Problem Set 5: More Regression Modeling

Linh Vu (Collab: Brice Laurent)

Due: October 27, 2023

This assignment is **due Friday, October 27 at 11:59pm**, handed in on Gradescope (remember, there are two separate submissions, one for your pdf, and another for you rmd file). Show your work and provide clear, explanations when asked. **Incorporate the relevant R output in this R markdown file**. Only the key output should be displayed for each problem and the relevant parts should be **highlighted** in some way. Make sure that you write-up any interpretation of R-code in your own words (don't just provide the output).

Collaboration policy (for this and all future homeworks): You are encouraged to discuss the problems with other students, but you must write up your solutions yourself and in your own words. Copying someone else's solution, or just making trivial changes is not acceptable.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.1     v purrr   0.3.4
## v tibble  3.2.1     v dplyr   1.1.1
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.3     v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Problem 1.

Let $X \sim \text{Unif}(a, b)$. Feel free to use results from the *Stat 110 Distribution Sheet* as seen on the midterm exam.

```
set.seed(139)
a = -0.5
b = 0.5
x = runif(10^6, a, b)
x2 = x^2
cov(x, x2)
```

```
## [1] -5.103786e-07
```

```
(a+b)*(a-b)^2/12
```

```
## [1] 0
```

(a) Determine the covariance between X and X^2.

Using the formula $Cov(X,Y) = E(XY) - E(X)E(Y)$, we get:

$$Cov(X, X^2) = E(X^3) - E(X)E(X^2)$$

Find $E(X^2)$ using LOTUS: $E(X^2) = \int_a^b X^2 \frac{1}{b-a} dX = \frac{b^3-a^3}{3(b-a)}$

Find $E(X^3)$ using LOTUS: $E(X^3) = \int_a^b X^3 \frac{1}{b-a} dX = \frac{b^4-a^4}{4(b-a)}$

Combining everything, we get:

$$Cov(X, X^2) = \frac{b^4 - a^4}{4(b-a)} - \frac{a+b}{2}\frac{b^3-a^3}{3(b-a)}$$

$$= \frac{3(b^2-a^2)(b^2+a^2) - 2(a+b)(b^3-a^3)}{12(b-a)}$$

Dividing both the numerator and denominator by $b-a$, we get:

$$= \frac{3(a+b)(b^2+a^2) - 2(a+b)(a^2+ab+b^2)}{12}$$

$$= \frac{(a+b)(3a^2+3b^2-2a^2-2ab-2b^2)}{12}$$

$$= \frac{(a+b)(a-b)^2}{12}$$

(b) Assume $b-a = 1$ (so that the variability of X is fixed). For what values of a (and b) will this covariance be zero? When will this covariance be large (and positive)? When will it be negative (and large in magnitude)? What does this mean for where the distribution of X is centered in each case?

Using results from part (a), we know that the covariance in this case is $\frac{a+b}{12}$

The covariance is 0 when $a + b = 0$. And since $b - a = 1$, we find that $a = -0.5$, $b = 0.5$.

The covariance is large and positive when $a + b$ is large and positive (i.e. a and b are both large and 1 unit apart). Similarly, the covariance is negative and large in magnitude when $a + b$ is negative and large in magnitude (i.e. a and b are both negative and large in magnitude). When X is centered at a negative value far from 0, the covariance is very negative, and if the center is positive and far from 0, the covariance is large and positive, and if the center is 0, the covariance is 0.

(c) What are the implications of the results in (b) for a quadratic regression model (when X and X^2 are both used as predictors)? Is this phenomenon specific to the Uniform distribution?

*Note: do not be afraid to check your answers empirically using R.

Because X and X^2 very rarely have covariance of 0 (i.e. not at all correlated), when X and X^2 are both used as predictors, we face potential issues of multicollinearity. As a result, we need to be careful about including many polynomial terms. This phenomenon is not specific to the Uniform distirbution (see R chunk below).

```
set.seed(139)
x = rnorm(10^6, 2, 4)
x2 = x^2
cov(x, x2)
```

```
## [1] 64.18893
```

```r
x = rpois(10^6, 4)
x2 = x^2
cov(x, x2)
```

```
## [1] 36.00201
```

Problem 2.

The file 'pregnancydata.csv' includes several variables to model the birthweight of babies (measured through an online survey). Those variables are defined below. Use this data set in R to answer the questions below:

id:	a unique identifier of the mother
weight:	birthweight of the newborn baby, in ounces
pregnancylength:	the length of the pregnancy, in days
country:	where the birth took place with categories United States (US), United Kingdom (UK), Canada (Can), and Other
motherage:	age of mother at childbirth, in years
multiples:	whether the baby was a 1=singleton or 2=twin
sex:	sex of the baby: girl or boy
induced:	a binary indicator for whether labor was induced with oxytocin
cesarean:	a binary indicator for whether a cesarean (c-section) was performed
previousbirths:	the number of births by the mother previous to this recorded one (from 0 to 10)

(a) Fit a regression model to predict weight from country and use the `relevel` command to make the "Other" group the reference group (call this **Model 1**). Interpret the results and provide a visual to support your conclusions.

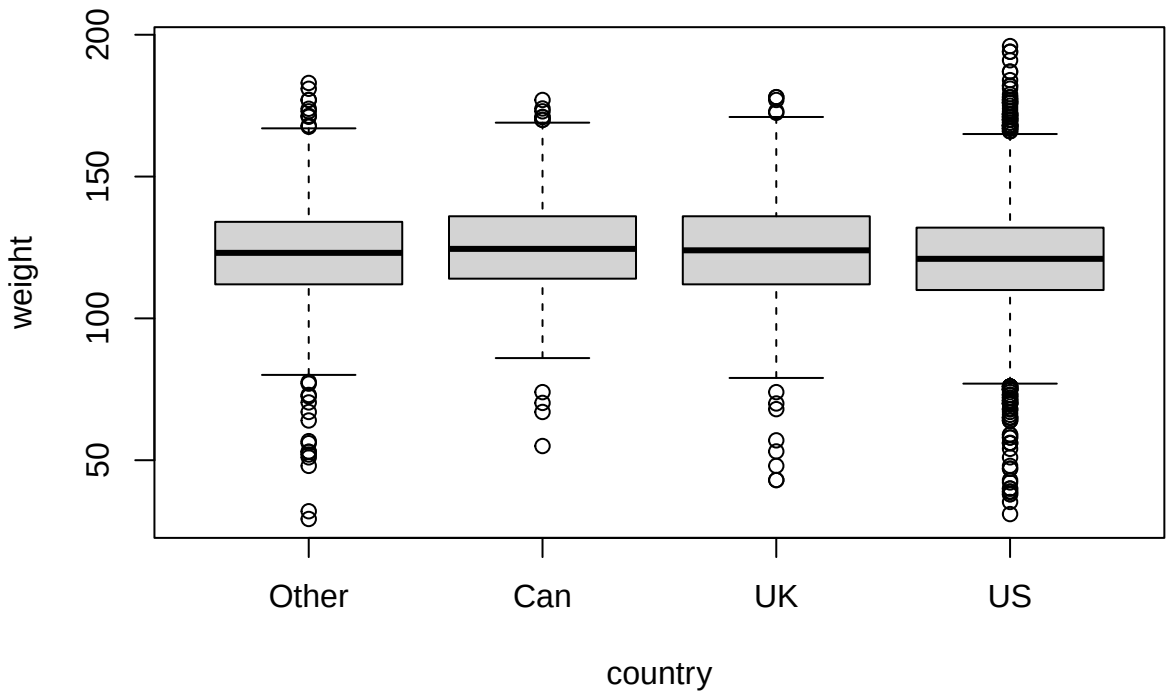
```
pregnancy <- read.csv("data/pregnancydata.csv")
pregnancy$country = relevel(as.factor(pregnancy$country), "Other")

mod1 <- lm(weight~country, pregnancy)
summary(mod1)
```

```
##
## Call:
## lm(formula = weight ~ country, data = pregnancy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -93.380 -11.311  -0.311  11.383  74.689
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 122.6570     0.6083 201.631   <2e-16 ***
## countryCan    2.2965     0.9712   2.365   0.0181 *
## countryUK     0.9596     0.7868   1.220   0.2227
## countryUS    -1.3458     0.6480  -2.077   0.0378 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.82 on 9061 degrees of freedom
## Multiple R-squared:  0.004002,   Adjusted R-squared:  0.003672
## F-statistic: 12.13 on 3 and 9061 DF,  p-value: 6.366e-08
```

```
x = c("Other", "Can", "UK", "US")
predict(mod1, new=data.frame(country=x))
```

```
##        1        2        3        4
## 122.6570 124.9535 123.6166 121.3112
```

```
plot(weight~country, pregnancy)
```



The intercept means that babies born in other countries weigh 122.657 ounces on average. The other slope estimates mean the difference between average weigh of babies born in Canada, UK, US and those born in other countries. Specifically, compared to babies born in other countries, babies born in Canada weigh 2.297 ounce more (this difference is significant due to small p-value); babies born in the UK weigh 0.96 ounce more (this difference is not significant); babies born in the US weigh 1.346 ounce less (this difference is significant).

The side-by-side boxplot shows that babies in Canada weigh slightly more on average, and babies in the US weigh slightly less on average.

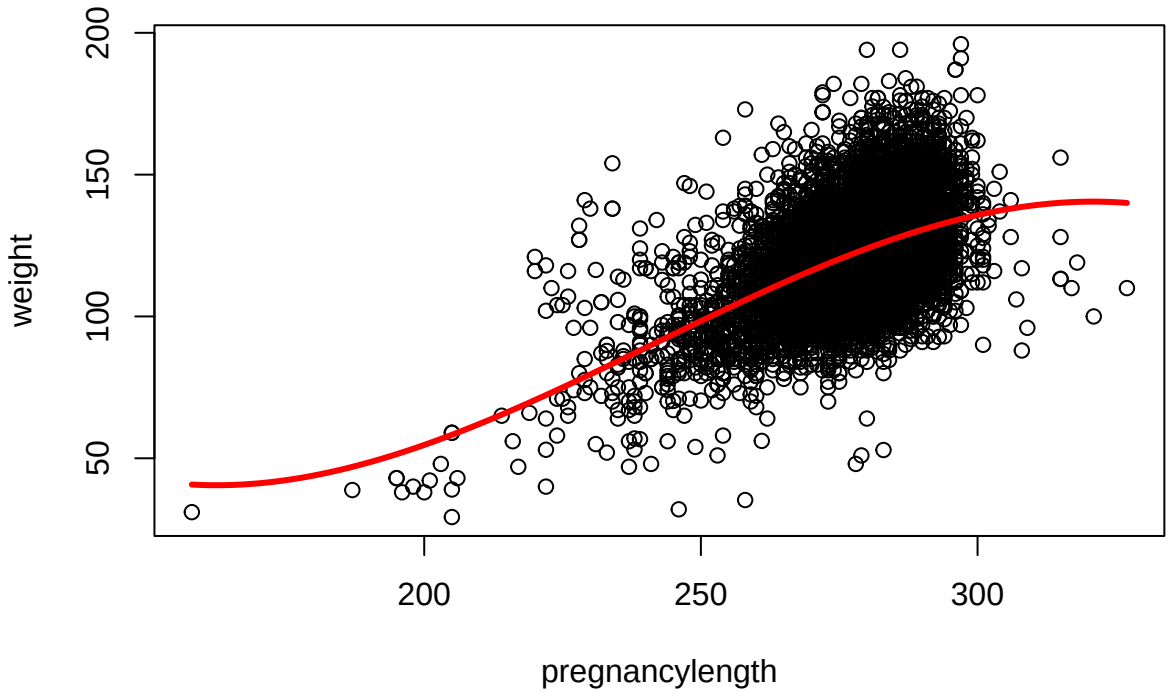
(b) Build a 3^{rd} order polynomial regression model to predict weight from `pregnancylength` (call this **Model 2**). Interpret the output and provide a visual to support the results of the model.

```
mod2 <- lm(weight~poly(pregnancylength, 3, raw=T), pregnancy)
summary(mod2)
```

```
##
## Call:
## lm(formula = weight ~ poly(pregnancylength, 3, raw = T), data = pregnancy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -74.486 -10.087  -0.761   9.364  70.727
##
```

```
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          5.712e+02  2.000e+02   2.856  0.00431 **
## poly(pregnancylength, 3, raw = T)1 -7.861e+00  2.353e+00  -3.341  0.00084 ***
## poly(pregnancylength, 3, raw = T)2  3.645e-02  9.195e-03   3.964 7.44e-05 ***
## poly(pregnancylength, 3, raw = T)3 -5.028e-05  1.193e-05  -4.213 2.54e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.33 on 9061 degrees of freedom
## Multiple R-squared:  0.2627, Adjusted R-squared:  0.2625
## F-statistic:  1076 on 3 and 9061 DF,  p-value: < 2.2e-16
```

```r
x = min(pregnancy$pregnancylength):max(pregnancy$pregnancylength)
yhat = predict(mod2, new=data.frame(pregnancylength=x))
plot(weight~pregnancylength, pregnancy)
lines(yhat~x,col="red",lwd=3)
```



$\hat{\beta}_0$: when length of the pregnancy is 0, the baby weighs 571.16 ounces on average. Overall, all of the estimates are significant due to the small p-values, so adding higher-order polynomial terms provide significant result. The R-squared value of 0.26 means that 26% of the variability in baby's weight can be explained by this model. The plot shows that at after around day 150, the weight of the baby goes up (at an increasing rate) until around day 280, until it levels off at around day 300 onwards and decreases at around day 320.

(c) Use **Model 2** to estimate the probability that a baby will weigh less than 7 pounds (112 ounces) when born on day 280.

The probability that a baby will weigh less than 7 pounds when born on day 280 is 0.218. We got the standard error from the model output in part (b), and we used the formula $t^* = \frac{fit-observed}{SE} = \frac{123.96-112}{15.33} = 0.78$. We then used the `pt` function to get the probability of getting a value more extreme than t^* from the t distribution.

```
# predict
new.data=data.frame(pregnancylength=280)
bound <- predict(mod2, new.data, interval="prediction")
bound
```

```
##        fit      lwr      upr
## 1 123.9578 93.90436 154.0113
```

```
# calculate probability
SE = 15.33
t_star = (bound[1]-112)/SE
1-pt(t_star, df=9061)
```

```
## [1] 0.2176975
```

(d) It is of medical interest to determine at what gestational age a developing fetus is gaining weight the fastest. Use **Model 2** to estimate this *period of fastest growth*.

From model 2, we know that the line of best fit is

$$\hat{weight} = 517 - 7.86 \cdot length + 3.65(10)^{-2} \cdot length^2 - 5.02(10)^{-5} \cdot length^3$$

We take the 2nd derivative with respect to length to find the period of fastest growth

$$2(3.65)(10)^{-2} - 6(5.02)(10)^{-5} \cdot length = 0$$

Solving for length gives us $length = 242.36$, meaning that the period of fastest growth is estimated as day 242 according to model 2.

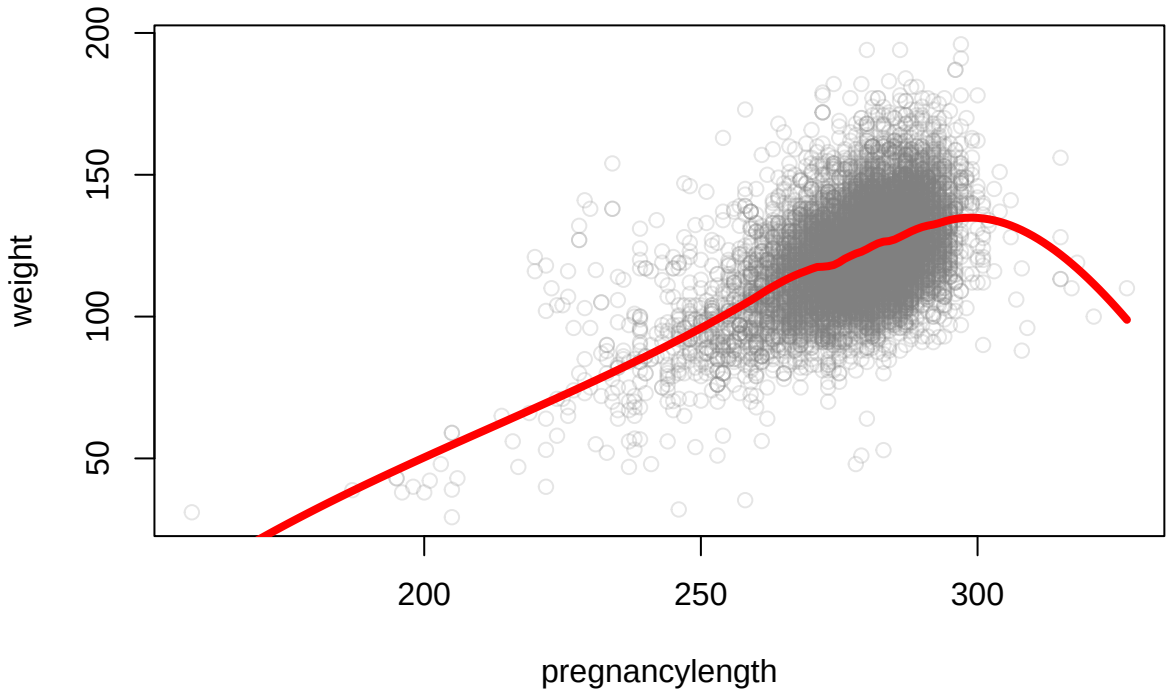
(e) Fit a LOESS model (call this **Model 3**) to predict weight from `pregnancylength` (use a smaller span of 0.3). Provide a visual to support the results of the model. How does this model compare to **Model 2** in its prediction accuracy?

The LOESS model and the 3rd degree polynomial model has similar R2 values (both approximately 0.26, the former has slightly higher R2 value). This means that within the range, the two models have similar prediction accuracy. But considering all the possible values (including outside the range), the LOESS model might have higher accuracy because it captures the true trend of baby's weight: it increases as the pregnancy progresses and perhaps plateaus/decreases if the pregnancy is abnormally long. On the other hand, the polynomial regression model has a weird/unrealistic curve that starts high and decreases over time before increasing at around day 150 (in reality, babies born prematurely do not weigh that much, and baby's weight increases over time)

```
mod3 <- loess(weight~pregnancylength, pregnancy, span=0.3)
summary(mod3)
```

```
## Call:
## loess(formula = weight ~ pregnancylength, data = pregnancy, span = 0.3)
##
## Number of Observations: 9065
## Equivalent Number of Parameters: 12.25
## Residual Standard Error: 15.32
## Trace of smoother matrix: 13.53  (exact)
##
## Control settings:
##   span     :  0.3
##   degree   :  2
##   family   :  gaussian
##   surface  :  interpolate      cell = 0.2
##   normalize:  TRUE
##  parametric:  FALSE
## drop.square:  FALSE
```

```
x=min(pregnancy$pregnancylength):max(pregnancy$pregnancylength)
yhat=predict(mod3, new=data.frame(pregnancylength=x))

plot(weight~pregnancylength, pregnancy, col=rgb(0.5, 0.5, 0.5, 0.2))
lines(yhat~x,col="red",lwd=4)
```




```
# calculate r2
# Predict the values using the model
```

```r
predicted_values <- predict(mod3)

# calculate the total sum of squares (TSS)
tss <- sum((pregnancy$weight - mean(pregnancy$weight))^2)

# calculate the residual sum of squares (RSS)
rss <- sum((pregnancy$weight - predicted_values)^2)

# calculate R2 of model 3
1 - (rss / tss)
```

```
## [1] 0.2649948
```

```r
# r2 of model 2
0.2627
```

```
## [1] 0.2627
```

Problem 3.

In this problem, we will attempt to investigate whether the COVID-19 related restrictions imposed by the government had any effect on the reporting of criminal activity in the Boston Police Department (BPD). We will be using the same combined dataset from last time (now named 'bpd.csv') that includes the number of daily incident reports filed (`count`) and various weather indicators on those days (`maxtemp` is the only weather variable we will use in this problem). Note: we also used these data in Pset 3.

Note: a state of emergency was declared in Massachusetts on March 10, 2020, and restrictions on non-essential businesses, schools, and MBTA service were mainly put into effect on March 17,2020 (see this City of Boston article for the timeline).

The R chunk below reads in the data and includes some code to create a variable called `dayinyear` in the `bpd` data frame that counts the number of days into the year, starting with 0 for Jan 1 (similar to what was done on the previous pset).

```
bpd = read.csv('data/bpd.csv')

jan1_19 = as.Date("1/1/19",format="%m/%d/%y")
jan1_20 = as.Date("1/1/20",format="%m/%d/%y")
jan1_21 = as.Date("1/1/21",format="%m/%d/%y")

bpd$dayinyear = as.Date(bpd$date,format="%m/%d/%y") - jan1_19
bpd$dayinyear[bpd$year==2020] =
  as.Date(bpd$date,format="%m/%d/%y")[bpd$year==2020] - jan1_20
bpd$dayinyear[bpd$year==2021] =
  as.Date(bpd$date,format="%m/%d/%y")[bpd$year==2021] - jan1_21
```

(a) Create a binary/dummy variable (call it `restrictions`) to indicate whether that day falls under the time period of state of emergency or restricted business operations in the city of Boston (all dates between and including March 10, 2020 and Friday, May 28, 2020). How many days fall in this time period in the data set?

80 days in the data set fall in this time period.

```
start = as.Date("03/10/2020", format="%m/%d/%y") - jan1_20
end = as.Date("05/28/2020", format="%m/%d/%y") - jan1_20

bpd$restrictions <- 1*(bpd$dayinyear >= start & bpd$dayinyear <= end & bpd$year == 2020)

sum(bpd$dayinyear >= start & bpd$dayinyear <= end & bpd$year == 2020)
```

```
## [1] 80
```

(b) Calculate the mean number of daily incident reports filed by the BPD during the restriction orders in Boston. Separately calculate the mean number of daily incident reports for a comparison group with the same calendar dates in the pre-pandemic portion of the data. Use these two groups to calculate a reasonable 95% confidence interval for the effect of COVID-19 restrictions on the reporting of crime in the BPD (based on a simple 2-group comparison method and not linear regression).

```
bpd20 <- bpd[bpd$year == 2020 & bpd$dayinyear >= start & bpd$dayinyear <= end, c("year", "count")]
bpd19 <- bpd[bpd$year == 2019 & bpd$dayinyear >= start & bpd$dayinyear <= end, c("year", "count")]

t.test(count~year, rbind(bpd20, bpd19))
```

```
##
##  Welch Two Sample t-test
##
## data:  count by year
## t = 21.694, df = 155.74, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 2019 and group 2020 is not equal to 0
## 95 percent confidence interval:
##   95.96207 115.18793
## sample estimates:
## mean in group 2019 mean in group 2020
##            264.700            159.125
```

The mean of daily incident reports during restriction period in 2020 is 159.125, and that value for 2019 is 264.7. Conducting the 2-sample t test, we get a 95% CI for the effect of restrictions on reporting of crime of (95.96, 115.19). This means that COVID restrictions tend to correlate with a decrease of between 95.96 and 115.19 daily crimes reported.

(c) Fit a linear regression model to predict `count` from `maxtemp` and `restrictions` (call it **model1**), and print out the `summary` results. Briefly interpret the coefficient estimates and use this model to estimate the effect of COVID-19 restrictions on the reporting of crime in the BPD (with 95% confidence).

```
model1 <- lm(count ~ maxtemp + restrictions, bpd)
summary(model1)
```

```
##
## Call:
## lm(formula = count ~ maxtemp + restrictions, data = bpd)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -146.34  -31.89   -6.60   31.44  120.94
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  181.60815    4.58837  39.580   <2e-16 ***
## maxtemp        0.70066    0.07195   9.739   <2e-16 ***
## restrictions -60.67812    4.92423 -12.322   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42.21 on 1093 degrees of freedom
## Multiple R-squared:  0.1993, Adjusted R-squared:  0.1979
## F-statistic: 136.1 on 2 and 1093 DF,  p-value: < 2.2e-16
```

```
confint(model1, "restrictions", type="confidence")
```

```
##                   2.5 %   97.5 %
## restrictions -70.34014 -51.0161
```

$\hat{\beta}_0$: when `maxtemp` is 0F and when there is no restriction, the expected number of daily reported crimes is 181.61 in Boston.

$\hat{\beta}_1$: the expected number of daily reported crimes increases by 0.7 as `maxtemp` increases by 1F, if we consider the same type of day (restriction vs non-restriction).

$\hat{\beta}_2$: `maxtemp` is held constant, the difference between expected number of daily reported crimes between restriction and non-restriction day is 60.68 (restriction days have fewer crimes).

All the coefficients are significant according to the model output.

The 95% confidence interval for the effect of COVID-19 restrictions is (-70.34, -51.02). Since 0 is not included in the interval, this means that restriction rules decrease the number of daily crimes in Boston.

(d) Fit a linear regression model to predict `count` from `maxtemp`, `restrictions`, `dayinyear` and all 2-way interactions between these 3 predictors (call it **model2**), and print out the `summary` results. Interpret what this model says about the relationship between crime reporting in the BPD and COVID-19 restrictions. Compute an estimate and 95% CI for the effect of restrictions on the 0th day of the year, assuming a `maxtemp` of 0 degrees. Also estimate `count` (and provide a 95% CI) on the 91st day of the year in 2020, assuming the temperature was 50 degrees. Do the same for 2019 and compare the difference.

```
model2 <- lm(count~(maxtemp + restrictions + dayinyear)^2, bpd)
summary(model2)
```

```
##
## Call:
## lm(formula = count ~ (maxtemp + restrictions + dayinyear)^2,
##     data = bpd)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -146.051  -31.478   -6.595   30.792  118.593
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)              1.925e+02  9.342e+00  20.609  < 2e-16 ***
## maxtemp                  4.720e-01  1.929e-01   2.447  0.01457 *
## restrictions            -7.807e+01  2.719e+01  -2.872  0.00416 **
## dayinyear               -5.861e-02  4.737e-02  -1.237  0.21623
## maxtemp:restrictions     9.130e-01  5.327e-01   1.714  0.08683 .
## maxtemp:dayinyear        1.174e-03  9.534e-04   1.232  0.21828
## restrictions:dayinyear -2.911e-01  2.443e-01  -1.192  0.23371
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42.21 on 1089 degrees of freedom
## Multiple R-squared:  0.2024, Adjusted R-squared:  0.198
## F-statistic: 46.06 on 6 and 1089 DF,  p-value: < 2.2e-16
```

```
new.data1 <- data.frame(dayinyear=as.difftime(91, unit="days"), maxtemp=50, restrictions=0)
new.data2 <- data.frame(dayinyear=as.difftime(91, unit="days"), maxtemp=50, restrictions=1)

predict(model2, new.data1, interval="prediction")
```

```
##        fit      lwr      upr
## 1 216.1435 133.2342 299.0528
```

```
predict(model2, new.data2, interval="prediction")
```

```
##        fit      lwr      upr
## 1 157.2327 73.60371 240.8617
```

From the model output, we know that considering two days, one with restriction and one without, with the same `maxtemp` of a and the same day in year b, the difference in daily crime count is $-78.07+0.913a-2.911b$.

The model output shows that the effect of restrictions on the 0th day of the year, assuming 0 degrees, is -0.7807 (this is the same as considering the 2-sample t test for restriction=1 vs restrictions=0). This estimate is significant because the p-value of 0.0416 is smaller than 0.05.

95% CI of `count` on the 91st day of the year in 2020, assuming the max temperature was 50 degrees is (73.60, 240.86), and the same 95% CI for 2019 is (133.23, 299.05). These two intervals are overlapping, but the interval for 2019 (non-restriction year) is higher than the interval for 2020 (restriction year), so it is likely that restriction is correlated with a decrease in daily count of crime.

(e) Perform a formal hypothesis test to determine whether **model2** performs significantly better at predicting `count` than **model1**.

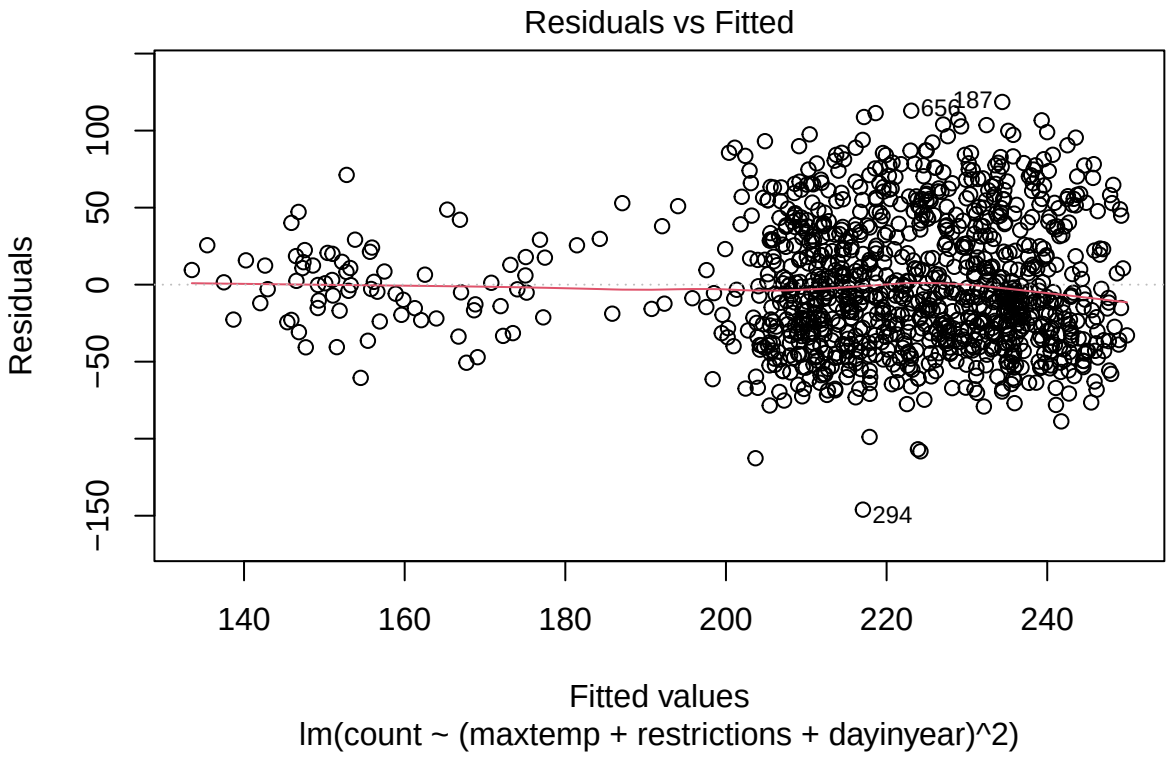
```
anova(model1, model2)
```

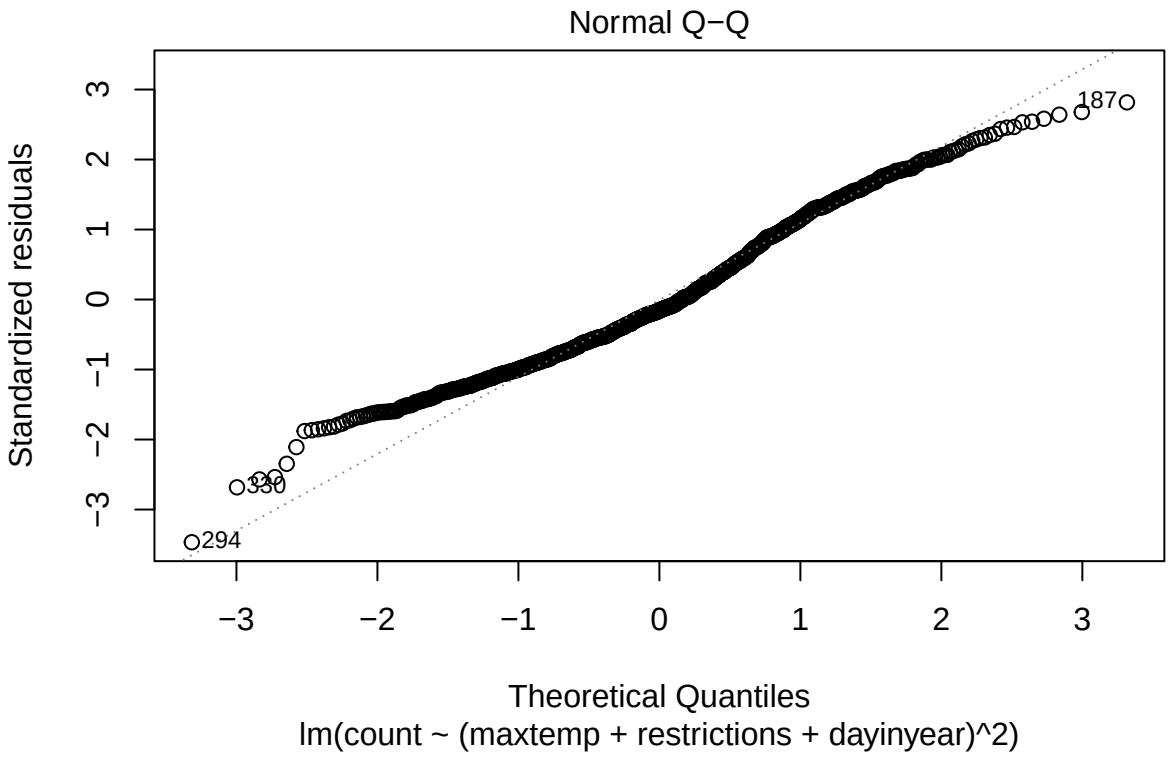
```
## Analysis of Variance Table
##
## Model 1: count ~ maxtemp + restrictions
## Model 2: count ~ (maxtemp + restrictions + dayinyear)^2
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1   1093 1947502
## 2   1089 1940064  4    7437.3 1.0437 0.3834
```

We conducted an SSE F-test to compare the 2 models because they have similar structures and one is nested within the other. The null hypothesis is that adding more terms to the original model does not add predictive power, and the alternative hypothesis is that at least one term, when added to the original model, increases the predictive power. The F-statistic is 1.044, with df = 4, 1089. The p-value is 0.3834, so we can retain the null hypothesis and conclude that model 2 does not perform significantly better than model 1.

(f) Investigate the assumptions for **model2**. Be sure to include and reference useful visuals.

```
plot(model2, which=c(1,2))
```


Residuals vs Fitted
Residuals
100
50
0
−50
−150
187
655
294
140
160
180
200
220
240
Fitted values
lm(count ~ (maxtemp + restrictions + dayinyear)^2)



The assumptions are: (1) Indepedence: the observations are not independent because the daily count of crime on neighboring days during COVID restriction would be more similar to each other than the daily count of crime during non-restriction period. (2) Linearity: this assumption is met because the points are similarly above the horizontal line and below the horizontal line in the 1st plot. (3) Normality: this assumption is met because the standardized residuals more or less follow the theoretical quantiles in the QQ plot (and we have a large dataset so we can apply the CLT). (4) Homoskedasticity: this seems to be violated because the variance of residuals increases as the fitted values increase.

(g) Determine which 4 dates **model2** did the worst job at predicting `count`. Can you think of a reason why any of these dates do not follow the relationships in this model? (all 4 are explainable with a little Google searching)

```
bpd$resids <- abs(model2$residuals)
bpd %>%
  slice_max(n=4, order_by=resids)
```

```
##       date year month day maxtemp meantemp mintemp dewpoint maxhumid meanhumid
## 1 12/14/21 2021   dec  14      51     44.8      36       26       51      34.5
## 2  11/1/19 2019   nov   1      72     57.2      45       65       81      49.3
## 3  5/15/19 2019   may  15      61     51.4      44       42       89      64.5
## 4 12/25/21 2021   dec  25      36     32.6      28       35       97      91.1
##   minhumid gust meanwind maxpress meanpress minpress precip count dayinyear
## 1       23   15     10.8     30.6      30.4     30.1   0.00    71  347 days
## 2       29   35     21.6     30.2      29.8     29.4   0.07   353  304 days
## 3       38   16      8.8     29.8      29.8     29.7   0.04   336  134 days
```

```
## 4       82   13      6.1     29.9      29.6     29.5   0.00    91  358 days
##   restrictions   resids
## 1            0 146.0514
## 2            0 118.5935
## 3            0 112.9283
## 4            0 112.6782
```

12/14/21 and 12/25/21 were Christmas holidays, and crime rates tend to go up during holiday time due to a variety of reasons such as increased alcohol asumption and heightened economic stress. There was a Knicks vs Celtics game on 11/1/19, so the high frequency of crime in Boston might have been related to this game. On 5/15/19, a police captain was placed on leave, so this might have affected the relationships in the model.

(h) Write a 200-300 word summary addressing whether there is evidence that COVID-19 reduced the amount of crime in Boston. Be sure to reference the results above (specifically, which approach you think was most reasonable) and mention any biases or confounders that may be present in this relationship.

Model 1 is simpler and only includes 2 predictors, whereas model 2 is more complex and includes all 2-way interactions among 3 predictors. In the model 2's output, we see that all of the estimates for coefficients that involve `dayinyear` are insignificant (due to large p-values). Also, the ESS F-test conducted in part (e) leads us to a similar conclusions: adding more terms to the model (`dayinyear` and 2-way interaction terms) do not improve the model's predictive power. Therefore, I think model 1 is the more reasonable approach for 2 reasons: it is simpler, and a more complex model doesn't necessarily improve the predictive power. Based on model 1's output, the estimate associated with `restrictions` is -60.678, meaning that holding temperature constant, on average, a day with restriction has 61 fewer reported crimes than a day without restriction. This is significant evidence that COVID-19 restriction is associated with fewer reported crimes. We can't draw a causal link because this is not a randomized control trial. Because of its simplicity, model 1 might fail to capture certain confounders; for example, during COVID restrictions period, the government might have more measures to alleviate economic pressures, leading to fewer reported crimes.

Problem 4.

Perform a simulation study (with 1,000 iterations) where the data are **generated** from the following sin function:

$$Y_i = \sin(X_i) + \varepsilon_i$$

where $\varepsilon_i \sim N(0, \sigma^2 = 0.1^2)$ and independent, and X_i are sampled independently from a Unif$(a = 0, b = 6)$, for $n = 50$ observations.

For each iteration, fit 4 different polynomial models (use the raw form in R): (i) 3^{rd} order, (ii) 5^{th} order, (iii) 7^{th} order, and (iv) $= 9^{th}$ order. Save the β_1 coefficient (linear term) estimates for each of the 4 models for each of the 1,000 iteration (presumably a 1000x4 matrix) and either separately save all 10 β coefficient estimates for the 9^{th} order or the model objects themselves (in a list).

Evaluate which model is *best* in each iteration two ways: (i) based on sequential ESS F-tests (you will perform 3 of them in each iteration) and (ii) out-of-sample mean squared error (based on a single test set of $n_{test} = 1000$ generated from the same data generating process as the regular $n = 50$ set of observations... this does not need to be recreated in each iteration).

```r
# params
set.seed(139)
nsims <- 1000
ntest <- 1000

# beta1 estimates
beta1 <- data.frame(m1 = as.numeric(),
                    m2 = as.numeric(),
                    m3 = as.numeric(),
                    m4 = as.numeric())
m4_list <- list()

# best model
best_model_ess <- rep(NULL, nsims)
best_model_mse <- rep(NULL, nsims)

# test data
x_test <- runif(ntest, 0, 6)
epsilon_test <- rnorm(ntest,0,0.1)
y_test <- sin(x_test) + epsilon_test

# simulate
for(i in 1:nsims){

  n <- 50
  x <- runif(n, 0, 6)
  epsilon <- rnorm(n,0,0.1)
  y <- sin(x) + epsilon

  # create lm
  m1 <- lm(y~poly(x,3,raw=T))
  m2 <- lm(y~poly(x,5,raw=T))
  m3 <- lm(y~poly(x,7,raw=T))
  m4 <- lm(y~poly(x,9,raw=T))

  # save beta1 estimates
  beta1[i,1] <- coefficients(m1)[2]
```

```
  beta1[i,2] <- coefficients(m2)[2]
  beta1[i,3] <- coefficients(m3)[2]
  beta1[i,4] <- coefficients(m4)[2]

  # save m4
  m4_list[[i]] <- m4

  # determine best model using out-of-sample MSE
  yhat1 <- predict(m1, new=data.frame(x=x_test))
  yhat2 <- predict(m2, new=data.frame(x=x_test))
  yhat3 <- predict(m3, new=data.frame(x=x_test))
  yhat4 <- predict(m4, new=data.frame(x=x_test))

  mse <- c(mean((y_test-yhat1)^2),
           mean((y_test-yhat2)^2),
           mean((y_test-yhat3)^2),
           mean((y_test-yhat4)^2))

  best_model_mse[i] <- which.min(mse)

  # sequential ESS F tests
  test1 <- anova(m1, m2)[2,6]
  test2 <- anova(m2, m3)[2,6]
  test3 <- anova(m3, m4)[2,6]

  # determine best model using ESS F tests
  if(test1 < 0.05 & test2 < 0.05 & test3 < 0.05){
    best_model_ess[i] = 4
  } else if(test1 < 0.05 & test2 < 0.05){
    best_model_ess[i] = 3
  } else if(test1 < 0.05){
    best_model_ess[i] = 2
  } else(best_model_ess[i] = 1)
}

table(best_model_ess)
```

```
## best_model_ess
##   1   2   3   4
##  93 857  49   1
```

```
table(best_model_mse)
```

```
## best_model_mse
##   1   2   3   4
##  15 848 121  16
```

(a) Based on the ESS F-tests, how often is each of the 4 models considered the best? Based on out-of-sample mean squared error?

Based on the ESS F-tests, model1 performs the best 9.3% of the time, model2 performs the best 85.7% of the time, model3 4.8% of the time, and model4 0.02% of the time.

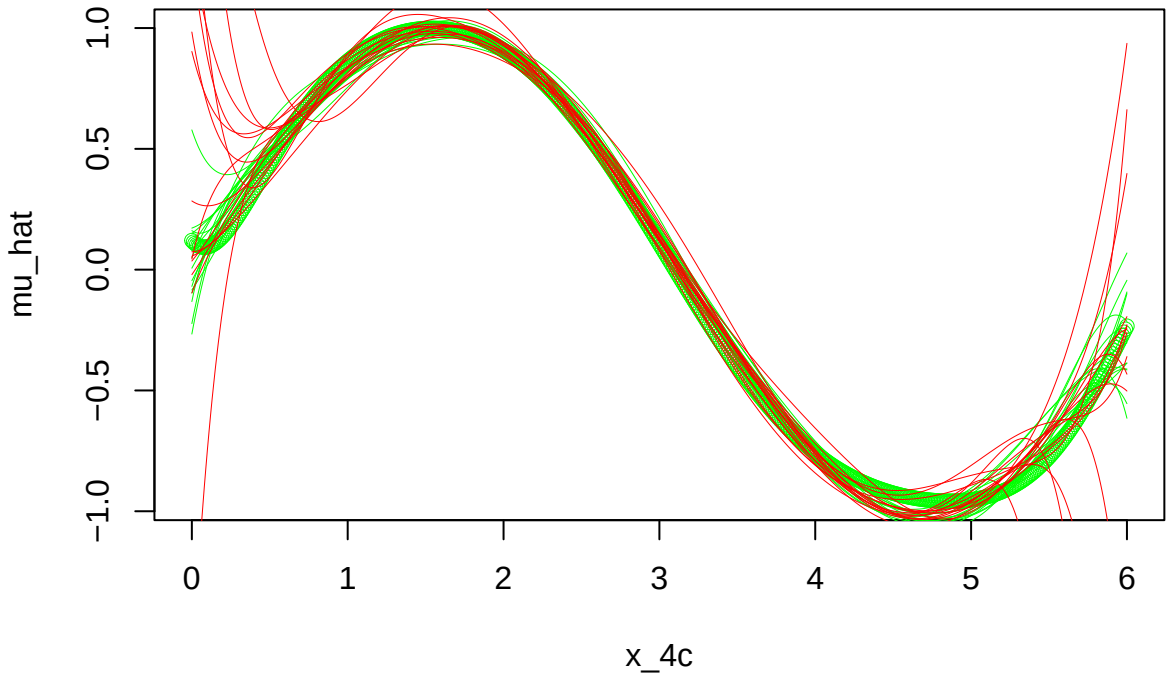
Based on the MSE method, model1 performs the best 1.5% of the time, model2 performs the best 84.8% of the time, model3 12.1% of the time, and model4 1.6% of the time.

(b) Which metric is more conservative when it comes to overfitting? How do you know?

The ESS F-test is more conservative to overfitting because it tends to fit simple models (lower-order polynomials) more of the time, whereas the MSE method determines 3rd and 4th order polynomials as the best models more often.

(c) Plot 10+ $\hat{\mu}_Y$ curves (the predicted curve) based on the estimated 9^{th} order polynomial model (for 10+ iterations): with at least 5 curves for when 9^{th} order model wins and at least 5 curves for when it does not win (based on out-of-sample mean square error). Be sure to color code these curves based on when this model wins vs. when it does not win. Interpret this plot: what does this say about how overfitting affects out-of-sample mean square error?

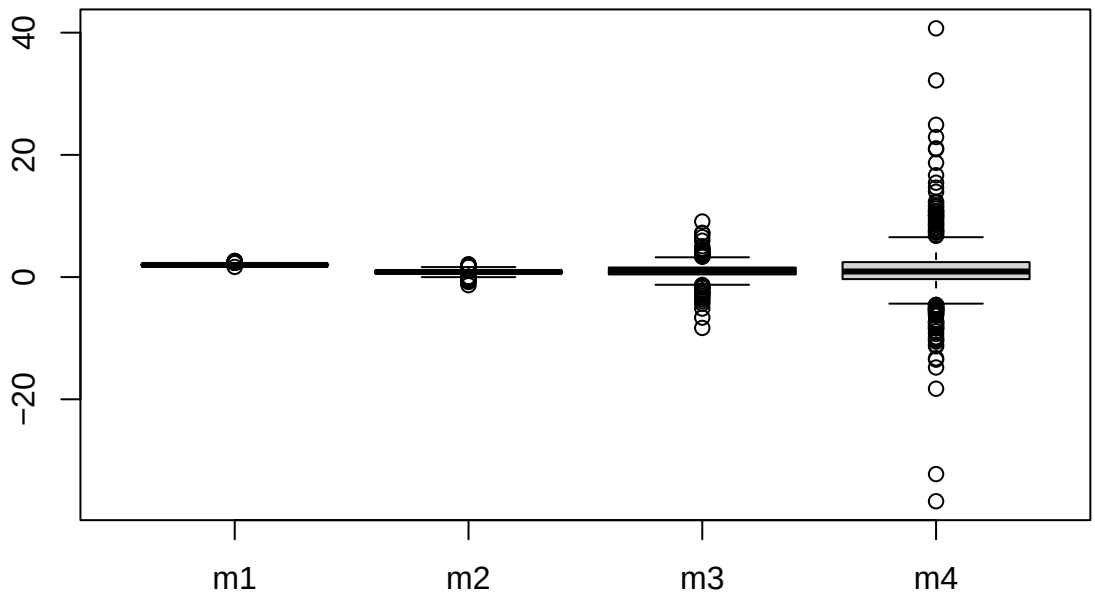
```r
# get model ids
win_id <- which(best_model_mse == 4)
lose_id <- which(best_model_mse == 1)

# plot
x_4c <- seq(0, 6, 0.01)
mu_hat <- predict(m4_list[[137]], new=data.frame(x=x_4c))
plot(mu_hat~x_4c, col="green", lwd=0.5)

for(i in win_id){
  mu_hat <- predict(m4_list[[i]], new=data.frame(x=x_4c))
  lines(mu_hat~x_4c, col="green", lwd=0.5)
}
for(i in lose_id){
  mu_hat <- predict(m4_list[[i]], new=data.frame(x=x_4c))
  lines(mu_hat~x_4c, col="red", lwd=0.5)
}
```



We expect the lines to follow the $sin(x)$ line with small variation. When model 4 wins, the plot behaves as expected (see the green lines), but when model 4 loses, the plot behaves weirdly (see the red lines and how they go way off near the boundaries). The training data includes more points in the middle of the range, so lost model 4 tends to behave badly outside of the range or at the boundaries. In general, this shows that overfitting (considering model 4 when it loses) leads to higher out-of-sample MSE.

(d) Provide the boxplots of $\hat{\beta}_1$ estimates in each of the 4 models (should be a side-by-side boxplot with 4 boxplots based on 1000 estimates each). Interpret this plot in context of this situation. What does this illustrate? Why is this not surprising?

```
boxplot(beta1)
```



As we fit more complex models, there are more outliers in the estimates of β_1. This illustrates that when we fit higher-order polynomials, we tend to get really varied estimates for the same coefficient. This is not surprising because the variance of all betas estimates is $\sigma^2(X^TX)^{-1}$ and it is large when we have lots of not-so-meaningful predictors. We estimate $sigma^2$ by calculating $\hat{\sigma}^2 = \frac{\sum(Y_i - \hat{Y})^2}{n-(p+1)}$. When a complex model doesn't improve $\sum(Y_i - \hat{Y})^2$, its $\hat{\sigma}^2$ increases because $n - (p+1)$ becomes smaller with more predictors in the model.