lab6

Anastasia Khudoyarova 11/30/2020

Load

```
data_f <- as.data.frame(read.table('zeta.csv', header = TRUE, sep=','))
data_f <- data_f[data_f$sex == "F",]
data_f <- subset.data.frame(data_f, select =
c("meanage","meaneducation","meanemployment","meanhouseholdincome"))
head(data_f)
```

```
##     meanage meaneducation meanemployment meanhouseholdincome
## 1  37.40335      10.91282      0.7400294            18533.84
## 3  31.80943      13.91337      1.0858555            40784.49
## 5  35.99079      10.09777      0.6287526            17496.53
## 7  37.26014      10.96916      0.8543247            19416.41
## 9  40.42732      11.57577      0.7815393            21607.34
## 11 38.24761      10.99235      0.7437151            17243.75
```

Filter as:

`8 < meaneducation < 18` `10,000 < meanhouseholdincome < 200,000` `0 < meanemployment < 3` `20 < meanage < 60`

```
c_df <-subset(data_f,
           8 < data_f$meaneducation &
           data_f$meaneducation < 18 &
           10000 < data_f$meanhouseholdincome &
           data_f$meanhouseholdincome < 200000 &
           0 < data_f$meanemployment &
           data_f$meanemployment < 3 &
           20 < data_f$meanage &
           data_f$meanage < 60)
head(c_df)
```

```
##     meanage meaneducation meanemployment meanhouseholdincome
## 1  37.40335      10.91282      0.7400294            18533.84
## 3  31.80943      13.91337      1.0858555            40784.49
## 5  35.99079      10.09777      0.6287526            17496.53
## 7  37.26014      10.96916      0.8543247            19416.41
```

```
## 9  40.42732      11.57577      0.7815393           21607.34
## 11 38.24761      10.99235      0.7437151           17243.75
```

Create a variable called log_income = log10(meanhouseholdincome)

```
c_df <- cbind(c_df, log10(c_df$meanhouseholdincome))
head(c_df, n=10)
```

```
##     meanage meaneducation meanemployment meanhouseholdincome
## 1  37.40335      10.91282      0.7400294           18533.84
## 3  31.80943      13.91337      1.0858555           40784.49
## 5  35.99079      10.09777      0.6287526           17496.53
## 7  37.26014      10.96916      0.8543247           19416.41
## 9  40.42732      11.57577      0.7815393           21607.34
## 11 38.24761      10.99235      0.7437151           17243.75
## 13 42.59515      11.15270      0.8582248           23200.96
## 15 34.21463      11.20447      0.6917640           18032.09
## 17 35.62713      10.85684      0.6221983           17908.28
## 19 37.25719      11.85254      0.8958583           27246.44
##    log10(c_df$meanhouseholdincome)
## 1                         4.267966
## 3                         4.610495
## 5                         4.242952
## 7                         4.288169
## 9                         4.334601
## 11                        4.236632
## 13                        4.365506
## 15                        4.256046
## 17                        4.253054
## 19                        4.435310
```

Rename the columns meanage, meaneducation, and meanemployment as age, education, and employment, respectively

```
colnames(c_df) <- c("age", "education", "employment", "income",
"log_income")
head(c_df)
```

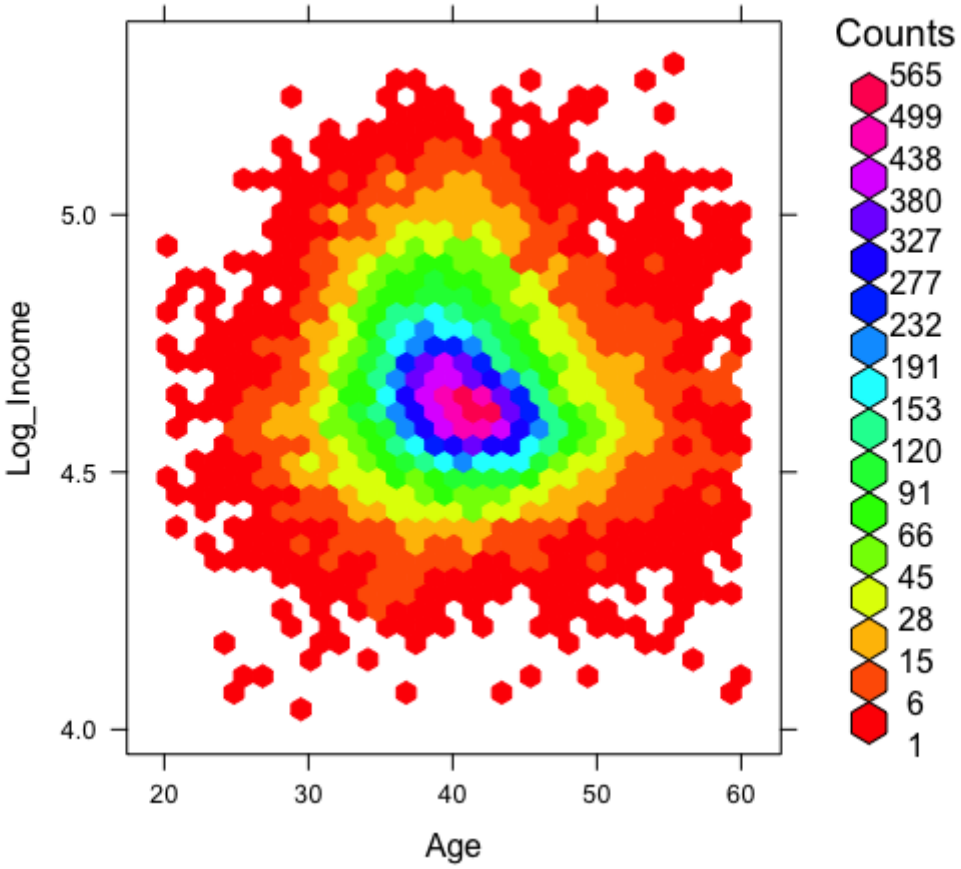
```
##         age education employment   income log_income
## 1  37.40335  10.91282  0.7400294 18533.84   4.267966
## 3  31.80943  13.91337  1.0858555 40784.49   4.610495
## 5  35.99079  10.09777  0.6287526 17496.53   4.242952
## 7  37.26014  10.96916  0.8543247 19416.41   4.288169
```

```
## 9  40.42732  11.57577  0.7815393 21607.34   4.334601
## 11 38.24761  10.99235  0.7437151 17243.75   4.236632
```

Linear Regression Analysis

a. Scatter plot that shows the effect age on log_income. There is almost no relationship.

```
library(hexbin)
hexbinplot(c_df$log_income ~ c_df$age, trans = sqrt, inv = function(x)
x^2, xlab="Age", ylab="Log_Income", colramp=rainbow)
```



b. Linear regression model between log_income and age.

t-value определяет величину разницы между вариациями в наборе данных. Или иначе просто считает разницу в величине standard error. Чем выше значение, тем более вероятно, что нет null hypothesis.

```
lin_reg1 <- lm(formula=log_income ~ age, data=c_df)
summary(lin_reg1)
```

```
##
## Call:
## lm(formula = log_income ~ age, data = c_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.65733 -0.08296 -0.01620  0.07178  0.67202
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.7877484  0.0064657   740.5   <2e-16 ***
## age         -0.0030739  0.0001584   -19.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1366 on 31427 degrees of freedom
## Multiple R-squared:  0.01184,    Adjusted R-squared:  0.01181
## F-statistic: 376.5 on 1 and 31427 DF,  p-value: < 2.2e-16
```

c. R-squared измеряет силу связи между моделью и зависимым значением по шкале от 0 до 100%

Чем больше это значение, тем лучше модель подходит для исследования.

d. F-statistic показывает можем ли мы отвергнуть null hypothesis. То есть случай когда все параметры нулевые

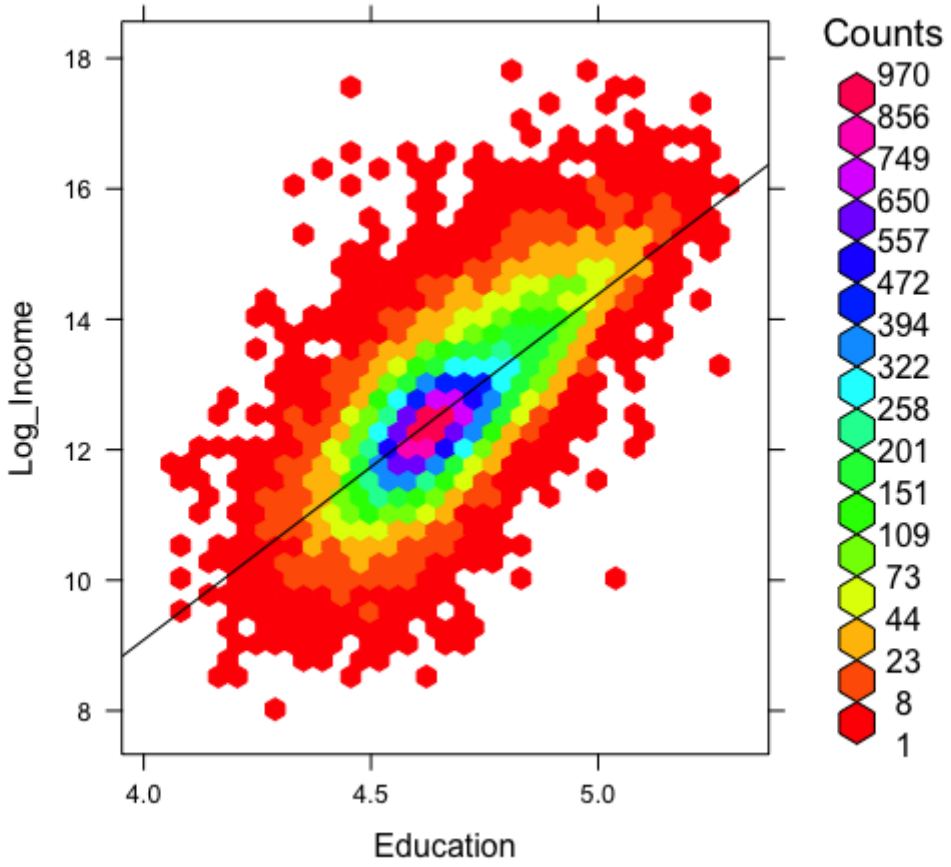
Чтобы получить хорошую модель F-statistic должна быть больше 1, а p-value очень маленьким.

e. Multiple R-squared: 0.01184, Adjusted R-squared: 0.01181

R-value около 0. Модель не очень хорошая.

f. Scatter plot that shows the effect education has on log_income.

```
hexbinplot(c_df$education ~ c_df$log_income, xlab="Education",
ylab="Log_Income", trans = sqrt, inv = function(x) x^2, type = c("q","r"),
colramp=rainbow)
```



g. Summary of a linear regression model between log_income and education.

Multiple R-squared: 0.5354, Adjusted R-squared: 0.5354

В этом случае у нас R-squared ближе к 1 => лучше, чем прошлая модель

```
lin_reg2 <- lm(formula=log_income ~ education, data=c_df)
summary(lin_reg2)
```

```
##
## Call:
## lm(formula = log_income ~ education, data = c_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.72721 -0.05349  0.00029  0.05796  0.64512
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.3896705  0.0067123   505.0   <2e-16 ***
## education   0.1010797  0.0005311   190.3   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.09369 on 31427 degrees of freedom
## Multiple R-squared:  0.5354, Adjusted R-squared:  0.5354
## F-statistic: 3.622e+04 on 1 and 31427 DF,  p-value: < 2.2e-16
```

h. Summary of a linear regression model between the dependent variable log_income, and the independent variables age, education, and employment

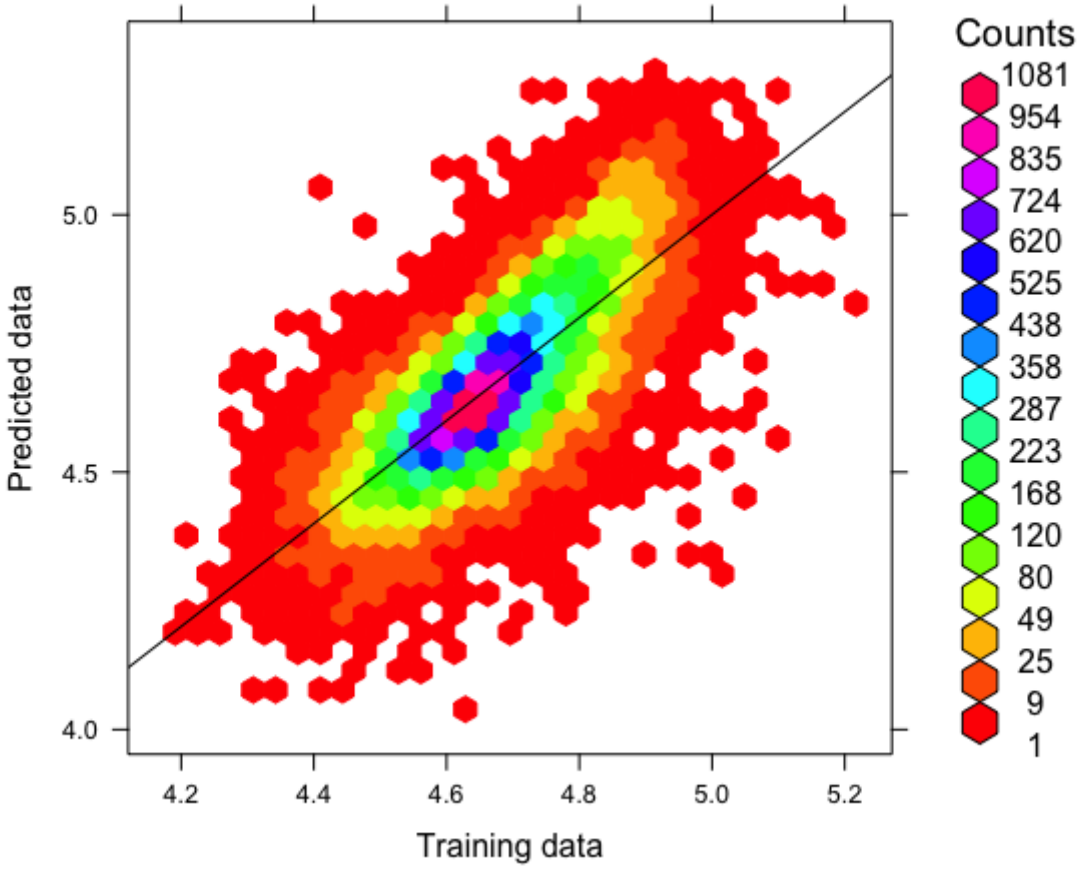
```
lin_reg3 <- lm(formula=log_income ~ age + education + employment,
data=c_df)
summary(lin_reg3)
```

```
##
## Call:
## lm(formula = log_income ~ age + education + employment, data = c_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.70315 -0.05023  0.00066  0.05213  0.64021
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.5123331  0.0076320  460.21   <2e-16 ***
## age         -0.0026030  0.0001109  -23.48   <2e-16 ***
## education    0.0912653  0.0005980  152.61   <2e-16 ***
## employment   0.0663722  0.0019559   33.94   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09017 on 31425 degrees of freedom
## Multiple R-squared:  0.5697, Adjusted R-squared:  0.5697
## F-statistic: 1.387e+04 on 3 and 31425 DF,  p-value: < 2.2e-16
```

i. С каждым шагом образования доход растет на 9 %

j. Graph that contains a y = x line and uses the multiple regression model to plot the predicted data points against the actual data points of the training set.

```
c_df["predict"] <- predict(lin_reg3, c_df)
hexbinplot(c_df$log_income ~ c_df$predict, ylab="Predicted data",
xlab="Training data", trans = sqrt, inv = function(x) x^2, type =
c("q","r"), colramp=rainbow)
```



k. Как видно на графике, линия y=x почти повторяет разброс значений. Так же ярко видна область, где точки как бы облепливают прямую y=x

Это означает, что модель действительно хорошая и предсказание совпадает с реальностью