The Analytics Edge

Test your knowledge of Discrete Choice and Model Selection in R

Note to all. I have compiled the answers in the following format – for each question, the qualitative or "written" solutions will be provided together with their sub-questions. The R scripts (as well as the console outputs) will be provided *after* each whole question, followed by all the relevant plots. If I have missed anything in the solutions, or if you have any questions, you may email me at benjamin_tanwj@mymail.sutd.edu.sg. Thank you!

1. This problem set uses data on the choice of the heating system in California houses. The dataset in the file **Heating.csv** consists of observations for 900 single-family houses in California that were newly built and had central air-conditioning. The choice is among heating systems. Five types of systems are considered to have been possible:

   - gas central (gc)
   - gas room (gr)
   - electric central (ec)
   - electric room (er)
   - heat pump (hp)

   There are 900 observations where the variables are:

   - **idcase**: observation number (1-900)
   - **depvar**: identifies the chosen alternative (gc, gr, ec, er, hp)
   - **ic.alt**: installation cost for the 5 alternatives (alt = gc, gr, ec, er, hp)
   - **oc.alt**: annual operating cost for the 5 alternatives (alt = gc, gr, ec, er, hp)
   - **income**: annual income of the household (in tens of thousands of dollars)
   - **agehed**: age of the household head
   - **rooms**: number of rooms in the house
   - **region**: a factor with levels ncostl (northern coastal region), scostl (southern coastal region), mountn (mountain region), valley (central valley region)

   Note that the attributes of the alternatives, namely, installation cost and operating cost, take a different value for each alternative. Therefore, there are 5 installation costs (one for each of the 5 systems) and 5 operating costs. To estimate the logit model, the researcher needs data on the attributes of all the alternatives, not just the attributes for the chosen alternative. For example, it is not sufficient for the researcher to determine how much was paid for the system that was actually installed (i.e., the bill for the installation). The researcher needs to determine how much it would have cost to install each of the systems if they had been installed. The importance of costs in the choice process (i.e., the coefficients of installation and operating costs) is determined through comparison of the costs of the chosen system with the costs of the non-chosen systems.

For these data, the costs were calculated as the amount the system would cost if it were installed in the house, given the characteristics of the house (such as size), the price of gas and electricity in the house location, and the weather conditions in the area (which determine the necessary capacity of the system and the amount it will be run.) These cost are conditional on the house having central air-conditioning. (That's why the installation cost of gas central is lower than that for gas room: the central system can use the air-conditioning ducts that have been installed.)

You'll see that the first household chose alternative 1 (gas central), has an income of $70,000, the head of household is 25 years old, the house has 6 rooms, and is located in the north coastal area.

(a) Run a logit model with installation cost and operating cost as the only explanatory variables, without intercepts.

   i. Do the estimated coefficients have the expected signs?

   ii. Are both coefficients significantly different from zero?

   iii. Use the average of the probabilities to compute the predicted share. Compute the actual shares of houses with each system. How closely do the predicted shares match the actual shares of houses with each heating system?

   iv. The ratio of coefficients usually provides economically meaningful information in discrete choice models. The willingness to pay (*wtp*) through higher installation cost for a one-dollar reduction in operating costs is the ratio of the operating cost coefficient to the installation cost coefficient. What is the estimated *wtp* from this model? Note that the annual operating cost recurs every year while the installation cost is a one-time payment. Does the result make sense?

Solution. Read the **csv** file as `heat <- read.csv("Heating.csv")`

```
data <- mlogit.data(heat, choice = "depvar", shape = "wide", varying = c(3:12))
```

This creates a data object for *mlogit* to run, where *depvar* (dependent variable) is the choice, *wide* indicates that each row is one choice situation, and *varying* indicates the indices of the variables that are alternative specific.

Train the model as follows:

```
model1 <- mlogit(depvar ~ ic + oc - 1, data)
```

Here *ic, oc* are used to predict the chocie variable, where the "-1" indicates that no intercepts are needed.

i. Both the coefficients of *ic, oc* are negative which makes sense since as the installation cost and operating cost for a system increases, the probability of choosing that system goes down.

ii. p-value for $H_0 : \beta_{ic} = 0$ is $< 2.2 \times 10^{-16}$.
p-value for $H_0 : \beta_{oc} = 0$ is $< 2.2 \times 10^{-16}$.
These p-values are very close to 0, indicating that we can reject the null hypotheses that the coefficients are zero.

iii. `pred1 <- predict(model1, newdata = data)`
This predicts the choice probabilities for each choice situation.
Running `table(heat$depvar)/900` gives us:

ec	*er*	*gc*	*gr*	*hp*
0.071	0.093	0.636	0.143	0.055

which are the observed shares in the dataset.
`apply(pred1,2,mean)` computes the average of the choice probabilities and gives:

ec	*er*	*gc*	*gr*	*hp*
0.104	0.051	0.516	0.24	0.087

While the model captures the data reasonably well, there are still differences in the results for *gc* and *gr*.

iv. Computing the ratio of the coefficients gives us:

$$\frac{\beta_{oc}}{\beta_{ic}} = 0.739.$$

This implies that the decision-makers are willing to pay \$0.73 higher in installation cost for a \$1 reduction in operating cost. Note that it seems unreasonable for the decision-maker to pay only 73 cents higher for a one-time payment for a \$1 reduction in annual costs.

(b) Add alternative-specific constants to the model in (a). With K alternatives, at most $K - 1$ alternative specific constants can be estimated. The coefficient of $K - 1$ constants are interpreted as relative to Kth alternative. Normalize the constant for the alternative hp to 0.

i. How well do the estimated probabilities match the shares of customers choosing each alternative in this case?

ii. Calculate the *wtp* that is implied by the estimate. Is this reasonable?

iii. Suppose you had included constants for alternatives ec, er, gc, hp with the constant for alternative gr normalized to zero. What would be the estimated coefficient of the constant for alternative gc? Can you figure this out logically rather than actually estimating the model?

Solution. Train the new model as follows:

```
model2 <- mlogit(depvar ~ ic + oc, data, reflevel = "hp")
```

This forces the alternative *hp* to be the reference level, and the other alternative specific constants are relative to this.

i. In this case, the estimated probabilities match the shares exactly. The presence of alternative specific constants ensures that the average probabilities equal the average share.

ii. In this case, we have

$$\frac{\beta_{oc}}{\beta_{ic}} = 4.56$$

which suggests an extra down-payment of \$4.56 for a \$1 saving in annual operating costs. This seems more reasonable.

iii. We do just that by changing the *reflevel = "gr"* argument as shown. Note that in the previous model, the intercept for *gr* is 0.308. Hence, in this new model, we would reduce all the alternative specific constants downward by 0.308 – so we can make the intercept for *gr* zero. Note that this does not change the quality of fit, though, since probabilities are not modified by translating the *ASCs* by the same amount.

(c) Now try some models with sociodemographic variables entering.

i. Enter installation cost divided by income, instead of installation cost. With this specification, the magnitude of the installation cost coefficient is inversely related to income, such that high income households are less concerned with installation costs than lower income households. Does dividing installation cost by income seem to make the model better or worse than the model in (b)?

ii. Instead of dividing installation cost by income, enter alternative-specific income effects. You can do this by using the | argument in the mlogit formula. What do the estimates imply about the impact of income on the choice of central systems versus room system? Do these income terms enter significantly?

Solution.

i. Train the model as follows:

```
data$iic <- data$ic/data$income
model4 <- mlogit(depvar ~ oc + iic, data)
```

Note that the new log likelihood value is -1010.2 which is worse than -1008.2 previously. Also, in the new model, installation cost divided by income is not significant in this model.

ii.
```
model5 <- mlogit(depvar ~ oc + ic | income, data, reflevel = "hp")
```

This estimates a coefficient for each alternative that is income dependent. As income rises, probabiltiy of choosing a heat pump increases relative to others, As income rises, probability of choosing gas rooms drops relative to others. It is worth noting, however, that none of these variables are significant.

R Scripts.

```
> #1a)
> heat <- read.csv("Heating.csv")
> head(heat)
  idcase depvar  ic.gc  ic.gr  ic.ec  ic.er   ic.hp  oc.gc  oc.gr  oc.ec  oc.er  oc.hp income
1      1     gc 866.00 962.64 859.90 995.76 1135.50 199.69 151.72 553.34 505.60 237.88      7
2      2     gc 727.93 758.89 796.82 894.69  968.90 168.66 168.66 520.24 486.49 199.19      5
3      3     gc 599.48 783.05 719.86 900.11 1048.30 165.58 137.80 439.06 404.74 171.47      4
4      4     er 835.17 793.06 761.25 831.04 1048.70 180.88 147.14 483.00 425.22 222.95      2
5      5     er 755.59 846.29 858.86 985.64  883.05 174.91 138.90 404.41 389.52 178.49      2
6      6     gc 666.11 841.71 693.74 862.56  859.18 135.67 140.97 398.22 371.04 209.27      6
  agehed rooms region
1     25     6 ncostl
2     60     5 scostl
3     65     2 ncostl
4     50     4 scostl
5     25     6 valley
6     65     7 scostl
> library(mlogit)
> data <- mlogit.data(heat, choice = "depvar", shape = "wide", varying = c(3:12))
> model1 <- mlogit(depvar ~ ic + oc - 1, data)
> summary(model1)

Call:
mlogit(formula = depvar ~ ic + oc - 1, data = data, method = "nr",
    print.level = 0)

Frequencies of alternatives:
      ec       er       gc       gr       hp
0.071111 0.093333 0.636667 0.143333 0.055556

nr method
4 iterations, 0h:0m:0s
g'(-H)^-1g = 1.56E-07
gradient close to zero

Coefficients :
      Estimate  Std. Error t-value  Pr(>|t|)
ic -0.00623187  0.00035277 -17.665 < 2.2e-16 ***
```

```
oc -0.00458008  0.00032216 -14.217 < 2.2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Log-Likelihood: -1095.2

> #iii)
> pred1 <- predict(model1, newdata = data)
> table(heat$depvar)/900

        ec         er         gc         gr         hp
0.07111111 0.09333333 0.63666667 0.14333333 0.05555556
> apply(pred1,2,mean)
        ec         er         gc         gr         hp
0.10413057 0.05141477 0.51695653 0.24030898 0.08718915

> #iv)
> model1$coef["oc"]/model1$coef["ic"]
       oc
0.7349453
```

```
> #b)
> #i)
> model2 <- mlogit(depvar ˜ ic + oc, data, reflevel = "hp")
> summary(model2)

Call:
mlogit(formula = depvar ˜ ic + oc, data = data, reflevel = "hp",
    method = "nr", print.level = 0)

Frequencies of alternatives:
      hp       ec       er       gc       gr
0.055556 0.071111 0.093333 0.636667 0.143333

nr method
6 iterations, 0h:0m:0s
g’(-H)ˆ-1g = 9.58E-06
successive function values within tolerance limits

Coefficients :
                  Estimate  Std. Error t-value  Pr(>|t|)
ec:(intercept)  1.65884594  0.44841936  3.6993 0.0002162 ***
```

```
er:(intercept)  1.85343697  0.36195509  5.1206 3.045e-07 ***
gc:(intercept)  1.71097930  0.22674214  7.5459 4.485e-14 ***
gr:(intercept)  0.30826328  0.20659222  1.4921 0.1356640
ic             -0.00153315  0.00062086 -2.4694 0.0135333 *
oc             -0.00699637  0.00155408 -4.5019 6.734e-06 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Log-Likelihood: -1008.2
McFadden R^2:  0.013691
Likelihood ratio test : chisq = 27.99 (p.value = 8.3572e-07)
> pred2 <- predict(model2, newdata = data)
> table(heat$depvar)/900

        ec         er         gc         gr         hp
0.07111111 0.09333333 0.63666667 0.14333333 0.05555556
> apply(pred2,2,mean)
        hp         ec         er         gc         gr
0.05555556 0.07111111 0.09333333 0.63666666 0.14333334

> #ii)
> model2$coef["oc"]/model2$coef["ic"]
      oc
4.563385

> #iii)
> model3 <- mlogit(depvar ~ ic + oc, data, reflevel = "gr")
> summary(model3)

Call:
mlogit(formula = depvar ~ ic + oc, data = data, reflevel = "gr",
    method = "nr", print.level = 0)

Frequencies of alternatives:
      gr       ec       er       gc       hp
0.143333 0.071111 0.093333 0.636667 0.055556

nr method
6 iterations, 0h:0m:0s
g'(-H)^-1g = 9.58E-06
successive function values within tolerance limits

Coefficients :
                  Estimate  Std. Error t-value  Pr(>|t|)
```

```
ec:(intercept)  1.35058266  0.50715442  2.6631 0.0077434 **
er:(intercept)  1.54517369  0.43298757  3.5686 0.0003588 ***
gc:(intercept)  1.40271602  0.13398657 10.4691 < 2.2e-16 ***
hp:(intercept) -0.30826328  0.20659222 -1.4921 0.1356640
ic             -0.00153315  0.00062086 -2.4694 0.0135333 *
oc             -0.00699637  0.00155408 -4.5019 6.734e-06 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Log-Likelihood: -1008.2
McFadden R^2:  0.013691
Likelihood ratio test : chisq = 27.99 (p.value = 8.3572e-07)
```

```
> #c)
> #i)
> data$iic <- data$ic/data$income
> model4 <- mlogit(depvar ~ oc + iic, data)
> summary(model4)

Call:
mlogit(formula = depvar ~ oc + iic, data = data, method = "nr",
    print.level = 0)

Frequencies of alternatives:
      ec       er       gc       gr       hp
0.071111 0.093333 0.636667 0.143333 0.055556

nr method
6 iterations, 0h:0m:0s
g’(-H)^-1g = 1.03E-05
successive function values within tolerance limits

Coefficients :
                 Estimate Std. Error t-value  Pr(>|t|)
er:(intercept)  0.0639934  0.1944893  0.3290  0.742131
gc:(intercept)  0.0563481  0.4650251  0.1212  0.903555
gr:(intercept) -1.4653063  0.5033845 -2.9109  0.003604 **
hp:(intercept) -1.8700773  0.4364248 -4.2850 1.827e-05 ***
oc             -0.0071066  0.0015518 -4.5797 4.657e-06 ***
iic            -0.0027658  0.0018944 -1.4600  0.144298
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
Log-Likelihood: -1010.2
McFadden R^2:  0.011765
Likelihood ratio test : chisq = 24.052 (p.value = 5.9854e-06)

> #ii)
> model5 <- mlogit(depvar ~ oc + ic | income, data, reflevel = "hp")
> summary(model5)

Call:
mlogit(formula = depvar ~ oc + ic | income, data = data, reflevel = "hp",
    method = "nr", print.level = 0)

Frequencies of alternatives:
      hp       ec       er       gc       gr
0.055556 0.071111 0.093333 0.636667 0.143333

nr method
6 iterations, 0h:0m:0s
g'(-H)^-1g = 6.27E-06
successive function values within tolerance limits

Coefficients :
                  Estimate  Std. Error t-value  Pr(>|t|)
ec:(intercept)  1.95445797  0.70353833  2.7780 0.0054688 **
er:(intercept)  2.30560852  0.62390478  3.6954 0.0002195 ***
gc:(intercept)  2.05517018  0.48639682  4.2253 2.386e-05 ***
gr:(intercept)  1.14158139  0.51828845  2.2026 0.0276231 *
oc             -0.00696000  0.00155383 -4.4792 7.491e-06 ***
ic             -0.00153534  0.00062251 -2.4664 0.0136486 *
ec:income      -0.06362917  0.11329865 -0.5616 0.5743846
er:income      -0.09685787  0.10755423 -0.9005 0.3678281
gc:income      -0.07178917  0.08878777 -0.8085 0.4187752
gr:income      -0.17981159  0.10012691 -1.7958 0.0725205 .
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Log-Likelihood: -1005.9
McFadden R^2:  0.01598
Likelihood ratio test : chisq = 32.67 (p.value = 1.2134e-05)
```

2. A sample of residential electricity customers were asked a series of choice experiments. The data is provided in the file **Electricity.csv**. In each experiment, four hypothetical electricity suppliers were described. The person was asked which of the four suppliers he/she would choose. As many as 12 experiments were presented to each person. Some people stopped before answering all 12. There are 361 people in the sample, and a total of 4308 experiments. In the experiments, the characteristics of each supplier were stated.

   - The price of the supplier was either one of these options:
     - a fixed price at a stated cents per kWh, with the price varying over suppliers and experiments (**pf1, pf2, pf3, pf4**)
     - a time-of-day (tod) rate under which the price is 11 cents per kWh from 8am to 8pm and 5 cents per kWh from 8pm to 8am. These tod prices did not vary over suppliers or experiments: whenever the supplier was said to offer tod, the prices were stated as above (**tod1, tod2, tod3, tod4**)
     - a seasonal rate under which the price is 10 cents per kWh in the summer, 8 cents per kWh in the winter, and 6 cents per kWh in the spring and fall. Like tod rates, these prices did not vary. Note that the price is for the electricity only, not transmission and distribution, which is supplied by the local regulated utility (**seas1, seas2, seas3, seas4**).
   - The length of contract that the supplier offered was also stated, in years (such as 1 year or 5 years.) During this contract period, the supplier guaranteed the prices and the buyer would have to pay a penalty if he/she switched to another supplier. The supplier could offer no contract in which case either side could stop the agreement at any time. This is recorded as a contract length of 0 (**cl1, cl2, cl3, cl4**).
   - Some suppliers were also described as being a local company or a "well-known" company. If the supplier was not local or well-known, then nothing was said about them in this regard (**loc1, loc2, loc3, loc4, wk1, wk2, wk3, wk4**).

   The actual choices made are captured in **choice** with **id** capturing the customer identity.

   (a) Run a mixed logit model without intercepts and a normal distribution for the 6 parameters of the model and taking into account the panel data structure.

      i. Using the estimated mean coefficients, determine the amount that a customer with average coefficients for price and length is willing to pay for an extra year of contract length.

      ii. Determine the share of the population who are estimated to dislike long term contracts (i.e. have a negative coefficient for the length.)

   Solution. This dataframe has 4308 observations with 26 variables. Also, each choice situation corresponds to one row, meaning we have a *shape = "wide"* situation here. Train the model as follows:

```
electricity <- read.csv("Electricity.csv")
data1 <- mlogit.data(electricity, id.var = "id", choice = "choice",
                     varying = 3:26, shape = "wide", sep = "")
model1 <- mlogit(choice ~ pf + cl + loc + wk + tod + seas - 1, data =
                 data1, rpar = c(pf = ’n’, cl = ’n’, loc = ’n’, wk = ’n’,
                 tod = ’n’, seas = ’n’), panel = TRUE)
```

We use the default settings to solve the mixed logit model with panel data to indicate multiple observations per individual.

i. The mean coefficient of contract length is around -0.18 indicating consumers prefer shorter contracts. Since the mean price coefficient is -0.84, a customer will pay $0.18/0.84 = 0.21$ cents per kWh ro reduce contract length by 1 year.

ii. The coefficient for length (*cl*) is normally distributed, with mean -0.18 and standard deviation 0.31 (*sd.rl*).

Share of people with negative coefficients is given as:

$$\begin{aligned} P(\mu + \sigma \cdot Z \leq 0) &= P\Big(Z \leq \frac{-\mu}{\sigma}\Big) \\ &= pnorm\Big(-\frac{model1\$coef[\text{“}cl\text{”}]}{model1\$coef[\text{“}sd.cl\text{”}]}\Big) \\ &= 0.719. \end{aligned}$$

Roughly 72% of the population dislike long-term contracts.

(b) The price coefficient is assumed to be normally distributed in these runs. This assumption means that some people are assumed to have positive price coefficients, since the normal distribution has support on both sides of zero. Using your estimates from before, determine the share of customers with positive price coefficients (Hint: Use the pnorm function to calculate this share). As you can see, this is pretty small share and can probably be ignored. However, in some situations, a normal distribution for the price coefficient will give a fairly large share with the wrong sign. Revise the model to make the price coefficient fixed rather than random. A fixed price coefficient also makes it easier to calculate the distribution of willingness to pay (*wtp*) for each non-price attribute. If the price coefficient is fixed, the distribution of *wtp* for an attribute has the same distribution as the attribute’s coefficient, simply scaled by the price coefficient. However, when the price coefficient is random, the distribution of *wtp* is the ratio of two distributions, which is harder to work with. What is the estimated value of the price coefficient/ Compare the log likelihood of the new model with the old model.

Solution. The share of customers with negative price coefficients is given as 0.9999998 (very close to 1) as should be expected. To create a model similar to the previous, but with deterministic price coefficients, just remove the *pf = “n”* from the vector in the *rpar* argument.

The new log-likelihood is given as -4110, which is smaller than that of the old model (-4089.6) and is expected. The estimated value of the price coefficient is -0.81.

(c) You think that everyone must like using a known company rather than an unknown one, and yet the normal distribution implies that some people dislike using a known company. Revise the model to give the coefficient of wk a uniform distribution (do this with the price coefficient fixed). What is the estimated distribution for the coefficient of wk and the estimated price coefficient?

Solution. Run the same model as in part b), but change *wk* = *“n”* to *wk* = *“u”* (as in uniform) in the *rpar* argument vector. From the results we see that the uniform distribution of *wk* is from 0.133 to 2.58 with mean = 1.36. The estimated price coefficient is -0.811.

R Scripts.

```
> #2
> #a)i)
> electricity <- read.csv("Electricity.csv")
> data1 <- mlogit.data(electricity, id.var = "id", choice = "choice",varying = 3:26,
                       shape = "wide", sep = "")
> model1 <- mlogit(choice ~ pf + cl + loc + wk + tod + seas - 1, data = data1, rpar =
                   c(pf = 'n', cl = 'n', loc = 'n', wk = 'n', tod = 'n', seas = 'n'),
                   panel = TRUE)
> summary(model1)

Call:
mlogit(formula = choice ~ pf + cl + loc + wk + tod + seas - 1,
    data = data1, rpar = c(pf = "n", cl = "n", loc = "n", wk = "n",
        tod = "n", seas = "n"), panel = TRUE)

Frequencies of alternatives:
      1       2       3       4
0.22702 0.26393 0.23816 0.27089

bfgs method
18 iterations, 0h:0m:43s
g'(-H)^-1g = 9.3E-07
gradient close to zero

Coefficients :
          Estimate Std. Error  t-value  Pr(>|t|)
```

```
pf      -0.8486024  0.0310904 -27.2947 < 2.2e-16 ***
cl      -0.1800805  0.0118515 -15.1948 < 2.2e-16 ***
loc      1.9510287  0.0721430  27.0439 < 2.2e-16 ***
wk       1.3941656  0.0586974  23.7517 < 2.2e-16 ***
tod     -8.1899414  0.2627907 -31.1653 < 2.2e-16 ***
seas    -8.3523617  0.2634401 -31.7050 < 2.2e-16 ***
sd.pf    0.1668789  0.0090082  18.5253 < 2.2e-16 ***
sd.cl    0.3092478  0.0149194  20.7279 < 2.2e-16 ***
sd.loc   1.1343809  0.0711787  15.9371 < 2.2e-16 ***
sd.wk    0.4388592  0.0694876   6.3156  2.69e-10 ***
sd.tod   2.1251956  0.1009321  21.0557 < 2.2e-16 ***
sd.seas  1.2218426  0.0838662  14.5689 < 2.2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Log-Likelihood: -4089.6

random coefficients
     Min.    1st Qu.     Median       Mean     3rd Qu. Max.
pf   -Inf -0.9611605 -0.8486024 -0.8486024 -0.73604435  Inf
cl   -Inf -0.3886650 -0.1800805 -0.1800805  0.02850392  Inf
loc  -Inf  1.1859005  1.9510287  1.9510287  2.71615703  Inf
wk   -Inf  1.0981595  1.3941656  1.3941656  1.69017164  Inf
tod  -Inf -9.6233640 -8.1899414 -8.1899414 -6.75651868  Inf
seas -Inf -9.1764820 -8.3523617 -8.3523617 -7.52824142  Inf
> #ii)
> pnorm(-model1$coef["cl"]/model1$coef["sd.cl"])
       cl
0.7198237



> #b)
> pnorm(-model1$coef["pf"]/model1$coef["sd.pf"])
       pf
0.9999998
> model2 <- mlogit(choice ~ pf + cl + loc + wk + tod + seas - 1, data = data1, rpar =
                   c(cl = ’n’, loc = ’n’, wk = ’n’, tod = ’n’, seas = ’n’), panel = TRUE)
> summary(model2)

Call:
mlogit(formula = choice ~ pf + cl + loc + wk + tod + seas - 1,
    data = data1, rpar = c(cl = "n", loc = "n", wk = "n", tod = "n",
        seas = "n"), panel = TRUE)
```

```
Frequencies of alternatives:
      1       2       3       4
0.22702 0.26393 0.23816 0.27089

bfgs method
15 iterations, 0h:0m:36s
g’(-H)^-1g = 9.59E-07
gradient close to zero

Coefficients :
         Estimate Std. Error t-value  Pr(>|t|)
pf      -0.810620   0.030282 -26.769 < 2.2e-16 ***
cl      -0.189633   0.012103 -15.668 < 2.2e-16 ***
loc      1.925156   0.071148  27.058 < 2.2e-16 ***
wk       1.350738   0.059260  22.794 < 2.2e-16 ***
tod     -7.857962   0.256376 -30.650 < 2.2e-16 ***
seas    -7.762911   0.254108 -30.550 < 2.2e-16 ***
sd.cl    0.309565   0.015371  20.140 < 2.2e-16 ***
sd.loc   1.101270   0.073389  15.006 < 2.2e-16 ***
sd.wk    0.760579   0.064427  11.805 < 2.2e-16 ***
sd.tod   2.299435   0.097326  23.626 < 2.2e-16 ***
sd.seas  1.791816   0.098986  18.102 < 2.2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Log-Likelihood: -4110.2

random coefficients
     Min.    1st Qu.     Median       Mean     3rd Qu. Max.
cl   -Inf -0.3984313 -0.1896326 -0.1896326  0.01916609  Inf
loc  -Inf  1.1823613  1.9251564  1.9251564  2.66795158  Inf
wk   -Inf  0.8377355  1.3507383  1.3507383  1.86374104  Inf
tod  -Inf -9.4089080 -7.8579624 -7.8579624 -6.30701684  Inf
seas -Inf -8.9714728 -7.7629111 -7.7629111 -6.55434946  Inf


> #c)
> model3 <- mlogit(choice ˜ pf + cl + loc + wk + tod + seas - 1, data = data1, rpar =
                   c(cl = ’n’, loc = ’n’, wk = ’u’, tod = ’n’, seas = ’n’), panel = TRUE)
> summary(model3)

Call:
mlogit(formula = choice ˜ pf + cl + loc + wk + tod + seas - 1,
    data = data1, rpar = c(cl = "n", loc = "n", wk = "u", tod = "n",
        seas = "n"), panel = TRUE)

Frequencies of alternatives:
      1       2       3       4
0.22702 0.26393 0.23816 0.27089

bfgs method
17 iterations, 0h:0m:42s
g'(-H)^-1g = 1.53E-07
gradient close to zero

Coefficients :
         Estimate Std. Error t-value  Pr(>|t|)
pf      -0.811314   0.030298 -26.778 < 2.2e-16 ***
cl      -0.188318   0.012061 -15.614 < 2.2e-16 ***
loc      1.928686   0.071206  27.086 < 2.2e-16 ***
wk       1.360809   0.059256  22.965 < 2.2e-16 ***
tod     -7.862986   0.256746 -30.625 < 2.2e-16 ***
seas    -7.779961   0.254544 -30.564 < 2.2e-16 ***
sd.cl    0.309688   0.015392  20.120 < 2.2e-16 ***
sd.loc   1.127218   0.074139  15.204 < 2.2e-16 ***
sd.wk    1.227168   0.106643  11.507 < 2.2e-16 ***
sd.tod   2.316823   0.098231  23.586 < 2.2e-16 ***
sd.seas  1.791521   0.099100  18.078 < 2.2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Log-Likelihood: -4110.5

random coefficients
          Min.    1st Qu.     Median       Mean     3rd Qu.     Max.
cl        -Inf -0.3971992 -0.1883179 -0.1883179  0.02056344      Inf
loc       -Inf  1.1683890  1.9286860  1.9286860  2.68898300      Inf
wk   0.1336413  0.7472252  1.3608091  1.3608091  1.97439299 2.587977
tod       -Inf -9.4256597 -7.8629861 -7.8629861 -6.30031245      Inf
seas      -Inf -8.9883233 -7.7799606 -7.7799606 -6.57159785      Inf
```

3. Suppose we perform best subset, forward stepwise, and backward stepwise selection on a single data set. For each approach, we obtain p+1 models, containing 0, 1, 2, ..., p predictors. Provide your answers for the following questions:

   (a) Which of the three models with k predictors has the smallest training sum of squared errors?

   Solution. Clearly, for every value of k, the best subset selection method will have the smallest training sum of squared errors – it finds the global optimum set for each k.

   (b) Which of the three models with k predictors has the smallest test sum of squared errors?

   Solution. This cannot be determined since a low training SSE does not necessarily translate to a low test set SSE.

   (c) True or False:

   i. The predictors in the k-variable model identified by forward stepwise are a subset of the predictors in the (k + 1)-variable model identified by forward stepwise selection.

   Solution. True. Each step in the forward stepwise selection method corresponds to adding only 1 variable to the previous set in an optimal manner.

   ii. The predictors in the k-variable model identified by backward stepwise are a subset of the predictors in the (k+ 1)- variable model identified by backward stepwise selection.

   Solution. True. In backward stepwise selection, we drop a variable at each step.

   iii. The predictors in the k-variable model identified by backward stepwise are a subset of the predictors in the (k + 1)- variable model identified by forward stepwise selection.

   Solution. False.

   iv. The predictors in the k-variable model identified by forward stepwise are a subset of the predictors in the (k + 1)-variable model identified by backward stepwise selection.

   Solution. False.

   v. The predictors in the k-variable model identified by best subset are a subset of the predictors in the (k+1)-variable model identified by best subset selection.

   Solution. False.

4. In this question, we will use the data in **College.csv** to investigate how well we can predict the number of applications received for universities and colleges in the US. The dataset has the following fields:

   - Private: Private/public indicator
   - Apps: Number of applications received
   - Accept: Number of applicants accepted
   - Enroll: Number of new students enrolled
   - Top10perc: New students from top 10% of high school class
   - Top25perc: New students from top 25% of high school class
   - F.Undergrad: Number of full-time undergraduate students
   - P.Undergrad: Number of part-time undergraduate students
   - Outstate: Out of state tuition
   - Room.Board: Room and board costs
   - Books: Estimated book costs
   - Personal: Estimated personal spending
   - PhD: Percent of faculty with PhDs
   - Terminal: Percent of faculty with a terminal degree
   - S.F.Ratio: Student/faculty ratio
   - perc.alumni: Percent of alumni who donate
   - Expend: Instructional expenditure per student
   - Grad.Rate: Graduation rate

   (a) Split the data set into a training set and a test set using the seed 1 and the sample() function with 80% in the training set and 20% in the test set. How many observations are there in the training and test sets?

   Solution. There are 621 observations in the training set, and 156 observations in the test set.

   (b) Fit a linear model using least squares on the training set. What is the average sum of squared error of the model on the training set? Report on the average sum of squared error on the test set obtained from the model.

   Solution. The average SSE for the training set is obtained as `mean(model1$residuals^2)`, and is reported as 1061946. The test mean squared error is given as 1075064.

(c) Use the backward stepwise selection method to select the variables for the regression model on the training set. Which is the first variable dropped from the set?

Solution. We see from the result that the subset of size 16 is obtained by dropping *P.Undergrad*.

(d) Plot the adjusted R^2 for all these models. If we choose the model based on the best adjusted R^2 value, which variables should be included in the model?

Solution. Plot found here. The best adjusted R^2 value corresponds to the subset of size 12 – In addition to the intercept, we would include *PrivateYes, Accept, Enroll, Top10perc, Top25perc, F.Undergrad, Outstate, Room.Board, PhD, S.F.Ratio, Expend, Grad.Rate*, and drop *P.Undergrad, Books, Terminal, perc.alumni, Personal.*

(e) Use the model identified in part (d) to estimate the average sum of squared test error. Does this improve on the model in part (b) in the prediction accuracy?

Solution. The average test set SSE is given as 1070293, which is less than 1075064 as in b), so yes, it improves on the accuracy.

(f) Fit a LASSO model on the training set. Use the command to define the grid for λ:
grid <- 10∧ seq(10,-2, length=100)
Plot the behavior of the coefficients as λ changes.

Solution. Plot found here.

(g) Set the seed to 1 before running the cross-validation with LASSO to choose the best λ. Use 10-fold cross validation. Report the test error obtained, along with the number of non-zero coefficient estimates.

Solution. The best λ is given as 0.497. There are 17 nonzeros in the complete model. Here LASSO gives the full model, and the test error will be the same as model 1.

R Scripts.

```
> #4
> #a)
> College <- read.csv("College.csv")
> set.seed(1)
> trainid <- sample(1:nrow(College),0.8*nrow(College))
> testid <- -trainid
> train<- College[trainid,]
> test<- College[testid,]


> #b)
> model1 <- lm(Apps~.,data = train)
> summary(model1)

Call:
lm(formula = Apps ~ ., data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-4835.7  -426.1   -32.7   307.9  7857.6

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -3.745e+02  4.544e+02  -0.824 0.410129
PrivateYes  -5.086e+02  1.532e+02  -3.320 0.000953 ***
Accept       1.591e+00  4.254e-02  37.407  < 2e-16 ***
Enroll      -8.866e-01  2.028e-01  -4.371 1.46e-05 ***
Top10perc    4.782e+01  6.470e+00   7.391 4.87e-13 ***
Top25perc   -1.238e+01  5.122e+00  -2.416 0.015972 *
F.Undergrad  5.589e-02  3.703e-02   1.509 0.131747
P.Undergrad  3.025e-03  4.282e-02   0.071 0.943700
Outstate    -8.484e-02  2.169e-02  -3.911 0.000102 ***
Room.Board   1.745e-01  5.338e-02   3.269 0.001140 **
Books       -4.336e-02  2.688e-01  -0.161 0.871924
Personal     3.255e-02  7.078e-02   0.460 0.645797
PhD         -7.251e+00  5.422e+00  -1.337 0.181657
Terminal    -5.150e+00  6.036e+00  -0.853 0.393902
S.F.Ratio    1.704e+01  1.468e+01   1.161 0.246228
perc.alumni -2.189e+00  4.654e+00  -0.470 0.638357
Expend       7.403e-02  1.456e-02   5.083 4.98e-07 ***
Grad.Rate    7.933e+00  3.334e+00   2.380 0.017629 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1046 on 603 degrees of freedom
Multiple R-squared:  0.9344,Adjusted R-squared:  0.9325
F-statistic:   505 on 17 and 603 DF,  p-value: < 2.2e-16

> predict1 <- predict(model1, newdata = test)
> mse1 <- mean((test$Apps - predict1)ˆ2)
> mse1
[1] 1075064


> #c)
> library(leaps)
> model2 <- regsubsets(Apps˜., data=train, nvmax=17, method = "backward")
> summary(model2)
Subset selection object
Call: regsubsets.formula(Apps ˜ ., data = train, nvmax = 17, method = "backward")
17 Variables  (and intercept)
            Forced in Forced out
PrivateYes      FALSE      FALSE
Accept          FALSE      FALSE
Enroll          FALSE      FALSE
Top10perc       FALSE      FALSE
Top25perc       FALSE      FALSE
F.Undergrad     FALSE      FALSE
P.Undergrad     FALSE      FALSE
Outstate        FALSE      FALSE
Room.Board      FALSE      FALSE
Books           FALSE      FALSE
Personal        FALSE      FALSE
PhD             FALSE      FALSE
Terminal        FALSE      FALSE
S.F.Ratio       FALSE      FALSE
perc.alumni     FALSE      FALSE
Expend          FALSE      FALSE
Grad.Rate       FALSE      FALSE
1 subsets of each size up to 17
Selection Algorithm: backward
          PrivateYes Accept Enroll Top10perc Top25perc F.Undergrad P.Undergrad Outstate
1  ( 1 )  " "        "*"    " "    " "       " "       " "         " "         " "
2  ( 1 )  " "        "*"    " "    "*"       " "       " "         " "         " "
3  ( 1 )  " "        "*"    "*"    "*"       " "       " "         " "         " "
4  ( 1 )  "*"        "*"    "*"    "*"       " "       " "         " "         " "
5  ( 1 )  " "        "*"    "*"    "*"       " "       " "         " "         "*"
6  ( 1 )  " "        "*"    "*"    "*"       " "       " "         " "         "*"
7  ( 1 )  "*"        "*"    "*"    "*"       " "       " "         " "         "*"
8  ( 1 )  "*"        "*"    "*"    "*"       " "       " "         " "         "*"
9  ( 1 )  "*"        "*"    "*"    "*"       "*"       " "         " "         "*"
10  ( 1 ) "*"        "*"    "*"    "*"       "*"       " "         " "         "*"
11  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
```

```
12  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
13  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
14  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
15  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
16  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
17  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         "*"         "*"
          Room.Board Books Personal PhD Terminal S.F.Ratio perc.alumni Expend Grad.Rate
1  ( 1 )  " "        " "   " "      " " " "      " "       " "         " "    " "
2  ( 1 )  " "        " "   " "      " " " "      " "       " "         " "    " "
3  ( 1 )  " "        " "   " "      " " " "      " "       " "         " "    " "
4  ( 1 )  " "        " "   " "      " " " "      " "       " "         " "    " "
5  ( 1 )  " "        " "   " "      " " " "      " "       " "         "*"    " "
6  ( 1 )  "*"        " "   " "      " " " "      " "       " "         "*"    " "
7  ( 1 )  "*"        " "   " "      " " " "      " "       " "         "*"    " "
8  ( 1 )  "*"        " "   " "      "*" " "      " "       " "         "*"    " "
9  ( 1 )  "*"        " "   " "      "*" " "      " "       " "         "*"    " "
10  ( 1 ) "*"        " "   " "      "*" " "      " "       " "         "*"    "*"
11  ( 1 ) "*"        " "   " "      "*" " "      " "       " "         "*"    "*"
12  ( 1 ) "*"        " "   " "      "*" " "      "*"       " "         "*"    "*"
13  ( 1 ) "*"        " "   " "      "*" "*"      "*"       " "         "*"    "*"
14  ( 1 ) "*"        " "   " "      "*" "*"      "*"       "*"         "*"    "*"
15  ( 1 ) "*"        " "   "*"      "*" "*"      "*"       "*"         "*"    "*"
16  ( 1 ) "*"        "*"   "*"      "*" "*"      "*"       "*"         "*"    "*"
17  ( 1 ) "*"        "*"   "*"      "*" "*"      "*"       "*"         "*"    "*"


> #d)
> summary(model2)
Subset selection object
Call: regsubsets.formula(Apps ~ ., data = train, nvmax = 17, method = "backward")
17 Variables  (and intercept)
            Forced in Forced out
PrivateYes      FALSE      FALSE
Accept          FALSE      FALSE
Enroll          FALSE      FALSE
Top10perc       FALSE      FALSE
Top25perc       FALSE      FALSE
F.Undergrad     FALSE      FALSE
P.Undergrad     FALSE      FALSE
Outstate        FALSE      FALSE
Room.Board      FALSE      FALSE
Books           FALSE      FALSE
Personal        FALSE      FALSE
PhD             FALSE      FALSE
```

```
Terminal        FALSE      FALSE
S.F.Ratio       FALSE      FALSE
perc.alumni     FALSE      FALSE
Expend          FALSE      FALSE
Grad.Rate       FALSE      FALSE
1 subsets of each size up to 17
Selection Algorithm: backward
          PrivateYes Accept Enroll Top10perc Top25perc F.Undergrad P.Undergrad Outstate
1  ( 1 )  " "        "*"    " "    " "       " "       " "         " "         " "
2  ( 1 )  " "        "*"    " "    "*"       " "       " "         " "         " "
3  ( 1 )  " "        "*"    "*"    "*"       " "       " "         " "         " "
4  ( 1 )  "*"        "*"    "*"    "*"       " "       " "         " "         " "
5  ( 1 )  " "        "*"    "*"    "*"       " "       " "         " "         "*"
6  ( 1 )  " "        "*"    "*"    "*"       " "       " "         " "         "*"
7  ( 1 )  "*"        "*"    "*"    "*"       " "       " "         " "         "*"
8  ( 1 )  "*"        "*"    "*"    "*"       " "       " "         " "         "*"
9  ( 1 )  "*"        "*"    "*"    "*"       "*"       " "         " "         "*"
10  ( 1 ) "*"        "*"    "*"    "*"       "*"       " "         " "         "*"
11  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
12  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
13  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
14  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
15  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
16  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         " "         "*"
17  ( 1 ) "*"        "*"    "*"    "*"       "*"       "*"         "*"         "*"
          Room.Board Books Personal PhD Terminal S.F.Ratio perc.alumni Expend Grad.Rate
1  ( 1 )  " "        " "   " "      " " " "      " "       " "         " "    " "
2  ( 1 )  " "        " "   " "      " " " "      " "       " "         " "    " "
3  ( 1 )  " "        " "   " "      " " " "      " "       " "         " "    " "
4  ( 1 )  " "        " "   " "      " " " "      " "       " "         " "    " "
5  ( 1 )  " "        " "   " "      " " " "      " "       " "         "*"    " "
6  ( 1 )  "*"        " "   " "      " " " "      " "       " "         "*"    " "
7  ( 1 )  "*"        " "   " "      " " " "      " "       " "         "*"    " "
8  ( 1 )  "*"        " "   " "      "*" " "      " "       " "         "*"    " "
9  ( 1 )  "*"        " "   " "      "*" " "      " "       " "         "*"    " "
10  ( 1 ) "*"        " "   " "      "*" " "      " "       " "         "*"    "*"
11  ( 1 ) "*"        " "   " "      "*" " "      " "       " "         "*"    "*"
12  ( 1 ) "*"        " "   " "      "*" " "      "*"       " "         "*"    "*"
13  ( 1 ) "*"        " "   " "      "*" "*"      "*"       " "         "*"    "*"
14  ( 1 ) "*"        " "   " "      "*" "*"      "*"       "*"         "*"    "*"
15  ( 1 ) "*"        " "   "*"      "*" "*"      "*"       "*"         "*"    "*"
16  ( 1 ) "*"        "*"   "*"      "*" "*"      "*"       "*"         "*"    "*"
17  ( 1 ) "*"        "*"   "*"      "*" "*"      "*"       "*"         "*"    "*"
> plot(summary(model2)$adjr2)
> which.max(summary(model2)$adjr2)
[1] 12
> summary(model2)$which
   (Intercept) PrivateYes Accept Enroll Top10perc Top25perc F.Undergrad P.Undergrad Outstate
1         TRUE      FALSE   TRUE  FALSE     FALSE     FALSE       FALSE       FALSE    FALSE
2         TRUE      FALSE   TRUE  FALSE      TRUE     FALSE       FALSE       FALSE    FALSE
3         TRUE      FALSE   TRUE   TRUE      TRUE     FALSE       FALSE       FALSE    FALSE
4         TRUE       TRUE   TRUE   TRUE      TRUE     FALSE       FALSE       FALSE    FALSE
5         TRUE      FALSE   TRUE   TRUE      TRUE     FALSE       FALSE       FALSE     TRUE
6         TRUE      FALSE   TRUE   TRUE      TRUE     FALSE       FALSE       FALSE     TRUE
7         TRUE       TRUE   TRUE   TRUE      TRUE     FALSE       FALSE       FALSE     TRUE
8         TRUE       TRUE   TRUE   TRUE      TRUE     FALSE       FALSE       FALSE     TRUE
9         TRUE       TRUE   TRUE   TRUE      TRUE      TRUE       FALSE       FALSE     TRUE
10        TRUE       TRUE   TRUE   TRUE      TRUE      TRUE       FALSE       FALSE     TRUE
11        TRUE       TRUE   TRUE   TRUE      TRUE      TRUE        TRUE       FALSE     TRUE
12        TRUE       TRUE   TRUE   TRUE      TRUE      TRUE        TRUE       FALSE     TRUE
13        TRUE       TRUE   TRUE   TRUE      TRUE      TRUE        TRUE       FALSE     TRUE
14        TRUE       TRUE   TRUE   TRUE      TRUE      TRUE        TRUE       FALSE     TRUE
15        TRUE       TRUE   TRUE   TRUE      TRUE      TRUE        TRUE       FALSE     TRUE
16        TRUE       TRUE   TRUE   TRUE      TRUE      TRUE        TRUE       FALSE     TRUE
17        TRUE       TRUE   TRUE   TRUE      TRUE      TRUE        TRUE        TRUE     TRUE
   Room.Board Books Personal   PhD Terminal S.F.Ratio perc.alumni Expend Grad.Rate
1       FALSE FALSE    FALSE FALSE    FALSE     FALSE       FALSE  FALSE     FALSE
2       FALSE FALSE    FALSE FALSE    FALSE     FALSE       FALSE  FALSE     FALSE
3       FALSE FALSE    FALSE FALSE    FALSE     FALSE       FALSE  FALSE     FALSE
4       FALSE FALSE    FALSE FALSE    FALSE     FALSE       FALSE  FALSE     FALSE
5       FALSE FALSE    FALSE FALSE    FALSE     FALSE       FALSE   TRUE     FALSE
6        TRUE FALSE    FALSE FALSE    FALSE     FALSE       FALSE   TRUE     FALSE
7        TRUE FALSE    FALSE FALSE    FALSE     FALSE       FALSE   TRUE     FALSE
8        TRUE FALSE    FALSE  TRUE    FALSE     FALSE       FALSE   TRUE     FALSE
9        TRUE FALSE    FALSE  TRUE    FALSE     FALSE       FALSE   TRUE     FALSE
10       TRUE FALSE    FALSE  TRUE    FALSE     FALSE       FALSE   TRUE      TRUE
11       TRUE FALSE    FALSE  TRUE    FALSE     FALSE       FALSE   TRUE      TRUE
12       TRUE FALSE    FALSE  TRUE    FALSE      TRUE       FALSE   TRUE      TRUE
13       TRUE FALSE    FALSE  TRUE     TRUE      TRUE       FALSE   TRUE      TRUE
14       TRUE FALSE    FALSE  TRUE     TRUE      TRUE        TRUE   TRUE      TRUE
15       TRUE FALSE     TRUE  TRUE     TRUE      TRUE        TRUE   TRUE      TRUE
16       TRUE  TRUE     TRUE  TRUE     TRUE      TRUE        TRUE   TRUE      TRUE
17       TRUE  TRUE     TRUE  TRUE     TRUE      TRUE        TRUE   TRUE      TRUE


> #e)
> model3 <- lm(Apps~Private+Accept+Enroll+Top10perc+Top25perc+F.Undergrad+Outstate+
               Room.Board+PhD+S.F.Ratio+Expend+Grad.Rate,data=train)
```

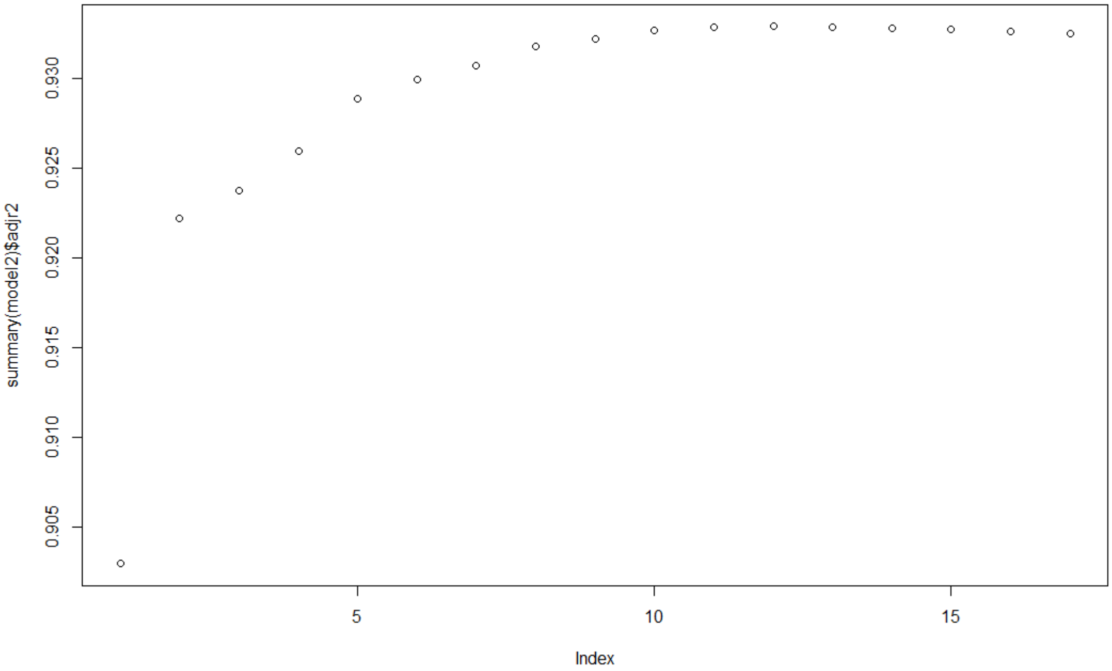
```
> summary(model3)

Call:
lm(formula = Apps ~ Private + Accept + Enroll + Top10perc + Top25perc +
    F.Undergrad + Outstate + Room.Board + PhD + S.F.Ratio + Expend +
    Grad.Rate, data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-4854.6  -427.5    -2.0   288.7  7795.8

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -445.32668  395.09820  -1.127 0.260133
PrivateYes  -505.02266  151.59344  -3.331 0.000916 ***
Accept         1.59459    0.04185  38.099  < 2e-16 ***
Enroll        -0.89222    0.19978  -4.466 9.50e-06 ***
Top10perc     48.08852    6.41076   7.501 2.25e-13 ***
Top25perc    -12.93159    5.07738  -2.547 0.011114 *
F.Undergrad    0.05688    0.03470   1.639 0.101710
Outstate      -0.09075    0.02079  -4.365 1.50e-05 ***
Room.Board     0.17368    0.05131   3.385 0.000758 ***
PhD          -10.72073    3.51836  -3.047 0.002411 **
S.F.Ratio     16.93860   14.54435   1.165 0.244631
Expend         0.07346    0.01447   5.077 5.10e-07 ***
Grad.Rate      7.49413    3.19970   2.342 0.019496 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1043 on 608 degrees of freedom
Multiple R-squared:  0.9342,Adjusted R-squared:  0.9329
F-statistic: 719.5 on 12 and 608 DF,  p-value: < 2.2e-16

> predict3 <- predict(model3,newdata=test)
> mse3 <- mean((test$Apps - predict3)^2)
> mse3
[1] 1070293


> #f)
> library(glmnet)
> grid <- 10^seq(10,-2, length=100)
> x <- model.matrix(Apps~.,College)
> y <- College$Apps
> model4 <- glmnet(x[trainid,],y[trainid],lambda=grid)
> plot(model4,xvar="lambda")
```

```
> #g)
> set.seed(1)
> cvmodel4<-cv.glmnet(x[trainid,],y[trainid],nfolds=10,lambda=grid)
> cvmodel4
$lambda
  [1] 1.000000e+10 7.564633e+09 5.722368e+09 4.328761e+09 3.274549e+09 2.477076e+09
  [7] 1.873817e+09 1.417474e+09 1.072267e+09 8.111308e+08 6.135907e+08 4.641589e+08
 [13] 3.511192e+08 2.656088e+08 2.009233e+08 1.519911e+08 1.149757e+08 8.697490e+07
 [19] 6.579332e+07 4.977024e+07 3.764936e+07 2.848036e+07 2.154435e+07 1.629751e+07
 [25] 1.232847e+07 9.326033e+06 7.054802e+06 5.336699e+06 4.037017e+06 3.053856e+06
 [31] 2.310130e+06 1.747528e+06 1.321941e+06 1.000000e+06 7.564633e+05 5.722368e+05
 [37] 4.328761e+05 3.274549e+05 2.477076e+05 1.873817e+05 1.417474e+05 1.072267e+05
 [43] 8.111308e+04 6.135907e+04 4.641589e+04 3.511192e+04 2.656088e+04 2.009233e+04
 [49] 1.519911e+04 1.149757e+04 8.697490e+03 6.579332e+03 4.977024e+03 3.764936e+03
 [55] 2.848036e+03 2.154435e+03 1.629751e+03 1.232847e+03 9.326033e+02 7.054802e+02
 [61] 5.336699e+02 4.037017e+02 3.053856e+02 2.310130e+02 1.747528e+02 1.321941e+02
 [67] 1.000000e+02 7.564633e+01 5.722368e+01 4.328761e+01 3.274549e+01 2.477076e+01
 [73] 1.873817e+01 1.417474e+01 1.072267e+01 8.111308e+00 6.135907e+00 4.641589e+00
 [79] 3.511192e+00 2.656088e+00 2.009233e+00 1.519911e+00 1.149757e+00 8.697490e-01
 [85] 6.579332e-01 4.977024e-01 3.764936e-01 2.848036e-01 2.154435e-01 1.629751e-01
 [91] 1.232847e-01 9.326033e-02 7.054802e-02 5.336699e-02 4.037017e-02 3.053856e-02
 [97] 2.310130e-02 1.747528e-02 1.321941e-02 1.000000e-02

$cvm
  [1] 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275
 [10] 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275
 [19] 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275
 [28] 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275
 [37] 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275
 [46] 16292275 16292275 16292275 16292275 16292275 16292275 16292275 16292275 15864368
 [55] 10498288  6766489  4620457  3384432  2671091  2253898  1928279  1718171  1602904
 [64]  1536242  1498546  1475057  1454658  1442283  1430273  1415723  1413550  1409690
 [73]  1409021  1408358  1409078  1409724  1403458  1400186  1397419  1395386  1394164
 [82]  1393483  1393074  1392907  1392753  1392703  1392999  1393188  1393307  1393365
 [91]  1393390  1393400  1393400  1393392  1393379  1393363  1393345  1393326  1393308
[100]  1393290

$cvsd
  [1] 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9
  [9] 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9
 [17] 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9
 [25] 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9
 [33] 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9
 [41] 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9
 [49] 4181491.9 4181491.9 4181491.9 4181491.9 4181491.9 4243349.2 3459516.1 2353980.3
 [57] 1675434.4 1250885.3  979602.8  803433.2  678835.4  587214.5  527860.9  487399.5
 [65]  458544.7  438198.1  427126.3  419106.2  415267.4  410543.9  408472.6  403114.0
 [73]  397355.5  392564.1  388952.4  384326.3  374441.8  368037.4  363681.8  360508.6
 [81]  358188.5  356458.8  355198.8  354336.7  353670.4  353196.1  353226.8  353152.7
 [89]  353015.5  352923.3  352824.1  352726.1  352630.8  352539.1  352451.8  352369.5
 [97]  352292.3  352220.4  352153.8  352092.3

$cvup
  [1] 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767
 [10] 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767
 [19] 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767
 [28] 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767
 [37] 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767
 [46] 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20473767 20107717
 [55] 13957804  9120469  6295892  4635317  3650694  3057332  2607114  2305386  2130765
 [64]  2023642  1957091  1913255  1881784  1861389  1845540  1826267  1822022  1812804
 [73]  1806376  1800922  1798030  1794050  1777900  1768224  1761101  1755895  1752353
 [82]  1749942  1748273  1747244  1746423  1745900  1746226  1746340  1746323  1746288
 [91]  1746214  1746126  1746031  1745931  1745831  1745732  1745637  1745547  1745462
[100]  1745382

$cvlo
  [1] 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783
 [10] 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783
 [19] 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783
 [28] 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783
 [37] 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783
 [46] 12110783 12110783 12110783 12110783 12110783 12110783 12110783 12110783 11621018
 [55]  7038772  4412508  2945023  2133547  1691488  1450465  1249443  1130957  1075044
 [64]  1048843  1040001  1036859  1027531  1023177  1015006  1005179  1005077  1006576
 [73]  1011665  1015793  1020125  1025397  1029016  1032149  1033737  1034878  1035976
 [82]  1037024  1037875  1038571  1039083  1039507  1039772  1040035  1040292  1040441
 [91]  1040566  1040674  1040769  1040853  1040927  1040993  1041053  1041106  1041154
[100]  1041198

$nzero
 s0  s1  s2  s3  s4  s5  s6  s7  s8  s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20 s21 s22
  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
s23 s24 s25 s26 s27 s28 s29 s30 s31 s32 s33 s34 s35 s36 s37 s38 s39 s40 s41 s42 s43 s44 s45
```

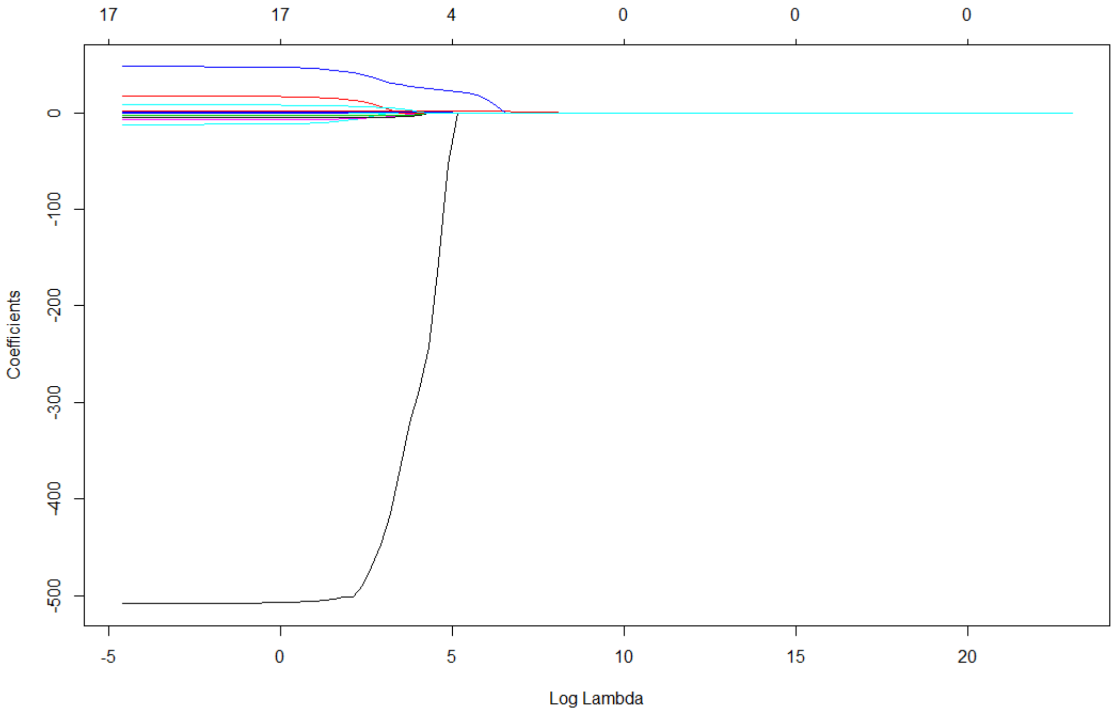
```
  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
s46 s47 s48 s49 s50 s51 s52 s53 s54 s55 s56 s57 s58 s59 s60 s61 s62 s63 s64 s65 s66 s67 s68
  0   0   0   0   0   0   0   1   1   1   1   1   1   1   2   2   3   3   3   4   5   6   9
s69 s70 s71 s72 s73 s74 s75 s76 s77 s78 s79 s80 s81 s82 s83 s84 s85 s86 s87 s88 s89 s90 s91
 10  11  12  13  14  15  16  16  16  17  17  17  17  17  17  17  17  17  17  17  17  17  17
s92 s93 s94 s95 s96 s97 s98 s99
 17  17  17  17  17  17  17  17

$name
                 mse
"Mean-Squared Error"

$glmnet.fit

Call:  glmnet(x = x[trainid, ], y = y[trainid], lambda = grid)

       Df    %Dev    Lambda
  [1,]  0 0.00000 1.000e+10
  [2,]  0 0.00000 7.565e+09
  [3,]  0 0.00000 5.722e+09
  [4,]  0 0.00000 4.329e+09
  [5,]  0 0.00000 3.275e+09
  [6,]  0 0.00000 2.477e+09
  [7,]  0 0.00000 1.874e+09
  [8,]  0 0.00000 1.417e+09
  [9,]  0 0.00000 1.072e+09
 [10,]  0 0.00000 8.111e+08
 [11,]  0 0.00000 6.136e+08
 [12,]  0 0.00000 4.642e+08
 [13,]  0 0.00000 3.511e+08
 [14,]  0 0.00000 2.656e+08
 [15,]  0 0.00000 2.009e+08
 [16,]  0 0.00000 1.520e+08
 [17,]  0 0.00000 1.150e+08
 [18,]  0 0.00000 8.697e+07
 [19,]  0 0.00000 6.579e+07
 [20,]  0 0.00000 4.977e+07
 [21,]  0 0.00000 3.765e+07
 [22,]  0 0.00000 2.848e+07
 [23,]  0 0.00000 2.154e+07
 [24,]  0 0.00000 1.630e+07
 [25,]  0 0.00000 1.233e+07
 [26,]  0 0.00000 9.326e+06
 [27,]  0 0.00000 7.055e+06
```

```
[28,]  0 0.00000 5.337e+06
[29,]  0 0.00000 4.037e+06
[30,]  0 0.00000 3.054e+06
[31,]  0 0.00000 2.310e+06
[32,]  0 0.00000 1.748e+06
[33,]  0 0.00000 1.322e+06
[34,]  0 0.00000 1.000e+06
[35,]  0 0.00000 7.565e+05
[36,]  0 0.00000 5.722e+05
[37,]  0 0.00000 4.329e+05
[38,]  0 0.00000 3.275e+05
[39,]  0 0.00000 2.477e+05
[40,]  0 0.00000 1.874e+05
[41,]  0 0.00000 1.417e+05
[42,]  0 0.00000 1.072e+05
[43,]  0 0.00000 8.111e+04
[44,]  0 0.00000 6.136e+04
[45,]  0 0.00000 4.642e+04
[46,]  0 0.00000 3.511e+04
[47,]  0 0.00000 2.656e+04
[48,]  0 0.00000 2.009e+04
[49,]  0 0.00000 1.520e+04
[50,]  0 0.00000 1.150e+04
[51,]  0 0.00000 8.697e+03
[52,]  0 0.00000 6.579e+03
[53,]  0 0.00000 4.977e+03
[54,]  1 0.02702 3.765e+03
[55,]  1 0.40180 2.848e+03
[56,]  1 0.61620 2.154e+03
[57,]  1 0.73890 1.630e+03
[58,]  1 0.80920 1.233e+03
[59,]  1 0.84930 9.326e+02
[60,]  1 0.87230 7.055e+02
[61,]  2 0.89340 5.337e+02
[62,]  2 0.90590 4.037e+02
[63,]  3 0.91320 3.054e+02
[64,]  3 0.91790 2.310e+02
[65,]  3 0.92050 1.748e+02
[66,]  4 0.92240 1.322e+02
[67,]  5 0.92390 1.000e+02
[68,]  6 0.92490 7.565e+01
[69,]  9 0.92670 5.722e+01
[70,] 10 0.92830 4.329e+01
[71,] 11 0.92990 3.275e+01
```

```
 [72,] 12 0.93140 2.477e+01
 [73,] 13 0.93250 1.874e+01
 [74,] 14 0.93310 1.417e+01
 [75,] 15 0.93350 1.072e+01
 [76,] 16 0.93380 8.111e+00
 [77,] 16 0.93400 6.136e+00
 [78,] 16 0.93420 4.642e+00
 [79,] 17 0.93420 3.511e+00
 [80,] 17 0.93430 2.656e+00
 [81,] 17 0.93430 2.009e+00
 [82,] 17 0.93430 1.520e+00
 [83,] 17 0.93430 1.150e+00
 [84,] 17 0.93440 8.697e-01
 [85,] 17 0.93440 6.579e-01
 [86,] 17 0.93440 4.977e-01
 [87,] 17 0.93440 3.765e-01
 [88,] 17 0.93440 2.848e-01
 [89,] 17 0.93440 2.154e-01
 [90,] 17 0.93440 1.630e-01
 [91,] 17 0.93440 1.233e-01
 [92,] 17 0.93440 9.326e-02
 [93,] 17 0.93440 7.055e-02
 [94,] 17 0.93440 5.337e-02
 [95,] 17 0.93440 4.037e-02
 [96,] 17 0.93440 3.054e-02
 [97,] 17 0.93440 2.310e-02
 [98,] 17 0.93440 1.748e-02
 [99,] 17 0.93440 1.322e-02
[100,] 17 0.93440 1.000e-02

$lambda.min
[1] 0.4977024

$lambda.1se
[1] 403.7017

attr(,"class")
[1] "cv.glmnet"
> cvmodel4$lambda.min
[1] 0.4977024
> signif(cvmodel4$lambda,4)
  [1] 1.000e+10 7.565e+09 5.722e+09 4.329e+09 3.275e+09 2.477e+09 1.874e+09 1.417e+09
  [9] 1.072e+09 8.111e+08 6.136e+08 4.642e+08 3.511e+08 2.656e+08 2.009e+08 1.520e+08
 [17] 1.150e+08 8.697e+07 6.579e+07 4.977e+07 3.765e+07 2.848e+07 2.154e+07 1.630e+07
 [25] 1.233e+07 9.326e+06 7.055e+06 5.337e+06 4.037e+06 3.054e+06 2.310e+06 1.748e+06
 [33] 1.322e+06 1.000e+06 7.565e+05 5.722e+05 4.329e+05 3.275e+05 2.477e+05 1.874e+05
 [41] 1.417e+05 1.072e+05 8.111e+04 6.136e+04 4.642e+04 3.511e+04 2.656e+04 2.009e+04
 [49] 1.520e+04 1.150e+04 8.697e+03 6.579e+03 4.977e+03 3.765e+03 2.848e+03 2.154e+03
 [57] 1.630e+03 1.233e+03 9.326e+02 7.055e+02 5.337e+02 4.037e+02 3.054e+02 2.310e+02
 [65] 1.748e+02 1.322e+02 1.000e+02 7.565e+01 5.722e+01 4.329e+01 3.275e+01 2.477e+01
 [73] 1.874e+01 1.417e+01 1.072e+01 8.111e+00 6.136e+00 4.642e+00 3.511e+00 2.656e+00
 [81] 2.009e+00 1.520e+00 1.150e+00 8.697e-01 6.579e-01 4.977e-01 3.765e-01 2.848e-01
 [89] 2.154e-01 1.630e-01 1.233e-01 9.326e-02 7.055e-02 5.337e-02 4.037e-02 3.054e-02
 [97] 2.310e-02 1.748e-02 1.322e-02 1.000e-02
> cvmodel4$nzero
 s0  s1  s2  s3  s4  s5  s6  s7  s8  s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20 s21 s22
  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
s23 s24 s25 s26 s27 s28 s29 s30 s31 s32 s33 s34 s35 s36 s37 s38 s39 s40 s41 s42 s43 s44 s45
  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
s46 s47 s48 s49 s50 s51 s52 s53 s54 s55 s56 s57 s58 s59 s60 s61 s62 s63 s64 s65 s66 s67 s68
  0   0   0   0   0   0   0   1   1   1   1   1   1   1   2   2   3   3   3   4   5   6   9
s69 s70 s71 s72 s73 s74 s75 s76 s77 s78 s79 s80 s81 s82 s83 s84 s85 s86 s87 s88 s89 s90 s91
 10  11  12  13  14  15  16  16  16  17  17  17  17  17  17  17  17  17  17  17  17  17  17
s92 s93 s94 s95 s96 s97 s98 s99
 17  17  17  17  17  17  17  17
```



Plot for Q4d. Click here to go back to the question.



Plot for Q4f. Click here to go back to the question.