STAT 501 Project

Multiple Linear Regression Model to Predict Electricity Demand of Five Cities in Spain

Links documentation

- Github Code
- Overleaf Report
- Kaggle Data

Note: If you need permission to use some links, please email at: *sjw6236@psu.edu*

Install all Packages and Import Library

Packages installation

```
# install.packages("dplyr")
# install.packages('car')
# install.packages("ggplot2")
```

Import library

```
#load package here
library(tidyverse)
library(dplyr)
library(car)
library (GGally)
library(readxl)
library(ggplot2)
library(broom)
library(ggpubr)
options(scipen=999)  # turn-off scientific notation like 1e+48
theme_set(theme_bw())  # pre-set the bw theme.
```

Data Processing

Load weather data

```
## Read raw data: weather features
weather_raw<-read.csv(file = "Data/weather_features.csv")
head(weather_raw,4)
```

```
##                       dt_iso city_name    temp temp_min temp_max pressure
## 1 2015-01-01 00:00:00+01:00  Valencia 270.475  270.475  270.475     1001
## 2 2015-01-01 01:00:00+01:00  Valencia 270.475  270.475  270.475     1001
## 3 2015-01-01 02:00:00+01:00  Valencia 269.686  269.686  269.686     1002
## 4 2015-01-01 03:00:00+01:00  Valencia 269.686  269.686  269.686     1002
```

```
##   humidity wind_speed wind_deg rain_1h rain_3h snow_3h clouds_all weather_id
## 1       77          1       62       0       0       0          0        800
## 2       77          1       62       0       0       0          0        800
## 3       78          0       23       0       0       0          0        800
## 4       78          0       23       0       0       0          0        800
##   weather_main weather_description weather_icon
## 1        clear        sky is clear          01n
## 2        clear        sky is clear          01n
## 3        clear        sky is clear          01n
## 4        clear        sky is clear          01n
```

```
# All the columns in the raw data
weather_raw_fields <- colnames(weather_raw)
print(weather_raw_fields)
```

```
##  [1] "dt_iso"             "city_name"           "temp"
##  [4] "temp_min"           "temp_max"            "pressure"
##  [7] "humidity"           "wind_speed"          "wind_deg"
## [10] "rain_1h"            "rain_3h"             "snow_3h"
## [13] "clouds_all"         "weather_id"          "weather_main"
## [16] "weather_description" "weather_icon"
```

Remove all duplicate data from the weather data

```
# All the duplicate data based on date
weather_duplicate<-weather_raw[duplicated(weather_raw[,1:2]),]
weather <- weather_raw[!duplicated(weather_raw[,1:2]), ]
```

Load energy data set modified from Kagel

Calculation: *(Keith,Can you explain here in bullets how new excel sheet is generated in bullet?)* * Use $$ for math equation, if any and reference the original data, and other links like I did in the beginning of this code

```
energy_raw<-read_excel("Data/energy_dataset-KO.xlsx")
```

```
## New names:
## * `` -> `...30`
## * `` -> `...36`
## * `` -> `...37`
## * `` -> `...38`
## * `` -> `...39`
```

- Five cities are:
  - Valencia
  - Barcelona
  - Bilbao
  - Seville
  - Madrid

Merge Energy Demand and Weather to one data set based on cities

```
# Valencia
Weather_Valencia <- weather[weather$city_name == 'Valencia',]
Data_Valencia <-merge(Weather_Valencia, energy_raw[,c("time","Valencia")], by.x = "dt_iso", by.y = "tim
colnames(Data_Valencia)[colnames(Data_Valencia) == "Valencia"] <- "energy"
```

```
# Barcelona
Weather_Barcelona <- weather[weather$city_name == " Barcelona",]
Data_Barcelona <-merge(Weather_Barcelona, energy_raw[,c("time","Barcelona")], by.x = "dt_iso", by.y = "
colnames(Data_Barcelona)[colnames(Data_Barcelona) == "Barcelona"] <- "energy"

# Bilbao
Weather_Bilbao <- weather[weather$city_name == 'Bilbao',]
Data_Bilbao <-merge(Weather_Bilbao, energy_raw[,c("time","Bilboa")], by.x = "dt_iso", by.y = "time")
colnames(Data_Bilbao)[colnames(Data_Bilbao) == "Bilboa"] <- "energy"

# Seville
Weather_Seville <- weather[weather$city_name == 'Seville',]
Data_Seville <-merge(Weather_Seville, energy_raw[,c("time","Seville")], by.x = "dt_iso", by.y = "time")
colnames(Data_Seville)[colnames(Data_Seville) == "Seville"] <- "energy"

# Madrid
Weather_Madrid <- weather[weather$city_name == 'Madrid',]
Data_Madrid <-merge(Weather_Madrid, energy_raw[,c("time","Madrid")], by.x = "dt_iso", by.y = "time")
colnames(Data_Madrid)[colnames(Data_Madrid) == "Madrid"] <- "energy"
```

Write all the data in separate cityname.csv sheets

```
write.csv(Data_Barcelona,"Data/Barcelona.csv", row.names = TRUE)
write.csv(Data_Valencia,"Data/Valencia.csv", row.names = TRUE)
write.csv(Data_Bilbao,"Data/Bilbao.csv", row.names = TRUE)
write.csv(Data_Seville,"Data/Seville.csv", row.names = TRUE)
write.csv(Data_Madrid,"Data/Madrid.csv", row.names = TRUE)
```

BARCELONA DATA

```
# Data with all predictors and response variable
## Read raw data: Barcelona
Barcelona_raw<-read.csv(file = "Data/Barcelona.csv")

# Rename column names
colnames(Barcelona_raw)[colnames(Barcelona_raw) == "X"] <- "DataID"
colnames(Barcelona_raw)[colnames(Barcelona_raw) == "dt_iso"] <- "time"

# Select only the columns of interest

Barcelona_Data<-Barcelona_raw %>%
          select(DataID,time,temp,humidity,pressure,wind_speed,rain_1h,rain_3h,snow_3h,weather_main,ene
```

See all the weather description in weather_main

```
print(unique(Barcelona_Data$weather_main))
```

```
## [1] "clear"        "clouds"       "rain"         "snow"         "drizzle"
## [6] "thunderstorm" "mist"         "fog"          "dust"
```

Give integer values for each description

```
Barcelona_Data<-transform(Barcelona_Data, weather_main = factor(weather_main,
      levels = c("clear", "clouds", "drizzle","dust","fog","haze","mist","rain","smoke","snow","squall"
       labels = c(1:12)))

Weather_description <- data.frame (Weather_Description  = c("clear", "clouds", "drizzle",
              "dust","fog","haze","mist","rain","smoke","snow","squall","thunderstorm"),
                  Index = c(1:12)
                  )
print(Weather_description)
```

```
##    Weather_Description Index
## 1                clear     1
## 2               clouds     2
## 3              drizzle     3
## 4                 dust     4
## 5                  fog     5
## 6                 haze     6
## 7                 mist     7
## 8                 rain     8
## 9                smoke     9
## 10                snow    10
## 11              squall    11
## 12        thunderstorm    12
```

More Data Processing

```
# Add two columns of rain duration
Barcelona_Data["rain_duration"] <- Barcelona_Data$rain_1h + Barcelona_Data$rain_3h

# Rename Column Names for clarity
Barcelona_Data <-merge(Barcelona_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nTime Ba
colnames(Barcelona_Data )[colnames(Barcelona_Data ) == "Specified Categorical Variable:\r\nTime Band (P

Barcelona_Data <-merge(Barcelona_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nSeason
colnames(Barcelona_Data )[colnames(Barcelona_Data ) == "Specified Categorical Variable:\r\nSeason (Spri

Barcelona_Data <-merge(Barcelona_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nTime of
colnames(Barcelona_Data )[colnames(Barcelona_Data ) == "Specified Categorical Variable:\r\nTime of day


colnames(Barcelona_Data )[colnames(Barcelona_Data ) == "snow_3h"] <- "snow_duration"

Barcelona_Data<-subset(Barcelona_Data,select=-c(rain_1h,rain_3h,DataID))

colnames(Barcelona_Data )[colnames(Barcelona_Data ) == "time"] <- "time_ID"

Barcelona_Data <- Barcelona_Data [, c("time_ID","temp","humidity","pressure","wind_speed","rain_duratio

colnames(Barcelona_Data )[colnames(Barcelona_Data ) == "energy"] <- "energy_demand"
```

Removal of rows with errorenous data

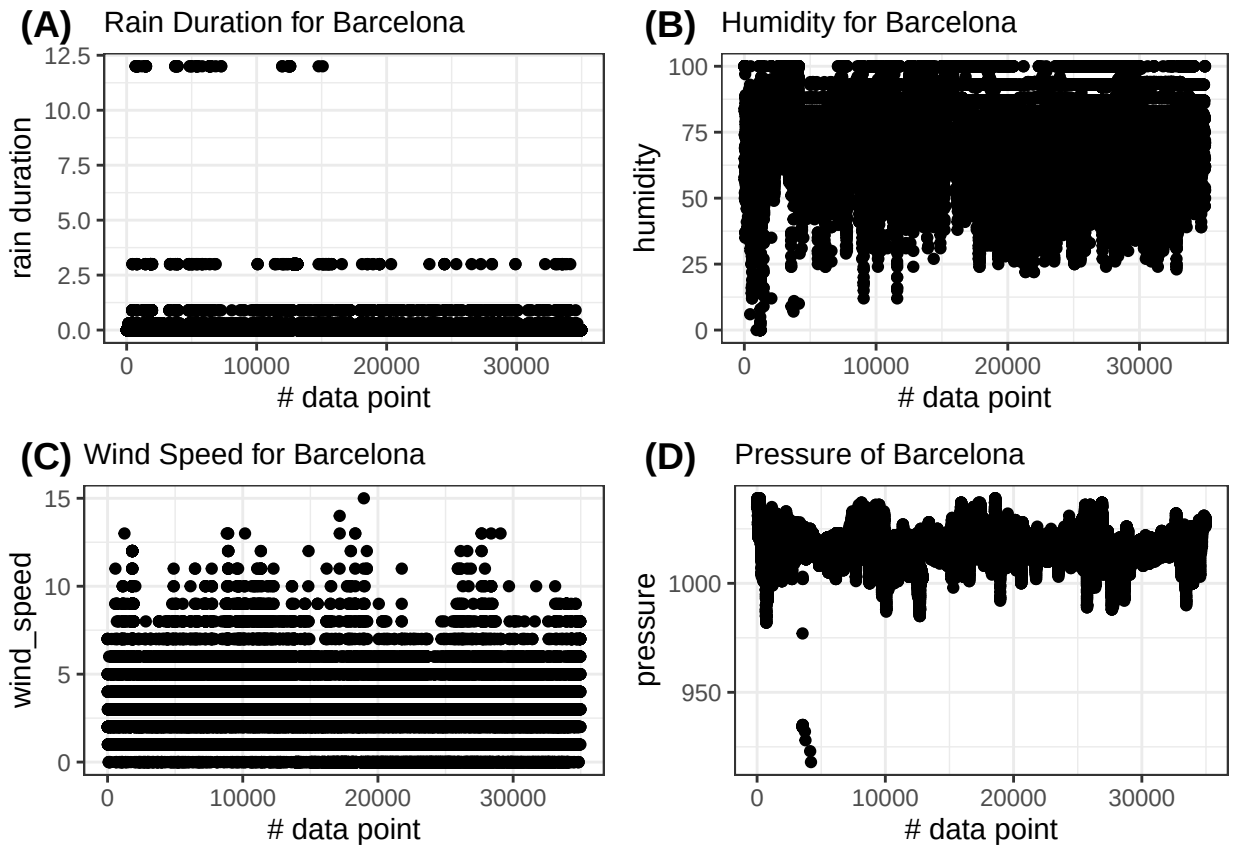
Reference to average weather data for Barcelona

```
Barcelona_Data[Barcelona_Data$temp < 270 ,]
```

```
##                       time_ID   temp humidity pressure wind_speed
## 1283 2015-02-23 10:00:00+01:00 262.24        0     1007          3
##      rain_duration snow_duration day_night time_band season weather_main
## 1283             0             0         1         1      4            2
##      energy_demand
## 1283      4331.571
```

```
Barcelona_Data<- subset(Barcelona_Data, temp>270) # remove temperature less than 270K
Barcelona_Data<- subset(Barcelona_Data, pressure >900 & pressure <1050) # remove pressure outside [900,
Barcelona_Data<-subset(Barcelona_Data, energy_demand >10) # Remove all 0 demand from the data
```

```
Barcelona.rain_duration <- ggplot(Barcelona_Data) +
          geom_point( aes(c(1:length(rain_duration)),rain_duration)) +
          labs(subtitle="Rain Duration for Barcelona",
          y="rain duration",
          x="# data point")

Barcelona.humidity <- ggplot(Barcelona_Data) +
          geom_point( aes(c(1:length(humidity)),humidity)) +
          labs(subtitle="Humidity for Barcelona",
          y="humidity",
          x="# data point")

Barcelona.wind_speed <- ggplot(Barcelona_Data) +
          geom_point( aes(c(1:length(wind_speed)),wind_speed)) +
          labs(subtitle="Wind Speed for Barcelona",
          y="wind_speed",
          x="# data point")

Barcelona.pressure <- ggplot(Barcelona_Data) +
          geom_point( aes(c(1:length(pressure)),pressure)) +
          labs(subtitle="Pressure of Barcelona",
          y="pressure",
          x="# data point")

ggarrange(Barcelona.rain_duration,Barcelona.humidity,Barcelona.wind_speed,Barcelona.pressure,
          labels = c("(A)", "(B)", "(C)","(D)"),
          ncol = 2, nrow = 2)
```




```
FullModel <- lm(energy_demand ~ temp + humidity + pressure + wind_speed + rain_duration + factor(day_nig
summary(FullModel)$adj.r.squared
```

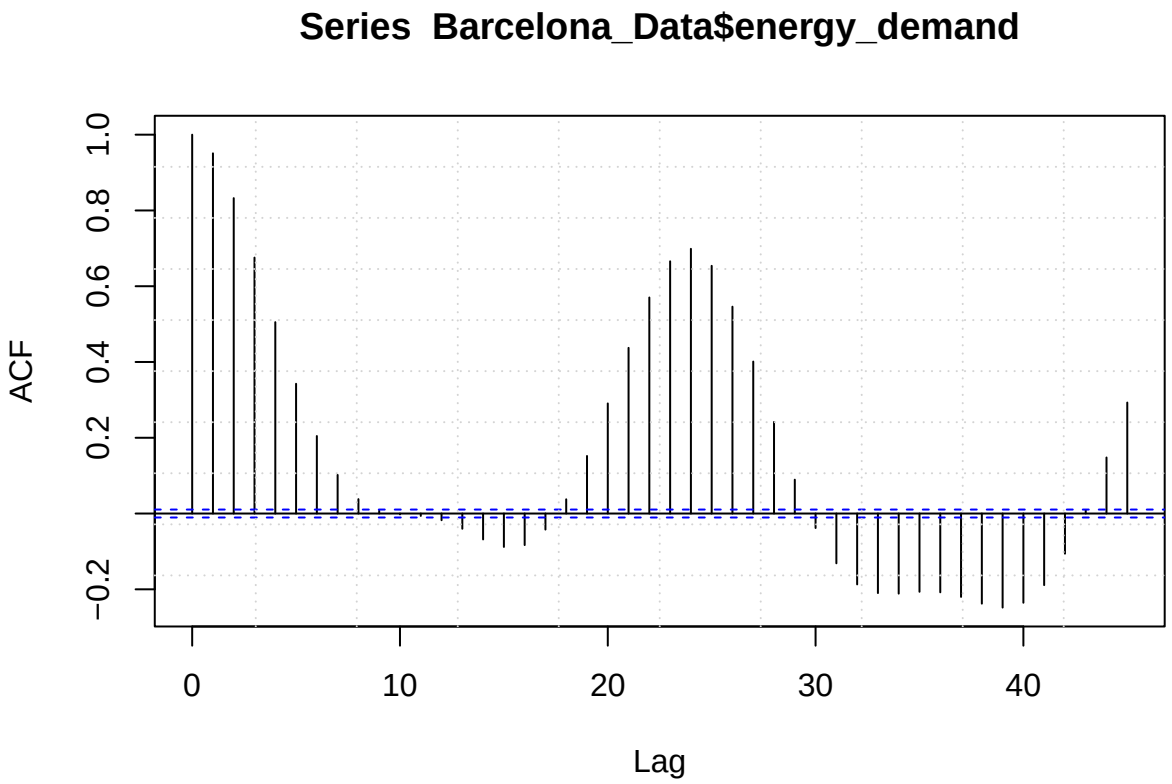
```
## [1] 0.3125999
```

Adjusted R2 of the model is very low *0.3127159*.

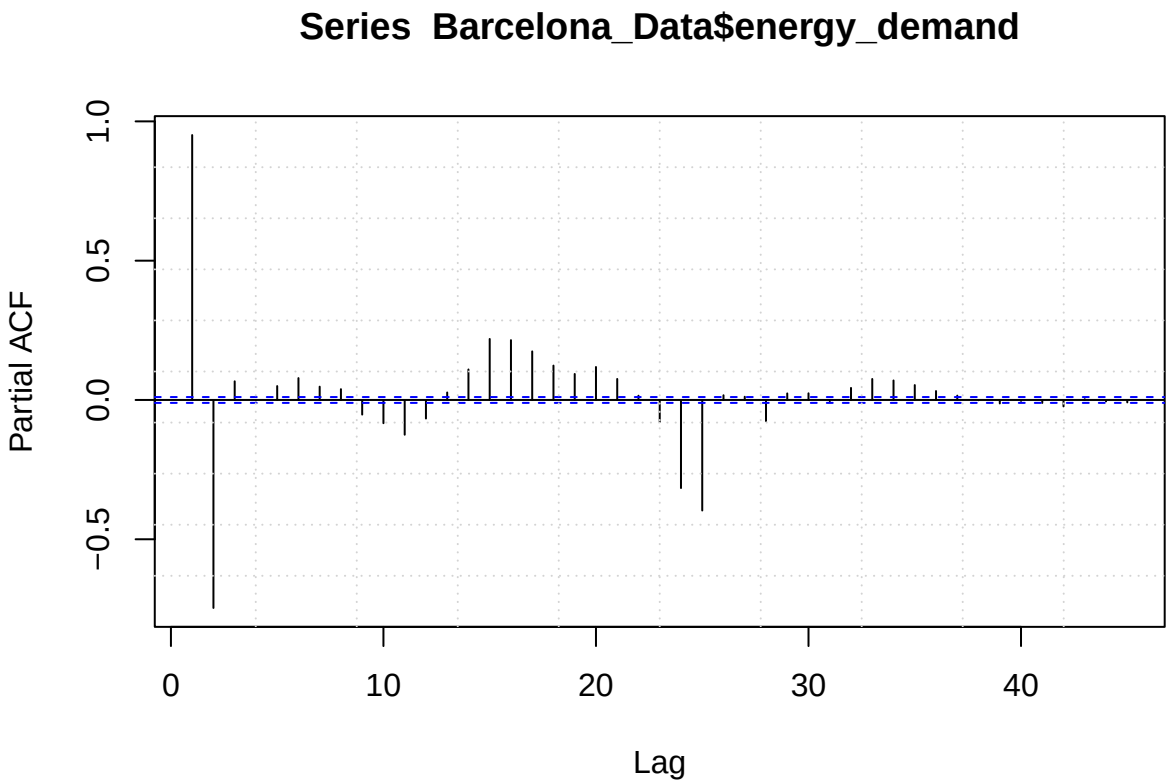
We perform Autocorrelation and Partial Autocorrelation to incorporate temporal variability/influence in the model.

Generation of AutoCorrelation and Partial Autocorrelation profile

```
Barcelona.acr <- acf(Barcelona_Data$energy_demand,plot = TRUE)
grid(10,10)
```




```
Barcelona.pacr<- pacf(Barcelona_Data$energy_demand,plot =TRUE)
grid(10,10)
```




```
pacf_values <-as.data.frame(Barcelona.pacr$acf)
```

```r
pacf_values_detect<-subset(pacf_values, abs(pacf_values) >0.25)

head(pacf_values_detect)
```

```
##            V1
## 1   0.9508500
## 2  -0.7465278
## 24 -0.3161943
## 25 -0.3967825
```

```r
# Check the highest partially auto correlated energy demand in order
idx<-order(abs(pacf_values_detect$V1),decreasing = TRUE)
pacf_values_detect<- pacf_values_detect[order(abs(pacf_values_detect$V1),decreasing = TRUE),]
```

Create energy demand lag variable, one at a time

```r
E_1<-Barcelona_Data$energy_demand[-1] # Get all the data except in first row
Barcelona_Data<-Barcelona_Data[-nrow(Barcelona_Data),]
Barcelona_Data$E_1 <- E_1

E_2<-E_1[-1]
Barcelona_Data<-Barcelona_Data[-nrow(Barcelona_Data),]
Barcelona_Data$E_2 <- E_2

E_25<-Barcelona_Data$energy_demand[25:nrow(Barcelona_Data)] # Get all the data except in first row
Barcelona_Data<-Barcelona_Data[1:(nrow(Barcelona_Data)-24),]

Barcelona_Data$E_25 <- E_25
```

Check the model now

```r
FullModel1 <- lm(energy_demand ~ E_1 + E_2 + E_25 + temp + humidity + pressure + wind_speed + rain_dura
summary(FullModel1)
```

```
##
## Call:
## lm(formula = energy_demand ~ E_1 + E_2 + E_25 + temp + humidity +
##     pressure + wind_speed + rain_duration + factor(day_night) +
##     factor(time_band) + factor(season) + factor(weather_main),
##     data = Barcelona_Data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1609.24   -50.99     8.82    62.86  2529.96
##
## Coefficients:
##                           Estimate  Std. Error  t value             Pr(>|t|)
## (Intercept)           -443.913651   95.856207   -4.631      0.0000036514802
## E_1                      1.602696    0.004053  395.472 < 0.0000000000000002
## E_2                     -0.714152    0.003737 -191.080 < 0.0000000000000002
## E_25                     0.030901    0.001588   19.454 < 0.0000000000000002
## temp                     2.564276    0.162888   15.743 < 0.0000000000000002
```

```
## humidity                 -0.476128    0.041091  -11.587 < 0.0000000000000002
## pressure                  0.008830    0.085399    0.103             0.917648
## wind_speed                0.503034    0.320373    1.570             0.116389
## rain_duration             0.088170    1.063993    0.083             0.933957
## factor(day_night)2        6.985750    1.654274    4.223      0.0000241842532
## factor(time_band)2       26.676131    7.987014    3.340             0.000839
## factor(time_band)3       14.395365    6.215916    2.316             0.020570
## factor(season)2         -15.109435    2.308903   -6.544      0.0000000000607
## factor(season)3          -2.076330    1.806323   -1.149             0.250367
## factor(season)4          22.721483    1.974512   11.507 < 0.0000000000000002
## factor(weather_main)2    -0.735382    1.323482   -0.556             0.578459
## factor(weather_main)3    -2.926329    8.477791   -0.345             0.729964
## factor(weather_main)4   181.549670   79.215696    2.292             0.021921
## factor(weather_main)5     0.717756   14.634756    0.049             0.960884
## factor(weather_main)7   -15.212228    6.012154   -2.530             0.011403
## factor(weather_main)8     6.053918    2.536987    2.386             0.017026
## factor(weather_main)10  105.739970   30.017678    3.523             0.000428
## factor(weather_main)12    2.309079    7.752191    0.298             0.765811
##
## (Intercept)            ***
## E_1                    ***
## E_2                    ***
## E_25                   ***
## temp                   ***
## humidity               ***
## pressure
## wind_speed
## rain_duration
## factor(day_night)2     ***
## factor(time_band)2     ***
## factor(time_band)3     *
## factor(season)2        ***
## factor(season)3
## factor(season)4        ***
## factor(weather_main)2
## factor(weather_main)3
## factor(weather_main)4  *
## factor(weather_main)5
## factor(weather_main)7  *
## factor(weather_main)8  *
## factor(weather_main)10 ***
## factor(weather_main)12
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 112 on 34930 degrees of freedom
## Multiple R-squared:  0.9591, Adjusted R-squared:  0.9591
## F-statistic: 3.724e+04 on 22 and 34930 DF,  p-value: < 0.00000000000000022
```

Generate Working and Evaluation Data

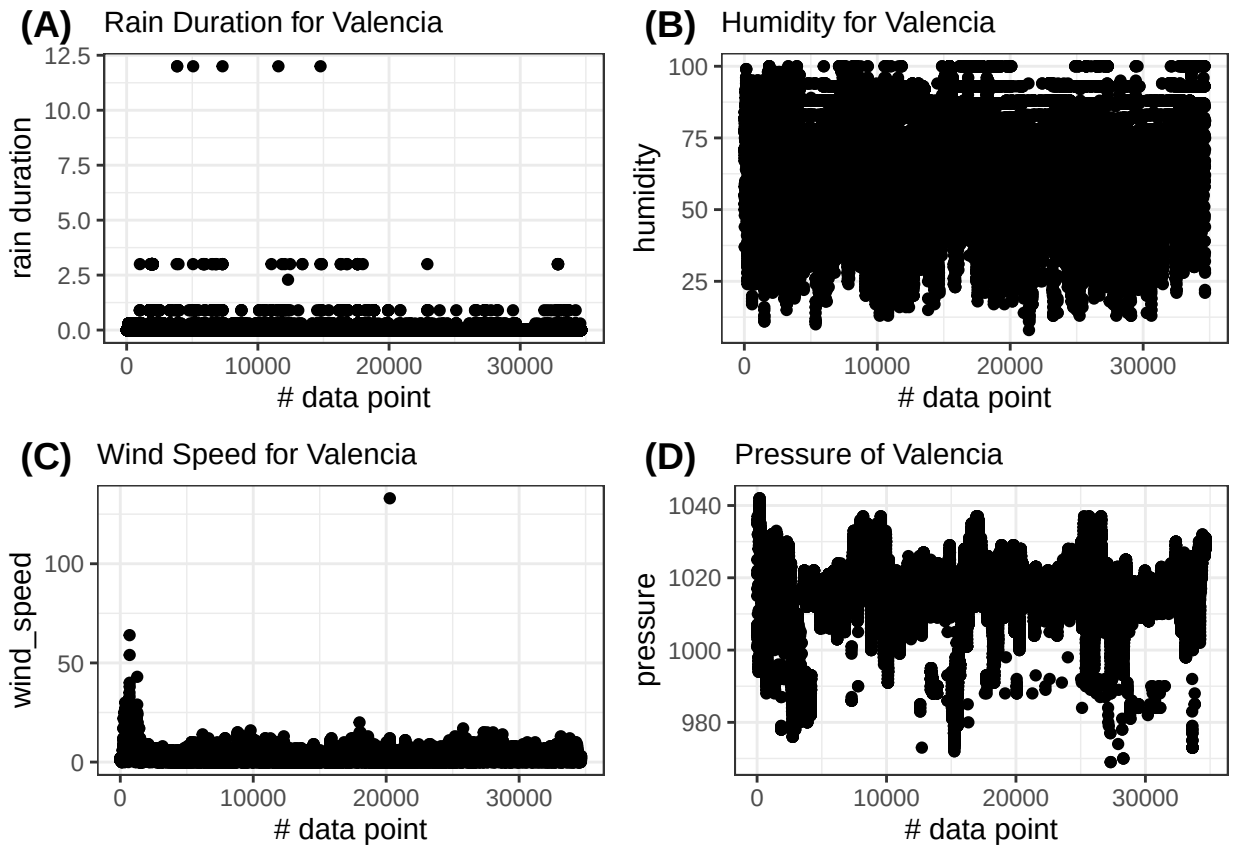
```
Barcelona_Data <- Barcelona_Data [, c("time_ID", "E_1","E_2","E_25",   "temp","humidity","pressure","wi
```

```
set.seed(501)
working_rows <- sample(1:nrow(Barcelona_Data),0.80*nrow(Barcelona_Data))

working_Barcelona <-Barcelona_Data[working_rows,]

working_Barcelona$index <- as.numeric(row.names(working_Barcelona))
working_Barcelona<- working_Barcelona[order(working_Barcelona$index), ]

working_Barcelona<-working_Barcelona %>%
                    relocate(index)


evaluation_Barcelona <-Barcelona_Data[-working_rows,]
evaluation_Barcelona$index <- as.numeric(row.names(evaluation_Barcelona))
evaluation_Barcelona<- evaluation_Barcelona[order(evaluation_Barcelona$index), ]

evaluation_Barcelona<-evaluation_Barcelona %>%
                    relocate(index)

write.csv(working_Barcelona,"Data/working_Barcelona.csv", row.names = FALSE)
write.csv(evaluation_Barcelona,"Data/evaluation_Barcelona.csv", row.names = FALSE)
```

Valencia DATA Preparation

```
Valencia_raw<-read.csv(file = "Data/Valencia.csv")
colnames(Valencia_raw)[colnames(Valencia_raw) == "X"] <- "DataID"
colnames(Valencia_raw)[colnames(Valencia_raw) == "dt_iso"] <- "time"
Valencia_Data<-Valencia_raw %>%
  select(DataID,time,temp,humidity,pressure,wind_speed,rain_1h,rain_3h,snow_3h,weather_main,energy)

Valencia_Data<-transform(Valencia_Data, weather_main = factor(weather_main,
                                                              levels = c("clear", "clouds", "drizzle","
                                                              labels = c(1:12)))

# Add two columns of rain duration
Valencia_Data["rain_duration"] <- Valencia_Data$rain_1h + Valencia_Data$rain_3h

# Rename Column Names for clarity
Valencia_Data <-merge(Valencia_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nTime Band
colnames(Valencia_Data )[colnames(Valencia_Data ) == "Specified Categorical Variable:\r\nTime Band (Peal

Valencia_Data <-merge(Valencia_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nSeason (Sp
colnames(Valencia_Data )[colnames(Valencia_Data ) == "Specified Categorical Variable:\r\nSeason (Spring

Valencia_Data <-merge(Valencia_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nTime of da
colnames(Valencia_Data )[colnames(Valencia_Data ) == "Specified Categorical Variable:\r\nTime of day (Da


colnames(Valencia_Data )[colnames(Valencia_Data ) == "snow_3h"] <- "snow_duration"

Valencia_Data<-subset(Valencia_Data,select=-c(rain_1h,rain_3h,DataID))

colnames(Valencia_Data )[colnames(Valencia_Data ) == "time"] <- "time_ID"
```

```
Valencia_Data <- Valencia_Data [, c("time_ID","temp","humidity","pressure","wind_speed","rain_duration"

colnames(Valencia_Data )[colnames(Valencia_Data ) == "energy"] <- "energy_demand"

VD_<- Valencia_Data[Valencia_Data$temp < 270 ,]
Valencia_Data<- subset(Valencia_Data, temp>270) # remove temperature less than 270K
Valencia_Data<- subset(Valencia_Data, pressure >900 & pressure <1050) # remove pressure outside [900,10
Valencia_Data<-subset(Valencia_Data, energy_demand >10) # Remove all 0 demand from the data

Valencia.rain_duration <- ggplot(Valencia_Data) +
  geom_point( aes(c(1:length(rain_duration)),rain_duration)) +
  labs(subtitle="Rain Duration for Valencia",
       y="rain duration",
       x="# data point")

Valencia.humidity <- ggplot(Valencia_Data) +
  geom_point( aes(c(1:length(humidity)),humidity)) +
  labs(subtitle="Humidity for Valencia",
       y="humidity",
       x="# data point")

Valencia.wind_speed <- ggplot(Valencia_Data) +
  geom_point( aes(c(1:length(wind_speed)),wind_speed)) +
  labs(subtitle="Wind Speed for Valencia",
       y="wind_speed",
       x="# data point")

Valencia.pressure <- ggplot(Valencia_Data) +
  geom_point( aes(c(1:length(pressure)),pressure)) +
  labs(subtitle="Pressure of Valencia",
       y="pressure",
       x="# data point")

ggarrange(Valencia.rain_duration,Valencia.humidity,Valencia.wind_speed,Valencia.pressure,
          labels = c("(A)", "(B)", "(C)","(D)"),
          ncol = 2, nrow = 2)
```



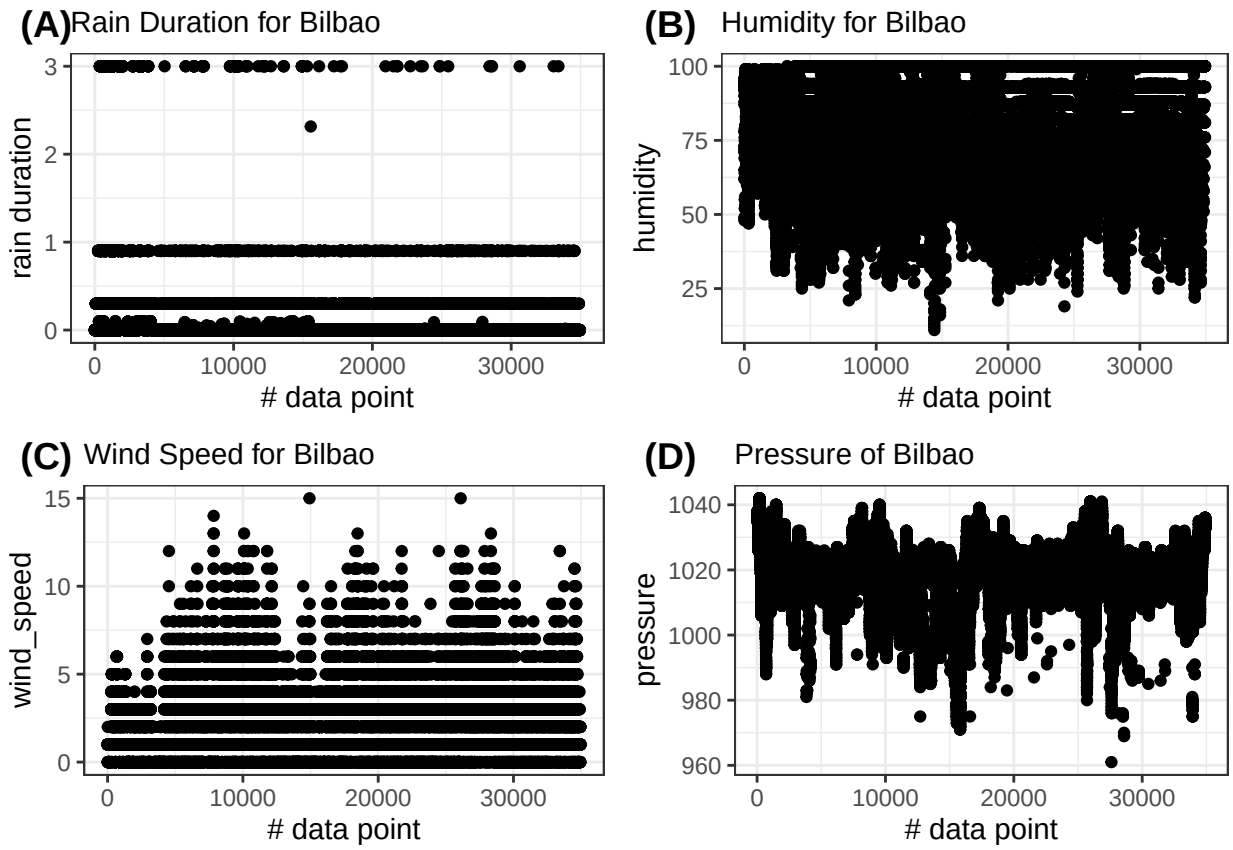

```
E_1<-Valencia_Data$energy_demand[-1] # Get all the data except in first row
Valencia_Data<-Valencia_Data[-nrow(Valencia_Data),]
Valencia_Data$E_1 <- E_1

E_2<-E_1[-1]
Valencia_Data<-Valencia_Data[-nrow(Valencia_Data),]
Valencia_Data$E_2 <- E_2

E_25<-Valencia_Data$energy_demand[25:nrow(Valencia_Data)] # Get all the data except in first row
Valencia_Data<-Valencia_Data[1:(nrow(Valencia_Data)-24),]

Valencia_Data$E_25 <- E_25

Valencia_Data <- Valencia_Data [, c("time_ID", "E_1","E_2","E_25",   "temp","humidity","pressure","wind


### Generate Working and Evaluation Data
set.seed(501)
working_rows <- sample(1:nrow(Valencia_Data),0.80*nrow(Valencia_Data))

working_Valencia <-Valencia_Data[working_rows,]

working_Valencia$index <- as.numeric(row.names(working_Valencia))
working_Valencia<- working_Valencia[order(working_Valencia$index), ]

working_Valencia<-working_Valencia %>%
  relocate(index)
```

```
evaluation_Valencia <-Valencia_Data[-working_rows,]
evaluation_Valencia$index <- as.numeric(row.names(evaluation_Valencia))
evaluation_Valencia<- evaluation_Valencia[order(evaluation_Valencia$index), ]

evaluation_Valencia<-evaluation_Valencia %>%
  relocate(index)

write.csv(working_Valencia,"Data/working_Valencia.csv", row.names = FALSE)
write.csv(evaluation_Valencia,"Data/evaluation_Valencia.csv", row.names = FALSE)
```

Bilbao DATA Preparation

```
Bilbao_raw<-read.csv(file = "Data/Bilbao.csv")
colnames(Bilbao_raw)[colnames(Bilbao_raw) == "X"] <- "DataID"
colnames(Bilbao_raw)[colnames(Bilbao_raw) == "dt_iso"] <- "time"
Bilbao_Data<-Bilbao_raw %>%
  select(DataID,time,temp,humidity,pressure,wind_speed,rain_1h,rain_3h,snow_3h,weather_main,energy)

Bilbao_Data<-transform(Bilbao_Data, weather_main = factor(weather_main,
                                                          levels = c("clear", "clouds", "drizzle","
                                                          labels = c(1:12)))

# Add two columns of rain duration
Bilbao_Data["rain_duration"] <- Bilbao_Data$rain_1h + Bilbao_Data$rain_3h

# Rename Column Names for clarity
Bilbao_Data <-merge(Bilbao_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nTime Band (Pea
colnames(Bilbao_Data )[colnames(Bilbao_Data ) == "Specified Categorical Variable:\r\nTime Band (Peak, O

Bilbao_Data <-merge(Bilbao_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nSeason (Spring
colnames(Bilbao_Data )[colnames(Bilbao_Data ) == "Specified Categorical Variable:\r\nSeason (Spring, Su

Bilbao_Data <-merge(Bilbao_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nTime of day (
colnames(Bilbao_Data )[colnames(Bilbao_Data ) == "Specified Categorical Variable:\r\nTime of day (Day v


colnames(Bilbao_Data )[colnames(Bilbao_Data ) == "snow_3h"] <- "snow_duration"

Bilbao_Data<-subset(Bilbao_Data,select=-c(rain_1h,rain_3h,DataID))

colnames(Bilbao_Data )[colnames(Bilbao_Data ) == "time"] <- "time_ID"

Bilbao_Data <- Bilbao_Data [, c("time_ID","temp","humidity","pressure","wind_speed","rain_duration","sno

colnames(Bilbao_Data )[colnames(Bilbao_Data ) == "energy"] <- "energy_demand"

BD_<- Bilbao_Data[Bilbao_Data$temp < 270 ,]
Bilbao_Data<- subset(Bilbao_Data, temp>270) # remove temperature less than 270K
Bilbao_Data<- subset(Bilbao_Data, pressure >900 & pressure <1050) # remove pressure outside [900,1050]m
Bilbao_Data<-subset(Bilbao_Data, energy_demand >10) # Remove all 0 demand from the data
```

```
Bilbao.rain_duration <- ggplot(Bilbao_Data) +
  geom_point( aes(c(1:length(rain_duration)),rain_duration)) +
  labs(subtitle="Rain Duration for Bilbao",
       y="rain duration",
       x="# data point")

Bilbao.humidity <- ggplot(Bilbao_Data) +
  geom_point( aes(c(1:length(humidity)),humidity)) +
  labs(subtitle="Humidity for Bilbao",
       y="humidity",
       x="# data point")

Bilbao.wind_speed <- ggplot(Bilbao_Data) +
  geom_point( aes(c(1:length(wind_speed)),wind_speed)) +
  labs(subtitle="Wind Speed for Bilbao",
       y="wind_speed",
       x="# data point")

Bilbao.pressure <- ggplot(Bilbao_Data) +
  geom_point( aes(c(1:length(pressure)),pressure)) +
  labs(subtitle="Pressure of Bilbao",
       y="pressure",
       x="# data point")

ggarrange(Bilbao.rain_duration,Bilbao.humidity,Bilbao.wind_speed,Bilbao.pressure,
          labels = c("(A)", "(B)", "(C)","(D)"),
          ncol = 2, nrow = 2)
```



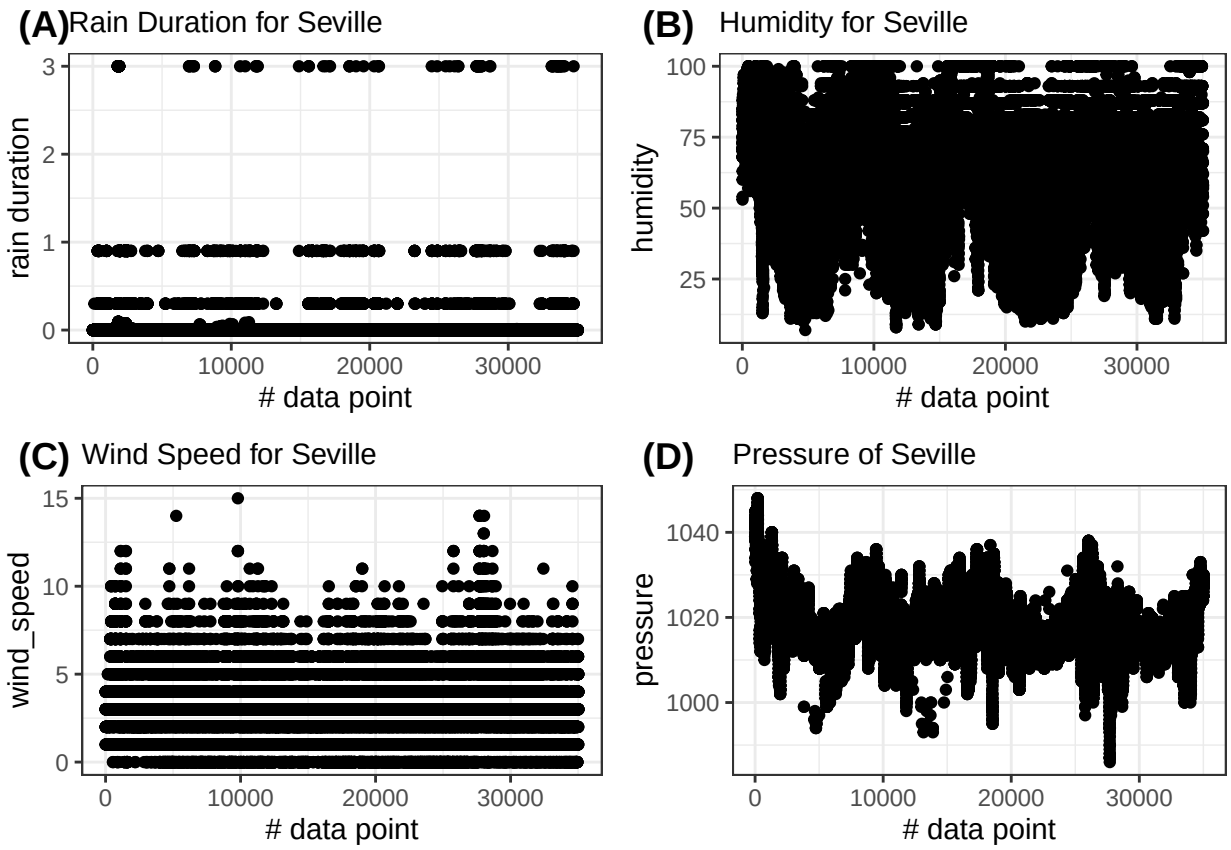

```
E_1<-Bilbao_Data$energy_demand[-1] # Get all the data except in first row
Bilbao_Data<-Bilbao_Data[-nrow(Bilbao_Data),]
Bilbao_Data$E_1 <- E_1

E_2<-E_1[-1]
Bilbao_Data<-Bilbao_Data[-nrow(Bilbao_Data),]
Bilbao_Data$E_2 <- E_2

E_25<-Bilbao_Data$energy_demand[25:nrow(Bilbao_Data)] # Get all the data except in first row
Bilbao_Data<-Bilbao_Data[1:(nrow(Bilbao_Data)-24),]

Bilbao_Data$E_25 <- E_25

Bilbao_Data <- Bilbao_Data [, c("time_ID", "E_1","E_2","E_25",   "temp","humidity","pressure","wind_spe


### Generate Working and Evaluation Data
set.seed(501)
working_rows <- sample(1:nrow(Bilbao_Data),0.80*nrow(Bilbao_Data))

working_Bilbao <-Bilbao_Data[working_rows,]

working_Bilbao$index <- as.numeric(row.names(working_Bilbao))
working_Bilbao<- working_Bilbao[order(working_Bilbao$index), ]

working_Bilbao<-working_Bilbao %>%
  relocate(index)
```

```
evaluation_Bilbao <-Bilbao_Data[-working_rows,]
evaluation_Bilbao$index <- as.numeric(row.names(evaluation_Bilbao))
evaluation_Bilbao<- evaluation_Bilbao[order(evaluation_Bilbao$index), ]

evaluation_Bilbao<-evaluation_Bilbao %>%
  relocate(index)

write.csv(working_Bilbao,"Data/working_Bilbao.csv", row.names = FALSE)
write.csv(evaluation_Bilbao,"Data/evaluation_Bilbao.csv", row.names = FALSE)
```

Seville DATA Preparation

```
Seville_raw<-read.csv(file = "Data/Seville.csv")
colnames(Seville_raw)[colnames(Seville_raw) == "X"] <- "DataID"
colnames(Seville_raw)[colnames(Seville_raw) == "dt_iso"] <- "time"
Seville_Data<-Seville_raw %>%
  select(DataID,time,temp,humidity,pressure,wind_speed,rain_1h,rain_3h,snow_3h,weather_main,energy)

Seville_Data<-transform(Seville_Data, weather_main = factor(weather_main,
                                                            levels = c("clear", "clouds", "drizzle","
                                                            labels = c(1:12)))

# Add two columns of rain duration
Seville_Data["rain_duration"] <- Seville_Data$rain_1h + Seville_Data$rain_3h

# Rename Column Names for clarity
Seville_Data <-merge(Seville_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nTime Band (
colnames(Seville_Data )[colnames(Seville_Data ) == "Specified Categorical Variable:\r\nTime Band (Peak,

Seville_Data <-merge(Seville_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nSeason (Spri
colnames(Seville_Data )[colnames(Seville_Data ) == "Specified Categorical Variable:\r\nSeason (Spring, S

Seville_Data <-merge(Seville_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nTime of day
colnames(Seville_Data )[colnames(Seville_Data ) == "Specified Categorical Variable:\r\nTime of day (Day


colnames(Seville_Data )[colnames(Seville_Data ) == "snow_3h"] <- "snow_duration"

Seville_Data<-subset(Seville_Data,select=-c(rain_1h,rain_3h,DataID))

colnames(Seville_Data )[colnames(Seville_Data ) == "time"] <- "time_ID"

Seville_Data <- Seville_Data [, c("time_ID","temp","humidity","pressure","wind_speed","rain_duration","s

colnames(Seville_Data )[colnames(Seville_Data ) == "energy"] <- "energy_demand"

SD_<-Seville_Data[Seville_Data$temp < 270 ,]
Seville_Data<- subset(Seville_Data, temp>270) # remove temperature less than 270K
Seville_Data<- subset(Seville_Data, pressure >900 & pressure <1050) # remove pressure outside [900,1050]
Seville_Data<-subset(Seville_Data, energy_demand >10) # Remove all 0 demand from the data
```

```
Seville.rain_duration <- ggplot(Seville_Data) +
  geom_point( aes(c(1:length(rain_duration)),rain_duration)) +
  labs(subtitle="Rain Duration for Seville",
       y="rain duration",
       x="# data point")

Seville.humidity <- ggplot(Seville_Data) +
  geom_point( aes(c(1:length(humidity)),humidity)) +
  labs(subtitle="Humidity for Seville",
       y="humidity",
       x="# data point")

Seville.wind_speed <- ggplot(Seville_Data) +
  geom_point( aes(c(1:length(wind_speed)),wind_speed)) +
  labs(subtitle="Wind Speed for Seville",
       y="wind_speed",
       x="# data point")

Seville.pressure <- ggplot(Seville_Data) +
  geom_point( aes(c(1:length(pressure)),pressure)) +
  labs(subtitle="Pressure of Seville",
       y="pressure",
       x="# data point")

ggarrange(Seville.rain_duration,Seville.humidity,Seville.wind_speed,Seville.pressure,
          labels = c("(A)", "(B)", "(C)","(D)"),
          ncol = 2, nrow = 2)
```



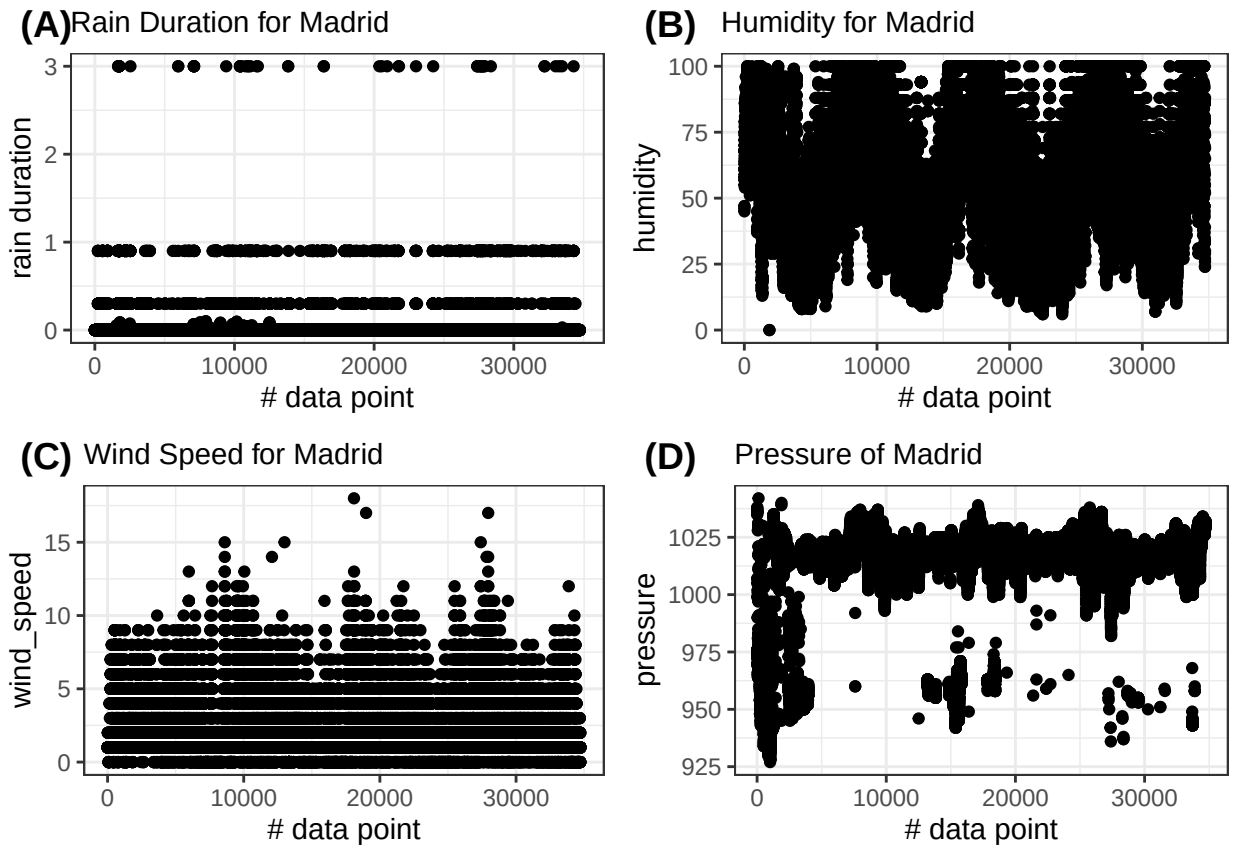

```
E_1<-Seville_Data$energy_demand[-1] # Get all the data except in first row
Seville_Data<-Seville_Data[-nrow(Seville_Data),]
Seville_Data$E_1 <- E_1

E_2<-E_1[-1]
Seville_Data<-Seville_Data[-nrow(Seville_Data),]
Seville_Data$E_2 <- E_2

E_25<-Seville_Data$energy_demand[25:nrow(Seville_Data)] # Get all the data except in first row
Seville_Data<-Seville_Data[1:(nrow(Seville_Data)-24),]

Seville_Data$E_25 <- E_25

Seville_Data <- Seville_Data [, c("time_ID", "E_1","E_2","E_25",   "temp","humidity","pressure","wind_sp


### Generate Working and Evaluation Data
set.seed(501)
working_rows <- sample(1:nrow(Seville_Data),0.80*nrow(Seville_Data))

working_Seville <-Seville_Data[working_rows,]

working_Seville$index <- as.numeric(row.names(working_Seville))
working_Seville<- working_Seville[order(working_Seville$index), ]

working_Seville<-working_Seville %>%
  relocate(index)
```

```
evaluation_Seville <-Seville_Data[-working_rows,]
evaluation_Seville$index <- as.numeric(row.names(evaluation_Seville))
evaluation_Seville<- evaluation_Seville[order(evaluation_Seville$index), ]

evaluation_Seville<-evaluation_Seville %>%
  relocate(index)

write.csv(working_Seville,"Data/working_Seville.csv", row.names = FALSE)
write.csv(evaluation_Seville,"Data/evaluation_Seville.csv", row.names = FALSE)
```

Madrid DATA Preparation

```
Madrid_raw<-read.csv(file = "Data/Madrid.csv")
colnames(Madrid_raw)[colnames(Madrid_raw) == "X"] <- "DataID"
colnames(Madrid_raw)[colnames(Madrid_raw) == "dt_iso"] <- "time"
Madrid_Data<-Madrid_raw %>%
  select(DataID,time,temp,humidity,pressure,wind_speed,rain_1h,rain_3h,snow_3h,weather_main,energy)

Madrid_Data<-transform(Madrid_Data, weather_main = factor(weather_main,
                                                          levels = c("clear", "clouds", "drizzle","
                                                          labels = c(1:12)))

# Add two columns of rain duration
Madrid_Data["rain_duration"] <- Madrid_Data$rain_1h + Madrid_Data$rain_3h

# Rename Column Names for clarity
Madrid_Data <-merge(Madrid_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nTime Band (Pea
colnames(Madrid_Data )[colnames(Madrid_Data ) == "Specified Categorical Variable:\r\nTime Band (Peak, O

Madrid_Data <-merge(Madrid_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nSeason (Spring
colnames(Madrid_Data )[colnames(Madrid_Data ) == "Specified Categorical Variable:\r\nSeason (Spring, Su

Madrid_Data <-merge(Madrid_Data, energy_raw[,c("time","Specified Categorical Variable:\r\nTime of day (
colnames(Madrid_Data )[colnames(Madrid_Data ) == "Specified Categorical Variable:\r\nTime of day (Day v


colnames(Madrid_Data )[colnames(Madrid_Data ) == "snow_3h"] <- "snow_duration"

Madrid_Data<-subset(Madrid_Data,select=-c(rain_1h,rain_3h,DataID))

colnames(Madrid_Data )[colnames(Madrid_Data ) == "time"] <- "time_ID"

Madrid_Data <- Madrid_Data [, c("time_ID","temp","humidity","pressure","wind_speed","rain_duration","sno

colnames(Madrid_Data )[colnames(Madrid_Data ) == "energy"] <- "energy_demand"

MD_ <- Madrid_Data[Madrid_Data$temp < 270 ,]
Madrid_Data<- subset(Madrid_Data, temp>270) # remove temperature less than 270K
Madrid_Data<- subset(Madrid_Data, pressure >900 & pressure <1050) # remove pressure outside [900,1050]m
Madrid_Data<-subset(Madrid_Data, energy_demand >10) # Remove all 0 demand from the data
```

```
Madrid.rain_duration <- ggplot(Madrid_Data) +
  geom_point( aes(c(1:length(rain_duration)),rain_duration)) +
  labs(subtitle="Rain Duration for Madrid",
       y="rain duration",
       x="# data point")

Madrid.humidity <- ggplot(Madrid_Data) +
  geom_point( aes(c(1:length(humidity)),humidity)) +
  labs(subtitle="Humidity for Madrid",
       y="humidity",
       x="# data point")

Madrid.wind_speed <- ggplot(Madrid_Data) +
  geom_point( aes(c(1:length(wind_speed)),wind_speed)) +
  labs(subtitle="Wind Speed for Madrid",
       y="wind_speed",
       x="# data point")

Madrid.pressure <- ggplot(Madrid_Data) +
  geom_point( aes(c(1:length(pressure)),pressure)) +
  labs(subtitle="Pressure of Madrid",
       y="pressure",
       x="# data point")

ggarrange(Madrid.rain_duration,Madrid.humidity,Madrid.wind_speed,Madrid.pressure,
          labels = c("(A)", "(B)", "(C)","(D)"),
          ncol = 2, nrow = 2)
```




```
E_1<-Madrid_Data$energy_demand[-1] # Get all the data except in first row
Madrid_Data<-Madrid_Data[-nrow(Madrid_Data),]
Madrid_Data$E_1 <- E_1

E_2<-E_1[-1]
Madrid_Data<-Madrid_Data[-nrow(Madrid_Data),]
Madrid_Data$E_2 <- E_2

E_25<-Madrid_Data$energy_demand[25:nrow(Madrid_Data)] # Get all the data except in first row
Madrid_Data<-Madrid_Data[1:(nrow(Madrid_Data)-24),]

Madrid_Data$E_25 <- E_25

Madrid_Data <- Madrid_Data [, c("time_ID", "E_1","E_2","E_25",   "temp","humidity","pressure","wind_spe


### Generate Working and Evaluation Data
set.seed(501)
working_rows <- sample(1:nrow(Madrid_Data),0.80*nrow(Madrid_Data))

working_Madrid <-Madrid_Data[working_rows,]

working_Madrid$index <- as.numeric(row.names(working_Madrid))
working_Madrid<- working_Madrid[order(working_Madrid$index), ]

working_Madrid<-working_Madrid %>%
  relocate(index)
```

```r
evaluation_Madrid <-Madrid_Data[-working_rows,]
evaluation_Madrid$index <- as.numeric(row.names(evaluation_Madrid))
evaluation_Madrid<- evaluation_Madrid[order(evaluation_Madrid$index), ]

evaluation_Madrid<-evaluation_Madrid %>%
  relocate(index)

write.csv(working_Madrid,"Data/working_Madrid.csv", row.names = FALSE)
write.csv(evaluation_Madrid,"Data/evaluation_Madrid.csv", row.names = FALSE)
```