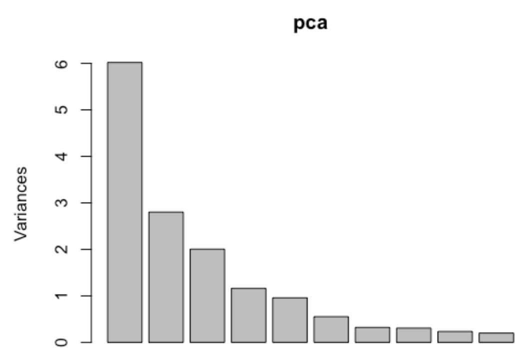
9.1

 # clear environment, load dataset

```
rm(list = ls())
set.seed(1)
dat <- read.table("uscrime.txt", stringsAsFactors = FALSE, header = TRUE)
```

 # using PCA (prcomp function) and scaling data; first goal is identifying the first "few" principal components to make our regression model and then compare that with the model of the total original variables

```
pca = prcomp(x=dat[,1:15], scale=TRUE)
plot(pca)
```

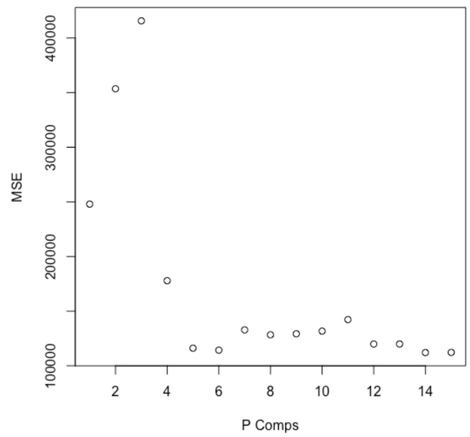


 # as perhaps expected, the most variance comes in the first principal variable; the variance seems to drastically reduce by the sixth or seventh variable; so we can target that area

 # another method may be brought in to determine the number of Principal Components that will be best to use; one idea is to use cross validation (20-fold as the option I went with) to see how "few" Principal Components to use

```
cv_res <- NULL
for (n_pca_comp in 1:ncol(pca$x)){
  pca_df <- as.data.frame(cbind(pca$x[,1:n_pca_comp], Crime=dat[,16]))
  cv <- cv.lm(pca_df, form.lm=Crime~., m=20, printit=FALSE)
  mean_squared_error <- attr(cv, 'ms')
  cv_res <- rbind(cv_res,data.frame(n_pca_comp, mean_squared_error))
}

plot(x=cv_res$n_pca_comp, y=cv_res$mean_squared_error, xlab="P Comps", ylab = "MSE")
```



 # This suggests that 6 principal components would b be best (14 and 15 also share a low mean squared error but in that case there wouldn't be any principal components, and besides we'll test against this (15) – the 6 PCs fits with what I found in last homework based on p-values

 # I'll also create a pca data frame for later use (pca_df) and run a linear regression model on it

```
pcomps = 6
pca_df <- as.data.frame(cbind(pca$x[,1:pcomps], Crime=dat[,16]))
regression <- lm(Crime~., data=pca_df)
summary(regression)
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-377.15 -172.23   25.81  132.10  480.38

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   905.09      35.35  25.604  < 2e-16 ***
PC1            65.22      14.56   4.478 6.14e-05 ***
PC2           -70.08      21.35  -3.283  0.00214 **
PC3            25.19      25.23   0.998  0.32409
PC4            69.45      33.14   2.095  0.04252 *
PC5          -229.04      36.50  -6.275 1.94e-07 ***
PC6           -60.21      48.04  -1.253  0.21734
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 242.3 on 40 degrees of freedom
Multiple R-squared:  0.6586,    Adjusted R-squared:  0.6074
F-statistic: 12.86 on 6 and 40 DF,  p-value: 4.869e-08
```

let's compare the regression model just created with the actual crime numbers

```
actual <- dat$Crime
predicted <- regression$fitted.values
rss <- sum((predicted - actual)^2)
tss <- sum((actual-mean(actual))^2)
rsq <- 1 - rss/tss
rsq
        [1] 0.6586023
```

a new r-squared value of .65 from our principal component regression model to the actual crime; this is a a lower R-squared than my adjusted R-squared from last homework, so PCA did not appear to help reduce this measure

next step is unscaling the coefficients back to original

```
coeff_convert <- (pca$rotation[,1:4] %*% regression$coefficients[2:5])/pca$scale
```

using the converted coefficients back to determine intercept

```
intercept2 <- regression$coefficients[1]-sum(coeff_convert*pca$center)
intercept2

        1666.485
```

creating new_state for estimate on new crime (given in 8.2 homework)

```
new_state <- data.frame(M=14.0,So=0, Ed=10.0, Po1 = 12.0, Po2 = 15.5, LF=0.640,
M.F=94.0,Pop=150,NW=1.1,U1=0.120,U2=3.6,Wealth=3200, Ineq=20.1, Prob=0.04,Time=39.0)
```

estimate in 8.2 was new crime for the “state” would be 1304, now looking for what this model will give me; had to look at equations on placing in new data; ended up doing manually

```
new_crime <- sum(
 coeff_convert[1,1] %*% new_state$M,
 coeff_convert[2,1] %*% new_state$So,
 ……….coeff_convert[15,1] %*% new_state$Time,
 intercept2
)
new_crime
        [1] 1112.678
```

a near estimate but still off from last estimate by 15%; with a lower R-squared, I would trust the answer given in Homework 8.2 over this PCA method; but if we were given years and years of crime data (and perhaps other variables as well) being able to identify and deploy Principle Components would be very effective; this dataset simply isn’t that large