Assignment 5 (S670)

FNU Anirudh

October 30, 2015

Question 1

```
infant_mortality<-
c(25.3,32.1,38.8,25.4,25.3,29.0,31.0,21.1,18.2,18.8,19.3,20.3,18.3,24.3,15.7,
24.0,16.3,19.0,16.8,17.5)
infant_mortality<-matrix(infant_mortality,c(4,5))
dimnames(infant_mortality)<-list(c("NorthEast","North
Central","South","West"), c("<8","9-11","12","13-15",">16"))
mat<-infant_mortality
twoway.median <- function(mat){
  meff.MP <- median(mat)
  beff.MP <- apply(mat,2,median,na.rm=T)
  mat.res <- mat -
matrix(rep(beff.MP,each=nrow(mat)),byrow=F,nrow=nrow(mat));

  beff.MP <- beff.MP - median(beff.MP)

  aeff.MP <- apply(mat.res,1,median,na.rm=T)

  res.MP <- mat.res -
matrix(rep(aeff.MP,each=ncol(mat)),byrow=T,ncol=ncol(mat))

  list(overall=meff.MP, row=aeff.MP, col=beff.MP, res=res.MP)
}
iter1 <- twoway.median(mat); #1st iteration
iter2<-twoway.median(iter1$res) #2nd iteration
mat<-rbind(iter2$res,iter2$col)
mat<-cbind(mat,iter2$row)
```

```
## Warning in cbind(mat, iter2$row): number of rows of result is not a
## multiple of vector length (arg 2)
```

```
mat[5,6]<-iter2$overall
rownames(mat)[5]<-"col effect"
colnames(mat)[6]<-"row effect"
mat
```

```
##                   <8  9-11    12 13-15   >16 row effect
## NorthEast     -1.55  0.00  0.55 -1.15  1.00       0.00
## North Central  1.55  0.00 -2.55  1.15  0.00       0.00
## South         10.30  4.05  0.00 -5.40 -0.15      -0.45
```

```
## West          -3.65 -6.40  0.45  2.35  0.00       0.00
## col effect    -0.05  0.00  0.45  0.15  0.00       0.00
```

```
# There is a difference when compared to table with row and column effect
#hence it depends if we start with row or column.
```

Question 2

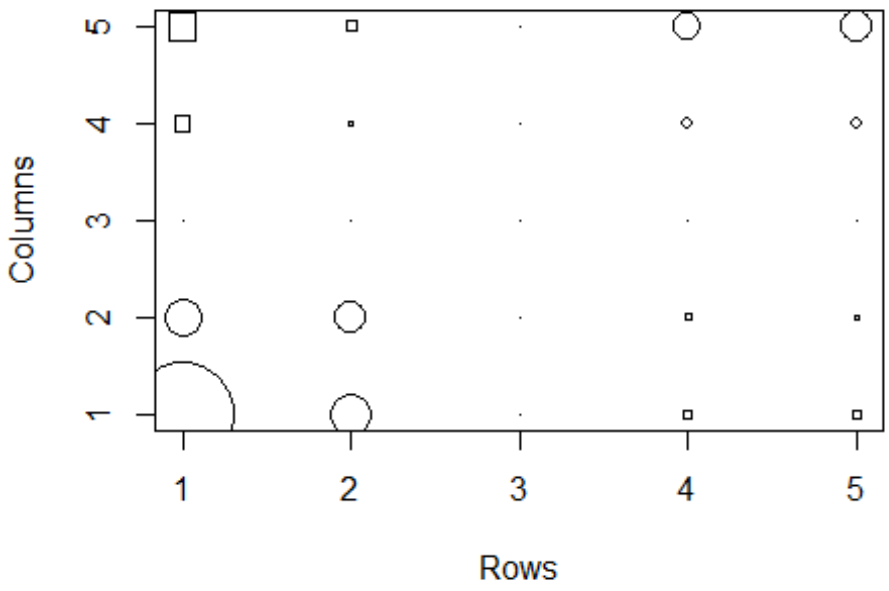
```
#a)
r1<-c(22.2,44.5,59.6,73.2,86.8)
r2<-c(10.5, 15.5, 29.0,36.5,46.2)
r3<-c(3.53, 5.76, 9.71, 14.0, 21.1)
r4<-c(1.04, 1.98, 2.45, 3.40, 5.40)
r5<-c(.641, .974, 1.80, 2.60, 3.64)

rowNames<-c("Food/Tobacco","Household","Medical/Health","Personal
care","Educ/research")
colNames<-c(1940, 1945, 1950, 1955, 1960)
personalExpenditureTable <- rbind(r1,r2,r3,r4,r5)
rownames(personalExpenditureTable)<-rowNames
colnames(personalExpenditureTable)<-colNames

medPolished<-medpolish(personalExpenditureTable)
```

```
## 1: 139.595
## Final: 139.595
```

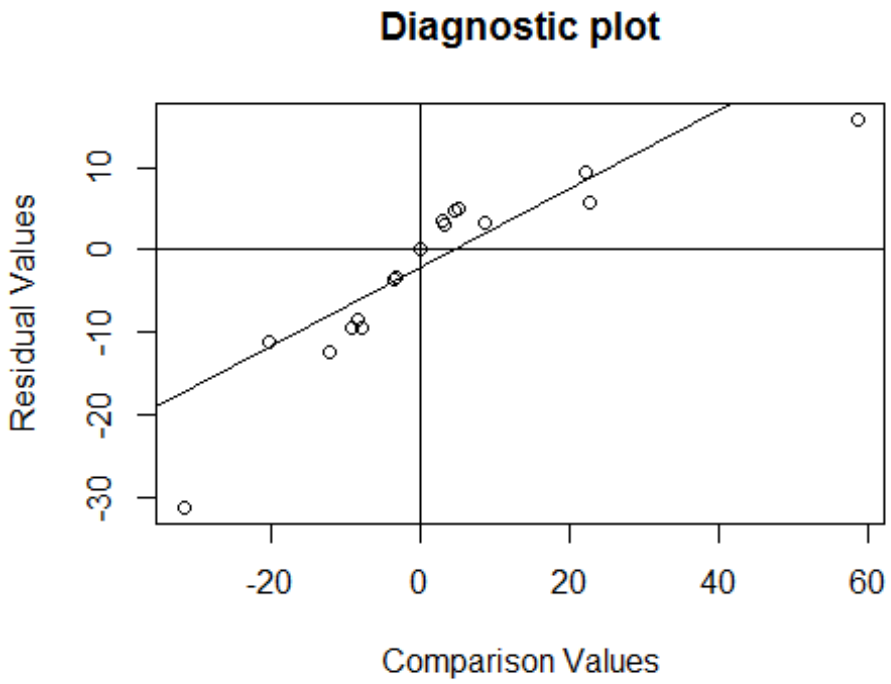
```
AnalogRSqr<- 1-((sum(abs(medPolished$residuals)))
/(sum(abs(personalExpenditureTable-medPolished$overall))))
AnalogRSqr
```

```
## [1] 0.6722237
```

```
#b)
plot(NA, NA, type = "n", xlim=c(1, 5), ylim=c(1, 5), xlab = "Rows",ylab =
"Columns")
for (i in 1:nrow(medPolished$residuals)){
  for (j in 1:ncol(medPolished$residuals)){
    if (medPolished$residuals[i,j]<0) {
      symbols(i,j,circles=abs(medPolished$residuals[i,j]/100),inches =
FALSE,add=T)
    }
    else {
      symbols(i,j,squares=abs(medPolished$residuals[i,j]/100),inches =
FALSE,add=T)}
  }
}
```




```r
#From Plot we can say that value in bottom left and upper right are
# negative, whereas for bottom right and upper left they are positive.

#c)
x<- vector()
y<- vector()
for(i in 1:length(medPolished$row)){
  for(j in 1:length(medPolished$col)){
    x<- c(x,(medPolished$row[i] * medPolished$col[j])/medPolished$overall)
  }

}


residuals<-vector()
for (i in 1:5){
  residuals<-c(residuals,medPolished$residuals[i,])
}
plot(x,residuals,xlab="Comparison Values",ylab="Residual
Values",main="Diagnostic plot")
abline(h=0,v=0)
fit<-lm(residuals~x)
abline(fit)
```



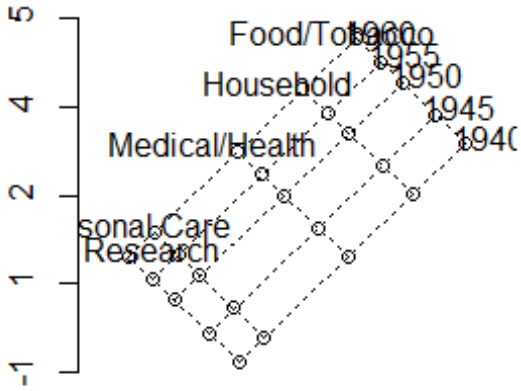

```
#d)

PETable.log<-log(personalExpenditureTable)
PETable.log<-matrix(PETable.log,c(5,5))
dimnames(PETable.log)=list(c("Food/Tobacco","Household","Medical/Health","Per
sonal Care","Educ / Research"),c("1940","1945","1950","1955","1960"))
PETable.log.MP <- medpolish(PETable.log)

## 1: 1.962294
## 2: 1.746089
## Final: 1.746089

MedianPolishdata<-rbind(PETable.log,PETable.log.MP$col)
MedianPolishdata<-cbind(MedianPolishdata,PETable.log.MP$row)

## Warning in cbind(MedianPolishdata, PETable.log.MP$row): number of rows of
## result is not a multiple of vector length (arg 2)

colnames(MedianPolishdata)[6]<-"Row Effect"
rownames(MedianPolishdata)[6]<-"Column Effect"
MedianPolishdata[6,6]<-medPolished$overall

#After transformation
MedianPolishdata

##                        1940        1945      1950      1955     1960
## Food/Tobacco     3.10009229  3.79548919 4.0876556 4.2931954 4.463607
## Household        2.35137526  2.74084002 3.3672958 3.5973123 3.832980
```

```
## Medical/Health   1.26129787  1.75093747 2.2731563 2.6390573 3.049273
## Personal Care    0.03922071  0.68309684 0.8960880 1.2237754 1.686399
## Educ / Research -0.44472582 -0.02634398 0.5877867 0.9555114 1.291984
## Column Effect   -0.98756329 -0.52221881 0.0000000 0.3276874 0.704197
##                 Row Effect
## Food/Tobacco     1.8144993
## Household        0.9964686
## Medical/Health   0.0000000
## Personal Care   -1.2909543
## Educ / Research -1.6853696
## Column Effect    9.7100000
```

```
sum_res<-sum(abs(PETable.log.MP$residual))
sum_data<-sum(abs(PETable.log-PETable.log.MP$overall))
Analogrsquare<-1-(sum_res/sum_data)
# AnalogRSquare after Transformation
Analogrsquare
```

```
## [1] 0.9444153
```

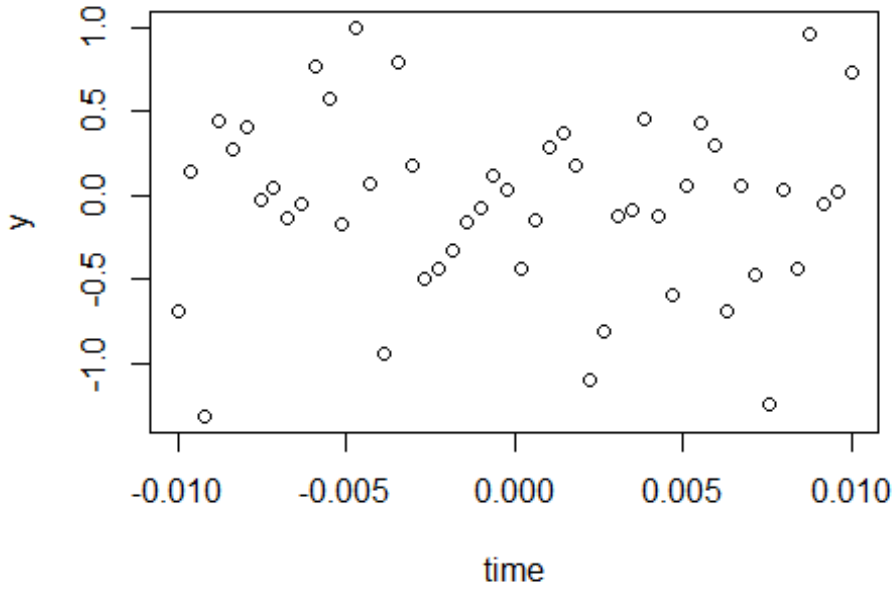
```
#e)
source('myplotfit.R')
myplotfit(PETable.log.MP)
```



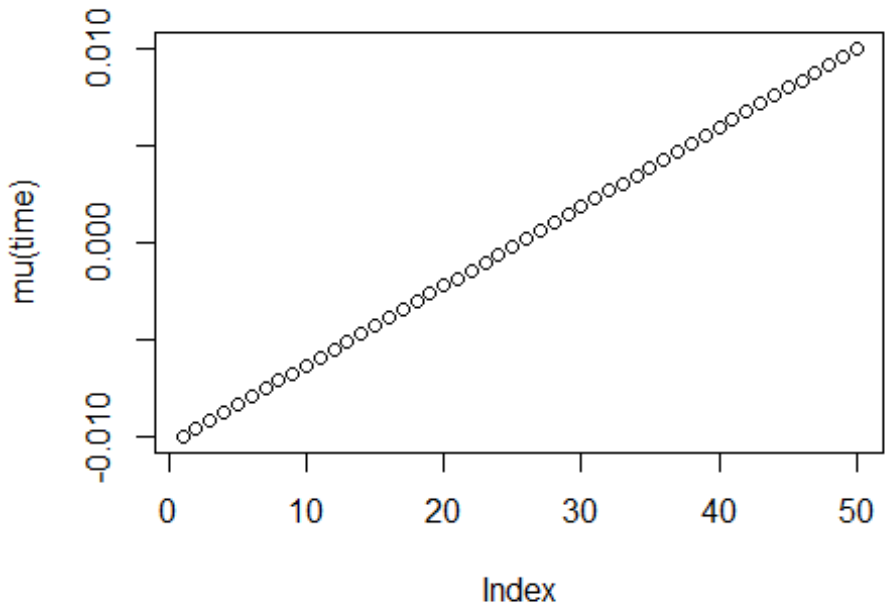

```
#Category has much larger effect than time because row effects are larger
#than column effects
```

Question 3

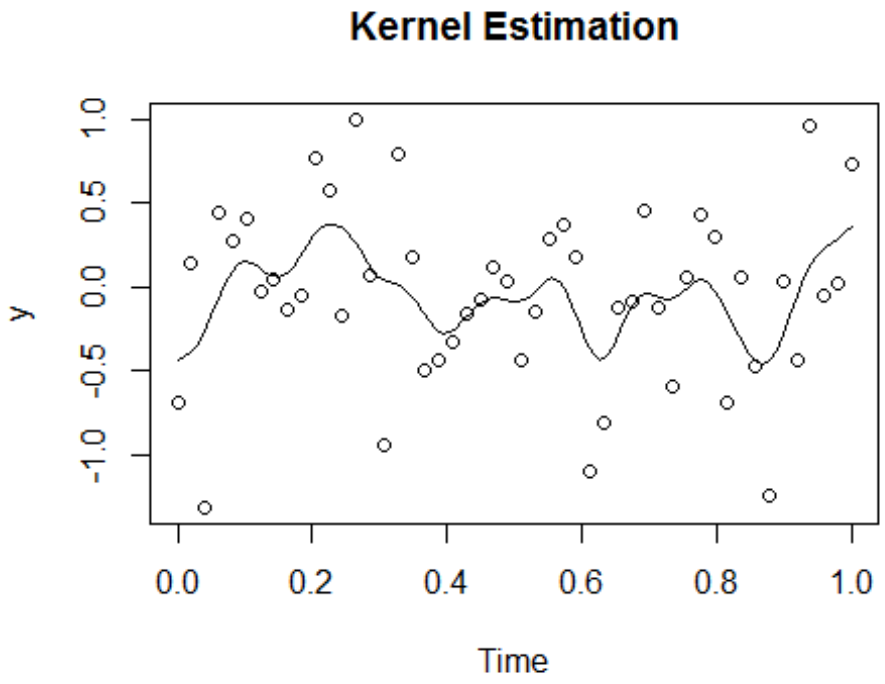
```r
#To find value of y, substitute i
y<- vector()
Mu<- expression(t + 0.5 * (exp(-50*(t-0.5)^2)))
mu<- function(dp){
  return (dp + 0.5 * (exp(-50*(dp-0.5)^2)))
}
x<- seq(0,1,length.out=50)
f<-function(t){ (2 * t - 1)/100}
time<- f(x)
t<-time
error <- rnorm(50,0,0.5)
y<-mu(time)+error
#y[i] <- mu(time) + error[i]
dataFrame<- cbind(time,y)
plot(dataFrame)
```




```r
plot(mu(time))
```




```
m2<-1
rk<-1/(2*sqrt(pi))
sigma2<-0.5^2
n<-50
ddut<-D(D(Mu , 't'),'t')
J2Mu <- function(t){( -(0.5 * (exp(-50 * ((t - 0.5)^2)) * (50 * 2) - exp(-50
* ((t -

0.5)^2)) * (50 * (2 * (t - 0.5))) * (50 * (2 * (t - 0.5))))))^2}
IntegrateJ2Mu <- integrate(J2Mu,lower=0,upper = 1)
J2muValue <- IntegrateJ2Mu$value

lamda_optimum<-(n^(-1/5))*(((sigma2)*rk)/(J2muValue*(m2^2)))^1/5
plot(x,y,main="Kernel Estimation",xlab="Time")
lines(ksmooth(x,y,kernel="normal",bandwidth=0.084))
```


Kernel Estimation
y
1.0
0.5
0.0
-0.5
-1.0
0.0
0.2
0.4
0.6
0.8
1.0
Time