DATA2020HW2

March 7, 2022

1 Transformations of Variables

(a)

$_1$ represents how much the mean of the dependent variable blood pressure changes given a one unit increase in the independent variable age while holding other independent variables constant.

$_2$ represents how much the mean of the dependent variable blood pressure changes given a one unit increase in the independent variable body mass index while holding other independent variables constant.

$_3$ represents how much the mean of the dependent variable blood pressure changes given that someone is pregnant while holding other independent variables constant.

$_0$ represents the expected mean value of the dependent variable blood pressure when someone is 0 years old, has a body mass index 0, and not pregnant.

(b)

$_1$ represents the average change of the dependent variable blood pressure when age minus mean of the age increases by 1 unit while holding other independent variables constant, which is equivalent to the average change in blood pressure when the age increases by 1 unit.

$_2$ represents the average change of the dependent variable blood pressure when BMI minus mean of the BMI increases by 1 unit while holding other independent variables constant, which is equivalent to the average change in blood pressure when the age increases by 1 unit.

$_3$ represents how much the mean of the dependent variable blood pressure changes given that someone is pregnant while holding other independent variables constant.

$_0$ represents the expected mean value of the dependent variable blood pressure when someone has average age, average BMI, and not pregnant. (c)

$_1$ represents a change of 1 standard deviation in age is associated with a change of $_1$ in blood pressure while holding other independent variables constant.

$_2$ represents a change of 1 standard deviation in BMI is associated with a change of $_2$ in blood pressure while holding other independent variables constant.

$_3$ represents how much the mean of the dependent variable blood pressure changes given that someone is pregnant while holding other independent variables constant.

$_0$ represents the expected mean value of the dependent variable blood pressure when someone has average age, average BMI, and not pregnant.

(d)

Since X_3 is a dummy varibale, it only takes two values: either 1 when somebody is pregnant or 0 when somebody is not pregnant. It is meaningless to center it or standardize it. For numerical variables, we typically apply log transform or standardize it. For categorical variables, we typically apply one-hot encoding.

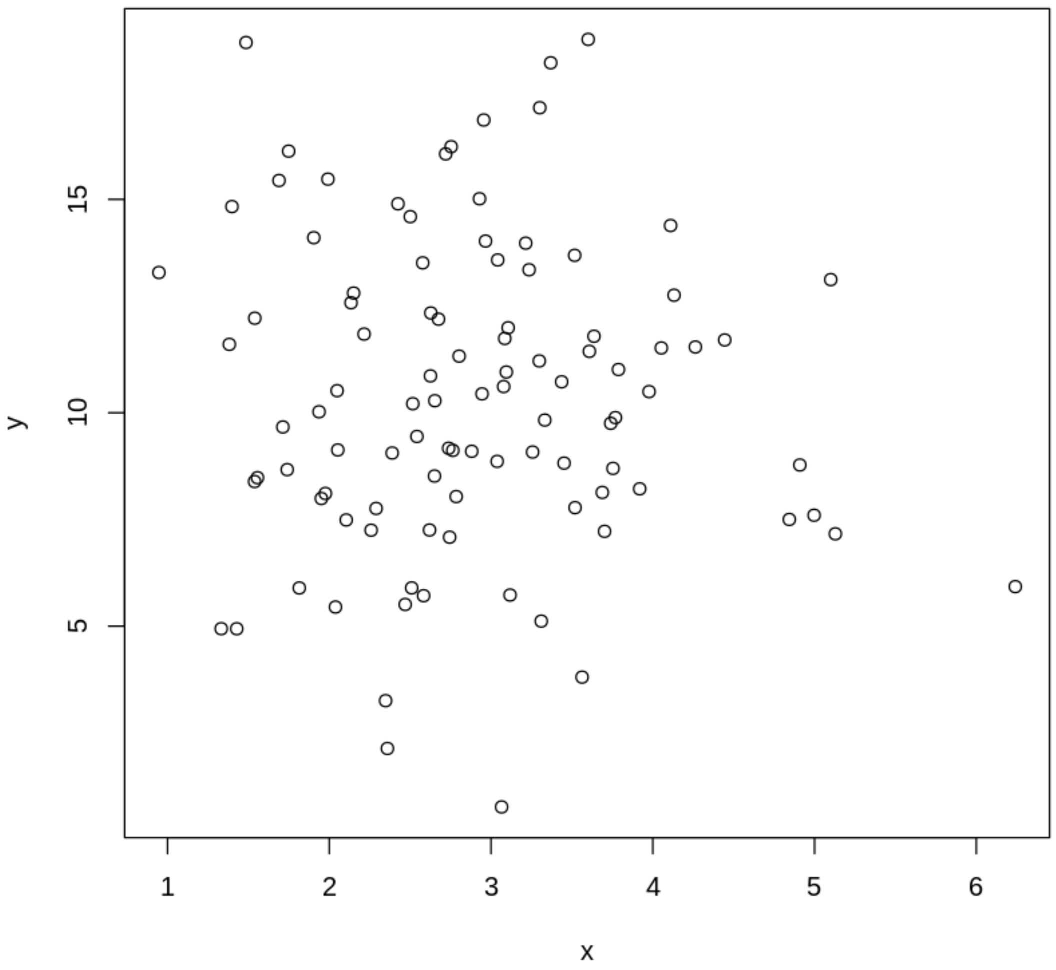
2 Simulation

(a)

[185]:

```r
set.seed(123)
y = rnorm(100, 10, 4)
x = rnorm(100, 3, 1)
model = lm(y ~ x)
summary(model)
plot(x,y)
```

```
Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max 
-9.5660 -2.3828 -0.1722  2.3689  8.5202 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   10.903      1.161   9.389 2.57e-15 ***
x             -0.187      0.381  -0.491    0.625    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.665 on 98 degrees of freedom
Multiple R-squared:  0.002453,	Adjusted R-squared:  -0.007726 
F-statistic: 0.241 on 1 and 98 DF,  p-value: 0.6246
```

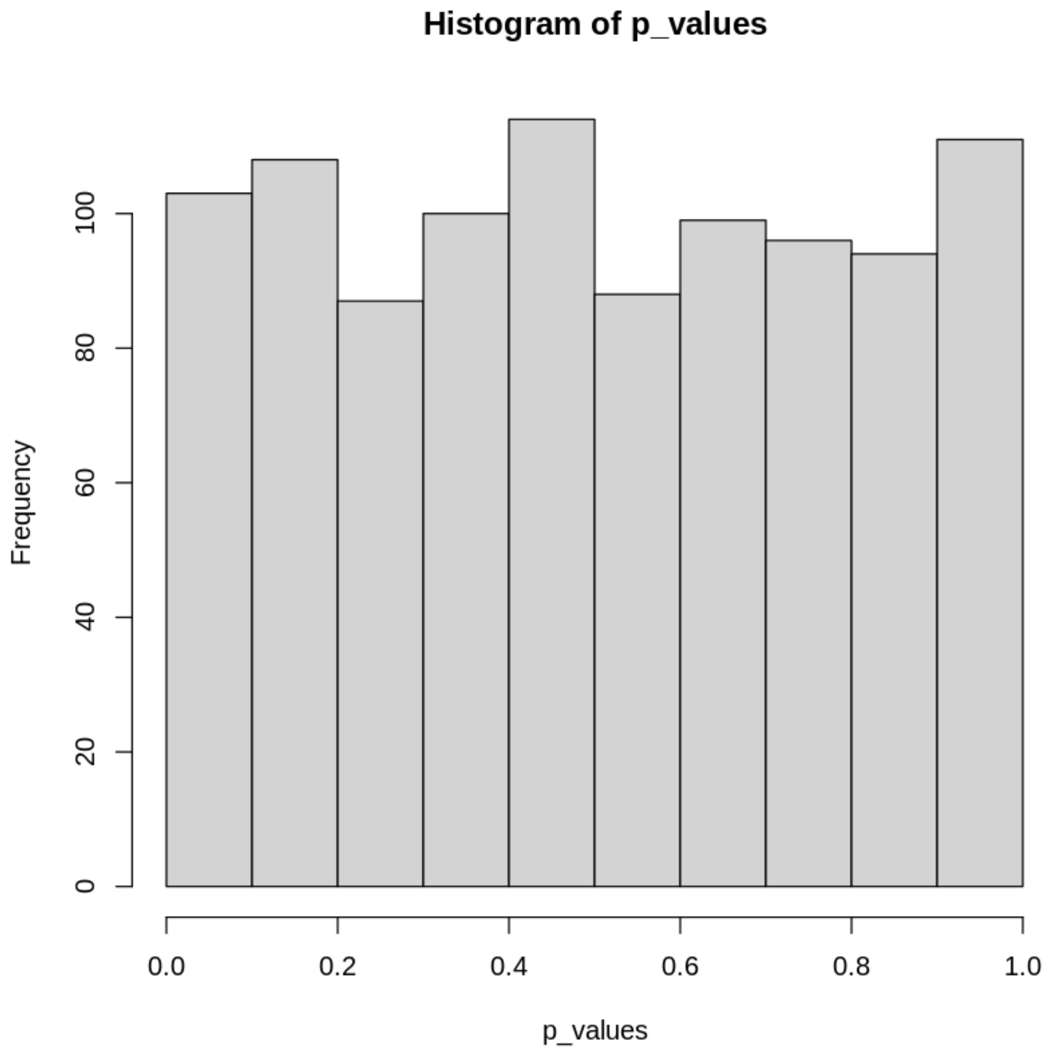


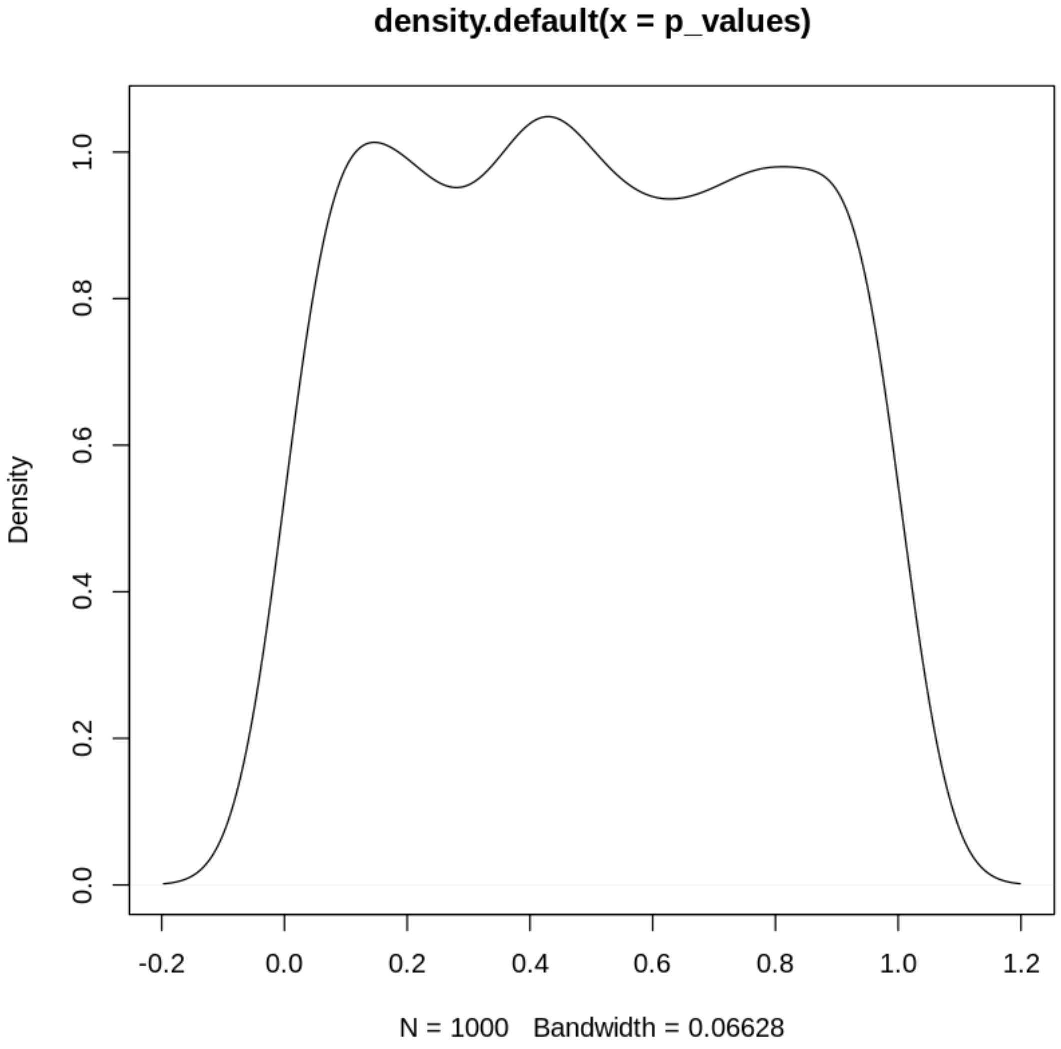
The p-value for the coefficient $_1$ is 0.625, which is really large. We failt to reject the null hypothesis that $_1 = 0$.

(b)

```
[186]: set.seed(123)
       p_values = vector()    # initialize an empty vector
       for (i in 1:1000){
         y = rnorm(100, 10, 4)
         x = rnorm(100, 3, 1)
         model = lm(y ~ x)
         p = summary(model)$coefficients[2,4]     # extract p_value for  1
         p_values = c(p_values, p)    # add each p_value to the vector
       }
```

```r
hist(p_values)
plot(density(p_values))
```





Above graphs are the distribution of the p values.

```
[187]: sum(p_values < 0.05) / 1000
```

0.053

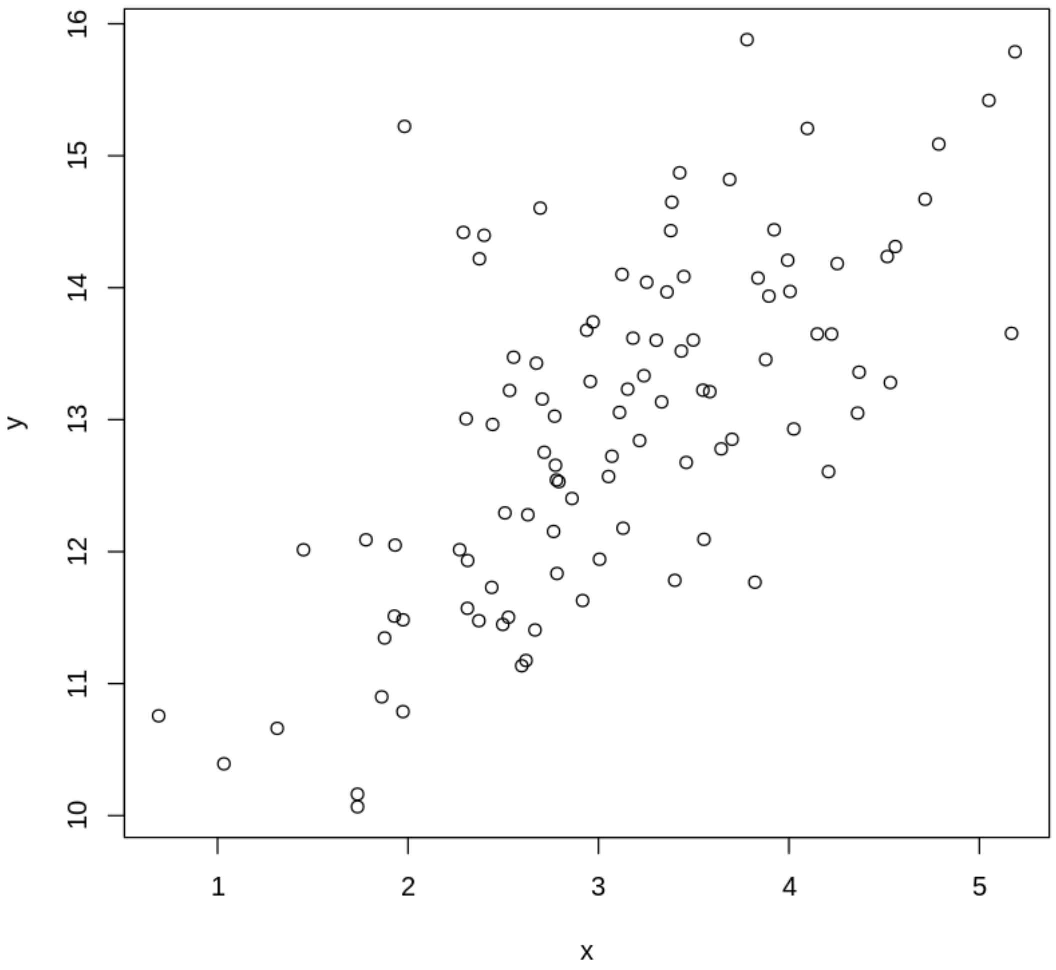
The proportion of times the p-value is less than 0.05 is 0.053. Only 5.3% of the time the estimated $_1$ is significant means that y and x does not have a linear relationship. This result matches my intuition since there is no linear relationship between y and x. If we fix the type-I error rate () to be 5%, then there are around 5% of the time when the null hypothesis is true and we reject the null. Thus, we are simulating the type-I error rate here.

(c)

[188]:

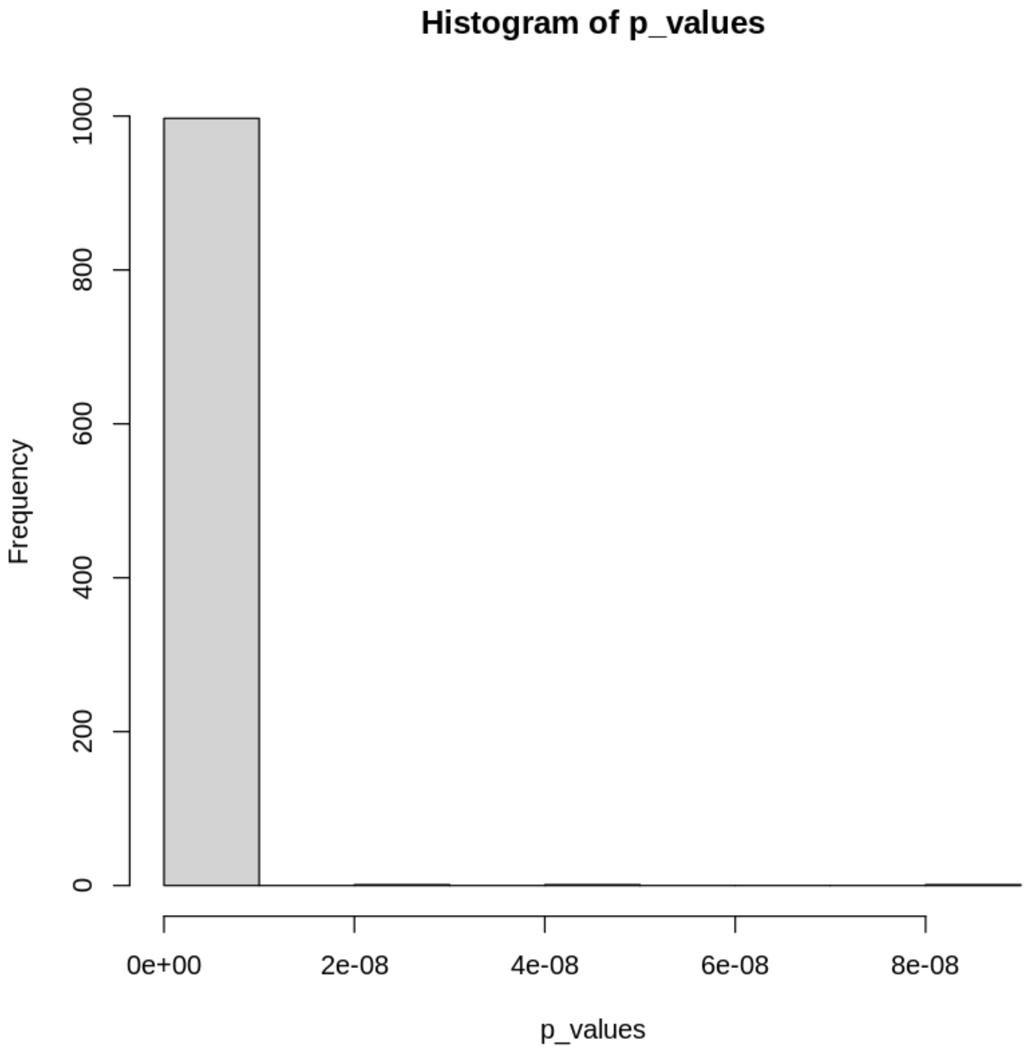
```r
set.seed(123)
x = rnorm(100, 3 ,1)
y = rnorm(100, 10+x, 1)
model = lm(y ~ x)
summary(model)
plot(x, y)
```

```

Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.9073 -0.6835 -0.0875  0.5806  3.2904 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  10.0546     0.3443  29.206  < 2e-16 ***
x             0.9475     0.1069   8.865  3.5e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9707 on 98 degrees of freedom
Multiple R-squared:  0.4451,	Adjusted R-squared:  0.4394 
F-statistic:  78.6 on 1 and 98 DF,  p-value: 3.497e-14
```



```
[189]: set.seed(123)
       p_values = vector()    # initialize an empty vector
       for (i in 1:1000){
         x = rnorm(100, 3 ,1)
         y = rnorm(100, 10+x, 1)
         model = lm(y ~ x)
         p = summary(model)$coefficients[2,4]     # extract p_value for  1
         p_values = c(p_values, p)    # add each p_value to the vector
       }

       hist(p_values)
       sum(p_values < 0.05) / 1000
```

```
1
```



The p value for $_1$ is really small here and we can reject the null hypothesis. Now, we are simulating type-II error: when the alternative hypothesis is true and we fail to reject the null. The proportion of times the p value is less than 0.05 is 1 here means that we correctly reject the null hypothesis every time for the 1000 simulations.

3 Linear Regression Application

[152]:

```r
install.packages("GGally")
install.packages("naniar")
library(GGally)
library(tidyverse)
library(ggplot2)
library(naniar)
```

```
Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)

Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)
```

[153]:

```
options(warn=-1)
```

[154]:

```
college = read.csv("college_scorecard.csv")
attach(college)
head(college)
names(college)
```

```
The following objects are masked from college (pos = 3):

    AANAPII, ACCREDAGENCY, ACTCMMID, ACTENMID, ACTMTMID, ACTWRMID,
    ADM_RATE, ANNHI, AVGFACSAL, C150_4, CCSIZSET, CCUGPROF, CITY,
    CONTROL, COSTT4_A, GRAD_DEBT_MDN, HBCU, HCM2, HIGHDEG, HSI,
    INEXPFTE, INSTNM, INSTURL, LOAN_EVER, MD_EARN_WNE_P10,
    MEDIAN_HH_INC, MENONLY, MN_EARN_WNE_P10, NANTI, NPCURL, NPT4_PRIV,
    NPT4_PUB, NUM4_PRIV, NUM4_PUB, OPEID, PAR_ED_PCT_1STGEN, PBI,
    PCIP01, PCIP03, PCIP04, PCIP05, PCIP09, PCIP10, PCIP11, PCIP12,
    PCIP13, PCIP14, PCIP15, PCIP16, PCIP19, PCIP22, PCIP23, PCIP24,
    PCIP25, PCIP26, PCIP27, PCIP29, PCIP30, PCIP31, PCIP38, PCIP39,
    PCIP40, PCIP41, PCIP42, PCIP43, PCIP44, PCIP45, PCIP46, PCIP47,
    PCIP48, PCIP49, PCIP50, PCIP51, PCIP52, PCIP54, PCTPELL, PELL_EVER,
    PFTFAC, POVERTY_RATE, PPTUG_EF, PREDDEG, REGION, RET_FT4, SAT_AVG,
    SATMTMID, SATVRMID, SATWRMID, SCH_DEG, STABBR, TRIBAL,
    TUITIONFEE_IN, TUITIONFEE_OUT, UNEMP_RATE, UNITID, WOMENONLY


The following objects are masked from college (pos = 4):

    AANAPII, ACCREDAGENCY, ACTCMMID, ACTENMID, ACTMTMID, ACTWRMID,
    ADM_RATE, ANNHI, AVGFACSAL, C150_4, CCSIZSET, CCUGPROF, CITY,
    CONTROL, COSTT4_A, GRAD_DEBT_MDN, HBCU, HCM2, HIGHDEG, HSI,
    INEXPFTE, INSTNM, INSTURL, LOAN_EVER, MD_EARN_WNE_P10,
    MEDIAN_HH_INC, MENONLY, MN_EARN_WNE_P10, NANTI, NPCURL, NPT4_PRIV,
    NPT4_PUB, NUM4_PRIV, NUM4_PUB, OPEID, PAR_ED_PCT_1STGEN, PBI,
    PCIP01, PCIP03, PCIP04, PCIP05, PCIP09, PCIP10, PCIP11, PCIP12,
    PCIP13, PCIP14, PCIP15, PCIP16, PCIP19, PCIP22, PCIP23, PCIP24,
    PCIP25, PCIP26, PCIP27, PCIP29, PCIP30, PCIP31, PCIP38, PCIP39,
    PCIP40, PCIP41, PCIP42, PCIP43, PCIP44, PCIP45, PCIP46, PCIP47,
    PCIP48, PCIP49, PCIP50, PCIP51, PCIP52, PCIP54, PCTPELL, PELL_EVER,
    PFTFAC, POVERTY_RATE, PPTUG_EF, PREDDEG, REGION, RET_FT4, SAT_AVG,
    SATMTMID, SATVRMID, SATWRMID, SCH_DEG, STABBR, TRIBAL,
    TUITIONFEE_IN, TUITIONFEE_OUT, UNEMP_RATE, UNITID, WOMENONLY


The following objects are masked from college (pos = 5):

    AANAPII, ACCREDAGENCY, ACTCMMID, ACTENMID, ACTMTMID, ACTWRMID,
    ADM_RATE, ANNHI, AVGFACSAL, C150_4, CCSIZSET, CCUGPROF, CITY,
    CONTROL, COSTT4_A, GRAD_DEBT_MDN, HBCU, HCM2, HIGHDEG, HSI,
    INEXPFTE, INSTNM, INSTURL, LOAN_EVER, MD_EARN_WNE_P10,
    MEDIAN_HH_INC, MENONLY, MN_EARN_WNE_P10, NANTI, NPCURL, NPT4_PRIV,
    NPT4_PUB, NUM4_PRIV, NUM4_PUB, OPEID, PAR_ED_PCT_1STGEN, PBI,
    PCIP01, PCIP03, PCIP04, PCIP05, PCIP09, PCIP10, PCIP11, PCIP12,
    PCIP13, PCIP14, PCIP15, PCIP16, PCIP19, PCIP22, PCIP23, PCIP24,
    PCIP25, PCIP26, PCIP27, PCIP29, PCIP30, PCIP31, PCIP38, PCIP39,
    PCIP40, PCIP41, PCIP42, PCIP43, PCIP44, PCIP45, PCIP46, PCIP47,
    PCIP48, PCIP49, PCIP50, PCIP51, PCIP52, PCIP54, PCTPELL, PELL_EVER,
    PFTFAC, POVERTY_RATE, PPTUG_EF, PREDDEG, REGION, RET_FT4, SAT_AVG,
    SATMTMID, SATVRMID, SATWRMID, SCH_DEG, STABBR, TRIBAL,
    TUITIONFEE_IN, TUITIONFEE_OUT, UNEMP_RATE, UNITID, WOMENONLY


The following objects are masked from college (pos = 7):

    AANAPII, ACCREDAGENCY, ACTCMMID, ACTENMID, ACTMTMID, ACTWRMID,
    ADM_RATE, ANNHI, AVGFACSAL, C150_4, CCSIZSET, CCUGPROF, CITY,
    CONTROL, COSTT4_A, GRAD_DEBT_MDN, HBCU, HCM2, HIGHDEG, HSI,
    INEXPFTE, INSTNM, INSTURL, LOAN_EVER, MD_EARN_WNE_P10,
    MEDIAN_HH_INC, MENONLY, MN_EARN_WNE_P10, NANTI, NPCURL, NPT4_PRIV,
    NPT4_PUB, NUM4_PRIV, NUM4_PUB, OPEID, PAR_ED_PCT_1STGEN, PBI,
    PCIP01, PCIP03, PCIP04, PCIP05, PCIP09, PCIP10, PCIP11, PCIP12,
    PCIP13, PCIP14, PCIP15, PCIP16, PCIP19, PCIP22, PCIP23, PCIP24,
    PCIP25, PCIP26, PCIP27, PCIP29, PCIP30, PCIP31, PCIP38, PCIP39,
    PCIP40, PCIP41, PCIP42, PCIP43, PCIP44, PCIP45, PCIP46, PCIP47,
    PCIP48, PCIP49, PCIP50, PCIP51, PCIP52, PCIP54, PCTPELL, PELL_EVER,
    PFTFAC, POVERTY_RATE, PPTUG_EF, PREDDEG, REGION, RET_FT4, SAT_AVG,
    SATMTMID, SATVRMID, SATWRMID, SCH_DEG, STABBR, TRIBAL,
    TUITIONFEE_IN, TUITIONFEE_OUT, UNEMP_RATE, UNITID, WOMENONLY
```

A data.frame: 6 × 95

	UNITID <int>	OPEID <int>	INSTNM <chr>	CITY <chr>	STABBR <chr>
1	100654	100200	Alabama A & M University	Normal	AL
2	100663	105200	University of Alabama at Birmingham	Birmingham	AL
3	100690	2503400	Amridge University	Montgomery	AL
4	100706	105500	University of Alabama in Huntsville	Huntsville	AL
5	100724	100500	Alabama State University	Montgomery	AL
6	100751	105100	The University of Alabama	Tuscaloosa	AL

1. 'UNITID' 2. 'OPEID' 3. 'INSTNM' 4. 'CITY' 5. 'STABBR' 6. 'ACCREDAGENCY' 7. 'INSTURL' 8. 'NPCURL' 9. 'SCH_DEG' 10. 'HCM2' 11. 'PREDDEG' 12. 'HIGHDEG' 13. 'CONTROL' 14. 'REGION' 15. 'CCUGPROF' 16. 'CCSIZSET' 17. 'HBCU' 18. 'PBI' 19. 'ANNHI' 20. 'TRIBAL' 21. 'AANAPII' 22. 'HSI' 23. 'NANTI' 24. 'MENONLY' 25. 'WOMENONLY' 26. 'ADM_RATE' 27. 'SATVRMID' 28. 'SATMTMID' 29. 'SATWRMID' 30. 'ACTCMMID' 31. 'ACTENMID' 32. 'ACTMTMID' 33. 'ACTWRMID' 34. 'SAT_AVG' 35. 'PCIP01' 36. 'PCIP03' 37. 'PCIP04' 38. 'PCIP05' 39. 'PCIP09' 40. 'PCIP10' 41. 'PCIP11' 42. 'PCIP12' 43. 'PCIP13' 44. 'PCIP14' 45. 'PCIP15' 46. 'PCIP16' 47. 'PCIP19' 48. 'PCIP22' 49. 'PCIP23' 50. 'PCIP24' 51. 'PCIP25' 52. 'PCIP26' 53. 'PCIP27' 54. 'PCIP29' 55. 'PCIP30' 56. 'PCIP31' 57. 'PCIP38' 58. 'PCIP39' 59. 'PCIP40' 60. 'PCIP41' 61. 'PCIP42' 62. 'PCIP43' 63. 'PCIP44' 64. 'PCIP45' 65. 'PCIP46' 66. 'PCIP47' 67. 'PCIP48' 68. 'PCIP49' 69. 'PCIP50' 70. 'PCIP51' 71. 'PCIP52' 72. 'PCIP54' 73. 'PPTUG_EF' 74. 'NPT4_PUB' 75. 'NPT4_PRIV' 76. 'NUM4_PUB' 77. 'NUM4_PRIV' 78. 'COSTT4_A' 79. 'TUITIONFEE_IN' 80. 'TUITIONFEE_OUT' 81. 'INEXPFTE' 82. 'AVGFACSAL' 83. 'PFTFAC' 84. 'PCTPELL' 85. 'C150_4' 86. 'RET_FT4' 87. 'PAR_ED_PCT_1STGEN' 88. 'GRAD_DEBT_MDN' 89. 'LOAN_EVER' 90. 'PELL_EVER' 91. 'MEDIAN_HH_INC' 92. 'POVERTY_RATE' 93. 'UNEMP_RATE' 94. 'MN_EARN_WNE_P10' 95. 'MD_EARN_WNE_P10'

[155]:

```
# drop useless columns
college = college %>% select(-c(UNITID, OPEID, CITY, STABBR, ACCREDAGENCY,␣
 ↪INSTURL, NPCURL, SCH_DEG, CCUGPROF, CCSIZSET))
head(college)
```

A data.frame: 6 × 85

	INSTNM <chr>	HCM2 <int>	PREDDEG <int>	HIGHDEG <int>	CONTRO <int>
1	Alabama A & M University	0	3	4	1
2	University of Alabama at Birmingham	0	3	4	1
3	Amridge University	0	2	4	2
4	University of Alabama in Huntsville	0	3	4	1
5	Alabama State University	0	3	4	1
6	The University of Alabama	0	3	4	1

[156]:

```
# all rows, columns start from 16 to the end
# 2 means apply to columns
# convert everything to numeric
college[,16:ncol(college)] <- apply(college[,16:ncol(college)], 2, as.numeric)

# all rows, columns start from 2 to 15
# 2 means apply to columns
```

```r
# convert everything to factor
college[,2:15] <- apply(college[,2:15], 2, as.factor)
```

[157]:

```r
# complete.cases(college$MD_EARN_WNE_P10) returns boolean statement
# takes all rows that do not have missing values in MD_EARN_WNE_P10 and all␣
↪columns
# 2 means apply to columns
# calculating the proportion of missing data

college = college[complete.cases(college$MD_EARN_WNE_P10),]
apply(college, 2,  function(x) sum(complete.cases(x))/nrow(college))
```

```
INSTNM 1 HCM2 1 PREDDEG 1 HIGHDEG 1 CONTROL 1 REGION 1 HBCU 1
PBI 1 ANNHI 1 TRIBAL 1 AANAPII 1 HSI 1 NANTI 1 MENONLY 1
WOMENONLY 1 ADM\_RATE 0.368681436868144 SATVRMID 0.252097225209722
SATMTMID 0.252097225209722 SATWRMID 0.151430415143042 ACTCMMID
0.259840825984083 ACTENMID 0.242632824263282 ACTMTMID 0.242632824263282
ACTWRMID 0.0660357066035707 SAT\_AVG 0.264357926435793 PCIP01
0.933318993331899 PCIP03 0.933318993331899 PCIP04 0.933318993331899 PCIP05
0.933318993331899 PCIP09 0.933318993331899 PCIP10 0.933318993331899 PCIP11
0.933318993331899 PCIP12 0.933318993331899 PCIP13 0.933318993331899 PCIP14
0.933318993331899 PCIP15 0.933318993331899 PCIP16 0.933318993331899 PCIP19
0.933318993331899 PCIP22 0.933318993331899 PCIP23 0.933318993331899 PCIP24
0.933318993331899 PCIP25 0.933318993331899 PCIP26 0.933318993331899 PCIP27
0.933318993331899 PCIP29 0.933318993331899 PCIP30 0.933318993331899 PCIP31
0.933318993331899 PCIP38 0.933318993331899 PCIP39 0.933318993331899 PCIP40
0.933318993331899 PCIP41 0.933318993331899 PCIP42 0.933318993331899 PCIP43
0.933318993331899 PCIP44 0.933318993331899 PCIP45 0.933318993331899 PCIP46
0.933318993331899 PCIP47 0.933318993331899 PCIP48 0.933318993331899 PCIP49
0.933318993331899 PCIP50 0.933318993331899 PCIP51 0.933318993331899 PCIP52
0.933318993331899 PCIP54 0.933318993331899 PPTUG\_EF 0.926865992686599
NPT4\_PUB 0.368681436868144 NPT4\_PRIV 0.524198752419875 NUM4\_PUB
0.372983437298344 NUM4\_PRIV 0.525059152505915 COSTT4\_A 0.651322865132287
TUITIONFEE\_IN 0.704882770488277 TUITIONFEE\_OUT 0.674123467412347
INEXPFTE 0.94665519466552 AVGFACSAL 0.701871370187137 PFTFAC
0.666594966659497 PCTPELL 0.927941492794149 C150\_4 0.438588943858894 RET\_FT4
0.397074639707464 PAR\_ED\_PCT\_1STGEN 0.94730049473005
GRAD\_DEBT\_MDN 0.883200688320069 LOAN\_EVER 0.856958485695849
PELL\_EVER 0.882125188212519 MEDIAN\_HH\_INC 0.890729189072919
POVERTY\_RATE 0.890729189072919 UNEMP\_RATE 0.890729189072919
MN\_EARN\_WNE\_P10 1 MD\_EARN\_WNE\_P10 1
```

We can see that columns containing admission rate, SAT, and ACT have a large portion of the data missing, so we remove them from the dataset.

[158]:

```r
college = college %>% select(-c(ADM_RATE, SATVRMID, SATMTMID, SATWRMID,␣
↪ACTCMMID, ACTENMID, ACTMTMID, ACTWRMID, SAT_AVG))
names(college)
```

1. 'INSTNM' 2. 'HCM2' 3. 'PREDDEG' 4. 'HIGHDEG' 5. 'CONTROL' 6. 'REGION' 7. 'HBCU' 8. 'PBI' 9. 'ANNHI' 10. 'TRIBAL' 11. 'AANAPII' 12. 'HSI' 13. 'NANTI' 14. 'MENONLY' 15. 'WOMENONLY' 16. 'PCIP01' 17. 'PCIP03' 18. 'PCIP04' 19. 'PCIP05' 20. 'PCIP09' 21. 'PCIP10' 22. 'PCIP11' 23. 'PCIP12' 24. 'PCIP13' 25. 'PCIP14' 26. 'PCIP15' 27. 'PCIP16' 28. 'PCIP19' 29. 'PCIP22' 30. 'PCIP23' 31. 'PCIP24' 32. 'PCIP25' 33. 'PCIP26' 34. 'PCIP27' 35. 'PCIP29' 36. 'PCIP30' 37. 'PCIP31' 38. 'PCIP38' 39. 'PCIP39' 40. 'PCIP40' 41. 'PCIP41' 42. 'PCIP42' 43. 'PCIP43' 44. 'PCIP44' 45. 'PCIP45' 46. 'PCIP46' 47. 'PCIP47' 48. 'PCIP48' 49. 'PCIP49' 50. 'PCIP50' 51. 'PCIP51' 52. 'PCIP52' 53. 'PCIP54' 54. 'PPTUG_EF' 55. 'NPT4_PUB' 56. 'NPT4_PRIV' 57. 'NUM4_PUB' 58. 'NUM4_PRIV' 59. 'COSTT4_A' 60. 'TUITIONFEE_IN' 61. 'TUITIONFEE_OUT' 62. 'INEXPFTE' 63. 'AVGFACSAL' 64. 'PFTFAC' 65. 'PCTPELL' 66. 'C150_4' 67. 'RET_FT4' 68. 'PAR_ED_PCT_1STGEN' 69. 'GRAD_DEBT_MDN' 70. 'LOAN_EVER' 71. 'PELL_EVER' 72. 'MEDIAN_HH_INC' 73. 'POVERTY_RATE' 74. 'UNEMP_RATE' 75. 'MN_EARN_WNE_P10' 76. 'MD_EARN_WNE_P10'

Then, we can combine columns NPT4_PUB and NPT4_PRIV, NUM4_PUB and NUM4_PRIV to a single column because they code public and private school separately.

[159]:

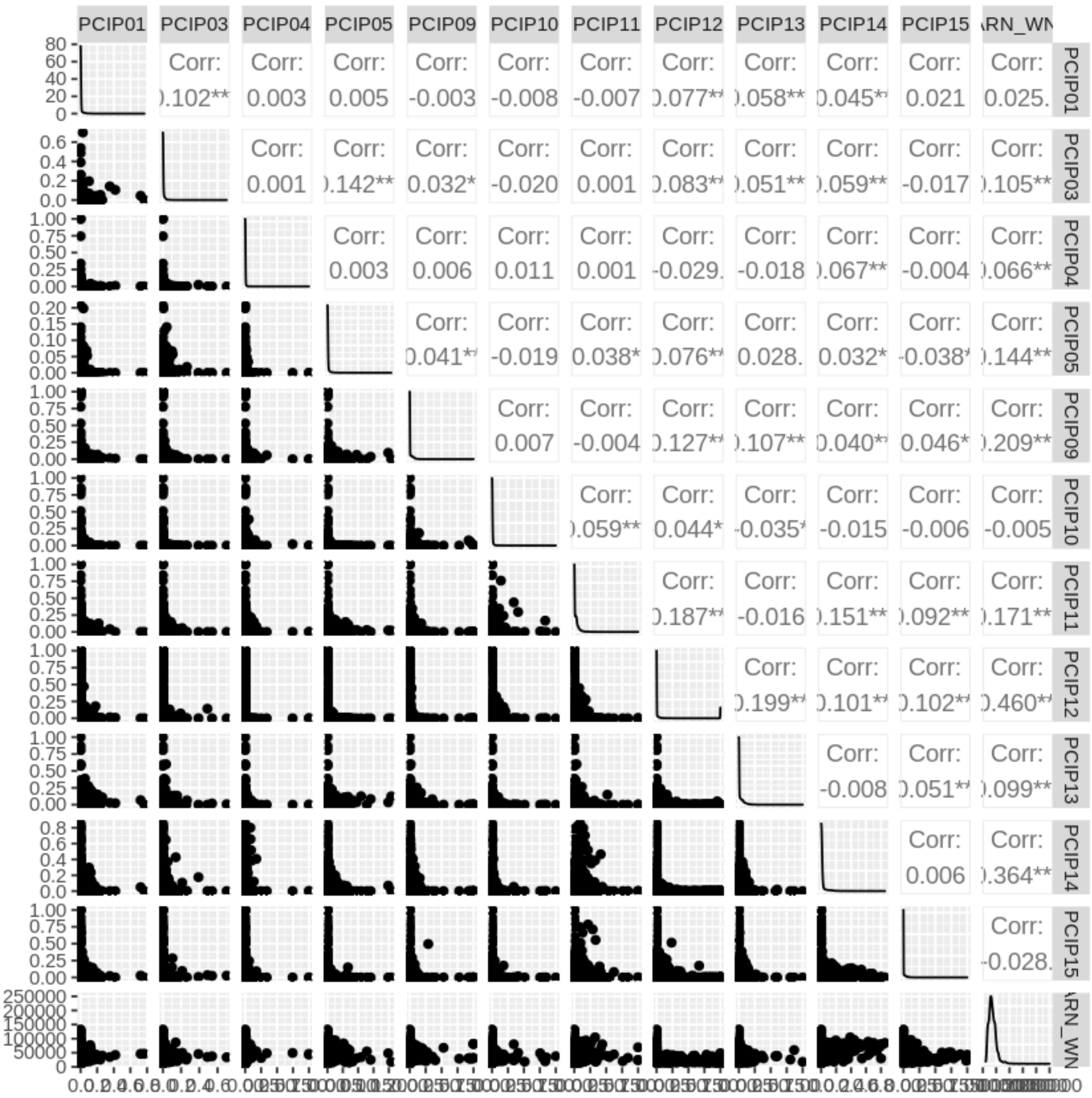
```
# is.na(college_df$NPT4_PRIV) contains TRUE or FALSE values
# college_df$NPT4_PUB: what to return if test is TRUE
# college_df$NPT4_PRIVL: what to return if test is FALSE

college$NPT <- ifelse(is.na(college$NPT4_PRIV),
                         college$NPT4_PUB,
                         college$NPT4_PRIV)
college$NUM <- ifelse(is.na(college$NUM4_PRIV),
                         college$NUM4_PUB,
                         college$NUM4_PRIV)
```

[160]:

```
y = college$MD_EARN_WNE_P10
college = college %>% select(-c(NPT4_PUB, NPT4_PRIV, NUM4_PUB, NUM4_PRIV))
```

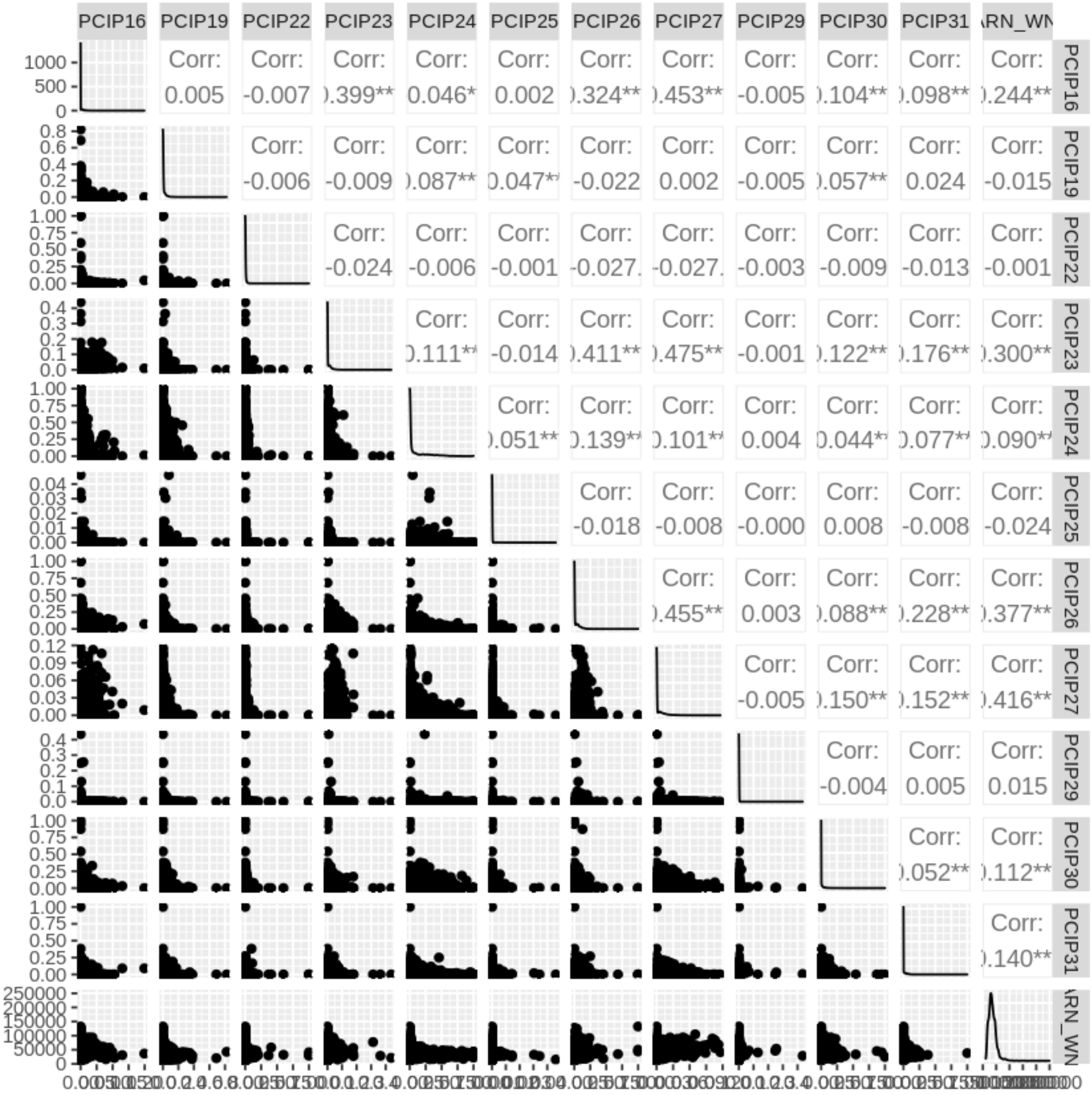
[161]:

```
ggpairs(college[,c(16:26, 72)])
```



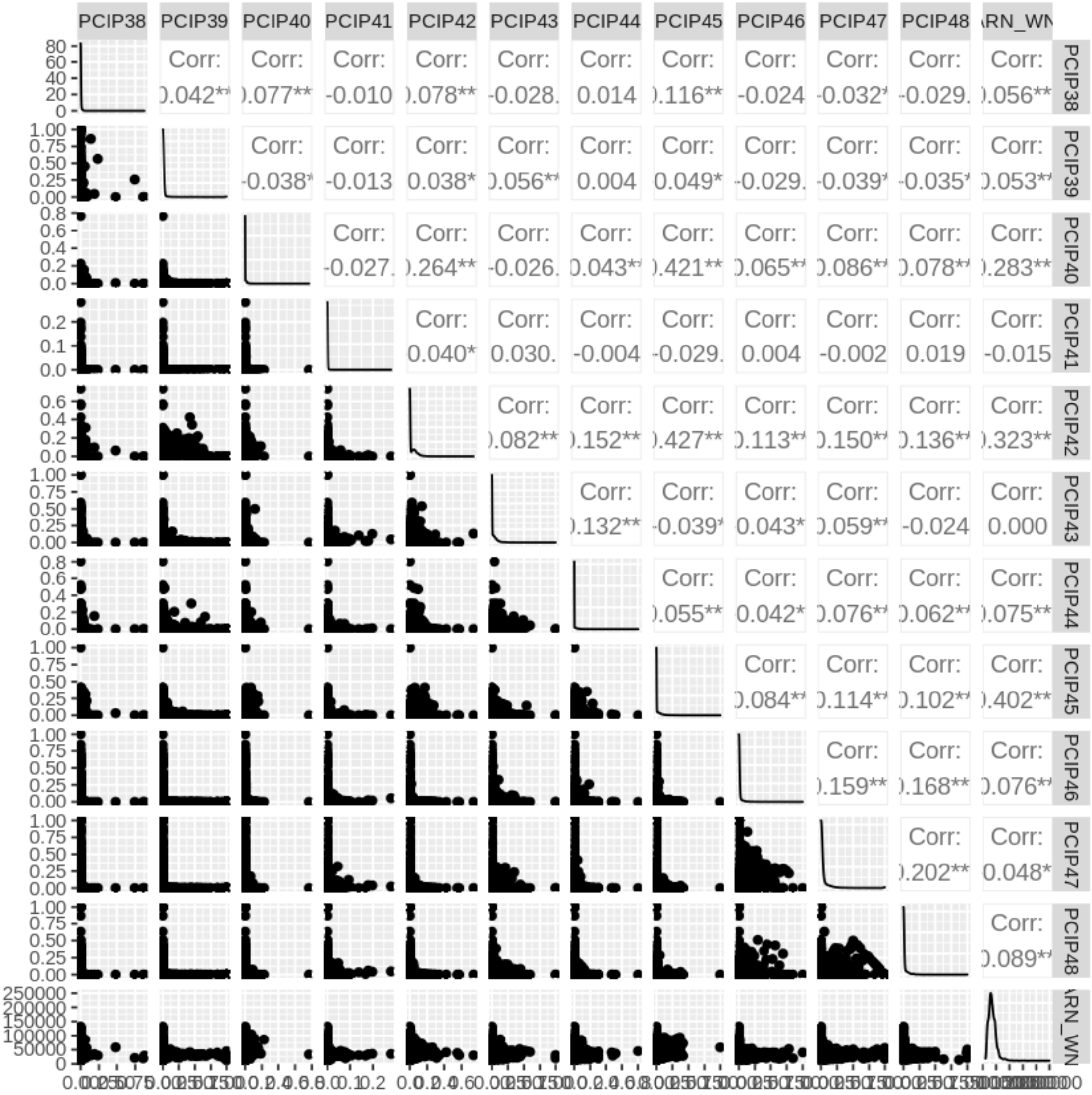
From above scatterplot matrix, we can see that no variable has a relatively strong correlation with the target variable.

```
[162]: ggpairs(college[,c(27:37, 72)])
```



From above scatterplot matrix, we can see that no variable has a relatively strong correlation with the target variable.

```
[163]: ggpairs(college[,c(38:48, 72)])
```



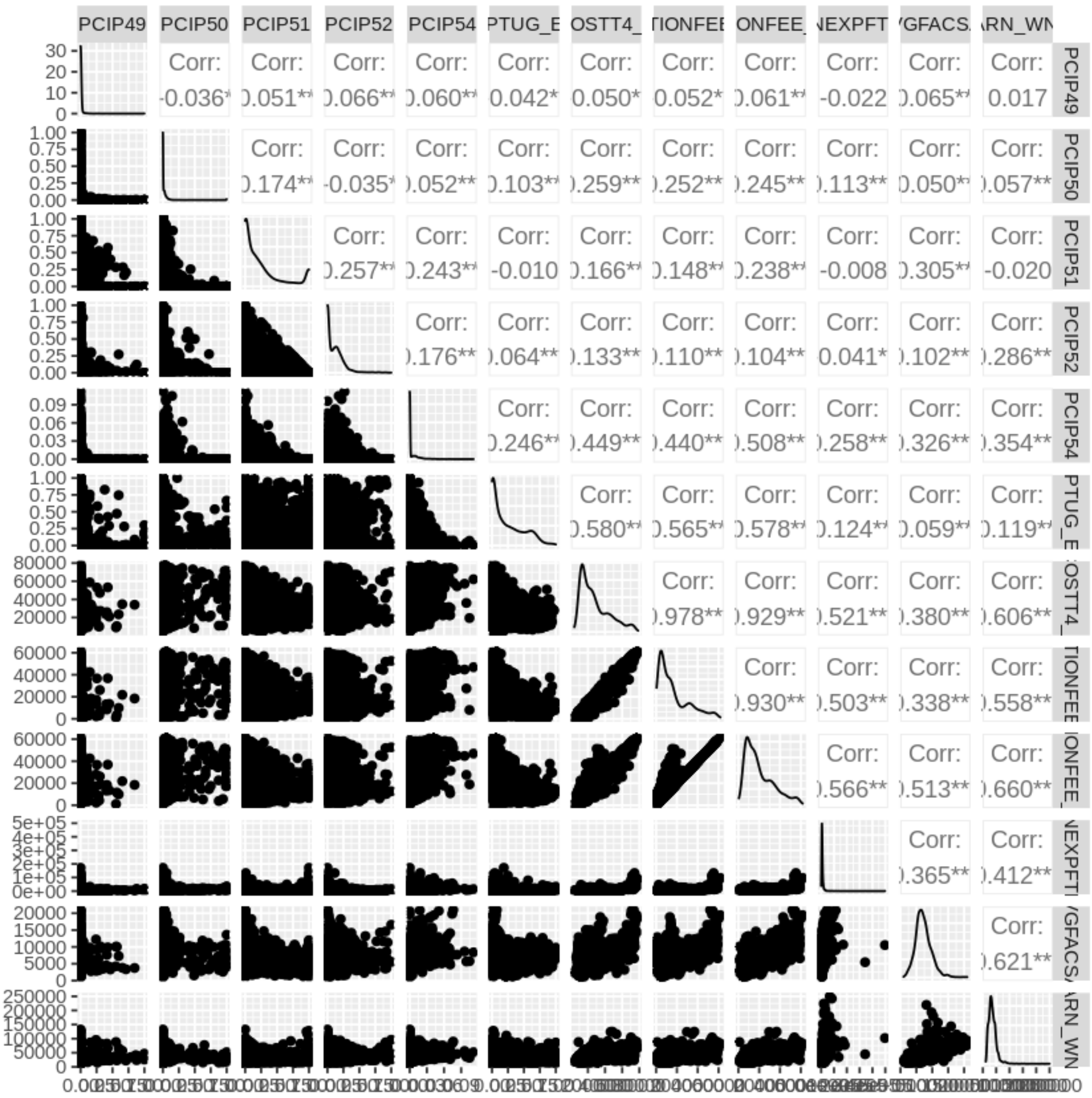
[164]:

```

```

From above scatterplot matrix, we can see that no variable has a relatively strong correlation with the target variable.

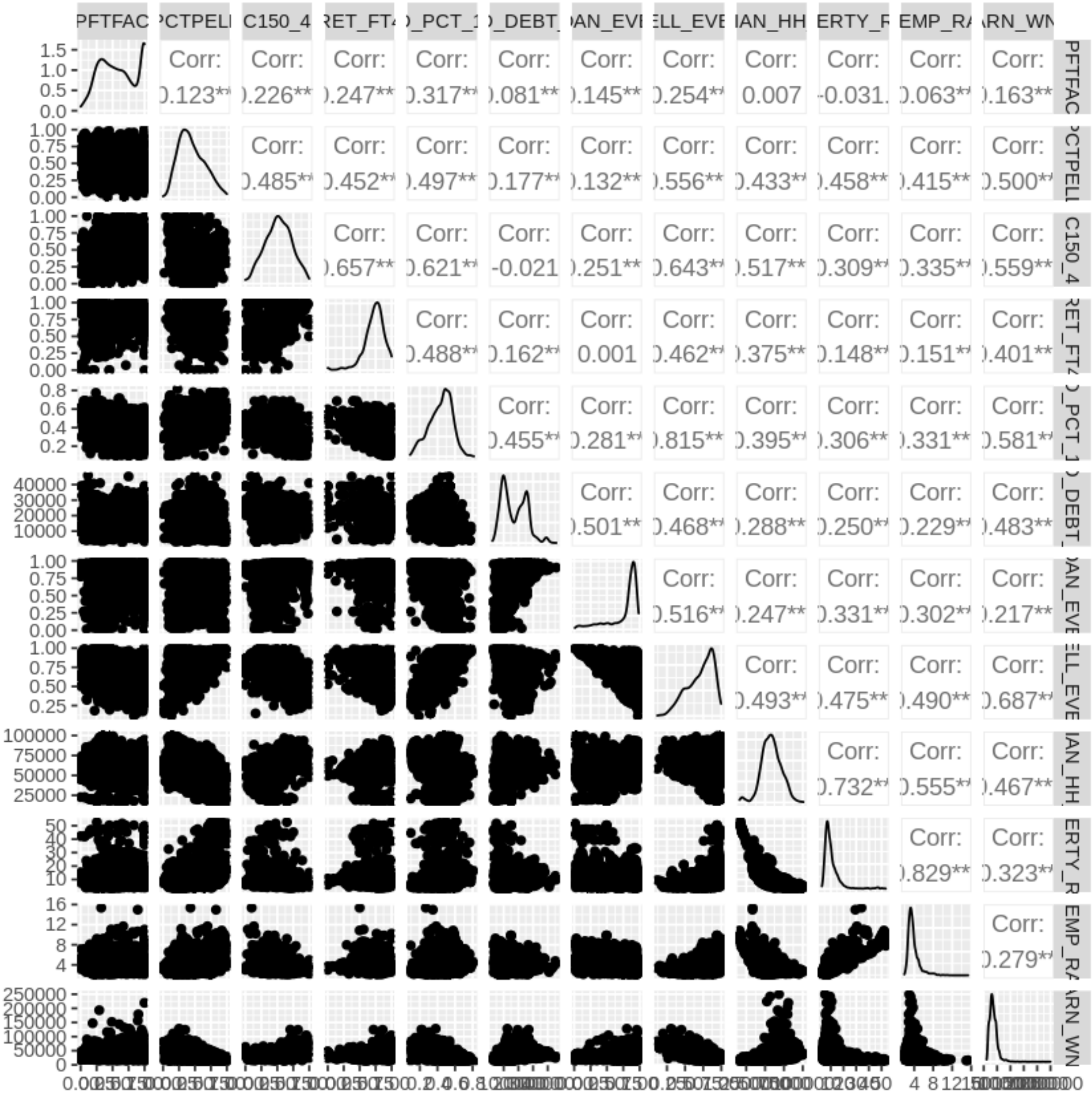
[165]:

```
ggpairs(college[,c(49:59, 72)])
```



From above scatterplot matrix, we can see that COSTT4_A, TUITIONFEE_IN, TUITIONFEE_OUT, and AVGFACSAL have a relatively strong correlation with the target variable.

```
[166]: ggpairs(college[,c(60:70, 72)])
```

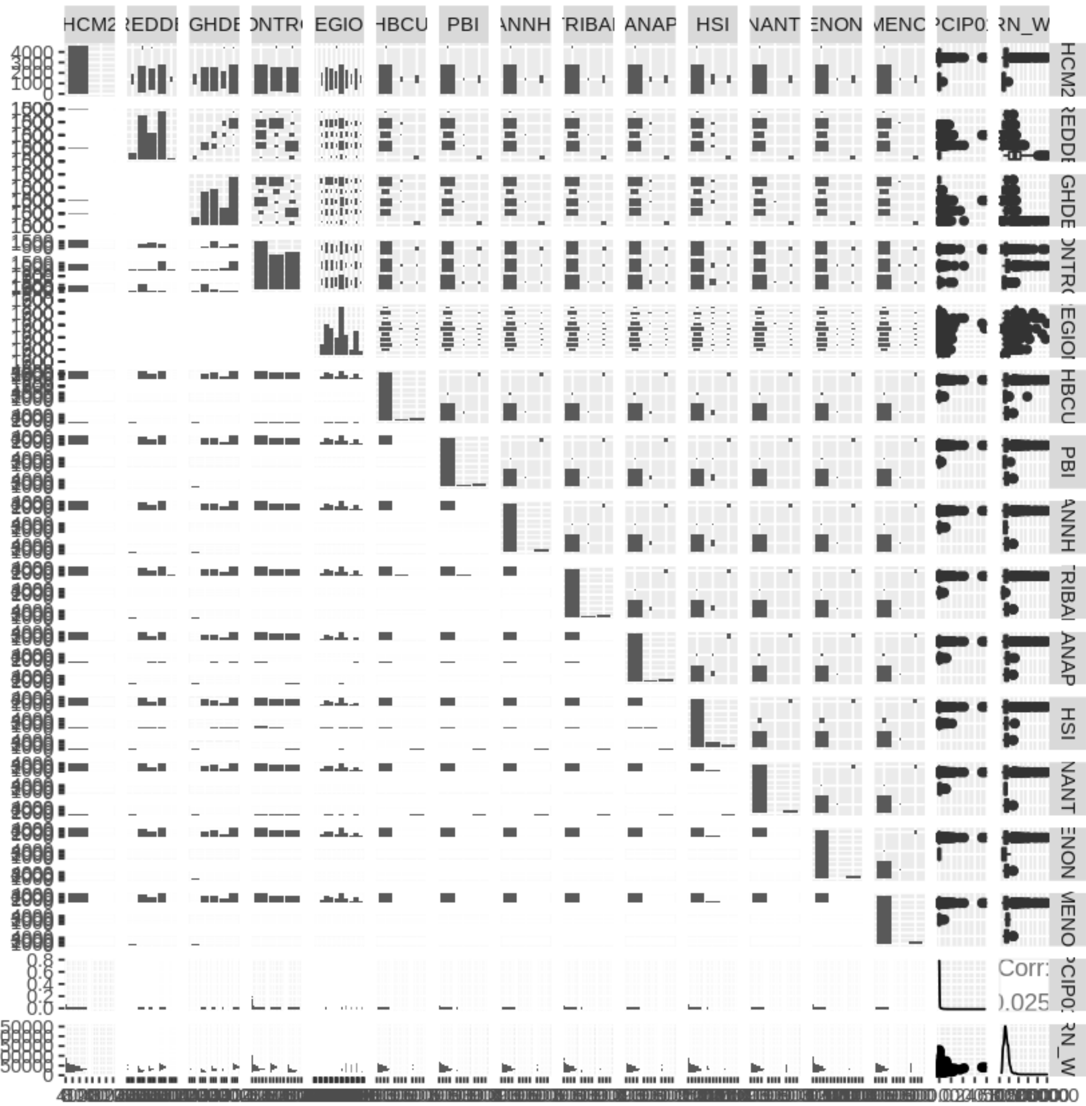


From above scatterplot matrix, we can see that PCTPELL, C150_4, PAR_ED_PCT_1STGEN, and PELL_EVER have a relatively strong correlation with the target variable.

[167]:
```
ggpairs(college[,c(2:16, 72)])
```

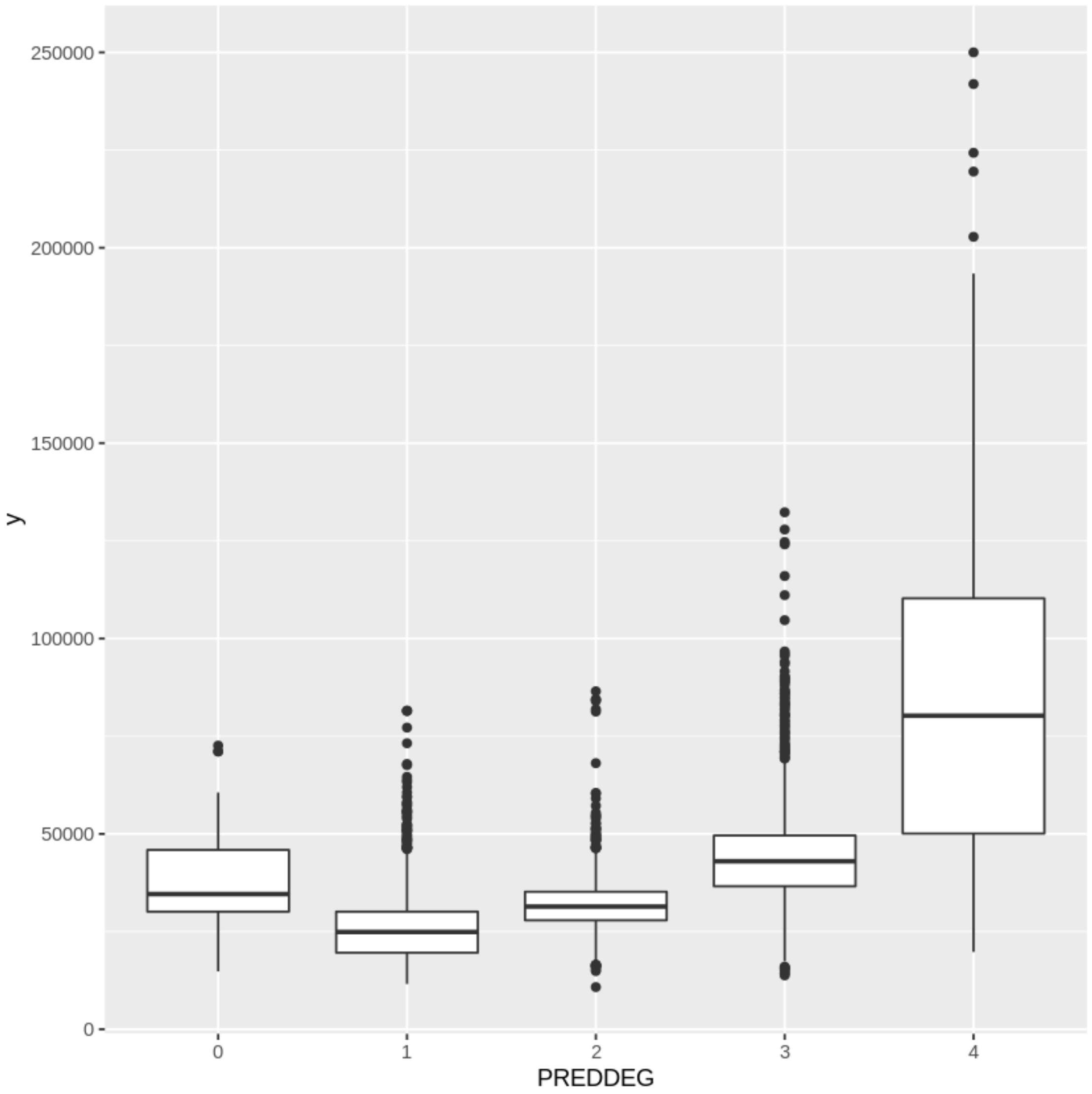
```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

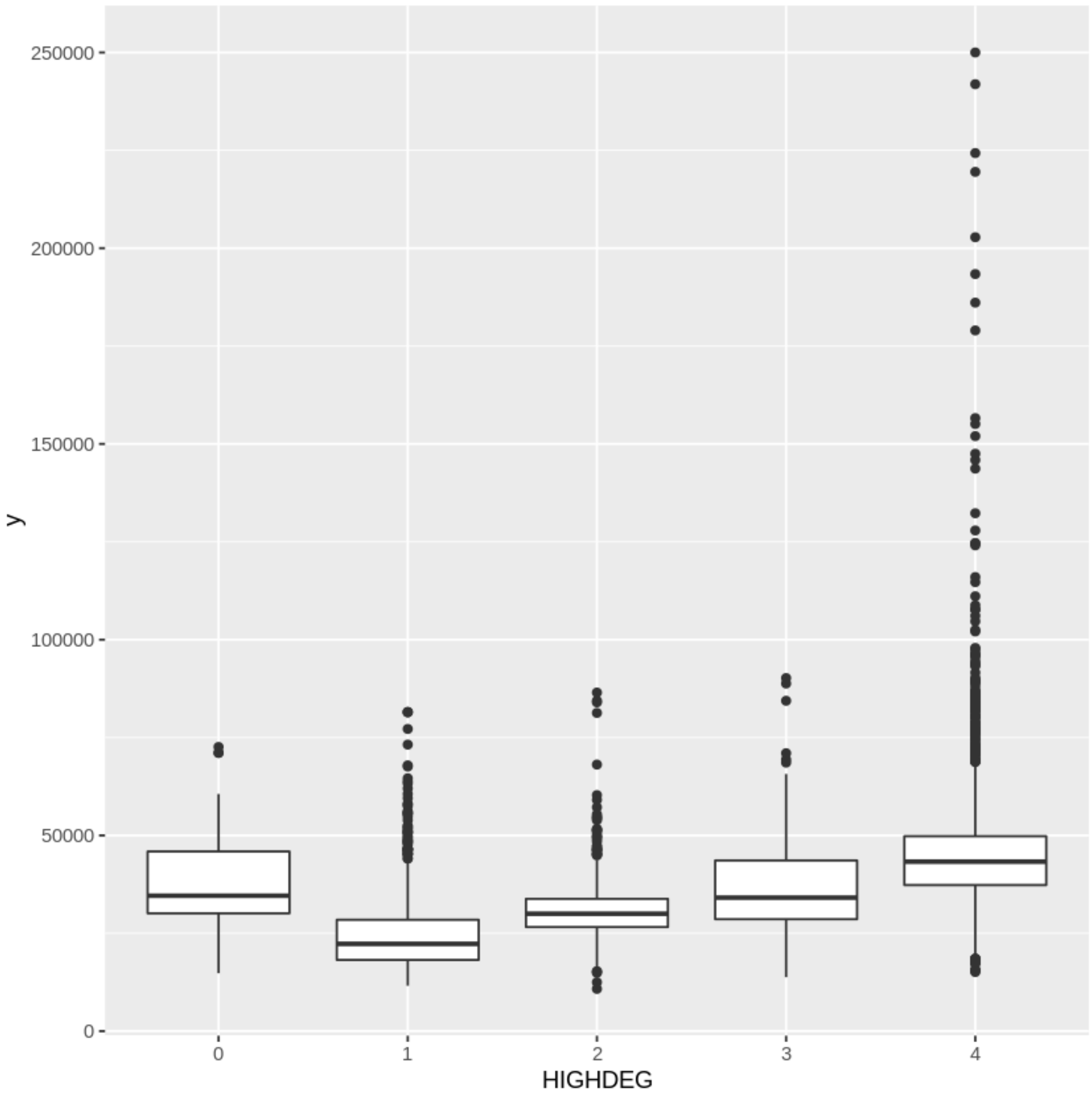
```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

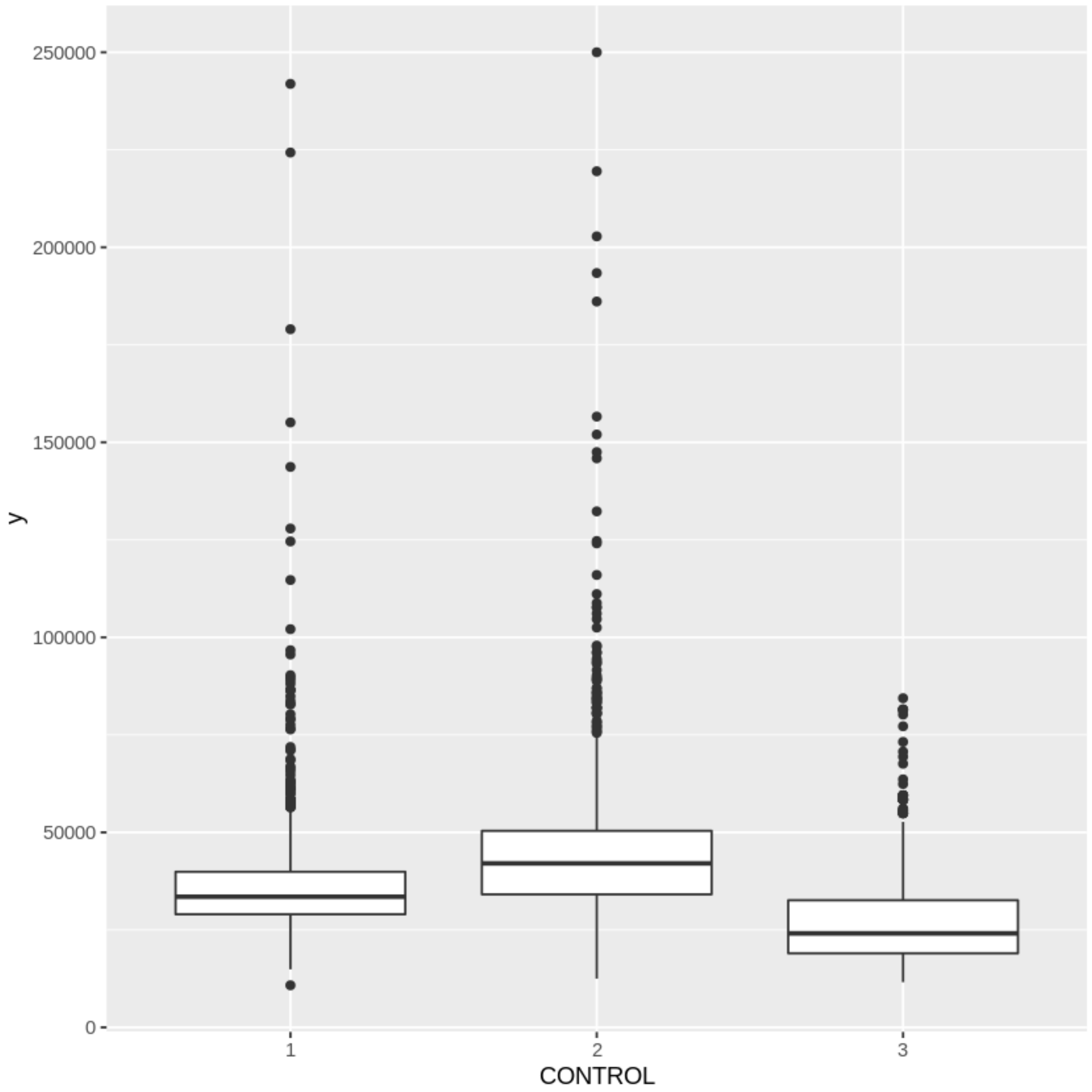


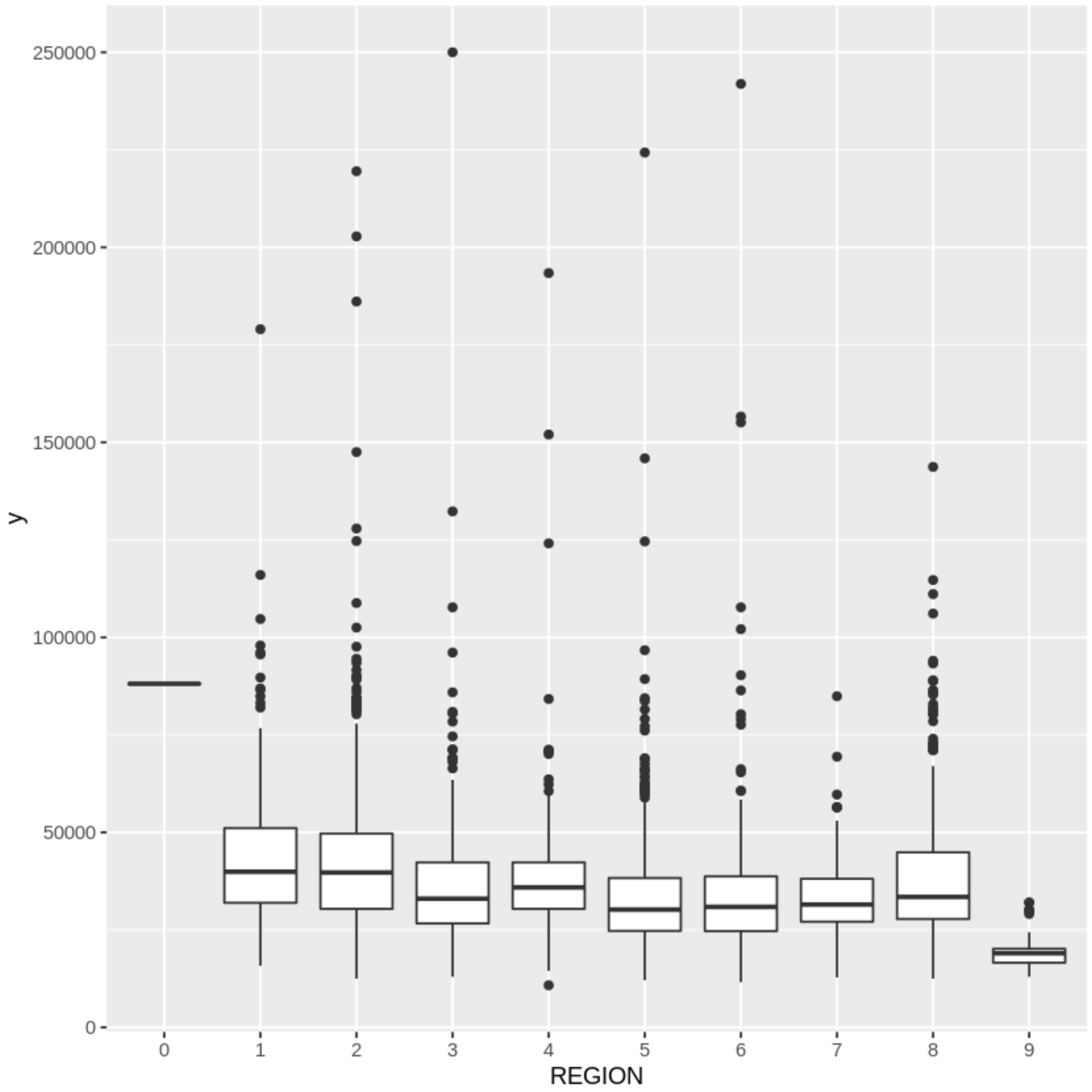
[168]:

```
ggplot(data = college) + geom_boxplot(aes(x = PREDDEG, y = y))
ggplot(data = college) + geom_boxplot(aes(x = HIGHDEG, y = y))
ggplot(data = college) + geom_boxplot(aes(x = CONTROL, y = y))
ggplot(data = college) + geom_boxplot(aes(x = REGION, y = y))
```


250000
200000
150000
100000
50000
0
y
0
1
2
3
4
PREDDEG


250000
200000
150000
100000
50000
0
y
0
1
2
3
4
HIGHDEG


250000
200000
150000
100000
50000
0
y
1
2
3
CONTROL



Now, we are experimenting the relationship between the dummy variables and the target variable. From above boxplot matrix, we can see that independent variables such as PREDDEG and HIGHDEG have a relatively signficiant correlation with the target variable.

[169]:
```
install.packages("leaps")
library(leaps)
```

```
Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)
```

[170]:
```
model1 <- regsubsets(y~., data = college[,c(3, 4, 55, 56, 57, 59, 61, 62, 64,␣
↪67)], nvmax = 10)
summary(model1)
```

```
Subset selection object
Call: regsubsets.formula(y ~ ., data = college[, c(3, 4, 55, 56, 57,
    59, 61, 62, 64, 67)], nvmax = 10)
12 Variables  (and intercept)
                  Forced in Forced out
PREDDEG2              FALSE      FALSE
PREDDEG3              FALSE      FALSE
HIGHDEG3              FALSE      FALSE
HIGHDEG4              FALSE      FALSE
COSTT4_A              FALSE      FALSE
TUITIONFEE_IN         FALSE      FALSE
TUITIONFEE_OUT        FALSE      FALSE
AVGFACSAL             FALSE      FALSE
PCTPELL               FALSE      FALSE
C150_4                FALSE      FALSE
PAR_ED_PCT_1STGEN     FALSE      FALSE
PELL_EVER             FALSE      FALSE
1 subsets of each size up to 10
Selection Algorithm: exhaustive
          PREDDEG2 PREDDEG3 HIGHDEG3 HIGHDEG4 COSTT4_A TUITIONFEE_IN
1  ( 1 )  " "      " "      " "      " "      " "      " "
2  ( 1 )  " "      " "      " "      " "      " "      " "
3  ( 1 )  " "      " "      " "      " "      " "      " "
4  ( 1 )  " "      " "      " "      " "      "*"      " "
5  ( 1 )  " "      " "      " "      " "      "*"      " "
6  ( 1 )  " "      "*"      " "      " "      "*"      " "
7  ( 1 )  " "      "*"      " "      " "      "*"      " "
8  ( 1 )  "*"      "*"      " "      " "      "*"      " "
9  ( 1 )  "*"      "*"      " "      " "      "*"      " "
10  ( 1 ) "*"      "*"      " "      " "      "*"      "*"
          TUITIONFEE_OUT AVGFACSAL PCTPELL C150_4 PAR_ED_PCT_1STGEN PELL_EVER
1  ( 1 )  " "            "*"       " "     " "    " "               " "
2  ( 1 )  " "            "*"       " "     " "    " "               "*"
3  ( 1 )  " "            "*"       " "     " "    "*"               "*"
4  ( 1 )  " "            "*"       " "     " "    "*"               "*"
5  ( 1 )  " "            "*"       " "     "*"    "*"               "*"
6  ( 1 )  " "            "*"       " "     "*"    "*"               "*"
7  ( 1 )  "*"            "*"       " "     "*"    "*"               "*"
8  ( 1 )  "*"            "*"       " "     "*"    "*"               "*"
9  ( 1 )  "*"            "*"       "*"     "*"    "*"               "*"
10  ( 1 ) "*"            "*"       "*"     "*"    "*"               "*"
```

[171]:

```
scores = summary(model1)
data.frame(
  Adj.R2 = which.max(scores$adjr2),
  CP = which.min(scores$cp),
  BIC = which.min(scores$bic)
)
```

A data.frame: 1 × 3

Adj.R2	CP	BIC
<int>	<int>	<int>
10	9	5

By running the best subset regression, the model with the highest adjusted R squared value, lowest Mallows's Cp value, and lowest BIC are model 10, model 9, and model 5 respectively. We will choose the model chosen by adjusted R squared as our model since it measures the percentage of variance in the target variable that is explained by the independent variables. Thus, the predictors for our model includes: PREDDEG, COSTT4_A, TUITIONFEE_IN, TUITIONFEE_OUT, AVGFACSAL, PCTPELL, C150_4, PAR_ED_PCT_1STGEN, and PELL_EVER.

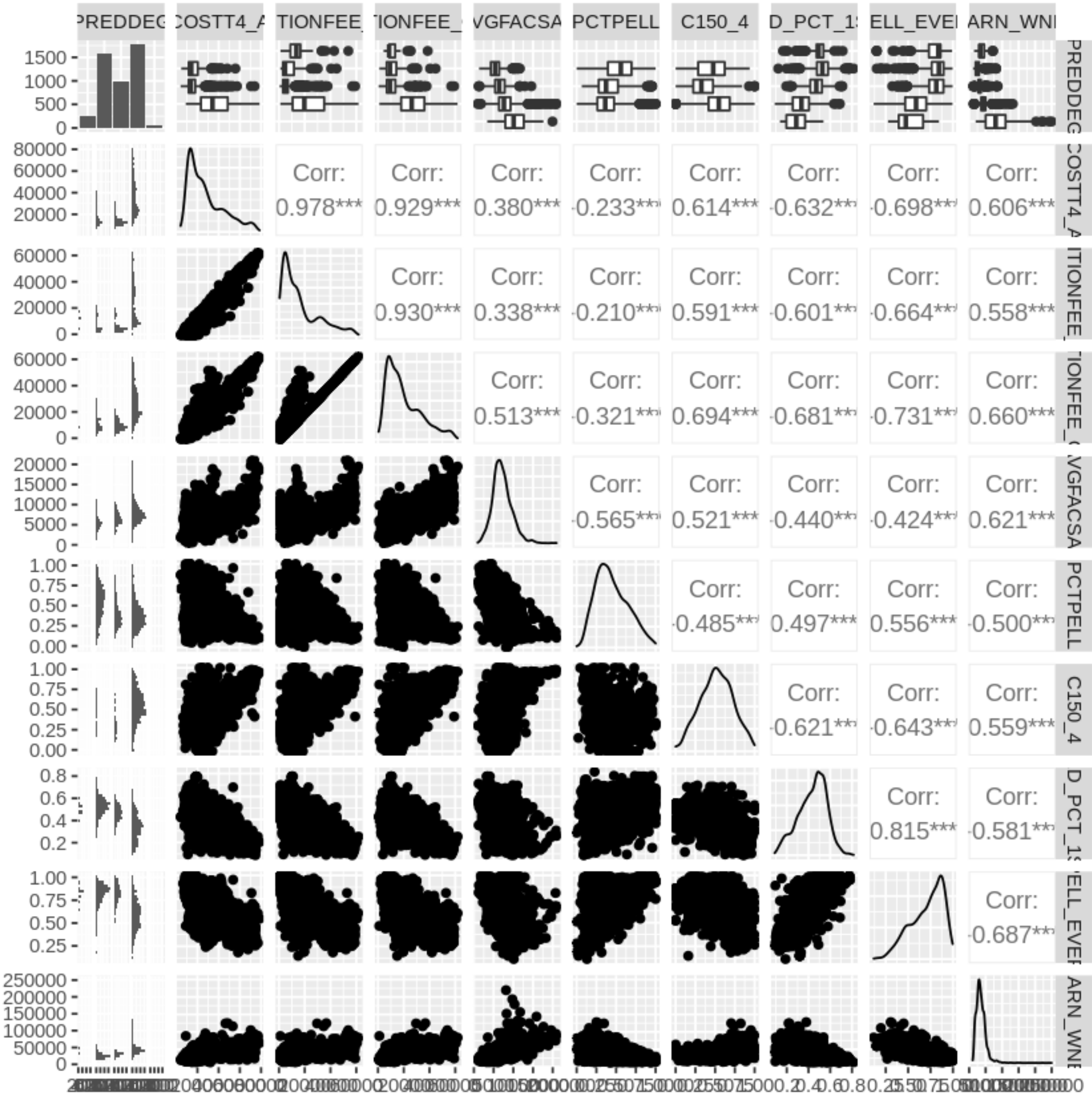
[172]:

```
grep("PREDDEG", colnames(college))
grep("COSTT4_A", colnames(college))
grep("TUITIONFEE_IN", colnames(college))
grep("TUITIONFEE_OUT", colnames(college))
grep("AVGFACSAL", colnames(college))
grep("PCTPELL", colnames(college))
grep("C150_4", colnames(college))
grep("PAR_ED_PCT_1STGEN", colnames(college))
grep("PELL_EVER", colnames(college))
ggpairs(college[,c(3,55,56,57,59,61,62,64,67, 72)])
```

```
3

55

56

57

59

61

62

64

67
```

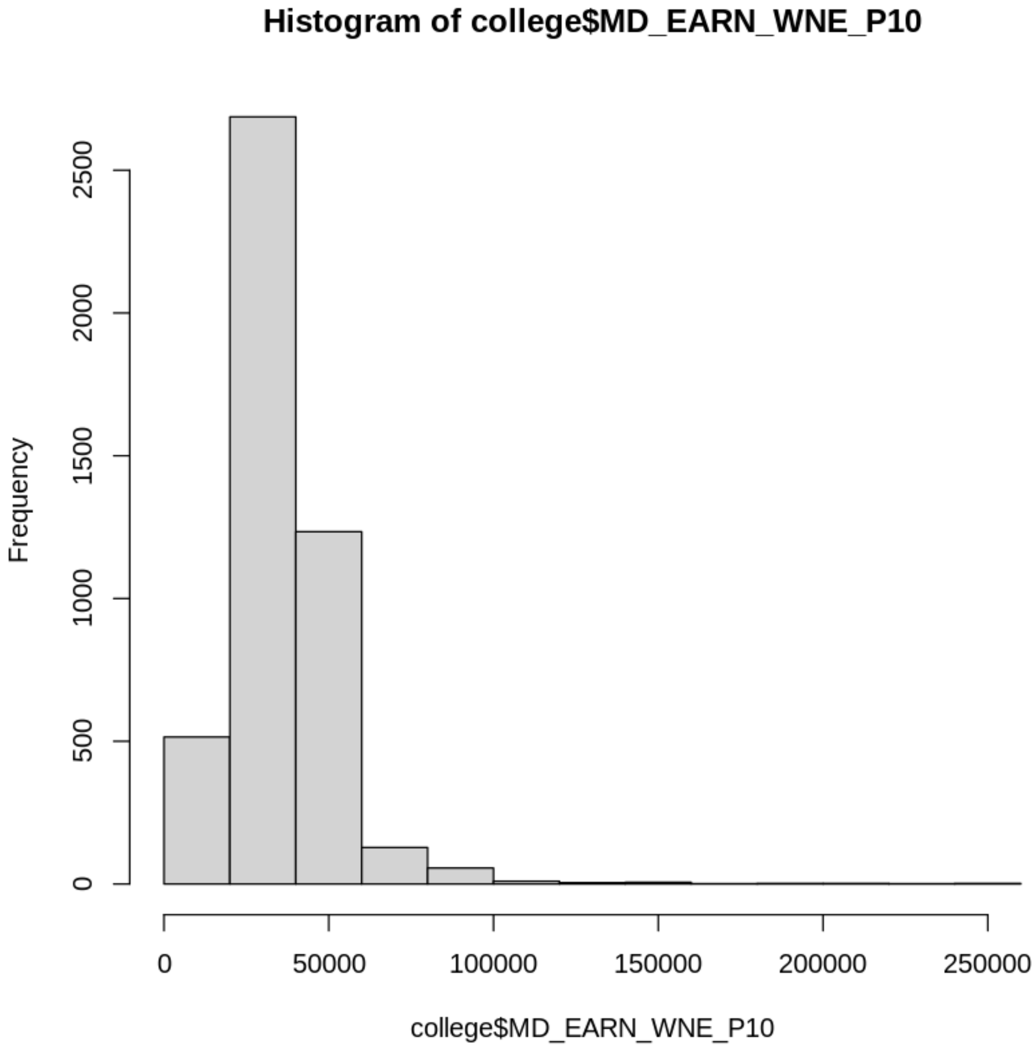
```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

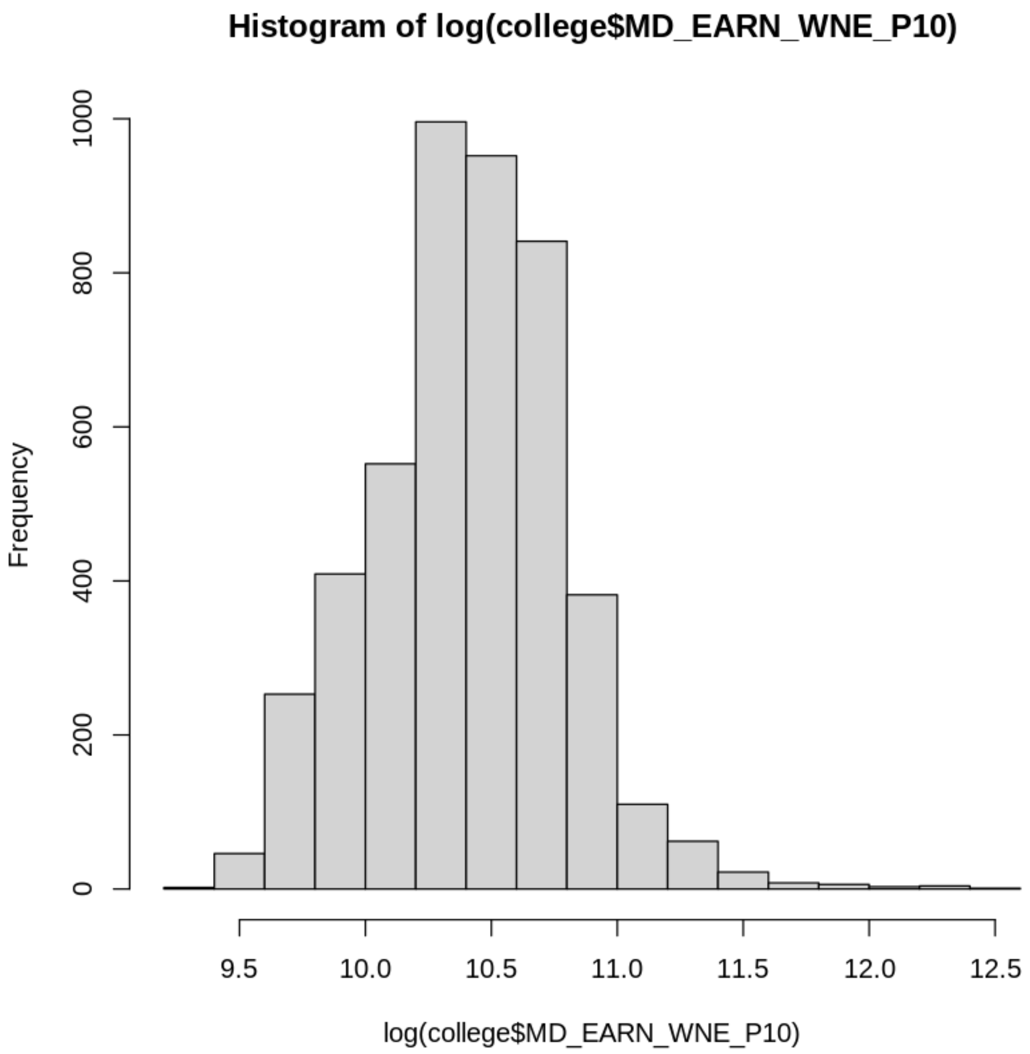
```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Since TUITIONFEE_IN and TUITIONFEE_OUT have strong correlation, we drop them from our model.

```
[173]: hist(college$MD_EARN_WNE_P10)
       hist(log(college$MD_EARN_WNE_P10))
```


Histogram of college$MD_EARN_WNE_P10
Frequency
0
500
1000
1500
2000
2500
0
50000
100000
150000
200000
250000
college$MD_EARN_WNE_P10



From the first graph, we can see that the dependent variable "MD_EARN_WNE_P10" is right-skewed. After applying log transform (all the values must be positive), the shape is more ideal.

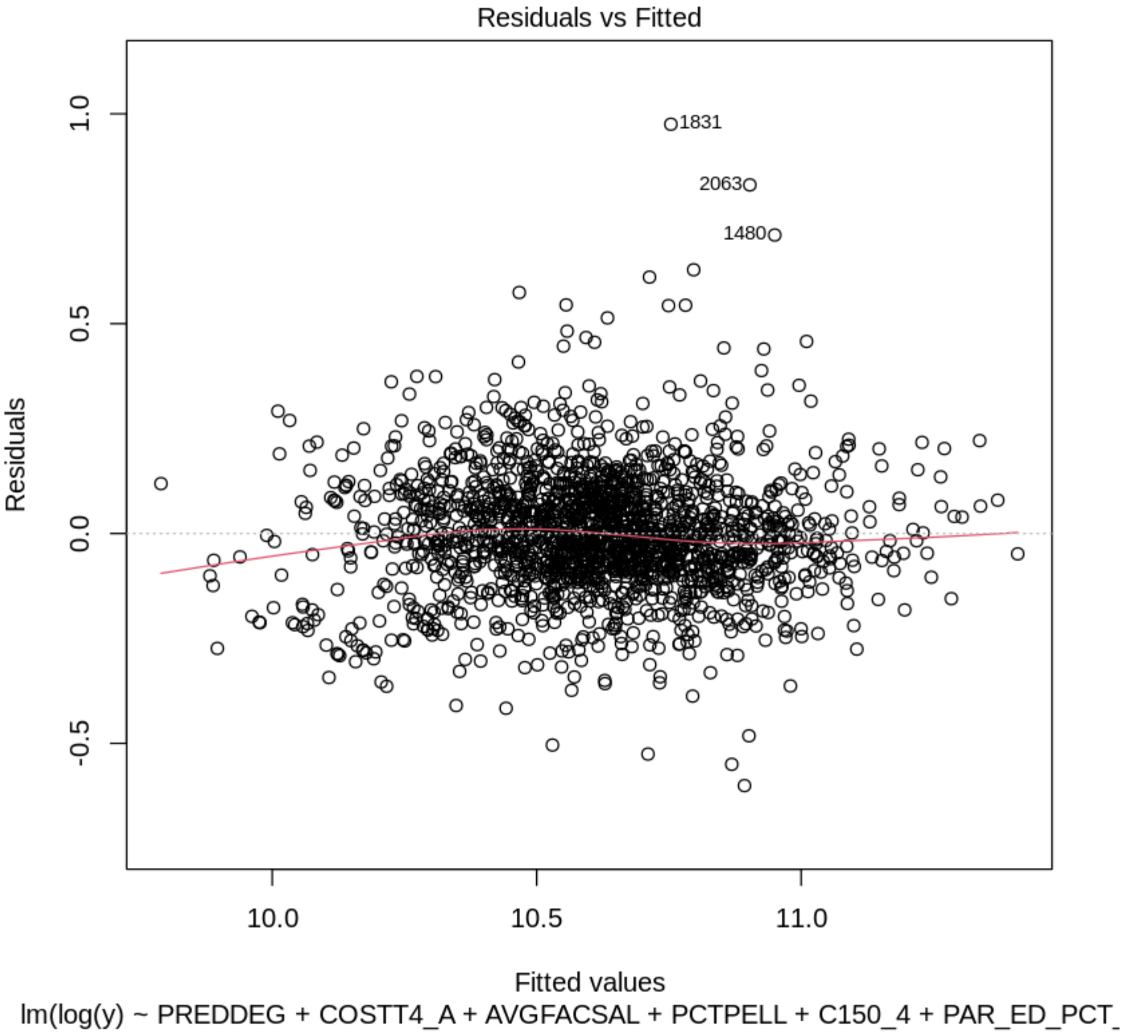
[174]:
```
model2 = lm(log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL+ C150_4 +␣
↪PAR_ED_PCT_1STGEN + PELL_EVER, data = college)
summary(model2)
```

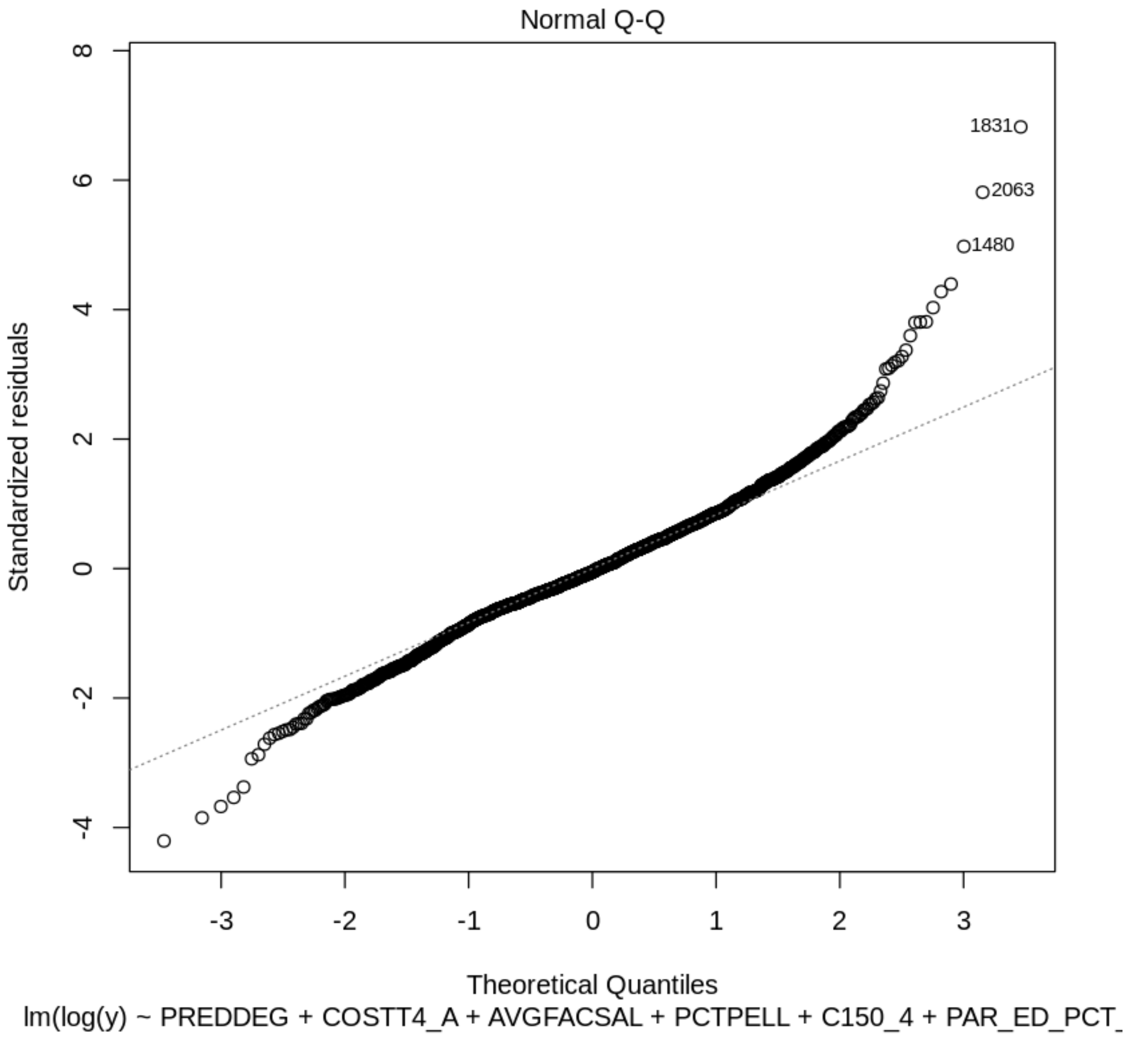
```
Call:
lm(formula = log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL +
    C150_4 + PAR_ED_PCT_1STGEN + PELL_EVER, data = college)

Residuals:
     Min       1Q   Median       3Q      Max
```

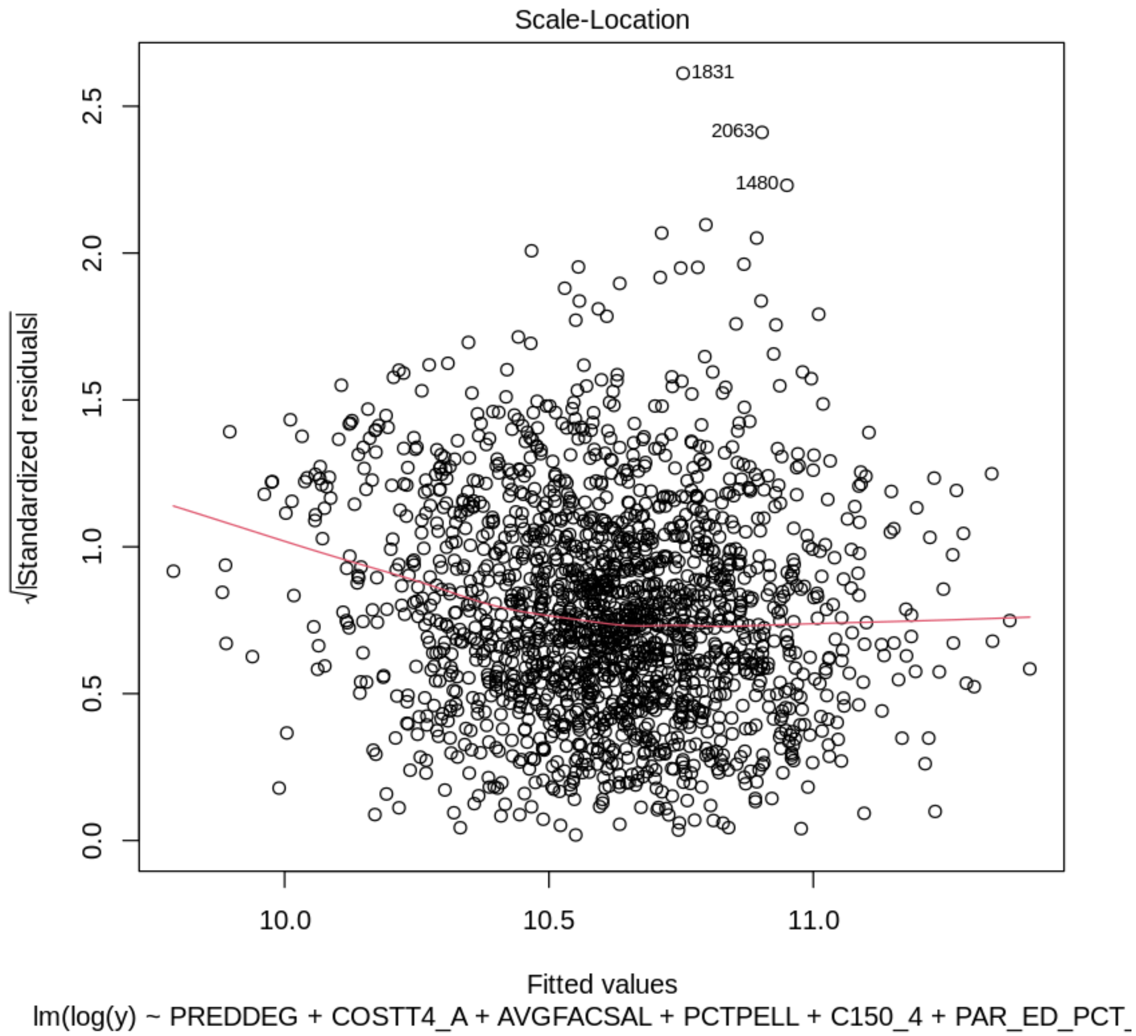
```
-0.60074 -0.08024 -0.00751  0.07937  0.97513 

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)    
(Intercept)        1.029e+01  3.833e-02 268.405  < 2e-16 ***
PREDDEG2           7.192e-02  2.162e-02   3.326 0.000899 ***
PREDDEG3           1.336e-01  2.075e-02   6.439 1.52e-10 ***
COSTT4_A           1.756e-06  2.807e-07   6.257 4.85e-10 ***
AVGFACSAL          4.962e-05  1.714e-06  28.943  < 2e-16 ***
PCTPELL           -2.177e-01  3.022e-02  -7.206 8.36e-13 ***
C150_4             9.203e-02  2.562e-02   3.593 0.000336 ***
PAR_ED_PCT_1STGEN  8.116e-01  5.049e-02  16.076  < 2e-16 ***
PELL_EVER         -7.833e-01  3.922e-02 -19.972  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1431 on 1856 degrees of freedom
  (2784 observations deleted due to missingness)
Multiple R-squared:  0.7272,	Adjusted R-squared:  0.7261 
F-statistic: 618.6 on 8 and 1856 DF,  p-value: < 2.2e-16
```

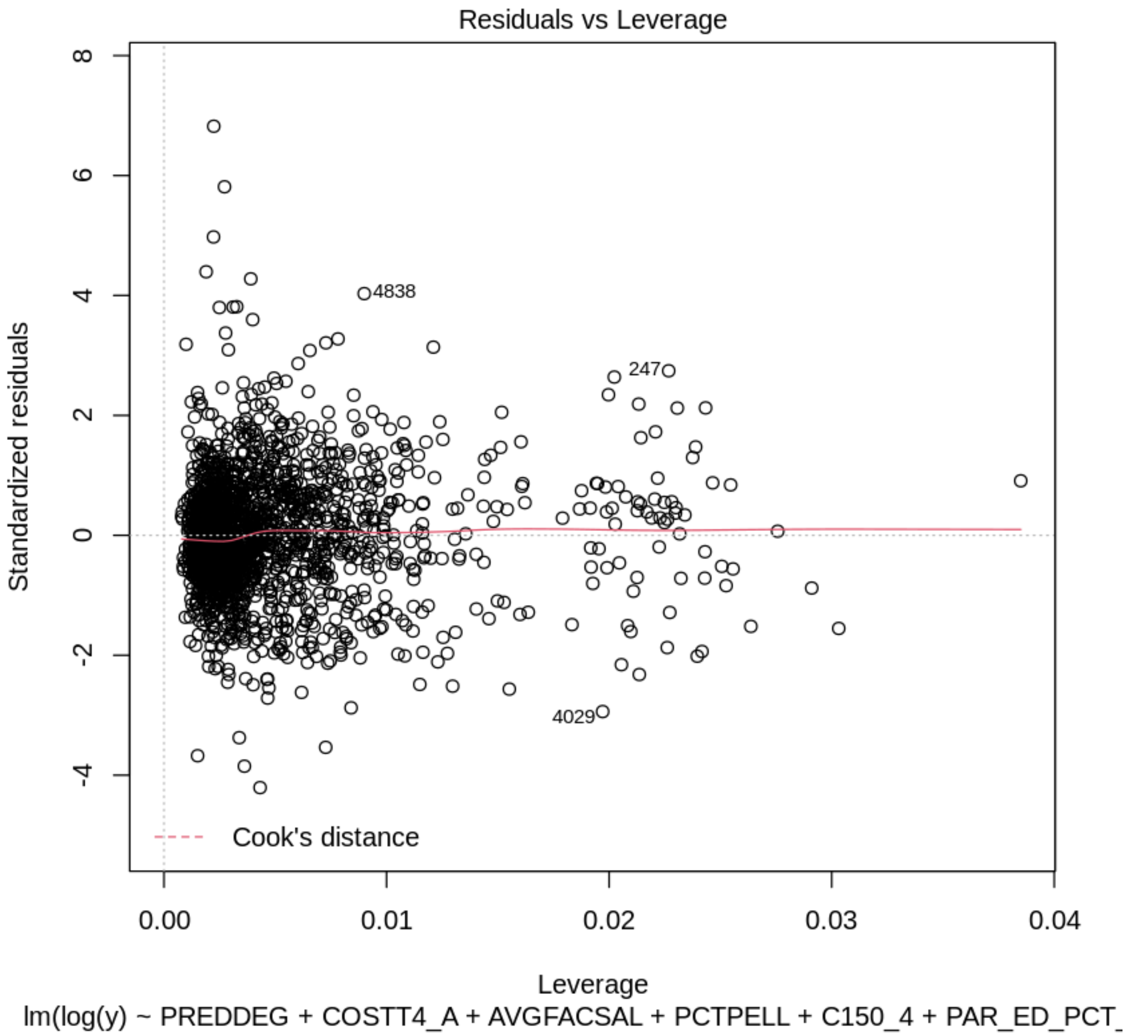
We can see from the summary table that all the coefficients have low p values, so that we reject all the nulls and all of them are significant in our model. Now, we need to check whether this model meets the “LINE” conditions or not.

```
[175]: plot(model2)
```


Residuals vs Fitted
1.0
0.5
0.0
-0.5
Residuals
1831
2063
1480
10.0
10.5
11.0
Fitted values
lm(log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL + C150_4 + PAR_ED_PCT_


Normal Q-Q
Standardized residuals
8
6
4
2
0
-2
-4
1831
2063
1480
-3
-2
-1
0
1
2
3
Theoretical Quantiles
lm(log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL + C150_4 + PAR_ED_PCT_

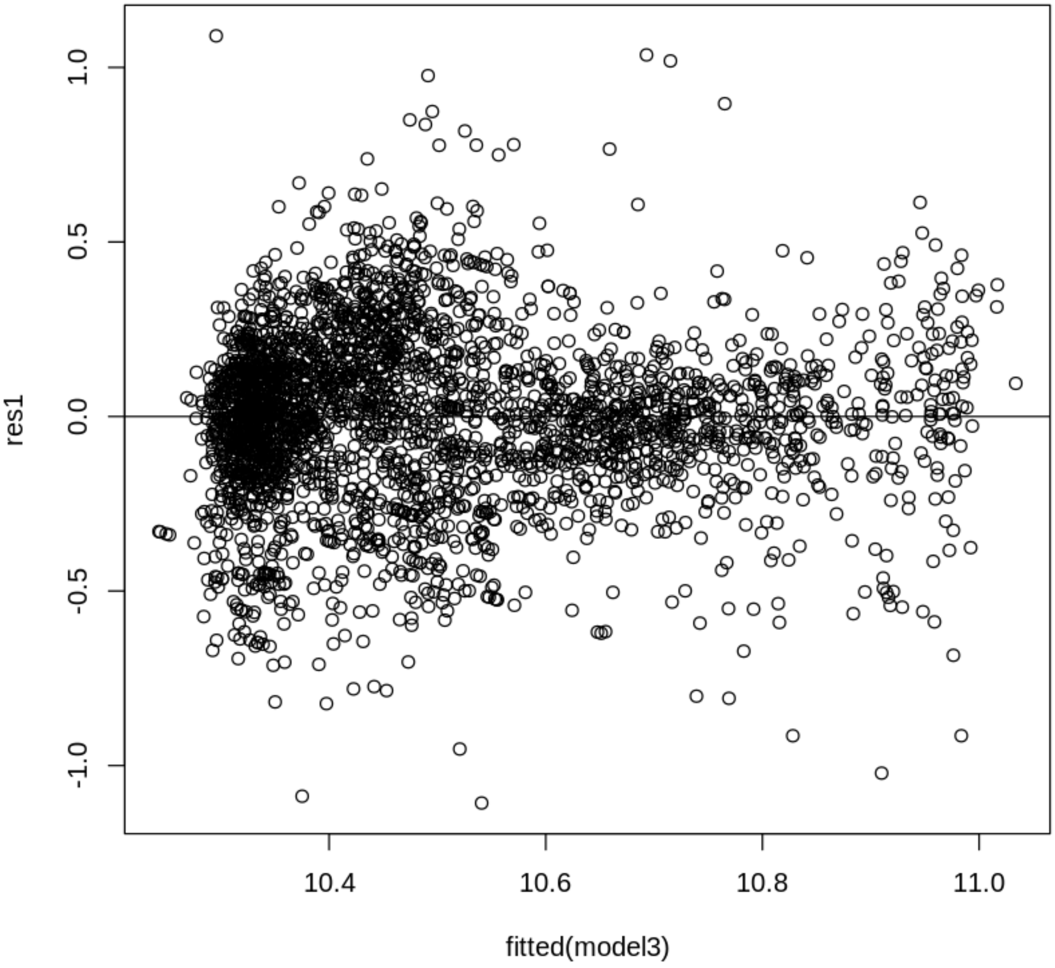

Scale-Location
1831
2063
1480
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
10.0
10.5
11.0
Fitted values
lm(log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL + C150_4 + PAR_ED_PCT_



From the residuals vs fitted values and standardized residuals vs fitted values graphs, we can see that these graphs are "well-behaved" because data points randomly bounce around. Moreover, from the residuals vs fitted values graph, we don't observe any drastic outliers. From the normal Q-Q plot, we can see that the residuals are approximately normally distributed as well. From the residuals vs leverage graph, we can see that there are no concerning influential points that need to be addressed (all cases are well inside of the Cook's distance lines).

[176]:
```
model3 = lm(log(y) ~ COSTT4_A, data = college)
summary(model2)
res1 = resid(model3)
plot(fitted(model3), res1)
abline(0, 0)
```

```
Call:
lm(formula = log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL +
    C150_4 + PAR_ED_PCT_1STGEN + PELL_EVER, data = college)

Residuals:
     Min       1Q   Median       3Q      Max
-0.60074 -0.08024 -0.00751  0.07937  0.97513

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        1.029e+01  3.833e-02 268.405  < 2e-16 ***
PREDDEG2           7.192e-02  2.162e-02   3.326 0.000899 ***
PREDDEG3           1.336e-01  2.075e-02   6.439 1.52e-10 ***
COSTT4_A           1.756e-06  2.807e-07   6.257 4.85e-10 ***
AVGFACSAL          4.962e-05  1.714e-06  28.943  < 2e-16 ***
PCTPELL           -2.177e-01  3.022e-02  -7.206 8.36e-13 ***
C150_4             9.203e-02  2.562e-02   3.593 0.000336 ***
PAR_ED_PCT_1STGEN  8.116e-01  5.049e-02  16.076  < 2e-16 ***
PELL_EVER         -7.833e-01  3.922e-02 -19.972  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1431 on 1856 degrees of freedom
  (2784 observations deleted due to missingness)
Multiple R-squared:  0.7272,       Adjusted R-squared:  0.7261
F-statistic: 618.6 on 8 and 1856 DF,  p-value: < 2.2e-16
```

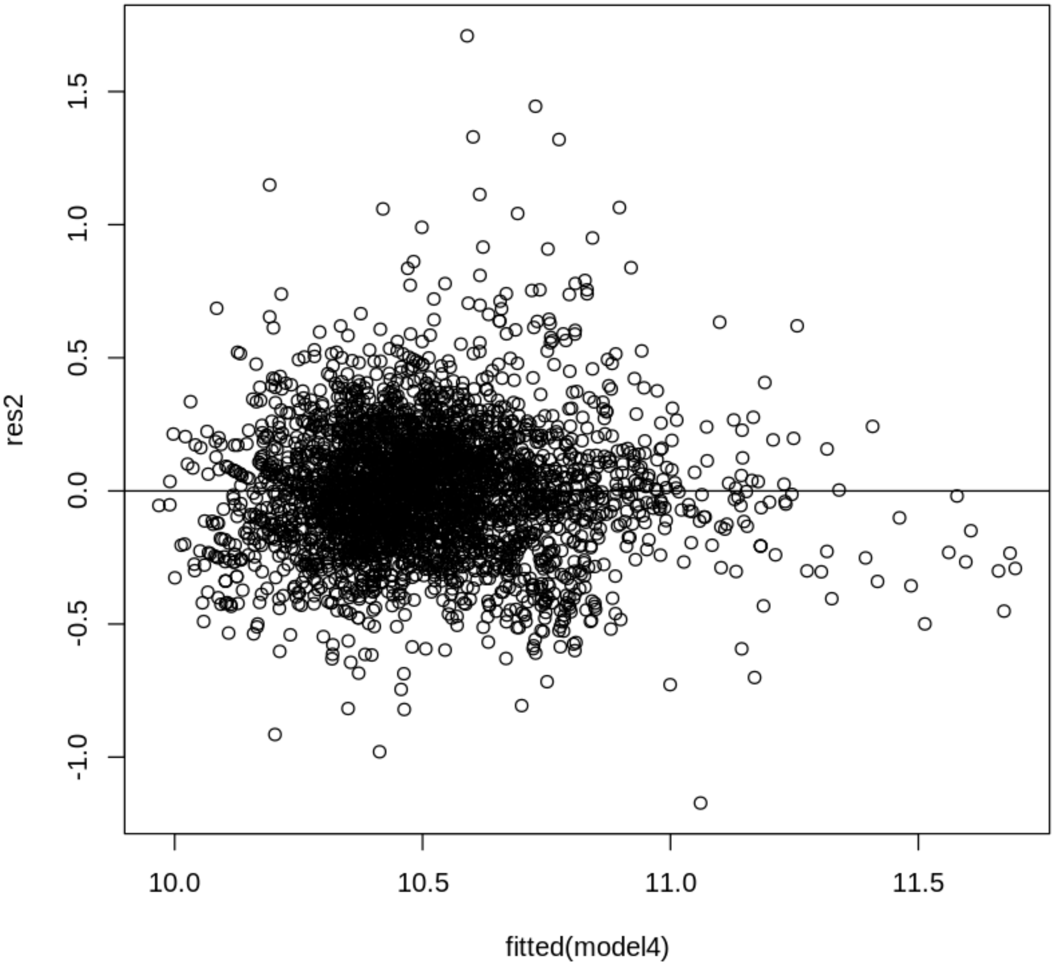


[177]:

```
model4 = lm(log(y) ~ AVGFACSAL, data = college)
summary(model4)
res2 = resid(model4)
plot(fitted(model4), res2)
abline(0, 0)
```

```
Call:
lm(formula = log(y) ~ AVGFACSAL, data = college)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.17234 -0.15149 -0.00558  0.13963  1.70926 
```

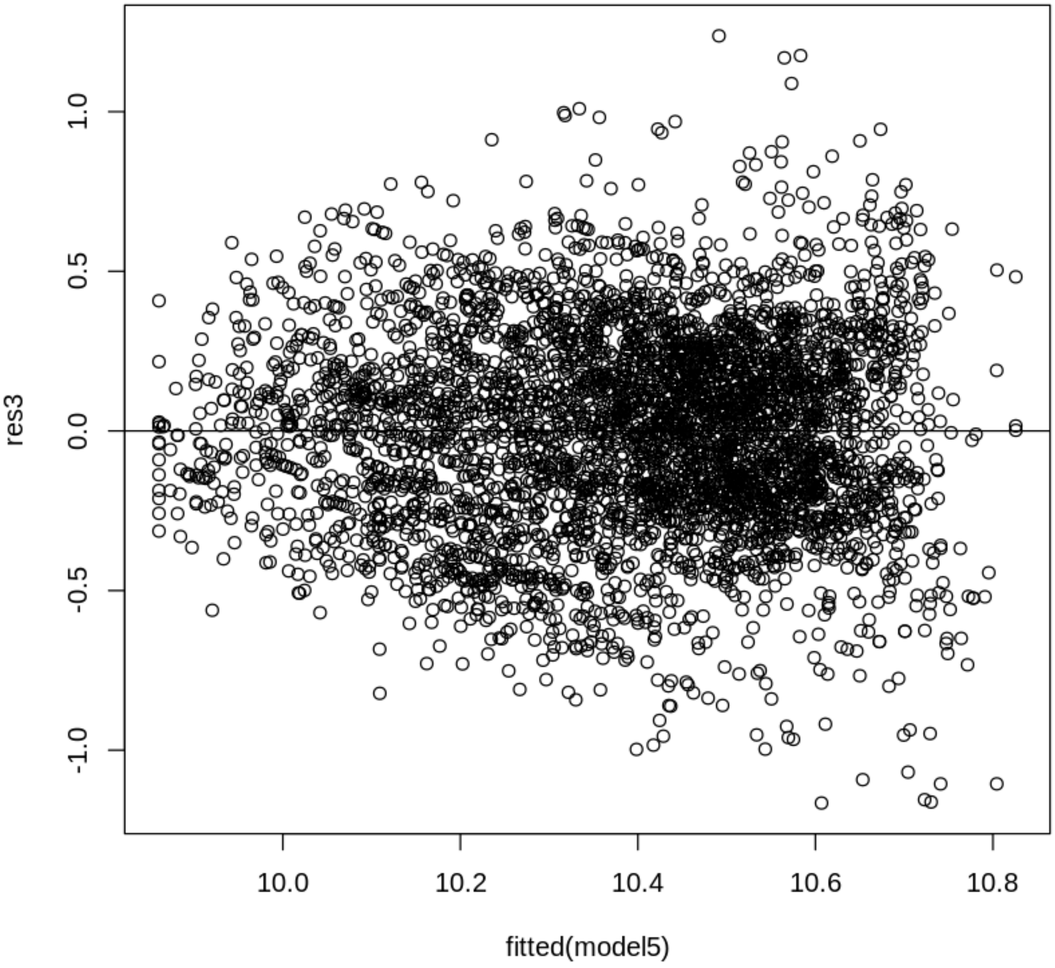
```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 9.887e+00  1.295e-02  763.52   <2e-16 ***
AVGFACSAL   8.616e-05  1.694e-06   50.87   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2439 on 3261 degrees of freedom
  (1386 observations deleted due to missingness)
Multiple R-squared:  0.4424,        Adjusted R-squared:  0.4423
F-statistic:  2588 on 1 and 3261 DF,  p-value: < 2.2e-16
```



[178]:

```
model5 = lm(log(y) ~ PCTPELL, data = college)
summary(model5)
res3 = resid(model5)
plot(fitted(model5), res3)
abline(0, 0)
```

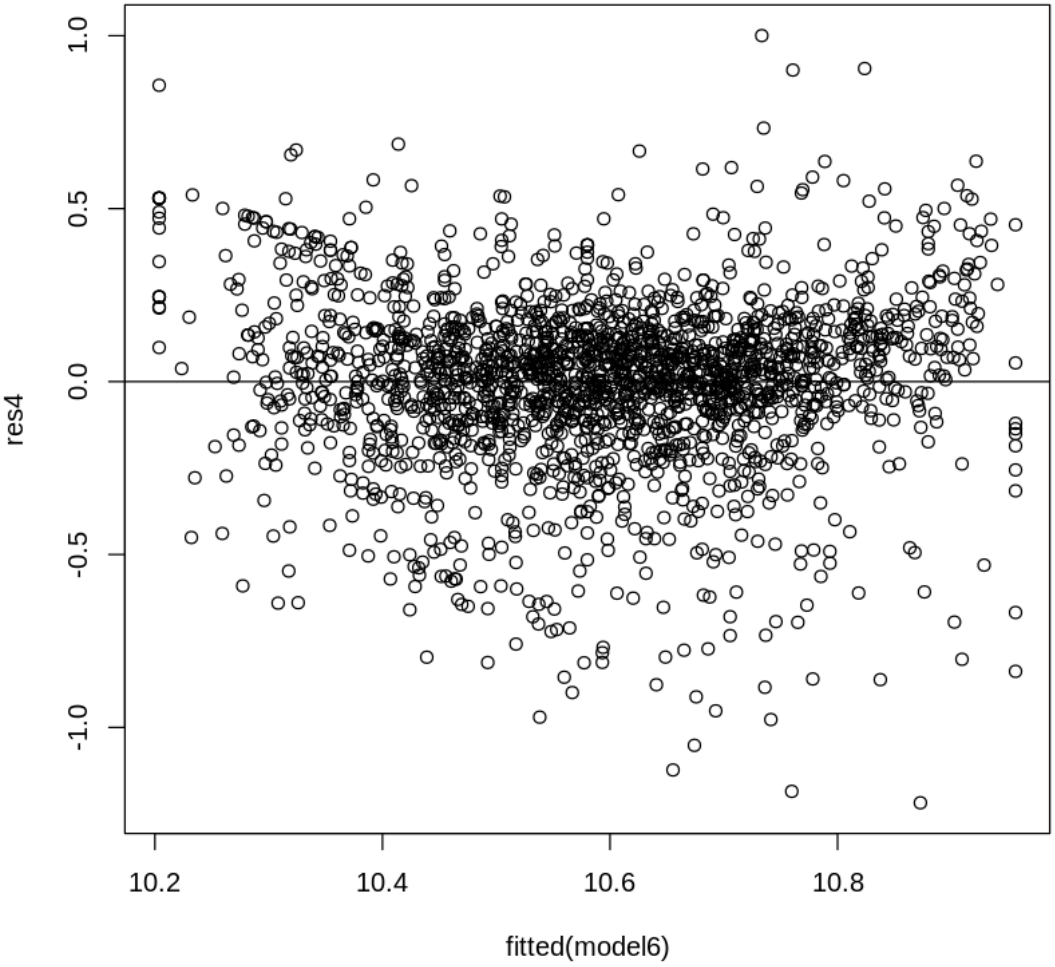
```
Call:
lm(formula = log(y) ~ PCTPELL, data = college)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.16545 -0.20849  0.01447  0.21310  1.23748 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 10.82573    0.01152  939.69   <2e-16 ***
PCTPELL     -0.96525    0.02336  -41.32   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3117 on 4312 degrees of freedom
  (335 observations deleted due to missingness)
Multiple R-squared:  0.2837,	Adjusted R-squared:  0.2835 
F-statistic:  1708 on 1 and 4312 DF,  p-value: < 2.2e-16
```



[179]:
```
model6 = lm(log(y) ~ C150_4, data = college)
summary(model6)
res4 = resid(model6)
plot(fitted(model6), res4)
abline(0, 0)
```

```
Call:
lm(formula = log(y) ~ C150_4, data = college)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.2178 -0.1073  0.0224  0.1343  1.0003 
```

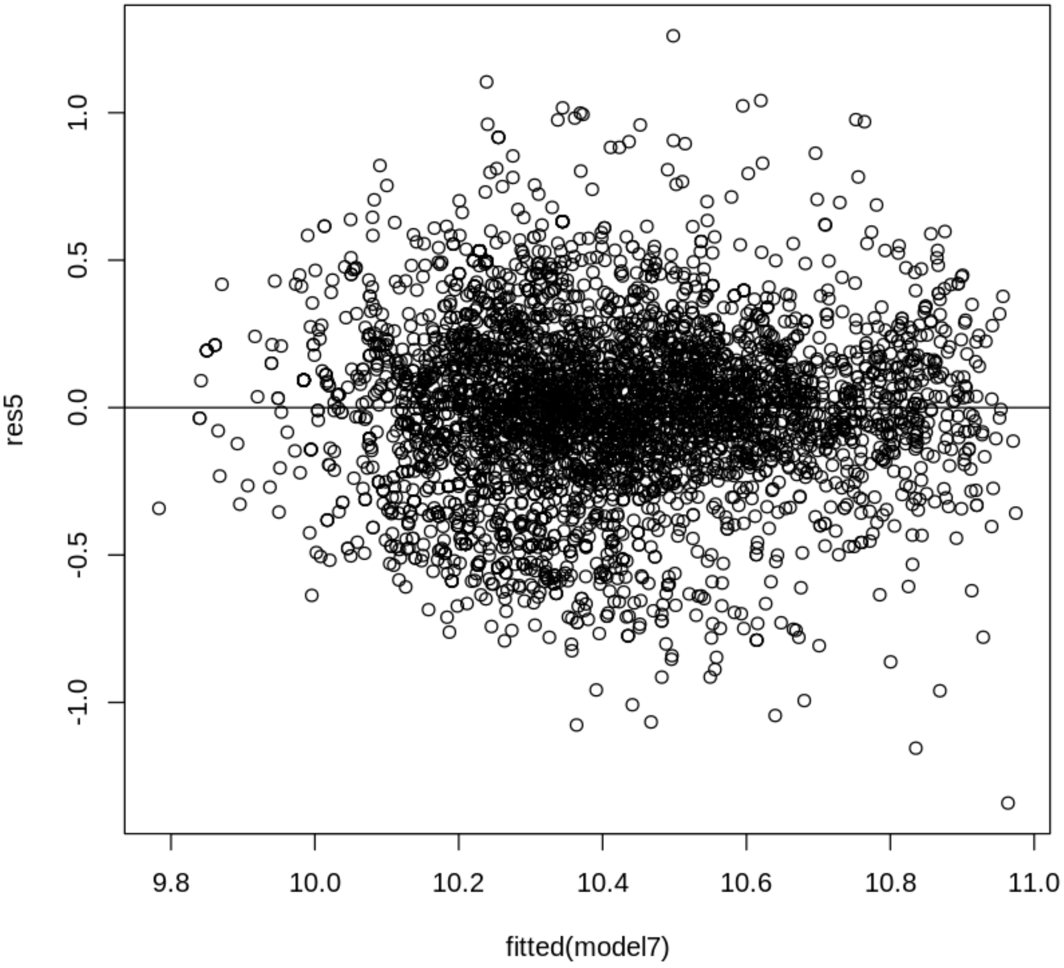
```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 10.20375    0.01479  690.08   <2e-16 ***
C150_4       0.75267    0.02647   28.43   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.249 on 2037 degrees of freedom
  (2610 observations deleted due to missingness)
Multiple R-squared:  0.2841,	Adjusted R-squared:  0.2837
F-statistic: 808.3 on 1 and 2037 DF,  p-value: < 2.2e-16
```



[180]:

```
model7 = lm(log(y) ~ PAR_ED_PCT_1STGEN, data = college)
summary(model7)
res5 = resid(model7)
plot(fitted(model7), res5)
abline(0, 0)
```

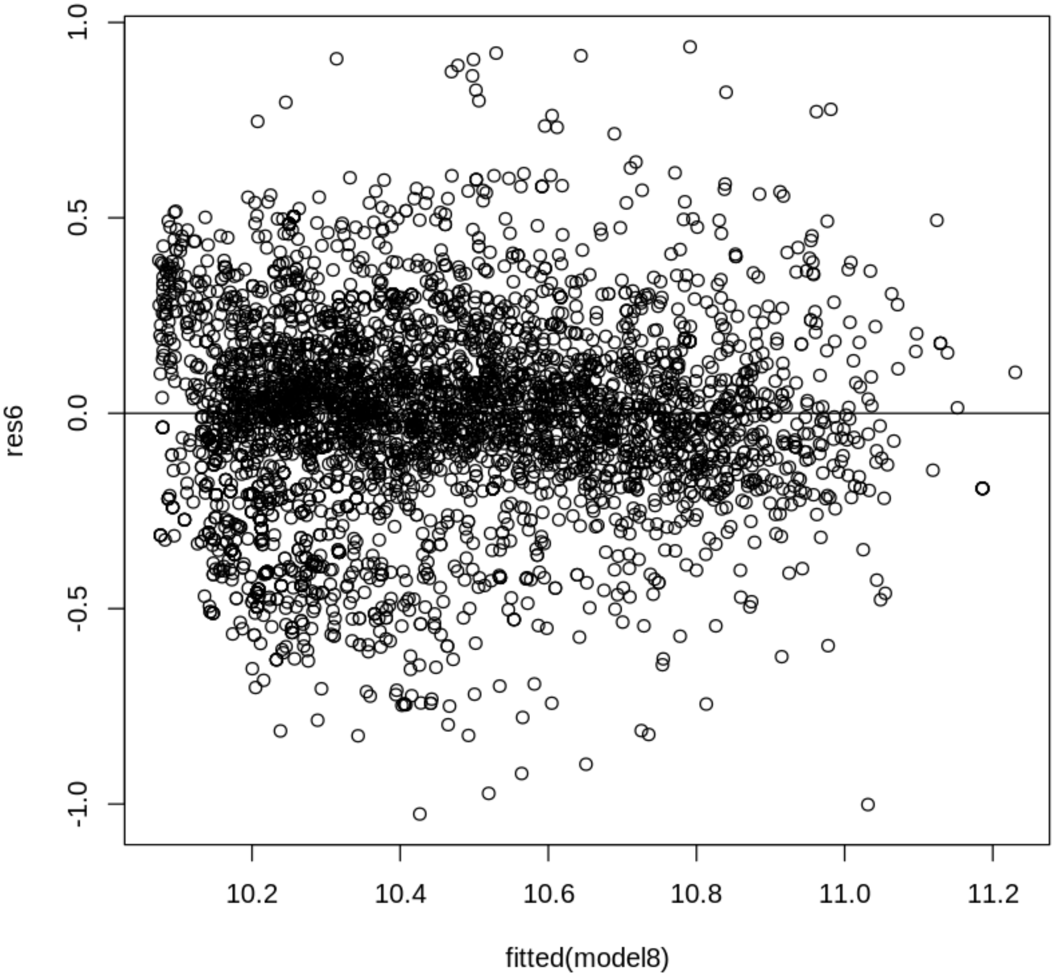
```
Call:
lm(formula = log(y) ~ PAR_ED_PCT_1STGEN, data = college)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.34117 -0.15599  0.01032  0.17080  1.26078 

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)    
(Intercept)       11.11973    0.01558  713.91   <2e-16 ***
PAR_ED_PCT_1STGEN -1.63923    0.03427  -47.84   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2911 on 4402 degrees of freedom
  (245 observations deleted due to missingness)
Multiple R-squared:  0.3421,	Adjusted R-squared:  0.3419 
F-statistic:  2288 on 1 and 4402 DF,  p-value: < 2.2e-16
```



[181]:
```
model8 = lm(log(y) ~ PELL_EVER, data = college)
summary(model8)
res6 = resid(model8)
plot(fitted(model8), res6)
abline(0, 0)
```

```
Call:
lm(formula = log(y) ~ PELL_EVER, data = college)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.02554 -0.12861  0.00847  0.13991  0.93754 
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 11.38107    0.01628  699.05   <2e-16 ***
PELL_EVER   -1.31101    0.02215  -59.18   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2541 on 4099 degrees of freedom
  (548 observations deleted due to missingness)
Multiple R-squared:  0.4607,        Adjusted R-squared:  0.4606
F-statistic:  3502 on 1 and 4099 DF,  p-value: < 2.2e-16
```



After plotting the residuals vs fitted values plot for each independent variable, I suspect that we should include PAR_ED_PCT_1STGEN squared term in our model since there seems to be a parabola trend in the residuals vs fitted values plot.

```
[182]: model9 = lm(log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL+ C150_4 +␣
        ↪PELL_EVER + PAR_ED_PCT_1STGEN + I(PAR_ED_PCT_1STGEN^2), data = college)
       summary(model9)
```

```
Call:
lm(formula = log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL +
    C150_4 + PELL_EVER + PAR_ED_PCT_1STGEN + I(PAR_ED_PCT_1STGEN^2),
    data = college)

Residuals:
     Min       1Q   Median       3Q      Max
-0.59059 -0.08125 -0.00733  0.07921  0.97500

Coefficients:
                         Estimate Std. Error t value Pr(>|t|)
(Intercept)             1.025e+01  4.790e-02 213.913  < 2e-16 ***
PREDDEG2                6.931e-02  2.168e-02   3.197 0.001414 **
PREDDEG3                1.273e-01  2.114e-02   6.023 2.06e-09 ***
COSTT4_A                1.848e-06  2.870e-07   6.440 1.52e-10 ***
AVGFACSAL               4.987e-05  1.721e-06  28.968  < 2e-16 ***
PCTPELL                -2.146e-01  3.027e-02  -7.090 1.90e-12 ***
C150_4                  9.722e-02  2.583e-02   3.763 0.000173 ***
PELL_EVER              -7.814e-01  3.923e-02 -19.920  < 2e-16 ***
PAR_ED_PCT_1STGEN       1.056e+00  1.679e-01   6.287 4.03e-10 ***
I(PAR_ED_PCT_1STGEN^2) -3.407e-01  2.236e-01  -1.524 0.127779
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1431 on 1855 degrees of freedom
  (2784 observations deleted due to missingness)
Multiple R-squared:  0.7276,	Adjusted R-squared:  0.7263
F-statistic: 550.5 on 9 and 1855 DF,  p-value: < 2.2e-16
```

After running the model, it turns out to be not significant, so we will go with the original model.

```
[183]: model10 = lm(log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL+ C150_4 +␣
        ↪PAR_ED_PCT_1STGEN + PELL_EVER, data = college)
       model11 = lm(log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL+ C150_4 +␣
        ↪PAR_ED_PCT_1STGEN + PELL_EVER + PREDDEG*PELL_EVER, data = college)
       anova(model10, model11)
```

A anova: 2 × 6

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	1856	38.01013	NA	NA	NA	NA
2	1854	37.34698	2	0.6631578	16.46043	8.204508e-08

After finishing construct model10, we suspect that PREDDEG and PELL_EVER might have an interaction effect, so that we conduct ANOVA to figure it out. The p value for the coefficient of the interaction term is really small so that we will go with the model with the interaction term.

So the final regression model is:

$$log(y) = {}_0 + {}_1 * PREDDEG2 + {}_2 * PREDDEG3 + {}_3 * COSTT4\_A + {}_4 * AVGFACSAL + {}_5 * PCTPELL + {}_6 * C150\_4 + {}_7 * PAR\_ED\_PCT\_1STGEN + {}_8 * PELL\_EVER + {}_9 * PREDDEG2 : PELL\_EVER + {}_{10} * PREDDEG3 : PELL\_EVER$$

[184]:
```
summary(model11)
```

```

Call:
lm(formula = log(y) ~ PREDDEG + COSTT4_A + AVGFACSAL + PCTPELL +
    C150_4 + PAR_ED_PCT_1STGEN + PELL_EVER + PREDDEG * PELL_EVER,
    data = college)

Residuals:
     Min       1Q   Median       3Q      Max
-0.59511 -0.08107 -0.00677  0.07762  0.98094

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)         1.017e+01  1.357e-01  74.987  < 2e-16 ***
PREDDEG2            4.343e-01  1.438e-01   3.021 0.002558 **
PREDDEG3            1.986e-01  1.358e-01   1.462 0.143987
COSTT4_A            1.972e-06  2.810e-07   7.016 3.19e-12 ***
AVGFACSAL           4.908e-05  1.704e-06  28.806  < 2e-16 ***
PCTPELL            -2.548e-01  3.070e-02  -8.299  < 2e-16 ***
C150_4              1.064e-01  2.567e-02   4.145 3.55e-05 ***
PAR_ED_PCT_1STGEN   8.097e-01  5.017e-02  16.140  < 2e-16 ***
PELL_EVER          -6.306e-01  1.648e-01  -3.826 0.000135 ***
PREDDEG2:PELL_EVER -4.552e-01  1.736e-01  -2.622 0.008803 **
PREDDEG3:PELL_EVER -6.142e-02  1.639e-01  -0.375 0.707886
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1419 on 1854 degrees of freedom
  (2784 observations deleted due to missingness)
Multiple R-squared:  0.732,	Adjusted R-squared:  0.7306
F-statistic: 506.4 on 10 and 1854 DF,  p-value: < 2.2e-16
```

From the summary table, we can see that our final model achieves an adjusted R-squared around

0.7306.

4 Summary

The model we select to predict log(median earnings of students working and not enrolled 10 years after entry) is as follow:

$log(y) = {}_0 + {}_1 * PREDDEG2 + {}_2 * PREDDEG3 + {}_3 * COSTT4\_A + {}_4 * AVGFACSAL + {}_5 * PCTPELL + {}_6 * C150\_4 + {}_7 * PAR\_ED\_PCT\_1STGEN + {}_8 * PELL\_EVER + {}_9 * PREDDEG2 : PELL\_EVER + {}_{10} * PREDDEG3 : PELL\_EVER$

${}_1$, ${}_2$, ${}_9$, and ${}_{10}$ all are coefficients for the dummy variable one-hot encoded bases on the independent variable "PREDDEG". Moreover, ${}_9$ and ${}_{10}$ show the interaction effect between categorical variable "PREDDEG" and continuous variable "PELL_EVER". If you have an associate's degree (PREDDEG2 = 1), your earning will increase by around 54% ($e^{0.4343} - 1$). If you have an bachelor's degree (PREDDEG3 = 1), your earning will increase by around 22% ($e^{0.1986} - 1$). When you have an associate's degree and share of students who received a Pell Grant while in school(PELL_EVER) increase by 1 unit, the earning will decrease by 57% ($e^{0.4343} - 1$). When you have an bachelor's degree and share of students who received a Pell Grant while in school(PELL_EVER) increase by 1 unit, the earning will decrease by 6% ($e^{0.06142} - 1$).

${}_3$, ${}_4$, ${}_6$, and ${}_7$ for "COSTT4_A", "AVGFACSAL", "C150_4", and "PAR_ED_PCT_1STGEN" all have positive coefficients, which means that increase the independent variable by 1 unit will increase the dependent variable. ${}_5$ and ${}_8$ for "PCTPELL" and "PELL_EVER" have negative coefficients, which means that increase the independent varibale by 1 unit will decrease the dependent variable.

After implementing this model, approximately 73% of the variance in the median earnings (target variable) can be explained by the independent variables of our choice.