STAT 423 Homework 2

1. In order to test hypotheses, we make the assumption that $y_i = \beta_0 + \beta_1 x_i + \epsilon_i$, and $\epsilon \overset{iid}{\sim} N(0, \sigma^2)$ for all $i \in \{1, \ldots, n\}$. Given a value of $0 < \alpha < 1$, this allows us to say that with a $1 - \alpha$ probability, the confidence interval, for example, $\hat{\beta}_1 \pm t_{n-(p+1)} SE_{\hat{\beta}_1}$ contains the true value of β_1. In this question, we assess how the corresponding confidence intervals and hypothesis tests perform subject to model miss-specification.

Consider the dataset $\{(x_i, y_i)\}_{i=1}^n$ drawn according to $x_i \sim \mathcal{N}(0, 1)$, $\epsilon_i \sim \mathcal{N}(0, 1)$ and $y_i = 0.5 + x_i + \epsilon_i$, for $i \in \{1, \ldots, n\}$, $n = 100$. Use the following R code to generate your data:

```
> set.seed(123)
> n <- 100
> x <- rnorm(n,0,1)
> eps <- rnorm(n,0,1)
> y <- 0.5 + x + eps
```

(a) Construct a 99% confidence interval for β_0. Also construct a 99% confidence interval for β_1. Make sure to explain what the relevant probability distribution is and what the relevant quantile (or critical value) is.

As was shown in the lecture 4 slides, the interval $\hat{\beta}_i - t_{1-\alpha/2,n-p-1} \cdot SE(\hat{\beta}_i) \leq \beta_i \leq \hat{\beta}_i + t_{1-\alpha/2,n-p-1} \cdot SE(\hat{\beta}_i)$ is a $100(1-\alpha)\%$ confidence interval for β_i, where $t_{1-\alpha/2,n-p-1}$ is the $1 - \alpha/2$ quantile of the t distribution with $n - p - 1$ degrees of freedom, where n is the number of observations per predictor, and p is the number of predictors. Also, $SE(\hat{\beta}_i)$ is the standard error of $\hat{\beta}_i$ and is calculated as $\sqrt{\frac{RSS}{n-p-1}((X'X)^{-1})_{i+1,i+1}}$, where RSS is the residual sum of squares, and $((X'X)^{-1})_{i+1,i+1}$ is the $(i+1, i+1)$ entry of the covariance matrix.

In our case, with $n = 100$ observations for our $p = 1$ predictors, and a significance level of $\alpha = 0.01$, we are finding the 0.995 quantile of a t distribution with $100 - 1 - 1 = 98$ degrees of freedom. In particular, this value is $t_{0.995,98}$.

Specifically, we have that the critical value from the relevant t distribution is $t_{0.995,98} = 2.626931$. Furthermore, our 99% confidence interval for β_0 is [0.1459217, 0.6484721], and our 99% confidence interval for β_1 is [0.6722273, 1.2228295].

(b) Suppose that instead of a 99% confidence interval, the task above was to construct a 95% interval for β_1. Would a 95% confidence interval for β_1 be wider than a 99% confidence interval? Why, or why not?

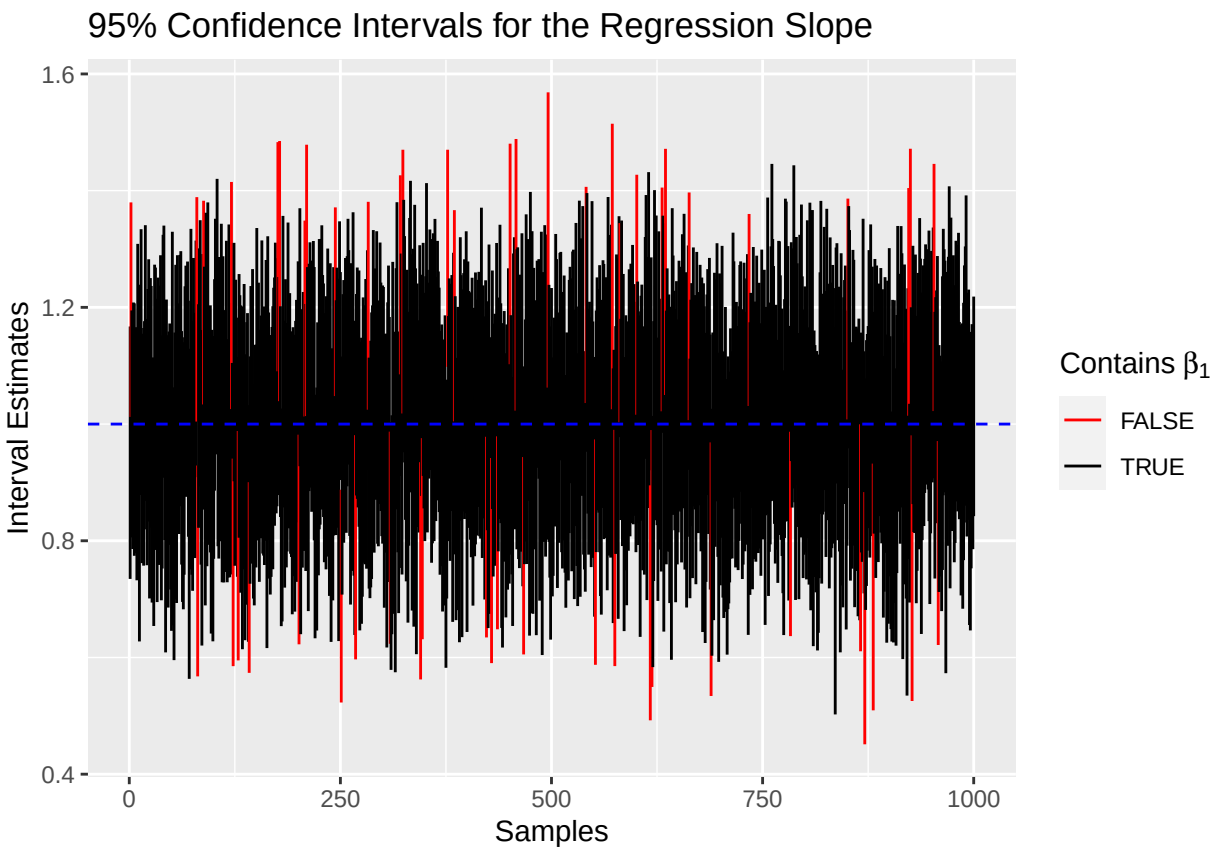
If instead of a 99% confidence interval, the task above was to construct a 95% interval for β_1, a 95% confidence interval for β_1 would not be wider than a 99% confidence interval for β_1. This is due to the fact that the confidence interval formula for β_1 is $\hat{\beta}_1 - t_{1-\alpha/2,n-p-1} \cdot SE(\hat{\beta}_1) \leq \beta_1 \leq \hat{\beta}_1 + t_{1-\alpha/2,n-p-1} \cdot SE(\hat{\beta}_1)$. As can be seen from the previous interval, the only thing that changes is the quantile of the t distribution that we look at. In the case of a 99% confidence interval we look at the 0.995th quantile, and for a 95% confidence interval we look at the 0.975th quantile. Therefore, the quantile for the latter will be smaller than the former, thus leading to a narrower interval. Furthermore, from a more intuitive lens, since we are less confident that a 95% interval will contain the true value of β_1 as compared to a 99% interval, we would expect the interval with less confidence to be narrower.

(c) Now suppose that instead of setting n to 100 above, we had chosen $n = 1000$, ran the same code and again constructed a 99% confidence interval for β_1. Do you expect the confidence interval to be wider than what you obtained in (a)? Why, or why not?

If instead of setting n to 100 above, we had chosen $n = 1000$, ran the same code and again constructed a 99% confidence interval for β_1, I would expect the confidence interval to be narrower than what we obtained in (a). The reason for this is twofold, with both reasons relying on the fact that the degrees of freedom increases when n increases. First off, and most importantly, the confidence interval gets narrower due to a decrease in the standard error (i.e. our estimate of β_1 gets more precise). To see this we must look at the formula for the standard error, which for $\hat{\beta}_1$ is $\sqrt{\frac{RSS}{n-p-1}((X'X)^{-1})_{1+1,1+1}}$. As can be seen in this formula, increasing n will decrease the value of $SE(\hat{\beta}_1)$, and therefore make the interval narrower around $\hat{\beta}_1$. The second reason why the interval gets narrower is that when we increase n, the degrees of freedom also increases, and thus the t distribution gets narrower. Therefore the critical value corresponding to the 0.995th quantile will be smaller for a larger n (holding p constant), and hence a narrower interval around $\hat{\beta}_1$.

(d) Generate $m = 1000$ datasets as above and construct a 95% confidence interval for β_1 in each of these datasets. What percentage of the datasets actually contains the true value $\beta_1 = 1$? Meaning, what is the actual coverage probability of the confidence interval you constructed? Plot the confidence intervals as follows: on the x-axis you want to have the replication (e.g. $i = 1, \ldots, 1000$) and on the y-axis the lower and upper bounds. You also want to plot a horizontal dashed line indicating the true population parameter. Using a different color, highlight the confidence intervals that do not contain the true population parameter.

In this problem, we will generate $m = 1000$ data sets as instructed above and construct a 95% confidence interval for β_1 in each of these data sets. Furthermore, we will plot the confidence intervals below.



As can be seen from the above plot, 946 out of the 1000 confidence intervals contained the true value of $\beta_1 = 1$. Hence the coverage probability of the confidence intervals constructed above is 0.946.

(e) Suppose that you now generate data as follows: $x_i \sim \mathcal{N}(0,1)$, $\epsilon_i \sim (1-\lambda)\mathcal{N}(0,1) + \lambda \text{ Unif}(-1,1)$, $0 \leq \lambda \leq 1$, and $y_i = 0.5 + x_i + \epsilon_i$, for $i \in \{1, \ldots, n\}$, $n = 100$. For example, if $\lambda = 0.1$ you can use the following R code chunk to generate your data:

```
> n <- 100
> lambda <- 0.1
> x <- rnorm(n,0,1)
> eps <- (1-lambda)*rnorm(n,0,1) + lambda*runif(n,-1,1)
> y <- 0.5 + x + eps
```

The data generating mechanism now no longer satisfies the normality assumptions that we generally make on ϵ

(i) Generate $m = 1000$ datasets as above and construct a 95% confidence interval for β_1 in each of these datasets. What percentage of the datasets actually contain the true value $\beta_1 = 1$? Meaning, what is the actual coverage probability of the confidence interval you constructed?

In this problem, we will generate $m = 1000$ datasets as instructed above and construct a 95% confidence interval for β_1 in each of these datasets. Furthermore, we will find the actual coverage probability of the confidence intervals we constructed. This is done below.

Based on the given datasets, with $n = 100$ and $\lambda = 0.01$, the actual coverage probability of the confidence intervals we constructed was 0.94, which is slightly lower than the nominal coverage probability of 0.95. It seems as if there is not much of an impact on the coverage probability when λ is low and n is relatively high, however, this coverage probability is lower than what we'd expect.

(ii) Consider modifying the sample size n, for example, suppose that $n = 3, 5, 10, 50$. What is the coverage probability for each individual n above.

In this problem, keeping λ the same as above for $n = 3, 5, 10, 50$, we will generate $m = 1000$ datasets as instructed above and construct a 95% confidence interval for β_1 in each of these datasets. Furthermore, we will find the actual coverage probability of the confidence intervals we constructed. This is done below.

Based on the given datasets, with $\lambda = 0.01$, the actual coverage probability of the confidence intervals we constructed was 0.72 when $n = 3$, 0.843 when $n = 5$, 0.918 when $n = 10$, and 0.945 when $n = 50$. It seems as if for small values of n the coverage probability is drastically off, but for moderate to large values of n, the coverage probability converges to the true coverage probability that was intended.

(iii) Consider increasing λ, for example, suppose that $\lambda = 0.1, 0.2, \ldots, 0.9, 1.0$. Now redo the above process for each value of λ. What is the coverage probability for each individual value of λ above? For which values do you see the coverage probability fall below 80%?

In this problem for $\lambda = 0.1, 0.2, \ldots, 0.9, 1.0$ and $n = 3, 5, 10, 50$, we will generate $m = 1000$ datasets as instructed above and construct a 95% confidence interval for β_1 in each of these datasets. Furthermore, we will find the actual coverage probability of the confidence intervals we constructed. This is done below.

```
##      0.1   0.2   0.3   0.4   0.5   0.6   0.7   0.8   0.9   1.0
## 3  0.721 0.690 0.694 0.699 0.697 0.697 0.708 0.691 0.710 0.673
## 5  0.834 0.865 0.854 0.855 0.850 0.836 0.860 0.874 0.863 0.838
## 10 0.896 0.922 0.918 0.897 0.914 0.928 0.906 0.894 0.918 0.906
## 50 0.938 0.947 0.946 0.946 0.942 0.954 0.946 0.940 0.961 0.928
```

As can be seen from the above matrix of the coverage probability for each n and λ pair, the coverage probability for the entire $n = 3$ row is below 80% (furthermore, the whole row below 73%). It turns out that besides the entire $n = 3$ row, all of the other n and λ pairings have a coverage probability above 80%. However, as n increases, the coverage probability converges to the desired value of 95%. Also, when $\lambda = 1$, we see that the coverage probability is at its lowers for each value of n. Intuitively this makes sense, because when $\lambda = 1$, we are simply sampling the errors from a Unif$(0, 1)$.

(iv) Consider now modifying α so that you build 80%, 85%, 90%, 95% and 99% confidence intervals. Choose $n = 100$ and let λ vary. Comment on the coverage probability as compared with the nominal levels 80%, 85%, 90%, 95% and 99%.

In this problem for $n = 100$, $\lambda = 0.1, 0.2, \ldots, 0.9, 1.0$, and confidence levels $100(1 - \alpha)\% = 80\%, 85\%, 90\%, 95\%, 99\%$, we will generate $m = 1000$ datasets as instructed above and construct confidence intervals for β_1 in each of these datasets. Furthermore, we will find the actual coverage probability of the confidence intervals we constructed. This is done below.

```
##        0.1   0.2   0.3   0.4   0.5   0.6   0.7   0.8   0.9   1.0
## 0.80 0.811 0.791 0.774 0.796 0.797 0.803 0.786 0.810 0.785 0.793
## 0.85 0.860 0.859 0.852 0.847 0.850 0.850 0.868 0.825 0.864 0.841
## 0.90 0.893 0.881 0.914 0.912 0.894 0.897 0.892 0.893 0.908 0.909
## 0.95 0.955 0.944 0.943 0.954 0.944 0.957 0.952 0.952 0.939 0.953
## 0.99 0.991 0.988 0.992 0.992 0.990 0.988 0.994 0.987 0.993 0.992
```

As can be seen from the above matrix of the coverage probability for each λ and $100(1 - \alpha)\%$ pair there doesn't seem to be much deviation from the nominal coverage probabilities. This seems to be caused by the large sample size $n = 100$. With smaller values of n we would probably see differences from the nominal coverage probabilities.

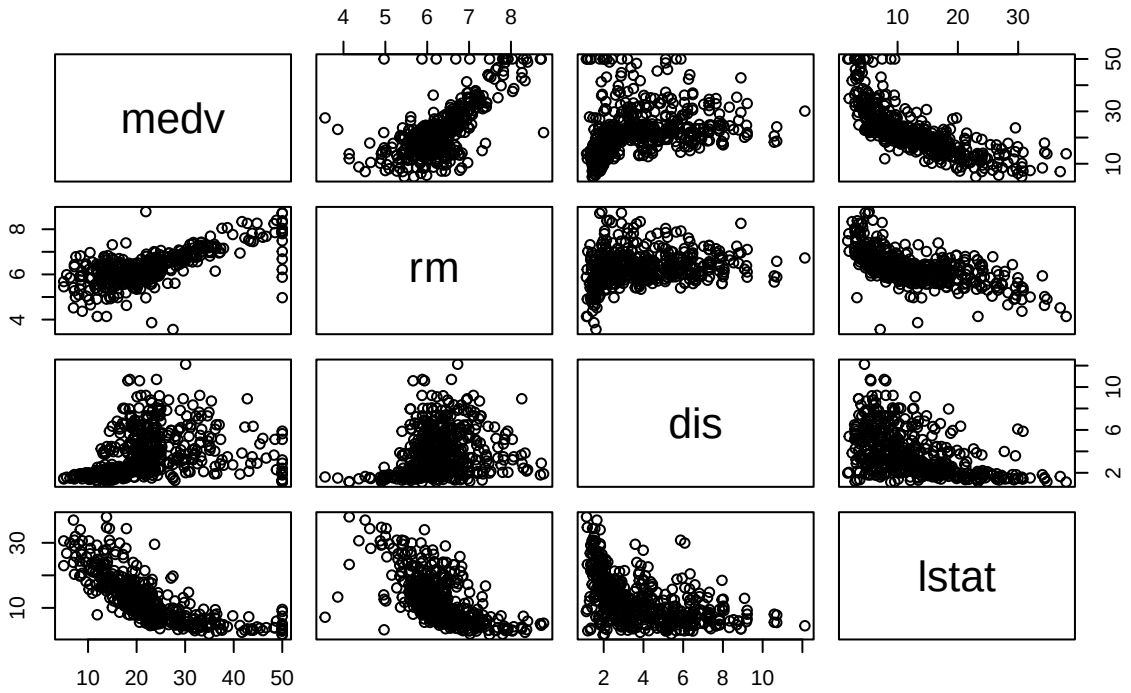
2. Consider the dataset `BostonHousing` from R package `mlbench`. Consider a multiple linear regression with response `medv` and predictors `rm`, `dis`, `lstat`, `age`, `crim`, `indus` and `nox`.

As computed in R, the estimated parameters in the linear regression model are $\hat{\beta}_0 = 2.8082926$, $\hat{\beta}_1 = 4.8733856$, $\hat{\beta}_2 = -0.4612815$, and $\hat{\beta}_3 = -0.7233308$.

(a) Examine the pairwise scatterplots for the following three predictors `rm`, `dis`, `lstats` and the response. What should the correlation matrix look like for these 4 variables (i.e. which correlations are large and positive, which are large and negative, and which are small)? Compute and print the correlation matrix to verify your results.

Below we will display and examine the scatterplots of the following three predictors `rm`, `dis`, `lstats` and the response, as well as describing and printing out what the correlation matrix should look like.

Pairwise Scatterplots Between the Response and Predictors



As can be seen from the above pairwise scatterplots of the following three predictors `rm`, `dis`, `lstat` and the response, it appears as if `medv` has a strong positive linear relationship with `rm`, a weak positive linear relationship with `dis`, and a strong negative linear relationship with `lstat`. Thus we should see a high postie correlation between `medv` and `rm`, a small positive correlation between `medv` and `dis`, and a high negative correlation between `medv` and `lstat`. Below we will compute and display the correlation matrix to validate these results.

```
##             medv         rm        dis      lstat
## medv   1.0000000  0.6953599  0.2499287 -0.7376627
## rm     0.6953599  1.0000000  0.2052462 -0.6138083
## dis    0.2499287  0.2052462  1.0000000 -0.4969958
## lstat -0.7376627 -0.6138083 -0.4969958  1.0000000
```

As can be seen from the above correlation matrix, `medv` has a strong positive linear relationship with `rm`, a weak positive linear relationship with `dis`, and a strong negative linear relationship with `lstat`. This is exactly what we expected based on the information provided from the scatterplot matrix.

(b) Fit the multiple linear regression with response `medv` and predictors `rm`, `dis` and `lstat`. Test whether **all** of the coefficients next to **all** predictors are zero at the **0.01**-level. What is the null hypothesis? What is the alternative hypothesis? What is the test statistic you are using and what is the distribution of the test statistic under H_0? And what is the decision of the test?

To start off this problem, we will use the `lm()` function in R to fit a multiple linear regression model with response `medv` and predictors `rm`, `dis` and `lstat`. This is done below.

We will now test whether all of the coefficients next to all of the predictors are zero at the 0.01 significance level. This test corresponds to the null hypothesis $H_0 : \beta_{rm} = \beta_{dis} = \beta_{lstat} = 0$, and alternative hypothesis H_1 : at least one $\beta_i \neq 0$. These competing hypotheses can be tested by comparing the empty model and the full model with the predictors `rm`, `dis` and `lstat`.

This test is called the ANOVA test and requires the use of the F statistic which is defined as $F = \frac{(RSS_q - RSS_p)(n-p-1)}{(p-q)RSS_p}$, where n is the number of observations, p is the number of covariates in the full model, q is the number of covariates in shared by both models, RSS_q is the residual sum of squares of the smaller model, and RSS_p is the residual sum of squares of the full model. Under the assumption of the null hypothesis (as well as our usual regression assumptions), $F \sim F_{p-q,n-p-1}$, where $p-q$ is the degrees of freedom of the numerator of the F ratio, and $n-p-1$ is the degrees of freedom of the denominator of the F ratio.

In our case, $F = \frac{(RSS_0 - RSS_3)(506-3-1)}{(3-0)RSS_3} = \frac{502(RSS_0 - RSS_3)}{3RSS_3}$, which follows the F distribution with 3 numerator degrees of freedom, and 502 denominator degrees of freedom. The corresponding F test is a one-sided upper-tailed test, which we will use the `anova` function in R to perform.

As calculated in the `anova()` function in R, our p-values was $5.2178766 \times 10^{-113}$, hence at the significance level of 0.01, we reject the null hypothesis and have found very significant evidence that at least one of the parameters in the model differs from zero. Thus we prefer the bigger model.

(c) Print the output of the test ran by using the `anova` function to compare the empty model and the four covariates. Show how to compute the F statistic using `Res.DF` and `RSS` as printed by the `anova` function. Is the F statistic computed here the same as the one obtained by running `summary(lm(formula = medv ~ rm + dis + lstat, data = bh.data))`?

To start off this problem, we will print out the output of the test ran by using the `anova()` function to compare the empty model and the full model. That is, we will print out the results from the test we ran in part (b). This is done below.

```
## Analysis of Variance Table
##
## Model 1: medv ~ 1
## Model 2: medv ~ rm + dis + lstat
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    505 42716
## 2    502 15088  3     27628 306.41 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We will now show how to compute the F statistic using `Res.DF` and `RSS` as printed by the `anova()` function. As described in part (b), the F statistic for this given test is $F = \frac{(RSS_0 - RSS_3)(506-3-1)}{(3-0)RSS_3} = \frac{502(RSS_0 - RSS_3)}{3RSS_3}$. Based on the output, RSS_0 corresponds to the `RSS` value in the first row of the output, and RSS_3 corresponds

to the RSS value in the second row of the output. Furthermore, the 502 in the numerator corresponds to the Res.df value in the second row of the output, and the 3 in the denominator corresponds to the difference of the Res.df values in the first and second row of the output (or alternatively the Df value in the second row of the output). Putting this together, we can calculate the F statistic in R as

```
# Construct the F statistic given the values Res.df and RSS
num_df <- anova_fit[1,1] - anova_fit[2,1]
denom_df <- anova_fit[2,1]
RSS_0 <- anova_fit[1,2]
RSS_3 <- anova_fit[2,2]
F_stat <- (denom_df * (RSS_0 - RSS_3)) / (num_df * RSS_3)
```

Running this calculation, we obtain $F = \frac{(42716-15088)(506-3-1)}{(3-0)15088} = \frac{502 \cdot (42716-15088)}{3 \cdot 15088} = 306.4081$, which is what is shown for the F value in the second row of the output. We will now see if this F value the same as the one obtained by running summary(lm(formula = medv ~ rm + dis + lstat, data = bh.data)). This is done below.

```
##
## Call:
## lm(formula = medv ~ rm + dis + lstat, data = BostonHousing)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.992  -3.133  -0.871   1.910  25.944
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.80829    3.36800   0.834 0.404781
## rm           4.87339    0.44456  10.962  < 2e-16 ***
## dis         -0.46128    0.13495  -3.418 0.000682 ***
## lstat       -0.72333    0.04933 -14.662  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.482 on 502 degrees of freedom
## Multiple R-squared:  0.6468, Adjusted R-squared:  0.6447
## F-statistic: 306.4 on 3 and 502 DF,  p-value: < 2.2e-16
```

As can be seen by running summary(lm(formulae = medv ~ rm + dis + lstat, data = bh.data)), we obtain an F statistic of 306.4 on 3 and 502 degrees of freedom with a p-value $< 2.2 \times 10^{-16}$, this matches what we obtained by hand and through anova().

(d) Consider null hypotheses: $H_0^1 : \beta_{rm} = 0$, $H_0^2 : \beta_{dis} = 0$, $H_0^3 : \beta_{lstat} = 0$, $H_0^4 : \beta_{age} = 0$, $H_0^5 : \beta_{crim} = 0$, $H_0^6 : \beta_{indus} = 0$, and $H_0^7 : \beta_{nox} = 0$. Which of these null hypotheses would be rejected if:

i. You test each hypothesis at the 0.01 level, without any FWER or FDR control?

In this part of the problem, we will consider the above set of null hypotheses and find out which of these null hypotheses would be rejected if we test each hypothesis at the 0.01 level, without any FWER or FDR control. To do this we will run seven different hypothesis tests, one for each null hypothesis. To save time we will use the lm() and summary() function to compute the p-values. This is done below.

```
##
## Call:
## lm(formula = medv ~ rm + dis + lstat + age + crim + indus + nox,
##     data = BostonHousing)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.5408  -3.2994  -0.9892   1.9849  27.7119
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.93238    3.95586   3.016 0.002689 **
## rm           4.81640    0.44161  10.907  < 2e-16 ***
## dis         -1.24214    0.20168  -6.159 1.51e-09 ***
## lstat       -0.57280    0.05544 -10.332  < 2e-16 ***
## age         -0.01323    0.01444  -0.917 0.359847
## crim        -0.11168    0.03175  -3.517 0.000476 ***
## indus       -0.16806    0.05885  -2.856 0.004475 **
## nox         -8.15678    3.85855  -2.114 0.035016 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.294 on 498 degrees of freedom
## Multiple R-squared:  0.6732, Adjusted R-squared:  0.6686
## F-statistic: 146.6 on 7 and 498 DF,  p-value: < 2.2e-16
```

As can be seen from the above summary of the multiple regression model, we would reject the following hypotheses: $H_0^1 : \beta_{rm} = 0$, $H_0^2 : \beta_{dis} = 0$, $H_0^3 : \beta_{lstat} = 0$, $H_0^5 : \beta_{crim} = 0$, and $H_0^6 : \beta_{indus} = 0$. That means we would fail to reject $H_0^4 : \beta_{age} = 0$ and $H_0^7 : \beta_{nox} = 0$.

ii. You control the FWER at the 0.01 level using the Bonferroni correction?

In this part of the problem, we will consider the above set of null hypotheses and find out which of these null hypotheses would be rejected if we test each hypothesis at the 0.01 level, given that we control the FWER at the 0.01 level using the Bonferroni correction.

In the Bonferroni correction procedure, given that m different hypothesis tests are being ran, in order to ensure that $FWER \leq \alpha$, one must perform each of m hypothesis tests at level α/m. In this case, for test i, one would reject H_0^i if the p-value for this test p_i is less than α/m. Alternatively, instead of rejecting hypothesis tests with p-values $p_i < \alpha/m$, one could adjust the calculated p-values p_i to obtain adjusted p-values p_i^*, where $p_i^* = p_i \cdot m$. Then one would reject only the null hypothesis corresponding to hypothesis tests where $p_i^* < \alpha$.

In our case $m = 7$, and we will use the second approach using the `p.adjust()` function in R. This procedure is done below.

```
##           rm          dis        lstat          age         crim        indus
## 3.885949e-24 1.057801e-08 5.859248e-22 1.000000e+00 3.329402e-03 3.132207e-02
##          nox
## 2.451128e-01
```

As can be seen from the Bonferroni correction procedure, we would reject the following hypotheses: $H_0^1 : \beta_{rm} = 0$, $H_0^2 : \beta_{dis} = 0$, $H_0^3 : \beta_{lstat} = 0$, and $H_0^5 : \beta_{crim} = 0$. That means we would fail to reject $H_0^4 : \beta_{age} = 0$, $H_0^6 : \beta_{indus} = 0$, and $H_0^7 : \beta_{nox} = 0$. It is important to note that the adjusted p-value for `age` was above 1 so it was automatically taken to be 1 since probabilities can't be greater than 1.

iii. You control the FWER at the 0.01 level using the Holm correction?

In this part of the problem, we will consider the above set of null hypotheses and find out which of these null hypotheses would be rejected if we test each hypothesis at the 0.01 level, given that we control the FWER at the 0.01 level using the Holm correction.

In the Holm correction procedure, given that m different hypothesis tests are being ran, in order to ensure that $FWER \leq \alpha$, one must first order the p-values from the m hypothesis tests from smallest to largest $p_{(1)}, \ldots, p_{(m)}$. Then, for $i \in \{1, \ldots, m\}$, one must calculate $\alpha/(m - i + 1)$. Next, one must let i_0 denote the smallest index such that $p_{(i_0)} \geq \frac{\alpha}{m-i_0+1}$. Once this is done one would reject only the null hypotheses corresponding to p-values $p_{(1)}, \ldots, p_{(i_0-1)}$. If $i_0 = 1$, no hypotheses are rejected, and if $p_{(i)} < \frac{\alpha}{m-i_0+1}$ for all i all the hypotheses are rejected.

Alternatively, one could use an adjusted p-value Holm method. In this case, one must first order the p-values from the m hypothesis tests from smallest to largest $p_{(1)}, \ldots, p_{(m)}$. Then, one must obtain the adjusted p-values $p^*_{(i)}$ as $p^*_{(i)} = p_{(i)} \cdot (m-i+1)$. Next, one must let i_0 denote the smallest index such that $p^*_{(i)} \geq \alpha$. Once this is done one would reject only the null hypotheses corresponding to the adjusted p-values $p^*_{(1)}, \ldots, p^*_{(i_0-1)}$.

We will now perform this procedure with our 7 p-values obtained in part (a). These p-values are

```
##           rm          dis        lstat          age         crim        indus
## 5.551356e-25 1.511144e-09 8.370354e-23 3.598472e-01 4.756288e-04 4.474581e-03
##          nox
## 3.501612e-02
```

We will now start the Holm procedure by ordering the p-values, this is done below.

```
##           rm        lstat          dis         crim        indus          nox
## 5.551356e-25 8.370354e-23 1.511144e-09 4.756288e-04 4.474581e-03 3.501612e-02
##          age
## 3.598472e-01
```

We will now use the `p.adjust()` function in R to adjust these p-values by $(m - i + 1)$. This is done below.

```
##           rm        lstat          dis         crim        indus          nox
## 3.885949e-24 5.022213e-22 7.555722e-09 1.902515e-03 1.342374e-02 7.003223e-02
##          age
## 3.598472e-01
```

As can be seen from the above output $i_0 = 5$. Thus, we would reject the following hypotheses: $H_0^1 : \beta_{rm} = 0$, $H_0^2 : \beta_{dis} = 0$, $H_0^3 : \beta_{lstat} = 0$, and $H_0^5 : \beta_{crim} = 0$. That means we would fail to reject $H_0^4 : \beta_{age} = 0$, $H_0^6 : \beta_{indus} = 0$, and $H_0^7 : \beta_{nox} = 0$.

iv. You control the FDR at the 0.01 level using the Benjamini-Hochberg procedure?

In this part of the problem, we will consider the above set of null hypotheses and find out which of these null hypotheses would be rejected if we test each hypothesis at the 0.01 level, given that we control the FDR at the 0.01 level using the Benjamini-Hochberg procedure.

In the Benjamini-Hochberg procedure, given that m different hypothesis tests are being ran, in order to control the FDR at level q, one must first order the p-values from the m hypothesis tests from smallest to largest $p_{(1)}, \ldots, p_{(m)}$. Then, for $i \in \{1, \ldots, m\}$, one must calculate $\frac{i}{m}q$. Next, one must let i_0 denote the largest index such that $p_{(i_0)} < \frac{i_0}{m}q$. Once this is done one would reject only the null hypotheses corresponding to p-values $p_{(1)}, \ldots, p_{(i_0)}$.

We will now perform this procedure with our 7 p-values obtained in part (a). These p-values are

```
##           rm          dis        lstat          age         crim        indus
## 5.551356e-25 1.511144e-09 8.370354e-23 3.598472e-01 4.756288e-04 4.474581e-03
##          nox
## 3.501612e-02
```

We will now start the Benjamini-Hochberg procedure by ordering the p-values, this is done below.

```
##           rm        lstat          dis         crim        indus          nox
## 5.551356e-25 8.370354e-23 1.511144e-09 4.756288e-04 4.474581e-03 3.501612e-02
##          age
## 3.598472e-01
```

We will now calculate $\frac{i}{m}q$ for all $i \in \{1, \ldots, 7\}$. This corresponds to the vector $\left[\frac{0.01}{7}, \frac{0.02}{7}, \frac{0.03}{7}, \frac{0.04}{7}, \frac{0.05}{7}, \frac{0.06}{7}, \frac{0.07}{7}\right]$, which simplifies to $[0.00143, 0.00286, 0.00429, 0.00571, 0.00714, 0.00857, 0.01]$.

Therefore, we can see that $i_0 = 5$, and thus we would reject the following hypotheses: $H_0^1 : \beta_{rm} = 0$, $H_0^2 : \beta_{dis} = 0$, $H_0^3 : \beta_{lstat} = 0$, $H_0^5 : \beta_{crim} = 0$, and $H_0^6 : \beta_{indus} = 0$. That means we would fail to reject $H_0^4 : \beta_{age} = 0$ and $H_0^7 : \beta_{nox} = 0$.

3. This question is about performing a linear regression without using the `lm()` function in R. You are given the following data: $\underline{x_1} = (4, 3, 3, 0, 1, 5)'$, $\underline{x_2} = (16, 8, 10, 15, 10, 21)'$, and $\underline{y} = (16, 8, 10, 15, 10, 21)'$ and assume that the following model holds: $\underline{y} = \beta_0 + \beta_1 \underline{x_1} + \beta_2 \underline{x_2} + \underline{\epsilon}$ with $\underline{\epsilon} = \mathcal{N}(\underline{0}, \sigma^2 I_6)$.

(a) The model above can be written in matrix form: $\underline{y} = X\underline{\beta} + \underline{\epsilon}$. What is the design matrix X in the above equation equal to?

The design matrix X in the above equation is equal to

$$X = \begin{bmatrix} 1 & x_{11} & x_{12} \\ 1 & x_{21} & x_{22} \\ 1 & x_{31} & x_{32} \\ 1 & x_{41} & x_{42} \\ 1 & x_{51} & x_{52} \\ 1 & x_{61} & x_{62} \end{bmatrix} = \begin{bmatrix} 1 & 4 & 6 \\ 1 & 3 & 2 \\ 1 & 3 & 7 \\ 1 & 0 & 9 \\ 1 & 1 & 3 \\ 1 & 5 & 0 \end{bmatrix}$$

The above design matrix stores the all ones column for the intercept as well as all of the observed data for our covariates. In R this matrix looks like

```
##      [,1] [,2] [,3]
## [1,]    1    4    6
## [2,]    1    3    2
## [3,]    1    3    7
## [4,]    1    0    9
## [5,]    1    1    3
## [6,]    1    5    0
```

(b) Calculate and print $X'X$ and $(X'X)^{-1}$. **Hint:** You can use R functions `t()` and `solve()`.

In this problem we will calculate and print $X'X$ and $(X'X)^{-1}$ using the R functions `t()` and `solve()`. This is done below.

As calculated in R, $X'X$ is

```
##      [,1] [,2] [,3]
## [1,]    6   16   27
## [2,]   16   60   54
## [3,]   27   54  179
```

Furthermore, as calculated in R, $(X'X)^{-1}$ is

```
##            [,1]        [,2]        [,3]
## [1,]  1.9385530 -0.34836472 -0.18731417
## [2,] -0.3483647  0.08548067  0.02675917
## [3,] -0.1873142  0.02675917  0.02576809
```

(c) Calculate the OLS estimates $\hat{\beta}_0$, $\hat{\beta}_1$, and $\hat{\beta}_2$. Compare your calculations to the output of the `lm()` summary.

In this problem, we will calculate the OLS estimates $\hat{\beta}_0$, $\hat{\beta}_1$, and $\hat{\beta}_2$. We will also compare our calculations to the output of the `lm()` summary. We will start by finding the estimates in R without using `lm()`. To do this we will use the fact that $\underline{\hat{\beta}} = (X'X)^{-1}X'\underline{y}$. This is done below.

```
##           [,1]
## [1,] 8.9172448
## [2,] 1.3332507
## [3,] 0.1912785
```

As calculated in R, our estimates of β_0, β_1, and β_2 are $\hat{\beta}_0 = 8.9172448$, $\hat{\beta}_1 = 1.3332507$, and $\hat{\beta}_2 = 0.1912785$. We will now compare this to running the output of the `lm()` summary below.

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##       1       2       3       4       5       6
##  0.6021 -5.2996 -4.2559  4.3612 -0.8243  5.4165
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   8.9172     7.8597   1.135    0.339
## x1            1.3333     1.6504   0.808    0.478
## x2            0.1913     0.9062   0.211    0.846
##
## Residual standard error: 5.645 on 3 degrees of freedom
## Multiple R-squared:  0.1989, Adjusted R-squared:  -0.3352
## F-statistic: 0.3724 on 2 and 3 DF,  p-value: 0.717
```

As can be seen above, our estimates we calculated by hand are exactly the same as those calculated through the `lm()` summary.

(d) Assuming that $\sigma = 0.25$, calculate $Var[\hat{\beta}_1]$ and $Cov[\hat{\beta}_1, \hat{\beta}_2]$. Remember from the lectures that $Var[\underline{\hat{\beta}}] = \sigma^2(X'X)^{-1}$.

In this problem, assuming that $\sigma = 0.25$, we will calculate $Var[\hat{\beta}_1]$ and $Cov[\hat{\beta}_1, \hat{\beta}_2]$. In particular, since $Var[\underline{\hat{\beta}}] = \sigma^2(X'X)^{-1}$, it follows that $Var[\hat{\beta}_1]$ will be the entry in the position $(2, 2)$ of the matrix $\sigma^2(X'X)^{-1}$, and $Cov[\hat{\beta}_1, \hat{\beta}_2]$ will be either the entry in position $(2, 3)$ or $(3, 2)$ of the matrix $\sigma^2(X'X)^{-1}$. These calculations are done in R below.

```
##             [,1]         [,2]         [,3]
## [1,]  0.12115956 -0.021772795 -0.011707136
## [2,] -0.02177279  0.005342542  0.001672448
## [3,] -0.01170714  0.001672448  0.001610505
```

Assuming that $\sigma = 0.25$, with the given data in the problem, $Var[\hat{\beta}_1] = 0.005342542$, and $Cov[\hat{\beta}_1, \hat{\beta}_2] = 0.001672448$.