```r
1   data2014 <-

    read.csv('/Users/Hamster/Desktop/Capstone_Data/2014data.csv',

    header=TRUE,sep = ",")

2   scatter.smooth(x=data2014$average_income,

    y=data2014$average_test_score, main="average_test_score ~

    average_income")

3   par(mfrow=c(1, 2))

4   boxplot(data2014$average_income, main="Average Income",

    sub=paste("Outlier rows: ", boxplot.stats(data2014$average_income)$out))

5   boxplot(data2014$average_test_score, main="Average Test Scores",

    sub=paste("Outlier rows: ",

    boxplot.stats(data2014$average_test_score)$out))

6   library(e1071)

7   par(mfrow=c(1, 2))
```

```r
8   plot(density(data2014$average_test_score), main="Density Plot: Avg
    Test Score", ylab="Frequency", sub=paste("Skewness:",
    round(e1071::skewness(data2014$average_test_score), 2)))

9   polygon(density(data2014$average_test_score), col="red")

10  plot(density(data2014$average_income), main="Density Plot: Avg
    Income", ylab="Frequency", sub=paste("Skewness:",
    round(e1071::skewness(data2014$average_income), 2)))

11  cor(data2014$average_test_score, data2014$average_income)

12  lm2014.svr <- lm(average_test_score ~ average_income,
    data=data2014)

13  print(lm2014.svr)

14  summary(lm2014.svr)

15  modelSummary_svr2014 <- summary(lm2014.svr)

16  modelCoeffs_svr2014 <- modelSummary_svr2014$coefficients
```

```r
17  beta.estimate_svr2014 <- modelCoeffs_svr2014["average_income", "Estimate"]

18  std.error_svr2014 <- modelCoeffs_svr2014["average_income", "Std. Error"]

19  t_value_svr2014 <- beta.estimate_svr2014/std.error_svr2014

20  t_value_svr2014

21  p_value_svr2014 <- 2*pt(-abs(t_value_svr2014), df=nrow(data2014)-ncol(data2014))

22  p_value_svr2014

23  f_statistic_svr2014 <- lm2014.svr$fstatistic[1]

24  f_svr2014 <- summary(lm2014.svr)$fstatistic

25  f_svr2014

26  model_p_svr2014 <- pf(f_svr2014[1], f_svr2014[2], f_svr2014[3], lower=FALSE)
```

```
27   model_p_svr2014

28   AIC(lm2014.svr)

29   BIC(lm2014.svr)

30   set.seed(100)

31   trainingRowIndex_svr2014 <- sample(1:nrow(data2014),

     0.8*nrow(data2014))

32   trainingData_svr2014 <- data2014[trainingRowIndex_svr2014, ]

33   testData_svr2014  <- data2014[-trainingRowIndex_svr2014, ]

34   lmMod_svr2014 <- lm(average_test_score ~ average_income,

     data=trainingData_svr2014)

35   testScorePred_svr2014 <- predict(lmMod_svr2014, testData_svr2014)

36   summary (lmMod_svr2014)

37   AIC (lmMod_svr2014)
```

```
38  actuals_preds_svr2014 <-

    data.frame(cbind(actuals=testData_svr2014$average_test_score,

    predicteds=testScorePred_svr2014))

39  correlation_accuracy_svr2014 <- cor(actuals_preds_svr2014)

40  correlation_accuracy_svr2014

41  head(actuals_preds_svr2014)

42  min_max_accuracy_svr2014 <- mean(apply(actuals_preds_svr2014,

    min) / apply(actuals_preds_svr2014, 1, max))

43  min_max_accuracy_svr2014

44  mape_svr2014 <- mean(abs((actuals_preds_svr2014$predicteds -

    actuals_preds_svr2014$actuals))/actuals_preds_svr2014$actuals)

45  mape_svr2014

46  library(DAAG)
```

```
47  cvResults_svr2014 <- suppressWarnings(CVlm(data=data2014,

    form.lm=average_test_score ~ average_income, m=5, dots=FAL

    seed=29, legend.pos="topleft", printit=FALSE, main="Small symbol

    are predicted values while bigger ones are actuals."));

    attr(cvResults_svr2014, 'ms')

48  lm2014.mvr1 <- lm(average_test_score ~

    average_income+race_white+household_married, data=data2014)

49  print(lm2014.mvr1)

50  summary(lm2014.mvr1)

51  AIC(lm2014.mvr1)

52  BIC(lm2014.mvr1)

53  library(DAAG)
```

```
54   cvResults_mvr_1_2014 <- suppressWarnings(CVlm(data=data2014,

     form.lm=average_test_score ~

     average_income+race_white+household_married, m=5, dots=FALSE,

     seed=29, legend.pos="topleft",  printit=FALSE, main="Small symbols

     are predicted values while bigger ones are actuals."));

     attr(cvResults_mvr_1_2014, 'ms')

55   lm2014.mvr2 <- lm(average_test_score ~

     average_income+race_white+education, data=data2014)

56   print(lm2014.mvr2)

57   summary(lm2014.mvr2)

58   AIC(lm2014.mvr2)

59   BIC(lm2014.mvr2)

60   library(DAAG)
```

```
61  cvResults_mvr_2_2014 <- suppressWarnings(CVlm(data=data2014,

    form.lm=average_test_score ~

    average_income+race_white+education, m=5, dots=FALSE, seed=29,

    legend.pos="topleft", printit=FALSE, main="Small symbols are

    predicted values while bigger ones are actuals."));

    attr(cvResults_mvr_2_2014, 'ms')

62  lm2014.mvr3 <- lm(average_test_score ~

    average_income+race_white+household_married+education,

    data=data2014)

63  print(lm2014.mvr3)

64  summary(lm2014.mvr3)

65  AIC(lm2014.mvr3)

66  BIC(lm2014.mvr3)
```

```
68  cvResults_mvr_3_2014 <- suppressWarnings(CVlm(data=data2014,

    form.lm=average_test_score ~

    average_income+race_white+household_married+education, m=5,

    dots=FALSE, seed=29, legend.pos="topleft", printit=FALSE,

    main="Small symbols are predicted values while bigger ones are

    actuals.")); attr(cvResults_mvr_3_2014, 'ms')

69  t(names(data2014))

70  data2014cor_mvr_3_2014 <- subset(data2014[c(3,4,5,6)])

71  cor(data2014cor_mvr_3_2014)

72  data2007 <-

    read.csv('/Users/Hamster/Desktop/Capstone_Data/2007data.csv',

    header=TRUE,sep = ",")
```

```
73   scatter.smooth(x=data2007$average_income,

     y=data2007$average_test_score, main="average_test_score ~

     average_income")

74   par(mfrow=c(1, 2))

75   boxplot(data2007$average_income, main="Average Income",

     sub=paste("Outlier rows: ",

     boxplot.stats(data2007$average_income)$out))

76   boxplot(data2007$average_test_score, main="Average Test Scores",

     sub=paste("Outlier rows: ",

     boxplot.stats(data2007$average_test_score)$out))

77   library(e1071)

78   par(mfrow=c(1, 2))
```

```
79  plot(density(data2007$average_test_score), main="Density Plot: Avg

    Test Score", ylab="Frequency", sub=paste("Skewness:",

    round(e1071::skewness(data2007$average_test_score), 2)))

80  polygon(density(data2007$average_test_score), col="red")

81  plot(density(data2007$average_income), main="Density Plot: Avg

    Income", ylab="Frequency", sub=paste("Skewness:",

    round(e1071::skewness(data2007$average_income), 2)))

82  cor(data2007$average_test_score, data2007$average_income)

83  lm2007.svr <- lm(average_test_score ~ average_income,

    data=data2007)

84  print(lm2007.svr)

85  summary(lm2007.svr)

86  modelSummary_svr2007 <- summary(lm2007.svr)

87  modelCoeffs_svr2007 <- modelSummary_svr2007$coefficients
```

```
88  beta.estimate_svr2007 <- modelCoeffs_svr2007["average_income",

    "Estimate"]

89  std.error_svr2007 <- modelCoeffs_svr2007["average_income", "std.

    Error"]

90  t_value_svr2007 <- beta.estimate_svr2007/std.error_svr2007

91  t_value_svr2007

92  p_value_svr2007 <- 2*pt(-abs(t_value_svr2007), df=nrow(data2007)-

    ncol(data2007))

93  p_value_svr2007

94  f_statistic_svr2007 <- lm2007.svr$fstatistic[1]

95  f_svr2007 <- summary(lm2007.svr)$fstatistic

96  f_svr2007

97  model_p_svr2007 <- pf(f_svr2007[1], f_svr2007[2], f_svr2007[3],

    lower=FALSE)
```

```
98    model_p_svr2007

99    AIC(lm2007.svr)

100   BIC(lm2007.svr)

101   set.seed(100)

102   trainingRowIndex_svr2007 <- sample(1:nrow(data2007),

      0.8*nrow(data2007))

103   trainingData_svr2007 <- data2007[trainingRowIndex_svr2007, ]

104   testData_svr2007 <- data2007[-trainingRowIndex_svr2007, ]

105   lmMod_svr2007 <- lm(average_test_score ~ average_income,

      data=trainingData_svr2007)

106   testscorePred_svr2007 <- predict(lmMod_svr2007, testData_svr2007)

107   summary (lmMod_svr2007)

108   AIC (lmMod_svr2007)
```

```
109   actuals_preds_svr2007 <-

      data.frame(cbind(actuals=testData_svr2007$average_test_score,

      predicteds=testscorePred_svr2007))

110   correlation_accuracy_svr2007 <- cor(actuals_preds_svr2007)

111   correlation_accuracy_svr2007

112   head(actuals_preds_svr2007)

113   min_max_accuracy_svr2007 <- mean(apply(actuals_preds_svr2007, 1,

      min) / apply(actuals_preds_svr2007, 1, max))

114   min_max_accuracy_svr2007

115   mape_svr2007 <- mean(abs((actuals_preds_svr2007$predicteds -

      actuals_preds_svr2007$actuals))/actuals_preds_svr2007$actuals)

116   mape_svr2007

117   library(DAAG)
```

```r
118  cvResults_svr2007 <- suppressWarnings(CVlm(data=data2007,

     form.lm=average_test_score ~ average_income, m=5, dots=FALSE,

     seed=29, legend.pos="topleft", printit=FALSE, main="Small symbols

     are predicted values while bigger ones are actuals."));

     attr(cvResults_svr2007, 'ms')

119  lm2007.mvr1 <- lm(average_test_score ~

     average_income+race_white+household_married, data=data2007)

120  print(lm2007.mvr1)

121  summary(lm2007.mvr1)

122  AIC(lm2007.mvr1)

123  BIC(lm2007.mvr1)

124  library(DAAG)
```

```r
125  cvResults_mvr_1_2007 <- suppressWarnings(CVlm(data=data2007,
     form.lm=average_test_score ~
     average_income+race_white+household_married, m=5, dots=FALSE,
     seed=29, legend.pos="topleft", printit=FALSE, main="Small symbols
     are predicted values while bigger ones are actuals."));
     attr(cvResults_mvr_1_2007, 'ms')
126  lm2007.mvr2 <- lm(average_test_score ~
     average_income+race_white+education, data=data2007)
127  print(lm2007.mvr2)
128  summary(lm2007.mvr2)
129  AIC(lm2007.mvr2)
130  BIC(lm2007.mvr2)
131  library(DAAG)
```

```
132  cvResults_mvr_2_2007 <- suppressWarnings(CVlm(data=data2007,

     form.lm=average_test_score ~

     average_income+race_white+education, m=5, dots=FALSE, seed=29

     legend.pos="topleft", printit=FALSE, main="Small symbols are

     predicted values while bigger ones are actuals."));

     attr(cvResults_mvr_2_2007, 'ms')

133  lm2007.mvr3 <- lm(average_test_score ~

     average_income+race_white+household_married+education,

     data=data2007)

134  print(lm2007.mvr3)

135  summary(lm2007.mvr3)

136  AIC(lm2007.mvr3)

137  BIC(lm2007.mvr3)

138  library(DAAG)
```

```
139  cvResults_mvr_3_2007 <- suppressWarnings(CVlm(data=data2007,

     form.lm=average_test_score ~

     average_income+race_white+household_married+education, m=5,

     dots=FALSE, seed=29, legend.pos="topleft", printit=FALSE,

     main="Small symbols are predicted values while bigger ones are

     actuals.")); attr(cvResults_mvr_3_2007, 'ms')

140  t(names(data2007))

141  data2007cor_mvr_3_2007 <- subset(data2007[c(3,4,5,6)])

142  cor(data2007cor_mvr_3_2007)
```