DATA 609 HW3

Bin Lin

2017-9-16

Page 113: #2. The following table gives the elongation e in inches per inch (in./in.) for a given stress S on a steel wire measured in pounds per square inch (lb/in:2). Test the model $e = c_1 S$ by plotting the data. Estimate c1 graphically.

The scatter plot of these two variables are shown below. They display linear relationship with positive correlation. Based on the potential model $e = c_1 S$ which will have a valid k, we are able to tell their relationship follows the proportionality model, therefore their best fit line will pass the origin. In other words, the intercept of regression line is equal to 0.

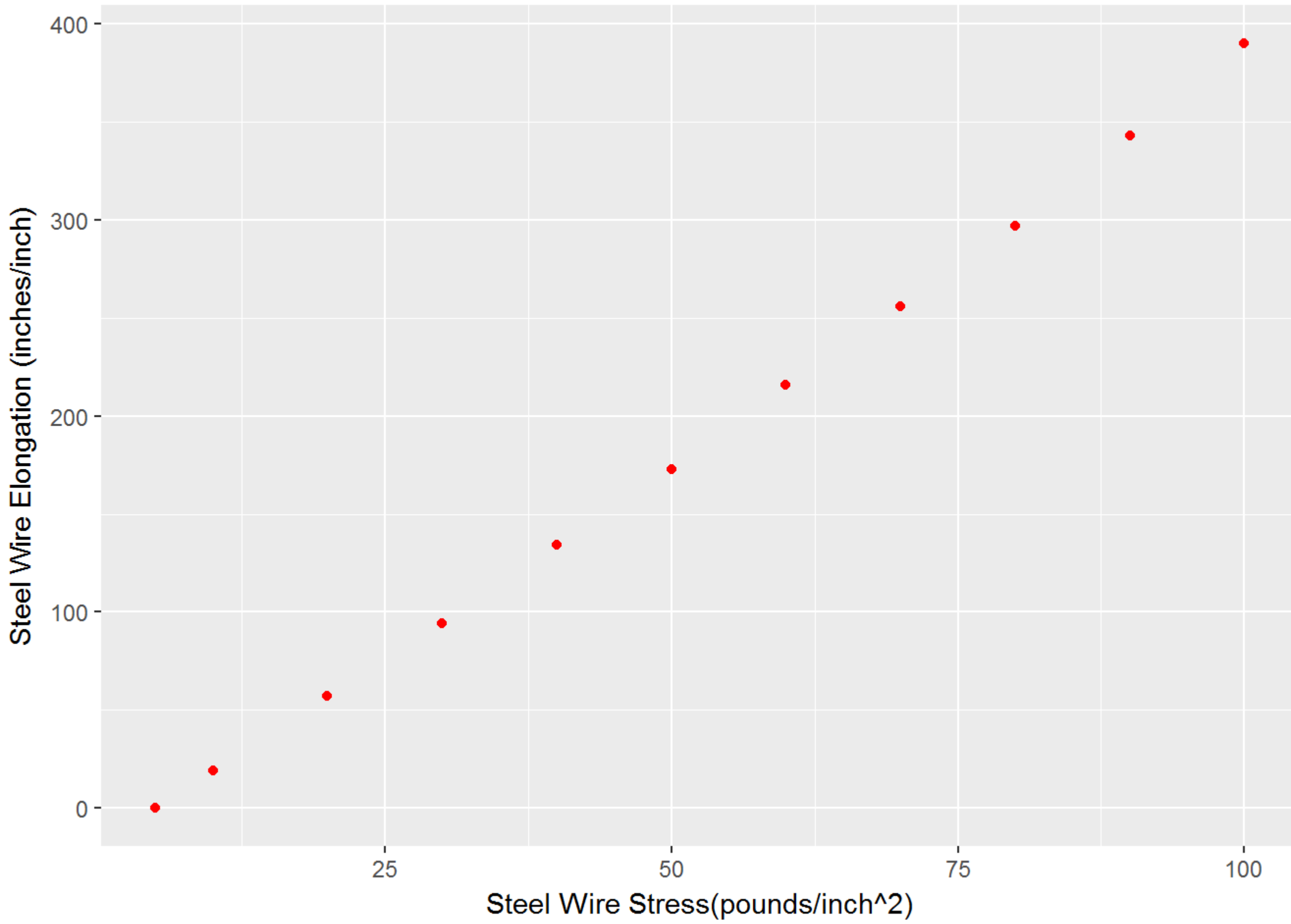
```
library("ggplot2")
```

```
## Warning: package 'ggplot2' was built under R version 3.3.3
```

```
library("boot")

S <- c(5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100)
e <- c(0, 19, 57, 94, 134, 173, 216, 256, 297, 343, 390)
df <- data.frame(S, e)

ggplot() + geom_point(data = df, aes(x = S, y = e), color = "red") + xlab("Steel Wire Stress(pou
nds/inch^2)") + ylab("Steel Wire Elongation (inches/inch)")
```



With the intercept being 0, the model for the best fit line is: e = 3.703 * S, where k = 3.703

```
m1 <- lm(e ~ S + 0)
summary(m1)
```

```
##
## Call:
## lm(formula = e ~ S + 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.517 -17.083 -12.165  -1.248  19.669
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## S  3.70331    0.07435   49.81 2.57e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.59 on 10 degrees of freedom
## Multiple R-squared:  0.996,  Adjusted R-squared:  0.9956
## F-statistic:  2481 on 1 and 10 DF,  p-value: 2.572e-13
```

TO fit the model in a graphical way, since the model will pass origin, the slope of the linear model can be estimated as

$c_1 = \frac{e_{last} - 0}{S_{last} - 0} = \frac{390}{100} = 3.9$

Page 121: #2a. For each of the following data sets, formulate the mathematical model that minimizes the largest deviation between the data and the line y = ax + b. If a computer is available, solve for the estimates of a and b.

$r - r_1 = r - (3.6 - (1a + b)) \geq 0$

$r + r_1 = r + (3.6 - (1a + b)) \geq 0$

$r - r_2 = r - (3 - (2.3a + b)) \geq 0$

$r + r_2 = r + (3 - (2.3a + b)) \geq 0$

$r - r_3 = r - (3.2 - (3.7a + b)) \geq 0$

$r + r_3 = r + (3.2 - (3.7a + b)) \geq 0$

$r - r_4 = r - (5.1 - (4.2a + b)) \geq 0$

$r + r_4 = r + (5.1 - (4.2a + b)) \geq 0$

$r - r_5 = r - (5.3 - (6.1a + b)) \geq 0$

$r + r_5 = r + (5.3 - (6.1a + b)) \geq 0$

$r - r_6 = r - (6.8 - (7a + b)) = \geq 0$

$r + r_6 = r + (6.8 - (7a + b)) = \geq 0$

$r + 1a + b - 3.6 \geq 0$

$r - 1a - b + 3.6 \geq 0$

$r + 2.3a + b - 3 \geq 0$

$r - 2.3a - b + 3 \geq 0$

$r + 3.7a + b - 3.2 \geq 0$

$r - 3.7a - b + 3.2 \geq 0$

$r + 4.2a + b - 5.1 \geq 0$

$r - 4.2a - b + 5.1 \geq 0$

$r + 6.1a + b - 5.3 \geq 0$

$r - 6.1a - b + 5.3 \geq 0$

$r + 7a + b - 6.8 \geq 0$

$r - 7a - b + 6.8 \geq 0$

To solve this linear inequations, I have to use the lp function in lpSolve package. The objective function is to minimize r. Therefore, the final result is that the minimum of r is equal to 0.92. In the meantime, a = 0.533, b = 2.147. The equation that minimize r is y = 0.533 * x + 2.147

```
#install.packages("lpSolve")
library("lpSolve")
```

```
## Warning: package 'lpSolve' was built under R version 3.3.2
```

```
f.obj <- c(1, 0, 0)

row1 = c(1, 1, 1)
row2 = c(1, -1, -1)
row3 = c(1, 2.3, 1)
row4 = c(1, -2.3, -1)
row5 = c(1, 3.7, 1)
row6 = c(1, -3.7, -1)
row7 = c(1, 4.2, 1)
row8 = c(1, -4.2, -1)
row9 = c(1, 6.1, 1)
row10 = c(1, -6.1, -1)
row11 = c(1, 7, 1)
row12 = c(1, -7, -1)

f.con <- rbind(row1, row2, row3, row4, row5, row6, row7, row8, row9, row10, row11, row12)
f.dir <- rep(">=", 12)
f.rhs <- c(3.6, -3.6, 3, -3, 3.2, -3.2, 5.1, -5.1, 5.3, -5.3, 6.8, -6.8)

lp ("min", f.obj, f.con, f.dir, f.rhs)$solution
```

```
## [1] 0.9200000 0.5333333 2.1466667
```