

HW6

Amin Yakubu

5/4/2019

```
library(ISLR)
library(factoextra)
library(gridExtra)
library(corrplot)
library(RColorBrewer)
library(gplots)
library(igraph)
library(jpeg)
library(imager)
```

Data preprocessing

```
data("USArrests")
seed = 1
usarrests = scale(USArrests)
```

Cluster Analysis

Question a – Without Scaling

```
set.seed(seed)
hc_noscale = hclust(dist(USArrests, method = 'euclidean'), method = 'complete')
```

Question b

```
hc_3_noscale = cutree(hc_noscale, 3)
```

States in the first Cluster

```
USArrests[hc_3_noscale == 1,]

##           Murder Assault UrbanPop Rape
## Alabama      13.2     236      58 21.2
## Alaska       10.0     263      48 44.5
## Arizona       8.1     294      80 31.0
## California    9.0     276      91 40.6
## Delaware      5.9     238      72 15.8
## Florida       15.4    335      80 31.9
## Illinois      10.4    249      83 24.0
```

```

## Louisiana      15.4    249    66 22.2
## Maryland     11.3    300    67 27.8
## Michigan     12.1    255    74 35.1
## Mississippi   16.1    259    44 17.1
## Nevada       12.2    252    81 46.0
## New Mexico    11.4    285    70 32.1
## New York      11.1    254    86 26.1
## North Carolina 13.0    337    45 16.1
## South Carolina 14.4    279    48 22.5

```

States in the 2nd Cluster

```
USArrests[hc_3_noscale == 2,]
```

```

##               Murder Assault UrbanPop Rape
## Arkansas      8.8    190    50 19.5
## Colorado      7.9    204    78 38.7
## Georgia      17.4    211    60 25.8
## Massachusetts 4.4    149    85 16.3
## Missouri      9.0    178    70 28.2
## New Jersey    7.4    159    89 18.8
## Oklahoma      6.6    151    68 20.0
## Oregon        4.9    159    67 29.3
## Rhode Island   3.4    174    87  8.3
## Tennessee     13.2    188    59 26.9
## Texas         12.7    201    80 25.5
## Virginia       8.5    156    63 20.7
## Washington     4.0    145    73 26.2
## Wyoming       6.8    161    60 15.6

```

States in the 3rd Cluster

```
USArrests[hc_3_noscale == 3,]
```

```

##               Murder Assault UrbanPop Rape
## Connecticut   3.3    110    77 11.1
## Hawaii        5.3     46    83 20.2
## Idaho          2.6    120    54 14.2
## Indiana        7.2    113    65 21.0
## Iowa           2.2     56    57 11.3
## Kansas          6.0    115    66 18.0
## Kentucky        9.7    109    52 16.3
## Maine           2.1     83    51  7.8
## Minnesota      2.7     72    66 14.9
## Montana         6.0    109    53 16.4
## Nebraska        4.3    102    62 16.5
## New Hampshire   2.1     57    56  9.5
## North Dakota    0.8     45    44  7.3
## Ohio            7.3    120    75 21.4
## Pennsylvania     6.3    106    72 14.9
## South Dakota    3.8     86    45 12.8
## Utah            3.2    120    80 22.9

```

```

## Vermont      2.2      48      32 11.2
## West Virginia 5.7      81      39  9.3
## Wisconsin    2.6      53      66 10.8

```

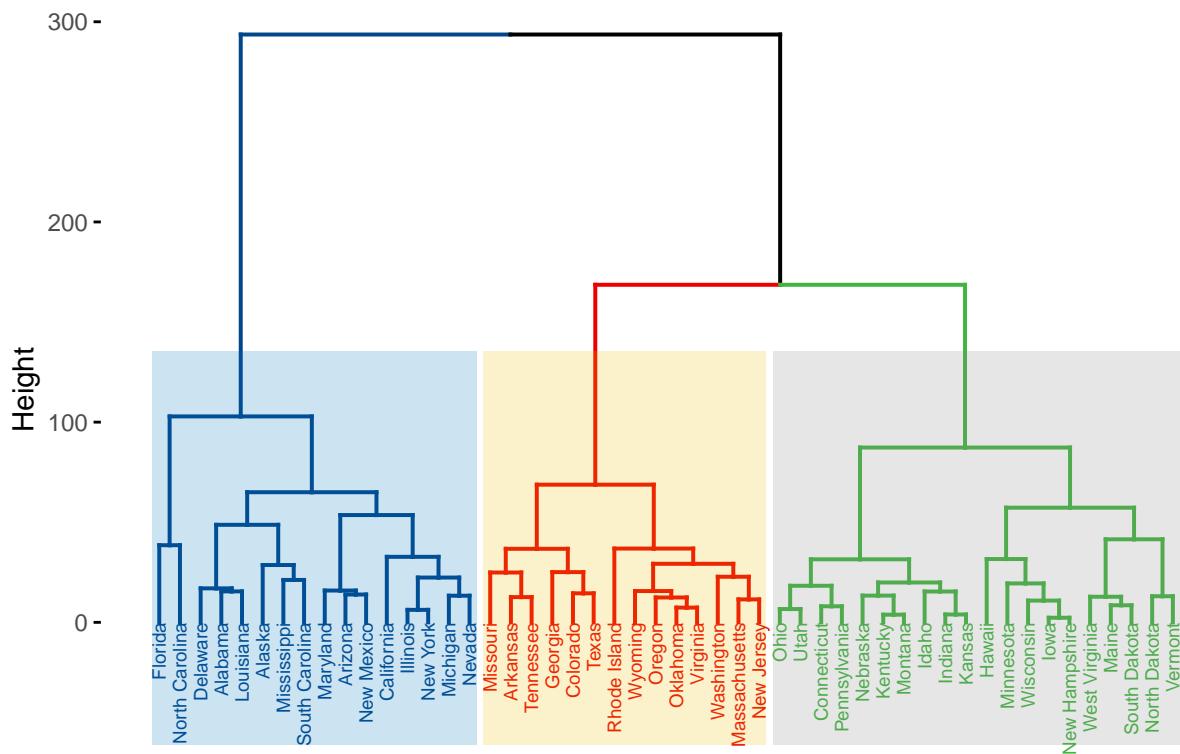
Visualization of the dendrogram and showing graphically which states belong to which clusters

```

fviz_dend(hc_noscale, k = 3,
          cex = 0.5,
          palette = "lancet",
          color_labels_by_k = TRUE,
          rect = TRUE,
          rect_fill = TRUE,
          rect_border = "jco",
          type = 'rectangle',
          labels_track_height = 60.5
)

```

Cluster Dendrogram



Question c

```

set.seed(seed)
hc_scaled = hclust(dist(usrarrests, method = 'euclidean'), method = 'complete')

fviz_dend(hc_scaled, k = 3,
          cex = 0.6,
          palette = "lancet",
          color_labels_by_k = TRUE,
          rect = TRUE,
          rect_fill = TRUE,

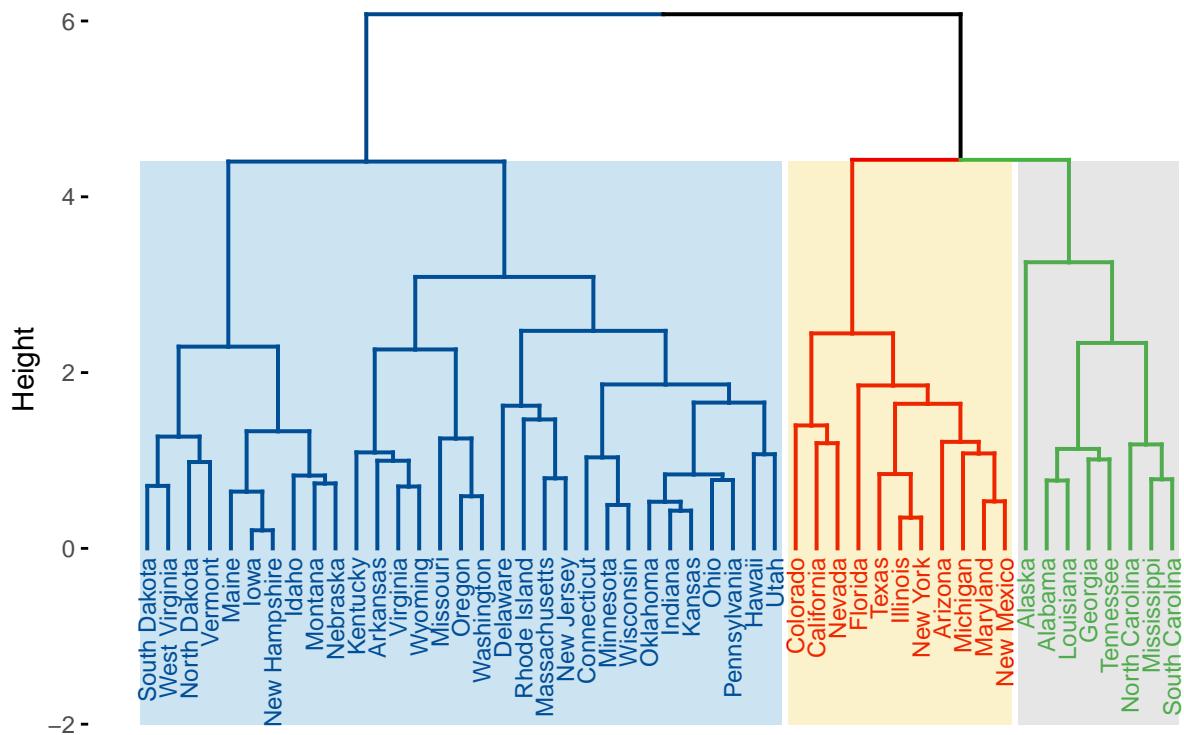
```

```

    rect_border = "jco",
    type = 'rectangle',
    labels_track_height = 1.5
)

```

Cluster Dendrogram



Question d

```

# number of states in each cluster when scaling
table(cutree(hc_scaled, 3))

##
##   1   2   3
##   8  11  31

# number of states in each cluster when not scaling
table(cutree(hc_noscale, 3))

##
##   1   2   3
##  16  14  20

```

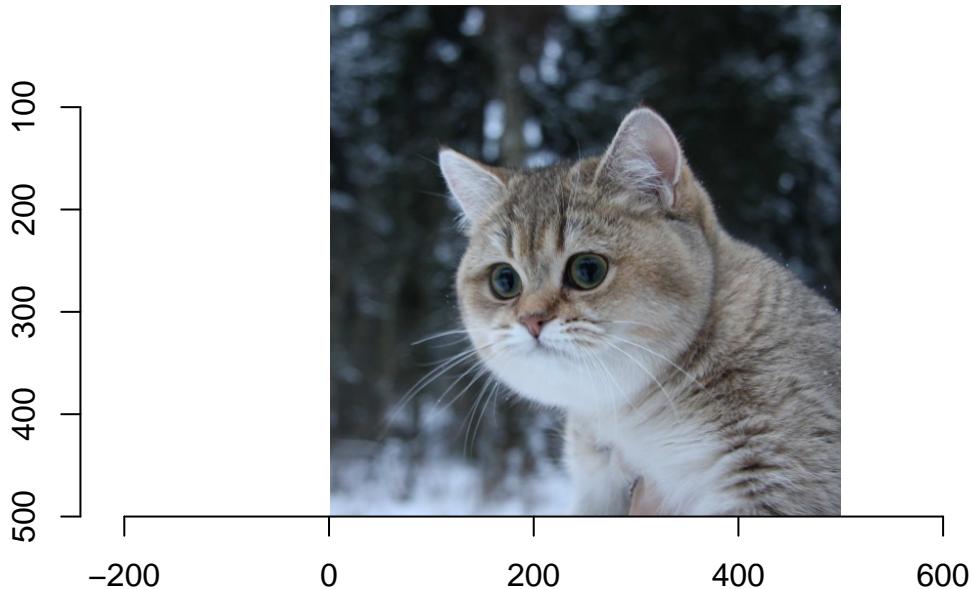
Scaling the variables impacts the clusters that are obtained, the branch lengths, and the height of the tree. So, scaling or not scaling affects the clusters obtained from cutting the dendrogram into 3 clusters as shown in the `table` function above. It is obvious that the states and the number of states in the clusters are different. For example, without scaling, Michigan clusters with Nevada while with scaling Michigan clusters nearby Arizona. The height of the unscaled tree is 300 while the height of the scaled tree is 6. Without scaling, we cut the tree at a height of about 150 whereas we cut the scaled tree at a height of about 4 to obtain 3 clusters.

In this scenario, scaling is more appropriate because `Murder`, `Assault`, and `Rape` all have units of per 100,000 people while `UrbanPop` is the percentage of the state population that lives in urban areas. Therefore, it is important to scale so that the units of `UrbanPop` has an equal contribution to the hierarchical clustering algorithm as the other variables. We will find more meaningful results when the variables are scaled to proportional units.

PCA

```
img <- readJPEG('cat.jpg')

plot(load.image('cat.jpg'))
```



```
dim(img)

## [1] 500 500 3

r <- img[,1]
g <- img[,2]
b <- img[,3]

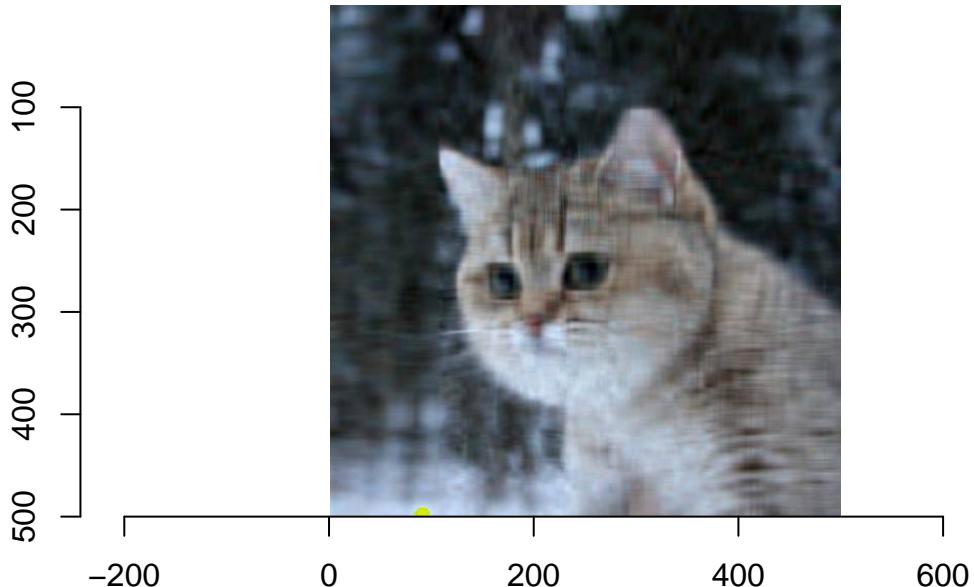
img.r.pca <- prcomp(r, center = FALSE)
img.g.pca <- prcomp(g, center = FALSE)
img.b.pca <- prcomp(b, center = FALSE)

rgb.pca <- list(img.r.pca, img.g.pca, img.b.pca)

# Approximate X with XV_kV_k^T
compress <- function(pr, k)
{
  compressed.img <- pr$x[,1:k] %*% t(pr$rotation[,1:k])
  compressed.img
}

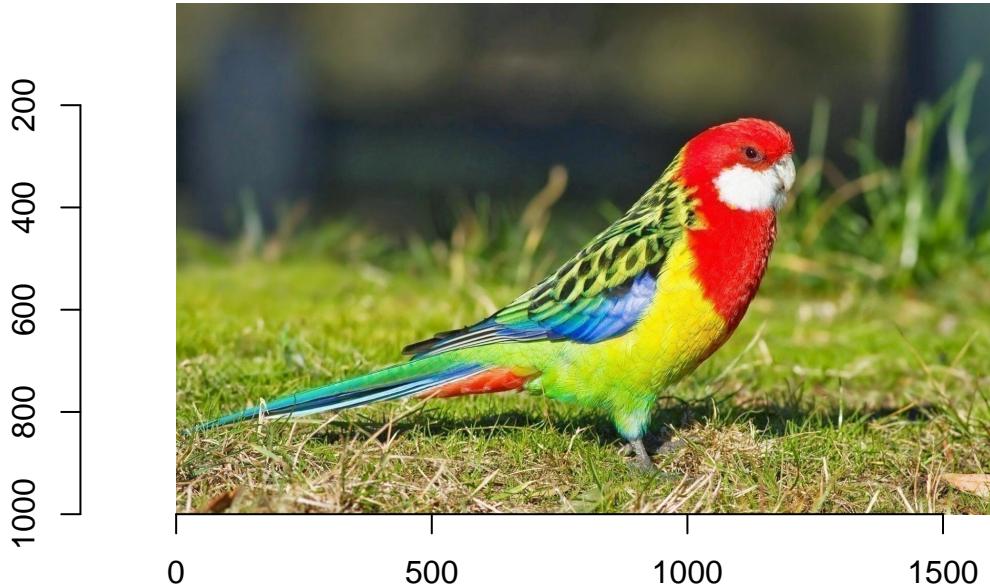
# Using first 20 PCs
```

```
pca20 <- sapply(rgb.pca, compress, k = 20, simplify = "array")  
  
writeJPEG(pca20, "pca20.jpg")  
  
# Try to increase the number of PCs!  
  
plot(load.image('pca20.jpg'))
```



Using Parrot image

```
plot(load.image('parrot.jpg'))
```



```
img <- readJPEG('parrot.jpg')
```

```

dim(img)

## [1] 1000 1600     3

r <- img[, , 1]
g <- img[, , 2]
b <- img[, , 3]

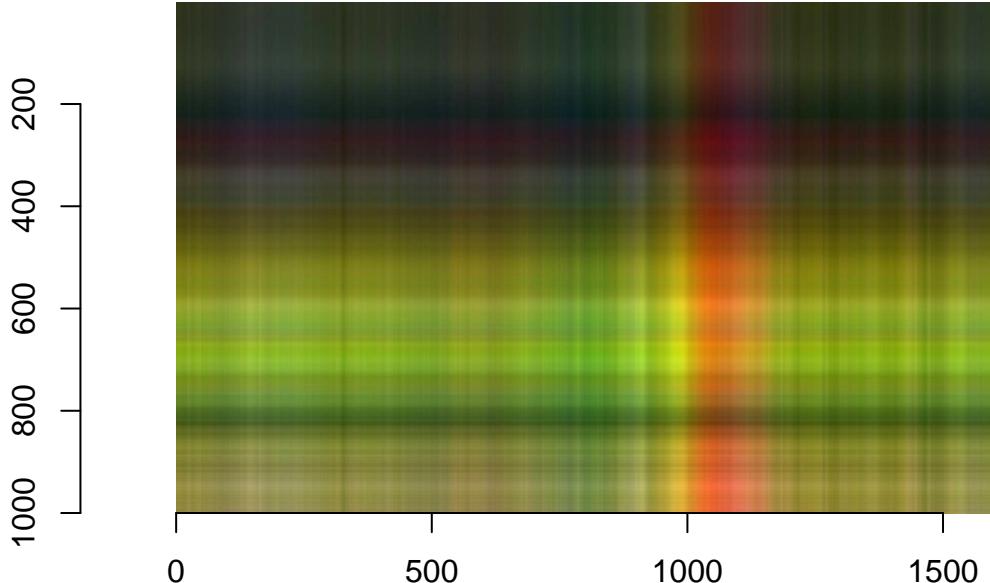
img.r.pca <- prcomp(r, center = FALSE)
img.g.pca <- prcomp(g, center = FALSE)
img.b.pca <- prcomp(b, center = FALSE)

rgb.pca <- list(img.r.pca, img.g.pca, img.b.pca)

# Approximate X with  $XV_kV_k^T$ 
compress <- function(pr, k)
{
  compressed.img <- pr$x[, 1:k] %*% t(pr$rotation[, 1:k])
  compressed.img
}

# Using first PC
p1 <- sapply(rgb.pca, compress, k = 1, simplify = "array")
writeJPEG(p1, "parrot1.jpg")
plot(load.image('parrot1.jpg'))

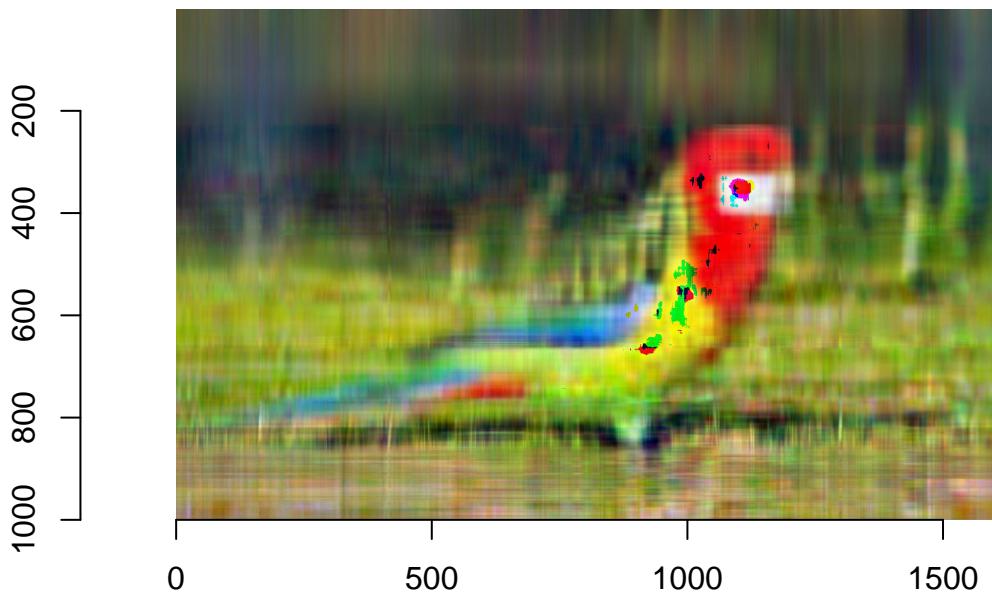
```



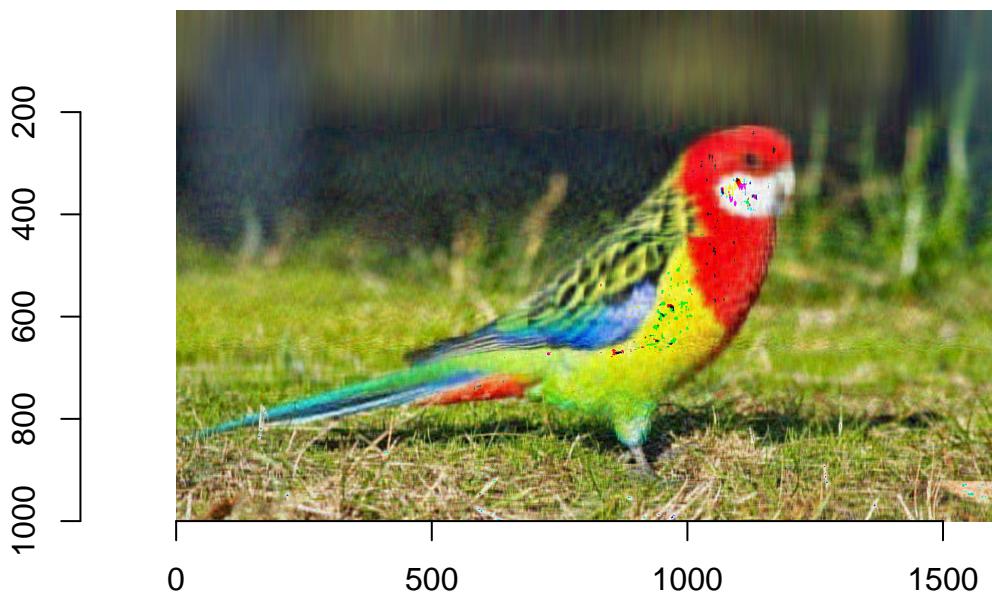
```

# Using first 10 PCs
p10 <- sapply(rgb.pca, compress, k = 10, simplify = "array")
writeJPEG(p10, "parrot10.jpg")
plot(load.image('parrot10.jpg'))

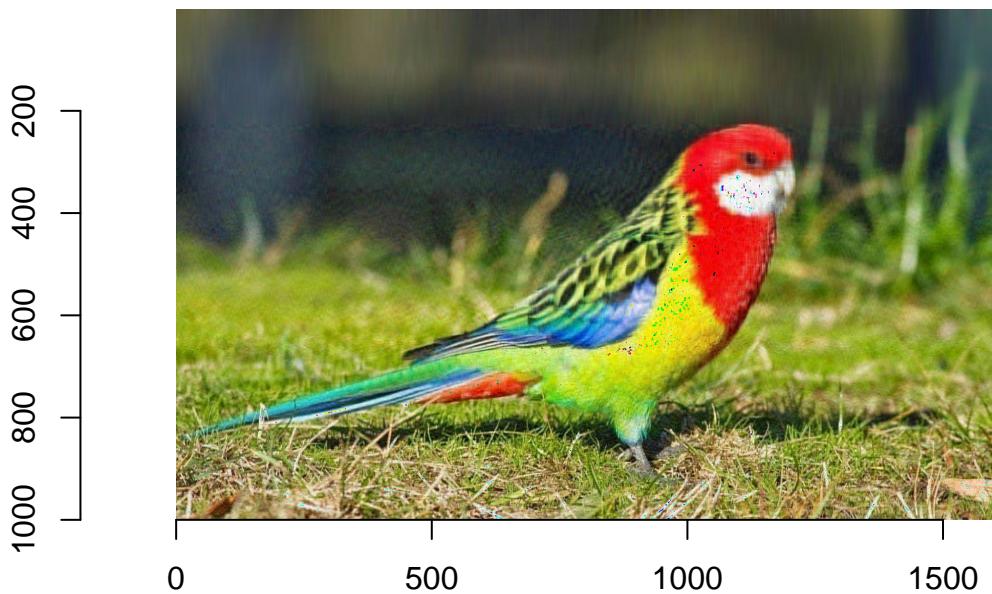
```



```
# Using first 50 PCs
p50 <- sapply(rgb.pca, compress, k = 50, simplify = "array")
writeJPEG(p50, "parrot50.jpg")
plot(load.image('parrot50.jpg'))
```



```
# Using first 100 PCs
p100 <- sapply(rgb.pca, compress, k = 100, simplify = "array")
writeJPEG(p100, "parrot100.jpg")
plot(load.image('parrot100.jpg'))
```



```
# Using first 200 PCs
p200 <- sapply(rgb.pca, compress, k = 200, simplify = "array")
writeJPEG(p200, "parrot200.jpg")
plot(load.image('parrot200.jpg'))
```

