Assignment 5.

Sergiu Buciumas

Ch10Ex1

Source code:

data <- read.csv("/Users/sbuciuma/Desktop/School/summer/nonparametric\_statistics/Lecture11/Exercise10\_1.csv")

head(data)

length(data)

hist(data$Temperature)

summary(data)

hist(data$Temperature, freq = F)

#max.data <- max(data$Temperature)

lines(density(data$Temperature), col = "red")



with rectangular kernel

lines(density(data$Temperature,bw='sj'))

lines(density(data$Temperature,kernel='rectangular',bw='sj'))



Ch10Ex2

Ch10Ex2 <- read.csv("/Users/sbuciuma/Desktop/School/summer/nonparametric\_statistics/Lecture11/Exercise10\_2.csv")

head(Ch10Ex2)

length(Ch10Ex2)

#hist(Ch10Ex2$Heights)

summary(data)

#hist(Ch10Ex2$Heights, freq = F)

#max.data <- max(data$Temperature)

plot(density(Ch10Ex2$Heights), col = "red")

#plot(data$Temperature)

#plot(density(data$Temperature))

lines(density(Ch10Ex2$Heights,bw='sj'))

lines(density(Ch10Ex2$Heights,kernel='rectangular',bw='sj'))



with normal kernel and rectangular.



Ch10Ex3

Ch10Ex3 <- read.csv("/Users/sbuciuma/Desktop/School/summer/nonparametric\_statistics/Lecture11/Exercise10\_3.csv")

head(Ch10Ex3)

length(Ch10Ex3)

summary(Ch10Ex3)

plot(Ch10Ex3)

### Loess regression

# Plot the data

plot(Ch10Ex3$MGP, Ch10Ex3$SP, ylim=range( c(Ch10Ex3$MGP, Ch10Ex3$SP) ))

# Fit a loess regression of mpg on weight (in half tons)

fitlo <- loess(SP~MGP,data=Ch10Ex3)

# To make predictions at values in a range similar to the original data

XforPred <- seq(10,70, length=400)

# Calculate the predictions and the standard errors

predInfo <- predict(fitlo,XforPred, se=TRUE)

lines(XforPred, predInfo$fit, col='red')

lines(XforPred, predInfo$fit-2\*predInfo$se.fit,lty=2)

lines(XforPred, predInfo$fit+2\*predInfo$se.fit,lty=2)

#with kernel method

plot(Ch10Ex3$MGP, Ch10Ex3$SP, ylim=range( c(Ch10Ex3$MGP, Ch10Ex3$SP) ))

plot(Ch10Ex3$MGP, Ch10Ex3$SP, main="bandwidth=15")

lines(ksmooth(Ch10Ex3$MGP, Ch10Ex3$SP, "normal", bandwidth=15))

#plot(density(data$Temperature))

lines(density(cars$MGP,bw='sj'))

lines(density(cars$MGP,kernel='rectangular',bw='sj'))



with kernel:



or with distribution:

# second option for Ch10Ex10 using the distributions

cars <- read.csv("/Users/sbuciuma/Desktop/School/summer/nonparametric\_statistics/Lecture11/Exercise10\_3.csv")

plot(cars)

plx<-predict(loess( cars$SP~ cars$MGP), se=T)

lines(cars$MGP,plx$fit)

lines(cars$MGP,plx$fit - qt(0.975,plx$df)\*plx$se, lty=2)

lines(cars$MGP,plx$fit + qt(0.975,plx$df)\*plx$se, lty=2)

