BST 232: Homework 2

Phuc Vu

09/23/24

Question 1 Variance-covariance matrices are symmetric positive semi-definite

The variance-covariance matrix of a random vector is always symmetric and positive semi- definite (PSD). There are various equivalent definitions of PSD, but we will use the one below: Definition: A $p \times p$ symmetric matrix $\mathbf{A}$ is PSD $\mathbf{u'Au} \geq 0 \forall \mathbf{u} \in \mathbb{R}^p$. Using this definition, show that for a zero-mean random vector $\mathbf{W} \in \mathbb{R}^p$, its covariance matrix is PSD. Recall that $Cov(\mathbf{W}) = E((\mathbf{W} - E(\mathbf{W}))(\mathbf{W} - E(\mathbf{W}))')$.

Proof:

$\forall \mathbf{u} \in \mathbb{R}^p$ and zero-mean random vector $\mathbf{W} \in \mathbb{R}^p$, let $u_i = \mathbf{u}_{i1}$ and $w_i = (\mathbf{W} - E(\mathbf{W}))_{i1}$ We have:

$$\begin{aligned}
\mathbf{u}'Cov(\mathbf{W})u &= \mathbf{u}'E((\mathbf{W} - E(\mathbf{W}))(\mathbf{W} - E(\mathbf{W}))')u \\
&= E(u'(\mathbf{W} - E(\mathbf{W}))(\mathbf{W} - E(\mathbf{W}))'u) \\
&= E(u'(\mathbf{W} - E(\mathbf{W}))(u'(\mathbf{W} - E(\mathbf{W})))') \\
&= E((\sum_{i=1}^{p} u_i w_i)(\sum_{i=1}^{p} u_i w_i)') \\
&= E((\sum_{i=1}^{p} u_i w_i)^2) \geq 0 \forall \mathbf{u} \in \mathbb{R}^p
\end{aligned}$$

Therefore, the variance-covariance matrix of a random vector is always positive semi- definite.

Question 2: Orthogonal columns of X

Two n-length vectors $\mathbf{b}$ and $\mathbf{c}$ are orthogonal if $\mathbf{b'c} = 0$. Show that if the columns of the design matrix $\mathbf{X}$ are mutually orthogonal then the elements of $\hat{\boldsymbol{\beta}}_{OLS}$ are the same as the OLS estimates from separate simple linear regressions of each column of $\mathbf{X}$ onto $\mathbf{y}$.

Proof

Let $\mathbf{x_1}, \ldots, \mathbf{x_{p+1}}$ denote the columns of $\mathbf{X}$. Note that $\mathbf{x_1} = (1, ..., 1)'$. We have $\mathbf{x_i}'\mathbf{x_j} = 0 \forall i \neq j$.

Let $\mathbf{C} = \mathbf{X'X} \in \mathbb{R}^{(p+1)\times(p+1)}$ then:

$$\mathbf{C}_{ij} = \mathbf{x_i'x_j} = \begin{cases} 0 \text{ if } i \neq j \\ \mathbf{x_i'x_i} \text{ if } i = j \end{cases}$$

Therefore: $\mathbf{C} = diag(\mathbf{x_1'x_1}, \ldots, \mathbf{x_{p+1}'x_{p+1}}) \Rightarrow \mathbf{C}^{-1} = diag(1/\mathbf{x_1'x_1}, \ldots, 1/(\mathbf{x_{p+1}'x_{p+1}}))$

We have:

$$
\begin{aligned}
\hat{\boldsymbol{\beta}}_{OLS} &= [\mathbf{X}'\mathbf{X}]^{-\mathbf{1}}\mathbf{X}'\mathbf{y} \\
&= \mathbf{C}^{-1}\mathbf{X}'\mathbf{y} \\
&= \mathbf{C}^{-1}(\mathbf{x_1'y}, \ldots, \mathbf{x'_{p+1}y}) \\
&= diag(1/\mathbf{x_1'x_1}, \ldots, 1/(\mathbf{x'_{p+1}x_{p+1}}))(\mathbf{x_1'y}, \ldots, \mathbf{x'_{p+1}y}) \\
&= (\frac{\mathbf{x_1'y}}{\mathbf{x_1'x_1}}, \frac{\mathbf{x_2'y}}{\mathbf{x_2'x_2}}, \ldots, \frac{\mathbf{x'_{p+1}y}}{\mathbf{x'_{p+1}x_{p+1}}})'
\end{aligned}
$$

For any given column $\mathbf{x_j}, j \geq 2$, we have $n\bar{x}_j = x_{j1} + ... + x_{jn} = \mathbf{x_j'x_1} = 0$, therefore $\bar{x}_j = 0 \forall j \geq 2$ Also, in simple linear regression, the coefficient regression is

$$
\begin{aligned}
\hat{\beta}_j &= \frac{\sum_{i=1}^n (x_{ji} - \bar{x}_j)(y_i - \bar{y})}{\sum_{i=1}^n (x_{ji} - \bar{x}_j)^2} \\
&= \frac{\sum_{i=1}^n x_{ji}(y_i - \bar{y})}{\sum_{i=1}^n x_{ji}^2} \text{ (because } \bar{x}_j = 0) \\
&= \frac{\sum_{i=1}^n x_{ji}y_i - x_{ji}\bar{y}}{\sum_{i=1}^n x_{ji}^2} \\
&= \frac{(\sum_{i=1}^n x_{ji}y_i) - n\bar{x}_j\bar{y}}{\sum_{i=1}^n x_{ji}^2} \\
&= \frac{(\sum_{i=1}^n x_{ji}y_i)}{\sum_{i=1}^n x_{ji}^2} \text{ (because } \bar{x}_j = 0) \\
&= \frac{\mathbf{x_j'y}}{\mathbf{x_j'x_j}}
\end{aligned}
$$

This is also the j^{th} entry of the $\hat{\boldsymbol{\beta}}_{OLS}$. Therefore, if the columns of the design matrix $\mathbf{X}$ are mutually orthogonal then the elements of $\hat{\beta}_{OLS}$ are the same as the OLS estimates from separate simple linear regressions of each column of $\mathbf{X}$ onto $\mathbf{y}$.

Question 3: Fitting and interpreting linear models. Consider assessing the association between LDL cholesterol level and body mass index (BMI) in the HERS data. In this problem, we will use log(LDL) as the outcome variable (we will explore how we made this decision when we get to regression diagnostics). Applying the model to all of the HERS data provided on Canvas (hers.csv in the Files >datasets folder):

a. Fit the simple linear model relating log(LDL) and BMI, and report and interpret the estimated slope $\hat{\beta}_1$. Is there strong evidence of a linear association between log(LDL) and BMI?

```
##
## Call:
## lm(formula = log(LDL) ~ BMI, data = her_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.33634 -0.15903  0.00839  0.16898  1.00656
##
## Coefficients:
```

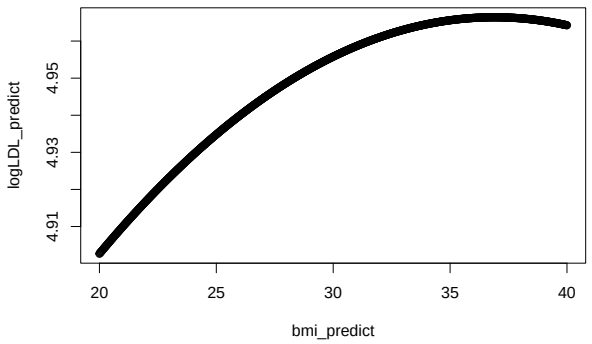
```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.8654563  0.0258892 187.934  < 2e-16 ***
## BMI         0.0027534  0.0008896   3.095  0.00199 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2576 on 2745 degrees of freedom
##   (16 observations deleted due to missingness)
## Multiple R-squared:  0.003478,   Adjusted R-squared:  0.003115
## F-statistic:  9.58 on 1 and 2745 DF,  p-value: 0.001987
```

$\hat{\beta} = 0.0028$

Interpretation: For every unit increase in BMI, the value of the log (LDL) is expected to increase by 0.0028. There is significant evidence of a linear association between lol(LDL) and BMI with p-value of less than 0.002.

b. Now fit a model for log(LDL) that allows for a quadratic relationship with BMI. Visualize the estimated association between BMI and log(LDL) for BMI values in the range [20,40]. Interpret the results.

```
##
## Call:
## lm(formula = log(LDL) ~ BMI + I(BMI^2), data = her_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.34219 -0.16068  0.00949  0.16869  1.00846
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.6624548  0.1015347  45.920   <2e-16 ***
## BMI          0.0164763  0.0066965   2.460   0.0139 *
## I(BMI^2)    -0.0002233  0.0001080  -2.068   0.0388 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2574 on 2744 degrees of freedom
##   (16 observations deleted due to missingness)
## Multiple R-squared:  0.005028,   Adjusted R-squared:  0.004303
## F-statistic: 6.933 on 2 and 2744 DF,  p-value: 0.0009923
```



At value x of BMI, an increase unit in BMI is associated with a change of $0.016 - 2 * 0.0002x + 0.0002^2$ in log(LDL). At small value, an increase unit in BMI is associated with a higher expected value of log(LDL). Nevertheless, at a high value (>35), an increase in BMI is associated with a lower expected value of log(LDL).

c. Construct a categorical version of the BMI variable following the CDC's BMI categories shown below. Specify and fit a linear model to assess the association between log(LDL) and the BMI categories, and interpret the results.

BMI Category	BMI Range (kg/m^2)
Underweight	Less than 18.5
Healthy Weight	18.5 to less than 25
Overweight	25 to less than 30
Obesity	30 or greater

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	4.9199687	0.0526019	93.5322225	0.0000000
BMI_cdcHealthy Weight	0.0032051	0.0534458	0.0599683	0.9521853
BMI_cdcOverweight	0.0248073	0.0532019	0.4662860	0.6410478
BMI_cdcObesity	0.0406855	0.0532727	0.7637223	0.4450984

Compared to people who are underweight BMI, people with Healthy weight BMI are expected to have a 0.003 higher lol(LDL).

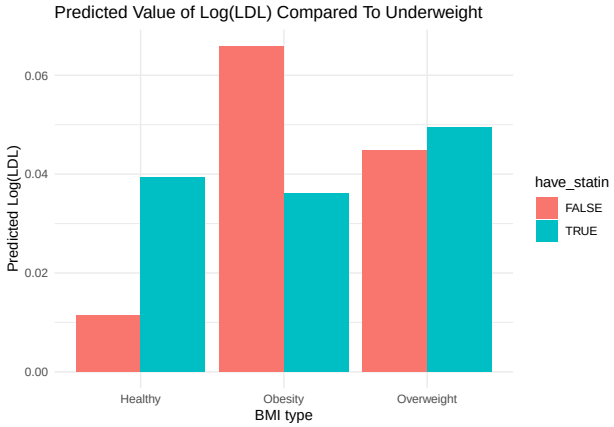
Compared to people who are underweight BMI, people with Overweight BMI are expected to have a 0.025 higher lol(LDL).

Compared to people who are underweight BMI, people with Obesity BMI are expected to have a 0.04 higher lol(LDL).

d. Fit a model for log(LDL) that allows for the association between the categorical BMI variable and log(LDL) to differ for those who do and do not take statins. Interpret the results (you may use visualizations to help).

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	4.9449144	0.0576059	85.8404778	0.0000000
BMI_cdcHealthy Weight	0.0114411	0.0587491	0.1947455	0.8456066
BMI_cdcOverweight	0.0448498	0.0584587	0.7672059	0.4430252
BMI_cdcObesity	0.0659436	0.0584805	1.1276181	0.2595800
statinsyes	-0.1197394	0.1262081	-0.9487459	0.3428336
BMI_cdcHealthy Weight:statinsyes	0.0278705	0.1276588	0.2183201	0.8271960

	Estimate	Std. Error	t value	Pr(>|t|)
BMI_cdcOverweight:statinsyes	0.0046839	0.1272070	0.0368213	0.9706302
BMI_cdcObesity:statinsyes	-0.0297528	0.1274002	-0.2335382	0.8153610



People with healthy BMI is expected to have higher 0.01 log(LDL) compared to people with underweight BMI among people who do not use statins while that of people who do use statin is 0.04. Similarly, people with Obesity BMI is expected to have higher 0.066 log(LDL) compared to people with underweight BMI among people who do not use statins while that of people who do use statin is 0.036. On the other hand, people with Overweight BMI is expected to have higher 0.044 log(LDL) compared to people with underweight BMI among people who do not use statins while that of people who do use statin is 0.050. Nevertheless, the interaction terms are not significant meaning there is not significant different association between the categorical BMI variable and log(LDL) among for those who do and do not take statins.

Question 4: Investigating the impacts of model misspecification In this problem, we will investigate the implications and impacts of misspecification of the systematic component of the linear model. Suppose interest lies in the relationship between a continuous outcome Y and a single continuous exposure variable x. Consider

the following three models: Model 1 $Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \varepsilon_i$ Model 2 $Y_i = \alpha_0 + \alpha_1 x_i + \varepsilon_i$ Model 3 $Y_i = \gamma_0 + \gamma_1 x_{i2}^* + \ldots + \gamma_{K-1} x_{iK}^* + \varepsilon_i$ where $\varepsilon_i \sim N(0, \sigma^2)$ and $\{x_{i1}^*, \ldots, x_{iK}^*\}$ is a collection of K indicator/dummy variables, each defined by

$$x_{ik} = \begin{cases} 1 \text{ if } x_i \in [c_k, c_{k+1}) \\ 0 \text{ otherwise} \end{cases}$$

for the partition $\{c_1, \ldots, c_{K+1}\}$ of the real line, where $c_1 \equiv -\infty$ and $c_{K+1} \equiv \infty$. Suppose Model 1 is the correct model and yet we fit Model 2 or Model 3. Despite Model 2 being misspecified, the OLS estimate of $\boldsymbol{\alpha}$ converges to some value, say $\boldsymbol{\alpha}^*$. That is, $\hat{\boldsymbol{\alpha}} \to \boldsymbol{\alpha}^*$ as $n \to \infty$. Similarly, despite Model 3 being misspecified, we have $\hat{\boldsymbol{\gamma}} \to \boldsymbol{\gamma}^*$ as $n \to \infty$ for some value $\boldsymbol{\gamma}^*$

(a) We can investigate the 'true' value of

$\boldsymbol{\alpha}^*$ and $\boldsymbol{\gamma}^*$ empirically by simulating a large dataset under the true model and examining the estimates from the misspecified model. Intuitively, we are simulating asymptotia by setting n to be large. For each of the following scenarios, simulate a single sample of size $n = 1,000,000$ from Model 1 with $\sigma^2 = 4$ as the error variance. Then fit Models 2 and 3 and report the values of $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\gamma}}$:

(i) $\boldsymbol{\beta} = (0, 0.2, 0.2, 0.1)$ and $x \sim Normal(0, 1)$

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.1982836	0.0020349	97.44241	0
x_i	0.5003285	0.0020334	246.05096	0

$$\hat{\boldsymbol{\alpha}} = (\alpha_0, \alpha_1)' = (0.1983, 0.5003)$$

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-2.133729	0.0543795	-39.23775	0
relevel(x_i_cat, ref = ref)x2	1.447911	0.0560902	25.81398	0
relevel(x_i_cat, ref = ref)x3	1.954465	0.0546538	35.76083	0
relevel(x_i_cat, ref = ref)x4	2.079942	0.0544883	38.17229	0
relevel(x_i_cat, ref = ref)x5	2.305178	0.0544883	42.30590	0
relevel(x_i_cat, ref = ref)x6	3.097035	0.0546522	56.66805	0
relevel(x_i_cat, ref = ref)x7	4.948074	0.0560718	88.24539	0
relevel(x_i_cat, ref = ref)x8	8.555224	0.0769183	111.22485	0

$$\hat{\boldsymbol{\gamma}} = (\gamma_0, \ldots, \gamma_8)' = (-2.1337, 1.4479, 1.9545, 2.0799, 2.3052, 3.097, 4.9481, 8.5552)$$

(ii) $\boldsymbol{\beta} = (0, 0.2, 0.2, -0.1)$ and $x \sim Normal(0, 1)$ Note: For estimation based on Model 3 use the following partition for x:

$\{c_1, c_2, ..., c_K, c_{K+1}\} = \{-\infty, -3, -2, -1, 0, 1, 2, 3, \infty\}$

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.1983833	0.0020347	97.50183	0
x_i	-0.1021339	0.0020332	-50.23266	0

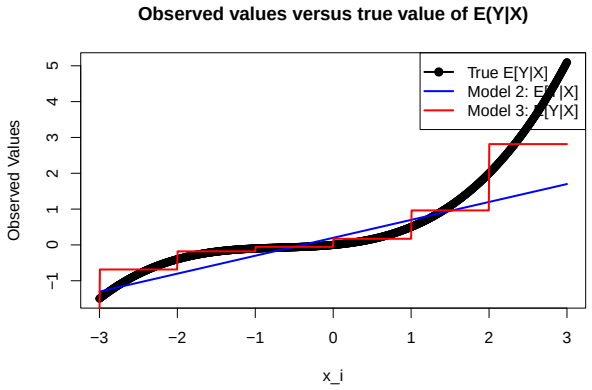
$$\hat{\boldsymbol{\alpha}} = (\alpha_0, \alpha_1)' = (0.1984, -0.1021)$$

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	5.113107	0.0542559	94.24057	0
relevel(x_i_cat, ref = ref)x2	-3.213646	0.0559627	-57.42475	0
relevel(x_i_cat, ref = ref)x3	-4.700456	0.0545296	-86.20011	0
relevel(x_i_cat, ref = ref)x4	-5.124804	0.0543645	-94.26755	0
relevel(x_i_cat, ref = ref)x5	-4.983884	0.0543645	-91.67532	0
relevel(x_i_cat, ref = ref)x6	-4.741936	0.0545280	-86.96330	0
relevel(x_i_cat, ref = ref)x7	-4.871976	0.0559443	-87.08615	0
relevel(x_i_cat, ref = ref)x8	-5.897588	0.0767435	-76.84808	0

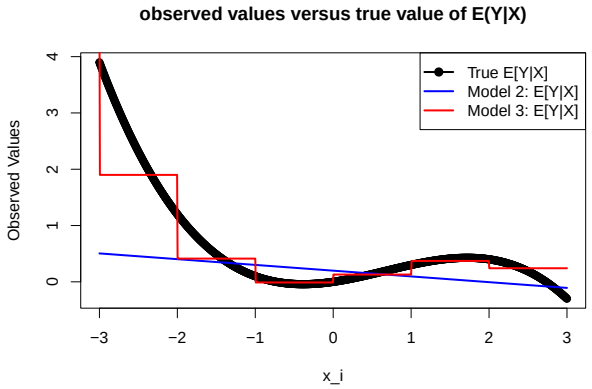
$$\hat{\boldsymbol{\gamma}} = (\gamma_0, \ldots, \gamma_8)' = (5.1131, -3.2136, -4.7005, -5.1248, -4.9839, -4.7419, -4.872, -5.8976)$$

(b) For a range of values of $x \in (-3, 3)$, compute the true values of $E[Y \mid x]$ under each of the following scenarios for Model 1:

(i) $\boldsymbol{\beta} = (0, 0.2, 0.2, 0.1)$



(ii) $\boldsymbol{\beta} = (0, 0.2, 0.2, -0.1)$



Plot the values versus x on separate figures (join the points to provide a smooth representation). Superimpose the values of $\hat{E}[Y \mid x]$ based on the approximate models you obtained in part (a) on the appropriate figure. You should end up with two figures, each with three lines.

(c) Suppose in a real data setting, where you don't know the true form of $E[Y \mid x]$ you fit both models (2) and (3) on your data and find that your plots of $\hat{E}[Y \mid x]$ from the two model fits display quite different trends. What might you conclude?

Since the step model (model 3) allow for flexibility of the model and capture the mean of effect of intervals of x on y, if I find the two models fits display quite different trends, I will add higher order term of x or transform the variable x to allow for more non-linear relations.

Code

```
knitr::opts_chunk$set(
  tidy = T, results = 'hold', echo = FALSE, warning = FALSE, message = FALSE,
  out.width = "50%"
)
```

```r
# This code cell sets the template format.
# You can change `out.width` to change the plot size.
# Note you may need to run install.packages("formatR") for this cell to work.

# You can also load your libraries here, like so:
library(ggplot2)
library(tidyverse)
# Code for 3a
her_data<-read.csv("/Users/quangphucvu/Downloads/hers.csv")

model3a<-lm(log(LDL)~BMI,data=her_data)
summary(model3a)
model3b<-lm(log(LDL)~BMI+I(BMI^2),data=her_data)
summary(model3b)

bmi_predict<-seq(20,40,length.out=1000)
her_predict<-data.frame(BMI=bmi_predict)
logLDL_predict<-predict(model3b,newdata = her_predict)
plot(bmi_predict,logLDL_predict)
# Code for 3c
BMI_cdc<-cut(her_data$BMI,breaks=c(-Inf,18.5,25,30,Inf),include.lowest = TRUE,labels = c("Underweight",
her_data$BMI_cdc<-BMI_cdc
# table(BMI_cdc)
model3c<-lm(log(LDL)~BMI_cdc,data = her_data)
knitr::kable(summary(model3c)$coefficients)
# Code for 3d
model3d<-lm(log(LDL)~BMI_cdc*statins,data = her_data)
knitr::kable(summary(model3d)$coefficients)
table_3d<-as.data.frame(summary(model3d)$coefficients)
table_3d$have_statin<-grepl("statin", rownames(table_3d), ignore.case = TRUE)
table_3d$predict<-table_3d$Estimate +(table_3d$have_statin)*rep(table_3d$Estimate[1:4])
plot_data<-table_3d[-c(1,5),]
plot_data$BMI_type<-rep(c("Healthy","Overweight","Obesity"),2)
ggplot(plot_data, aes(x = BMI_type, y = predict, fill = have_statin)) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.9)) +
  labs(title = "Predicted Value of Log(LDL) Compared To Underweight",
       x = "BMI type",
       y = "Predicted Log(LDL)") +
  theme_minimal()
# code for 4a
# set up data
set.seed(2024-09-19)
n<-1e6
x_i <- rnorm(n, 0, 1)
x_i_category<-cut(x_i,breaks=c(-Inf,-3,-2,-1,0,1,2,3,Inf),include.lowest = TRUE,labels = c("x1","x2","x3
epsilon_i<-rnorm(n,0,2)
Y_1<-0.2*x_i+0.2*x_i^2+0.1*x_i^3+epsilon_i
Y_2<-0.2*x_i+0.2*x_i^2-0.1*x_i^3+epsilon_i
data<-data.frame(Y_1 = Y_1,
           Y_2 = Y_2,
           x_i = x_i,
           x_i_cat<-as.factor(x_i_category))
```

```r
ref<-"x1"
formula_1<-"x_i+I(x_i^2)+I(x_i^3)"
formula_2<-"x_i"
formula_3<-"relevel(x_i_cat,ref=ref)"

#code for 4a(i)
model_3a_2_i<-lm(as.formula(paste0("Y_1~",formula_2)))
model_3a_3_i<-lm(as.formula(paste0("Y_1~",formula_3)))
result_3a_2_i<-summary(model_3a_2_i)$coefficients
knitr::kable(result_3a_2_i)
result_3a_3_i<-summary(model_3a_3_i)$coefficients
knitr::kable(result_3a_3_i)
#code for 4a(ii)
model_3a_2_ii<-lm(as.formula(paste0("Y_2~",formula_2)))
model_3a_3_ii<-lm(as.formula(paste0("Y_2~",formula_3)))
result_3a_2_ii<-summary(model_3a_2_ii)$coefficients
knitr::kable(result_3a_2_ii)
result_3a_3_ii<-summary(model_3a_3_ii)$coefficients
knitr::kable(result_3a_3_ii)
# code for 4b
x<-seq(-3,3,length.out=1000)
E_Y_1<-0.2*x+0.2*x^2+0.1*x^3
E_Y_2<-0.2*x+0.2*x^2-0.1*x^3
x_i_cat_test<-cut(x,breaks=c(-Inf,-3,-2,-1,0,1,2,3,Inf),include.lowest = TRUE,labels = c("x1","x2","x3"
data_predict<-data.frame(
          x_i = x,
          x_i_cat<-as.factor(x_i_cat_test))
hat_E_Y_2_i<-predict(model_3a_2_i,newdata = data_predict)
hat_E_Y_3_i<-predict(model_3a_3_i,newdata = data_predict)
hat_E_Y_2_ii<-predict(model_3a_2_ii,newdata = data_predict)
hat_E_Y_3_ii<-predict(model_3a_3_ii,newdata = data_predict)
plot(x, E_Y_1, main = "Observed values versus true value of E(Y|X)",
     xlab = "x_i", ylab = "Observed Values", pch = 19)
lines(x,hat_E_Y_2_i, col = "blue", lwd = 2)
lines(x,hat_E_Y_3_i, col = "red", lwd = 2)
legend("topright", legend = c("True E[Y|X]", "Model 2: E[Y|X]", "Model 3: E[Y|X]"),
       col = c("black", "blue", "red"), lwd = 2, pch = c(19, NA, NA))
plot(x, E_Y_2, main = "observed values versus true value of E(Y|X)",
     xlab = "x_i", ylab = "Observed Values", pch = 19)
lines(x,hat_E_Y_2_ii, col = "blue", lwd = 2)
lines(x,hat_E_Y_3_ii, col = "red", lwd = 2)
legend("topright", legend = c("True E[Y|X]", "Model 2: E[Y|X]", "Model 3: E[Y|X]"),
       col = c("black", "blue", "red"), lwd = 2, pch = c(19, NA, NA))
```