ST516: Computational Sttistics Exam (Part B)

Katrine Eriksen and Katrine Bach

2 juni 2016

Task 4

Standard Normal Distribution:

Matrix regression:

Linear regression:

Residual standard error:

R^2**:**

R^2_{adj}**:**

F-Statistics:

Task 5

Theory ##Standard Normal Density ##Naive density estimation The Naive kernel method uses each point of estimation x as the center og the bin of width $2h$. ### Kernel density estimation The Kernel density estimation is an eppraoch that is rooted in the histogram methodology. The basic idea is to estimate the density function at a point x using neighboring observations. To express it more transparently, we consider the weight function

55

which is called the kernel weight. The kernel estimate of $f(x)$ is defined as

55

###Naive estimator: ###Gaussian Kernel ###Silverman's suggestion for Gaussian method: ###Density:

Here we apply our function densi that estimates the density using the naive estimator ot thre gaussian kernel. First we run it foe the Kernel density estimation without specifying d or h.

```
densi <- function(x,d=NULL,h=NULL, method="naive"){
n<-length(x)

if(length(h)==0){
  if (method=="kernel"){
    h <- 0.9*min(c(IQR(x)/1.34,sd(x)))*n^(-1/5)
  }else{
    h<- (max(x)-min(x))/(1+log2(n))
  }
```

```
  #Make warning with other methods
}else{
  h <- h
}

if(length(d)==0){
  minimum<-densi(x,min(x),h,method)
  maksimum<-densi(x,max(x),h, method)

  quantile1<-quantile(x,names=FALSE)[2]
  q1<-densi(x,quantile1,h,method)

  quantile3<-quantile(x,names=FALSE)[4]
  q3<-densi(x,quantile3,h,method)

  medians <- densi(x,median(x),h,method)
  means <- densi(x,mean(x),h,method)

  smoke <- matrix(c(min(x),minimum,quantile1,q1,median(x),medians,mean(x),means,quantile3,q3,max(x),maks
  colnames(smoke) <- c("x","y")
  rownames(smoke) <- c("Min","1st Quantile","Median", "Mean", "3dr Quantile", "Max")
  smoke <- as.table(smoke)
  print(sprintf("bandwidth: %f", h))
  return(smoke)

}else{
  d=d
}

if(method=="kernel"){
  kern<-0
  for (i in 1:n) {
    kern[i] <-(1/h)*dnorm((d-x[i])/h)
  }
  summ<- sum(kern)
  f<-(1/n)*summ
  return(f)
}else{
  nav <- 0
  for(i in 1:n){
    if(abs((d-x[i])/h)<1){
      w <- 1/2
  }else{
      w <- 0
  }
    nav[i] <-(1/h)*w
  }
  summ <- sum(nav)
  f <- (1/n)*summ
  return(f)
}

}
```

```
densi(faithful$eruptions, method="kernel")
```

```
## [1] "bandwidth: 0.334777"
```

```
##                       x         y
## Min           1.6000000 0.2132889
## 1st Quantile 2.1627500 0.3106146
## Median        4.0000000 0.3850462
## Mean          3.4877831 0.1547187
## 3dr Quantile 4.4542500 0.4782279
## Max           5.1000000 0.1577281
```

Forklar hvad værdierne viser

Now we run it again using the Naive density estimation and again without specifying d and h.

```
densi(faithful$eruptions)
```

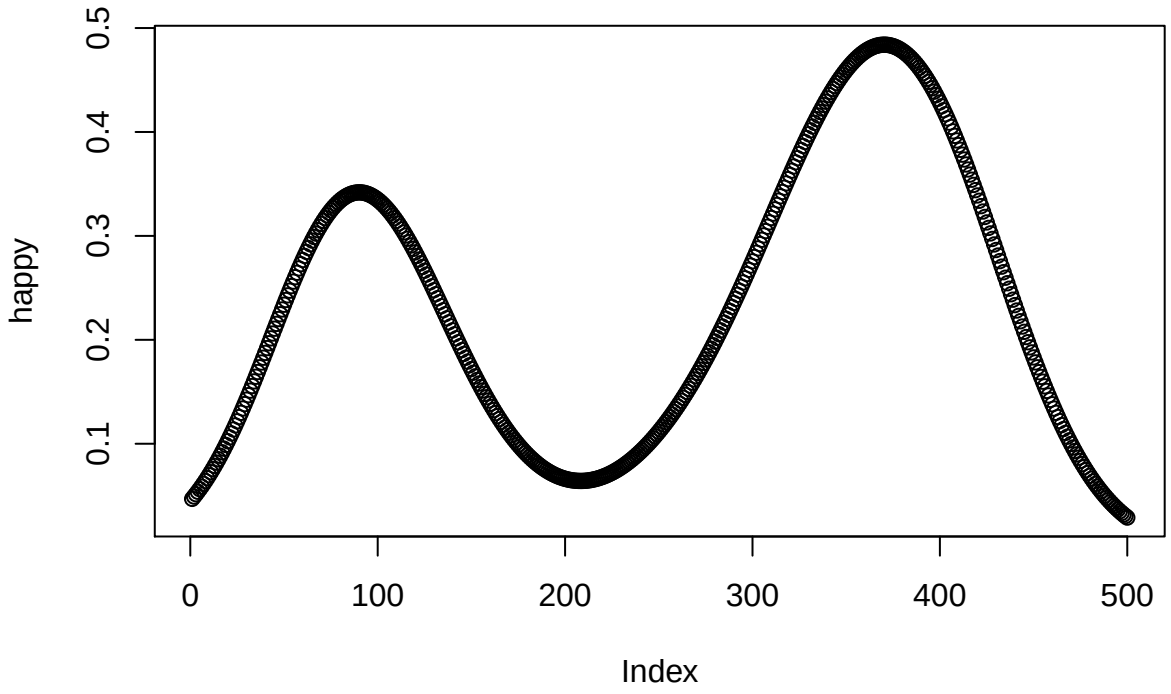
```
## [1] "bandwidth: 0.385146"
```

```
##                       x         y
## Min           1.6000000 0.2434142
## 1st Quantile 2.1627500 0.3913718
## Median        4.0000000 0.3913718
## Mean          3.4877831 0.1336392
## 3dr Quantile 4.4542500 0.5536479
## Max           5.1000000 0.1240935
```

Forklar noget om værdierne

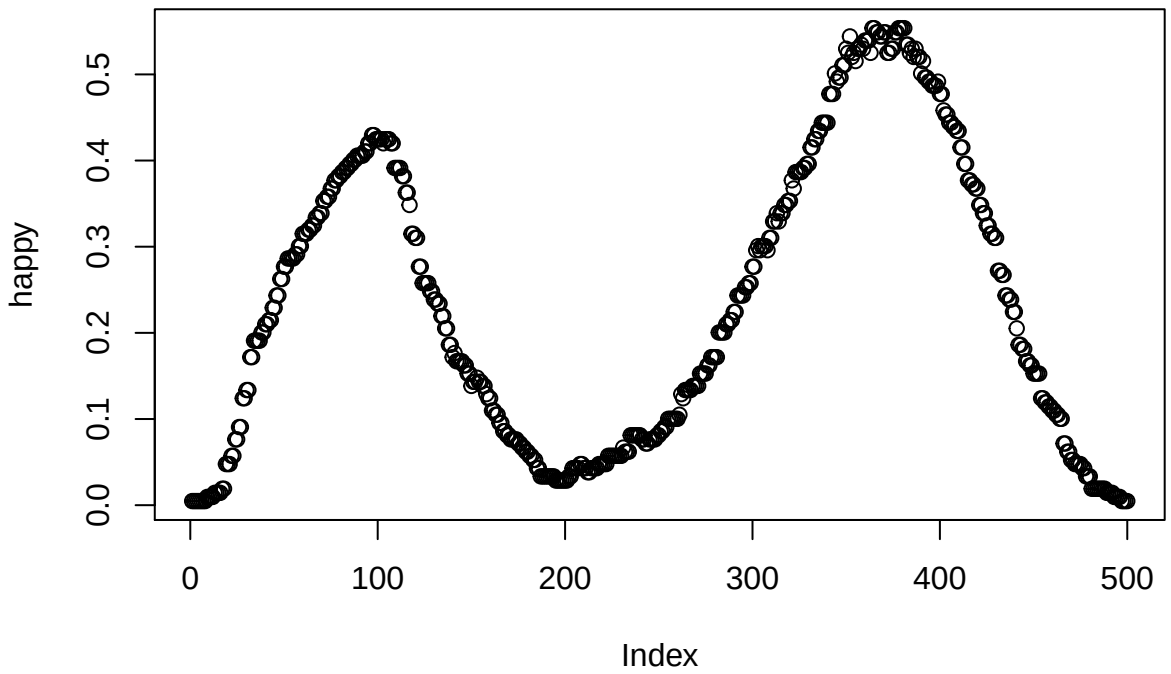
Now we use our function densiplot, that calls our density function and creates a density plot based on the specified method. We start by showing the plot for the Kernel denisty estimation.

```
densiplot <- function(x,n=500, method="naive", from= min(x)-(sd(x)/3), to=max(x)+(sd(x)/3)){
  happy<-c()
  for (d in seq(from,to,length.out = n)) {
    happy1<-densi(x,d=d,method=method)
    happy<-c(happy,happy1)
  }
  plot(happy)
}
densiplot(faithful$eruptions, method="kernel")
```



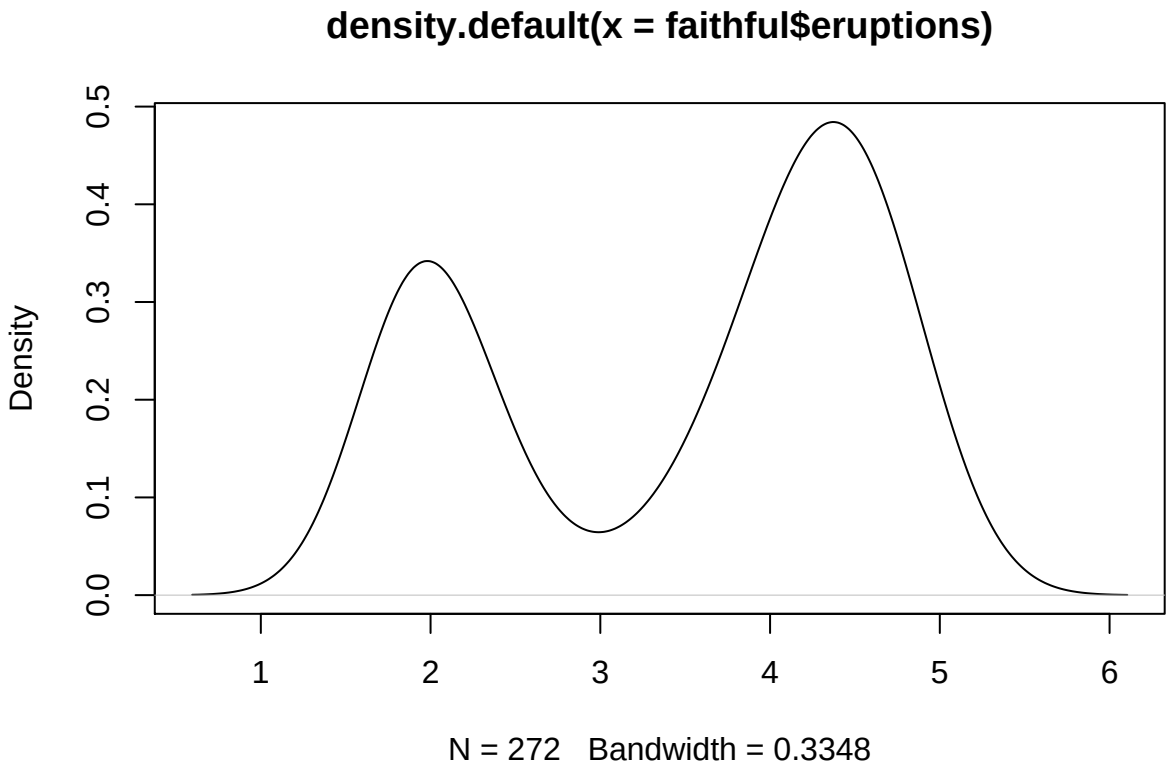
Now we run it again using the Naive density estimation and again without specifying d and h.

```
densiplot(faithful$eruptions)
```



And now we want to se how our densiplot function looks againts the already build-in function *density()*

```
plot(density(faithful$eruptions))
```




```
density(faithful$eruptions)
```

```
##
## Call:
##  density.default(x = faithful$eruptions)
##
## Data: faithful$eruptions (272 obs.); Bandwidth 'bw' = 0.3348
##
##        x                y
##  Min.   :0.5957   Min.   :0.0002262
##  1st Qu.:1.9728   1st Qu.:0.0514171
##  Median :3.3500   Median :0.1447010
##  Mean   :3.3500   Mean   :0.1813462
##  3rd Qu.:4.7272   3rd Qu.:0.3086071
##  Max.   :6.1043   Max.   :0.4842095
```

Her kan man også sammenligne tabellen med hinanden. Husk at lav funktionsbeskrivelser til densi() og densiplot()!!!!

Task 6

Markov chain:

Transition Probability:

Task 7

Beta Distribution:

Metropolis-Hastings sampler:

Rejection rate:
