# Lesson 02

Christopher A. Swenson (chris@cswenson.com)

11/27/2021

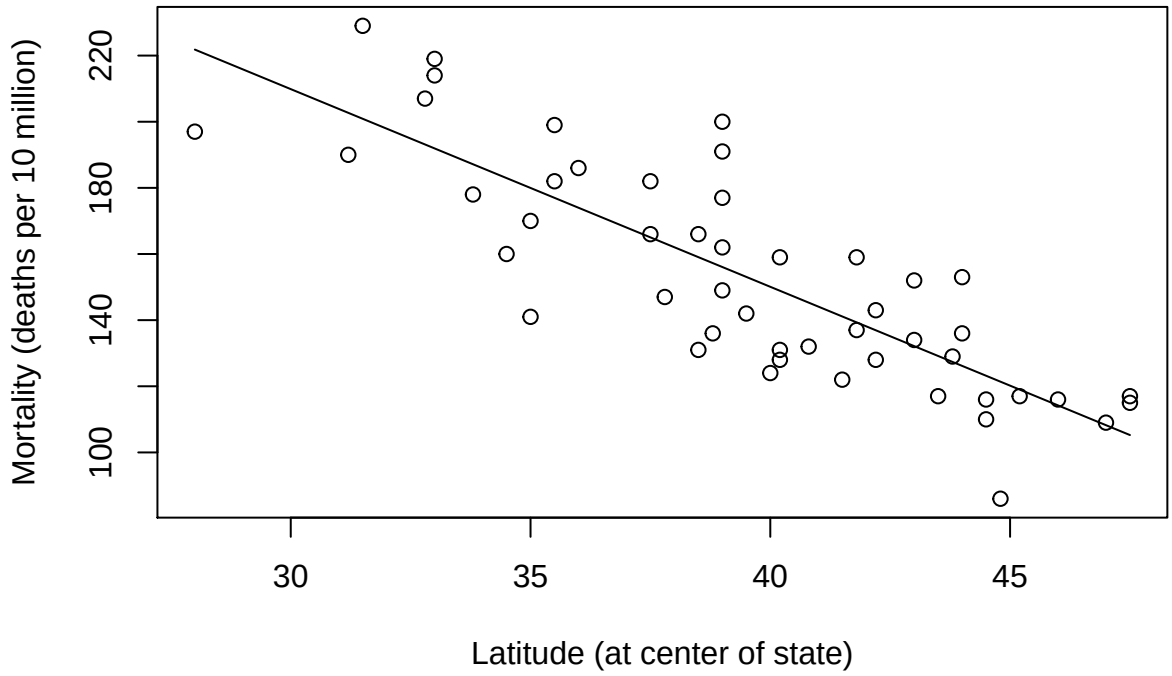
## Skin cancer

Load the skin cancer data. Fit a simple linear regression model with y = Mort and x = Lat. Display a scatterplot of the data with the simple linear regression line. Display model results. Calculate confidence intervals for the model parameters (regression coefficients).

```
skincancer <- read.table("./Data/skincancer.txt", header=T)
attach(skincancer)

model <- lm(Mort ~ Lat)

plot(x=Lat, y=Mort,
     xlab="Latitude (at center of state)", ylab="Mortality (deaths per 10 million)",
     panel.last = lines(sort(Lat), fitted(model)[order(Lat)]))
```




```
summary(model)
```

```
##
## Call:
## lm(formula = Mort ~ Lat)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -38.972 -13.185   0.972  12.006  43.938
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 389.1894    23.8123   16.34  < 2e-16 ***
## Lat          -5.9776     0.5984   -9.99 3.31e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.12 on 47 degrees of freedom
## Multiple R-squared:  0.6798, Adjusted R-squared:  0.673
## F-statistic:  99.8 on 1 and 47 DF,  p-value: 3.309e-13
```

```
# Coefficients:
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept) 389.1894    23.8123   16.34  < 2e-16 ***
# Lat          -5.9776     0.5984   -9.99 3.31e-13 ***
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
# Residual standard error: 19.12 on 47 degrees of freedom
# Multiple R-squared:  0.6798,  Adjusted R-squared:  0.673
# F-statistic:  99.8 on 1 and 47 DF,  p-value: 3.309e-13

confint(model, level=0.95)
```

```
##                  2.5 %     97.5 %
## (Intercept) 341.285151 437.093552
## Lat          -7.181404  -4.773867
```

```
#                  2.5 %     97.5 %
# (Intercept) 341.285151 437.093552
# Lat          -7.181404  -4.773867

detach(skincancer)
```

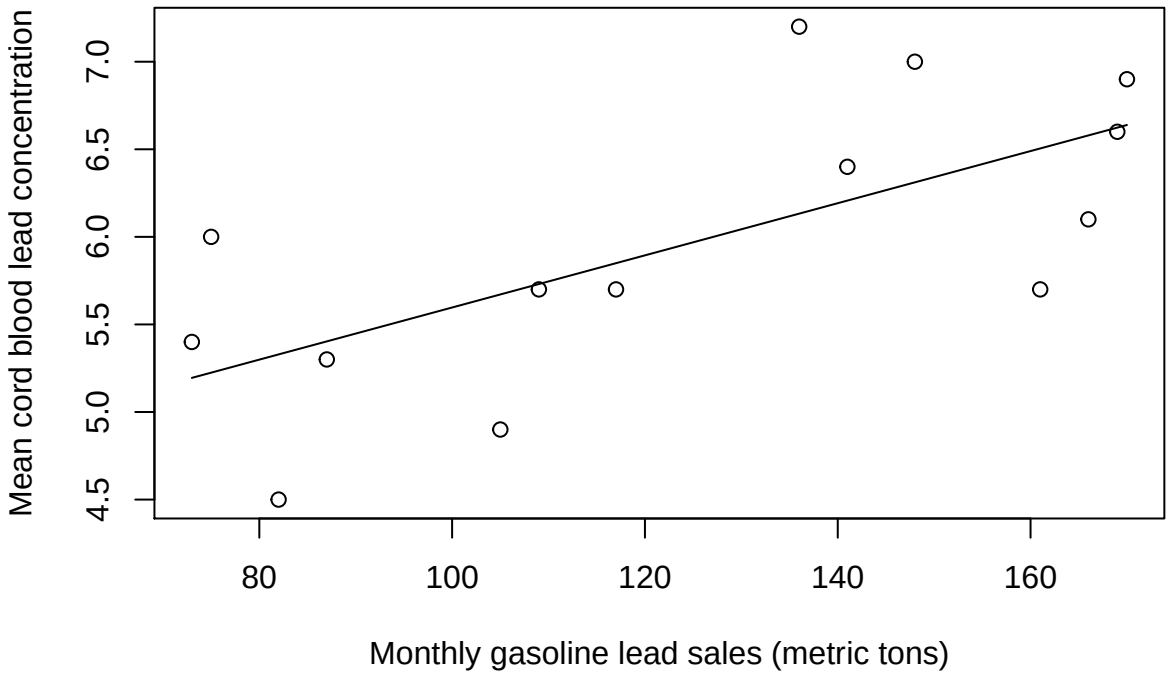
## Cord blood lead concentration

Load the cord blood lead concentration data. Fit a simple linear regression model with y = Cord and x = Sold. Display a scatterplot of the data with the simple linear regression line. Display model results. Calculate confidence intervals for the model parameters (regression coefficients).

```
cordblood <- read.table("./Data/leadcord.txt", header=T)
attach(cordblood)

model <- lm(Cord ~ Sold)

plot(x=Sold, y=Cord,
     xlab="Monthly gasoline lead sales (metric tons)",
     ylab="Mean cord blood lead concentration",
     panel.last = lines(sort(Sold), fitted(model)[order(Sold)]))
```




```
summary(model)
```

```
##
## Call:
## lm(formula = Cord ~ Sold)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82877 -0.39679 -0.02723  0.24729  1.06742
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.108182   0.608806   6.748 2.05e-05 ***
## Sold        0.014885   0.004719   3.155   0.0083 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6162 on 12 degrees of freedom
## Multiple R-squared:  0.4533, Adjusted R-squared:  0.4078
## F-statistic: 9.952 on 1 and 12 DF,  p-value: 0.008303
```

```
# Coefficients:
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept) 4.108182   0.608806   6.748 2.05e-05 ***
# Sold        0.014885   0.004719   3.155   0.0083 **
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
#
# Residual standard error: 0.6162 on 12 degrees of freedom
# Multiple R-squared:  0.4533,  Adjusted R-squared:  0.4078
# F-statistic: 9.952 on 1 and 12 DF,  p-value: 0.008303

confint(model, level=0.95)
```

```
##                   2.5 %     97.5 %
## (Intercept) 2.781707607 5.43465712
## Sold        0.004604418 0.02516608
```

```
#                   2.5 %     97.5 %
# (Intercept) 2.781707607 5.43465712
# Sold        0.004604418 0.02516608

detach(cordblood)
```

### Skin cancer

Load the skin cancer data. Fit a simple linear regression model with y = Mort and x = Lat. Display analysis of variance table.

```
skincancer <- read.table("./Data/skincancer.txt", header=T)
attach(skincancer)

model <- lm(Mort ~ Lat)

anova(model)
```

```
## Analysis of Variance Table
##
## Response: Mort
##           Df Sum Sq Mean Sq F value    Pr(>F)
## Lat        1  36464   36464  99.797 3.309e-13 ***
## Residuals 47  17173     365
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Analysis of Variance Table
# Response: Mort
#           Df Sum Sq Mean Sq F value    Pr(>F)
# Lat        1  36464   36464  99.797 3.309e-13 ***
# Residuals 47  17173     365
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
# Note: R anova function does not display the total sum of squares.
# Add regression and residual sums of squares to get total sum of squares.
# SSR + SSE = SSTO, i.e., 36464 + 17173 = 53637.

detach(skincancer)
```

### Height and grade point average

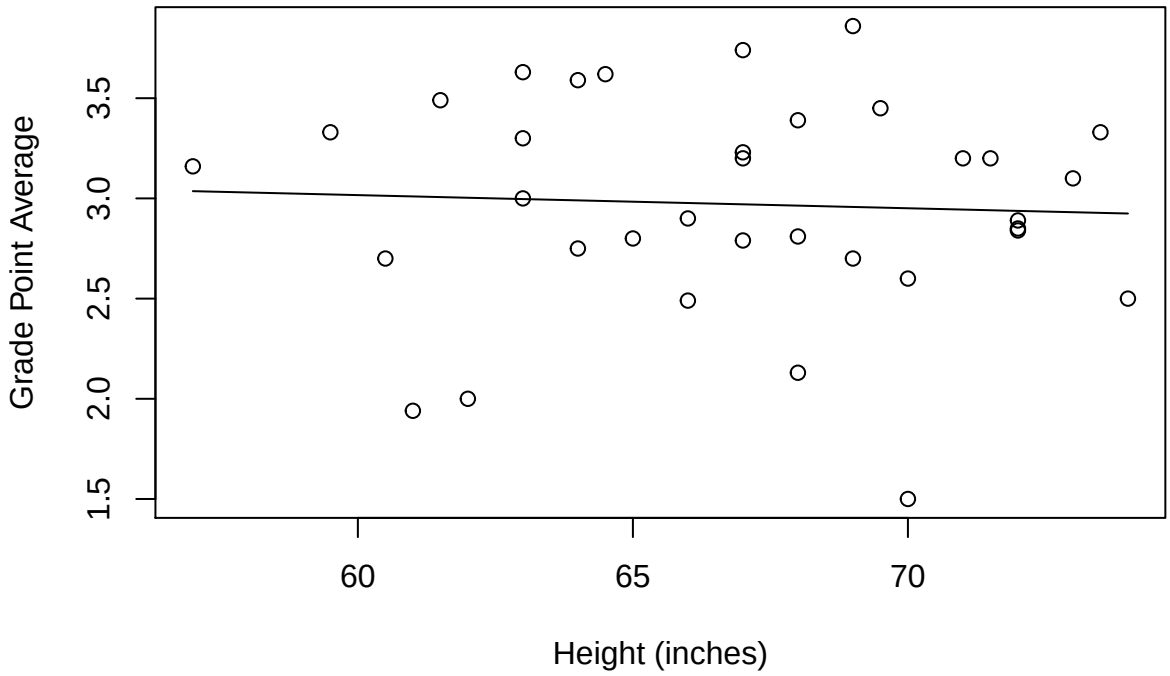
Load the height and grade point average data. Fit a simple linear regression model with y = gpa and x = height. Display a scatterplot of the data with the simple linear regression line. Display model results. Display analysis of variance table.

```
heightgpa <- read.table("./Data/heightgpa.txt", header=T)
attach(heightgpa)

model <- lm(gpa ~ height)

plot(x=height, y=gpa,
```

```
     xlab="Height (inches)", ylab="Grade Point Average",
     panel.last = lines(sort(height), fitted(model)[order(height)]))
```




```
summary(model)
```

```
##
## Call:
## lm(formula = gpa ~ height)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.45081 -0.24878  0.00325  0.35622  0.90263
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.410214   1.434616   2.377   0.0234 *
## height      -0.006563   0.021428  -0.306   0.7613
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5423 on 33 degrees of freedom
## Multiple R-squared:  0.002835,   Adjusted R-squared:  -0.02738
## F-statistic: 0.09381 on 1 and 33 DF,  p-value: 0.7613
```

```
# Coefficients:
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)  3.410214   1.434616   2.377   0.0234 *
# height      -0.006563   0.021428  -0.306   0.7613
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
#
# Residual standard error: 0.5423 on 33 degrees of freedom
# Multiple R-squared:  0.002835,  Adjusted R-squared:  -0.02738
```

```
# F-statistic: 0.09381 on 1 and 33 DF,  p-value: 0.7613

anova(model)
```

```
## Analysis of Variance Table
##
## Response: gpa
##           Df Sum Sq Mean Sq F value Pr(>F)
## height     1 0.0276 0.02759  0.0938 0.7613
## Residuals 33 9.7055 0.29411
```

```
# Analysis of Variance Table
# Response: gpa
#           Df Sum Sq Mean Sq F value Pr(>F)
# height     1 0.0276 0.02759  0.0938 0.7613
# Residuals 33 9.7055 0.29411
# SSTO = SSR + SSE = 0.0276 + 9.7055 = 9.7331.

detach(heightgpa)
```

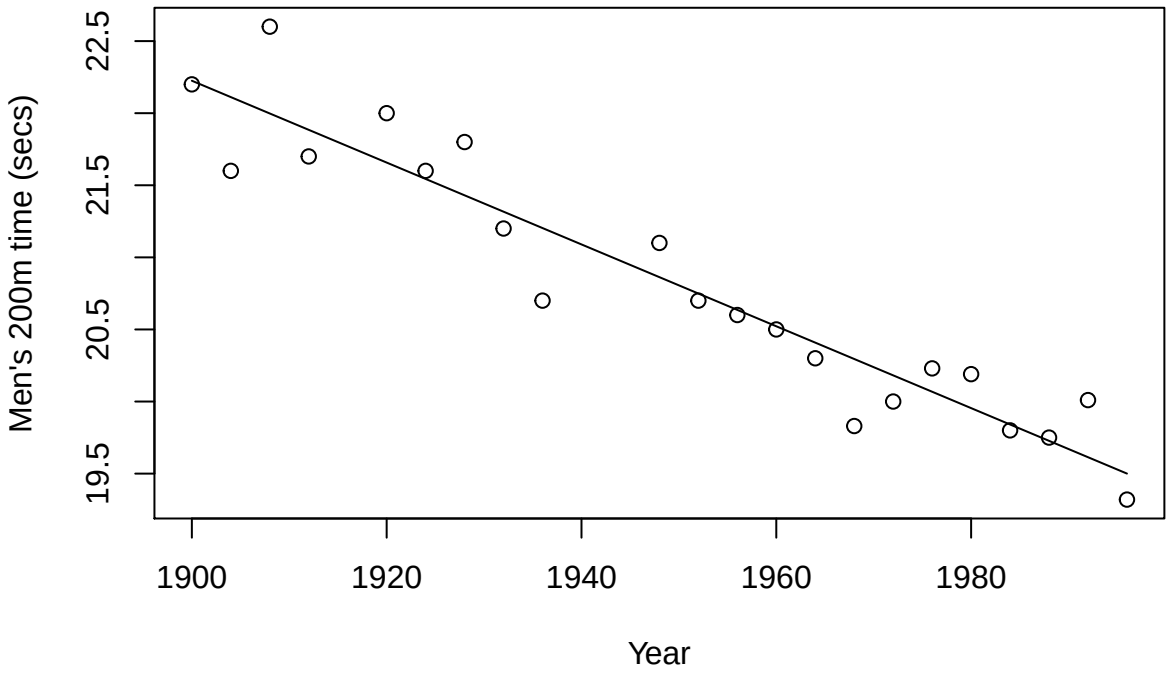
## Sprinters

Load the sprinters data. Fit a simple linear regression model with y = Men200m and x = Year. Display a scatterplot of the data with the simple linear regression line. Display model results. Display analysis of variance table.

```
sprinters <- read.table("./Data/mens200m.txt", header=T)
attach(sprinters)

model <- lm(Men200m ~ Year)

plot(x=Year, y=Men200m,
     xlab="Year", ylab="Men's 200m time (secs)",
     panel.last = lines(sort(Year), fitted(model)[order(Year)]))
```




```
summary(model)
```

```
##
## Call:
## lm(formula = Men200m ~ Year)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.51154 -0.16441 -0.03034  0.21721  0.60199
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 76.153369   4.152226   18.34 5.61e-14 ***
## Year        -0.028383   0.002129  -13.33 2.07e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2981 on 20 degrees of freedom
## Multiple R-squared:  0.8988, Adjusted R-squared:  0.8938
## F-statistic: 177.7 on 1 and 20 DF,  p-value: 2.074e-11
```

```
# Coefficients:
#              Estimate Std. Error t value Pr(>|t|)
# (Intercept) 76.153369   4.152226   18.34 5.61e-14 ***
# Year        -0.028383   0.002129  -13.33 2.07e-11 ***
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
#
# Residual standard error: 0.2981 on 20 degrees of freedom
# Multiple R-squared:  0.8988,  Adjusted R-squared:  0.8938
# F-statistic: 177.7 on 1 and 20 DF,  p-value: 2.074e-11

anova(model)
```

```
## Analysis of Variance Table
##
## Response: Men200m
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## Year       1 15.7964 15.7964  177.72 2.074e-11 ***
## Residuals 20  1.7777  0.0889
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Analysis of Variance Table
# Response: Men200m
#           Df  Sum Sq Mean Sq F value    Pr(>F)
# Year       1 15.7964 15.7964  177.72 2.074e-11 ***
# Residuals 20  1.7777  0.0889
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
# SSTO = SSR + SSE = 15.7964 + 1.7777 = 17.5741.

detach(sprinters)
```

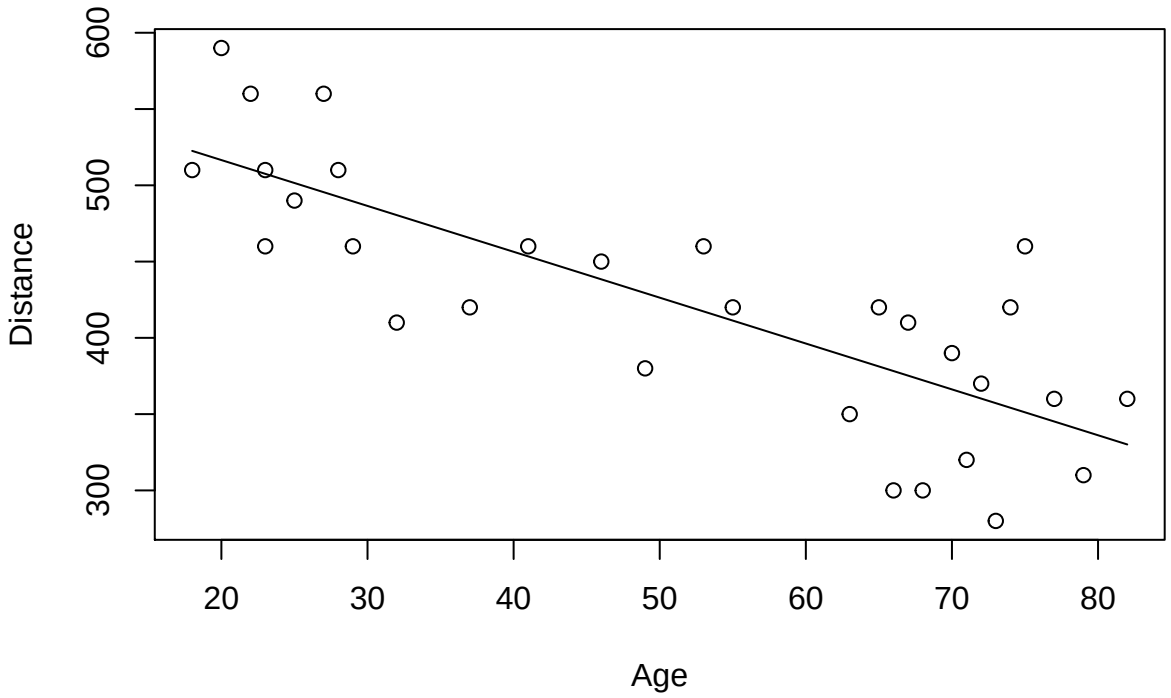
### Highway sign reading distance and driver age

Load the signdist data. Fit a simple linear regression model with y = Distance and x = Age. Display a scatterplot of the data with the simple linear regression line. Display model results. Calculate confidence intervals for the slope.

```
signdist <- read.table("./Data/signdist.txt", header=T)
attach(signdist)

model <- lm(Distance ~ Age)

plot(x=Age, y=Distance,
     xlab="Age", ylab="Distance",
     panel.last = lines(sort(Age), fitted(model)[order(Age)]))
```




```
summary(model)
```

```
##
## Call:
## lm(formula = Distance ~ Age)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -78.231 -41.710   7.646  33.552 108.831
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 576.6819    23.4709  24.570  < 2e-16 ***
## Age          -3.0068     0.4243  -7.086 1.04e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 49.76 on 28 degrees of freedom
## Multiple R-squared:  0.642,  Adjusted R-squared:  0.6292
## F-statistic: 50.21 on 1 and 28 DF,  p-value: 1.041e-07
```

```
# Coefficients:
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept) 576.6819    23.4709  24.570  < 2e-16 ***
# Age          -3.0068     0.4243  -7.086 1.04e-07 ***
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
# Residual standard error: 49.76 on 28 degrees of freedom
# Multiple R-squared:  0.642,  Adjusted R-squared:  0.6292
# F-statistic: 50.21 on 1 and 28 DF,  p-value: 1.041e-07

confint(model, parm="Age", level=0.95)
```

```
##         2.5 %   97.5 %
## Age -3.876051 -2.13762
```

```
#                  2.5 %    97.5 %
# Age          -3.876051  -2.13762

confint(model, parm="Age", level=0.99)
```

```
##         0.5 %   99.5 %
## Age -4.179391 -1.83428
```

```
#                  0.5 %    99.5 %
# Age          -4.179391  -1.83428

detach(signdist)
```

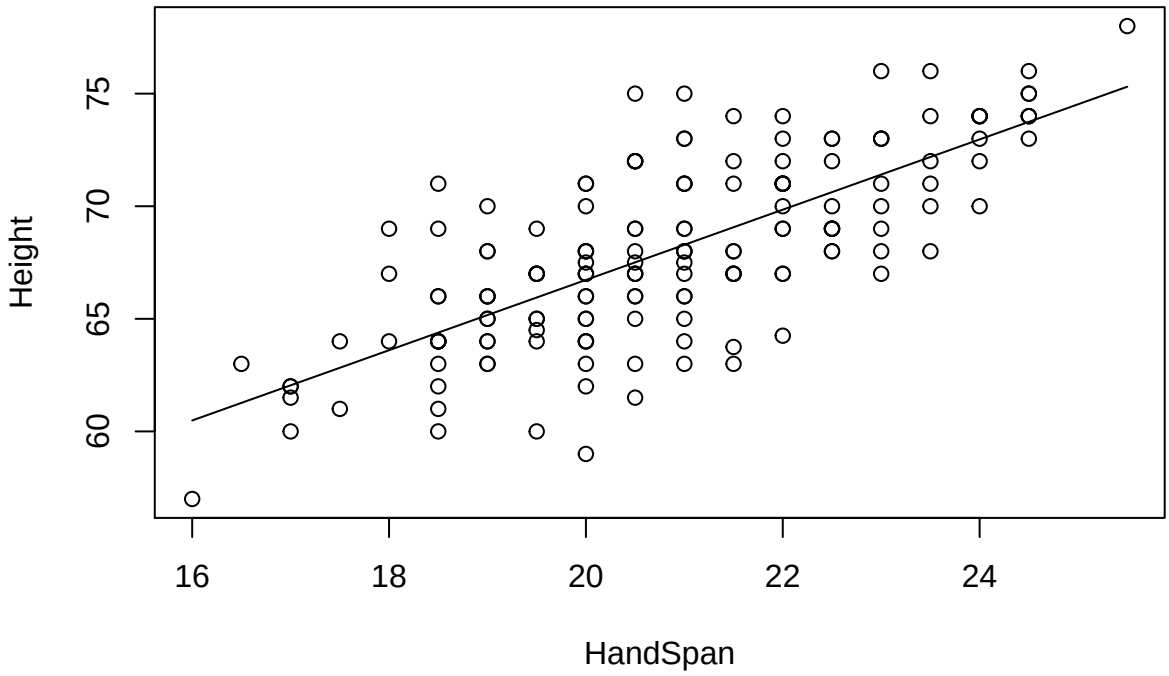
## Handcode and height

Load the handheight data. Fit a simple linear regression model with y = Height and x = Handcode Display a scatterplot of the data with the simple linear regression line. Display model results. Display analysis of variance table.

```
handheight <- read.table("./Data/handheight.txt", header=T)
attach(handheight)

model <- lm(Height ~ HandSpan)

plot(x=HandSpan, y=Height,
     xlab="HandSpan", ylab="Height",
     panel.last = lines(sort(HandSpan), fitted(model)[order(HandSpan)]))
```




```
summary(model)
```

```
##
## Call:
```

```
## lm(formula = Height ~ HandSpan)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7266 -1.7266 -0.1666  1.4933  7.4933
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  35.5250     2.3160   15.34   <2e-16 ***
## HandSpan      1.5601     0.1105   14.11   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.744 on 165 degrees of freedom
## Multiple R-squared:  0.5469, Adjusted R-squared:  0.5442
## F-statistic: 199.2 on 1 and 165 DF,  p-value: < 2.2e-16
```

```
# Coefficients:
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)  35.5250     2.3160   15.34   <2e-16 ***
# HandSpan      1.5601     0.1105   14.11   <2e-16 ***
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
# Residual standard error: 2.744 on 165 degrees of freedom
# Multiple R-squared:  0.5469,  Adjusted R-squared:  0.5442
# F-statistic: 199.2 on 1 and 165 DF,  p-value: < 2.2e-16

anova(model)
```

```
## Analysis of Variance Table
##
## Response: Height
##            Df Sum Sq Mean Sq F value    Pr(>F)
## HandSpan    1 1500.1 1500.06  199.17 < 2.2e-16 ***
## Residuals 165 1242.7    7.53
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Analysis of Variance Table
# Response: Height
#            Df Sum Sq Mean Sq F value    Pr(>F)
# HandSpan    1 1500.1 1500.06  199.17 < 2.2e-16 ***
# Residuals 165 1242.7    7.53
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
# SSTO = SSR + SSE = 1500.1 + 1242.7 = 2742.8.

detach(handheight)
```

## Checking account deposits

Load the newaccounts data. Fit a simple linear regression model with y = New and x = Size Display a scatterplot of the data with the simple linear regression line. Display model results. Display lack of fit analysis of variance table. Display usual analysis of variance table.

```
library(EnvStats) # EnvStats must be installed first
```

```
##
## Attaching package: 'EnvStats'
```

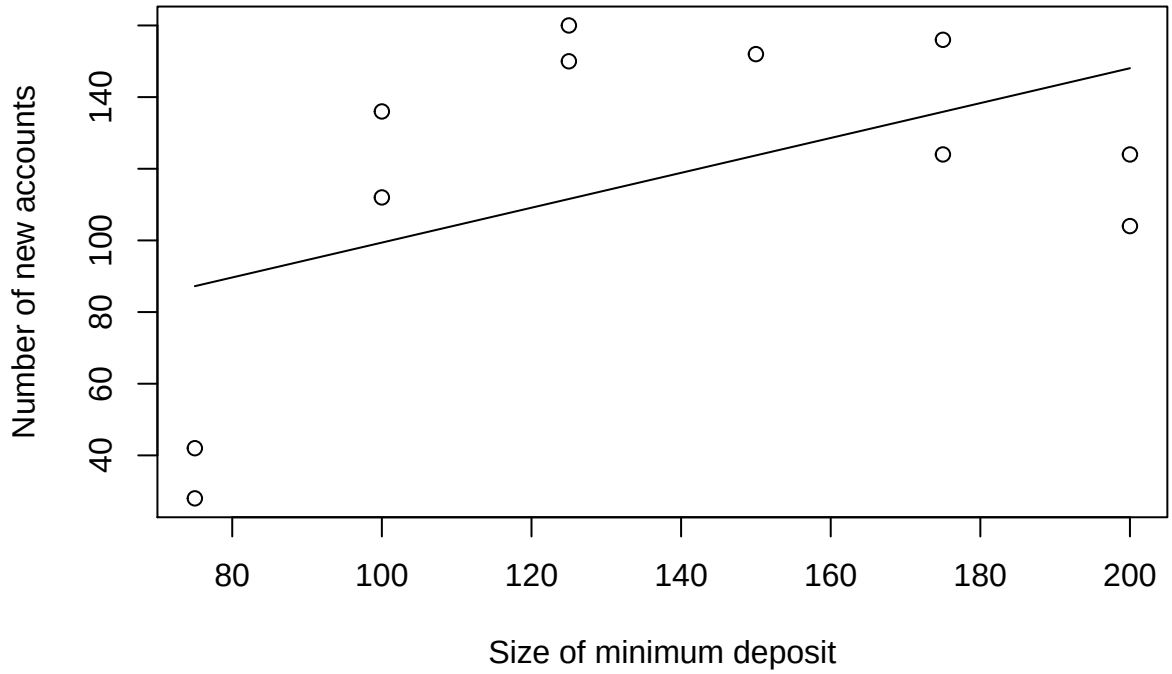
```
## The following objects are masked from 'package:stats':
##
##     predict, predict.lm
```

```
## The following object is masked from 'package:base':
##
##     print.default
```

```
newaccounts <- read.table("./Data/newaccounts.txt", header=T)
attach(newaccounts)

model <- lm(New ~ Size)

plot(x=Size, y=New,
     xlab="Size of minimum deposit", ylab="Number of new accounts",
     panel.last = lines(sort(Size), fitted(model)[order(Size)]))
```




```
summary(model)
```

```
##
## Call:
## lm(formula = New ~ Size)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -59.23 -34.06  12.61  32.44  48.44
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  50.7225    39.3979   1.287     0.23
## Size          0.4867     0.2747   1.772     0.11
##
## Residual standard error: 40.47 on 9 degrees of freedom
## Multiple R-squared:  0.2586, Adjusted R-squared:  0.1762
## F-statistic: 3.139 on 1 and 9 DF,  p-value: 0.1102
```

```
# Coefficients:
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)  50.7225    39.3979   1.287     0.23
# Size          0.4867     0.2747   1.772     0.11
#
# Residual standard error: 40.47 on 9 degrees of freedom
# Multiple R-squared:  0.2586,  Adjusted R-squared:  0.1762
# F-statistic: 3.139 on 1 and 9 DF,  p-value: 0.1102

# replaced since alr3 is not available
#alr3::pureErrorAnova(model) # Lack of fit anova table
EnvStats::anovaPE(model) # Lack of fit anova table
```

```
##                 Df  Sum Sq Mean Sq F value   Pr(>F)
## Size             1  5141.3  5141.3  22.393 0.005186 **
## Lack of Fit      4 13593.6  3398.4  14.801 0.005594 **
## Pure Error       5  1148.0   229.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Analysis of Variance Table
# Response: New
#              Df  Sum Sq Mean Sq F value   Pr(>F)
# Size          1  5141.3  5141.3  22.393 0.005186 **
# Residuals     9 14741.6  1638.0
#  Lack of fit  4 13593.6  3398.4  14.801 0.005594 **
#  Pure Error   5  1148.0   229.6
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
# NOTE: The F value for Size uses MSPE in its denominator.
# So, F value for Size is 5141.3 / 229.6 = 22.393.
# Thus it differs from the F value for Size in the usual anova table:

anova(model)
```

```
## Analysis of Variance Table
##
## Response: New
##           Df  Sum Sq Mean Sq F value Pr(>F)
## Size       1  5141.3  5141.3  3.1389 0.1102
## Residuals  9 14741.6  1638.0
```

```
# Analysis of Variance Table
# Response: New
#           Df  Sum Sq Mean Sq F value Pr(>F)
# Size       1  5141.3  5141.3  3.1389 0.1102
# Residuals  9 14741.6  1638.0
# NOTE: Here the F value for Size uses MSE in its denominator.
# So, F value for Size is 5141.3 / 1638.0 = 3.1389.
```

```
detach(newaccounts)
```