регрессия

2024-04-03

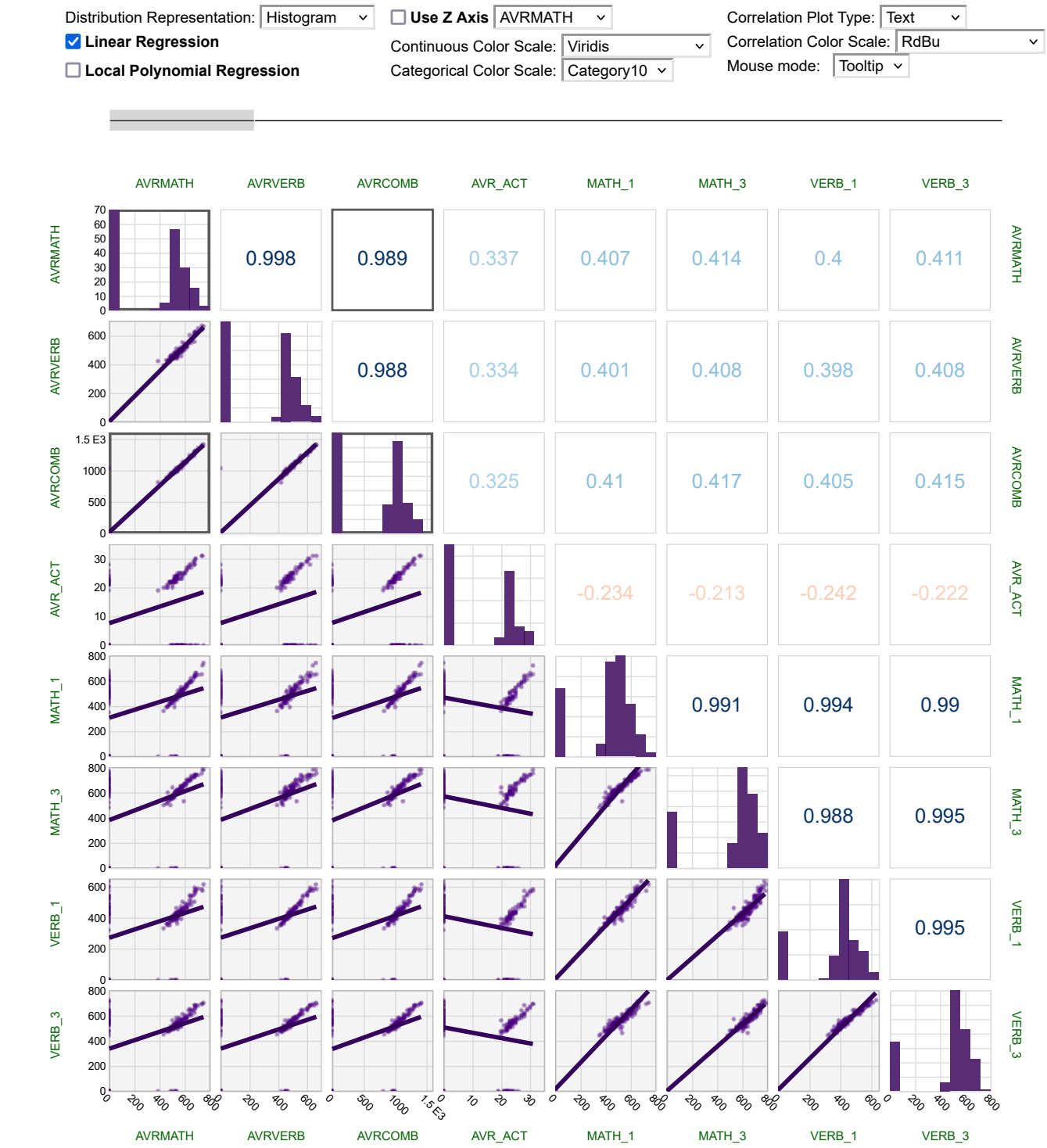
Предварительный анализ данных

```
library(readxl)
library(tidyverse)
library(kableExtra)

data <- read_excel("C:/Users/yanas/Documents/7R/I_shortname.xls")


print_df <- function(df)
{
  df |>
    kable(format = "html") |>
    kable_styling() |>
    kableExtra::scroll_box(width = "100%", height = "100%")
}

head(data, 8) |>
 print_df()
```

...1	PPIND	FICE	STATE	TYPE	AVRMATH	AVRVERB	AVRCOMB	AVR_ACT	MATH_1	MATH_3	VERB_1	VERB_3
Auburn University-Ma	1	1009	AL	I	575	501	1076	24	520	638	453	559
University of Alabam	1	1051	AL	I	NA	NA	NA	23	NA	NA	NA	NA
University of Alabam	1	1052	AL	I	NA	NA	NA	21	NA	NA	NA	NA
University of Alaska	1	1063	AK	I	499	462	961	22	NA	NA	NA	NA
Arizona State Univer	1	1081	AZ	I	521	453	974	23	450	590	390	500
Northern Arizona Uni	1	1082	AZ	I	495	444	939	22	420	560	380	500
University of Arizon	1	1083	AZ	I	526	462	983	23	450	600	400	520
University of Arkans	1	1108	AR	I	NA	NA	NA	NA	NA	NA	NA	NA

Построим графики зависимостей признаков и выберем подходящие

```
library(scatterPlotMatrix)
library(dplyr)

data |>
  select(-PPIND, -FICE, -STATE, -TYPE, -...1) |>
  scatterPlotMatrix(regressionType = 1,
                    corrPlotType = "Text",
                    plotProperties = list(noCatColor = "Indigo"),
                    controlWidgets = TRUE,
                    height = 1050,
                    width = 1000)
```



Оставим в новом датасете только рассматриваемые

признаки

```
df <-
  data |>
  select(...1,
         PPIND,
         ADD_FEE,
         BOOK,
         NEW10,
         PH_D,
         SAL_ALL,
         SF_RATIO,
         GRADUAT,
         INSTRUCT)
```

Посмотрим на моды:

```
library(dplyr)
modes<-summarize(df, across(ADD_FEE:INSTRUCT, function(x) max(table(x))))
print(modes)
```

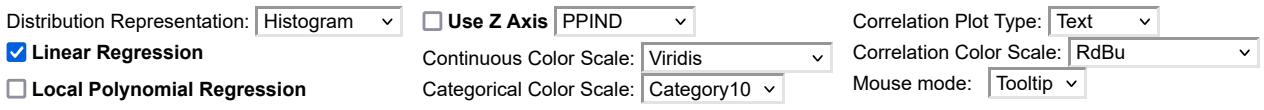
```
## # A tibble: 1 × 8
##   ADD_FEE  BOOK NEW10  PH_D SAL_ALL SF_RATIO GRADUAT INSTRUCT
##     <int> <int> <int> <int>   <int>    <int>   <int>    <int>
## 1       4    29    10    10       5        5       7        3
```

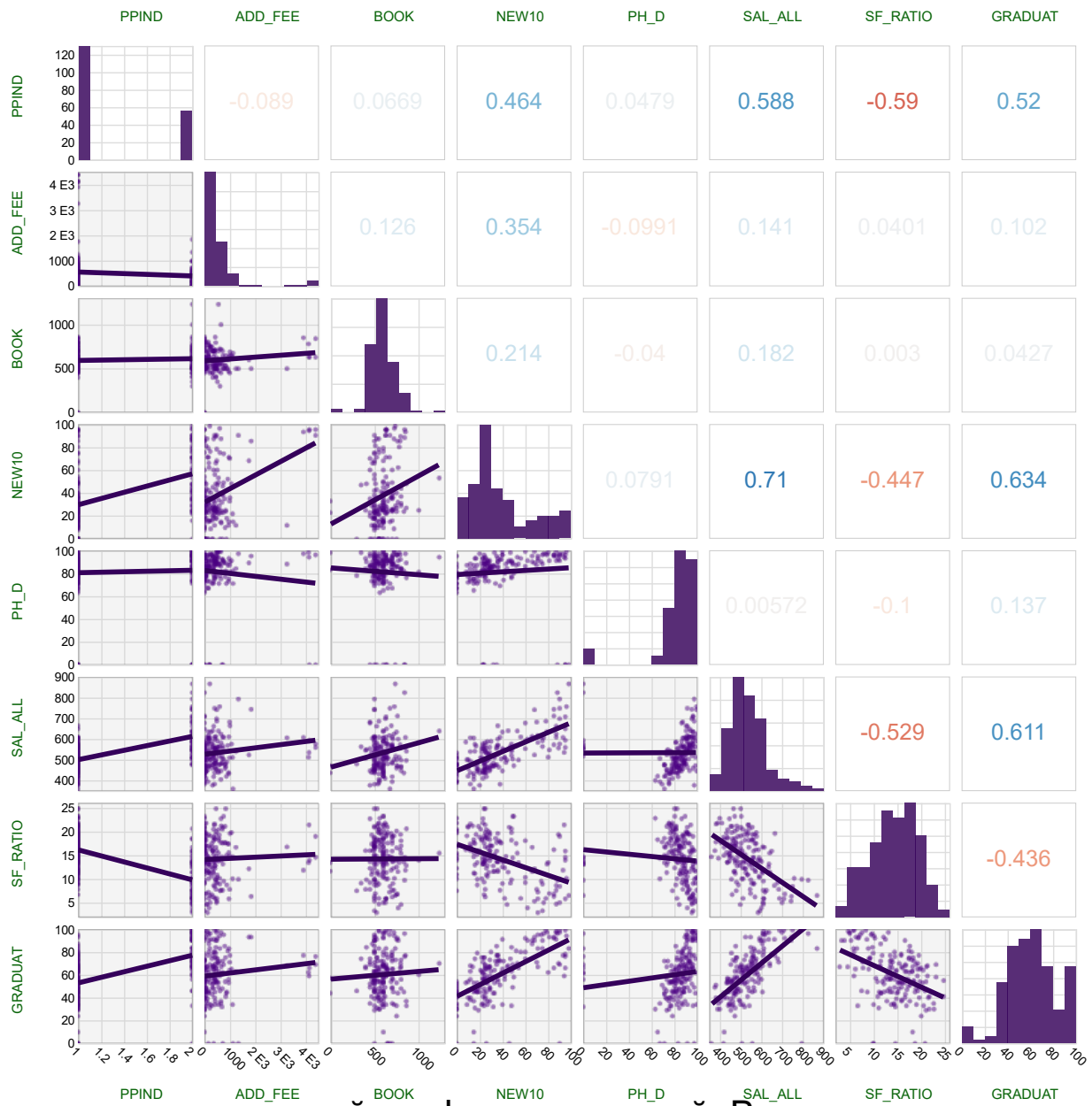
Рассматриваемые признаки(первые 4 есть в задании, остальные — те, которые влияют на переменную NEW10):

1. ***College name*** — название колледжа — **качественный признак**
2. ***PPIND*** — Гос/частное заведение (гос = 1, частный = 2) — **качественный признак**
3. ***ADD_FEE*** — дополнительные сборы — **количественный непрерывный признак**
4. ***BOOK*** — примерная стоимость учебников — **количественный дискретный признак**
5. ***NEW10(Процент студентов из лучших слоев школьных выпускников)*** — **количественный дискретный признак**
6. ***PH_D***— количество преподавателей со степенью Ph.D.— **количественный дискретный признак**
7. ***SAL_ALL*** — Средняя заработная плата — **количественный непрерывный признак**
8. ***SF_RATIO*** — соотношение студентов к преподавателям — **количественный непрерывный признак**
9. ***GRADUAT*** — процент выпускников — **количественный дискретный признак**
10. ***INSTRUCT*** — расходы на обучение в расчете на одного учащегося — **количественный непрерывный признак**

Построим MatrixPlot для рассматриваемых признаков:

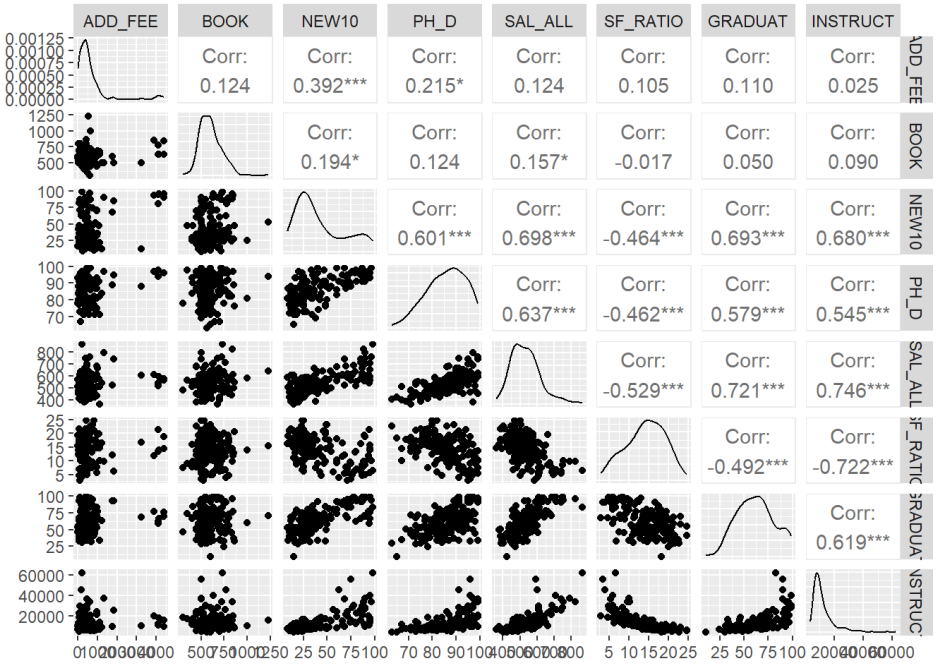
```
library(scatterPlotMatrix)
library(dplyr)


categories <- list( c(1,2), NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL)

df |>
  select(-...1)|>
  scatterPlotMatrix(regressionType = 1,
                        corrPlotType = "Text",
                        categorical = categories,
                        plotProperties = list(noCatColor =  "Indigo"),
                        controlWidgets = TRUE,
                        height = 1050, width = 1000)
```





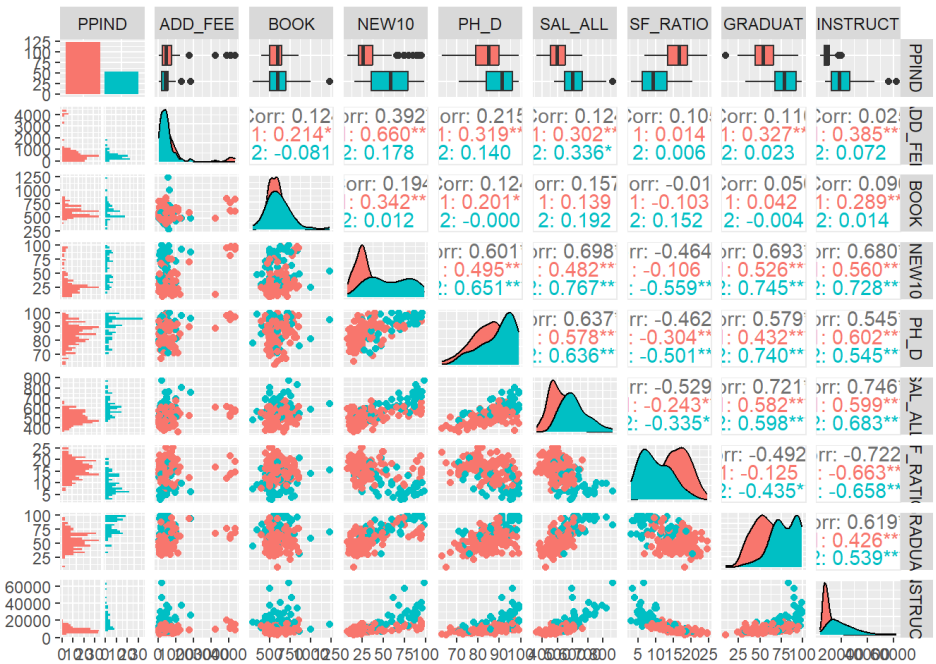
Построим также другой график плотностей. Видно, что распределения большинства переменных несимметричны, с хвостом вправо.

```
library(GGally)
library(dplyr)
df|>
  select(-...1, -PPIND) |>
    mutate_all(as.numeric) |>
      ggpairs()
```



Посмотрим на однородность/неоднородность по типу заведения(государственное или частное)

```
library(GGally)
library(dplyr)

df$PPIND <- factor(df$PPIND)

# Построим графики с разделением по 'PPIND'
ggpairs(df, columns = 2:ncol(df), aes(colour = PPIND))
```



Прологорифмируем несимметричные с хвостом вправо распределения(“SAL_ALL”, “NEW10”, “BOOK”, “ADD_FEE”, “INSTRUCT”, “GRADUAT”) и снова построим графики:

```
library(scatterPlotMatrix)

data_log <- df |>
  mutate_at(vars("SAL_ALL",
                 "NEW10",
                 "BOOK",
                 "GRADUAT",
                 "ADD_FEE",
                 "INSTRUCT"),
            ~log(.))

categories <- list( c(1,2), NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL)


data_log|>
  select(-...1) |>
  scatterPlotMatrix(regressionType = 1,
                        corrPlotType = "Text",
                        categorical = categories,
                        plotProperties = list(noCatColor =  "Indigo"),
                        controlWidgets = TRUE,
                        height = 1050, width = 1000)
```

Distribution Representation: Histogram

☑ **Linear Regression**

☐ **Local Polynomial Regression**

☐ **Use Z Axis** PPIND

Continuous Color Scale: Viridis

Categorical Color Scale: Category10

Correlation Plot Type: Text

Correlation Color Scale: RdBu

Mouse mode: Tooltip

Далее будут рассматриваться только частные университеты

```
library(dplyr)
private_colleges <- data %>%
  filter(PPIND == 2)

private_colleges_log <- data_log %>%
  filter(PPIND == 2)

private_colleges_log <- private_colleges_log %>%
  rename_with(~ paste0(.x, "_LOG"))

head(private_colleges_log, 8)
```

```
## # A tibble: 8 × 10
##   ...1_LOG          PPIND_LOG ADD_FEE_LOG BOOK_LOG NEW10_LOG PH_D_LOG SAL_ALL_LOG
##   <chr>             <fct>           <dbl>    <dbl>     <dbl>     <dbl>        <dbl>
## 1 California Inst… 2                5.35     6.60      4.58        NA           76
## 2 Stanford Univer… 2               NA        6.73      4.48        NA           71
## 3 University of S… 2                5.80     6.40      3.81        90           46
## 4 University of D… 2                5.80     5.99      3.37        84           6.23
## 5 Yale University  2               NA        6.45      4.55        96           6.56
## 6 University of D… 2                6.02     6.47      3.09        82           6.32
## 7 American Univer… 2                5.35     6.71      3.56        NA           6.32
## 8 Catholic Univer… 2                6.13     6.77      3.18        90           6.18
## # ℹ 3 more variables: SF_RATIO_LOG <dbl>, GRADUAT_LOG <dbl>, INSTRUCT_LOG <dbl>
```

Построим ggpairs

```
library(GGally)
library(dplyr)
ggpairs(private_colleges_log, columns = 3:ncol(private_colleges_log))
```

Так как данных мало, то заполним значения NA медианными значениями, чтобы не удалять строки.

```
private_colleges_na <- private_colleges_log %>% mutate_if(is.numeric, ~ifelse(is.na(.), median(., na.rm = TRUE), .))
```

Построим полную модель регрессии

```
library(lm.beta)

ModelFull <- lm(NEW10_LOG ~
              ADD_FEE_LOG +
              BOOK_LOG +
              PH_D_LOG +
              SAL_ALL_LOG +
              SF_RATIO_LOG +
              GRADUAT_LOG +
              INSTRUCT_LOG,
            data = private_colleges_na)

ModelFull <- lm.beta(ModelFull)
summary(ModelFull)
```

```
##
## Call:
## lm(formula = NEW10_LOG ~ ADD_FEE_LOG + BOOK_LOG + PH_D_LOG +
##     SAL_ALL_LOG + SF_RATIO_LOG + GRADUAT_LOG + INSTRUCT_LOG,
##     data = private_colleges_na)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.57095 -0.22877  0.02211  0.15558  0.71251
##
## Coefficients:
##               Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept)  -6.879955           NA   2.082580  -3.304  0.00188 **
## ADD_FEE_LOG  -0.036103    -0.056304   0.059376  -0.608  0.54621
## BOOK_LOG     -0.107100    -0.055867   0.173597  -0.617  0.54038
## PH_D_LOG      0.006024     0.095995   0.008210   0.734  0.46696
## SAL_ALL_LOG   0.798841     0.239369   0.534198   1.495  0.14179
## SF_RATIO_LOG  0.005606     0.051085   0.017100   0.328  0.74457
## GRADUAT_LOG   0.254060     0.120514   0.266512   0.953  0.34554
## INSTRUCT_LOG  0.499924     0.534301   0.191419   2.612  0.01220 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2863 on 45 degrees of freedom
## Multiple R-squared:  0.6853, Adjusted R-squared:  0.6363
## F-statistic:    14 on 7 and 45 DF,  p-value: 1.792e-09
```

Значимость коэффициентов

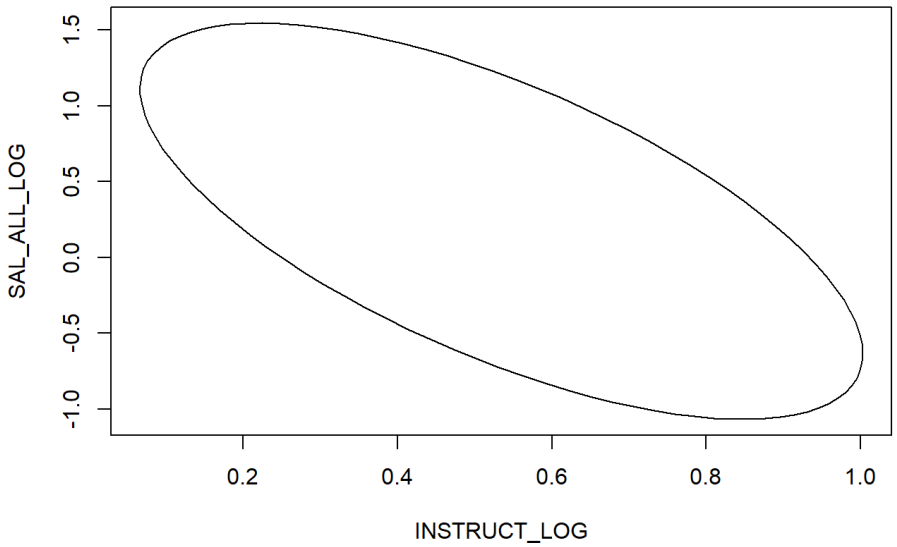
P-значения (Pr(>|t|)) показывают статистическую значимость коэффициентов. Коэффициенты с p-значением меньше 0.05 считаются статистически значимыми. В данном случае, только коэффициент для INSTRUCT является статистически значимым (p = 0.01220), что указывает на его значимое влияние на переменную NEW10.

Модель в целом

1. Multiple R-squared и Adjusted R-squared показывают долю вариативности зависимой переменной, объясненную моделью. Значение R-squared равное 0.6853 означает, что примерно 68.53% вариативности NEW10 объясняется моделью. Adjusted R-squared (скорректированный R^2) учитывает количество предикторов в модели и для этой модели равен 0.6363, что является хорошим показателем эффективности модели.
2. Residual standard error отражает среднеквадратичное отклонение остатков модели. Меньшие значения указывают на лучшее соответствие модели данным.
3. F-статистика и её p-значение тестируют нулевую гипотезу о том, что все коэффициенты модели равны нулю против альтернативной гипотезы, что хотя бы один из них не равен нулю. Значение p меньше 0.05 (в данном случае 1.792e-09) указывает на то, что модель в целом значима.

Доверительный эллипсоид

```
library(ellipse)
cov_matrix <- vcov(ModelFull)

selected_cov <- cov_matrix[c("INSTRUCT_LOG", "SAL_ALL_LOG"), c("INSTRUCT_LOG", "SAL_ALL_LOG")]

ellipse_data <- ellipse(selected_cov,
                        centre = coef(ModelFull)[c("INSTRUCT_LOG", "SAL_ALL_LOG")],
                        level = 0.95)
plot(ellipse_data,
     type = 'l')
```



Так как признак SAL_ALL_LOG не значим, то нельзя сделать выводы на основе данного эллипсоида.

Оценка мультиколлинераности

Мультиколлинеарность возникает, когда в модели регрессии присутствуют коррелирующие между собой переменные, что приводит к тому, что матрица предикторов не обладает полным рангом. Это означает, что данные не позволяют однозначно определить оценки коэффициентов методом наименьших квадратов. Кроме того, наличие мультиколлинеарности может усложнить процесс интерпретации значений коэффициентов регрессии, поскольку становится непонятно, какой из взаимосвязанных предикторов вносит больший вклад в реакцию зависимой переменной.

```
library(olsrr)
ols_vif_tol(ModelFull)
```

```
##      Variables Tolerance      VIF
## 1  ADD_FEE_LOG 0.8155906 1.226105
## 2     BOOK_LOG 0.8528172 1.172584
## 3     PH_D_LOG 0.4084779 2.448113
## 4  SAL_ALL_LOG 0.2729357 3.663867
## 5 SF_RATIO_LOG 0.2879792 3.472474
## 6  GRADUAT_LOG 0.4375726 2.285335
## 7 INSTRUCT_LOG 0.1670878 5.984877
```

1. Tolerance переменной определяется как $1 - R^2$, где R^2 — коэффициент детерминации модели регрессии, построенной для данной независимой переменной против всех остальных независимых переменных в модели. Таким образом, толерантность измеряет уникальность информации, которую несет данная переменная.
2. VIF для переменной является обратной величиной толерантности, то есть $VIF = 1/Tolerance$. VIF показывает, насколько увеличивается дисперсия коэффициента оценки при наличии мультиколлинеарности. Высокие значения VIF(>5) указывают на то, что переменная сильно коррелирует с другими переменными в модели.

Несмотря на то, что признак INSTRUCT_LOG имеет VIF>5, пока что не будем его удалять, так как он был единственным значимым в модели.

Корреляции

```
ols_correlations(ModelFull)
```

```
##                  Correlations
## -------------------------------------------------
## Variable        Zero Order    Partial     Part
## -------------------------------------------------
## ADD_FEE_LOG          0.064     -0.090    -0.051
## BOOK_LOG             0.010     -0.092    -0.052
## PH_D_LOG             0.593      0.109     0.061
## SAL_ALL_LOG          0.694      0.218     0.125
## SF_RATIO_LOG        -0.557      0.049     0.027
## GRADUAT_LOG          0.613      0.141     0.080
## INSTRUCT_LOG         0.788      0.363     0.218
## -------------------------------------------------
```

1. Zero Order Correlation — это просто корреляция Пирсона между каждой независимой переменной и зависимой переменной. Она не учитывает влияние других независимых переменных в модели. Эти значения показывают прямую связь каждой переменной с зависимой переменной без корректировки на влияние других факторов.
2. Partial Correlation — частная корреляция между независимой переменной и зависимой переменной учитывает (исключает) влияние всех других независимых переменных в модели. То есть, это корреляция между переменной и зависимой переменной при условии, что эффекты всех остальных переменных в модели контролируются или “удаляются”.

Построим новую модель, исключив признаки ADD_FEE, BOOK и SF_RATIO, так как их корреляции с зависимой перменной по таблице выше очень малы

```
library(lm.beta)

model <- lm(NEW10_LOG ~
              PH_D_LOG +
              SAL_ALL_LOG +
              GRADUAT_LOG +
              INSTRUCT_LOG,
            data = private_colleges_na)

model <- lm.beta(model)
summary(model)
```

```
##
## Call:
## lm(formula = NEW10_LOG ~ PH_D_LOG + SAL_ALL_LOG + GRADUAT_LOG +
##     INSTRUCT_LOG, data = private_colleges_na)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.55576 -0.20509  0.03505  0.16917  0.71026
##
## Coefficients:
##               Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept)  -6.882079           NA   1.962754  -3.506 0.000996 ***
## PH_D_LOG      0.005313     0.084673   0.007652   0.694 0.490828
## SAL_ALL_LOG   0.688999     0.206455   0.407452   1.691 0.097323 .
## GRADUAT_LOG   0.277519     0.131641   0.257195   1.079 0.285971
## INSTRUCT_LOG  0.482382     0.515552   0.115076   4.192 0.000118 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2794 on 48 degrees of freedom
## Multiple R-squared:  0.6804, Adjusted R-squared:  0.6537
## F-statistic: 25.54 on 4 and 48 DF,  p-value: 2.24e-11
```

Оценка мультиколлинеараности для новой модели

Видим, что значения “выровнялись”: нет сильно больших значений VIF и малых значений Tolerance.

```
library(olsrr)
ols_vif_tol(model)
```

```
##      Variables Tolerance      VIF
## 1     PH_D_LOG 0.4477510 2.233384
## 2  SAL_ALL_LOG 0.4467219 2.238529
## 3  GRADUAT_LOG 0.4473857 2.235208
## 4 INSTRUCT_LOG 0.4402240 2.271571
```

Корреляции для новой модели

```
ols_correlations(model)
```

```
##                  Correlations
## -----------------------------------------------
## Variable        Zero Order    Partial    Part
## -----------------------------------------------
## PH_D_LOG             0.593      0.100    0.057
## SAL_ALL_LOG          0.694      0.237    0.138
## GRADUAT_LOG          0.613      0.154    0.088
## INSTRUCT_LOG         0.788      0.518    0.342
## -----------------------------------------------
```

Автоматическая пошаговая регрессия по AIC

AIC — инофрмационный критерий.

$$\mathrm{AIC} = 2k - 2\ln L,$$

где k — количество параметров модели, L — значение функции правдоподобия в точке ОМП(ее максимальное значение). Выбираем модель с меньшим значением AIC (так как вычитается большее).

Backward

```
library(MASS)
backward_model <- stepAIC(ModelFull, direction = "backward")
```

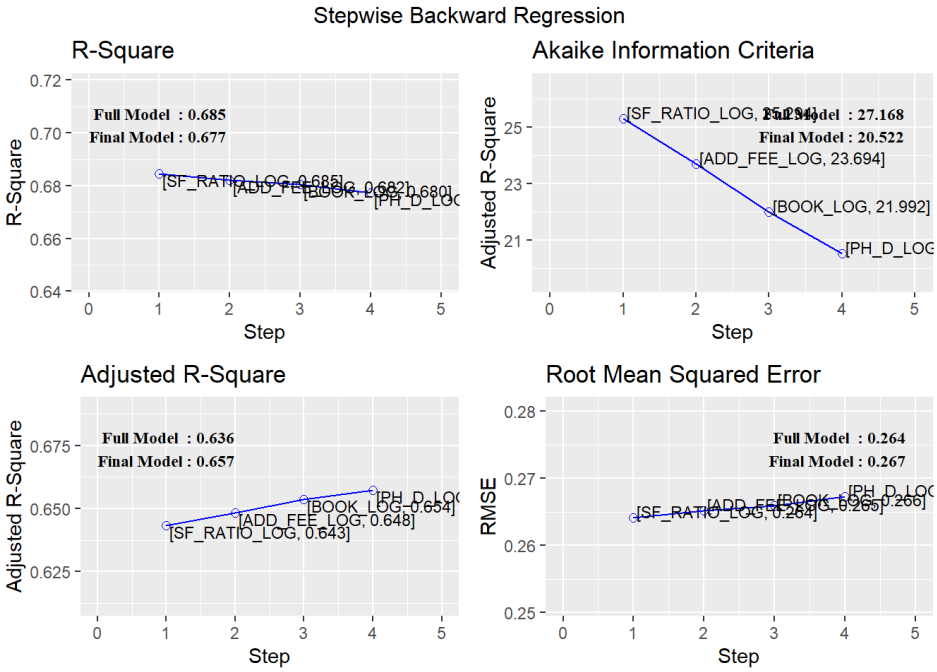
```
## Start:  AIC=-125.24
## NEW10_LOG ~ ADD_FEE_LOG + BOOK_LOG + PH_D_LOG + SAL_ALL_LOG +
##     SF_RATIO_LOG + GRADUAT_LOG + INSTRUCT_LOG
##
##                Df Sum of Sq    RSS     AIC
## - SF_RATIO_LOG  1   0.00881 3.6979 -127.11
## - ADD_FEE_LOG   1   0.03031 3.7194 -126.81
## - BOOK_LOG      1   0.03120 3.7203 -126.79
## - PH_D_LOG      1   0.04413 3.7332 -126.61
## - GRADUAT_LOG   1   0.07450 3.7636 -126.18
## <none>                      3.6891 -125.24
## - SAL_ALL_LOG   1   0.18333 3.8724 -124.67
## - INSTRUCT_LOG  1   0.55917 4.2483 -119.76
##
## Step:  AIC=-127.11
## NEW10_LOG ~ ADD_FEE_LOG + BOOK_LOG + PH_D_LOG + SAL_ALL_LOG +
##     GRADUAT_LOG + INSTRUCT_LOG
##
##                Df Sum of Sq    RSS     AIC
## - ADD_FEE_LOG   1   0.02801 3.7259 -128.71
## - BOOK_LOG      1   0.03048 3.7284 -128.68
## - PH_D_LOG      1   0.03667 3.7346 -128.59
## - GRADUAT_LOG   1   0.07957 3.7775 -127.98
## <none>                      3.6979 -127.11
## - SAL_ALL_LOG   1   0.27015 3.9681 -125.38
## - INSTRUCT_LOG  1   1.08618 4.7841 -115.47
##
## Step:  AIC=-128.71
## NEW10_LOG ~ BOOK_LOG + PH_D_LOG + SAL_ALL_LOG + GRADUAT_LOG +
##     INSTRUCT_LOG
##
##                Df Sum of Sq    RSS     AIC
## - BOOK_LOG      1   0.02101 3.7470 -130.42
## - PH_D_LOG      1   0.03285 3.7588 -130.25
## - GRADUAT_LOG   1   0.09256 3.8185 -129.41
## <none>                      3.7259 -128.71
## - SAL_ALL_LOG   1   0.24381 3.9698 -127.35
## - INSTRUCT_LOG  1   1.24078 4.9667 -115.48
##
## Step:  AIC=-130.42
## NEW10_LOG ~ PH_D_LOG + SAL_ALL_LOG + GRADUAT_LOG + INSTRUCT_LOG
##
##                Df Sum of Sq    RSS     AIC
## - PH_D_LOG      1   0.03763 3.7846 -131.89
## - GRADUAT_LOG   1   0.09089 3.8378 -131.15
## <none>                      3.7470 -130.42
## - SAL_ALL_LOG   1   0.22321 3.9702 -129.35
## - INSTRUCT_LOG  1   1.37167 5.1186 -115.88
##
## Step:  AIC=-131.89
## NEW10_LOG ~ SAL_ALL_LOG + GRADUAT_LOG + INSTRUCT_LOG
##
##                Df Sum of Sq    RSS     AIC
## <none>                      3.7846 -131.89
## - GRADUAT_LOG   1   0.21757 4.0022 -130.92
## - SAL_ALL_LOG   1   0.27874 4.0633 -130.12
## - INSTRUCT_LOG  1   1.44004 5.2246 -116.80
```

Итоговая модель по backward:

```
print(summary(backward_model))
```

```
##
## Call:
## lm(formula = NEW10_LOG ~ SAL_ALL_LOG + GRADUAT_LOG + INSTRUCT_LOG,
##     data = private_colleges_na)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5847 -0.1851  0.0295  0.1640  0.7292
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -7.2874     1.8640  -3.910 0.000284 ***
## SAL_ALL_LOG    0.7511     0.3954   1.900 0.063364 .
## GRADUAT_LOG    0.3689     0.2198   1.678 0.099642 .
## INSTRUCT_LOG   0.4912     0.1138   4.318 7.65e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2779 on 49 degrees of freedom
## Multiple R-squared:  0.6772, Adjusted R-squared:  0.6574
## F-statistic: 34.26 on 3 and 49 DF,  p-value: 4.397e-12
```

```
k2 <- ols_step_backward_p(ModelFull)
plot(k2)
```



Forward

```
null_model <- lm(NEW10_LOG ~ 1, data = private_colleges_na)
forward_model <- stepAIC(null_model,
                         scope = list(lower = null_model,
                                      upper = ModelFull),
                         direction = "forward")
```

```
## Start:  AIC=-77.96
## NEW10_LOG ~ 1
##
##                Df Sum of Sq     RSS      AIC
## + INSTRUCT_LOG  1    7.2740  4.4488 -127.316
## + SAL_ALL_LOG   1    5.6510  6.0718 -110.832
## + GRADUAT_LOG   1    4.4106  7.3121 -100.980
## + PH_D_LOG      1    4.1156  7.6072  -98.883
## + SF_RATIO_LOG  1    3.6319  8.0909  -95.616
## <none>                      11.7228  -77.964
## + ADD_FEE_LOG   1    0.0479 11.6749  -76.181
## + BOOK_LOG      1    0.0011 11.7217  -75.969
##
## Step:  AIC=-127.32
## NEW10_LOG ~ INSTRUCT_LOG
##
##                Df Sum of Sq    RSS     AIC
## + SAL_ALL_LOG   1   0.44662 4.0022 -130.92
## + GRADUAT_LOG   1   0.38545 4.0633 -130.12
## + PH_D_LOG      1   0.35563 4.0931 -129.73
## <none>                      4.4488 -127.32
## + SF_RATIO_LOG  1   0.08648 4.3623 -126.36
## + ADD_FEE_LOG   1   0.00825 4.4405 -125.42
## + BOOK_LOG      1   0.00298 4.4458 -125.35
##
## Step:  AIC=-130.92
## NEW10_LOG ~ INSTRUCT_LOG + SAL_ALL_LOG
##
##                Df Sum of Sq    RSS     AIC
## + GRADUAT_LOG   1  0.217569 3.7846 -131.89
## + PH_D_LOG      1  0.164315 3.8378 -131.15
## <none>                      4.0022 -130.92
## + BOOK_LOG      1  0.029967 3.9722 -129.32
## + ADD_FEE_LOG   1  0.026861 3.9753 -129.28
## + SF_RATIO_LOG  1  0.000165 4.0020 -128.93
##
## Step:  AIC=-131.89
## NEW10_LOG ~ INSTRUCT_LOG + SAL_ALL_LOG + GRADUAT_LOG
##
##                Df Sum of Sq    RSS     AIC
## <none>                      3.7846 -131.89
## + PH_D_LOG      1  0.037632 3.7470 -130.42
## + BOOK_LOG      1  0.025784 3.7588 -130.25
## + ADD_FEE_LOG   1  0.014525 3.7701 -130.09
## + SF_RATIO_LOG  1  0.000533 3.7840 -129.89
```

Итоговая модель по forward:

```
print(summary(forward_model))
```

```
##
## Call:
## lm(formula = NEW10_LOG ~ INSTRUCT_LOG + SAL_ALL_LOG + GRADUAT_LOG,
##     data = private_colleges_na)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5847 -0.1851  0.0295  0.1640  0.7292
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -7.2874     1.8640  -3.910 0.000284 ***
## INSTRUCT_LOG   0.4912     0.1138   4.318 7.65e-05 ***
## SAL_ALL_LOG    0.7511     0.3954   1.900 0.063364 .
## GRADUAT_LOG    0.3689     0.2198   1.678 0.099642 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2779 on 49 degrees of freedom
## Multiple R-squared:  0.6772, Adjusted R-squared:  0.6574
## F-statistic: 34.26 on 3 and 49 DF,  p-value: 4.397e-12
```

```
k1 <- ols_step_forward_p(ModelFull)
plot(k1)
```

Backward и forward дали одинаковые результаты, построим итоговую модель, оставив только 3 признака:

Итоговая модель:

```
library(lm.beta)

reduced_model <- lm(NEW10_LOG ~
              SAL_ALL_LOG +
              GRADUAT_LOG +
              INSTRUCT_LOG,
            data = private_colleges_na)

reduced_model <- lm.beta(reduced_model)
summary(reduced_model)
```

```
##
## Call:
## lm(formula = NEW10_LOG ~ SAL_ALL_LOG + GRADUAT_LOG + INSTRUCT_LOG,
##     data = private_colleges_na)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5847 -0.1851  0.0295  0.1640  0.7292
##
## Coefficients:
##              Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept)   -7.2874           NA     1.8640  -3.910 0.000284 ***
## SAL_ALL_LOG    0.7511       0.2251     0.3954   1.900 0.063364 .
## GRADUAT_LOG    0.3689       0.1750     0.2198   1.678 0.099642 .
## INSTRUCT_LOG   0.4912       0.5250     0.1138   4.318 7.65e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2779 on 49 degrees of freedom
## Multiple R-squared:  0.6772, Adjusted R-squared:  0.6574
## F-statistic: 34.26 on 3 and 49 DF,  p-value: 4.397e-12
```

Заметим, что SAL_ALL_LOG и GRADUAT_LOG стали значимы, а также увеличились значения Multiple R-squared и Adjusted R-squared.

Посмтрим на корреляции и мультиколлинераность новой модели:

```
ols_vif_tol(reduced_model)
```

```
##      Variables Tolerance      VIF
## 1  SAL_ALL_LOG 0.4693680 2.130524
## 2  GRADUAT_LOG 0.6061049 1.649879
## 3 INSTRUCT_LOG 0.4456808 2.243758
```

Все показатели VIF меньше 5, значит, мультиколлинеарности нет.

```
ols_correlations(reduced_model)
```

```
##                  Correlations
## -----------------------------------------------
## Variable        Zero Order    Partial    Part
## -----------------------------------------------
## SAL_ALL_LOG          0.694      0.262    0.154
## GRADUAT_LOG          0.613      0.233    0.136
## INSTRUCT_LOG         0.788      0.525    0.350
## -----------------------------------------------
```

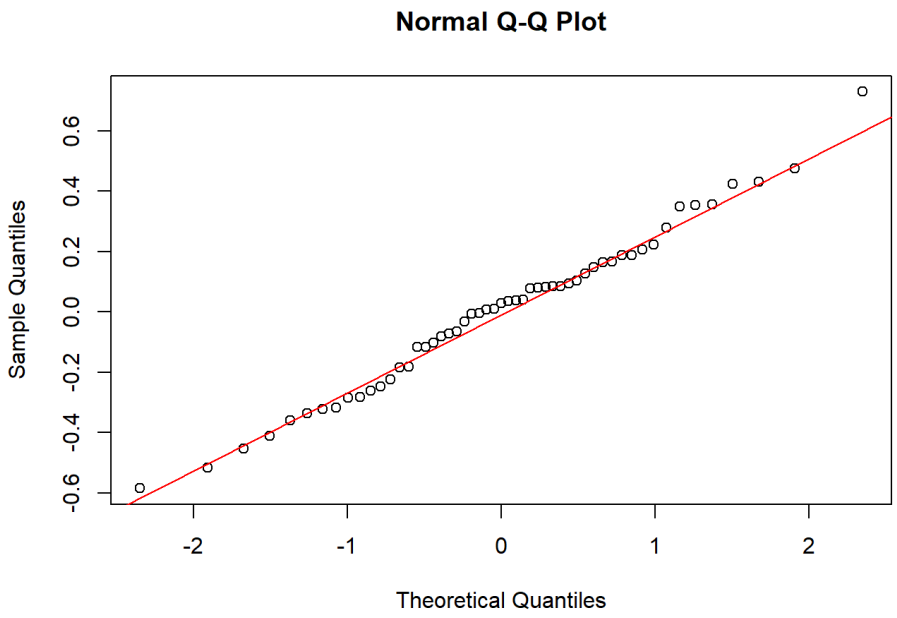
Частные корреляции увеличились и теперь все регрессоры влияют на зависимую переменную.

Нормальность остатков

Нормальность остатков имеет ключевое значение по нескольким причинам:

1. Точность статистических тестов: Нормальное распределение остатков обеспечивает точность при проверке статистических гипотез.
2. Соответствие оценок OLS и MLE: Если остатки распределены нормально, оценки коэффициентов регрессии, полученные методом наименьших квадратов (OLS), совпадают с оценками, полученными методом максимального правдоподобия (MLE). Это совпадение делает оценки асимптотически эффективными, что является одним из преимуществ метода максимального правдоподобия.

```
qqnorm(residuals(reduced_model))
qqline(residuals(reduced_model), col = "red")
```



Точки отклоняются от линии, что может свидетельствовать о наличии выбросов или нарушении нормальности, но так как данных мало, такая ситуация естественна.

Проведение теста Шапиро-Уилка на остатках (H0: остатки распределены нормально)

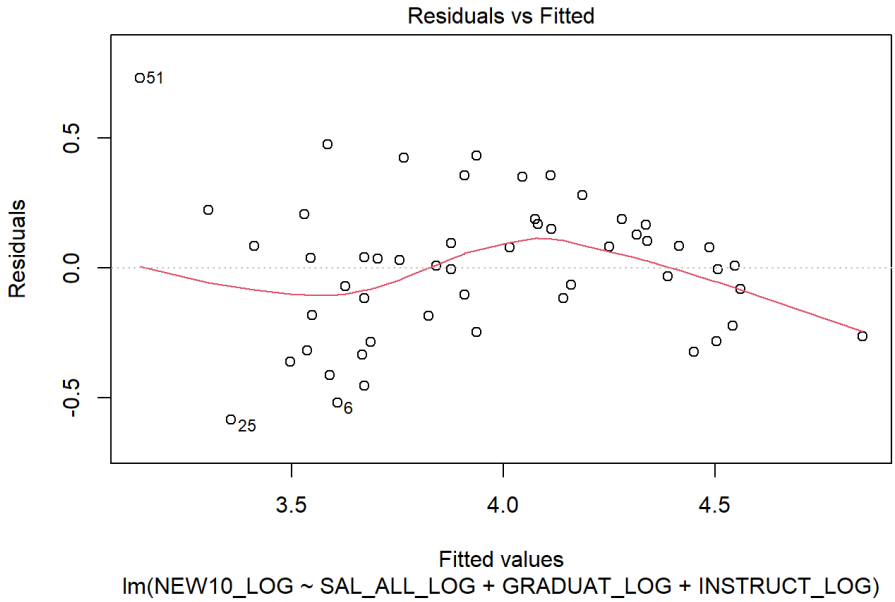
```
shapiro.test(residuals(reduced_model))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(reduced_model)
## W = 0.9892, p-value = 0.9116
```

Можно сказать, что нет достаточных оснований отвергнуть нулевую гипотезу о нормальности распределения остатков.

Predicted vs Residuals

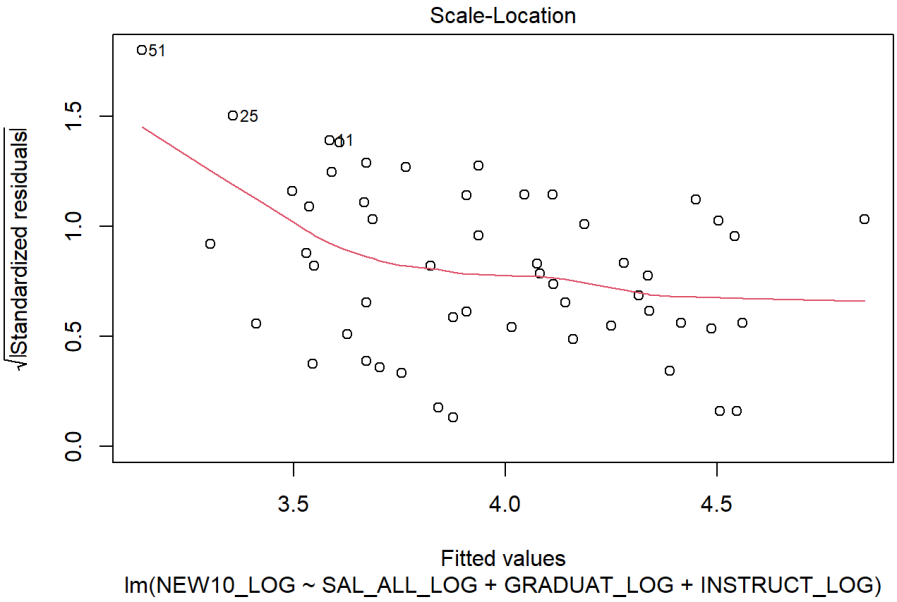
```
plot(reduced_model, which = 1)
```



По построению регрессии модель должна быть такой, чтобы остатки были ортогональны предсказаниям (проекция на регрессоры), поэтому корреляция между ними нулевая. Также ошибки независимы и одинакого распределены, поэтому для каждого значения предсказания распределение остатков выглядят одинаково.

На данном графике не видно большой нелинейности(она скорее объясняется малым объемом выборки данных), поэтому можно считать, что модель регрессии верна. Для выяснения гомоскедастичности посмотрим на другой график:

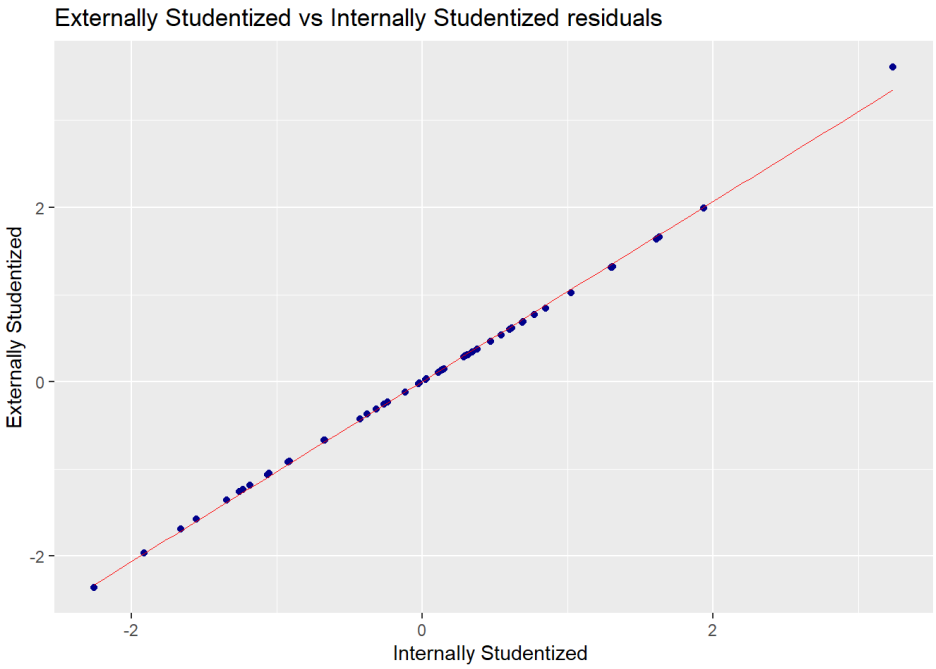
```
plot(reduced_model, which = 3)
```



Остатки гомоскедачтины, так как довольно равномерно распределны относительно $y = 0$(одинаковая дисперсия).

Residuls vs Deleted Residuals

```
library(ggplot2)
data_res <- data.frame(Internally_Studentized = rstandard(reduced_model), Externally_Studentized = rstudent(reduced_model))

ggplot(data_res, aes(x = Internally_Studentized,
                     y = Externally_Studentized)) +
  geom_point(color = "darkblue") +
  geom_smooth(method = "lm",
              se = FALSE,
              color = "red",
              lwd = 0.1) +
  labs(title = "Externally Studentized vs Internally Studentized residuals",
       x = "Internally Studentized ",
       y = "Externally Studentized")
```



На графике сравниваются 2 вида остатков(можно видеть, что разница в дисперсии, то есть сравниается она):

- **Стандартизированные остатки(Internally studentized)** рассчитываются по формуле:

$$\frac{r_i}{\hat{\sigma}\sqrt{1-h_{ii}}}$$

- **Стьюдентизированные удалённые остатки(Externally studentized)** рассчитываются по формуле:

$$\frac{r_i}{\hat{\sigma}^{(i)}\sqrt{1-h_{ii}}}$$

  где $\sigma^{(i)}$ — это оценка стандартного отклонения остатков, полученная после исключения i-го наблюдения из данных, а h_{ii} — рычаг $(D(y_i - \hat{y}_i) = Dr_i = \sigma^2(1 - h_{ii}))$.

Сравнение этих двух типов остатков на графике позволяет выявить аномальные данные, такие как выбросы или наблюдения с чрезмерно большим влиянием на результаты регрессии. Удаленные остатки по модулю обычно больше, чем просто остатки. Поэтому в идеале в I четверти они должны лежать над прямой, а в III четверти под ней.

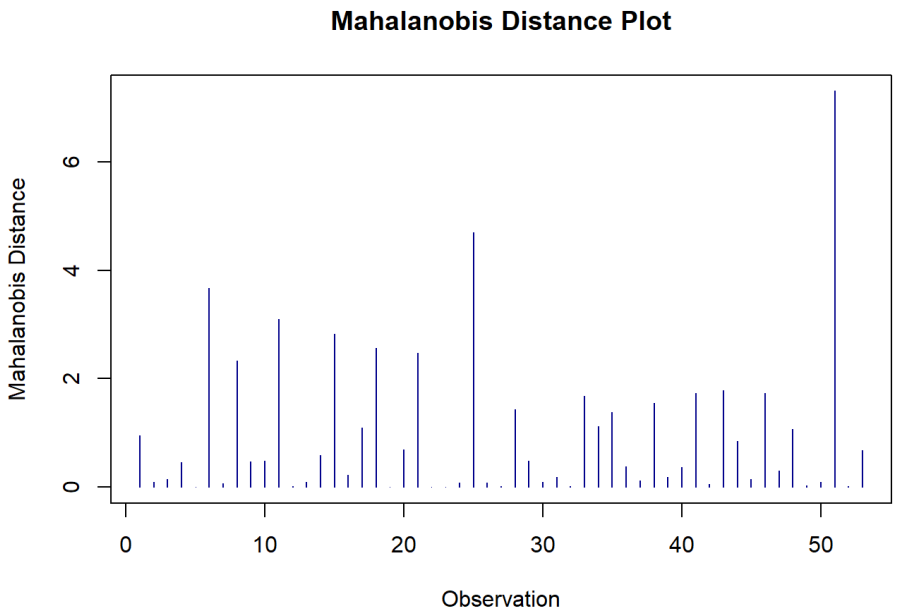
На таком графике большинство точек должно лежать вдоль линии $y = x$, что указывает на то, что оба типа остатков имеют схожие дисперсии. Однако точки, расположенные далеко от этой линии, могут указывать на наличие выбросов. Здесь виден 1 выброс.

Выбросы по Махаланобису

Расстояние Махаланобиса служит индикатором для выявления выбросов среди независимых переменных, позволяя оценить, насколько далеко каждое наблюдение отклоняется от среднего значения регрессоров.

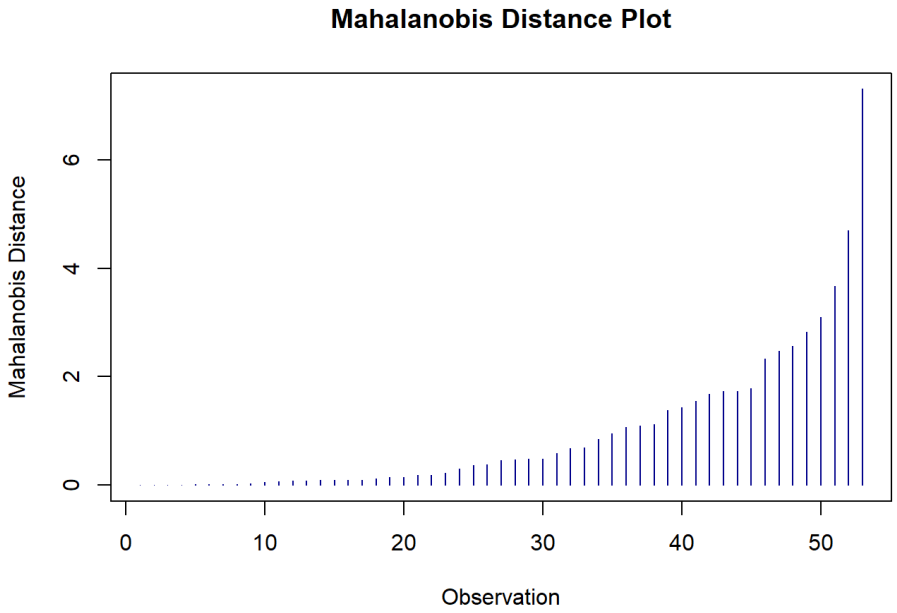
$$D_i = r_M^2(x_i, \overline{\boldsymbol{x}}, \boldsymbol{S_{xx}}) = (n-1)(1-h_{ii}) = (x_i - \overline{\boldsymbol{x}})^T \boldsymbol{S_{xx}^{-1}}(x_i - \overline{\boldsymbol{x}}).$$

```
residuals <- residuals(reduced_model)
residuals_matrix <- as.matrix(residuals)
cov_matrix <- cov(residuals_matrix)

mahalanobis_distance <- mahalanobis(residuals_matrix, center = colMeans(residuals_matrix), cov = cov_matrix)

df <- length(coef(reduced_model))
alpha <- 0.05
mahalanobis_threshold <- qchisq(1 - alpha, df)


plot(mahalanobis_distance,
     main = "Mahalanobis Distance Plot",
     xlab = "Observation",
     ylab = "Mahalanobis Distance",
     type = "h",
     col = "darkblue")

abline(h = mahalanobis_threshold, col = "red", lty = 2)
```



Упорядочим:

```
residuals <- residuals(reduced_model)
residuals_matrix <- as.matrix(residuals)
cov_matrix <- cov(residuals_matrix)

mahalanobis_distance <- mahalanobis(residuals_matrix, center = colMeans(residuals_matrix), cov = cov_matrix)

df <- length(coef(reduced_model))
alpha <- 0.05
mahalanobis_threshold <- qchisq(1 - alpha, df)



sorted_indices <- order(mahalanobis_distance)
sorted_cooks_distance <- mahalanobis_distance[sorted_indices]


plot(sorted_cooks_distance,
     main = "Mahalanobis Distance Plot",
     xlab = "Observation",
     ylab = "Mahalanobis Distance",
     type = "h",
     col = "darkblue")

abline(h = mahalanobis_threshold, col = "red", lty = 2)
```



Здесь нет выбросов.

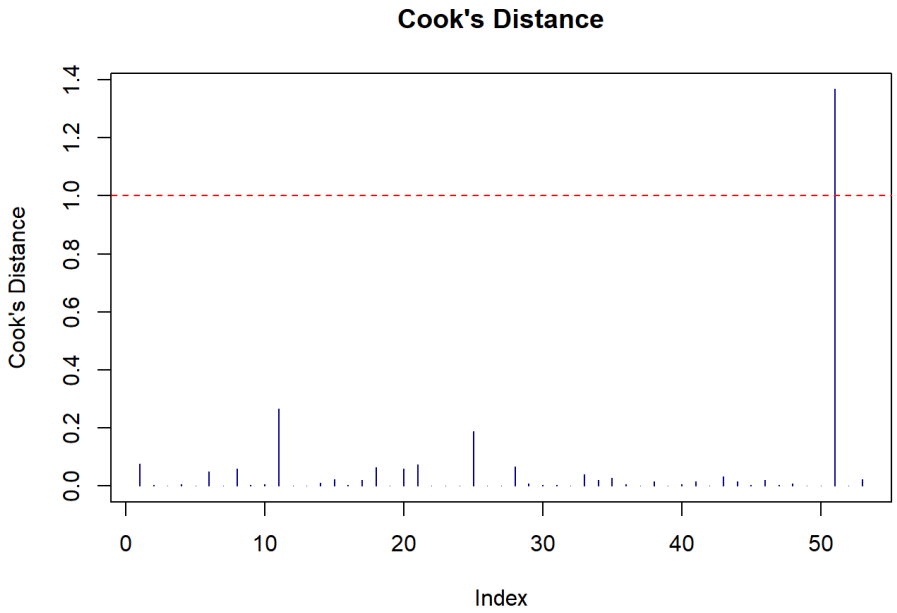
Выбросы по Куку

Это расстояние Махаланобиса между оценками коэффициентов и оценками коэффициентов регрессии, полученными по выборке без i-го индивида.

$$D_i = r_M^2(\hat{b}, \hat{b}^{(i)}, \mathrm{cov}\hat{b})$$

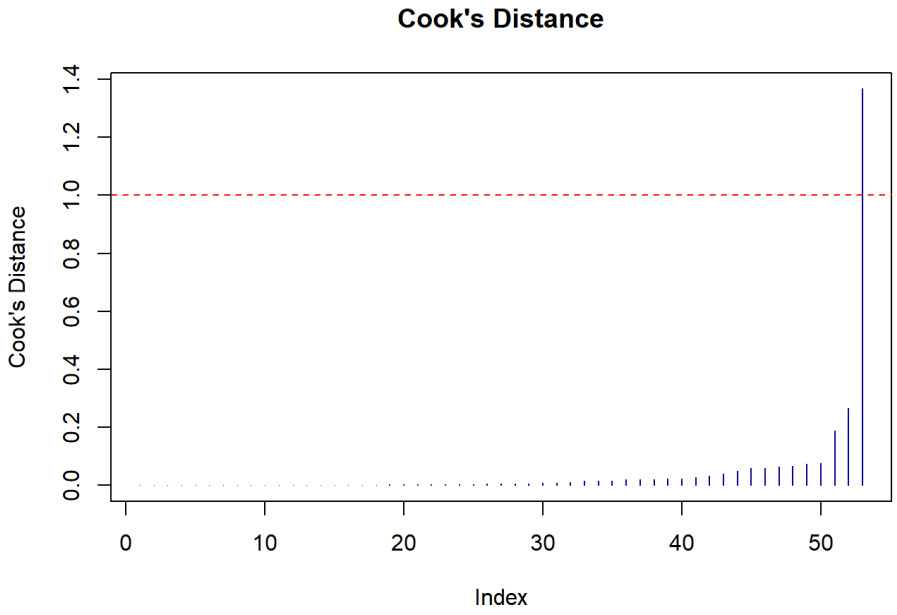
Расстояние Кука используется для идентификации точек данных, которые оказывают влияние на определенные параметры регрессионной модели. Этот показатель помогает обнаружить наблюдения, отклонение которых от общей модели регрессии является существенным.

```
cooks_distance <- cooks.distance(reduced_model)

plot(cooks_distance,
col = "darkblue",
type = "h",
main = "Cook's Distance",
ylab = "Cook's Distance")

abline(h = 1, col = "red", lty = 2)
```



Упорядочим:

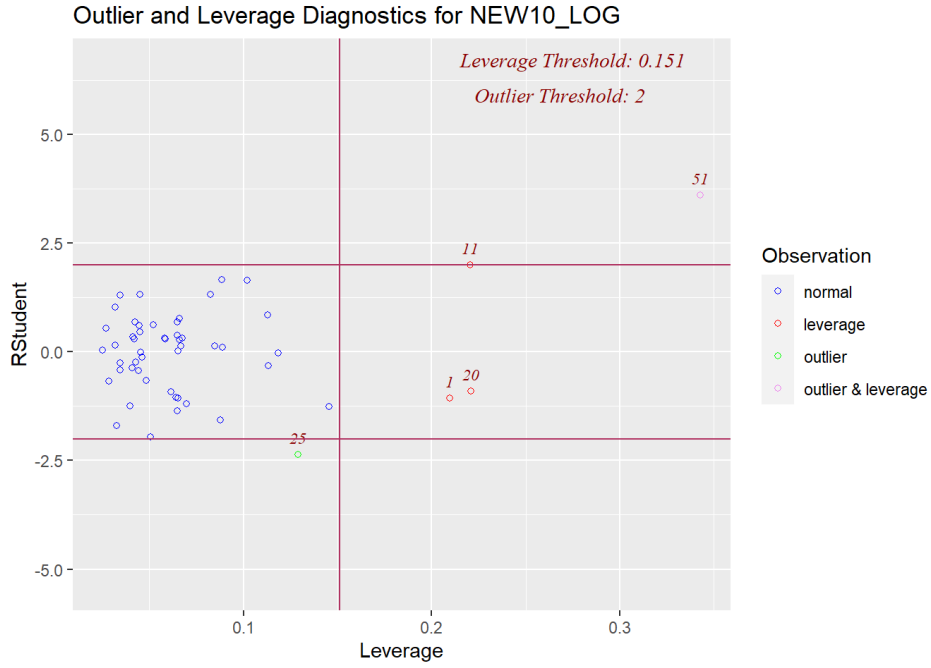
```
cooks_distance <- cooks.distance(reduced_model)
sorted_indices <- order(cooks_distance)

sorted_cooks_distance <- cooks_distance[sorted_indices]

plot(sorted_cooks_distance,
col = "darkblue",
type = "h",
main = "Cook's Distance",
ylab = "Cook's Distance")

abline(h = 1, col = "red", lty = 2)
```



Из этого графика следует, что наблюдение под номером 51 можно классифицировать как выброс, поскольку есть резкое увеличение расстояния по сравнению с другими данными.

Outlier and Leverage

```
ols_plot_resid_lev(reduced_model)
```



Оси графика:

1. Ось X: На этой оси отображаются значения рычага для каждого наблюдения. Значение рычага для каждого наблюдения отражает его потенциальное влияние на модель. Большие значения рычага указывают на наблюдения, которые имеют большое влияние на построенную модель ($D(y_i - \hat{y}_i) = Dr_i = \sigma^2(1 - h_{ii})$).
2. Ось Y: На этой оси отображаются студентизированные остатки (остатки, делённые на оценку их стандартного отклонения, учитывая число степеней свободы $\frac{r_i}{\hat{\sigma}^{(i)}\sqrt{1-h_{ii}}}$).

Из представленного графика следует, что наблюдение номер 51 является выбросом. Мы не будем удалять другие наблюдения, которые график определяет как выбросы, поскольку они входят в 5% данных, отклоняющихся более чем на два стандартных отклонения. Это соответствует общепринятому статистическому правилу, согласно которому такое отклонение считается нормальным.

Выведем выброс 51:

```
selected_rows <- private_colleges[c(51), ]
print_df(selected_rows)
```

...1	PPIND	FICE	STATE	TYPE	AVRMATH	AVRVERB	AVRCOMB	AVR_ACT	MATH_1	MATH_3	VERB_1	VERB_3
Brigham Young Univer	2	3670	UT	I	NA	NA	NA	NA	NA	NA	NA	NA

Удалим выброс 51:

```
outlier_rows <- c(51)

outliers <- private_colleges[outlier_rows, ]
cleaned_data <- private_colleges_na[-outlier_rows, ]
```

Построим модель без выбросов

```
library(lm.beta)

cleaned_model <- lm(NEW10_LOG ~
                      SAL_ALL_LOG +
                      GRADUAT_LOG +
                      INSTRUCT_LOG,
                    data = cleaned_data)

cleaned_model <- lm.beta(cleaned_model)
summary(cleaned_model)
```

```
##
## Call:
## lm(formula = NEW10_LOG ~ SAL_ALL_LOG + GRADUAT_LOG + INSTRUCT_LOG,
##     data = cleaned_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.50368 -0.15602  0.01736  0.12435  0.65187
##
## Coefficients:
##              Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept)   -7.3255           NA     1.6699  -4.387 6.27e-05 ***
## SAL_ALL_LOG    0.4221       0.1258     0.3657   1.154 0.254166
## GRADUAT_LOG    0.8192       0.3305     0.2330   3.516 0.000969 ***
## INSTRUCT_LOG   0.5085       0.5283     0.1020   4.984 8.52e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.249 on 48 degrees of freedom
## Multiple R-squared:  0.746,  Adjusted R-squared:  0.7302
## F-statistic:    47 on 3 and 48 DF,  p-value: 2.528e-14
```

Видно, что после удаления выброса качество регрессионной модели улучшилось. Мы видим более высокий скорректированный R-квадрат (Adjusted R-squared: 0.7302), больше признаков стали значимыми, значение p-value уменьшилось.

Прогноз

```
new_data <- data.frame(SAL_ALL_LOG = c(log(600)),
                       GRADUAT_LOG = c(log(90)),
                       INSTRUCT_LOG = c(log(19000)))

log_predictions <- predict(cleaned_model, newdata = new_data)

predictions <- exp(log_predictions)

predictions
```

```
##        1
## 58.59051
```

Доверительный и предсказательный интервалы

Доверительный интервал представляет собой оценку того диапазона, в пределах которого, с определённой степенью уверенности, ожидается нахождение среднего значения прогнозируемой переменной.

Предсказательный интервал представляет собой оценку того диапазона, в который с определённой вероятностью попадает будущее значение наблюдаемой переменной. Этот интервал используется для измерения неопределённости, связанной с прогнозами новых наблюдений.

По сравнению с доверительным интервалом, который оценивает неопределённость среднего значения, предсказательный интервал более широк. Это различие объясняется тем, что предсказательный интервал учитывает не только ошибку в оценке параметров модели, но и вариабельность самой случайной ошибки в предсказываемых данных.

```
conf_int_log <- predict(cleaned_model, newdata = new_data, interval = "confidence")
pred_int_log <- predict(cleaned_model, newdata = new_data, interval = "prediction")

conf_int <- exp(conf_int_log)
pred_int <- exp(pred_int_log)


cat("Confident interval:\n")
```

```
## Confident interval:
```

```
print(conf_int)
```

```
##        fit      lwr      upr
## 1 58.59051 53.11757 64.62734
```

```
cat("Predicted interval:\n")
```

```
## Predicted interval:
```

```
print(pred_int)
```

```
##        fit      lwr     upr
## 1 58.59051 35.18008 97.5793
```

Пусть администрация университета планирует привлечь больше студентов, которые были отлчиниками в школе. Они хотят оценить, как изменение параметров, таких как расходы на обучение, количество преподавателей со степенью Ph.D. и соотношение студентов к преподавателям, может повлиять на процент таких студентов.

Доверительный интервал помогает администрации понять, в каких пределах может измениться этот процент студентов при изменении параметров университета. Он предоставляет оценку диапазона, где мы ожидаем нахождение среднего значения процента студентов, которые были отлчиниками в школе.

С другой стороны, предсказательный интервал дает более широкую оценку диапазона, в который с определенной вероятностью могут попасть будущие значения процента студентов-бывших отличников. Этот интервал учитывает неопределенность, связанную не только с оценкой параметров модели, но и с вариабельностью случайной ошибки в данных.

Использование предсказательного интервала более полезно при планировании, так как он помогает администрации оценить вероятный диапазон изменений процента студентов, которые хорошо учились в школе, учитывая неопределенность будущих значений.