Mini_Project_Class08

Zainab Ashir

```
# Import the data and save it in wisc.df, call our data 'cancer_data'
fna.data <- "https://marcos-diazg.github.io/BIMM143_SP23/class-material/class8/WisconsinCa
wisc.df <- read.csv(fna.data, row.names=1)
cancer_data <- wisc.df[,-1]
```

Q1. How many observations are in this dataset?

```
# Use nrow() function to find the number of rows, which are the observations of our data a
obs<- nrow(cancer_data)
obs
```

```
[1] 569
```

Thus, there is 569 observations in our dataset!!

Q2. How many of the observations have a malignant diagnosis?

```
# Make diagnosis variable that has all the number of observations in diagnosis column, and
diagnosis <- wisc.df$diagnosis
table(diagnosis)
```

```
diagnosis
  B   M
357 212
```

Thus, there appears to be 212 melignant diagnosis

Q3. How many variables/features in the data are suffixed with _mean?

```r
grep("_mean",colnames(cancer_data), ignore.case = FALSE)
```

```
 [1]  1  2  3  4  5  6  7  8  9 10
```

Thus, there is 10 columns with suffix _mean

```r
# Check column means and standard deviations
colMeans(cancer_data)
```

```
          radius_mean          texture_mean        perimeter_mean
         1.412729e+01          1.928965e+01          9.196903e+01
            area_mean       smoothness_mean      compactness_mean
         6.548891e+02          9.636028e-02          1.043410e-01
       concavity_mean    concave.points_mean         symmetry_mean
         8.879932e-02          4.891915e-02          1.811619e-01
fractal_dimension_mean             radius_se            texture_se
         6.279761e-02          4.051721e-01          1.216853e+00
          perimeter_se               area_se         smoothness_se
         2.866059e+00          4.033708e+01          7.040979e-03
       compactness_se         concavity_se    concave.points_se
         2.547814e-02          3.189372e-02          1.179614e-02
          symmetry_se    fractal_dimension_se          radius_worst
         2.054230e-02          3.794904e-03          1.626919e+01
       texture_worst       perimeter_worst            area_worst
         2.567722e+01          1.072612e+02          8.805831e+02
     smoothness_worst     compactness_worst       concavity_worst
         1.323686e-01          2.542650e-01          2.721885e-01
 concave.points_worst        symmetry_worst fractal_dimension_worst
         1.146062e-01          2.900756e-01          8.394582e-02
```

```r
apply(cancer_data,2,sd)
```

```
          radius_mean          texture_mean        perimeter_mean
         3.524049e+00          4.301036e+00          2.429898e+01
            area_mean       smoothness_mean      compactness_mean
         3.519141e+02          1.406413e-02          5.281276e-02
       concavity_mean    concave.points_mean         symmetry_mean
         7.971981e-02          3.880284e-02          2.741428e-02
```

```
 fractal_dimension_mean               radius_se              texture_se
           7.060363e-03            2.773127e-01            5.516484e-01
           perimeter_se                 area_se          smoothness_se
           2.021855e+00            4.549101e+01            3.002518e-03
        compactness_se            concavity_se       concave.points_se
           1.790818e-02            3.018606e-02            6.170285e-03
            symmetry_se    fractal_dimension_se            radius_worst
           8.266372e-03            2.646071e-03            4.833242e+00
         texture_worst         perimeter_worst              area_worst
           6.146258e+00            3.360254e+01            5.693570e+02
      smoothness_worst       compactness_worst         concavity_worst
           2.283243e-02            1.573365e-01            2.086243e-01
  concave.points_worst          symmetry_worst fractal_dimension_worst
           6.573234e-02            6.186747e-02            1.806127e-02
```

```r
# Perform PCA on cancer.data by using summary of prcomp()
cancer.pr <- prcomp(cancer_data, scale. = TRUE)
summary(cancer.pr)
```

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

Q4. From your results, what proportion of the original variance is captured by the first principal components (PC1)?

It is 0.4427

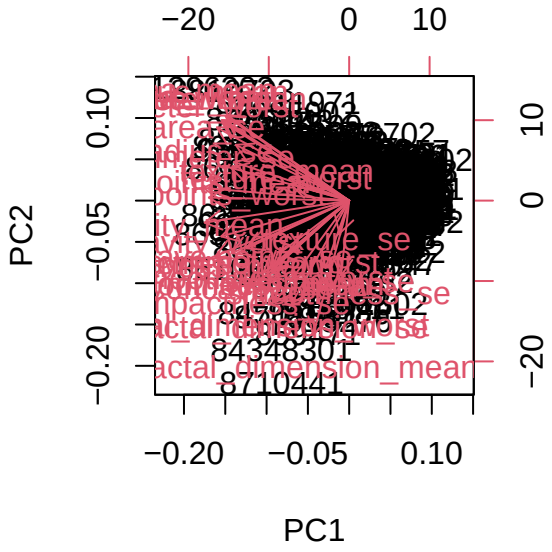
Q5. How many principal components (PCs) are required to describe at least 70% of the original variance in the data?

It would be 3 components since tye cumulative under PC3 is 0.72636, which is the first one that hit at least 70%

Q6. How many principal components (PCs) are required to describe at least 90% of the original variance in the data?

It would be 7 components since the cumulative under PC3 is 0.91010, which is the first one that hit at least 90%

```
#Creating biplot for our results
biplot(cancer.pr)
```

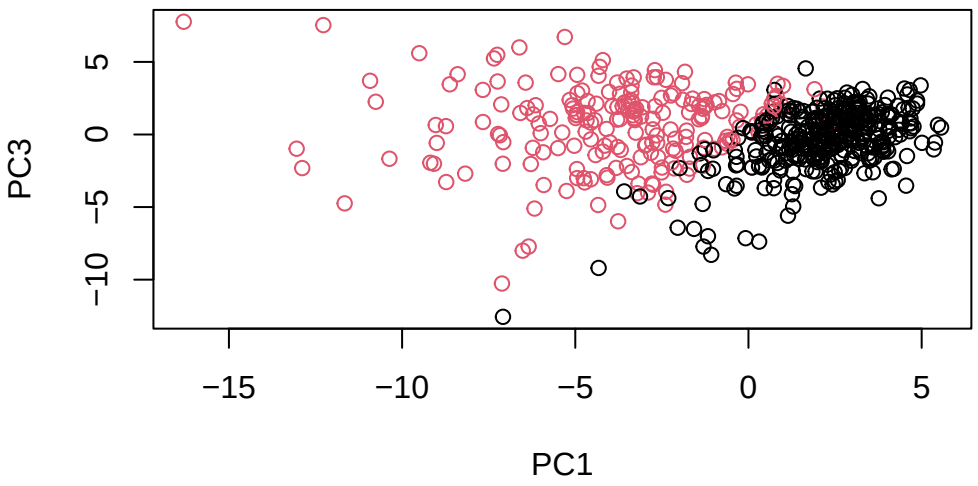


Q7. What stands out to you about this plot? Is it easy or difficult to understand? Why?

Nothing stands out in this plot, it is a total mess. It's so diffcult to understand since there seems to be so many observations to fit into this plot with no meaning or clear visuale.

Q8. Generate a similar plot for principal components 1 and 3. What do you notice about these plots?

```
# Make scatter plot for components 1 and 3
diagnosis_vector <- as.numeric(diagnosis=="M")
plot(cancer.pr$x,col=(diagnosis_vector+1), xlab = "PC1", ylab = "PC3")
```



This plot seems to have a better representation of Melignant and benign observations in the PC3 vs PC! axies

Q9. For the first principal component, what is the component of the loading vector (i.e. wisc.pr$rotation[,1]) for the feature concave.points_mean? This tells us how much this original feature contributes to the first PC.

```
#Use the function
cancer.pr$rotation[,1]
```

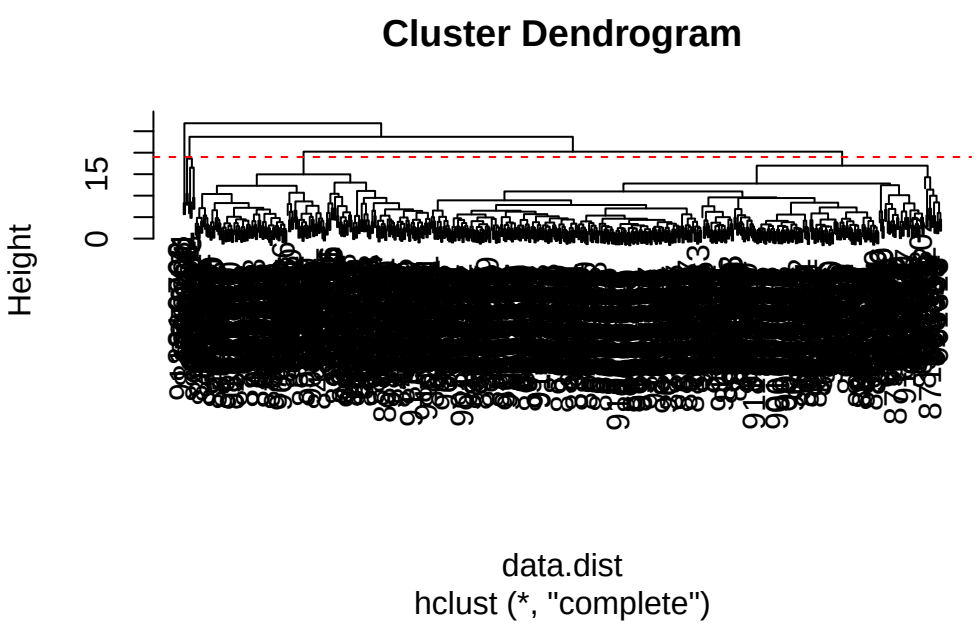
```
            radius_mean            texture_mean          perimeter_mean
            -0.21890244             -0.10372458             -0.22753729
              area_mean         smoothness_mean        compactness_mean
            -0.22099499             -0.14258969             -0.23928535
         concavity_mean     concave.points_mean           symmetry_mean
            -0.25840048             -0.26085376             -0.13816696
 fractal_dimension_mean               radius_se              texture_se
            -0.06436335             -0.20597878             -0.01742803
           perimeter_se                 area_se           smoothness_se
            -0.21132592             -0.20286964             -0.01453145
         compactness_se            concavity_se       concave.points_se
            -0.17039345             -0.15358979             -0.18341740
           symmetry_se    fractal_dimension_se             radius_worst
            -0.04249842             -0.10256832             -0.22799663
         texture_worst         perimeter_worst              area_worst
            -0.10446933             -0.23663968             -0.22487053
       smoothness_worst       compactness_worst         concavity_worst
            -0.12795256             -0.21009588             -0.22876753
  concave.points_worst          symmetry_worst fractal_dimension_worst
            -0.25088597             -0.12290456             -0.13178394
```

It looks like its -0.26085376

Q10. Using the plot() and abline() functions, what is the height at which the clustering model has 4 clusters?

```
#do the heierchy
data.scaled <- scale(cancer_data)
data.dist <- dist(data.scaled)
cancer.hclust <- hclust(data.dist, method="complete")
```

```
#plot the hclust function of our data
plot(cancer.hclust)
abline(h=19, col="red", lty=2)
```



The height is 18

Q12. Which method gives your favorite results for the same data.dist dataset? Explain your reasoning.

My favourite method is "complete" since the heirchy tree it gives is very clear and easier for me to trace the linkage.

Q13. How well does the newly created model with four clusters separate out the two diagnoses?

```
# Using Kmeans
cancer.km <- kmeans(scale(cancer_data), centers=2, nstart=20)
table(cancer.km$cluster, diagnosis)
```

```
  diagnosis
     B   M
1   14 175
2 343  37
```

It does seem like it seperates them well enough.

Q14. How well do the hierarchical clustering models you created in previous sections (i.e. before PCA) do in terms of separating the diagnoses? Again, use the table() function to compare the output of each model (wisc.km$cluster and wisc.hclust.clusters) with the vector containing the actual diagnoses.

```
# Create new model for 4 clusters and look at its results
cancer.hclust.clusters <- cutree(cancer.hclust, k=4)
table(cancer.hclust.clusters, diagnosis_vector)
```

```
                     diagnosis_vector
cancer.hclust.clusters   0   1
                     1  12 165
                     2   2   5
                     3 343  40
                     4   0   2
```

It does seem like it seperates, but not that clear

Optional

```
cancer.pr.clust <- hclust(data.dist, method="ward.D2")
g <- cutree(cancer.pr.clust, k=2)
```

Prediction

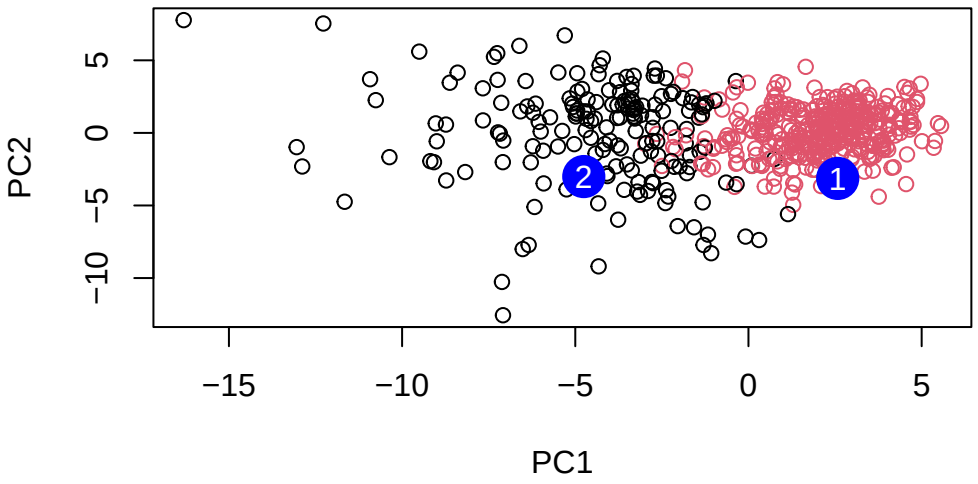
```
#prepare the data
url <- "https://tinyurl.com/new-samples-CSV"
new <- read.csv(url)
npc <- predict(cancer.pr, newdata=new)
npc
```

```
          PC1       PC2        PC3        PC4       PC5        PC6        PC7
[1,]  2.576616 -3.135913  1.3990492 -0.7631950  2.781648 -0.8150185 -0.3959098
[2,] -4.754928 -3.009033 -0.1660946 -0.6052952 -1.140698 -1.2189945  0.8193031
           PC8       PC9       PC10      PC11      PC12      PC13     PC14
[1,] -0.2307350 0.1029569 -0.9272861 0.3411457  0.375921 0.1610764 1.187882
[2,] -0.3307423 0.5281896 -0.4855301 0.7173233 -1.185917 0.5893856 0.303029
         PC15       PC16        PC17        PC18        PC19       PC20
```

```
[1,] 0.3216974 -0.1743616 -0.07875393 -0.11207028 -0.08802955 -0.2495216
[2,] 0.1299153  0.1448061 -0.40509706  0.06565549  0.25591230 -0.4289500
           PC21       PC22       PC23       PC24        PC25          PC26
[1,]  0.1228233 0.09358453 0.08347651  0.1223396  0.02124121  0.078884581
[2,] -0.1224776 0.01732146 0.06316631 -0.2338618 -0.20755948 -0.009833238
             PC27        PC28         PC29         PC30
[1,]  0.220199544 -0.02946023 -0.015620933  0.005269029
[2,] -0.001134152  0.09638361  0.002795349 -0.019015820
```

Q16. Which of these new patients should we prioritize for follow up based on your results?

```
#first need to plot the data
plot(cancer.pr$x[,1:2],col=g)
points(npc[,1], npc[,2], col="blue", pch=16, cex=3)
text(npc[,1], npc[,2], c(1,2), col="white")
```



It looks like patient #1 should be prioritized