Local Linear Regression

Assignment 2

Ignasi Mane, Antoni Company & Chiara Barbi

3/8/2019

Introduction

```
library(glmnet)
library(caret)
library(sm)
library(KernSmooth)
library(ggplot2)
library(gridExtra)
set.seed(1234)
data(aircraft)
```

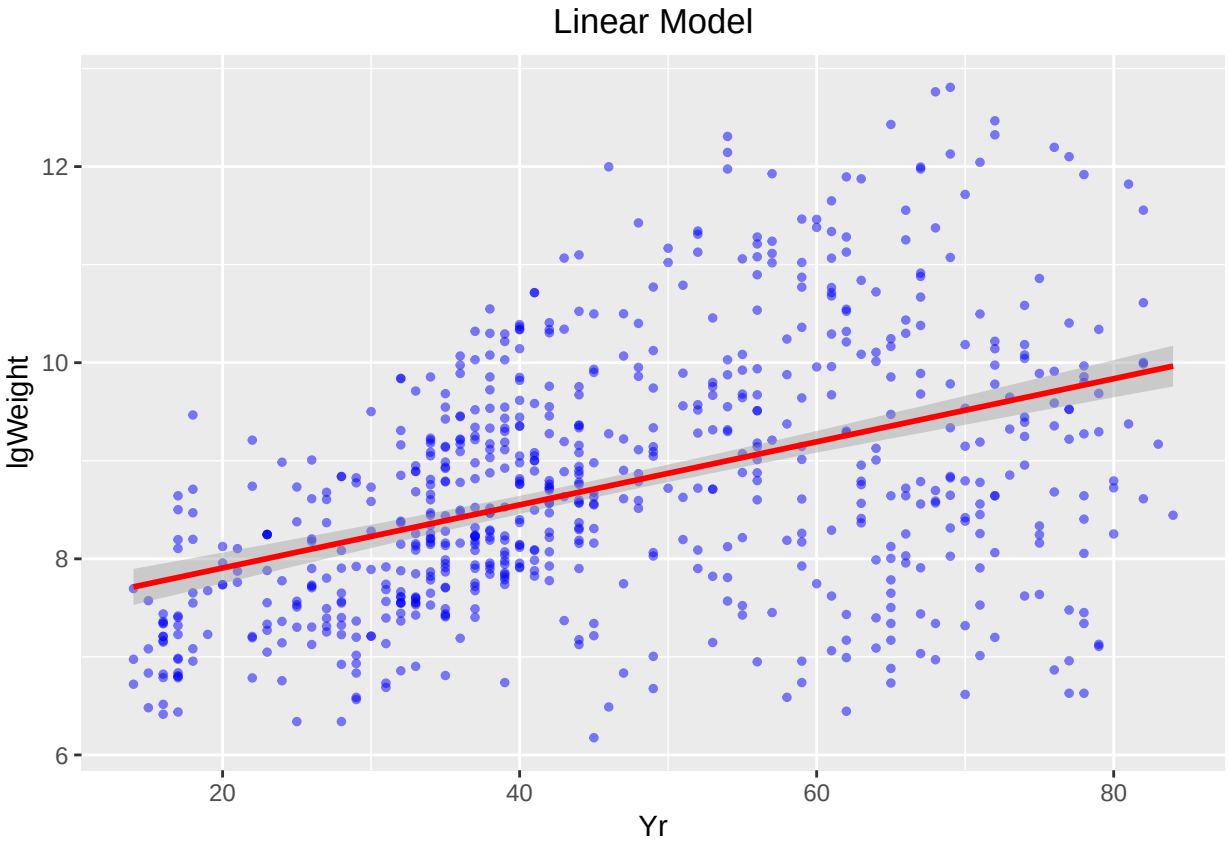
We use the Aircraft dataset from the R library sm. These data record six characteristics of aircraft designs which appeared during the twentieth century. However, we will use only two variables, a response variable (Weight) and one predictor (Yr). Moreover, to normalize the response, we will apply a log transformation.

```
lgWeight <- log(aircraft$Weight) #log response
Yr <- aircraft$Yr #predictor
```

Descriptive Statistics

First we plot the data and fit a simple linear regression. As we can see in the chart, it seems that there is heterocedasticity because the variance increases when Yr does so.

```
plot2<-ggplot(data=aircraft)+
  geom_point(aes(x=Yr, y=lgWeight), alpha=0.5, size=1, color="blue")+
  geom_smooth(aes(x=Yr, y=lgWeight), method="lm", color="red")+
  labs(title = "Linear Model")+
  theme(plot.title = element_text(hjust = 0.5))

plot2
```



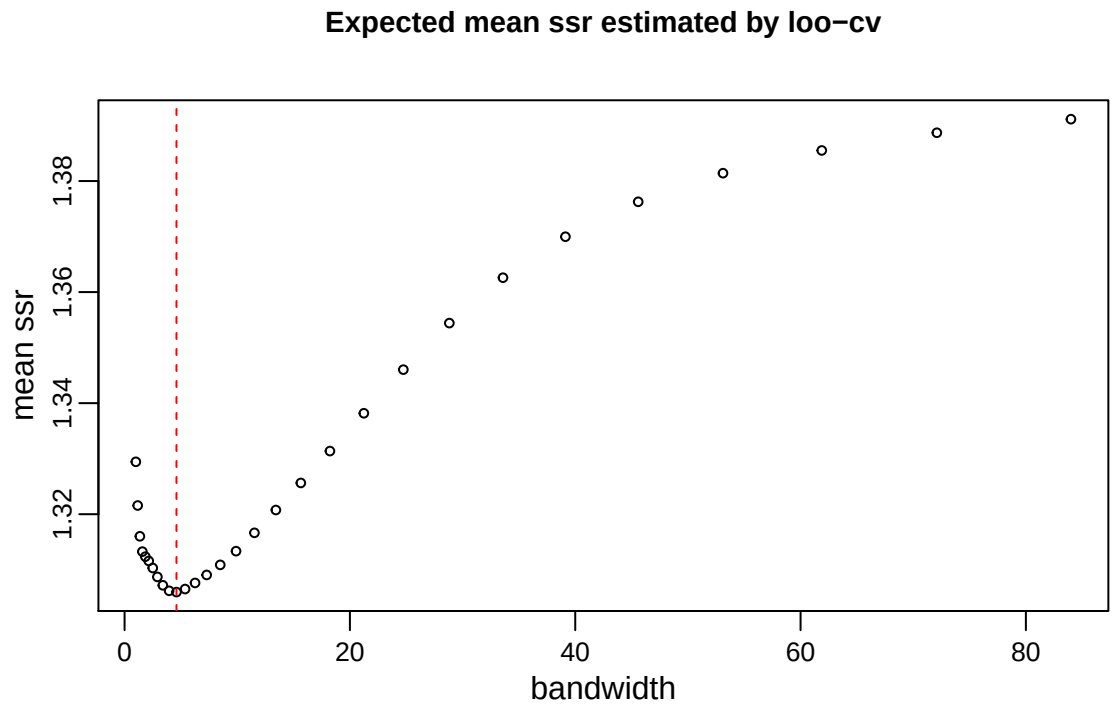
Questions

1. Use the function loc.pol.reg that you can find in ATENEA and choose all the bandwidth values you need by leave-one-out cross-validation (you have not to program it again! Just look for the right function in the*.Rmd files you can find in ATENEA)

With the aim to fit the optimum local lineal regression we must determine the optimal bandwidth. To find it, we apply leave-one-out cross-validation using the normal distribution as the Kernel function.

```
n <- length(Yr)
h.v <-  exp(seq(from=log(1), to = log(max(Yr)), length=30))
results <- data.frame(h=as.numeric(),mssr=as.numeric())

for (i in 1:length(h.v)) {
  #loocv is k-fold cv when k=n
  mssr <- k.fold.cv(x=Yr,y=lgWeight,h=h.v[i],k = n)
  results[i,] <- c(h.v[i],mssr)
```

```
}

h.opt <- results[results$mssr==min(results$mssr),][,1]

par(mgp=c(1.5,0.5,0))
plot(x=results$h,y=results$mssr,
     cex.axis=0.8, cex=0.6, cex.main=0.9,
     xlab = "bandwidth", ylab = "mean ssr",
     main = "Expected mean ssr estimated by loo-cv")
abline(v=h.opt,col=2,lty=2)
```



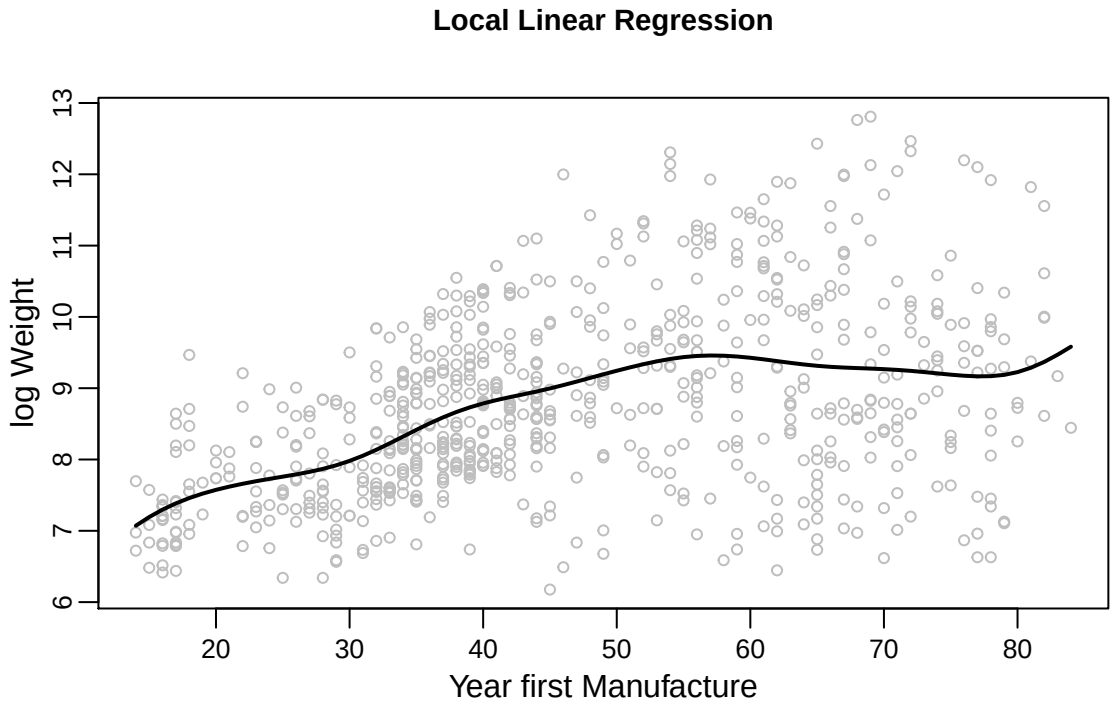
The the optimal value for the bandwidth is:

```
h.opt
```

```
## [1] 4.608341
```

Once the optimal bandwidth is found, we use the function locpolreg() to fit a local linear regression to express lgWeight as a function of Yr.

```
f.model <- locpolreg(x=Yr,y=lgWeight,h=h.opt,q=1,r=0,tg=NULL,
                     type.kernel="normal", nosubplot=TRUE, doing.plot=TRUE,
                     cex.main=0.9, cex.axis=0.8, cex=0.7,
                     xlab = "Year first Manufacture",
                     ylab = "log Weight",main= "Local Linear Regression")
```




```
y.hat <- f.model$mtgr
```

In the previous steps we have estimated the optimal Local Linear Regression for the data analysed (lgWeight as function of Yr), obtaining an estimated response for each obserbation ($\hat{m}(Yr)$).

Now, in the following steps we will estimate the conditional variance ($\hat{\sigma}^2$) of lgWeigth given Yr to see the effect of the variance on the previous estimated Local linear regression model at each local point of the regression.

In order to estimate the conditional variance, we calculate the estimated residuals form the previous regression, where:

$$\hat{\epsilon} = lgWeight - \hat{Y}$$

```
res <- lgWeight-y.hat
```

And then, we have to transform them:

$$Z = log(\hat{\epsilon}^2)$$

```
Z <- log(res^2)
```

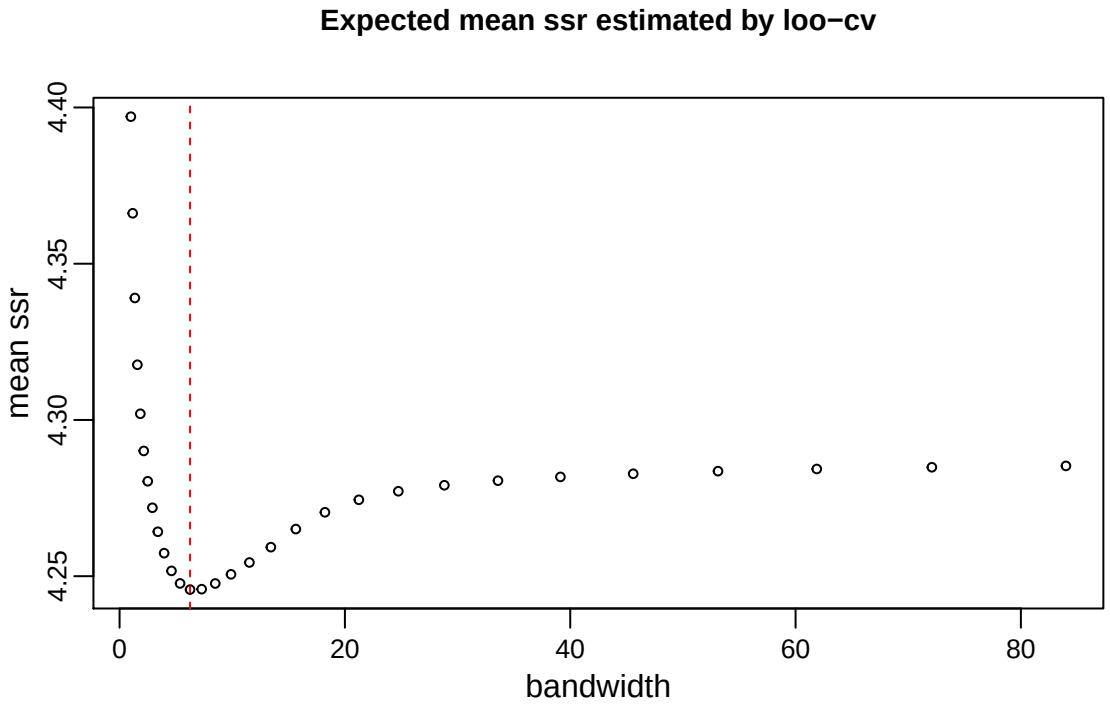
Once we have the data transformed we will fit a nonparametric regression of Z as a function of Yr. The estimated regression obtained will be $\hat{q}(Yr)$. Where $\hat{q}(Yr)$ is an estimation of $log(\sigma^2(Yr))$, therefore, the estimated conditional variance of the model will be:

$$\hat{\sigma}^2 = e^{\hat{q}(Yr)}$$

To proceed with the estimated regression of $\hat{q}(Yr)$ we have previously obtained Z from the estimated residuals of the previous model.

Once the data is prepared, apply again leave-one-out cross-validation using as the Kernel function the normal distribution.

```
n <- length(Yr) #loocv is k-fold cv when k=n
h.v <-  exp(seq(from=log(1), to = log(max(Yr)), length=30))
results <- data.frame(h=as.numeric(),mssr=as.numeric())

for (i in 1:length(h.v)) {

  mssr <- k.fold.cv(x=Yr,y=Z,h=h.v[i],k=n)
  results[i,] <- c(h.v[i],mssr)

}

h.opt <- results[results$mssr==min(results$mssr),][,1]

par(mgp=c(1.5,0.5,0))
plot(x=results$h,y=results$mssr,
     cex.axis=0.8, cex=0.6, cex.main=0.9,
     xlab = "bandwidth", ylab = "mean ssr",
     main = "Expected mean ssr estimated by loo-cv")
abline(v=h.opt,col=2,lty=2)
```



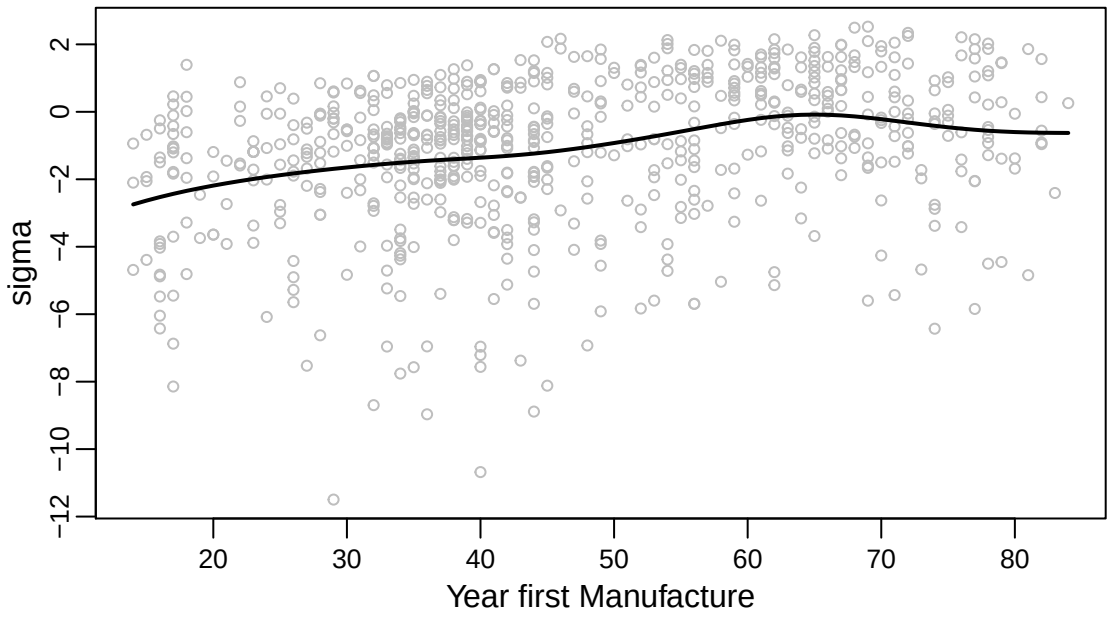
The optimal bandwidth for $\hat{q}(Yr)$ is

```
h.opt
```

```
## [1] 6.255377
```

When the optimal bandwidth is established, we can proceed with the local linear regression of $\hat{q}(Yr)$.

```
z.model <- locpolreg(x=Yr,y=Z,h=h.opt,q=1,r=0,tg=NULL,type.kernel="normal",
                     nosubplot=TRUE,doing.plot=TRUE, cex.main=0.9, cex.axis=0.8,
                     cex=0.7,xlab = "Year first Manufacture", ylab = "sigma")
```



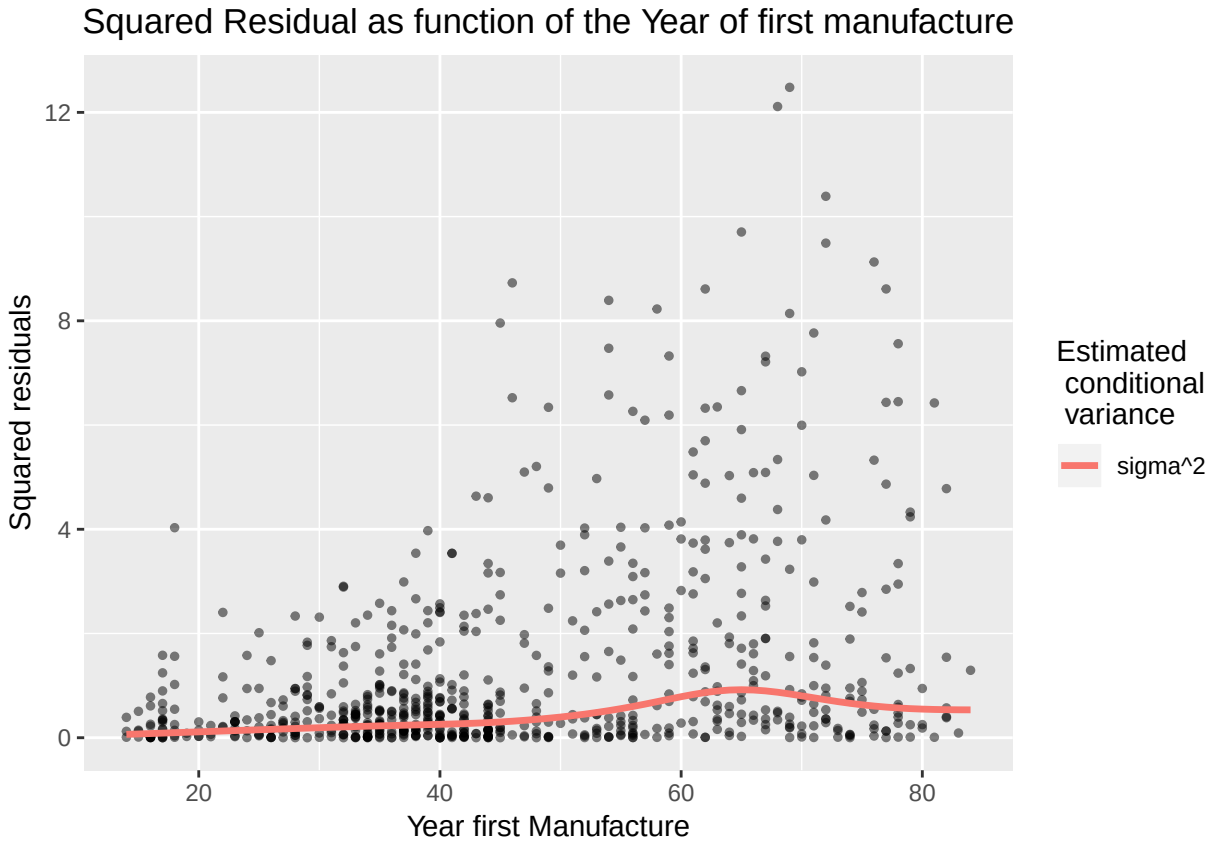
Once the regression for the $\hat{q}(Yr)$ is completed, we proceed applying $\hat{\sigma}^2 = e^{\hat{q}(Yr)}$ to obtain the conditional variance of the model ($\hat{\sigma}^2$).

The code implemented to obtain this transformation is the following:

```
e.val <- z.model$mtgr
es.val <- z.model$S%*%lgWeight
sigma2<-exp(e.val)
sigma <- sqrt(sigma2)
```

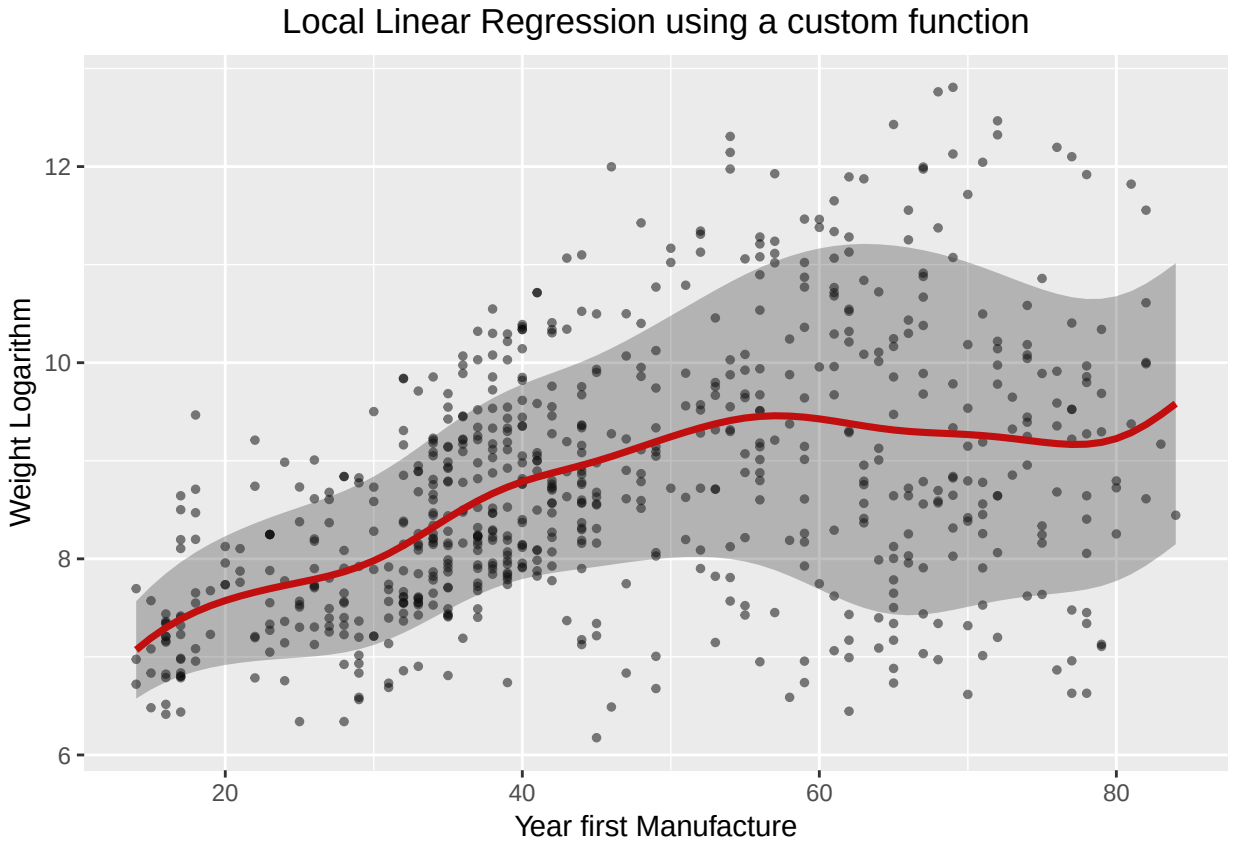
Once the transformation is done, we proceed with the graphic of $\hat{\epsilon}^2$ against Yr superimposing the estimated conditional variance of the model ($\hat{\sigma}^2$) and the estimated conditional standard deviation ($\hat{\sigma} = \sqrt{\hat{\sigma}^2}$).

```
ggplot(as.data.frame(cbind(Yr,res^2)),aes(Yr,res^2))+
  geom_point(col="black",alpha=0.5, size=1)+
  geom_line(aes(y=sigma2,x=Yr,colour="red"),size=1.2)+
  ggtitle("Squared Residual as function of the Year of first manufacture")+
  xlab("Year first Manufacture")+ylab("Squared residuals")+
  theme(plot.title = element_text(hjust = 0.5))+
  scale_color_discrete(name = "Estimated \n conditional \n variance",
                       labels =paste("sigma^2"))
```



Now, we can finally plot $\hat{m}(Yr)$ and superimposing the bands of $\hat{m}(Yr) \pm 1.96\hat{\sigma}(Yr)$ using the estimated conditional standard deviation obtained.

```
regression.customm.function <- ggplot(aircraft, aes(Yr,lgWeight))+
  ggtitle("Local Linear Regression using a custom function")+
  xlab("Year first Manufacture")+ylab("Weight Logarithm")+
  theme(plot.title = element_text(hjust = 0.5))+
  geom_point(col="black", alpha=0.5, size=1)+
  geom_line(aes(y=f.model$mtgr, x=Yr), col="red",size=1.25)+
  geom_ribbon(aes(ymin=f.model$mtgr-1.96*sigma,
                  ymax=f.model$mtgr+1.96*sigma), alpha=0.3)

regression.customm.function
```



2. Use the function sm.regression from library sm and choose all the bandwidth values you need by direct plug-in (use the function dpill from the same library KernSmooth).

The objective of this exercise is to replicate the results obtained in the previous point by using the functions available in the library sm of R.

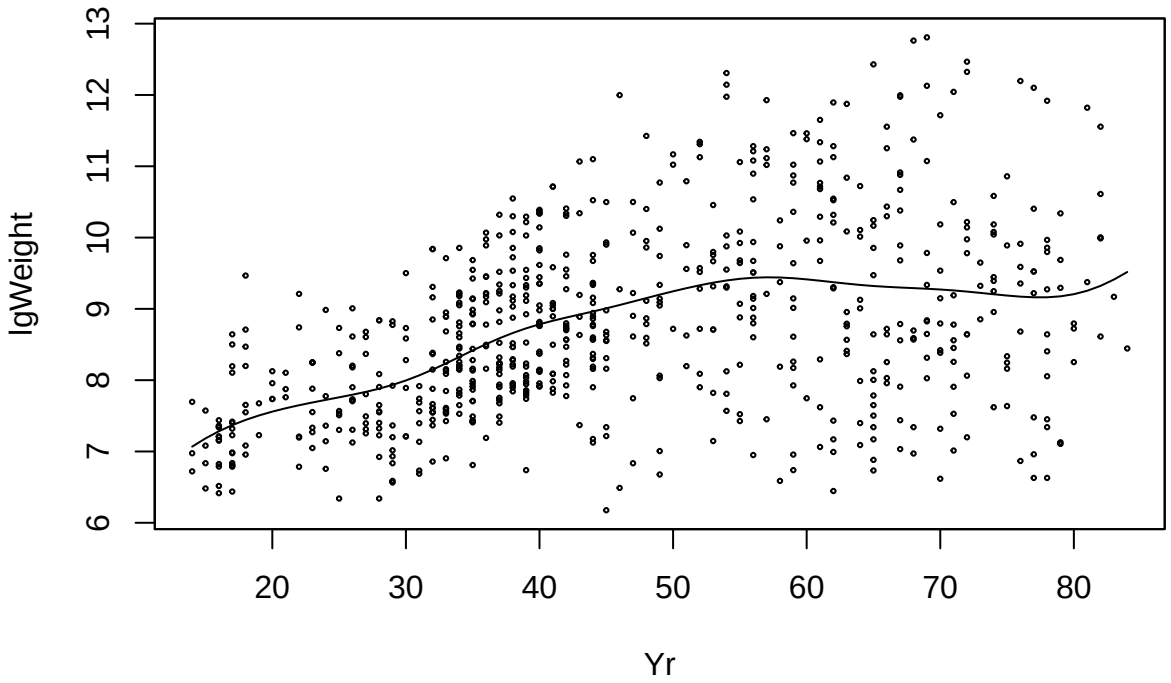
As before, we will start finding the optimal bandwidth. We will use the function dpill() available in the library sm.

```
h.dp <- dpill(x=Yr,y=lgWeight)
h.dp
```

```
## [1] 5.021118
```

Using the optimal bandwidth, we will fit the optimal local linear regression ($\hat{m}(Yr)$) represented in the following plot.

```
s.model <- sm.regression(x=Yr, y=lgWeight, h=h.dp, eval.points=Yr)
```



Once $\hat{m}(Yr)$ is found, we need to compute Z where, just as a reminder, $Z = log((lgWeight - \hat{y})^2)$.

```
y.hat <- s.model$estimate
eval.points <- s.model$eval.points

res <- lgWeight-y.hat
Z <- log(res^2)
```

When Z is computed, we will proceed with the regression of Z as a function of Yr to finally obtain $\hat{q}(Yr)$. Where $\hat{\sigma}^2 = e^{\hat{q}(Yr)}$ corresponds to the estimated conditional variance of the model.

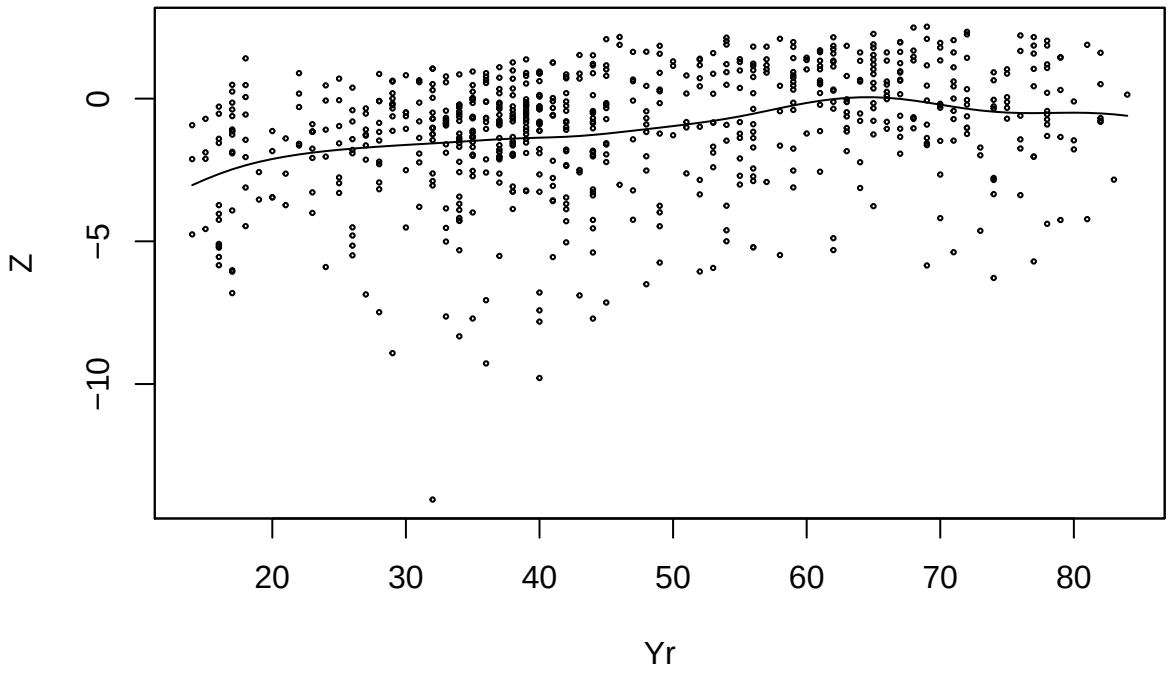
First, we need to obtain the optimal bandwidth, which will be:

```
h.dp <- dpill(x=Yr,y=Z)
h.dp
```

```
## [1] 4.287659
```

Finally, we will do the local linear regression to obtain $\hat{q}(Yr)$ displayed in the following plot.

```
z.model <- sm.regression(x=Yr, y=Z, h=h.dp, eval.points=Yr)
```

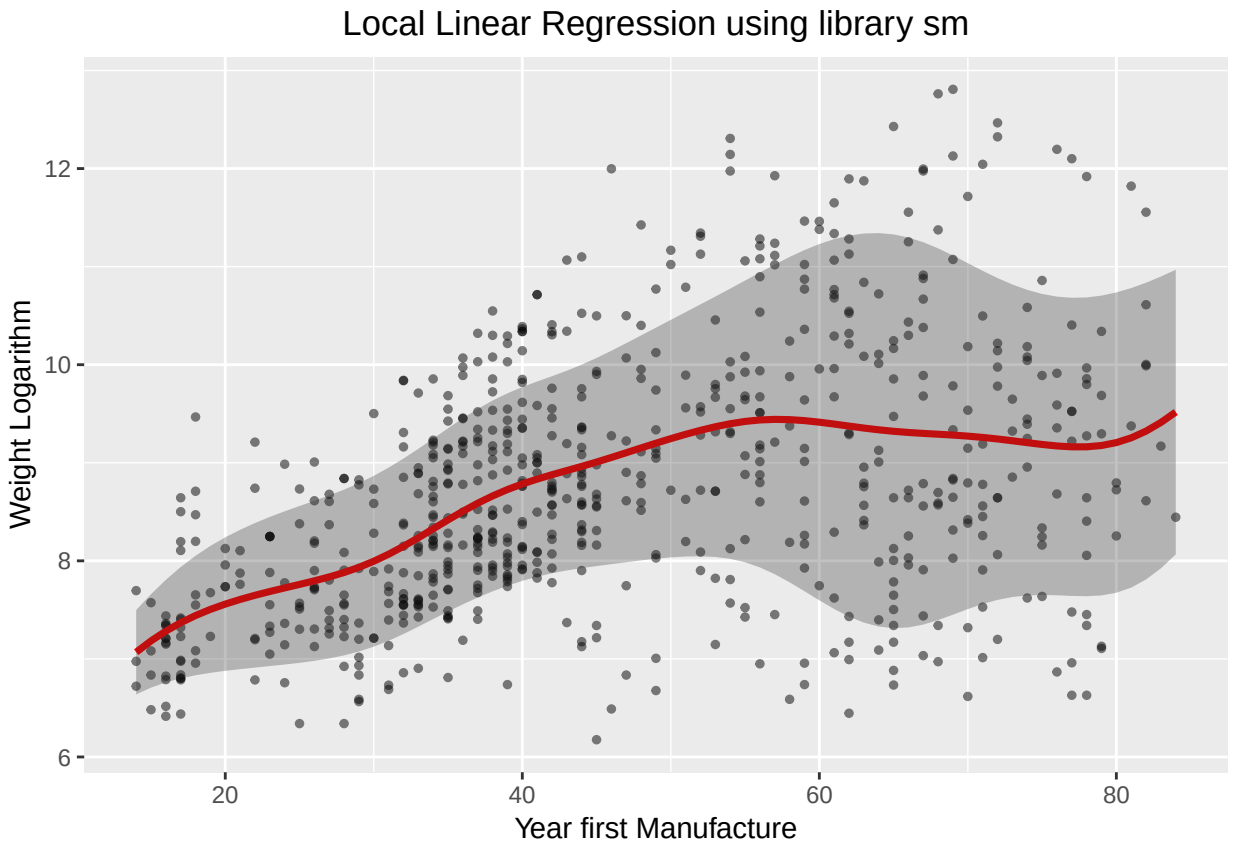


Finally, we apply the transformation $\hat{\sigma}^2 = e^{\hat{q}(Yr)}$ to obtain the estimated conditional variance and standard deviation.

```
y.hat <- z.model$estimate
sigma2<-exp(y.hat)
sigma <- sqrt(sigma2)
```

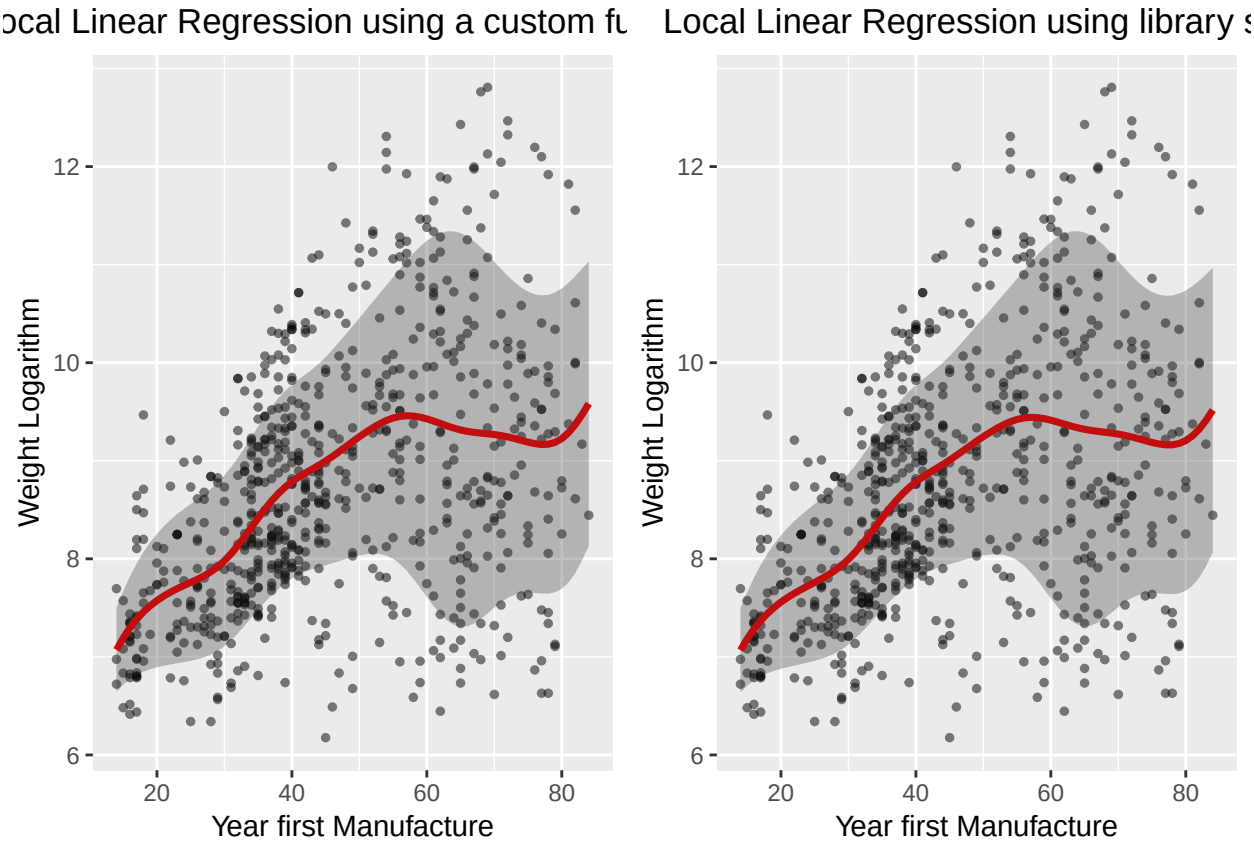
Now, we can finally plot $\hat{m}(Yr)$ and superimposing the bands of $\hat{m}(Yr) \pm 1.96\hat{\sigma}(Yr)$ using the estimated conditional standard deviation obtained with the library sm.

```
regression.r.function <-ggplot(aircraft, aes(Yr,lgWeight))+
  ggtitle("Local Linear Regression using library sm")+
  xlab("Year first Manufacture")+ylab("Weight Logarithm")+
  theme(plot.title = element_text(hjust=0.5))+
  geom_point(col="black",alpha=0.5, size=1)+
  geom_line(aes(y=s.model$estimate, x=Yr),col="red",size=1.25)+
  geom_ribbon(aes(ymin=s.model$estimate-1.96*sigma,
                  ymax=s.model$estimate+1.96*sigma),alpha=0.3)

regression.r.function
```



3. Results comparision.

```
grid.arrange(regression.customm.function, regression.r.function, ncol=2)
```



As we can see, both models are quite similar. The difference is due to the method used to chose the bandwidth parameter.