Lesson 11

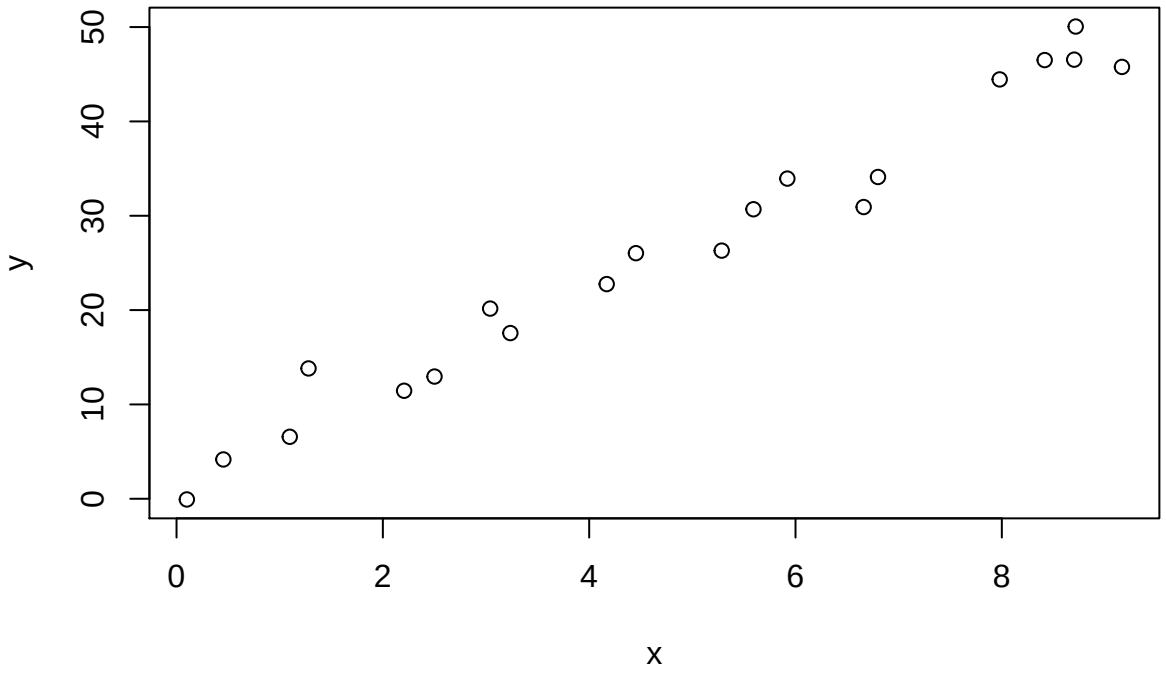
Christopher A. Swenson (chris@cswenson.com)

11/27/2021

Influence 1 (no influential points)

Load the influence1 data. Create a scatterplot of the data.

```
influence1 <- read.table("./Data/influence1.txt", header=T)
attach(influence1)

plot(x, y)
```

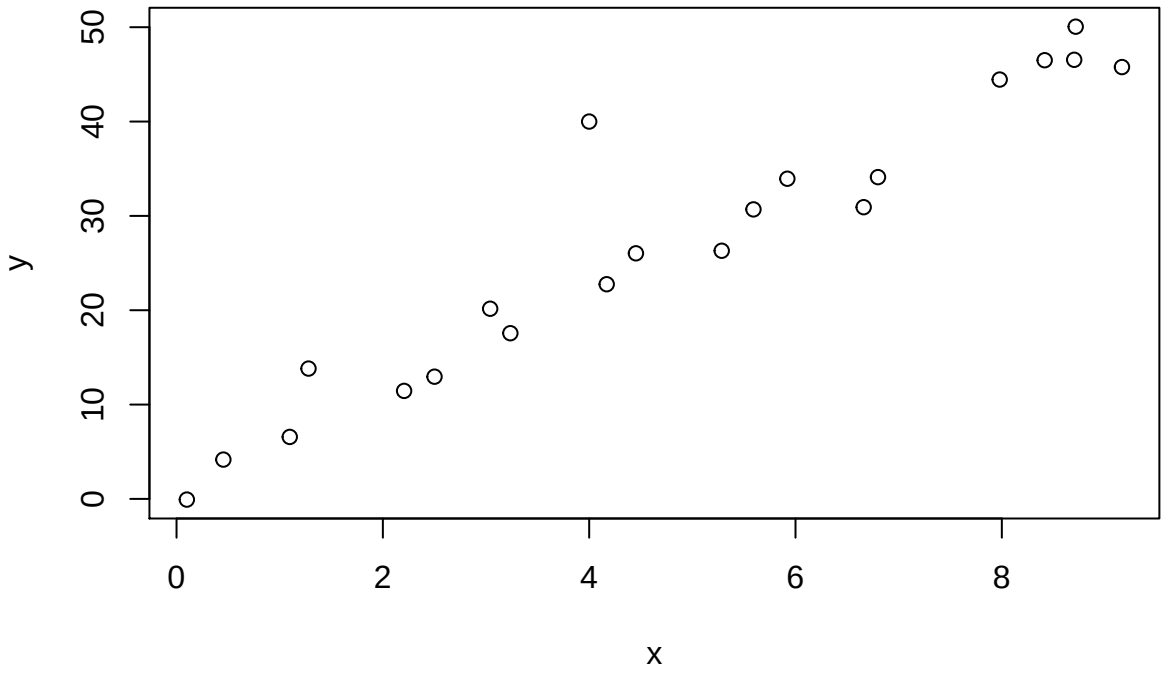



```
detach(influence1)
```

Influence 2 (outlier, low leverage, not influential)

Load the influence2 data. Create a scatterplot of the data. Fit a simple linear regression model to all the data. Fit a simple linear regression model to the data excluding observation #21. Add regression lines to the scatterplot, one for each model. Calculate leverages, standardized residuals, studentized residuals, DFFITS, Cook's distances.

```
influence2 <- read.table("./Data/influence2.txt", header=T)
attach(influence2)

plot(x, y)
```



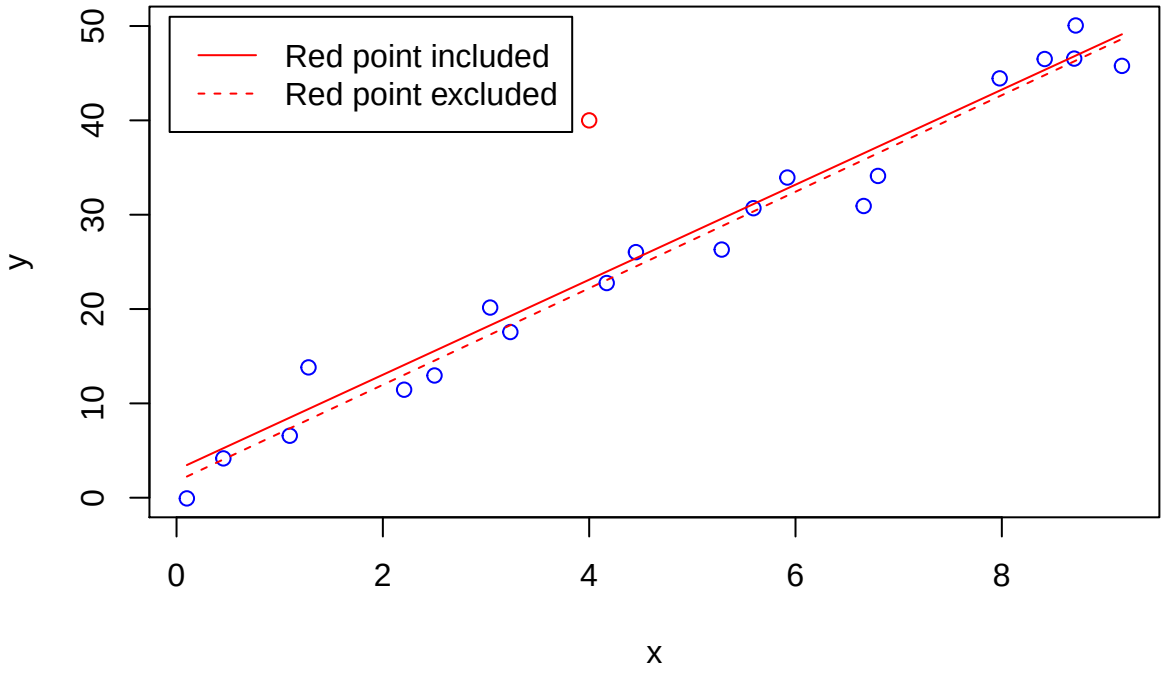

```
model.1 <- lm(y ~ x)
summary(model.1)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.587 -2.620 -1.077  1.157 16.893
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.9576     2.0091   1.472    0.157
## x             5.0373     0.3633  13.865 2.18e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.711 on 19 degrees of freedom
## Multiple R-squared:  0.9101, Adjusted R-squared:  0.9053
## F-statistic: 192.2 on 1 and 19 DF,  p-value: 2.179e-11
```

```
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)   2.9576     2.0091   1.472    0.157
# x             5.0373     0.3633  13.865 2.18e-11 ***
# ---
# Residual standard error: 4.711 on 19 degrees of freedom
# Multiple R-squared:  0.9101,  Adjusted R-squared:  0.9053
# F-statistic: 192.2 on 1 and 19 DF,  p-value: 2.179e-11

model.2 <- lm(y ~ x, subset=1:20) # exclude obs #21
summary(model.2)
```

```
##
## Call:
## lm(formula = y ~ x, subset = 1:20)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8911 -1.7580 -0.0998  1.7552  5.5365
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.7322     1.1205   1.546     0.14
## x             5.1169     0.2003  25.551 1.35e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.592 on 18 degrees of freedom
## Multiple R-squared:  0.9732, Adjusted R-squared:  0.9717
## F-statistic: 652.8 on 1 and 18 DF,  p-value: 1.353e-15
```

```
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)   1.7322     1.1205   1.546     0.14
# x             5.1169     0.2003  25.551 1.35e-15 ***
# ---
# Residual standard error: 2.592 on 18 degrees of freedom
# Multiple R-squared:  0.9732,  Adjusted R-squared:  0.9717
# F-statistic: 652.8 on 1 and 18 DF,  p-value: 1.353e-15

plot(x=x, y=y, col=ifelse(Row<=20, "blue", "red"),
     panel.last = c(lines(sort(x), fitted(model.1)[order(x)], col="red"),
                    lines(sort(x[-21]), fitted(model.2)[order(x[-21])],
                          col="red", lty=2)))
legend("topleft", col="red", lty=c(1,2),
       inset=0.02, legend=c("Red point included", "Red point excluded"))
```



```
lev <- hatvalues(model.1)
round(lev, 6)
```

```
##        1        2        3        4        5        6        7        8
## 0.176297 0.157454 0.127015 0.119313 0.086145 0.077744 0.065028 0.061276
##        9       10       11       12       13       14       15       16
## 0.048147 0.049628 0.049313 0.051829 0.055760 0.069310 0.072580 0.109616
##       17       18       19       20       21
## 0.127489 0.141136 0.140453 0.163492 0.050974
```

```
#        1        2        3        4        5        6        7        8        9
# 0.176297 0.157454 0.127015 0.119313 0.086145 0.077744 0.065028 0.061276 0.048147
#       10       11       12       13       14       15       16       17       18
# 0.049628 0.049313 0.051829 0.055760 0.069310 0.072580 0.109616 0.127489 0.141136
#       19       20       21
# 0.140453 0.163492 0.050974
sum(lev) # 2
```

```
## [1] 2
```

```
sta <- rstandard(model.1)
round(sta, 6)
```

```
##         1         2         3         4         5         6         7         8
## -0.826351 -0.249154 -0.435445  0.998187 -0.581904 -0.574462  0.413791 -0.371226
##         9        10        11        12        13        14        15        16
##  0.139767 -0.262514 -0.713173 -0.095897  0.252734 -1.229353 -0.683161  0.292644
##        17        18        19        20        21
##  0.262144  0.731458 -0.055615 -0.776800  3.681098
```

```
#         1         2         3         4         5         6         7         8
# -0.826351 -0.249154 -0.435445  0.998187 -0.581904 -0.574462  0.413791 -0.371226
#         9        10        11        12        13        14        15        16
#  0.139767 -0.262514 -0.713173 -0.095897  0.252734 -1.229353 -0.683161  0.292644
#        17        18        19        20        21
#  0.262144  0.731458 -0.055615 -0.776800  3.681098

stu <- rstudent(model.1)
round(stu, 6)
```

```
##         1         2         3         4         5         6         7         8
## -0.819167 -0.242905 -0.425962  0.998087 -0.571499 -0.564060  0.404582 -0.362643
##         9        10        11        12        13        14        15        16
##  0.136110 -0.255977 -0.703633 -0.093362  0.246408 -1.247195 -0.673261  0.285483
##        17        18        19        20        21
##  0.255615  0.722190 -0.054136 -0.768382  6.690129
```

```
#         1         2         3         4         5         6         7         8
# -0.819167 -0.242905 -0.425962  0.998087 -0.571499 -0.564060  0.404582 -0.362643
#         9        10        11        12        13        14        15        16
#  0.136110 -0.255977 -0.703633 -0.093362  0.246408 -1.247195 -0.673261  0.285483
#        17        18        19        20        21
#  0.255615  0.722190 -0.054136 -0.768382  6.690129

dffit <- dffits(model.1)
round(dffit, 6)
```

```
##         1         2         3         4         5         6         7         8
## -0.378974 -0.105007 -0.162478  0.367368 -0.175466 -0.163769  0.106698 -0.092652
##         9        10        11        12        13        14        15        16
##  0.030612 -0.058495 -0.160254 -0.021828  0.059879 -0.340354 -0.188345  0.100168
##        17        18        19        20        21
##  0.097710  0.292757 -0.021884 -0.339696  1.550500
```

```
#         1         2         3         4         5         6         7         8
# -0.378974 -0.105007 -0.162478  0.367368 -0.175466 -0.163769  0.106698 -0.092652
#         9        10        11        12        13        14        15        16
#  0.030612 -0.058495 -0.160254 -0.021828  0.059879 -0.340354 -0.188345  0.100168
#        17        18        19        20        21
#  0.097710  0.292757 -0.021884 -0.339696  1.550500

cook <- cooks.distance(model.1)
round(cook, 6)
```

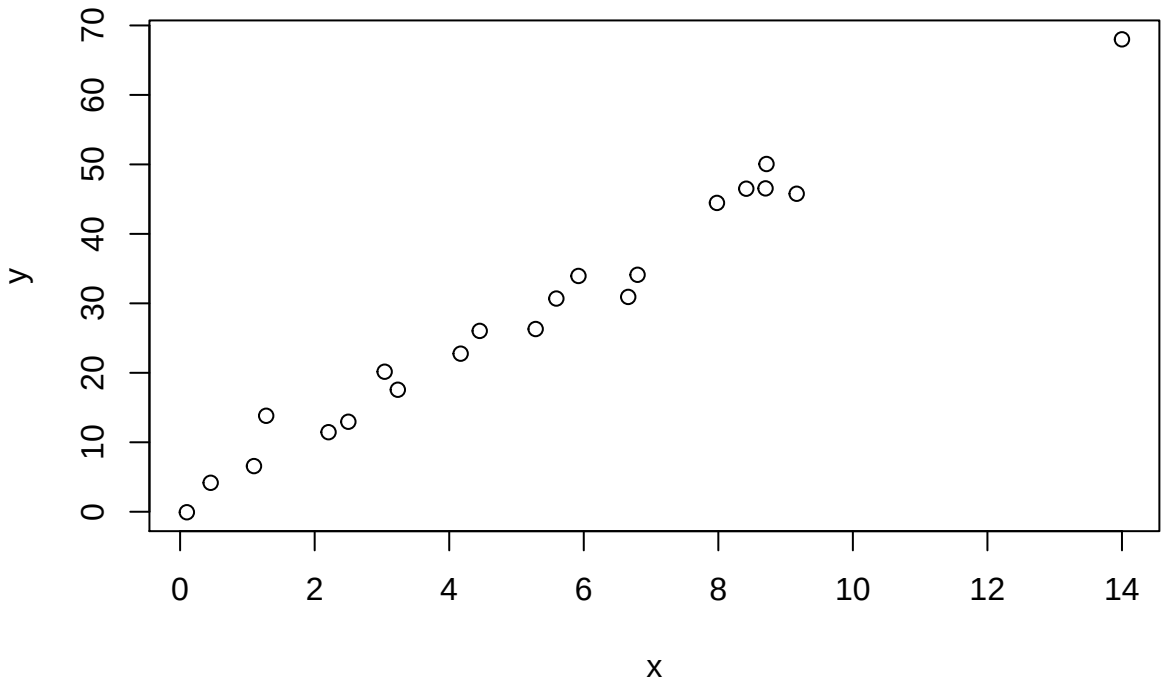
```
##        1        2        3        4        5        6        7        8
## 0.073076 0.005800 0.013794 0.067493 0.015960 0.013909 0.005954 0.004498
##        9       10       11       12       13       14       15       16
## 0.000494 0.001799 0.013191 0.000251 0.001886 0.056275 0.018262 0.005272
##       17       18       19       20       21
## 0.005021 0.043960 0.000253 0.058968 0.363914
```

```
#        1        2        3        4        5        6        7        8
# 0.073076 0.005800 0.013794 0.067493 0.015960 0.013909 0.005954 0.004498
#        9       10       11       12       13       14       15       16
# 0.000494 0.001799 0.013191 0.000251 0.001886 0.056275 0.018262 0.005272
#       17       18       19       20       21
# 0.005021 0.043960 0.000253 0.058968 0.363914

detach(influence2)
```

Influence 3 (high leverage, not an outlier, not influential)

Load the influence3 data. Create a scatterplot of the data. Fit a simple linear regression model to all the data. Fit a simple linear regression model to the data excluding observation #21. Add regression lines to the scatterplot, one for each model. Calculate leverages, DFFITS, Cook's distances.

```
influence3 <- read.table("./Data/influence3.txt", header=T)
attach(influence3)

plot(x, y)
```



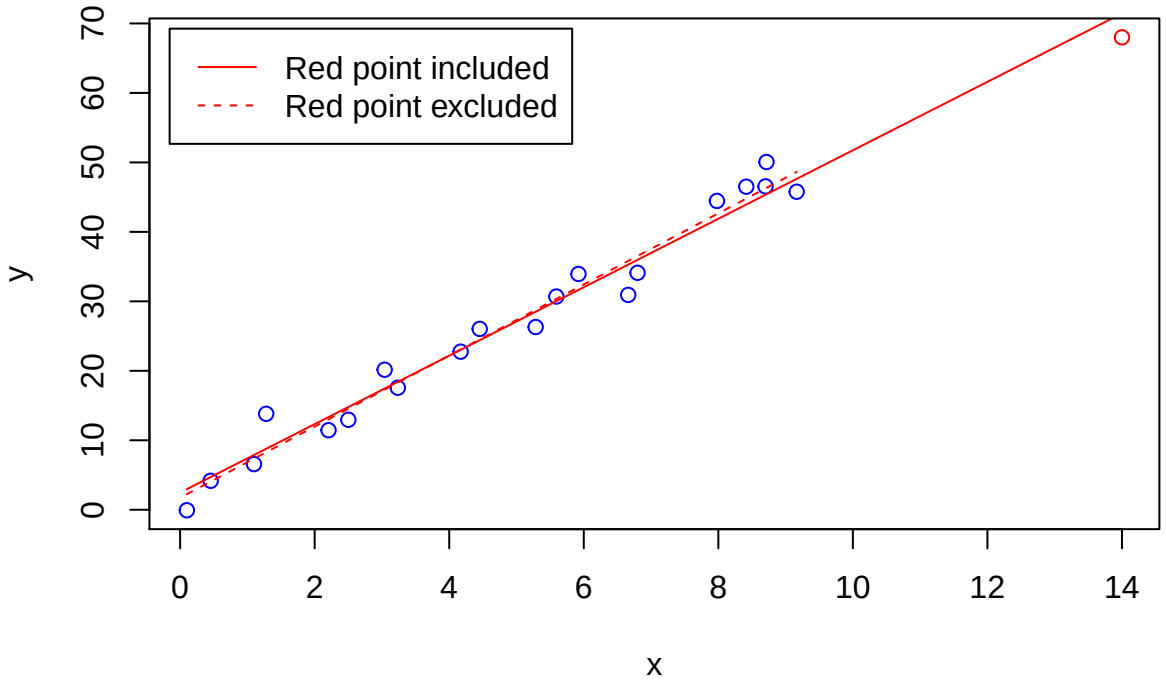

```
model.1 <- lm(y ~ x)
summary(model.1)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.3636 -1.8607 -0.5376  2.2987  5.0434
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.4679     1.0757   2.294   0.0333 *
## x             4.9272     0.1719  28.661   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.709 on 19 degrees of freedom
## Multiple R-squared:  0.9774, Adjusted R-squared:  0.9762
## F-statistic: 821.4 on 1 and 19 DF,  p-value: < 2.2e-16
```

```
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)   2.4679     1.0757   2.294   0.0333 *
# x             4.9272     0.1719  28.661   <2e-16 ***
# ---
# Residual standard error: 2.709 on 19 degrees of freedom
# Multiple R-squared:  0.9774,  Adjusted R-squared:  0.9762
# F-statistic: 821.4 on 1 and 19 DF,  p-value: < 2.2e-16

model.2 <- lm(y ~ x, subset=1:20) # exclude obs #21
summary(model.2)
```

```
##
## Call:
## lm(formula = y ~ x, subset = 1:20)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8911 -1.7580 -0.0998  1.7552  5.5365
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.7322     1.1205   1.546     0.14
## x             5.1169     0.2003  25.551 1.35e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.592 on 18 degrees of freedom
## Multiple R-squared:  0.9732, Adjusted R-squared:  0.9717
## F-statistic: 652.8 on 1 and 18 DF,  p-value: 1.353e-15
```

```
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)   1.7322     1.1205   1.546     0.14
# x             5.1169     0.2003  25.551 1.35e-15 ***
# ---
# Residual standard error: 2.592 on 18 degrees of freedom
# Multiple R-squared:  0.9732,  Adjusted R-squared:  0.9717
# F-statistic: 652.8 on 1 and 18 DF,  p-value: 1.353e-15

plot(x=x, y=y, col=ifelse(Row<=20, "blue", "red"),
     panel.last = c(lines(sort(x), fitted(model.1)[order(x)], col="red"),
                    lines(sort(x[-21]), fitted(model.2)[order(x[-21])],
                          col="red", lty=2)))
legend("topleft", col="red", lty=c(1,2),
       inset=0.02, legend=c("Red point included", "Red point excluded"))
```




```
lev <- hatvalues(model.1)
round(lev, 6)
```

```
##        1        2        3        4        5        6        7        8
## 0.153481 0.139367 0.116292 0.110382 0.084374 0.077557 0.066879 0.063589
##        9       10       11       12       13       14       15       16
## 0.050033 0.052121 0.047632 0.048156 0.049557 0.055893 0.057574 0.078121
##       17       18       19       20       21
## 0.088549 0.096634 0.096227 0.110048 0.357535
```

```
#        1        2        3        4        5        6        7        8        9
# 0.153481 0.139367 0.116292 0.110382 0.084374 0.077557 0.066879 0.063589 0.050033
#       10       11       12       13       14       15       16       17       18
# 0.052121 0.047632 0.048156 0.049557 0.055893 0.057574 0.078121 0.088549 0.096634
#       19       20       21
# 0.096227 0.110048 0.357535
sum(lev) # 2
```

```
## [1] 2
```

```
dffit <- dffits(model.1)
round(dffit, 6)
```

```
##         1         2         3         4         5         6         7         8
## -0.525036 -0.083882 -0.182326  0.758981 -0.218230 -0.201548  0.277728 -0.082294
##         9        10        11        12        13        14        15        16
##  0.138643 -0.022210 -0.184873  0.055235  0.197411 -0.424484 -0.172490  0.299173
##        17        18        19        20        21
##  0.309606  0.630493  0.149474 -0.250945 -1.238416
```

```
#         1         2         3         4         5         6         7         8
# -0.525036 -0.083882 -0.182326  0.758981 -0.218230 -0.201548  0.277728 -0.082294
#         9        10        11        12        13        14        15        16
#  0.138643 -0.022210 -0.184873  0.055235  0.197411 -0.424484 -0.172490  0.299173
#        17        18        19        20        21
```

```
#  0.309606  0.630493  0.149474 -0.250945 -1.238416

cook <- cooks.distance(model.1)
round(cook, 6)
```

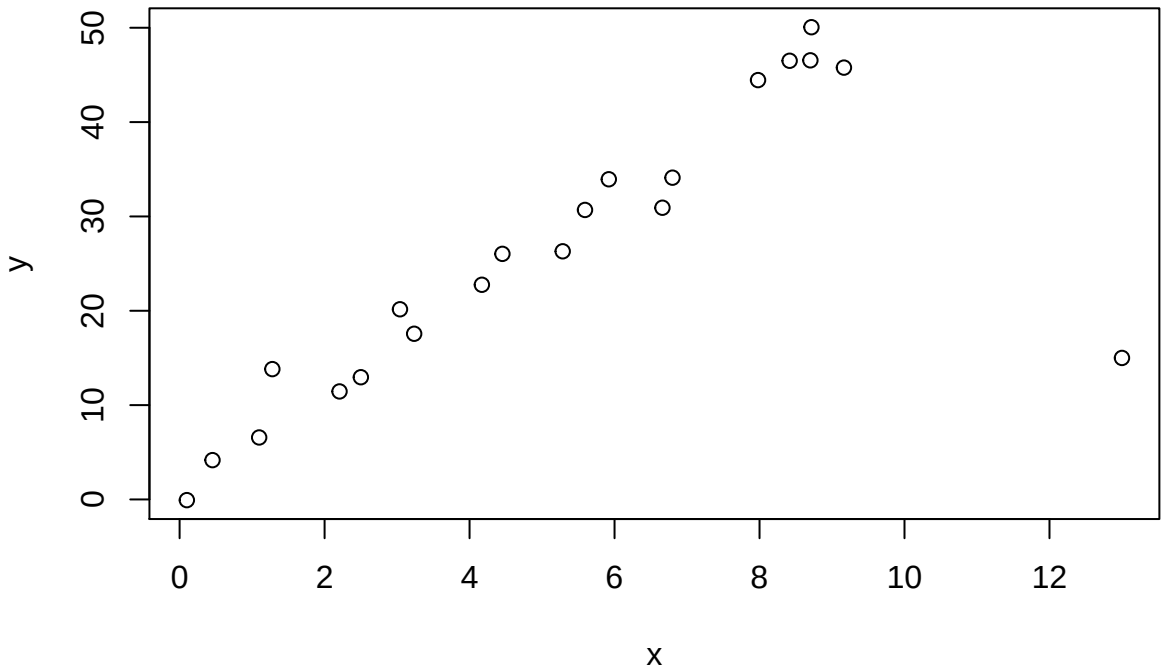
```
##        1        2        3        4        5        6        7        8
## 0.134157 0.003705 0.017302 0.241690 0.024433 0.020879 0.038412 0.003555
##        9       10       11       12       13       14       15       16
## 0.009943 0.000260 0.017379 0.001605 0.019748 0.081344 0.015289 0.044620
##       17       18       19       20       21
## 0.047961 0.173901 0.011656 0.032322 0.701965
```

```
#        1        2        3        4        5        6        7        8
# 0.134157 0.003705 0.017302 0.241690 0.024433 0.020879 0.038412 0.003555
#        9       10       11       12       13       14       15       16
# 0.009943 0.000260 0.017379 0.001605 0.019748 0.081344 0.015289 0.044620
#       17       18       19       20       21
# 0.047961 0.173901 0.011656 0.032322 0.701965

detach(influence3)
```

Influence 4 (outlier, high leverage, influential)

Load the influence4 data. Create a scatterplot of the data. Fit a simple linear regression model to all the data. Fit a simple linear regression model to the data excluding observation #21. Add regression lines to the scatterplot, one for each model. Calculate leverages, DFFITS, Cook's distances.

```
influence4 <- read.table("./Data/influence4.txt", header=T)
attach(influence4)

plot(x, y)
```




```
model.1 <- lm(y ~ x)
summary(model.1)
```

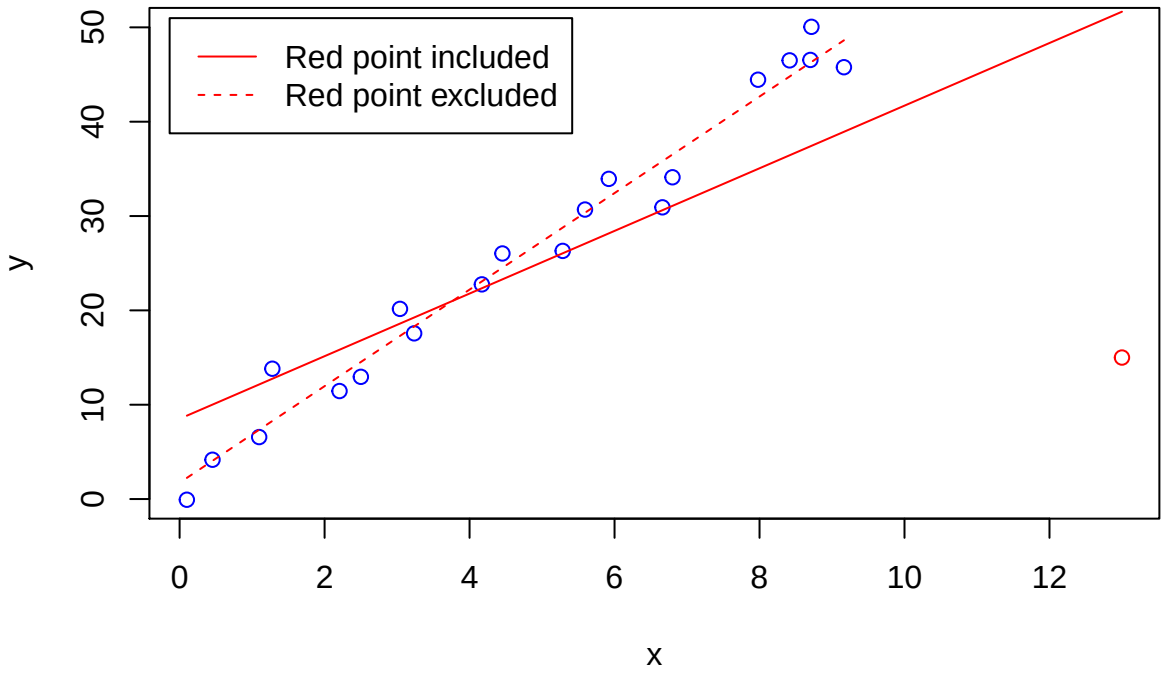
```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -36.662  -3.851   1.063   5.779  12.617
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   8.5046     4.2224   2.014 0.058374 .
## x             3.3198     0.6862   4.838 0.000114 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.45 on 19 degrees of freedom
## Multiple R-squared:  0.5519, Adjusted R-squared:  0.5284
## F-statistic: 23.41 on 1 and 19 DF,  p-value: 0.0001143
```

```
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)   8.5046     4.2224   2.014 0.058374 .
# x             3.3198     0.6862   4.838 0.000114 ***
# ---
# Residual standard error: 10.45 on 19 degrees of freedom
# Multiple R-squared:  0.5519,  Adjusted R-squared:  0.5284
# F-statistic: 23.41 on 1 and 19 DF,  p-value: 0.0001143

model.2 <- lm(y ~ x, subset=1:20) # exclude obs #21
summary(model.2)
```

```
##
## Call:
## lm(formula = y ~ x, subset = 1:20)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8911 -1.7580 -0.0998  1.7552  5.5365
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.7322     1.1205   1.546     0.14
## x             5.1169     0.2003  25.551 1.35e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.592 on 18 degrees of freedom
## Multiple R-squared:  0.9732, Adjusted R-squared:  0.9717
## F-statistic: 652.8 on 1 and 18 DF,  p-value: 1.353e-15
```

```
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)   1.7322     1.1205   1.546     0.14
# x             5.1169     0.2003  25.551 1.35e-15 ***
# ---
# Residual standard error: 2.592 on 18 degrees of freedom
# Multiple R-squared:  0.9732,  Adjusted R-squared:  0.9717
```

```
# F-statistic: 652.8 on 1 and 18 DF,  p-value: 1.353e-15

plot(x=x, y=y, col=ifelse(Row<=20, "blue", "red"),
     panel.last = c(lines(sort(x), fitted(model.1)[order(x)], col="red"),
                    lines(sort(x[-21]), fitted(model.2)[order(x[-21])],
                          col="red", lty=2)))
legend("topleft", col="red", lty=c(1,2),
       inset=0.02, legend=c("Red point included", "Red point excluded"))
```




```
lev <- hatvalues(model.1)
round(lev, 6)
```

```
##        1        2        3        4        5        6        7        8
## 0.158964 0.143985 0.119522 0.113263 0.085774 0.078589 0.067369 0.063924
##        9       10       11       12       13       14       15       16
## 0.049897 0.052019 0.047667 0.048354 0.049990 0.057084 0.058943 0.081446
##       17       18       19       20       21
## 0.092800 0.101587 0.101146 0.116146 0.311532
#        1        2        3        4        5        6        7        8        9
# 0.158964 0.143985 0.119522 0.113263 0.085774 0.078589 0.067369 0.063924 0.049897
#       10       11       12       13       14       15       16       17       18
# 0.052019 0.047667 0.048354 0.049990 0.057084 0.058943 0.081446 0.092800 0.101587
#       19       20       21
# 0.101146 0.116146 0.311532
sum(lev) # 2
```

```
## [1] 2
```

```
dffit <- dffits(model.1)
round(dffit, 6)
```

```
##          1          2          3          4          5          6          7
##  -0.402761  -0.243756  -0.205848   0.037612  -0.131355  -0.109593   0.040473
```

```
##          8          9         10         11         12         13         14
##  -0.042401   0.060224   0.009181   0.005430   0.078165   0.127828   0.007230
##         15         16         17         18         19         20         21
##   0.073067   0.280501   0.323599   0.436114   0.308869   0.249206 -11.467011
```

```
#         1          2          3          4          5          6          7
# -0.402761  -0.243756  -0.205848   0.037612  -0.131355  -0.109593   0.040473
#         8          9         10         11         12         13         14
# -0.042401   0.060224   0.009181   0.005430   0.078165   0.127828   0.007230
#        15         16         17         18         19         20         21
#  0.073067   0.280501   0.323599   0.436114   0.308869   0.249206 -11.467011
```

```
cook <- cooks.distance(model.1)
round(cook, 6)
```

```
##        1        2        3        4        5        6        7        8
## 0.081718 0.030755 0.021983 0.000746 0.009014 0.006290 0.000863 0.000947
##        9       10       11       12       13       14       15       16
## 0.001907 0.000044 0.000016 0.003203 0.008478 0.000028 0.002804 0.039575
##       17       18       19       20       21
## 0.052293 0.091802 0.048085 0.031938 4.048013
```

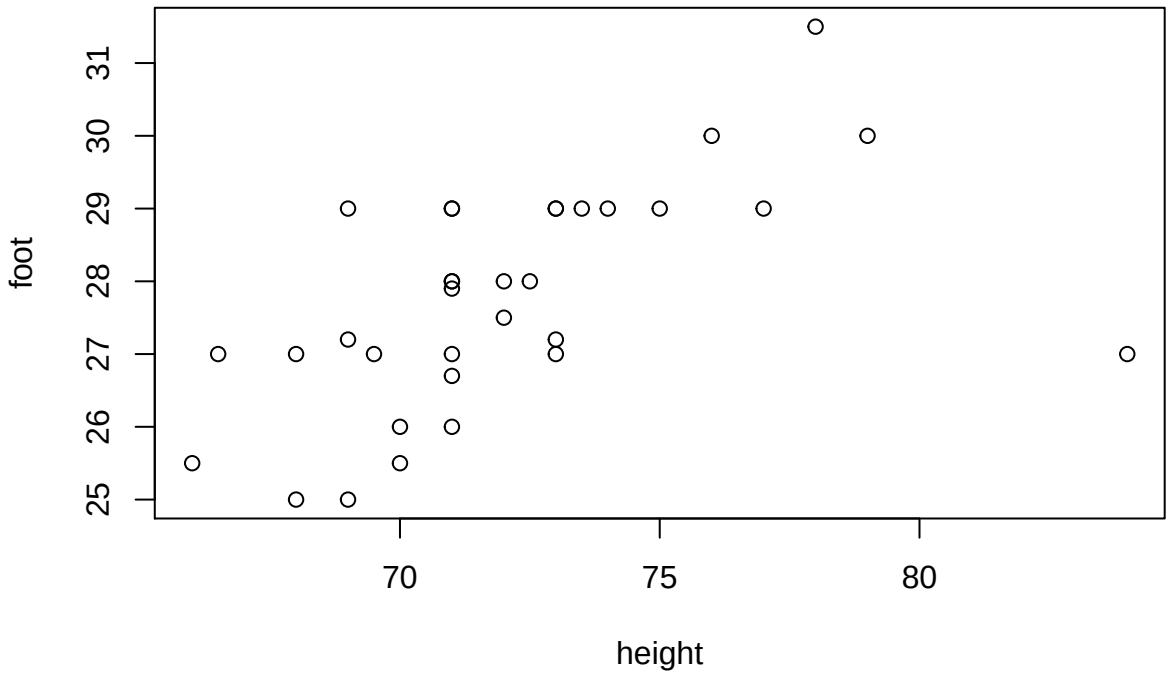
```
#        1        2        3        4        5        6        7        8
# 0.081718 0.030755 0.021983 0.000746 0.009014 0.006290 0.000863 0.000947
#        9       10       11       12       13       14       15       16
# 0.001907 0.000044 0.000016 0.003203 0.008478 0.000028 0.002804 0.039575
#       17       18       19       20       21
# 0.052293 0.091802 0.048085 0.031938 4.048013
```

```
detach(influence4)
```

Foot length and height (outlier, high leverage, influential)

Load the height_foot data. Create a scatterplot of the data. Fit a simple linear regression model to all the data. Fit a simple linear regression model to the data excluding observation #28. Calculate DFFITS and Cook's distance for obs #28.

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

plot(height, foot)
```




```
model.1 <- lm(foot ~ height)
summary(model.1)
```

```
##
## Call:
## lm(formula = foot ~ height)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5447 -0.8100  0.1903  0.7897  2.3559
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.93577    4.43778   2.464 0.019477 *
## height       0.23344    0.06151   3.795 0.000643 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.286 on 31 degrees of freedom
## Multiple R-squared:  0.3173, Adjusted R-squared:  0.2952
## F-statistic: 14.41 on 1 and 31 DF,  p-value: 0.0006428
```

```
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept) 10.93577    4.43778   2.464 0.019477 *
# height       0.23344    0.06151   3.795 0.000643 ***
# ---
# Residual standard error: 1.286 on 31 degrees of freedom
# Multiple R-squared:  0.3173,  Adjusted R-squared:  0.2952
# F-statistic: 14.41 on 1 and 31 DF,  p-value: 0.0006428

which(height>80) # 28
```

```
## [1] 28
```

```r
model.2 <- lm(foot ~ height, subset=(1:33)[-28]) # exclude obs #28
summary(model.2)
```

```
##
## Call:
## lm(formula = foot ~ height, subset = (1:33)[-28])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.74925 -0.81825  0.07875  0.58075  2.25075
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.25313    4.33232   0.058    0.954
## height       0.38400    0.06038   6.360 5.12e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.028 on 30 degrees of freedom
## Multiple R-squared:  0.5741, Adjusted R-squared:  0.5599
## F-statistic: 40.45 on 1 and 30 DF,  p-value: 5.124e-07
```

```r
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)  0.25313    4.33232   0.058    0.954
# height       0.38400    0.06038   6.360 5.12e-07 ***
# ---
# Residual standard error: 1.028 on 30 degrees of freedom
# Multiple R-squared:  0.5741,  Adjusted R-squared:  0.5599
# F-statistic: 40.45 on 1 and 30 DF,  p-value: 5.124e-07

dffit <- dffits(model.1)
dffit[28] # -3.200223
```

```
##        28
## -3.200223
```

```r
cook <- cooks.distance(model.1)
cook[28] # 3.274466
```

```
##       28
## 3.274466
```

```r
detach(heightfoot)
```

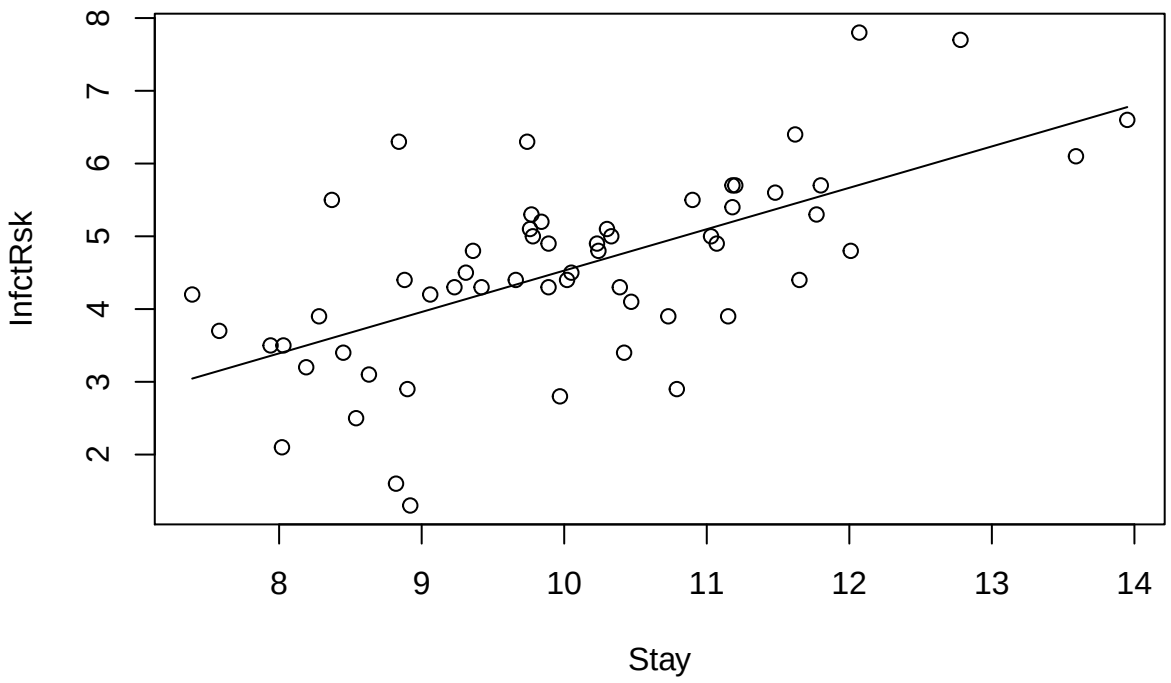
Hospital infection risk (two outliers, high leverages)

Load the infection risk data. Fit a simple linear regression model to all the data. Create a scatterplot of the data and add the regression line. Display influence measures for influential points, including DFFITS, Cook's distances, and leverages (hat).

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

model <- lm(InfctRsk ~ Stay)
summary(model)
```

```
##
## Call:
## lm(formula = InfctRsk ~ Stay)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6145 -0.4660  0.1388  0.4970  2.4310
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.15982    0.95580  -1.213     0.23
## Stay         0.56887    0.09416   6.041  1.3e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.024 on 56 degrees of freedom
## Multiple R-squared:  0.3946, Adjusted R-squared:  0.3838
## F-statistic:  36.5 on 1 and 56 DF,  p-value: 1.302e-07
```

```
plot(x=Stay, y=InfctRsk,
     panel.last = lines(sort(Stay), fitted(model)[order(Stay)]))
```




```
summary(influence.measures(model))
```

```
## Potentially influential observations of
##   lm(formula = InfctRsk ~ Stay) :
##
##    dfb.1_ dfb.Stay dffit cov.r   cook.d hat
## 2   0.33  -0.28     0.44  0.86_*  0.09   0.03
## 12  0.16  -0.17    -0.18  1.17_*  0.02   0.12_*
## 26  0.07  -0.07    -0.08  1.21_*  0.00   0.15_*
## 27 -0.31   0.27    -0.41  0.89_*  0.08   0.03
## 55 -0.34   0.29    -0.46  0.83_*  0.10   0.03
```

```
#     dfb.1_ dfb.Stay dffit   cov.r   cook.d hat
# 2   -0.13   0.09    -0.23    0.94_*  0.02   0.01
# 34  -0.04   0.05     0.05    1.07_*  0.00   0.05
# 40  -0.20   0.17    -0.27    0.94_*  0.04   0.01
# 47   0.85  -0.90    -0.92_*  1.30_*  0.42   0.25_*
# 53  -0.16   0.20     0.30    0.94_*  0.04   0.02
# 93  -0.14   0.09    -0.25    0.92_*  0.03   0.01
# 104 -0.11   0.12     0.14    1.07_*  0.01   0.05_*
# 112  0.64  -0.68    -0.70_*  1.19_*  0.24   0.18_*

detach(infectionrisk)
```