Homework 3 – Due March 10

Name: Navya Mittal netID: navya_0215 Collaborated with:

Your homework **must be submitted in Word or PDF format, created by calling "Knit Word" or "Knit PDF" from RStudio on your R Markdown document. Submission in other formats may receive a grade of 0**. Your responses must be supported by both textual explanations and the code you generate to produce your result. Note that all R code used to produce your results must be shown in your knitted file.

We are working with the World Top Incomes Database (wtid-report.csv), and the Pareto distribution, as in the lab 5. We also continue to practice working with data frames, manipulating data from one format to another, and writing functions to automate repetitive tasks.

We saw in the lab 5 that if the upper tail of the income distribution followed a perfect Pareto distribution, then

$$\left(\frac{P99}{P99.9}\right)^{-a+1} = 10 \qquad (\star) \tag{1}$$

$$\left(\frac{P99.5}{P99.9}\right)^{-a+1} = 5 \qquad (\star\star) \tag{2}$$

$$\left(\frac{P99}{P99.5}\right)^{-a+1} = 2 \qquad (\star\star\star) \tag{3}$$

We could estimate the Pareto exponent by solving any one of these equations for a; in the lab we used

$$a = 1 - \frac{\log 10}{\log\left(P99/P99.9\right)}\,, \qquad (*) \tag{4}$$

Because of measurement error and sampling noise, we can't find one value of a which will satisfy all three equations $(\star)$–$(\star\star\star)$. Generally, trying to make all three equations come close to balancing gives a better estimate of a than just solving one of them. (This is analogous to finding the slope and intercept of a regression line by trying to come close to all the points in a scatterplot, and not just running a line through two of them.)

1. We estimate a by minimizing

$$\left(\left(\frac{P99}{P99.9}\right)^{-a+1} - 10\right)^2 + \left(\left(\frac{P99.5}{P99.9}\right)^{-a+1} - 5\right)^2 + \left(\left(\frac{P99}{P99.5}\right)^{-a+1} - 2\right)^2$$

Write a function, `percentile_ratio_discrepancies`, which takes as inputs `a`, `P99`, `P99.5` and `P99.9`, and returns the value of the expression above. Check that when `a=2`, `P99=1e6`, `P99.5=2e6` and `P99.9=1e7`, your function returns `0`.

```r
percentile_ratio_discrepancies <- function(a, x, y, z){ #x=P99, y=P99.5 z=P99.9
  ans = ((x/z)^(-a+1) - 10)^2 + ((y/z)^(-a+1) -5)^2 + ((x/y)^(-a+1) -2)^2
  return (ans)
}
percentile_ratio_discrepancies(2,1e6,2e6,1e7)
```

```
## [1] 0
```

2. Write a function, `exponent.multi_ratios_est`, which takes as inputs `P99`, `P99.5`, `P99.9`, and estimates `a`. It should minimize your `percentile_ratio_discrepancies` function. The starting value for the minimization should come from (∗). Check that when `P99=1e6`, `P99.5=2e6` and `P99.9=1e7`, your function returns an `a` of 2.

Hint: Use the built-in nonlinear optimization function `nlm()`; check its documentation to see how to pass additional arguments of `percentile_ratio_discrepancies` to `nlm`. Look at the examples on the help page. You should be passing to `nlm`: f = `percentile_ratio_discrepancies`, p = the value from (∗), and `P99`, `P99.5`, `P99.9`. Keep the default arguments beyond that to `nlm`.

```
?nlm
exponent.multi_ratios_est <- function(x, y, z){ #x=P99, y=P99.5, z=P99.9
  #startvar = 1 - log(10)/(log (x/y))
  a = nlm(f=percentile_ratio_discrepancies, p=(1 - log(10)/(log (x/z))), x,y,z)
  return (a$estimate)
}

exponent.multi_ratios_est(x=1e6, y=2e6, z=1e7)
```

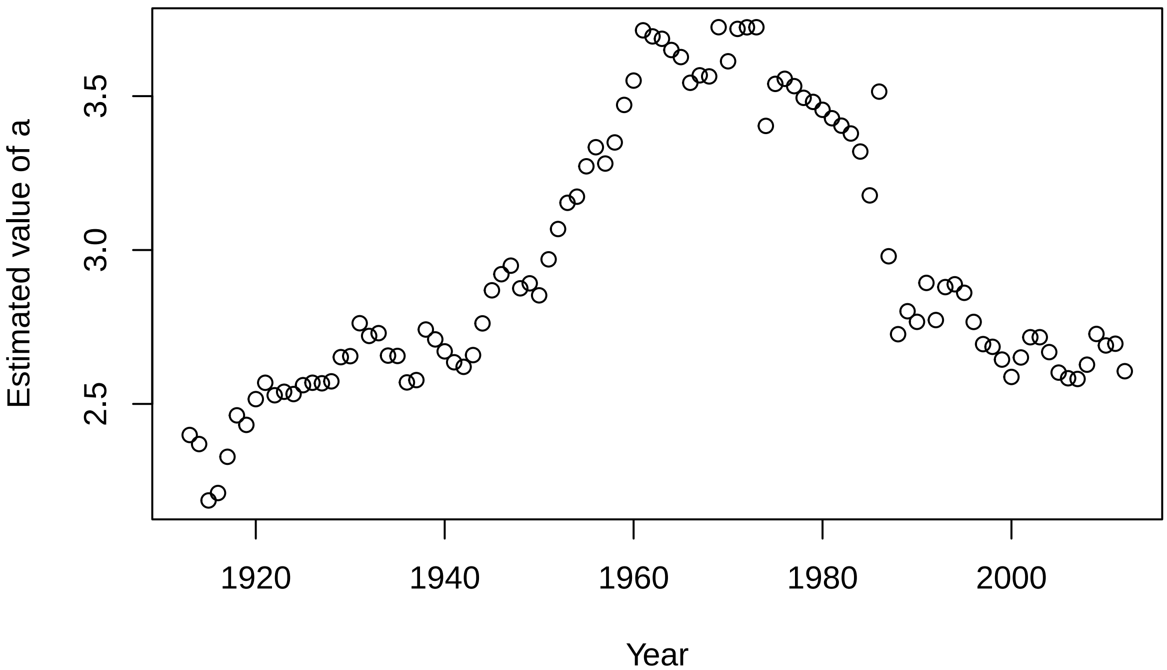
```
## [1] 2
```

3. Write a function which uses `exponent.multi_ratios_est` to estimate a for the US for every year from 1913 to 2012. (There are many ways you could do this, including loops.) Plot the estimates; make sure the labels of the plot are appropriate.

```
data = read.csv("wtid-report.csv", header=TRUE)

data_new = data[-c(1,3,4,8)]
head(data_new)
```

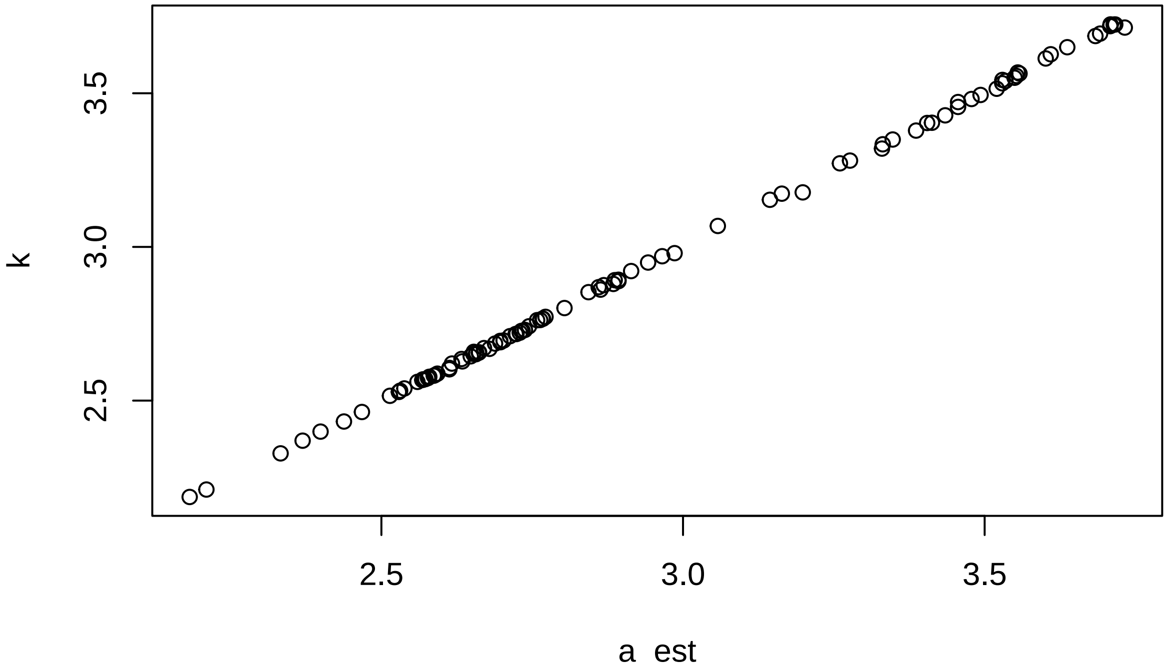
```
##   Year P99.income.threshold P99.5.income.threshold P99.9.income.threshold
## 1 1913             80087.90               131337.2               415206.4
## 2 1914             74012.72               122935.9               397671.6
## 3 1915             62392.24               118717.4               437522.8
## 4 1916             74869.18               133777.1               502094.2
## 5 1917             92341.21               149697.9               519558.7
## 6 1918             92221.06               143219.7               442731.1
```

```
est = c()

exponent.new <- function(xvec, yvec, zvec){ #x=P99, y=P99.5, z=P99.9
  output = array(dim=length(xvec))
  for (i in 1:length(xvec)){
    output[i] = exponent.multi_ratios_est(xvec[i], yvec[i], zvec[i])

  }
  return (output)
}

?with

k = with(data_new,exponent.new(P99.income.threshold, P99.5.income.threshold, P99.9.income.threshold))

plot(data_new$Year, k , xlab = 'Year', ylab = 'Estimated value of a')
```



4. Use (∗) to estimate a for the US for every year, as in the lab. Make a scatter-plot of these estimates against those from problem 3. If they are identical or completely independent, something is wrong with at least one part of your code. Otherwise, can you say anything about how the two estimates compare?

```
#a = 1 - \frac{\log{10}}{\log{(P99/P99.9)}} ~,\qquad (*)

a_est = with(data=data_new, (1-log (10)/(log (data_new$P99.income.threshold/data_new$P99.9.income.thres

plot(a_est, k)
```



The two estimates are linear.

5. Fit a regression with

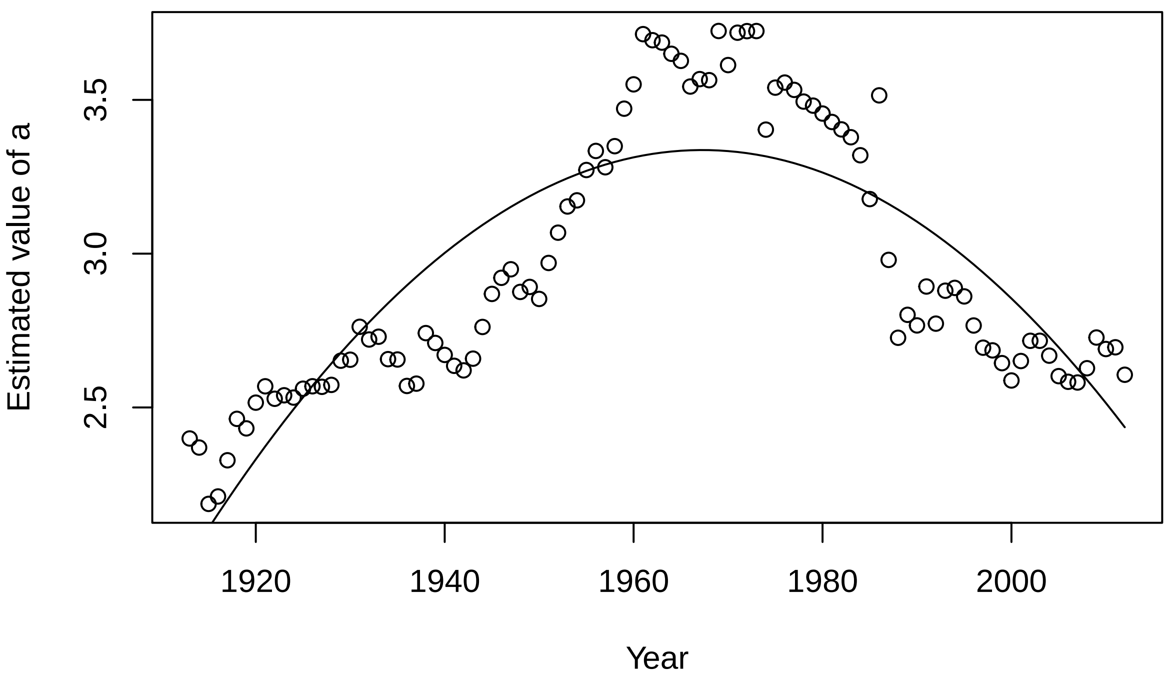
- response: estimated a in problem 3
- two covariates: year and square of year

The regression should contain an intercept term. Produce a scatter plot in problem 3 again and then overlay it with a line representing the fitted values of the regression.
Hint: `lm(y ~  x +I(x^2))` will regress y on both x and x^2.

```
?I()
b = lm(k ~ data_new$Year + I((data_new$Year)^2))
summary(b)
```

```
##
## Call:
## lm(formula = k ~ data_new$Year + I((data_new$Year)^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.42906 -0.19960 -0.00509  0.19263  0.40173
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -1.739e+03  1.206e+02  -14.42   <2e-16 ***
## data_new$Year         1.771e+00  1.229e-01   14.41   <2e-16 ***
## I((data_new$Year)^2) -4.502e-04  3.132e-05  -14.38   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2334 on 97 degrees of freedom
## Multiple R-squared:  0.7076, Adjusted R-squared:  0.7016
## F-statistic: 117.4 on 2 and 97 DF,  p-value: < 2.2e-16
```

```
plot(data_new$Year, k , xlab = 'Year', ylab = 'Estimated value of a')
par(new=TRUE)
lines(data_new$Year, fitted(b))
```



```
?lines
```