

# Lecture 3

Jussi Klemelä

February 3, 2009

## 1 Kernel and nearest neighborhood estimator

**Kernel estimator.** The kernel estimator of the regression function is defined as

$$\hat{f}(x) = \sum_{i=1}^n p_i(x) Y_i, \quad (1)$$

where

$$p_i(x) = \frac{K_h(x - X_i)}{\sum_{i=1}^n K_h(x - X_i)}, \quad i = 1, \dots, n, \quad (2)$$

$K : \mathbf{R}^d \rightarrow \mathbf{R}$  is the kernel function,  $K_h(x) = K(x/h)/h^d$ , and  $h > 0$  is the smoothing parameter.

**Curse of dimensionality.** Let us consider the case where  $K(x) = I_{[-1/2, 1/2]^d}(x)$ . Then the support of  $K_h$  is  $[-h/2, h/2]^d$  and this support has volume  $h^d$ . When the explanatory variable has uniform distribution on  $[0, 1]^d$ , then the number of observations in the support of  $K_h$  is  $\approx n \cdot h^d$ . For example, when  $n = 1000$ ,  $h = 0.1$  and  $d = 3$ , then there is  $\approx 1$  observation in the support of  $K_h$ . In general, the local neighborhoods are almost empty in high dimensional spaces, and thus kernel estimators are not efficient in high dimensional spaces.

**Nearest neighborhood estimator.** We define the nearest neighborhood estimator for the regression function as

$$\hat{f}(x) = \sum_{i=1}^n p_i(x) Y_i, \quad x \in \mathbf{R}^d, \quad (3)$$

where

$$p_i(x) = \begin{cases} 1/k, & \text{when } X_i \in N_k(x), \\ 0, & \text{otherwise} \end{cases}$$

and  $N_k(x)$  is the collection of the  $k$  nearest realizations of explanatory variables to  $x$ :

$$N_k(x) = \{X_i : \|X_i - x\| \leq r_k(x), \quad i = 1, \dots, n\}, \quad (4)$$

$$r_k(x) = \min\{r > 0 : \#\{X_i \in B_r(x)\} = k\},$$

where  $B_r(x) = \{y \in \mathbf{R}^d : \|x - y\| \leq r\}$ . Note that  $\#\{i : X_i \in N_k(x)\} = k$ .

## 2 Linear regression

The linear regression model assumes that regression function has the form

$$f(x) = \beta_0 + x^T \beta_1, \quad x \in \mathbf{R}^d,$$

where  $\beta_0 \in \mathbf{R}$  and  $\beta_1 \in \mathbf{R}^d$ . The least squares estimator  $\hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1)$  of  $\beta = (\beta_0, \beta_1)$  is defined by

$$\hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1) = \operatorname{argmin}_{\beta_0 \in \mathbf{R}, \beta_1 \in \mathbf{R}^d} \sum_{i=1}^n (Y_i - \beta_0 - \beta_1^T X_i)^2.$$

Let us denote by  $\mathbf{X}$  the  $n \times (d+1)$ -matrix whose  $i$ th row is  $(1, X_i^T)$ , where  $X_i$  is interpreted as a column vector of length  $d$ . Let  $\mathbf{y}$  be the column vector of length  $n$  whose  $i$ th element is  $Y_i$ . Then

$$\hat{\beta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}, \quad (5)$$

where we assume that  $\mathbf{X}^T \mathbf{X}$  is nonsingular. In the case  $d = 1$  we get

$$\beta_1 = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^n (X_i - \bar{X})^2}, \quad \beta_0 = \bar{Y} - \beta_1 \bar{X},$$

where

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i, \quad \bar{Y} = \frac{1}{n} \sum_{i=1}^n Y_i.$$

**Capital asset pricing model.** The capital asset pricing model (CAPM) takes  $Y = R - r$ , where  $R$  is a daily (monthly) return of a stock and  $r$  is the risk free rate for the period of one day (month). The explanatory variable is  $X = R_M - r$ , where  $R_M$  is the daily (monthly) return of a market portfolio. The market portfolio is taken to be some index like S&P 500 index or the Wilshire 5000 index.

### 3 Single index models

There exists a large number of regression function estimators which are based on the single index model assumption

$$f(x) = g(x^T\theta), \quad x \in \mathbf{R}^d, \quad (6)$$

where  $g : \mathbf{R} \rightarrow \mathbf{R}$  is the unknown link function and  $\theta \in \mathbf{R}^d$  is the unknown index vector. Note that unlike in the generalized linear model, to be defined later, the link function  $g$  is now unknown and needs to be estimated. An index summarizes many variables into one number, for example stock index, inflation index, cost-of-living index, or price index are such indices. A linear index  $x^T\theta$  can be generalized to a non-linear index  $v_\theta(x)$ , depending on parameter  $\theta$ , and then the single index model can be written as

$$f(x) = g(v_\theta(x)), \quad x \in \mathbf{R}^d.$$

We shall consider linear single index models (6).

**Identification.** The vector  $\theta$  is not uniquely defined: the use of the vector  $c\theta$  and the link function  $g_c(u) = g(u/c)$ , with some  $c > 0$ , leads to the same regression function  $f$ . To assure the uniqueness we shall assume that  $\|\theta\| = 1$ . (We could also assume that the first component of  $\theta$  is equal to one.) Also, the sign of the coefficient vector  $\theta$  is not unique, because the use of the vector  $-\theta$  and the link function  $g_-(u) = g(-u)$  leads to the same regression function.

**Estimation.** For a given  $\theta$ , the link function  $g$  can be estimated by applying univariate nonparametric regression with  $X_i^T\theta$ ,  $i = 1, \dots, n$ , as observations of a univariate explanatory variable. Thus we can proceed in the estimation of the regression function in the single-index model by first estimating the parameter vector  $\theta$  and then estimating the link function  $g$ . We consider both the minimization estimation approach and the average derivative approach.

**M-estimation approach.** In the minimization-estimation (M-estimation) approach one finds for each fixed  $\theta$  a nonparametric estimator  $\hat{g}_\theta$  of  $g_\theta(t) = E(Y_1 | X_1^T\theta = t)$  and then estimates  $\theta$  by

$$\hat{\theta} = \operatorname{argmin}_\theta \sum_{i=1}^n \psi(Y_i, \hat{g}_\theta(X_i^T\theta)),$$

where  $\psi$  is a contrast function. The contrast function  $\psi$  can be chosen for example as

$$\psi(y, z) = |y - z|^2,$$

which leads to a semiparametric least squares estimator.

## 4 Illustrations

We look at the following code in

<http://cc.oulu.fi/~jklemela/finatool/>

```
# we obtain returns of the DAX stock index

ticker<-c("^GDAXI")
destfile<-"~/pois"
ry<-read.yahoo(ticker, source="web", destfile=destfile)
#save(file="/home/jsk/Arti/statfina/Dax.var",list=c("ry"))
#load(file="/Users/jsk/Karhu/Arti/statfina/DaxMdax.var")
dm<-data.manip(ry,ticker)
method<-"return"
df<-data.final(dm,ticker,method=method)
n<-dim(df)[1]
S<-matrix(df[1:n,1],n,1)
plot(S,type="l")

# we calculate volatilities for the 5 day periods

perlen<-5
pernum<-floor(n/perlen)
volas<-matrix(0,pernum,1)
for (i in 1:pernum){
  beg<-(i-1)*perlen+1
  end<-(i-1)*perlen+perlen
  period<-S[beg:end]
  volas[i]<-sqrt(sum(period^2)/perlen)*sqrt(252)
}
plot(volas,type="l")

## 2D kernel estimation
```

```

# we use now the volatilities of two periods to predict
# the volatility of the next period

dendat<-matrix(0,pernum-2,3)
for (i in 1:(pernum-2)){
  dendat[i,1]<-volas[i]
  dendat[i,2]<-volas[i+1]
  dendat[i,3]<-volas[i+2]
}
plot(dendat[,1],dendat[,2])

library(scatterplot3d)
scatterplot3d(dendat)

# we make the logarithmic transform for the explanatory variables
# and estimate the regression function

x<-matrix(0,dim(dendat)[1],2)
x[,1]<-log(dendat[,1])
x[,2]<-log(dendat[,2])
plot(x)
y<-dendat[,3]
scatterplot3d(x[,1],x[,2],y)

h<-0.7
N<-c(40,40)
pcf<-pcf.kernesti(x,y,h,N)

dp<-draw.pcf(pcf,pnum=N)    # need package "denpro"

contour(dp$x,dp$y,dp$z,drawlabels=FALSE)

persp(dp$x,dp$y,dp$z,phi=30,theta=30)

```

## 5 Examination

A possible question in the examination:

- 2) a) Define a kernel estimator of a regression function.
- b) Define a nearest neighborhood estimator of a regression function.