Hw9_Ex_2

Manuel Alejandro Garcia Acosta

11/13/2019

We will use the following packages for this exercise.

```
library('KernSmooth')
```

```
## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009
```

```
library('ggplot2')
```

Homework 9

Exercise 2

Part (d)

First I downloaded the data from (https://books.google.com/ngrams). My query was for the words 'fake news' between the years 1900 and 2000 with smoothing of 0. I got the following data after viewing the Page Source. They are the values for the frequency each year.

```
v <- c(0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 6.75992761767219e-10,
       2.6898412386344717e-09, 0.0, 0.0, 7.05148273105749e-10, 0.0, 0.0,
       2.15676076997795e-09, 7.86069154212754e-10, 1.0530158078836394e-08,
       8.05010003102069e-10, 2.750184524558108e-09, 6.70825084370108e-09, 0.0,
       7.831103543409768e-10, 3.3711879954267943e-09, 0.0, 1.3376082463878447e-08,
       5.1371777942677e-09, 2.7137407876409725e-09, 5.488800525199622e-09,
       2.3588215825043335e-09, 0.0, 2.297882772950288e-09, 4.0264418466051666e-09,
       4.509577600231296e-09, 4.163419387026579e-09, 0.0, 4.119892427212335e-09,
       6.3566565344785886e-09, 8.508320270550485e-09, 1.964633256079651e-08,
       3.816146953994348e-09, 1.6158185900394528e-09, 1.7579219857566386e-08,
       9.113046317565932e-09, 0.0, 4.693678778977528e-09, 4.117363783251449e-09,
       5.547278192352678e-09, 2.7532935931162683e-09, 2.5676545334363254e-09,
       4.069465209255441e-09, 1.1604386340735573e-09, 1.1896278406808847e-09,
       1.176486685849909e-09, 1.1991960757740117e-09, 1.7211456704302464e-09,
       2.1776846992338506e-09, 5.247626777560299e-10, 0.0, 0.0,
       2.3102715296374754e-09, 8.147825192317271e-10, 1.1145626643838114e-09,
       2.369873186580662e-09, 0.0, 1.9535804085535347e-09, 1.735336985220215e-09,
       1.927556558811716e-09, 7.308098570746324e-09, 7.218641240314128e-10,
       1.4529188963052775e-09, 1.4018888272460117e-09, 2.822707845240302e-09,
       2.0576063075594675e-09, 7.053182482508191e-10, 2.593494530245266e-09,
       2.323692571692959e-09, 8.988535471488035e-10, 1.546565875365502e-09,
       1.091439050249221e-09, 2.10411466028404e-09, 1.6525671941991504e-09,
       1.6463121976784123e-09, 1.968059271106881e-09, 1.3374278351463431e-09,
       1.0758712809533222e-09, 6.953358444583557e-10, 4.928406016624365e-10,
       1.2826857354042431e-09, 9.224092600845779e-10, 1.453305253917847e-09,
       2.1524351190294055e-09, 2.2081922956829203e-09, 1.503659197155116e-09,
       2.795319975490429e-09, 3.2015656792339087e-09, 2.530552434265587e-09,
```

```
       2.612969396409426e-09, 2.861999526260206e-09, 2.8343078994907955e-09,
       2.2857795656250346e-09, 4.338205350506996e-09)
```

After getting the frequencies I put them together with their corresponding year in a data frame.

```
years <- 1900:2000 # Create the vector of dates
Data <- data.frame(year=years,  Y=v) # Merge the information into a data frame
```

Next I just used the 'ggplot2' package to create a plot of the information I got.

```
plot <- ggplot(data = Data, aes( x = year, y=Y))+
  geom_line()+xlab('Year')+ylab('Frequency') # Create a plot with the data
```

In the code that follows I used the package 'KernSmooth'. First I used the function dpill() to select a bandwith for local linear regression.

NOTE: The dpill() function chooses the bandwith for local regression using the Gaussian kernel.

$$\frac{1}{\sqrt{2\pi\sigma}} e^{\frac{x^2}{2\sigma^2}}$$

```
h <- dpill(x= Data$year, y = Data$Y) # Select bandwith for local lin. regression
h # This is the bandwith
```

```
## [1] 3.760177
```

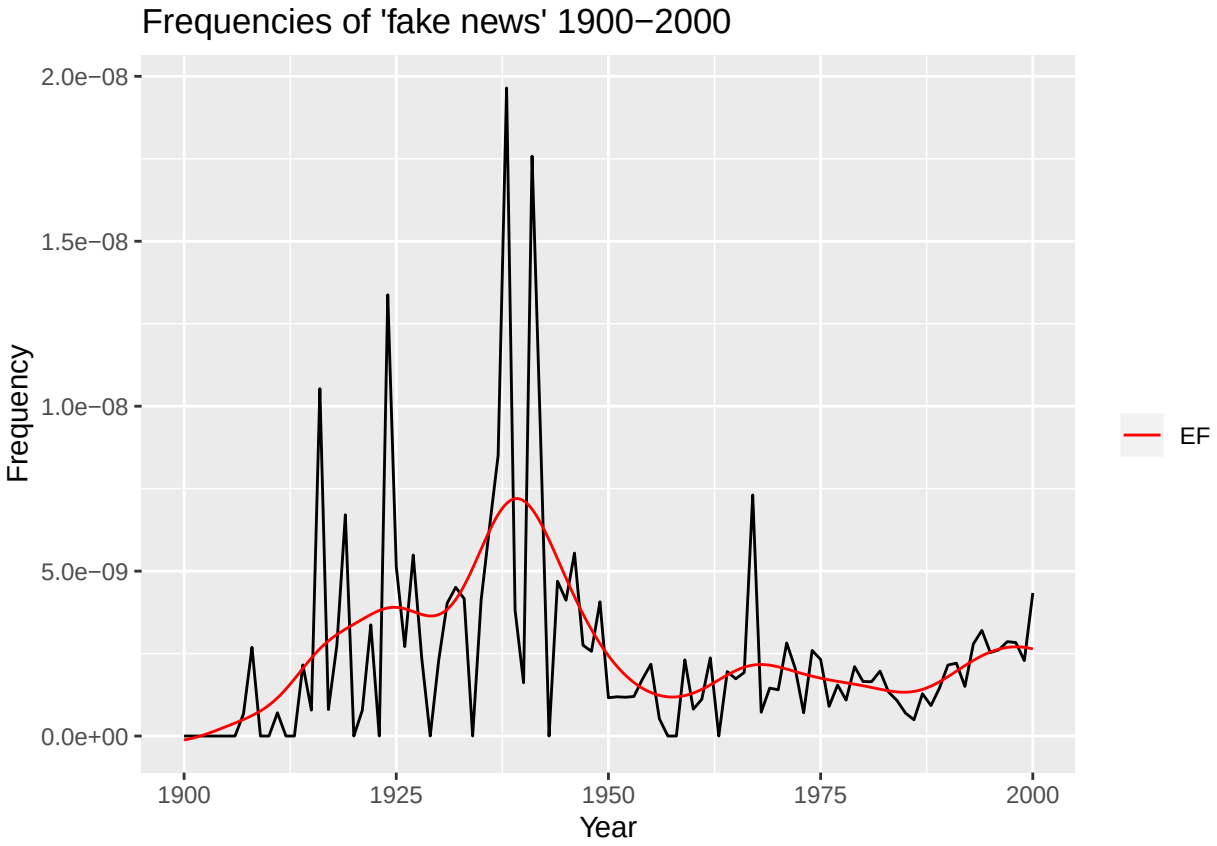
Next, I used the function locpoly() to estimate the regression function using polynomials of degree 1 and using the bandwith I found with the dpill() function.

NOTE: Since dpill() computes the bandwith with a Gaussian kernel, in the locpoly() function I also used the Gaussian kernel.

```
# Esimate the regression function
fit <- locpoly(x = Data$year, y = Data$Y, degree = 1,  bandwidth = h)
```

Finally, I plotted both the original data and the estimated function together.

```
plot+geom_line(data = data.frame(x= fit$x, y = fit$y), aes(x=x,y=y,
  color = 'red'))+ scale_color_manual(labels = 'EF', values = 'red')+
  labs(color = '')+ggtitle('Frequencies of \'fake news\' 1900-2000')
```



Where EF stands for 'Estimated Function'.