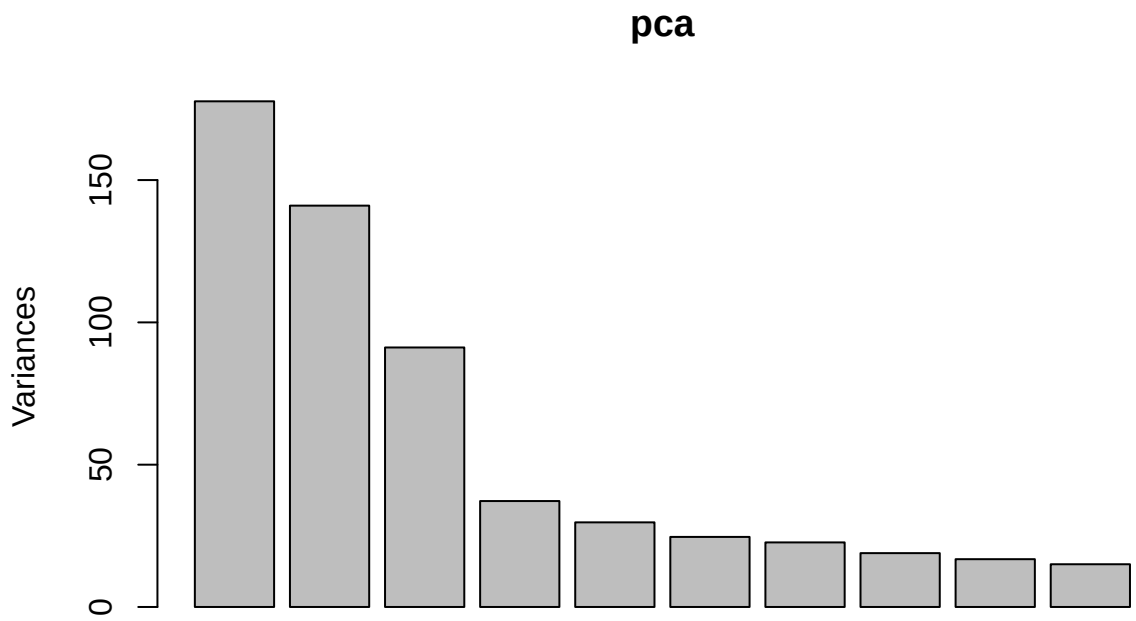
Logistic regression based on PC scores

```
load(file = "analyticData.rda")
analyticData = analyticData %>% select(-permth_exm)
# NA: alive
# 1: deceased
analyticData$mortstat = ifelse(analyticData$mortstat %>% is.na,1,0)
# 1: alive, 0 : deceased

pca = prcomp(analyticData %>% select(-SEQN,-mortstat) %>% na.omit() ,
             center = T,
             scale. = T)

if(F){
  save(pca,file = 'pca.rda')
}
screeplot(pca)
```




```
pcscore = data.frame(SEQN = analyticData %>% na.omit %>% select(SEQN),
                     pca$x,
                     mortstat = analyticData %>% na.omit %>% select(mortstat))
if(F){
  save(pcscore,file = 'pcscore.rda')
}
# first 5 PCs
y = pcscore[,c(2:6,which(colnames(pcscore) == 'mortstat'))]
```

```r
y$mortstat = as.factor(y$mortstat)
set.seed(100)
# trainIdx = sample(c(TRUE, FALSE), dim(y)[1], replace = TRUE, prob = c(.7, .3))
trainIdx = sample(dim(y)[1],0.7*dim(y)[1])
fit = glm(mortstat ~ ., family = "binomial", data = y, subset = trainIdx)
summary(fit)
```

```
##
## Call:
## glm(formula = mortstat ~ ., family = "binomial", data = y, subset = trainIdx)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.0870   0.1862   0.3769   0.5671   1.1903
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.492502   0.087984  28.329  < 2e-16 ***
## PC1          0.094210   0.006781  13.893  < 2e-16 ***
## PC2         -0.052478   0.009802  -5.354 8.62e-08 ***
## PC3         -0.067613   0.010009  -6.756 1.42e-11 ***
## PC4         -0.034857   0.016337  -2.134   0.0329 *
## PC5          0.010609   0.015775   0.673   0.5013
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2236.5  on 2924  degrees of freedom
## Residual deviance: 1916.7  on 2919  degrees of freedom
## AIC: 1928.7
##
## Number of Fisher Scoring iterations: 6
```

```r
yPred =  (predict(fit, y[-trainIdx,], type = "response") > 0.5) * 1
ytest = y[-trainIdx, ]
ptab = table(ytest[,"mortstat"], yPred %>% factor(levels = levels(ytest[,"mortstat"] )))
# result
ptab
```

```
##
##        0    1
##   0    0  171
##   1    1 1082
```

```r
# acc
sum(diag(ptab)) / sum(ptab)
```

```
## [1] 0.8628389
```

BMI prediction based on raw data using lm

```r
load(file = 'analyticData.rda')
analyticData = analyticData %>% select(-mortstat,-permth_exm) %>%
  inner_join(Covariate_D %>% select(SEQN,BMI),by = "SEQN")
```

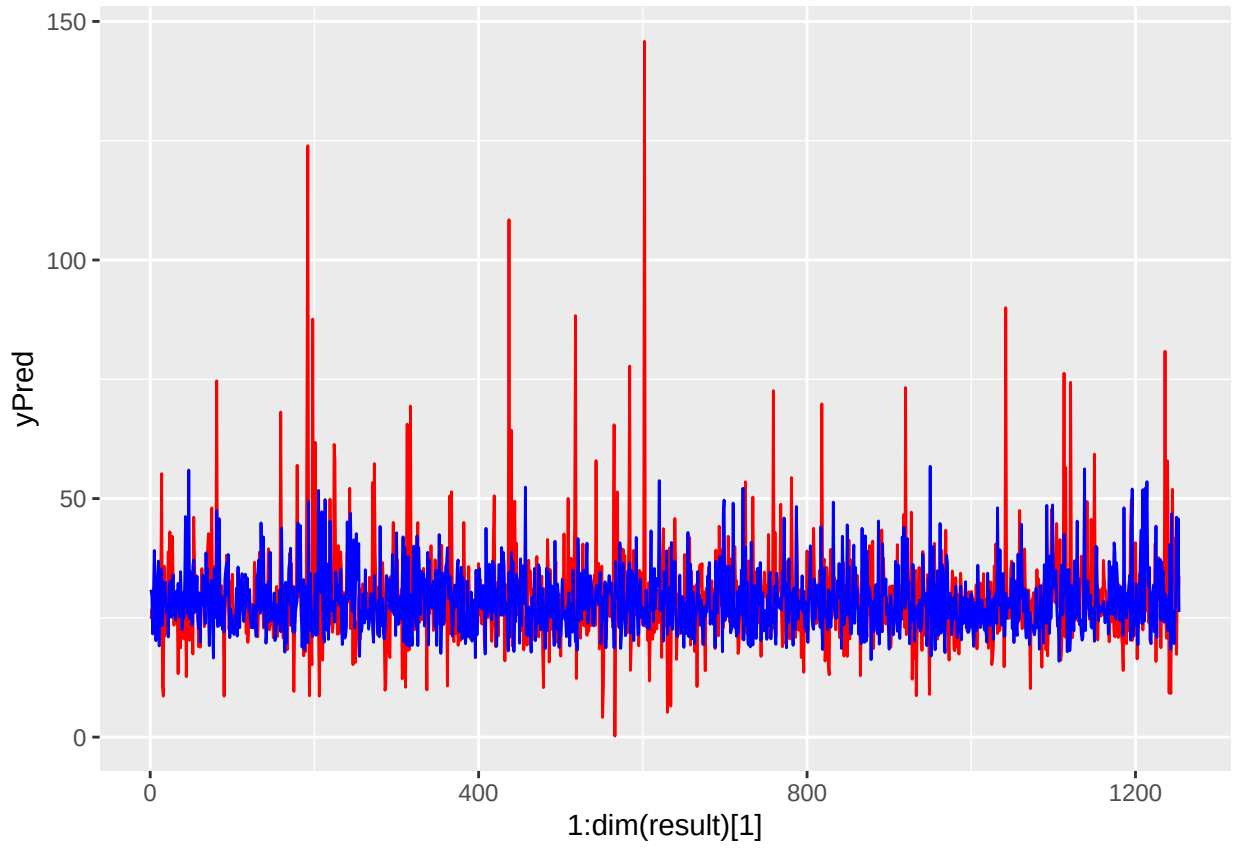
```r
analyticData$'log(BMI+1)' = log(analyticData$BMI+1)


y = analyticData %>% select(-SEQN)


set.seed(100)
BMI = y$BMI
y = y %>% select(-BMI)
trainIdx = sample(nrow(y),0.7*nrow(y))

fit = lm( y$`log(BMI+1)` ~  ., data = y, subset = trainIdx)
# summary(fit)
yPred = exp(predict(fit,y[-trainIdx,]))-1
result = cbind(yPred,yTrue = BMI[-trainIdx]) %>% na.omit() %>% as.data.frame()
modelmse = mean(summary(fit)$residuals^2)
# model MSE
modelmse
```

```
## [1] 0.02202508
```

```r
# MSE of yPred and yTrue
mean((result[,1]-result[,2])^2)
```

```
## [1] 151.2781
```

```r
# visualization
library(ggplot2)
ggplot(data=result , aes(x = 1:dim(result)[1])) +
  geom_line(aes(y = yPred),color = 'red') +
  geom_line(aes(y = yTrue),color = 'blue')
```



BMI prediction based on PC scores using lm

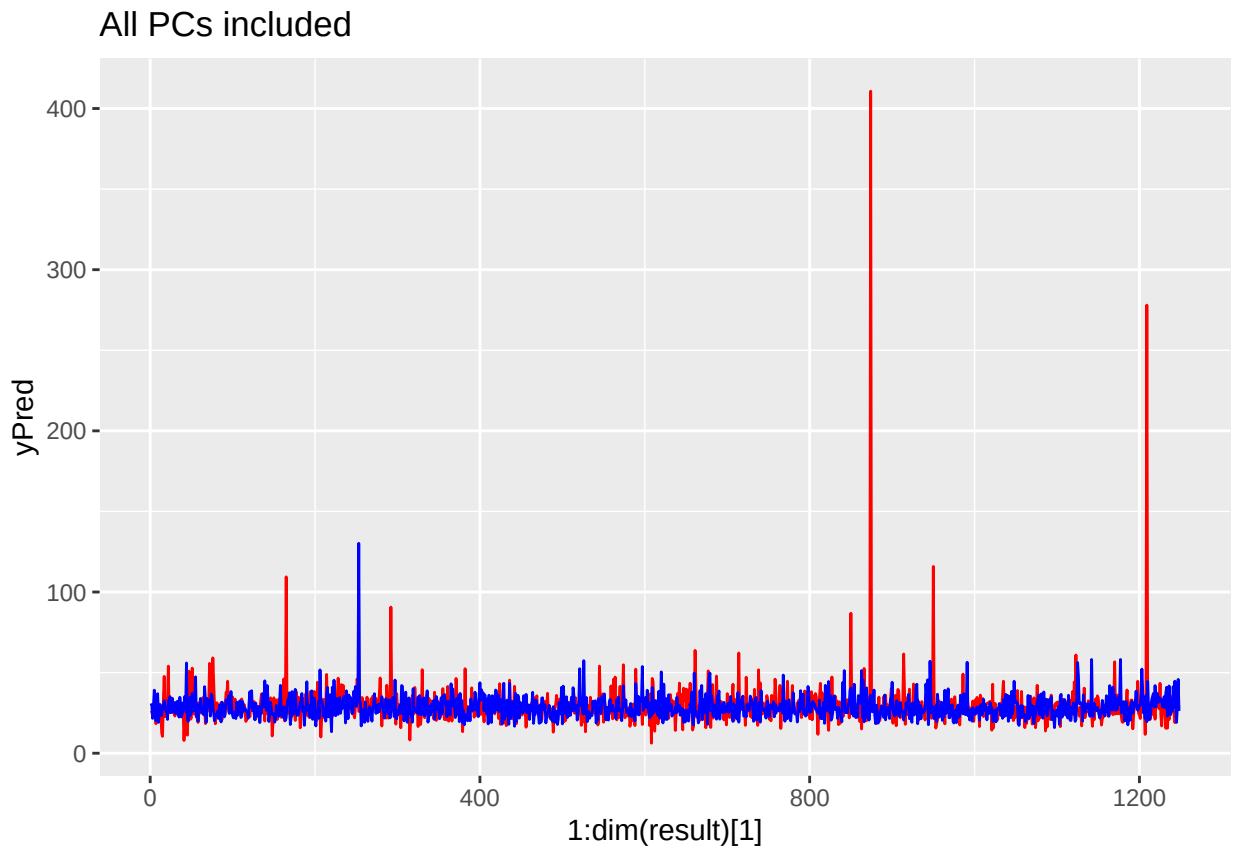
```
load(file = 'pcscore.rda')

pcscore = pcscore %>% select(-mortstat) %>%
  inner_join(Covariate_D %>% select(SEQN,BMI),by = "SEQN")

y = pcscore %>% select(-SEQN)



BMI = y$BMI
y = y %>% select(-BMI) %>% mutate('log(BMI+1)' = log(BMI+1))
set.seed(100)
trainIdx = sample(nrow(y),0.7*nrow(y))

fit = lm( y$`log(BMI+1)` ~  ., data = y, subset = trainIdx)
# summary(fit)
yPred = exp(predict(fit,y[-trainIdx,]))-1
result = cbind(yPred,yTrue = BMI[-trainIdx]) %>% na.omit() %>% as.data.frame()
modelmse = mean(summary(fit)$residuals^2)
# model MSE
modelmse
```

```
## [1] 0.02209232
```

```r
# MSE of yPred and yTrue
mean((result[,1]-result[,2])^2)
```

```
## [1] 281.5869
```

```r
# visualization
library(ggplot2)
ggplot(data=result , aes(x = 1:dim(result)[1])) +
  geom_line(aes(y = yPred),color = 'red') +
  geom_line(aes(y = yTrue),color = 'blue') +
  labs(title = "All PCs included")
```




```r
ysub = y[,c(1:100,which(colnames(y) == 'log(BMI+1)'))]
fit = lm( ysub$`log(BMI+1)` ~., data = ysub, subset = trainIdx)
# summary(fit)
yPred = exp(predict(fit,ysub[-trainIdx,]))-1
result = cbind(yPred,yTrue = BMI[-trainIdx]) %>% na.omit() %>% as.data.frame()
modelmse = mean(summary(fit)$residuals^2)
# model MSE
modelmse
```

```
## [1] 0.04203967
```

```r
# MSE of yPred and yTrue
mean((result[,1]-result[,2])^2)
```

```
## [1] 51.59289
```

```r
# visualization
library(ggplot2)
ggplot(data=result , aes(x = 1:dim(result)[1])) +
  geom_line(aes(y = yPred),color = 'red') +
  geom_line(aes(y = yTrue),color = 'blue') +
  labs(title  = "First 100 PCs included")
```