08 HW-Lab Class08(ML Mini Project)

May 4, 2024

NAME: Edwin Ruiz

PID: A17136339

1 1. Exploratory data analysis

[2]:
```
# Save your input data file into your Project directory
fna.data <- "WisconsinCancer.csv"

# Complete the following code to input the data and store as wisc.df
wisc.df <- read.csv(fna.data, row.names=1)

wisc.df
```

	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
842302	M	17.990	10.38	122.80	1001.0
842517	M	20.570	17.77	132.90	1326.0
84300903	M	19.690	21.25	130.00	1203.0
84348301	M	11.420	20.38	77.58	386.1
84358402	M	20.290	14.34	135.10	1297.0
843786	M	12.450	15.70	82.57	477.1
844359	M	18.250	19.98	119.60	1040.0
84458202	M	13.710	20.83	90.20	577.9
844981	M	13.000	21.82	87.50	519.8
84501001	M	12.460	24.04	83.97	475.9
845636	M	16.020	23.24	102.70	797.8
84610002	M	15.780	17.89	103.60	781.0
846226	M	19.170	24.80	132.40	1123.0
846381	M	15.850	23.95	103.70	782.7
84667401	M	13.730	22.61	93.60	578.3
84799002	M	14.540	27.54	96.73	658.8
848406	M	14.680	20.13	94.74	684.5
84862001	M	16.130	20.68	108.10	798.8
849014	M	19.810	22.15	130.00	1260.0
8510426	B	13.540	14.36	87.46	566.3
8510653	B	13.080	15.71	85.63	520.0
8510824	B	9.504	12.44	60.34	273.9
8511133	M	15.340	14.26	102.50	704.4
851509	M	21.160	23.04	137.20	1404.0
852552	M	16.650	21.38	110.00	904.6
852631	M	17.140	16.40	116.00	912.7
852763	M	14.580	21.53	97.41	644.8
852781	M	18.610	20.25	122.10	1094.0
852973	M	15.300	25.27	102.40	732.4
A data.frame: 569 × 31 853201	M	17.570	15.05	115.00	955.1
921362	B	7.691	25.44	48.34	170.4
921385	B	11.540	14.44	74.65	402.9
921386	B	14.470	24.99	95.81	656.4
921644	B	14.740	25.42	94.70	668.6
922296	B	13.210	28.06	84.88	538.4
922297	B	13.870	20.70	89.77	584.8
922576	B	13.620	23.23	87.19	573.2
922577	B	10.320	16.35	65.31	324.9
922840	B	10.260	16.58	65.85	320.8
923169	B	9.683	19.34	61.05	285.7
923465	B	10.820	24.21	68.89	361.6
923748	B	10.860	21.48	68.51	360.5
923780	B	11.130	22.44	71.49	378.4
924084	B	12.770	29.43	81.35	507.9
924342	B	9.333	21.94	59.01	264.0
924632	B	12.880	28.92	82.50	514.3
924934	B	10.290	27.61	65.67	321.4
924964	B	10.160	19.59	64.73	311.7
925236	B	9.423	27.88	59.26	271.3
925277	B	14.590	22.68	96.39	657.1

[3]:
```
# We can use -1 here to remove the first column
wisc.data <- wisc.df[,-1]

# Create diagnosis vector for later
diagnosis <- factor(wisc.df$diagnosis)
```

1.0.1 Q1. How many observations are in this dataset?

[4]:
```
nrow(wisc.data)
```

569

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

[5]:
```
sum(diagnosis == "M")
```

212

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

[6]:
```
length(grep("_mean", names(wisc.data)))
```

10

2 2. Principal Component Analysis

2.1 Performing PCA

[7]:
```
# Check column means and standard deviations
colMeans(wisc.data)

apply(wisc.data,2,sd)
```

radius_mean 14.1272917398946 **texture_mean** 19.2896485061512 **perimeter_mean** 91.9690333919157 **area_mean** 654.889103690686 **smoothness_mean** 0.096360281195079 **compactness_mean** 0.104340984182777 **concavity_mean** 0.0887993158172232 **concave.points_mean** 0.0489191458699472 **symmetry_mean** 0.181161862917399 **fractal_dimension_mean** 0.0627976098418278 **radius_se** 0.405172056239016 **texture_se** 1.21685342706503 **perimeter_se** 2.86605922671353 **area_se** 40.337079086116 **smoothness_se** 0.00704097891036907 **compactness_se** 0.0254781388400703 **concavity_se** 0.0318937163444639 **concave.points_se** 0.0117961370826011 **symmetry_se** 0.0205422987697715 **fractal_dimension_se** 0.00379490386643234 **radius_worst** 16.2691898066784 **texture_worst** 25.677223198594 **perimeter_worst** 107.261212653779 **area_worst** 880.583128295254 **smoothness_worst** 0.132368594024605 **compactness_worst** 0.254265043936731 **concavity_worst** 0.272188483304042 **concave.points_worst** 0.114606223198594 **symmetry_worst** 0.290075571177505 **fractal_dimension_worst** 0.0839458172231985

radius_mean 3.52404882621208 **texture_mean** 4.30103576816695 **perimeter_mean** 24.2989810387549 **area_mean** 351.914129181653 **smoothness_mean** 0.0140641281376736 **compactness_mean** 0.0528127579325122 **concavity_mean** 0.0797198087078935 **concave.points_mean** 0.0388028448591536 **symmetry_mean** 0.0274142813360357 **fractal_dimension_mean** 0.00706036279508446 **radius_se** 0.277312732986104 **texture_se** 0.551648392617202 **perimeter_se** 2.02185455404211 **area_se** 45.4910055161318 **smoothness_se** 0.00300251794383907 **compactness_se** 0.0179081793256774 **concavity_se** 0.0301860603229884 **concave.points_se** 0.00617028517404687 **symmetry_se** 0.0082663715287984 **fractal_dimension_se** 0.0026460709670891 9 **radius_worst** 4.83324158046932 **texture_worst** 6.14625762303832 **perimeter_worst** 33.6025422690364 **area_worst** 569.356992669949 **smoothness_worst** 0.0228324294048355 **compactness_worst** 0.157336488913742 **concavity_worst** 0.208624280608132 **concave.points_worst** 0.0657323411959421 **symmetry_worst** 0.0618674675375187 **fractal_dimension_worst** 0.018061267348894

[8]:
```
# Perform PCA on wisc.data by completing the following code
wisc.pr <- prcomp(wisc.data, scale. = TRUE)

# Look at summary of results
summary(wisc.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
```

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

0.4427 or 44.27%

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

[9]:
```
pcs_70 <- min(which(cumsum(wisc.pr$sdev^2 / sum(wisc.pr$sdev^2)) >= 0.7))
pcs_70
```

3

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

[10]:
```
pcs_90 <- min(which(cumsum(wisc.pr$sdev^2 / sum(wisc.pr$sdev^2)) >= 0.9))
pcs_90
```

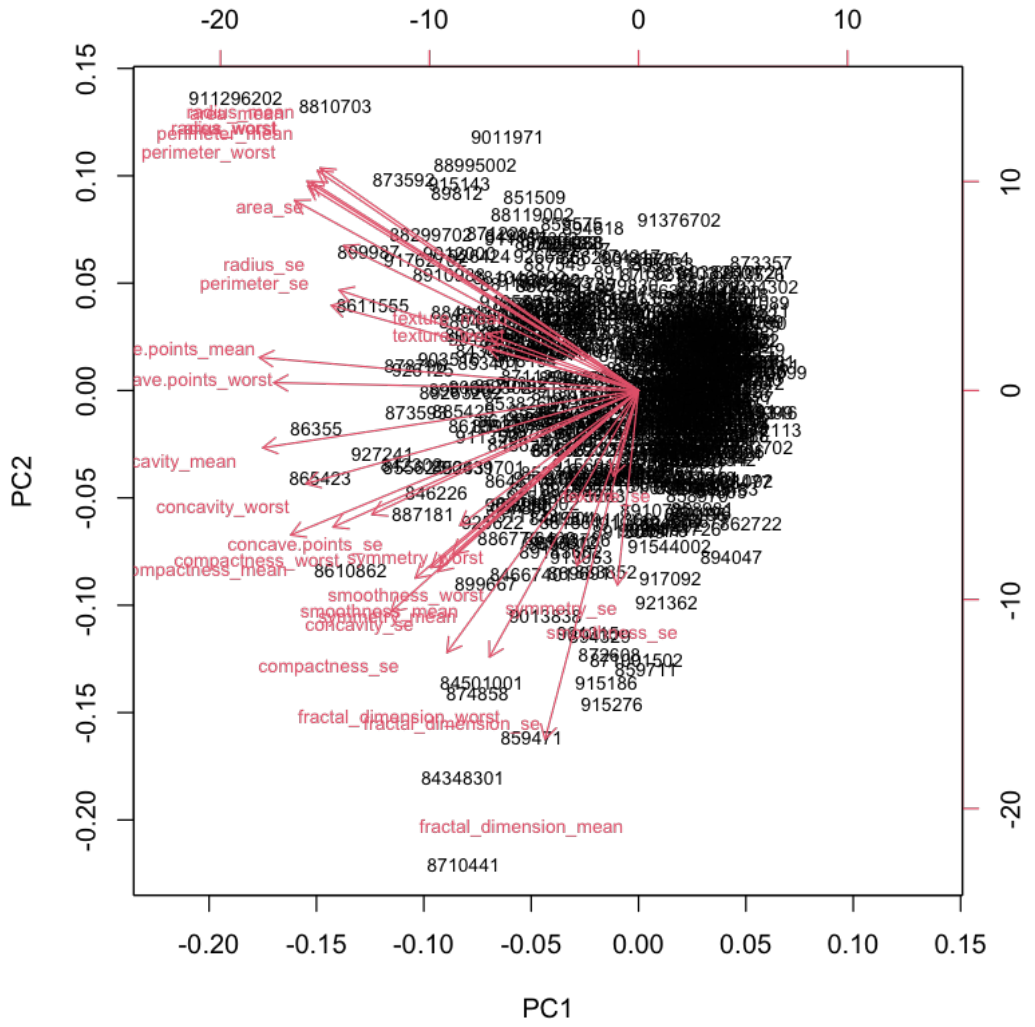
7

2.2 Interpreting PCA results

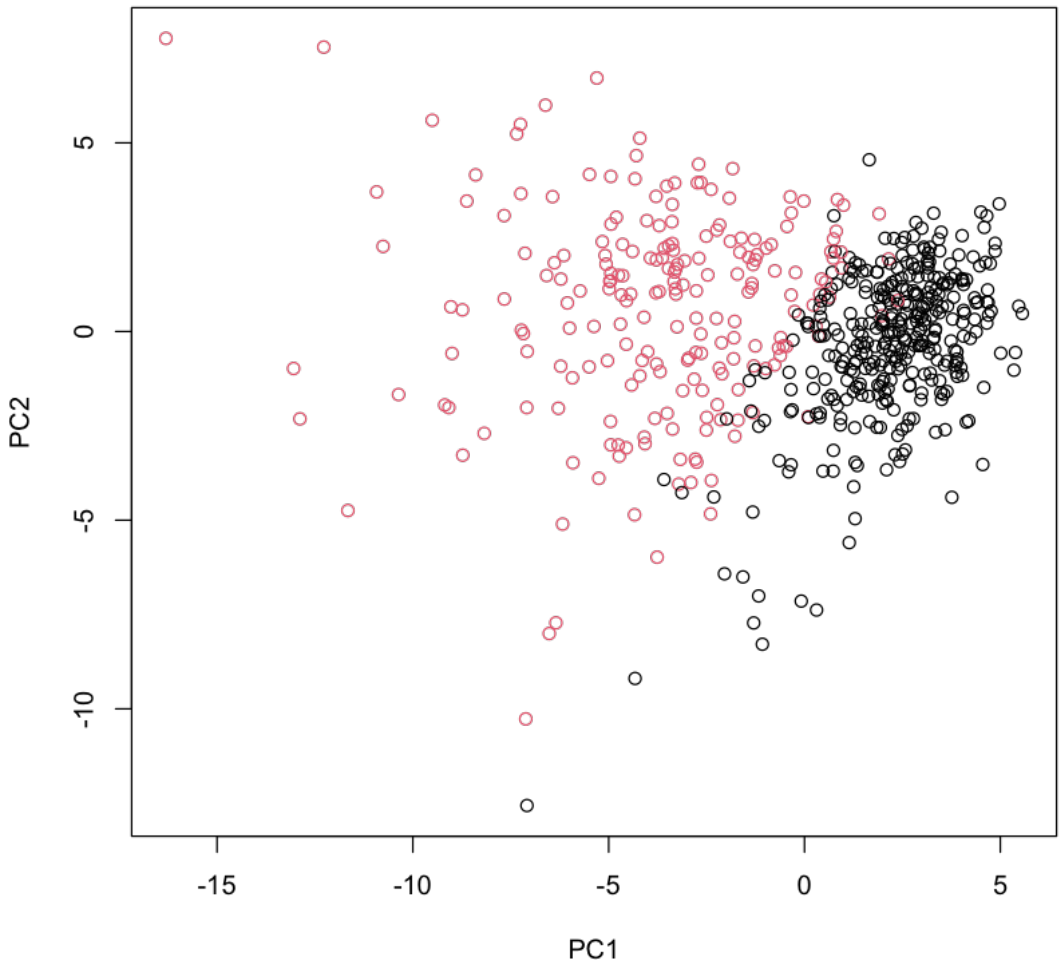
2.2.1 Q7. What stands out to you about this plot? Is it easy or difficult to understand? Why?

Everything seems to be at the center of this black cluster and it is difficult to interpret because it is not easy to determine which data belongs do what cluster in this plot

[11]:
```
biplot(wisc.pr, cex = 0.7)
```

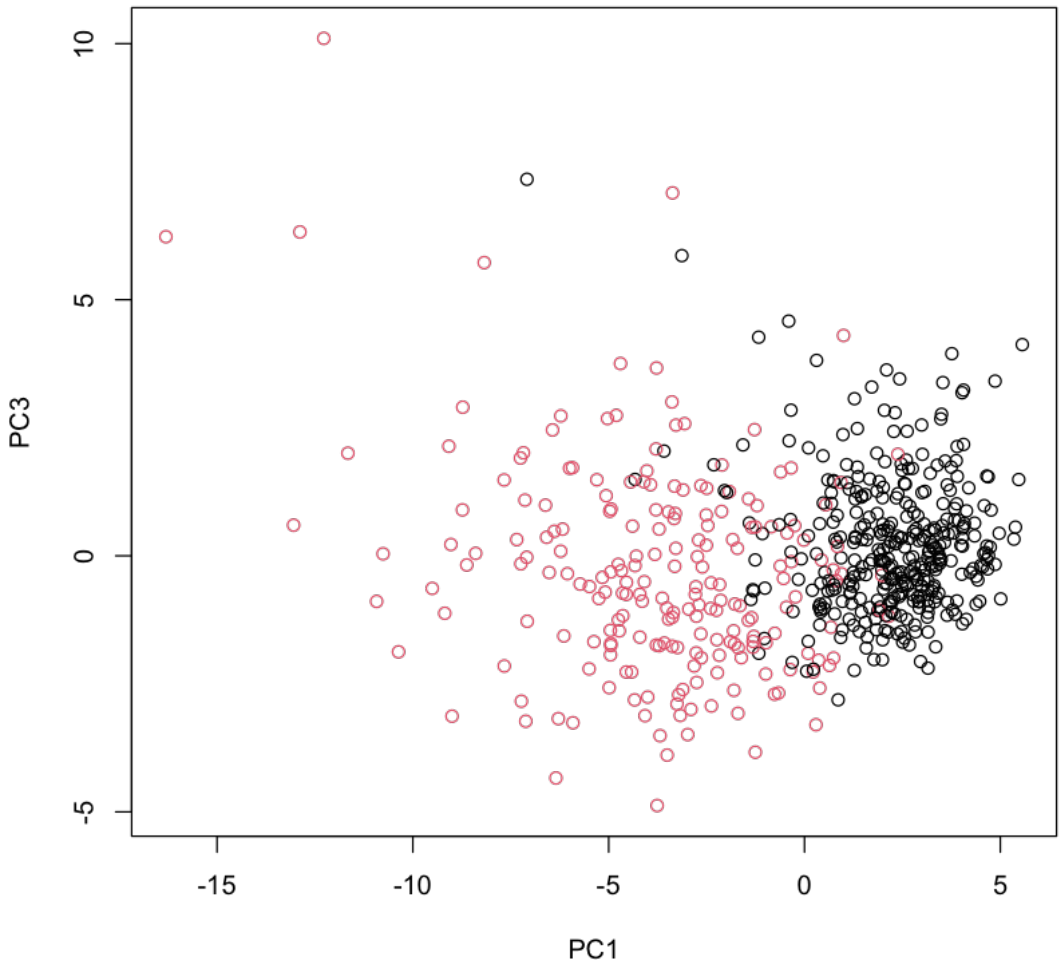


[14]:
```
# Scatter plot observations by components 1 and 2
plot(wisc.pr$x[,c(1,2)], col = diagnosis,
     xlab = "PC1", ylab = "PC2")
```



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

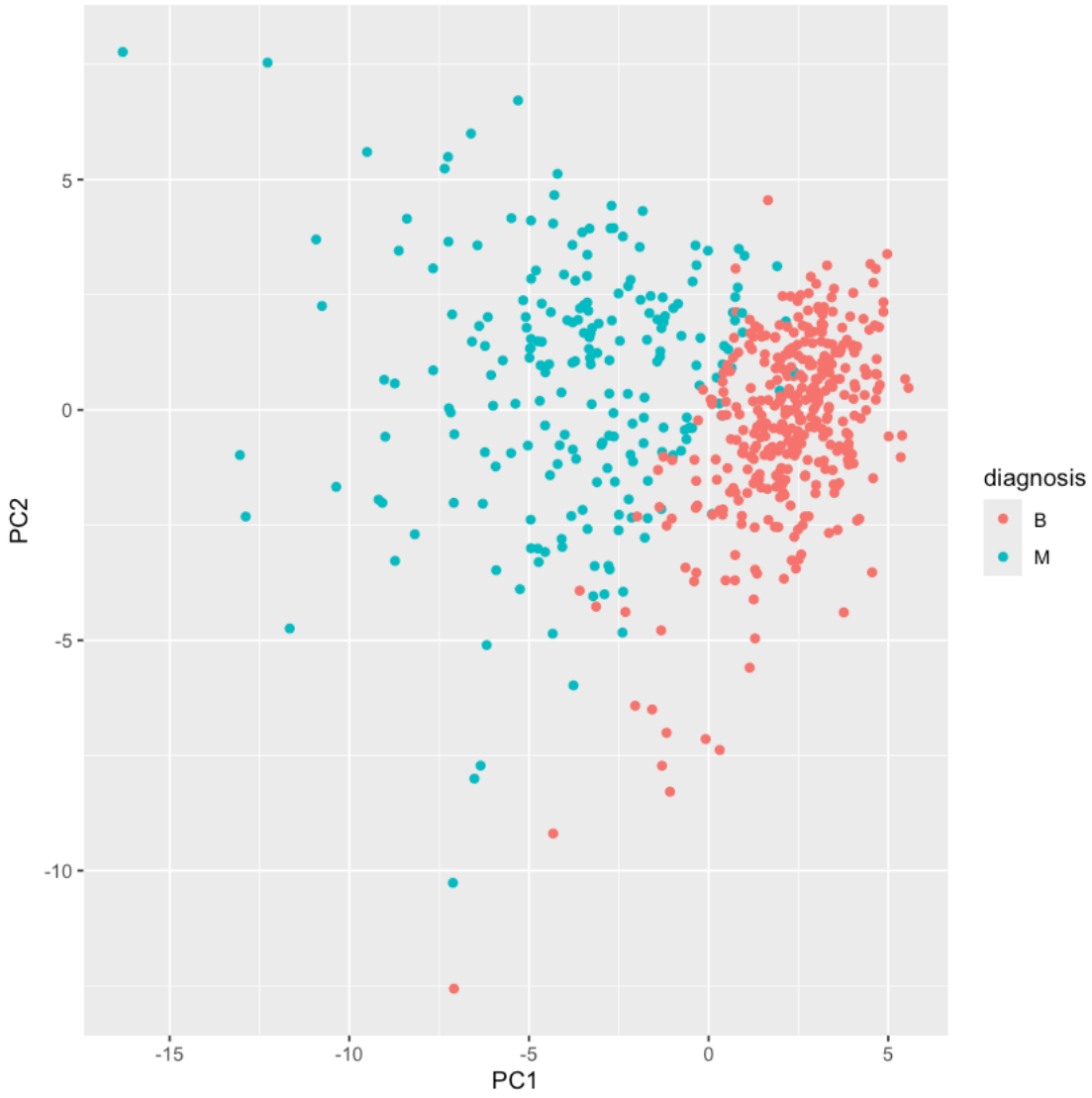
[12]:
```
# Repeat for components 1 and 3
plot(wisc.pr$x[,c(1,3)], col = diagnosis,
     xlab = "PC1", ylab = "PC3")
```



The only difference is in PC3 which shows the variability in the data and PC1 remains consistent overall

[13]:
```
# Create a data frame from PCA results for plotting
df <- as.data.frame(wisc.pr$x)
df$diagnosis <- diagnosis

# Load the ggplot2 package
library(ggplot2)

# Make a scatter plot colored by diagnosis
ggplot(df) +
  aes(PC1, PC2, col=diagnosis) +
```

```
  geom_point()
```

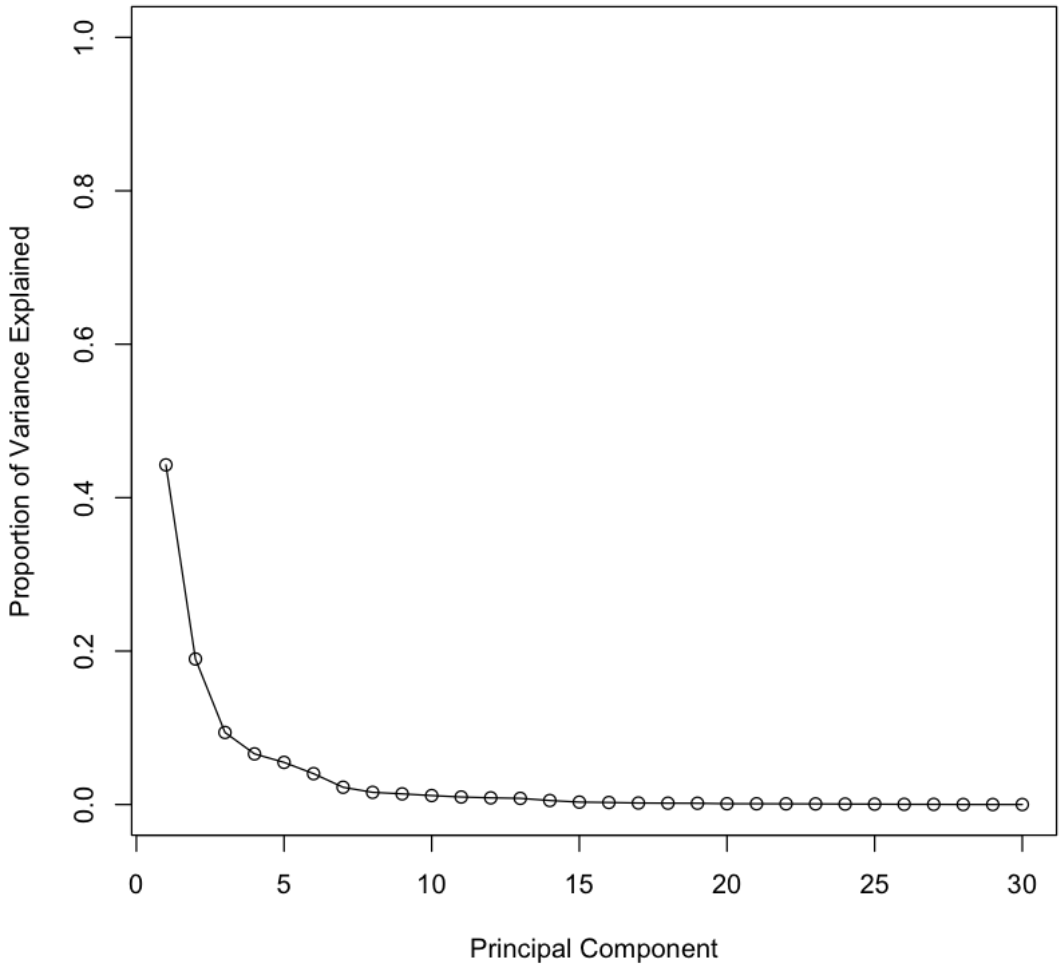


2.3 Variance explained

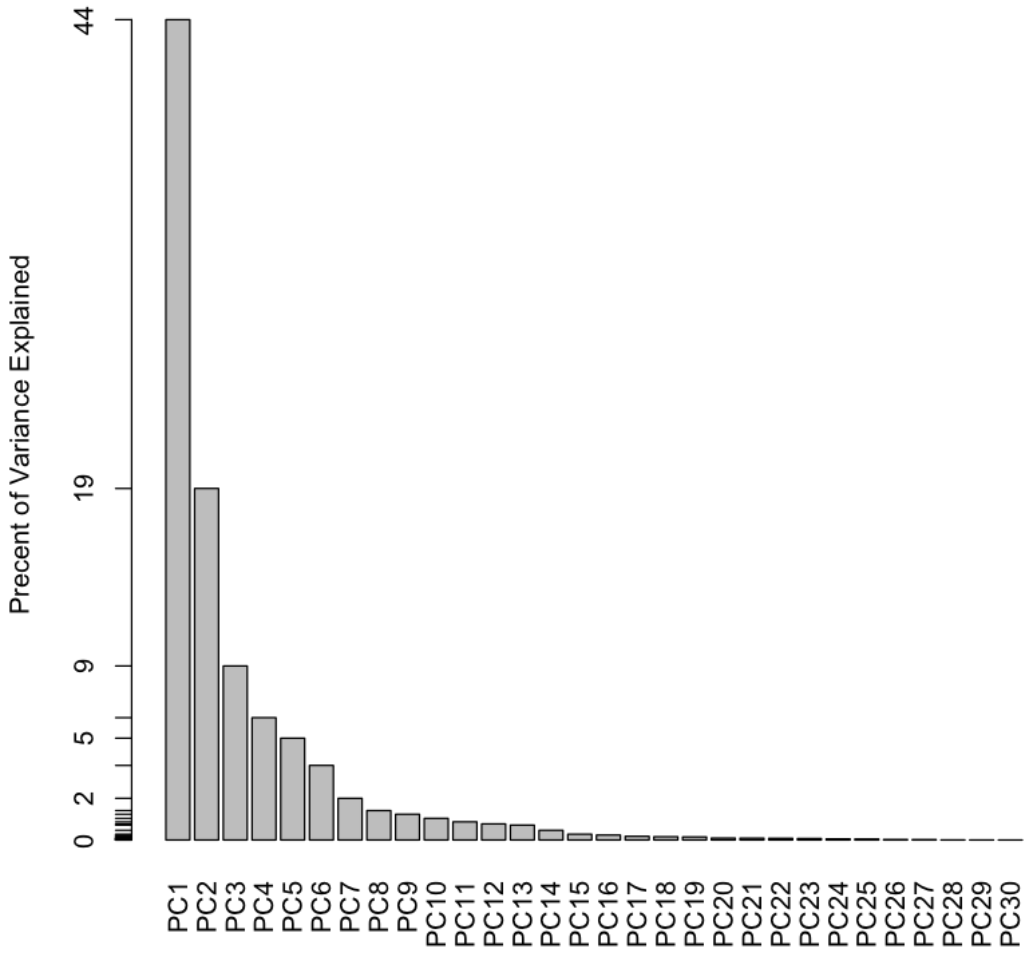
[14]:
```
# Calculate variance of each component
pr.var <- wisc.pr$sdev^2
head(pr.var)

# Variance explained by each principal component: pve
pve <- pr.var / sum(pr.var)

# Plot variance explained for each principal component
```

```
plot(pve, xlab = "Principal Component",
     ylab = "Proportion of Variance Explained",
     ylim = c(0, 1), type = "o")
```

1. 13.2816076822579 2. 5.69135461320994 3. 2.81794897722942 4. 1.98064047464105 5. 1.64873054770388 6. 1.207356611965



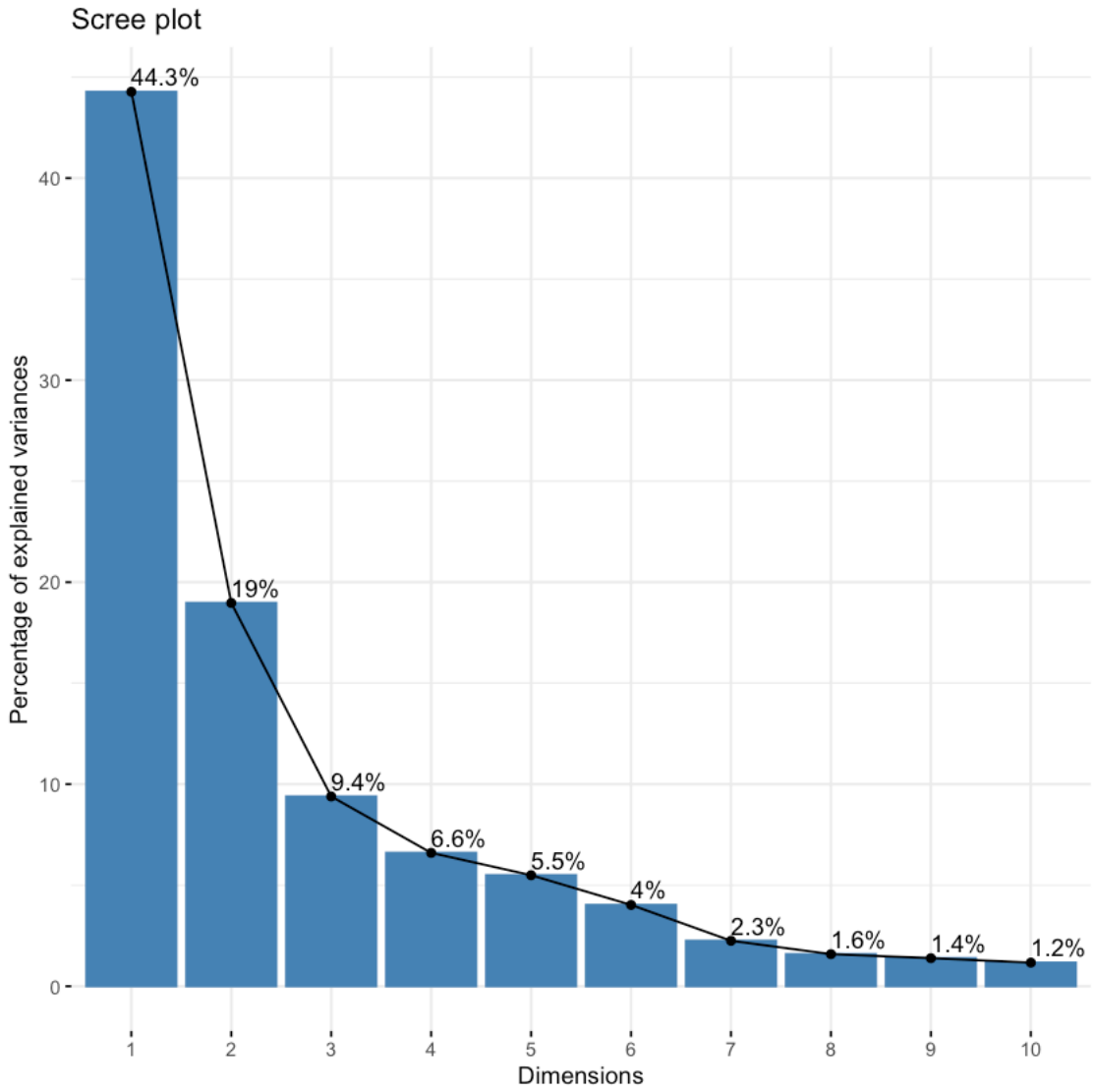
[15]:
```
# Alternative scree plot of the same data, note data driven y-axis
barplot(pve, ylab = "Precent of Variance Explained",
     names.arg=paste0("PC",1:length(pve)), las=2, axes = FALSE)
axis(2, at=pve, labels=round(pve,2)*100 )
```



[16]:
```
## ggplot based graph
install.packages("factoextra")
library(factoextra)
fviz_eig(wisc.pr, addlabels = TRUE)
```

```
The downloaded binary packages are in
/var/folders/vw/6c5wjngs433234dthdjypz800000gn/T//RtmpaUnjdT/downloaded_packages
```

```
Welcome! Want to learn more? See two factoextra-related books at
https://goo.gl/ve3WBa
```



2.4 Communicating PCA results

2.4.1 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?

[17]:
```
wisc.pr$rotation["concave.points_mean", 1]
```

-0.26085375838574

2.4.2 Q10. What is the minimum number of principal components required to explain 80% of the variance of the data?

[18]:
```
pcs_80 <- min(which(cumsum(wisc.pr$sdev^2 / sum(wisc.pr$sdev^2)) >= 0.8))
pcs_80
```

5

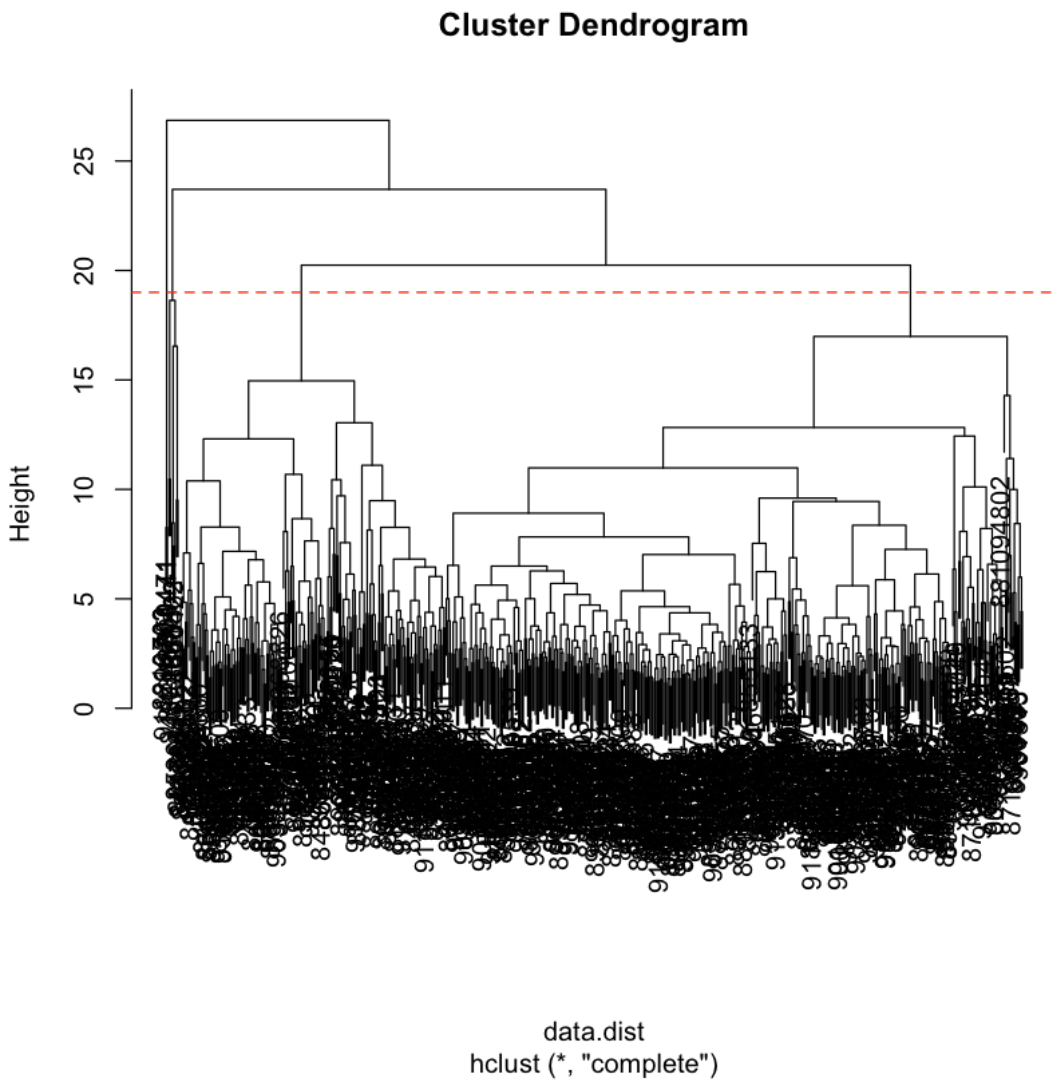
2.5 3. Hierarchical clustering

[19]:
```
data.scaled <- scale(wisc.data)
data.dist <- dist(data.scaled)
wisc.hclust <- hclust(data.dist, method="complete")
```

2.6 Results of hierarchical clustering

2.6.1 Q11. Using the plot() and abline() functions, what is the height at which the clustering model has 4 clusters?

[34]:
```
plot(wisc.hclust)
abline(h=19, col="red", lty=2)
```



2.7 Selecting number of clusters

[21]:
```
wisc.hclust.clusters <- cutree(wisc.hclust, k=4)
table(wisc.hclust.clusters, diagnosis)
```

```
                    diagnosis
wisc.hclust.clusters   B   M
                   1  12 165
                   2   2   5
                   3 343  40
                   4   0   2
```

2.7.1 Q12. Can you find a better cluster vs diagnoses match by cutting into a different number of clusters between 2 and 10?

clusters	B	M
1	357	210
2	0	2

[22]:
```
cluster_matches <- sapply(2:10, function(k) table(cutree(wisc.hclust, k),
 ↪diagnosis))
cluster_matches
```

```
[[1]]
   diagnosis
      B   M
  1 357 210
  2   0   2

[[2]]
   diagnosis
      B   M
  1 355 205
  2   2   5
  3   0   2

[[3]]
   diagnosis
      B   M
  1  12 165
  2   2   5
  3 343  40
  4   0   2

[[4]]
   diagnosis
      B   M
  1  12 165
  2   0   5
  3 343  40
  4   2   0
  5   0   2

[[5]]
   diagnosis
      B   M
  1  12 165
  2   0   5
  3 331  39
  4   2   0
```

```
  5  12   1
  6   0   2

[[6]]
   diagnosis
      B   M
  1  12 165
  2   0   3
  3 331  39
  4   2   0
  5  12   1
  6   0   2
  7   0   2

[[7]]
   diagnosis
      B   M
  1  12  86
  2   0  79
  3   0   3
  4 331  39
  5   2   0
  6  12   1
  7   0   2
  8   0   2

[[8]]
   diagnosis
      B   M
  1  12  86
  2   0  79
  3   0   3
  4 331  39
  5   2   0
  6  12   0
  7   0   2
  8   0   2
  9   0   1

[[9]]
    diagnosis
       B   M
  1   12  86
  2    0  59
  3    0   3
  4  331  39
  5    0  20
  6    2   0
```

```
7   12   0
8    0   2
9    0   2
10   0   1
```

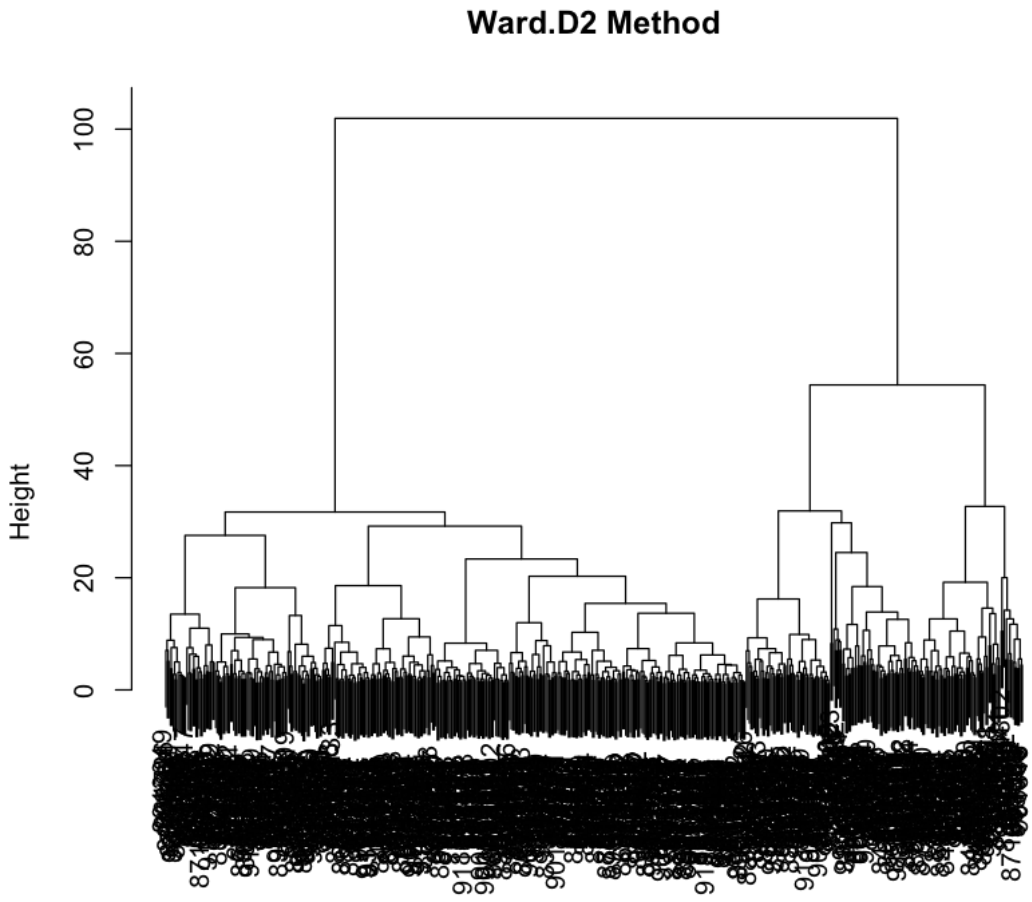
2.8 Using different methods

2.8.1 Q13. Which method gives your favorite results for the same data.dist dataset? Explain your reasoning.

- I like the Ward.D2 method because it not only visually organizes the data but it also makes it easier to interpret the results, although we would have to zoom in to the results to determine this.

https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/hclust

[23]:
```
wisc.pr.hclust <- hclust(dist(wisc.pr$x), method="ward.D2")
plot(wisc.pr.hclust, main="Ward.D2 Method", sub="", xlab="", ylab="Height")
```



3 4. OPTIONAL: K-means clustering

3.1 K-means clustering and comparing results

[24]:
```
wisc.km <- kmeans(scale(wisc.data), centers=2, nstart=20)
table(wisc.km$cluster, diagnosis)
```

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

3.1.1 Q14. How well does k-means separate the two diagnoses? How does it compare to your hclust results?

It is good, it almost separates malignant and benign cases

[25]:
```
table(wisc.km$cluster, diagnosis)
```

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

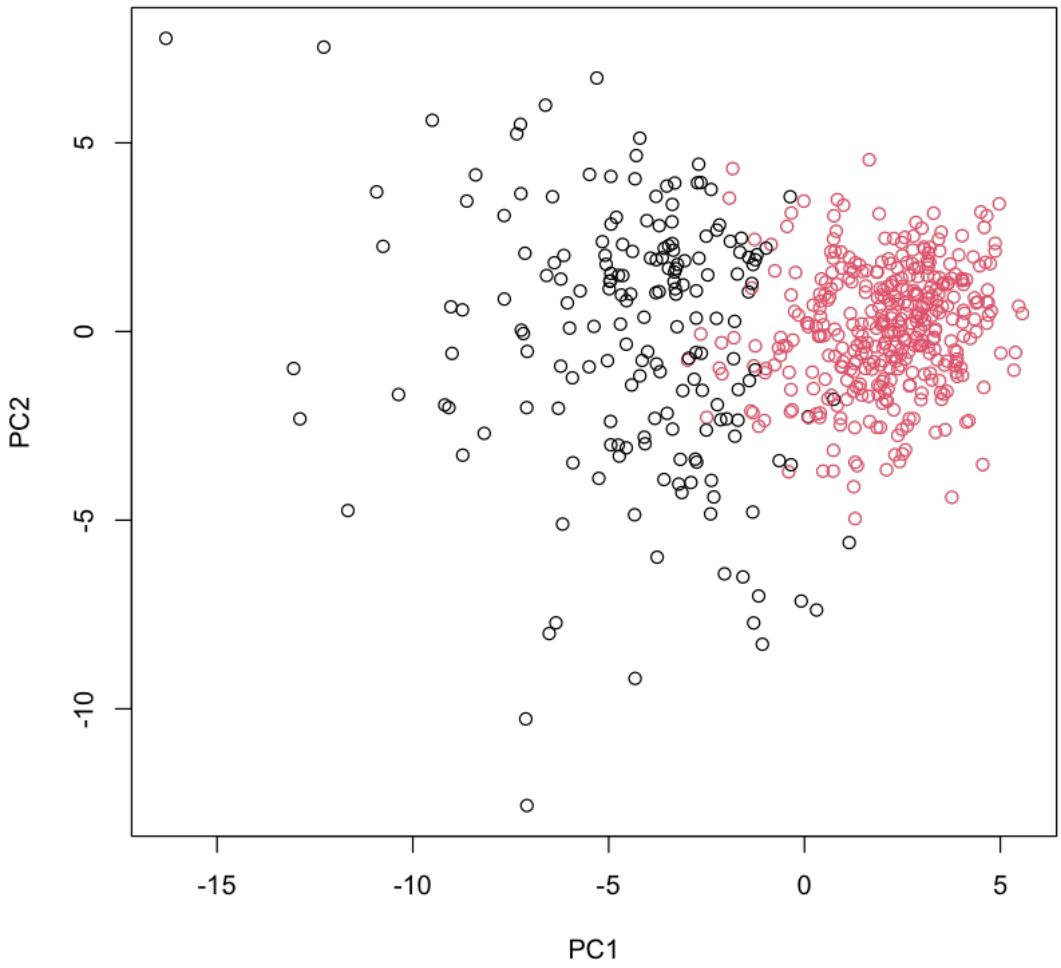
4 5. Combining methods

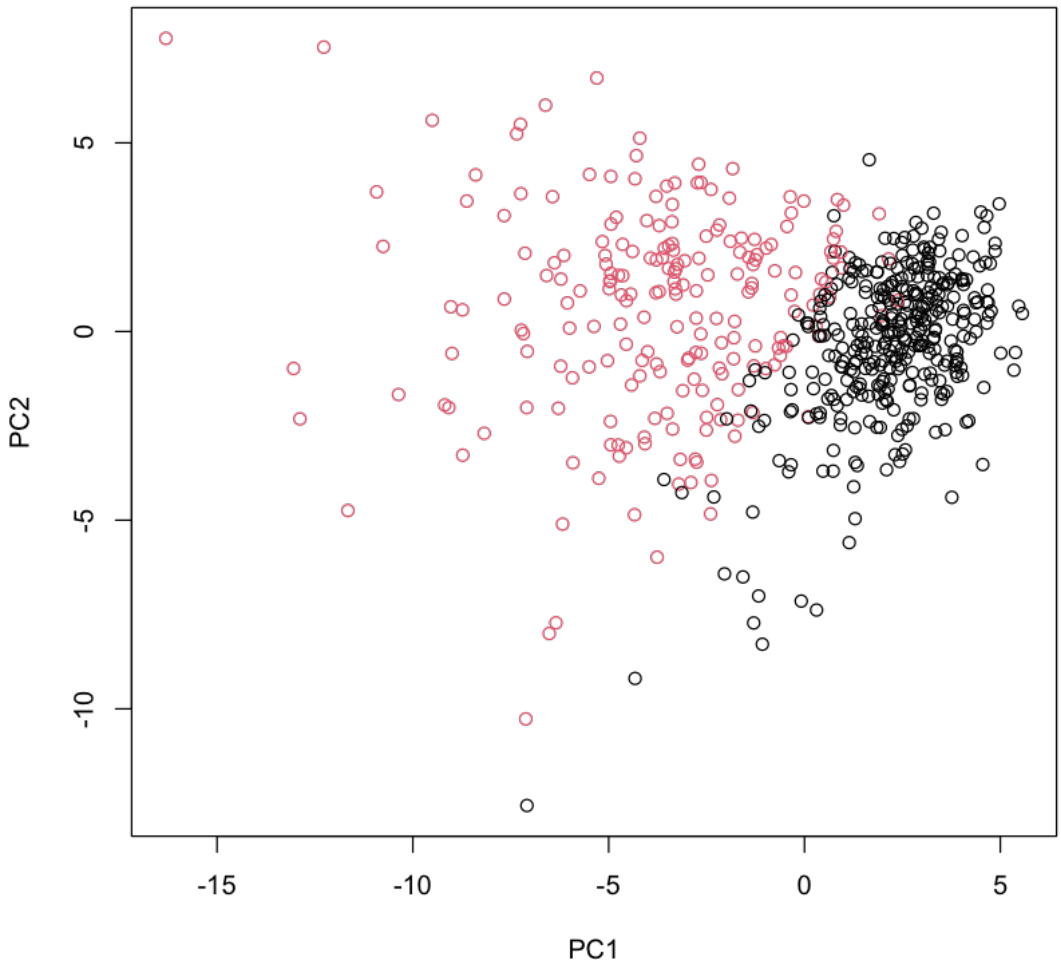
4.1 Clustering on PCA results

[26]:
```
grps <- cutree(wisc.pr.hclust, k=2)
table(grps)
table(grps, diagnosis)
plot(wisc.pr$x[,1:2], col=grps)
plot(wisc.pr$x[,1:2], col=diagnosis)
```

```
grps
  1   2
184 385
```

```
    diagnosis
grps   B   M
   1  20 164
   2 337  48
```

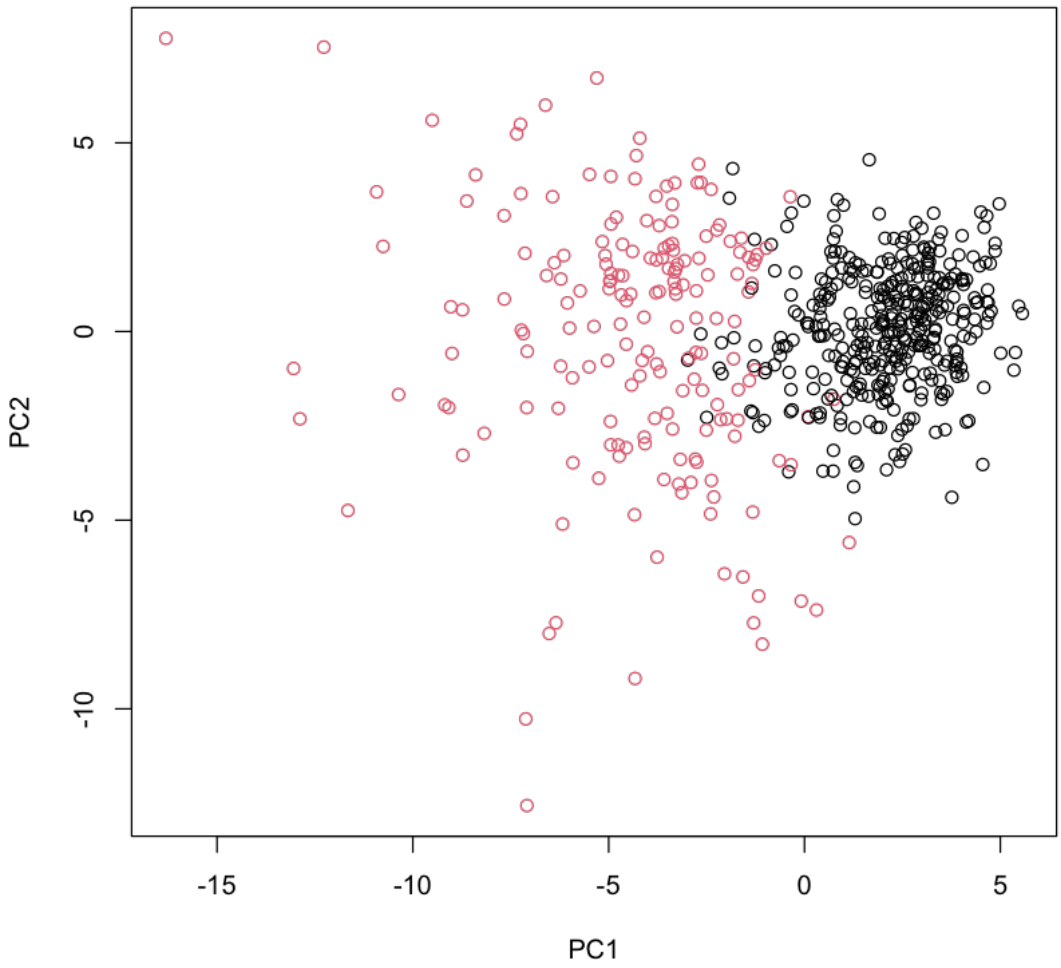

PC2
5
0
-5
-10
-15
-10
-5
0
5
PC1



[27]:
```
g <- as.factor(grps)
levels(g)

g <- relevel(g,2)
levels(g)

# Plot using our re-ordered factor
plot(wisc.pr$x[,1:2], col=g)
```

1. '1' 2. '2'

1. '2' 2. '1'



[28]:
```
library(rgl)
plot3d(wisc.pr$x[,1:3], xlab="PC 1", ylab="PC 2", zlab="PC 3", cex=1.5, size=1,␣
  ↪type="s", col=grps)

## Use the distance along the first 7 PCs for clustering i.e. wisc.pr$x[, 1:7]
wisc.pr.hclust <- hclust(dist(wisc.pr$x[, 1:7]), method="ward.D2")

wisc.pr.hclust.clusters <- cutree(wisc.pr.hclust, k=2)
```

```
Warning message in rgl.init(initValue, onlyNULL):
"RGL: unable to open X11 display"
Warning message:
"'rgl.init' failed, running with 'rgl.useNULL = TRUE'."
```

4.1.1 Q15. How well does the newly created model with four clusters separate out the two diagnoses?

It is not that great, rather it is mixed

[29]:
```
# Compare to actual diagnoses
table(wisc.pr.hclust.clusters, diagnosis)
```

```
                    diagnosis
wisc.pr.hclust.clusters   B   M
                      1  28 188
                      2 329  24
```

4.1.2 Q16. How well do the k-means and 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.

K-means shows to be better than hierarchical clustering

[30]:
```
table(wisc.km$cluster, diagnosis)
table(wisc.hclust.clusters, diagnosis)
```

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

```
                   diagnosis
wisc.hclust.clusters   B   M
                   1  12 165
                   2   2   5
                   3 343  40
                   4   0   2
```

4.2 6. Sensitivity/Specificity

4.2.1 Q17. Which of your analysis procedures resulted in a clustering model with the best specificity? How about sensitivity?

Hierarchical Clustering is best for specificity
- Specificity $= \frac{TN}{TN+FP} = \frac{343}{343+(357-343)} = \frac{343}{343+14} = \frac{343}{357} = 0.961$

k-means is best for sensitivity - Sensitivity $= \frac{TP}{TP+FN} = \frac{165}{165+(212-165)} = \frac{165}{165+47} = \frac{165}{212} = 0.778$

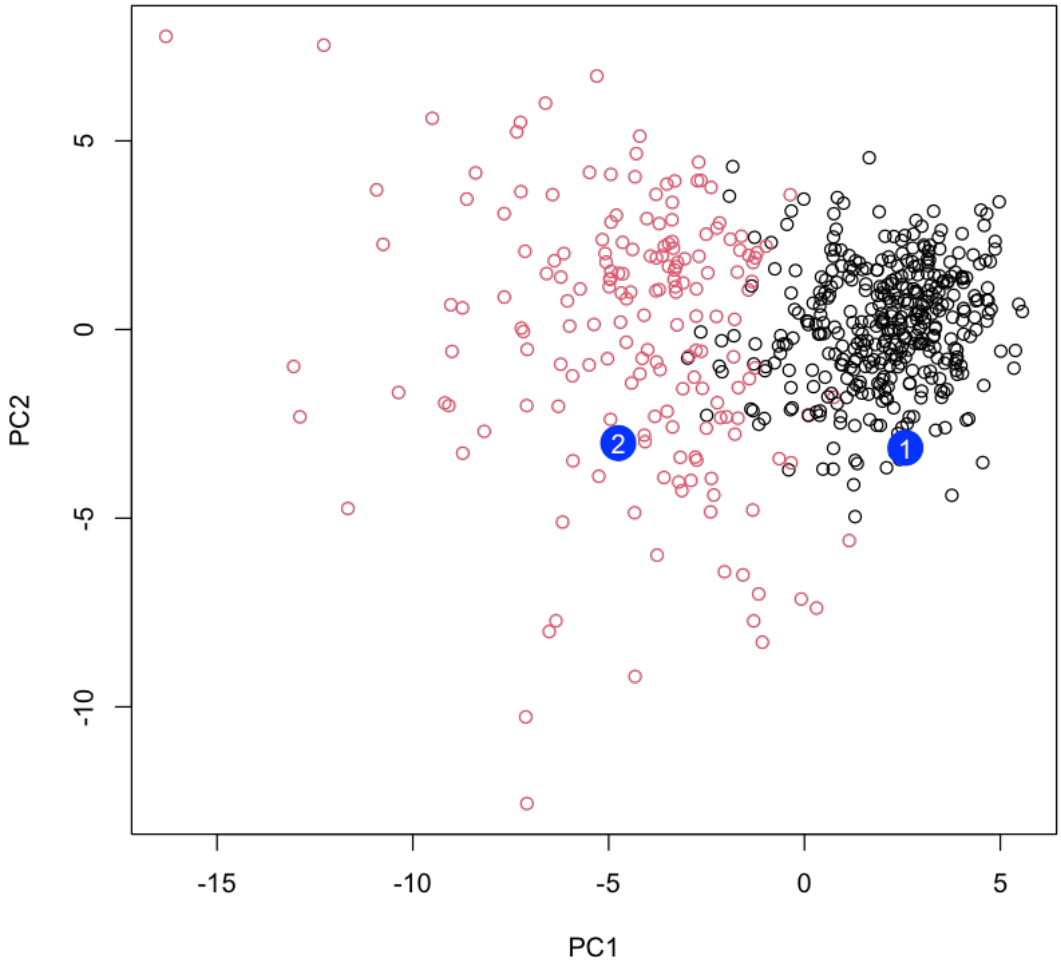
5 7. Prediction

[31]:
```
#url <- "new_samples.csv"
url <- "https://tinyurl.com/new-samples-CSV"
new <- read.csv(url)
npc <- predict(wisc.pr, newdata=new)
npc
```

A matrix: 2 × 30 of type dbl

PC1	PC2	PC3	PC4	PC5	PC6	PC7
2.576616	-3.135913	1.3990492	-0.7631950	2.781648	-0.8150185	-0.395909
-4.754928	-3.009033	-0.1660946	-0.6052952	-1.140698	-1.2189945	0.819303

[32]:
```
plot(wisc.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")
```



5.0.1 Q18. Which of these new patients should we prioritize for follow up based on your results?

Prioritize the new patients close or within the malignant clusters