whikehart_HW1

sophie whikehart

2025-01-26

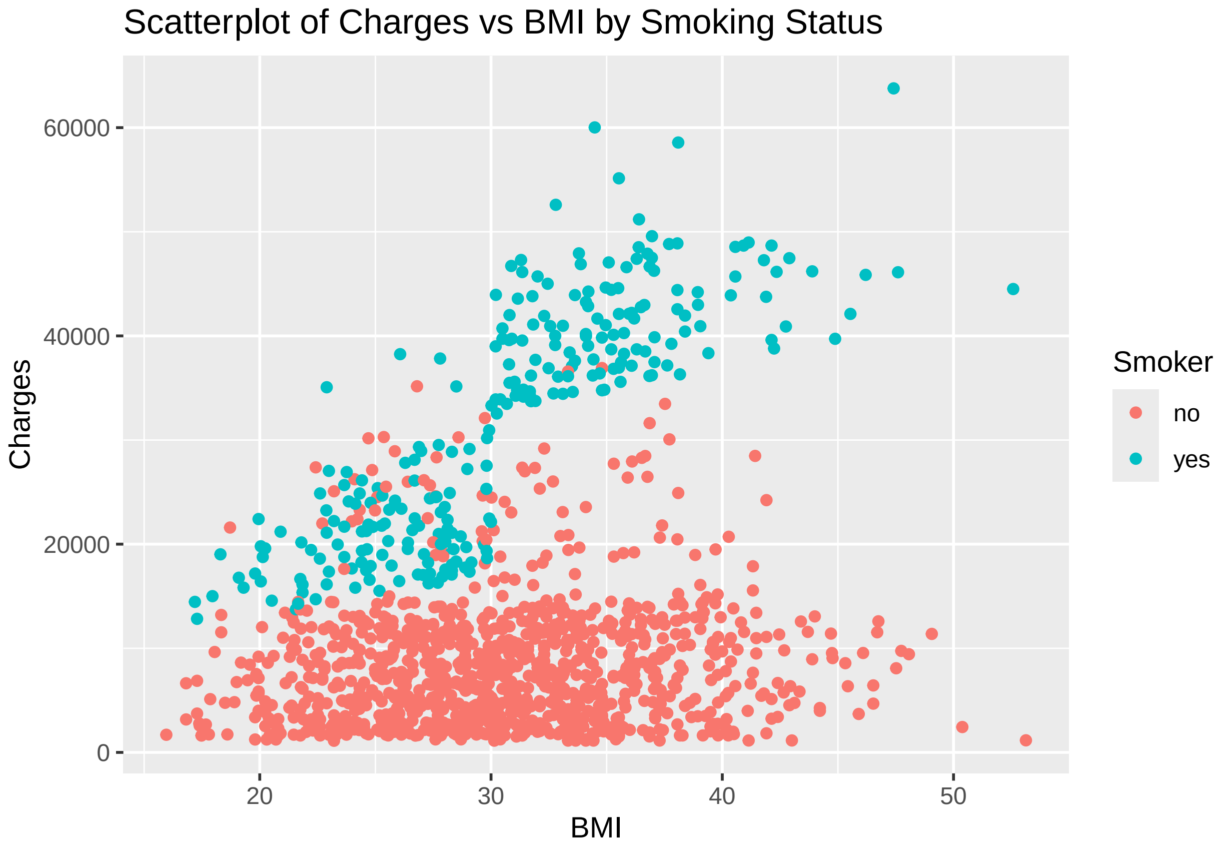
Question 1

```
# a) load dataset
load("Medical_Cost_2.RData")

# b) check for missing data and remove missing values
any(is.na(df))
```

```
## [1] TRUE
```

```
medical_data_clean <- na.omit(df)
```

```
# c) scatterplot with bmi on the x-axis,
# charges on the y-axis and with the color of each dot
# representing whether subject is smoker or not

ggplot(medical_data_clean, aes(x = bmi, y = charges, color = smoker)) +
  geom_point() +
  labs(
    title = "Scatterplot of Charges vs BMI by Smoking Status",
    x = "BMI",
    y = "Charges",
    color = "Smoker"
  )
```



```
# d) Fit a least squares model with intercept in order to predict

  ## charges using bmi as the only predictor

bmi_model <- lm(charges ~ bmi, data = medical_data_clean)

  ## charges using bmi and smoker as predictor

bmi_smoker_model <- lm(charges ~ bmi + smoker, data = medical_data_clean)


  ## charges using bmi and smoker as in the previous model;
  ## but allowing for an interaction term between the variables bmi and smoker

bmi_smoker_interaction_model <- lm(charges ~ bmi + smoker + bmi:smoker, data = medical_data_clean)
```

Model 1

```
# present result in form of table where you report estimated regression
# coefficients and their interpretation [be careful with dummy variable]

# report 95% confidence interval for coefficient of the variable
# bmi and explain meaning of this confidence interval
```

```r
#summary(bmi_model)

# extract coefficient
coefficients <- bmi_model$coefficients

# extract confidence intervals
conf_intervals <- confint(bmi_model)

# create a data frame for the table

bmi_table <- data.frame(
  Variable = names(coefficients),
  Coefficient_Estimates = coefficients,
  CI_Lower = conf_intervals[, 1],
  CI_Upper = conf_intervals[, 2]
)

#table 1

print(bmi_table)
```

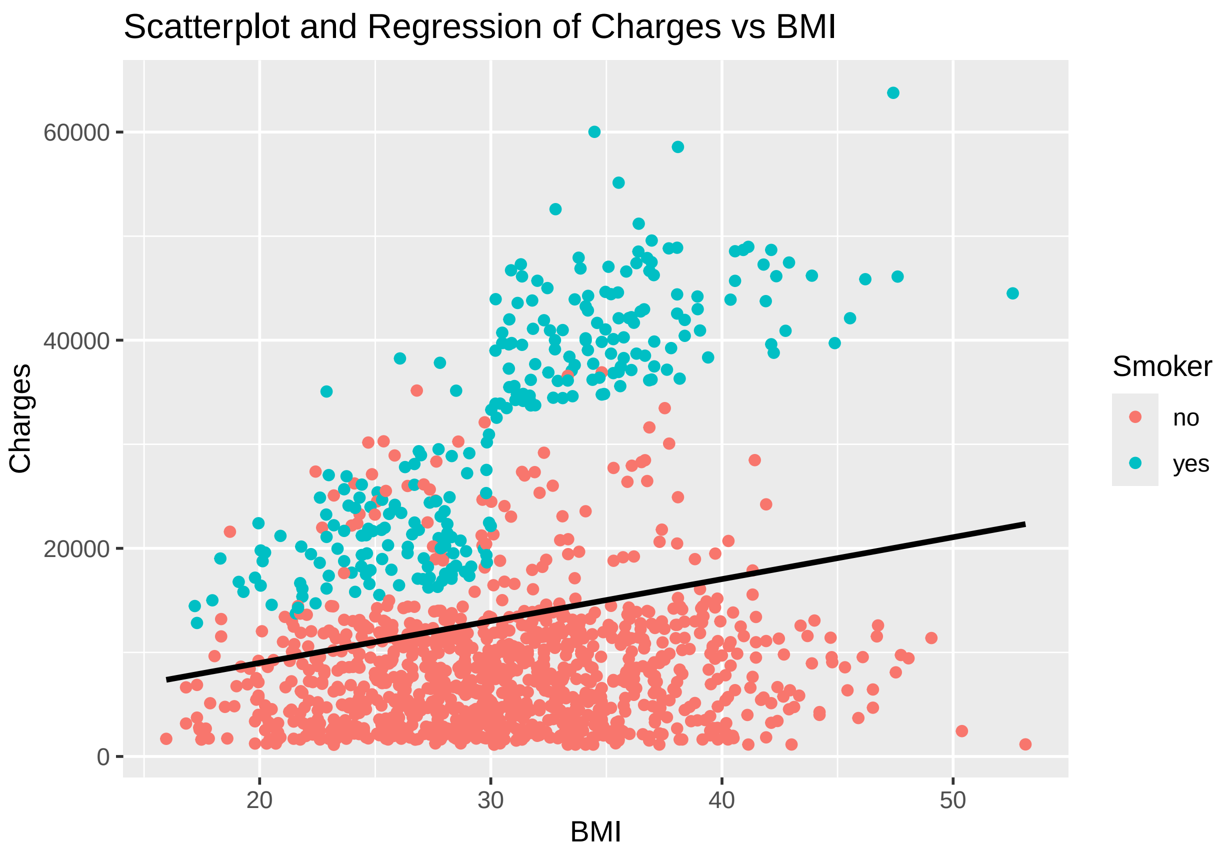
```
##                Variable Coefficient_Estimates   CI_Lower  CI_Upper
## (Intercept) (Intercept)              938.4422 -2361.3490 4238.2333
## bmi                 bmi              402.6474   297.0107  508.2841
```

- For the first model with `charges` using `bmi` only as the predictor, the estimated regression `coefficient` is 938.44. This means that when BMI is 0 (not realistic), the estimated charge is 938.44. The confidence intervals for the intercept are (-2361.3490, 4238.2333). Because this is such a large confidence interval this suggests large uncertainty in the estimate of the intercept coefficient. The BMI coefficient estimate is 402.64 which means that for each unit increase in BMI the outcome variable is expected to increase 402.65 units.

- The 95% confidence interval for the coefficient of variable `bmi` is (297.0107, 508.2841) which spans a large range but not as large as the confidence intervals for the intercept coefficient.

```r
# draw the regression line(s) of the model on the scatterplot produced in
# point (c)

# scatterplot with regression line based on relationship between charges and bmi

ggplot(medical_data_clean, aes(x = bmi, y = charges)) +
  geom_point(aes(color = smoker)) +
  geom_smooth(method = "lm", se = FALSE, color = "black") +
  # Adds regression line
  labs(
    title = "Scatterplot and Regression of Charges vs BMI",
    x = "BMI",
    y = "Charges",
    color = "Smoker"
  )
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



```
# report the (training set) mean squared error of the model

# predict values of charges using the model

prediction1 <- predict(bmi_model, medical_data_clean)

# calculate the residuals

residuals1 <- medical_data_clean$charges - prediction1

# compute mean squared error (MSE)

mse1 <- mean(residuals1^2)
```

- [1] "Training set MSE of charges using bmi as the only predictor is 138358366.167428"

```
# predict the medical costs billed by a health insurance company to a
# smoker with a bmi that is 29 and 31.5

predict(bmi_model, data.frame(bmi = c(29, 31.5), smoker = "yes"))
```

```
##        1        2
## 12615.22 13621.84
```

```
# compute a predicted difference in charges between a smoker with bmi
# 31.5 and one with bmi 29

# do the same for non-smokers

# comment on results

# difference between smoker with bmi 31.65 and bmi 29

bmi_31_5_smoker <- predict(bmi_model, data.frame(bmi = c(31.5), smoker = "yes"))

bmi_29_smoker <- predict(bmi_model, data.frame(bmi = c(29), smoker = "yes"))

difference1 <- (bmi_31_5_smoker - bmi_29_smoker)

# difference between non-smokers with bmi 31.65 and bmi 29

bmi_31_65_no_smoke <- predict(bmi_model, data.frame(bmi = c(31.5), smoker = "no"))
bmi_29_no_smoke <- predict(bmi_model, data.frame(bmi = c(29), smoker = "no"))

difference2 <- (bmi_31_65_no_smoke - bmi_29_no_smoke)
```

- "Predicted difference for model of charges using bmi as the only predictor between smoker with bmi 31.5 and bmi 29 is 1006.61848551234"
- "Predicted difference for model of charges using bmi as the only predictor between non-smoker with bmi 31.5 and bmi 29 is 1006.61848551234"
- The two numbers will be the same because model 1 only accounts for an interaction between charges and bmi and does not account for differences on if patient is a smoker or not.

Model 2

```
# table 2

# present result in form of table where you report estimated regression
# coefficients and their interpretation [be careful with dummy variable]

# report 95% confidence interval for coefficient of the variable bmi
# and explain meaning of this confidence interval

summary(bmi_smoker_model)
```

```
##
## Call:
## lm(formula = charges ~ bmi + smoker, data = medical_data_clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16295.4  -4711.3   -840.1   3698.3  28192.9
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3711.68    1018.78  -3.643  0.00028 ***
## bmi           398.47      32.46  12.275  < 2e-16 ***
## smokeryes   23218.77     491.04  47.285  < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 7097 on 1275 degrees of freedom
## Multiple R-squared:  0.6521, Adjusted R-squared:  0.6515
## F-statistic:  1195 on 2 and 1275 DF,  p-value: < 2.2e-16
```

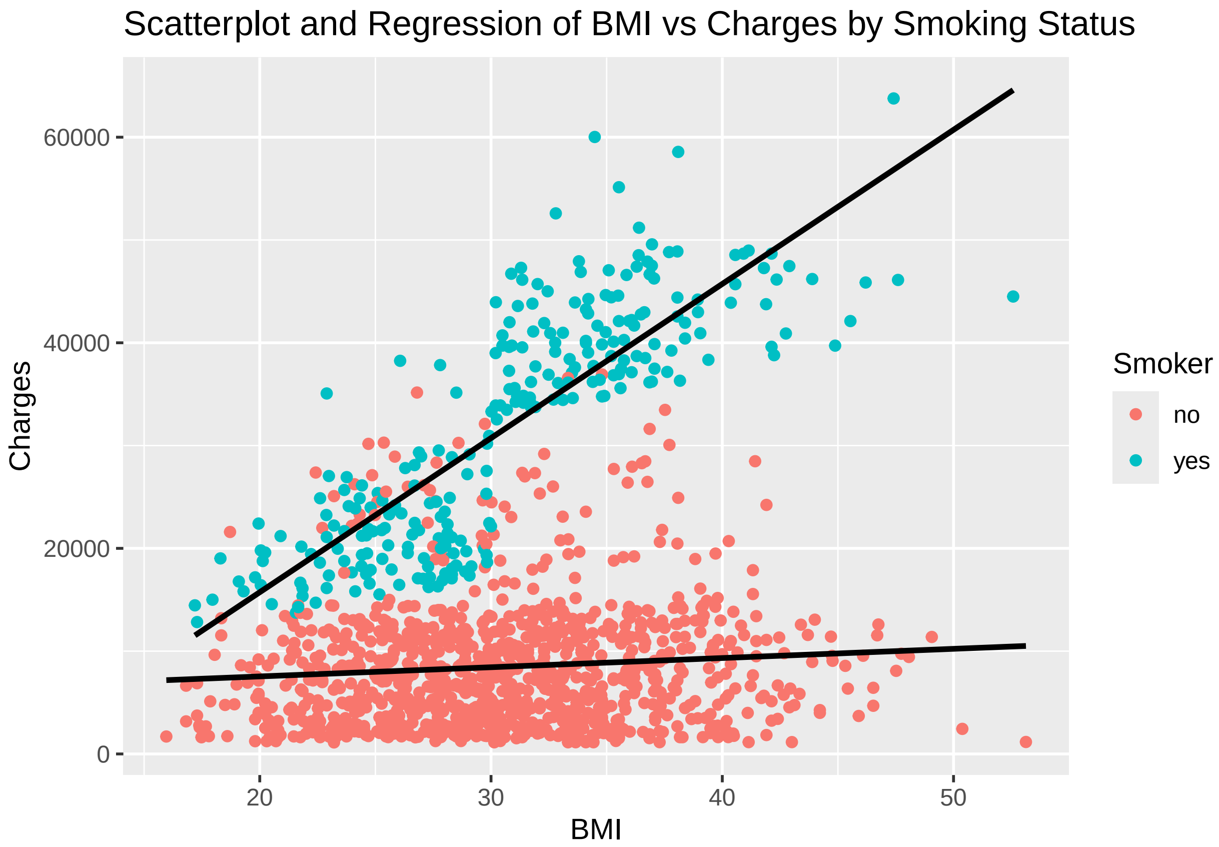
```r
# extract coefficient
coefficients <- bmi_smoker_model$coefficients

# extract confidence intervals
conf_intervals <- confint(bmi_smoker_model)

# create a data frame for the table

bmi_smoker_table <- data.frame(
  Variable = names(coefficients),
  Coefficient_Estimates = coefficients,
  CI_Lower = conf_intervals[, 1],
  CI_Upper = conf_intervals[, 2]
)


print(bmi_smoker_table)
```

```
##                Variable Coefficient_Estimates   CI_Lower   CI_Upper
## (Intercept) (Intercept)            -3711.6846 -5710.3461 -1713.0232
## bmi                 bmi              398.4665   334.7816   462.1514
## smokeryes     smokeryes            23218.7734 22255.4346 24182.1122
```

- The intercept coefficient estimate for the model with charges using `bmi` and `smoker` as predictors is -3711.68. The intercept represents the expected charges for a person with a bmi of 0 and who does not smoke. This intercept doesn’t make sense in the real world because it includes negative charges. The 95% CI is between -5710.35 and -1713.0. The negative charges could also be a result from extrapolating beyond the range of data.
- The coefficient estimate for smokeryes is 23218.77. This suggests that smoking increases medical charges by about 23,218.77 units when compared to non-smokers. The 95% CI lies between 23,2255.43 and 24,182.11.
- The coefficient estimate for bmi is 398.46 so for each unit increase in BMI, charges are expected to increase by about 398.47 units. The 95% CI is between 334.78 and 462.15 units which spans much smaller than when the model has bmi as the only predictor for charges.

```r
# scatterplot with bmi and smoker as predictors

ggplot(medical_data_clean, aes(x = bmi, y = charges, color = smoker)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, aes(group = smoker), color = "black") +
  # Adds separate regression lines for each smoker group
  labs(
    title = "Scatterplot and Regression of BMI vs Charges by Smoking Status",
    x = "BMI",
    y = "Charges",
    color = "Smoker"
  )
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



```
# report the (training set) mean squared error of the model

# predict values of charges using the model

prediction2 <- predict(bmi_smoker_model, medical_data_clean)

# calculate the residuals

residuals2 <- medical_data_clean$charges - prediction2

# compute mean squared error (MSE)

mse2 <- mean(residuals2^2)
```

- "Training set MSE of charges using bmi and smoker as predictors is 50246295.5202116"

```
# predict the medical costs billed by a health insurance company to a smoker
# with a bmi that is 29 and 31.5

predict(bmi_smoker_model, data.frame(bmi = c(29, 31.5), smoker = c("yes")))
```

```
##        1        2
## 31062.62 32058.78
```

```
# compute a predicted difference in charges between a smoker with bmi 31.5
# and one with bmi 29

# do the same for non-smokers

# comment on results

# difference between smoker with bmi 31.65 and bmi 29

bmi_31_5_smoker1 <- predict(bmi_smoker_model, data.frame(bmi = c(31.5), smoker = "yes"))

bmi_29_smoker1 <- predict(bmi_smoker_model, data.frame(bmi = c(29), smoker = "yes"))

difference2 <- (bmi_31_5_smoker1 - bmi_29_smoker1)

# difference between non-smokers with bmi 31.65 and bmi 29

bmi_31_5_no_smoke1 <- predict(bmi_smoker_model, data.frame(bmi = c(31.5), smoker = "no"))
bmi_29_no_smoke1 <- predict(bmi_smoker_model, data.frame(bmi = c(29), smoker = "no"))

difference3 <- (bmi_31_5_no_smoke1 - bmi_29_no_smoke1)
```

- "Predicted difference for model of charges using bmi and smoke as the only predictor between smoker with bmi 31.5 and bmi 29 is 996.16629780822"
- "Predicted difference for model of charges using bmi and smoker as the only predictor between non-smoker with bmi 31.5 and bmi 29 is 996.166297808219"
- The difference between these two results shows that smoking doesn't appear to have a differential effect on the predicted charges between these two BMI values. This was surprising because I thought that the two values would be different.

Model 3

```
# table 3


# present result in form of table where you report estimated regression
# coefficients and their interpretation [be careful with dummy variable]
```

```r
# report 95% confidence interval for coefficient of the variable bmi
# and explain meaning of this confidence interval

summary(bmi_smoker_interaction_model)
```

```
##
## Call:
## lm(formula = charges ~ bmi + smoker + bmi:smoker, data = medical_data_clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20086.4  -4386.4   -901.9   3005.2  28045.9
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     5750.97     991.45   5.801 8.33e-09 ***
## bmi               89.47      31.76   2.817  0.00492 **
## smokeryes     -20008.39    2122.67  -9.426  < 2e-16 ***
## bmi:smokeryes   1410.05      67.84  20.784  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6135 on 1274 degrees of freedom
## Multiple R-squared:  0.7402, Adjusted R-squared:  0.7396
## F-statistic:  1210 on 3 and 1274 DF,  p-value: < 2.2e-16
```

```r
# extract coefficient
coefficients <- bmi_smoker_interaction_model$coefficients

# extract confidence intervals
conf_intervals <- confint(bmi_smoker_interaction_model)

# create a data frame for the table

bmi_smoker_interaction_table <- data.frame(
  Variable = names(coefficients),
  Coefficient_Estimates = coefficients,
  CI_Lower = conf_intervals[, 1],
  CI_Upper = conf_intervals[, 2]
)


print(bmi_smoker_interaction_table)
```

```
##                    Variable Coefficient_Estimates     CI_Lower    CI_Upper
## (Intercept)     (Intercept)            5750.97441   3805.92094   7696.0279
## bmi                     bmi              89.47383     27.16986    151.7778
## smokeryes         smokeryes          -20008.38717 -24172.70361 -15844.0707
## bmi:smokeryes bmi:smokeryes            1410.05387   1276.95956   1543.1482
```

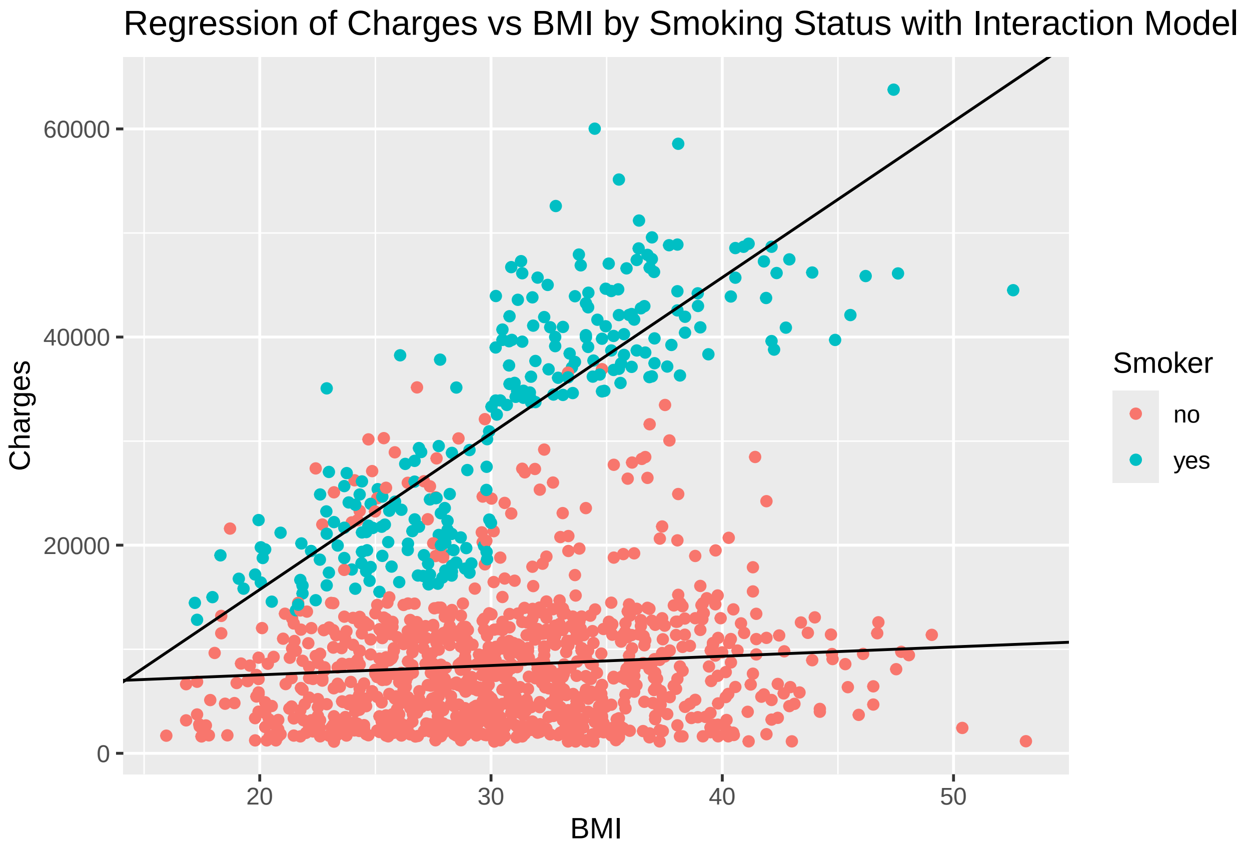
- The coefficient intercept for the model of `charges` using `bmi` and `smoker` as in the previous model but allowing for an interaction term between `bmi` and `smoker` is 5750.97. This intercept suggests that charges will be 5750.97 when both bmi and smoker are 0. CI 95% is between 3805,92 and 7696.03.

- The intercept for smokeryes is -20008.39. This means that smokers are associated with a decrease in charges. The coefficient for smokeryes being negative seems counter-inuitive but it could be that there are confounding variables we are not accounting for in this model or the interaction term plays a significant role in this model. Even though smokers have lower charges on average, when you account for BMI the charges could be higher due to the interaction effect.

- The coefficient estimate for the interaction term `bmi:smokeryes` is 1410.05, so the effect for smokers of BMI on charges is increased by 1410,04 per unit increase.

- The coefficient estimate for BMI is 89.47 so for each unit increase in BMI the charges increase by 89.47 assuming smoking status remains the same. The 95% CI for BMI is 21.17 and 151.78 which is a larger span than when the model was just charges with smoking and bmi as predictors.

```
ggplot(medical_data_clean, aes(x = bmi, y = charges, color = smoker)) +
  geom_point() +
  geom_abline(intercept = coef(bmi_smoker_interaction_model)[1], slope = coef(bmi_smoker_interaction_mo
  geom_abline(intercept = coef(bmi_smoker_interaction_model)[1] + coef(bmi_smoker_interaction_model)[3]
  labs(
    title = "Regression of Charges vs BMI by Smoking Status with Interaction Model",
    x = "BMI",
    y = "Charges",
    color = "Smoker"
  )
```



```
# report the (training set) mean squared error of the model
```

```r
# predict values of charges using the model

prediction3 <- predict(bmi_smoker_interaction_model, medical_data_clean)

# calculate the residuals

residuals3 <- medical_data_clean$charges - prediction3

# compute mean squared error (MSE)

mse3 <- mean(residuals3^2)
```

- "Training set MSE of Charges using bmi and smoker with interaction term is 37522940.5056037"

```r
# predict the medical costs billed by a health insurance company to
# a smoker with a bmi that is 29 and 31.5

predict(bmi_smoker_interaction_model, data.frame(bmi = c(29, 31.5), smoker = c("yes")))
```

```
##        1        2
## 29228.89 32977.71
```

```r
# compute a predicted difference in charges between a smoker
# with bmi 31.5 and one with bmi 29

# do the same for non-smokers

# comment on results

# difference between smoker with bmi 31.65 and bmi 29

bmi_31_5_smoker2 <- predict(bmi_smoker_interaction_model, data.frame(bmi = c(31.5), smoker = "yes"))

bmi_29_smoker2 <- predict(bmi_smoker_interaction_model, data.frame(bmi = c(29), smoker = "yes"))

difference3 <- (bmi_31_5_smoker2 - bmi_29_smoker2)

# difference between non-smokers with bmi 31.65 and bmi 29

bmi_31_5_no_smoke2 <- predict(bmi_smoker_interaction_model, data.frame(bmi = c(31.5), smoker = "no"))
bmi_29_no_smoke2 <- predict(bmi_smoker_interaction_model, data.frame(bmi = c(29), smoker = "no"))

difference4 <- (bmi_31_5_no_smoke2 - bmi_29_no_smoke2)
```

- "Predicted difference for model of charges using bmi and smoke with interaction term between smoker with bmi 31.5 and bmi 29 is 3748.81924147658"
- "Predicted difference for model of charges using bmi and smoker with an intereaction term between non-smoker with bmi 31.5 and bmi 29 is 223.684578048067"

The results of this make more sense as the predicted difference in charges for smokers with bmi 31.5 and 29 is 3748.42 while non-smokers the differences is 223.68. Therefore, for smokers the interaction term suggests the relationship between BMI and charges is stronger and that smoking status modifies the effect of BMI on medical charges.

```r
# e) now define and add a new boolean variable smoker_bmi30p that is only
# true if the subject is a smoker and has a bmi grearter than 30

medical_data_clean$smoker_bmi30p <- with(medical_data_clean, smoker == "yes" & bmi > 30)

# use this newly defined variable, together with bmi and smoker to fit the
# linear model represented in figure 1 by carefully defining the
# interaction terms (allow each of the three straight lines
# to have their own intercept and slop, but use command lm only one)


figure_1_lm_model <- lm(charges ~ bmi * smoker_bmi30p, data = medical_data_clean)

# present results in form of table where you report the estimated coefficients
# of the model

table <- figure_1_lm_model$coefficients
table
```

```
##           (Intercept)                   bmi     smoker_bmi30pTRUE
##            12429.4462              -84.0272             8641.5277
## bmi:smoker_bmi30pTRUE
##              659.2990
```

- From this linear regression model, charges is the outcome of interest. BMI is the continous variable with smoker_bmi30p as the binary variable indicating if the individual is a smoker which we want them to be and with a bmi > 30. The interaction term allows for the affect of BMI on charges to differ dependning on if someone is a smoker with a bmi > 30.
- The intercept is 12429.44 which is the predicted value of the outcome when all predictor variables are zero.
- BMI of -81.02 suggest that for every 1 unit increase in BMI, the outcome variable will decrease by 84.02 units.
- smoker_bmi30pTRUE is 8641.52 which suggests that if a person is a smoker with a bmi of 30 or higher the outcome variable will increase by 8641.52 units.
- bmi:smoker_bmi30pTRUEis 659.29 and is the interaction term. It shows that for each unit increase in BMI the outcome will change by an addition 659.29 units for smokers with bmi > 30 compared to non-smokers or those with bmi < 30.

```r
# interpret the non-significant variable in the model (p > 0.05)
# and explain how Figure 1 would change if we were to discard
# those variables, ex - perform variable selection

summary(figure_1_lm_model)
```

```
##
## Call:
## lm(formula = charges ~ bmi * smoker_bmi30p, data = medical_data_clean)
##
```

```
## Residuals:
##    Min     1Q Median     3Q    Max 
##  -9524  -5358  -1423   3156  28007 
## 
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)    
## (Intercept)           12429.45    1045.67  11.887  < 2e-16 ***
## bmi                     -84.03      34.12  -2.462   0.0139 *  
## smoker_bmi30pTRUE      8641.53    5210.54   1.658   0.0975 .  
## bmi:smoker_bmi30pTRUE   659.30     146.44   4.502 7.34e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 6964 on 1274 degrees of freedom
## Multiple R-squared:  0.6652, Adjusted R-squared:  0.6644 
## F-statistic: 843.8 on 3 and 1274 DF,  p-value: < 2.2e-16
```

- In this model, the non-significant variable is `smoker_bmi30pTRUE` with a p-value of 0.0975. This variable indicates if a smoker is a person with a BMI over 30. The p-value is non-significant since it is above p > 0.05 cutoff so that is has weak association with the variable `charges`. Accounting for other variables like `bmi`, and the interaction between `bmi` and smoking status, the effect of being a smoker with a BMI over 30 on `charges` is not strong enough to effect this model
- If we were to discard the non-significant variable then the model would just be `bmi` and `bmi:smoker_bmi30pTRUE`
  - The model would become more simple and it may shift the coefficients for `bmi` and the interaction term.

```
# compute a predicted difference in charges between a smoker
# with bmi 31.5 and one with bmi 29

# do the same for non-smokers

# comment on results

# difference between smoker with bmi 31.65 and bmi 29

bmi_31_5_smoker3 <- predict(figure_1_lm_model, data.frame(bmi = c(31.5), smoker_bmi30p = TRUE))

bmi_29_smoker3 <- predict(figure_1_lm_model, data.frame(bmi = c(29), smoker_bmi30p = TRUE))

difference4 <- (bmi_31_5_smoker3 - bmi_29_smoker3)

# difference between non-smokers with bmi 31.65 and bmi 29

bmi_31_5_no_smoke3 <- predict(figure_1_lm_model, data.frame(bmi = c(31.5), smoker_bmi30p = FALSE))
bmi_29_no_smoke3 <- predict(figure_1_lm_model, data.frame(bmi = c(29), smoker_bmi30p = FALSE))

difference5 <- (bmi_31_5_no_smoke3 - bmi_29_no_smoke3)
```

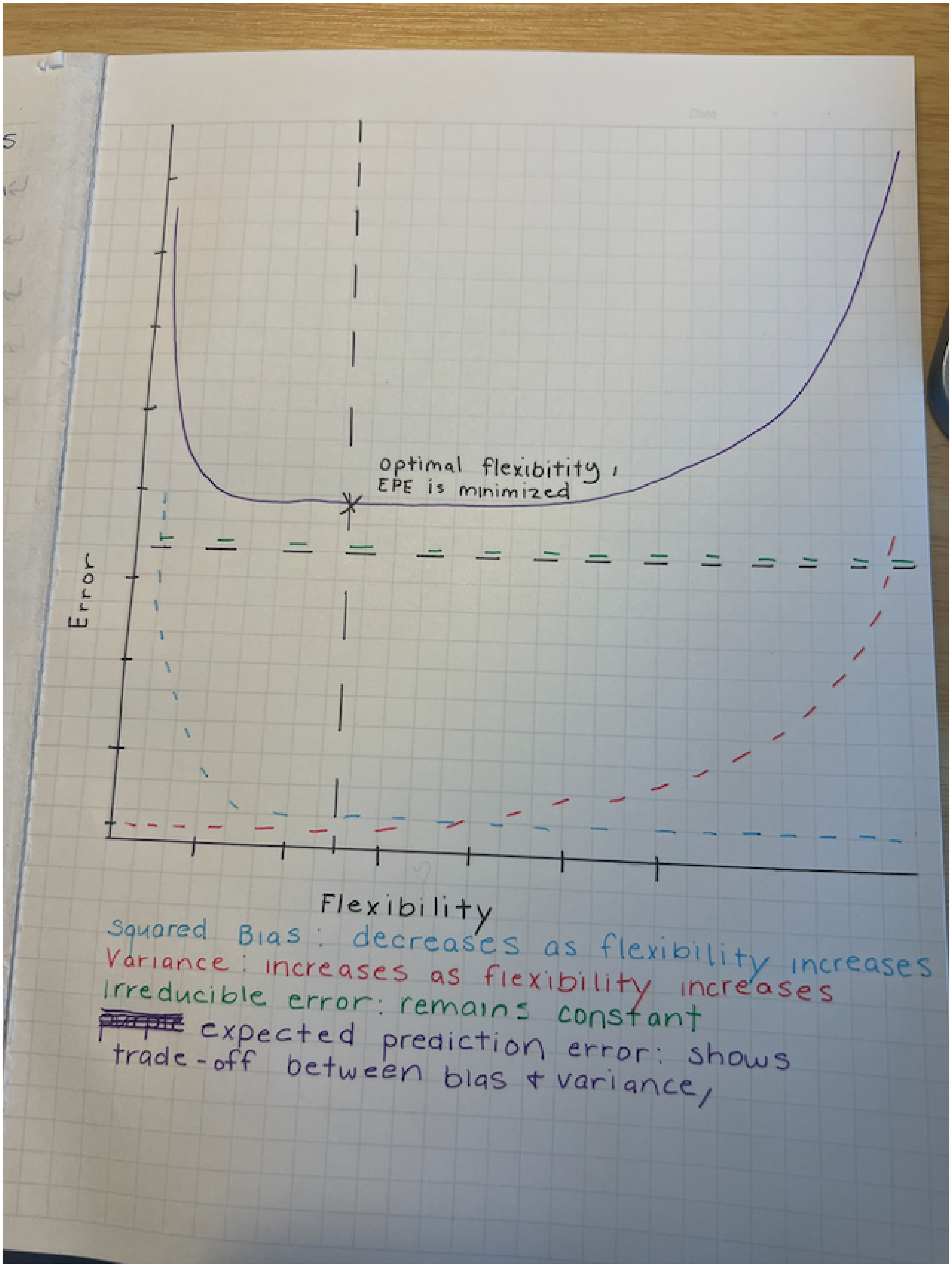
- "Calculated difference in charges between person with bmi 31.5 and bmi 29 for model with charges by bmi and if the person is a smoker with bmi over 30 is 1438.17940117325"

- "Calculated difference in charges between person with bmi 31.5 and bmi 29 for model with charges by bmi and if the person is not a smoker with bmi over 30 is -210.067990031132"
- The results of the differences of the charges suggest the smoking significantly increases the charges for individuals with a BMI over 30 while being a non-smoker results in decrease in charges. When compared to previous models, this is most similar to model three which models an interaction between charges, bmi and smoking accounting for an interaction term for bmi and smoking.

Question 2

a) Make a plot, like the one we saw in class with "flexibility" on the x-axis

- Sketch the following curves: squared bias, variance, irreducible error, expected prediction error
- Be sure to label each curve
- Indicate which level of flexibility is "best"



Squared Bias: decreases as flexibility increases
Variance: increases as flexibility increases
Irreducible error: remains constant
~~purple~~ expected prediction error: shows trade-off between bias & variance,

b) Make a plot with ‘flexibility’ on the x-axis

- Sketch curves corresponding to training error and test error
- Be sure to label each curve
- Indicate which level of flexibility is ‘best’!

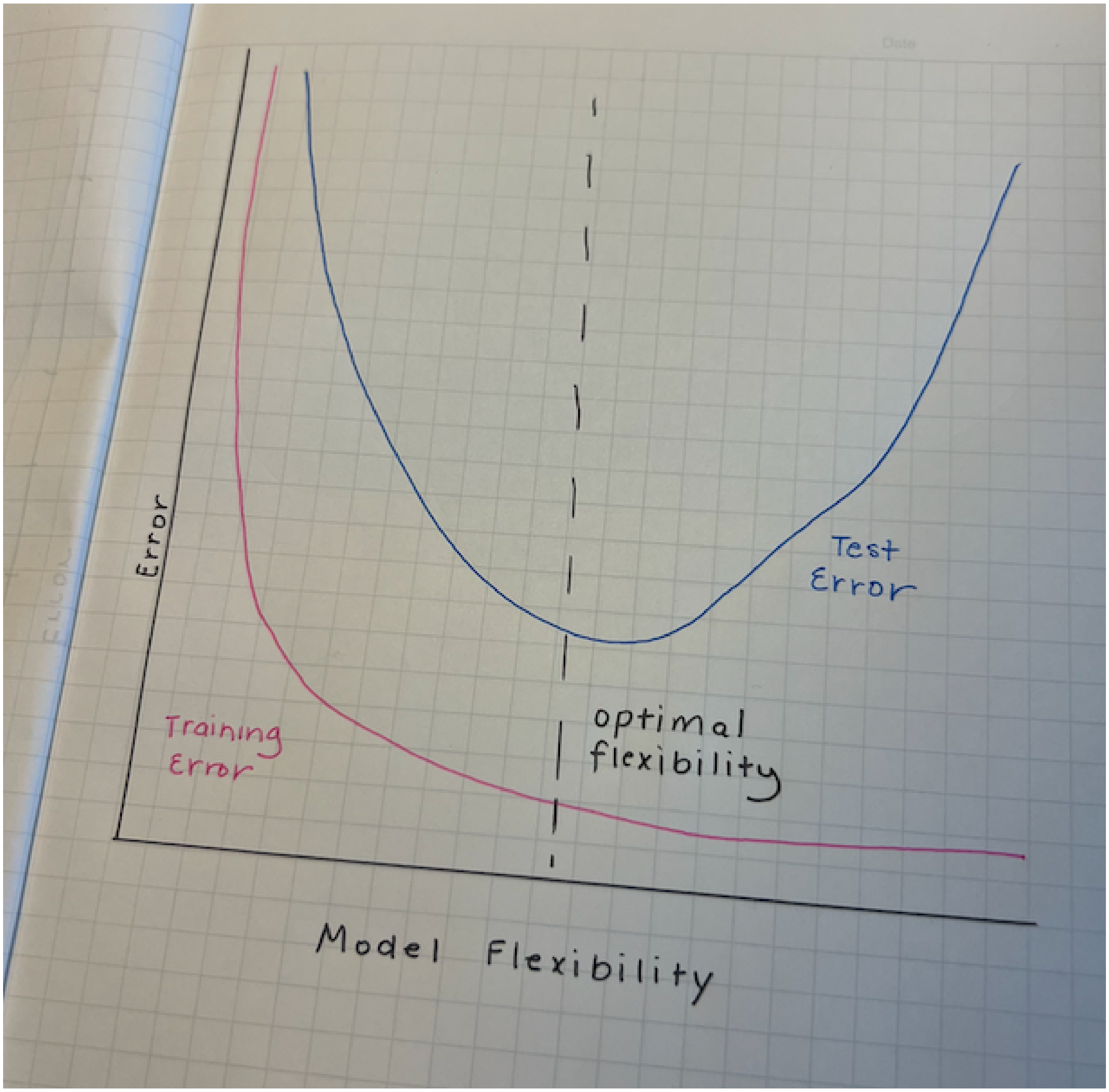


Figure 1: Training Error and Test Error Plot

Question 3

- Numerical explorations of bias-variance trade-off phenomenon
  - Generate stimulated data and use these data to perform **linear regression**
  - Set the seed with `set.seed(0)` before you begin

```r
# a) Use the `rnorm()` function to generate a predictor vector X of
# length n = 30, use `runif()` to generate a noise vector E of length n = 30

set.seed(0) # for reproducibility
n <- 30
X <- rnorm(n) # predictor vector of length n = 30
epsilon <- runif(n, min = -0.5, max = 0.5)

# b) generate response vector

Y <- 3 + 2*X + 3*X^3 + epsilon

# c) Fit models with various forms

# model 1
model1 <- lm(Y ~ X)

# model 2 quadratic term
X2 <- X^2
model2 <- lm(Y ~ X + X2)

# model 3 cubic and quadratic terms
X3 <- X^3
X4 <- X^4
model3 <- lm(Y ~ X + X2 + X3 + X4)

# model 4 linear and cubic terms
model4 <- lm(Y ~ X + X3)
```

```r
# d) For each of the models above compute the training mean squared error
 # Comment on the results.

# model 1 linear
y_pred1 <- predict(model1)
mse1 <- mean((Y - y_pred1)^2)
print(paste("MSE for Model 1 is:", mse1))
```

```
## [1] "MSE for Model 1 is: 26.580334321571"
```

```r
# model 2 linear + quadratic
y_pred2 <- predict(model2)
mse2 <- mean((Y - y_pred2)^2)
print(paste("MSE for Model 2 is:", mse2))
```

```
## [1] "MSE for Model 2 is: 13.2048064723429"
```

```r
# model 3 linear + quadratic + cubic + quartic
y_pred3 <- predict(model3)
mse3 <- mean((Y - y_pred3)^2)
print(paste("MSE for Model 3 is:", mse3))
```

```
## [1] "MSE for Model 3 is: 0.0550314059392257"
```

```
# model 4 linear + cubic
y_pred4 <- predict(model4)
mse4 <- mean((Y - y_pred4)^2)
print(paste("MSE for Model 4 is:", mse4))
```

```
## [1] "MSE for Model 4 is: 0.0569268496769212"
```

- Based on the results for MSE, model 1 has the highest MSE which is likely due to poor fit in the data.
- Model 2 has lower MSE so the quadratic term improves the fit but it is still not the lowest.
- Model 3 MSE is the lowest as it captures the full complexity of the data. It fits the data almost perfectly but may be over fitting.
- Finally, Model 4 has second lowest MSE and is close to Model 3. It has a good fit and may be the best for getting bias-variance trade-off and is simplier than model 3.

```
# e) generate 10k (new) test observations following steps 3(a) and 3(b)
  # compute the test MSE of the models fitted in 3(c) on these test observations
  # report and comment on results

set.seed(0)
n_test <- 10000
X_test <- rnorm(n_test)
epsilon_test <- runif(n_test, min = -0.5, max = 0.5)
Y_test <- 3 + 2*X_test + 3*X_test^3 + epsilon_test

# prediction for model 1
y_pred1_test <- predict(model1, newdata = data.frame(X = X_test))
mse1_test <- mean((Y_test - y_pred1_test)^2)
print(paste("Test MSE for Model 1", mse1_test))
```

```
## [1] "Test MSE for Model 1 49.0271447535418"
```

```
# prediction for model 2
y_pred2_test <- predict(model2, newdata = data.frame(X = X_test, X2 = X_test^2))
mse2_test <- mean((Y_test - y_pred2_test)^2)
print(paste("Test MSE for Model 2", mse2_test))
```

```
## [1] "Test MSE for Model 2 78.7527358109701"
```

```
# prediction for model 3
y_pred3_test <- predict(model3, newdata = data.frame(X = X_test, X2 = X_test^2, X3 = X_test^3, X4 = X_te
mse3_test <- mean((Y_test - y_pred3_test)^2)
print(paste("Test MSE for Model 3", mse3_test))
```

```
## [1] "Test MSE for Model 3 0.115771413400653"
```

```
# prediction for model 4
y_pred4_test <- predict(model4, newdata = data.frame(X = X_test, X3 = X_test^3))
mse4_test <- mean((Y_test - y_pred4_test)^2)
print(paste("Test MSE for Model 4", mse4_test))
```

```
## [1] "Test MSE for Model 4 0.0983129079480685"
```

- Model 1 as predicted, has the highest test MSE. This was expected that the linear model would not be able to capture the data and have a poor fit.
- Model 2 had a slightly higher test MSE than Model 1. This shows that over fitting the data resulted in higher MSE.
- Model 3 performs better than model 1 and 2 but also likely over fits the data as well and it is the most complex model.
- Model 4 has the lowest test MSE and capture the data without over fitting so it is the optimal model for this data.

```
# f) compute the training and test MSE of the true regression function f^true
  # compare to those of models fitted in 3(c)
  # comment on results

# training MSE for true function
Y_train_true <- 3 + 2*X + 3*X^3
train_mse_true <- mean((Y - Y_train_true)^2)
print(paste("Training MSE for true function f_true", train_mse_true))
```

```
## [1] "Training MSE for true function f_true 0.0665123315193317"
```

```
# test MSE for true function on test data
Y_test_true <- 3 + 2*X_test + 3*X_test^3
test_mse_true <- mean((Y_test - Y_test_true)^2)
print(paste("Test MSE for true function f_true", test_mse_true))
```

```
## [1] "Test MSE for true function f_true 0.0822663850280152"
```

- The results of the training MSE for the true function is very low 0.06 which is expected because the true function perfectly fits the training data.

- The test MSE for the true function is a bit higher and is also expected because the model is not aligned with every new test point.

- Model 1 had a test MSE of 49.03 which is much higher than the true function MSE. This confirms that a linear model is not the right model to fit the true data.

- Model 2 had a test MSE of 78.75 which was also much higher than true MSE. This shows that quadratic function is not right to capture this data.

- Model 3 had test MSE of 0.116 which is higher than true function test MSE. This shows that overfitting leads to increase in test MSE compared to true function.

- Mode 4 had lowest test MSE of 0.09 which is higher than true function test MSE but is the best performing model compared to the others.