Homework 1

Asif Hasan - 301376671

2023-09-22

1.

Yes, I have read and acknowledge the SFU Student Academic Integrity Policy.

2. Problem Set 3, Question 2

a(i).

We would expect to see lower bias in the linear model compared to Figure 6 because the true model has less curvature than in Figure 6.

a(ii).

We would expect to see higher variance in the linear model compared to Figure 6 if we used the same sample given in Figure 6.

b.

As we increase the sample size in this situation, we expect only the variance to get smaller.

3. Problem Set 4, Application

Question 1

```
# get the data
air.data <- airquality
head(air.data)
```

```
##   Ozone Solar.R Wind Temp Month Day
## 1    41     190  7.4   67     5   1
## 2    36     118  8.0   72     5   2
## 3    12     149 12.6   74     5   3
## 4    18     313 11.5   62     5   4
## 5    NA      NA 14.3   56     5   5
## 6    28      NA 14.9   66     5   6
```

```
# get the dimensions
dim(air.data)
```

```
## [1] 153   6
```

```
# remove NA
air.data2 <- na.omit(airquality[ ,1:4])
dim(air.data2)
```

```
## [1] 111   4
```

We got rid of 42 rows of data with missing values and 2 columns that we will not be using for fitting regression models.

Question 2

```
set.seed(4099183)
# get number of rows
n <- nrow(air.data2)
# set sampling fraction
sf <- 0.75
# generate sample
reorder <- sample.int(n)
set <- ifelse(test = (reorder < sf * n), yes = 1, no=2)
# show observations in the validation set
air.data2[set==2, ]
```

```
##     Ozone Solar.R Wind Temp
## 1      41     190  7.4   67
## 4      18     313 11.5   62
## 12     16     256  9.7   69
## 15     18      65 13.2   58
## 16     14     334 11.5   64
## 17     34     307 12.0   66
## 19     30     322 11.5   68
## 21      1       8  9.7   59
## 22     11     320 16.6   73
## 24     32      92 12.0   61
## 29     45     252 14.9   81
## 41     39     323 11.5   87
## 44     23     148  8.0   82
## 50     12     120 11.5   73
## 63     49     248  9.2   85
## 67     40     314 10.9   83
## 70     97     272  5.7   92
## 71     85     175  7.4   89
## 81     63     220 11.5   85
## 82     16       7  6.9   74
## 85     80     294  8.6   86
## 92     59     254  9.2   81
## 99    122     255  4.0   89
## 108    22      71 10.3   77
## 113    21     259 15.5   77
## 128    47      95  7.4   87
## 137     9      24 10.9   71
## 153    20     223 11.5   68
```

Question 3

```
# fit 5 models on the train set
model.solar <- lm(Ozone ~ Solar.R, data=air.data2[set==1, ])
model.wind <- lm(Ozone ~ Wind, data=air.data2[set==1, ])
model.temp <- lm(Ozone ~ Temp, data=air.data2[set==1, ])
model.all <- lm(Ozone ~ ., data=air.data2[set==1, ])
model.comp <- lm(Ozone ~ Temp+Wind+Solar.R+I(Temp^2)+I(Wind^2)+I(Solar.R^2)
                 +Temp*Wind+Temp*Solar.R+Wind*Solar.R, data=air.data2[set==1, ])
# predict Ozone using the fitted models on validation set
pred.solar <- predict(model.solar, newdata=air.data2[set==2, ])
pred.wind <- predict(model.wind, newdata=air.data2[set==2, ])
pred.temp <- predict(model.temp, newdata=air.data2[set==2, ])
pred.all <- predict(model.all, newdata=air.data2[set==2, ])
pred.comp <- predict(model.comp, newdata=air.data2[set==2, ])
```

```
# calculate MSPE for the 5 models
(MSPE.solar <- mean((air.data2[set==2, "Ozone"] - pred.solar)^2))
```

```
## [1] 903.2816
```

```
(MSPE.wind <- mean((air.data2[set==2, "Ozone"] - pred.wind)^2))
```

```
## [1] 541.2048
```

```
(MSPE.temp <- mean((air.data2[set==2, "Ozone"] - pred.temp)^2))
```

```
## [1] 409.8842
```

```
(MSPE.all <- mean((air.data2[set==2, "Ozone"] - pred.all)^2))
```

```
## [1] 262.7439
```

```
(MSPE.comp <- mean((air.data2[set == 2, "Ozone"] - pred.comp)^2))
```

```
## [1] 271.8901
```

a.

The model with all variables denoted as ‘model.all’ with the formula = Temp + Wind + Solar.R is the best model out the the five fitted models.

Question 4

```
# set number of folds
V <- 5
# sample the folds
folds <- floor((sample.int(n) - 1) * V / n) + 1
# create matrix for MSPEs for 5 models
MSPEs.cv <- matrix(NA, nrow = V, ncol = 5)
colnames(MSPEs.cv) <- c("solar-c", "wind-c", "temp-c", "all-c", "comp-c")
# run cross-validation in for-loop
for (v in 1:V) {
  # fit 5 models on fold == !v
  model.solar.cv <- lm(Ozone ~ Solar.R, data=air.data2[folds!=v, ])
  model.wind.cv <- lm(Ozone ~ Wind, data=air.data2[folds!=v, ])
  model.temp.cv <- lm(Ozone ~ Temp, data=air.data2[folds!=v, ])
  model.all.cv <- lm(Ozone ~ ., data=air.data2[folds!=v, ])
  model.comp.cv <- lm(Ozone ~ Temp+Wind+Solar.R+I(Temp^2)+I(Wind^2)+I(Solar.R^2)
                 +Temp*Wind+Temp*Solar.R+Wind*Solar.R, data=air.data2[folds!=v, ])

  # predict Ozone using the fitted models on fold == v
  pred.solar.cv <- predict(model.solar.cv, newdata=air.data2[folds==v, ])
  pred.wind.cv <- predict(model.wind.cv, newdata=air.data2[folds==v, ])
  pred.temp.cv <- predict(model.temp.cv, newdata=air.data2[folds==v, ])
  pred.all.cv <- predict(model.all.cv, newdata=air.data2[folds==v, ])
  pred.comp.cv <- predict(model.comp.cv, newdata=air.data2[folds==v, ])

  # calculated MSPEs for 5 models for each v fold
  MSPEs.cv[v, 1] <- mean((air.data2[folds==v, "Ozone"] - pred.solar.cv)^2)
  MSPEs.cv[v, 2] <- mean((air.data2[folds==v, "Ozone"] - pred.wind.cv)^2)
  MSPEs.cv[v, 3] <- mean((air.data2[folds==v, "Ozone"] - pred.temp.cv)^2)
  MSPEs.cv[v, 4] <- mean((air.data2[folds==v, "Ozone"] - pred.all.cv)^2)
  MSPEs.cv[v, 5] <- mean((air.data2[folds==v, "Ozone"] - pred.comp.cv)^2)
}
##MSPEs.cv
# calculate mean MSPEs of v folds
(MSPEcv <- apply(X = MSPEs.cv, MARGIN = 2, FUN = mean))
```

```
##   solar-c    wind-c    temp-c     all-c    comp-c
## 1050.8892  746.5681  607.1550  476.7882  371.7492
```

```
# calculate 95% CI for each model
MSPEcv.sd <- apply(X = MSPEs.cv, MARGIN = 2, FUN = sd)
MSPEcv.CIl <- MSPEcv - qt(p = .975, df = V - 1) * MSPEcv.sd / sqrt(V)
MSPEcv.CIu <- MSPEcv + qt(p = .975, df = V - 1) * MSPEcv.sd / sqrt(V)
round(cbind(MSPEcv.CIl, MSPEcv.CIu), 2)
```

```
##          MSPEcv.CIl MSPEcv.CIu
## solar-c     514.78    1587.00
## wind-c      402.28    1090.86
## temp-c      171.45    1042.86
## all-c       178.09     775.49
## comp-c      198.08     545.42
```

a.

The two good models for prediction are as follows: The first model, ‘model.comp,’ incorporates curvature and interactions with the formula = Temp + Wind + Solar.R + (Temp^2) + (Wind^2) + (Solar.R^2) + Temp*Wind + Temp*Solar.R + Wind*Solar.R. The second model, denoted as ‘model.all,’ utilizes all three variables with the formula = Solar.R + Wind + Temp. In contrast, the three models using a single variable each are considered poor choices for prediction.

Question 5

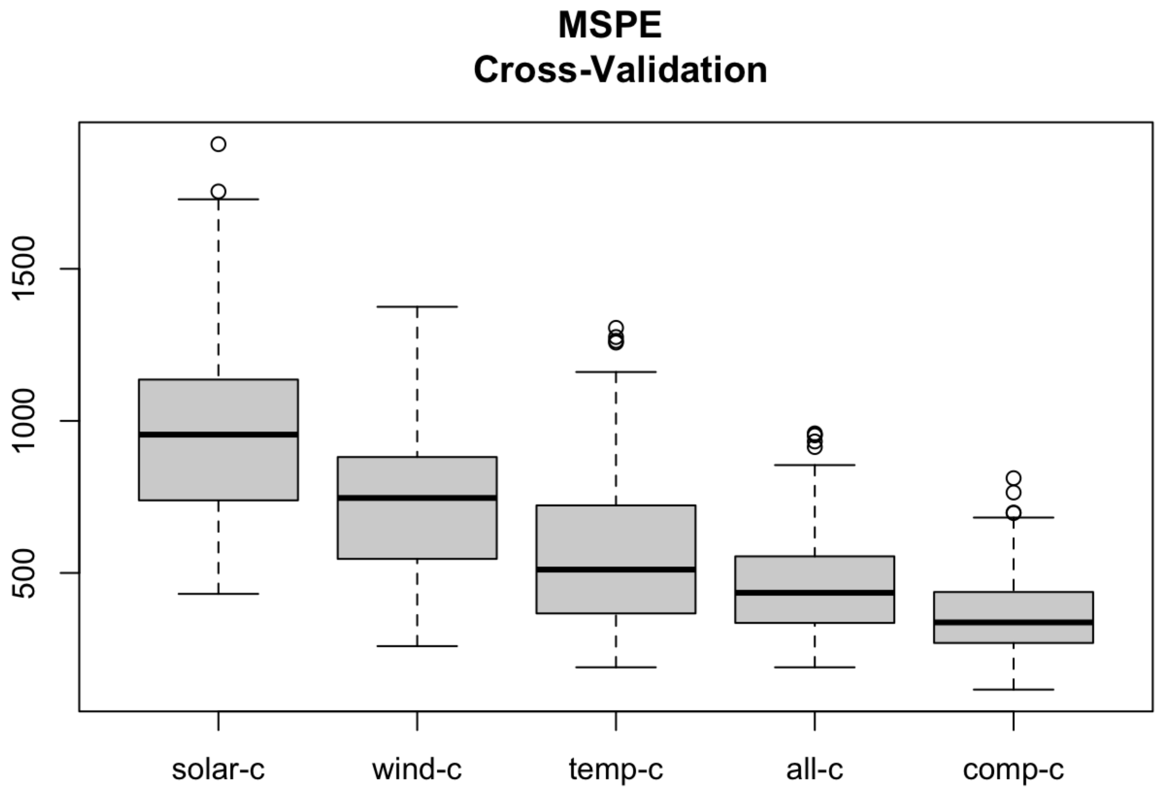
```
# repeat cross-validation 20 times
R <- 20
# create matrix for MSPEs for 5 models
MSPEs.cv20 <- matrix(NA, nrow = V * R, ncol = 5)
colnames(MSPEs.cv20) <- c("solar-c", "wind-c", "temp-c", "all-c", "comp-c")
# run 20 times
for (r in 1:R) {
  # sample the folds
  folds <- floor((sample.int(n) - 1) * V / n) + 1
  # run cross-validation each run
  for (v in 1:V) {
    # fit 5 models on fold == !v
    model.solar.cv <- lm(Ozone ~ Solar.R, data=air.data2[folds!=v, ])
    model.wind.cv <- lm(Ozone ~ Wind, data=air.data2[folds!=v, ])
    model.temp.cv <- lm(Ozone ~ Temp, data=air.data2[folds!=v, ])
    model.all.cv <- lm(Ozone ~ ., data=air.data2[folds!=v, ])
    model.comp.cv <- lm(Ozone ~ Temp+Wind+Solar.R+I(Temp^2)+I(Wind^2)+I(Solar.R^2)
                        +Temp*Wind+Temp*Solar.R+Wind*Solar.R, data=air.data2[folds!=v, ])

    # predict Ozone using the fitted models on fold == v
    pred.solar.cv <- predict(model.solar.cv, newdata=air.data2[folds==v, ])
    pred.wind.cv <- predict(model.wind.cv, newdata=air.data2[folds==v, ])
    pred.temp.cv <- predict(model.temp.cv, newdata=air.data2[folds==v, ])
    pred.all.cv <- predict(model.all.cv, newdata=air.data2[folds==v, ])
    pred.comp.cv <- predict(model.comp.cv, newdata=air.data2[folds==v, ])

    # calculated MSPEs for 5 models for each v fold
    MSPEs.cv20[(r - 1) * V + v, 1] <- mean((air.data2[folds==v, "Ozone"] - pred.solar.cv)^
2)
    MSPEs.cv20[(r - 1) * V + v, 2] <- mean((air.data2[folds==v, "Ozone"] - pred.wind.cv)^2)
    MSPEs.cv20[(r - 1) * V + v, 3] <- mean((air.data2[folds==v, "Ozone"] - pred.temp.cv)^2)
    MSPEs.cv20[(r - 1) * V + v, 4] <- mean((air.data2[folds==v, "Ozone"] - pred.all.cv)^2)
    MSPEs.cv20[(r - 1) * V + v, 5] <- mean((air.data2[folds==v, "Ozone"] - pred.comp.cv)^2)
  }
}
##MSPEs.cv20
##(MSPEcv20 <- apply(X = MSPEs.cv20, MARGIN = 2, FUN = mean))
```

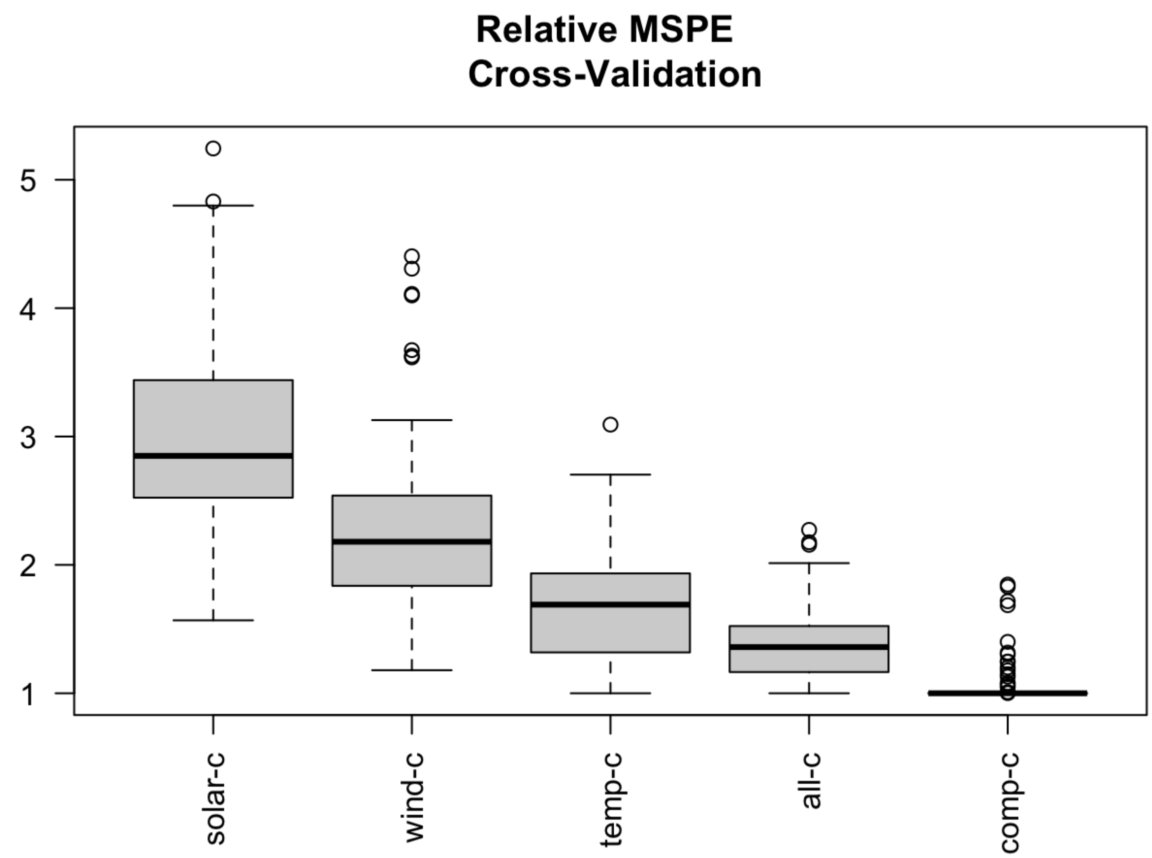
a.

```
# create boxplots for MSPEs
boxplot(MSPEs.cv20, main = "MSPE \n Cross-Validation")
```



b.

```
# create boxplots for RMSPEs
low.cv <- apply(MSPEs.cv20, 1, min)
boxplot(MSPEs.cv20 / low.cv,
        las = 2,
        main = "Relative MSPE \n Cross-Validation"
)
```



Question 6

Based on the analysis, I would suggest using the model that allows curvature and interactions denoted as 'model.comp' with the formula = Temp + Wind + Solar.R + (Temp^2) + (Wind^2) + (Solar.R^2) + Temp*Wind + Temp*Solar.R + Wind*Solar.R as it has the lowest MSPE among all five models and would be the best model for prediction.

4. Problem Set, Question 5B, Categorical Explanatories

```
# read data
ins <- read.csv("Insurance.csv", header=TRUE)
# convert zone and make to categorical vars
ins$zone <- as.factor(ins$zone)
ins$make <- as.factor(ins$make)
##class(ins$zone)
##class(ins$make)
# remove claims == 0
ins <- ins[ins$claims>0,]
##nrow(ins)
```

a(i).

```
# build a model using all vars
model <- lm(per ~ ., data=ins)
summary(model)
```

```
##
## Call:
## lm(formula = per ~ ., data = ins)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.0994 -0.7170  0.0734  0.8393  3.7574
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.186e+01  1.321e-01  89.770  < 2e-16 ***
## km          -3.434e-01  2.064e-02 -16.641  < 2e-16 ***
## zone2       -1.376e-01  9.717e-02  -1.416    0.157
## zone3       -2.143e-02  9.753e-02  -0.220    0.826
## zone4        4.317e-01  9.692e-02   4.454 8.95e-06 ***
## zone5       -1.042e+00  1.043e-01  -9.983  < 2e-16 ***
## zone6       -4.440e-01  1.009e-01  -4.401 1.14e-05 ***
## zone7       -2.862e+00  1.378e-01 -20.767  < 2e-16 ***
## bonus        2.301e-01  1.405e-02  16.381  < 2e-16 ***
## make2       -1.403e+00  1.140e-01 -12.314  < 2e-16 ***
## make3       -1.710e+00  1.189e-01 -14.382  < 2e-16 ***
## make4       -1.834e+00  1.240e-01 -14.789  < 2e-16 ***
## make5       -1.317e+00  1.138e-01 -11.568  < 2e-16 ***
## make6       -8.253e-01  1.129e-01  -7.312 3.95e-13 ***
## make7       -1.716e+00  1.153e-01 -14.878  < 2e-16 ***
## make8       -2.070e+00  1.199e-01 -17.260  < 2e-16 ***
## make9        1.459e+00  1.209e-01  12.071  < 2e-16 ***
## insured     -5.724e-05  1.151e-05  -4.975 7.15e-07 ***
## claims       3.029e-03  3.519e-04   8.608  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.179 on 1778 degrees of freedom
## Multiple R-squared:  0.6477, Adjusted R-squared:  0.6442
## F-statistic: 181.6 on 18 and 1778 DF,  p-value: < 2.2e-16
```

A total of 18 parameters and the intercept, are estimated in the model.

a(ii).

When make and zone are both at their first levels, 1, the intercept of the regression model is 11.86.

a(iii).

when make and zone are both at their last levels, 9 and 7, respectively,the intercept of the regression model is 10.457 (11.86 + 1.459 -2.862).