

Linear Regression

Contents

1	Getting Set Up	1
1.1	Setting chunk options and generating R script	1
1.2	Importing Libraries	1
2	Basics	2
3	Simple Linear Regression	4
4	Multiple Linear Regression	5
4.1	Reading in the data	5
4.2	Exploring the data	6
4.3	Fitting the model	8
4.4	Likelihood ratio test null model versus full model	8
4.5	Sequential building of the model	8
4.6	Final Model	9
4.7	Predicting a new observation	9

1 Getting Set Up

1.1 Setting chunk options and generating R script

```
knitr::opts_chunk$set(warning=FALSE, message=FALSE)
knitr::purl("linear-regression.Rmd")
```

```
##
##
## processing file: linear-regression.Rmd
```

```
## output file: linear-regression.R
```

1.2 Importing Libraries

```
install.packages('pastecs')
```

2 Basics

Reading the data in R.

```
library(pastecs)
kalama = read.table("kalama.txt", header=T)
attach(kalama)
kalama
```

```
##      age height
## 1    18   76.1
## 2    19   77.0
## 3    20   78.1
## 4    21   78.2
## 5    22   78.8
## 6    23   79.7
## 7    24   79.9
## 8    25   81.1
## 9    26   81.2
## 10   27   81.8
## 11   28   82.8
## 12   29   83.5
```

Descriptive statistics in R.

```
options(digits=2)
descrip.kalama = stat.desc(kalama[,c("age", "height")], basic=TRUE, desc=TRUE)
descrip.kalama
```

```
##              age  height
## nbr.val      12.00 12.000
## nbr.null      0.00  0.000
## nbr.na        0.00  0.000
## min          18.00 76.100
## max          29.00 83.500
## range         11.00  7.400
## sum          282.00 958.200
## median        23.50 79.800
## mean          23.50 79.850
## SE.mean        1.04  0.665
## CI.mean.0.95   2.29  1.463
## var           13.00  5.301
## std.dev        3.61  2.302
## coef.var       0.15  0.029
```

Estimating Correlations in R.

```
cov.age.height = cov(age, height)
corr.age.height = cor(age, height)
cov.age.height
```

```
## [1] 8.3
```

```
corr.age.height
```

```
## [1] 0.99
```

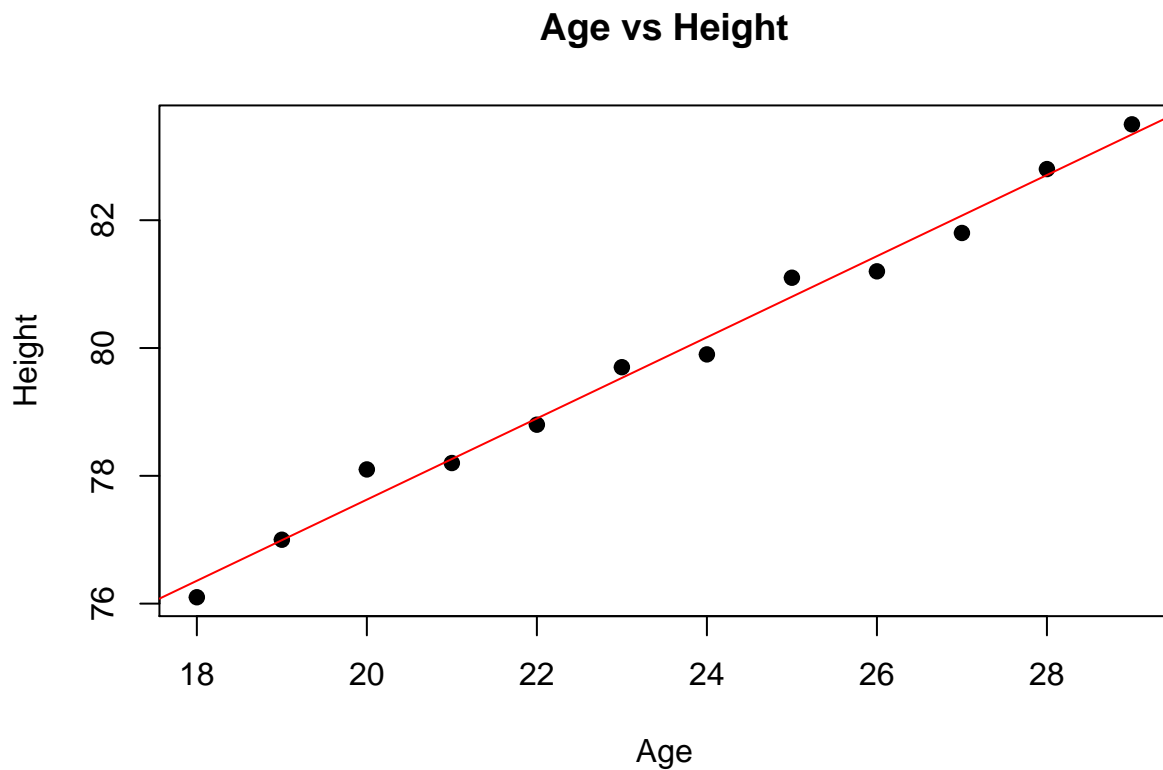
Testing if the population correlation is zero.

```
corr.age.height.test = cor.test(age, height, alternative="two.sided", method="pearson")
corr.age.height.test
```

```
##
## Pearson's product-moment correlation
##
## data: age and height
## t = 30, df = 10, p-value = 4e-11
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.98 1.00
## sample estimates:
## cor
## 0.99
```

Scatterplot with line.

```
plot(age, height, main="Age vs Height", xlab="Age", ylab="Height", pch=19)
abline(lm(height~age), col="red")
```



3 Simple Linear Regression

```
res = lm(height~age, data=kalama)
kalama.anova = anova(res)
kalama.summary = summary(res)
kalama.anova
```

```
## Analysis of Variance Table
##
## Response: height
##           Df Sum Sq Mean Sq F value    Pr(>F)
## age         1   57.7    57.7    880 4.4e-11 ***
## Residuals  10    0.7     0.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
kalama.summary
```

```
##
## Call:
## lm(formula = height ~ age, data = kalama)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.2724 -0.2425 -0.0276  0.1601  0.4724
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  64.9283     0.5084   127.7 < 2e-16 ***
## age          0.6350     0.0214    29.7 4.4e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.26 on 10 degrees of freedom
## Multiple R-squared:  0.989, Adjusted R-squared:  0.988
## F-statistic: 880 on 1 and 10 DF, p-value: 4.43e-11
```

4 Multiple Linear Regression

4.1 Reading in the data

```
satisfaction = read.table("satisfaction.txt", header=T)
attach(satisfaction)
satisfaction
```

```
##      satis age severity anxiety
## 1      48  50        51      2.3
## 2      57  36        46      2.3
## 3      66  40        48      2.2
## 4      70  41        44      1.8
## 5      89  28        43      1.8
## 6      36  49        54      2.9
## 7      46  42        50      2.2
## 8      54  45        48      2.4
## 9      26  52        62      2.9
## 10     77  29        50      2.1
## 11     89  29        48      2.4
## 12     67  43        53      2.4
## 13     47  38        55      2.2
## 14     51  34        51      2.3
## 15     57  53        54      2.2
## 16     66  36        49      2.0
## 17     79  33        56      2.5
## 18     88  29        46      1.9
## 19     60  33        49      2.1
## 20     49  55        51      2.4
## 21     77  29        52      2.3
## 22     52  44        58      2.9
## 23     60  43        50      2.3
## 24     86  23        41      1.8
## 25     43  47        53      2.5
## 26     34  55        54      2.5
## 27     63  25        49      2.0
```

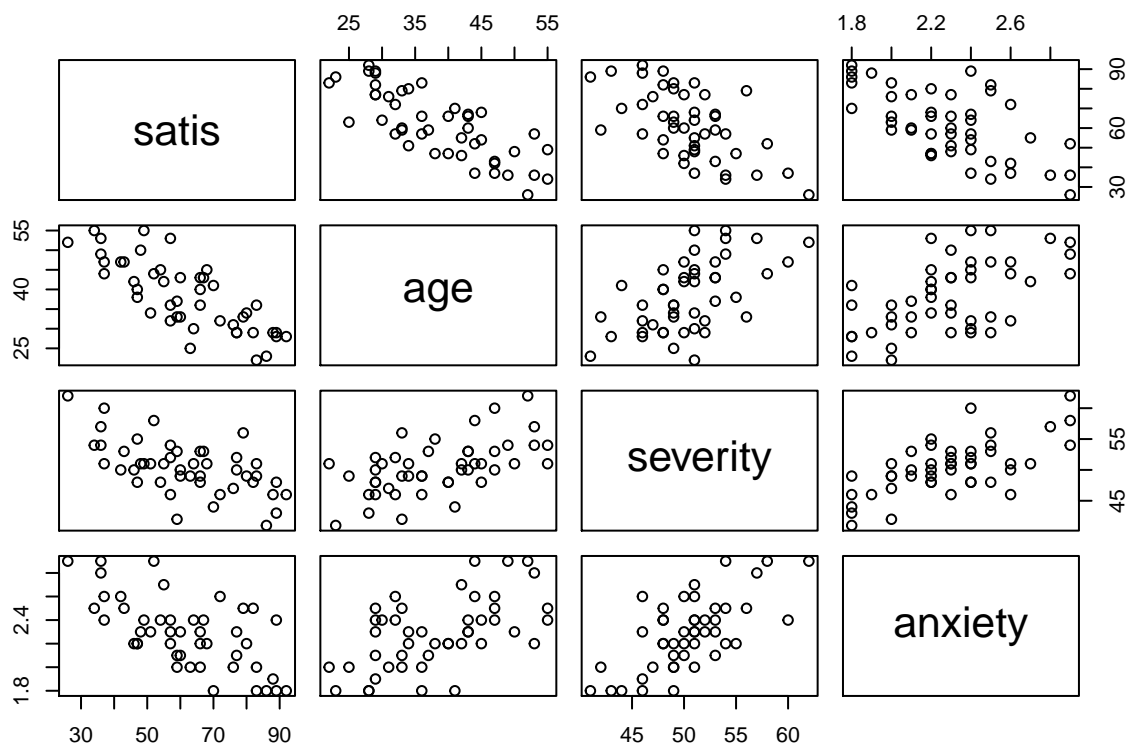
```
## 28    72  32    46    2.6
## 29    57  32    52    2.4
## 30    55  42    51    2.7
## 31    59  33    42    2.0
## 32    83  36    49    1.8
## 33    76  31    47    2.0
## 34    47  40    48    2.2
## 35    36  53    57    2.8
## 36    80  34    49    2.2
## 37    82  29    48    2.5
## 38    64  30    51    2.4
## 39    37  47    60    2.4
## 40    42  47    50    2.6
## 41    66  43    53    2.3
## 42    83  22    51    2.0
## 43    37  44    51    2.6
## 44    68  45    51    2.2
## 45    59  37    53    2.1
## 46    92  28    46    1.8
```

4.2 Exploring the data

```
cor(satisfaction)
```

```
##          satis    age severity anxiety
## satis      1.00 -0.79   -0.60  -0.64
## age       -0.79  1.00    0.57   0.57
## severity  -0.60  0.57    1.00   0.67
## anxiety   -0.64  0.57    0.67   1.00
```

```
plot(satisfaction)
```



Descriptive statistics

```
options(digits=2)
descrip.satisfaction = stat.desc(satisfaction,basic=TRUE, desc=TRUE)
descrip.satisfaction
```

```
##          satis      age severity anxiety
## nbr.val    46.00    46.00  4.6e+01  46.000
## nbr.null     0.00     0.00  0.0e+00   0.000
## nbr.na       0.00     0.00  0.0e+00   0.000
## min         26.00    22.00  4.1e+01   1.800
## max         92.00    55.00  6.2e+01   2.900
## range       66.00    33.00  2.1e+01   1.100
## sum        2832.00  1766.00  2.3e+03  105.200
## median      60.00    37.50  5.0e+01   2.300
## mean        61.57    38.39  5.0e+01   2.287
## SE.mean      2.54     1.31  6.4e-01   0.044
## CI.mean.0.95  5.12     2.65  1.3e+00   0.089
## var         297.10    79.53  1.9e+01   0.090
## std.dev      17.24     8.92  4.3e+00   0.299
## coef.var      0.28     0.23  8.6e-02   0.131
```

4.3 Fitting the model

```
satisfaction.lm = lm(satis~age+severity+anxiety, data=satisfaction)
satisfaction.summary = summary(satisfaction.lm)
satisfaction.summary

##
## Call:
## lm(formula = satis ~ age + severity + anxiety, data = satisfaction)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.35  -6.42   0.52   8.37  17.16
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  158.491     18.126   8.74  5.3e-11 ***
## age          -1.142      0.215  -5.31  3.8e-06 ***
## severity     -0.442      0.492  -0.90   0.374
## anxiety     -13.470      7.100  -1.90   0.065 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10 on 42 degrees of freedom
## Multiple R-squared:  0.682, Adjusted R-squared:  0.659
## F-statistic: 30.1 on 3 and 42 DF, p-value: 1.54e-10
```

4.4 Likelihood ratio test null model versus full model

```
satisfaction.lm.int = lm(satis~1, data=satisfaction) # Null model
anova(satisfaction.lm.int,satisfaction.lm) # Null versus full

## Analysis of Variance Table
##
## Model 1: satis ~ 1
## Model 2: satis ~ age + severity + anxiety
##   Res.Df  RSS Df Sum of Sq    F Pr(>F)
## 1      45 13369
## 2      42  4249   3     9120 30.1 1.5e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4.5 Sequential building of the model

```
satisfaction.anova = anova(satisfaction.lm)
satisfaction.anova
```



```
## Analysis of Variance Table
##
## Response: satis
##           Df Sum Sq Mean Sq F value    Pr(>F)
## age         1   8275     8275   81.80 2.1e-11 ***
## severity    1    481      481    4.75  0.035 *
## anxiety     1    364      364    3.60  0.065 .
## Residuals  42   4249      101
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4.6 Final Model

```
satisfaction.lm.final = lm(satis~age+anxiety, data=satisfaction)
satisfaction.final.summary = summary(satisfaction.lm.final)
satisfaction.final.summary
```

```
##
## Call:
## lm(formula = satis ~ age + anxiety, data = satisfaction)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.445  -7.328   0.673   8.513  18.053
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   145.941     11.525   12.66 4.2e-16 ***
## age           -1.200       0.204   -5.88 5.4e-07 ***
## anxiety       -16.742       6.081   -2.75 0.0086 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10 on 43 degrees of freedom
## Multiple R-squared:  0.676, Adjusted R-squared:  0.661
## F-statistic: 44.9 on 2 and 43 DF, p-value: 2.98e-11
```

4.7 Predicting a new observation

```
newdata = data.frame(age=43, anxiety=2.7)
pred.w.plim = predict(satisfaction.lm.final, newdata, interval="predict")
pred.w.clim = predict(satisfaction.lm.final, newdata, interval = "confidence")
pred.w.plim
```

```
## fit lwr upr
## 1 49 28 70
```

```
pred.w.clim
```

```
##   fit lwr upr  
## 1  49  44  54
```