5 de diciembre de 2024.
Dictamen C.I. 25/2024

DICTAMEN
QUE PRESENTA LA COMISIÓN DE INVESTIGACIÓN DE LA DIVISIÓN DE CIENCIAS DE LA COMUNICACIÓN Y DISEÑO

ANTECEDENTES

I. El Consejo Divisional de Ciencias de la Comunicación y Diseño, en la sesión 12.24, celebrada el 30 de abril de 2024, integró esta Comisión en los términos señalados en el artículo 56 de Reglamento Interno de los Órganos Colegiados Académicos.

II. El Consejo Divisional designó para esta Comisión a las siguientes personas integrantes:

a) Órganos personales:
  - ✓ Dra. Margarita Espinosa Meneses, Jefa del Departamento de Ciencias de la Comunicación;
  - ✓ Mtra. Brenda García Parra, Jefa del Departamento de Teoría y Procesos del Diseño;
  - ✓ Dr. Carlos Roberto Jaimez González, Jefe del Departamento de Tecnologías de la Información.

b) Representantes propietarios:
- Personal académico:
  - ✓ Mtro. Daniel Cuitláhuac Peña Rodríguez, Departamento de Ciencias de la Comunicación;
  - ✓ Mtro. Luis Antonio Rivera Díaz, Departamento de Teoría y Procesos del Diseño;
  - ✓ Dr. Dominique Emile Henri Decouchant, Departamento de Tecnologías de la Información.

CONSIDERACIONES

I. La Comisión recibió, para análisis y discusión, el tercer reporte parcial de resultados del proyecto de investigación denominado **"El diseño ante el cambio climático: divulgación, normatividad e información climatológico"** presentado por el Dr. Christopher Heard Wade, aprobado en la Sesión 14.21 celebrada el 11 de junio de 2021, mediante el Acuerdo DCCD.CD.16.14.21.

II. El Consejo Divisional en la Sesión 19.21 celebrada el 16 de diciembre de 2021, mediante Acuerdo DCCD.CD.11.19.21, aprobó una recalendarización de dicho proyecto por un periodo del 14 de junio de 2021 al 13 de junio de 2024.

III. El Consejo Divisional en la Sesión 22.22 celebrada el 19 de octubre de 2022, mediante Acuerdo DCCD.CD.04.22.22, aprobó el primer reporte parcial de resultados del proyecto de investigación.

IV. El Consejo Divisional en la Sesión 16.23 celebrada el 12 de septiembre de 2023, mediante Acuerdo DCCD.CD.06.16.23, aprobó el segundo reporte parcial de resultados del proyecto de investigación.

V. La Comisión de Investigación sesionó el 5 de diciembre de 2024, fecha en la que concluyó su trabajo de análisis y evaluación del reporte parcial de resultados, con el presente Dictamen.

VI. La Comisión tomó en consideración los siguientes elementos:

- *"Lineamientos para la creación de grupos de investigación y para el registro, seguimiento y evaluación de proyectos de investigación. División de Ciencias de la Comunicación y Diseño"* aprobados en la Sesión 20.24 del Consejo Divisional, celebrada el 18 de noviembre de 2024, mediante Acuerdo DCCD.CD.23.20.24.
- Protocolo de investigación.
- Relevancia para el Departamento.
- Objetivos planteados.
- Resultados obtenidos.

VII. **Objetivo general**:

Desarrollar información, modelos y formas de presentación significativa y accesible para facilitar el diseño de artefactos y sistemas que toman en cuenta las necesidades ocasionadas por el cambio climático. El objetivo concuerda con "Desarrollar investigación enfocada en la sustentabilidad" como prioridad definida por la planeación institucional.

VIII. **Objetivos particulares:**

- Actualizar y desarrollar bases de datos meteorológicos aptos para uso en el diseño de edificios y vivienda representativos con efectos de cambio climático para los principales centros urbanos de la República Mexicana.

- Desarrollar estudios puntuales del impacto posible del cambio climático sobre lugares y regiones de la República Mexicana con respecto a temperaturas, precipitación y eventos meteorológicos extremos con énfasis en la resiliencia de los diseños significativos.
- Diseñar un sistema de divulgación de la ciencia usando modelos de arquitectura de información para el cambio climático, de semiótica para propiciar diseño significativo, duradero y ambientalmente responsable.
- Desarrollo de tecnología para el uso de agua no-tratada, gris o del mar como fuente o sumidero del calor sin el uso de aditivos químicos perjudicial al medio ambiente.
- Desarrollo y diseño de objetos y artefactos enfocados a la sustentabilidad empleando modelos semióticos para su representación abstracta.

IX. **Actividades realizadas del 14 de junio de 2023 al 14 de junio de 2024**:

- Desarrollo de archivos de datos meteorológicos de años típicos futuros basado en los resultados de simulaciones de cambio climático regional CMIP6 y los datos horarios de la base de datos de sesenta sitios de la República Mexicana con control de calidad mejorada.
- Se desarrollará un sistema para revisión de los datos horarios de cada sitio para reducir el impacto de datos con errores de registro original y mejorar la fidelidad de los años típicos basados en los datos históricos.
- Se generará un base de datos de años típicos históricos mejorado para uso como referencia o punto de partida para generación de años típicos estimados para años futuros. Estos archivos serán libres de las restricciones del licenciamiento de los archivos con los cuales se cuentan actualmente tanto los del ASHRAE.

X. **Productos**:

- ✓ Se desarrolló código para extraer los datos y normalizarlos sobre la hora regular en la forma necesaria para archivos EPW. Se integró el código para simular radiación solar horario directa y difusa siguiendo la metodología publicada en “Development of 3012 IWEC2 Weather Files for International Locations (RP-1477)”.
- ✓ Se desarrolló código en Matlab para ensamblar años típicos a partir de una serie de archivos de datos meteorológicos horarios anuales, en formato EPW para un sitio dado conforme con el método de TMY3.
- ✓ Se terminó de codificar y probar simular radiación solar horario directa y difusa siguiendo la metodología publicada en “Development of 3012 IWEC2 Weather Files for International Locations (RP-1477)”.

- ✓ Se realizaron simulaciones del comportamiento térmico de una casa típica con la última versión del ESP-r, programa de simulación térmica de edificaciones con datos meteorológicos ajustados al futuro de acuerdo con los modelos regionales climatológicos para analizar tendencias en la necesidad de uso nocturno de aire acondicionado en el futuro en la Cd. de Villahermosa. Los resultados se incorporaron en el producto siguiente.
- ✓ Capítulo del libro "Investigación transdisciplinaria sobre cambio global en el sureste de México": "Confort Térmico Nocturno en Viviendas de la Ciudad de Villahermosa – Tabasco ante Escenarios de Cambio Climático.

DICTAMEN

ÚNICO:

Tras evaluar el tercer reporte parcial de resultados del proyecto de investigación denominado **"El diseño ante el cambio climático: divulgación, normatividad e información climatológico"** presentado por el Dr. Christopher Heard Wade, la Comisión de Investigación recomienda al Consejo Divisional de Ciencias de la Comunicación y Diseño aceptarlo.

VOTOS:

Integrantes	Sentido de los votos
Dra. Margarita Espinosa Meneses	A favor
Mtra. Brenda García Parra	A favor
Dr. Carlos Roberto Jaimez González	------
Mtro. Daniel Cuitláhuac Peña Rodríguez	A favor
Mtro. Luis Antonio Rivera Díaz	A favor
Dr. Dominique Emile Henri Decouchant	A favor
Total de los votos	**5 votos a favor**

UNIVERSIDAD AUTÓNOMA METROPOLITANA
UNIDAD CUAJIMALPA
SECRETARÍA ACADÉMICA
DIVISIÓN DE CIENCIAS DE LA COMUNICACIÓN Y DISEÑO

Coordinadora

Mtra. Silvia Gabriela García Martínez

Secretaria del Consejo Divisional de Ciencias de la Comunicación y Diseño

Ciudad de México a 3 de diciembre de 2024
DTPD.148.24

Dra. Gloria Angélica Martínez de la Peña
Presidenta del Consejo Divisional
División de Ciencias de la Comunicación y Diseño
Presente

Asunto: 2do. Informe de Actividades del Proyecto:
"El diseño ante el cambio climático: Divulgación, normatividad e información climatológica"

Estimada Dra. Martínez,

Por medio de la presente, me permito hacer entrega del informe del proyecto de Investigación: "El diseño ante el cambio climático: Divulgación, normatividad e información climatológica", a nombre del Dr. Christopher Lionel Heard Wade.

El proyecto fue aprobado por el Consejo Divisional de la DCCD en la Sesión 14.21 mediante el acuerdo DCCD.CD.16.14.21 el 11 de junio del 2021, y a partir de su prórroga, abarca un periodo del 14 de junio del 2024 al 13 de junio del 2025.

El presente tercer reporte contempla las actividades realizadas a partir del 13 de junio del 2023 al 12 de junio de 2024, por lo que se envía para su revisión y se anexan registro, así como proyecto, protocolo con cronograma de actividades considerando el presupuesto, y los productos de trabajo realizados.

Sin otro particular, reciba un cordial saludo

Atentamente

Casa abierta al tiempo

Mtra. Brenda García Parra
Jefa del Departamento de Teoría y Procesos del Diseño

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA
Unidad Cuajimalpa

MDI. Brenda García Parra 5 de noviembre de 2024
Jefa del Departamento de Teoría y Procesos del Diseño.
División de Ciencias de la Comunicación y Diseño.
Universidad Autónoma Metropolitana. Unidad Cuajimalpa

Por medio del presente solicito que se canaliza el informe de avance anual del proyecto "El diseño ante el cambio climático: Divulgación, normatividad e información climatológico" que cubre actividades del periodo 14 de junio de 2023 al 14 de junio de 2024 a la Comisión de Investigación del Consejo Divisional para su análisis, dictaminación y registro.

Atentamente

Dr. Christopher Heard

Informe de avance anual del proyecto "El diseño ante el cambio climático: Divulgación, normatividad e información climatológico".

Agosto 2024

Autores:

Dr. Christopher Lionel Heard Wade
Dr. Sazcha Marcelo Olivera Villarroel
Dra. Esperanza García López
Dra. Lucero Fabiola García Franco

Introducción

El proyecto fue aprobado en la sesión 14.21 celebrada el 11 de junio de 2021 mediante el acuerdo DCCD.CD.15.14.21 bajo el nombre de "El diseño ante el cambio climático: Divulgación, normatividad e información climatológico", iniciando el 14 de junio de 2021. Se notó un error en la propuesta original donde en el cronograma no se indicó que los periodos de tiempo eran trimestres y por lo tanto el proyecto fue aprobado con duración de un año. Actualmente se ha aprobado la recalendarización para tener una duración de tres años.

Actividades del periodo 14 de junio de 2023 al 14 de junio de 2024.

Las actividades principales durante periodo fueron:

6.1 Desarrollo de archivos de datos meteorológicos de años típicos futuros basado en los resultados de simulaciones de cambio climático regional CMIP6 y los datos horarios de la base de datos de sesenta sitios de la República Mexicana con control de calidad mejorada.

6.1.1 Se desarrollará un sistema para revisión de los datos horarios de cada sitio para reducir el impacto de datos con errores de registro original y mejorar la fidelidad de los años tipicos basados en los datos históricos.

6.1.2 Se generará un base de datos de años típicos históricos mejorado para uso como referencia o punto de partida para generación de años típicos estimados para años futuros. Estos archivos serán libres de las restricciones del licenciamiento de los archivos con los cuales se cuentan actualmente tanto los del ASHRAE como los del proyecto CONACyT/SENER Fondo de Sustentabilidad.

Productos del período

1. Se desarrollo código para extraer los datos y normalizarlos sobre la hora regular en la forma necesaria para archivos EPW. Si integró el código para simular radiación solar horario directa y difusa siguiendo la metodología publicada en "Development of 3012 IWEC2 Weather Files for International Locations (RP-1477)".

   En código se encuentra en el Apéndice 1.

2. Se desarrollo código en Matlab para ensamblar años típicos a partir de una serie de archivos de datos meteorológicos horarios anuales, en formato EPW para un sitio dado conforme con el método de TMY3.

   En código se encuentra en el Apéndice 2.

3. Se terminó de codificar y probar simular radiación solar horario directa y difusa siguiendo la metodología publicada en "Development of 3012 IWEC2 Weather Files for International Locations (RP-1477)".

   En código se encuentra en el Apéndice 3.

4. Se realizaron simulaciones del comportamiento térmico de una casa típica con la última versión del ESP-r, programa de simulación térmica de edificaciones con datos meteorológicos ajustados al futuro de acuerdo con los modelos regionales climatológicos para analizar tendencias en la necesidad de uso nocturno de aire acondicionado en el futuro en la Cd. de Villahermosa. Los resultados se incorporaron en el producto siguiente.

5. El escribió un capítulo invitado del libro "Investigación transdisciplinaria sobre cambio global en el sureste de México": "Confort Térmico Nocturno en Viviendas de la Ciudad de Villahermosa – Tabasco ante Escenarios de Cambio Climático", Christopher Lionel Heard*, Sazcha Marcelo Olivera Villarroel.

   Se encuentra en el Apéndice 4

Desarrollo de una base de datos de años típicos históricos mejorado

Con el fin de actualizar los datos meteorológicos disponibles para la construcción de años típicos mejorados se inició el desarrollo de código de Matlab para procesar los datos en bruto obtenidos del Integrated Surface Dataset (Global), National Centers for Environmental Information de la NOAA de Estados Unidos[1].

El procesamiento de los datos requiere el uso de sistemas de interpolación para normalizar el horario de los datos, reposición de datos faltantes y estimación de radiación solar a partir de datos alternos (Cobertura de nubes, temperatura del aire y la humedad relativa).

Se desarrollo código para extraer los datos y normalizarlos sobre la hora exacta en la forma necesaria para archivos EPW. Se terminó de codificar y probar simular radiación solar horario directa y difusa siguiendo la metodología publicada en “Development of 3012 IWEC2Weather Files for International Locations (RP-1477)”[2]

Se actualizaron y se mejoraron los métodos y código para los ajustes de datos meteorológicos horarios locales hacia el futuro basado en los resultados de la ronda CMIP6 del programa del EPCC y los métodos mejorados reportados en Eames et al[3] y en consulta en persona con el Dr. Matt Eames en la Universidad de Exeter, Reino Unido durante una estancia en el Reino Unido.

Al código de ajuste se agregó la metodología de uso de ensambles de resultados de una selección (siete) de modelos regionales de cambio climático del CMIP6 para mejorar la confiabilidad de los resultados de los ajustes[4],[5].

El código de Matlab se encuentra en el Apéndice 5.

Actualización de simulaciones de una casa típica de la Ciudad de México

Se realizaron simulaciones del comportamiento térmico de una casa típica con la última versión del ESP-r, programa de simulación térmica de edificaciones con datos ajustados al futuro.

Referencias

1. https://www.ncei.noaa.gov/access/search/data-search/global-hourly?dataset abbv=DS3505&countryabbv=&georegionabbv=&resolution=40 Consultado 25 Noviembre 2021.

2. Joe, Yu, et al. "Development of 3012 IWEC2 weather files for international locations (RP-1477)." Ashrae Transactions 120.1 (2014).

3. Eames, M. E., Xie, H., Mylona, A., Shilston, R., & Hacker, J. (2024). A revised morphing algorithm for creating future weather for building performance

evaluation. Building Services Engineering Research & Technology, 45(1), 5-20.

4. Almazroui, M., Islam, M. N., Saeed, F., Saeed, S., Ismail, M., Ehsan, M. A., ... & Barlow, M. (2021). Projected changes in temperature and precipitation over the United States, Central America, and the Caribbean in CMIP6 GCMs. Earth Systems and Environment, 5, 1-24.

5. Milinski, S., Maher, N., & Olonscheck, D. (2020). How large does a large ensemble need to be?. Earth System Dynamics, 11(4), 885-901.

```
%## Copyright (C) 2021 Christopher Heard
%##
%## This program is free software: you can redistribute it and/or modify it
%## under the terms of the GNU General Public License as published by
%## the Free Software Foundation, either version 3 of the License, or
%## (at your option) any later version.
%##
%## This program is distributed in the hope that it will be useful, but
%## WITHOUT ANY WARRANTY; without even the implied warranty of
%## MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
%## GNU General Public License for more details.
%##
%## You should have received a copy of the GNU General Public License
%## along with this program.  If not, see
%## <https://www.gnu.org/licenses/>.


%## Author: Christopher Heard
%## Created: 2024-09-18

% This script is to read in the data from a global hourly weather file downloaded from
% the National Climate Data Center USA.

% Incorporates code from "ToZHM_05bis" for solar estimation

% Set up classes for Koppen-Gieger classification of sites for coefficients
% for solar radiation estimation from JOE, Yu, et al. Development of 3012
% IWEC2 Weather Files for International Locations (RP-1477).
% Ashrae Transactions, 2014, vol. 120, no 1.
% Source KoppenGeiger.m : Beck, H., Zimmermann, N., McVicar, T. et al.
% Present and future Kˆppen-Geiger climate classification maps at 1-km resolution.
% Sci Data 5, 180214 (2018). https://doi.org/10.1038/sdata.2018.214
clear

classes = {...
        1,1,'Af','Tropical, rainforest',[0 0 255];...
        2,1,'Am','Tropical, monsoon',[0 120 255];...
        3,1,'Aw','Tropical, savannah',[70 170 250];... % also As - same coefficients for solar estimation
        4,2,'BWh','Arid, desert, hot',[255 0 0];...
        5,2,'BWk','Arid, desert, cold',[255 150 150];...
        6,2,'BSh','Arid, steppe, hot',[245 165 0];...
        7,2,'BSk','Arid, steppe, cold',[255 220 100];...
        8,3,'Csa','Temperate, dry summer, hot summer',[255 255 0];...
        9,3,'Csb','Temperate, dry summer, warm summer',[200 200 0];...
        10,3,'Csc','Temperate, dry summer, cold summer',[150 150 0];...
        11,3,'Cwa','Temperate, dry winter, hot summer',[150 255 150];...
        12,3,'Cwb','Temperate, dry winter, warm summer',[100 200 100];...
        13,3,'Cwc','Temperate, dry winter, cold summer',[50 150 50];...
        14,3,'Cfa','Temperate, no dry season, hot summer',[200 255 80];...
        15,3,'Cfb','Temperate, no dry season, warm summer',[100 255 80];...
        16,3,'Cfc','Temperate, no dry season, cold summer',[50 200 0];...
        17,4,'Dsa','Cold, dry summer, hot summer',[255 0 255];...
        18,4,'Dsb','Cold, dry summer, warm summer',[200 0 200];...
        19,4,'Dsc','Cold, dry summer, cold summer',[150 50 150];...
        20,4,'Dsd','Cold, dry summer, very cold winter',[150 100 150];...
        21,4,'Dwa','Cold, dry winter, hot summer',[170 175 255];...
        22,4,'Dwb','Cold, dry winter, warm summer',[90 120 220];...
        23,4,'Dwc','Cold, dry winter, cold summer',[75 80 180];...
        24,4,'Dwd','Cold, dry winter, very cold winter',[50 0 135];...
        25,4,'Dfa','Cold, no dry season, hot summer',[0 255 255];...
        26,4,'Dfb','Cold, no dry season, warm summer',[55 200 255];...
        27,4,'Dfc','Cold, no dry season, cold summer',[0 125 125];...
        28,4,'Dfd','Cold, no dry season, very cold winter',[0 70 95];...
        29,5,'ET','Polar, tundra',[178 178 178];...
        30,5,'EF','Polar, frost',[102 102 102];...
        };

KoppenClass = {...
'MEX_ACAPULCO_768050','Aw',-6;...
'MEX_ACAPULCO-G-ALVAREZ_768056','Aw',-6;...
'MEX_AGUASCALIENTES_765710','BSh',-6;...
'MEX_ALTAR-SON_761130','BSh',-7;...
'MEX_ARRIAGA_768400','Aw',-6;...
'MEX_BAHIAS-DE-HUATULCO_768485','Aw',-6;...
'MEX_CABO-SAN-LUCAS_767503','BWh',-7;...
'MEX_CAMPECHE-IGNACIO_766950','Aw',-6;...
'MEX_CAMPECHE-IGNACIO_766961','Aw',-6;...
'MEX_CANCUN_765950','Aw',-5;...
'MEX_CANCUN-IAP_765905','Aw',-5;...
%'MEX_CANCUN-IAP_765906','Aw',-5;...
'MEX_CHETUMAL_767500','Aw',-5;...
'MEX_CHICHEN-ITZA_767501','Aw',-6;...
'MEX_CHIHUAHUA-G-FIERRO_762253','BSh',-6;...
'MEX_CHIHUAHUA-IAP_762252','BSh',-6;...
'MEX_CHIHUAHUA-UNIVERSIT_762250','BSh',-6;...
'MEX_CHILPANCINGO_767620','Aw',-6;...
```

```
'MEX_CHOIX_763110','Csa',-7;...
'MEX_CIUDAD-CONSTITUCION_764020','BWh',-7;...
'MEX_CIUDAD-DEL-CARMEN-I_767493','Aw',-6;...
'MEX_CIUDAD-GUZMAN_766560','Csb',-6;...
'MEX_CIUDAD-JUAREZ_760751','BWh',-7;...
'MEX_CIUDAD-OBREGON-SON_762580','BWh',-7;...
'MEX_CIUDAD-VICTORIA_764910','Cfa',-6;...
%'MEX_CIUDAD-VICTORIA_764915','Cfa',-6;...
'MEX_COATZACOALCOS_767410','Am',-6;...
'MEX_COLIMA_766580','Aw',-6;...
'MEX_COLONIA-JUAN-CARRAS_764580','Aw',-7;...
'MEX_COLOTLAN_765190','Csa',-6;...
'MEX_COMITAN_768480','Cwb',-6;...
'MEX_COZUMEL-AP_766480','Aw',-5;...
'MEX_COZUMEL-IAP_766493','Aw',-5;...
'MEX_C-P-A-CARLOS-ROVIRO_767433','Af',-6;...
'MEX_CUERNAVACA_767260','Aw',-6;...
'MEX_CULIACAN(CITY)_764120','BSh',-7;...
'MEX_DE-GUANAJUATO-IAP_765773','BSh',-6;...
'MEX_DON-MIGUEL-Y-HIDALG_766133','Csa',-6;...
'MEX_DURANGO(CITY)_764230','BSk',-6;...
'MEX_DURANGO-IAP_764235','BSk',-6;...
'MEX_EJIDO-NUEVO-LEON-BC_760400','BWh',-8;...
'MEX_EMPALME_762560','BWh',-7;...
'MEX_FELIPE-CARRILLO-PUE_766980','Aw',-5;...
'MEX_FES-CUAUTITLAN_766700','Cwb',-6;...
'MEX_GEOGRAFIA-UNAM_766810','Cwb',-6;...
'MEX_GUADALAJARA_766120','Csa',-6;...
'MEX_GUADALAJARA-IAP_766131','Csa',-6;...
'MEX_GUANAJUATO_765770','Csa',-6;...
'MEX_GUAYMAS-G-YANEZ-AP_762555','BWh',-7;...
'MEX_HACIENDA-YLANG-YLAN_766920','Aw',-6;...
'MEX_HERMOSILLO-IAP_761600','BWh',-7;...
'MEX_HUAJUAPAN-DE-LEON_767730','Cwb',-6;...
'MEX_ISLA-GUADALUPE_761510','BSh',-8;...
'MEX_ISLA-SOCORRO_767230','BWh',-7;...
'MEX_IXTAPA-ZIHUATANEJO_767584','Aw',-6;...
'MEX_IXTEPEC_767502','Aw',-6;...
'MEX_JALAPA_766870','Cfa',-6;...
'MEX_JUAREZ-G-GONZALEZ_760753','BWh',-7;...
'MEX_LAGOS-DE-MORENO_765730','Csa',-6;...
'MEX_LA-PAZ(CITY)_764050','BWh',-7;...
'MEX_LA-PAZ-G-MARQUEZ-AP_764055','BWh',-7;...
'MEX_LEON-SAN-CARLOS_765752','BSh',-6;...
'MEX_LICENCIADO-GUSTAVO_766013','Aw',-6;...
'MEX_LICENCIADO-Y-GEN-IG_766546','Cwb',-6;...
'MEX_LORETO_763050','BWh',-7;...
'MEX_LOS-CABOS-IAP_764056','BWh',-7;...
'MEX_MANZANILLO_766540','Aw',-6;...
'MEX_MATAMOROS-G-CANALES_763993','Cfa',-6;...
'MEX_MATAMOROS-IAP_763991','Cfa',-6;...
'MEX_MATLAPA_765850','Aw',-6;...
'MEX_MAZATLAN-G-BUELNA-I_764591','Aw',-7;...
'MEX_MAZATLAN-G-BUELNA-II_764593','Aw',-7;...
'MEX_MERIDA-IAP_766440','Aw',-6;...
'MEX_MEXICALI_760051','BWh',-8;...
'MEX_MEXICALI-G-SANCHEZ_760053','BWh',-8;...
'MEX_MEXICO-CITY_766800','Cwb',-6;...
'MEX_MEXICO-CITY-B-JUAREZ_766793','Cwb',-6;...
'MEX_MEXICO-CITY-IAP_766790','Cwb',-6;...
'MEX_MINATITLAN_767383','Am',-6;...
'MEX_MONCLOVA_763420','BSh',-6;...
'MEX_MONTERREY(CITY)_763930','BSh',-6;...
'MEX_MONTERREY-G-ESCOBEDO_763943','BSh',-6;...
'MEX_MONTERREY-IAP_763940','BSh',-6;...
'MEX_MORELIA_766650','Cwb',-6;...
'MEX_MORELIA-G-MUJICA-AP_766655','Cwb',-6;...
'MEX_NUEVA-CASAS-GRANDES_761220','BSk',-6;...
'MEX_NUEVO-LAREDO_762861','BSh',-6;...
'MEX_OAXACA_767750','Cwa',-6;...
'MEX_ORIZABA-VER_767370','Cwb',-6;...
'MEX_PACHUCA_766320','BSk',-6;...
'MEX_PARRAL_763230','BSk',-6;...
'MEX_PIEDRAS-NEGRAS_762430','Cfa',-6;...
'MEX_PILARES-DE-NACOZARI_761180','BSh',-7;...
'MEX_PLAYA-DE-ORO-IAP_766534','Aw',-6;...
'MEX_PROGRESO_765930','BSh',-6;...
'MEX_PUEBLA_766850','Cwb',-6;...
'MEX_PUERTO-ANGEL_768550','Am',-6;...
'MEX_PUERTO-ESCONDIDO_768556','Aw',-6;...
'MEX_PUERTO-PENASCO_760610','BWh',-7;...
'MEX_QUERETARO_766250','BSk',-6;...
'MEX_QUETZALCOATL-IAP_762863','BSh',-6;...
'MEX_REYNOSA-G-BLANCO-AP_763503','BSh',-6;...
'MEX_REYNOSA-IAP_763501_75','BSh',-6;...
'MEX_RIO-VERDE_765810','Cfa',-6;...
```

```
'MEX_SALINA-CRUZ_768330','Aw',-6;...
'MEX_SALTILLO_763900','BSh',-6;...
'MEX_SAN-FELIPE_760550','BWh',-8;...
'MEX_SAN-LUIS-POTOSI_765390','BSk',-6;...
'MEX_SANTA-ROSALIA_762530','BSh',-7;...
'MEX_SN-CRISTOBAL-LAS-C_768450','Cwb',-6;...
'MEX_SOMBRERETE_764710','BSk',-6;...
'MEX_SOTO-LA-MARINA_764990','BSh',-6;...
'MEX_TAJIN_766127','Af',-6;...
'MEX_TAMPICO_765480','BSh',-6;...
'MEX_TAMPICO-GEN-FJ-MINA_765491','BSh',-6;...
'MEX_TAMPICO-G-MINA-AP_765493','BSh',-6;...
'MEX_TAMUIN-SLP_765430_02','Aw',-6;...
'MEX_TAPACHULA-CITY_769030','Aw',-6;...
'MEX_TAPACHULA-IAP_769043','Aw',-6;...
'MEX_TEMOSACHIC_762200','BSk',-6;...
'MEX_TEPEHUANES_763730','Csb',-6;...
'MEX_TEPIC_765560','Cwa',-6;...
'MEX_TIJUANA-G-RODRIGUE_760013','BSh',-8;...
'MEX_TLAXCALA_766830','Cwb',-6;...
'MEX_TOLUCA_766750','Cwb',-6;...
'MEX_TOLUCA-LA-LOPEZ-AP_766753','Cwb',-6;...
'MEX_TORREON-AP_763820','BWh',-6;...
'MEX_TULANCINGO_766340','Cfb',-6;...
'MEX_TUXPAN-VER_766400','Am',-6;...
'MEX_TUXTLA-GUTIERREZ_760754','Aw',-6;...
'MEX_TUXTLA-GUTIERREZ-A_768430','Aw',-6;...
'MEX_VALLADOLID_766470','Aw',-6;...
'MEX_VALLE-DEL-FUERTE-IN_763615','BWh',-7;...
'MEX_VERACRUZ-GEN-JARA_766910','Aw',-6;...
'MEX_VERACRUZ-G-JARA-AP_766913','Aw',-6;...
'MEX_VILLAHERMOSA_767441_75','Am',-6;...
'MEX_VILLAHERMOSA-TAB_767430','Am',-6;...
'MEX_XOXOCOTLAN-IAP_767755','Cwb',-6;...
'MEX_ZACATECAS(CITY)_765250','Cwb',-6;...
'MEX_ZACATECAS-G-RUIZ-AP_765255','Cwb',-6;...
'MEX_ZAMORA_766620','Csa',-6};

% Tau d and b for ASHRAE clear sky models from chapter 14 2021 fundamentals
% handbook database

taud =
[2.55600000000000,2.53900000000000,2.49100000000000,2.46200000000000,2.42800000000000,2.42300000000000,2.42800000000000,
2.40300000000000,2.48300000000000,2.43900000000000,2.55400000000000,2.55000000000000;2.51900000000000,2.53200000000000,2
.45600000000000,2.42700000000000,2.39700000000000,2.39600000000000,2.38400000000000,2.38800000000000,2.44400000000000,2.
49400000000000,2.52900000000000,2.53800000000000;2.53700000000000,2.53000000000000,2.43200000000000,2.41600000000000,2.3
8500000000000,2.31500000000000,2.31100000000000,2.34800000000000,2.38700000000000,2.47400000000000,2.53200000000000,2.54
500000000000;2.54200000000000,2.47800000000000,2.45500000000000,2.33300000000000,2.27300000000000,2.33600000000000,2.384
00000000000,2.43400000000000,2.38500000000000,2.51600000000000,2.49600000000000,2.52700000000000;2.65300000000000,2.5790
0000000000,2.53700000000000,2.41100000000000,2.29500000000000,2.39500000000000,2.38400000000000,2.44300000000000,2.42500
000000000,2.59900000000000,2.63500000000000,2.64800000000000;2.59400000000000,2.54900000000000,2.50000000000000,2.403000
00000000,2.39200000000000,2.48000000000000,2.49800000000000,2.55900000000000,2.51600000000000,2.58100000000000,2.5710000
0000000,2.59600000000000;2.50300000000000,2.42800000000000,2.33300000000000,2.29200000000000,2.24300000000000,2.39300000
000000,2.43900000000000,2.46600000000000,2.39500000000000,2.46300000000000,2.44100000000000,2.48900000000000;2.514000000
00000,2.44200000000000,2.34800000000000,2.30400000000000,2.25100000000000,2.40400000000000,2.44400000000000,2.4670000000
0000,2.39900000000000,2.47600000000000,2.43900000000000,2.49000000000000;2.64700000000000,2.60700000000000,2.56800000000
000,2.46600000000000,2.40300000000000,2.36000000000000,2.40000000000000,2.46000000000000,2.41700000000000,2.616000000000
00,2.68100000000000,2.65000000000000;2.63200000000000,2.53500000000000,2.57400000000000,2.51200000000000,2.3860000000000
0,2.41200000000000,2.41700000000000,2.46000000000000,2.44500000000000,2.58300000000000,2.64200000000000,2.63900000000000
;2.59600000000000,2.55000000000000,2.59600000000000,2.45300000000000,2.37100000000000,2.35200000000000,2.37600000000000,
2.43900000000000,2.39700000000000,2.57800000000000,2.62500000000000,2.63100000000000;2.60300000000000,2.53400000000000,2
.58800000000000,2.44200000000000,2.37000000000000,2.37000000000000,2.38000000000000,2.45700000000000,2.40000000000000,2.
57700000000000,2.65200000000000,2.64000000000000;2.55000000000000,2.50800000000000,2.50300000000000,2.42200000000000,2.3
2500000000000,2.38100000000000,2.41100000000000,2.44500000000000,2.43500000000000,2.52400000000000,2.59400000000000,2.56
800000000000;2.49800000000000,2.44500000000000,2.42300000000000,2.36100000000000,2.26100000000000,2.34300000000000,2.396
00000000000,2.43600000000000,2.42100000000000,2.47300000000000,2.51000000000000,2.51600000000000;2.39100000000000,2.3490
0000000000,2.28400000000000,2.14400000000000,2.06500000000000,2.29100000000000,2.31700000000000,2.35000000000000,2.33800
000000000,2.40600000000000,2.38900000000000,2.40300000000000;2.58200000000000,2.53200000000000,2.55100000000000,2.470000
00000000,2.31300000000000,2.29500000000000,2.37700000000000,2.37700000000000,2.39000000000000,2.52500000000000,2.6140000
0000000,2.58700000000000;2.49600000000000,2.43900000000000,2.45400000000000,2.34100000000000,2.25900000000000,2.30300000
000000,2.36500000000000,2.39200000000000,2.39800000000000,2.46800000000000,2.52100000000000,2.51700000000000;2.463000000
00000,2.42700000000000,2.42800000000000,2.33100000000000,2.21800000000000,2.32300000000000,2.36300000000000,2.4020000000
0000,2.40400000000000,2.44500000000000,2.48900000000000,2.48900000000000;2.45500000000000,2.41000000000000,2.42100000000
000,2.31000000000000,2.19400000000000,2.29700000000000,2.33700000000000,2.37300000000000,2.38400000000000,2.436000000000
00,2.47800000000000,2.47200000000000;2.56500000000000,2.52000000000000,2.41700000000000,2.29900000000000,2.3100000000000
0,2.37300000000000,2.30800000000000,2.39100000000000,2.46200000000000,2.50600000000000,2.52500000000000,2.53700000000000
;2.55200000000000,2.51300000000000,2.46200000000000,2.33900000000000,2.33400000000000,2.30800000000000,2.24700000000000,
2.38500000000000,2.44100000000000,2.44200000000000,2.50200000000000,2.51400000000000;2.56400000000000,2.52100000000000,2
.52300000000000,2.41800000000000,2.28900000000000,2.28800000000000,2.38500000000000,2.38700000000000,2.37400000000000,2.
51600000000000,2.58600000000000,2.57000000000000;2.51300000000000,2.47300000000000,2.49700000000000,2.36700000000000,2.2
3700000000000,2.26300000000000,2.30800000000000,2.30600000000000,2.36600000000000,2.46400000000000,2.54700000000000,2.53
400000000000;2.41300000000000,2.38500000000000,2.34100000000000,2.29800000000000,2.16100000000000,2.30500000000000,2.314
00000000000,2.37000000000000,2.36800000000000,2.38700000000000,2.43900000000000,2.44200000000000;2.28500000000000,2.2580
0000000000,2.21900000000000,2.10600000000000,2.18200000000000,2.18600000000000,2.26900000000000,2.29500000000000,2.42400
000000000,2.28500000000000,2.39900000000000,2.32900000000000;2.32500000000000,2.29100000000000,2.18700000000000,2.112000
00000000,2.05700000000000,2.23900000000000,2.28400000000000,2.32200000000000,2.31200000000000,2.33200000000000,2.3280000
```

```
0000000,2.31900000000000;2.55400000000000,2.50900000000000,2.40200000000000,2.29200000000000,2.29200000000000,2.34800000
000000,2.28300000000000,2.36600000000000,2.43900000000000,2.48600000000000,2.50500000000000,2.52100000000000;2.450000000
00000,2.42400000000000,2.42300000000000,2.27700000000000,2.05600000000000,2.15500000000000,2.26600000000000,2.2920000000
0000,2.30000000000000,2.42400000000000,2.50500000000000,2.47200000000000;2.42700000000000,2.39300000000000,2.40600000000
000,2.30200000000000,2.09600000000000,2.24700000000000,2.30000000000000,2.30600000000000,2.30300000000000,2.389000000000
00,2.46300000000000,2.46200000000000;2.43400000000000,2.40000000000000,2.40600000000000,2.31200000000000,2.1230000000000
0,2.27000000000000,2.31700000000000,2.33100000000000,2.32800000000000,2.39500000000000,2.46400000000000,2.46800000000000
;2.20200000000000,2.14400000000000,2.14500000000000,1.98200000000000,2.14300000000000,1.97500000000000,2.04500000000000,
2.10500000000000,2.44900000000000,2.19000000000000,2.44500000000000,2.23100000000000;2.08800000000000,2.03800000000000,2
.02400000000000,1.87000000000000,2.11700000000000,1.89900000000000,1.98000000000000,2.03700000000000,2.44800000000000,2.
07000000000000,2.36600000000000,2.11900000000000;2.15500000000000,2.17500000000000,2.11500000000000,2.02900000000000,2.1
7800000000000,2.09600000000000,2.20500000000000,2.21700000000000,2.46500000000000,2.23400000000000,2.37600000000000,2.22
400000000000;2.23600000000000,2.27300000000000,2.18200000000000,2.11100000000000,2.17700000000000,2.18200000000000,2.285
00000000000,2.29200000000000,2.45400000000000,2.32100000000000,2.40500000000000,2.30700000000000;2.41600000000000,2.3900
0000000000,2.29400000000000,2.18500000000000,2.14500000000000,2.31700000000000,2.34400000000000,2.36700000000000,2.34600
000000000,2.42000000000000,2.38800000000000,2.39500000000000;2.43800000000000,2.40500000000000,2.28700000000000,2.126000
00000000,2.04700000000000,2.27300000000000,2.32600000000000,2.35800000000000,2.36400000000000,2.41700000000000,2.4360000
0000000,2.42000000000000;2.44300000000000,2.40900000000000,2.28900000000000,2.12600000000000,2.04800000000000,2.27700000
000000,2.32400000000000,2.36000000000000,2.36400000000000,2.42300000000000,2.44200000000000,2.42600000000000;2.570000000
00000,2.51000000000000,2.38000000000000,2.24200000000000,2.23700000000000,2.37300000000000,2.31500000000000,2.3980000000
0000,2.44800000000000,2.51500000000000,2.53400000000000,2.52200000000000;2.37000000000000,2.32600000000000,2.24100000000
000,2.15900000000000,2.04700000000000,2.26000000000000,2.33000000000000,2.36200000000000,2.33600000000000,2.366000000000
00,2.36300000000000,2.33300000000000;2.48700000000000,2.44200000000000,2.29300000000000,2.15200000000000,2.0440000000000
0,2.31200000000000,2.29600000000000,2.32300000000000,2.34300000000000,2.43300000000000,2.46200000000000,2.47400000000000
;2.55100000000000,2.50000000000000,2.39700000000000,2.25800000000000,2.22000000000000,2.30000000000000,2.18500000000000,
2.28300000000000,2.37600000000000,2.47200000000000,2.50300000000000,2.52200000000000;2.44400000000000,2.44900000000000,2
.25900000000000,2.27000000000000,2.10700000000000,2.32700000000000,2.40100000000000,2.38300000000000,2.37800000000000,2.
43400000000000,2.47800000000000,2.50900000000000;2.41200000000000,2.40600000000000,2.33400000000000,2.17000000000000,1.9
0900000000000,2.27300000000000,2.39800000000000,2.37900000000000,2.38300000000000,2.43500000000000,2.48400000000000,2.49
900000000000;2.47100000000000,2.39100000000000,2.21200000000000,1.92600000000000,1.80700000000000,2.20800000000000,2.245
00000000000,2.25700000000000,2.24400000000000,2.33300000000000,2.42400000000000,2.47000000000000;2.57200000000000,2.4940
0000000000,2.33500000000000,2.12000000000000,1.99600000000000,2.35700000000000,2.39500000000000,2.35100000000000,2.36700
000000000,2.49000000000000,2.53300000000000,2.57200000000000;2.44600000000000,2.37200000000000,2.15200000000000,1.884000
00000000,1.79900000000000,2.22200000000000,2.21600000000000,2.24900000000000,2.23300000000000,2.37900000000000,2.4270000
0000000,2.45800000000000;2.46900000000000,2.40200000000000,2.20100000000000,1.93100000000000,1.86500000000000,2.26200000
000000,2.24800000000000,2.27800000000000,2.26300000000000,2.41400000000000,2.45600000000000,2.48500000000000;2.550000000
00000,2.52800000000000,2.52300000000000,2.37400000000000,2.36800000000000,2.31600000000000,2.25300000000000,2.4160000000
0000,2.47300000000000,2.53100000000000,2.53500000000000,2.54500000000000;2.52200000000000,2.45500000000000,2.36800000000
000,2.17500000000000,2.16500000000000,2.37300000000000,2.32300000000000,2.40800000000000,2.44100000000000,2.513000000000
00,2.51200000000000,2.49600000000000];
taub =
[0.310000000000000,0.321000000000000,0.333000000000000,0.344000000000000,0.363000000000000,0.366000000000000,0.394000000
000000,0.396000000000000,0.359000000000000,0.364000000000000,0.320000000000000,0.322000000000000;0.288000000000000,0.290
000000000000,0.310000000000000,0.322000000000000,0.339000000000000,0.341000000000000,0.386000000000000,0.381000000000000
,0.347000000000000,0.317000000000000,0.293000000000000,0.289000000000000;0.291000000000000,0.294000000000000,0.321000000
000000,0.326000000000000,0.344000000000000,0.390000000000000,0.434000000000000,0.422000000000000,0.394000000000000,0.341
000000000000,0.301000000000000,0.291000000000000;0.298000000000000,0.314000000000000,0.326000000000000,0.371000000000000
,0.409000000000000,0.407000000000000,0.404000000000000,0.388000000000000,0.396000000000000,0.347000000000000,0.326000000
000000,0.305000000000000;0.266000000000000,0.271000000000000,0.281000000000000,0.320000000000000,0.373000000000000,0.369
000000000000,0.384000000000000,0.366000000000000,0.369000000000000,0.311000000000000,0.281000000000000,0.267000000000000
;0.271000000000000,0.281000000000000,0.298000000000000,0.331000000000000,0.349000000000000,0.342000000000000,0.347000000
000000,0.330000000000000,0.337000000000000,0.308000000000000,0.289000000000000,0.274000000000000;0.329000000000000,0.350
000000000000,0.387000000000000,0.410000000000000,0.442000000000000,0.409000000000000,0.402000000000000,0.393000000000000
,0.409000000000000,0.378000000000000,0.362000000000000,0.336000000000000;0.328000000000000,0.349000000000000,0.386000000
000000,0.410000000000000,0.443000000000000,0.409000000000000,0.403000000000000,0.395000000000000,0.409000000000000,0.376
000000000000,0.365000000000000,0.338000000000000;0.301000000000000,0.304000000000000,0.307000000000000,0.337000000000000
,0.374000000000000,0.419000000000000,0.425000000000000,0.406000000000000,0.409000000000000,0.350000000000000,0.312000000
000000,0.301000000000000;0.260000000000000,0.276000000000000,0.270000000000000,0.292000000000000,0.346000000000000,0.355
000000000000,0.374000000000000,0.361000000000000,0.360000000000000,0.310000000000000,0.273000000000000,0.263000000000000
;0.320000000000000,0.328000000000000,0.314000000000000,0.349000000000000,0.399000000000000,0.432000000000000,0.440000000
000000,0.420000000000000,0.421000000000000,0.369000000000000,0.333000000000000,0.319000000000000;0.318000000000000,0.331
000000000000,0.316000000000000,0.353000000000000,0.400000000000000,0.426000000000000,0.437000000000000,0.413000000000000
,0.419000000000000,0.368000000000000,0.328000000000000,0.317000000000000;0.265000000000000,0.276000000000000,0.278000000
000000,0.299000000000000,0.354000000000000,0.348000000000000,0.352000000000000,0.344000000000000,0.345000000000000,0.310
000000000000,0.274000000000000,0.272000000000000;0.294000000000000,0.307000000000000,0.318000000000000,0.341000000000000
,0.392000000000000,0.374000000000000,0.364000000000000,0.355000000000000,0.358000000000000,0.335000000000000,0.305000000
000000,0.298000000000000;0.400000000000000,0.414000000000000,0.438000000000000,0.493000000000000,0.527000000000000,0.462
000000000000,0.458000000000000,0.446000000000000,0.445000000000000,0.420000000000000,0.410000000000000,0.398000000000000
;0.305000000000000,0.315000000000000,0.306000000000000,0.326000000000000,0.399000000000000,0.427000000000000,0.411000000
000000,0.409000000000000,0.402000000000000,0.356000000000000,0.315000000000000,0.312000000000000;0.287000000000000,0.306
000000000000,0.300000000000000,0.327000000000000,0.387000000000000,0.382000000000000,0.374000000000000,0.366000000000000
,0.364000000000000,0.335000000000000,0.299000000000000,0.295000000000000;0.304000000000000,0.315000000000000,0.314000000
000000,0.347000000000000,0.405000000000000,0.379000000000000,0.373000000000000,0.360000000000000,0.360000000000000,0.342
000000000000,0.309000000000000,0.305000000000000;0.310000000000000,0.324000000000000,0.319000000000000,0.356000000000000
,0.417000000000000,0.392000000000000,0.387000000000000,0.374000000000000,0.372000000000000,0.350000000000000,0.316000000
000000,0.313000000000000;0.362000000000000,0.376000000000000,0.401000000000000,0.436000000000000,0.451000000000000,0.449
000000000000,0.465000000000000,0.442000000000000,0.411000000000000,0.397000000000000,0.376000000000000,0.373000000000000
;0.368000000000000,0.379000000000000,0.391000000000000,0.431000000000000,0.449000000000000,0.467000000000000,0.481000000
000000,0.445000000000000,0.426000000000000,0.422000000000000,0.389000000000000,0.384000000000000;0.341000000000000,0.346
000000000000,0.336000000000000,0.364000000000000,0.435000000000000,0.458000000000000,0.437000000000000,0.434000000000000
,0.432000000000000,0.388000000000000,0.353000000000000,0.348000000000000;0.306000000000000,0.318000000000000,0.305000000
000000,0.339000000000000,0.416000000000000,0.422000000000000,0.413000000000000,0.408000000000000,0.395000000000000,0.354
000000000000,0.313000000000000,0.312000000000000;0.329000000000000,0.338000000000000,0.353000000000000,0.375000000000000
,0.433000000000000,0.395000000000000,0.398000000000000,0.379000000000000,0.378000000000000,0.368000000000000,0.333000000
000000,0.326000000000000;0.363000000000000,0.374000000000000,0.391000000000000,0.439000000000000,0.422000000000000,0.425
000000000000,0.406000000000000,0.397000000000000,0.355000000000000,0.387000000000000,0.341000000000000,0.350000000000000
```

```
;0.420000000000000,0.433000000000000,0.470000000000000,0.506000000000000,0.530000000000000,0.475000000000000,0.465000000
000000,0.451000000000000,0.449000000000000,0.440000000000000,0.427000000000000,0.421000000000000;0.358000000000000,0.372
000000000000,0.400000000000000,0.435000000000000,0.451000000000000,0.448000000000000,0.465000000000000,0.441000000000000
,0.411000000000000,0.395000000000000,0.375000000000000,0.371000000000000;0.386000000000000,0.386000000000000,0.376000000
000000,0.425000000000000,0.530000000000000,0.511000000000000,0.477000000000000,0.467000000000000,0.464000000000000,0.423
000000000000,0.386000000000000,0.388000000000000;0.331000000000000,0.340000000000000,0.334000000000000,0.377000000000000
,0.460000000000000,0.419000000000000,0.404000000000000,0.401000000000000,0.404000000000000,0.373000000000000,0.331000000
000000,0.328000000000000;0.329000000000000,0.339000000000000,0.336000000000000,0.375000000000000,0.452000000000000,0.413
000000000000,0.400000000000000,0.394000000000000,0.397000000000000,0.372000000000000,0.331000000000000,0.326000000000000
;0.377000000000000,0.394000000000000,0.401000000000000,0.469000000000000,0.429000000000000,0.481000000000000,0.464000000
000000,0.446000000000000,0.349000000000000,0.408000000000000,0.327000000000000,0.367000000000000;0.415000000000000,0.433
000000000000,0.446000000000000,0.519000000000000,0.448000000000000,0.516000000000000,0.494000000000000,0.476000000000000
,0.359000000000000,0.449000000000000,0.357000000000000,0.403000000000000;0.390000000000000,0.390000000000000,0.416000000
000000,0.462000000000000,0.425000000000000,0.453000000000000,0.424000000000000,0.420000000000000,0.349000000000000,0.401
000000000000,0.345000000000000,0.370000000000000;0.371000000000000,0.367000000000000,0.399000000000000,0.438000000000000
,0.429000000000000,0.432000000000000,0.403000000000000,0.400000000000000,0.353000000000000,0.382000000000000,0.339000000
000000,0.353000000000000;0.359000000000000,0.369000000000000,0.401000000000000,0.444000000000000,0.464000000000000,0.416
000000000000,0.410000000000000,0.402000000000000,0.406000000000000,0.380000000000000,0.373000000000000,0.364000000000000
;0.393000000000000,0.405000000000000,0.444000000000000,0.506000000000000,0.541000000000000,0.470000000000000,0.458000000
000000,0.444000000000000,0.435000000000000,0.420000000000000,0.402000000000000,0.401000000000000;0.393000000000000,0.405
000000000000,0.444000000000000,0.506000000000000,0.541000000000000,0.471000000000000,0.460000000000000,0.446000000000000
,0.436000000000000,0.420000000000000,0.402000000000000,0.401000000000000;0.361000000000000,0.380000000000000,0.415000000
000000,0.459000000000000,0.475000000000000,0.446000000000000,0.464000000000000,0.437000000000000,0.413000000000000,0.393
000000000000,0.374000000000000,0.377000000000000;0.371000000000000,0.386000000000000,0.417000000000000,0.457000000000000
,0.503000000000000,0.440000000000000,0.420000000000000,0.408000000000000,0.414000000000000,0.399000000000000,0.382000000
000000,0.380000000000000;0.380000000000000,0.394000000000000,0.440000000000000,0.494000000000000,0.541000000000000,0.460
000000000000,0.468000000000000,0.457000000000000,0.445000000000000,0.418000000000000,0.396000000000000,0.388000000000000
;0.369000000000000,0.381000000000000,0.412000000000000,0.459000000000000,0.483000000000000,0.467000000000000,0.495000000
000000,0.468000000000000,0.439000000000000,0.411000000000000,0.389000000000000,0.382000000000000;0.339000000000000,0.338
000000000000,0.400000000000000,0.408000000000000,0.476000000000000,0.414000000000000,0.397000000000000,0.396000000000000
,0.396000000000000,0.377000000000000,0.345000000000000,0.327000000000000;0.407000000000000,0.406000000000000,0.429000000
000000,0.484000000000000,0.595000000000000,0.479000000000000,0.442000000000000,0.444000000000000,0.437000000000000,0.424
000000000000,0.400000000000000,0.388000000000000;0.365000000000000,0.392000000000000,0.446000000000000,0.553000000000000
,0.614000000000000,0.478000000000000,0.465000000000000,0.458000000000000,0.460000000000000,0.433000000000000,0.388000000
000000,0.367000000000000;0.314000000000000,0.336000000000000,0.382000000000000,0.457000000000000,0.511000000000000,0.408
000000000000,0.396000000000000,0.404000000000000,0.395000000000000,0.362000000000000,0.333000000000000,0.315000000000000
;0.397000000000000,0.423000000000000,0.496000000000000,0.597000000000000,0.639000000000000,0.491000000000000,0.496000000
000000,0.484000000000000,0.482000000000000,0.441000000000000,0.416000000000000,0.399000000000000;0.395000000000000,0.420
000000000000,0.485000000000000,0.584000000000000,0.616000000000000,0.482000000000000,0.490000000000000,0.479000000000000
,0.474000000000000,0.434000000000000,0.412000000000000,0.396000000000000;0.379000000000000,0.384000000000000,0.384000000
000000,0.428000000000000,0.447000000000000,0.475000000000000,0.489000000000000,0.447000000000000,0.420000000000000,0.408
000000000000,0.392000000000000,0.387000000000000;0.380000000000000,0.399000000000000,0.421000000000000,0.485000000000000
,0.498000000000000,0.453000000000000,0.469000000000000,0.446000000000000,0.428000000000000,0.404000000000000,0.390000000
000000,0.389000000000000];
LatDecim =
[32.5410000000000,32.6310000000000,29.7840000000000,35.0030000000000,30.2000000000000,28.5140000000000,31.8250000000000,
25.7780000000000,30.3500000000000,24.6140000000000,25.1720000000000,23.1610000000000,22.8970000000000,23.7580000000000,2
2.2960000000000,26.1890000000000,29.5110000000000,21.0140000000000,20.9830000000000,23.7670000000000,21.2890000000000,20
.6800000000000,20.5220000000000,25.6340000000000,20.8420000000000,29.6000000000000,29.4690000000000,19.4440000000000,26.
2170000000000,19.8500000000000,19.3370000000000,19.4360000000000,22.2500000000000,19.5500000000000,24.1250000000000,19.1
460000000000,20.4310000000000,27.3640000000000,26.6560000000000,19.4030000000000,23.0060000000000,17.0020000000000,23.62
80000000000,23.6310000000000,18.3190000000000,22.8720000000000,14.7940000000000,19.8200000000000,22.1240000000000];
LongDecim =
[-116.970000000000,-115.242000000000,-119.310000000000,-105.339000000000,-107.161000000000,-101.239000000000,-102.717000
000000,-100.107000000000,-111.411000000000,-110.006000000000,-110.111000000000,-106.266000000000,-102.687000000000,-109.
875000000000,-97.8660000000000,-112.839000000000,-104.911000000000,-101.266000000000,-101.483000000000,-95.5390000000000
,-94.5220000000000,-105.254000000000,-103.311000000000,-103.694000000000,-101.575000000000,-101.172000000000,-95.5220000
000000,-107.186000000000,-102.831000000000,-101.026000000000,-99.5660000000000,-99.0720000000000,-100.472000000000,-99.6
330000000000,-105.042000000000,-96.1870000000000,-97.1140000000000,-95.0750000000000,-97.9830000000000,-99.1080000000000
,-91.2780000000000,-96.7210000000000,-106.494000000000,-94.4750000000000,-93.3530000000000,-94.9640000000000,-92.3700000
000000,-84.9450000000000,-93.9410000000000];
LatLong = zeros(49,2);
LatLong (:,1) = LatDecim;
LatLong (:,2) = LongDecim;

%file_location = 'C:\Users\usuario\Documents\Christopher\NatClimDataCntr\New Foundland\Campeche766961_1989.csv';
%file_location = 'C:\Users\usuario\Documents\Christopher\NatClimDataCntr\New Foundland\76001399999_2011.csv'; %This is
for Tijuana
%file_location = 'C:\Users\usuario\Documents\Christopher\NatClimDataCntr\New Foundland\76679399999CDMXAirprt2014.csv';
%file_location = 'C:\Users\usuario\Documents\Christopher\NatClimDataCntr\New
Foundland\QUERETARO_INTERCONTINENTAL2022.csv';
%file_location = 'C:\Users\usuario\Documents\Christopher\NatClimDataCntr\New Foundland\765810RioVerde1980.csv';


CSVpath = 'C:\Users\usuario\Documents\Christopher\NatClimDataCntr\New Foundland\';

while 1
MetStationNumber = input('øN˙mero de identificacion del estaciÛn meteorolÛgico? ');
CSVFileList = dir([CSVpath,'**/*',num2str(MetStationNumber),'*.csv']);
if size(CSVFileList) > 0
    break
end

end

file_location = strcat(CSVFileList(11).folder,'\',CSVFileList(11).name);
```

```
optionsforimport = detectImportOptions(file_location);

% Notes for extracting EPW file to copy header and get 'FilePlace'

  Pathstem = 'C:\Users\usuario\Documents\Christopher\RawYearlyData\';
  EpwFileList = dir([Pathstem,'**/*',num2str(MetStationNumber),'*.epw']);
  FilePlace = extractBefore(string(EpwFileList(1).name),strlength(string(EpwFileList(1).name))-7);
  EpwFileNewFileName = extractBefore(string(EpwFileList(1).name),strlength(string(EpwFileList(1).name))-6);



WF_file = file_location;
    rWF_data = readtable(WF_file,optionsforimport);

    %extract the information from the header

    header = rWF_data(1,1);

    %Remove rows with duplicate date-times
    %Find column number for date-time
    tidx = find(string(rWF_data.Properties.VariableNames) == "DATE");
    %Make index for the duplicated rows for the date-time
    [C,ia] = unique(rWF_data(:,tidx),'rows');
    %Use the index 'ia' to make the cleaned up table
    WF_data = rWF_data(ia,:);

    %Find the size of the new table
    flag_end = size(WF_data,1);

    %Set up the fields in the data structure - helps with speed as the size
    %is decided before iterating through elements of the structure

WS_data.Pressure = zeros(flag_end,1);
WS_data.DryBulb = zeros(flag_end,1);
WS_data.DewPoint = zeros(flag_end,1);
WS_data.WindDir = zeros(flag_end,1);
WS_data.WindSpeed = zeros(flag_end,1);
WS_data.CloudCover = zeros(flag_end,1);
WS_data.OpaqueCover = zeros(flag_end,1);
WS_data.Visibility = zeros(flag_end,1);
WS_data.Ceiling = zeros(flag_end,1);
WS_data.Year = zeros(flag_end,1);
WS_data.Month = zeros(flag_end,1);
WS_data.Hour = zeros(flag_end,1);
WS_data.Minute = zeros(flag_end,1);
WS_data.Second = zeros(flag_end,1);
WS_data.HourDecimal = zeros(flag_end,1);
%WS_data.PresFilled = zeros(flag_end,1);
WS_data.Precipit = zeros(flag_end,1);
%CloudCoverRaw = zeros(flag_end,1);

Latitude = WF_data.LATITUDE(1);
Longitude = WF_data.LONGITUDE(1);
Altitude = WF_data.ELEVATION (1);

    %WF_date = [WF_data.data(1:8760,1:5)]; %, WF_data.textdata(9:flag_end,:)];
    %once extracted the data is all that is left to get.
    Temp3 = str2num(cell2mat(WF_data.SLP));
    WS_data.Pressure = Temp3(:,1)*10; % NOAA data in hectoPascal and EPW should be in Pa
    Temp3 = str2num(cell2mat(WF_data.TMP));
    WS_data.DryBulb = Temp3(:,1)/10;
    Temp3 = str2num(cell2mat(WF_data.DEW));
    WS_data.DewPoint = Temp3(:,1)/10;



    Temp3 = regexp(WF_data.WND,'\d*','match');
    Temp5 = regexp(WF_data.GF1,'\d*','match');
    %Temp6 = regexp(WF_data.DATE,'\d*','match');

    %Precipitation - need to fill empty cells to get that full array as
    %numbers
    if ismember('AA1',WF_data.Properties.VariableNames)
        Temp7 = WF_data.AA1;
    else
       Temp7 = cell(flag_end,1);
    end
    INDEX = cellfun('isempty',Temp7);
    Temp7(INDEX) = {'00,0000,0,0'};
    Temp7 = regexp(Temp7,'\d*','match');

    TempXX=regexp(WF_data.VIS,'\d*','match');
    for i = 2:flag_end
            if isempty(TempXX{i,1})
                TempXX{i,1} = TempXX{i-1,1};
```

```
        end
        Vis(i,1) = str2double(TempXX{i,1}{1});
    end
    Vis(1,1) = str2double(TempXX{1,1}{1});
    if isempty(TempXX{1,1})
            TempXX{1,1} = TempXX{2,1};
        end
    WS_data.Visibility = Vis;

    WS_data.PresentWeather = WF_data{1:flag_end,30};

    % "Cloud Cover
    % The ISH data format contains fields for total sky cover
    % and opaque sky cover, but the latter are mostly missing. The
    % total sky cover is read as the cloud cover data. To interpolate
    % for missing cloud cover data, linear approximations are used." Huang
    % et al.


    %Detect which has better quality Total Coverage or Lowest Cloud Cover
    %in GF1 - columnn 3 is quality for the former and column 5 for the
    %latter.
    %GF1 = vertcat(Temp5{:,1}); % Flatten the array of arrays of cells

    %Fill empty cells before applying vertcat to make sure all resulting
    %arrays are the same size

    %if isempty(Temp5{1,1})  % Need to improve this to take into account more than one missing value at start of year
    %    Temp5{1,1} = Temp5{2,1};
    %end

    Tmp5Cntr = 0;
    while isempty(Temp5{Tmp5Cntr+1,1})
        Tmp5Cntr = Tmp5Cntr + 1;
    end
    if isempty(Temp5{1,1})
        for i = 1:Tmp5Cntr
            Temp5{1,1} = Temp5{Tmp5Cntr+1,1};
        end
    end

    Wnd = zeros(flag_end,4);

    for i = 2:flag_end
            if isempty(Temp5{i,1})
                Temp5{i,1} = Temp5{i-1,1};
            end
            for j = 1:4
            Wnd(i,j) = str2double(Temp3{i,1}{j});
            end
    end

    GF1 = str2double(vertcat(Temp5{1:flag_end,1})); % Convert to array of real numbers in 2 dimensions

    WS_data.Ceiling = GF1(1:flag_end,8);

    WS_data.WindDir(1:flag_end,1) = Wnd(1:flag_end,1);
    WS_data.WindSpeed(1:flag_end,1) = Wnd(1:flag_end,3);

    %TotalCoverQual = ; Chooses which field has most data

    if  sum(GF1(:,3))<sum(GF1(:,5))
        CoverIndex = 1;
    else
        CoverIndex = 4;
    end

    Temp4 = zeros(size(Temp3,1),1);

    %Temp4 = Temp6{i,1}{2};
        WS_data.Year = year(WF_data.DATE);
        %Temp4 = Temp6{i,1}{1};
        WS_data.Month = month(WF_data.DATE);
        %Temp4 = Temp6{i,1}{3};
        WS_data.Day = day(WF_data.DATE);
        %Temp4 = Temp6{i,1}{4};
        WS_data.Hour = hour(WF_data.DATE);
        %Temp4 = Temp6{i,1}{5};
        WS_data.Minute = minute(WF_data.DATE);
        %Temp4 = Temp6{i,1}{6};
        WS_data.Second = second(WF_data.DATE);

% For opaque cover the field index is 2

    for i=1:flag_end
```

```
            switch GF1(i,CoverIndex) %str2double(Temp4) %( - use switch to assign values of cloud cover in tenths)
assume that 99 is clear sky
                case 0
                    WS_data.CloudCover(i,1) = 0;
                case 1
                    WS_data.CloudCover(i,1) = 1;
                case 2
                    WS_data.CloudCover(i,1) = 2.5;
                case 3
                    WS_data.CloudCover(i,1) = 4;
                case 4
                    WS_data.CloudCover(i,1) = 5;
                case 5
                    WS_data.CloudCover(i,1) = 6;
                case 6
                    WS_data.CloudCover(i,1) = 7.5;
                case 7
                    WS_data.CloudCover(i,1) = 9;
                case 8
                    WS_data.CloudCover(i,1) = 10;
                case 99
                    WS_data.CloudCover(i,1) = 0;
                otherwise
                    WS_data.CloudCover(i,1) = 0;
            end

            switch GF1(i,2) %str2double(Temp4) %( - use switch to assign values of cloud cover in tenths) assume that 99
is clear sky
                case 0
                    WS_data.OpaqueCover(i,1) = 0;
                case 1
                    WS_data.OpaqueCover(i,1) = 1;
                case 2
                    WS_data.OpaqueCover(i,1) = 2.5;
                case 3
                    WS_data.OpaqueCover(i,1) = 4;
                case 4
                    WS_data.OpaqueCover(i,1) = 5;
                case 5
                    WS_data.OpaqueCover(i,1) = 6;
                case 6
                    WS_data.OpaqueCover(i,1) = 7.5;
                case 7
                    WS_data.OpaqueCover(i,1) = 9;
                case 8
                    WS_data.OpaqueCover(i,1) = 10;
                case 99
                    WS_data.OpaqueCover(i,1) = 0;
                otherwise
                    WS_data.OpaqueCover(i,1) = 0;
            end

        Temp4 = Temp7{i,1}{2};
        WS_data.Precipit(i,1) = str2double(Temp4);



    end

%SKY-CONDITION-OBSERVATION total lowest cloud cover code
%The code that represents the fraction of the celestial dome covered by all low clouds present. If
%no low clouds are present; the code denotes the fraction covered by all middle level clouds present.
%DOM: A specific domain comprised of the characters in the ASCII character set.
% 00 = None
% 01 = One okta or 1/10 or less but not zero
%02 = Two oktas or 2/10 - 3/10
% 03 = Three oktas or 4/10
% 04 = Four oktas or 5/10
% 05 = Five oktas or 6/10
% 06 = Six oktas or 7/10 - 8/10
% 07 = Seven oktas or 9/10 or more but not 10/10
% 08 = Eight oktas or 10/10
% 09 = Sky obscured, or cloud amount cannot be estimated
% 10 = Partial obscuration
% 11 = Thin Scattered
% 12 = Scattered
% 13 = Dark Scattered
% 14 = Thin Broken
% 15 = Broken
% 16 = Dark Broken
% 17 = Thin Overcast
% 18 = Overcast
% 19 = Dark overcast
% 99 = Missing
```

```
%Decimal hour of the year

%Day number for day before the first day of each month except leap years
Daynumber = [0,31,59,90,120,151,181,212,243,273,304,334,365];

Leap = ~mod(WS_data.Year(1),4); %This works for 2000 because it is exactly divisable
%by 400 as well - otherwise years which are divisable by 100 are not leap
%years - the data processed here doesn't extend back to 1900 or to 2100
%which are not leap years.
if Leap
        Daynumber = [0,31,60,91,121,152,182,213,244,274,305,335,366];
end


for i=1:flag_end
   switch WS_data.Month(i)
       case 1
           WS_data.HourDecimal(i,1) = Daynumber(1)*24;
       case 2
           WS_data.HourDecimal(i,1) = Daynumber(2)*24;
       case 3
           WS_data.HourDecimal(i,1) = Daynumber(3)*24;
       case 4
           WS_data.HourDecimal(i,1) = Daynumber(4)*24;
       case 5
           WS_data.HourDecimal(i,1) = Daynumber(5)*24;
       case 6
           WS_data.HourDecimal(i,1) = Daynumber(6)*24;
       case 7
           WS_data.HourDecimal(i,1) = Daynumber(7)*24;
       case 8
           WS_data.HourDecimal(i,1) = Daynumber(8)*24;
       case 9
           WS_data.HourDecimal(i,1) = Daynumber(9)*24;
       case 10
           WS_data.HourDecimal(i,1) = Daynumber(10)*24;
       case 11
           WS_data.HourDecimal(i,1) = Daynumber(11)*24;
       case 12
           WS_data.HourDecimal(i,1) = Daynumber(12)*24;
   end

end

WS_data.HourDecimal = WS_data.HourDecimal+WS_data.Day*24+WS_data.Hour+WS_data.Minute/60+WS_data.Second/3600-24;

% Data filling from TMY2 manual
% Before filling missing data check for periods of missing data that exceed
% 47 hours - if there is such a gap then reject the file: TMY2 class B
% stations  can have upto 47 hours' break in the data.

% But https://energy-models.com/users-manual-tmy2s contains the following:
% Note the treatment of 5 hour gaps.

% The NSRDB contains filled data for total and opaque sky cover, dry bulb
% temperature, relative humidity, and atmospheric pressure. NSRDB data gaps
% up to 5 hours were filled by linear interpolation. Gaps from 6 to 47 hours
% were filled for the above elements by using data from adjacent days for
% identical hours and then by adjusting the data so that there were no abrupt
% changes in data values between the filled and measured data. Many Class B
% stations did not operate for parts of the night and/or early morning and
% late afternoon. For these stations, NSRDB data were filled from sunrise to
% sunset to allow model estimates of solar radiation. However, nighttime data
% were not necessarily filled.

% Fill missing data using spline interpolation - need to add detection of
%
************************************************************************************************************************
************************************************************************
% whole days missing and use TMY3 method. TMY3 uses TMY2 data filling
% methods "TMY2s for Class A stations were restricted to the selection of
% typical months that had no more than 2 consecutive hours of data missing,
% whereas Class B stations could have up to 47 consecutive hours of data missing.
% Gaps from 6 to 47 hours were filled for the above elements by using data
% from adjacent days for identical hours and then by adjusting the data so
% that there were no abrupt changes in data values between the filled and measured data.

% To detect breaks of more than **five hours** - setup an array of the
% differences in hours between data points. Set up an index of these gaps.
% Data missing at beginning or end of year will require separate method.
% Splice in data, half from before break and half from after and smooth the
% transitions (splice over 4 data points). Use indices for each half of
% splice synchronised with time of day to pick up and paste the fill-in
% data.
```

```
% The data resampling process will interpolate the gaps of less than five
% hours with a spline.

WS_data.HourDecimalDiff = zeros(flag_end,1);
%WS_data.GapMidPointHrDec = zeros(flag_end,1);
WS_data.HourDecimalDiff(1:flag_end-1,1) = WS_data.HourDecimal(2:flag_end)-WS_data.HourDecimal(1:flag_end-1);

% Detection of excessive gaps in data i.e. periods of greater than 47 hours
% duration.

if sum(WS_data.HourDecimalDiff > 47) | (WS_data.HourDecimal(1,1) > 47)
    Mssg = 'Existe uno o m·s discontinuidades mayores de 47 horas en los datos';
    disp(Mssg);
    Mssg = ['Es necesario descartar ',file_location];
    disp(Mssg);
end


WS_data.HourDecimalDiff = zeros(flag_end,1);

WS_data.HourDecimalDiff(1:flag_end-1,1) = WS_data.HourDecimal(2:flag_end)-WS_data.HourDecimal(1:flag_end-1);

WS_data.HourDecimalMissingINDEX = ((WS_data.HourDecimalDiff > 5)& (WS_data.HourDecimalDiff < 24)) ;

%Start of step through file to find gaps of between 5 and 47 hours and
%fill them to new data structure (Has to be expanded to accomadate more
%data points.

NewDataCntr = 1;
OldDataCntr = 1;

% Needs code to catch a gap at the end of the year (24 April 2023)

while OldDataCntr <= flag_end-1
    if WS_data.HourDecimal(1,1)>5 % Case of first data point more than five hours into the year
        if WS_data.HourDecimal(1,1)>24
            FilSrc=24;
        else
            FilSrc=48;
        end
        INDEXfil = WS_data.HourDecimal > FilSrc & WS_data.HourDecimal < FilSrc + WS_data.HourDecimal(1,1);
        TmpArry = WS_data.Pressure(INDEXfil);
        LngthTarry = size(TmpArry,1);
        WS_dataNw.Pressure(1:LngthTarry,1) = TmpArry; % This needs to be done for all variables in WS_data
        TmpArry = WS_data.DryBulb(INDEXfil);
        WS_dataNw.DryBulb(1:LngthTarry,1) = TmpArry;
        TmpArry = WS_data.DewPoint(INDEXfil);
        WS_dataNw.DewPoint(1:LngthTarry,1) = TmpArry;
        TmpArry = WS_data.WindDir(INDEXfil);
        WS_dataNw.WindDir(1:LngthTarry,1) = TmpArry;
        TmpArry = WS_data.WindSpeed(INDEXfil);
        WS_dataNw.WindSpeed(1:LngthTarry,1) = TmpArry;
        TmpArry = WS_data.CloudCover(INDEXfil);
        WS_dataNw.CloudCover(1:LngthTarry,1) = TmpArry;
        TmpArry = WS_data.Precipit(INDEXfil);
        WS_dataNw.Precipit(1:LngthTarry,1) = TmpArry;
        TmpArry = WS_data.WS_data.OpaqueCover(INDEXfil);
        WS_dataNw.WS_data.OpaqueCover(1:LngthTarry,1) = TmpArry;
        TmpArry = WS_data.WS_data.Visibility(INDEXfil);
        WS_dataNw.WS_data.Visibility(1:LngthTarry,1) = TmpArry;
        TmpArry = WS_data.WS_data.Ceiling(INDEXfil);
        WS_dataNw.WS_data.Ceiling(1:LngthTarry,1) = TmpArry;
        TmpArry = WS_data.HourDecimal(INDEXfil)-FilSrc;
        WS_dataNw.HourDecimal(1:LngthTarry,1) = TmpArry;
        NewDataCntr = LngthTarry+1;
    end


    WS_dataNw.Pressure(NewDataCntr,1) = WS_data.Pressure(OldDataCntr,1);
    WS_dataNw.DryBulb(NewDataCntr,1) = WS_data.DryBulb(OldDataCntr,1);
    WS_dataNw.DewPoint(NewDataCntr,1) = WS_data.DewPoint(OldDataCntr,1);
    WS_dataNw.WindDir(NewDataCntr,1) = WS_data.WindDir(OldDataCntr,1);
    WS_dataNw.WindSpeed(NewDataCntr,1) = WS_data.WindSpeed(OldDataCntr,1);
    WS_dataNw.CloudCover(NewDataCntr,1) = WS_data.CloudCover(OldDataCntr,1);
    WS_dataNw.Precipit(NewDataCntr,1) = WS_data.Precipit(OldDataCntr,1);
    WS_dataNw.OpaqueCover(NewDataCntr,1) = WS_data.OpaqueCover(OldDataCntr,1);
    WS_dataNw.Visibility(NewDataCntr,1) = WS_data.Visibility(OldDataCntr,1);
    WS_dataNw.Ceiling(NewDataCntr,1) = WS_data.Ceiling(OldDataCntr,1);
    WS_dataNw.HourDecimal(NewDataCntr,1) = WS_data.HourDecimal(OldDataCntr,1);

    if WS_data.HourDecimal(OldDataCntr+1,1)-WS_data.HourDecimal(OldDataCntr,1)>5 % Catch a gap of more than five hours
        if WS_data.HourDecimal(OldDataCntr+1,1)-WS_data.HourDecimal(OldDataCntr,1)>24 % Catch if the gap is more than 24
hours
                                                                                        % - if it is more than 47 hours
this will have been reported earlier.
```

```
            FilSrc = 48;
        else
            FilSrc = 24;
        end
        % Index for data to take from before gap: a whole day or two days
        % before for the first half of the gap
        INDEXfil = WS_data.HourDecimal < ((WS_data.HourDecimal(OldDataCntr+1)-WS_data.HourDecimal(OldDataCntr))/2 +
WS_data.HourDecimal(OldDataCntr)-FilSrc)...
            & (WS_data.HourDecimal > WS_data.HourDecimal(OldDataCntr)-FilSrc);
        clear TmpArry;
        TmpArry = WS_data.Pressure(INDEXfil);
        LngthTarry = size(TmpArry,1);
        WS_dataNw.Pressure(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.DryBulb(INDEXfil);
        WS_dataNw.DryBulb(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.DewPoint(INDEXfil);
        WS_dataNw.DewPoint(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.WindDir(INDEXfil);
        WS_dataNw.WindDir(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.WindSpeed(INDEXfil);
        WS_dataNw.WindSpeed(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.CloudCover(INDEXfil);
        WS_dataNw.CloudCover(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.OpaqueCover(INDEXfil);
        WS_dataNw.OpaqueCover(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.Visibility(INDEXfil);
        WS_dataNw.Visibility(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.Ceiling(INDEXfil);
        WS_dataNw.Ceiling(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.Precipit(INDEXfil);
        WS_dataNw.Precipit(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.HourDecimal(INDEXfil)+FilSrc;
        WS_dataNw.HourDecimal(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        NewDataCntr = NewDataCntr + LngthTarry;
        % Check this logic to make sure that it is getting the right
        % data interval
        INDEXfil = WS_data.HourDecimal > ((WS_data.HourDecimal(OldDataCntr+1)-WS_data.HourDecimal(OldDataCntr))/2 +
WS_data.HourDecimal(OldDataCntr)+FilSrc)...
            & (WS_data.HourDecimal < WS_data.HourDecimal(OldDataCntr+1)+FilSrc);
        TmpArry = WS_data.Pressure(INDEXfil);
        LngthTarry = size(TmpArry,1);
        WS_dataNw.Pressure(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.DryBulb(INDEXfil);
        WS_dataNw.DryBulb(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.DewPoint(INDEXfil);
        WS_dataNw.DewPoint(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.WindDir(INDEXfil);
        WS_dataNw.WindDir(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.WindSpeed(INDEXfil);
        WS_dataNw.WindSpeed(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.CloudCover(INDEXfil);
        WS_dataNw.CloudCover(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.Visibility(INDEXfil);
        WS_dataNw.Visibility(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.OpaqueCover(INDEXfil);
        WS_dataNw.OpaqueCover(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.Precipit(INDEXfil);
        WS_dataNw.Precipit(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
        TmpArry = WS_data.HourDecimal(INDEXfil)-FilSrc;
        WS_dataNw.HourDecimal(NewDataCntr+1:NewDataCntr+LngthTarry,1) = TmpArry;
    end
    OldDataCntr = OldDataCntr+1;
    NewDataCntr = NewDataCntr+1;
end
clear TmpArry;
if WS_data.HourDecimal(flag_end,1)< 8755 % Case of last data point more than five hours before the end of the year
    if WS_data.HourDecimal(flag_end,1)> 8736
        FilSrc=24;
    else
        FilSrc=48;
    end
    INDEXfil = WS_data.HourDecimal < 8760 - FilSrc & WS_data.HourDecimal > WS_data.HourDecimal(flag_end,1) - FilSrc;
    TmpArry = WS_data.Pressure(INDEXfil);
    LngthTarry = size(TmpArry,1);
    WS_dataNw.Pressure(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry; % This needs to be done for all variables in
WS_data
    TmpArry = WS_data.DryBulb(INDEXfil);
    WS_dataNw.DryBulb(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry;
    TmpArry = WS_data.DewPoint(INDEXfil);
    WS_dataNw.DewPoint(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry;
    TmpArry = WS_data.WindDir(INDEXfil);
    WS_dataNw.WindDir(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry;
    TmpArry = WS_data.WindSpeed(INDEXfil);
    WS_dataNw.WindSpeed(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry;
    TmpArry = WS_data.CloudCover(INDEXfil);
    WS_dataNw.CloudCover(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry;
```

```
    TmpArry = WS_data.Visibility(INDEXfil);
    WS_dataNw.Visibility(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry;
    TmpArry = WS_data.OpaqueCover(INDEXfil);
    WS_dataNw.OpaqueCover(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry;
    TmpArry = WS_data.Ceiling(INDEXfil);
    WS_dataNw.Ceiling(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry;
    TmpArry = WS_data.Precipit(INDEXfil);
    WS_dataNw.Precipit(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry;
    TmpArry = WS_data.HourDecimal(INDEXfil) + FilSrc;
    WS_dataNw.HourDecimal(NewDataCntr+1:LngthTarry+NewDataCntr,1) = TmpArry;
    NewDataCntr = NewDataCntr + LngthTarry+1;
    end

INDEX = (WS_dataNw.DryBulb > 998); % Dry bulb needs to go before Pressure as it is used if there is no pressure data

 WS_dataNw.DryBulb(INDEX) = NaN;
 A = WS_dataNw.DryBulb;
 t = WS_dataNw.HourDecimal;

 WS_dataNw.DryBulb = fillmissing(A,'spline','samplepoints',t);



% Pressure -
% First put Nan in for missing values - signalled by 99999 HectoPascal or
% 999990 Pa
% But if all the pressure data are missing then put in value based on
% temperature and altitude

 %Station Pressure = Sea Level Pressure x e^( -elevation / (temperature x 29.263))
 %elevation = the elevation of the station in meters
 %Standard sea level pressure in Pa is 101325 Pa
 %temperature in K = temperature (∞C) + 273.15

 if all(WS_dataNw.Pressure > 999980) % In Pa
     WS_dataNw.Pressure = 101325*exp(-Altitude./((WS_dataNw.DryBulb + 273.15)*29.263));
 else
 INDEX = (WS_dataNw.Pressure > 999980); % In Pa

 WS_dataNw.Pressure(INDEX) = NaN;
 % Adjust reported sealevel pressure (Field SLP) to station pressure
 WS_dataNw.Pressure(~INDEX) = WS_dataNw.Pressure(~INDEX).*exp(-Altitude./((WS_dataNw.DryBulb(~INDEX) + 273.15)*29.263));

 A = WS_dataNw.Pressure;
 WS_dataNw.Pressure = fillmissing(A,'spline','samplepoints',t);
 end



 INDEX = (WS_dataNw.DewPoint > 998);

 WS_dataNw.DewPoint(INDEX) = NaN;
 A = WS_dataNw.DewPoint;

 WS_dataNw.DewPoint = fillmissing(A,'spline','samplepoints',t);


 INDEX = (WS_dataNw.WindDir > 998);

 WS_dataNw.WindDir(INDEX) = NaN;
 A = WS_dataNw.WindDir;

 WS_dataNw.WindDir = fillmissing(A,'spline','samplepoints',t);


 INDEX = (WS_dataNw.WindSpeed > 999.8);

 WS_dataNw.WindSpeed(INDEX) = NaN;
 A = WS_dataNw.WindSpeed;

 WS_dataNw.WindSpeed = fillmissing(A,'spline','samplepoints',t);

 % Remove negative windspeeds
 INDEX = (WS_dataNw.WindSpeed < 0);
 WS_dataNw.WindSpeed(INDEX) = 0;


 INDEX = (WS_dataNw.CloudCover > 98);

 WS_dataNw.CloudCover(INDEX) = NaN;
 A = WS_dataNw.CloudCover;

 WS_dataNw.CloudCover = fillmissing(A,'linear','samplepoints',t);

 %Interpolation into regular on the hour data
```

```
 WS_data_resampled.Pressure = zeros(8760,1);
 WS_data_resampled.DryBulb = zeros(8760,1);
 WS_data_resampled.DewPoint = zeros(8760,1);
 WS_data_resampled.WindDir = zeros(8760,1);
 WS_data_resampled.WindSpeed = zeros(8760,1);
 WS_data_resampled.CloudCover = zeros(8760,1);
 WS_data_resampled.Visibility = zeros(8760,1);
 WS_data_resampled.OpaqueCover = zeros(8760,1);
 WS_data_resampled.Ceiling = zeros(8760,1);
 WS_data_resampled.Year = zeros(8760,1);
 WS_data_resampled.Month = zeros(8760,1);
 WS_data_resampled.DayOfMonth = zeros (8760,1);
 WS_data_resampled.Hour = zeros(8760,1);
 WS_data_resampled.Minute = zeros(8760,1);
 WS_data_resampled.Second = zeros(8760,1);
 WS_data_resampled.DayOfYear = zeros(8760,1);
 WS_data_resampled.Precipit = zeros(8760,1);
 WS_data_resampled.PresentWeather = zeros(8760,1);

 EPWData = zeros(8760,29);

 Hournumber = [0,744,1416,2160,2880,3624,4344,5088,5832,6552,7296,8016,8760];


% ********************************************

%     UTC offset pickup from KoppenClass

    KoppenStr = KoppenClass(:,1);
    KoppClass = KoppenClass(contains(KoppenStr,FilePlace),2);
    UTCdiff = KoppenClass(contains(KoppenStr,FilePlace),3);
    UTCdiffd = cell2mat(UTCdiff); %str2double(cell2mat(UTCdiff)); % Time zone difference from UTC (GMT)
    % hours - the time in the raw data file is UTC



 RelativeToUTC = UTCdiffd;
 for imonth = 1:12
     for irsmpld = Hournumber(imonth)+1:Hournumber(imonth+1)
         WS_data_resampled.Month(irsmpld) = imonth;
         WS_data_resampled.DayOfMonth(irsmpld) = fix((irsmpld-Hournumber(imonth))/24)+1;
     end
 end

 for irsmpld = 1:8760
    WS_data_resampled.Year(irsmpld) = WS_data.Year(1,1);
 end


 %This is data on the hour
 regular_hours = (1:1:8760);

 A = mod(regular_hours+23,24)+1;
 WS_data_resampled.Hour = reshape(A,[],1);
 A = fix((regular_hours-0.1)/24)+1;
 WS_data_resampled.DayOfYear = reshape(A,[],1);



 A = interp1(WS_dataNw.HourDecimal,WS_dataNw.Pressure,regular_hours,'spline','extrap');
 WS_data_resampled.Pressure = reshape(A,[],1);
 A = interp1(WS_dataNw.HourDecimal,WS_dataNw.DryBulb,regular_hours,'spline','extrap');
 WS_data_resampled.DryBulb = reshape(A,[],1);
 A = interp1(WS_dataNw.HourDecimal,WS_dataNw.DewPoint,regular_hours,'spline','extrap');
 WS_data_resampled.DewPoint = reshape(A,[],1);
 % Check that no dewpoint exceeds drybulb - if it does then it is equal to
 % drybulb
INDEX = (WS_data_resampled.DewPoint > WS_data_resampled.DryBulb);
WS_data_resampled.DewPoint(INDEX) = WS_data_resampled.DryBulb(INDEX);


 A = interp1(WS_dataNw.HourDecimal,WS_dataNw.WindDir,regular_hours,'nearest','extrap'); %This then uses preceding wind
direction to fill half of gap and following wind direction to fill the second half of gap
 % Wind direction needs to be rounded to nearest 10∞
 A = round(A,-1);
 % Add 360 to negative directions
 IndexWinD = A < 0;
 A(IndexWinD) = A(IndexWinD) + 360;
 % Correct values greater than 360
 IndexWinD = A > 360;
 A(IndexWinD) = mod(A(IndexWinD),360);
 WS_data_resampled.WindDir = reshape(A,[],1);

 A = interp1(WS_dataNw.HourDecimal,WS_dataNw.WindSpeed,regular_hours,'spline','extrap');
 WS_data_resampled.WindSpeed = reshape(A,[],1);
```

```
% Remove negative windspeeds
INDEX = (WS_data_resampled.WindSpeed < 0);
WS_data_resampled.WindSpeed(INDEX) = 0;


A = interp1(WS_dataNw.HourDecimal,WS_dataNw.CloudCover,regular_hours,'linear','extrap');
WS_data_resampled.CloudCover = reshape(A,[],1);
INDEX = (WS_data_resampled.CloudCover < 0);
WS_data_resampled.CloudCover(INDEX) = 0;

A = interp1(WS_dataNw.HourDecimal,WS_dataNw.Visibility,regular_hours,'linear','extrap');
WS_data_resampled.Visibility = reshape(A,[],1);
INDEX = (WS_data_resampled.Visibility < 0);
WS_data_resampled.Visibility(INDEX) = 0;

A = interp1(WS_dataNw.HourDecimal,WS_dataNw.Ceiling,regular_hours,'linear','extrap');
WS_data_resampled.Ceiling = reshape(A,[],1);
INDEX = (WS_data_resampled.Ceiling < 0);
WS_data_resampled.Ceiling(INDEX) = 0;

A = interp1(WS_dataNw.HourDecimal,WS_dataNw.OpaqueCover,regular_hours,'linear','extrap');
WS_data_resampled.OpaqueCover = reshape(A,[],1);
INDEX = (WS_data_resampled.OpaqueCover < 0);
WS_data_resampled.OpaqueCover(INDEX) = 0;



% Sum of precipitation reported over the regular hour
INPRECIP = (fix(WS_dataNw.HourDecimal) == regular_hours-1); %fix rounds each element to the nearest integer toward zero
for i = 1:8760
    WS_data_resampled.Precipit(i,1) = sum(WS_dataNw.Precipit(INPRECIP(:,i)));
end


%      **********************************************************************


% processing the solar.

% Solar calculations https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7085731/

% Declination angle

WS_data_resampled.Declination = -0.40927971*sin(2*pi*(WS_data_resampled.DayOfYear+284)/365); % in radians ASHRAE
fundamentals 2021 SI equation 10 from Chpt. 14 page 14.10
                                                                                                    % 23.45 degrees =
0.40927971 radians
% Local time - in Mexico there are two time zones which cover most of the
% country UTC -6 Baja California UTC -8 The state of Baja California Sur, Sonora and
% Chihuahua uses UTC -7. Quintana Roo uses UTC -5.

%ASHRAE fundamentals 2021 SI equations 5 & 6 from Chpt. 14 page 14.9
%Equation of time

B = (WS_data_resampled.DayOfYear-1)*2*pi/365; % B is in radians - Equation 6 Chpt 14 page 14.9 2021
E = 2.2918*(0.0075+0.1868*cos(B) - 3.2077*sin(B) - 1.4615*cos(2*B) - 4.089*sin(2*B)); % E is in minutes - Equation 5
Chpt 14 page 14.9 2021

% ASHRAE handbook 2019 HVAC Applications Chapter 36 Equation 5
% ET = 9.87 sin 2B ñ 7.53 cos B ñ 1.5 sin B     (5)
% where B = 0.989(N ñ 81) & N = day of year, with January 1 = 1
% B = 0.989*(WS_data_resampled.DayOfYear - 81)*2*pi/365; % B is in radians
% E = 9.87*sin(2*B) - 7.53*cos(B) - 1.5*sin(B);


Lst = RelativeToUTC*15; % 0 degrees because the raw data files are in UTC  -  but resampled is in local time zone civil
% time as are EPW files
Loc = Longitude; % in degrees

WS_data_resampled.SolarTime = mod(WS_data_resampled.Hour + (4*(Lst-Loc)-E)/60,24); % In hours
% According to AHSHRAE Applications 2019 Chpt. 36 Equation 4 - needs equation of time.
% AST = LST + ET + (4 min) (LST meridian ñ Local longitude)



WS_data_resampled.HourAngle = (WS_data_resampled.SolarTime-12)*pi/12; % In radians
% Solar altitude angle - equation 12 Chpt 14 page 14.10 2021
% WS_data_resampled.SolAltAngle = asin(sin(WS_data_resampled.Declination)*sin(Latitude*pi*2/360)...
%    +cos(WS_data_resampled.Declination).*cos(Latitude*pi*2/360).*cos(WS_data_resampled.HourAngle));

% Solar altitude angle from SolarAzEl Programed by Darin C. Koblick
% 2/17/2009 and Darin C. Koblick 4/16/2013 Vectorized for Speed

DateVector = zeros (8760,6);
DateVector(1:8760,1) = WS_data_resampled.Year;
DateVector(1:8760,2) = WS_data_resampled.Month;
```

```
 DateVector(1:8760,3) = WS_data_resampled.DayOfMonth;
 DateVector(1:8760,4) = WS_data_resampled.Hour;
 DateVector(1:8760,5) = WS_data_resampled.Minute;
 DateVector(1:8760,6) = WS_data_resampled.Second;

 if UTCdiffd <= -7.9
     UTCcity = 'America/Tijuana';
 elseif UTCdiffd <= -6.9
     UTCcity = 'America/Hermosillo';
 elseif UTCdiffd <= -5.9
     UTCcity = 'America/Mexico_City';
 elseif UTCdiffd <= -4.9
     UTCcity = 'America/Cancun';
 end

 t1 = datetime(DateVector,'TimeZone',UTCcity); % datetime(DateVector,'TimeZone','America/Mexico_City');  EPW time is
local timezone
 t1.TimeZone = 'UTC';
 %jd = juliandate(t1);
 Latvector(1:8760,1) = Latitude;
 Longvector(1:8760,1) = Longitude;
 Altvector(1:8760,1) = Altitude/1000;
 % Darin Koblick (2024). Vectorized Solar Azimuth and Elevation Estimation
 % (https://www.mathworks.com/matlabcentral/fileexchange/23051-vectorized-solar-azimuth-and-elevation-estimation),
 % MATLAB Central File Exchange. Retrieved September 18, 2024.
 [Az,El] = SolarAzEl(t1,Latvector,Longvector,Altvector);
 WS_data_resampled.SolAltAngle = deg2rad(El);

 % Relative humidity

 % For relative humidity: saturated partial pressure (pws) and dew point partial pressure (pw) is needed
 % The saturation pressure over ice for the temperature range of -100 to 0∞C is given by
 % ln pws = C1/T + C2 + C3T + C4T^2 + C5T^3 + C6T^4 + C7 ln T
 % where
C1 = -5.6745359e+3;
C2 = 6.3925247e00;
C3 = -9.6778430e-3;
C4 = 6.2215701e-7;
C5 = 2.0747825e-9;
C6 = -9.4840240e-13;
C7 = 4.1635019e00;
%
% The saturation pressure over liquid water for the temperature range of 0 to 200∞C is given by
% ln pws = C8/T + C9 + C10T + C11T^2 + C12^T 3 + C13 ln T
% where
%
C8 = -5.8002206e+03;
C9 = 1.3914993e+00;
C10 = -4.8640239e-02;
C11 = 4.1764768e-05;
C12 = -1.4452093e-08;
C13 = 6.5459673E+00;
%
% In both Equations (5) and (6),
% pws = saturation pressure, Pa
% T = absolute temperature, K = ∞C + 273.15
%
  T = WS_data_resampled.DryBulb+273.15;
  % Then equation 6
  pws = exp(C8./T + C9 + C10.*T + C11*T.^2 + C12*T.^3 + C13*log(T))/1e3;
  INDEX = (WS_data_resampled.DryBulb < 0);
  % Then equation 5
  pws(INDEX) = exp(C1./T(INDEX) + C2 + C3*T(INDEX) + C4*T(INDEX).^2 + C5*T(INDEX).^3 + C6*T(INDEX).^4 +
C7*log(T(INDEX))))/1e3;

    T = WS_data_resampled.DewPoint+273.15;
  % Then equation 6
  pw = exp(C8./T + C9 + C10.*T + C11*T.^2 + C12*T.^3 + C13*log(T))/1e3;
  INDEX = (WS_data_resampled.DewPoint < 0);
  % Then equation 5
  pw(INDEX) = exp(C1./T(INDEX) + C2 + C3*T(INDEX) + C4*T(INDEX).^2 + C5*T(INDEX).^3 + C6*T(INDEX).^4 +
C7*log(T(INDEX))))/1e3;

 WS_data_resampled.RelativeHumidity = (pw./pws)*100;

 WS_data_resampled.EstimHourlySol = zeros(8760,1);
 WS_data_resampled.EstimDaylySol = zeros(365,24,1);
 WS_data_resampled.Taub = zeros(365,1);
 WS_data_resampled.Taud = zeros(365,1);

 % Day light hours index
 INDEX = (WS_data_resampled.SolAltAngle > 0); %Don't estimate solar radiation at night

 % Use correlations from Development of 3012 IWEC2Weather Files
 % for International Locations (RP-1477) Huang et al ASHRAE Transactions
 % Volume 120 part 1 pp 340 - 355 - Table 2
```

```
 % Change the following to make using the coefficients easier

ConstsSol(1,1:7) = [0.97136,0.24936,-0.32165,0.03768,-0.0076,0.00794,-2.23175]; % Af
ConstsSol(2,1:7) = [0.71868,-0.11359,-0.07259,0.01038,-0.00285,0.00866,-8.42023]; % Am
%ConstsSol.As = [0.8089,0.07355,-0.40101,-0.00424,-0.00242,0.00342,-8.395]; % As has the same criteria as Aw and same
the regression coefficients
ConstsSol(3,1:7) = [0.8089,0.07355,-0.40101,-0.00424,-0.00242,0.00342,-8.395]; % Aw
ConstsSol(6,1:7) = [0.68149,-0.04697,-0.2842,0.01726,-0.00081,0.00453,-8.91306]; % BSh
ConstsSol(7,1:7) = [0.57692,-0.08062,-0.24399,0.0261,0.00054,0.00277,-1.21103]; % BSk
ConstsSol(4,1:7) = [0.51315,0.1554,-0.42157,0.01427,-0.00035,0.00469,-9.55426]; % BWh
ConstsSol(5,1:7) = [0.6997,-0.03001,-0.27693,0.01529,-0.00169,0.0051,-9.23071]; % BWk
ConstsSol(14,1:7) = [0.67839,0.03646,-0.39075,0.01359,-0.00148,0.0073,-8.71373]; % Cfa
ConstsSol(15,1:7) = [0.7437,-0.02988,-0.26353,0.02606,-0.00323,-8E-05,-1.97366]; % Cfb
ConstsSol(16,1:7) = [0.7437,-0.02988,-0.26353,0.02606,-0.00323,-8E-05,-1.97366]; % Cfc
ConstsSol(8,1:7) = [0.54698,0.21871,-0.42476,0.0233,-0.00038,0.00183,-9.83335]; % Csa
ConstsSol(9,1:7) = [0.6425,0.06685,-0.35491,0.01935,-0.00111,1E-05,-5.40989]; % Csb
ConstsSol(11,1:7) = [0.68175,0.15988,-0.43455,0.01972,-0.00303,0.00201,-6.67731]; % Cwa
ConstsSol(12,1:7) = [0.65533,-0.00683,-0.13621,0.0324,-0.00252,-0.0032,0.17022]; % Cwb
ConstsSol(25,1:7) = [0.73067,0.04956,-0.32687,0.01269,-0.00298,0.00581,0.30725]; % Dfa
ConstsSol(26,1:7) = [0.69491,-0.10822,-0.22999,0.01232,-0.00091,0.0039,-3.46883]; % Dfb
ConstsSol(27,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dfc
ConstsSol(17,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dsa
ConstsSol(18,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dsb
ConstsSol(19,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dsc
ConstsSol(21,1:7) = [0.66459,0.59148,-0.80274,0.03374,-0.00376,0.01662,-21.1779]; % Dwa
ConstsSol(22,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dwb
ConstsSol(23,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dwc
ConstsSol(24,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dwd
ConstsSol(29,1:7) = [0.77029,0.00687,-0.35561,0.01849,-0.00149,-0.00278,-6.41702]; % ET
ConstsSol(30,1:7) = [0.77029,0.00687,-0.35561,0.01849,-0.00149,-0.00278,-6.41702]; % EF

for i = 1:12
    Tmonth(1,1,i) = mean(WS_data_resampled.DryBulb(Hournumber(i)+1:Hournumber(i+1),1));
    P(1,1,i) = sum(WS_data_resampled.Precipit(Hournumber(i)+1:Hournumber(i+1),1));
end

%[Class, BroadClass] = KoppenGeiger(Tmonth,P,classes);
 %Sc0 = 0.5598;
 %Sc1 = 0.4982;
 %Sc2 = -0.6762;
 %Sc3 = 0.02842;
 %Sc4 = -0.00317;
 %Sc5 = 0.014;
 %Sd = -17.853;
 %Sk = 0.843;


 Class = cell2mat(classes(contains(classes(:,3),KoppClass),1));
 %BroadClass = 3;

 % This is a bodge to not rewrite regression equation code below

 Sc0 = ConstsSol(Class,1);
 Sc1 = ConstsSol(Class,2);
 Sc2 = ConstsSol(Class,3);
 Sc3 = ConstsSol(Class,4);
 Sc4 = ConstsSol(Class,5);
 Sc5 = ConstsSol(Class,6);
 Sd = ConstsSol(Class,7);
 %Sk = 0.843; %- This is not used in the Huang et al ASHRAE version of the
 %equation and looks to just have been a constant factor in the Qingyuan
 %paper - the coefficients can just be divided through by Sk

 I0 = 1367.7; % W/m^2 from
 % Use T to hold dry bulb at hour n-3 - use a wrap around on the data at
 % the beginning of the year just to make logical consistancy although
 % these hours will always be nighttime with a negative solar altitude
 % angle and so the data will not be used.
 T(1:3) = WS_data_resampled.DryBulb(8758:8760);
 T(4:8760) = WS_data_resampled.DryBulb(1:8757);

 WS_data_resampled.EstimHourlySol = zeros(8760,1);
 %WS_data_resampled.EstimHourlySol(INDEX)=
(I0*sin(WS_data_resampled.SolAltAngle(INDEX)).*(Sc0+Sc1.*WS_data_resampled.CloudCover(INDEX)./10 ...
 %    +Sc2.*((WS_data_resampled.CloudCover(INDEX)./10).^2)+Sc3.*(WS_data_resampled.DryBulb(INDEX)-T(INDEX))...
 %    +Sc4.*WS_data_resampled.RelativeHumidity(INDEX)+Sc5.*WS_data_resampled.WindSpeed(INDEX))...
 %    +Sd); %./Sk;

 WS_data_resampled.EstimHourlySol= (I0*sin(WS_data_resampled.SolAltAngle).*(Sc0+Sc1.*WS_data_resampled.CloudCover./
10 ...
     +Sc2.*((WS_data_resampled.CloudCover./10).^2)+Sc3.*(WS_data_resampled.DryBulb - T)...
     +Sc4.*WS_data_resampled.RelativeHumidity+Sc5.*WS_data_resampled.WindSpeed)...
     +Sd);
 % Limits based on extraterrestrial radiation (I0*sin(SolAltAngle)
 clear INDEX;
 INDEX = (WS_data_resampled.EstimHourlySol(:,1) < 0.1*I0*sin(WS_data_resampled.SolAltAngle(:,1)));
```

```
 WS_data_resampled.EstimHourlySol(INDEX,1) = 0.1*I0*sin(WS_data_resampled.SolAltAngle(INDEX,1));

 INDEX = (WS_data_resampled.EstimHourlySol(:,1) > 0.9*I0*sin(WS_data_resampled.SolAltAngle(:,1)));
 WS_data_resampled.EstimHourlySol(INDEX,1) = 0.9*I0*sin(WS_data_resampled.SolAltAngle(INDEX,1));

 % For extra terrestial horizontal radiation: field 11 in EPW I0*sin(SolAltAngle)

 WS_data_resampled.ExtraTHRad = I0*WS_data_resampled.SolAltAngle;
 WS_data_resampled.SolAltAngleDayly = reshape(WS_data_resampled.SolAltAngle,[24,365]);

  % Day light hours index
 INDEXDAYTIME = (WS_data_resampled.SolAltAngleDayly > 0); %Don't estimate solar radiation at night
 INDEXDAYTIME7 = (WS_data_resampled.SolAltAngleDayly > 0.3); %Cutoff at 17 degrees solar angle - for direct/diffuse
split below this angle considered 100% diffuse.
                                                              %Direct/diffuse split for cos solar angle >0.3

 % Remove values outside of daylight hours
 INDEXDAYTIMEyr = reshape(INDEXDAYTIME,8760,1);
 WS_data_resampled.ExtraTHRad(~INDEXDAYTIMEyr) = 0;

 % For extra terrestial direct normal: field 12 during daylight hours

 WS_data_resampled.ExtraTDNRad = zeros(8760,1);
 WS_data_resampled.ExtraTDNRad(INDEXDAYTIMEyr) = I0;

 % Horizontal Infrared Radiation Intensity
 % Put as missing (9999) - ESP-r, EnergyPlus & TRNSYS all have models that
 % estimate this value: https://publications.ibpsa.org/proceedings/bs/2017/papers/BS2017_569.pdf

 WS_data_resampled.HIRI = zeros(8760,1);
 WS_data_resampled.HIRI(1:8760,1) = 9999;

 INDEX = (WS_data_resampled.EstimHourlySol(:,1) < 0);
 WS_data_resampled.EstimHourlySol(INDEX,1) = 0;

 % Need to detect clear days and use ASHRAE clear sky model to smooth out
 % direct/total radiation over the day as per Gueymard & Thevenard
 % Source for tau b & d (beam and diffuse pseudo optical depths) values
 % from ASHRAE data for Mexican sites (40) and interpolated for the sites
 % that are not in the ASHRAE data set.
 %"To improve the hourly solar profiles in the IWEC2
 %weather files, the ASHRAE 2009 Clear Sky Model (Gueymard
 %and Thevenard 2013), or ACSM09 for brevity, is used to
 %recalculate the hourly distribution of global horizontal radiation
 %while leaving the total daily solar radiation computed by
 %the ZHM unchanged. For clear days, defined as those where
 %the aggregate cloud cover during daylight hours is no more
 %than 10 and the cloud cover during any daylight hour is no
 %more than 1, the hourly solar profile from ACSM09 is scaled
 %to the daily total and substituted for the hourly profile from the
 %ZHM." Development of 3012 IWEC2Weather Files for International Locations (RP-1477)
 % Huang et al. Ashrae Transactions, 2014, vol. 120, no 1, 340-355.

 % Direct Normal Solar Radiation from model by Zhang et al (2004): sigmoidal
 % Gompertz function.



 % Reshape estimated hourly solar energy data into daily sets of 24 hours by 365 days
 % Reshape all the weather data



 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.EstimHourlySol(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.EstimHourlySol(1:-RelativeToUTC-1,1);
 %WS_data_resampled.EstimDaylySol = reshape(Temp8,[24,365]);
 WS_data_resampled.EstimDaylySol = reshape(WS_data_resampled.EstimHourlySol,[24,365]);

 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.SolarTime(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.SolarTime(1:-RelativeToUTC-1,1);
 %WS_data_resampled.SolarTimeDayly = reshape(Temp8,[24,365]);
 WS_data_resampled.SolarTimeDayly = reshape(WS_data_resampled.SolarTime,[24,365]);

 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.Pressure(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.Pressure(1:-RelativeToUTC-1,1);
 %WS_data_resampled.PressureDayly = reshape(Temp8,[24,365]);
 WS_data_resampled.PressureDayly = reshape(WS_data_resampled.Pressure,[24,365]);

 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.DryBulb(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.DryBulb(1:-RelativeToUTC-1,1);
 %WS_data_resampled.DryBulbDayly = reshape(Temp8,[24,365]);
 WS_data_resampled.DryBulbDayly = reshape(WS_data_resampled.DryBulb,[24,365]);
```

```
 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.DewPoint(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.DewPoint(1:-RelativeToUTC-1,1);
 %WS_data_resampled.DewPointDayly = reshape(Temp8,[24,365]);
 WS_data_resampled.DewPointDayly = reshape(WS_data_resampled.DewPoint,[24,365]);

 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.WindDir(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.WindDir(1:-RelativeToUTC-1,1);
 %WS_data_resampled.WindDirDayly = reshape(Temp8,[24,365]);
 WS_data_resampled.WindDirDayly = reshape(WS_data_resampled.WindDir,[24,365]);

 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.WindSpeed(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.WindSpeed(1:-RelativeToUTC-1,1);
 %WS_data_resampled.WindSpeedDayly = reshape(Temp8,[24,365]);
 WS_data_resampled.WindSpeedDayly = reshape(WS_data_resampled.WindSpeed,[24,365]);

 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.CloudCover(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.CloudCover(1:-RelativeToUTC-1,1);
 %WS_data_resampled.CloudCoverDayly = reshape(Temp8,[24,365]);
 WS_data_resampled.CloudCoverDayly = reshape(WS_data_resampled.CloudCover,[24,365]);

 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.Precipit(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.Precipit(1:-RelativeToUTC-1,1);
 %WS_data_resampled.PrecipitDayly = reshape(Temp8,[24,365]);
 WS_data_resampled.PrecipitDayly = reshape(WS_data_resampled.Precipit,[24,365]);

 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.Declination(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.Declination(1:-RelativeToUTC-1,1);
 %WS_data_resampled.DeclinationDayly = reshape(Temp8,[24,365]);
 WS_data_resampled.DeclinationDayly = reshape(WS_data_resampled.Declination,[24,365]);

 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.SolAltAngle(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.SolAltAngle(1:-RelativeToUTC-1,1);
 %WS_data_resampled.SolAltAngleDayly = reshape(Temp8,[24,365]);
 %WS_data_resampled.SolAltAngleDayly = reshape(WS_data_resampled.SolAltAngle,[24,365]);

 %Temp8 = zeros(8760,1);
 %Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.HourAngle(-RelativeToUTC:8760,1);
 %Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.HourAngle(1:-RelativeToUTC-1,1);
 %WS_data_resampled.HourAngleDayly = reshape(Temp8,[24,365]);
 WS_data_resampled.HourAngleDayly = reshape(WS_data_resampled.HourAngle,[24,365]);

  % Day light hours index
 INDEXDAYTIME = (WS_data_resampled.SolAltAngleDayly > 0); %Don't estimate solar radiation at night
 INDEXDAYTIME7 = (WS_data_resampled.SolAltAngleDayly > 0.3); %Cutoff at 17 degrees solar angle - for direct/diffuse
split below this angle considered 100% diffuse.
                                                              %Direct/diffuse split for cos solar angle >0.3
 WS_data_resampled.DayAverageCldCvr = zeros(24,365);
 WS_data_resampled.DaylightCloudCover(1:24,1:365) = nan;
 WS_data_resampled.DaylightCloudCover(INDEXDAYTIME) = WS_data_resampled.CloudCoverDayly(INDEXDAYTIME);
 WS_data_resampled.DayAverageCldCvr(1,1:365) = sum(WS_data_resampled.DaylightCloudCover,'omitnan'); %Total cloud cover
during daylight hours in column 1
 WS_data_resampled.DayAverageCldCvr(2,1:365) = max(WS_data_resampled.DaylightCloudCover); %Maximum cloud cover at any
point of daylight hours for a given day in column 2.
 INDEXCLEARDAY = (WS_data_resampled.DayAverageCldCvr(1,:) < 10 & WS_data_resampled.DayAverageCldCvr(2,:) <= 1);
 WS_data_resampled.DayAverageCldCvr(3,1:365) = sum(WS_data_resampled.EstimDaylySol); %Total daily solar energy estimated
from ZHM model in column 3.

 % Interpolate values for Taub and Taud for ASHRAE clear sky model using
 % the 'nearest' parameter - it is likely there are very small difference in
 % latitude and longitude between the values for the sites from the raw
 % data source and the ASHRAE data table - for the 49 sites which ASHRAE
 % provides data, it is likely that there is a one to one correspondance
 % with raw data sites. For any other site the values of the nearest
 % available site is used.


 taub12 = zeros(12,1);
 taud12 = zeros(12,1);
 taub365 = zeros(365,1);
 taud365 = zeros(365,1);
 taub397 = zeros(397,1);
 taud397 = zeros(397,1);

 QLatLong = [Latitude,Longitude];

     k = dsearchn(LatLong,QLatLong); %Nearest point search k = dsearchn(X,XI) performs the search without using a
triangulation.
                                        %With large X and small XI, this approach is faster and uses much less memory.
```

```
    taub12 = taub(k,:);
    taud12 = taud(k,:);
    taub14(1) = taub12(12);
    taub14(2:13) = taub12;
    taub14(14) = taub12(1);
    taud14(1) = taud12(12);
    taud14(2:13) = taud12;
    taud14(14) = taud12(1);
tau14x = [1,32,63,91,122,152,183,213,244,275,315,336,366,397];

DaysOfTheYear = (1:397);
taub397 = interp1(tau14x,taub14,DaysOfTheYear,'linear');
taud397 = interp1(tau14x,taud14,DaysOfTheYear,'linear');
taub365 = taub397(12:376);
taud365 = taud397(12:376);

% Air mass from equation 16 Chpt. 14 ASHRAE Fundamentals Handbook 2021 SI

WS_data_resampled.Airmass(1:24,1:365) = zeros(24,365);
WS_data_resampled.Airmass(1:24,1:365) = nan;
WS_data_resampled.Airmass(INDEXDAYTIME) = 1.0./(sin(WS_data_resampled.SolAltAngleDayly(INDEXDAYTIME))...
    +0.50572.*((6.07995+WS_data_resampled.SolAltAngleDayly(INDEXDAYTIME).*360./(2.*pi)).^(-1.6364)));

% Clear sky solar radiation: equations 19, 20, 17 & 18 Chpt. 14 ASHRAE Fundamentals Handbook 2021 SI
% Equation 19 & 20 Air mass exponents ab and ad are correlated to taub and taud through
% the following empirical relationships:

ab = zeros(24,365);
ad = zeros(24,365);
E0 = zeros(24,365);
taub24365 = zeros(24,365);
taud24365 = zeros(24,365);
WS_data_resampled.DailyEb = zeros(24,365);
WS_data_resampled.DailyEd = zeros(24,365);
WS_data_resampled.DailyEbh = zeros(24,365);
WS_data_resampled.DailyASHRAEHoriz = zeros(24,365);
WS_data_resampled.DailyZHMmntAvg = zeros(24,12); % Averages for each hour of the day for each month

for Idaytau = 1:365
    for Ihourtau = 1:24
        ab(Ihourtau,Idaytau) = 1.454 -0.406*taub365(Idaytau) - 0.268*taud365(Idaytau) +
0.021*taub365(Idaytau)*taub365(Idaytau);
        ad(Ihourtau,Idaytau) = 0.507 -0.205*taub365(Idaytau) - 0.080*taud365(Idaytau) +
0.190*taub365(Idaytau)*taub365(Idaytau);
        % Extraterrestrial Solar Radiation Equation 4 Chpt. 14 ASHRAE
        % Fundamentals Handbook 2021 SI in W/m^2
        E0(Ihourtau,Idaytau) = 1367*(1+0.033*cos(2*pi*(Idaytau-3)/365));
        taub24365(Ihourtau,Idaytau) = taub365(Idaytau);
        taud24365(Ihourtau,Idaytau) = taud365(Idaytau);
    end
end
CLEARDAYINDEX = reshape(INDEXCLEARDAY,[365,1]);

ab(~INDEXDAYTIME)= nan;
E0(~INDEXDAYTIME)= nan;
taub24365(~INDEXDAYTIME)= nan;
taud24365(~INDEXDAYTIME)= nan;
WS_data_resampled.DailyEb(~INDEXDAYTIME)= nan;
WS_data_resampled.DailyEd(~INDEXDAYTIME)= nan;
WS_data_resampled.DailyEbh(~INDEXDAYTIME)= nan;
WS_data_resampled.DailyASHRAEHoriz(~INDEXDAYTIME)= nan;

WS_data_resampled.DailyEb(INDEXDAYTIME) = E0(INDEXDAYTIME).*exp(-
taub24365(INDEXDAYTIME).*WS_data_resampled.Airmass(INDEXDAYTIME).^(ab(INDEXDAYTIME))); %Beam
WS_data_resampled.DailyEd(INDEXDAYTIME) = E0(INDEXDAYTIME).*exp(-
taud24365(INDEXDAYTIME).*WS_data_resampled.Airmass(INDEXDAYTIME).^(ad(INDEXDAYTIME))); %Diffuse on horizontal surface

% Beam on horizontal surface

WS_data_resampled.DailyEbh(INDEXDAYTIME)= WS_data_resampled.DailyEb(INDEXDAYTIME).*cos(pi/2-
WS_data_resampled.SolAltAngleDayly(INDEXDAYTIME));

WS_data_resampled.DailyASHRAEHoriz(INDEXDAYTIME)= WS_data_resampled.DailyEbh(INDEXDAYTIME)
+WS_data_resampled.DailyEd(INDEXDAYTIME);

WS_data_resampled.DayAverageCldCvr(4,1:365) = sum(WS_data_resampled.DailyASHRAEHoriz,'omitnan'); %Total daily beam
solar energy from ASHRAE clear sky model on horizontal surface in column 4.
WS_data_resampled.DayAverageCldCvr(5,1:365) = sum(WS_data_resampled.DailyEd,'omitnan'); %Total daily diffuse solar
energy on horizontal surface from ASHRAE clear sky model in column 5.
WS_data_resampled.DayAverageCldCvr(6,1:365) = sum(WS_data_resampled.EstimDaylySol); %ZHM daily totals in column 6.
WS_data_resampled.DayAverageCldCvr(7,1:365) = WS_data_resampled.DayAverageCldCvr(6,1:365)./
WS_data_resampled.DayAverageCldCvr(4,1:365); % ZHM daily total/ASHRAE clear sky daily total
WS_data_resampled.DayAverageCldCvr(8,1:365) =
WS_data_resampled.DayAverageCldCvr(4,1:365)+WS_data_resampled.DayAverageCldCvr(5,1:365); % ASHRAE clear sky daily total
on horizontal surface in column 8
```

```
 %Clear days hourly ASHRAE scaled to ZHM daily totals
 WS_data_resampled.DailyClearScaledToZHM = nan(24,365);
 WS_data_resampled.DailyClearScaledToZHM(1:24,CLEARDAYINDEX) =
WS_data_resampled.DailyASHRAEHoriz(1:24,CLEARDAYINDEX).*WS_data_resampled.DayAverageCldCvr(7,INDEXCLEARDAY);

 % Cloudy days - needs ZHM monthly average profile (hour by hour) - then
 % scale the ASHRAE clear sky to the monthly average. Hourly difference
 % between that result and hourly monthly

 % Set month days' indecies

 INDEXJANDAYS = false(365,1);
 INDEXJANDAYS (1:31) = true;
 %INDEXmonthDAYS (1,1:31) = true;
 INDEXFEBDAYS = false(365,1);
 INDEXFEBDAYS (32:59) = true;
 %INDEXmonthDAYS (2,32:59) = true;
 INDEXMARDAYS = false(365,1);
 INDEXMARDAYS (60:90) = true;
 %INDEXmonthDAYS(3,60:90) = true;
 INDEXAPRDAYS = false(365,1);
 INDEXAPRDAYS (91:120) = true;
 %INDEXmonthDAYS (4,91:120) = true;
 INDEXMAYDAYS = false(365,1);
 INDEXMAYDAYS (121:151) = true;
 %INDEXmonthDAYS (5,121:151) = true;
 INDEXJUNDAYS = false(365,1);
 INDEXJUNDAYS (152:181) = true;
 %INDEXmonthDAYS (6,152:181) = true;
 INDEXJULDAYS = false(365,1);
 INDEXJULDAYS (182:212) = true;
 %INDEXmonthDAYS (7,182:212) = true;
 INDEXAUGDAYS = false(365,1);
 INDEXAUGDAYS (213:243) = true;
 %INDEXmonthDAYS (8,213:243) = true;
 INDEXSEPDAYS = false(365,1);
 INDEXSEPDAYS (244:273) = true;
 %INDEXmonthDAYS (9,244:273) = true;
 INDEXOCTDAYS = false(365,1);
 INDEXOCTDAYS (274:304) = true;
 %INDEXmonthDAYS (10,274:304) = true;
 INDEXNOVDAYS = false(365,1);
 INDEXNOVDAYS (305:334) = true;
 %INDEXmonthDAYS (11,305:334) = true;
 INDEXDECDAYS = false(365,1);
 INDEXDECDAYS (335:365) = true;
 %INDEXmonthDAYS (12,335:365) = true;

 CLEARDAYINDEX = reshape(INDEXCLEARDAY,[365,1]);
 WS_data_resampled.EstimDaylySol(~INDEXDAYTIME)= nan;
 WS_data_resampled.DailyZHMmntAvg = zeros(24,12);
 WS_data_resampled.DailyZHMmntAvg(:,1) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXJANDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,2) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXFEBDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,3) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXMARDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,4) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXAPRDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,5) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXMAYDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,6) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXJUNDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,7) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXJULDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,8) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXAUGDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,9) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXSEPDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,10) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXOCTDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,11) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXNOVDAYS&~CLEARDAYINDEX),2,'omitnan');
 WS_data_resampled.DailyZHMmntAvg(:,12) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXDECDAYS&~CLEARDAYINDEX),2,'omitnan');

 %Daily totals from DailyZHMmntAvg
 WS_data_resampled.DailyZHMmntAvgTot = zeros(12,1);
 for i = 1:12
     WS_data_resampled.DailyZHMmntAvgTot(i) = sum(WS_data_resampled.DailyZHMmntAvg(:,i),'omitnan');
 end
 %Scale ASHRAE clear sky to monthly average ZHM total on a horizontal
 %surface on cloudy days

 WS_data_resampled.DailyCldyZHMratioASH = nan(24,365);
 %WS_data_resampled.DailyCldyAHSminusZHMhrly = nan(24,365);
```

```
 for i = 1:24
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXJANDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(1)./
WS_data_resampled.DayAverageCldCvr(8,INDEXJANDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXFEBDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(2)./
WS_data_resampled.DayAverageCldCvr(8,INDEXFEBDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXMARDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(3)./
WS_data_resampled.DayAverageCldCvr(8,INDEXMARDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXAPRDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(4)./
WS_data_resampled.DayAverageCldCvr(8,INDEXAPRDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXMAYDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(5)./
WS_data_resampled.DayAverageCldCvr(8,INDEXMAYDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXJUNDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(6)./
WS_data_resampled.DayAverageCldCvr(8,INDEXJUNDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXJULDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(7)./
WS_data_resampled.DayAverageCldCvr(8,INDEXJULDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXAUGDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(8)./
WS_data_resampled.DayAverageCldCvr(8,INDEXAUGDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXSEPDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(9)./
WS_data_resampled.DayAverageCldCvr(8,INDEXSEPDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXOCTDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(10)./
WS_data_resampled.DayAverageCldCvr(8,INDEXOCTDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXNOVDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(11)./
WS_data_resampled.DayAverageCldCvr(8,INDEXNOVDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXDECDAYS&~CLEARDAYINDEX) = WS_data_resampled.DailyZHMmntAvgTot(12)./
WS_data_resampled.DayAverageCldCvr(8,INDEXDECDAYS&~CLEARDAYINDEX);
 end

 WS_data_resampled.DailyCldyASHScaled = nan(24,365);
 WS_data_resampled.DailyCldyASHScaled =
WS_data_resampled.DailyASHRAEHoriz(:,:).*WS_data_resampled.DailyCldyZHMratioASH(:,:);

WS_data_resampled.DailyCldyASHminusZHMhrly = nan(24,365);


  for i = 1:24
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXJANDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXJANDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,1);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXFEBDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXFEBDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,2);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXMARDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXMARDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,3);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXAPRDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXAPRDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,4);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXMAYDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXMAYDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,5);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXJUNDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXJUNDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,6);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXJULDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXJULDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,7);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXAUGDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXAUGDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,8);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXSEPDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXSEPDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,9);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXOCTDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXOCTDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,10);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXNOVDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXNOVDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,11);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXDECDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXDECDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,12);
  end
     WS_data_resampled.DailyCldySmoothed = zeros(24,365);
     WS_data_resampled.DailyCldySmoothed(:,~CLEARDAYINDEX) = WS_data_resampled.DailyCldyASHScaled(:,~CLEARDAYINDEX)
+WS_data_resampled.DailyCldyASHminusZHMhrly(:,~CLEARDAYINDEX);

 figure (1);

 set(gcf,'Position',[50,100,1500,700]);
 subplot (6,1,1);
 plttemp = reshape(WS_data_resampled.DailyCldySmoothed,[1,8760]);
 plot(plttemp(1:240));
 title ('DailyCldySmoothed');
 ax = gca;
 ax.XLim = [1 240];
 ax.YLim = [-100 max(plttemp)*1.1];



 subplot (6,1,3);
 plttemp = reshape(WS_data_resampled.EstimHourlySol,[1,8760]);

 plot(plttemp(1:240));

 title ('EstimHourlySol ZHM');
 ax = gca;
 ax.XLim = [1 240];
```

```
ax.YLim = [-100 max(plttemp)*1.1];

% Direct/diffuse partition using "Zhang et al. (2004) has been used to estimate the direct normal
% radiation using a sigmoidal Gompertz function where growth
% is slowest at the start and end of the function (Parton and Innis 1972)"
% Following "Development of 3012 IWEC2Weather Files for International
% Locations (RP-1477)" Huang et al.

A1 = zeros (24,365);
A2 = zeros (24,365);
A3 = zeros (24,365);
A4 = 2.99;
SinH = zeros (24,365);
SinH = max(sin(WS_data_resampled.SolAltAngleDayly),0.2);  %Zhang et al. (2004) "When Kt is smaller than 0.2, the direct
radiation Kn is almost zero"
%"Therefore values smaller than 0.3 were excluded in the modelling
% process."
A1(INDEXDAYTIME) = -0.1556.*SinH(INDEXDAYTIME).*SinH(INDEXDAYTIME) + 0.1028.*SinH(INDEXDAYTIME) + 1.3748;
A2(INDEXDAYTIME) = 0.7973.*SinH(INDEXDAYTIME).*SinH(INDEXDAYTIME) + 0.1509.*SinH(INDEXDAYTIME) + 3.035;
A3(INDEXDAYTIME) = 5.4307.*SinH(INDEXDAYTIME) + 7.2182;

WS_data_resampled.DailyTotalHorizontal = WS_data_resampled.DailyCldySmoothed +
max(WS_data_resampled.DailyClearScaledToZHM,0);
INDEXNANS0 = false(24,365);
INDEXNANS0 = isnan(WS_data_resampled.DailyTotalHorizontal);
WS_data_resampled.DailyTotalHorizontal(INDEXNANS0) = 0;
INDEXNEGATIVES = false(24,365);
INDEXNEGATIVES = WS_data_resampled.DailyTotalHorizontal < 0;
WS_data_resampled.DailyTotalHorizontal(INDEXNEGATIVES) = 0;


subplot (6,1,1);
plttemp = reshape(WS_data_resampled.EstimHourlySol,[1,8760]);
plot(plttemp(1:240));
title ('Global Horizontal Radiation - Raw ZHM');
ax = gca;
ax.XLim = [1 240];
ax.YLim = [-100 max(plttemp)*1.1];

% Clearness index Kt

WS_data_resampled.Kt = zeros (24,365);
WS_data_resampled.Kt = (WS_data_resampled.DailyTotalHorizontal./SinH)/I0;

WS_data_resampled.Knormal = zeros (24,365);
%WS_data_resampled.Knormal =
%(A1.*A2).^(-((A3.*A2).^(-(A4.*WS_data_resampled.Kt))));   % Incorrect
%interpretation of the formula due to lack of parenteses in the right
%place: Parton, William J., and George S. Innis. "Some graphs and their functional forms." (1972).
% https://api.mountainscholar.org/server/api/core/bitstreams/d0457fb2-8adb-41f9-9220-e13fc26be4e0/content`
%

WS_data_resampled.Knormal = A1.*(A2.^(-A3.*(A2.^(-A4.*WS_data_resampled.Kt))));

WS_data_resampled.DailyDirectNormal = zeros (24,365);
WS_data_resampled.DailyDirectNormal7  = zeros (24,365);
WS_data_resampled.DailyDirectNormal(INDEXDAYTIME) = WS_data_resampled.Knormal(INDEXDAYTIME).*I0;
WS_data_resampled.DailyDirectNormal7(INDEXDAYTIME7) = WS_data_resampled.Knormal(INDEXDAYTIME7).*I0;

% Horizontal diffuse irradiance

WS_data_resampled.DailyHorizDiffuse = WS_data_resampled.DailyTotalHorizontal -
WS_data_resampled.DailyDirectNormal7.*SinH;
% Catch values below zero
IndexNegatives = WS_data_resampled.DailyHorizDiffuse < 0;
WS_data_resampled.DailyHorizDiffuse(IndexNegatives) = 0;



subplot (6,1,2);
WS_data_resampled.DailyCldySmoothed(:,~CLEARDAYINDEX) = max(WS_data_resampled.DailyCldySmoothed(:,~CLEARDAYINDEX),0);

plttemp = reshape(WS_data_resampled.DailyTotalHorizontal,[1,8760]);
plot(plttemp(1:240));

title ('DailyTotalHorizontal');
ax = gca;
ax.XLim = [1 240];
ax.YLim = [-100 max(plttemp)*1.1];



subplot (6,1,4);

plttemp2 = reshape(WS_data_resampled.DailyTotalHorizontal,[1,8760]);
```

```
 hold;
 INDEXNANS = false(1,8760);
 INDEXNANS = isnan(plttemp2);
 plttemp2(INDEXNANS) = 0;
 plot(plttemp2(1:600));

 title ('Difference EPW global horizontal (J. Huang data - Blue line) - global horizontal this model (ZHM smoothed - Red
line)');
 ax = gca;

 ax.YLim = [-100 1400];
 hold;


 WhyOne(1:8760,1) = 1:8760;

 subplot (6,1,5);
 plttemp = reshape(WS_data_resampled.DailyDirectNormal7,[1,8760]);
 plot(plttemp);
 title ('DirectDailyNormal 7');
 ax = gca;
 ax.XLim = [1 8760];
 ax.YLim = [-100 1400];








%         Graph results


 subplot (6,1,6);
 plttemp = reshape(WS_data_resampled.DailyHorizDiffuse,[1,8760]);
 plot(plttemp);
 title ('Diffuse Horizontal Radiation');
 ax = gca;
 ax.XLim = [1 8760];
 %ax.YLim = [-100 max(plttemp)*1.1];
 ax.YLim = [-100 1400];

% Solar total on horizontal surface, Field:14 N Global Horizontal Radiation
% = reshape(WS_data_resampled.DailyTotalHorizontal,[1,8760]);

% Field:15 O Direct Normal Radiation
% = reshape(WS_data_resampled.DailyDirectNormal7,[1,8760]);

% Field:16 P Diffuse Horizontal Radiation - use the difference total minus
% direct on horizontal surface*SinH
% = reshape(WS_data_resampled.DailyHorizDiffuse,[1,8760]);

% Field:17 Q Global Horizontal Illuminance
% Following Perez et al 1990

% Atmospheric brightness factor

WS_data_resampled.DailyBrightnessFactor = WS_data_resampled.DailyHorizDiffuse.*WS_data_resampled.Airmass./E0;

% Zenit

WS_data_resampled.DailyZenit = pi/2-WS_data_resampled.SolAltAngleDayly;

% Sky clearness
% Equation 1, Perez, Richard, et al.
% "Modeling daylight availability and irradiance components from direct and global irradiance."
% Solar energy 44.5 (1990): 271-289.

% Sky clearness = [(Dh + I)/Dh + kZ^3]/[1 + kZ^3]

WS_data_resampled.DailySkyClearness = ((WS_data_resampled.DailyHorizDiffuse...
    + WS_data_resampled.DailyDirectNormal7)./WS_data_resampled.DailyHorizDiffuse...
    + 1.041.*(WS_data_resampled.DailyZenit.^3))./(1 + 1.041.*((WS_data_resampled.DailyZenit.^3)));

WS_data_resampled.PreciptableH2O = exp(0.07.*WS_data_resampled.DewPointDayly - 0.075);

% Sky clearness bins used to give values for constants a through d for equation 6 in
% Perez et al

% Sky clearness bin edges
SkyBinEdges = [0,1,2,3,4,5,6,7,20];

% Assign coefficients for equation 6 Perez et al
```

```
Aseq6 = [96.63  -0.47   11.5    -9.16;
107.54  0.79    1.79    -1.19;...
98.73   0.7     4.4     -6.95;...
92.72   0.56    8.36    -8.31;...
86.73   0.98    7.1     -10.94;...
88.34   1.39    6.06    -7.6;...
78.63   1.47    4.93    -11.37;...
99.65   1.86    -4.46   -3.15];

Aseq6Bin = discretize(WS_data_resampled.DailySkyClearness,SkyBinEdges);
Aseq6Bin(isnan(Aseq6Bin)) = 1;
Aeq6 = Aseq6(Aseq6Bin,1); Aeq6 = reshape(Aeq6,[24,365]);
Beq6 = Aseq6(Aseq6Bin,2); Beq6 = reshape(Beq6,[24,365]);
Ceq6 = Aseq6(Aseq6Bin,3); Ceq6 = reshape(Ceq6,[24,365]);
Deq6 = Aseq6(Aseq6Bin,4); Deq6 = reshape(Deq6,[24,365]);

% Equation 6 Perez et al Global Horizontal Illuminance

WS_data_resampled.GlobHorizIllum = zeros(24,365);
WS_data_resampled.GlobHorizIllum = WS_data_resampled.DailyTotalHorizontal.*(Aeq6...
    + Beq6.*WS_data_resampled.PreciptableH2O + Ceq6.*cos(WS_data_resampled.DailyZenit)...
    + Deq6.*log(WS_data_resampled.DailyBrightnessFactor));

% Diffuse Horizontal Illuminance - Equation 7 Perez et al

Aseq7 = [97.24  -0.46   12      -8.91;...
107.22  1.15    0.59    -3.95;...
104.97  2.96    -5.53   -8.77;...
102.39  5.59    -13.95  -13.9;...
100.71  5.94    -22.75  -23.74;...
106.42  3.83    -36.15  -28.83;...
141.88  1.9     -53.24  -14.03;...
152.23  0.35    -45.27  -7.98];


Aeq7 = Aseq7(Aseq6Bin,1); Aeq7 = reshape(Aeq7,[24,365]);
Beq7 = Aseq7(Aseq6Bin,2); Beq7 = reshape(Beq7,[24,365]);
Ceq7 = Aseq7(Aseq6Bin,3); Ceq7 = reshape(Ceq7,[24,365]);
Deq7 = Aseq7(Aseq6Bin,4); Deq7 = reshape(Deq7,[24,365]);

WS_data_resampled.DiffHorizIllum = NaN(24,365);
WS_data_resampled.DiffHorizIllum = WS_data_resampled.DailyHorizDiffuse.*(Aeq7...
    + Beq7.*WS_data_resampled.PreciptableH2O + Ceq7.*cos(WS_data_resampled.DailyZenit)...
    + Deq7.*reallog(WS_data_resampled.DailyBrightnessFactor));

% Field 18 Direct Normal Illuminance Equation 8 Perez et al

Aseq8 = [57.2   -4.55   -2.98   117.12;...
98.99   -3.46   -1.21   12.38;...
109.83  -4.9    -1.71   -8.81;...
110.34  -5.84   -1.99   -4.56;...
106.36  -3.97   -1.75   -6.16;...
107.19  -1.25   -1.51   -26.73;...
105.75  0.77    -1.26   -34.44;...
101.18  1.58    -1.1    -8.29];

Aeq8 = Aseq8(Aseq6Bin,1); Aeq8 = reshape(Aeq8,[24,365]);
Beq8 = Aseq8(Aseq6Bin,2); Beq8 = reshape(Beq8,[24,365]);
Ceq8 = Aseq8(Aseq6Bin,3); Ceq8 = reshape(Ceq8,[24,365]);
Deq8 = Aseq8(Aseq6Bin,4); Deq8 = reshape(Deq8,[24,365]);

WS_data_resampled.DirNormIllum = NaN(24,365);
WS_data_resampled.DirNormIllum = WS_data_resampled.DailyDirectNormal7.*...
    (Aeq8 + Beq8.*WS_data_resampled.PreciptableH2O + ...
    Ceq8.*exp(5.73.*WS_data_resampled.DailyZenit-5) + Deq8.*WS_data_resampled.DailyBrightnessFactor);

%Field:20 T Zenith Luminance in Cd/m2.
% Equation 10 Perez et al

Aseq10 = [40.86 26.77   -29.59  -45.75;...
26.58   14.73   58.46   -21.25;...
19.34   2.28    100     0.25;...
13.25   -1.39   124.79  15.66;...
14.47   -5.09   160.09  9.13;...
19.76   -3.88   154.61  -19.21;...
28.39   -9.67   151.58  -69.39;...
42.91   -19.62  130.8   -164.08];

Aeq10 = Aseq10(Aseq6Bin,1); Aeq10 = reshape(Aeq10,[24,365]);
Ceq10 = Aseq10(Aseq6Bin,2); Ceq10 = reshape(Ceq10,[24,365]);
CPreq10 = Aseq10(Aseq6Bin,3); CPreq10 = reshape(CPreq10,[24,365]);
Deq10 = Aseq10(Aseq6Bin,4); Deq10 = reshape(Deq10,[24,365]);

WS_data_resampled.ZenitIllum = zeros(24,365);
WS_data_resampled.ZenitIllum = WS_data_resampled.DailyHorizDiffuse.*(Aeq10...
    + Ceq10.*cos(WS_data_resampled.DailyZenit)...
```

```
    + CPreq10.*exp(-3.*WS_data_resampled.DailyZenit) + ...
    Deq10.*WS_data_resampled.DailyBrightnessFactor);


%Field:21 U Wind Direction : WS_data_resampled.WindDir
%This is the Wind Direction in degrees where the convention is that North=0.0, East=90.0, South=180.0, West=270.0. (Wind
direction in degrees at the time indicated. If calm, direction equals zero.) Values can range from 0 to 360. Missing
value is 999.

%Field:22 V Wind Speed : WS_data_resampled.WindSpeed
%This is the wind speed in m/sec. (Wind speed at time indicated.) Values can range from 0 to 40. Missing value is 999.

%Field:23 W Total Sky Cover
% For Total sky cover use WS_data_resampled.CloudCover
%This is the value for total sky cover (tenths of coverage). (i.e. 1 is 1/10 covered. 10 is total coverage). (Amount of
sky dome in tenths covered by clouds or obscuring phenomena at the hour indicated at the time indicated.) Minimum value
is 0; maximum value is 10; missing value is 99.


%Field:24 X Opaque Sky Cover
%This is the value for opaque sky cover (tenths of coverage). (i.e. 1 is 1/10 covered. 10 is total coverage). (Amount of
sky dome in tenths covered by clouds or obscuring phenomena that prevent observing the sky or higher cloud layers at the
time indicated.) This is not used unless the field for Horizontal Infrared Radiation Intensity is missing and then it is
used to calculate Horizontal Infrared Radiation Intensity. Minimum value is 0; maximum value is 10; missing value is 99.
% WS_data_resampled.OpaqueCover - it has been converted as is cloud cover
% into tenths as an integer.

%Field:25 Y Visibility : WS_data_resampled.Visibility
%This is the value for visibility in km. (Horizontal visibility at the time indicated.) It is not currently used in
EnergyPlus calculations. Missing value is 9999.

%Field:26 Z Ceiling Height : WS_data_resampled.Ceiling
%This is the value for ceiling height in m. (77777 is unlimited ceiling height. 88888 is cirroform ceiling.) It is not
currently used in EnergyPlus calculations. Missing value is 99999.
% Not in file therefore use cloud base - GF1 column 8 - SKY-CONDITION-OBSERVATION lowest cloud base height dimension

%Field:27 AA Present Weather Observation
% If the value of the field is 0, then the observed weather codes are taken from the following field.
% If the value of the field is 9, then ìmissingî weather is assumed.
% Since the primary use of these fields (Present Weather Observation and Present Weather Codes) is for rain/wet
surfaces,
% a missing observation field or a missing weather code implies no rain.

%Field:28 AB Present Weather Codes
%Present weather codes - these are nine digits as a single integer - 9
%indicates missing data - therefore frquently it is just 999999999
%WF_data.REM{1,1}
hours_regular = reshape(regular_hours,[8760,1]);
MetCommentsIndex = dsearchn(WS_data.HourDecimal,hours_regular);
WS_data_resampled.PresentWeather = string(WF_data.REM(MetCommentsIndex));

%Field:28 AB Precipitable Water, units mm, missing 999 : WS_data_resampled.PreciptableH2O

%Field:29 AC Aerosol Optical Depth, units thousandths, missing .999 - put
%in as missing

%Field:30 AD Snow Depth, units cm, missing 999 - put in as missing

%Field:31 AE Days Since Last Snowfall, missing 99 - put in as missing

%Field:32 AF Albedo, missing 999 - put in as missing

%Field:33 AG Liquid Precipitation Depth, units mm, missing 999
%WS_data_resampled.Precipit - this is from field 4 in AA1 and is in mm

%Field:34 AH Liquid Precipitation Quantity, units hr, missing 99 - put in as missing

%Field:35 AI allways 1

 NewFileLocation = strcat(EpwFileList(1).folder,'\');
 WF_EPWfile = [NewFileLocation,EpwFileList(end).name];
 WF_EPWdata = importdata(WF_EPWfile,',',8);

 %extract the information from the header, note due to format the

 EPWheader = WF_EPWdata.textdata(1:8,1);

 EPWdate = zeros(8760,5);
 EPWdate(1:8760,1) = WS_data_resampled.Year;
 EPWdate(1:8760,2) = WS_data_resampled.Month;
 EPWdate(1:8760,3) = WS_data_resampled.DayOfMonth;
 EPWdate(1:8760,4) = WS_data_resampled.Hour;
 EPWdate(1:8760,5) = WS_data_resampled.Minute;

EpwFileNewFileName = strcat(EpwFileNewFileName,'_',string(WS_data.Year(1,1)-2000),'.epw');
```

```
EPWData(1:8760,1) = WS_data_resampled.DryBulb;
EPWData(1:8760,2) = WS_data_resampled.DewPoint;
EPWData(1:8760,3) = WS_data_resampled.RelativeHumidity;
EPWData(1:8760,4) = WS_data_resampled.Pressure;
EPWData(1:8760,5) = WS_data_resampled.ExtraTHRad;
EPWData(1:8760,6) = WS_data_resampled.ExtraTDNRad;
EPWData(1:8760,7) = WS_data_resampled.HIRI;
EPWData(1:8760,8) = reshape(WS_data_resampled.DailyTotalHorizontal,[8760,1]);
EPWData(1:8760,9) = reshape(WS_data_resampled.DailyDirectNormal7,[8760,1]);
EPWData(1:8760,10) = reshape(WS_data_resampled.DailyHorizDiffuse,[8760,1]);
EPWData(1:8760,11) = reshape(WS_data_resampled.GlobHorizIllum,[8760,1]);
EPWData(1:8760,12) = reshape(WS_data_resampled.DirNormIllum,[8760,1]);
EPWData(1:8760,13) = reshape(WS_data_resampled.DiffHorizIllum,[8760,1]);
EPWData(1:8760,14) = reshape(WS_data_resampled.ZenitIllum,[8760,1]);
EPWData(1:8760,15) = WS_data_resampled.WindDir;
EPWData(1:8760,16) = WS_data_resampled.WindSpeed;
EPWData(1:8760,17) = WS_data_resampled.CloudCover;
EPWData(1:8760,18) = WS_data_resampled.OpaqueCover;
EPWData(1:8760,19) = WS_data_resampled.Visibility;
EPWData(1:8760,20) = WS_data_resampled.Ceiling;
%EPWData(1:8760,21) = WS_data_resampled.PresentWeather;
EPWData(1:8760,22) = reshape(WS_data_resampled.PreciptableH2O,[8760,1]);
EPWData(1:8760,23) = 0.999;
EPWData(1:8760,24) = 999;
EPWData(1:8760,25) = 99;
EPWData(1:8760,26) = 999;
EPWData(1:8760,27) = WS_data_resampled.Precipit;
EPWData(1:8760,28) = 99;
EPWPresentWeather = WS_data_resampled.PresentWeather;

EPWData( isnan(EPWData) ) = 0;

INDEXday24hr = EPWdate(1:8760,4) > 23;
EPWdate(INDEXday24hr,3) = EPWdate(INDEXday24hr,3)-1;

create_epw_from_CSV(EPWdate,EPWData,EPWPresentWeather,EPWheader,NewFileLocation,EpwFileNewFileName);

 fclose('all'); % At the end close all files

 % Check time and date at end of each month - hour 24 should be zero? or
 % day should be one less - en EPW file the last hour of the day is 24 -
 % Check day of month for hour 24.
```

```
%## Copyright (C) 2021 Christopher Heard
%##
%## This program is free software: you can redistribute it and/or modify it
%## under the terms of the GNU General Public License as published by
%## the Free Software Foundation, either version 3 of the License, or
%## (at your option) any later version.
%##
%## This program is distributed in the hope that it will be useful, but
%## WITHOUT ANY WARRANTY; without even the implied warranty of
%## MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
%## GNU General Public License for more details.
%##
%## You should have received a copy of the GNU General Public License
%## along with this program.  If not, see
%## <https://www.gnu.org/licenses/>.


%## Author: Christopher Heard
%## Created: 2021-04-15

% This script is to read in an epw file
% It extracts data from the header to find the latitude and longitude for interpolation of data from
CMIP5 model results
% from a historical file and a future file.
% The year for each month¥s data in the TMY epw file (Typical Meteorological Year - made up of real
months)
%
% Users Manual for TMY3 Data Sets
% S. Wilcox and W. Marion
% https://www.nrel.gov/docs/fy08osti/43156.pdf
%
%
% EPW data field descriptions taken from: https://bigladdersoftware.com/epx/docs/8-3/auxiliary-
programs/energyplus-weather-file-epw-data-dictionary.html

%Data Field Descriptions
%Descriptions of the fields are taken from the IWEC manual - as descriptive of what should be
contained in the data fields.
%Field:1 Year
%This is the Year of the data. Not really used in EnergyPlus. Used in the Weather Converter program
for display in audit file.
%Field:2 Month
%This is the month (1-12) for the data. Cannot be missing.
%Field:3 Day
%This is the day (dependent on month) for the data. Cannot be missing.
%Field:4 Hour
%This is the hour of the data. (1 - 24). Hour 1 is 00:01 to 01:00. Cannot be missing.
%Field:5 Minute
%This is the minute field. (1..60)
%Field:6 Data Source and Uncertainty Flags
%The data source and uncertainty flags from various formats (usually shown with each field) are
consolidated in the E/E+ EPW format. More is shown about Data Source and Uncertainty in Data Sources/
Uncertainty section later in this document.
%Field:7 Dry Bulb Temperature
%This is the dry bulb temperature in C at the time indicated. Note that this is a full numeric field
(i.e. 23.6) and not an integer representation with tenths. Valid values range from -70 C to 70 C.
Missing value for this field is 99.9.
%Field:8 Dew Point Temperature
%This is the dew point temperature in C at the time indicated. Note that this is a full numeric field
(i.e. 23.6) and not an integer representation with tenths. Valid values range from -70 C to 70 C.
Missing value for this field is 99.9.
%Field:9 Relative Humidity
%This is the Relative Humidity in percent at the time indicated. Valid values range from 0% to 110%.
Missing value for this field is 999.
%Field:10 Atmospheric Station Pressure
%This is the station pressure in Pa at the time indicated. Valid values range from 31,000 to 120,000.
(These values were chosen from the ìstandard barometric pressureî for all elevations of the World).
Missing value for this field is 999999.
%Field:11 Extraterrestrial Horizontal Radiation
%This is the Extraterrestrial Horizontal Radiation in Wh/m2. It is not currently used in EnergyPlus
calculations. It should have a minimum value of 0; missing value for this field is 9999.
%Field:12 Extraterrestrial Direct Normal Radiation
```

%This is the Extraterrestrial Direct Normal Radiation in Wh/m2. (Amount of solar radiation in Wh/m2 received on a surface normal to the rays of the sun at the top of the atmosphere during the number of minutes preceding the time indicated). It is not currently used in EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 9999.
%Field:13 Horizontal Infrared Radiation Intensity
%This is the Horizontal Infrared Radiation Intensity in Wh/m2. If it is missing, it is calculated from the Opaque Sky Cover field as shown in the following explanation. It should have a minimum value of 0; missing value for this field is 9999.
%Field:14 Global Horizontal Radiation
%This is the Global Horizontal Radiation in Wh/m2. (Total amount of direct and diffuse solar radiation in Wh/m2 received on a horizontal surface during the number of minutes preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 9999.
%Field:15 Direct Normal Radiation
%This is the Direct Normal Radiation in Wh/m2. (Amount of solar radiation in Wh/m2 received directly from the solar disk on a surface perpendicular to the sunís rays, during the number of minutes preceding the time indicated.) If the field is ìmissing (9999)î or invalid (<0), it is set to 0. Counts of such missing values are totaled and presented at the end of the runperiod.
%Field:16 Diffuse Horizontal Radiation
%This is the Diffuse Horizontal Radiation in Wh/m2. (Amount of solar radiation in Wh/m2 received from the sky (excluding the solar disk) on a horizontal surface during the number of minutes preceding the time indicated.) If the field is ìmissing (9999)î or invalid (<0), it is set to 0. Counts of such missing values are totaled and presented at the end of the runperiod.
%Field:17 Global Horizontal Illuminance
%This is the Global Horizontal Illuminance in lux. (Average total amount of direct and diffuse illuminance in hundreds of lux received on a horizontal surface during the number of minutes preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 999999 and will be considered missing of >= 999900.
%Field:18 Direct Normal Illuminance
%This is the Direct Normal Illuminance in lux. (Average amount of illuminance in hundreds of lux received directly from the solar disk on a surface perpendicular to the sunís rays, during the number of minutes preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 999999 and will be considered missing of >= 999900.
%Field:19 Diffuse Horizontal Illuminance
%This is the Diffuse Horizontal Illuminance in lux. (Average amount of illuminance in hundreds of lux received from the sky (excluding the solar disk) on a horizontal surface during the number of minutes preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 999999 and will be considered missing of >= 999900.
%Field:20 Zenith Luminance
%This is the Zenith Illuminance in Cd/m2. (Average amount of luminance at the skyís zenith in tens of Cd/m2 during the number of minutes preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 9999.
%Field:21 Wind Direction
%This is the Wind Direction in degrees where the convention is that North=0.0, East=90.0, South=180.0, West=270.0. (Wind direction in degrees at the time indicated. If calm, direction equals zero.) Values can range from 0 to 360. Missing value is 999.
%Field:22 Wind Speed
%This is the wind speed in m/sec. (Wind speed at time indicated.) Values can range from 0 to 40. Missing value is 999.
%Field:23 Total Sky Cover
%This is the value for total sky cover (tenths of coverage). (i.e. 1 is 1/10 covered. 10 is total coverage). (Amount of sky dome in tenths covered by clouds or obscuring phenomena at the hour indicated at the time indicated.) Minimum value is 0; maximum value is 10; missing value is 99.
%Field:24 Opaque Sky Cover
%This is the value for opaque sky cover (tenths of coverage). (i.e. 1 is 1/10 covered. 10 is total coverage). (Amount of sky dome in tenths covered by clouds or obscuring phenomena that prevent observing the sky or higher cloud layers at the time indicated.) This is not used unless the field for Horizontal Infrared Radiation Intensity is missing and then it is used to calculate Horizontal Infrared Radiation Intensity. Minimum value is 0; maximum value is 10; missing value is 99.
%Field:25 Visibility
%This is the value for visibility in km. (Horizontal visibility at the time indicated.) It is not currently used in EnergyPlus calculations. Missing value is 9999.
%Field:26 Ceiling Height
%This is the value for ceiling height in m. (77777 is unlimited ceiling height. 88888 is cirroform ceiling.) It is not currently used in EnergyPlus calculations. Missing value is 99999.
%Field:27 Present Weather Observation
%If the value of the field is 0, then the observed weather codes are taken from the following field. If the value of the field is 9, then ìmissingî weather is assumed. Since the primary use of these fields (Present Weather Observation and Present Weather Codes) is for rain/wet surfaces, a missing

observation field or a missing weather code implies ìno rainî.

%TMY 2 data fields

% Field 1 Year (2 digits)
% Field 2 Month
% Field 3 Day
% Field 4 Hour – Note TMY hour is that of local standard time – in EPW the time zone is indicated
% Field 5 Extraterrestrial Horizontal Radiation: (11 EPW 5 in WF_data) – Amount of solar radiation in Wh/m2 received on a horizontal surface at the top of the atmosphere during the 60 minutes preceding the hour indicated
% Field 6 Extraterrestrial Direct Normal Radiation: (12 EPW 6 in WF_data)– Amount of solar radiation in Wh/m2 received on a surface normal to the sun at the top of the atmosphere during the 60 minutes preceding the hour indicated
% Field 7 Global Horizontal Radiation: (14 EPW 8 in WR_data) – Total amount of direct and diffuse solar radiation in Wh/m2 received on a horizontal surface during the 60 minutes preceding the hour indicated
% Field 8 Direct Normal Radiation: (15 EPW 9 in WR_data) – Amount of solar radiation in Wh/m2 received within a 5.7¬∞ field of view centered on the sun during the 60 minutes preceding the hour indicated
% Field 9 Diffuse Horizontal Radiation: (16 EPW 10 in WR_data) – Amount of solar radiation in Wh/m2 received from the sky (excluding the solar disk) on a horizontal surface during the 60 minutes preceding the hour indicated
% Field 10 Global Horiz. Illuminance: (17 EPW 11 in WR_data) – Average total amount of direct and diffuse illuminance in hundreds of lux received on a horizontal surface during the 60 minutes preceding the hour indicated
% Field 11 Direct Normal Illuminance: (18 EPW 12 in WR_data) – Average amount of direct normal illuminance in hundreds of lux received within a 5.7¬∞ field of view centered on the sun during the 60 minutes preceding the hour indicated.
% Field 12 Diffuse Horiz. Illuminance: (19 EPW 13 in WR_data) – Average amount of illuminance in hundreds of lux received from the sky (excluding the solar disk) on a horizontal surface during the 60 minutes preceding the hour indicated.
% Field 13 Zenith Luminance: (20 EPW 14 in WR_data) – Average amount of luminance at the sky's zenith in tens of Cd/m2 during the 60 minutes preceding the hour indicated.
% Field 14 Total Sky Cover: (23 EPW 17 in WR_data) – Amount of sky dome in tenths covered by clouds or obscuring phenomena at the hour indicated
% Field 15 Opaque Sky Cover: (24 EPW 18 in WR_data) – Amount of sky dome in tenths covered by clouds or obscuring phenomena that prevent observing the sky or higher cloud layers at the hour indicated
% Field 16 Dry Bulb Temperature: (7 EPW in C, 1 in WR_data) – Dry bulb temperature in tenths of oC at the hour indicated -500 to 500 = -50.0 to 50.0oC
% Field 17 Dew Point Temperature: (8 EPW in C, 2 in WR_data) – Dew point temperature in tenths of oC at the hour indicated -600 to 300 = -60.0 to 30.0oC
% Field 18 Relative Humidity: (9 EPW 3 in WR_data) – Relative humidity in percent at the hour indicated
% Field 19 Atmospheric Pressure: (10 EPW in Pa, 4 in WR_data) – Atmospheric pressure at station in millibars at the hour indicated (Pa / 100)
% Field 20 Wind Direction: (21 EPW 15 in WR_data) – Wind direction in degrees at the hour indicated. (N = 0 or 360, E = 90, S = 180,W = 270). For calm winds, wind direction equals zero.
% Field 21 Wind Speed: (22 EPW in m/s, 16 in WR_data) – Wind speed in tenths of meters per second at the hour indicated.
% Field 22 Visibility: (25 EPW in km, 17 in WR_data) – Horizontal visibility in tenths of kilometers at the hour indicated.
% Field 23 Ceiling Height: (26 EPW 18 in WR_data) – Ceiling height in meters at the hour indicated.
% Field 24 Present Weather: (27 EPW 19 in WR_data) – Present weather conditions denoted by a 10-digit number.

% For Cumulative distribution functions and Weighting values for FS statistics: TMY3 weightings and parameters
% Daily max dry bulb temperature 1/20
% Daily min dry bulb temperature 1/20
% Daily mean dry bulb temperature 2/20
% Daily max dewpoint temperature 1/20
% Daily min dewpoint temperature 1/20
% Daily mean dewpoint temperatue 2/20
% Daily max wind velocity 1/20
% Daily mean wind velocity 1/20
% Daily global radiation 5/20
% Daily direct radiation 5/20

```
%Day number for first day of each month except leap years
Daynumber = [1,32,60,91,121,152,182,213,244,274,305,335,366];

%Leap = ~mod(Year,4); %This works for 2000 because it is exactly divisable
%by 400 as well - otherwise years which are divisable by 100 are not leap
%years - the data processed here doesn't extend back to 1900 or to 2100
%which are not leap years.
%    if Leap
%        Daynumber = [1,32,61,92,122,153,183,214,245,275,306,336];
%    end


% Load epw file
% Provide location of files
%file_location = 'C:\Users\usuario\Documents\Christopher\RawYearlyData\MORELOS-TLAXCALA-PUEBLA\TLAXCALA_766830\';
%file_location = 'C:\Users\usuario\Documents\Christopher\RawYearlyData\MORELOS-TLAXCALA-PUEBLA\PUEBLA_766850\';
%file_location = 'C:\Users\usuario\Documents\Christopher\RawYearlyData\BAJA-CALIF-NORTE\MEXICALI-G-SANCHEZ_760053\';
%file_location = 'C:\Users\usuario\Documents\Christopher\RawYearlyData\BAJA-CALIF-NORTE\TIJUANA-G-RODRIGUE_760013\';
file_location = 'C:\Users\usuario\Documents\Christopher\TMY3s\MEXICO-CITY_766800\';
file_list = dir([file_location,'*.epw']);
n_files = size(file_list,1);

%Set up arrays for data and dates

WS_data = zeros(n_files,8784,29);
WS_date = cell(n_files,8784,5);
WF_date_TMY = zeros(8760,5);


%run through all of the files
for ifile = 1:n_files
  %ifile=1;
    WF_file = [file_location,file_list(ifile).name];
    WF_data = importdata(WF_file,',',8);

    %extract the information from the header, note due to format the

    header = WF_data.textdata(1:8,1);
    flag_end = size(WF_data.textdata,1);
    WS_date(ifile,1:flag_end-8,1:5) = WF_data.textdata(9:flag_end,1:5); %if matlab

    %WF_date = [WF_data.data(1:8760,1:5)]; %, WF_data.textdata(9:flag_end,:)];
    %once extracted the data is all that is left to get.
    WS_data(ifile,1:flag_end-8,1:29) = WF_data.data; % If Matlab

    %WF_data = WF_data.data(:,7:end);

    %WS_data(ifile,1:flag_end-8,1:29) = WF_data(1:flag_end-8,1:29);
    Year = str2double(cell2mat(WS_date(ifile,1,1)));

    %Day number for first day of each month except leap years
    Daynumber = [1,32,60,91,121,152,182,213,244,274,305,335,366];

    Leap(ifile) = ~mod(Year,4);
    if Leap(ifile)
        Daynumber = [1,32,61,92,122,153,183,214,245,275,306,336,367];
    end

    % Data for January for each year file

    StartHour = 1;
    EndHour = (Daynumber(2)-1)*24;

    January(ifile).year = Year;

    % Load hourly data for the month

    January(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
```

```
    January(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    January(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    January(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    January(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day = 1 : Daynumber(2) - Daynumber(1) % For each day of the month the start hour is (the day
of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        January(ifile).dmxdbt(day) = max(January(ifile).dbt(StartHour:EndHour));
        January(ifile).dmidbt(day) = min(January(ifile).dbt(StartHour:EndHour));
        January(ifile).dmndbt(day) = mean(January(ifile).dbt(StartHour:EndHour));
        January(ifile).dmxdpt(day) = max(January(ifile).dpt(StartHour:EndHour));
        January(ifile).dmidpt(day) = min(January(ifile).dpt(StartHour:EndHour));
        January(ifile).dmndpt(day) = mean(January(ifile).dpt(StartHour:EndHour));
        January(ifile).dmxwnd(day) = max(January(ifile).wnd(StartHour:EndHour));
        January(ifile).dmnwnd(day) = mean(January(ifile).wnd(StartHour:EndHour));
        January(ifile).dsmgrd(day) = sum(January(ifile).grd(StartHour:EndHour));
        January(ifile).dsmdrd(day) = sum(January(ifile).drd(StartHour:EndHour));
    end



    % Data for February for each year file

    StartHour = (Daynumber(2)-1)*24+1;
    EndHour = (Daynumber(3)-1)*24;
    if Leap(ifile)
        EndHour = (Daynumber(3)-2)*24;
    end
    February(ifile).year = Year;


    % Load hourly data for the month

    February(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    February(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    February(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    February(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    February(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.
    Daynumber3 = Daynumber(3);
    if Leap(ifile)
        Daynumber3 = Daynumber(3)-1;
    end
    for day = 1 : Daynumber3 - Daynumber(2) % For each day of the month the start hour is (the day of
the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        February(ifile).dmxdbt(day) = max(February(ifile).dbt(StartHour:EndHour));
        February(ifile).dmidbt(day) = min(February(ifile).dbt(StartHour:EndHour));
        February(ifile).dmndbt(day) = mean(February(ifile).dbt(StartHour:EndHour));
        February(ifile).dmxdpt(day) = max(February(ifile).dpt(StartHour:EndHour));
        February(ifile).dmidpt(day) = min(February(ifile).dpt(StartHour:EndHour));
        February(ifile).dmndpt(day) = mean(February(ifile).dpt(StartHour:EndHour));
        February(ifile).dmxwnd(day) = max(February(ifile).wnd(StartHour:EndHour));
        February(ifile).dmnwnd(day) = mean(February(ifile).wnd(StartHour:EndHour));
        February(ifile).dsmgrd(day) = sum(February(ifile).grd(StartHour:EndHour));
        February(ifile).dsmdrd(day) = sum(February(ifile).drd(StartHour:EndHour));
    end

    % Data for March for each year file
```

```
    StartHour = (Daynumber(3)-1)*24+1;
    EndHour = (Daynumber(4)-1)*24;

    March(ifile).year = Year;


    % Load hourly data for the month

    March(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    March(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    March(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    March(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    March(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day =  1 : Daynumber(4) - Daynumber(3) % For each day of the month the start hour is (the day
of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        March(ifile).dmxdbt(day) = max(March(ifile).dbt(StartHour:EndHour));
        March(ifile).dmidbt(day) = min(March(ifile).dbt(StartHour:EndHour));
        March(ifile).dmndbt(day) = mean(March(ifile).dbt(StartHour:EndHour));
        March(ifile).dmxdpt(day) = max(March(ifile).dpt(StartHour:EndHour));
        March(ifile).dmidpt(day) = min(March(ifile).dpt(StartHour:EndHour));
        March(ifile).dmndpt(day) = mean(March(ifile).dpt(StartHour:EndHour));
        March(ifile).dmxwnd(day) = max(March(ifile).wnd(StartHour:EndHour));
        March(ifile).dmnwnd(day) = mean(March(ifile).wnd(StartHour:EndHour));
        March(ifile).dsmgrd(day) = sum(March(ifile).grd(StartHour:EndHour));
        March(ifile).dsmdrd(day) = sum(March(ifile).drd(StartHour:EndHour));
    end

    % Data for April for each year file

    StartHour = (Daynumber(4)-1)*24+1;
    EndHour = (Daynumber(5)-1)*24;

    April(ifile).year = Year;

    % Load hourly data for the month

    April(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    April(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    April(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    April(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    April(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day =  1 : Daynumber(5) - Daynumber(4) % For each day of the month the start hour is (the day
of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        April(ifile).dmxdbt(day) = max(April(ifile).dbt(StartHour:EndHour));
        April(ifile).dmidbt(day) = min(April(ifile).dbt(StartHour:EndHour));
        April(ifile).dmndbt(day) = mean(April(ifile).dbt(StartHour:EndHour));
        April(ifile).dmxdpt(day) = max(April(ifile).dpt(StartHour:EndHour));
        April(ifile).dmidpt(day) = min(April(ifile).dpt(StartHour:EndHour));
        April(ifile).dmndpt(day) = mean(April(ifile).dpt(StartHour:EndHour));
        April(ifile).dmxwnd(day) = max(April(ifile).wnd(StartHour:EndHour));
        April(ifile).dmnwnd(day) = mean(April(ifile).wnd(StartHour:EndHour));
        April(ifile).dsmgrd(day) = sum(April(ifile).grd(StartHour:EndHour));
        April(ifile).dsmdrd(day) = sum(April(ifile).drd(StartHour:EndHour));
    end

    % Data for May for each year file
```

```
    StartHour = (Daynumber(5)-1)*24+1;
    EndHour = (Daynumber(6)-1)*24;

    May(ifile).year = Year;

    % Load hourly data for the month

    May(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    May(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    May(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    May(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    May(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day =  1 : Daynumber(6) - Daynumber(5) % For each day of the month the start hour is (the day
of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        May(ifile).dmxdbt(day) = max(May(ifile).dbt(StartHour:EndHour));
        May(ifile).dmidbt(day) = min(May(ifile).dbt(StartHour:EndHour));
        May(ifile).dmndbt(day) = mean(May(ifile).dbt(StartHour:EndHour));
        May(ifile).dmxdpt(day) = max(May(ifile).dpt(StartHour:EndHour));
        May(ifile).dmidpt(day) = min(May(ifile).dpt(StartHour:EndHour));
        May(ifile).dmndpt(day) = mean(May(ifile).dpt(StartHour:EndHour));
        May(ifile).dmxwnd(day) = max(May(ifile).wnd(StartHour:EndHour));
        May(ifile).dmnwnd(day) = mean(May(ifile).wnd(StartHour:EndHour));
        May(ifile).dsmgrd(day) = sum(May(ifile).grd(StartHour:EndHour));
        May(ifile).dsmdrd(day) = sum(May(ifile).drd(StartHour:EndHour));
    end

    % Data for June for each year file

    StartHour = (Daynumber(6)-1)*24+1;
    EndHour = ((Daynumber(7)-1)*24);

    June(ifile).year = Year;

    % Load hourly data for the month

    June(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    June(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    June(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    June(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    June(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day =  1 : Daynumber(7) - Daynumber(6) % For each day of the month the start hour is (the day
of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        June(ifile).dmxdbt(day) = max(June(ifile).dbt(StartHour:EndHour));
        June(ifile).dmidbt(day) = min(June(ifile).dbt(StartHour:EndHour));
        June(ifile).dmndbt(day) = mean(June(ifile).dbt(StartHour:EndHour));
        June(ifile).dmxdpt(day) = max(June(ifile).dpt(StartHour:EndHour));
        June(ifile).dmidpt(day) = min(June(ifile).dpt(StartHour:EndHour));
        June(ifile).dmndpt(day) = mean(June(ifile).dpt(StartHour:EndHour));
        June(ifile).dmxwnd(day) = max(June(ifile).wnd(StartHour:EndHour));
        June(ifile).dmnwnd(day) = mean(June(ifile).wnd(StartHour:EndHour));
        June(ifile).dsmgrd(day) = sum(June(ifile).grd(StartHour:EndHour));
        June(ifile).dsmdrd(day) = sum(June(ifile).drd(StartHour:EndHour));
    end

    % Data for July for each year file
```

```
    StartHour = (Daynumber(7)-1)*24+1;
    EndHour = ((Daynumber(8)-1)*24);

    July(ifile).year = Year;

    % Load hourly data for the month

    July(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    July(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    July(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    July(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    July(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day =  1 : Daynumber(8) - Daynumber(7) % For each day of the month the start hour is (the day
of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        July(ifile).dmxdbt(day) = max(July(ifile).dbt(StartHour:EndHour));
        July(ifile).dmidbt(day) = min(July(ifile).dbt(StartHour:EndHour));
        July(ifile).dmndbt(day) = mean(July(ifile).dbt(StartHour:EndHour));
        July(ifile).dmxdpt(day) = max(July(ifile).dpt(StartHour:EndHour));
        July(ifile).dmidpt(day) = min(July(ifile).dpt(StartHour:EndHour));
        July(ifile).dmndpt(day) = mean(July(ifile).dpt(StartHour:EndHour));
        July(ifile).dmxwnd(day) = max(July(ifile).wnd(StartHour:EndHour));
        July(ifile).dmnwnd(day) = mean(July(ifile).wnd(StartHour:EndHour));
        July(ifile).dsmgrd(day) = sum(July(ifile).grd(StartHour:EndHour));
        July(ifile).dsmdrd(day) = sum(July(ifile).drd(StartHour:EndHour));
    end

    % Data for August for each year file

    StartHour = (Daynumber(8)-1)*24+1;
    EndHour = ((Daynumber(9)-1)*24);

    August(ifile).year = Year;

    % Load hourly data for the month

    August(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    August(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    August(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    August(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    August(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day =  1 : Daynumber(9) - Daynumber(8) % For each day of the month the start hour is (the day
of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        August(ifile).dmxdbt(day) = max(August(ifile).dbt(StartHour:EndHour));
        August(ifile).dmidbt(day) = min(August(ifile).dbt(StartHour:EndHour));
        August(ifile).dmndbt(day) = mean(August(ifile).dbt(StartHour:EndHour));
        August(ifile).dmxdpt(day) = max(August(ifile).dpt(StartHour:EndHour));
        August(ifile).dmidpt(day) = min(August(ifile).dpt(StartHour:EndHour));
        August(ifile).dmndpt(day) = mean(August(ifile).dpt(StartHour:EndHour));
        August(ifile).dmxwnd(day) = max(August(ifile).wnd(StartHour:EndHour));
        August(ifile).dmnwnd(day) = mean(August(ifile).wnd(StartHour:EndHour));
        August(ifile).dsmgrd(day) = sum(August(ifile).grd(StartHour:EndHour));
        August(ifile).dsmdrd(day) = sum(August(ifile).drd(StartHour:EndHour));
    end

    % Data for September for each year file
```

```
    StartHour = (Daynumber(9)-1)*24+1;
    EndHour = ((Daynumber(10)-1)*24);

    September(ifile).year = Year;

    % Load hourly data for the month

    September(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    September(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    September(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    September(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    September(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day =  1 : Daynumber(10) - Daynumber(9) % For each day of the month the start hour is (the
day of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        September(ifile).dmxdbt(day) = max(September(ifile).dbt(StartHour:EndHour));
        September(ifile).dmidbt(day) = min(September(ifile).dbt(StartHour:EndHour));
        September(ifile).dmndbt(day) = mean(September(ifile).dbt(StartHour:EndHour));
        September(ifile).dmxdpt(day) = max(September(ifile).dpt(StartHour:EndHour));
        September(ifile).dmidpt(day) = min(September(ifile).dpt(StartHour:EndHour));
        September(ifile).dmndpt(day) = mean(September(ifile).dpt(StartHour:EndHour));
        September(ifile).dmxwnd(day) = max(September(ifile).wnd(StartHour:EndHour));
        September(ifile).dmnwnd(day) = mean(September(ifile).wnd(StartHour:EndHour));
        September(ifile).dsmgrd(day) = sum(September(ifile).grd(StartHour:EndHour));
        September(ifile).dsmdrd(day) = sum(September(ifile).drd(StartHour:EndHour));
    end

    % Data for October for each year file

    StartHour = (Daynumber(10)-1)*24+1;
    EndHour = ((Daynumber(11)-1)*24);

    October(ifile).year = Year;

    % Load hourly data for the month

    October(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    October(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    October(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    October(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    October(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day =  1 : Daynumber(11) - Daynumber(10) % For each day of the month the start hour is (the
day of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        October(ifile).dmxdbt(day) = max(October(ifile).dbt(StartHour:EndHour));
        October(ifile).dmidbt(day) = min(October(ifile).dbt(StartHour:EndHour));
        October(ifile).dmndbt(day) = mean(October(ifile).dbt(StartHour:EndHour));
        October(ifile).dmxdpt(day) = max(October(ifile).dpt(StartHour:EndHour));
        October(ifile).dmidpt(day) = min(October(ifile).dpt(StartHour:EndHour));
        October(ifile).dmndpt(day) = mean(October(ifile).dpt(StartHour:EndHour));
        October(ifile).dmxwnd(day) = max(October(ifile).wnd(StartHour:EndHour));
        October(ifile).dmnwnd(day) = mean(October(ifile).wnd(StartHour:EndHour));
        October(ifile).dsmgrd(day) = sum(October(ifile).grd(StartHour:EndHour));
        October(ifile).dsmdrd(day) = sum(October(ifile).drd(StartHour:EndHour));
    end

    % Data for November for each year file
```

```
    StartHour = (Daynumber(11)-1)*24+1;
    EndHour = ((Daynumber(12)-1)*24);

    November(ifile).year = Year;

    % Load hourly data for the month

    November(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    November(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    November(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    November(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    November(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day =  1 : Daynumber(12) - Daynumber(11) % For each day of the month the start hour is (the
day of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        November(ifile).dmxdbt(day) = max(November(ifile).dbt(StartHour:EndHour));
        November(ifile).dmidbt(day) = min(November(ifile).dbt(StartHour:EndHour));
        November(ifile).dmndbt(day) = mean(November(ifile).dbt(StartHour:EndHour));
        November(ifile).dmxdpt(day) = max(November(ifile).dpt(StartHour:EndHour));
        November(ifile).dmidpt(day) = min(November(ifile).dpt(StartHour:EndHour));
        November(ifile).dmndpt(day) = mean(November(ifile).dpt(StartHour:EndHour));
        November(ifile).dmxwnd(day) = max(November(ifile).wnd(StartHour:EndHour));
        November(ifile).dmnwnd(day) = mean(November(ifile).wnd(StartHour:EndHour));
        November(ifile).dsmgrd(day) = sum(November(ifile).grd(StartHour:EndHour));
        November(ifile).dsmdrd(day) = sum(November(ifile).drd(StartHour:EndHour));
    end

    % Data for December for each year file

    StartHour = (Daynumber(12)-1)*24+1;
    EndHour = ((Daynumber(13)-1)*24);

    December(ifile).year = Year;

    % Load hourly data for the month

    December(ifile).dbt = WS_data(ifile,StartHour:EndHour,1);
    December(ifile).dpt = WS_data(ifile,StartHour:EndHour,2);
    December(ifile).wnd = WS_data(ifile,StartHour:EndHour,16);
    December(ifile).grd = WS_data(ifile,StartHour:EndHour,8);
    December(ifile).drd = WS_data(ifile,StartHour:EndHour,9);

    % Produce daily data: Max. dry bulb, Mean dry bulb, Max dew point, Mean
    % dew point, Max wind speed, Mean wind speed, Total global radiation
    % and Total direct radiation.

    for day =  1 : Daynumber(13) - Daynumber(12) % For each day of the month the start hour is (the
day of the month minus one) times 24 + 1 and the end hour is the start hour plus 23
        StartHour = (day -1)*24 + 1;
        EndHour = StartHour + 23;

        December(ifile).dmxdbt(day) = max(December(ifile).dbt(StartHour:EndHour));
        December(ifile).dmidbt(day) = min(December(ifile).dbt(StartHour:EndHour));
        December(ifile).dmndbt(day) = mean(December(ifile).dbt(StartHour:EndHour));
        December(ifile).dmxdpt(day) = max(December(ifile).dpt(StartHour:EndHour));
        December(ifile).dmidpt(day) = min(December(ifile).dpt(StartHour:EndHour));
        December(ifile).dmndpt(day) = mean(December(ifile).dpt(StartHour:EndHour));
        December(ifile).dmxwnd(day) = max(December(ifile).wnd(StartHour:EndHour));
        December(ifile).dmnwnd(day) = mean(December(ifile).wnd(StartHour:EndHour));
        December(ifile).dsmgrd(day) = sum(December(ifile).grd(StartHour:EndHour));
        December(ifile).dsmdrd(day) = sum(December(ifile).drd(StartHour:EndHour));
    end
```

```
end

% for daily maximum dry bulb, mean dry bulb, maximum dew point, mean dew point, maximum wind speed,
mean wind speed, global solar, direct solar
Binwidth = 0.1;
LwrBinLimit = -20;
UprBinLimit = 80;
NumberOfBins = (UprBinLimit - LwrBinLimit)/Binwidth;

Januarypersistancestats = MonthStats (January,n_files,31,1);

Februarypersistancestats = MonthStats (February,n_files,28,1);

Marchpersistancestats = MonthStats (March,n_files,31,1);

Aprilpersistancestats = MonthStats (April,n_files,30,1);

Maypersistancestats = MonthStats (May,n_files,31,1);

Junepersistancestats = MonthStats (June,n_files,30,1);

Julypersistancestats = MonthStats (July,n_files,31,1);

Augustpersistancestats = MonthStats (August,n_files,31,1);

Septemberpersistancestats = MonthStats (September,n_files,30,1);

Octoberpersistancestats = MonthStats (October,n_files,31,1);

Novemberpersistancestats = MonthStats (November,n_files,30,1);

Decemberpersistancestats = MonthStats (December,n_files,31,1);

% For each best month-year of each month of the year the main parameters are graphed to allow
eyeballing as a quality check.
% Parameters to be inspected: Dry bulb temperature, Dew point, Air
% pressure, Wind speed and direction, Global solar radiantion, Direct solar radiation

%
% Should plot the data for the month from the selected year and then ask if
% it is OK and if it is not then plot the next best and so on until an
% acceptable one is found from those available. It should then register
% which one is the acceptable one for incorporation into the TMY file.
%
% January
% Need to find file number that corresponds to each best year

StartHour = 1;
EndHour = (Daynumber(2)-1)*24;
HourArray = 1:744;
BestJanuary = -1;

for ibest = 1 : size(Januarypersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestJanuary < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Januarypersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
```

```
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end

         if IsOK == 's'
            BestJanuary = [Januarypersistancestats.bestyears(ibest),ifile];
         end
        end
       end


   end

end

% February
% Need to find file number that corresponds to each best year

StartHour = (Daynumber(2)-1)*24+1;
    EndHour = (Daynumber(3)-1)*24;
    if Leap(ifile)
        EndHour = (Daynumber(3)-2)*24;
    end
HourArray = 1:EndHour-StartHour+1;
BestFebruary = -1;

for ibest = 1 : size(Februarypersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestFebruary < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Februarypersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
```

```
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end

         if IsOK == 's'
            BestFebruary = [Februarypersistancestats.bestyears(ibest),ifile];
         end
        end
       end


   end

end

% March
% Need to find file number that corresponds to each best year

StartHour = (Daynumber(3)-1)*24+1;
EndHour = (Daynumber(4)-1)*24;
HourArray = 1:EndHour-StartHour+1;
BestMarch = -1;

for ibest = 1 : size(Marchpersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestMarch < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Marchpersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
```

```
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end

         if IsOK == 's'
            BestMarch = [Marchpersistancestats.bestyears(ibest),ifile];
         end
        end
       end


   end

end

% April
% Need to find file number that corresponds to each best year

StartHour = (Daynumber(4)-1)*24+1;
EndHour = (Daynumber(5)-1)*24;
HourArray = 1:EndHour-StartHour+1;
BestApril = -1;

for ibest = 1 : size(Aprilpersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestApril < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Aprilpersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
```

```
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end

         if IsOK == 's'
            BestApril = [Aprilpersistancestats.bestyears(ibest),ifile];
         end
        end
       end


   end

end

% May
% Need to find file number that corresponds to each best year

StartHour = (Daynumber(5)-1)*24+1;
EndHour = (Daynumber(6)-1)*24;
HourArray = 1:EndHour-StartHour+1;
BestMay = -1;

for ibest = 1 : size(Maypersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestMay < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Maypersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
```

```
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end

         if IsOK == 's'
            BestMay = [Maypersistancestats.bestyears(ibest),ifile];
         end
        end
       end


   end

end

% June
% Need to find file number that corresponds to each best year

StartHour = (Daynumber(6)-1)*24+1;
EndHour = ((Daynumber(7)-1)*24);
HourArray = 1:EndHour-StartHour+1;
BestJune = -1;

for ibest = 1 : size(Junepersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestJune < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Junepersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');
```

```
          IsOK = input('Los datos son buenos? s/n [n]: ', 's');
          if isempty(IsOK)
             IsOK = 'n';
          end

          if IsOK == 's'
             BestJune = [Junepersistancestats.bestyears(ibest),ifile];
          end
         end
        end


    end

end

% July
% Need to find file number that corresponds to each best year

StartHour = (Daynumber(7)-1)*24+1;
EndHour = ((Daynumber(8)-1)*24);
HourArray = 1:EndHour-StartHour+1;
BestJuly = -1;

for ibest = 1 : size(Julypersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestJuly < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Julypersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end
```

```
         if IsOK == 's'
            BestJuly = [Julypersistancestats.bestyears(ibest),ifile];
         end
        end
       end


   end

end

% August
% Need to find file number that corresponds to each best year

StartHour = (Daynumber(8)-1)*24+1;
EndHour = ((Daynumber(9)-1)*24);
HourArray = 1:EndHour-StartHour+1;
BestAugust = -1;

for ibest = 1 : size(Augustpersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestAugust < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Augustpersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end

         if IsOK == 's'
            BestAugust = [Augustpersistancestats.bestyears(ibest),ifile];
         end
        end
```

```
        end


    end

end

% September
% Need to find file number that corresponds to each best year

StartHour = (Daynumber(9)-1)*24+1;
EndHour = ((Daynumber(10)-1)*24);
HourArray = 1:EndHour-StartHour+1;
BestSeptember = -1;

for ibest = 1 : size(Septemberpersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestSeptember < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Septemberpersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end

         if IsOK == 's'
            BestSeptember = [Septemberpersistancestats.bestyears(ibest),ifile];
         end
        end
       end


   end
```

```
end

% October
% Need to find file number that corresponds to each best year

StartHour = (Daynumber(10)-1)*24+1;
EndHour = ((Daynumber(11)-1)*24);
HourArray = 1:EndHour-StartHour+1;
BestOctober = -1;

for ibest = 1 : size(Octoberpersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestOctober < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Octoberpersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end

         if IsOK == 's'
            BestOctober = [Octoberpersistancestats.bestyears(ibest),ifile];
         end
        end
       end


   end

end

% November
% Need to find file number that corresponds to each best year
```

```
StartHour = (Daynumber(11)-1)*24+1;
EndHour = ((Daynumber(12)-1)*24);
HourArray = 1:EndHour-StartHour+1;
BestNovember = -1;

for ibest = 1 : size(Novemberpersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestNovember < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Novemberpersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end

         if IsOK == 's'
            BestNovember = [Novemberpersistancestats.bestyears(ibest),ifile];
         end
        end
       end


   end

end

% December
% Need to find file number that corresponds to each best year

StartHour = (Daynumber(12)-1)*24+1;
EndHour = ((Daynumber(13)-1)*24);
HourArray = 1:EndHour-StartHour+1;
BestDecember = -1;
```

```
for ibest = 1 : size(Decemberpersistancestats.bestyears)
   % The variables to be graphed are based on the data used in ESP-r - i.e.
   % Dry bulb, horizontal solar total, normal solar direct, wind speed,
   % wind direction, relative humidity and atmospheric pressure - these are
   % the data that ESP-r uses in its climate data files. At present (July
   % 2021)atmospheric pressure is not used in ESP-r but it may well be used
   % in the future.
   for ifile = 1:n_files
       if BestDecember < 0
        if str2double(cell2mat(WS_date(ifile,1,1))) ==  Decemberpersistancestats.bestyears(ibest)
         DrybulbPlt = WS_data(ifile,StartHour:EndHour,1);
         DHsolPlt = WS_data(ifile,StartHour:EndHour,8);
         DRNsolPlt = WS_data(ifile,StartHour:EndHour,9);
         WndSpdPlt = WS_data(ifile,StartHour:EndHour,16);
         WindDirPlt = WS_data(ifile,StartHour:EndHour,15);
         RHPlt = WS_data(ifile,StartHour:EndHour,3);
         PressurePlt = WS_data(ifile,StartHour:EndHour,4);


         figure(1);
         set(gcf,'Position',[50,50,2500,1000]);
         subplot (7,1,1);
         plot(HourArray,DrybulbPlt,'b-');
         ylabel('Dryblb');
         subplot (7,1,2);
         plot(HourArray,DHsolPlt,'b-');
         ylabel('HorSol');
         subplot (7,1,3);
         plot(HourArray,DRNsolPlt,'b-');
         ylabel('DirSol');
         subplot (7,1,4);
         plot(HourArray,WndSpdPlt,'b-');
         ylabel('WndSpd');
         subplot (7,1,5);
         plot(HourArray,WindDirPlt,'b-');
         ylabel('WndDir');
         subplot (7,1,6);
         plot(HourArray,RHPlt,'b-');
         ylabel('RH');
         subplot (7,1,7);
         plot(HourArray,PressurePlt,'b-');
         ylabel('Press');

         IsOK = input('Los datos son buenos? s/n [n]: ', 's');
         if isempty(IsOK)
            IsOK = 'n';
         end

         if IsOK == 's'
            BestDecember = [Decemberpersistancestats.bestyears(ibest),ifile];
         end
        end
       end


   end

end

%
% Users Manual for TMY3 Data Sets
% S. Wilcox and W. Marion
% https://www.nrel.gov/docs/fy08osti/43156.pdf
%
% "We used linear curve-fitting techniques to remove discontinuities created
% by concatenating months from different years to form the TMY2 and TMY3 data
% sets. These techniques were applied for 6 hours on each side of the month
% interfaces for dry bulb temperature, dew point temperature, wind speed,
% wind direction, atmospheric pressure, and precipitable water. Relative
% humidity for 6 hours on each side of the month interfaces was calculated
% using psychometric relationships (ASHRAE 1993) and curve-fitted values of
% dry bulb temperature and dew point temperature."
```

```
%

% Intialize TMY3DataYear
TMY3DataYear = zeros(8784,29);
if ~mod(str2double(cell2mat(WS_date(1,1,1))),4) %is a leap year then use next file
    WF_date_TMY(1:8760,2:5) = str2double([WS_date(2,1:8760,2:5)]); %[b{:}])
else
    WF_date_TMY(1:8760,2:5) = str2double([WS_date(1,1:8760,2:5)]);
end

% January added to TMY data array
%
StartHour = 1;
EndHour = (Daynumber(2)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestJanuary(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestJanuary(1);

% February added to TMY data array

StartHour = (Daynumber(2)-1)*24+1;
    EndHour = (Daynumber(3)-1)*24;
    if Leap(ifile)
        EndHour = (Daynumber(3)-2)*24;
    end

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestFebruary(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestFebruary(1);

% Linear interpolation data splicing from minus six hours to plus six hours
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    %InterpParms = ['slope',num2str(slope1,4),' const',num2str(const1,4)];
    %disp (InterpParms)
    for ihour = StartHour-6 : StartHour+5
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end

%InterpVals = [num2str(TMY3DataYear(StartHour-10:StartHour+10,1))];
%disp(InterpVals)



% March added to TMY data array

StartHour = (Daynumber(3)-1)*24+1;
EndHour = (Daynumber(4)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestMarch(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestMarch(1);

% Linear interpolation data splicing from minus six hours to plus six hours
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    for ihour = StartHour-6 : StartHour+5
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end

% April added to TMY data array

StartHour = (Daynumber(4)-1)*24+1;
EndHour = (Daynumber(5)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestApril(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestApril(1);

% Linear interpolation data splicing from minus six hours to plus six hours
```

```
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    for ihour = StartHour-6 : StartHour+5
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end

% May added to TMY data array

StartHour = (Daynumber(5)-1)*24+1;
EndHour = (Daynumber(6)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestMay(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestMay(1);

% Linear interpolation data splicing from minus six hours to plus six hours
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    for ihour = StartHour-6 : StartHour+5
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end

% June added to TMY data array

StartHour = (Daynumber(6)-1)*24+1;
EndHour = (Daynumber(7)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestJune(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestJune(1);

% Linear interpolation data splicing from minus six hours to plus six hours
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    for ihour = StartHour-6 : StartHour+5
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end

% July added to TMY data array

StartHour = (Daynumber(7)-1)*24+1;
EndHour = (Daynumber(8)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestJuly(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestJuly(1);

% Linear interpolation data splicing from minus six hours to plus six hours
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    for ihour = StartHour-6 : StartHour+5
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end

% August added to TMY data array

StartHour = (Daynumber(8)-1)*24+1;
EndHour = (Daynumber(9)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestAugust(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestAugust(1);

% Linear interpolation data splicing from minus six hours to plus six hours
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    for ihour = StartHour-6 : StartHour+5
```

```
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end

% September added to TMY data array

StartHour = (Daynumber(9)-1)*24+1;
EndHour = (Daynumber(10)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestSeptember(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestSeptember(1);

% Linear interpolation data splicing from minus six hours to plus six hours
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    for ihour = StartHour-6 : StartHour+5
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end

% October added to TMY data array

StartHour = (Daynumber(10)-1)*24+1;
EndHour = (Daynumber(11)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestOctober(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestOctober(1);

% Linear interpolation data splicing from minus six hours to plus six hours
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    for ihour = StartHour-6 : StartHour+5
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end

% November added to TMY data array

StartHour = (Daynumber(11)-1)*24+1;
EndHour = (Daynumber(12)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestNovember(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestNovember(1);

% Linear interpolation data splicing from minus six hours to plus six hours
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    for ihour = StartHour-6 : StartHour+5
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end

% December added to TMY data array

StartHour = (Daynumber(12)-1)*24+1;
EndHour = (Daynumber(13)-1)*24;

TMY3DataYear(StartHour : EndHour,:) = WS_data(BestDecember(2),StartHour : EndHour,:);
WF_date_TMY(StartHour : EndHour,1) = BestDecember(1);

% Linear interpolation data splicing from minus six hours to plus six hours
for idata = 1:29
    slope1 = (TMY3DataYear(StartHour-6,idata) - TMY3DataYear(StartHour+5,idata))/(-11.0);
    const1 = TMY3DataYear(StartHour-6,idata) - slope1*(StartHour-6);
    for ihour = StartHour-6 : StartHour+5
    TMY3DataYear(ihour,idata) = const1 +  slope1*ihour;
    end
end
```

```
% Set up EPW eight header lines
% 1st line: Location data - copy from header(1)
% 2nd line: Design conditions - initially none "DESIGN CONDITIONS,0" -
% ESP-r generates its own design conditions when EPW file is imported in
% weather processor.
% 3rd line: TYPICAL/EXTREME PERIODS - for the moment copy from header(3)
% Idem w.r.t. ESP-r weather processing.
% 4th line: ground temperatures - for the moment leave as from header(4) -
% at a later date - use the model from "Heat Transfer Model to Predict
% Temperature Distribution in the Ground"
% Barbara Larwa, Energies 2019, 12, 25; doi:10.3390/en12010025
% 5th line: holidays and daylight savings
%HOLIDAYS/DAYLIGHT SAVING,
%     A1, \field LeapYear Observed
%      \type choice
%      \key Yes
%      \key No
%      \note Yes if Leap Year will be observed for this file - not the case
%      \note No if Leap Year days (29 Feb) should be ignored in this file
%     A2, \field Daylight Saving Start Day
%     A3, \field Daylight Saving End Day
%     N1, \field Number of Holidays (essentially unlimited)
%     A4, \field Holiday 1 Name
%     A5, \field Holiday 1 Day
%     \note repeat above two fields until Number of Holidays is reached
% -- etc to # of Holidays entered
% using: "HOLIDAYS/DAYLIGHT SAVINGS,No,0,0,0" - ESP-r allows use of
% holidays and daylight saving separately from met data.
% 6th & 7th line: comments
% 8th line: data periods "DATA PERIODS,1,1,Data,Sunday, 1/ 1,12/31" - from
% header(8)


create_epw_TMY3(WF_date_TMY,TMY3DataYear,header,file_location,file_list(1).name)
```

```
% Author: Christopher Heard
% Created: 2021-03-17

% Reads in EPW file and uses the cloudcover etc. to apply the methods of
% "Development of 3012 IWEC2Weather Files for International Locations
% (RP-1477)"
% by Huang et al
%

% This script is to read in an epw file
% It extracts data from the header to find the latitude and longitude for interpolation of data from
CMIP5 model results
% from a historical file and a future file.
% The year for each month¥s data in the TMY epw file (Typical Meteorological Year - made up of real
months)
%
% EPW data field descriptions taken from: https://bigladdersoftware.com/epx/docs/8-3/auxiliary-
programs/energyplus-weather-file-epw-data-dictionary.html

%Data Field Descriptions
%Descriptions of the fields are taken from the IWEC manual - as descriptive of what should be
contained in the data fields.
%Field:1 Year
%This is the Year of the data. Not really used in EnergyPlus. Used in the Weather Converter program
for display in audit file.
%Field:2 Month
%This is the month (1-12) for the data. Cannot be missing.
%Field:3 Day
%This is the day (dependent on month) for the data. Cannot be missing.
%Field:4 Hour
%This is the hour of the data. (1 - 24). Hour 1 is 00:01 to 01:00. Cannot be missing.
%Field:5 Minute
%This is the minute field. (1..60)
%Field:6 Data Source and Uncertainty Flags
%The data source and uncertainty flags from various formats (usually shown with each field) are
consolidated in the E/E+ EPW format. More is shown about Data Source and Uncertainty in Data Sources/
Uncertainty section later in this document.
%Field:7 Dry Bulb Temperature
%This is the dry bulb temperature in C at the time indicated. Note that this is a full numeric field
(i.e. 23.6) and not an integer representation with tenths. Valid values range from -70 C to 70 C.
Missing value for this field is 99.9.
%Field:8 Dew Point Temperature
%This is the dew point temperature in C at the time indicated. Note that this is a full numeric field
(i.e. 23.6) and not an integer representation with tenths. Valid values range from -70 C to 70 C.
Missing value for this field is 99.9.
%Field:9 Relative Humidity
%This is the Relative Humidity in percent at the time indicated. Valid values range from 0% to 110%.
Missing value for this field is 999.
%Field:10 Atmospheric Station Pressure
%This is the station pressure in Pa at the time indicated. Valid values range from 31,000 to 120,000.
(These values were chosen from the ìstandard barometric pressureî for all elevations of the World).
Missing value for this field is 999999.
%Field:11 Extraterrestrial Horizontal Radiation
%This is the Extraterrestrial Horizontal Radiation in Wh/m2. It is not currently used in EnergyPlus
calculations. It should have a minimum value of 0; missing value for this field is 9999.
%Field:12 Extraterrestrial Direct Normal Radiation
%This is the Extraterrestrial Direct Normal Radiation in Wh/m2. (Amount of solar radiation in Wh/m2
received on a surface normal to the rays of the sun at the top of the atmosphere during the number of
minutes preceding the time indicated). It is not currently used in EnergyPlus calculations. It should
have a minimum value of 0; missing value for this field is 9999.
%Field:13 Horizontal Infrared Radiation Intensity
%This is the Horizontal Infrared Radiation Intensity in Wh/m2. If it is missing, it is calculated
from the Opaque Sky Cover field as shown in the following explanation. It should have a minimum value
of 0; missing value for this field is 9999.
%Field:14 Global Horizontal Radiation
%This is the Global Horizontal Radiation in Wh/m2. (Total amount of direct and diffuse solar
radiation in Wh/m2 received on a horizontal surface during the number of minutes preceding the time
indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0;
missing value for this field is 9999.
%Field:15 Direct Normal Radiation
%This is the Direct Normal Radiation in Wh/m2. (Amount of solar radiation in Wh/m2 received directly
from the solar disk on a surface perpendicular to the sunís rays, during the number of minutes
```

preceding the time indicated.) If the field is ìmissing (9999)î or invalid (<0), it is set to 0. Counts of such missing values are totaled and presented at the end of the runperiod.
%Field:16 Diffuse Horizontal Radiation
%This is the Diffuse Horizontal Radiation in Wh/m2. (Amount of solar radiation in Wh/m2 received from the sky (excluding the solar disk) on a horizontal surface during the number of minutes preceding the time indicated.) If the field is ìmissing (9999)î or invalid (<0), it is set to 0. Counts of such missing values are totaled and presented at the end of the runperiod.
%Field:17 Global Horizontal Illuminance
%This is the Global Horizontal Illuminance in lux. (Average total amount of direct and diffuse illuminance in hundreds of lux received on a horizontal surface during the number of minutes preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 999999 and will be considered missing of >= 999900.
%Field:18 Direct Normal Illuminance
%This is the Direct Normal Illuminance in lux. (Average amount of illuminance in hundreds of lux received directly from the solar disk on a surface perpendicular to the sunís rays, during the number of minutes preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 999999 and will be considered missing of >= 999900.
%Field:19 Diffuse Horizontal Illuminance
%This is the Diffuse Horizontal Illuminance in lux. (Average amount of illuminance in hundreds of lux received from the sky (excluding the solar disk) on a horizontal surface during the number of minutes preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 999999 and will be considered missing of >= 999900.
%Field:20 Zenith Luminance
%This is the Zenith Illuminance in Cd/m2. (Average amount of luminance at the skyís zenith in tens of Cd/m2 during the number of minutes preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 9999.
%Field:21 Wind Direction
%This is the Wind Direction in degrees where the convention is that North=0.0, East=90.0, South=180.0, West=270.0. (Wind direction in degrees at the time indicated. If calm, direction equals zero.) Values can range from 0 to 360. Missing value is 999.
%Field:22 Wind Speed
%This is the wind speed in m/sec. (Wind speed at time indicated.) Values can range from 0 to 40. Missing value is 999.
%Field:23 Total Sky Cover
%This is the value for total sky cover (tenths of coverage). (i.e. 1 is 1/10 covered. 10 is total coverage). (Amount of sky dome in tenths covered by clouds or obscuring phenomena at the hour indicated at the time indicated.) Minimum value is 0; maximum value is 10; missing value is 99.
%Field:24 Opaque Sky Cover
%This is the value for opaque sky cover (tenths of coverage). (i.e. 1 is 1/10 covered. 10 is total coverage). (Amount of sky dome in tenths covered by clouds or obscuring phenomena that prevent observing the sky or higher cloud layers at the time indicated.) This is not used unless the field for Horizontal Infrared Radiation Intensity is missing and then it is used to calculate Horizontal Infrared Radiation Intensity. Minimum value is 0; maximum value is 10; missing value is 99.
%Field:25 Visibility
%This is the value for visibility in km. (Horizontal visibility at the time indicated.) It is not currently used in EnergyPlus calculations. Missing value is 9999.
%Field:26 Ceiling Height
%This is the value for ceiling height in m. (77777 is unlimited ceiling height. 88888 is cirroform ceiling.) It is not currently used in EnergyPlus calculations. Missing value is 99999.
%Field:27 Present Weather Observation
%If the value of the field is 0, then the observed weather codes are taken from the following field. If the value of the field is 9, then ìmissingî weather is assumed. Since the primary use of these fields (Present Weather Observation and Present Weather Codes) is for rain/wet surfaces, a missing observation field or a missing weather code implies ìno rainî.

%TMY 2 data fields

% Field 1 Year (2 digits)
% Field 2 Month
% Field 3 Day
% Field 4 Hour - Note TMY hour is that of local standard time - in EPW the time zone is indicated
% Field 5 Extraterrestrial Horizontal Radiation: (11 EPW 5 in WF_data) - Amount of solar radiation in Wh/m2 received on a horizontal surface at the top of the atmosphere during the 60 minutes preceding the hour indicated
% Field 6 Extraterrestrial Direct Normal Radiation: (12 EPW 6 in WF_data)- Amount of solar radiation in Wh/m2 received on a surface normal to the sun at the top of the atmosphere during the 60 minutes preceding the hour indicated

```
% Field 7 Global Horizontal Radiation: (14 EPW 8 in WR_data) - Total amount of direct and diffuse
solar radiation in Wh/m2 received on a horizontal surface during the 60 minutes preceding the hour
indicated
% Field 8 Direct Normal Radiation: (15 EPW 9 in WR_data) - Amount of solar radiation in Wh/m2
received within a 5.7¬∞ field of view centered on the sun during the 60 minutes preceding the hour
indicated
% Field 9 Diffuse Horizontal Radiation: (16 EPW 10 in WR_data) - Amount of solar radiation in Wh/m2
received from the sky (excluding the solar disk) on a horizontal surface during the 60 minutes
preceding the hour indicated
% Field 10 Global Horiz. Illuminance: (17 EPW 11 in WR_data) - Average total amount of direct and
diffuse illuminance in hundreds of lux received on a horizontal surface during the 60 minutes
preceding the hour indicated
% Field 11 Direct Normal Illuminance: (18 EPW 12 in WR_data) - Average amount of direct normal
illuminance in hundreds of lux received within a 5.7¬∞ field of view centered on the sun during the
60 minutes preceding the hour indicated.
% Field 12 Diffuse Horiz. Illuminance: (19 EPW 13 in WR_data) - Average amount of illuminance in
hundreds of lux received from the sky (excluding the solar disk) on a horizontal surface during the
60 minutes preceding the hour indicated.
% Field 13 Zenith Luminance: (20 EPW 14 in WR_data) - Average amount of luminance at the sky's zenith
in tens of Cd/m2 during the 60 minutes preceding the hour indicated.
% Field 14 Total Sky Cover: (23 EPW 17 in WR_data) - Amount of sky dome in tenths covered by clouds
or obscuring phenomena at the hour indicated
% Field 15 Opaque Sky Cover: (24 EPW 18 in WR_data) - Amount of sky dome in tenths covered by clouds
or obscuring phenomena that prevent observing the sky or higher cloud layers at the hour indicated
% Field 16 Dry Bulb Temperature: (7 EPW in C, 1 in WR_data) - Dry bulb temperature in tenths of oC at
the hour indicated -500 to 500 = -50.0 to 50.0oC
% Field 17 Dew Point Temperature: (8 EPW in C, 2 in WR_data) - Dew point temperature in tenths of oC
at the hour indicated -600 to 300 = -60.0 to 30.0oC
% Field 18 Relative Humidity: (9 EPW 3 in WR_data) - Relative humidity in percent at the hour
indicated
% Field 19 Atmospheric Pressure: (10 EPW in Pa, 4 in WR_data) - Atmospheric pressure at station in
millibars at the hour indicated (Pa / 100)
% Field 20 Wind Direction: (21 EPW 15 in WR_data) - Wind direction in degrees at the hour indicated.
( N = 0 or 360, E = 90, S = 180,W = 270 ). For calm winds, wind direction equals zero.
% Field 21 Wind Speed: (22 EPW in m/s, 16 in WR_data) - Wind speed in tenths of meters per second at
the hour indicated.
% Field 22 Visibility: (25 EPW in km, 17 in WR_data) - Horizontal visibility in tenths of kilometers
at the hour indicated.
% Field 23 Ceiling Height: (26 EPW 18 in WR_data) - Ceiling height in meters at the hour indicated.
% Field 24 Present Weather: (27 EPW 19 in WR_data) - Present weather conditions denoted by a 10-digit
number.

% For Cumulative distribution functions and  Weighting values for FS statistics: TMY3 weightings and
parameters
% Daily max dry bulb temperature 1/20
% Daily min dry bulb temperature 1/20
% Daily mean dry bulb temperature 2/20
% Daily max dewpoint temperature 1/20
% Daily min dewpoint temperature 1/20
% Daily mean dewpoint temperatue 2/20
% Daily max wind velocity 1/20
% Daily mean wind velocity 1/20
% Daily global radiation 5/20
% Daily direct radiation 5/20


%Day number for first day of each month except leap years
Daynumber = [1,32,60,91,121,152,182,213,244,274,305,335];

 WS_data_resampled.EstimHourlySol = zeros(8760,1);
 WS_data_resampled.RelativeHumidity = zeros(8760,1);
 WS_data_resampled.SolAltAngle = zeros(8760,1);
 WS_data_resampled.HourAngle = zeros(8760,1);
 WS_data_resampled.SolarTime = zeros(8760,1);
 WS_data_resampled.Declination = zeros(8760,1);
 WS_data_resampled.Precipit = zeros(8760,1);
 WS_data_resampled.DayOfYear = zeros(8760,1);
 WS_data_resampled.Second = zeros(8760,1);
 WS_data_resampled.Minute = zeros(8760,1);
 WS_data_resampled.Hour = zeros(8760,1);
 WS_data_resampled.DayOfMonth = zeros(8760,1);
 WS_data_resampled.Month = zeros(8760,1);
```

```
 WS_data_resampled.Year = zeros(8760,1);
 WS_data_resampled.CloudCover = zeros(8760,1);
 WS_data_resampled.WindSpeed = zeros(8760,1);
 WS_data_resampled.WindDir = zeros(8760,1);
 WS_data_resampled.DewPoint = zeros(8760,1);
 WS_data_resampled.DryBulb = zeros(8760,1);
 WS_data_resampled.Pressure = zeros(8760,1);

 % Set up classes for Koppen-Gieger classification of sites for coefficients
% for solar radiation estimation from JOE, Yu, et al. Development of 3012
% IWEC2 Weather Files for International Locations (RP-1477).
% Ashrae Transactions, 2014, vol. 120, no 1.
% Source KoppenGeiger.m : Beck, H., Zimmermann, N., McVicar, T. et al.
% Present and future Kˆppen-Geiger climate classification maps at 1-km resolution.
% Sci Data 5, 180214 (2018). https://doi.org/10.1038/sdata.2018.214


classes = {...
        1,1,'Af','Tropical, rainforest',[0 0 255];...
        2,1,'Am','Tropical, monsoon',[0 120 255];...
        3,1,'Aw','Tropical, savannah',[70 170 250];... % also As - same coefficients for solar
estimation
        4,2,'BWh','Arid, desert, hot',[255 0 0];...
        5,2,'BWk','Arid, desert, cold',[255 150 150];...
        6,2,'BSh','Arid, steppe, hot',[245 165 0];...
        7,2,'BSk','Arid, steppe, cold',[255 220 100];...
        8,3,'Csa','Temperate, dry summer, hot summer',[255 255 0];...
        9,3,'Csb','Temperate, dry summer, warm summer',[200 200 0];...
        10,3,'Csc','Temperate, dry summer, cold summer',[150 150 0];...
        11,3,'Cwa','Temperate, dry winter, hot summer',[150 255 150];...
        12,3,'Cwb','Temperate, dry winter, warm summer',[100 200 100];...
        13,3,'Cwc','Temperate, dry winter, cold summer',[50 150 50];...
        14,3,'Cfa','Temperate, no dry season, hot summer',[200 255 80];...
        15,3,'Cfb','Temperate, no dry season, warm summer',[100 255 80];...
        16,3,'Cfc','Temperate, no dry season, cold summer',[50 200 0];...
        17,4,'Dsa','Cold, dry summer, hot summer',[255 0 255];...
        18,4,'Dsb','Cold, dry summer, warm summer',[200 0 200];...
        19,4,'Dsc','Cold, dry summer, cold summer',[150 50 150];...
        20,4,'Dsd','Cold, dry summer, very cold winter',[150 100 150];...
        21,4,'Dwa','Cold, dry winter, hot summer',[170 175 255];...
        22,4,'Dwb','Cold, dry winter, warm summer',[90 120 220];...
        23,4,'Dwc','Cold, dry winter, cold summer',[75 80 180];...
        24,4,'Dwd','Cold, dry winter, very cold winter',[50 0 135];...
        25,4,'Dfa','Cold, no dry season, hot summer',[0 255 255];...
        26,4,'Dfb','Cold, no dry season, warm summer',[55 200 255];...
        27,4,'Dfc','Cold, no dry season, cold summer',[0 125 125];...
        28,4,'Dfd','Cold, no dry season, very cold winter',[0 70 95];...
        29,5,'ET','Polar, tundra',[178 178 178];...
        30,5,'EF','Polar, frost',[102 102 102];...
        };

    %UTC offset corrected 8 December 2023 - needs change in call to
    %timezone to change it according to the UTC offset.

KoppenClass = {...
'MEX_ACAPULCO_768050','Aw',-6;...
'MEX_ACAPULCO-G-ALVAREZ_768056','Aw',-6;...
'MEX_AGUASCALIENTES_765710','BSh',-6;...
'MEX_ALTAR-SON_761130','BSh',-7;...
'MEX_ARRIAGA_768400','Aw',-6;...
'MEX_BAHIAS-DE-HUATULCO_768485','Aw',-6;...
'MEX_CABO-SAN-LUCAS_767503','BWh',-7;...
'MEX_CAMPECHE-IGNACIO_766950','Aw',-6;...
'MEX_CAMPECHE-IGNACIO_766961','Aw',-6;...
'MEX_CANCUN_765950','Aw',-5;...
'MEX_CANCUN-IAP_765905','Aw',-5;...
'MEX_CANCUN-IAP_765906','Aw',-5;...
'MEX_CHETUMAL_767500','Aw',-5;...
'MEX_CHICHEN-ITZA_767501','Aw',-6;...
'MEX_CHIHUAHUA-G-FIERRO_762253','BSh',-6;...
'MEX_CHIHUAHUA-IAP_762252','BSh',-6;...
'MEX_CHIHUAHUA-UNIVERSIT_762250','BSh',-6;...
```

```
'MEX_CHILPANCINGO_767620','Aw',-6;...
'MEX_CHOIX_763110','Csa',-7;...
'MEX_CIUDAD-CONSTITUCION_764020','BWh',-7;...
'MEX_CIUDAD-DEL-CARMEN-I_767493','Aw',-6;...
'MEX_CIUDAD-GUZMAN_766560','Csb',-6;...
'MEX_CIUDAD-JUAREZ_760751','BWh',-7;...
'MEX_CIUDAD-OBREGON-SON_762580','BWh',-7;...
'MEX_CIUDAD-VICTORIA_764910','Cfa',-6;...
'MEX_CIUDAD-VICTORIA_764915','Cfa',-6;...
'MEX_COATZACOALCOS_767410','Am',-6;...
'MEX_COLIMA_766580','Aw',-6;...
'MEX_COLONIA-JUAN-CARRAS_764580','Aw',-7;...
'MEX_COLOTLAN_765190','Csa',-6;...
'MEX_COMITAN_768480','Cwb',-6;...
'MEX_COZUMEL-AP_766480','Aw',-5;...
'MEX_COZUMEL-IAP_766493','Aw',-5;...
'MEX_C-P-A-CARLOS-ROVIRO_767433','Af',-6;...
'MEX_CUERNAVACA_767260','Aw',-6;...
'MEX_CULIACAN(CITY)_764120','BSh',-7;...
'MEX_DE-GUANAJUATO-IAP_765773','BSh',-6;...
'MEX_DON-MIGUEL-Y-HIDALG_766133','Csa',-6;...
'MEX_DURANGO(CITY)_764230','BSk',-6;...
'MEX_DURANGO-IAP_764235','BSk',-6;...
'MEX_EJIDO-NUEVO-LEON-BC_760400','BWh',-8;...
'MEX_EMPALME_762560','BWh',-7;...
'MEX_FELIPE-CARRILLO-PUE_766980','Aw',-5;...
'MEX_FES-CUAUTITLAN_766700','Cwb',-6;...
'MEX_GEOGRAFIA-UNAM_766810','Cwb',-6;...
'MEX_GUADALAJARA_766120','Csa',-6;...
'MEX_GUADALAJARA-IAP_766131','Csa',-6;...
'MEX_GUANAJUATO_765770','Csa',-6;...
'MEX_GUAYMAS-G-YANEZ-AP_762555','BWh',-7;...
'MEX_HACIENDA-YLANG-YLAN_766920','Aw',-6;...
'MEX_HERMOSILLO-IAP_761600','BWh',-7;...
'MEX_HUAJUAPAN-DE-LEON_767730','Cwb',-6;...
'MEX_ISLA-GUADALUPE_761510','BSh',-8;...
'MEX_ISLA-SOCORRO_767230','BWh',-7;...
'MEX_IXTAPA-ZIHUATANEJO_767584','Aw',-6;...
'MEX_IXTEPEC_767502','Aw',-6;...
'MEX_JALAPA_766870','Cfa',-6;...
'MEX_JUAREZ-G-GONZALEZ_760753','BWh',-7;...
'MEX_LAGOS-DE-MORENO_765730','Csa',-6;...
'MEX_LA-PAZ(CITY)_764050','BWh',-7;...
'MEX_LA-PAZ-G-MARQUEZ-AP_764055','BWh',-7;...
'MEX_LEON-SAN-CARLOS_765752','BSh',-6;...
'MEX_LICENCIADO-GUSTAVO_766013','Aw',-6;...
'MEX_LICENCIADO-Y-GEN-IG_766546','Cwb',-6;...
'MEX_LORETO_763050','BWh',-7;...
'MEX_LOS-CABOS-IAP_764056','BWh',-7;...
'MEX_MANZANILLO_766540','Aw',-6;...
'MEX_MATAMOROS-G-CANALES_763993','Cfa',-6;...
'MEX_MATAMOROS-IAP_763991','Cfa',-6;...
'MEX_MATLAPA_765850','Aw',-6;...
'MEX_MAZATLAN-G-BUELNA-I_764591','Aw',-7;...
'MEX_MAZATLAN-G-BUELNA-II_764593','Aw',-7;...
'MEX_MERIDA-IAP_766440','Aw',-6;...
'MEX_MEXICALI_760051','BWh',-8;...
'MEX_MEXICALI-G-SANCHEZ_760053','BWh',-8;...
'MEX_MEXICO-CITY_766800','Cwb',-6;...
'MEX_MEXICO-CITY-B-JUAREZ_766793','Cwb',-6;...
'MEX_MEXICO-CITY-IAP_766790','Cwb',-6;...
'MEX_MINATITLAN_767383','Am',-6;...
'MEX_MONCLOVA_763420','BSh',-6;...
'MEX_MONTERREY(CITY)_763930','BSh',-6;...
'MEX_MONTERREY-G-ESCOBEDO_763943','BSh',-6;...
'MEX_MONTERREY-IAP_763940','BSh',-6;...
'MEX_MORELIA_766650','Cwb',-6;...
'MEX_MORELIA-G-MUJICA-AP_766655','Cwb',-6;...
'MEX_NUEVA-CASAS-GRANDES_761220','BSk',-6;...
'MEX_NUEVO-LAREDO_762861','BSh',-6;...
'MEX_OAXACA_767750','Cwa',-6;...
'MEX_ORIZABA-VER_767370','Cwb',-6;...
```

```
'MEX_PACHUCA_766320','BSk',-6;...
'MEX_PARRAL_763230','BSk',-6;...
'MEX_PIEDRAS-NEGRAS_762430','Cfa',-6;...
'MEX_PILARES-DE-NACOZARI_761180','BSh',-7;...
'MEX_PLAYA-DE-ORO-IAP_766534','Aw',-6;...
'MEX_PROGRESO_765930','BSh',-6;...
'MEX_PUEBLA_766850','Cwb',-6;...
'MEX_PUERTO-ANGEL_768550','Am',-6;...
'MEX_PUERTO-ESCONDIDO_768556','Aw',-6;...
'MEX_PUERTO-PENASCO_760610','BWh',-7;...
'MEX_QUERETARO_766250','BSk',-6;...
'MEX_QUETZALCOATL-IAP_762863','BSh',-6;...
'MEX_REYNOSA-G-BLANCO-AP_763503','BSh',-6;...
'MEX_REYNOSA-IAP_763501_75','BSh',-6;...
'MEX_RIO-VERDE_765810','Cfa',-6;...
'MEX_SALINA-CRUZ_768330','Aw',-6;...
'MEX_SALTILLO_763900','BSh',-6;...
'MEX_SAN-FELIPE_760550','BWh',-8;...
'MEX_SAN-LUIS-POTOSI_765390','BSk',-6;...
'MEX_SANTA-ROSALIA_762530','BSh',-7;...
'MEX_SN-CRISTOBAL-LAS-C_768450','Cwb',-6;...
'MEX_SOMBRERETE_764710','BSk',-6;...
'MEX_SOTO-LA-MARINA_764990','BSh',-6;...
'MEX_TAJIN_766127','Af',-6;...
'MEX_TAMPICO_765480','BSh',-6;...
'MEX_TAMPICO-GEN-FJ-MINA_765491','BSh',-6;...
'MEX_TAMPICO-G-MINA-AP_765493','BSh',-6;...
'MEX_TAMUIN-SLP_765430_02','Aw',-6;...
'MEX_TAPACHULA-CITY_769030','Aw',-6;...
'MEX_TAPACHULA-IAP_769043','Aw',-6;...
'MEX_TEMOSACHIC_762200','BSk',-6;...
'MEX_TEPEHUANES_763730','Csb',-6;...
'MEX_TEPIC_765560','Cwa',-6;...
'MEX_TIJUANA-G-RODRIGUE_760013','BSh',-8;...
'MEX_TLAXCALA_766830','Cwb',-6;...
'MEX_TOLUCA_766750','Cwb',-6;...
'MEX_TOLUCA-LA-LOPEZ-AP_766753','Cwb',-6;...
'MEX_TORREON-AP_763820','BWh',-6;...
'MEX_TULANCINGO_766340','Cfb',-6;...
'MEX_TUXPAN-VER_766400','Am',-6;...
'MEX_TUXTLA-GUTIERREZ_760754','Aw',-6;...
'MEX_TUXTLA-GUTIERREZ-A_768430','Aw',-6;...
'MEX_VALLADOLID_766470','Aw',-6;...
'MEX_VALLE-DEL-FUERTE-IN_763615','BWh',-7;...
'MEX_VERACRUZ-GEN-JARA_766910','Aw',-6;...
'MEX_VERACRUZ-G-JARA-AP_766913','Aw',-6;...
'MEX_VILLAHERMOSA_767441_75','Am',-6;...
'MEX_VILLAHERMOSA-TAB_767430','Am',-6;...
'MEX_XOXOCOTLAN-IAP_767755','Cwb',-6;...
'MEX_ZACATECAS(CITY)_765250','Cwb',-6;...
'MEX_ZACATECAS-G-RUIZ-AP_765255','Cwb',-6;...
'MEX_ZAMORA_766620','Csa',-6};

% Tau d and b for ASHRAE clear sky models from chapter 14 2021 fundamentals
% handbook database

taud =
[2.55600000000000,2.53900000000000,2.49100000000000,2.46200000000000,2.42800000000000,2.4230000000000
0,2.42800000000000,2.40300000000000,2.48300000000000,2.43900000000000,2.55400000000000,2.550000000000
00;2.51900000000000,2.53200000000000,2.45600000000000,2.42700000000000,2.39700000000000,2.39600000000
000,2.38400000000000,2.38800000000000,2.44400000000000,2.49400000000000,2.52900000000000,2.5380000000
0000;2.53700000000000,2.53000000000000,2.43200000000000,2.41600000000000,2.38500000000000,2.315000000
00000,2.31100000000000,2.34800000000000,2.38700000000000,2.47400000000000,2.53200000000000,2.54500000
000000;2.54200000000000,2.47800000000000,2.45500000000000,2.33300000000000,2.27300000000000,2.3360000
0000000,2.38400000000000,2.43400000000000,2.38500000000000,2.51600000000000,2.49600000000000,2.527000
00000000;2.65300000000000,2.57900000000000,2.53700000000000,2.41100000000000,2.29500000000000,2.39500
000000000,2.38400000000000,2.44300000000000,2.42500000000000,2.59900000000000,2.63500000000000,2.6480
0000000000;2.59400000000000,2.54900000000000,2.50000000000000,2.40300000000000,2.39200000000000,2.480
00000000000,2.49800000000000,2.55900000000000,2.51600000000000,2.58100000000000,2.57100000000000,2.59
600000000000;2.50300000000000,2.42800000000000,2.33300000000000,2.29200000000000,2.24300000000000,2.3
9300000000000,2.43900000000000,2.46600000000000,2.39500000000000,2.46300000000000,2.44100000000000,2.
48900000000000;2.51400000000000,2.44200000000000,2.34800000000000,2.30400000000000,2.25100000000000,2
```

.40400000000000,2.44400000000000,2.46700000000000,2.39900000000000,2.47600000000000,2.43900000000000,
2.49000000000000;2.64700000000000,2.60700000000000,2.56800000000000,2.46600000000000,2.40300000000000
,2.36000000000000,2.40000000000000,2.46000000000000,2.41700000000000,2.61600000000000,2.6810000000000
0,2.65000000000000;2.63200000000000,2.53500000000000,2.57400000000000,2.51200000000000,2.386000000000
00,2.41200000000000,2.41700000000000,2.46000000000000,2.44500000000000,2.58300000000000,2.64200000000
000,2.63900000000000;2.59600000000000,2.55000000000000,2.59600000000000,2.45300000000000,2.3710000000
0000,2.35200000000000,2.37600000000000,2.43900000000000,2.39700000000000,2.57800000000000,2.625000000
00000,2.63100000000000;2.60300000000000,2.53400000000000,2.58800000000000,2.44200000000000,2.37000000
000000,2.37000000000000,2.38000000000000,2.45700000000000,2.40000000000000,2.57700000000000,2.6520000
0000000,2.64000000000000;2.55000000000000,2.50800000000000,2.50300000000000,2.42200000000000,2.325000
00000000,2.38100000000000,2.41100000000000,2.44500000000000,2.43500000000000,2.52400000000000,2.59400
000000000,2.56800000000000;2.49800000000000,2.44500000000000,2.42300000000000,2.36100000000000,2.2610
0000000000,2.34300000000000,2.39600000000000,2.43600000000000,2.42100000000000,2.47300000000000,2.510
00000000000,2.51600000000000;2.39100000000000,2.34900000000000,2.28400000000000,2.14400000000000,2.06
500000000000,2.29100000000000,2.31700000000000,2.35000000000000,2.33800000000000,2.40600000000000,2.3
8900000000000,2.40300000000000;2.58200000000000,2.53200000000000,2.55100000000000,2.47000000000000,2.
31300000000000,2.29500000000000,2.37700000000000,2.37700000000000,2.39000000000000,2.52500000000000,2
.61400000000000,2.58700000000000;2.49600000000000,2.43900000000000,2.45400000000000,2.34100000000000,
2.25900000000000,2.30300000000000,2.36500000000000,2.39200000000000,2.39800000000000,2.46800000000000
,2.52100000000000,2.51700000000000;2.46300000000000,2.42700000000000,2.42800000000000,2.3310000000000
0,2.21800000000000,2.32300000000000,2.36300000000000,2.40200000000000,2.40400000000000,2.445000000000
00,2.48900000000000,2.48900000000000;2.45500000000000,2.41000000000000,2.42100000000000,2.31000000000
000,2.19400000000000,2.29700000000000,2.33700000000000,2.37300000000000,2.38400000000000,2.4360000000
0000,2.47800000000000,2.47200000000000;2.56500000000000,2.52000000000000,2.41700000000000,2.299000000
00000,2.31000000000000,2.37300000000000,2.30800000000000,2.39100000000000,2.46200000000000,2.50600000
000000,2.52500000000000,2.53700000000000;2.55200000000000,2.51300000000000,2.46200000000000,2.3390000
0000000,2.33400000000000,2.30800000000000,2.24700000000000,2.38500000000000,2.44100000000000,2.442000
00000000,2.50200000000000,2.51400000000000;2.56400000000000,2.52100000000000,2.52300000000000,2.41800
000000000,2.28900000000000,2.28800000000000,2.38500000000000,2.38700000000000,2.37400000000000,2.5160
0000000000,2.58600000000000,2.57000000000000;2.51300000000000,2.47300000000000,2.49700000000000,2.367
00000000000,2.23700000000000,2.26300000000000,2.30800000000000,2.30600000000000,2.36600000000000,2.46
400000000000,2.54700000000000,2.53400000000000;2.41300000000000,2.38500000000000,2.34100000000000,2.2
9800000000000,2.16100000000000,2.30500000000000,2.31400000000000,2.37000000000000,2.36800000000000,2.
38700000000000,2.43900000000000,2.44200000000000;2.28500000000000,2.25800000000000,2.21900000000000,2
.10600000000000,2.18200000000000,2.18600000000000,2.26900000000000,2.29500000000000,2.42400000000000,
2.28500000000000,2.39900000000000,2.32900000000000;2.32500000000000,2.29100000000000,2.18700000000000
,2.11200000000000,2.05700000000000,2.23900000000000,2.28400000000000,2.32200000000000,2.3120000000000
0,2.33200000000000,2.32800000000000,2.31900000000000;2.55400000000000,2.50900000000000,2.402000000000
00,2.29200000000000,2.29200000000000,2.34800000000000,2.28300000000000,2.36600000000000,2.43900000000
000,2.48600000000000,2.50500000000000,2.52100000000000;2.45000000000000,2.42400000000000,2.4230000000
0000,2.27700000000000,2.05600000000000,2.15500000000000,2.26600000000000,2.29200000000000,2.300000000
00000,2.42400000000000,2.50500000000000,2.47200000000000;2.42700000000000,2.39300000000000,2.40600000
000000,2.30200000000000,2.09600000000000,2.24700000000000,2.30000000000000,2.30600000000000,2.3030000
0000000,2.38900000000000,2.46300000000000,2.46200000000000;2.43400000000000,2.40000000000000,2.406000
00000000,2.31200000000000,2.12300000000000,2.27000000000000,2.31700000000000,2.33100000000000,2.32800
000000000,2.39500000000000,2.46400000000000,2.46800000000000;2.20200000000000,2.14400000000000,2.1450
0000000000,1.98200000000000,2.14300000000000,1.97500000000000,2.04500000000000,2.10500000000000,2.449
00000000000,2.19000000000000,2.44500000000000,2.23100000000000;2.08800000000000,2.03800000000000,2.02
400000000000,1.87000000000000,2.11700000000000,1.89900000000000,1.98000000000000,2.03700000000000,2.4
4800000000000,2.07000000000000,2.36600000000000,2.11900000000000;2.15500000000000,2.17500000000000,2.
11500000000000,2.02900000000000,2.17800000000000,2.09600000000000,2.20500000000000,2.21700000000000,2
.46500000000000,2.23400000000000,2.37600000000000,2.22400000000000;2.23600000000000,2.27300000000000,
2.18200000000000,2.11100000000000,2.17700000000000,2.18200000000000,2.28500000000000,2.29200000000000
,2.45400000000000,2.32100000000000,2.40500000000000,2.30700000000000;2.41600000000000,2.3900000000000
0,2.29400000000000,2.18500000000000,2.14500000000000,2.31700000000000,2.34400000000000,2.367000000000
00,2.34600000000000,2.42000000000000,2.38800000000000,2.39500000000000;2.43800000000000,2.40500000000
000,2.28700000000000,2.12600000000000,2.04700000000000,2.27300000000000,2.32600000000000,2.3580000000
0000,2.36400000000000,2.41700000000000,2.43600000000000,2.42000000000000;2.44300000000000,2.409000000
00000,2.28900000000000,2.12600000000000,2.04800000000000,2.27700000000000,2.32400000000000,2.36000000
000000,2.36400000000000,2.42300000000000,2.44200000000000,2.42600000000000;2.57000000000000,2.5100000
0000000,2.38000000000000,2.24200000000000,2.23700000000000,2.37300000000000,2.31500000000000,2.398000
00000000,2.44800000000000,2.51500000000000,2.53400000000000,2.52200000000000;2.37000000000000,2.32600
000000000,2.24100000000000,2.15900000000000,2.04700000000000,2.26000000000000,2.33000000000000,2.3620
0000000000,2.33600000000000,2.36600000000000,2.36300000000000,2.33300000000000;2.48700000000000,2.442
00000000000,2.29300000000000,2.15200000000000,2.04400000000000,2.31200000000000,2.29600000000000,2.32
300000000000,2.34300000000000,2.43300000000000,2.46200000000000,2.47400000000000;2.55100000000000,2.5
0000000000000,2.39700000000000,2.25800000000000,2.22000000000000,2.30000000000000,2.18500000000000,2.
28300000000000,2.37600000000000,2.47200000000000,2.50300000000000,2.52200000000000;2.44400000000000,2
.44900000000000,2.25900000000000,2.27000000000000,2.10700000000000,2.32700000000000,2.40100000000000,
2.38300000000000,2.37800000000000,2.43400000000000,2.47800000000000,2.50900000000000;2.41200000000000
,2.40600000000000,2.33400000000000,2.17000000000000,1.90900000000000,2.27300000000000,2.3980000000000
0,2.37900000000000,2.38300000000000,2.43500000000000,2.48400000000000,2.49900000000000;2.471000000000

```
00,2.39100000000000,2.21200000000000,1.92600000000000,1.80700000000000,2.20800000000000,2.24500000000
000,2.25700000000000,2.24400000000000,2.33300000000000,2.42400000000000,2.47000000000000;2.5720000000
0000,2.49400000000000,2.33500000000000,2.12000000000000,1.99600000000000,2.35700000000000,2.395000000
00000,2.35100000000000,2.36700000000000,2.49000000000000,2.53300000000000,2.57200000000000;2.44600000
000000,2.37200000000000,2.15200000000000,1.88400000000000,1.79900000000000,2.22200000000000,2.2160000
0000000,2.24900000000000,2.23300000000000,2.37900000000000,2.42700000000000,2.45800000000000;2.469000
00000000,2.40200000000000,2.20100000000000,1.93100000000000,1.86500000000000,2.26200000000000,2.24800
000000000,2.27800000000000,2.26300000000000,2.41400000000000,2.45600000000000,2.48500000000000;2.5500
0000000000,2.52800000000000,2.52300000000000,2.37400000000000,2.36800000000000,2.31600000000000,2.253
00000000000,2.41600000000000,2.47300000000000,2.53100000000000,2.53500000000000,2.54500000000000;2.52
200000000000,2.45500000000000,2.36800000000000,2.17500000000000,2.16500000000000,2.37300000000000,2.3
2300000000000,2.40800000000000,2.44100000000000,2.51300000000000,2.51200000000000,2.49600000000000];
taub =
[0.310000000000000,0.321000000000000,0.333000000000000,0.344000000000000,0.363000000000000,0.36600000
0000000,0.394000000000000,0.396000000000000,0.359000000000000,0.364000000000000,0.320000000000000,0.3
22000000000000;0.288000000000000,0.290000000000000,0.310000000000000,0.322000000000000,0.339000000000
000,0.341000000000000,0.386000000000000,0.381000000000000,0.347000000000000,0.317000000000000,0.29300
0000000000,0.289000000000000;0.291000000000000,0.294000000000000,0.321000000000000,0.326000000000000,
0.344000000000000,0.390000000000000,0.434000000000000,0.422000000000000,0.394000000000000,0.341000000
000000,0.301000000000000,0.291000000000000;0.298000000000000,0.314000000000000,0.326000000000000,0.37
1000000000000,0.409000000000000,0.407000000000000,0.404000000000000,0.388000000000000,0.3960000000000
00,0.347000000000000,0.326000000000000,0.305000000000000;0.266000000000000,0.271000000000000,0.281000
000000000,0.320000000000000,0.373000000000000,0.369000000000000,0.384000000000000,0.366000000000000,0
.369000000000000,0.311000000000000,0.281000000000000,0.267000000000000;0.271000000000000,0.2810000000
00000,0.298000000000000,0.331000000000000,0.349000000000000,0.342000000000000,0.347000000000000,0.330
000000000000,0.337000000000000,0.308000000000000,0.289000000000000,0.274000000000000;0.3290000000000
0,0.350000000000000,0.387000000000000,0.410000000000000,0.442000000000000,0.409000000000000,0.4020000
00000000,0.393000000000000,0.409000000000000,0.378000000000000,0.362000000000000,0.336000000000000;0.
328000000000000,0.349000000000000,0.386000000000000,0.410000000000000,0.443000000000000,0.40900000000
0000,0.403000000000000,0.395000000000000,0.409000000000000,0.376000000000000,0.365000000000000,0.3380
00000000000;0.301000000000000,0.304000000000000,0.307000000000000,0.337000000000000,0.374000000000000
,0.419000000000000,0.425000000000000,0.406000000000000,0.409000000000000,0.350000000000000,0.31200000
0000000,0.301000000000000;0.260000000000000,0.276000000000000,0.270000000000000,0.292000000000000,0.3
46000000000000,0.355000000000000,0.374000000000000,0.361000000000000,0.360000000000000,0.310000000000
000,0.273000000000000,0.263000000000000;0.320000000000000,0.328000000000000,0.314000000000000,0.34900
0000000000,0.399000000000000,0.432000000000000,0.440000000000000,0.420000000000000,0.421000000000000,
0.369000000000000,0.333000000000000,0.319000000000000;0.318000000000000,0.331000000000000,0.316000000
000000,0.353000000000000,0.400000000000000,0.426000000000000,0.437000000000000,0.413000000000000,0.41
9000000000000,0.368000000000000,0.328000000000000,0.317000000000000;0.265000000000000,0.2760000000000
00,0.278000000000000,0.299000000000000,0.354000000000000,0.348000000000000,0.352000000000000,0.344000
000000000,0.345000000000000,0.310000000000000,0.274000000000000,0.272000000000000;0.294000000000000,0
.307000000000000,0.318000000000000,0.341000000000000,0.392000000000000,0.374000000000000,0.3640000000
00000,0.355000000000000,0.358000000000000,0.335000000000000,0.305000000000000,0.298000000000000;0.400
000000000000,0.414000000000000,0.438000000000000,0.493000000000000,0.527000000000000,0.4620000000000
0,0.458000000000000,0.446000000000000,0.445000000000000,0.420000000000000,0.410000000000000,0.3980000
00000000;0.305000000000000,0.315000000000000,0.306000000000000,0.326000000000000,0.399000000000000,0.
427000000000000,0.411000000000000,0.409000000000000,0.402000000000000,0.356000000000000,0.31500000000
0000,0.312000000000000;0.287000000000000,0.306000000000000,0.300000000000000,0.327000000000000,0.3870
00000000000,0.382000000000000,0.374000000000000,0.366000000000000,0.364000000000000,0.335000000000000
,0.299000000000000,0.295000000000000;0.304000000000000,0.315000000000000,0.314000000000000,0.34700000
0000000,0.405000000000000,0.379000000000000,0.373000000000000,0.360000000000000,0.360000000000000,0.3
42000000000000,0.309000000000000,0.305000000000000;0.310000000000000,0.324000000000000,0.319000000000
000,0.356000000000000,0.417000000000000,0.392000000000000,0.387000000000000,0.374000000000000,0.37200
0000000000,0.350000000000000,0.316000000000000,0.313000000000000;0.362000000000000,0.376000000000000,
0.401000000000000,0.436000000000000,0.451000000000000,0.449000000000000,0.465000000000000,0.442000000
000000,0.411000000000000,0.397000000000000,0.376000000000000,0.373000000000000;0.368000000000000,0.37
9000000000000,0.391000000000000,0.431000000000000,0.449000000000000,0.467000000000000,0.4810000000000
00,0.445000000000000,0.426000000000000,0.422000000000000,0.389000000000000,0.384000000000000;0.341000
000000000,0.346000000000000,0.336000000000000,0.364000000000000,0.435000000000000,0.458000000000000,0
.437000000000000,0.434000000000000,0.432000000000000,0.388000000000000,0.353000000000000,0.3480000000
00000;0.306000000000000,0.318000000000000,0.305000000000000,0.339000000000000,0.416000000000000,0.422
000000000000,0.413000000000000,0.408000000000000,0.395000000000000,0.354000000000000,0.3130000000000
0,0.312000000000000;0.329000000000000,0.338000000000000,0.353000000000000,0.375000000000000,0.4330000
00000000,0.395000000000000,0.398000000000000,0.379000000000000,0.378000000000000,0.368000000000000,0.
333000000000000,0.326000000000000;0.363000000000000,0.374000000000000,0.391000000000000,0.43900000000
0000,0.422000000000000,0.425000000000000,0.406000000000000,0.397000000000000,0.355000000000000,0.3870
00000000000,0.341000000000000,0.350000000000000;0.420000000000000,0.433000000000000,0.470000000000000
,0.506000000000000,0.530000000000000,0.475000000000000,0.465000000000000,0.451000000000000,0.44900000
0000000,0.440000000000000,0.427000000000000,0.421000000000000;0.358000000000000,0.372000000000000,0.4
00000000000000,0.435000000000000,0.451000000000000,0.448000000000000,0.465000000000000,0.441000000000
000,0.411000000000000,0.395000000000000,0.375000000000000,0.371000000000000;0.386000000000000,0.38600
0000000000,0.376000000000000,0.425000000000000,0.530000000000000,0.511000000000000,0.477000000000000,
```

```
0.467000000000000,0.464000000000000,0.423000000000000,0.386000000000000,0.388000000000000;0.331000000
000000,0.340000000000000,0.334000000000000,0.377000000000000,0.460000000000000,0.419000000000000,0.40
4000000000000,0.401000000000000,0.404000000000000,0.373000000000000,0.331000000000000,0.328000000000
00;0.329000000000000,0.339000000000000,0.336000000000000,0.375000000000000,0.452000000000000,0.413000
000000000,0.400000000000000,0.394000000000000,0.397000000000000,0.372000000000000,0.331000000000000,0
.326000000000000;0.377000000000000,0.394000000000000,0.401000000000000,0.469000000000000,0.429000000
00000,0.481000000000000,0.464000000000000,0.446000000000000,0.349000000000000,0.408000000000000,0.327
000000000000,0.367000000000000;0.415000000000000,0.433000000000000,0.446000000000000,0.519000000000000
0,0.448000000000000,0.516000000000000,0.494000000000000,0.476000000000000,0.359000000000000,0.4490000
00000000,0.357000000000000,0.403000000000000;0.390000000000000,0.390000000000000,0.416000000000000,0.
462000000000000,0.425000000000000,0.453000000000000,0.424000000000000,0.420000000000000,0.349000000000
0000,0.401000000000000,0.345000000000000,0.370000000000000;0.371000000000000,0.367000000000000,0.3990
00000000000,0.438000000000000,0.429000000000000,0.432000000000000,0.403000000000000,0.400000000000000
,0.353000000000000,0.382000000000000,0.339000000000000,0.353000000000000;0.359000000000000,0.36900000
0000000,0.401000000000000,0.444000000000000,0.464000000000000,0.416000000000000,0.410000000000000,0.4
02000000000000,0.406000000000000,0.380000000000000,0.373000000000000,0.364000000000000;0.393000000000
000,0.405000000000000,0.444000000000000,0.506000000000000,0.541000000000000,0.470000000000000,0.45800
0000000000,0.444000000000000,0.435000000000000,0.420000000000000,0.402000000000000,0.401000000000000;
0.393000000000000,0.405000000000000,0.444000000000000,0.506000000000000,0.541000000000000,0.471000000
000000,0.460000000000000,0.446000000000000,0.436000000000000,0.420000000000000,0.402000000000000,0.40
1000000000000;0.361000000000000,0.380000000000000,0.415000000000000,0.459000000000000,0.475000000000
00,0.446000000000000,0.464000000000000,0.437000000000000,0.413000000000000,0.393000000000000,0.374000
000000000,0.377000000000000;0.371000000000000,0.386000000000000,0.417000000000000,0.457000000000000,0
.503000000000000,0.440000000000000,0.420000000000000,0.408000000000000,0.414000000000000,0.399000000
00000,0.382000000000000,0.380000000000000;0.380000000000000,0.394000000000000,0.440000000000000,0.494
000000000000,0.541000000000000,0.460000000000000,0.468000000000000,0.457000000000000,0.445000000000000
0,0.418000000000000,0.396000000000000,0.388000000000000;0.369000000000000,0.381000000000000,0.4120000
00000000,0.459000000000000,0.483000000000000,0.467000000000000,0.495000000000000,0.468000000000000,0.
439000000000000,0.411000000000000,0.389000000000000,0.382000000000000;0.339000000000000,0.33800000000
0000,0.400000000000000,0.408000000000000,0.476000000000000,0.414000000000000,0.397000000000000,0.3960
00000000000,0.396000000000000,0.377000000000000,0.345000000000000,0.327000000000000;0.407000000000000
,0.406000000000000,0.429000000000000,0.484000000000000,0.595000000000000,0.479000000000000,0.44200000
0000000,0.444000000000000,0.437000000000000,0.424000000000000,0.400000000000000,0.388000000000000;0.3
65000000000000,0.392000000000000,0.446000000000000,0.553000000000000,0.614000000000000,0.478000000000
000,0.465000000000000,0.458000000000000,0.460000000000000,0.433000000000000,0.388000000000000,0.36700
0000000000;0.314000000000000,0.336000000000000,0.382000000000000,0.457000000000000,0.511000000000000,
0.408000000000000,0.396000000000000,0.404000000000000,0.395000000000000,0.362000000000000,0.333000000
000000,0.315000000000000;0.397000000000000,0.423000000000000,0.496000000000000,0.597000000000000,0.63
9000000000000,0.491000000000000,0.496000000000000,0.484000000000000,0.482000000000000,0.441000000000
00,0.416000000000000,0.399000000000000;0.395000000000000,0.420000000000000,0.485000000000000,0.584000
000000000,0.616000000000000,0.482000000000000,0.490000000000000,0.479000000000000,0.474000000000000,0
.434000000000000,0.412000000000000,0.396000000000000;0.379000000000000,0.384000000000000,0.384000000
00000,0.428000000000000,0.447000000000000,0.475000000000000,0.489000000000000,0.447000000000000,0.420
000000000000,0.408000000000000,0.392000000000000,0.387000000000000;0.380000000000000,0.399000000000000
0,0.421000000000000,0.485000000000000,0.498000000000000,0.453000000000000,0.469000000000000,0.4460000
00000000,0.428000000000000,0.404000000000000,0.390000000000000,0.389000000000000];
LatDecim =
[32.5410000000000,32.6310000000000,29.7840000000000,35.0030000000000,30.2000000000000,28.514000000000
0,31.8250000000000,25.7780000000000,30.3500000000000,24.6140000000000,25.1720000000000,23.16100000000
00,22.8970000000000,23.7580000000000,22.2960000000000,26.1890000000000,29.5110000000000,21.0140000000
000,20.9830000000000,23.7670000000000,21.2890000000000,20.6800000000000,20.5220000000000,25.634000000
0000,20.8420000000000,29.6000000000000,29.4690000000000,19.4440000000000,26.2170000000000,19.85000000
00000,19.3370000000000,19.4360000000000,22.2500000000000,19.5500000000000,24.1250000000000,19.1460000
000000,20.4310000000000,27.3640000000000,26.6560000000000,19.4030000000000,23.0060000000000,17.002000
0000000,23.6280000000000,23.6310000000000,18.3190000000000,22.8720000000000,14.7940000000000,19.82000
00000000,22.1240000000000];
LongDecim =
[-116.970000000000,-115.242000000000,-119.310000000000,-105.339000000000,-107.161000000000,-101.23900
0000000,-102.717000000000,-100.107000000000,-111.411000000000,-110.006000000000,-110.111000000000,-10
6.266000000000,-102.687000000000,-109.875000000000,-97.8660000000000,-112.839000000000,-104.911000000
000,-101.266000000000,-101.483000000000,-95.5390000000000,-94.5220000000000,-105.254000000000,-103.31
1000000000,-103.694000000000,-101.575000000000,-101.172000000000,-95.5220000000000,-107.186000000000,
-102.831000000000,-101.026000000000,-99.5660000000000,-99.0720000000000,-100.472000000000,-99.6330000
000000,-105.042000000000,-96.1870000000000,-97.1140000000000,-95.0750000000000,-97.9830000000000,-99.
1080000000000,-91.2780000000000,-96.7210000000000,-106.494000000000,-94.4750000000000,-93.35300000000
00,-94.9640000000000,-92.3700000000000,-84.9450000000000,-93.9410000000000];
LatLong = zeros(49,2);
LatLong (:,1) = LatDecim;
LatLong (:,2) = LongDecim;

Hournumber = [0,744,1416,2160,2880,3624,4344,5088,5832,6552,7296,8016,8760];
```

```
%File location for recording files with suspicious results

%file = ['C:\Users\usuario\Documents\Christopher\Oddfilenames.txt'];

 %       fid = fopen(file,'w');


% Load epw file
% Provide location of files
%file_location = 'C:\Users\usuario\Documents\Christopher\RawYearlyData\BAJA-CALIF-NORTE\MEXICALI-G-
SANCHEZ_760053\';
%file_location = 'C:\Wthr\MEX_RIO-VERDE_765810_80.epw';
%file_location = 'C:\Wthr\MEX_MEXICO-CITY-B-JUAREZ_766793_14.epw';
%file_location = 'C:\Wthr\MEX_MERIDA-IAP_766440_08.epw';



file_location = 'C:\Wthr\Oddballs\';
file_list = dir([file_location,'*.epw']);
n_files = size(file_list,1);
%run through all of the files
for ifile = 1:n_files
    if ifile < 1
      clear (WF_date);
    end
  %ifile=1;
    WF_file = [file_location,file_list(ifile).name];
    WF_data = importdata(WF_file,',',8);

    %WF_file = [file_location];


    %extract the information from the header

    header = WF_data.textdata(1:8,1);
    flag_end = size(WF_data.textdata,1);


    %remove 29th of February for leap year
    if flag_end == 8792
        WF_datec(1:1416,1:5) = WF_data.textdata(9:1424,1:5);
        WF_datec(1417:8760,1:5) = WF_data.textdata(1449:8792,1:5);
        WS_data(1:1416,:) = WF_data.data(1:1416,:);
        WS_data(1417:8760,:) = WF_data.data(1441:8784,:);
    else
    WF_datec(1:8760,1:5) = WF_data.textdata(9:flag_end,1:5); %if matlab
    WS_data = WF_data.data; % If Matlab
    end

    %WF_date = [WF_data.data(1:8760,1:5)]; %, WF_data.textdata(9:flag_end,:)];
    %once extracted the data is all that is left to get.
    %WS_data = WF_data.data; % If Matlab

    %WF_data = WF_data.data(:,7:end);

    % Assign climate classification from file name and classification list
    % KoppenClass

    FilePlace =
extractBefore(string(file_list(ifile).name),strlength(string(file_list(ifile).name))-7);
    KoppenStr = KoppenClass(:,1);
    KoppClass = KoppenClass(contains(KoppenStr,FilePlace),2);
    UTCdiff = KoppenClass(contains(KoppenStr,FilePlace),3);
    UTCdiffd = cell2mat(UTCdiff); %str2double(cell2mat(UTCdiff)); % Time zone difference from UTC
(GMT)
    %extract the important information - this can be edited accordingly - Need to deal with Typical/
Extreme periods and ground temperatures
    C = strsplit(header{1,1},',','CollapseDelimiters',false);
    Lat = str2double(C{7});
    Lon = str2double(C{8});
    Altitude = str2double(C{10});
    RelativeToUTC = UTCdiffd; %str2double(C{9});
```

```
    Year = str2num(cell2mat(WF_datec(1,1)));
    Leap = ~mod(Year,4);
    if Leap
        Daynumber = [1,32,61,92,122,153,183,214,245,275,306,336];
    end



 Latitude = Lat;
 Longitude = Lon;

 WF_date = str2double(WF_datec);
 WS_data_resampled.DayOfYear = 1:1:8760;
 WS_data_resampled.DayOfYear = reshape(WS_data_resampled.DayOfYear,8760,1);
 WS_data_resampled.Second = zeros(8760,1); %Not in EPW therefore zero
 WS_data_resampled.Minute = WF_date(:,5);
 WS_data_resampled.Hour = WF_date(:,4);
 WS_data_resampled.DayOfMonth = WF_date(:,3);
 WS_data_resampled.Month = WF_date(:,2);
 WS_data_resampled.Year = WF_date(:,1);

 WS_data_resampled.RelativeHumidity = WS_data(:,3);
 WS_data_resampled.Precipit = zeros(8760,1);
 WS_data_resampled.CloudCover = WS_data(:,17);
 WS_data_resampled.WindSpeed = WS_data(:,16);
 WS_data_resampled.WindDir = WS_data(:,15);
 WS_data_resampled.DewPoint = WS_data(:,2);
 WS_data_resampled.DryBulb = WS_data(:,1);
 WS_data_resampled.Pressure = WS_data(:,4);

 % Solar calculations https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7085731/

 % Declination angle

 WS_data_resampled.Declination = -0.40927971*sin(2*pi*(WS_data_resampled.DayOfYear+284)/365); % in
radians ASHRAE fundamentals 2021 SI equation 10 from Chpt. 14 page 14.10
                                                                                                   % 23.45
degrees = 0.40927971 radians
 % Local time - in Mexico there are two time zones which cover most of the
 % country UTC -6 Baja California UTC -8 The state of Baja California Sur, Sonora and
 % Chihuahua uses UTC -7. Quintana Roo uses UTC -5.

 %ASHRAE fundamentals 2021 SI equations 5 & 6 from Chpt. 14 page 14.9
 %Equation of time

 B = (WS_data_resampled.DayOfYear-1)*2*pi/365; % B is in radians - Equation 6 Chpt 14 page 14.9 2021
 E = 2.2918*(0.0075+0.1868*cos(B) - 3.2077*sin(B) - 1.4615*cos(2*B) - 4.089*sin(2*B)); % E is in
minutes - Equation 5 Chpt 14 page 14.9 2021

 % ASHRAE handbook 2019 HVAC Applications Chapter 36 Equation 5
 % ET = 9.87 sin 2B ñ 7.53 cos B ñ 1.5 sin B     (5)
 % where B = 0.989(N ñ 81) & N = day of year, with January 1 = 1
 % B = 0.989*(WS_data_resampled.DayOfYear - 81)*2*pi/365; % B is in radians
 % E = 9.87*sin(2*B) - 7.53*cos(B) - 1.5*sin(B);


 Lst = RelativeToUTC*15; % 0 degrees because the raw data files are in UTC  -  but resampled is in
local time zone civil
 % time as are EPW files
 Loc = Longitude; % in degrees

 WS_data_resampled.SolarTime = mod(WS_data_resampled.Hour + (4*(Lst-Loc)-E)/60,24); % In hours
 % According to AHSHRAE Applications 2019 Chpt. 36 Equation 4 - needs equation of time.
 % AST = LST + ET + (4 min) (LST meridian ñ Local longitude)



 WS_data_resampled.HourAngle = (WS_data_resampled.SolarTime-12)*pi/12; % In radians
 % Solar altitude angle - equation 12 Chpt 14 page 14.10 2021
 % WS_data_resampled.SolAltAngle = asin(sin(WS_data_resampled.Declination)*sin(Latitude*pi*2/360)...
 %    +cos(WS_data_resampled.Declination).*cos(Latitude*pi*2/360).*cos(WS_data_resampled.HourAngle));
```

```
 % Solar altitude angle from SolarAzEl Programed by Darin C. Koblick
 % 2/17/2009 and Darin C. Koblick 4/16/2013 Vectorized for Speed

 DateVector = zeros (8760,6);
 DateVector(1:8760,1) = WS_data_resampled.Year;
 DateVector(1:8760,2) = WS_data_resampled.Month;
 DateVector(1:8760,3) = WS_data_resampled.DayOfMonth;
 DateVector(1:8760,4) = WS_data_resampled.Hour;
 DateVector(1:8760,5) = WS_data_resampled.Minute;
 DateVector(1:8760,6) = WS_data_resampled.Second;

 if UTCdiffd <= -7.9
     UTCcity = 'America/Tijuana';
 elseif UTCdiffd <= -6.9
     UTCcity = 'America/Hermosillo';
 elseif UTCdiffd <= -5.9
     UTCcity = 'America/Mexico_City';
 elseif UTCdiffd <= -4.9
     UTCcity = 'America/Cancun';
 end

 t1 = datetime(DateVector,'TimeZone',UTCcity); % datetime(DateVector,'TimeZone','America/
Mexico_City');  EPW time is local timezone
 t1.TimeZone = 'UTC';
 %jd = juliandate(t1);
 Latvector(1:8760,1) = Latitude;
 Longvector(1:8760,1) = Longitude;
 Altvector(1:8760,1) = Altitude/1000;
 [Az,El] = SolarAzEl(t1,Latvector,Longvector,Altvector);
 WS_data_resampled.SolAltAngle = deg2rad(El);
 %Check declination, solar angle etc. by plotting them over the year -
 %looks alright for Tijuana

 % Relative humidity

 % For relative humidity saturated partial pressure (pws) and dew point partial pressure (pw) is
needed
% The saturation pressure over ice for the temperature range of -100 to 0∞C is given by
% ln pws = C1/T + C2 + C3T + C4T^2 + C5T^3 + C6T^4 + C7 ln T
% where
C1 = -5.6745359e+3;
C2 = 6.3925247e00;
C3 = -9.6778430e-3;
C4 = 6.2215701e-7;
C5 = 2.0747825e-9;
C6 = -9.4840240e-13;
C7 = 4.1635019e00;
%
% The saturation pressure over liquid water for the temperature range of 0 to 200∞C is given by
% ln pws = C8/T + C9 + C10T + C11T^2 + C12^T 3 + C13 ln T
% where
%
C8 = -5.8002206e+03;
C9 = 1.3914993e+00;
C10 = -4.8640239e-02;
C11 = 4.1764768e-05;
C12 = -1.4452093e-08;
C13 = 6.5459673E+00;
%
% In both Equations (5) and (6),
% pws = saturation pressure, Pa
% T = absolute temperature, K = ∞C + 273.15
%
  T = WS_data_resampled.DryBulb+273.15;
  % Then equation 6
  pws = exp(C8./T + C9 + C10.*T + C11*T.^2 + C12*T.^3 + C13*log(T))/1e3;
  INDEX = (WS_data_resampled.DryBulb < 0);
  % Then equation 5
  pws(INDEX) = exp(C1./T(INDEX) + C2 + C3*T(INDEX) + C4*T(INDEX).^2 + C5*T(INDEX).^3 + C6*T(INDEX).^4
+ C7*log(T(INDEX)))/1e3;

    T = WS_data_resampled.DewPoint+273.15;
```

```
  % Then equation 6
  pw = exp(C8./T + C9 + C10.*T + C11*T.^2 + C12*T.^3 + C13*log(T))/1e3;
  INDEX = (WS_data_resampled.DewPoint < 0);
  % Then equation 5
  pw(INDEX) = exp(C1./T(INDEX) + C2 + C3*T(INDEX) + C4*T(INDEX).^2 + C5*T(INDEX).^3 + C6*T(INDEX).^4
+ C7*log(T(INDEX)))/1e3;

 WS_data_resampled.RelativeHumidity = (pw./pws)*100;

 WS_data_resampled.EstimHourlySol = zeros(8760,1);
 WS_data_resampled.EstimDaylySol = zeros(365,24,1);
 WS_data_resampled.Taub = zeros(365,1);
 WS_data_resampled.Taud = zeros(365,1);

 % Day light hours index
 INDEX = (WS_data_resampled.SolAltAngle > 0); %Don't estimate solar radiation at night

 % Use correlations from Development of 3012 IWEC2Weather Files
 % for International Locations (RP-1477) Huang et al ASHRAE Transactions
 % Volume 120 part 1 pp 340 - 355 - Table 2
 % Change the following to make using the coefficients easier

ConstsSol(1,1:7) = [0.97136,0.24936,-0.32165,0.03768,-0.0076,0.00794,-2.23175]; % Af
ConstsSol(2,1:7) = [0.71868,-0.11359,-0.07259,0.01038,-0.00285,0.00866,-8.42023]; % Am
%ConstsSol.As = [0.8089,0.07355,-0.40101,-0.00424,-0.00242,0.00342,-8.395]; % As has the same
criteria as Aw and same the regression coefficients
ConstsSol(3,1:7) = [0.8089,0.07355,-0.40101,-0.00424,-0.00242,0.00342,-8.395]; % Aw
ConstsSol(6,1:7) = [0.68149,-0.04697,-0.2842,0.01726,-0.00081,0.00453,-8.91306]; % BSh
ConstsSol(7,1:7) = [0.57692,-0.08062,-0.24399,0.0261,0.00054,0.00277,-1.21103]; % BSk
ConstsSol(4,1:7) = [0.51315,0.1554,-0.42157,0.01427,-0.00035,0.00469,-9.55426]; % BWh
ConstsSol(5,1:7) = [0.6997,-0.03001,-0.27693,0.01529,-0.00169,0.0051,-9.23071]; % BWk
ConstsSol(14,1:7) = [0.67839,0.03646,-0.39075,0.01359,-0.00148,0.0073,-8.71373]; % Cfa
ConstsSol(12,1:7) = [0.7437,-0.02988,-0.26353,0.02606,-0.00323,-8E-05,-1.97366]; % Cfb
ConstsSol(16,1:7) = [0.7437,-0.02988,-0.26353,0.02606,-0.00323,-8E-05,-1.97366]; % Cfc
ConstsSol(8,1:7) = [0.54698,0.21871,-0.42476,0.0233,-0.00038,0.00183,-9.83335]; % Csa
ConstsSol(9,1:7) = [0.6425,0.06685,-0.35491,0.01935,-0.00111,1E-05,-5.40989]; % Csb
ConstsSol(11,1:7) = [0.68175,0.15988,-0.43455,0.01972,-0.00303,0.00201,-6.67731]; % Cwa
ConstsSol(12,1:7) = [0.65533,-0.00683,-0.13621,0.0324,-0.00252,-0.0032,0.17022]; % Cwb
ConstsSol(25,1:7) = [0.73067,0.04956,-0.32687,0.01269,-0.00298,0.00581,0.30725]; % Dfa
ConstsSol(26,1:7) = [0.69491,-0.10822,-0.22999,0.01232,-0.00091,0.0039,-3.46883]; % Dfb
ConstsSol(27,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dfc
ConstsSol(17,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dsa
ConstsSol(18,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dsb
ConstsSol(19,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dsc
ConstsSol(21,1:7) = [0.66459,0.59148,-0.80274,0.03374,-0.00376,0.01662,-21.1779]; % Dwa
ConstsSol(22,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dwb
ConstsSol(23,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dwc
ConstsSol(24,1:7) = [0.73447,-0.03502,-0.18087,0.01923,-0.00225,-0.00443,-3.61457]; % Dwd
ConstsSol(29,1:7) = [0.77029,0.00687,-0.35561,0.01849,-0.00149,-0.00278,-6.41702]; % ET
ConstsSol(30,1:7) = [0.77029,0.00687,-0.35561,0.01849,-0.00149,-0.00278,-6.41702]; % EF

for i = 1:12
    Tmonth(1,1,i) = mean(WS_data_resampled.DryBulb(Hournumber(i)+1:Hournumber(i+1),1));
    P(1,1,i) = sum(WS_data_resampled.Precipit(Hournumber(i)+1:Hournumber(i+1),1));
end

%[Class, BroadClass] = KoppenGeiger(Tmonth,P,classes);
 %Sc0 = 0.5598;
 %Sc1 = 0.4982;
 %Sc2 = -0.6762;
 %Sc3 = 0.02842;
 %Sc4 = -0.00317;
 %Sc5 = 0.014;
 %Sd = -17.853;
 %Sk = 0.843;


 Class = cell2mat(classes(contains(classes(:,3),KoppClass),1));
 %BroadClass = 3;

 % This is a bodge to not rewrite regression equation code below
```

```
 Sc0 = ConstsSol(Class,1);
 Sc1 = ConstsSol(Class,2);
 Sc2 = ConstsSol(Class,3);
 Sc3 = ConstsSol(Class,4);
 Sc4 = ConstsSol(Class,5);
 Sc5 = ConstsSol(Class,6);
 Sd = ConstsSol(Class,7);
 %Sk = 0.843; %- This is not used in the Huang et al ASHRAE version of the
 %equation and looks to just have been a factor in the Qingyuan
 %paper - the coefficients can just be divided through by Sk

 I0 = 1367.7; %W/m^2 from
 % Use T to hold dry bulb at hour n-3 - use a wrap around on the data at
 % the beginning of the year just to make logical consistancy although
 % these hours will always be nighttime with a negative solar altitude
 % angle and so the data will not be used.
 T(1:3) = WS_data_resampled.DryBulb(8758:8760);
 T(4:8760) = WS_data_resampled.DryBulb(1:8757);

 WS_data_resampled.EstimHourlySol = zeros(8760,1);
 %WS_data_resampled.EstimHourlySol(INDEX)=
(I0*sin(WS_data_resampled.SolAltAngle(INDEX)).*(Sc0+Sc1.*WS_data_resampled.CloudCover(INDEX)./10 ...
 %    +Sc2.*((WS_data_resampled.CloudCover(INDEX)./10).^2)+Sc3.*(WS_data_resampled.DryBulb(INDEX)-
T(INDEX))...
 %    +Sc4.*WS_data_resampled.RelativeHumidity(INDEX)+Sc5.*WS_data_resampled.WindSpeed(INDEX))...
 %    +Sd); %./Sk;

 WS_data_resampled.EstimHourlySol=
(I0*sin(WS_data_resampled.SolAltAngle).*(Sc0+Sc1.*WS_data_resampled.CloudCover./10 ...
     +Sc2.*((WS_data_resampled.CloudCover./10).^2)+Sc3.*(WS_data_resampled.DryBulb - T)...
     +Sc4.*WS_data_resampled.RelativeHumidity+Sc5.*WS_data_resampled.WindSpeed)...
     +Sd);
 % Limits based on extraterrestrial radiation (I0*sin(SolAltAngle)
 clear INDEX;
 INDEX = (WS_data_resampled.EstimHourlySol(:,1) < 0.1*I0*sin(WS_data_resampled.SolAltAngle(:,1)));
 WS_data_resampled.EstimHourlySol(INDEX,1) = 0.1*I0*sin(WS_data_resampled.SolAltAngle(INDEX,1));

 INDEX = (WS_data_resampled.EstimHourlySol(:,1) > 0.9*I0*sin(WS_data_resampled.SolAltAngle(:,1)));
 WS_data_resampled.EstimHourlySol(INDEX,1) = 0.9*I0*sin(WS_data_resampled.SolAltAngle(INDEX,1));

 INDEX = (WS_data_resampled.EstimHourlySol(:,1) < 0);
 WS_data_resampled.EstimHourlySol(INDEX,1) = 0;

 % Need to detect clear days and use ASHRAE clear sky model to smooth out
 % direct/total radiation over the day as per Gueymard & Thevenard
 % Source for tau b & d (beam and diffuse pseudo optical depths) values
 % from ASHRAE data for Mexican sites (40) and interpolated for the sites
 % that are not in the ASHRAE data set.
 %"To improve the hourly solar profiles in the IWEC2
 %weather files, the ASHRAE 2009 Clear Sky Model (Gueymard
 %and Thevenard 2013), or ACSM09 for brevity, is used to
 %recalculate the hourly distribution of global horizontal radiation
 %while leaving the total daily solar radiation computed by
 %the ZHM unchanged. For clear days, defined as those where
 %the aggregate cloud cover during daylight hours is no more
 %than 10 and the cloud cover during any daylight hour is no
 %more than 1, the hourly solar profile from ACSM09 is scaled
 %to the daily total and substituted for the hourly profile from the
 %ZHM." Development of 3012 IWEC2Weather Files for International Locations (RP-1477)
 % Huang et al. Ashrae Transactions, 2014, vol. 120, no 1, 340-355.

 % Direct Normal Solar Radiation from model by Zhang et al (2004): sigmoidal
 % Gompertz function.



 % Reshape estimated hourly solar energy data into daily sets of 24 hours by 365 days
 % Reshape all the weather data



 %Temp8 = zeros(8760,1);
```

```
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.EstimHourlySol(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.EstimHourlySol(1:-RelativeToUTC-1,1);
%WS_data_resampled.EstimDaylySol = reshape(Temp8,[24,365]);
WS_data_resampled.EstimDaylySol = reshape(WS_data_resampled.EstimHourlySol,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.SolarTime(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.SolarTime(1:-RelativeToUTC-1,1);
%WS_data_resampled.SolarTimeDayly = reshape(Temp8,[24,365]);
WS_data_resampled.SolarTimeDayly = reshape(WS_data_resampled.SolarTime,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.Pressure(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.Pressure(1:-RelativeToUTC-1,1);
%WS_data_resampled.PressureDayly = reshape(Temp8,[24,365]);
WS_data_resampled.PressureDayly = reshape(WS_data_resampled.Pressure,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.DryBulb(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.DryBulb(1:-RelativeToUTC-1,1);
%WS_data_resampled.DryBulbDayly = reshape(Temp8,[24,365]);
WS_data_resampled.DryBulbDayly = reshape(WS_data_resampled.DryBulb,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.DewPoint(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.DewPoint(1:-RelativeToUTC-1,1);
%WS_data_resampled.DewPointDayly = reshape(Temp8,[24,365]);
WS_data_resampled.DewPointDayly = reshape(WS_data_resampled.DewPoint,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.WindDir(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.WindDir(1:-RelativeToUTC-1,1);
%WS_data_resampled.WindDirDayly = reshape(Temp8,[24,365]);
WS_data_resampled.WindDirDayly = reshape(WS_data_resampled.WindDir,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.WindSpeed(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.WindSpeed(1:-RelativeToUTC-1,1);
%WS_data_resampled.WindSpeedDayly = reshape(Temp8,[24,365]);
WS_data_resampled.WindSpeedDayly = reshape(WS_data_resampled.WindSpeed,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.CloudCover(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.CloudCover(1:-RelativeToUTC-1,1);
%WS_data_resampled.CloudCoverDayly = reshape(Temp8,[24,365]);
WS_data_resampled.CloudCoverDayly = reshape(WS_data_resampled.CloudCover,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.Precipit(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.Precipit(1:-RelativeToUTC-1,1);
%WS_data_resampled.PrecipitDayly = reshape(Temp8,[24,365]);
WS_data_resampled.PrecipitDayly = reshape(WS_data_resampled.Precipit,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.Declination(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.Declination(1:-RelativeToUTC-1,1);
%WS_data_resampled.DeclinationDayly = reshape(Temp8,[24,365]);
WS_data_resampled.DeclinationDayly = reshape(WS_data_resampled.Declination,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.SolAltAngle(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.SolAltAngle(1:-RelativeToUTC-1,1);
%WS_data_resampled.SolAltAngleDayly = reshape(Temp8,[24,365]);
WS_data_resampled.SolAltAngleDayly = reshape(WS_data_resampled.SolAltAngle,[24,365]);

%Temp8 = zeros(8760,1);
%Temp8(1:8761+RelativeToUTC,1) = WS_data_resampled.HourAngle(-RelativeToUTC:8760,1);
%Temp8(8762+RelativeToUTC:8760,1) = WS_data_resampled.HourAngle(1:-RelativeToUTC-1,1);
%WS_data_resampled.HourAngleDayly = reshape(Temp8,[24,365]);
WS_data_resampled.HourAngleDayly = reshape(WS_data_resampled.HourAngle,[24,365]);

 % Day light hours index
```

```
 INDEXDAYTIME = (WS_data_resampled.SolAltAngleDayly > 0); %Don't estimate solar radiation at night
 INDEXDAYTIME7 = (WS_data_resampled.SolAltAngleDayly > 0.3); %Cutoff at 17 degrees solar angle - for
direct/diffuse split below this angle considered 100% diffuse.
                                                            %Direct/diffuse split for cos solar
angle >0.3
 WS_data_resampled.DayAverageCldCvr = zeros(24,365);
 WS_data_resampled.DaylightCloudCover(1:24,1:365) = nan;
 WS_data_resampled.DaylightCloudCover(INDEXDAYTIME) =
WS_data_resampled.CloudCoverDayly(INDEXDAYTIME);
 WS_data_resampled.DayAverageCldCvr(1,1:365) = sum(WS_data_resampled.DaylightCloudCover,'omitnan');
%Total cloud cover during daylight hours in column 1
 WS_data_resampled.DayAverageCldCvr(2,1:365) = max(WS_data_resampled.DaylightCloudCover); %Maximum
cloud cover at any point of daylight hours for a given day in column 2.
 INDEXCLEARDAY = (WS_data_resampled.DayAverageCldCvr(1,:) < 10 &
WS_data_resampled.DayAverageCldCvr(2,:) <= 1);
 WS_data_resampled.DayAverageCldCvr(3,1:365) = sum(WS_data_resampled.EstimDaylySol); %Total daily
solar energy estimated from ZHM model in column 3.

 % Interpolate values for Taub and Taud for ASHRAE clear sky model using
 % the 'nearest' parameter - it is likely there are very small difference in
 % latitude and longitude between the values for the sites from the raw
 % data source and the ASHRAE data table - for the 49 sites which ASHRAE
 % provides data, it is likely that there is a one to one correspondance
 % with raw data sites. For any other site the values of the nearest
 % available site is used.


 taub12 = zeros(12,1);
 taud12 = zeros(12,1);
 taub365 = zeros(365,1);
 taud365 = zeros(365,1);
 taub397 = zeros(397,1);
 taud397 = zeros(397,1);

 QLatLong = [Latitude,Longitude];

     k = dsearchn(LatLong,QLatLong); %Nearest point search k = dsearchn(X,XI) performs the search
without using a triangulation.
                                        %With large X and small XI, this approach is faster and uses
much less memory.
     taub12 = taub(k,:);
     taud12 = taud(k,:);
     taub14(1) = taub12(12);
     taub14(2:13) = taub12;
     taub14(14) = taub12(1);
     taud14(1) = taud12(12);
     taud14(2:13) = taud12;
     taud14(14) = taud12(1);
 tau14x = [1,32,63,91,122,152,183,213,244,275,315,336,366,397];

 DaysOfTheYear = (1:397);
 taub397 = interp1(tau14x,taub14,DaysOfTheYear,'linear');
 taud397 = interp1(tau14x,taud14,DaysOfTheYear,'linear');
 taub365 = taub397(12:376);
 taud365 = taud397(12:376);

 % Air mass from equation 16 Chpt. 14 ASHRAE Fundamentals Handbook 2021 SI

 WS_data_resampled.Airmass(1:24,1:365) = zeros(24,365);
 WS_data_resampled.Airmass(1:24,1:365) = nan;
 WS_data_resampled.Airmass(INDEXDAYTIME) = 1.0./
(sin(WS_data_resampled.SolAltAngleDayly(INDEXDAYTIME))...
     +0.50572.*((6.07995+WS_data_resampled.SolAltAngleDayly(INDEXDAYTIME).*360./
(2.*pi)).^(-1.6364)));

 % Clear sky solar radiation: equations 19, 20, 17 & 18 Chpt. 14 ASHRAE Fundamentals Handbook 2021 SI
 % Equation 19 & 20 Air mass exponents ab and ad are correlated to taub and taud through
 % the following empirical relationships:

 ab = zeros(24,365);
 ad = zeros(24,365);
 E0 = zeros(24,365);
```

```
 taub24365 = zeros(24,365);
 taud24365 = zeros(24,365);
 WS_data_resampled.DailyEb = zeros(24,365);
 WS_data_resampled.DailyEd = zeros(24,365);
 WS_data_resampled.DailyEbh = zeros(24,365);
 WS_data_resampled.DailyASHRAEHoriz = zeros(24,365);
 WS_data_resampled.DailyZHMmntAvg = zeros(24,12); % Averages for each hour of the day for each month

 for Idaytau = 1:365
     for Ihourtau = 1:24
         ab(Ihourtau,Idaytau) = 1.454 -0.406*taub365(Idaytau) - 0.268*taud365(Idaytau) +
0.021*taub365(Idaytau)*taub365(Idaytau);
         ad(Ihourtau,Idaytau) = 0.507 -0.205*taub365(Idaytau) - 0.080*taud365(Idaytau) +
0.190*taub365(Idaytau)*taub365(Idaytau);
         % Extraterrestrial Solar Radiation Equation 4 Chpt. 14 ASHRAE
         % Fundamentals Handbook 2021 SI in W/m^2
         E0(Ihourtau,Idaytau) = 1367*(1+0.033*cos(2*pi*(Idaytau-3)/365));
         taub24365(Ihourtau,Idaytau) = taub365(Idaytau);
         taud24365(Ihourtau,Idaytau) = taud365(Idaytau);
     end
 end
 CLEARDAYINDEX = reshape(INDEXCLEARDAY,[365,1]);

 ab(~INDEXDAYTIME)= nan;
 E0(~INDEXDAYTIME)= nan;
 taub24365(~INDEXDAYTIME)= nan;
 taud24365(~INDEXDAYTIME)= nan;
 WS_data_resampled.DailyEb(~INDEXDAYTIME)= nan;
 WS_data_resampled.DailyEd(~INDEXDAYTIME)= nan;
 WS_data_resampled.DailyEbh(~INDEXDAYTIME)= nan;
 WS_data_resampled.DailyASHRAEHoriz(~INDEXDAYTIME)= nan;

 WS_data_resampled.DailyEb(INDEXDAYTIME) = E0(INDEXDAYTIME).*exp(-
taub24365(INDEXDAYTIME).*WS_data_resampled.Airmass(INDEXDAYTIME).^(ab(INDEXDAYTIME))); %Beam
 WS_data_resampled.DailyEd(INDEXDAYTIME) = E0(INDEXDAYTIME).*exp(-
taud24365(INDEXDAYTIME).*WS_data_resampled.Airmass(INDEXDAYTIME).^(ad(INDEXDAYTIME))); %Diffuse on
horizontal surface

 % Beam on horizontal surface

 WS_data_resampled.DailyEbh(INDEXDAYTIME)= WS_data_resampled.DailyEb(INDEXDAYTIME).*cos(pi/2-
WS_data_resampled.SolAltAngleDayly(INDEXDAYTIME));

 WS_data_resampled.DailyASHRAEHoriz(INDEXDAYTIME)= WS_data_resampled.DailyEbh(INDEXDAYTIME)
+WS_data_resampled.DailyEd(INDEXDAYTIME);

 WS_data_resampled.DayAverageCldCvr(4,1:365) = sum(WS_data_resampled.DailyASHRAEHoriz,'omitnan');
%Total daily beam solar energy from ASHRAE clear sky model on horizontal surface in column 4.
 WS_data_resampled.DayAverageCldCvr(5,1:365) = sum(WS_data_resampled.DailyEd,'omitnan'); %Total daily
diffuse solar energy on horizontal surface from ASHRAE clear sky model in column 5.
 WS_data_resampled.DayAverageCldCvr(6,1:365) = sum(WS_data_resampled.EstimDaylySol); %ZHM daily
totals in column 6.
 WS_data_resampled.DayAverageCldCvr(7,1:365) = WS_data_resampled.DayAverageCldCvr(6,1:365)./
WS_data_resampled.DayAverageCldCvr(4,1:365); % ZHM daily total/ASHRAE clear sky daily total
 WS_data_resampled.DayAverageCldCvr(8,1:365) =
WS_data_resampled.DayAverageCldCvr(4,1:365)+WS_data_resampled.DayAverageCldCvr(5,1:365); % ASHRAE
clear sky daily total on horizontal surface in column 8

 %Clear days hourly ASHRAE scaled to ZHM daily totals
 WS_data_resampled.DailyClearScaledToZHM = nan(24,365);
 WS_data_resampled.DailyClearScaledToZHM(1:24,CLEARDAYINDEX) =
WS_data_resampled.DailyASHRAEHoriz(1:24,CLEARDAYINDEX).*WS_data_resampled.DayAverageCldCvr(7,INDEXCLE
ARDAY);

 % Cloudy days - needs ZHM monthly average profile (hour by hour) - then
 % scale the ASHRAE clear sky to the monthly average. Hourly difference
 % between that result and hourly monthly

 % Set month days' indecies

 INDEXJANDAYS = false(365,1);
 INDEXJANDAYS (1:31) = true;
```

```
%INDEXmonthDAYS (1,1:31) = true;
INDEXFEBDAYS = false(365,1);
INDEXFEBDAYS (32:59) = true;
%INDEXmonthDAYS (2,32:59) = true;
INDEXMARDAYS = false(365,1);
INDEXMARDAYS (60:90) = true;
%INDEXmonthDAYS(3,60:90) = true;
INDEXAPRDAYS = false(365,1);
INDEXAPRDAYS (91:120) = true;
%INDEXmonthDAYS (4,91:120) = true;
INDEXMAYDAYS = false(365,1);
INDEXMAYDAYS (121:151) = true;
%INDEXmonthDAYS (5,121:151) = true;
INDEXJUNDAYS = false(365,1);
INDEXJUNDAYS (152:181) = true;
%INDEXmonthDAYS (6,152:181) = true;
INDEXJULDAYS = false(365,1);
INDEXJULDAYS (182:212) = true;
%INDEXmonthDAYS (7,182:212) = true;
INDEXAUGDAYS = false(365,1);
INDEXAUGDAYS (213:243) = true;
%INDEXmonthDAYS (8,213:243) = true;
INDEXSEPDAYS = false(365,1);
INDEXSEPDAYS (244:273) = true;
%INDEXmonthDAYS (9,244:273) = true;
INDEXOCTDAYS = false(365,1);
INDEXOCTDAYS (274:304) = true;
%INDEXmonthDAYS (10,274:304) = true;
INDEXNOVDAYS = false(365,1);
INDEXNOVDAYS (305:334) = true;
%INDEXmonthDAYS (11,305:334) = true;
INDEXDECDAYS = false(365,1);
INDEXDECDAYS (335:365) = true;
%INDEXmonthDAYS (12,335:365) = true;

CLEARDAYINDEX = reshape(INDEXCLEARDAY,[365,1]);
WS_data_resampled.EstimDaylySol(~INDEXDAYTIME)= nan;
WS_data_resampled.DailyZHMmntAvg = zeros(24,12);
WS_data_resampled.DailyZHMmntAvg(:,1) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXJANDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,2) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXFEBDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,3) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXMARDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,4) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXAPRDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,5) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXMAYDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,6) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXJUNDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,7) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXJULDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,8) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXAUGDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,9) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXSEPDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,10) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXOCTDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,11) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXNOVDAYS&~CLEARDAYINDEX),2,'omitnan');
WS_data_resampled.DailyZHMmntAvg(:,12) =
mean(WS_data_resampled.EstimDaylySol(:,INDEXDECDAYS&~CLEARDAYINDEX),2,'omitnan');

%Daily totals from DailyZHMmntAvg
WS_data_resampled.DailyZHMmntAvgTot = zeros(12,1);
for i = 1:12
    WS_data_resampled.DailyZHMmntAvgTot(i) = sum(WS_data_resampled.DailyZHMmntAvg(:,i),'omitnan');
end
%Scale ASHRAE clear sky to monthly average ZHM total on a horizontal
%surface on cloudy days

WS_data_resampled.DailyCldyZHMratioASH = nan(24,365);
```

```
 %WS_data_resampled.DailyCldyAHSminusZHMhrly = nan(24,365);

 for i = 1:24
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXJANDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(1)./
WS_data_resampled.DayAverageCldCvr(8,INDEXJANDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXFEBDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(2)./
WS_data_resampled.DayAverageCldCvr(8,INDEXFEBDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXMARDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(3)./
WS_data_resampled.DayAverageCldCvr(8,INDEXMARDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXAPRDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(4)./
WS_data_resampled.DayAverageCldCvr(8,INDEXAPRDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXMAYDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(5)./
WS_data_resampled.DayAverageCldCvr(8,INDEXMAYDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXJUNDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(6)./
WS_data_resampled.DayAverageCldCvr(8,INDEXJUNDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXJULDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(7)./
WS_data_resampled.DayAverageCldCvr(8,INDEXJULDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXAUGDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(8)./
WS_data_resampled.DayAverageCldCvr(8,INDEXAUGDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXSEPDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(9)./
WS_data_resampled.DayAverageCldCvr(8,INDEXSEPDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXOCTDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(10)./
WS_data_resampled.DayAverageCldCvr(8,INDEXOCTDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXNOVDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(11)./
WS_data_resampled.DayAverageCldCvr(8,INDEXNOVDAYS&~CLEARDAYINDEX);
     WS_data_resampled.DailyCldyZHMratioASH(i,INDEXDECDAYS&~CLEARDAYINDEX) =
WS_data_resampled.DailyZHMmntAvgTot(12)./
WS_data_resampled.DayAverageCldCvr(8,INDEXDECDAYS&~CLEARDAYINDEX);
 end

 WS_data_resampled.DailyCldyASHScaled = nan(24,365);
 WS_data_resampled.DailyCldyASHScaled =
WS_data_resampled.DailyASHRAEHoriz(:,:).*WS_data_resampled.DailyCldyZHMratioASH(:,:);

WS_data_resampled.DailyCldyASHminusZHMhrly = nan(24,365);


  for i = 1:24
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXJANDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXJANDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,1);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXFEBDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXFEBDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,2);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXMARDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXMARDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,3);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXAPRDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXAPRDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,4);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXMAYDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXMAYDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,5);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXJUNDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXJUNDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,6);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXJULDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXJULDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,7);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXAUGDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXAUGDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,8);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXSEPDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXSEPDAYS&~CLEARDAYINDEX)-WS_data_resampled.DailyZHMmntAvg(i,9);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXOCTDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXOCTDAYS&~CLEARDAYINDEX)-
WS_data_resampled.DailyZHMmntAvg(i,10);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXNOVDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXNOVDAYS&~CLEARDAYINDEX)-
```

```
WS_data_resampled.DailyZHMmntAvg(i,11);
     WS_data_resampled.DailyCldyASHminusZHMhrly(i,INDEXDECDAYS&~CLEARDAYINDEX) =
WS_data_resampled.EstimDaylySol(i,INDEXDECDAYS&~CLEARDAYINDEX)-
WS_data_resampled.DailyZHMmntAvg(i,12);
  end
     WS_data_resampled.DailyCldySmoothed = zeros(24,365);
     WS_data_resampled.DailyCldySmoothed(:,~CLEARDAYINDEX) =
WS_data_resampled.DailyCldyASHScaled(:,~CLEARDAYINDEX)
+WS_data_resampled.DailyCldyASHminusZHMhrly(:,~CLEARDAYINDEX);

 figure (1);
 %set(gcf,'Position',[50,900,750,500]);
 set(gcf,'Position',[10,50,2500,1400]);
 subplot (6,1,1);
 plttemp = reshape(WS_data_resampled.DailyCldySmoothed,[1,8760]);
 plot(plttemp(1:240));
 title ('DailyCldySmoothed');
 ax = gca;
 ax.XLim = [1 240];
 ax.YLim = [-100 max(plttemp)*1.1];


 %figure (3);
 %set(gcf,'Position',[1750,900,750,500]);
 subplot (6,1,3);
 plttemp = reshape(WS_data_resampled.EstimHourlySol,[1,8760]);
 %for i = 1:24
 %    plttempIndex(i,1:365) = CLEARDAYINDEX;
 %end
 %plttempIndexC = reshape(plttempIndex,[1,8760]);
 plot(plttemp(1:240));
 %plot(plttemp(~plttempIndex));
 title ('EstimHourlySol ZHM');
 ax = gca;
 ax.XLim = [1 240];
 ax.YLim = [-100 max(plttemp)*1.1];

 % Direct/diffuse partition using "Zhang et al. (2004) has been used to estimate the direct normal
 % radiation using a sigmoidal Gompertz function where growth
 % is slowest at the start and end of the function (Parton and Innis 1972)"
 % Following "Development of 3012 IWEC2Weather Files for International
 % Locations (RP-1477)" Huang et al.

 A1 = zeros (24,365);
 A2 = zeros (24,365);
 A3 = zeros (24,365);
 A4 = 2.99;
 SinH = zeros (24,365);
 SinH = max(sin(WS_data_resampled.SolAltAngleDayly),0.2);  %Zhang et al. (2004) "When Kt is smaller
than 0.2, the direct radiation Kn is almost zero"
 %"Therefore values smaller than 0.3 were excluded in the modelling
 % process."
 A1(INDEXDAYTIME) = -0.1556*SinH(INDEXDAYTIME).*SinH(INDEXDAYTIME) + 0.1028.*SinH(INDEXDAYTIME) +
1.3748;
 A2(INDEXDAYTIME) = 0.7973*SinH(INDEXDAYTIME).*SinH(INDEXDAYTIME) + 0.1509*SinH(INDEXDAYTIME) +
3.035;
 A3(INDEXDAYTIME) = 5.4307*SinH(INDEXDAYTIME) + 7.2182;

 WS_data_resampled.DailyTotalHorizontal = WS_data_resampled.DailyCldySmoothed +
max(WS_data_resampled.DailyClearScaledToZHM,0);
 INDEXNANS0 = false(24,365);
 INDEXNANS0 = isnan(WS_data_resampled.DailyTotalHorizontal);
 WS_data_resampled.DailyTotalHorizontal(INDEXNANS0) = 0;
 INDEXNEGATIVES = false(24,365);
 INDEXNEGATIVES = WS_data_resampled.DailyTotalHorizontal < 0;
 WS_data_resampled.DailyTotalHorizontal(INDEXNEGATIVES) = 0;

 subplot (6,1,1);
 plttemp = reshape(WS_data_resampled.EstimHourlySol,[1,8760]);
 plot(plttemp(1:240));
 title ('Global Horizontal Radiation - Raw ZHM');
 ax = gca;
```

```
 ax.XLim = [1 240];
 ax.YLim = [-100 max(plttemp)*1.1];

 WS_data_resampled.Kt = zeros (24,365);
 WS_data_resampled.Kt = (WS_data_resampled.DailyTotalHorizontal./SinH)/I0;

 WS_data_resampled.Knormal = zeros (24,365);
 WS_data_resampled.Knormal = (A1.*A2).^(-((A3.*A2).^(-(A4.*WS_data_resampled.Kt))));

 WS_data_resampled.DailyDirectNormal = zeros (24,365);
 WS_data_resampled.DailyDirectNormal7  = zeros (24,365);
 WS_data_resampled.DailyDirectNormal(INDEXDAYTIME) = WS_data_resampled.Knormal(INDEXDAYTIME).*I0;
 WS_data_resampled.DailyDirectNormal7(INDEXDAYTIME7) = WS_data_resampled.Knormal(INDEXDAYTIME7).*I0;

 %figure (2);
 %set(gcf,'Position',[900,900,750,500]);
 subplot (6,1,2);
 WS_data_resampled.DailyCldySmoothed(:,~CLEARDAYINDEX) =
max(WS_data_resampled.DailyCldySmoothed(:,~CLEARDAYINDEX),0);
 %plttemp = reshape(WS_data_resampled.DailyCldySmoothed,[1,8760]);
 plttemp = reshape(WS_data_resampled.DailyTotalHorizontal,[1,8760]);
 plot(plttemp(1:240));
 %plot(WS_data_resampled.DailyCldySmoothed(:,~CLEARDAYINDEX));
 title ('DailyTotalHorizontal');
 ax = gca;
 ax.XLim = [1 240];
 ax.YLim = [-100 max(plttemp)*1.1];


 %figure (4);
 %set(gcf,'Position',[50,300,750,500]);
 subplot (6,1,4);
 %plttemp = reshape(WS_data_resampled.DailyDirectNormal,[1,8760]);
 plttemp1 = reshape(WS_data(1:8760,8),[1,8760]);
 plttemp2 = reshape(WS_data_resampled.DailyTotalHorizontal,[1,8760]);
 %plttemp(2:8760) = plttemp2(1:8759);plttemp(1)=plttemp2(8760);
 % plttemp = reshape(WS_data(1:8760,8),[1,8760])-reshape(WS_data_resampled.DailyTotalHorizontal,
[1,8760]);
 %plttemp = plttemp1-plttemp2;
 plot(plttemp1(4000:4600));
 hold;
 INDEXNANS = false(1,8760);
 INDEXNANS = isnan(plttemp2);
 plttemp2(INDEXNANS) = 0;
 plot(plttemp2(4000:4600));
 %plot(plttemp(1:240));
 title ('Difference EPW global horizontal (J. Huang data - Blue line) - global horizontal this model
(ZHM smoothed - Red line)');
 ax = gca;
 %ax.XLim = [4380 4428];
 ax.YLim = [-100 1400];
 hold;
 ExOne = WS_data(1:8760,8);
 ExTwo = reshape(plttemp2,[8760,1]);

 WhyOne(1:8760,1) = 1:8760;
 CurveOne = [ExOne,WhyOne];
 CurveTwo = [ExTwo,WhyOne];
 Distances = frechetDistance(CurveOne,CurveTwo);

 %fprintf(fid,'%s',file_list(ifile).name);
 %fprintf(fid,'%s',' Frechet Distance ');
 %fprintf(fid,'%g\r\n',Distances);

 Kns = zeros(100,1);
 Kts = zeros(100,1);
 sinhangle = zeros(100,1);

 for i = 1:100
     hourangle = (i-0.99)*pi/100;
     sinhangle(i) = sin(hourangle);
     A1s = -0.1556*sinhangle(i)*sinhangle(i) + 0.1028*sinhangle(i) + 1.3748;
```

```
     A2s = 0.7973*sinhangle(i)*sinhangle(i) + 0.1509*sinhangle(i) + 3.035;
     A3s = 5.4307*sinhangle(i) + 7.2182;
     Kts(i) = 0.35/sinhangle(i);
     Kns(i) =  (A1s*A2s)^(-((A3s*A2s)^(-(A4*Kts(i)))));
 end

%figure (5);
%set(gcf,'Position',[900,300,750,500]);
subplot (6,1,5);
 plttemp = reshape(WS_data_resampled.DailyDirectNormal7,[1,8760]);
 plot(plttemp);
 title ('DirectDailyNormal 7');
 ax = gca;
 ax.XLim = [1 8760];
 ax.YLim = [-100 1400];

%figure (6);
%set(gcf,'Position',[1750,300,750,500]);
%subplot (6,1,6);
%plot(Kts(2:100),sinhangle(2:100));






%         Graph results

 %figure (1);
 %set(gcf,'Position',[50,900,750,500]);
 set(gcf,'Position',[10,50,2500,1400]);
 subplot (6,1,6);
 plttemp = reshape(WS_data(1:8760,8),[1,8760]);
 plot(plttemp(1:240));
 title ('Global Horizontal Radiation From EPW');
 ax = gca;
 ax.XLim = [1 240];
 ax.YLim = [-100 max(plttemp)*1.1];

 figure (2);
 set(gcf,'Position',[10,50,2500,1400]);
 plttemp1 = reshape(WS_data(1:8760,8),[1,8760]);                      % y axis daily from EPW file
 plttemp2 = reshape(WS_data_resampled.DailyTotalHorizontal,[1,8760]); % x axis daily ZHM estimate

 hold
 plot (plttemp2,plttemp1,'linestyle','none','marker','o');
 hold



 JoeMonthHorizTot = zeros(12,1);
 ZHMonthHorizTot = zeros(12,1);

 plttemp1 = reshape(WS_data(1:8760,8),[24,365]);
 plttemp1(plttemp1==0)=NaN;
 plttemp2 = WS_data_resampled.DailyTotalHorizontal;
 plttemp2(plttemp2==0)=NaN;


 JoeMonthHorizTot = zeros(12,1);
 ZHMonthHorizTot = zeros(12,1);

 JoeMonthHorizTot(1,1) = mean(plttemp1(:,INDEXJANDAYS),'all','omitnan');
 JoeMonthHorizTot(2,1) = mean(plttemp1(:,INDEXFEBDAYS),'all','omitnan');
 JoeMonthHorizTot(3,1) = mean(plttemp1(:,INDEXMARDAYS),'all','omitnan');
 JoeMonthHorizTot(4,1) = mean(plttemp1(:,INDEXAPRDAYS),'all','omitnan');
 JoeMonthHorizTot(5,1) = mean(plttemp1(:,INDEXMAYDAYS),'all','omitnan');
 JoeMonthHorizTot(6,1) = mean(plttemp1(:,INDEXJUNDAYS),'all','omitnan');
 JoeMonthHorizTot(7,1) = mean(plttemp1(:,INDEXJULDAYS),'all','omitnan');
 JoeMonthHorizTot(8,1) = mean(plttemp1(:,INDEXAUGDAYS),'all','omitnan');
```

```
 JoeMonthHorizTot(9,1) = mean(plttemp1(:,INDEXSEPDAYS),'all','omitnan');
 JoeMonthHorizTot(10,1) = mean(plttemp1(:,INDEXOCTDAYS),'all','omitnan');
 JoeMonthHorizTot(11,1) = mean(plttemp1(:,INDEXNOVDAYS),'all','omitnan');
 JoeMonthHorizTot(12,1) = mean(plttemp1(:,INDEXDECDAYS),'all','omitnan');

 ZHMonthHorizTot(1,1) = mean(plttemp2(:,INDEXJANDAYS),'all','omitnan');
 ZHMonthHorizTot(2,1) = mean(plttemp2(:,INDEXFEBDAYS),'all','omitnan');
 ZHMonthHorizTot(3,1) = mean(plttemp2(:,INDEXMARDAYS),'all','omitnan');
 ZHMonthHorizTot(4,1) = mean(plttemp2(:,INDEXAPRDAYS),'all','omitnan');
 ZHMonthHorizTot(5,1) = mean(plttemp2(:,INDEXMAYDAYS),'all','omitnan');
 ZHMonthHorizTot(6,1) = mean(plttemp2(:,INDEXJUNDAYS),'all','omitnan');
 ZHMonthHorizTot(7,1) = mean(plttemp2(:,INDEXJULDAYS),'all','omitnan');
 ZHMonthHorizTot(8,1) = mean(plttemp2(:,INDEXAUGDAYS),'all','omitnan');
 ZHMonthHorizTot(9,1) = mean(plttemp2(:,INDEXSEPDAYS),'all','omitnan');
 ZHMonthHorizTot(10,1) = mean(plttemp2(:,INDEXOCTDAYS),'all','omitnan');
 ZHMonthHorizTot(11,1) = mean(plttemp2(:,INDEXNOVDAYS),'all','omitnan');
 ZHMonthHorizTot(12,1) = mean(plttemp2(:,INDEXDECDAYS),'all','omitnan');

 figure (3);
 set(gcf,'Position',[10,50,2500,1400]);
 hold
 plot(ZHMonthHorizTot,JoeMonthHorizTot,'linestyle','none','marker','o');
 ax = gca;
 ax.XLim = [0,1200];
 ax.YLim = [0,1200];
 title ('Global Horizontal Radiation ZHM(x) vs EPW(y) monthly averages IWEC vs IWEC2');
 hold

 for imonth = 1:12
 if ZHMonthHorizTot(imonth) < 100
     file_list(ifile)
 end
 end

end
figure(3);
xlabel('Monthly global horizontal radiation ZHM this program');
ylabel('Monthly global horizontal radiation EPW file');

figure(2);
xlabel('Daily global horizontal radiation ZHM this program');
ylabel('Daily global horizontal radiation EPW file');
ax = gca;
 ax.XLim = [-400,1200];
 ax.YLim = [-400,1200];

 fclose('all');
```

CONFORT TÉRMICO NOCTURNO EN VIVIENDAS DE LA CIUDAD DE VILLAHERMOSA – TABASCO ANTE ESCENARIOS DE CAMBIO CLIMÁTICO

Christopher Lionel Heard*, Sazcha Marcelo Olivera Villarroel**

Resumen:
El estudio analiza el comportamiento de los habitantes de una vivienda típica en Villahermosa, Tabasco, México, frente al aumento de las temperaturas nocturnas. Para ello, se realizan simulaciones en una casa de dos pisos, con viviendas contiguas. En esta región, el clima actual exige el uso de aire acondicionado en las casas unifamiliares. Durante los meses más cálidos, que van de mediados de febrero a mediados de junio, la estrategia habitual es mantener el aire acondicionado encendido a una temperatura estable, sin aumentar la potencia de los equipos. La simulación parte del supuesto de que la temperatura interior nocturna de las viviendas es de 25°C, mientras que la temperatura exterior es superior a la interior, lo que obliga a tener el aire acondicionado funcionando. Para llevar a cabo este estudio, se utilizará el programa ESP-r, una herramienta que permite determinar tanto el confort térmico dentro de la vivienda como la cantidad de energía consumida para lograrlo, incluyendo la potencia de los sistemas de aire acondicionado actuales. Según los modelos y escenarios desarrollados, es probable que el crecimiento de esta región conlleve un aumento significativo en el uso de aires acondicionados en las viviendas unifamiliares sobre todo en horarios nocturnos. Esto, a su vez, se traducirá en un incremento considerable en la demanda y el consumo de energía eléctrica en la región de Tabasco. Esta situación plantea la necesidad de realizar una inversión sustancial en la infraestructura de transmisión y distribución de energía eléctrica en la región. Será necesario transportar energía desde fuentes más lejanas para satisfacer la creciente demanda. El año típico de referencia utilizado en este análisis fue desarrollado por WhiteBox Technologies, basándose en datos históricos horarios recopilados entre 1975 y 1989.

Palabras clave: Cambio climático, confort térmico, aire acondicionado, vivienda, transmisión y distribución eléctrica.

* Doctor en Ingeniería del Medio Ambiente en Edificios en la Universidad de Sheffield.
Correo electrónico: cheard@correo.cua.uam.mx. https://orcid.org/0000-0003-2207-0512
** Doctor en Economía con especialidad en economía de los recursos naturales por la Facultad de Economía de la Universidad Nacional Autónoma de México.
Correo electrónico: satzcha@msn.com. https://orcid.org/0000-0003-1864-7374

Introducción.

Villahermosa, la ciudad clave del sureste mexicano, destaca por su posición estratégica, su pujante desarrollo económico y cultural, y su papel como centro político y administrativo del estado de Tabasco. Aunque las ciudades medianas como Villahermosa no son grandes emisores de gases de efecto invernadero, desempeñan un rol crucial en los procesos de adaptación al cambio climático global. (Vollman, 2019)

Este papel crucial se debe a varios factores. En primer lugar, la creciente urbanización en la región atrae una migración interna cada vez mayor, lo que conlleva un aumento en la demanda de recursos, infraestructura y servicios. Esto a su vez genera mayores emisiones de gases de efecto invernadero a medida que la población crece y las actividades económicas se expanden. Además, la ubicación geográfica de Villahermosa, situada en una zona propensa a inundaciones y fenómenos meteorológicos extremos, la convierte en un punto vulnerable a los impactos del cambio climático (Olivera et al, 2020).

En este contexto, Villahermosa no solo se convertirá en una fuente futura de emisiones, sino también en un centro neurálgico donde se manifestarán los efectos del calentamiento global, como inundaciones, sequías, olas de calor y otros peligros que se espera se intensifiquen. (Satterthwaite, 2009). La ciudad enfrenta, por lo tanto, un doble desafío: mitigar sus propias emisiones y, al mismo tiempo, prepararse para afrontar los impactos del cambio climático. Esto requerirá la implementación de políticas y estrategias integrales que aborden tanto la reducción de gases de efecto invernadero como la mejora de la resiliencia urbana.

Por un lado, Villahermosa deberá implementar medidas de eficiencia energética, fomentar el uso de energías renovables, promover el transporte público y la movilidad sostenible, y adoptar prácticas de economía circular que disminuyan la huella de carbono de la ciudad. Por otro lado, deberá invertir en infraestructura resiliente, mejorar los sistemas de alerta temprana, desarrollar planes de emergencia y contingencia, y fortalecer la capacidad de respuesta de la población ante los impactos climáticos (CEPAL, 2020). Solo a través de este abordaje integral, Villahermosa podrá preservar su desarrollo y contribuir de manera efectiva a la adaptación regional ante los efectos del cambio climático. Esto requerirá la coordinación y el compromiso de los diferentes actores involucrados, desde el gobierno local hasta la ciudadanía y el sector privado, con el fin de construir una ciudad más sostenible, resiliente y preparada para los desafíos futuros.

El clima de la región se caracteriza por ser cálido y húmedo durante gran parte del año, con temperaturas extremas que superan los 35°C. Los máximos diarios suelen rondar alrededor de los 26°C (Información Climatológica por Estado, 2024). Esta situación climática ha generado una alta demanda de aires acondicionados domésticos para mantener el confort térmico de los habitantes. No obstante, si las temperaturas siguieran aumentando como resultado del cambio climático, la potencia requerida de estos sistemas de climatización también se incrementaría considerablemente (Heard y Olivera, 2013).

Esto supondría una carga sustancial para la infraestructura eléctrica de la región, que probablemente no podría ser satisfecha únicamente con paneles fotovoltaicos en techos y sistemas de almacenamiento de energía local. Con el fin de planificar a medio y largo plazo, es importante evaluar el efecto que tendría el aumento de la potencia de los aires acondicionados, en comparación con mantener los niveles actuales de enfriamiento, como una alternativa para lograr el confort térmico deseado. Este análisis permitirá a los responsables de la toma de decisiones tomar medidas informadas y desarrollar estrategias adecuadas para hacer frente a los desafíos energéticos y climáticos futuros (de Buen, et al, 2017).

Además, la creciente demanda de sistemas de climatización también tendrá un impacto en el consumo de energía y las emisiones de gases de efecto invernadero, lo que a su vez podría exacerbar los efectos del cambio climático. Por lo tanto, es crucial explorar soluciones alternativas, como la mejora de la eficiencia energética de los edificios, la promoción de diseños arquitectónicos y urbanos más resilientes al clima, y el desarrollo de tecnologías de climatización más sostenibles, para lograr un equilibrio entre la necesidad de confort térmico y la preservación del medio ambiente. (Mendoza, 2016)

El crecimiento tan significativo del uso de energía eléctrica para confort térmico tiene implicaciones de gran relevancia que lo convierten en un tema prioritario en la agenda pública. Por un lado, la infraestructura eléctrica se ha visto profundamente impactada por este fenómeno. Hoy en día, el creciente equipamiento de aparatos de climatización en los hogares y el desarrollo inmobiliario asociado al sector servicios a lo largo y ancho del país han provocado que la demanda de energía para refrigeración y climatización sea el principal factor en el consumo eléctrico nacional, no sólo de manera regional o estacional, sino presente incluso en zonas y épocas tradicionalmente fuera de la época de calor y de las regiones con climas más cálidos.

Este aumento desmesurado en el consumo eléctrico para climatización ha tenido también fuertes repercusiones económicas a nivel de los hogares mexicanos. Tal como se mencionaba anteriormente, bajo una definición de pobreza energética que establece que un hogar se encuentra en esta situación cuando gasta más del 10% de sus ingresos en energía, en México cerca del 11% del total de los hogares, es decir, alrededor de 3.5 millones de familias, se encuentran en esta precaria situación. Esto demuestra que el impacto del uso de electricidad para confort térmico, especialmente en zonas de clima cálido, tiene un peso económico muy significativo para una proporción considerable de la población (CONUEE, 2020).

Más allá de los retos en materia de infraestructura eléctrica y de las consecuencias económicas para los hogares, este fenómeno también plantea importantes desafíos ambientales. El aumento en el consumo energético destinado a la climatización conlleva un incremento sustancial en las emisiones de gases de efecto invernadero, lo cual contribuye al agravamiento del cambio climático. Esto demanda el diseño e implementación urgente de políticas públicas integrales que promuevan la eficiencia energética, el uso de tecnologías limpias y el desarrollo de alternativas sostenibles para el acondicionamiento térmico de edificaciones.

El problema de la pobreza energética en México es un asunto complejo que tiene implicaciones en diversos ámbitos. La situación es particularmente grave en la frontera norte del país, donde cerca del 18% de los hogares tienen gastos energéticos que superan el 10% de sus ingresos, lo cual se debe a la necesidad de mantener el confort térmico en climas cálidos (CONUEE, 2020).

Este fenómeno representa una carga significativa para las finanzas públicas, ya que los subsidios al consumo de electricidad para el confort térmico de estos usuarios ascienden a más de 40 mil millones de pesos anuales. Se estima que, solo por este concepto, el erario tiene que asumir un gasto adicional de 400 millones de pesos cada año. Además, el consumo energético asociado a la climatización de edificios residenciales y comerciales en México tiene un impacto considerable en el medio ambiente. Se calcula que las emisiones de gases de efecto invernadero derivadas de esta actividad superan los 75 millones de toneladas de CO_2 al año, lo que representa aproximadamente el 12% del total de emisiones del país. Peor aún, se proyecta que esta cifra podría incrementarse hasta 6.7 veces para el año 2050 si no se adoptan medidas para abordar este problema. (SEMARNAT, 2022)

Es evidente que la pobreza y el consumo energéticos para confort térmico en México tienen implicaciones sociales, económicas y ambientales que requieren una atención urgente por parte de las autoridades y la sociedad en general. Desarrollar políticas y estrategias integrales que aborden estas cuestiones de manera efectiva y sostenible se convierte en una prioridad crucial para el bienestar y el desarrollo del país.

En el presente trabajo, nos centramos en la necesidad de modelar y desarrollar escenarios de simulación que nos permitan analizar los incrementos en la potencia de uso de aires acondicionados domésticos y sus efectos en la demanda eléctrica de la ciudad de Villahermosa. Para llevar a cabo este estudio, utilizaremos el programa ESP-r, una herramienta que nos posibilita determinar tanto el confort térmico dentro de la vivienda como la cantidad de energía consumida para lograrlo, incluyendo la potencia de uso de los actuales sistemas de aire acondicionado.

La importancia de este análisis radica en la creciente dependencia de la población hacia los sistemas de climatización, especialmente en regiones con climas cálidos. El uso masivo de aires acondicionados ha generado un aumento significativo en la demanda eléctrica, lo que puede repercutir en la estabilidad y eficiencia del sistema energético de la ciudad. Mediante la simulación de diferentes escenarios, podremos evaluar el impacto de estos incrementos en la potencia de uso de los aires acondicionados y plantear estrategias para mitigar el impacto en la red eléctrica. Además, esta investigación también se enfocará en analizar la relación entre el confort térmico de los ocupantes y el consumo energético de los sistemas de climatización. Esto nos permitirá identificar oportunidades de mejora en la eficiencia de los equipos y en las prácticas de uso, con el fin de reducir el consumo de energía sin comprometer el bienestar de los habitantes.

Material y métodos.

Para desarrollar estos modelos de simulación, es crucial generar dos enfoques principales en la creación de escenarios. Por un lado, la simulación del clima futuro, y por otro, la simulación del comportamiento de los edificios a partir de esos escenarios climáticos. Dado que la simulación térmica detallada de edificios requiere datos meteorológicos locales a nivel horario, con parámetros como temperatura, humedad, radiación solar y viento, no se pueden usar directamente los resultados de los modelos regionales de cambio climático, ya que carecen de algunas de las variables necesarias para el proceso de simulación.

Tradicionalmente, la simulación térmica de edificios ha utilizado años representativos de datos horarios derivados de periodos de entre diez y treinta años de mediciones. Una metodología ampliamente aplicada es la construcción de un Año Meteorológico Típico (TMY, por sus siglas en inglés), que se ensambla a partir de meses de datos medidos que se acercan al comportamiento promedio mensual del periodo total. En el marco de este proyecto, se desarrolló el archivo TMY para Villahermosa, el cual sirvió como base para elaborar los años meteorológicos típicos futuros.

Desafortunadamente, México no cuenta con un modelo probabilístico para generar proyecciones climáticas futuras como el que existe en el Reino Unido (Harris et al., 2014). Ante esta limitación, la opción más viable es generar archivos de datos climáticos futuros mediante el ajuste de información histórica (por ejemplo, Años Meteorológicos Típicos o TMY, por sus siglas en inglés) en función de los resultados de modelos regionales de cambio climático.

Existen diversos métodos para llevar a cabo estos ajustes, entre los cuales se encuentra el utilizado para el estándar CIBSE TM49 (Eames, 2016; M. Eames, comunicación personal, 2019). Este método ha sido programado para trabajar con el conjunto de datos CMIP5, publicado por el Programa Mundial de Investigación del Clima (CMIP5 - home, 2022). Actualmente, hay una amplia variedad de enfoques para ajustar los datos TMY y generar estimaciones del comportamiento térmico futuro de los edificios. En este sentido, se puede desarrollar un conjunto de bases de datos ajustadas y ordenadas de acuerdo con el cambio previsto para un parámetro meteorológico específico, como la temperatura.

Esto permitiría realizar una serie de simulaciones térmicas de edificios y producir un rango de posibles impactos del cambio climático sobre el uso de energía o el confort térmico interior. Aquí está una versión más fluida y atractiva del texto original: La probabilidad relativa de cada escenario y su asignación pueden ser, en ocasiones, algo arbitrarias. Una manera alternativa de abordar este desafío sería reducir la cantidad de simulaciones, buscando criterios para seleccionar uno o dos modelos regionales de cambio climático y eligiendo los escenarios más apropiados para la región de estudio.

En este sentido, el trabajo de Sheffield et al. (2013) resulta de gran interés. Estos investigadores evaluaron los resultados obtenidos en los experimentos de simulación climática a escala regional para Norteamérica, incluyendo Centroamérica y México, llevados a cabo en el marco del CMIP5. Su análisis nos brinda una valiosa perspectiva sobre cómo abordar de manera más eficiente y pertinente los desafíos de la modelización climática regional. Al reducir la cantidad de simulaciones y enfocar en los modelos y escenarios más relevantes para una región en particular, podemos obtener proyecciones más precisas y confiables, facilitando la toma de decisiones informadas en materia de adaptación y mitigación del cambio climático. Este enfoque más selectivo y estratégico representa una alternativa prometedora a los enfoques tradicionales, que a menudo se caracterizan por una proliferación de simulaciones y un grado considerable de arbitrariedad en la asignación de probabilidades.

Los modelos de cambio climático regional han demostrado que el modelo MIRCO5 (Watanabe et al., 2010) es el más adecuado para reproducir el comportamiento histórico (hindcasting) en Centroamérica y México. Por eso, se decidió utilizar los resultados mensuales promedio del escenario MIRCO5 rcp45 r1itp1 entre 2006 y 2100 para generar Años Típicos Meteorológicos (TMY) para los años 2020, 2030 y cada década posterior hasta 2100.

En la zona urbana de Villahermosa, el tipo de vivienda más común es la casa de dos pisos, con casas similares a los lados. Se ha considerado que, en el futuro, estas casas usarán iluminación LED y refrigeradores de consumo estándar, ya que es probable que no se utilicen tecnologías menos eficientes que las actuales. Este enfoque permite obtener una visión realista del consumo energético futuro de las viviendas típicas de la región, teniendo en cuenta las proyecciones climáticas más precisas disponibles. De esta manera, se puede planificar mejor la infraestructura energética y las medidas de eficiencia que se necesitarán para satisfacer la demanda a largo plazo.

En el caso de Villahermosa, ubicada a los 17.59° de latitud norte, la situación presenta algunos desafíos interesantes. Durante el verano, la fachada orientada hacia el norte recibe insolación directa al mediodía, mientras que las fachadas sin sombra orientadas hacia el oeste reciben una cantidad considerable de radiación solar directa por la tarde.

Un análisis de la disposición de las calles en Villahermosa revela que muchas de las manzanas en las zonas residenciales tienen calles con ejes norte-sur, lo que resulta en fachadas orientadas hacia el este u oeste. Este patrón plantea consideraciones importantes a la hora de diseñar viviendas eficientes desde el punto de vista energético. La orientación de las edificaciones y la distribución de las calles son factores clave que deben tenerse en cuenta para optimizar el aprovechamiento de la luz solar y minimizar los efectos negativos de la radiación directa, especialmente en climas cálidos como el de Villahermosa. Un enfoque integral que considere estos aspectos puede contribuir a mejorar el confort y la eficiencia energética de las construcciones en esta localidad.

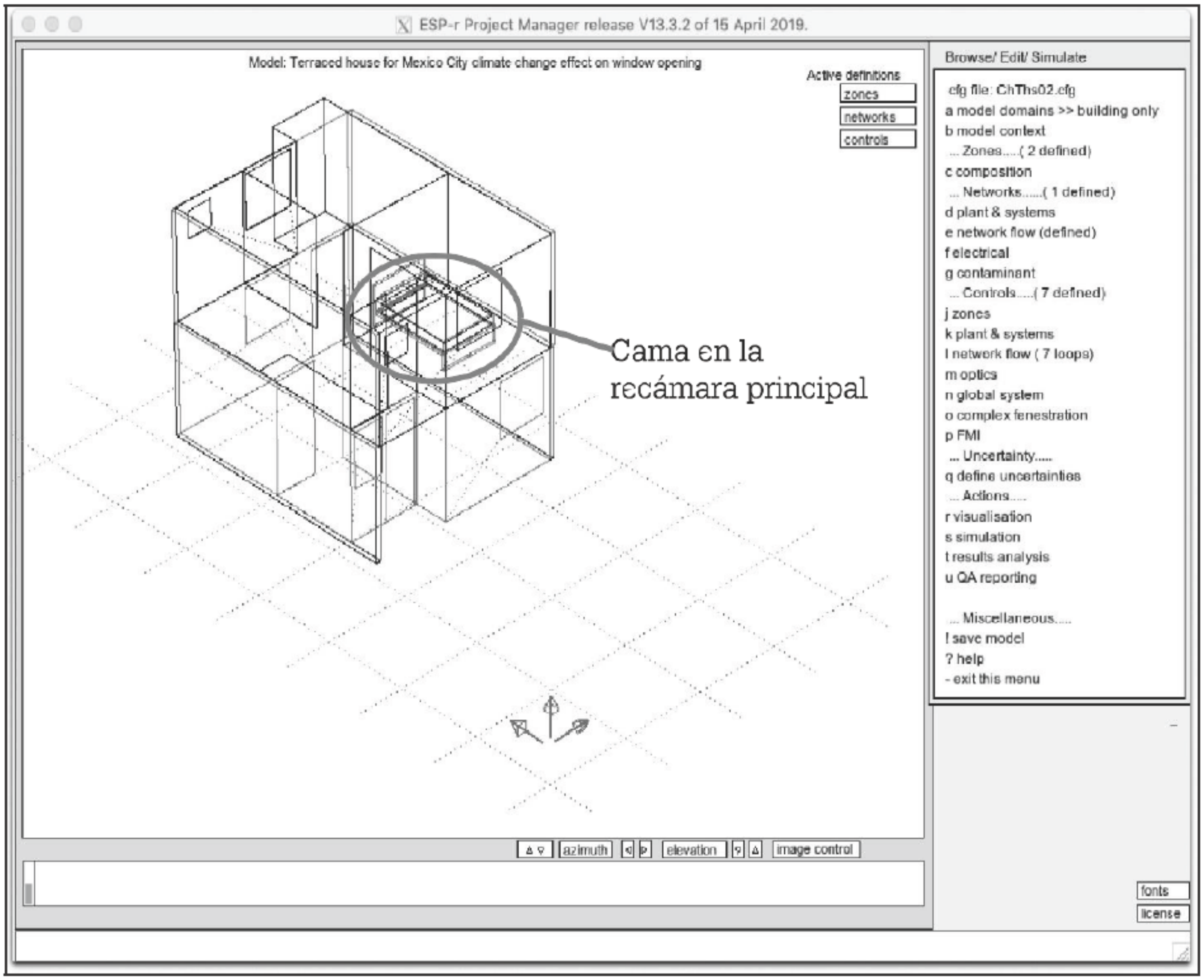

Figura 1. Casa representativa con cama y sensor de ambiente de temperatura de radiación encima de la almohada en la recámara principal.

Modelo de Simulación.

La simulación mediante el programa ESP-r es apropiada para situaciones donde la temperatura de la edificación se controla principalmente a través del uso de sistemas de aire acondicionado o calefacción, y no de forma natural. Este programa permite controlar de manera aproximada la temperatura interior mediante estrategias como la apertura de ventanas, debido a su sistema de uso de volúmenes finitos y conservación de energía (ESP-r Overview, 2022).

La simulación considera que las propiedades de los materiales de construcción, como la densidad, conductividad y capacidad térmicas, se mantienen constantes a lo largo de la simulación. Esto simplifica el modelo, pero puede introducir cierta incertidumbre si estos parámetros varían significativamente en función de las condiciones. Para modelar el flujo de aire a través de grietas y escapes alrededor de ventanas y puertas, el programa utiliza la ecuación 1. Esta ecuación considera factores como la diferencia de presión a través de la abertura y el coeficiente de descarga. Adicionalmente, las ventanas abiertas se modelan como orificios utilizando la ecuación 2, que relaciona el caudal de aire con la diferencia de presión.

La simulación utiliza datos meteorológicos horarios, y realiza cuatro pasos de tiempo por hora tanto para la simulación térmica como del flujo de aire. Esto permite capturar con mayor detalle la dinámica de los intercambios de calor y las variaciones de las condiciones exteriores a lo largo del día. En general, el conjunto de programas ESP-r tiene capacidades y limitaciones bien documentadas, las cuales se pueden encontrar en la reseña realizada por Hand J. (ESP-r Overview, 2022). Estos antecedentes son importantes para entender el alcance y las restricciones del modelo de simulación utilizado.

$$\dot{m} = \rho k \Delta P^{x} \qquad (kg/s)$$
$$donde$$
$$x = 0.5 + 0.5\exp(-500W)$$
$$k = L9.7(0.0092)^{x}/1000$$
$$W = ancho\ de\ la\ grieta\ (mm)$$
$$L = largo\ de\ la\ grieta\ (m)$$

$$\dot{m} = C_d A^2 \sqrt{2\rho\Delta P} \qquad (kg/s)$$
$$donde$$
$$C_d = factor\ de\ descarga = 0.65$$
$$A = \text{área}\ abierta\ \ (m^2)$$

t

El uso generalizado de aires acondicionados ha incrementado significativamente el consumo eléctrico en las zonas residenciales de Villahermosa y otras ciudades cálidas de México. En las últimas décadas, el consumo se ha disparado, llegando a ser 6.6 veces mayor en las áreas de climas cálidos, en contraste con un aumento de solo 3.2 veces en las zonas templadas. Este incremento en el consumo eléctrico se debe principalmente a la creciente necesidad de mantener temperaturas frescas y confortables en el interior de las viviendas durante los meses más calurosos del año. (Baez, 2020;

La mayoría de las viviendas en Villahermosa carecen de un diseño adecuado para soportar las altas temperaturas y ser energéticamente eficientes, lo que conlleva a un uso excesivo e ineficiente de los aires acondicionados. Muchas de estas casas y apartamentos, fueron construidos sin tener en cuenta principios de arquitectura bioclimática, como la orientación de las ventanas, la incorporación de aislamiento térmico o la utilización de materiales autóctonos que ayuden a regular la temperatura interior. Esto genera un aumento considerable en la demanda eléctrica y un mayor impacto ambiental, debido a las emisiones de gases de efecto invernadero provenientes de las plantas de generación eléctrica. (de Buen, et al, 2018)

Si bien se han implementado normativas de eficiencia energética para los equipos de aire acondicionado, lo que ha ayudado a mitigar el crecimiento del consumo eléctrico, aún existe un gran potencial para mejorar la eficiencia energética de las propias viviendas y reemplazar los equipos antiguos por modelos más eficientes. Además, es crucial promover entre la población la importancia de adoptar hábitos de ahorro energético, como mantener una temperatura adecuada en los aires acondicionados y evitar un uso excesivo de estos.

La guía de la CONAVI sobre el uso eficiente de la energía en el hogar subraya la importancia de adquirir aires acondicionados eficientes como una medida clave para ahorrar energía. Además, destaca la relevancia del diseño bioclimático de las viviendas para reducir la necesidad de estos equipos. Esto incluye la incorporación de elementos arquitectónicos como toldos, voladizos, vegetación y ventilación natural, que permiten mantener temperaturas más agradables sin depender tanto de los aires acondicionados.

Modelando el confort térmico en el dormitorio principal.

A la hora de determinar los niveles de vestimenta y actividad de los ocupantes de los dormitorios, nos basamos en los trabajos de Lin y Deng (Lai et al., 2018; Lin & Deng, 2008a). Lin y Deng desarrollaron un modelo de confort térmico específico para recámaras en climas subtropicales, incluyendo medidas de aislamiento para ropa de cama y pijamas (Ecuación 3). Por lo que, para evaluar el confort térmico de la vivienda, se utilizó el "Voto Medio Estimado" (PMV por sus siglas en inglés), definido por Fanger (ASHRAE, 2021). El PMV es una escala que mide la satisfacción de un grupo de personas con las condiciones térmicas que experimentan, en una escala que va de -3 (frío) a +3 (caliente), pasando por neutro (0).

Además, consideramos la "unidad clo" como medida de la resistencia térmica de la ropa, el concepto de "clo" es una unidad de medida clave para comprender el aislamiento térmico proporcionado por la ropa que llevamos puesta. Esta escala nos permite cuantificar con precisión el nivel de confort que nos brindan nuestras prendas en diferentes entornos. Por ejemplo, estar completamente desnudos: en ese caso, tendríamos 0 clo de aislamiento. Por el contrario, un atuendo típico de hombre, con camisa, pantalones, ropa interior, calcetines y zapatos, se ubica en 1 ciclo. Una indumentaria muy abrigada, como ropa de lana, abrigo y sombrero, alcanza valores entre 3 y 4 clo.

Por otro lado, la ropa ligera de verano se encuentra en torno a los 0.5 clo, mientras que un uniforme militar de invierno llega a 1.5 clo. Como puedes ver, la escala de clo nos permite cuantificar con precisión el nivel de aislamiento de nuestras prendas, lo cual es fundamental para evaluar el confort térmico en distintos ambientes.

Esta unidad de medida se ha convertido en una herramienta indispensable para ingenieros, diseñadores y expertos en climatización, quienes la utilizan para optimizar las condiciones de bienestar de las personas en espacios interiores o exteriores. Comprender el clo de nuestra ropa nos ayuda a tomar decisiones más informadas sobre qué vestir en función de las circunstancias. En el estudio 1 clo equivale a 0.155 (m2·K/W) (ASHRAE, 2021).

Así mismo se considera que todos los seres vivos consumen energía para mantener sus funciones vitales. A esta cantidad de energía que gasta un organismo en un período de tiempo determinado se le conoce como tasa metabólica. Puede medirse en unidades como Joules, calorías o kilocalorías por unidad de tiempo, o bien en función del oxígeno que consume o el dióxido de carbono que produce.

La tasa metabólica basal (TMB) o tasa metabólica estándar (TMS) es la medición de la tasa metabólica de un animal en reposo, tranquilo y sin estrés, cuando no está activamente digiriendo alimentos. Según la teoría metabólica de la ecología, la tasa metabólica de un organismo está relacionada con su tamaño corporal y temperatura. Específicamente, se ha observado que la tasa metabólica total (B) se relaciona con la masa corporal (M) siguiendo la ley de Kleiber: B = B0 * M^b, donde B0 es una constante y b un exponente cercano a 3/4. Además, la tasa metabólica aumenta a medida que lo hace la temperatura corporal, siguiendo el factor de Boltzmann: B = B0 * e^(-E/kT), donde E es la energía de activación, T la temperatura absoluta y k la constante de Boltzmann.

De acuerdo con la norma ASHRAE de 2021, esta medida permite comparar la tasa metabólica de organismos de diferente tamaño y temperatura. En el caso del estudio la tasa metabólica se define en términos de producción de calor por unidad de área de la piel, donde 1 met equivale a 58W/m2 (ASHRAE, 2021). Con estos parámetros, pudimos modelar de manera más precisa el confort térmico de la vivienda y las necesidades de sus ocupantes.

$$PMV = 0.0998\left\{40 - \frac{1}{R_t}\left[\left(34.6 - \frac{4.7\bar{\bar{t}}_r + h_c t_a}{4.7 + h_c}\right) + 0.3762(5.52 - p_a)\right]\right\} - 0.0998[0.056(34 - t_a) + 0.692(5.87 - p_a)]$$

Es importante tener en cuenta que los valores de confort térmico en las viviendas modeladas dependen en gran medida de las costumbres y preferencias individuales de los residentes. Por lo tanto, los valores absolutos de PMV (Predicted Mean Vote) no necesariamente reflejan la realidad típica de estas viviendas. Sin embargo, es probable que durante el resto del siglo se produzcan cambios significativos en la distribución de PMV, lo que podría ser un indicador de las consecuencias del cambio climático en México, especialmente en términos del potencial de un mayor confort térmico y un uso más extendido del aire acondicionado en los hogares.

En el caso de los residentes que duermen en los pisos superiores de una vivienda representativa, se han considerado dos escenarios diferentes. En el primero, los ocupantes llevan un pijama ligero con una sábana de algodón ligera doblada para exponer el 52% del cuerpo (1,8 clo), mientras que, en el segundo escenario, se encuentran completamente cubiertos con ropa (2,7 clo). En ambos casos, se asume un nivel de actividad de 0,7 met.

Para el dormitorio principal, se ha colocado un sensor de temperatura de radiación ambiente ligeramente por encima de la superficie de la almohada, con el fin de registrar las condiciones térmicas del espacio de descanso. Por otro lado, se supone que el espacio de la planta baja de la vivienda se ocupa durante un breve período de tiempo por la mañana en los días laborables, y durante todo el día los fines de semana y festivos.

Esta distribución del uso del espacio es relevante para comprender las dinámicas de confort térmico y el potencial de ahorro energético en la vivienda. Aquí una versión más elaborada del pasaje original: Durante la noche, cuando las personas pasan la mayor parte de su tiempo en el hogar, se vuelve fundamental garantizar condiciones de confort adecuadas. Es en este momento del día cuando las necesidades de climatización se hacen más evidentes y relevantes para el bienestar de los ocupantes.

Dado que la vivienda unifamiliar se utiliza primordialmente durante las horas nocturnas, se considera que las condiciones de confort en ese período serían el principal factor motivacional para la adquisición e implementación de unidades de aire acondicionado. Esto se debe a que, por lo general, es durante la noche cuando las personas buscan descansar, relajarse y disfrutar del hogar, por lo que contar con un ambiente térmicamente confortable se vuelve una prioridad. Más allá del uso diurno, las exigencias de climatización se intensifican en la noche, cuando las personas permanecen por períodos prolongados en la vivienda.

Por ello, se entiende que la necesidad de garantizar un clima interior agradable y acorde a las preferencias de los ocupantes durante la noche sería el principal incentivo para la adquisición y uso de unidades de aire acondicionado en las casas unifamiliares.

Resultados.

La figura 2 muestra los valores del PMV de confort térmico para la recámara principal usando la estrategia propuesta poner el ventilador a potencias mínimas. Aunque los resultados indican que se obtienen temperaturas templadas se debe incrementar la potencia de aire acondicionado para finales de siglo para lograr condiciones de confort. Sin embargo, en las ocasiones cuando las condiciones de confort son de incomodidad por calor sería muy difícil reducir el arropamiento a un nivel mucho menos de 1.8 clo. Se puede suponer que bajo tales condiciones el uso de ventiladores y aire acondicionado sería contemplado con una mayor potencia sobre todo si dichas condiciones ocurren en periodos de mayor duración.

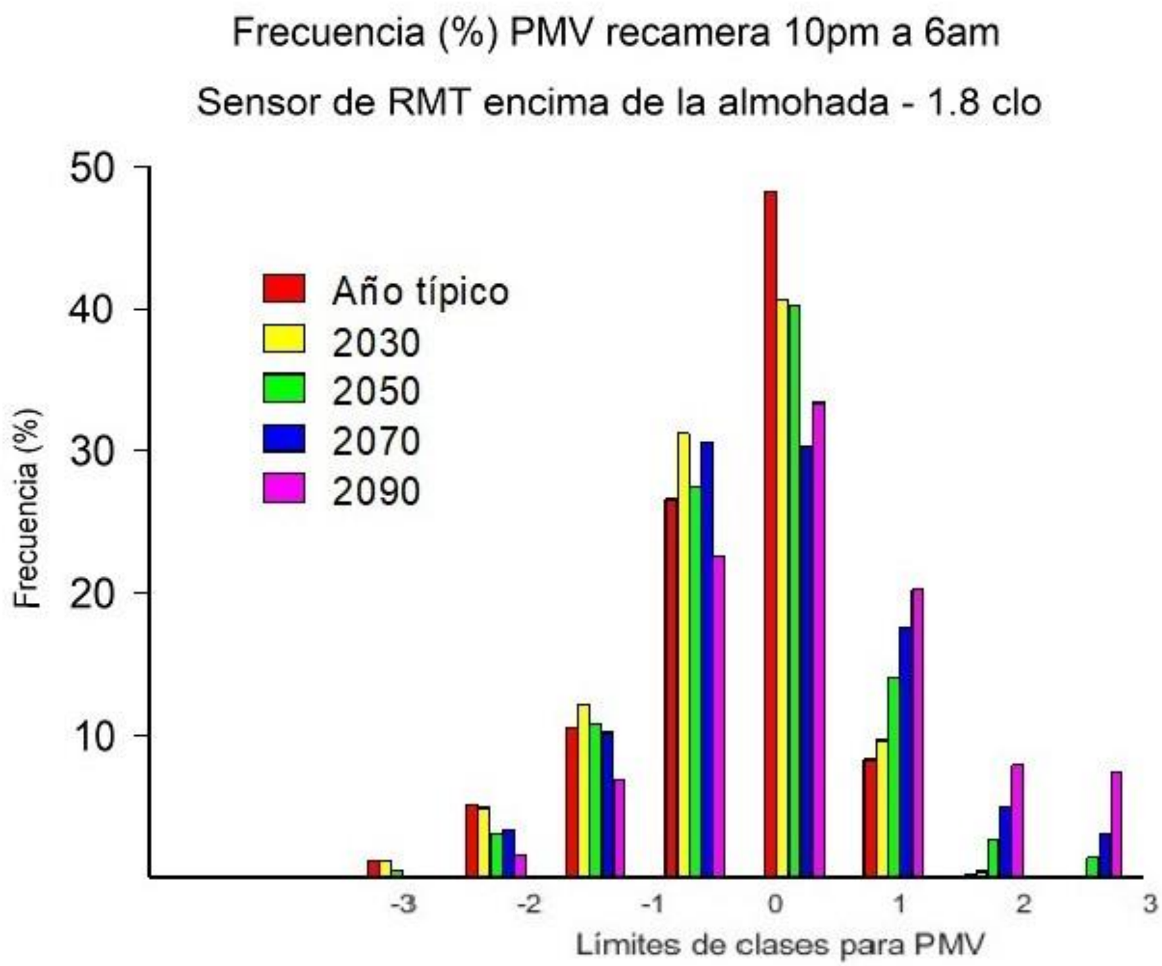

Figura 2. PMV frecuencia entre 22:00 y 06:00 para una recámara con ventana de feb. 15 a jun. 15 y 1.8 clo de arropamiento. Sensor de temperatura medio radiante sobre la almohada de la cama.

La sensación de mayor incomodidad por altas temperaturas en la parte final del siglo será más común. El modelo climático predice un incremento de temperaturas en la década actual pero posteriormente la frecuencia de temperaturas templadas a calorosas incrementa de manera significativamente. Acercándose el final del siglo se puede apreciar que se predicen rachas con incomodidad por calor, mucho más severos, especialmente para vivienda con una orientación desfavorable.

La orientación de la casa sí tiene un impacto sobre la percepción de altos índices de PMV en la recámara. No obstante, esto es una característica del diseño específico de la casa. En este caso la disposición de una ventana con orientación al este sin sombreado externo acumularía en junio radiación solar directa de manera importante, en particular cuando la luz solar cae sobre el piso antes de medio-día y por lo tanto almacenaría calor para su posterior liberación. Durante junio usualmente inicia la estación de lluvias y las tardes frecuentemente son nubladas, lo cual reduce la radiación solar entrando a una ventana con orientación hacia el oeste. Las diferencias en las frecuencias de PMV para las cuatro orientaciones de la casa típica muestran la necesidad para los diseñadores de realizar simulaciones detallados para sus proyectos específicos con datos de escenarios climáticos futuros (Heard et al 2013).

Esto permitiría a los diseñadores ganar una perspicacia en relación con la resiliencia o no de su propuesta de diseño. Sin embargo, los cambios de las distribuciones de las percepciones de PMV son similar para las cuatro orientaciones simuladas. En términos absolutos se encuentra que los PMV indicados caen dentro de los rangos de confort. Tomando en cuenta la estrategia optimista del uso de aire acondicionado en potencia mínima, y considerando que los supuestos de uso de ropa de cama son racionales y que la casa simulada no es particularmente propicia a sobrecalentamiento, es probable que, en la práctica, eventos de calor extremo fomentarán la compra de unidades de aire acondicionado para quienes aún no tengan este dispositivo y una vez instalados su uso típicamente se vuelve habitual y a mayores potencias por las rachas de calor previstas.

Electrodoméstico	Potencia W	Potencia (KW)	Tiempo de uso	Horas mes	**Precio consumo básico**	Precio consumo intermedio	Precio consumo excedente
Ventilador de techo (CB)	103	0.103	8	240	**$29.56**	$35.99	$105.30
Ventilador de torre (CB)	50	0.05	8	240	**$14.35**	$17.47	$51.11
Ventilador de piso	90	0.09	8	240	**$25.83**	$31.45	$92.01
Aire acondicionado de ventana	1700	1.7	8	240	**$487.95**	$593.97	$1,737.88
Mini split	1600	1.6	8	240	**$459.25**	$559.03	$1,635.66
Clima portátil	1400	1.4	8	240	**$401.84**	$489.15	$1,431.20

Tabla 1. ***Consumo de energía eléctrica y Potencia de electrodomésticos***
Fuente: SENER 2016, actualizado por Selectra (2024)

La Tabla 1 nos muestra una clara tendencia: a medida que aumenta la potencia del equipo de enfriamiento, también se incrementa el precio que el usuario final debe pagar. Esto se debe a que equipos más potentes requieren un mayor consumo de energía eléctrica, lo cual conlleva mayores costos de operación para el usuario. Asimismo, cuantas más horas requiera el equipo estar encendido, mayor será el consumo eléctrico y, por ende, el gasto mensual en la factura de luz.

Es importante destacar que la tabla 1 únicamente refleja el promedio de horas de uso en una ciudad típica de México. Este dato puede variar significativamente dependiendo de factores como el clima de la región, los hábitos de uso de los consumidores y la eficiencia energética de los propios equipos de aire acondicionado. Por ejemplo, en zonas con climas más cálidos y húmedos, probablemente los usuarios requieran mantener encendidos los equipos por un mayor número de horas al día, aumentando así su consumo eléctrico. Del mismo modo, aquellos usuarios que estén más conscientes del impacto ambiental y económico del uso excesivo de aire acondicionado tenderán a optimizar los tiempos de uso, buscando un mayor balance entre confort y ahorro.

Al analizar esta información, se evidencia la necesidad de encontrar un equilibrio entre la capacidad de los equipos de aire acondicionado y su consumo energético. Seleccionar la potencia adecuada, así como optimizar los tiempos de uso, puede resultar clave para controlar tanto los costos como el impacto ambiental. Sin embargo, este balance debe considerar también otros factores como el nivel de abrigo o vestimenta (expresado en clo) y la actividad metabólica de los ocupantes (expresada en PVM), los cuales influyen directamente en las necesidades de enfriamiento. De esta manera, se podrá lograr un ambiente interior confortable, eficiente y sostenible en el tiempo.

Mientras que la figura 3 nos muestra la cantidad de horas de uso del aire acondicionado en función de su potencia de refrigeración, la cual va desde el apagado hasta los 8 kW, aumentando de 1 en 1 kW. Al analizar el uso típico actual y los escenarios de cambio climático para 2030, 2050, 2070 y 2090, observamos un incremento constante en las horas de uso del aire acondicionado.

Particularmente, en los escenarios futuros para finales de siglo, veremos que será necesario encender el aire acondicionado incluso en horas de la mañana y en momentos del día en los que actualmente no suele ser necesario. Esto se debe a que los aumentos proyectados en las temperaturas y los niveles de humedad harán que los espacios interiores se vuelvan cada vez más calurosos e incómodos, incluso durante las primeras horas del día, obligando a los usuarios a utilizar los equipos de aire acondicionado por períodos más prolongados.

Además, se aprecia que el equipo se encenderá a niveles de potencia cada vez más cercanos a sus límites de diseño, es decir, por encima de los 5 kW. Esto implica que, a medida que los equipos actuales lleguen al final de su vida útil, tenderán a ser reemplazados por modelos de mayor capacidad de refrigeración, capaces de hacer frente a las condiciones climáticas más exigentes.

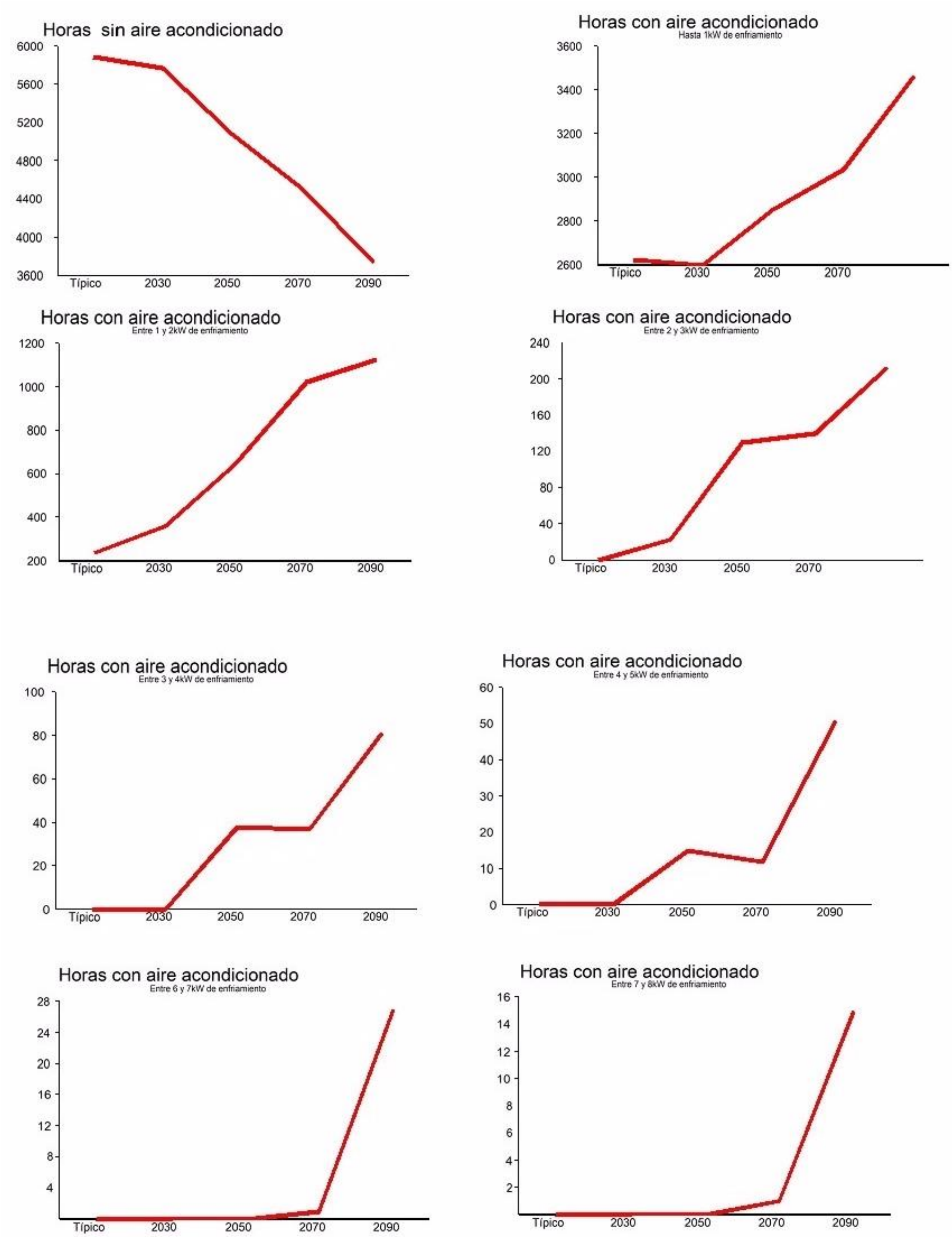

Figura 3. Horas de uso y no uso de aire acondicionado según potencia de enfriamiento

La Tabla 2 revela un preocupante incremento en el uso de aire acondicionado a medida que avanza el cambio climático. Actualmente, el aire acondicionado se utiliza durante 7.68 horas típicamente por la noche, es decir, el 32.70% del día. Sin embargo, se prevé que para finales de siglo esta cifra aumente casi un 57.32%, llegando a 13.75 horas de funcionamiento diario. Esto significa que prácticamente la mitad del día el aire acondicionado estará en pleno funcionamiento. Como se observa en la Figura 3, este aumento en el uso del aire acondicionado también conllevará un incremento notable en el consumo energético de estos equipos.

Uso de aire acondicionado	**Año Típico**	**2030**	**2050**	**2070**	**2090**
Horas sin necesidad de aire acondicionado	67.30%	65.91%	57.86%	51.36%	42.68%
Horas con necesidad de aire acondicionado	32.70%	34.09%	42.14%	48.64%	57.32%

Tabla 2. Horas de uso y no uso de aire acondicionado, expresado en porcentajes

Discusión y Conclusiones.

La combinación de los diferentes escenarios de cambio climático y la simulación de una vivienda típica con uso de aire acondicionado propuesta en este estudio muestra el impacto potencial del cambio climático sobre las condiciones interiores de confort térmico en las viviendas de Villahermosa. Es crucial considerar el confort térmico nocturno, ya que es el periodo más importante para brindar una buena calidad de sueño y el momento en el que los ocupantes pasan más tiempo en sus hogares. Por lo tanto, el confort térmico en los dormitorios se considera el factor decisivo más importante a la hora de adquirir y utilizar unidades de aire acondicionado.

Si bien los valores específicos del índice PMV (Voto Medio Previsto) para una vivienda y sus ocupantes individuales pueden diferir de los resultados presentados aquí, este estudio logró su objetivo de evaluar los cambios potenciales del clima, el PMV y el uso de aire acondicionado a lo largo del siglo. Es evidente que simular una gama más amplia de tipos de vivienda, niveles de aislamiento, vestimenta y actividad de los ocupantes sería de gran utilidad para obtener resultados aún más representativos de la realidad.

Los hallazgos de este estudio generan razones para preocuparse por la percepción de incomodidad térmica sobre todo en el horario nocturno que podría llevar a los habitantes a instalar aire acondicionado doméstico de manera cada vez más generalizada en las extensas áreas urbanas. Este aumento en el uso de sistemas de climatización, si bien puede mejorar las condiciones de confort interior, también conlleva implicaciones en términos de mayor demanda energética, emisiones de gases de efecto invernadero y costos económicos para los hogares. Por lo tanto, es crucial explorar soluciones de diseño y estrategias de adaptación que permitan mitigar los efectos del cambio climático y lograr condiciones de confort térmico interior de manera más sostenible.

El incremento en el uso de aire acondicionado doméstico debido a las altas temperaturas tendrá un impacto significativo en las inversiones requeridas para la transmisión y distribución de energía eléctrica en áreas urbanas como Villahermosa. A medida que las temperaturas extremas se vuelvan más frecuentes e intensas, se producirán eventos de sobrecalentamiento que conllevarán consecuencias importantes para la salud pública y requerirán respuestas urgentes por parte de los gobiernos locales. Este cambio en los hábitos de consumo y en la demanda de equipos más potentes tendrá importantes implicaciones en términos de consumo energético, costos operativos y la necesidad de una mayor inversión en infraestructura eléctrica para satisfacer la creciente demanda de energía

Es fundamental desarrollar programas integrales para prevenir los efectos adversos del cambio climático en la habitabilidad de las viviendas en Villahermosa. Estos programas deberán abordar no solo el aumento en el consumo de energía eléctrica, sino también garantizar el confort térmico y la protección de la salud de los habitantes. Esto implicará la implementación de medidas de eficiencia energética, el uso de materiales y diseños constructivos adaptados al clima, y el desarrollo de soluciones de enfriamiento pasivo y basadas en energías renovables.

Es importante destacar que los resultados del estudio actual se basan en las proyecciones de un solo modelo regional de clima futuro, el cual ha demostrado una buena capacidad para representar las condiciones históricas (hindcasting). Si bien este modelo indica que los cambios serán de una magnitud considerable, existe cierta incertidumbre sobre la exactitud de estas proyecciones. Para tener una mejor comprensión de los posibles escenarios climáticos futuros, sería deseable realizar estudios que incorporen los resultados de una gama más amplia de modelos regionales de cambio climático.

A pesar de esta incertidumbre, los hallazgos presentados en este estudio son suficientes para concluir que es necesario adaptar el diseño de edificios y viviendas unifamiliares en Villahermosa. Esto implicará la adopción de estrategias de diseño bioclimático, la selección de materiales y soluciones constructivas más resilientes al clima, y la integración de sistemas de enfriamiento eficientes y sostenibles. Estas medidas son fundamentales para garantizar la habitabilidad de las viviendas, reducir el consumo energético y proteger la salud de los residentes ante los desafíos que plantea el cambio climático en la región.

Sería valioso ampliar y expandir este tipo de estudio a otras áreas urbanas con climas tropicales a lo largo de la República Mexicana. Analizar y comparar las características y condiciones específicas de diferentes ciudades y regiones con este tipo de clima en todo el país nos permitiría obtener una imagen más integral y enriquecedora. Al examinar en detalle las particularidades de cada localidad, podríamos identificar patrones, retos y oportunidades comunes, así como las diferencias que puedan existir.

Esto nos ayudaría a desarrollar soluciones más acertadas y adaptadas a las necesidades de las distintas comunidades, en lugar de aplicar enfoques genéricos. Un análisis comparativo a escala nacional sin duda aportaría una valiosa perspectiva para entender mejor la dinámica de los entornos urbanos tropicales mexicanos y diseñar estrategias de desarrollo más eficaces y sostenibles. Vale la pena invertir esfuerzos en esta línea de investigación tan relevante para el futuro de nuestras ciudades.

Por ejemplo, se podría estudiar cómo varían los patrones de desarrollo urbano, las dinámicas sociodemográficas, los desafíos ambientales y las soluciones de infraestructura en ciudades como Cancún, Acapulco, Veracruz, Tampico o Mérida, entre otras. Esto permitiría obtener una visión más integral y detallada de la realidad urbana en los entornos cálidos y húmedos de México. Asimismo, sería relevante involucrar a autoridades locales, expertos en planificación y desarrollo, así como a la propia ciudadanía, con el fin de comprender mejor las necesidades, percepciones y aspiraciones de quienes habitan estas regiones tropicales. De este modo, se podrían formular estrategias más efectivas y adaptadas a las particularidades de cada contexto urbano.

Agradecimientos.
El desarrollo de este trabajo se benefició de la infraestructura y los recursos provenientes de dos importantes proyectos de investigación. Por un lado, el proyecto "Cambio climático y su impacto sobre el diseño de vivienda y edificios, y las necesidades de modificación de las NOM-020-ENER-2011 y NOM-008-ENER-2001" (No. 246157), financiado por el Fondo Sectorial CONACyT-SENER-Sustentabilidad Energética. Y por otro, el proyecto "El diseño ante el cambio climático: Divulgación, normatividad e información climatológica", desarrollado por la División de Ciencias de la Comunicación y Diseño de la Unidad Cuajimalpa de la Universidad Autónoma Metropolitana. El acceso a esta infraestructura y recursos ha sido clave para el desarrollo del presente trabajo.

Referencias.

Andersen, R., Fabi, V., Toftum, J., Corgnati, S. P., & Olesen, B. W. (2013). Window opening behavior modeled from measurements in Danish dwellings. Building and Environment, 69, 101-113. https://doi.org/10.1016/j.buildenv.2013.07.005

ASHRAE. (2021). Handbook Fundamentals (SI Edition), Capítulo 9. https://www.ashrae.org/technical-resources/ashrae-handbook

CMIP5 - home. (2022). Llnl.Gov. Retrieved Autumn 11, 2018, from https://esgf-node.llnl.gov/projects/cmip5/

Baez Garcia, W. G. (2020). Desempeño De Modelos De Confort Térmico En Vivienda De Interés Social Ubicada En Clima Cálido De México. https://rinacional.tecnm.mx/jspui/handle/TecNM/1436

CONUEE (2020) En México, las políticas públicas de eficiencia energética han logrado abaratar los servicios energéticos. https://www.gob.mx/conuee/es/articulos/en-mexico-las-politicas-publicas-de-eficiencia-energetica-han-logrado-abaratar-los-servicios-energeticos?idiom=es

de Buen O., Navarrete J., & Jaime E. 2017. Resultados relevantes relativos a eficiencia energética del Módulo Hogares y Medio Ambiente de la Encuesta Nacional de los Hogares de INEGI, 2015. In C. N. p. e. U. E. d. l. Energía (Ed.), Cuadernos: 8. CDMX: Comisión Nacional para el Uso Eficiente de la Energía. https://www.conuee.gob.mx/transparencia/boletines/Cuadernos/cuaderno5nvociclo.pdf

Eames, M. (2019.). [Letter to Christopher Heard].

Eames, M. (2016). An update of the UK's design summer years: Probabilistic design summer years for enhanced overheating risk analysis in building design. *Building Services Engineering Research and Technology*. https://doi.org/10.1177/0143624416631131

ESP-r Overview. (2022). Strath.Ac.Uk. Retrieved Spring 1, 2019, from http://www.esru.strath.ac.uk/Programs/ESP-r_overview.htm

Font, J. G. (2018). Ciudades adaptativas y resilientes ante el cambio climático. Estrategias locales para contribuir a la sostenibilidad urbana. *Revista Aragonesa de Administración Pública*, *52*, 102-158.

Harris, C. N. P., Quinn, A. D., & Bridgeman, J. (2014). The use of probabilistic weather generator information for climate change adaptation in the UK water sector: Climate change adaptation in the water sector using weather generators. Meteorological Applications, 21(2), 129-140. https://doi.org/10.1002/met.1335

Heard, C. L., & Olivera Villarroel, S. M. (2013). Evaluación económica de la resistencia térmica de la vivienda de interés social en las ciudades tropicales de México. *Acta Universitaria*, *23*(4), 17–29. https://doi.org/10.15174/au.2013.458

Información Climatológica por Estado. (2024.). Gob.Mx. Retrieved Summer 6, 2019, from https://smn.conagua.gob.mx/es/informacion-climatologica-por-estado?estado

Jones, R. V., Fuertes, A., Gregori, E., & Giretti, A. (2017). Stochastic behavioural models of occupants' main bedroom window operation for UK residential buildings. Building and Environment, 118, 144-158. https://doi.org/10.1016/j.buildenv.2017.03.033

Lai, D., Jia, S., Qi, Y., & Liu, J. (2018). Window-opening behavior in Chinese residential buildings across different climate zones. Building and Environment, 142, 234-243. https://doi.org/10.1016/j.buildenv.2018.06.030

Lin, Z., & Deng, S. (2008a). A study on the thermal comfort in sleeping environments in the subtropics-developing a thermal comfort model for sleeping environments. Building and Environment, 43(1), 70-81. https://doi.org/10.1016/j.buildenv.2006.11.026

Lin, Z., & Deng, S. (2008b). A study on the thermal comfort in sleeping environments in the subtropics-Measuring the total insulation values for the bedding systems commonly used in the subtropics. Building and Environment, 43 (5), 905-916. https://doi.org/10.1016/j.buildenv.2006.11.026

Markovic, R., Grintal, E., Wölki, D., Frisch, J., & Van Treeck, C. (2018). Window opening model using deep learning methods. Building and Environment, 145, 319-329. https://doi.org/10.1016/j.buildenv.2018.09.024

Mendoza Cárdenas, M. C. (2016). Evaluación del impacto del uso de estrategias de climatización pasiva en el consumo de energía eléctrica de dispositivos de acondicionamiento de aire en viviendas en Monterrey (Doctoral dissertation, Universidad Autónoma de Nuevo León). http://eprints.uanl.mx/14150/

Secretaría de Medio Ambiente y Recursos Naturales, (2022). Plan de Acción en Enfriamiento México. Refrigerantes con Bajo Potencial de Calentamiento Global y Eficiencia Energética en Equipos de Refrigeración y Aire Acondicionado. México. Semarnat. https://www.gob.mx/cms/uploads/attachment/file/775010/PAE_FINAL__1_.pdf

Secretaria de Energía. (2016). Estudio de caracterización del uso de aire acondicionado en vivienda de interés social. Mexico: CONUEE/GIZ. http://www.gob.mx/cms/uploads/attachment/file/164221/Conuee_2016_Estudio_Caracterizaci_n_Uso_Aire_Acondicionado_en_Vivienda.pdf

Sheffield, J., Barrett, A. P., Colle, B., Nelun Fernando, D., Fu, R., Geil, K. L., Hu, Q., Kinter, J., Kumar, S., Langenbrunner, B., Lombardo, K., Long, L. N., Maloney, E., Mariotti, A., Meyerson, J. E., Mo, K. C., David Neelin, J., Nigam, S., Pan, Z., … Yin, L. (2013). North American climate in CMIP5 experiments. Part I: Evaluation of historical simulations of continental and regional climatology. Journal of Climate, 26(23), 9209-9245. https://doi.org/10.1175/JCLI-D-12-00592.1

Vollman, F. (2019). Las ciudades,“causa y solución” del cambio climático. *Noticias ONU*. https://news.un.org/es/story/2019/09/1462322

Watanabe, M., Suzuki, T., O'ishi, R., Komuro, Y., Watanabe, S., Emori, S., Takemura, T., Chikira, M., Ogura, T., Sekiguchi, M., Takata, K., Yamazaki, D., Yokohata, T., Nozawa, T., Hasumi, H., Tatebe, H., & Kimoto, M. (2010). Improved climate simulation by MIROC5: Mean states, variability, and climate sensitivity. Journal of Climate, 23(23), 6312-6335. https://doi.org/10.1175/2010jcli3679.1

Weather Files for Current and Future Climate. (2022). Exeter.Ac.Uk. Retrieved February 6, 2016, from http://www.exeter.ac.uk/media/universityofexeter/research/newsandevents/

```
% Copyright (C) 2018 Christopher Heard
%
% This program is free software: you can redistribute it and/or modify it
% under the terms of the GNU General Public License as published by
% the Free Software Foundation, either version 3 of the License, or
% (at your option) any later version.
%
% This program is distributed in the hope that it will be useful, but
% WITHOUT ANY WARRANTY; without even the implied warranty of
% MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
% GNU General Public License for more details.
%
% You should have received a copy of the GNU General Public License
% along with this program.  If not, see
% <https://www.gnu.org/licenses/>.


% Author: Christopher Heard
% Created: 2018-12-27

% This script is to read in an epw file
% It extracts data from the header to find the latitude and longitude for interpolation of data from
CMIP5 model results
% from a historical file and a future file.
% The year for each month¥s data in the TMY epw file (Typical Meteorological Year - made up of real
months)
%
% EPW data field descriptions taken from: https://bigladdersoftware.com/epx/docs/8-3/auxiliary-
programs/energyplus-weather-file-epw-data-dictionary.html

%Data Field Descriptions
%Descriptions of the fields are taken from the IWEC manual - as descriptive of what should be
contained in the data fields.
%Field:1 Year
%This is the Year of the data. Not really used in EnergyPlus. Used in the Weather Converter program
for display in audit file.
%Field:2 Month
%This is the month (1-12) for the data. Cannot be missing.
%Field:3 Day
%This is the day (dependent on month) for the data. Cannot be missing.
%Field:4 Hour
%This is the hour of the data. (1 - 24). Hour 1 is 00:01 to 01:00. Cannot be missing.
%Field:5 Minute
%This is the minute field. (1..60)
%Field:6 Data Source and Uncertainty Flags
%The data source and uncertainty flags from various formats (usually shown with each field) are
consolidated in the E/E+ EPW format. More is shown about Data Source and Uncertainty in Data Sources/
Uncertainty section later in this document.
%Field:7 Dry Bulb Temperature
%This is the dry bulb temperature in C at the time indicated. Note that this is a full numeric field
(i.e. 23.6) and not an integer representation with tenths. Valid values range from -70 C to 70 C.
Missing value for this field is 99.9.
%Field:8 Dew Point Temperature
%This is the dew point temperature in C at the time indicated. Note that this is a full numeric field
(i.e. 23.6) and not an integer representation with tenths. Valid values range from -70 C to 70 C.
Missing value for this field is 99.9.
%Field:9 Relative Humidity
%This is the Relative Humidity in percent at the time indicated. Valid values range from 0% to 110%.
Missing value for this field is 999.
%Field:10 Atmospheric Station Pressure
%This is the station pressure in Pa at the time indicated. Valid values range from 31,000 to 120,000.
(These values were chosen from the ìstandard barometric pressureî for all elevations of the World).
Missing value for this field is 999999.
%Field:11 Extraterrestrial Horizontal Radiation
%This is the Extraterrestrial Horizontal Radiation in Wh/m2. It is not currently used in EnergyPlus
calculations. It should have a minimum value of 0; missing value for this field is 9999.
%Field:12 Extraterrestrial Direct Normal Radiation
%This is the Extraterrestrial Direct Normal Radiation in Wh/m2. (Amount of solar radiation in Wh/m2
received on a surface normal to the rays of the sun at the top of the atmosphere during the number of
minutes preceding the time indicated). It is not currently used in EnergyPlus calculations. It should
have a minimum value of 0; missing value for this field is 9999.
%Field:13 Horizontal Infrared Radiation Intensity
```

```
%This is the Horizontal Infrared Radiation Intensity in Wh/m2. If it is missing, it is calculated
from the Opaque Sky Cover field as shown in the following explanation. It should have a minimum value
of 0; missing value for this field is 9999.
%Field:14 Global Horizontal Radiation
%This is the Global Horizontal Radiation in Wh/m2. (Total amount of direct and diffuse solar
radiation in Wh/m2 received on a horizontal surface during the number of minutes preceding the time
indicated.) It is not currently used in EnergyPlus calculations. It should have a minimum value of 0;
missing value for this field is 9999.
%Field:15 Direct Normal Radiation
%This is the Direct Normal Radiation in Wh/m2. (Amount of solar radiation in Wh/m2 received directly
from the solar disk on a surface perpendicular to the sunís rays, during the number of minutes
preceding the time indicated.) If the field is ìmissing ( 9999)î or invalid (<0), it is set to 0.
Counts of such missing values are totaled and presented at the end of the runperiod.
%Field:16 Diffuse Horizontal Radiation
%This is the Diffuse Horizontal Radiation in Wh/m2. (Amount of solar radiation in Wh/m2 received from
the sky (excluding the solar disk) on a horizontal surface during the number of minutes preceding the
time indicated.) If the field is ìmissing ( 9999)î or invalid (<0), it is set to 0. Counts of such
missing values are totaled and presented at the end of the runperiod.
%Field:17 Global Horizontal Illuminance
%This is the Global Horizontal Illuminance in lux. (Average total amount of direct and diffuse
illuminance in hundreds of lux received on a horizontal surface during the number of minutes
preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a
minimum value of 0; missing value for this field is 999999 and will be considered missing of >=
999900.
%Field:18 Direct Normal Illuminance
%This is the Direct Normal Illuminance in lux. (Average amount of illuminance in hundreds of lux
received directly from the solar disk on a surface perpendicular to the sunís rays, during the number
of minutes preceding the time indicated.) It is not currently used in EnergyPlus calculations. It
should have a minimum value of 0; missing value for this field is 999999 and will be considered
missing of >= 999900.
%Field:19 Diffuse Horizontal Illuminance
%This is the Diffuse Horizontal Illuminance in lux. (Average amount of illuminance in hundreds of lux
received from the sky (excluding the solar disk) on a horizontal surface during the number of minutes
preceding the time indicated.) It is not currently used in EnergyPlus calculations. It should have a
minimum value of 0; missing value for this field is 999999 and will be considered missing of >=
999900.
%Field:20 Zenith Luminance
%This is the Zenith Illuminance in Cd/m2. (Average amount of luminance at the skyís zenith in tens of
Cd/m2 during the number of minutes preceding the time indicated.) It is not currently used in
EnergyPlus calculations. It should have a minimum value of 0; missing value for this field is 9999.
%Field:21 Wind Direction
%This is the Wind Direction in degrees where the convention is that North=0.0, East=90.0,
South=180.0, West=270.0. (Wind direction in degrees at the time indicated. If calm, direction equals
zero.) Values can range from 0 to 360. Missing value is 999.
%Field:22 Wind Speed
%This is the wind speed in m/sec. (Wind speed at time indicated.) Values can range from 0 to 40.
Missing value is 999.
%Field:23 Total Sky Cover
%This is the value for total sky cover (tenths of coverage). (i.e. 1 is 1/10 covered. 10 is total
coverage). (Amount of sky dome in tenths covered by clouds or obscuring phenomena at the hour
indicated at the time indicated.) Minimum value is 0; maximum value is 10; missing value is 99.
%Field:24 Opaque Sky Cover
%This is the value for opaque sky cover (tenths of coverage). (i.e. 1 is 1/10 covered. 10 is total
coverage). (Amount of sky dome in tenths covered by clouds or obscuring phenomena that prevent
observing the sky or higher cloud layers at the time indicated.) This is not used unless the field
for Horizontal Infrared Radiation Intensity is missing and then it is used to calculate Horizontal
Infrared Radiation Intensity. Minimum value is 0; maximum value is 10; missing value is 99.
%Field:25 Visibility
%This is the value for visibility in km. (Horizontal visibility at the time indicated.) It is not
currently used in EnergyPlus calculations. Missing value is 9999.
%Field:26 Ceiling Height
%This is the value for ceiling height in m. (77777 is unlimited ceiling height. 88888 is cirroform
ceiling.) It is not currently used in EnergyPlus calculations. Missing value is 99999.
%Field:27 Present Weather Observation
%If the value of the field is 0, then the observed weather codes are taken from the following field.
If the value of the field is 9, then ìmissingî weather is assumed. Since the primary use of these
fields (Present Weather Observation and Present Weather Codes) is for rain/wet surfaces, a missing
observation field or a missing weather code implies ìno rainî.


% list of ensamble models and their paths
```

```
ModelFileLocations = [ "C:\Users\usuario\Documents\Christopher\CMIP6files\ACCES-CM2\", ...
                       "C:\Users\usuario\Documents\Christopher\CMIP6files\AWI-CM-1-1MR\", ...
                       "C:\Users\usuario\Documents\Christopher\CMIP6files\EC-Earth3\", ...
                       "C:\Users\usuario\Documents\Christopher\CMIP6files\EC-Earth3-Veg\", ...
                       "C:\Users\usuario\Documents\Christopher\CMIP6files\GFDL-ESM4\", ...
                       "C:\Users\usuario\Documents\Christopher\CMIP6files\MPI-ESM1-2-HR\", ...
                       "C:\Users\usuario\Documents\Christopher\CMIP6files\UKESM1-0-LL\"] ;

HModelFileNameStubs = [ "_Amon_ACCESS-CM2_historical_r1i1p1f1_gn_185001-201412.nc", ...
                        "_Amon_AWI-CM-1-1-MR_historical_r1i1p1f1_gn_185001-201412.nc", ...
                        "_Amon_EC-Earth3_historical_r1i1p1f1_gr_185001-201412.nc", ...
                        "_Amon_EC-Earth3-Veg_historical_r1i1p1f1_gr_185001-201412.nc", ...
                        "_Amon_GFDL-ESM4_historical_r1i1p1f1_gr1_195001-201412.nc", ...
                        "_Amon_MPI-ESM1-2-HR_historical_r1i1p1f1_gn_196501-201412.nc", ...
                        "_Amon_UKESM1-0-LL_historical_r1i1p1f2_gn_185001-201412.nc"] ;

FModelFileNameStubs = [ "_Amon_ACCESS-CM2_ssp245_r1i1p1f1_gn_201501-210012.nc", ...
                        "_Amon_AWI-CM-1-1-MR_ssp245_r1i1p1f1_gn_291501-210012.nc", ...
                        "_Amon_EC-Earth3_ssp245_r1i1p1f1_gr_201501-210012.nc", ...
                        "_Amon_EC-Earth3-Veg_ssp245_r1i1p1f1_gr_201501-210012.nc", ...
                        "_Amon_GFDL-ESM4_ssp245_r1i1p1f1_gr1_201501-210012.nc", ...
                        "_Amon_MPI-ESM1-2-HR_ssp245_r1i1p1f1_gn_201501-210012.nc", ...
                        "_Amon_UKESM1-0-LL_ssp245_r1i1p1f2_gn_201501-210012.nc"] ;


% Load epw file
% Provide location of EPW files
file_location = 'C:\Users\usuario\Documents\Christopher\TMY3s\MEXICO-CITY_766800\'; % Address of EPW
TMY3 files
file_list = dir([file_location,'*TMY3.epw']);
n_files = size(file_list,1);
%run through all of the files
%for ifile = 2:6 %n_files
  ifile=1;
    WF_file = [file_location,file_list(ifile).name];
    WF_data = importdata(WF_file,',',8);

    %extract the information from the header, note due to format the

    header = WF_data.textdata(1:8,1);
    flag_end = size(WF_data.textdata,1);
    WF_date = WF_data.textdata(9:flag_end,1:5); % if matlab
    %WF_date(1:flag_end-8,1:5) = WF_data.textdata(9:flag_end,1:5); %if matlab

    WF_date_TMY(1:8760,2:5) = str2double([WF_date(1:8760,2:5)]); %[b{:}])

    %WF_date = [WF_data.data(1:8760,1:5)]; %, WF_data.textdata(9:flag_end,:)];
    %once extracted the data is all that is left to get.
    %WF_data = WF_data.data; % If Matlab

    WF_data = WF_data.data(:,1:end);

    %extract the important information - this can be edited accordingly - Need to deal with Typical/
Extreme periods and ground temperatures
    C = strsplit(header{1,1},',');
    Lat = str2double(C{7});
    Lon = str2double(C{8});
    Altitude = str2double(C{10});

    % The call to routine that reads in the data from the CMIP 6 models' results for the historical
    % modelling data for each model considered in the ensamble corresponding to the year and month of
TMY-epw file data for the four longitude and
    % latitude points around the site longitude and latitude has been commented out.

    %[Model_historical] = ReadHistorical(WF_date,Lat,Lon);

    % The call to ReadHistoricalFifteenYrAvgMth returns the average values
    % around the year of each month in the TMY file
    %
    numbmodels = length(ModelFileLocations);
    Model_historicalFifteenYrAvg = zeros(numbmodels,12,10);
```

```
    for imodel = 1:numbmodels
        HModelFilePath = strcat(ModelFileLocations(imodel));
        HModelFileNameStub = strcat(HModelFileNameStubs(imodel));

        [Model_historicalFifteenYrAvg(imodel,1:12,1:10)] = ReadHistoricalFifteenYrAvgMth
(WF_date,Lat,Lon, ...
        HModelFilePath,HModelFileNameStub);
    end

    Model_Future = zeros(numbmodels,12,10);

 for iFutureYear = 1:1
      FutureYear = 2050+2*iFutureYear;   %2020+10*iFutureYear ;
      WF_date_TMY(1:8760,1) = FutureYear;

        for imodel = 1:numbmodels
            HModelFilePath = strcat(ModelFileLocations(imodel));
            HModelFileNameStub = strcat(FModelFileNameStubs(imodel));

            [Model_Future(imodel,1:12,1:10)] = ReadFutureYear(FutureYear,Lat,Lon, ...
            HModelFilePath,HModelFileNameStub);
        end

% need to put in averaging of models' results and then do the difference.
% then change the line below to work with the ensemble average values.

    EnsembleHistorical = mean (Model_historicalFifteenYrAvg);
    EnsembleFuture = mean (Model_Future);


    Ensemble_differences = EnsembleFuture - EnsembleHistorical;

    Relative_ensemble_differences = 100.*Ensemble_differences./(EnsembleFuture + EnsembleHistorical);

    WF_data_morphed = WF_data;

    % Morphing has to be done in a specific order since the dewpoint morphing depends on future
values not only of specific humidity,
    % but also the dry bulb temperature and the station air pressure.
    % First: dry bulb temperature and air pressure then dewpoint and relative humidity
    % Then the change in radiation values is applied as a stretch function
    %

    CmltvMnthDyNmbr = [0,31,59,90,120,151,181,212,243,273,304,334,365];


    % morph5_JHa is used to apply a weighted stretch which
    % ensures correct change in tmean, tmin and tmax on a daily basis over the month using weighted
strectch

    % [outdata,m,n,sclcheck,morph_flag] =
morph5_JHa(indata,varname,changefactor1,dTMIN,dTMAX,tmorph_opt)
    varname = 'tas';
    tmorph_opt = 2;

    [EPWhrlydbtFutureMonth2,m,n,sclcheck,morph_flag] = ...

morph5_JHa(WF_data(:,1),varname,Ensemble_differences(1,:,1),Ensemble_differences(1,:,7),Ensemble_diff
erences(1,:,6),tmorph_opt);
    EPWhrlydbtFutureMonth2 = reshape(EPWhrlydbtFutureMonth2,1,8760);

    for i=1:12 % as change factors are derived and applied on a monthly basis the routine is applied
month by month

      %WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,1) - hourly air temperatures

      % TemperatureMorph
      % re-written to use bounded stretching following Eames et al "A
      % revised morphing algorithm for creating future weather for building
      % performance evaluation, Building Services Engineering Research &
      % Technology 2003
```

```
      EPWhrlydbtFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24) = ...
      TemperatureMorph (WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,1),...
      EnsembleHistorical(1,i,6),...
      EnsembleHistorical(1,i,7), EnsembleHistorical(1,i,1),...
      EnsembleFuture(1,i,6), EnsembleFuture(1,i,7), EnsembleFuture(1,i,1));

      %PWhrlypresFutureMonth = PressureMorph (EPW_hourly_pressure_month, PresavgHist, PresavgFuture)

      PWhrlypresFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24) = ...
      PressureMorph (WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,4),...
      EnsembleHistorical(1,i,4),EnsembleFuture(1,i,4),Altitude);

      % The dewpoint and relative humidity use previously morphed dry bulb and pressure data
      % For the morph5_JHa dry bulb morphing a second set of dewpoint and relative humidity is
generated.

      %[EPWhrlyDptFutureMonth, EPWhrlyRHFutureMonth] = DewPointMorph (EPW_hourly_dry_bulb_month,...
      % EPW_hourly_dew_point_month,EPW_hourly_pressure_month, EPW_hourly_relative_humidity_month,...
      % FutureEPW_hourly_pressure_month, FutureEPW_hourly_dry_bulb_month, HussHist, HussFuture)

      [EPWhrlyDptFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24),...
       EPWhrlyRHFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24)] = DewPointMorph (...

WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,1),WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthD
yNmbr(i+1)*24,2),...

WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,4),WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthD
yNmbr(i+1)*24,3),...
       PWhrlypresFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24),...
       EPWhrlydbtFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24),...
       EnsembleHistorical(1,i,2),EnsembleFuture(1,i,2));

       [EPWhrlyDptFutureMonth2(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24),...
       EPWhrlyRHFutureMonth2(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24)] = DewPointMorph (...

WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,1),WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthD
yNmbr(i+1)*24,2),...

WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,4),WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthD
yNmbr(i+1)*24,3),...
       PWhrlypresFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24),...
       EPWhrlydbtFutureMonth2(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24),...
       EnsembleHistorical(1,i,2),EnsembleFuture(1,i,2));


       % EPWhrlyRadFutureMonth = RadiationMorph (EPW_hourly_Rad_month, RadavgHist, RadavgFuture)
       % Horizontal Infrared Radiation Intensity - morphed with
surface_downwelling_longwave_flux_in_air

       EPWhrlyHorizInfraRadFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24) =
RadiationMorph...
       (WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,7), EnsembleHistorical(1,i,10),
EnsembleFuture(1,i,10));

       % Global Horizontal Radiation - morphed with surface_downwelling_shortwave_flux_in_air - put a
limit on the maximum value?

       EPWhrlyHorizShortRadFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24) =
RadiationMorph...
       (WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,8), EnsembleHistorical(1,i,5),
EnsembleFuture(1,i,5));

       % Direct Normal Radiation - morphed with surface_downwelling_shortwave_flux_in_air - put a
limit on the maximum value?

       EPWhrlyNormalShortRadFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24) =
RadiationMorph...
       (WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,9), EnsembleHistorical(1,i,5),
EnsembleFuture(1,i,5));

       % Diffuse Horizontal Radiation - morphed with surface_downwelling_shortwave_flux_in_air - put
```

```
a limit on the maximum value?

       EPWhrlyDiffuseShortRadFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24) =
RadiationMorph...
       (WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,10), EnsembleHistorical(1,i,5),
EnsembleFuture(1,i,5));

       % Radiation fields are 5 through to 14 the extraterrestrial fields 5 and 6 should be
unchanged.
       % fields 11 through to 14 are luminosity fields - left unchanged
       % If there is radiation data present then building thermal modelling programs should not need
to use the cloud cover fields.
       % These are therefore left unchanged.

       % Wind speed and direction

       % [EPWhrlyWndSpdFutureMonth, EPWhrlyWndDirFutureMonth] = WindMorph
(EPW_hourly_wind_speed_month,...
       % EPW_hourly_wind_direction_month, eastward_windHist, northward_windHist, eastward_windFutr,
northward_windFutr)

       [EPWhrlyWndSpdFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24),...
       EPWhrlyWndDirFutureMonth(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24)] = WindMorph (...

WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnthDyNmbr(i+1)*24,16),WF_data(CmltvMnthDyNmbr(i)*24+1:CmltvMnth
DyNmbr(i+1)*24,15),...

EnsembleHistorical(1,i,8),EnsembleHistorical(1,i,9),EnsembleFuture(1,i,8),EnsembleFuture(1,i,9));

    end


    MorphedDryBulbDifs = EPWhrlydbtFutureMonth - transpose(WF_data(1:8760,1));
    MorphedPressureDifs = PWhrlypresFutureMonth - transpose(WF_data(1:8760,4));
    MorphedDewPointDifs = EPWhrlyDptFutureMonth - transpose(WF_data(1:8760,2));
    MorphedRHDifs = EPWhrlyRHFutureMonth - transpose(WF_data(1:8760,3));

    WF_data_morphed = WF_data;
    WF_data_morphed(:,1)= EPWhrlydbtFutureMonth;
    WF_data_morphed(:,4)= PWhrlypresFutureMonth;
    WF_data_morphed(:,2)= EPWhrlyDptFutureMonth;
    WF_data_morphed(:,3)= EPWhrlyRHFutureMonth;
    WF_data_morphed(:,7)= EPWhrlyHorizInfraRadFutureMonth;
    WF_data_morphed(:,8)= EPWhrlyHorizShortRadFutureMonth;
    WF_data_morphed(:,9)= EPWhrlyNormalShortRadFutureMonth;
    WF_data_morphed(:,10)= EPWhrlyDiffuseShortRadFutureMonth;
    WF_data_morphed(:,15)= EPWhrlyWndDirFutureMonth;
    WF_data_morphed(:,16)= EPWhrlyWndSpdFutureMonth;

    WF_data_morphed2 = WF_data;
    WF_data_morphed2(:,1)= EPWhrlydbtFutureMonth2;
    WF_data_morphed2(:,4)= PWhrlypresFutureMonth;
    WF_data_morphed2(:,2)= EPWhrlyDptFutureMonth2;
    WF_data_morphed2(:,3)= EPWhrlyRHFutureMonth2;
    WF_data_morphed2(:,7)= EPWhrlyHorizInfraRadFutureMonth;
    WF_data_morphed2(:,8)= EPWhrlyHorizShortRadFutureMonth;
    WF_data_morphed2(:,9)= EPWhrlyNormalShortRadFutureMonth;
    WF_data_morphed2(:,10)= EPWhrlyDiffuseShortRadFutureMonth;
    WF_data_morphed2(:,15)= EPWhrlyWndDirFutureMonth;
    WF_data_morphed2(:,16)= EPWhrlyWndSpdFutureMonth;


 % Write morphed EPW file
 % File location
      NewFileLocation = 'C:\Users\usuario\Documents\Christopher\TMY3s\MEXICO-CITY_766800\Morphed\';
      NewFileName = [NewFileLocation,file_list(ifile).name(1:end-4),'_',int2str(FutureYear),'.ep7'];
      fid = fopen(NewFileName,'w');
 %print the header first
        for k = 1:size(header,1)
            fprintf(fid,'%s\r\n',header{k});
        end
        for j = 1:size(WF_data_morphed,1)
```

```
            %then the date
            %fprintf(fid,'%g,',WF_date{j,1});
            fprintf(fid,'%g,',FutureYear);
            for k = 2:5
                fprintf(fid,'%g,',WF_date_TMY(j,k));
            end
            fprintf(fid,'?9?9?9?9E0?9?9?9?9*9?9?9?9?9*9?9?9?9?9*_*9*9*9?9?9,');
            %next print the data
            %for k = 1:size(WF_data_morphed,2)-1
                 %fprintf(fid,'%g,',WF_data_morphed(j,k));
                 fprintf(fid,'%.1f,',WF_data_morphed(j,1:2));
                 fprintf(fid,'%.2i,',WF_data_morphed(j,3));
                 fprintf(fid,'%1.0f,',WF_data_morphed(j,4:15));
                 fprintf(fid,'%.1f,',WF_data_morphed(j,16));
                 fprintf(fid,'%1.0f,',WF_data_morphed(j,17:18));
                 fprintf(fid,'%.1f,',WF_data_morphed(j,19));
                 fprintf(fid,'%1.0f,',WF_data_morphed(j,20:23));
                 fprintf(fid,'%.4f,',WF_data_morphed(j,24));
                 fprintf(fid,'%1.0f,',WF_data_morphed(j,25:26));
                 fprintf(fid,'%.3f,',WF_data_morphed(j,27));
                 fprintf(fid,'%.1f,',WF_data_morphed(j,28));
                 %i,%i,%i,%i,%i,%i,%i,%i,%i,%i,%i,%i,%i,%.1f,%i,%i,%.1f,%i,%i,%i,%i,%.4f,%i,%i,%.3f,
%.1f,',...
                 %WF_data_morphed(j,k));
            %end
            fprintf(fid,'%.1f\r\n',WF_data_morphed(j,end));
        end
        %fclose(fid);


end
fclose('all');

% Author: Christopher Heard
% Created: 2024

function EPWhrlydbtFutureMonth = TemperatureMorph (EPW_hourly_dry_bulb_month, TmaxHist, TminHist,
TavgHist,...
   TmaxFuture, TminFuture, TavgFuture)

% EPWhrlydbtFutureMonth and EPW_hourly_dry_bulb_month are arrays of hourly drybulb temperatures for a
month

% Step through days to normalize temperature to the maximum and minimum
% range for each day.

% Find number of days in month

Numbhours = size(EPW_hourly_dry_bulb_month,1);
Numbdays = Numbhours/24;

EPW_hourly_dry_bulb_month_days = reshape(EPW_hourly_dry_bulb_month,[24,Numbdays]);

Xqis = zeros(24,Numbdays);

Daymax = zeros(Numbdays,1);
Daymin = zeros(Numbdays,1);
Daymean = zeros(Numbdays,1);
DeltaXqis = zeros(Numbdays,1);

% https://www.metoffice.gov.uk/research/climate/maps-and-data/data/haduk-grid/datasets
% Maximum air temperature :     tasmax  Average of daily maximum air temperature over the calendar
month, season or year
% https://iridl.ldeo.columbia.edu/
SOURCES/.NASA/.NCCS_Dataserver/.CREATE_REANALYSIS/.MERRA2/.mon/.atmos/.tasmax/?Set-Language=es
% "monthly mean of the daily-maximum near-surface air temperature."

% Similarly for tasmin
% Therefore the average of the daily maximums y minimums and the average
% temperature for the morphed month should correspond to those derived from the
% future regional climate model differences from the historical to the
% future year.
```

```
DeltaTas = TavgFuture - TavgHist;
DeltaTasmax = TmaxFuture - TmaxHist;
DeltaTasmin = TminFuture - TminHist;

Daymean = mean(EPW_hourly_dry_bulb_month_days); % Promedio de las temperaturas en el archivo aÒo base
Daymax = max(EPW_hourly_dry_bulb_month_days); % Temperatura m·xima de cada dÌa del mes del archivo
aÒo base
Daymin = min(EPW_hourly_dry_bulb_month_days); % Temperatura mÌnima de cada dÌa del mes del archivo
aÒo base
%Xqis =(EPW_hourly_dry_bulb_month_days-Daymin)./(Daymax-Daymin);


EPWmax = max(EPW_hourly_dry_bulb_month); % La temperatura m·xima en el mes del archivo del aÒo base
EPWmin = min(EPW_hourly_dry_bulb_month); % La temperatura mÌnima en el mes del archivo del aÒo base
Xqis = (EPW_hourly_dry_bulb_month_days-EPWmin)./(EPWmax-EPWmin); % Valores normalizados respeto a las
temperaturas m·xima y mÌnima del mes

DeltaXqis = DeltaTas./(EPWmax-EPWmin); % Cambio de temperatura promedio entre histÛricas y futuras en
el modelo/s regional de clima normalizado
                                       %  al Max & Min del mes del archivo del aÒo base

%for iday = 1:Numbdays

 %   Daymean(iday)= mean(EPW_hourly_dry_bulb_month(iday*24-23:iday*24));
 %   Daymax(iday) = max(EPW_hourly_dry_bulb_month(iday*24-23:iday*24));
 %   Daymin(iday) = min(EPW_hourly_dry_bulb_month(iday*24-23:iday*24));

 %   for ihour = iday*24-23:iday*24 % hours of the day 'iday'

  %      Xqis(ihour)=(EPW_hourly_dry_bulb_month(ihour)-Daymin(iday))/ ...  % Normalized on a daily
basis
  %          (Daymax(iday)-Daymin(iday));
  %      DeltaXqis(iday) = DeltaTas/(Daymax(iday)-Daymin(iday));

  %  end

%end
%DeltaXqisResh = reshape(DeltaXqis,[24,1]);

TmonMean = mean(EPW_hourly_dry_bulb_month); % Promedio de la temperatura en el mes del archivo base
de datos horarios
TmonMax = mean(Daymax); % Promedio de las temperaturas m·ximas diarias en el mes del archivo base de
datos horarios
TmonMin = mean(Daymin); % Promedio de las temperaturas mÌnimas diarias en el mes del archivo base de
datos horarios


FutureMonMeanT = TmonMean + DeltaTas; % Temperatura promedio del mes futuro
FutureMonMeanTmax = TmonMax + DeltaTasmax; % Promedio de las temperaturas m·ximas diarias del mes
futuro
FutureMonMeanTmin = TmonMin + DeltaTasmin; % Promedio de las temperaturas mÌnimas diarias del mes
futuro

%Ess = zeros(Numbdays);

Ess = ((FutureMonMeanT-FutureMonMeanTmin)/(FutureMonMeanTmax-FutureMonMeanTmin))*((Daymax-Daymin)./
(Daymean-Daymin)) - 1;


MonthMean = mean(Daymean);
MonthMeanMax = mean(Daymax);
MonthMeanMin = mean(Daymin);
%Ess = ((FutureMonMeanT-FutureMonMeanTmin)/(FutureMonMeanTmax-FutureMonMeanTmin))*((MonthMeanMax-
MonthMeanMin)/(MonthMean-MonthMeanMin)) - 1;

%Essresh = reshape(Ess,[1,Numbdays]);

Rmm = 1;
Rnn = 1;
XqisDays = reshape(Xqis,[24,Numbdays]);
XqisPrimeDays = zeros(24,Numbdays);
```

```
%Gee = (XqisDays(:,1:Numbdays).^Rmm).*((1-XqisDays(:,1:Numbdays)).^Rnn);
Gee = (Xqis.^Rmm).*((1-Xqis).^Rnn);
    %XqisPrimeDays = XqisDays(:,1:Numbdays) +
(Essresh(1,1:Numbdays).*mean(XqisDays(:,1:Numbdays)).*Gee)./mean(Gee);
    XqisPrimeDays = Xqis + (Ess.*mean(Xqis).*Gee)./mean(Gee);

for iday = 1:Numbdays

    Rmm = 1;
    Rnn = 1;


    %Gee(:,iday) = (XqisDays(:,iday).^Rmm).*((1-XqisDays(:,iday)).^Rnn);
    %XqisPrimeDays(:,iday) = XqisDays(:,iday) +
(Essresh(1,iday).*mean(XqisDays(:,iday)).*Gee(:,iday))./mean(Gee(:,iday));

    %XqisPrimeDaysm(:,iday) = XqisDays(:,iday) + (Essm.*mean(XqisDays(:,iday)).*Gee(:,iday))./
mean(Gee(:,iday));

    Iterat = 0;
    while 1
        if min(XqisPrimeDays(:,iday))>= 0 & max(XqisPrimeDays(:,iday)) <= 1
            break
        end
        IterM = 0;
        IterN = 0;

        if min(XqisPrimeDays(:,iday)) < 0

            %[minXqis,IndexminXprime] = min(XqisPrimeDays(:,iday));
                     formatSpec = 'min(XqisPrimeDays(:,iday)) < 0  %4.3f  iday %6d\n';
                     fprintf(formatSpec,min(XqisPrimeDays(:,iday)),iday);
            % put in another while loop that exits when min(XqisPrimeDays(:,iday)) is
            % between 0 and 0.001 (say)

            %Rnn = 0.5;
            IterN = 0;
            while min(XqisPrimeDays(:,iday)) < 0 || min(XqisPrimeDays(:,iday)) > 0.0001

                if min(XqisPrimeDays(:,iday)) < 0
                    Rnn = Rnn/2;
                elseif min(XqisPrimeDays(:,iday)) > 0.001
                    Rnn = (Rnn+1)/2;
                end
                Gee(:,iday) = (XqisDays(:,iday).^Rmm).*((1-XqisDays(:,iday)).^Rnn);
                XqisPrimeDays(:,iday) = XqisDays(:,iday) +
(Ess(1,iday).*mean(XqisDays(:,iday)).*Gee(:,iday))./mean(Gee(:,iday));
                IterN = IterN + 1;
                if IterN > 25;
                    break;
                end
            end


        end

        if max(XqisPrimeDays(:,iday)) > 1

               formatSpec = 'max(XqisPrimeDays(:,iday)) > 1  %4.3f  iday %6d\n';
                     fprintf(formatSpec,max(XqisPrimeDays(:,iday)),iday);

            IterM = 0;
            %Rmm = 0.5;
            while max(XqisPrimeDays(:,iday)) > 1 || max(XqisPrimeDays(:,iday)) < 0.9999

                if max(XqisPrimeDays(:,iday)) > 1
                    Rmm = Rmm/2;
                elseif max(XqisPrimeDays(:,iday)) < 0.999
                    Rmm = (Rmm+1)/2;
                end
                Gee(:,iday) = (XqisDays(:,iday).^Rmm).*((1-XqisDays(:,iday)).^Rnn);
```

```
                XqisPrimeDays(:,iday) = XqisDays(:,iday) +
(Ess(1,iday).*mean(XqisDays(:,iday)).*Gee(:,iday))./mean(Gee(:,iday));
                IterM = IterM + 1;
                if IterM > 25;
                    break;
                end
            end
        end

        if IterN > 25 || IterM > 25
            % Go to do a straight shift i.e. equation 1 of Eames et al and
            % then breaks out of while loop
            XqisPrimeDays(:,iday) = XqisDays(:,iday) + DeltaXqis; %DeltaXqis(iday);

            break; % Breaks out if the number of iterations exceeds 25
        end

    end
end


EPWhrlydbtFutureMonth = FutureMonMeanTmin + XqisPrimeDays*(FutureMonMeanTmax - FutureMonMeanTmin);

EPWhrlydbtFutureMonthm = FutureMonMeanTmin + XqisPrimeDays*(FutureMonMeanTmax - FutureMonMeanTmin);
EPWhrlydbtFutureMonthm = reshape(EPWhrlydbtFutureMonthm,[Numbhours,1]);

FuDaymin = zeros(Numbdays);
FuDaymean = zeros(Numbdays);


EPWhrlydbtFutureMonth_days = reshape(EPWhrlydbtFutureMonth,[24,Numbdays]);
FuDaymean = mean(EPWhrlydbtFutureMonth);
FuDaymin = min(EPWhrlydbtFutureMonth);

Index = FuDaymean < FuDaymin;



EPWhrlydbtFutureMonth_days(:,Index) = EPW_hourly_dry_bulb_month_days(:,Index)+DeltaTas;


EPWhrlydbtFutureMonth = reshape(EPWhrlydbtFutureMonth_days,[Numbhours,1]);


formatSpec = 'Future month mean %4.3f mean of morphed days %4.3f\n';
                     fprintf(formatSpec,FutureMonMeanT,mean(EPWhrlydbtFutureMonth));
formatSpec = 'Future month mean maximum %4.3f mean maximum of morphed days %4.3f\n';
                     fprintf(formatSpec,FutureMonMeanTmax,mean(max(EPWhrlydbtFutureMonth_days)));
formatSpec = 'Future month mean minimum %4.3f mean minimum of morphed days %4.3f\n\n';
                     fprintf(formatSpec,FutureMonMeanTmin,mean(min(EPWhrlydbtFutureMonth_days)));

figure (2);
plot (EPW_hourly_dry_bulb_month);
hold
plot (EPWhrlydbtFutureMonth);

hold
IsOK = input('øSeguir? s/n [s]: ', 's');

return;
```