# STAD80: Assignment 2

Vladislav Trukhin

Due: Feb 3

## Contents



## Question 1

### 1a

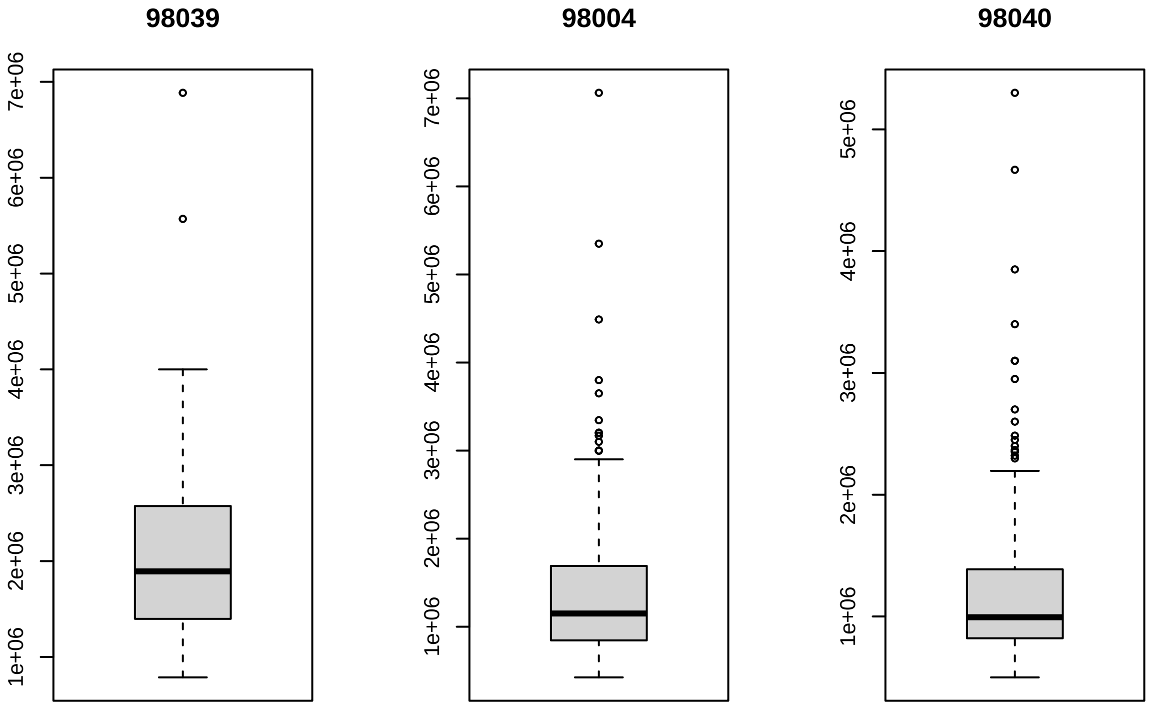
```
data <- read.csv("/Users/vladislavtrukhin/Downloads/_data_hw2/housingprice.csv")

mean_prices <- tapply(data$price, data$zipcode, mean)
sorted_mean_prices <- sort(mean_prices, decreasing = TRUE)
labels(sorted_mean_prices) # Zipcode order by avg housing price
```

```
## [[1]]
##  [1] "98039" "98004" "98040" "98112" "98102" "98109" "98105" "98006" "98119"
## [10] "98005" "98033" "98199" "98075" "98074" "98077" "98053" "98177" "98008"
## [19] "98052" "98122" "98115" "98116" "98007" "98027" "98029" "98144" "98103"
## [28] "98024" "98107" "98117" "98072" "98136" "98065" "98034" "98059" "98011"
## [37] "98070" "98125" "98166" "98028" "98014" "98045" "98019" "98126" "98155"
## [46] "98010" "98056" "98118" "98133" "98038" "98146" "98108" "98058" "98092"
## [55] "98106" "98022" "98042" "98178" "98055" "98198" "98031" "98030" "98003"
## [64] "98188" "98023" "98148" "98001" "98032" "98168" "98002"
```

```
top_mean_prices <- sorted_mean_prices[0:3]
labels(top_mean_prices) # Top 3 zipcodes with most expensive avg housing price
```

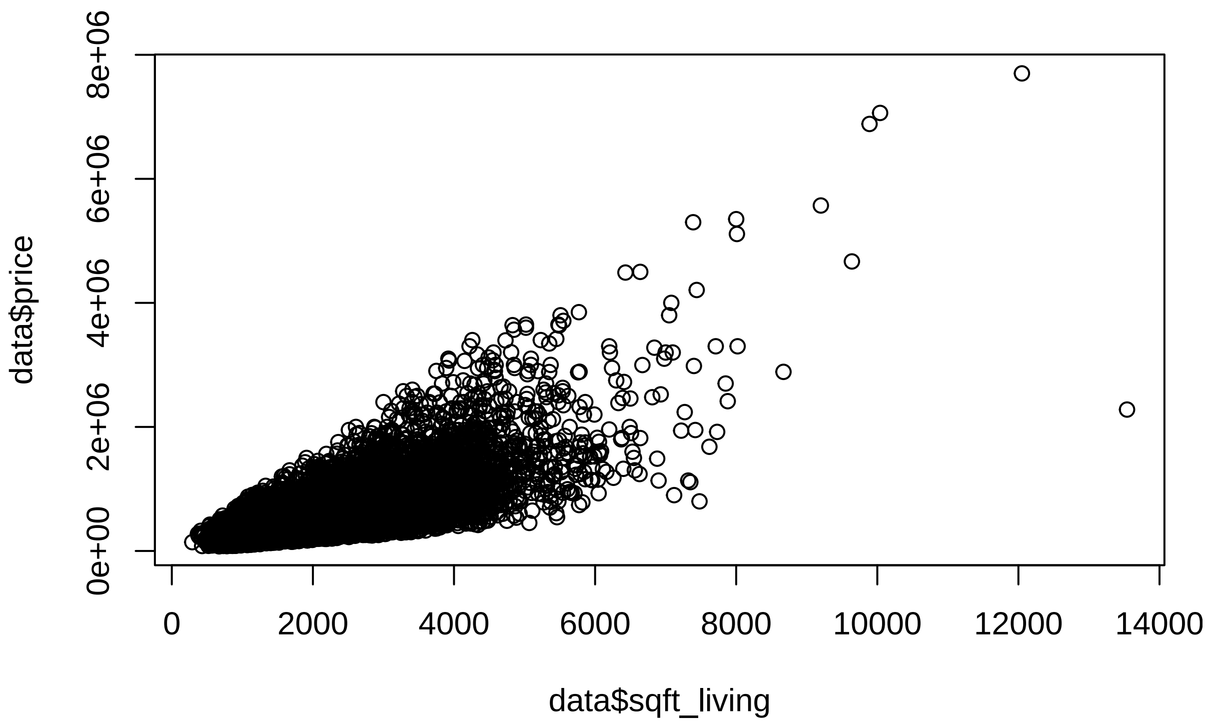
```
## [[1]]
## [1] "98039" "98004" "98040"
```

```
par(mfrow=c(1,3))
boxplot(data[which(data$zipcode == labels(top_mean_prices[1])),]$price)
title(labels(top_mean_prices[1]))
boxplot(data[which(data$zipcode == labels(top_mean_prices[2])),]$price)
title(labels(top_mean_prices[2]))
boxplot(data[which(data$zipcode == labels(top_mean_prices[3])),]$price)
title(labels(top_mean_prices[3]))
```




**1b**

```
plot(data$sqft_living, data$price)
```




**1c**

```r
train <- read.csv("/Users/vladislavtrukhin/Downloads/_data_hw2/train.data.csv")
test <- read.csv("/Users/vladislavtrukhin/Downloads/_data_hw2/test.data.csv")

fit <- lm(price ~ bedrooms + bathrooms + sqft_living + sqft_lot, train)
summary(fit)$r.squared # Training R2
```

```
## [1] 0.5101139
```

```r
cor(predict(fit, test), test$price)^2 # Test R2
```

```
## [1] 0.5050777
```

### 1d

```r
fit <- lm(price ~ zipcode +
            bedrooms +
            bathrooms +
            sqft_living +
            sqft_lot, train)
summary(fit)$r.squared # Training R2
```

```
## [1] 0.5162971
```

```r
cor(predict(fit, test), test$price)^2 # Test R2
```

```
## [1] 0.5120952
```

### 1e

```r
fancy = read.csv("/Users/vladislavtrukhin/Downloads/_data_hw2/fancyhouse.csv")
predict(fit, fancy) # Predicted price
```

```
##        1
## 15642273
```

The predicted price is not reasonable as the actual price of the home is $100+ million, which makes the predicted 10 times off of the actual price.

### 1f

$R^2 = 1 - \frac{RSS}{TSS}$

As the value of TSS is the same for both models, only need to observe their respective RSS.

$RSS_{d+1} - RSS_d = ||\mathbf{Y} - \mathbf{X}_{d+1}\hat{\beta_{d+1}}||_2^2 - ||\mathbf{Y} - \mathbf{X}_d\hat{\beta}_d||_2^2$

$= \sum_{i=1}^n (y_i - \hat{\beta}_{d+1,0} + \hat{\beta}_{d+1,1}x_{i,1} + ... + \hat{\beta}_{d+1,d+1}x_{i,d+1})^2 - \sum_{i=1}^n (y_i - \hat{\beta}_{d,0} + \hat{\beta}_{d,1}x_{i,1} + ... + \hat{\beta}_{d,d}x_{i,d})^2$

$= \sum_{i=1}^n (y_i - \hat{y}_{id+1})^2 - \sum_{i=1}^n (y_i - \hat{y}_{id})^2$

The addition of one covariate term brings one more degree of freedom when finding the minimum argument $\hat{\beta}_{d+1}$ for $||\mathbf{Y} - \mathbf{X}_{d+1}\beta_{d+1}||_2^2$. The minimum argument can be computed in closed form without issues as it is assumed $n > d + 1$, or number of features + 1 do not exceed number of samples within the training data, $\mathbf{X}_{d+1}$. This means that the estimated $\mathbf{Y}$, $\hat{\mathbf{Y}}_{d+1} = \mathbf{X}_{d+1}\beta_{d+1}$, will be closer to the true value of $\mathbf{Y}$ than if were using the minimum argument obtained from the model without the additional covariate term, $\hat{\mathbf{Y}}_d = \mathbf{X}_d\beta_d$.

$\Rightarrow (y_i - \hat{\beta}_{d+1,0} + \hat{\beta}_{d+1,1}x_{i,1} + ... + \hat{\beta}_{d+1,d+1}x_{i,d+1})^2 \leq (y_i - \hat{\beta}_{d,0} + \hat{\beta}_{d,1}x_{i,1} + ... + \hat{\beta}_{d,d}x_{i,d})^2$

$\Rightarrow \sum_{i=1}^n (y_i - \hat{\beta}_{d+1,0} + \hat{\beta}_{d+1,1}x_{i,1} + ... + \hat{\beta}_{d+1,d+1}x_{i,d+1})^2 - \sum_{i=1}^n (y_i - \hat{\beta}_{d,0} + \hat{\beta}_{d,1}x_{i,1} + ... + \hat{\beta}_{d,d}x_{i,d})^2 \leq 0$

$\Rightarrow ||\mathbf{Y} - \mathbf{X}_{d+1}\hat{\beta_{d+1}}||_2^2 - ||\mathbf{Y} - \mathbf{X}_d\hat{\beta}_d||_2^2 \leq 0$

$\Rightarrow RSS_{d+1} - RSS_d \leq 0$

$\Rightarrow RSS_{d+1} \leq RSS_d$

$\Rightarrow \frac{RSS_{d+1}}{TSS} \leq \frac{RSS_d}{TSS}$

$\Rightarrow 1 - \frac{RSS_{d+1}}{TSS} \geq 1 - \frac{RSS_d}{TSS}$

$\Rightarrow R^2_{d+1} \geq R^2_d$

Therefore, if $n > d + 1$, adding an additional covariate never lowers $R^2$ over training data.

## Question 2

### 2a

```
fit <- lm(price ~ zipcode +
            bedrooms +
            bathrooms +
            bedrooms * bathrooms +
            sqft_living +
            sqft_lot, train)
summary(fit)$r.squared # Training R2
```

```
## [1] 0.5223738
```

```
cor(predict(fit, test), test$price)^2 # Test R2
```

```
## [1] 0.5165772
```

### 2b

```
fit <- lm(price ~ zipcode +
            bedrooms +
            bathrooms +
            bedrooms * bathrooms +
            sqft_living +
            sqft_lot +
            sqft_living * bedrooms, train)
summary(fit)$r.squared # Training R2
```

```
## [1] 0.5262117
```

```
cor(predict(fit, test), test$price)^2 # Test R2
```

```
## [1] 0.5228651
```

Adding a new covariate that multiplies sqft_living and bedrooms. It models both the number of bedrooms and the size of each, which may influence the price.

### 2c

```
fit <- lm(price ~ zipcode +
            poly(bedrooms, 2) +
            poly(bathrooms, 3) +
            sqft_living +
```

```
             sqft_lot, train)
summary(fit)$r.squared # Training R2
```

```
## [1] 0.5423359
```

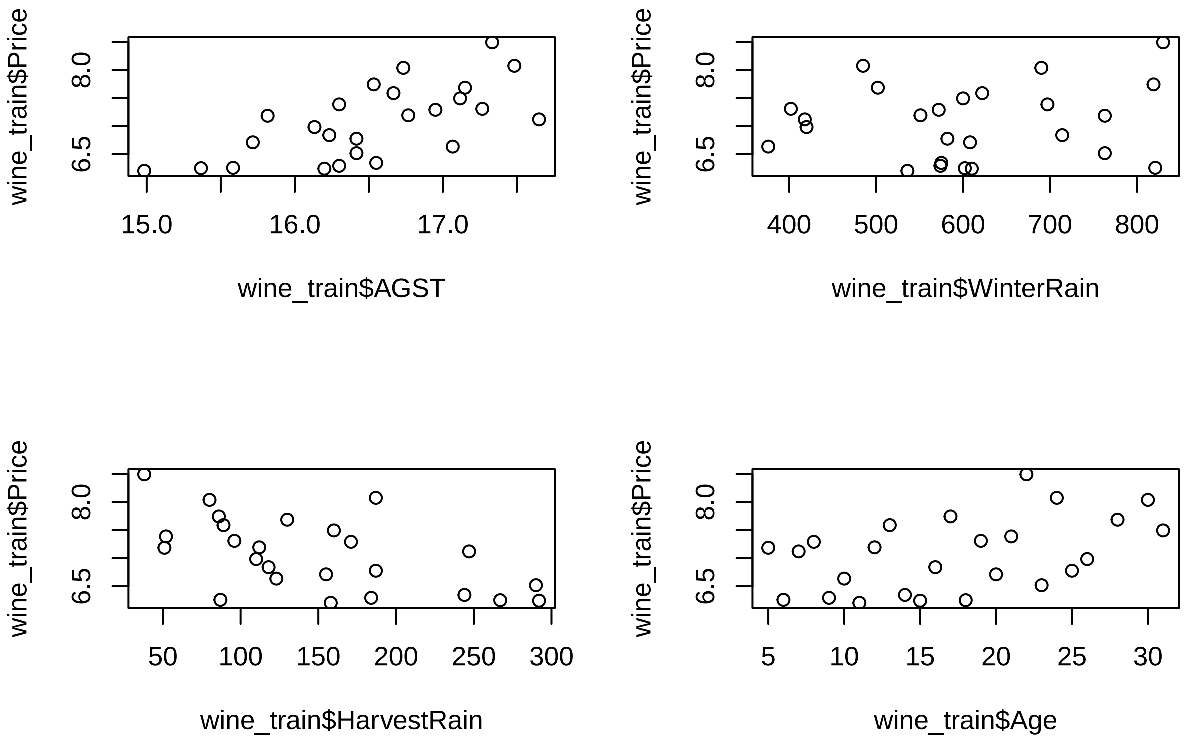
```
cor(predict(fit, test), test$price)^2 # Test R2
```

```
## [1] 0.5285267
```

## Question 3

### 3.a

```
wine_train <-
  read.csv("/Users/vladislavtrukhin/Downloads/_data_hw2/wine.csv")
wine_test <-
  read.csv("/Users/vladislavtrukhin/Downloads/_data_hw2/winetest.csv")
par(mfrow=c(2,2))
plot(wine_train$AGST, wine_train$Price)
plot(wine_train$WinterRain, wine_train$Price)
plot(wine_train$HarvestRain, wine_train$Price)
plot(wine_train$Age, wine_train$Price)
```




```
cor(wine_train$AGST, wine_train$Price)
```

```
## [1] 0.6595629
```

```
cor(wine_train$WinterRain, wine_train$Price)
```

```
## [1] 0.1366505
```

```
cor(wine_train$HarvestRain, wine_train$Price)
```

```
## [1] -0.5633219
```

```r
cor(wine_train$Age, wine_train$Price)
```

```
## [1] 0.4477679
```

According to the graph, AGST and Price seem to be the most correlated. The variance is smaller and has a strong positive correlation. The Pearson's correlation number suggests the same, with the magnitude of the value for AGST and Price higher than all three.

### 3.b

```r
fit <- lm(Price ~ AGST, wine_train)
summary(fit)$coeff # Coefficients
```

```
##               Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) -3.4177613  2.4935130 -1.370661 0.1837099385
## AGST         0.6350943  0.1509154  4.208282 0.0003350495
```

```r
summary(fit)$r.squared # Training R2
```

```
## [1] 0.4350232
```

```r
rss <- sum((predict(fit,wine_test)-wine_test$Price)^2)
tss <- sum((wine_test$Price-mean(wine_test$Price))^2)
rsq <- 1 - rss/tss
rsq # Test R2
```

```
## [1] 0.31426
```

### 3.c

```r
fit <- lm(Price ~ AGST + HarvestRain, wine_train)
summary(fit)$r.squared # Training R2
```

```
## [1] 0.7073708
```

```r
rss <- sum((predict(fit,wine_test)-wine_test$Price)^2)
tss <- sum((wine_test$Price-mean(wine_test$Price))^2)
rsq <- 1 - rss/tss
rsq # Test R2
```

```
## [1] -2.503339
```

```r
fit <- lm(Price ~ AGST + HarvestRain + Age, wine_train)
summary(fit)$r.squared # Training R2
```

```
## [1] 0.7900362
```

```r
rss <- sum((predict(fit,wine_test)-wine_test$Price)^2)
tss <- sum((wine_test$Price-mean(wine_test$Price))^2)
rsq <- 1 - rss/tss
rsq # Test R2
```

```
## [1] -0.5080824
```

```r
fit <- lm(Price ~ AGST + HarvestRain + Age + WinterRain, wine_train)
summary(fit)$r.squared # Training R2
```

```
## [1] 0.8285662
```

```r
summary(fit)$coeff # Coefficients
```

```
##                 Estimate   Std. Error   t value     Pr(>|t|)
## (Intercept) -3.429980187 1.7658975180 -1.942344 6.631093e-02
## AGST         0.607209348 0.0987022158  6.151932 5.197012e-06
## HarvestRain -0.003971534 0.0008537981 -4.651608 1.537556e-04
## Age          0.023930832 0.0080968750  2.955564 7.818874e-03
## WinterRain   0.001075505 0.0005072784  2.120148 4.669359e-02
```

```r
rss <- sum((predict(fit,wine_test)-wine_test$Price)^2)
tss <- sum((wine_test$Price-mean(wine_test$Price))^2)
rsq <- 1 - rss/tss
rsq # Test R2
```

```
## [1] 0.3343905
```

```r
fit <- lm(Price ~ AGST + HarvestRain + Age + WinterRain + FrancePop, wine_train)
summary(fit)$r.squared # Training R2
```

```
## [1] 0.8293592
```

```r
rss <- sum((predict(fit,wine_test)-wine_test$Price)^2)
tss <- sum((wine_test$Price-mean(wine_test$Price))^2)
rsq <- 1 - rss/tss
rsq # Test R2
```

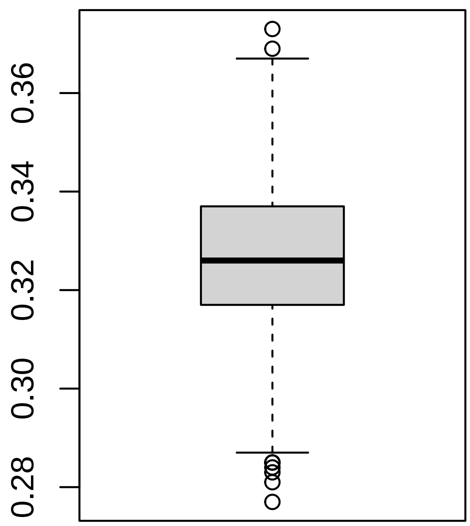
```
## [1] 0.2120672
```

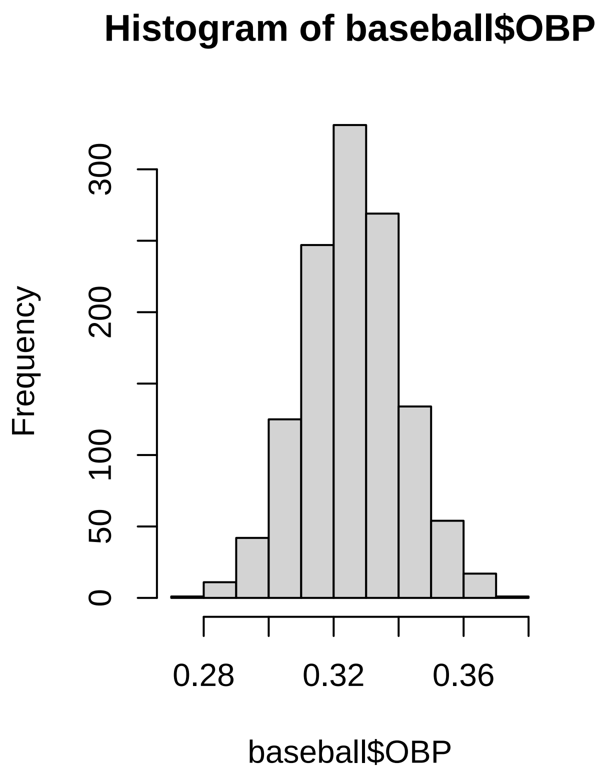
The linear model depending on AGST, HarvestRain, Age, and WinterRain performed the best, as it had a high $R^2$ value for the training data and the highest $R^2$ value for the test data. That particular model agrees with Prof. Ashenfelter's findings, since HarvestRain has a negative coefficient and WinterRain has a positive one.

## Question 4

### 4.a

```r
baseball = read.csv("/Users/vladislavtrukhin/Downloads/_data_hw2/baseball.csv")
par(mfrow=c(1, 2))
boxplot(baseball$OBP)
hist(baseball$OBP)
```



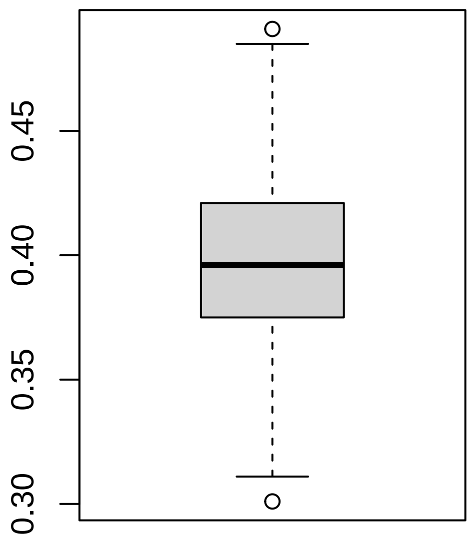

```
mean(baseball$OBP)
```

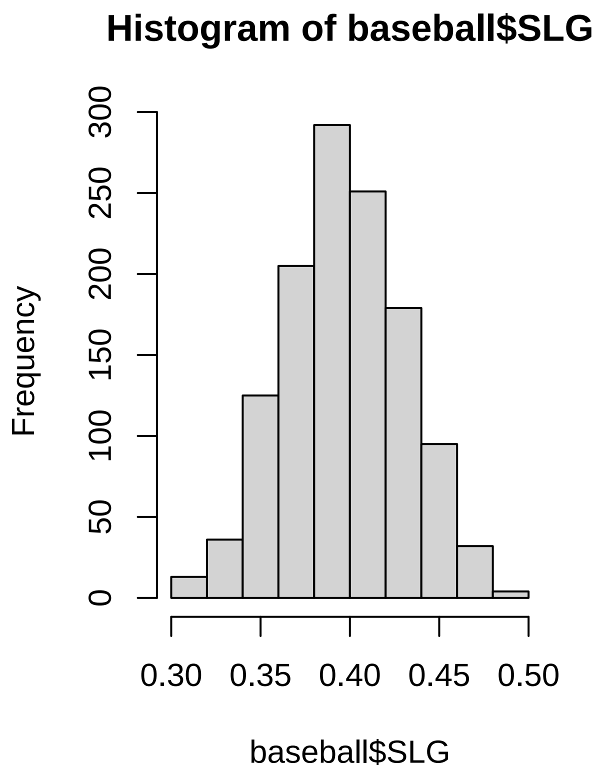
```
## [1] 0.3263312
```

```
median(baseball$OBP)
```

```
## [1] 0.326
```

```
boxplot(baseball$SLG)
hist(baseball$SLG)
```



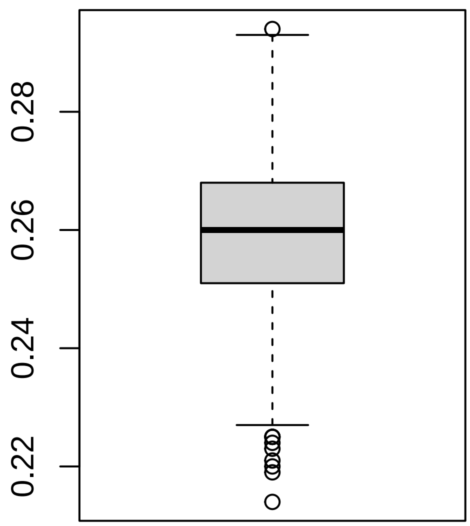

```
mean(baseball$SLG)
```

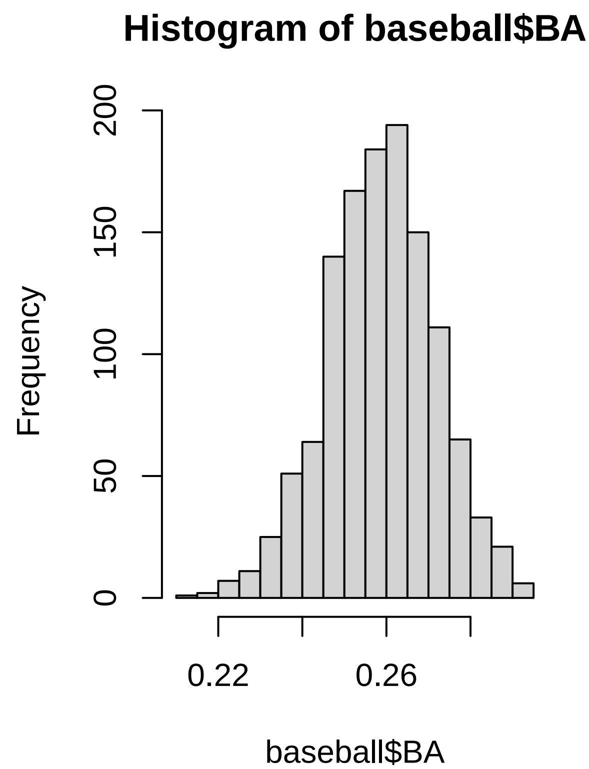
```
## [1] 0.3973417
```

```
median(baseball$SLG)
```

```
## [1] 0.396
```

```
boxplot(baseball$BA)
hist(baseball$BA)
```




```r
mean(baseball$BA)
```

```
## [1] 0.2592727
```

```r
median(baseball$BA)
```

```
## [1] 0.26
```

**4.b**

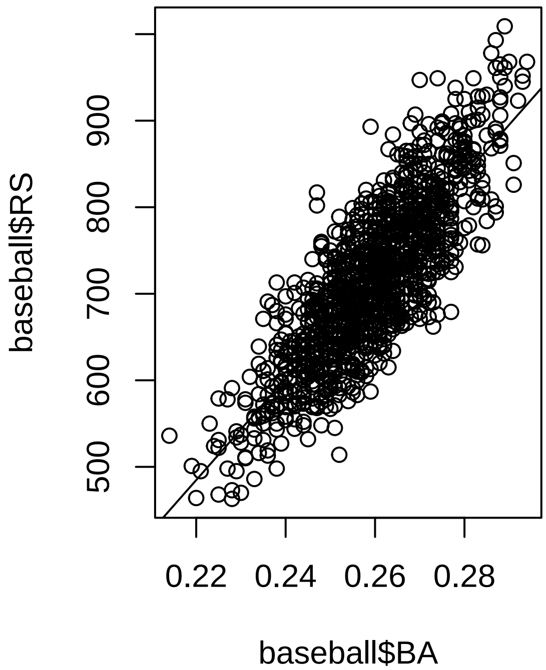
```r
par(mfrow=c(1,2))
fit <- lm(baseball$RS ~ baseball$BA)
summary(fit)$coeff # Coefficients
```

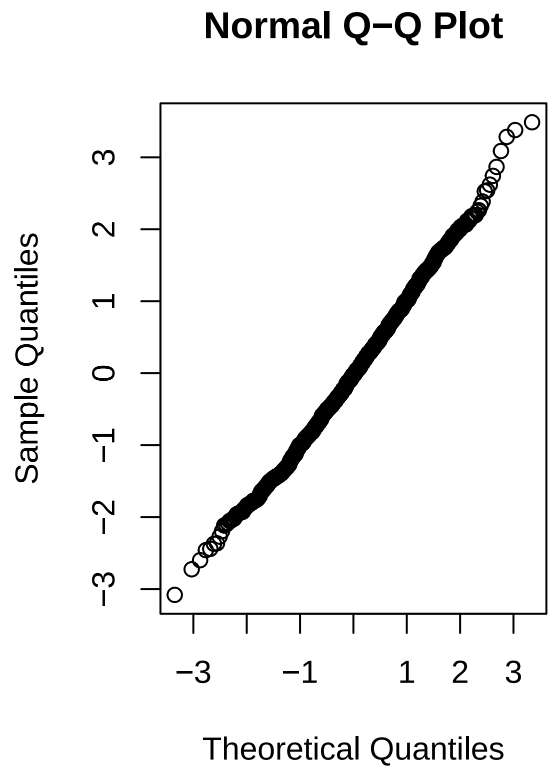
```
##              Estimate Std. Error   t value      Pr(>|t|)
## (Intercept) -805.511   29.51107 -27.29522 1.207747e-128
## baseball$BA 5864.840  113.68182  51.58995 6.416404e-310
```

```r
summary(fit)$r.squared # Training R2
```

```
## [1] 0.6839284
```

```r
plot(baseball$BA, baseball$RS)
abline(fit)
qqnorm(rstandard(fit))
```



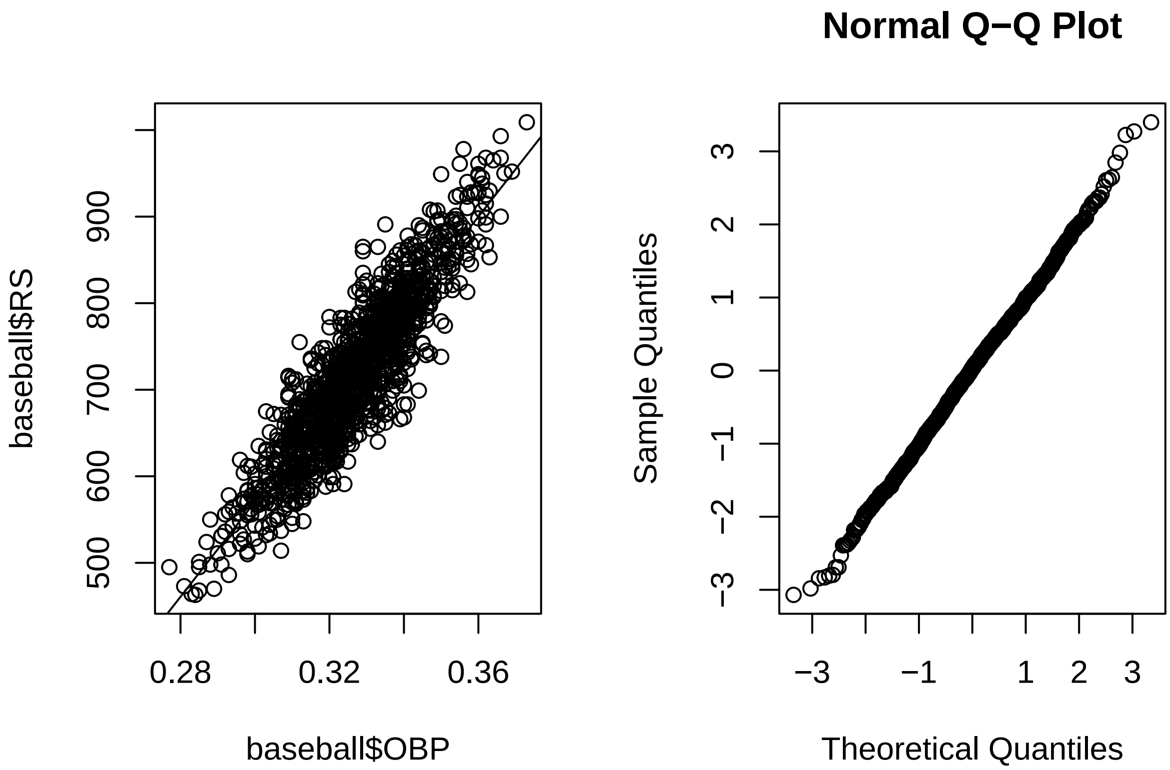

```
fit <- lm(baseball$RS ~ baseball$OBP)
summary(fit)$coeff # Coefficients
```

```
##                 Estimate Std. Error   t value      Pr(>|t|)
## (Intercept)  -1076.602   24.69729 -43.59192 9.443235e-252
## baseball$OBP  5490.386   75.60177  72.62246  0.000000e+00
```

```
summary(fit)$r.squared # Training R2
```

```
## [1] 0.8108862
```

```
plot(baseball$OBP, baseball$RS)
abline(fit)
qqnorm(rstandard(fit))
```



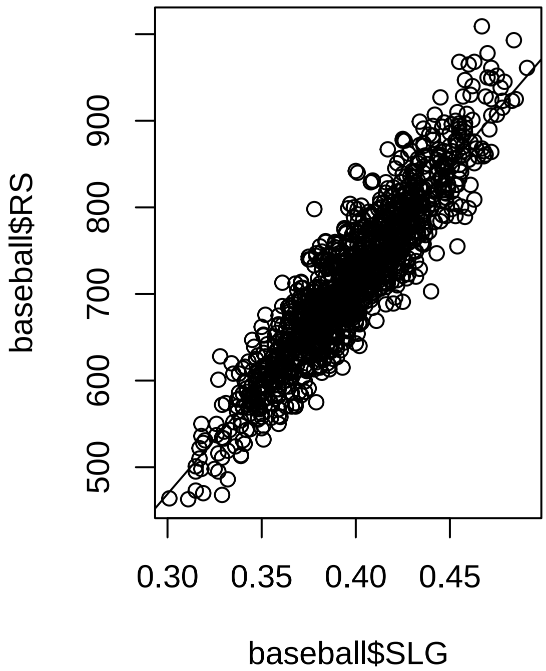

```
fit <- lm(baseball$RS ~ baseball$SLG)
summary(fit)$coeff # Coefficients
```

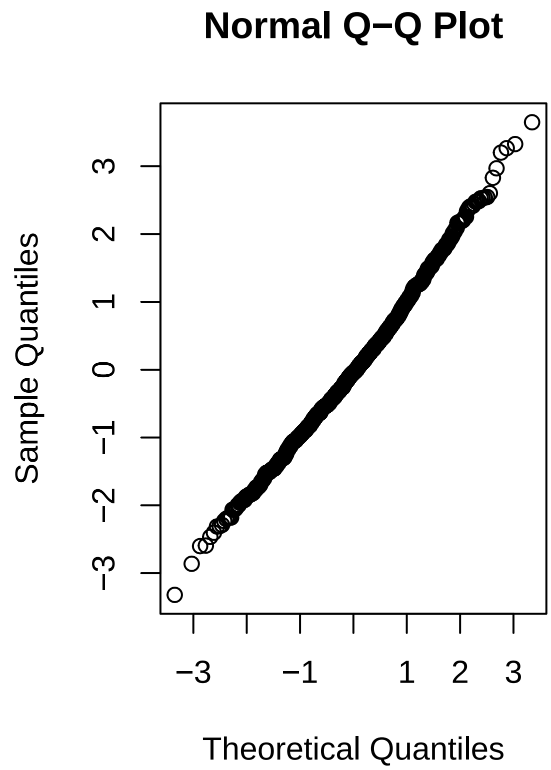
```
##                Estimate Std. Error   t value      Pr(>|t|)
## (Intercept)    -289.368   12.35223 -23.42638 1.190308e-100
## baseball$SLG   2527.925   30.97887  81.60158  0.000000e+00
```

```
summary(fit)$r.squared # Training R2
```

```
## [1] 0.8440831
```

```
plot(baseball$SLG, baseball$RS)
abline(fit)
qqnorm(rstandard(fit))
```




The analysis is not consistent with the intuition, as the $R^2$ is the lowest relative to RS's and BA's $R^2$.

### 4.c

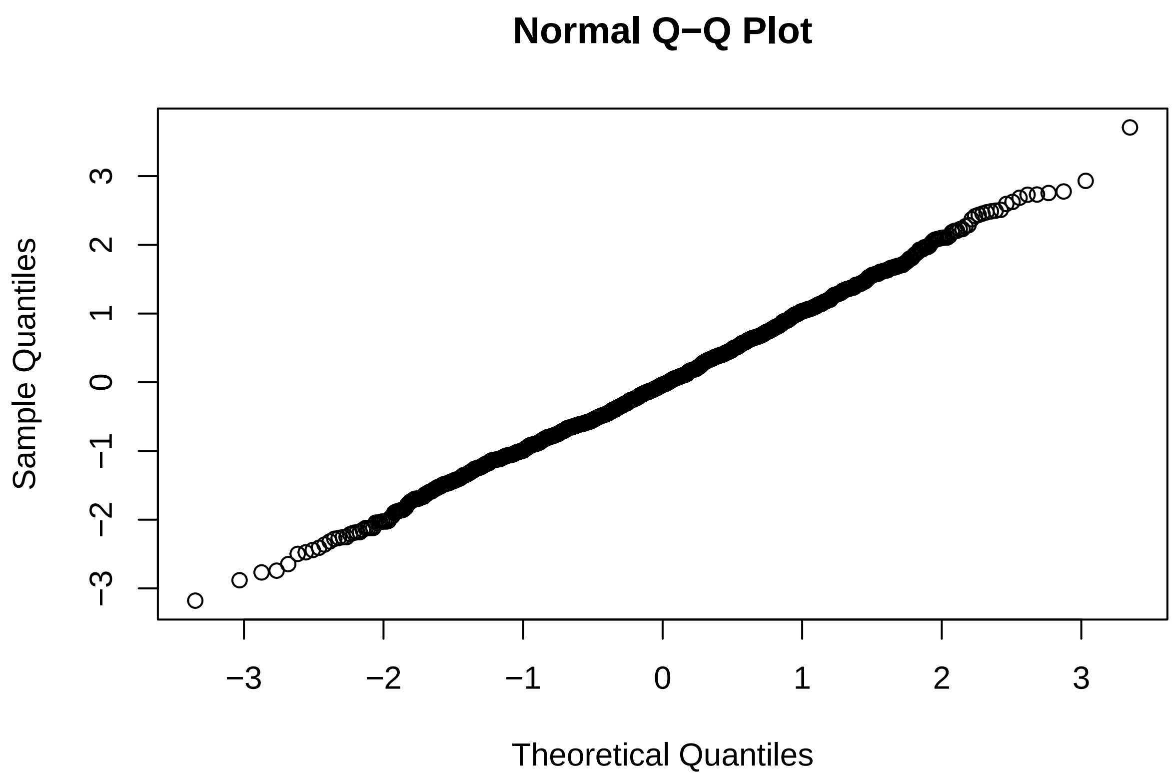
```
fit <- lm(baseball$RS ~ baseball$BA + baseball$SLG + baseball$OBP)
summary(fit)$coeff # Coefficients
```

```
##                 Estimate Std. Error    t value      Pr(>|t|)
## (Intercept)   -806.0845   17.39190 -46.348260 5.904672e-272
## baseball$BA   -134.9050  113.73431  -1.186141  2.357959e-01
## baseball$SLG 1533.8848   37.75868  40.623372 2.187242e-229
## baseball$OBP 2900.9403   97.87168  29.640243 4.386860e-146
```

```
summary(fit)$r.squared # Training R2
```

```
## [1] 0.9248834
```

```
qqnorm(rstandard(fit))
```




```
fit <- lm(baseball$RS ~ baseball$OBP + baseball$SLG)
summary(fit)$r.squared # Training R2
```

```
## [1] 0.9247974
```

The results are consistent of that in 4.b. The coefficient of BA has a low significance level, consistent with the low $R^2$ value obtained from 4.b. The two models have near equivalent $R^2$ values, which makes the later model a better model due to being more simple.

### 4.d

```
baseball$RD = baseball$RS - baseball$RA
baseball_pre_2002 = baseball[which(baseball$Year < 2002), ]

fit1 <- lm(W ~ RD, baseball_pre_2002)
fit2 <- lm(RS ~ OBP + SLG, baseball_pre_2002)
fit3 <- lm(RA ~ OOBP + OSLG, baseball_pre_2002)

oak_pred <- data.frame("OBP" = .349,
                       "SLG" = .430,
                       "OOBP" = .307,
                       "OSLG" = .373)
oak_pred$RS <- predict(fit2, oak_pred)
oak_pred$RA <- predict(fit3, oak_pred)
oak_pred$RD <- oak_pred$RS - oak_pred$RA
predict(fit1, oak_pred) #Predicted Wins
```

```
##        1
## 103.1386
```

```r
baseball[which(baseball$Year == 2002 & baseball$Team == 'OAK'), ]$W #Actual Wins
```

```
## [1] 103
```