Public universities

Dmitry

25 марта 2022 г

Читаем данные

```
data<-read_xls("PUBLIC_shortname.xls") %>% data.frame()
```

```
## New names:
## * `` -> ...1
```

```
kable_new(data[1:5, 1:10])
```

...1	PPIND	FICE	STATE	TYPE	AVRMATH	AVRVERB	AVRCOMB	AVR_ACT	MATH_1
Alabama Agri. & Mech	1	1002	AL	IIA	NA	NA	NA	17	NA
University of Montev	1	1004	AL	IIA	NA	NA	NA	21	NA
Auburn University-Ma	1	1009	AL	I	575	501	1076	24	520
University of North	1	1016	AL	IIB	NA	NA	NA	NA	NA
Jacksonville State U	1	1020	AL	IIA	NA	NA	NA	20	NA

Взглянем на признаки

```
names(data)
```

```
##  [1] "...1"     "PPIND"    "FICE"     "STATE"    "TYPE"     "AVRMATH" 
##  [7] "AVRVERB"  "AVRCOMB"  "AVR_ACT"  "MATH_1"   "MATH_3"   "VERB_1"  
## [13] "VERB_3"   "ACT_1"    "ACT_3"    "APP_REC"  "APP_ACC"  "NEW_STUD"
## [19] "NEW10"    "NEW25"    "FULLTIME" "PARTTIME" "IN_STATE" "OUT_STAT"
## [25] "R_B_COST" "ROOM"     "BOARD"    "ADD_FEE"  "BOOK"     "PERSONAL"
## [31] "PH_D"     "TERM_D"   "SF_RATIO" "DONATE"   "INSTRUCT" "GRADUAT" 
## [37] "SAL_FULL" "SAL_AC"   "SAL_AS"   "SAL_ALL"  "COMP_FUL" "COMP_AC" 
## [43] "COMP_AS"  "COMP_ALL" "NUM_FULL" "NUM_AC"   "NUM_AS"   "NUM_INS" 
## [49] "NUM_ALL"
```

Отберём интересующие признаки

```
todrop<-c("FICE", "PPIND", "ROOM", "BOARD", "NUM_FULL", "NUM_AC",   "NUM_AS", "NUM_INS", "NUM_ALL", "NEW10", "NEW25", "MATH_
1", "MATH_3", "VERB_1", "VERB_3", "ACT_1", "ACT_3", "IN_STATE", "NEW_STUD", "SAL_AS", "SAL_FULL", "COMP_FUL", "COMP_AC", "CO
MP_AS", "COMP_ALL", "DONATE", "INSTRUCT", "PH_D", "FULLTIME", "PARTTIME", "BOOK", "ADD_FEE", "APP_REC", "APP_ACC", "PERSONA
L", "AVR_ACT")

northeast<-c("ME", "MA", "RI", "CT", "NH", "VT", "NY", "PA", "NJ", "DE", "MD")
southeast<-c("WV", "VA", "KY", "TN", "NC", "SC", "GA", "AL", "MS", "AS", "LA", "FL")
midwest<-c("OH", "IN", "MI", "IL", "MO", "WI", "MN", "IA", "KS", "NE", "SD", "ND")
southwest<-c("TX", "OK", "NM", "AZ")
west<-c("CO", "WY", "MT", "ID", "WA", "OR", "UT", "NV", "CA", "AK", "HI")

region<-Vectorize(function(name){
    if(name %in% northeast) {return("Northeast")}
    else if (name %in% southeast) {return("Southeast")}
    else if (name %in% midwest) {return("Midwest")}
    else if (name %in% southwest) {return("Southwest")}
    else {return("West")}
})


fdata<-data %>% mutate(APP_ACC_r = APP_ACC/APP_REC, REGION = region(STATE)) %>%
             dplyr::select(-all_of(todrop)) %>% rename(UNIV = ...1)

kable_new(fdata[1:7,])
```

UNIV	STATE	TYPE	AVRMATH	AVRVERB	AVRCOMB	OUT_STAT	R_B_COST	TERM_D	SF_RATIO	GRADUAT	SAL_AC	SAL_ALL	APP
Alabama Agri. & Mech	AL	IIA	NA	NA	NA	3400	2550	53	14.3	40	369	350	
University of Montev	AL	IIA	NA	NA	NA	4440	3030	72	18.9	51	385	388	

UNIV	STATE	TYPE	AVRMATH	AVRVERB	AVRCOMB	OUT_STAT	R_B_COST	TERM_D	SF_RATIO	GRADUAT	SAL_AC	SAL_ALL	APP
Auburn University-Ma	AL	I	575	501	1076	6300	3933	91	16.7	69	437	455	
University of North	AL	IIB	NA	NA	NA	2970	2536	68	19.4	76	412	411	
Jacksonville State U	AL	IIA	NA	NA	NA	2610	2600	67	20.1	33	389	386	
Livingston Universit	AL	IIB	NA	NA	NA	1740	2449	58	18.8	36	304	300	
Troy State Universit	AL	IIA	510	470	980	2883	2570	50	23.0	48	385	350	

Взглянем на pairs-plot

```
pairs<-fdata %>% dplyr::select(-all_of(c("TYPE", "STATE", "UNIV", "REGION"))) %>%
            ggpairs(lower = list(continuous = wrap("smooth", alpha = 0.3, size=0.7)),
                    diag=list(continuous=wrap("barDiag",binwidth = function(x) 2 * IQR(x) / (length(x)^(1/3)))))+
            theme(axis.text.x = element_text(angle = 45, hjust = 1))

typepairs<-fdata %>% dplyr::select(-all_of(c("TYPE", "STATE", "UNIV", "REGION"))) %>%
            ggpairs(ggplot2::aes(colour=fdata$TYPE),
                    lower = list(continuous = wrap("smooth", alpha = 0.3, size=0.7)),
                    diag=list(continuous=wrap("barDiag",binwidth = function(x) 2 * IQR(x) / (length(x)^(1/3)))))+
                    theme(axis.text.x = element_text(angle = 45, hjust = 1))

#pairs

#typepairs
```

Признаки с результатами экзаменов сильно коррелируют, отсекаем часть из них

```
fdata<-dplyr::select(fdata, -all_of(c("AVRVERB", "AVRCOMB")))
```

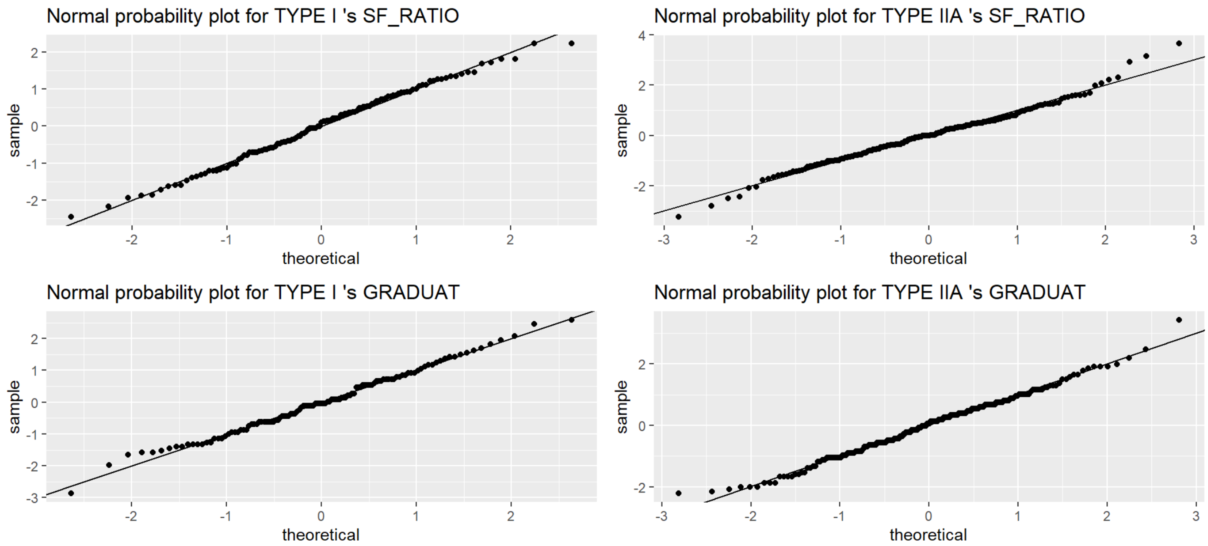
```
fdata %>% group_by(TYPE) %>% summarize(n = n())
```

```
## # A tibble: 3 x 2
##   TYPE      n
##   <chr> <int>
## 1 I       123
## 2 IIA     220
## 3 IIB      96
```

```
fdata %>% group_by(REGION) %>% summarize(n = n())
```

```
## # A tibble: 5 x 2
##   REGION        n
##   <chr>     <int>
## 1 Midwest     112
## 2 Northeast    86
## 3 Southeast   119
## 4 Southwest    45
## 5 West         77
```

```
fdata_I<-filter(fdata, TYPE == "I")
fdata_noIIB<-filter(fdata, TYPE != "IIB")

q1<-ggplot(filter(fdata_noIIB, TYPE == "I", !is.na(SF_RATIO)), aes(sample = (SF_RATIO - mean(SF_RATIO))/sd(SF_RATIO) )) + ge
om_qq() + labs(title = "Normal probability plot for TYPE I 's SF_RATIO ")+geom_abline()
q2<-ggplot(filter(fdata_noIIB, TYPE == "IIA", !is.na(SF_RATIO)), aes(sample = (SF_RATIO - mean(SF_RATIO))/sd(SF_RATIO) )) +
 geom_qq() + labs(title = "Normal probability plot for TYPE IIA 's SF_RATIO ")+geom_abline()
q3<-ggplot(filter(fdata_noIIB, TYPE == "I", !is.na(GRADUAT)), aes(sample = (GRADUAT - mean(GRADUAT))/sd(GRADUAT) )) + geom_q
q() + labs(title = "Normal probability plot for TYPE I 's GRADUAT ")+geom_abline()
q4<-ggplot(filter(fdata_noIIB, TYPE == "IIA", !is.na(GRADUAT)), aes(sample = (GRADUAT - mean(GRADUAT))/sd(GRADUAT) )) + geom
_qq() + labs(title = "Normal probability plot for TYPE IIA 's GRADUAT ")+geom_abline()

grid.arrange(q1, q2,q3,q4, nrow = 2)
```




```
shapiro.test(filter(fdata_noIIB, TYPE == "I")$SF_RATIO)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  filter(fdata_noIIB, TYPE == "I")$SF_RATIO
## W = 0.99247, p-value = 0.7501
```

```
shapiro.test(filter(fdata_noIIB, TYPE == "IIA")$SF_RATIO)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  filter(fdata_noIIB, TYPE == "IIA")$SF_RATIO
## W = 0.98653, p-value = 0.03696
```

```
shapiro.test(filter(fdata_noIIB, TYPE == "I")$GRADUAT)
```

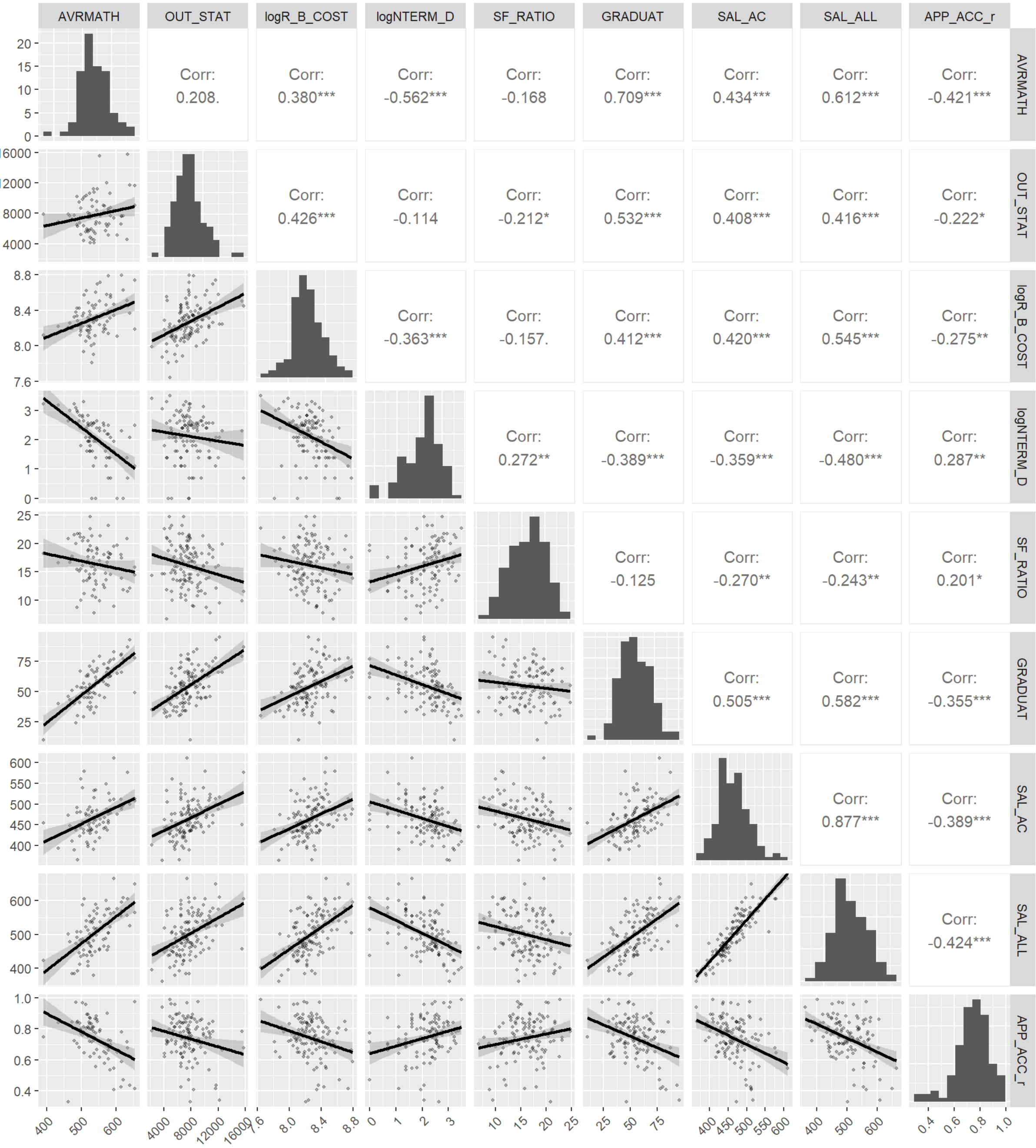
```
##
##  Shapiro-Wilk normality test
##
## data:  filter(fdata_noIIB, TYPE == "I")$GRADUAT
## W = 0.993, p-value = 0.8163
```

```
shapiro.test(filter(fdata_noIIB, TYPE == "IIA")$GRADUAT)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  filter(fdata_noIIB, TYPE == "IIA")$GRADUAT
## W = 0.99252, p-value = 0.3921
```

Рассмотрим собственно университеты. Здесь распределение переменной TERM_D скошено вправо. Развернув значения этого признака, получаем новый признак — NTERM_D — число представителей преподавательского состава без высшего образования. Он уже скошен влево и разумно его логарифмировать. Скошенность R_B_COST тоже требует логарифмирования.

```
fdata_I_t <- mutate(fdata_I, TERM_D = log(100 - TERM_D), R_B_COST = log(R_B_COST)) %>% rename(logNTERM_D = TERM_D, logR_B_CO
ST = R_B_COST)

fdata_I_t %>% dplyr::select(-all_of(c("TYPE", "STATE", "UNIV", "REGION"))) %>%
            ggpairs(lower = list(continuous = wrap("smooth", alpha = 0.3, size=0.7)), upper = list(continuous = wrap("cor",
 method = "pearson")),
                    diag=list(continuous=wrap("barDiag",binwidth = function(x) 2 * IQR(x) / (length(x)^(1/3)))))+
            theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Будем предсказывать средний результат по математике

Взглянем на отфильтрованные данные

```
kable_new(fdata_I_t[1:5,])
```

UNIV	STATE	TYPE	AVRMATH	OUT_STAT	logR_B_COST	logNTERM_D	SF_RATIO	GRADUAT	SAL_AC	SAL_ALL	APP_ACC_r	REGION
Auburn University-Ma	AL	I	575	6300	8.277	2.197	16.7	69	437	455	0.900	Southeas
University of Alabam	AL	I	NA	5424	8.169	2.197	17.3	50	447	463	0.787	Southeas
University of Alabam	AL	I	NA	4440	8.552	1.386	6.7	33	445	461	0.701	Southeas
University of Alaska	AK	I	499	5226	8.186	NA	10.0	NA	560	508	0.771	West
Arizona State Univer	AZ	I	521	7434	8.487	1.946	18.9	48	449	489	0.805	Southwe

```
summary(fdata_I_t)
```

```
##      UNIV              STATE               TYPE              AVRMATH
##  Length:123         Length:123         Length:123         Min.   :390.0
##  Class :character   Class :character   Class :character   1st Qu.:511.8
##  Mode  :character   Mode  :character   Mode  :character   Median :535.5
##                                                           Mean   :541.1
##                                                           3rd Qu.:574.2
##                                                           Max.   :655.0
##                                                           NA's   :43
##     OUT_STAT      logR_B_COST      logNTERM_D       SF_RATIO
##  Min.   : 2279   Min.   :7.641   Min.   :0.000   Min.   : 6.70
##  1st Qu.: 6326   1st Qu.:8.134   1st Qu.:1.609   1st Qu.:13.45
##  Median : 7446   Median :8.246   Median :2.303   Median :16.50
##  Mean   : 7658   Mean   :8.267   Mean   :2.133   Mean   :16.13
##  3rd Qu.: 8838   3rd Qu.:8.389   3rd Qu.:2.583   3rd Qu.:18.95
##  Max.   :15732   Max.   :8.796   Max.   :3.497   Max.   :24.70
##  NA's   :1                       NA's   :11
##     GRADUAT          SAL_AC         SAL_ALL        APP_ACC_r
##  Min.   :10.00   Min.   :364.0   Min.   :362.0   Min.   :0.3301
##  1st Qu.:44.50   1st Qu.:436.0   1st Qu.:455.0   1st Qu.:0.6767
##  Median :54.00   Median :460.0   Median :495.0   Median :0.7575
##  Mean   :54.73   Mean   :464.6   Mean   :500.2   Mean   :0.7415
##  3rd Qu.:66.00   3rd Qu.:490.0   3rd Qu.:551.0   3rd Qu.:0.8361
##  Max.   :95.00   Max.   :611.0   Max.   :665.0   Max.   :0.9886
##  NA's   :4                                       NA's   :1
##     REGION
##  Length:123
##  Class :character
##  Mode  :character
##
##
##
##
```

Немного преобразуем данные: удалим столбец с типом, разобьём их на две части: где AVR_MATH известно и где неизвестно (можем применить потом для предсказания готовые данные), пропуски заполним средними значениями (пропусков мало, искусственно дисперсию мы не занизим)

```
fdata_I_t_nT<-fdata_I_t %>% dplyr::select(-TYPE)
tdata<-fdata_I_t_nT %>% filter(!is.na(AVRMATH))
testdata<-fdata_I_t_nT %>% filter(is.na(AVRMATH)) %>% dplyr::select(-AVRMATH)

NA2mean <- function(x) replace(x, is.na(x), mean(x, na.rm = TRUE))
tdata<-replace(tdata, TRUE, lapply(tdata, NA2mean))
```

```
## Warning in mean.default(x, na.rm = TRUE): argument is not numeric or logical:
## returning NA

## Warning in mean.default(x, na.rm = TRUE): argument is not numeric or logical:
## returning NA

## Warning in mean.default(x, na.rm = TRUE): argument is not numeric or logical:
## returning NA
```

Строим стандартную модель со всеми признаками

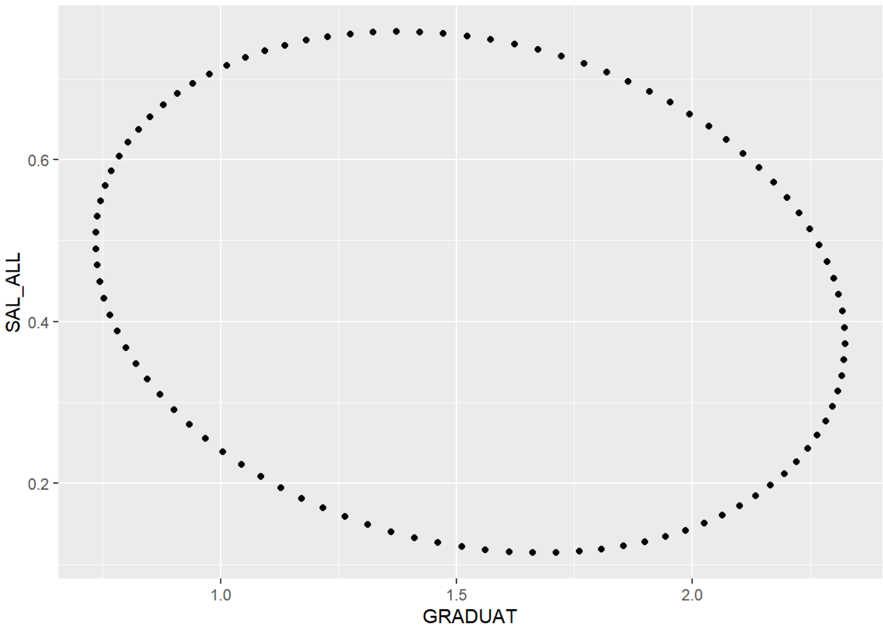
```
model_default<-lm(AVRMATH ~ OUT_STAT + logR_B_COST + logNTERM_D + SF_RATIO + GRADUAT + SAL_AC + SAL_ALL + APP_ACC_r ,data =
tdata)

summary(lm.beta(model_default))
```

```
##
## Call:
## lm(formula = AVRMATH ~ OUT_STAT + logR_B_COST + logNTERM_D +
##     SF_RATIO + GRADUAT + SAL_AC + SAL_ALL + APP_ACC_r, data = tdata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -68.407 -19.743  -3.784  18.248 131.036
##
## Coefficients:
##               Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept) 523.891586     0.000000 193.068318   2.714  0.00835 **
## OUT_STAT     -0.002053    -0.096077   0.002028  -1.012  0.31490
## logR_B_COST  -5.169727    -0.021083  23.398687  -0.221  0.82577
## logNTERM_D  -12.044243    -0.184798   5.578688  -2.159  0.03423 *
## SF_RATIO     -0.096340    -0.007215   1.084499  -0.089  0.92946
## GRADUAT       1.529480     0.477414   0.317686   4.814 8.07e-06 ***
## SAL_AC       -0.375415    -0.348017   0.163795  -2.292  0.02488 *
## SAL_ALL       0.436100     0.564773   0.128760   3.387  0.00116 **
## APP_ACC_r   -38.938710    -0.107048  29.824943  -1.306  0.19591
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31.12 on 71 degrees of freedom
## Multiple R-squared:  0.6407, Adjusted R-squared:  0.6002
## F-statistic: 15.82 on 8 and 71 DF,  p-value: 4.019e-13
```

Возможно, мы получим более хорошую модель, если поработаем с коррелированностью признаков

Взглянем на доверительный эллипс для, например, GRADUAT и SAL_ALL

```
ellipse68<-ellipse(model_default, which = c(6, 8))
ggplot()+geom_point(aes(x = ellipse68[,1], y = ellipse68[,2]))+
  xlab("GRADUAT")+ylab("SAL_ALL")
```




```
#проанализировать эллипс
```

Построим таблицу избыточности и частных корреляций

```
pcorrelations<-pcor(tdata[c(-1,-2,-12)])$estimate
spcorrelations<-spcor(tdata[c(-1,-2,-12)])$estimate

formula<-.~OUT_STAT + logR_B_COST + logNTERM_D + SF_RATIO + GRADUAT + SAL_AC + SAL_ALL + APP_ACC_r

modelOUS<-lm(update(formula, OUT_STAT ~ .-OUT_STAT), data = tdata)
modelRBC<-lm(update(formula, logR_B_COST ~ .-logR_B_COST), data = tdata)
modelNTD<-lm(update(formula, logNTERM_D ~ .-logNTERM_D), data = tdata)
modelSFR<-lm(update(formula, SF_RATIO ~ .-SF_RATIO), data = tdata)
modelGRA<-lm(update(formula, GRADUAT ~ .-GRADUAT), data = tdata)
modelSAC<-lm(update(formula, SAL_AC ~ .-SAL_AC), data = tdata)
modelSAL<-lm(update(formula, SAL_ALL ~ .-SAL_ALL), data = tdata)
modelAPC<-lm(update(formula, APP_ACC_r ~ .-APP_ACC_r), data = tdata)


mcorOUT<-cor(tdata$OUT_STAT,    modelOUS$fitted.values)^2
mcorRBC<-cor(tdata$logR_B_COST, modelRBC$fitted.values)^2
mcorNTD<-cor(tdata$logNTERM_D,  modelNTD$fitted.values)^2
mcorSFR<-cor(tdata$SF_RATIO,    modelSFR$fitted.values)^2
mcorGRA<-cor(tdata$GRADUAT,     modelGRA$fitted.values)^2
mcorSAC<-cor(tdata$SAL_AC,      modelSAC$fitted.values)^2
mcorSAL<-cor(tdata$SAL_ALL,     modelSAL$fitted.values)^2
mcorAPC<-cor(tdata$APP_ACC_r,   modelAPC$fitted.values)^2

rsq<-c(mcorOUT, mcorRBC, mcorNTD, mcorSFR, mcorGRA, mcorSAC, mcorSAL, mcorAPC)

info<-data.frame(tolerance = 1 - rsq,
                Rsq        =  rsq,
               partialcors = pcorrelations[1, 2:9],
           semipartialcors = spcorrelations[1, 2:9], row.names = names(tdata)[4:11])
print(info)
```

```
##             tolerance       Rsq partialcors semipartialcors
## OUT_STAT    0.5616913 0.4383087 -0.11926311    -0.072006234
## logR_B_COST 0.5558144 0.4441856 -0.02621185    -0.015718102
## logNTERM_D  0.6907932 0.3092068 -0.24820521    -0.153592993
## SF_RATIO    0.7672331 0.2327669 -0.01054207    -0.006319799
## GRADUAT     0.5146940 0.4853060  0.49609971     0.342507076
## SAL_AC      0.2195172 0.7804828 -0.26247097    -0.163055021
## SAL_ALL     0.1820152 0.8179848  0.37295186     0.240950499
## APP_ACC_r   0.7528322 0.2471678 -0.15311623    -0.092880795
```

Уберём logR_B_COST и SF_RATIO

```
feat.nums<-c(4,6,8:11)

model1<-lm(AVRMATH ~ OUT_STAT + logNTERM_D + GRADUAT + SAL_AC + SAL_ALL + APP_ACC_r ,data = tdata)

summary(model1)
```

```
##
## Call:
## lm(formula = AVRMATH ~ OUT_STAT + logNTERM_D + GRADUAT + SAL_AC +
##     SAL_ALL + APP_ACC_r, data = tdata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -68.088 -18.973  -3.872  17.666 132.158
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 480.522663  54.139581   8.876 3.19e-13 ***
## OUT_STAT     -0.002175   0.001767  -1.230 0.222523
## logNTERM_D  -12.063651   5.385851  -2.240 0.028144 *
## GRADUAT       1.529781   0.312813   4.890 5.81e-06 ***
## SAL_AC       -0.362562   0.151454  -2.394 0.019242 *
## SAL_ALL       0.422783   0.114705   3.686 0.000435 ***
## APP_ACC_r   -38.276019  29.295429  -1.307 0.195466
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30.7 on 73 degrees of freedom
## Multiple R-squared:  0.6404, Adjusted R-squared:  0.6108
## F-statistic: 21.66 on 6 and 73 DF,  p-value: 1.877e-14
```

Посмотрим на результаты перебора

```
leaps(tdata[,c(-1,-2,-3, -12)], tdata[, 3], method = "adjr2", names = names(tdata)[4:11], nbest = 1)
```

```
## $which
##   OUT_STAT logR_B_COST logNTERM_D SF_RATIO GRADUAT SAL_AC SAL_ALL APP_ACC_r
## 1    FALSE       FALSE      FALSE    FALSE    TRUE  FALSE   FALSE     FALSE
## 2    FALSE       FALSE      FALSE    FALSE    TRUE  FALSE    TRUE     FALSE
## 3    FALSE       FALSE       TRUE    FALSE    TRUE  FALSE    TRUE     FALSE
## 4    FALSE       FALSE       TRUE    FALSE    TRUE   TRUE    TRUE     FALSE
## 5    FALSE       FALSE       TRUE    FALSE    TRUE   TRUE    TRUE      TRUE
## 6     TRUE       FALSE       TRUE    FALSE    TRUE   TRUE    TRUE      TRUE
## 7     TRUE        TRUE       TRUE    FALSE    TRUE   TRUE    TRUE      TRUE
## 8     TRUE        TRUE       TRUE     TRUE    TRUE   TRUE    TRUE      TRUE
##
## $label
## [1] "(Intercept)" "OUT_STAT"    "logR_B_COST" "logNTERM_D"  "SF_RATIO"
## [6] "GRADUAT"     "SAL_AC"      "SAL_ALL"     "APP_ACC_r"
##
## $size
## [1] 2 3 4 5 6 7 8 9
##
## $adjr2
## [1] 0.4682037 0.5383781 0.5802288 0.6037046 0.6081005 0.6108023 0.6056797
## [8] 0.6001703
```

```
leaps(tdata[,c(-1,-2,-3, -12)], tdata[, 3], method = "Cp", names = names(tdata)[4:11], nbest = 1)
```

```
## $which
##   OUT_STAT logR_B_COST logNTERM_D SF_RATIO GRADUAT SAL_AC SAL_ALL APP_ACC_r
## 1    FALSE       FALSE      FALSE    FALSE    TRUE  FALSE   FALSE     FALSE
## 2    FALSE       FALSE      FALSE    FALSE    TRUE  FALSE    TRUE     FALSE
## 3    FALSE       FALSE       TRUE    FALSE    TRUE  FALSE    TRUE     FALSE
## 4    FALSE       FALSE       TRUE    FALSE    TRUE   TRUE    TRUE     FALSE
## 5    FALSE       FALSE       TRUE    FALSE    TRUE   TRUE    TRUE      TRUE
## 6     TRUE       FALSE       TRUE    FALSE    TRUE   TRUE    TRUE      TRUE
## 7     TRUE        TRUE       TRUE    FALSE    TRUE   TRUE    TRUE      TRUE
## 8     TRUE        TRUE       TRUE     TRUE    TRUE   TRUE    TRUE      TRUE
##
## $label
## [1] "(Intercept)" "OUT_STAT"    "logR_B_COST" "logNTERM_D"  "SF_RATIO"
## [6] "GRADUAT"     "SAL_AC"      "SAL_ALL"     "APP_ACC_r"
##
## $size
## [1] 2 3 4 5 6 7 8 9
##
## $Cp
## [1] 27.744454 14.900056  7.790495  4.337031  4.532291  5.058828  7.007891
## [8]  9.000000
```

Взглянем на пошаговую регрессию по C_p-критерию Mallow (эквивалентен AIC в нормальной модели)

```
(ols_step_backward_p(model_default))
```

```
##
##
##                              Elimination Summary
## ---------------------------------------------------------------------------
##        Variable                   Adj.
## Step     Removed      R-Square    R-Square     C(p)       AIC        RMSE
## ---------------------------------------------------------------------------
##    1    SF_RATIO         0.6406      0.6057    7.0079    785.5421    30.9038
##    2    logR_B_COST      0.6404      0.6108    5.0588    783.5994    30.7024
## ---------------------------------------------------------------------------
```

```
(ols_step_forward_p(model_default))
```

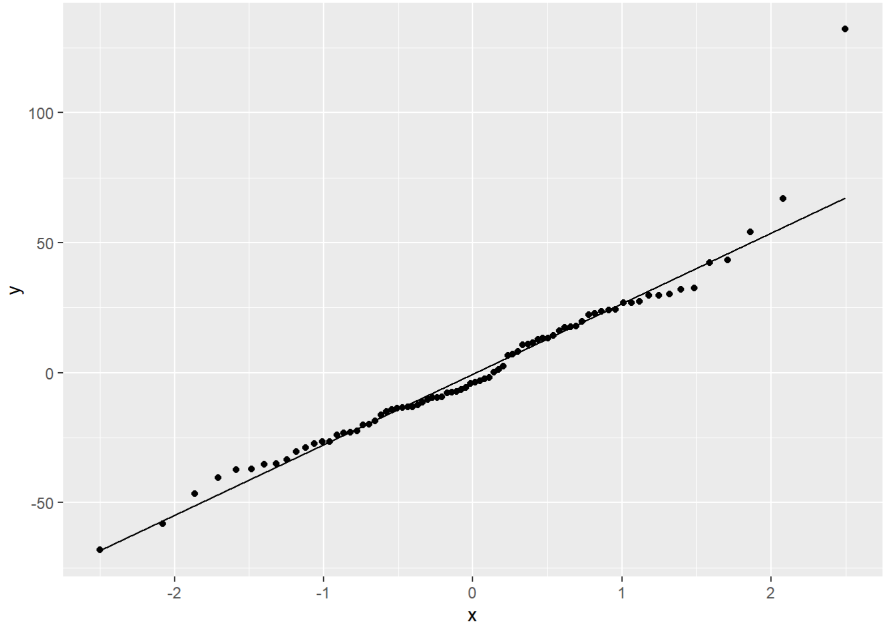
```
##
##                               Selection Summary
## ---------------------------------------------------------------------------
##        Variable                   Adj.
## Step    Entered      R-Square    R-Square     C(p)        AIC        RMSE
## ---------------------------------------------------------------------------
##    1    GRADUAT         0.4749      0.4682    27.7445    803.8732    35.8889
##    2    SAL_ALL         0.5501      0.5384    14.9001    793.5198    33.4373
##    3    logNTERM_D      0.5962      0.5802     7.7905    786.8711    31.8855
##    4    SAL_AC          0.6238      0.6037     4.3370    783.2075    30.9811
##    5    APP_ACC_r       0.6329      0.6081     4.5323    783.2413    30.8088
##    6    OUT_STAT        0.6404      0.6108     5.0588    783.5994    30.7024
## ---------------------------------------------------------------------------
```

Проанализируем остатки новой модели. Сначала нормальность

```
shapiro.test(model1$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  model1$residuals
## W = 0.93934, p-value = 0.0008877
```

```
ggplot(tdata, aes(sample = model1$residuals))+geom_qq()+geom_qq_line()
```



Каков график predicted-residuals

```
ggplot(tdata ,aes(x = model1$fitted.values,y=model1$residuals))+
  geom_point()+
  geom_text(aes(label=ifelse(abs(model1$residuals)>50,as.character(reorder(1:dim(tdata)[1],model1$fitted.values)),'')),hjust
=0,vjust=0)+
  xlab("Predicted")+ylab("Residuals")
```

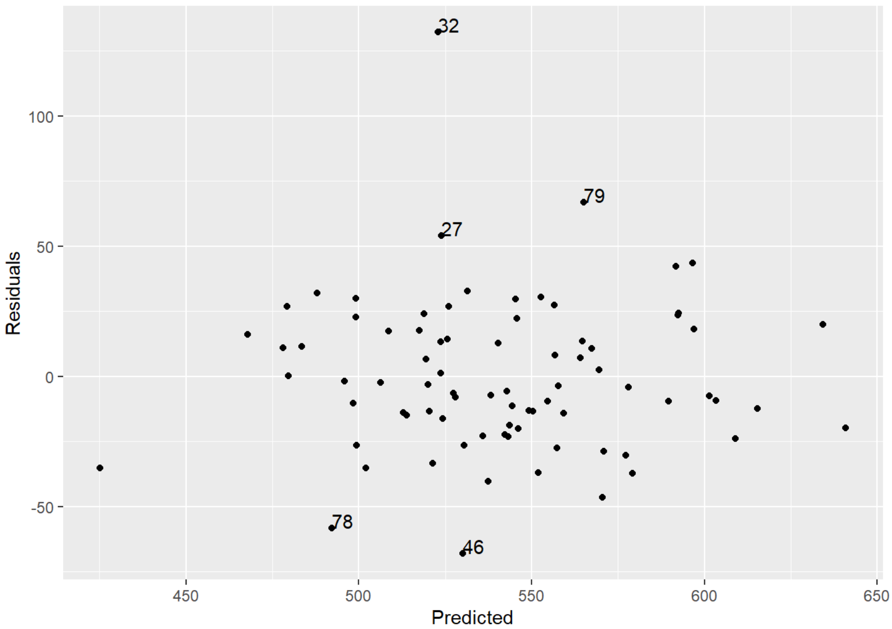


График residuals-deleted residuals

```
del.resid<-tdata$AVRMATH
form1<-AVRMATH ~ OUT_STAT + logNTERM_D + GRADUAT + SAL_AC + SAL_ALL + APP_ACC_r
for(i in 1:dim(tdata)[1])
{
  del.resid[i]<-del.resid[i]-predict.lm(lm(form1, data = tdata[-i,]), tdata[i,])
}
ggplot(tdata, aes(x = model1$residuals, y = del.resid))+
  geom_point()+
  geom_label_repel(aes(label=ifelse(abs(model1$residuals - del.resid)>7,as.character(reorder(1:dim(tdata)[1], model1$residua
ls)),'')),
                  box.padding   = 0.35,
                  point.padding = 0.5,
                  segment.color = 'grey50')+
  xlab("Residuals")+ylab("Deleted residuals")+geom_abline(slope = 1, intercept = 0)
```

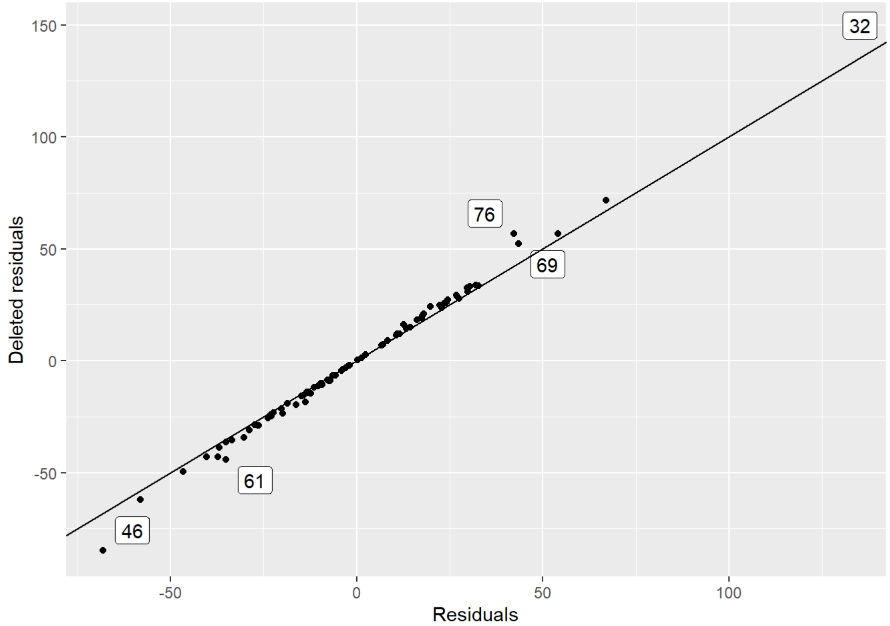


График расстояний Кука

```
gdata1<-data.frame('n' = 1:dim(tdata)[1], 'dist' = cooks.distance(model1))
ggplot(gdata1, aes(x = n, y = dist))+geom_point()+geom_text(aes(label=ifelse(dist >0.04,as.character(n),'')),hjust=0,vjust=0
)+xlab("n")+ylab("Cook's distance")
```

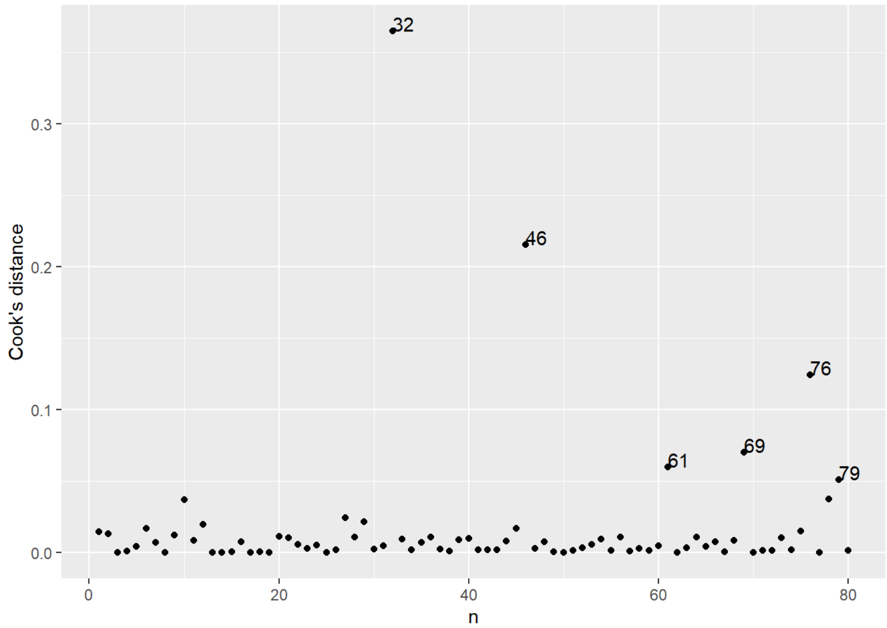
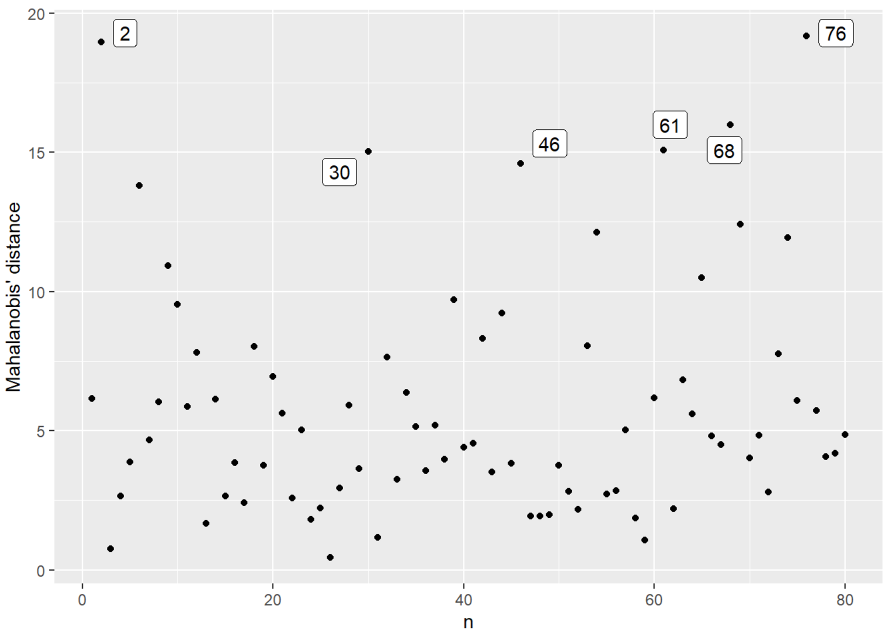
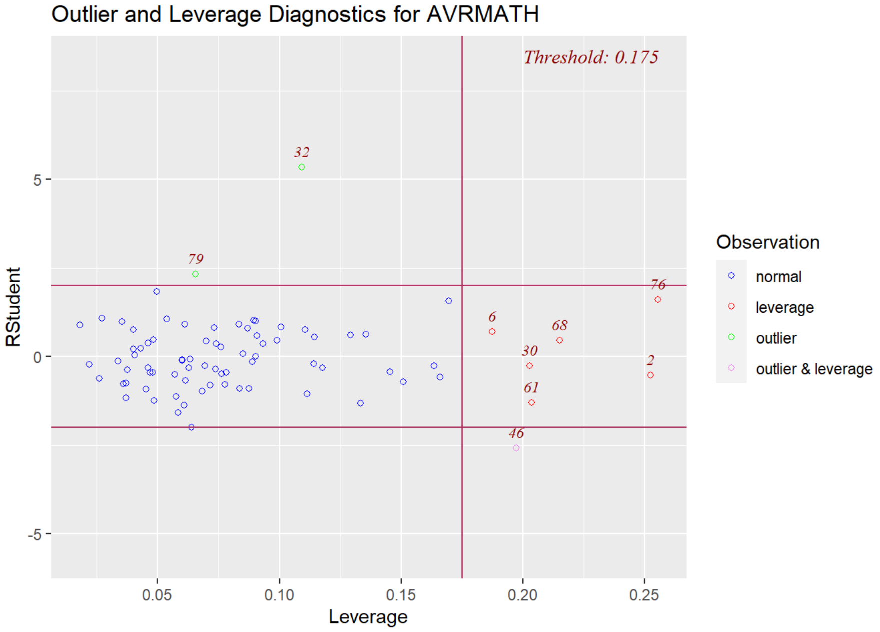


График расстояний Махаланобиса

```
gdata2<-data.frame('n' = 1:dim(tdata)[1], 'dist' = mahalanobis(tdata[feat.nums], apply(tdata[feat.nums], 2, mean) ,cov(tdata
[feat.nums])))

ggplot(gdata2, aes(x = n, y = dist))+geom_point()+geom_label_repel(aes(label=ifelse(dist >qchisq(0.95, 7),as.character(n),''
)),
                 box.padding   = 0.35,
                 point.padding = 0.5,
                 segment.color = 'grey50')+
                 xlab("n")+ylab("Mahalanobis' distance")
```




```
ols_plot_resid_lev(model1)
```



Кандидаты на удаление:

```
to.delete<-c(2,30,32,46,61,68,69,76,79)
kable_new(tdata[to.delete,])
```

	UNIV	STATE	AVRMATH	OUT_STAT	logR_B_COST	logNTERM_D	SF_RATIO	GRADUAT	SAL_AC	SAL_ALL	APP_ACC_r	REGION
2	University of Alaska	AK	499	5226	8.186	2.051	10.0	56.312	560	508	0.771	West
30	University of Southe	MS	531	4652	7.812	0.000	18.7	45.000	438	438	0.717	Southeast
32	University of Missou	MO	655	9057	8.189	2.485	14.4	49.000	508	564	0.973	Midwest

	UNIV	STATE	AVRMATH	OUT_STAT	logR_B_COST	logNTERM_D	SF_RATIO	GRADUAT	SAL_AC	SAL_ALL	APP_ACC_r	REGION
46	Kent State Universit	OH	462	7854	8.207	2.639	20.8	46.000	473	502	0.330	Midwest
61	Texas Southern Unive	TX	390	7860	8.120	3.219	18.2	10.000	455	423	0.747	Southwest
68	University of Vermon	VT	553	15516	8.503	2.303	9.9	79.000	450	458	0.784	Northeast
69	College of William a	VA	640	11720	8.366	2.079	12.1	93.000	499	525	0.436	Southeast
76	University of Michig	MI	634	15732	8.447	0.693	11.5	87.000	577	605	0.676	Midwest
79	University of Texas	TX	632	4536	8.508	1.386	20.8	56.312	505	557	0.716	Southwest

```
model2<-lm(form1, data = tdata[-to.delete,])
summary(lm.beta(model2))
```

```
##
## Call:
## lm(formula = form1, data = tdata[-to.delete, ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -66.101 -13.050   0.565  11.704  43.322
##
## Coefficients:
##               Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept)  6.213e+02    0.000e+00  4.339e+01  14.320  < 2e-16 ***
## OUT_STAT    -3.598e-03   -1.641e-01  1.522e-03  -2.364 0.021099 *
## logNTERM_D  -8.726e+00   -1.521e-01  4.251e+00  -2.053 0.044201 *
## GRADUAT      1.222e+00    4.076e-01  2.460e-01   4.964 5.40e-06 ***
## SAL_AC      -4.767e-01   -5.108e-01  1.273e-01  -3.746 0.000388 ***
## SAL_ALL      4.111e-01    6.392e-01  9.377e-02   4.384 4.43e-05 ***
## APP_ACC_r   -1.231e+02   -3.764e-01  2.436e+01  -5.054 3.87e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.14 on 64 degrees of freedom
## Multiple R-squared:  0.7623, Adjusted R-squared:   0.74
## F-statistic: 34.21 on 6 and 64 DF,  p-value: < 2.2e-16
```

R^2_{adj} улучшился значительно

Эта же модель получается при автоматическом отборе признаков по AIC в обоих направлениях

```
model_default_d<-model_default<-lm(AVRMATH ~ OUT_STAT + logR_B_COST + logNTERM_D + SF_RATIO + GRADUAT + SAL_AC + SAL_ALL + A
PP_ACC_r ,data = tdata[-to.delete,])

(ols_step_forward_aic(model_default_d))
```

```
##
##                          Selection Summary
## ---------------------------------------------------------------------
## Variable        AIC       Sum Sq         RSS        R-Sq      Adj. R-Sq
## ---------------------------------------------------------------------
## GRADUAT       691.457    55555.070    64804.113    0.46158      0.45377
## APP_ACC_r     668.464    74784.352    45574.831    0.62134      0.61021
## logNTERM_D    659.501    81305.269    39053.914    0.67552      0.66099
## OUT_STAT      658.108    83127.235    37231.949    0.69066      0.67191
## SAL_ALL       655.472    85480.912    34878.271    0.71022      0.68792
## SAL_AC        643.400    91751.735    28607.448    0.76232      0.74003
## ---------------------------------------------------------------------
```

```
(ols_step_backward_aic(model_default_d))
```

```
##
##
##                        Backward Elimination Summary
## ----------------------------------------------------------------------
## Variable         AIC         RSS        Sum Sq       R-Sq      Adj. R-Sq
## ----------------------------------------------------------------------
## Full Model     646.977    28437.685    91921.498    0.76373      0.73324
## logR_B_COST    644.980    28438.897    91920.286    0.76372      0.73746
## SF_RATIO       643.400    28607.448    91751.735    0.76232      0.74003
## ----------------------------------------------------------------------
```

Остатки нормальны

```
shapiro.test(model2$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  model2$residuals
## W = 0.98331, p-value = 0.4667
```

Пересмотрим модель

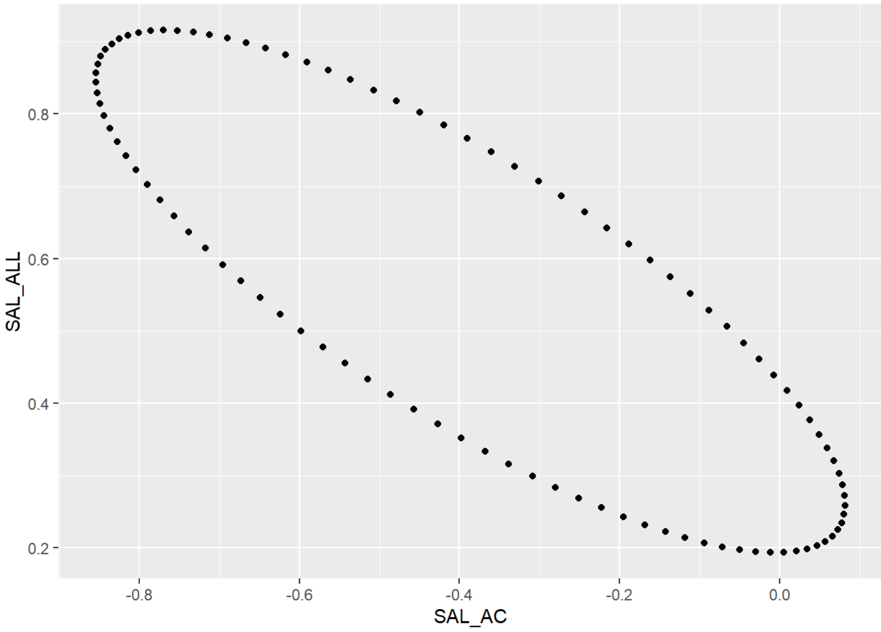
Признак GRADUAT скорее является следствием из результатов AVRMATH

```
model_default_noGRADUAT<-lm(AVRMATH ~ OUT_STAT + logR_B_COST + logNTERM_D + SF_RATIO + SAL_AC + SAL_ALL + APP_ACC_r, data =
tdata)

summary.beta(model_default_noGRADUAT)
```

```
##
## Call:
## lm(formula = AVRMATH ~ OUT_STAT + logR_B_COST + logNTERM_D +
##     SF_RATIO + SAL_AC + SAL_ALL + APP_ACC_r, data = tdata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -89.700 -17.757  -1.835  16.781 117.664
##
## Coefficients:
##               Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept) 581.183129     0.000000 220.391507   2.637 0.010239 *
## OUT_STAT      0.001823     0.085340   0.002129   0.856 0.394590
## logR_B_COST  -8.488326    -0.034617  26.749355  -0.317 0.751913
## logNTERM_D  -20.334665    -0.312000   6.068757  -3.351 0.001286 **
## SF_RATIO      0.201231     0.015070   1.238320   0.163 0.871365
## SAL_AC       -0.386467    -0.358262   0.187314  -2.063 0.042700 *
## SAL_ALL       0.554535     0.718152   0.144550   3.836 0.000265 ***
## APP_ACC_r   -61.318931    -0.168574  33.693778  -1.820 0.072933 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.59 on 72 degrees of freedom
## Multiple R-squared:  0.5233, Adjusted R-squared:  0.477
## F-statistic: 11.29 on 7 and 72 DF,  p-value: 1.444e-09
```

Посмотрим на доверит. эллипсоид для зарплат

```
ellipse67<-ellipse(model_default_noGRADUAT, which = c(6, 7))
ggplot()+geom_point(aes(x = ellipse67[,1], y = ellipse67[,2]))+
  xlab("SAL_AC")+ylab("SAL_ALL")
```



Эллипс захватывает часть прямой SAL_AC == 0, поэтому его можно исключить

```
model3<-lm(AVRMATH ~ OUT_STAT + logR_B_COST + logNTERM_D + SF_RATIO+ SAL_ALL + APP_ACC_r, data = tdata)
summary.beta(model3)
```

```
##
## Call:
## lm(formula = AVRMATH ~ OUT_STAT + logR_B_COST + logNTERM_D +
##     SF_RATIO + SAL_ALL + APP_ACC_r, data = tdata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -99.471 -19.908  -1.233  21.783 118.109
##
## Coefficients:
##               Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept) 401.781589     0.000000 206.981499   1.941 0.056101 .
## OUT_STAT      0.001325     0.062006   0.002162   0.613 0.541911
## logR_B_COST   4.795119     0.019555  26.535831   0.181 0.857101
## logNTERM_D  -21.927373    -0.336438   6.152271  -3.564 0.000648 ***
## SF_RATIO      0.816844     0.061174   1.228352   0.665 0.508151
## SAL_ALL       0.311692     0.403658   0.085763   3.634 0.000515 ***
## APP_ACC_r   -50.598867    -0.139103  34.025267  -1.487 0.141296
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.38 on 73 degrees of freedom
## Multiple R-squared:  0.4952, Adjusted R-squared:  0.4537
## F-statistic: 11.93 on 6 and 73 DF,  p-value: 2.732e-09
```

Продолжим отбор. Автоматический отбор признаком по AIC и adj. R^2 в отдельности дают:

```
(ols_step_backward_aic(model3))
```

```
##
##
##                        Backward Elimination Summary
## ---------------------------------------------------------------------
## Variable         AIC          RSS         Sum Sq       R-Sq      Adj. R-Sq
## ---------------------------------------------------------------------
## Full Model     810.730    96593.857    94744.530    0.49517      0.45367
## logR_B_COST    808.765    96637.065    94701.323    0.49494      0.46082
## SF_RATIO       807.285    97266.850    94071.538    0.49165      0.46454
## OUT_STAT       805.632    97690.072    93648.316    0.48944      0.46928
## ---------------------------------------------------------------------
```

```
(ols_step_forward_aic(model3))
```

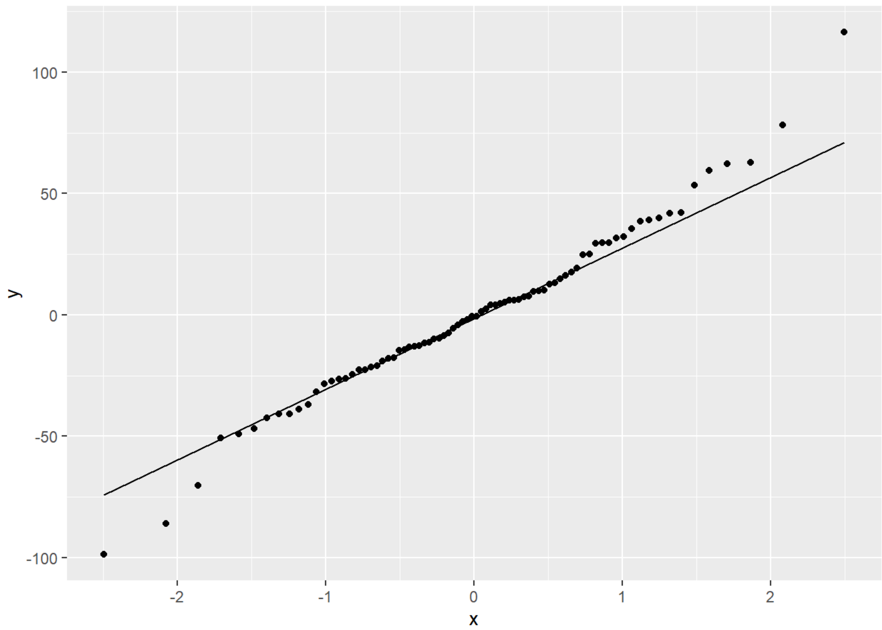
```
##
##                          Selection Summary
## -----------------------------------------------------------------------
## Variable        AIC       Sum Sq        RSS         R-Sq      Adj. R-Sq
## -----------------------------------------------------------------------
## SAL_ALL       817.890    71634.927    119703.461    0.37439      0.36637
## logNTERM_D    806.084    90608.806    100729.582    0.47355      0.45988
## APP_ACC_r     805.632    93648.316     97690.072    0.48944      0.46928
## -----------------------------------------------------------------------
```

```
model4<-lm(AVRMATH ~ logNTERM_D + SAL_ALL + APP_ACC_r, data = tdata)
summary.beta(model4)
```

```
##
## Call:
## lm(formula = AVRMATH ~ logNTERM_D + SAL_ALL + APP_ACC_r, data = tdata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -98.812 -21.166  -0.578  18.005 116.497
##
## Coefficients:
##              Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept) 456.80639      0.00000   54.81047   8.334 2.52e-12 ***
## logNTERM_D  -20.94239     -0.32132    5.91031  -3.543 0.000679 ***
## SAL_ALL       0.32588      0.42203    0.07384   4.414 3.31e-05 ***
## APP_ACC_r   -51.45573     -0.14146   33.46188  -1.538 0.128265
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.85 on 76 degrees of freedom
## Multiple R-squared:  0.4894, Adjusted R-squared:  0.4693
## F-statistic: 24.29 on 3 and 76 DF,  p-value: 4.008e-11
```

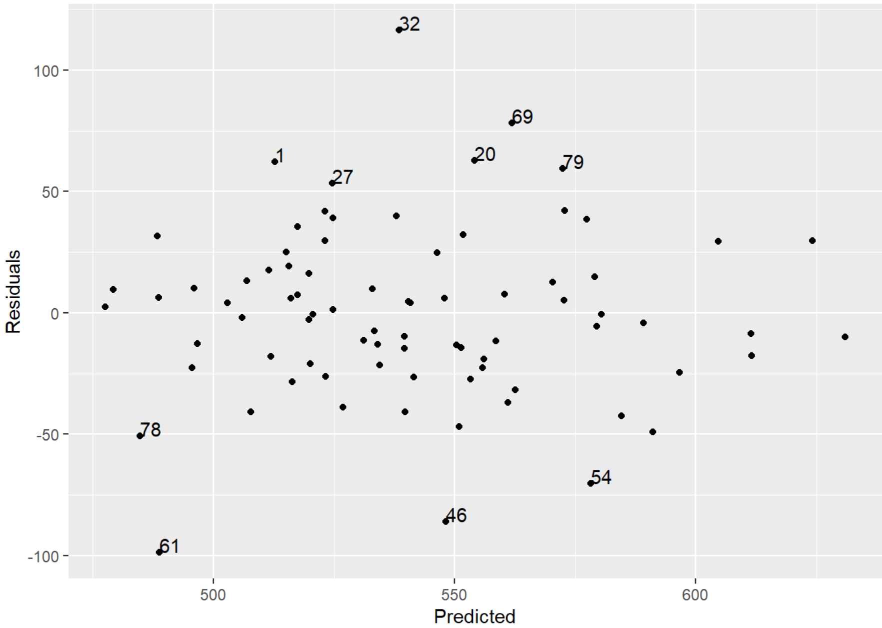
Остатки

```
ggplot(tdata, aes(sample = model4$residuals))+geom_qq()+geom_qq_line()
```



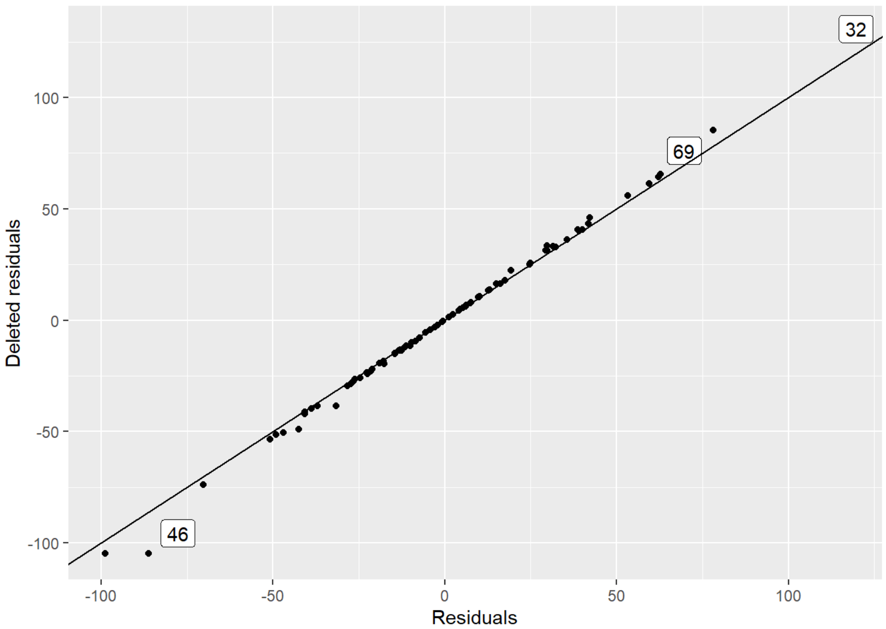
predicted-residuals:

```
ggplot(tdata ,aes(x = model4$fitted.values,y=model4$residuals))+
  geom_point()+
  geom_text(aes(label=ifelse(abs(model4$residuals)>50,as.character(reorder(1:dim(tdata)[1],model4$fitted.values)),'')),hjust
=0,vjust=0)+
  xlab("Predicted")+ylab("Residuals")
```



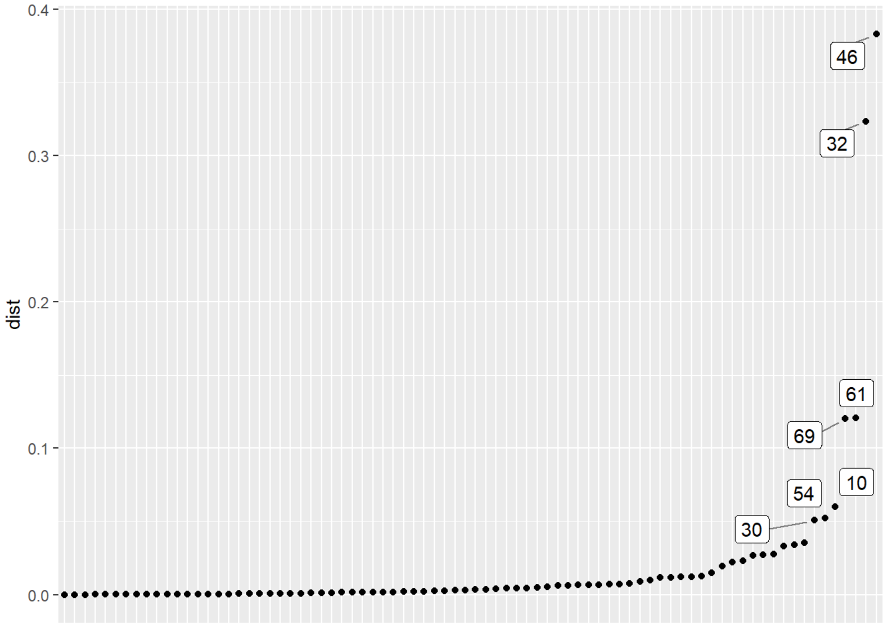
residuals-deleted residuals

```
del.resid<-tdata$AVRMATH
form2<-AVRMATH ~ logNTERM_D + SAL_ALL + APP_ACC_r
for(i in 1:dim(tdata)[1])
{
  del.resid[i]<-del.resid[i]-predict.lm(lm(form2, data = tdata[-i,]), tdata[i,])
}
ggplot(tdata, aes(x = model4$residuals, y = del.resid))+
  geom_point()+
  geom_label_repel(aes(label=ifelse(abs(model4$residuals - del.resid)>7,as.character(reorder(1:dim(tdata)[1], model4$residua
ls)),'')),
                 box.padding   = 0.35,
                 point.padding = 0.5,
                 segment.color = 'grey50')+
  xlab("Residuals")+ylab("Deleted residuals")+geom_abline(slope = 1, intercept = 0)
```



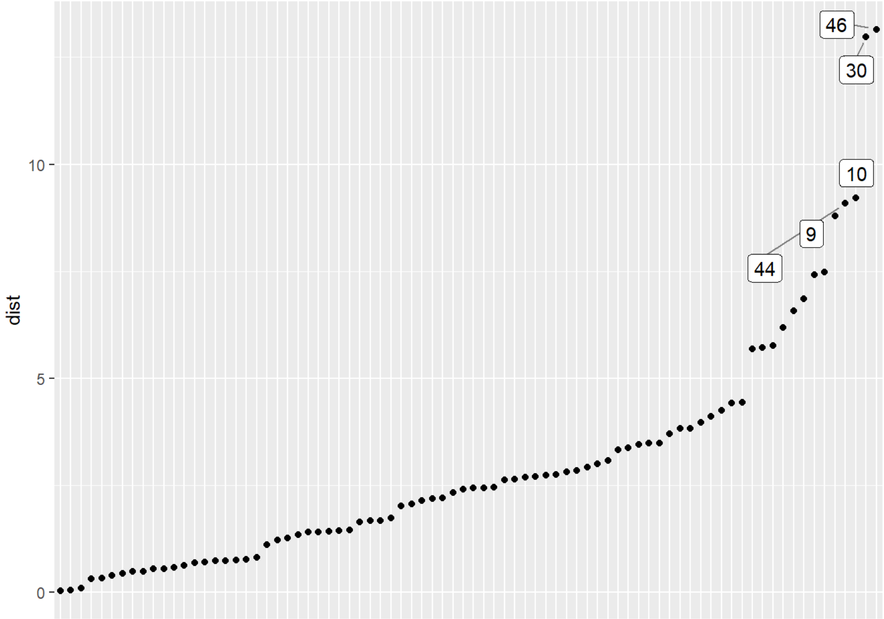
Расстояния Кука

```
gdata1<-data.frame('n' = 1:dim(tdata)[1], 'dist' = cooks.distance(model4))

ggplot(gdata1, aes(x = reorder(n, dist), y = dist))+geom_point()+geom_label_repel( aes(label=ifelse(dist >0.04,as.character(n),'')),
                  box.padding   = 0.35,
                  point.padding = 0.5,
                  segment.color = 'grey50')+
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```

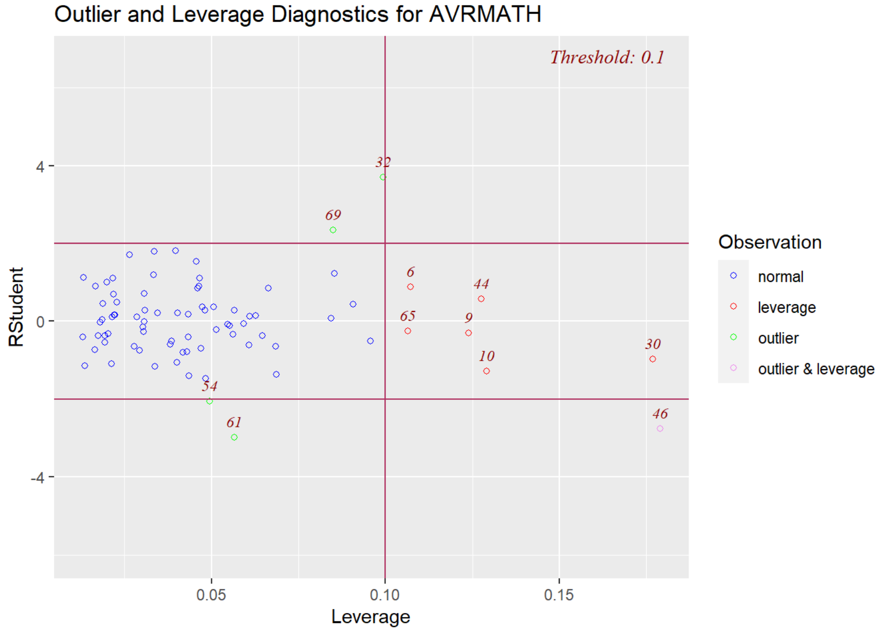


Расстояния Махаланобиса

```
feat.nums<-c(6,10,11)

gdata2<-data.frame('n' = 1:dim(tdata)[1], 'dist' = mahalanobis(tdata[feat.nums], apply(tdata[feat.nums], 2, mean) ,cov(tdata[feat.nums])))

ggplot(gdata2, aes(x = reorder(n, dist), y = dist))+geom_point()+geom_label_repel( aes(label=ifelse(dist >8,as.character(n),'')),
                  box.padding   = 0.35,
                  point.padding = 0.5,
                  segment.color = 'grey50')+
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```




```
ols_plot_resid_lev(model4)
```



Имеет смысл отбросить 30, 32, 46

```
model5<-lm(AVRMATH ~ logNTERM_D + SAL_ALL + APP_ACC_r, data = tdata[-c(30,32,46),])
summary.beta(model5)
```

```
##
## Call:
## lm(formula = AVRMATH ~ logNTERM_D + SAL_ALL + APP_ACC_r, data = tdata[-c(30,
##     32, 46), ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -105.995  -17.051    0.281   19.125   73.601
##
## Coefficients:
##              Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept)  574.69884      0.00000   52.90799  10.862  < 2e-16 ***
## logNTERM_D   -22.40657     -0.34294    5.61241  -3.992 0.000154 ***
## SAL_ALL        0.21184      0.28572    0.06847   3.094 0.002799 **
## APP_ACC_r   -128.79616     -0.34367   32.69981  -3.939 0.000186 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30.89 on 73 degrees of freedom
## Multiple R-squared:  0.595,  Adjusted R-squared:  0.5783
## F-statistic: 35.75 on 3 and 73 DF,  p-value: 2.524e-14
```

Что-нибудь спрогнозируем. Например, для Государственного университета штата Иллинойс. По данным на 2000-2002 годы 99% поступивших сдавали экзамен ACT, а не SAT;

C9. Percent and number of first-time, first-year (freshman) students enrolled in fall 2001 who submitted national standardized (SAT/ACT) test scores. Include information for **ALL enrolled, degree-seeking, first-time, first-year (freshman) students who submitted test scores**. Do not include partial test scores (e.g., mathematics scores but not verbal for a category of students) or combine other standardized test results (such as TOEFL) in this item. SAT scores should be recentered scores. The 25th percentile is the score that 25 percent scored at or below; the 75th percentile score is the one that 25 percent scored at or above.

Percent submitting SAT scores _____ Number submitting SAT scores _____
Percent submitting ACT scores __99%___ Number submitting ACT scores __3,307___

	25th Percentile	75th Percentile
SAT I Verbal		
SAT I Math		
ACT Composite	20	25
ACT English	20	25
ACT Math	19	25

Percent of first-time, first-year (freshman) students with scores in each range:

	SAT I Verbal	SAT I Math
700-800		
600-699		
500-599		
400-499		
300-399		
200-299		

	ACT Composite	ACT English	ACT Math
30-36	3	4	4
24-29	37	35	36
18-23	59	52	50
12-17	1	9	10
6-11	--	0	--
Below 6	--	--	--

Why Do Some States Require the ACT?

In 2001, when states were first implementing statewide assessment programs, Illinois and Colorado decided that, rather than creating their own tests for high school juniors, they would contract with ACT, Inc. to use the ACT as a statewide assessment . (The ACT is generally considered more content based than the SAT, and therefore a better for assessments.)

This plan had the added advantages of **providing every student with the chance to take a college admissions test** and, ideally, **encouraging students who might not have otherwise considered college to apply**.

```
predict.lm(model5, testdata[9,])
```

```
##        9
## 510.6915
```

Тем не менее результаты можно сопоставить

ACT MATH SCORE	SAT MATH SCORE (Before March 2016)	SAT MATH SCORE (After March 2016)
30	680	710
29	660	690
28	640	660
27	620	640
26	600	620
25	580	600
24	560	580
23	540	570
22	520	550
21	500	530
20	480	510