Homework Week 3

Ben Arancibia

September 12, 2015

113

2) The following table gives the elongation e in inches per inch (in./in.) for a given stress S on a streel wire measured in pounds per square inch (lb/in^2). Test the model $e = c1S$ by potting the data. Estimate $c1$ graphically.

```
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)
S <- S * 10^-3
e <- e * 10^5

c <- e / S

df <- data.frame(S, e, c)
knitr::kable(df)
```

S	e	c
0.005	0	0
0.010	1900000	190000000
0.020	5700000	285000000
0.030	9400000	313333333
0.040	13400000	335000000
0.050	17300000	346000000
0.060	21600000	360000000
0.070	25600000	365714286
0.080	29700000	371250000
0.090	34300000	381111111
0.100	39000000	390000000

Now estimate the model where c1 is the median and plot the results.

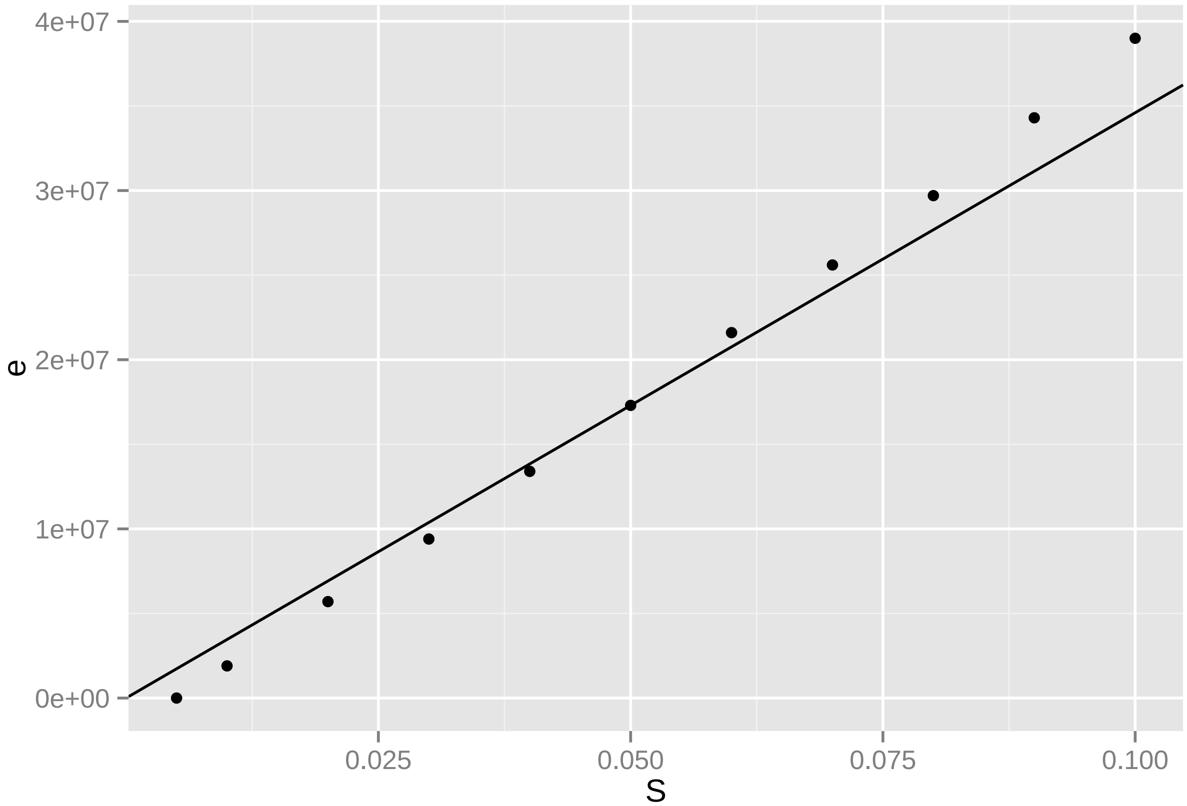
```
c1 <- median(c)
c1
```

```
## [1] 3.46e+08
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.1.3
```

```r
f1 <-ggplot(data=df) + geom_point(aes(x=S, y=e)) + geom_abline(intercept=0, slope=c1)
f1
```

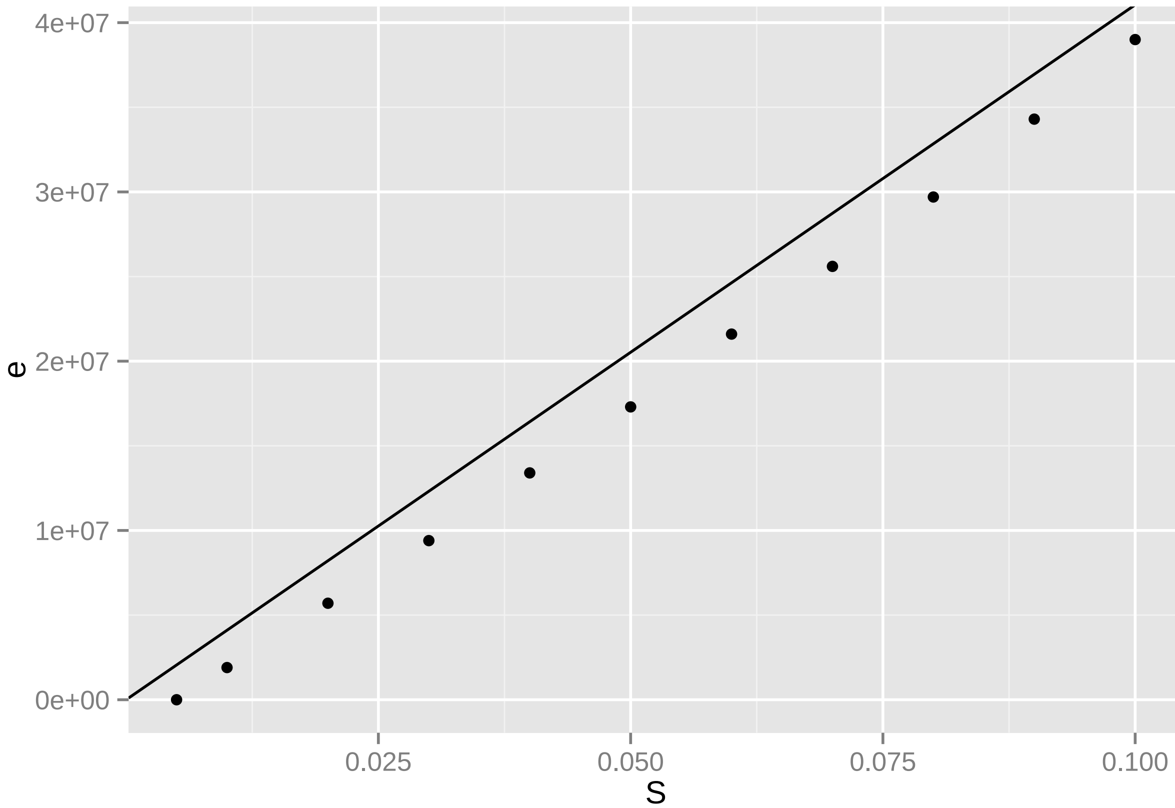


Estimate the model using slow and plot the results.

```r
c1 <- (e[length(e)] - e[1]) / (S[length(S)] - S[1])
c1
```

```
## [1] 410526316
```

```r
f2 <-ggplot(data=df) + geom_point(aes(x=S, y=e)) + geom_abline(intercept=0, slope=c1)
f2
```



Median approach is a little better than purely by the slow approach. The best approach would be a mixture of slope plus intercept.

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.

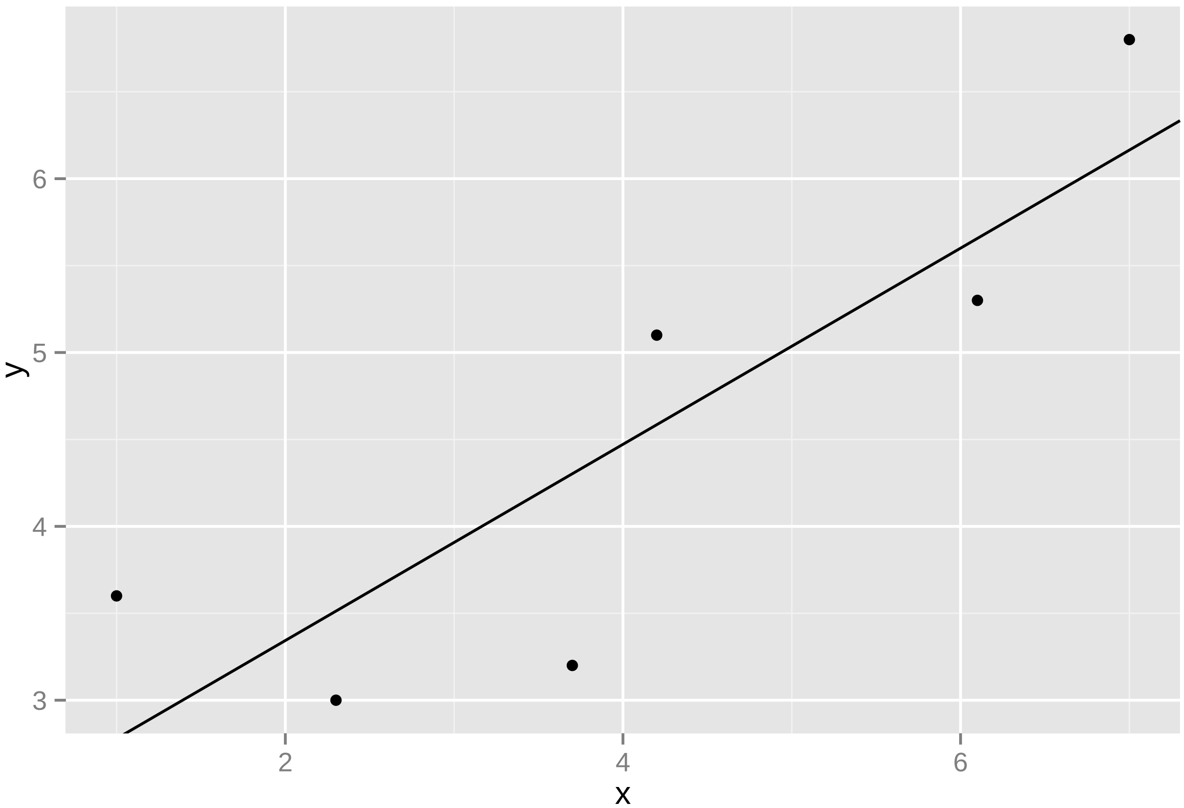
```
x <- c(1, 2.3, 3.7, 4.2, 6.1, 7)
y <- c(3.6, 3, 3.2, 5.1, 5.3, 6.8)

df <- data.frame(x, y)

lm <- lm(y ~ x, df)
lm$coefficients
```

```
## (Intercept)           x
##   2.2148534   0.5642337
```

```
f3 <- ggplot(data = df) + geom_point(aes(x = x, y = y)) + geom_abline(intercept = 2.214853,
    slope = 0.5642337)
f3
```



127

10) Fit the data with the models given using least squares. Data for planets. Fit the model $y = ax^(3/2)$

```
body <- c("Mercury", "Venus", "Earth", "Mars",
          "Jupiter", "Saturn", "Uranus", "Neptune")
period <- c(7.6 * 10^6, 1.94 * 10^7, 3.16 * 10^7,
            5.94 * 10^7, 3.74 * 10^8, 9.35 * 10^8,
            2.64 * 10^9, 5.22 * 10^9)
distance <- c(5.79 * 10^10, 1.08 * 10^11, 1.5 * 10^11,
              2.28 * 10^11, 7.79 * 10^11, 1.43 * 10^12,
              2.87 * 10^12, 4.5 * 10^12)
df <- data.frame(body, period, distance)
df
```

```
##      body   period distance
## 1 Mercury 7.60e+06 5.79e+10
## 2   Venus 1.94e+07 1.08e+11
## 3   Earth 3.16e+07 1.50e+11
## 4    Mars 5.94e+07 2.28e+11
## 5 Jupiter 3.74e+08 7.79e+11
## 6  Saturn 9.35e+08 1.43e+12
## 7  Uranus 2.64e+09 2.87e+12
## 8 Neptune 5.22e+09 4.50e+12
```

Least-squares equation for power curve where $n = 3/2$, x=period and y=distance.

```r
n <- 3/2

model <- function (x, y, n)
{
  numerator <- sum(y * x^n)
  denominator <- sum(x^(2*n))
  print(numerator)
  print(denominator)
  result <- numerator / denominator
  return (result)
}

a <- model(df$period, df$distance, n)
```

```
## [1] 2.133105e+27
## [1] 1.615064e+29
```

```r
a
```

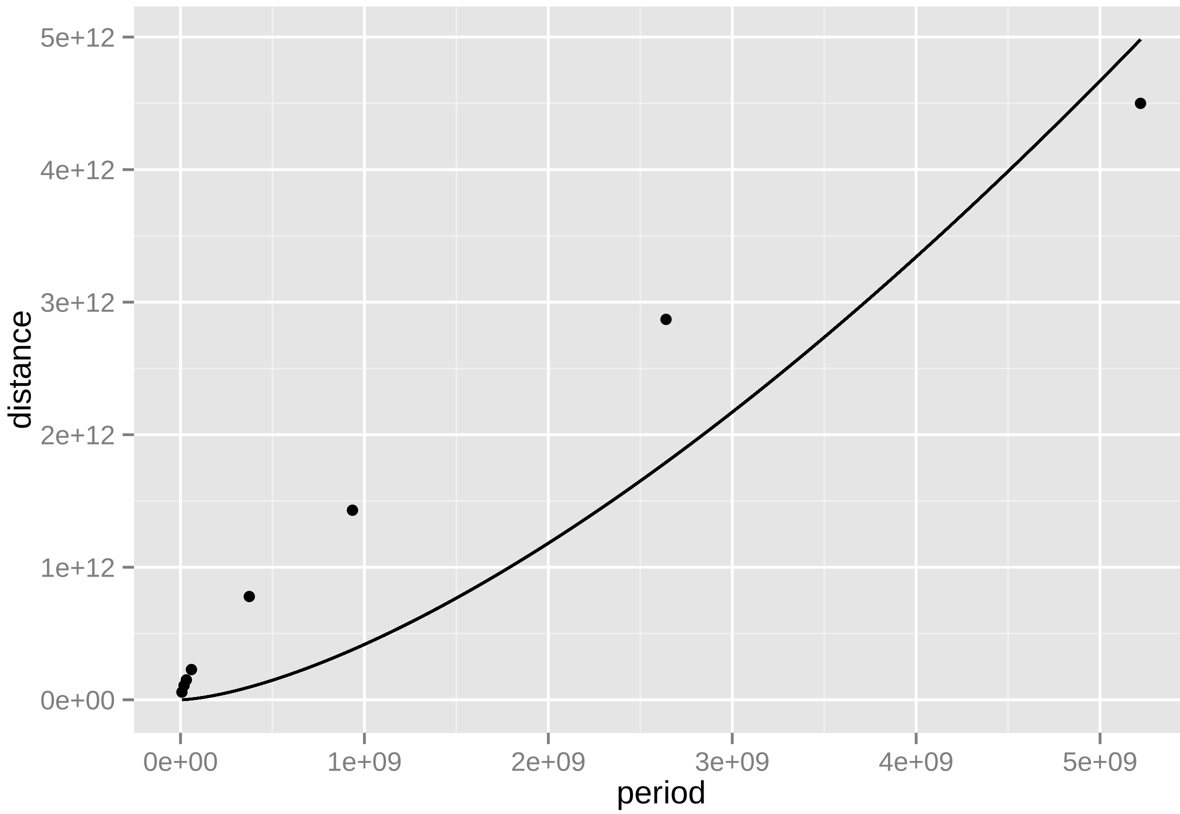
```
## [1] 0.01320756
```

```r
planet <- function(a, x, n)
{
  yestimate <- a * (x^(n))
  return(yestimate)
}

x_vals <- seq(min(df$period), max(df$period), by=100000)
y_mest <- planet(a, x_vals, n)
dfplanet<- data.frame(x_vals, y_mest)
```

Original data points and the model plotted.

```r
f4 <- ggplot(df) + geom_point(aes(x=period, y=distance)) + geom_line(data=dfplanet, aes(x=x_vals, y=y_me
f4
```



Squared deviations may be minimized, but the model does not do a good job representing the data.