Homework #3

Fan Yang (fy2232)

October 24, 2017

Part 1: Estimating a on US data

Recall from the lab, we first get the data and function in lab.

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.2
```

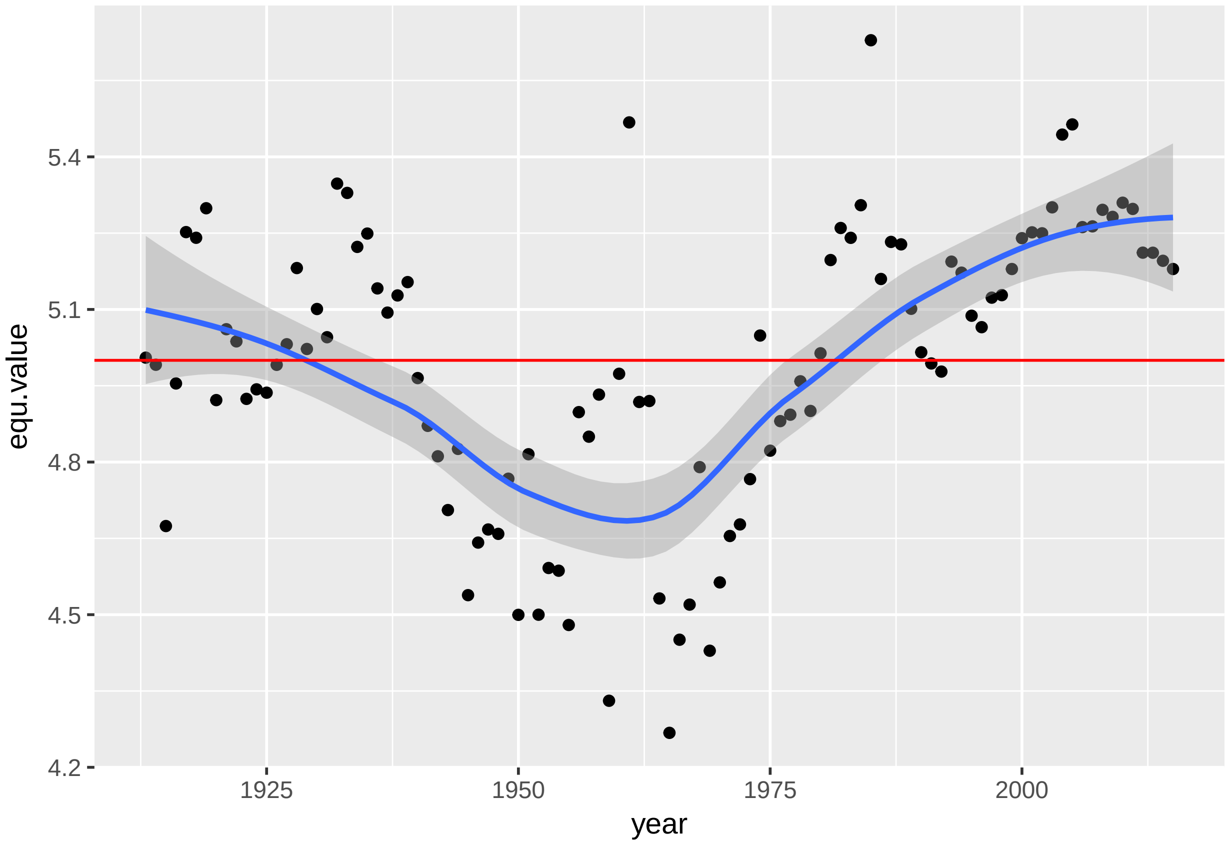
```
homework <- read.csv("wtid-homework.csv", header = TRUE)
report <- read.csv("wtid-report.csv", header = TRUE)
report <- data.frame(year=report$Year,P99=report$P99.income.threshold,
                   P99.5=report$P99.5.income.threshold,
                   P99.9=report$P99.9.income.threshold)
exponent.est_ratio <- function(P99,P99.9){
  return (1 - log(10)/log(P99/P99.9))
}
```

i.

```
lefthand.equ <- function(P99.5,P99.9,a){
  return ((P99.5/P99.9)^(1-a))
}
a <- exponent.est_ratio(report$P99,report$P99.9)
equ.value <- lefthand.equ(report$P99.5,report$P99.9,a)

report2 <- cbind(report, equ.value)
ggplot(data = report2) +
  geom_point(mapping = aes(x = year, y = equ.value)) +
  geom_smooth(mapping = aes(x = year, y = equ.value)) +
  geom_abline(intercept = 5, slope = 0,col = "red")
```

```
## `geom_smooth()` using method = 'loess'
```



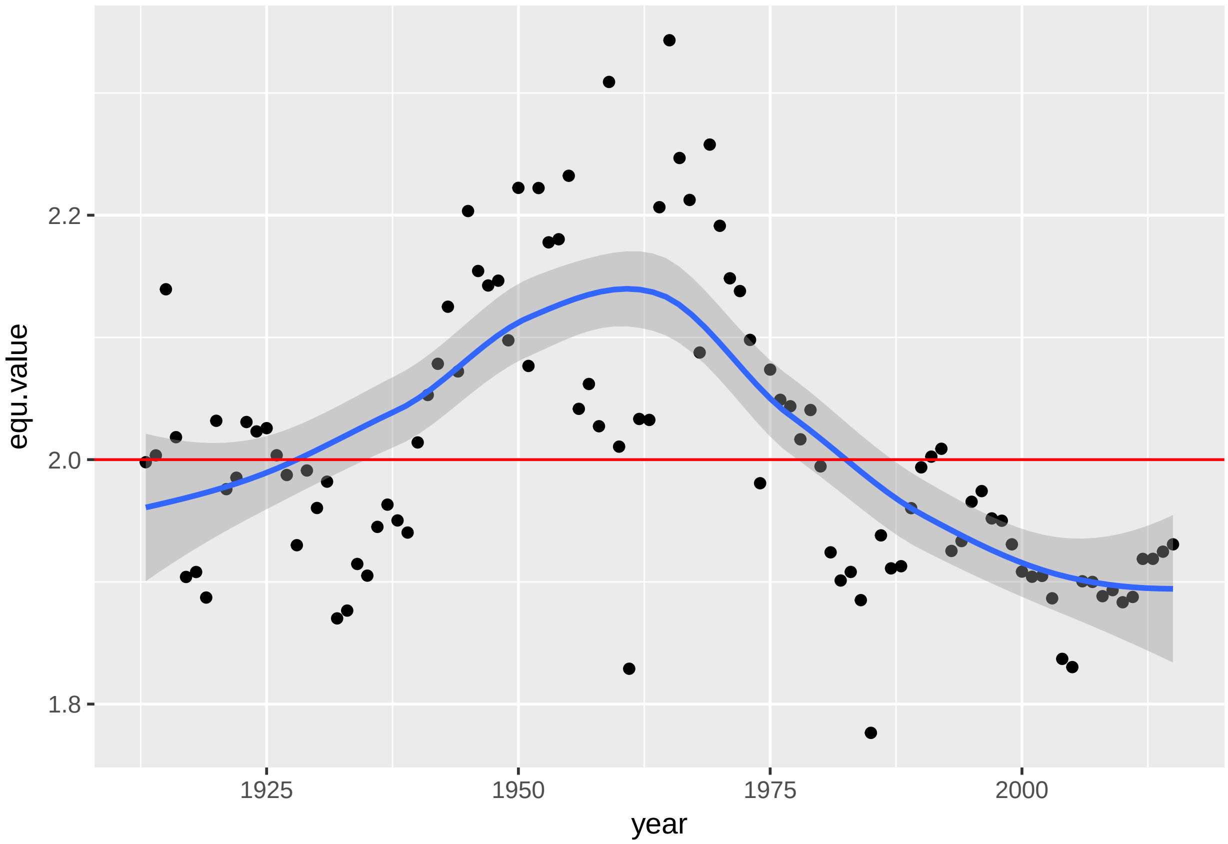
Most of the points lay between 4.6 and 5.4. In spite of some deviation, this is a good fit.

ii.

```
lefthand.equ2 <- function(P99,P99.5,a){
  return ((P99/P99.5)^(1-a))
}
a <- exponent.est_ratio(report$P99,report$P99.9)
equ.value <- lefthand.equ2(report$P99,report$P99.5,a)

report3 <- cbind(report, equ.value)
ggplot(data = report3) +
  geom_point(mapping = aes(x = year, y = equ.value)) +
  geom_smooth(mapping = aes(x = year, y = equ.value)) +
  geom_abline(intercept = 2, slope = 0,col = "red")
```

```
## `geom_smooth()` using method = 'loess'
```



Compared with the previous fit, this one seems better. Most of the points lay between 1.8 and 2.2, which has a smaller deviation than the previous one.

iii.

```
percentile_ratio_discrepancies <- function(P99,P99.5,P99.9,a){
  part1 = ((P99/P99.9)^(1-a) - 10) ^ 2
  part2 = ((P99.5/P99.9)^(1-a) - 5) ^ 2
  part3 = ((P99/P99.5)^(1-a) - 2) ^ 2
  return (part1+part2+part3)
}
percentile_ratio_discrepancies(1e6,2e6,1e7,2)
```

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

iv.

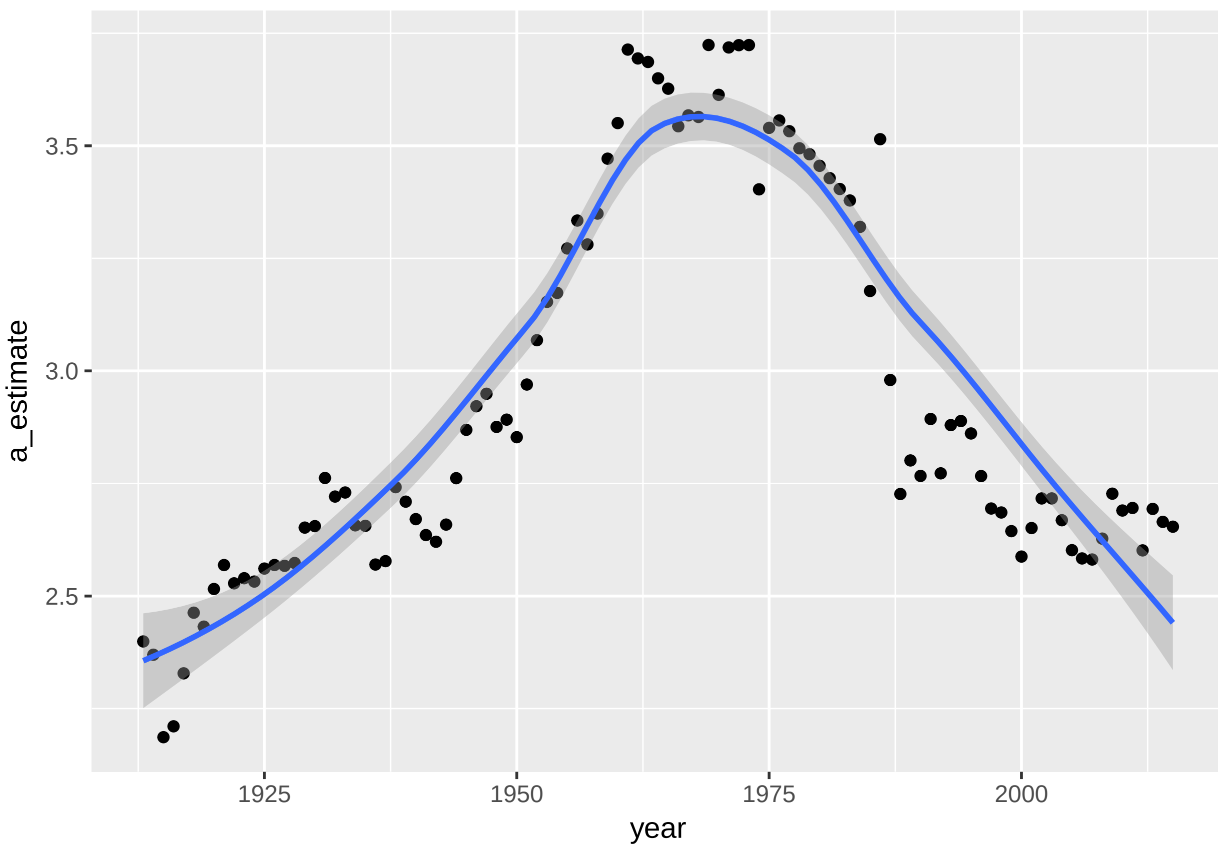
```
exponent.multi_ratios_est <- function(P99,P99.5,P99.9){
  a <- 1 - log(10)/log(P99/P99.9)
  f <- function(a,vector)
    {percentile_ratio_discrepancies(vector[1],vector[2],vector[3],a)}
  a <- 1 - log(10)/log(P99/P99.9)
  vec <- c(P99,P99.5,P99.9)
  return (nlm(f,a,vec)$estimate)
}
exponent.multi_ratios_est(1e6,2e6,1e7)
```

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

v.

```
###PASS###########################
exponent.every_ratios_est <- function(P99,P99.5,P99.9){
  f <- function(vector)
    {exponent.multi_ratios_est(vector[1],vector[2],vector[3])}
  data <- as.matrix(cbind(P99,P99.5,P99.9))
  a_estimate <- apply(data,1,f)
  return (a_estimate)
}
a_estimate <- exponent.every_ratios_est(report$P99,report$P99.5,report$P99.9)
report4 <- data.frame(report,a_estimate)
ggplot(data = report4) +
  geom_point(mapping = aes(x = year, y = a_estimate)) +
  geom_smooth(mapping = aes(x = year, y = a_estimate))
```

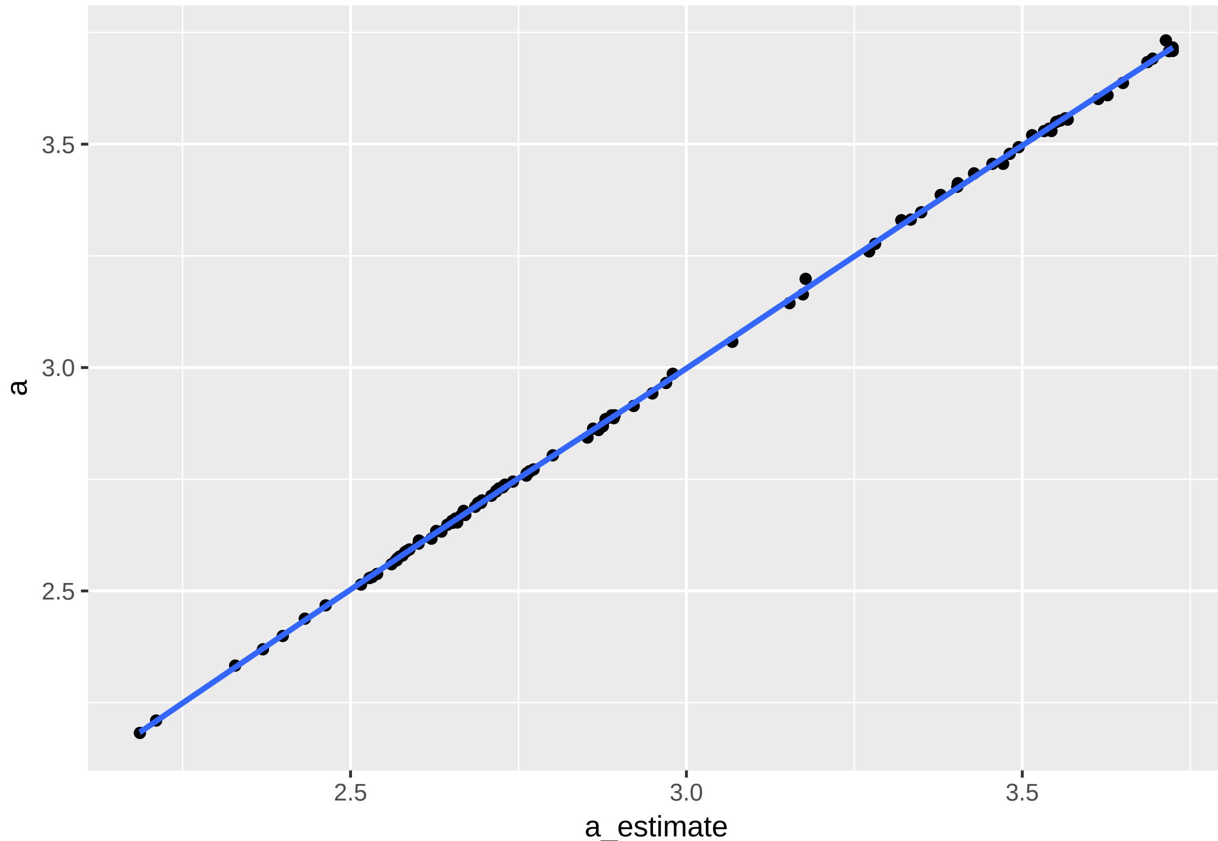
```
## `geom_smooth()` using method = 'loess'
```



vi.

```
a <- exponent.est_ratio(report$P99,report$P99.9)
datavi = data.frame(a,a_estimate)
ggplot(data = datavi) +
  geom_point(mapping = aes(x = a_estimate, y = a)) +
  geom_smooth(mapping = aes(x = a_estimate, y = a))
```

```
## `geom_smooth()` using method = 'loess'
```



The two estimates are very similar but not identical, which leads the conclusion that our estimates fit good.

Part 2: Data for Other Countries

vii.

```
homework <- na.omit(homework)
a_country<-list()
for (i in levels(homework$Country)){
  temp=homework[homework$Country==i,]
  a_country[[i]]=exponent.every_ratios_est(temp$P99,temp$P99.5,temp$P99.9)
}
```

viii.

```
##method 1
# iter=1
# g<-ggplot()
# for (i in levels(homework$Country)){
#   g<-g+ geom_point(mapping = aes(x = homework$Year[homework$Country==i] ,
#                                  y = a_country[[i]]), col = iter) +
#     geom_smooth(mapping = aes(x = homework$Year[homework$Country==i] ,
#                                  y = a_country[[i]]), col = iter) +
#     labs(title = i, x = "year", y = "estimate")
#   iter <- iter + 1
#   print(g)
```

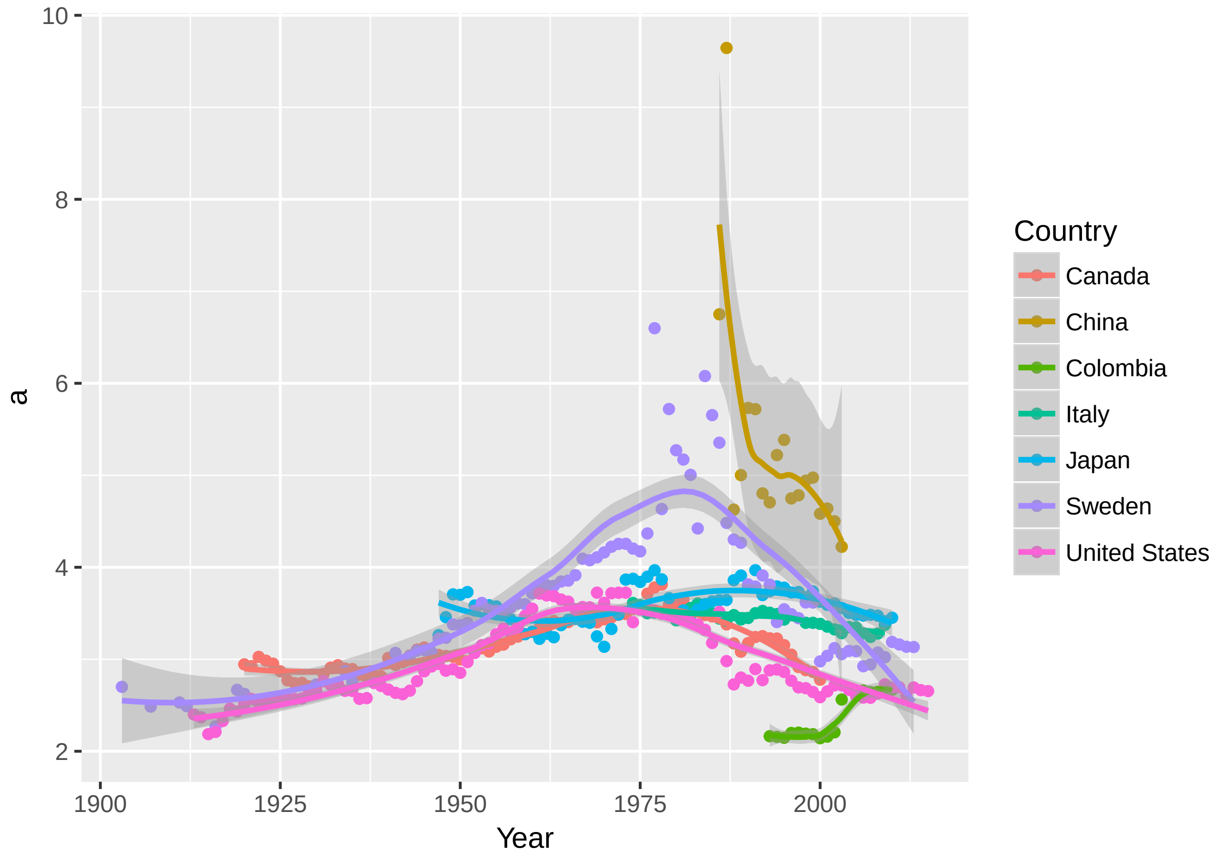
```r
# }

##method 2
newdata <- data.frame(a=unlist(a_country),Country=homework$Country,Year=homework$Year)
names(newdata)
```

```
## [1] "a"       "Country" "Year"
```

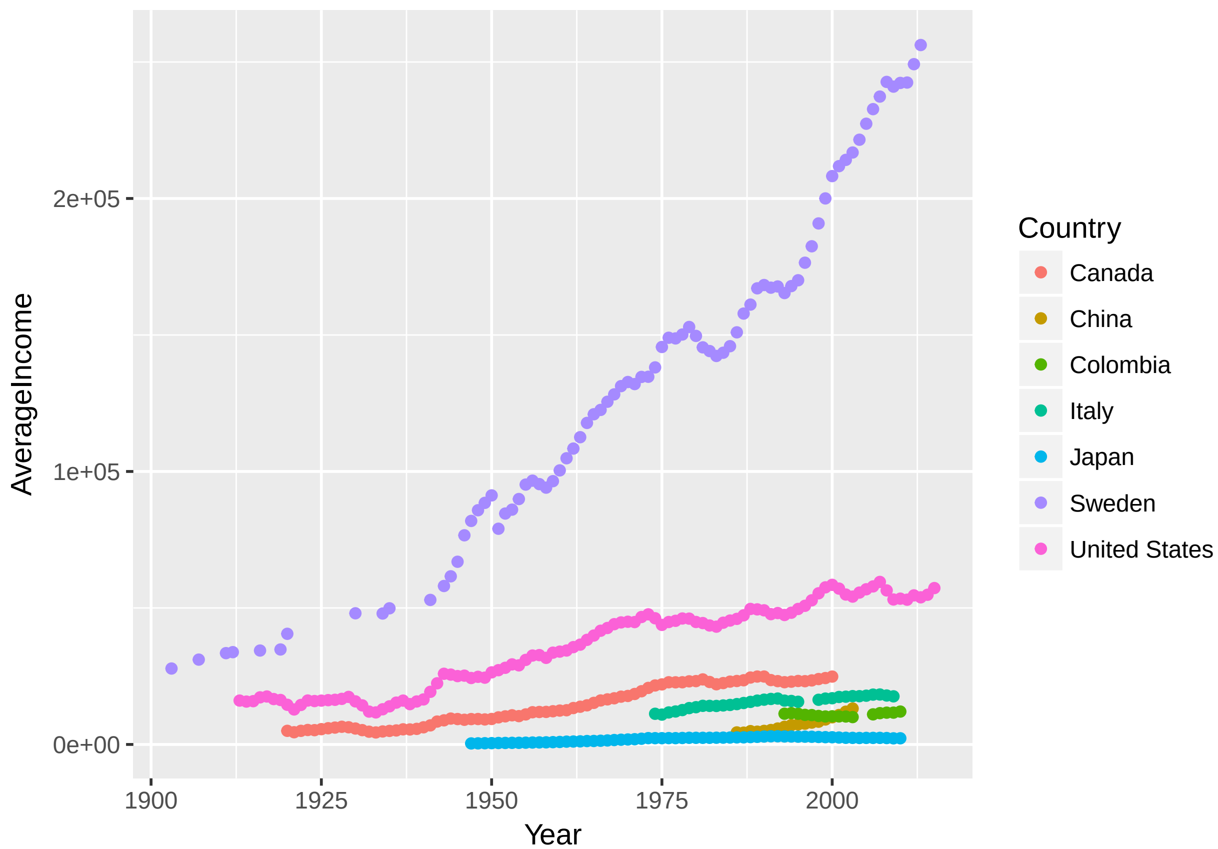
```r
ggplot(data = newdata) +
  geom_point(mapping = aes(x=Year, y=a, color=Country))+
  geom_smooth(mapping = aes(x=Year, y=a, color=Country))
```

```
## `geom_smooth()` using method = 'loess'
```



ix.

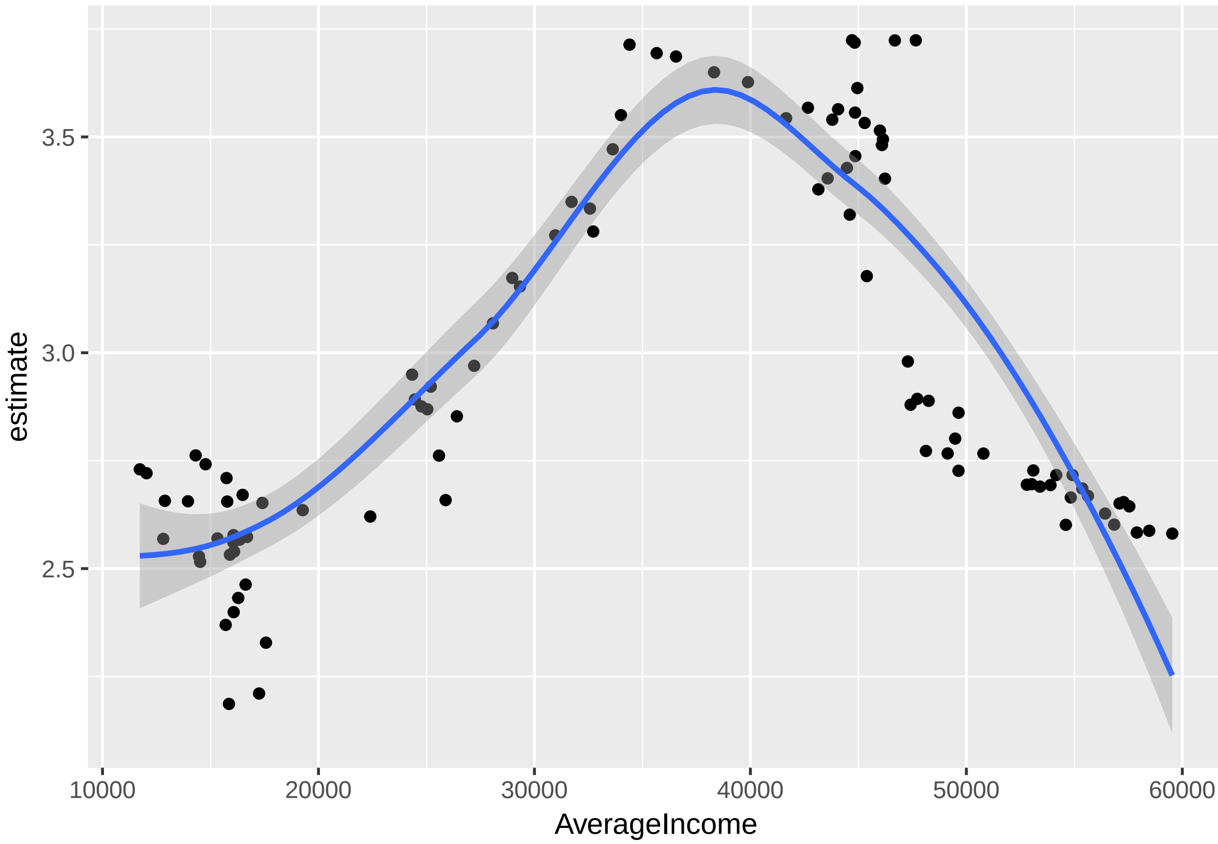
```r
ggplot(data = homework) +
  geom_point(mapping = aes(x=Year, y=AverageIncome, color=Country))
```



x.

```
datax <- data.frame(estimate=a_country$`United States`,
                    AverageIncome=homework$AverageIncome[homework$Country=="United States"])
ggplot(data = datax) +
  geom_point(mapping = aes(x=AverageIncome, y=estimate)) +
  geom_smooth(mapping = aes(x=AverageIncome, y=estimate))
```

```
## `geom_smooth()` using method = 'loess'
```



At the very beginning of economic growth, which indicates low average income, the estimated exponent is small and the income inequality is high. As economic growing, average income grows and estimated exponent reaches a high value, which also implies less income inequality. Finally, when economic grows to some high extent, estimated exponent turns down and represents a high income inequality.

xi.

```
datax <- data.frame(estimate=a_country$`United States`,
                    AverageIncome=homework$AverageIncome[homework$Country=="United States"])
modelxi <- lm(estimate~AverageIncome+I(AverageIncome^2),data=datax)
summary(modelxi)
```

```
##
## Call:
## lm(formula = estimate ~ AverageIncome + I(AverageIncome^2), data = datax)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.50724 -0.18364 -0.02531  0.18689  0.54918
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         8.230e-01  1.515e-01   5.432 3.93e-07 ***
## AverageIncome       1.394e-04  1.015e-05  13.740  < 2e-16 ***
## I(AverageIncome^2) -1.891e-09  1.451e-10 -13.027  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.2466 on 100 degrees of freedom
## Multiple R-squared:  0.6679, Adjusted R-squared:  0.6612
## F-statistic: 100.5 on 2 and 100 DF,  p-value: < 2.2e-16
```

The regression function is $y = 0.823 + 1.394 \times 10^{-4}x - 1.891 \times 10^{-9}x^2$ The symmetry axis of function is $(1.394 * 10^{-4})/(2 * 1.891 * 10^{-9}) = 36858.8$ and is vary similar with the plot in part (x). What's more, the p-value of F-test less than 2.2e-16, which indicates our model is a good fit.

xii.

```
Kuznet <- function(estimate, AverageIncome){
  model_sep <- lm(estimate~AverageIncome+I(AverageIncome^2))
  return(model_sep$coefficients[3])
}
dataxii <- data.frame(estimate=unlist(a_country),
                      AverageIncome=homework$AverageIncome,
                      Country = homework$Country)
x2coefficient<-c()
for (i in levels(dataxii$Country)){
  data_temp <- dataxii[dataxii$Country==i,]
  x2coefficient[i] <- Kuznet(data_temp$estimate,data_temp$AverageIncome)
}
x2coefficient
```

```
##        Canada         China      Colombia         Italy         Japan
## -3.360837e-09  5.257536e-08  2.867133e-07 -6.591048e-09  1.889447e-07
##        Sweden United States
## -1.496762e-10 -1.890556e-09
```

So Canada, Italy, Sweden and United States compatible with the hypothesis.