LSM

Bojana Milošević

11/25/2019

Primer PCA

```
library(readr)
wdbc <- read_csv("breast-cancer-wisconsin_wdbc.csv")
#deo izlaza
prcomp(wdbc[c(3:32)], center = TRUE, scale = TRUE)
```

```
## Standard deviations (1, .., p=30):
##  [1] 3.64439401 2.38565601 1.67867477 1.40735229 1.28402903 1.09879780
##  [7] 0.82171778 0.69037464 0.64567392 0.59219377 0.54213992 0.51103950
## [13] 0.49128148 0.39624453 0.30681422 0.28260007 0.24371918 0.22938785
## [19] 0.22243559 0.17652026 0.17312681 0.16564843 0.15601550 0.13436892
## [25] 0.12442376 0.09043030 0.08306903 0.03986650 0.02736427 0.01153451
##
## Rotation (n x k) = (30 x 30):
##                                  PC1          PC2          PC3
## radius (nucA)            -0.21890244  0.233857132 -0.008531243
## texture (nucA)           -0.10372458  0.059706088  0.064549903
## perimeter (nucA)         -0.22753729  0.215181361 -0.009314220
## area (nucA)              -0.22099499  0.231076711  0.028699526
## smoothness (nucA)        -0.14258969 -0.186113023 -0.104291904
## compactness (nucA)       -0.23928535 -0.151891610 -0.074091571
## concavity (nucA)         -0.25840048 -0.060165363  0.002733838
## concave points (nucA)    -0.26085376  0.034767500 -0.025563541
## symmetry (nucA)          -0.13816696 -0.190348770 -0.040239936
## fractal dimension (nucA) -0.06436335 -0.366575471 -0.022574090
## radius (nucB)            -0.20597878  0.105552152  0.268481387
## texture (nucB)           -0.01742803 -0.089979682  0.374633665
## perimeter (nucB)         -0.21132592  0.089457234  0.266645367
## area (nucB)              -0.20286964  0.152292628  0.216006528
## smoothness (nucB)        -0.01453145 -0.204430453  0.308838979
## compactness (nucB)       -0.17039345 -0.232715896  0.154779718
## concavity (nucB)         -0.15358979 -0.197207283  0.176463743
## concave points (nucB)    -0.18341740 -0.130321560  0.224657567
```

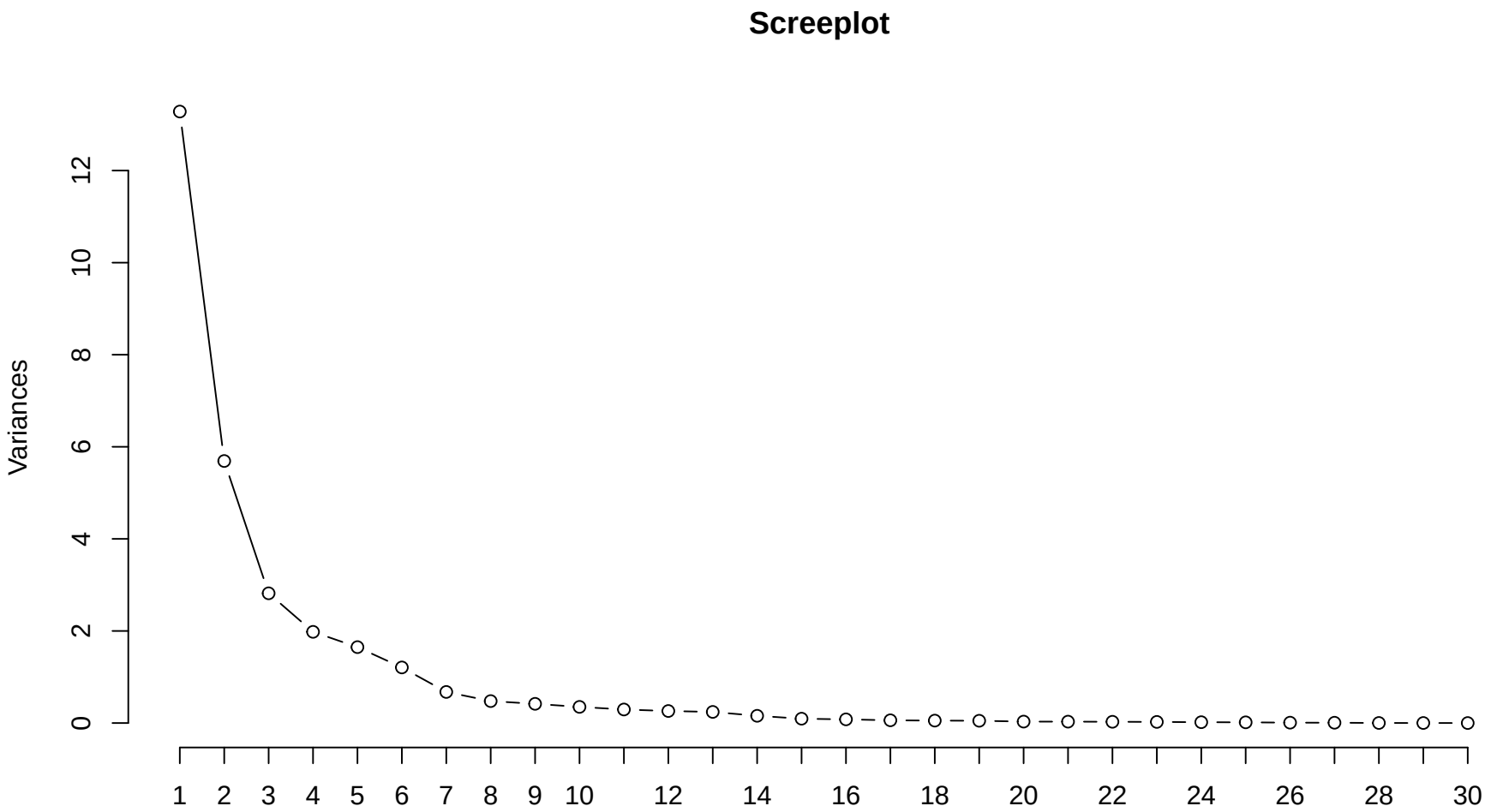
Primer

```
wdbc.pr <- prcomp(wdbc[c(3:32)], center = TRUE, scale = TRUE)
summary(wdbc.pr)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6
## Standard deviation     3.6444 2.3857 1.67867 1.40735 1.28403 1.09880
## Proportion of Variance 0.4427 0.1897 0.09393 0.06602 0.05496 0.04025
## Cumulative Proportion  0.4427 0.6324 0.72636 0.79239 0.84734 0.88759
##                           PC7     PC8    PC9    PC10   PC11    PC12
## Standard deviation     0.82172 0.69037 0.6457 0.59219 0.5421 0.51104
## Proportion of Variance 0.02251 0.01589 0.0139 0.01169 0.0098 0.00871
## Cumulative Proportion  0.91010 0.92598 0.9399 0.95157 0.9614 0.97007
##                           PC13    PC14    PC15    PC16    PC17    PC18
## Standard deviation     0.49128 0.39624 0.30681 0.28260 0.24372 0.22939
## Proportion of Variance 0.00805 0.00523 0.00314 0.00266 0.00198 0.00175
## Cumulative Proportion  0.97812 0.98335 0.98649 0.98915 0.99113 0.99288
##                           PC19    PC20   PC21    PC22    PC23   PC24
## Standard deviation     0.22244 0.17652 0.1731 0.16565 0.15602 0.1344
## Proportion of Variance 0.00165 0.00104 0.0010 0.00091 0.00081 0.0006
## Cumulative Proportion  0.99453 0.99557 0.9966 0.99749 0.99830 0.9989
##                           PC25    PC26    PC27    PC28    PC29    PC30
## Standard deviation     0.12442 0.09043 0.08307 0.03987 0.02736 0.01153
## Proportion of Variance 0.00052 0.00027 0.00023 0.00005 0.00002 0.00000
## Cumulative Proportion  0.99942 0.99969 0.99992 0.99997 1.00000 1.00000
```

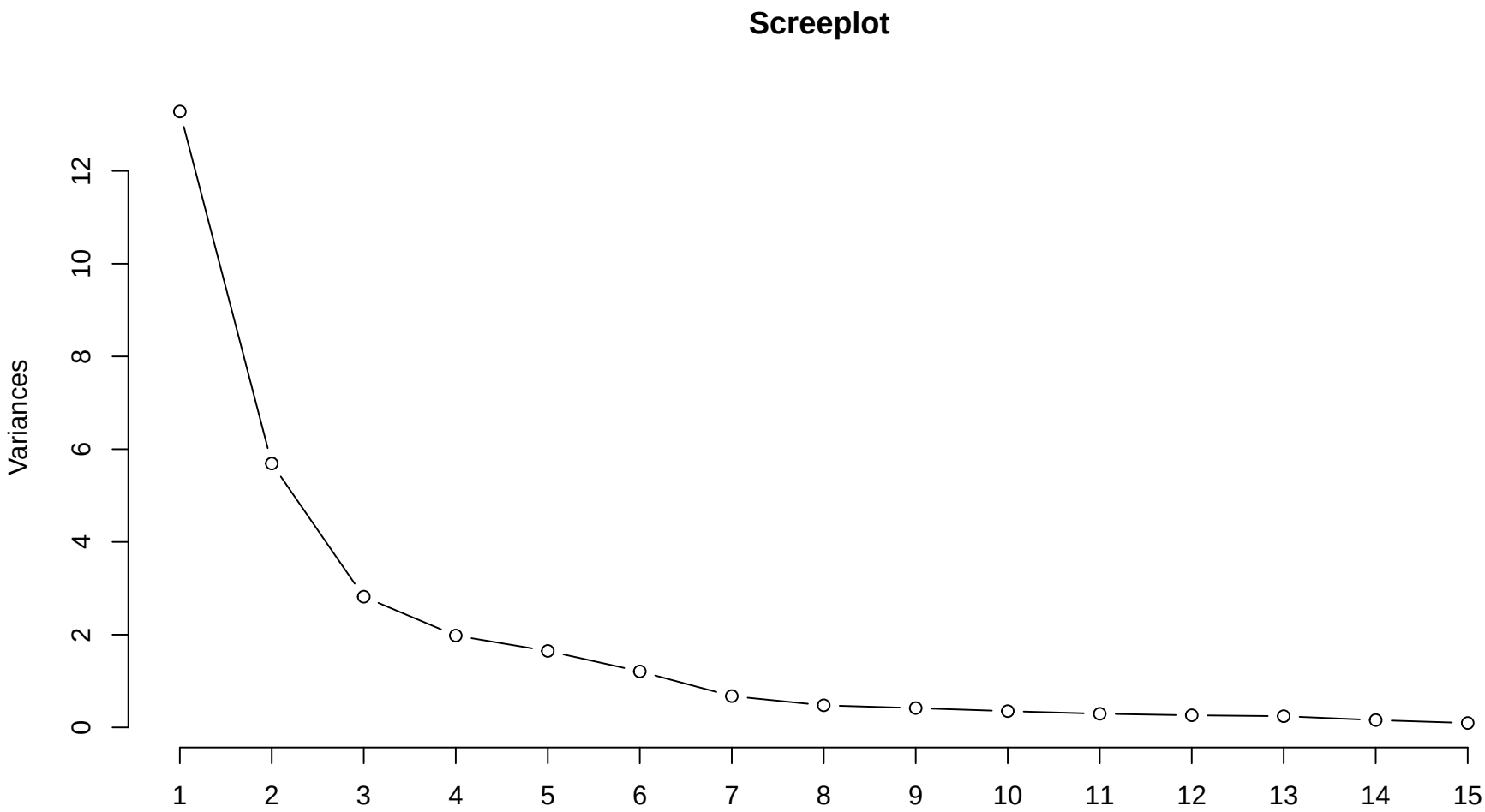
Primer

```
screeplot(wdbc.pr, type = "l", npcs = 30, main = "Screeplot ")
```



Primer

```
screeplot(wdbc.pr, type = "l", npcs = 15, main = "Screeplot ")
```

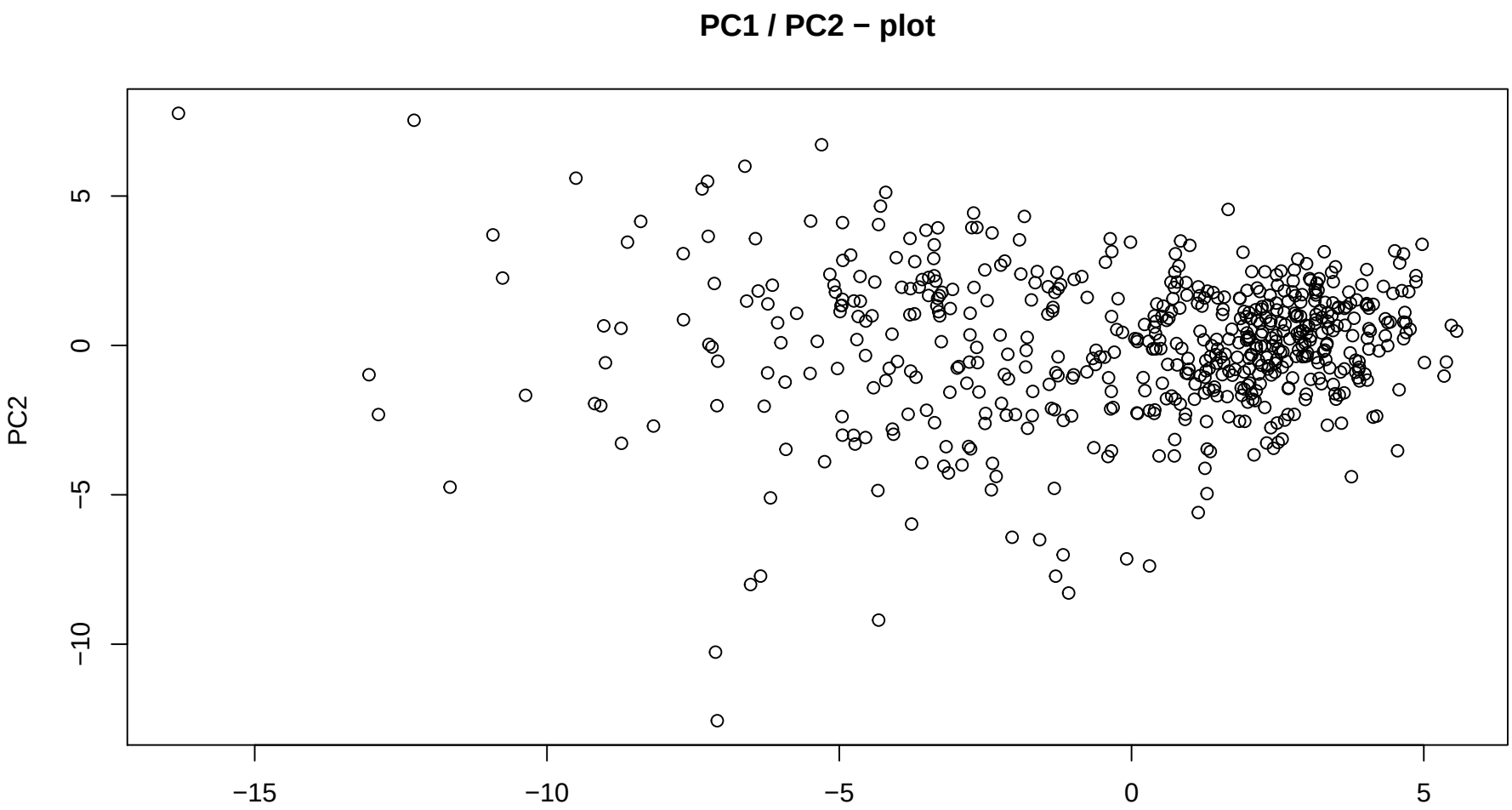


Primer

Imajući u vidu rezultate analize glavnih komponenti, najbolje bi bilo da zadržimo prvih 6 komponenti u daljoj analizi. Kako imamo kategoričku zavisnu promenljivu primer ćemo nastaviti nakon prikaza logističke regresije

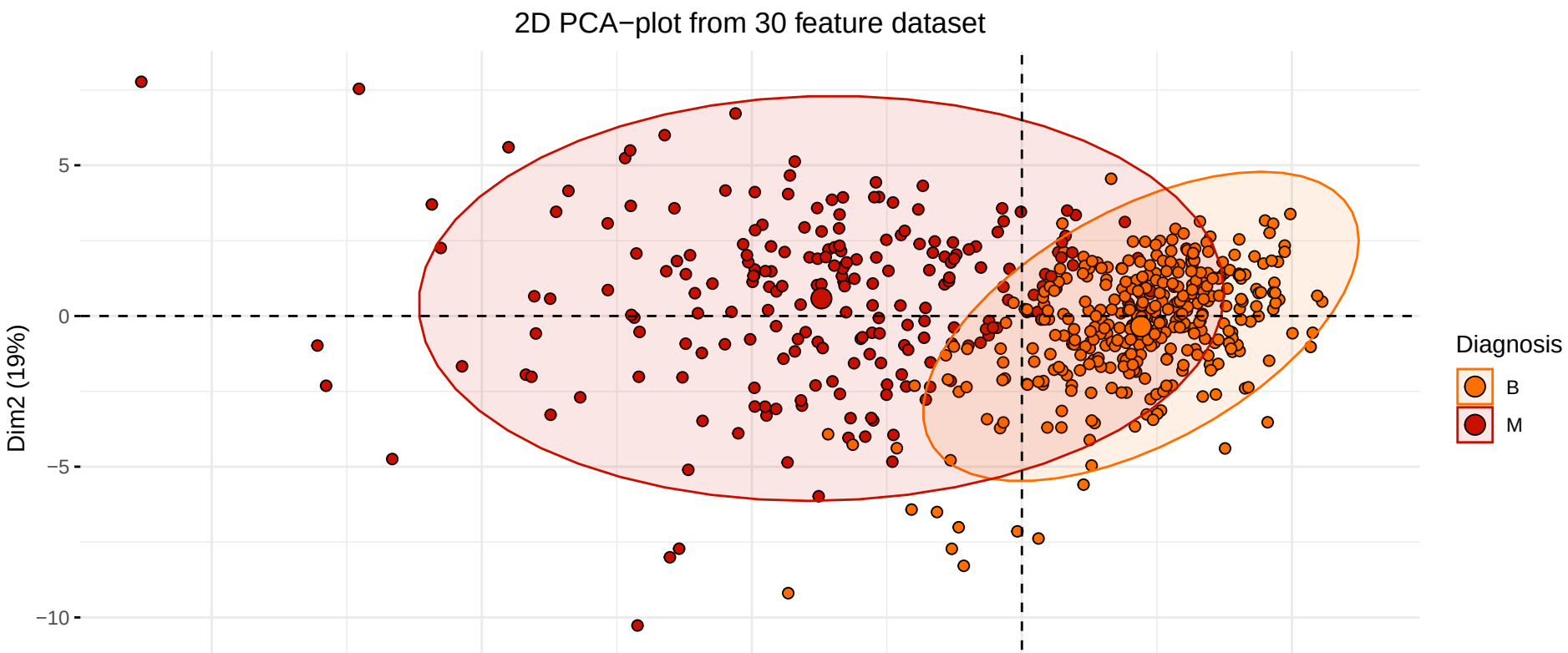
Primer- Prve dve komponente

```
plot(wdbc.pr$x[,1],wdbc.pr$x[,2], xlab="PC1 ", ylab = "PC2 ",
     main = "PC1 / PC2 - plot")
```



Primer- grupisanje na osnovu vrednosti prve dve komponente

```
library("factoextra")
fviz_pca_ind(wdbc.pr, geom.ind = "point", pointshape = 21,
             pointsize = 2,
             fill.ind = wdbc$`diagnosis (M=malignant; B=benign)`,
             palette = "futurama", addEllipses = TRUE,
             legend.title = "Diagnosis") +
  ggtitle("2D PCA-plot from 30 feature dataset") +
  theme(plot.title = element_text(hjust = 0.5))
```



Primer (sa podeljenim skupom podataka)

```
set.seed(10)
nrow(wdbc)
```

```
## [1] 569
```

```
indeksi<-sample(1:569,size=560,replace=FALSE)

wdbcS=wdbc[indeksi,]
wdbcN=wdbc[-indeksi,]
wdbc.prS <- prcomp(wdbcS[c(3:32)], center =TRUE, scale = TRUE)
summary(wdbc.prS)
```

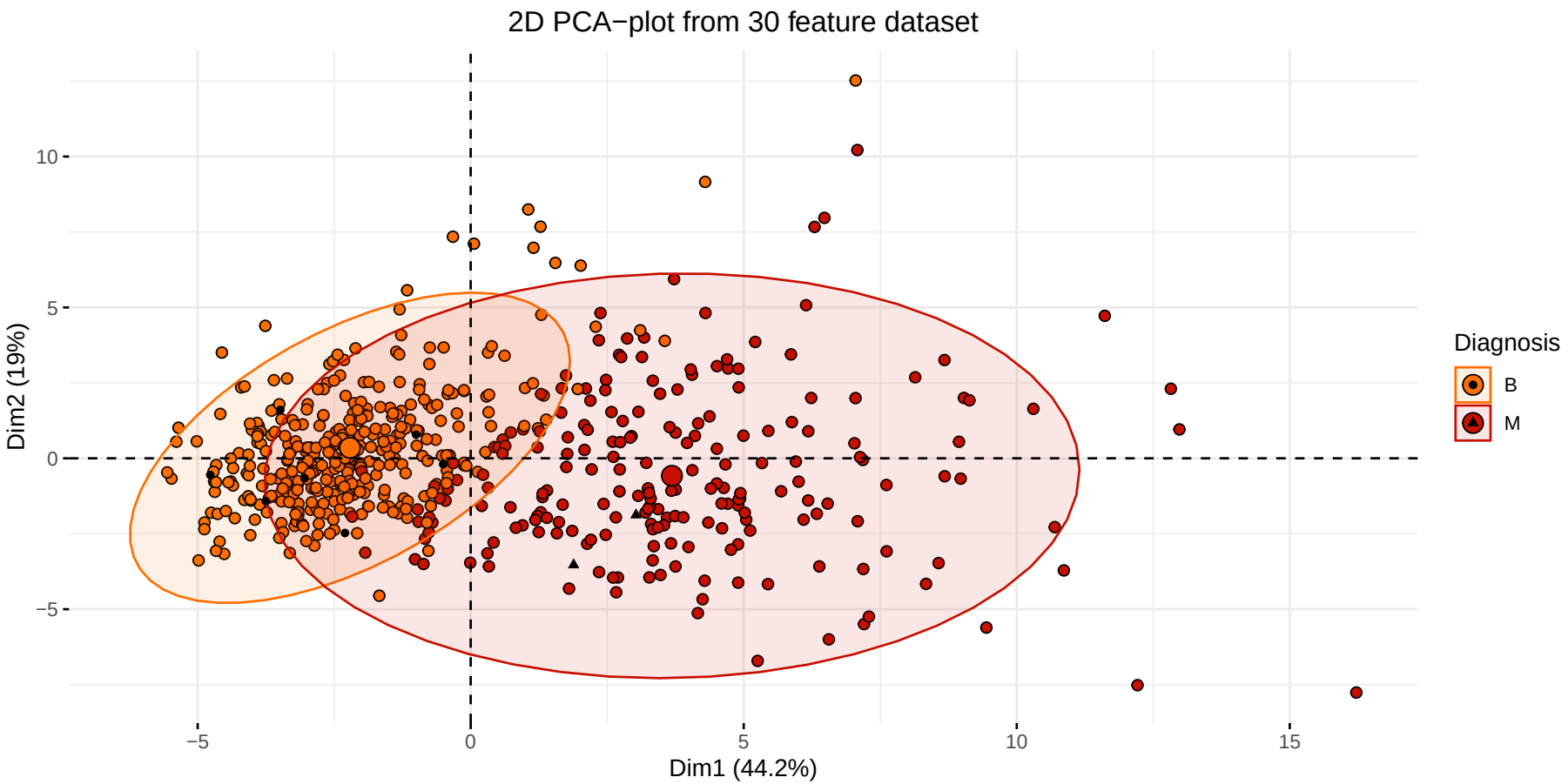
```
## Importance of components:
##                           PC1    PC2     PC3     PC4    PC5     PC6
## Standard deviation     3.6425 2.3879 1.67493 1.41057 1.2869 1.10114
## Proportion of Variance 0.4423 0.1901 0.09351 0.06632 0.0552 0.04042
## Cumulative Proportion  0.4423 0.6323 0.72583 0.79216 0.8474 0.88778
##                           PC7     PC8     PC9    PC10    PC11    PC12
## Standard deviation     0.82261 0.69106 0.64370 0.59039 0.53921 0.51058
## Proportion of Variance 0.02256 0.01592 0.01381 0.01162 0.00969 0.00869
## Cumulative Proportion  0.91033 0.92625 0.94006 0.95168 0.96138 0.97006
##                          PC13    PC14    PC15    PC16    PC17    PC18
## Standard deviation     0.4928 0.39523 0.30738 0.28116 0.24457 0.23045
## Proportion of Variance 0.0081 0.00521 0.00315 0.00264 0.00199 0.00177
## Cumulative Proportion  0.9782 0.98337 0.98652 0.98915 0.99115 0.99292
##                           PC19    PC20   PC21    PC22    PC23   PC24
## Standard deviation     0.22040 0.17648 0.1731 0.16601 0.15587 0.1346
## Proportion of Variance 0.00162 0.00104 0.0010 0.00092 0.00081 0.0006
## Cumulative Proportion  0.99453 0.99557 0.9966 0.99749 0.99830 0.9989
##                           PC25    PC26    PC27    PC28    PC29    PC30
## Standard deviation     0.12430 0.09035 0.08239 0.03993 0.02737 0.01158
## Proportion of Variance 0.00052 0.00027 0.00023 0.00005 0.00002 0.00000
```

Primer

```
wdbcNSkalirano=scale(wdbcN[c(3:32)],center=wdbc.prS$center,scale=wdbc.prS$scale)
noviPodaci=data.frame(cbind(data.frame(wdbcNSkalirano%*%wdbc.prS$rotation,wdbcN[,2])[,1:2],wdbcN[,2]))
noviPodaci[,3]=as.factor(noviPodaci[,3])
names(noviPodaci)[3]='diagnosis'
```

Primer

```
fviz_pca_ind(wdbc.prS, geom.ind = "point", pointshape = 21,
             pointsize = 2,
             fill.ind = wdbcS$‘diagnosis (M=malignant; B=benign)‘,
             palette = "futurama", addEllipses = TRUE,
             legend.title = "Diagnosis") +
  ggtitle("2D PCA-plot from 30 feature dataset") +
  theme(plot.title = element_text(hjust = 0.5))+geom_point(data=noviPodaci,aes(x=PC1,y=PC2,pch=diagnosis))
```



Metode regularizacije-Nazubljena regresija

$$\hat{\delta}_c = (X^T X + cI)^{-1} X^T Y. \tag{1}$$

Konstanta c je mali pozitivan broj koji utiče, na pristrasnost ocene, a sa druge povećava stabilnost ocena. Ovaj metod se ne mora primeniti na centriran model. Može se pokazati da je

$$\hat{\delta}_c = (c(X^T X)^{-1} + I)^{-1} \hat{\delta}$$

Osobine ocene

$$E(\hat{\delta}_c) = (X^T X + cI)^{-1}(X^T X + cI - cI)\delta = \delta - c(XX^T + cI)^{-1}\delta.$$

$$Cov(\hat{\delta}_c) = (X^T X + cI)^{-1} X^T X (X^T X + cI)^{-1}\sigma^2.$$

$$tr((X^T X + cI)^{-1} X^T X (X^T X + cI)^{-1}\sigma^2) = \sigma^2 tr((X^T X + cI)^{-2} X^T X)$$

$$= \sigma^2 \sum_{i=1}^{p} \frac{1}{\lambda_i (1 + c\lambda_i^{-1})^2}$$

To je manje od sume disperzija pojedinačnih komponenti ocene metodom najmanjih kvadrata $\sigma^2 \sum_{j=1}^{p} \frac{1}{\lambda_j}$.

Interpretacija parametra c

Drugi način da se ova ocena interpretira je ona zapravo predstavlja ocenu najmanjih kvadrata kada za parametre postoji gornje ograničenje, odnosno rešava se problem

$$\min_{\delta}(y_0 - X\delta)^T(y_0 - X\delta), \quad ||\delta||_2 \leq C,$$

za neko C koje se može izraziti u funkciji od c. Taj problem je ekvivalentan traženju minimuma funkcije

$$(y_0 - X\delta)^T(y_0 - X\delta) + c||\delta||_2$$

Važno

Ova metoda se ne može koristiti za selekciju prediktora

Lasso regresija

- Tibširani 1996.

$$\hat{\delta}_l = \arg\min_{\delta}(y_0 - X\delta)^T(y_0 - X\delta) + c\sum_{j=1}^{p}|\delta_j|$$

Ova metoda se može koristiti za selekciju prediktora

Uopštenje

$$\hat{\delta}_l = \arg\min_{\delta}(y_0 - X\delta)^T(y_0 - X\delta) + c\sum_{j=1}^{p}|\delta_j|^q$$

Sličnosti i razlike

- Nazubljena regresija daje stabilnije ocene
- Sa Lasso regresijom možemo izvršiti selekciju prediktora, ali ona nije uvek stabilna
- U slučaju Lasso regresije nemamo analitički izraz za ocenu β pa ne možemo naći tačnu pristrasnost
- U oba slučaja radi se o konveksnoj optimizaciji

Odabir parametra c

- Videti kako zavise ocene od vrednosti c i odabrati ono za koje se stabilizuje grafik
- Krosvalidacija

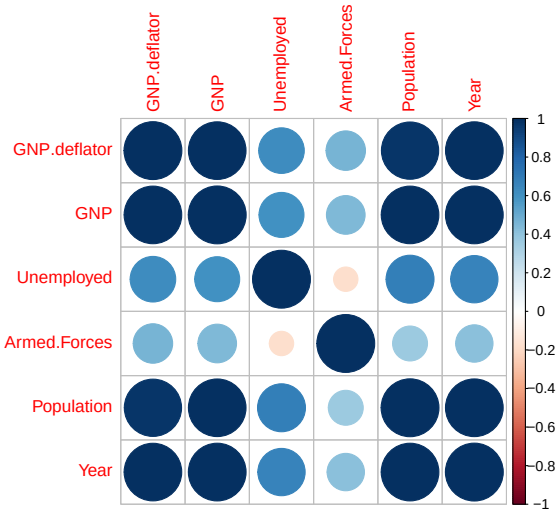
Primer- Longley

Ekonomske promenljive, posmatrane godišnje u periodu od 1947 do 1962 (n=16).

- GNP.deflator GNP implicit price deflator (1954=100)
- GNP Gross National Product.
- Unemployed number of unemployed.
- Armed.Forces number of people in the armed forces.
- Population noninstitutionalized population $\geq$ 14 years of age.
- Year the year (time).
- Employed number of people employed.

Primer-Longley

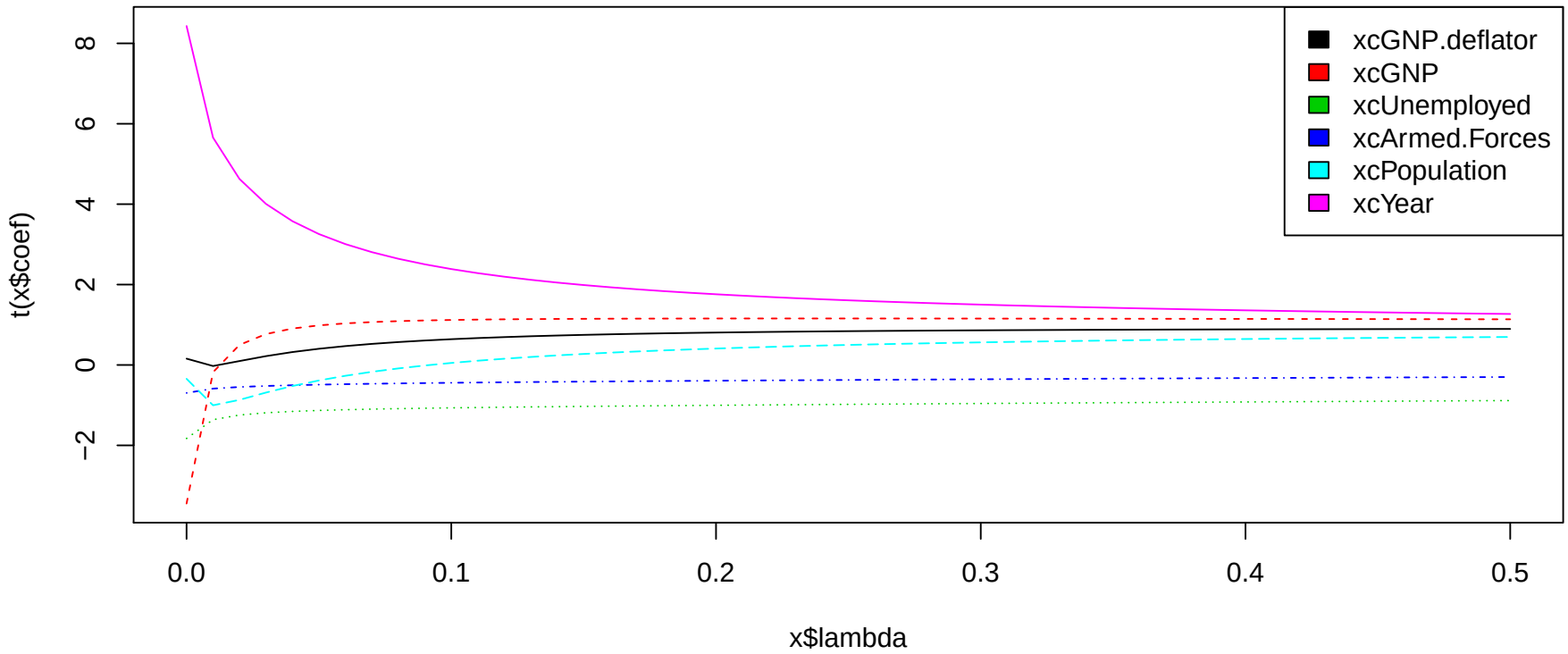
```
x <- as.matrix(longley[,1:6])
y <- as.matrix(longley[,7])
xc=scale(x,center=TRUE,scale=TRUE)
yc=y-mean(y)
library("corrplot")
corrplot(cor(xc))
```




```
MB=lm(yc~xc-1)
```

Primer-Longley

```
library('lars')
library('MASS')
fit.ridge=lm.ridge(yc~xc-1,lambda=seq(from=0,to=0.5,by=0.01))
plot(fit.ridge,lwd=4,col=1:6)
legend("topright",legend=rownames(fit.ridge$coef),
       fill=1:6)
```



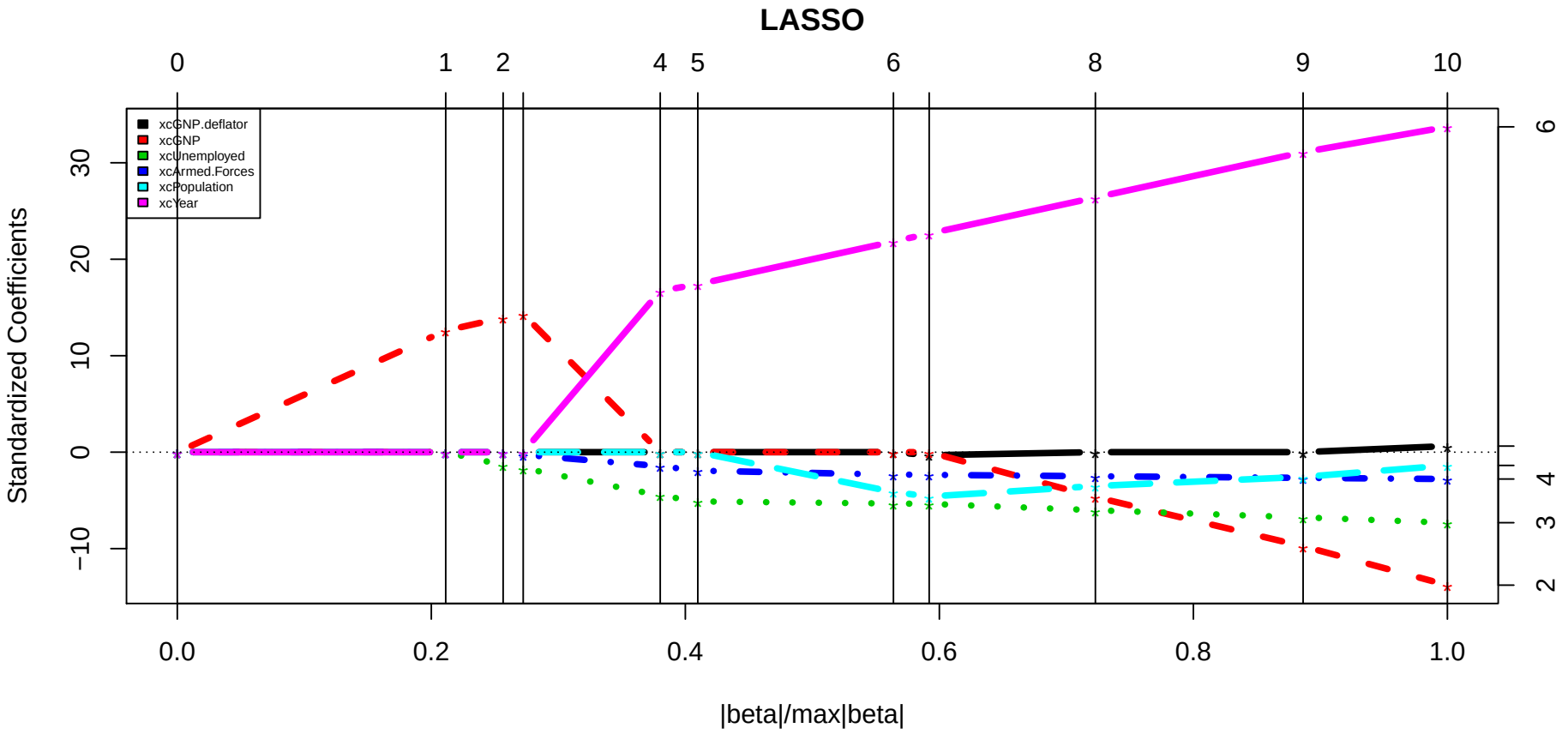
Primer-Longley

```
#GCV je ocena dobijena krosvalidacijom
#Golub, G. H., Heath, M., & Wahba, G. (1979).
#Generalized cross-validation as a method for choosing a good ridge
select(lm.ridge(yc~xc-1,lambda=seq(from=0,to=0.5,by=0.01)))
```

```
## modified HKB estimator is 0.003847822
## modified L-W estimator is 0.02906578
## smallest value of GCV  at 0
```

Primer- Longley

```
fit.lasso=lars(x=xc,y=yc,type="lasso")
plot(fit.lasso,lwd=4)
legend("topleft",legend=rownames(fit.ridge$coef),fill=1:6,cex=0.5)
```



na

x-osi je odnos norme u slucaju regularizacije i maksimalne L1 norme (u slučaju ONK) # Primer- Longley

```
fit.lasso=lars(x=xc,y=yc,type="lasso")
coef.lars(fit.lasso,s=0,mode="lambda")
```

Primer- Longley

```
rmse<-function (x, y) sqrt (mean ( (x-y)^2) )
fitRIDGE=lm.ridge(yc~xc-1,lambda=0.03)
yR=mean(y)+xc%*%matrix(fitRIDGE$coef,ncol=1)
rmse(yR,y)
```

```
## [1] 0.3071134
```

```
coef.P=coef.lars(fit.lasso,s=0.3,mode="lambda")
yLassoC=xc%*%matrix(coef.P,ncol=1)+mean(y)
rmse(yLassoC,y)
```

```
## [1] 0.5021149
```

```
coef.P=coef.lars(fit.lasso,s=0.03,mode="lambda")
yLassoC=xc%*%matrix(coef.P,ncol=1)+mean(y)
rmse(yLassoC,y)
```

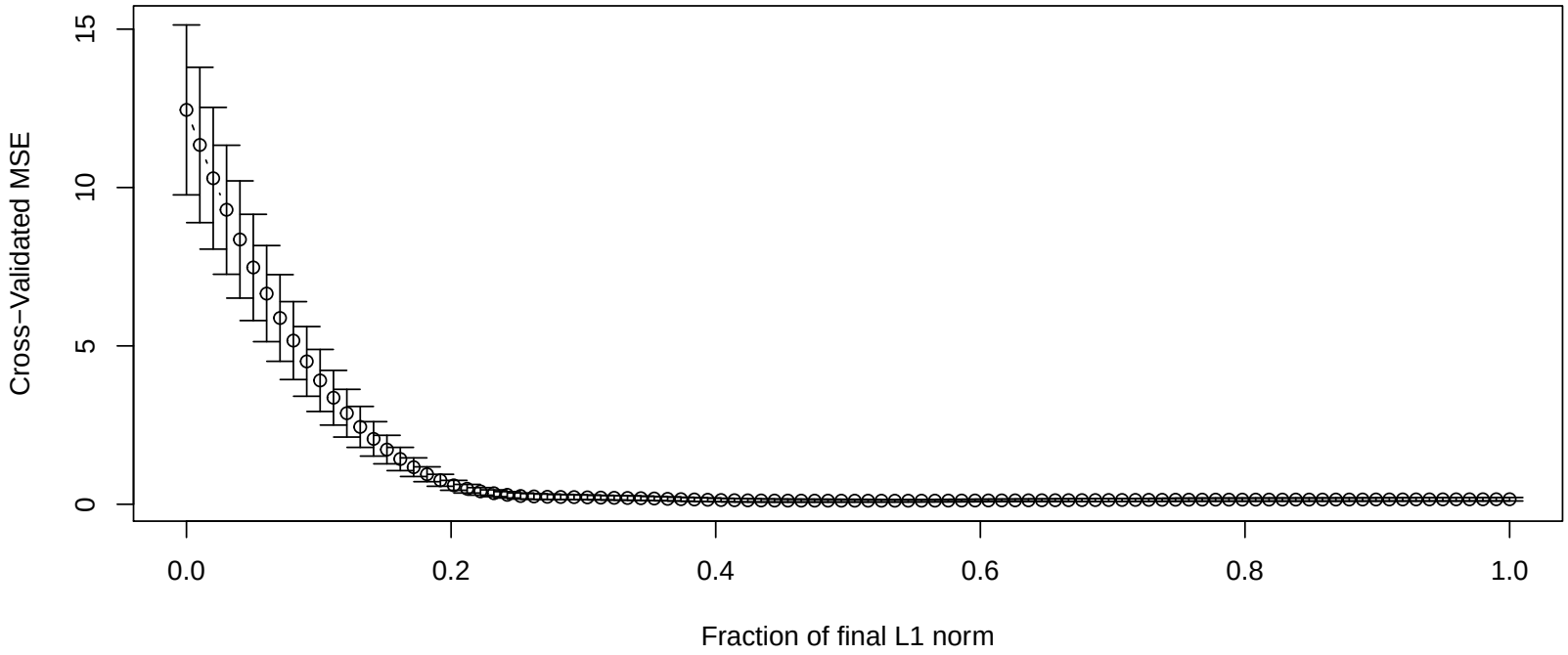
```
## [1] 0.2898697
```

```
coef.P=coef.lars(fit.lasso,s=0,mode="lambda")
yLassoC=xc%*%matrix(coef.P,ncol=1)+mean(y)
rmse(yLassoC,y)
```

```
## [1] 0.2286406
```

Primer-Longley

```
cv.lars(x=xc,y=yc,intercept=FALSE,type="lasso",plot.it = TRUE)
```




```
#nakon s=0.2 nije primetan pad u MSE
coef.P=coef.lars(fit.lasso,s=0.2,mode="lambda")
yLassoC=xc%*%matrix(coef.P,ncol=1)+mean(y)
rmse(yLassoC,y)
```

```
## [1] 0.4680767
```