# STAT 652 Assignment 1

Dhruv Patel

13/10/2021

Lecture 4: Applications A
1. Compute a summary on TWcp and TWrat. Report the minimum, maximum, and mean for each variable.
Answer:

```
data = na.omit(airquality)
filter_data = (data[,1:4])
head(filter_data)
```

```
##   Ozone Solar.R Wind Temp
## 1    41     190  7.4   67
## 2    36     118  8.0   72
## 3    12     149 12.6   74
## 4    18     313 11.5   62
## 7    23     299  8.6   65
## 8    19      99 13.8   59
```

```
filter_data$TWcp = filter_data$Temp*filter_data$Wind
filter_data$TWrat = filter_data$Temp/filter_data$Wind
```

Min, Max and Mean values for TWcp are:

```
min(filter_data$TWcp)
```

```
## [1] 216.2
```

```
max(filter_data$TWcp)
```

```
## [1] 1490.4
```

```
mean(filter_data$TWcp)
```

```
## [1] 756.527
```

Min, Max and Mean values for TWrat are:

```
min(filter_data$TWrat)
```

```
## [1] 3.034826
```

```r
max(filter_data$TWrat)
```

```
## [1] 40.86957
```

```r
mean(filter_data$TWrat)
```

```
## [1] 9.419117
```

2. Create two new models: Temp + Wind + TWcp and Temp + Wind + TWrat. Fit these two models in lm().

(a) Report the t-test results for the two new variables.
Answer:
TWrat summary:

```r
lm_twrat = lm(Ozone ~ Temp + Wind + TWrat, data = filter_data)
summary(lm_twrat)
```

```
##
## Call:
## lm(formula = Ozone ~ Temp + Wind + TWrat, data = filter_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -55.241 -10.969  -3.506  11.568  80.805
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -85.5258    22.5920  -3.786 0.000253 ***
## Temp          1.4214     0.2557   5.559 2.01e-07 ***
## Wind         -0.6654     0.9090  -0.732 0.465756
## TWrat         2.5121     0.6272   4.005 0.000115 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.36 on 107 degrees of freedom
## Multiple R-squared:  0.636,  Adjusted R-squared:  0.6258
## F-statistic: 62.31 on 3 and 107 DF,  p-value: < 2.2e-16
```

TWcp summary:

```r
lm_twcp = lm(Ozone ~ Temp + Wind + TWcp, data = filter_data)
summary(lm_twcp)
```

```
##
## Call:
## lm(formula = Ozone ~ Temp + Wind + TWcp, data = filter_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -40.930 -11.193  -3.034   8.193  97.456
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -239.8918    48.6200  -4.934 2.97e-06 ***
## Temp           4.0005     0.5935   6.741 8.26e-10 ***
## Wind          13.5975     4.2835   3.174 0.001961 **
## TWcp          -0.2173     0.0545  -3.987 0.000123 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.37 on 107 degrees of freedom
## Multiple R-squared:  0.6355, Adjusted R-squared:  0.6253
## F-statistic: 62.19 on 3 and 107 DF,  p-value: < 2.2e-16
```

(b) Based on the test results, which variable seems to be the most useful, or are neither particularly helpful? (1 sentence)
Answer:
t values for TWcp and TWrat are -3.987 and 4.005 respectively.Since both values are below significance level(0.5). Both are important.

(c) From the model with the cross-product term, compute and report the slope of the Temp effect when Wind is at its minimum value. Repeat for the maximum value of Wind. (You can do this by hand from the output if you want.)

```
min(filter_data$Wind)
```

```
## [1] 2.3
```

```
max(filter_data$Wind)
```

```
## [1] 20.7
```

3. Fit each model on the training data and report the MSPEs from the validation data.

(a) Which model wins this competition?
Answer:

```
set.seed(2928893)
rows = nrow(filter_data)
train_split = 0.75
reorder_col = sample.int(n=rows, size=rows, replace=FALSE)
set = ifelse(test = ((train_split*rows) > reorder_col), yes=1, no=2)

train_data = filter_data[set==1,]
test_data = filter_data[set==2,]

fit.TWcp = lm(Ozone ~ Temp + Wind + TWcp, data = train_data)
fit.TWrat = lm(Ozone ~ Temp + Wind + TWrat, data = train_data)

pred.TWcp = predict(fit.TWcp, newdata=test_data)
```

```r
pred.TWrat = predict(fit.TWrat,newdata=test_data)

MSPE.TWcp = mean((test_data$Ozone - pred.TWcp)^2)
MSPE.TWrat = mean((test_data$Ozone - pred.TWrat)^2)

MSPE.TWcp
```

```
## [1] 286.4392
```

```r
MSPE.TWrat
```

```
## [1] 290.9852
```

So,when set.seed(2928893) is fixed. We get MSPE for TWcp as 286.4392 and TWrat as 290.9852. Which shows TWcp wins the competition.

4. Add these models the five you compared in the previous exercise, and rerun the CV 20 times.

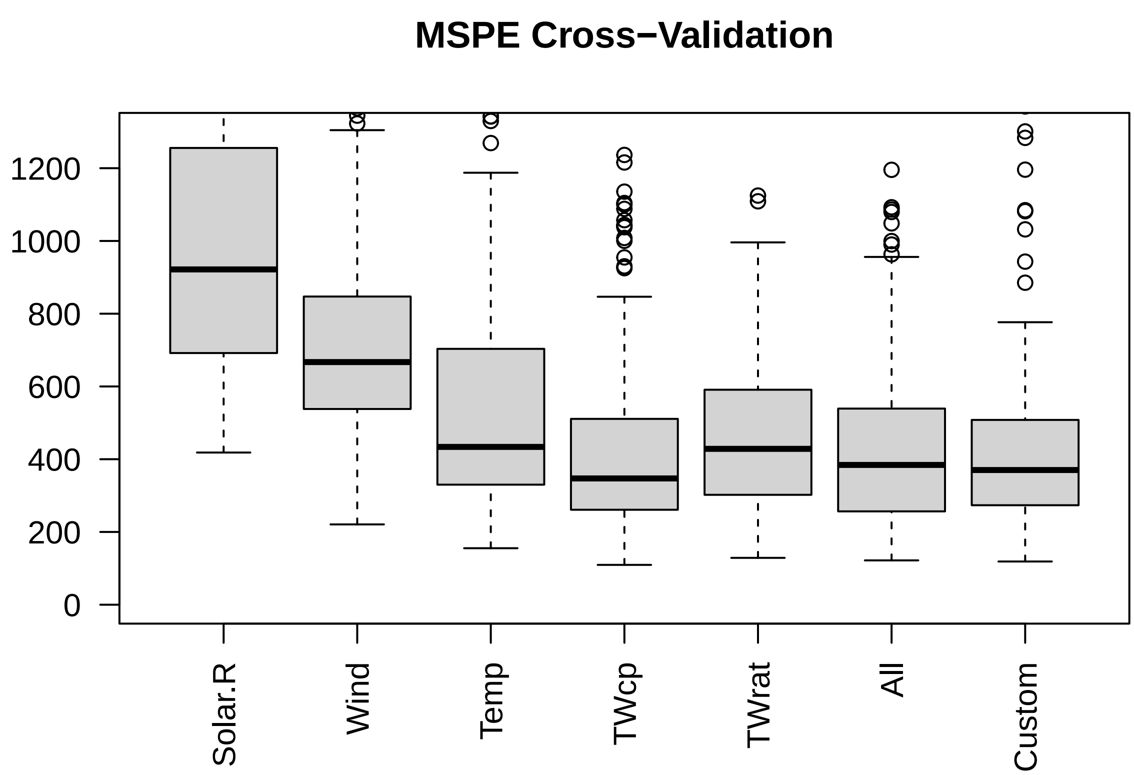
(a) Make boxplots of the RMSPE, and narrow focus if necessary to see best models better.
Answer: V = 7 corresponding to different models and R = 20 number of times it runs.

```r
knitr::opts_chunk$set(warning = FALSE, message = FALSE)

data$TWcp = data$Temp * data$Wind
data$TWrat = data$Temp / data$Wind

V=7
R=20

mat_CV = matrix(NA, nrow=V*R, ncol=7)
colnames(mat_CV) = c("Solar.R", "Wind", "Temp","TWcp","TWrat","All","Custom")

for (i in 1:R){
  folds = floor((sample.int(rows)-1)*V/rows) + 1
  for(j in 1:V){

    # Training Model
    fit.Solar.R = lm(Ozone ~ Solar.R, data = data[folds!=j,])
    fit.Wind = lm(Ozone ~ Wind, data = data[folds!=j,])
    fit.Temp = lm(Ozone ~Temp, data = data[folds!=j,])
    fit.TWcp = lm(Ozone ~ Temp + Wind + TWcp, data = data[folds!=j,])
    fit.TWrat = lm(Ozone ~ Temp + Wind + TWrat, data = data[folds!=j,])
    fit.All = lm(Ozone ~ ., data = data[folds!=j,])
    fit.Custom = lm(Ozone ~ .^2 + Solar.R^2 + Wind^2 + Temp^2, data = data[folds!=j,])

    # Model Prediction
    pred.Solar.R = predict(fit.Solar.R, newdata = data[folds==j,])
    pred.Wind = predict(fit.Wind, newdata = data[folds==j,])
    pred.Temp = predict(fit.Temp, newdata = data[folds==j,])
    pred.TWcp = predict(fit.TWcp, newdata = data[folds==j,])
```

```
    pred.TWrat = predict(fit.TWrat,newdata = data[folds==j,])
    pred.All = predict(fit.All, newdata = data[folds==j,])
    pred.Custom = predict(fit.Custom,newdata = data[folds==j,])

    r = j+V*(i-1)
    # Calculating MSPE for each attributes
    mat_CV[r,1] = mean((data[folds==j,"Ozone"] - pred.Solar.R)^2)
    mat_CV[r,2] = mean((data[folds==j,"Ozone"] - pred.Wind)^2)
    mat_CV[r,3] = mean((data[folds==j,"Ozone"] - pred.Temp)^2)
    mat_CV[r,4] = mean((data[folds==j,"Ozone"] - pred.TWcp)^2)
    mat_CV[r,5] = mean((data[folds==j,"Ozone"] - pred.TWrat)^2)
    mat_CV[r,6] = mean((data[folds==j,"Ozone"] - pred.All)^2)
    mat_CV[r,7] = mean((data[folds==j,"Ozone"] - pred.Custom)^2)
  }
}
```

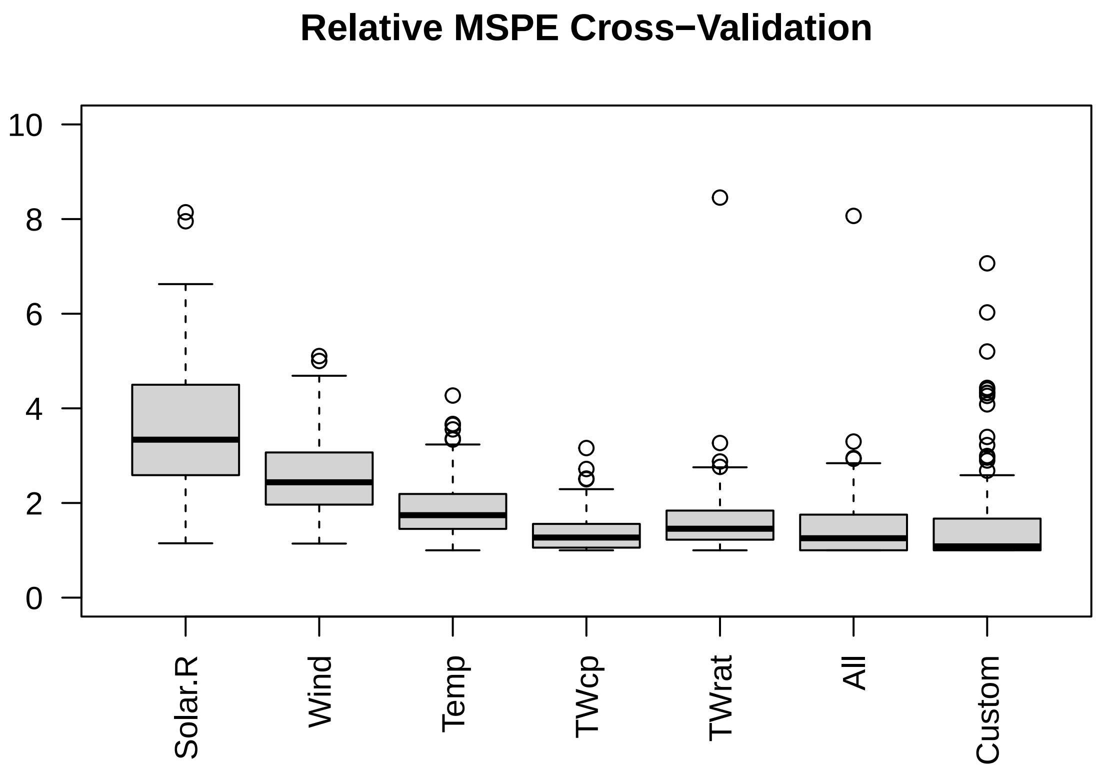
MSPE Cross-Validation Boxplot:

```
# MSPE Boxplot
boxplot(mat_CV, las=2, ylim=c(0,1300),main="MSPE Cross-Validation")
```




Relative MSPE Cross-Validation Boxplot:

```
rel_CV = mat_CV/apply(mat_CV, 1, min)
boxplot(rel_CV, las=2,ylim=c(0,10),main="Relative MSPE Cross-Validation")
```




(b) Are any of the new models competitive, or even best? (1 sentence)
Answer: The model with second-order for three varialbes (Solar.R, Wind and Temp) is best model till now.