UK Complex Statistical Methods, WS 2024/25
Exercise sheet 2

Khodosevich Leonid

November 28, 2024

1 Problem 1

Problem 1 1. Find an explicit expression for the weight function $W_i(x)$ of a local linear estimator $\hat{f}_n(x)=\sum_{i=1}^n W_i(x) Y_i$.(3 points) 2. Show that for any linear estimator $\hat{f}_n(x;h)$ under assumptions of Section 2.4

$$\mathrm{E}\{\mathrm{GCV}(h)\}=\mathrm{E}\left[\frac{1}{n} \sum_{i=1}^n\left\{f\left(x_i\right)-\widehat{f}_n\left(x_i ; h\right)\right\}^2\right]\left\{1+O\left(n^{-1}\right)\right\}+\sigma^2+o\left(n^{-1}\right)$$

as well as

$$\mathrm{E}[\exp \{AIC(h)\}]=\mathrm{E}\left[\frac{1}{n} \sum_{i=1}^n\left\{f\left(x_i\right)-\widehat{f}_n\left(x_i ; h\right)\right\}^2\right]\left\{1+O\left(n^{-1}\right)\right\}+\sigma^2+o\left(n^{-1}\right)$$

Hint: use the Taylor expansions $(1-x/n)^{-2}=1+2x/n+o\left(n^{-1}\right)$ and $\exp(2x/n)=1+2x/n+o\left(n^{-1}\right)$.(3 points)

Solution

I)

definitions reminder

$$X=X(x)=\left(\begin{array}{cccc}1 & \left(\frac{X_1-x}{h}\right) & \ldots & \left(\frac{X_1-x}{h}\right)^{\ell}/\ell! \\ \vdots & \vdots & \ldots & \vdots \\ 1 & \left(\frac{X_n-x}{h}\right) & \ldots & \left(\frac{X_n-x}{h}\right)^{\ell}/\ell!\end{array}\right)$$

$$Y=\left(\begin{array}{c}Y_1 \\ \vdots \\ Y_n\end{array}\right)$$

$$V=V(x)=\operatorname{diag}\left\{K\left(\frac{X_1-x}{h}\right), \ldots, K\left(\frac{X_n-x}{h}\right)\right\}$$

$P(x) = \left(1, x, x^2/2!, \ldots, x^\ell/\ell!\right)^t$
$W_i(x) = \frac{1}{nh} P(0)^t \left(\frac{1}{nh} X^t V X\right)^{-1} P\left(\frac{X_i - x}{h}\right) K\left(\frac{X_i - x}{h}\right)$.

In linear case:

$$X = X(x) = \begin{pmatrix} 1 & \left(\frac{X_1 - x}{h}\right) \\ \vdots & \vdots \\ 1 & \left(\frac{X_n - x}{h}\right) \end{pmatrix}$$

$$Y = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}$$

$$V = V(x) = \text{diag}\left\{K\left(\frac{X_1 - x}{h}\right), \ldots, K\left(\frac{X_n - x}{h}\right)\right\}$$

$P(x) = (1, x)^t$

Let $Z_i = \frac{X_i - x}{h}$

$$X^t V X = \begin{pmatrix} 1 & \left(\frac{X_1 - x}{h}\right) \\ \vdots & \vdots \\ 1 & \left(\frac{X_n - x}{h}\right) \end{pmatrix}^t \begin{pmatrix} K\left(\frac{X_1 - x}{h}\right) & \ldots & 0 \\ \vdots & \ldots & \vdots \\ 0 & \ldots & K\left(\frac{X_1 - x}{h}\right) \end{pmatrix} \begin{pmatrix} 1 & \left(\frac{X_1 - x}{h}\right) \\ \vdots & \vdots \\ 1 & \left(\frac{X_n - x}{h}\right) \end{pmatrix}$$

$$= \begin{pmatrix} 1 & (Z_1) \\ \vdots & \vdots \\ 1 & (Z_n) \end{pmatrix}^t \begin{pmatrix} K(Z_1) & \ldots & 0 \\ \vdots & \ldots & \vdots \\ 0 & \ldots & K(Z_n) \end{pmatrix} \begin{pmatrix} 1 & (Z_1) \\ \vdots & \vdots \\ 1 & (Z_n) \end{pmatrix}$$

$$= \begin{pmatrix} 1 & \ldots & 1 \\ (Z_1) & \ldots & (Z_n) \end{pmatrix} \begin{pmatrix} K(Z_1) & K(Z_1)(Z_1) \\ \vdots & \vdots \\ K(Z_n) & K(Z_n)(Z_n) \end{pmatrix}$$

$$= \begin{pmatrix} \sum_{i=1}^n K(Z_i) & \sum_{i=1}^n K(Z_i)(Z_i) \\ \sum_{i=1}^n K(Z_i)(Z_i) & \sum_{i=1}^n K(Z_i)(Z_i^2) \end{pmatrix}$$

$$= \begin{pmatrix} V_1 & V_2 \\ V_2 & V_3 \end{pmatrix}, \text{where } V_1 = \sum_{i=1}^n K(Z_i), V_2 = \sum_{i=1}^n K(Z_i)(Z_i), V_3 = \sum_{i=1}^n K(Z_i)\left(Z_i^2\right)$$

Then

$$\left(\frac{1}{nh} X^t V X\right)^{-1} = nh \begin{pmatrix} V_1 & V_2 \\ V_2 & V_3 \end{pmatrix}^{-1} = \frac{nh}{V_1 V_3 - V_2^2} \begin{pmatrix} V_3 & -V_2 \\ -V_2 & V_1 \end{pmatrix}$$

Then

$$W_i(x) = \frac{1}{nh}\begin{pmatrix} 1 & 0 \end{pmatrix}\frac{nh}{V_1V_3 - V_2^2}\begin{pmatrix} V_3 & -V_2 \\ -V_2 & V_1 \end{pmatrix}\begin{pmatrix} 1 & Z_i \end{pmatrix} K\left(Z_i\right)$$
$$= \frac{1}{V_1V_3 - V_2^2}\begin{pmatrix} V_3 & -V_2 \end{pmatrix}\begin{pmatrix} 1 \\ Z_i \end{pmatrix} K\left(Z_i\right)$$
$$= \frac{1}{V_1V_3 - V_2^2}\left(V_3 - V_2 Z_i\right) K\left(Z_i\right)$$

where $V_1 = \sum_{i=1}^{n} K\left(Z_i\right),$

$$V_2 = \sum_{i=1}^{n} K\left(Z_i\right)\left(Z_i\right),$$
$$V_3 = \sum_{i=1}^{n} K\left(Z_i\right)\left(Z_i^2\right),$$
$$Z_i = \frac{X_i - x}{h}$$

II) Proof

1)

$$GCV(h) = \frac{n^{-1}\sum_{i=1}^{n}\left\{Y_i - \widehat{f}_n\left(X_i;h\right)\right\}^2}{\left\{1 - n^{-1}\sum_{i=1}^{n} W_i\left(X_i;h\right)\right\}^2}]$$
$$= n^{-1}\sum_{i=1}^{n}\left\{Y_i - \widehat{f}_n\left(X_i;h\right)\right\}^2\left(1 + \frac{2}{n}\sum_{i=1}^{n} W_i + o(n^{-1})\right)$$
$$= n^{-1}\sum_{i=1}^{n}\left\{Y_i - \widehat{f}_n\left(X_i;h\right)\right\}^2\left(1 + \frac{2}{n}\sum_{i=1}^{n} W_i + o(n^{-1})\right)$$

$$
\begin{aligned}
\mathrm{E}\left[GCV(h)\right] &= \mathrm{E}\left[n^{-1}\sum_{i=1}^{n}\left\{Y_i-\widehat{f}_n\left(X_i;h\right)\right\}^2\right]\left(1+\frac{2}{n}\sum_{i=1}^{n}W_i+o(n^{-1})\right)\\
&= \left(\mathrm{E}\left[n^{-1}\sum_{i=1}^{n}\left\{f\left(X_i\right)-\widehat{f}_n\left(X_i;h\right)\right\}^2\right)+\sigma^2-\mathrm{E}\left\{\frac{2\sigma^2}{n}\sum_{i=1}^{n}W_i\left(X_i;h\right)\right\}\right)
\end{aligned}
$$

$$
\left(1+\frac{2}{n}\sum_{i=1}^{n}W_i+o(n^{-1})\right)
$$

$$
= (\mathrm{E}\left[n^{-1}\sum_{i=1}^{n}\left\{f\left(X_i\right)-\widehat{f}_n\left(X_i;h\right)\right\}^2\right]\left(1+O(n^{-1})\right)+C
$$

$$
C=\left(\sigma^2+\sigma^2\mathrm{E}\left[\frac{2\sigma^2}{n}\sum_{i=1}^{n}W_i\right]+\sigma^2 o(n^{-1})-\sigma^2\mathrm{E}\left[\frac{2\sigma^2}{n}\sum_{i=1}^{n}W_i\right]-\mathrm{E}\left[n^{-1}\sum_{i=1}^{n}\left\{f\left(X_i\right)\right\}\right]\right)
$$

$$
= \left(\sigma^2+0+o(n^{-1})-0-o(n^{-1})\right)
$$

$$
= \left(\sigma^2+o(n^{-1})\right)
$$

Finally:

$$
\mathrm{E}\{\mathrm{GCV}(h)\}=\mathrm{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left\{f\left(x_i\right)-\widehat{f}_n\left(x_i;h\right)\right\}^2\right]\left\{1+O\left(n^{-1}\right)\right\}+\sigma^2+o\left(n^{-1}\right)
$$

2)

$$
\begin{aligned}
\mathrm{E}[\exp\{AIC(h)\}] &= \mathrm{E}\exp\left(\ln\left[n^{-1}\sum_{i=1}^{n}\left\{Y_i-\widehat{f}_n\left(X_i;h\right)\right\}^2 * \exp\left\{\frac{2}{n}\sum_{i=1}^{n}W_i\left(X_i;h\right)\right\}\right]\right)\\
&= \mathrm{E}\left(n^{-1}\sum_{i=1}^{n}\left\{Y_i-\widehat{f}_n\left(X_i;h\right)\right\}^2 * \exp\left\{\frac{2}{n}\sum_{i=1}^{n}W_i\left(X_i;h\right)\right\}\right)\\
&= \mathrm{E}\left(n^{-1}\sum_{i=1}^{n}\left\{Y_i-\widehat{f}_n\left(X_i;h\right)\right\}^2 * \left(1+\frac{2}{n}\sum_{i=1}^{n}W_i+o(n^{-1})\right)\right)
\end{aligned}
$$

Next derivations are exactly like with GCV. ■

Excercise sheet 2 (problems 2, 3)

Khodosevich Leonid

Problem 2

Subtask 1.

1. Let $X_1, \ldots, X_{300}$ be equispaced points on $[0, 1]$. Set seed to 2425 and simulate Y_i according to the model

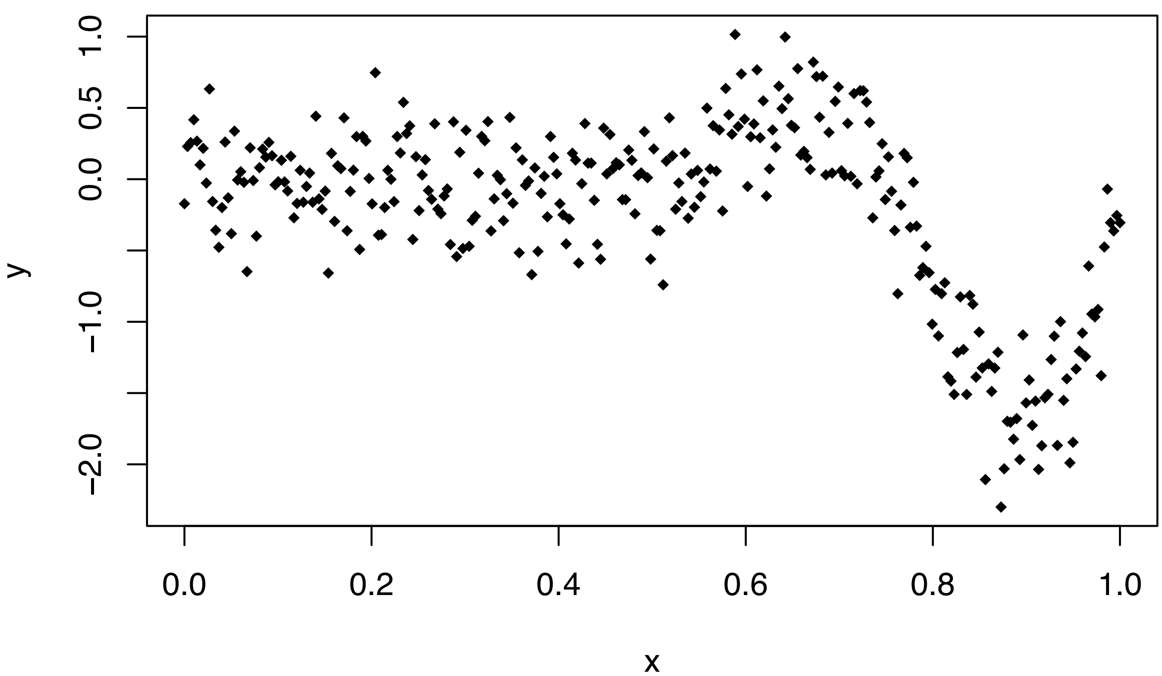
$$Y_i = f(X_i) + 0.3\epsilon_i, \quad f(X_i) = 3X_i^3 \sin(4\pi X_i) \tan(\pi X_i/4), \quad i = 1, \ldots, 300,$$

where $\epsilon_1, \ldots, \epsilon_{300} \overset{i.i.d.}{\sim} \mathcal{N}(0, 1)$.

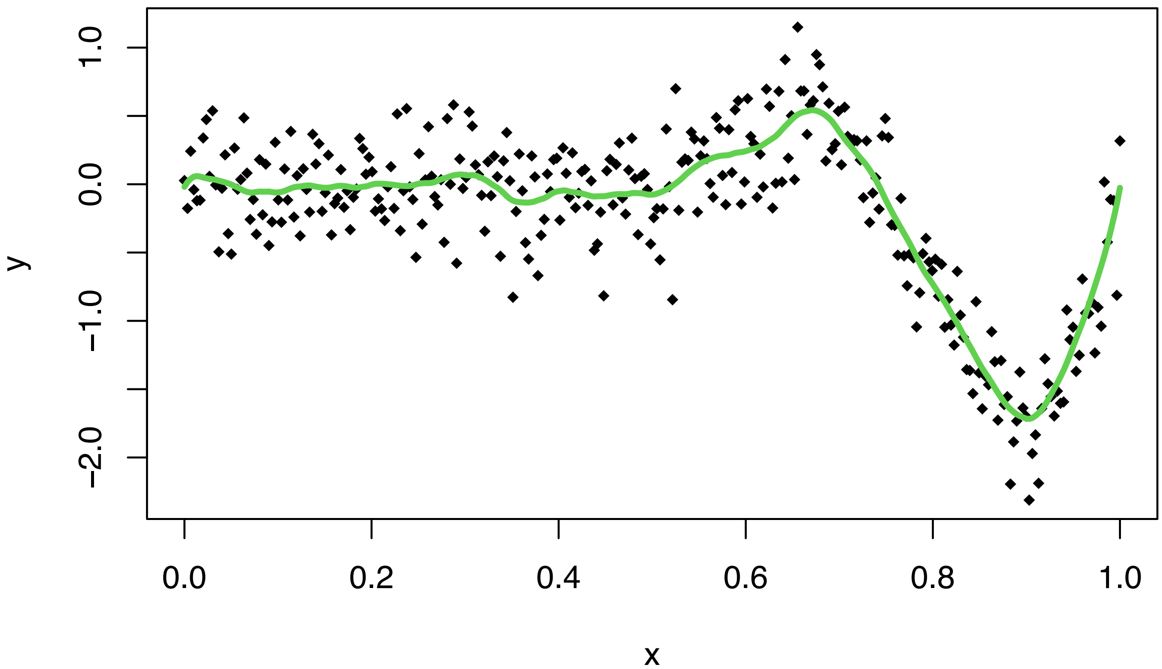
(a) Estimate regression function f by local polynomials of degree 1, 2 and 3, choosing every time the bandwidth by Akaike Information Criterion (AIC). Plot the true function f (without the data points), as well as all three estimators, putting a legend. Comment on the results. Compare also the bandwidths you obtained for all three estimators, giving theoretical justification. (3 points)

(b) Estimate the first derivative of f again using local polynomials of degree 1, 2 and 3, taking corresponding bandwidths from (a). Plot the true derivative f' and all three estimators, putting a legend. Comment on the results, giving theoretical justification. (3 points)

Data:



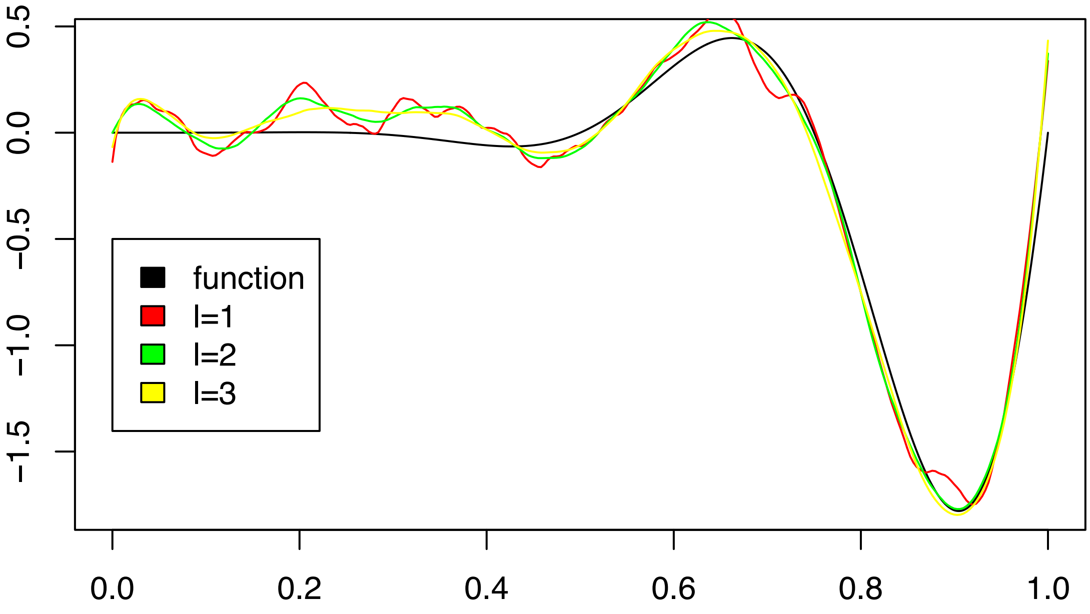
Estimation of local polynomial of degree 1




```
## [1] 0.04176153
```

a) Estimating regression function f by local polynomials of degree 1, 2 and 3, choosing every time the bandwidth by Akaike Information Criterion (AIC), as well as all three estimators, putting a legend. Comment on the results. Compare also the bandwidths you obtained for all three estimators, giving theoretical justification. (3 points):

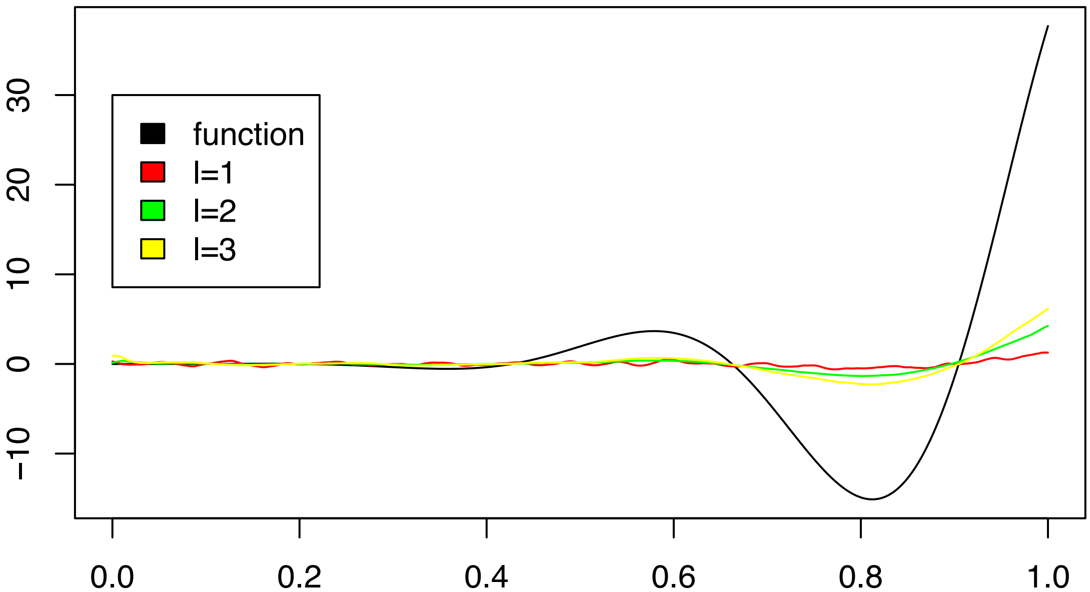
```
## [1] 1.00000000 0.03110416
## [1] 2.0000000 0.1032305
## [1] 3.000000 0.150491
```



As discussed in lecture, the bigger the grade of local polynomial, the more information it needs to fit properly. Thats why bandwidth of higher grades is bigger.

b) Estimate the first derivative of f again using local polynomials of degree 1, 2 and 3, taking corresponding bandwidths from (a). Plot the true derivative f' and all three estimators, putting a legend. Comment on the results, giving theoretical justification. (3 points)

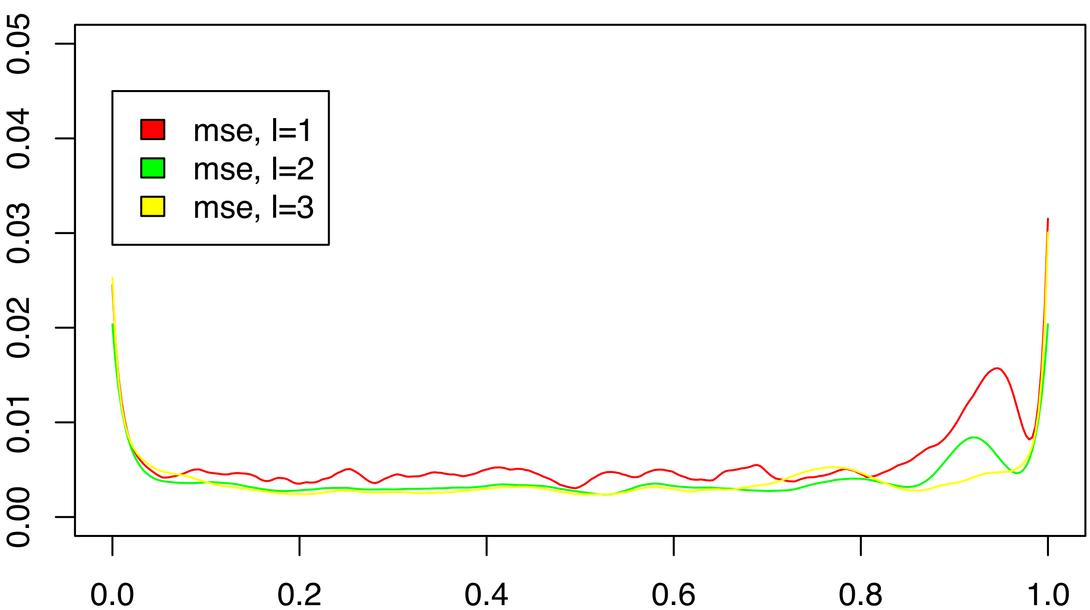
```
## [1] 1.00000000 0.03110416
## [1] 2.0000000 0.1032305
## [1] 3.000000 0.150491
```



As discussed on lecture, deriatives are being estimated poorly, because derivatives have lower l-constant of Hoelder class.

Subtask 2.

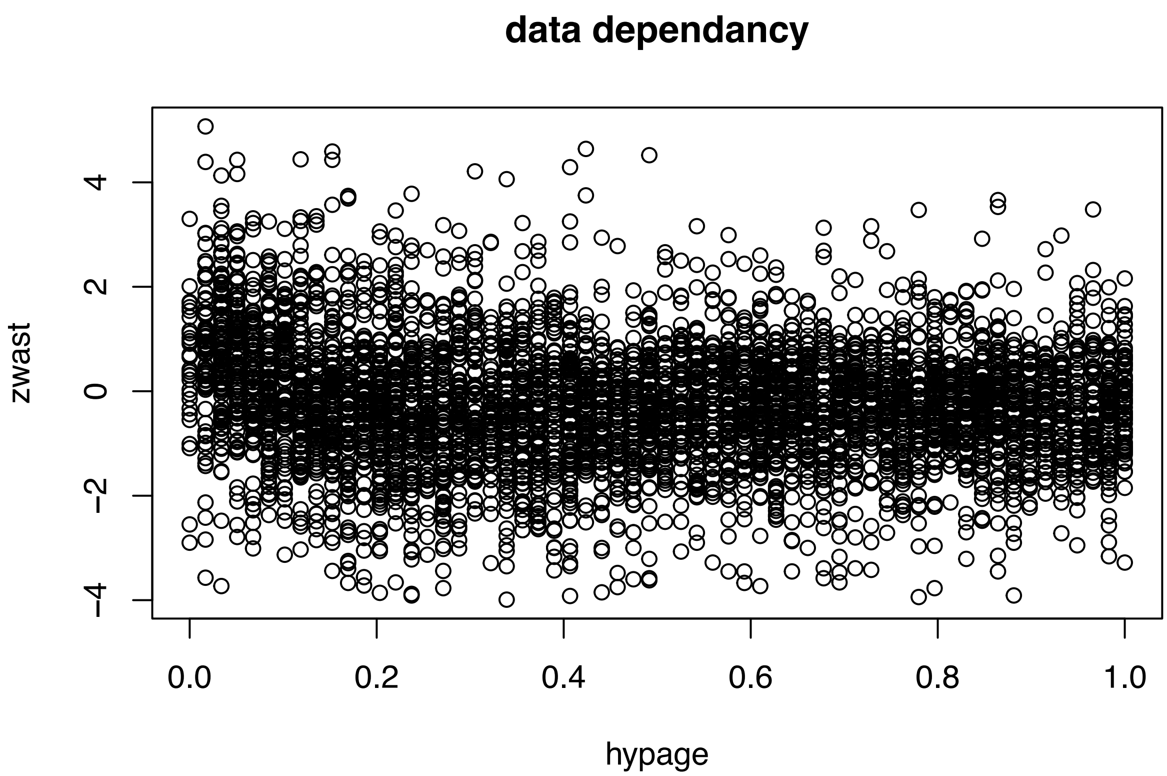
Simulate 200 samples as in 1, setting seed again to 2425. Obtain three estimators for each sample as in 1(a). With this, get Monte Carlo estimators of the mean squared errors at each point x of all three estimators (takes about 20 minutes) and plot these on one plot as a function of x, putting a legend. Comment on the results, giving theoretical justification. (3 points)



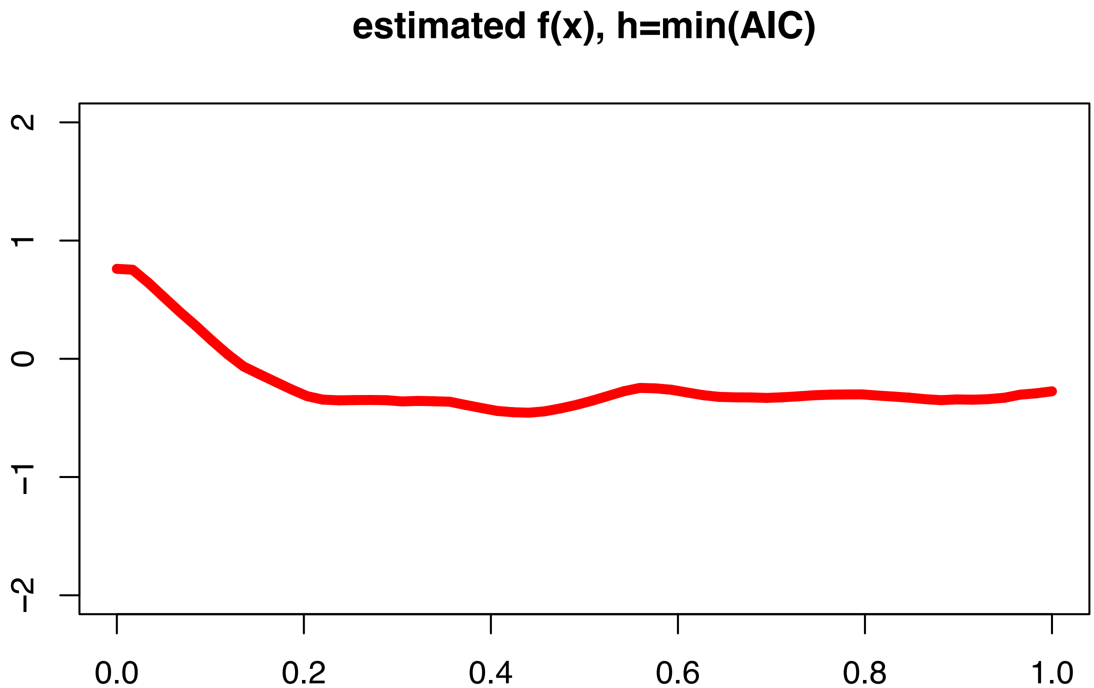
1) LPE with higher grade with proper choice of h approximate function better
2) There is major boundary problem for all of 3 estimators.

Problem 3

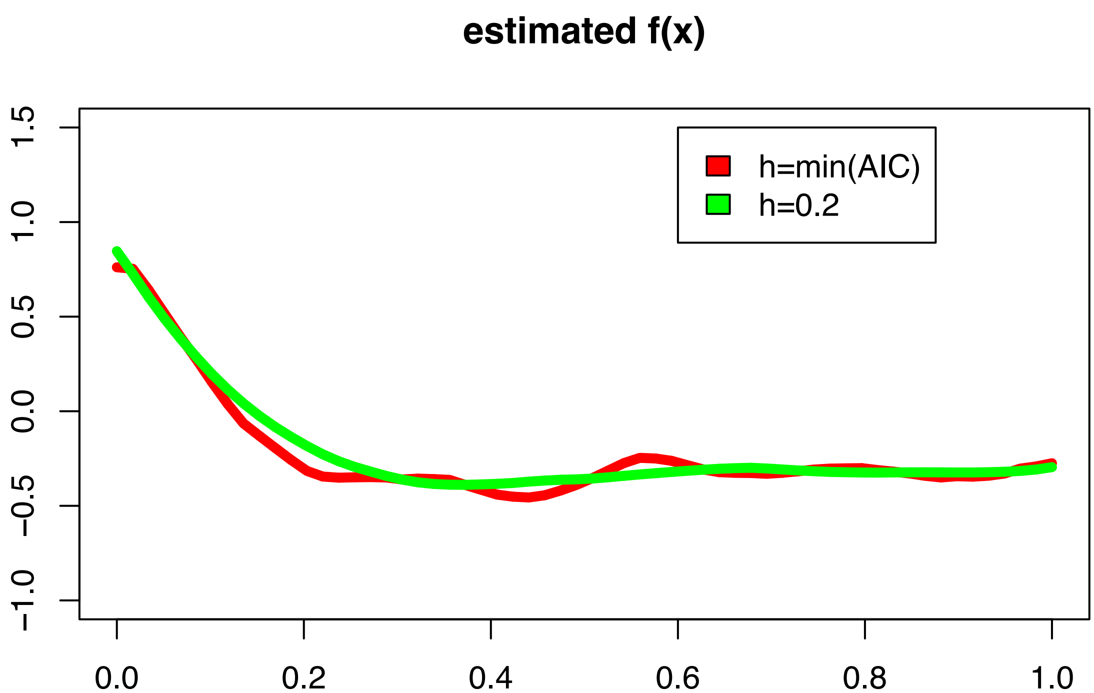
Read the dataset "Kenya DHS" into R and consider variables zwast as a response and hypage as a covariate. Variable zwast is the weight-for-height Z-score. Scale the covariate into [0, 1] interval. Estimate f and its first derivative in the model zwasti =f(hypagei)+ eps_i, i=1,...,4686, with local polynomial estimator of degree 1, choosing the bandwidth with AIC. Plot the resulting estimator without the data. Is the chosen bandwidth plausible? Now, set the bandwidth to h = 0.2. Add this estimator to the previous plot and comment on the differences. Finally, plot both estimators (with two different bandwidths) of the first derivative of f on one plot. Comment on the results. How can you interpret both f and its first derivative for these data? (5 points)




```
## [1] 0.08469634
```

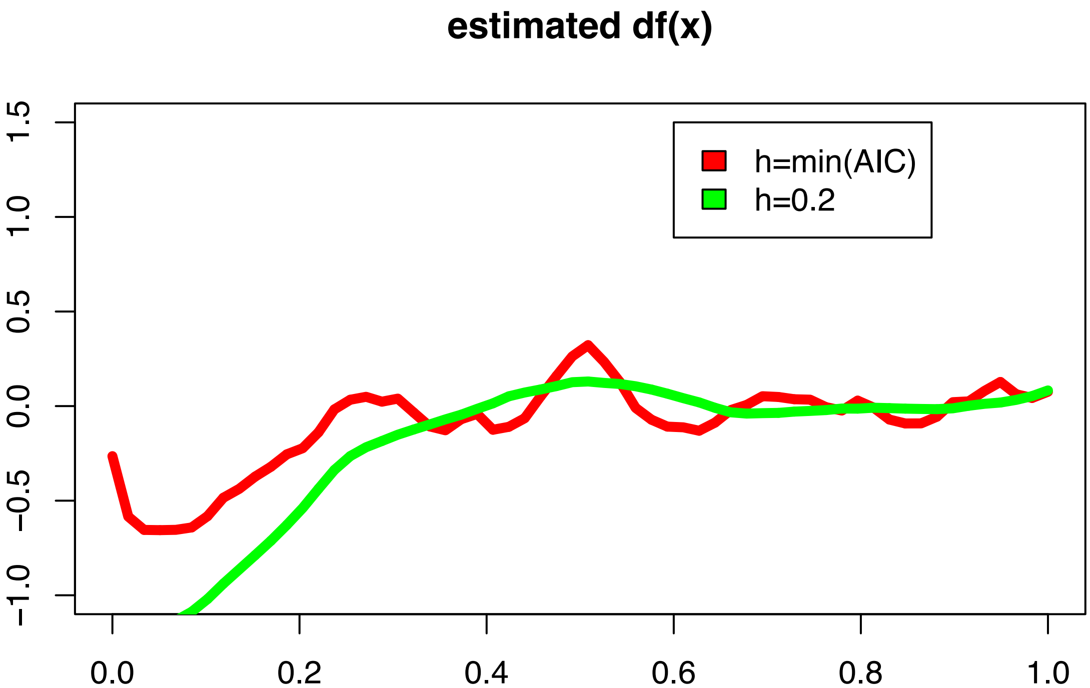



```
## [1] 0.08469634

## [1] 0.2
```



Finally, plot both estimators (with two different bandwidths) of the first derivative of f on one plot. Comment on the results. How can you interpret both f and its first derivative for these data? (5 points)

```
## [1] 0.08469634

## [1] 0.2
```



From both plots we can see, that h chosen from AIC minimization is overfitting to data, causing much higher varience and lower bias. While h=0.2 yields smooth approximation for both f(x) and df(x). I dont really know meaning for hypage variable, so cannot really interpret correctly. But from what I see, the groth of BMI(zwast variable) is highly dependent on hypage if hypage is low, is it growth, BMI start to change slowly (got that from df graph)

```
set.seed(2425)
knitr::opts_chunk$set(warning = FALSE, message = FALSE)
library('ubiquity')
library(svMisc)

# kernel
k=function(x) (3*(1-x^2)/4)*(x<=1&x>=-1)

# local polynomial fit func
locpoly=function(x,y,h,l,ker,der=0){
  n = length(x)
  f=matrix(NA,n,der+1)
  W=rep(0,n)
  for (i in 1:length(x)){
    X=rep(1,n)
    XX=as.vector(outer(x,x[i],"-"))/h
    for (j in 1:l) X=cbind(X,XX^j/factorial(j))
    fit.lm=lm(y~X-1,weights=ker(XX))

    f[i,]=(fit.lm$coef)[1:(der+1)]
    H=influence(fit.lm, do.coef=FALSE)$hat
    index=which(labels(H)==i)
    W[i]=H[index]
  }
  return(list(f=f,W=W))
}

# GCV func
GCV=function(x,y,l,ker){
  gcv=function(x,y,h,l,ker){
    fit.lp=locpoly(x,y,h,l,ker)
    return(sum((y-fit.lp$f[,1])^2)/(1-mean(fit.lp$W))^2)
  }
  return(optimize(gcv,interval=c(0.001,1),x=x,y=y,l=l,ker=ker)$minimum)
}

AIC=function(x,y,l,ker){
  gcv=function(x,y,h,l,ker){
    fit.lp=locpoly(x,y,h,l,ker)
    return(log(sum((y-fit.lp$f[,1])^2)) + (2*mean(fit.lp$W)))
  }
  return(optimize(gcv,interval=c(0.001,1),x=x,y=y,l=l,ker=ker)$minimum)
}




n = 300
x = linspace(0, 1, 300)
f = function(x){3*x^3 * sin(4*pi*x) * tan(x*pi/4)}
y = f(x) + 0.3*rnorm(n)
plot(x,y,pch=18,cex=0.8)




n = 300
x = linspace(0, 1, 300)
f = function(x){3*x^3 * sin(4*pi*x) * tan(x*pi/4)}
y = f(x) + 0.3*rnorm(n)
plot(x,y,pch=18,cex=0.8)
h.gcv = AIC(x,y,l=1,ker=k)
fit.lp = locpoly(x, y, h=h.gcv, l = 1, ker=k)
```

```
lines(x,fit.lp$f[,1],col=3,lwd=3)
h.gcv



n = 300
x = linspace(0, 1, n)
f = function(x){3*x^3 * sin(4*pi*x) * tan(x*pi/4)}
y = f(x) + 0.3*rnorm(n)
clrs = c("black","red", "green", "yellow")
plot(x, f(x), type='n', ann=F)
lines(linspace(0, 1, 1000), f(linspace(0, 1, 1000)))
h_list = c()
for (ll in 1:3) {
  h.aic = AIC(x,y,l=ll,ker=k)
  fit.lp = locpoly(x, y, h=h.aic, l=ll, ker=k)
  lines(x,fit.lp$f[,1],lwd=1, col=clrs[ll+1])
  print(c(ll, h.aic))
  h_list = c(h_list, h.aic)
}
legend(0, -0.5, legend=c("function", "l=1", "l=2", "l=3"),
       fill = clrs)






n = 300
x = linspace(0, 1, n)
df = function(x){3/4 * x^2 * (pi*x*1/cos((pi*x)/4)^2*sin(4*pi*x) + 4*(4*pi*x*cos(4*pi*x) + 3*sin(4*pi*x))*tan((pi*x)/4))}
y = f(x) + 0.3*rnorm(n)
clrs = c("black","red", "green", "yellow")
plot(x, df(x), type='n', ann=F)
lines(linspace(0, 1, 1000),df(linspace(0, 1, 1000)))
for (ll in 1:3) {
  h.aic = h_list[ll]#AIC(x,y,l=ll,ker=k)
  fit.lp = locpoly(x, y, h=h.aic, l=ll, ker=k, der=1)
  lines(x,fit.lp$f[,2],lwd=1, col=clrs[ll+1])
  print(c(ll, h.aic))
}
legend(0, 30, legend=c("function", "l=1", "l=2", "l=3"),
       fill = clrs)



n = 300
n_samples = 200
x = linspace(0, 1, n)
f = function(x){3*x^3 * sin(4*pi*x) * tan(x*pi/4)}
y = f(x) + 0.3*rnorm(n)
clrs = c("red", "green", "yellow")

mse_1 = replicate(n, 0)
mse_2 = replicate(n, 0)
mse_3 = replicate(n, 0)
for (i in 1:n_samples) {
  break # remove to compute
  y = f(x) + 0.3*rnorm(n)

  ll = 1
  h.aic = AIC(x,y,l=ll,ker=k)
  fit.lp = locpoly(x, y, h=h.aic, l=ll, ker=k)
  mse_1 = mse_1 + (fit.lp$f[,1] - f(x))**2

```

```
  ll = 2
  h.aic = AIC(x,y,l=ll,ker=k)
  fit.lp = locpoly(x, y, h=h.aic, l=ll, ker=k)
  mse_2 = mse_2 + (fit.lp$f[,1] - f(x))**2

  ll = 3
  h.aic = AIC(x,y,l=ll,ker=k)
  fit.lp = locpoly(x, y, h=h.aic, l=ll, ker=k)
  mse_3 = mse_3 + (fit.lp$f[,1] - f(x))**2

  progress(i, n_samples)
}
mse_1 = mse_1 / n_samples
mse_2 = mse_2 / n_samples
mse_3 = mse_3 / n_samples



# did that so computation is reduced
load("mse_1.RData")
load("mse_2.RData")
load("mse_3.RData")
```