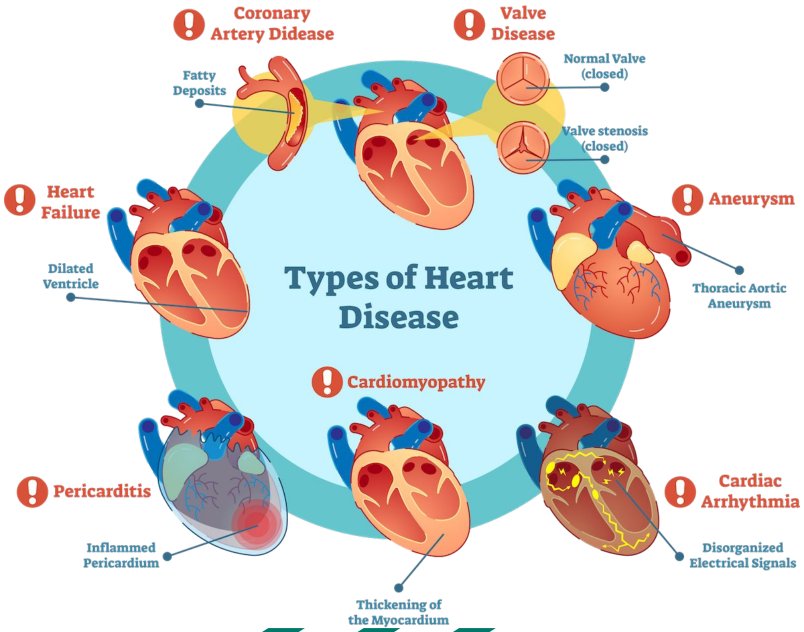
Predicting Heart Disease Status Using Logistic Regression and Clustering Algorithms

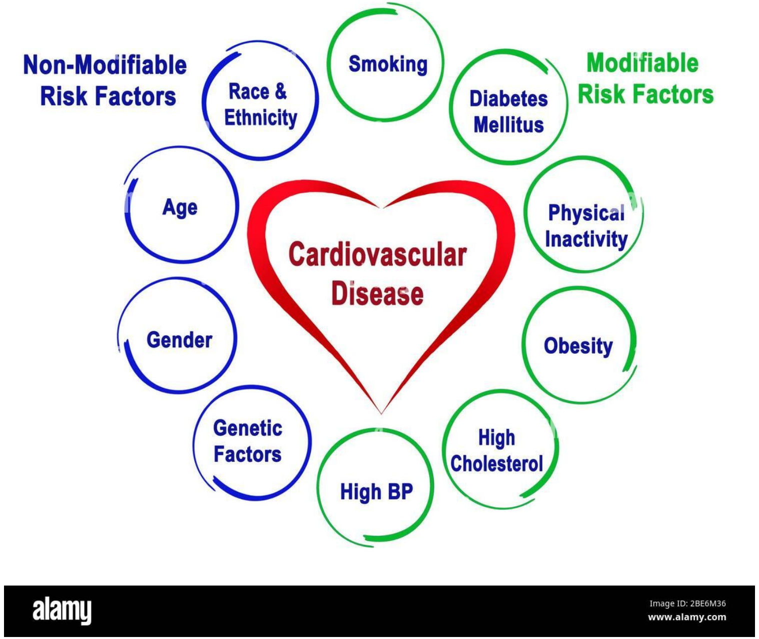
Professor Antonio Punzo

Phuong Huynh, Bufan Zhou, and Phuc Thinh Nguyen

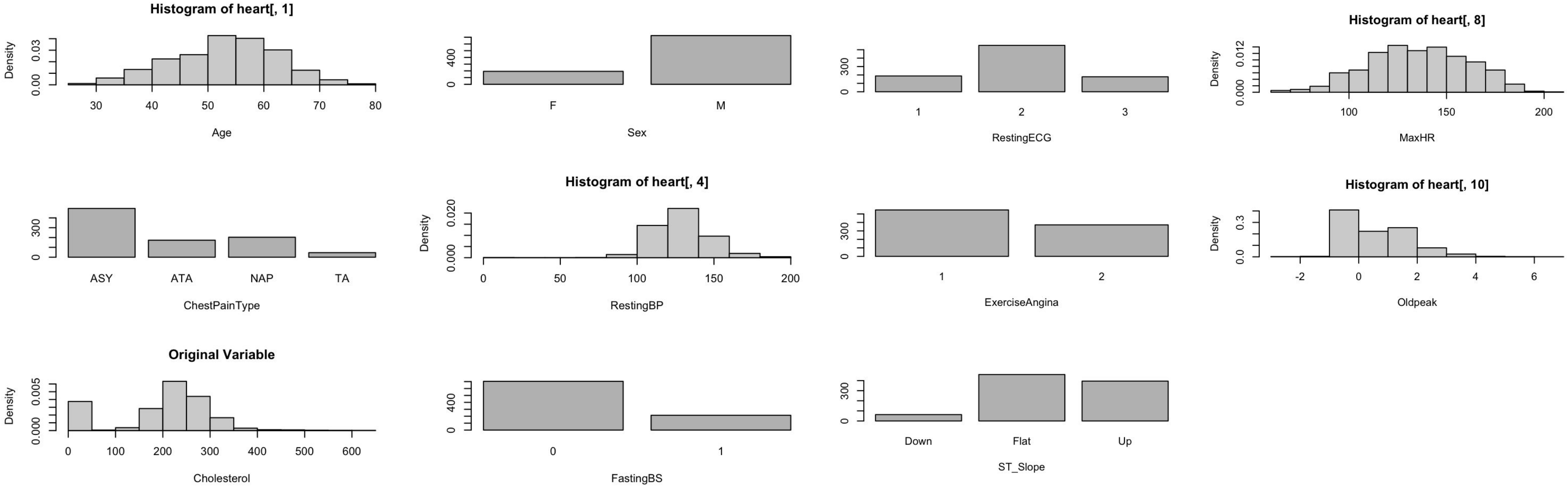
May 9th, 2024

Objective and Motivation

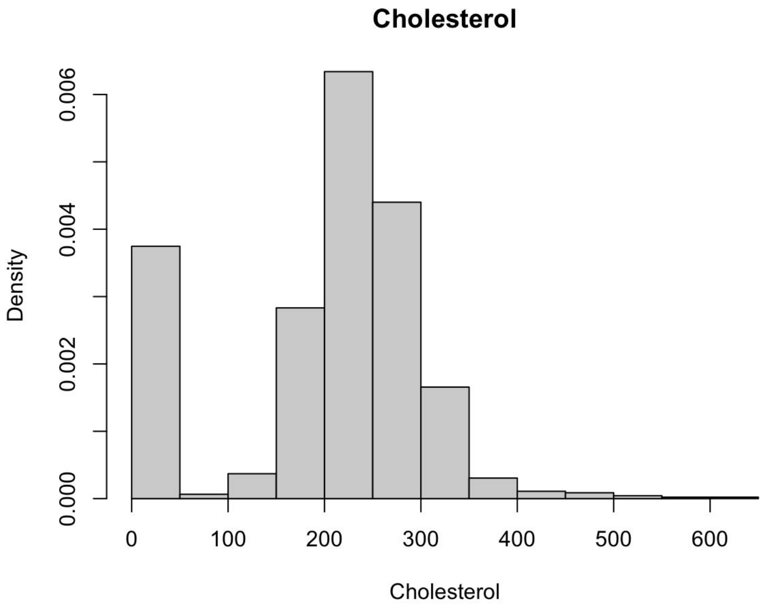


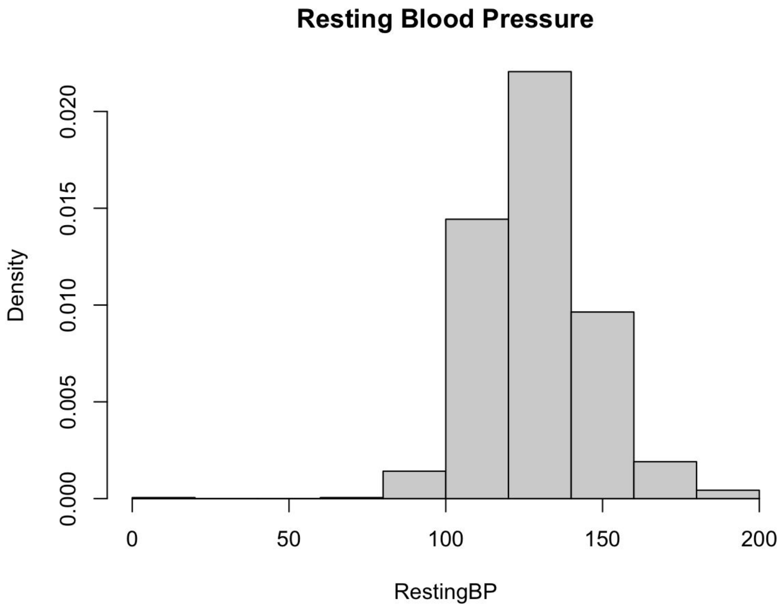


Analysis of univariate distributions

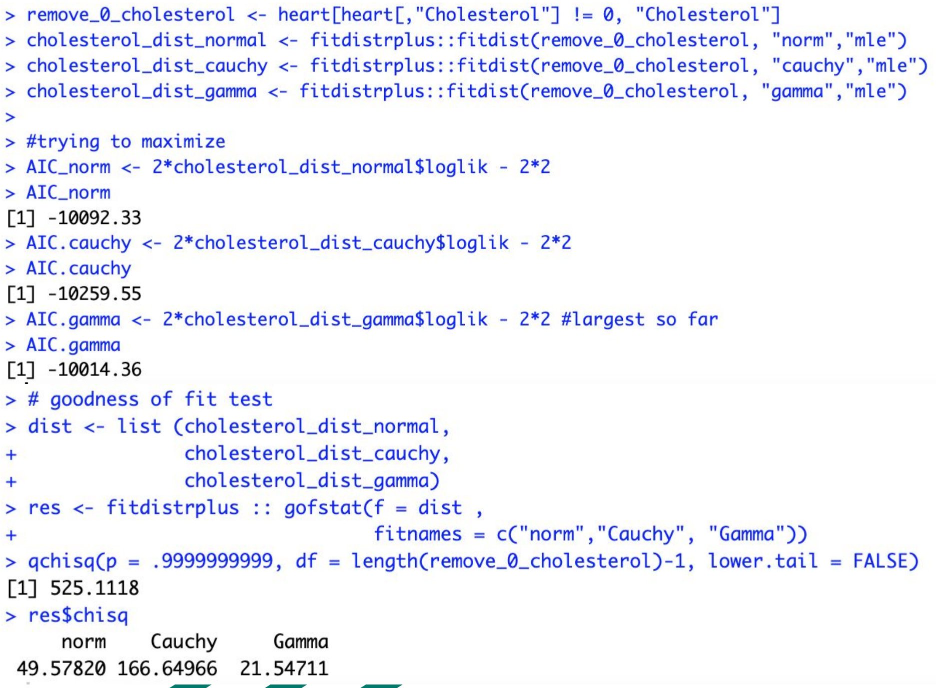


Analysis of univariate distributions

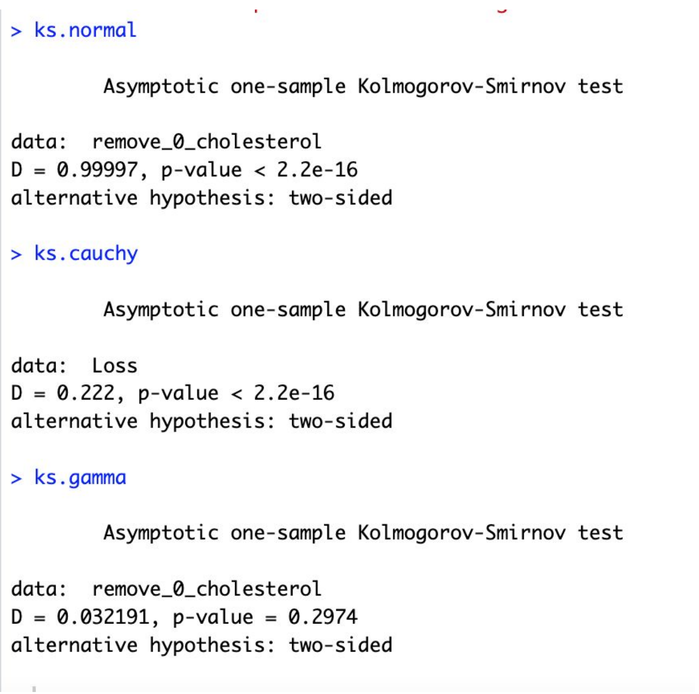




Analysis of univariate distributions

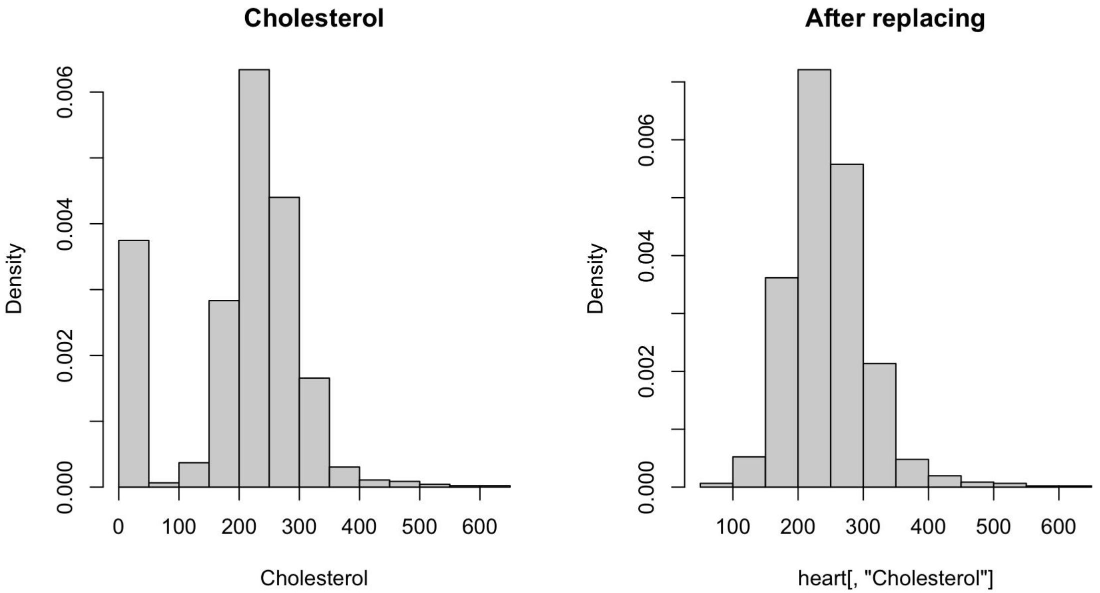


```
> remove_0_cholesterol <- heart[heart[,"Cholesterol"] != 0, "Cholesterol"]
> cholesterol_dist_normal <- fitdistrplus::fitdist(remove_0_cholesterol, "norm","mle")
> cholesterol_dist_cauchy <- fitdistrplus::fitdist(remove_0_cholesterol, "cauchy","mle")
> cholesterol_dist_gamma <- fitdistrplus::fitdist(remove_0_cholesterol, "gamma","mle")
>
> #trying to maximize
> AIC_norm <- 2*cholesterol_dist_normal$loglik - 2*2
> AIC_norm
[1] -10092.33
> AIC.cauchy <- 2*cholesterol_dist_cauchy$loglik - 2*2
> AIC.cauchy
[1] -10259.55
> AIC.gamma <- 2*cholesterol_dist_gamma$loglik - 2*2 #largest so far
> AIC.gamma
[1] -10014.36
> # goodness of fit test
> dist <- list (cholesterol_dist_normal,
+               cholesterol_dist_cauchy,
+               cholesterol_dist_gamma)
> res <- fitdistrplus :: gofstat(f = dist ,
+                                fitnames = c("norm","Cauchy", "Gamma"))
> qchisq(p = .9999999999, df = length(remove_0_cholesterol)-1, lower.tail = FALSE)
[1] 525.1118
> res$chisq
     norm    Cauchy     Gamma 
 49.57820 166.64966  21.54711 
```



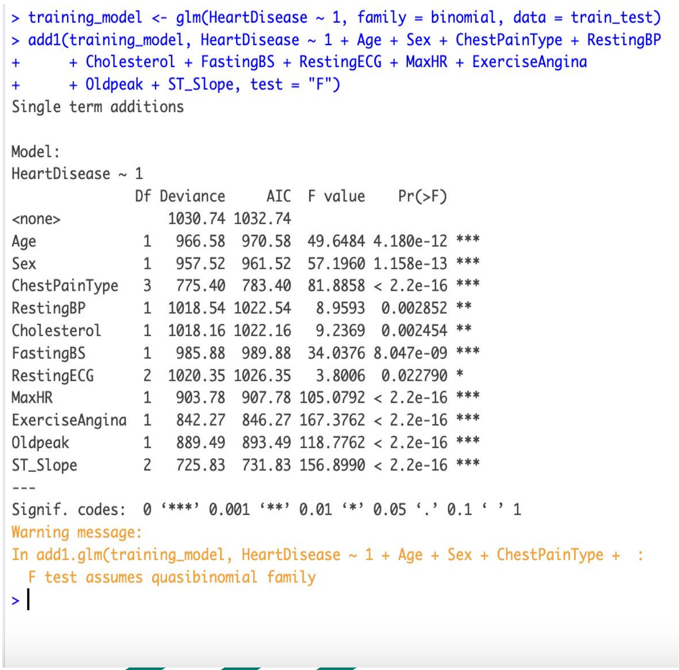
```
> ks.normal

	Asymptotic one-sample Kolmogorov-Smirnov test

data:  remove_0_cholesterol
D = 0.99997, p-value < 2.2e-16
alternative hypothesis: two-sided

> ks.cauchy

	Asymptotic one-sample Kolmogorov-Smirnov test

data:  Loss
D = 0.222, p-value < 2.2e-16
alternative hypothesis: two-sided

> ks.gamma

	Asymptotic one-sample Kolmogorov-Smirnov test

data:  remove_0_cholesterol
D = 0.032191, p-value = 0.2974
alternative hypothesis: two-sided
```

Analysis of univariate distributions



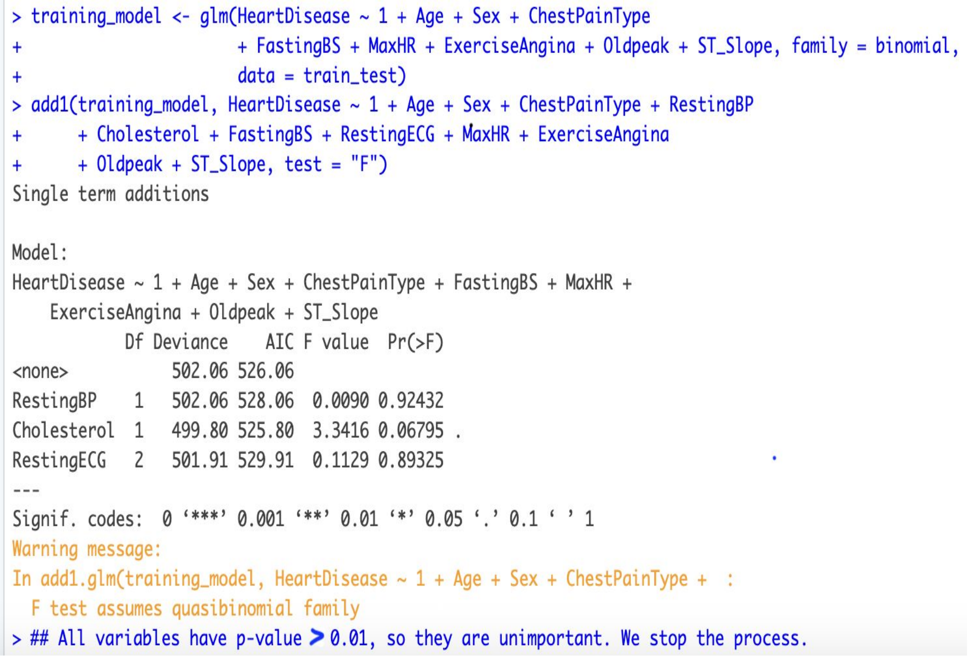
Logistic Regression Analysis

Forward Selection



```
> training_model <- glm(HeartDisease ~ 1, family = binomial, data = train_test)
> add1(training_model, HeartDisease ~ 1 + Age + Sex + ChestPainType + RestingBP
+      + Cholesterol + FastingBS + RestingECG + MaxHR + ExerciseAngina
+      + Oldpeak + ST_Slope, test = "F")
Single term additions

Model:
HeartDisease ~ 1
               Df Deviance     AIC  F value     Pr(>F)    
<none>             1030.74 1032.74                        
Age             1   966.58  970.58  49.6484 4.180e-12 ***
Sex             1   957.52  961.52  57.1960 1.158e-13 ***
ChestPainType   3   775.40  783.40  81.8858 < 2.2e-16 ***
RestingBP       1  1018.54 1022.54   8.9593  0.002852 ** 
Cholesterol     1  1018.16 1022.16   9.2369  0.002454 ** 
FastingBS       1   985.88  989.88  34.0376 8.047e-09 ***
RestingECG      2  1020.35 1026.35   3.8006  0.022790 *  
MaxHR           1   903.78  907.78 105.0792 < 2.2e-16 ***
ExerciseAngina  1   842.27  846.27 167.3762 < 2.2e-16 ***
Oldpeak         1   889.49  893.49 118.7762 < 2.2e-16 ***
ST_Slope        2   725.83  731.83 156.8990 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Warning message:
In add1.glm(training_model, HeartDisease ~ 1 + Age + Sex + ChestPainType +  :
  F test assumes quasibinomial family
>
```

...

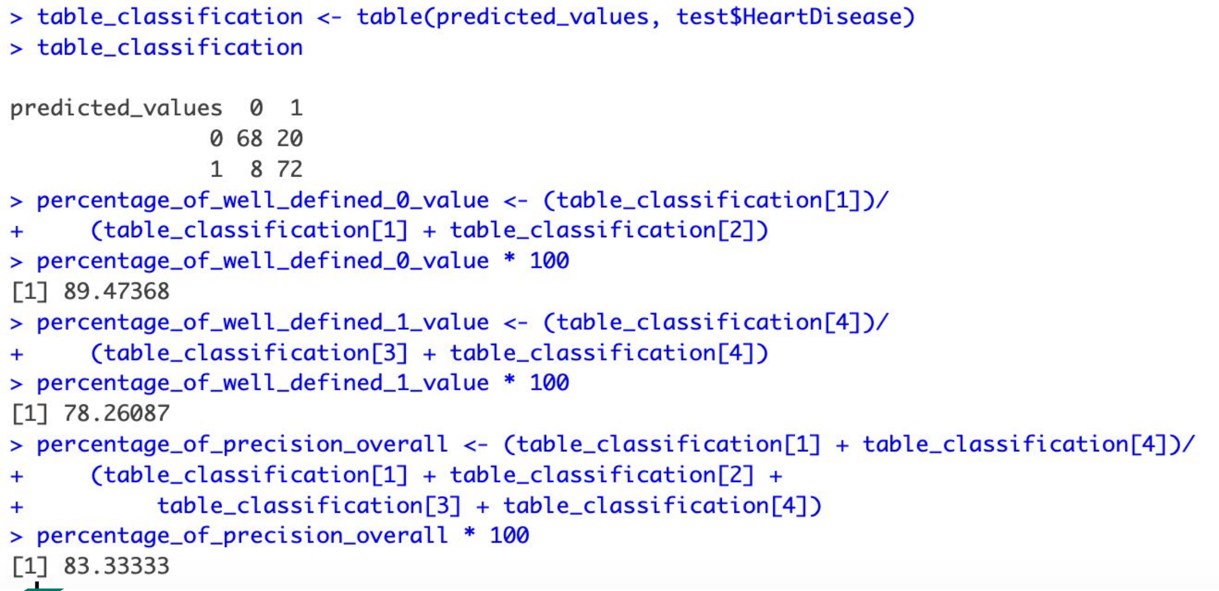


```
> training_model <- glm(HeartDisease ~ 1 + Age + Sex + ChestPainType
+                       + FastingBS + MaxHR + ExerciseAngina + Oldpeak + ST_Slope, family = binomial,
+                       data = train_test)
> add1(training_model, HeartDisease ~ 1 + Age + Sex + ChestPainType + RestingBP
+      + Cholesterol + FastingBS + RestingECG + MaxHR + ExerciseAngina
+      + Oldpeak + ST_Slope, test = "F")
Single term additions

Model:
HeartDisease ~ 1 + Age + Sex + ChestPainType + FastingBS + MaxHR + 
    ExerciseAngina + Oldpeak + ST_Slope
            Df Deviance    AIC F value  Pr(>F)  
<none>           502.06 526.06                  
RestingBP    1   502.06 528.06  0.0090 0.92432  
Cholesterol  1   499.80 525.80  3.3416 0.06795 .
RestingECG   2   501.91 529.91  0.1129 0.89325  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Warning message:
In add1.glm(training_model, HeartDisease ~ 1 + Age + Sex + ChestPainType +  :
  F test assumes quasibinomial family
> ## All variables have p-value > 0.01, so they are unimportant. We stop the process.
```

Logistic Regression Analysis

Confusion Matrix

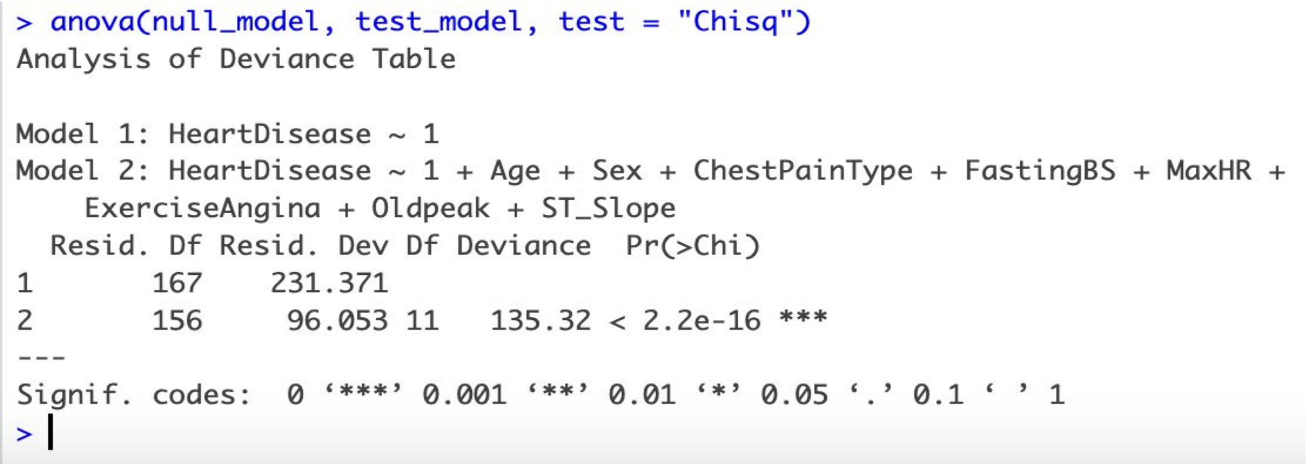
predicted_values <- predict(training_model, test, “response”)
predicted_values[predicted_values > 0.75] = 1
predicted_values[predicted_values <= 0.75] = 0



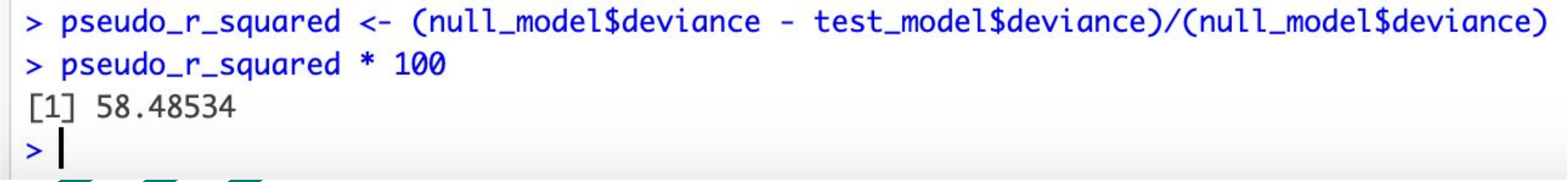
```
> table_classification <- table(predicted_values, test$HeartDisease)
> table_classification

predicted_values  0  1
               0 68 20
               1  8 72
> percentage_of_well_defined_0_value <- (table_classification[1])/
+     (table_classification[1] + table_classification[2])
> percentage_of_well_defined_0_value * 100
[1] 89.47368
> percentage_of_well_defined_1_value <- (table_classification[4])/
+     (table_classification[3] + table_classification[4])
> percentage_of_well_defined_1_value * 100
[1] 78.26087
> percentage_of_precision_overall <- (table_classification[1] + table_classification[4])/
+     (table_classification[1] + table_classification[2] +
+          table_classification[3] + table_classification[4])
> percentage_of_precision_overall * 100
[1] 83.33333
```

Logistic Regression Analysis

Pseudo R-squared



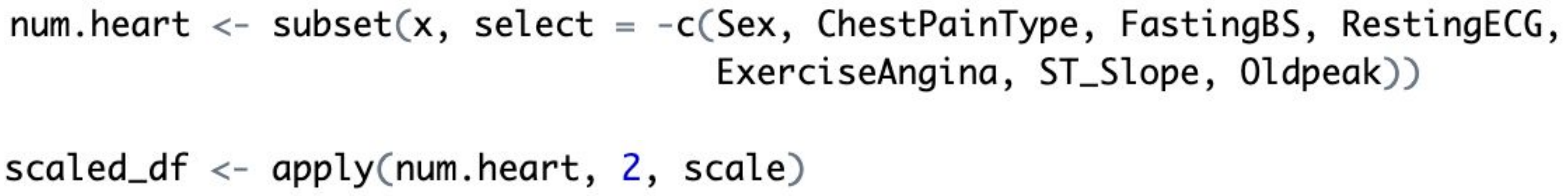
```
> anova(null_model, test_model, test = "Chisq")
Analysis of Deviance Table

Model 1: HeartDisease ~ 1
Model 2: HeartDisease ~ 1 + Age + Sex + ChestPainType + FastingBS + MaxHR + 
    ExerciseAngina + Oldpeak + ST_Slope
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)    
1       167    231.371                          
2       156     96.053 11   135.32 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> 
```



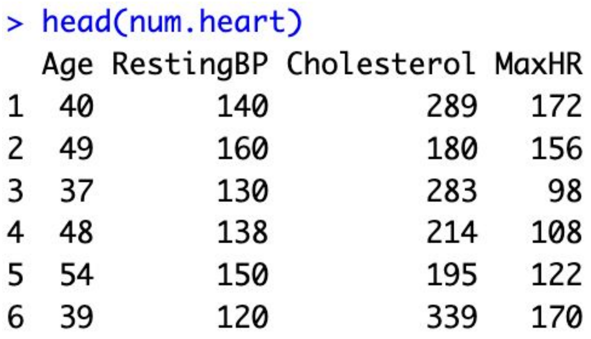
```
> pseudo_r_squared <- (null_model$deviance - test_model$deviance)/(null_model$deviance)
> pseudo_r_squared * 100
[1] 58.48534
> 
```

Principal Component Analysis

- Preparing for the data

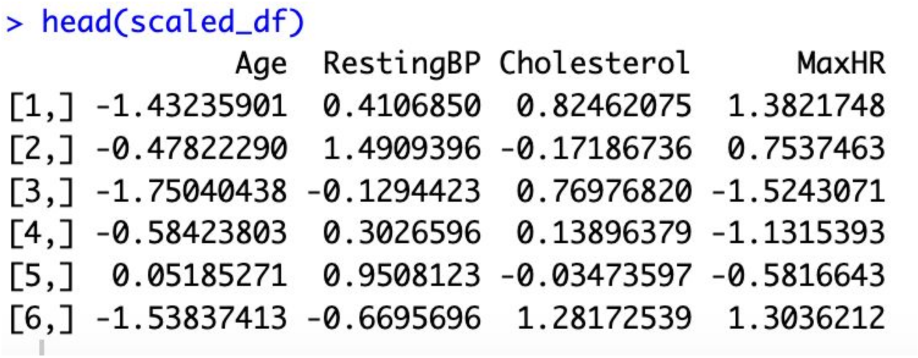


```
num.heart <- subset(x, select = -c(Sex, ChestPainType, FastingBS, RestingECG,
                                   ExerciseAngina, ST_Slope, Oldpeak))

scaled_df <- apply(num.heart, 2, scale)
```



```
> head(num.heart)
  Age RestingBP Cholesterol MaxHR
1  40       140         289   172
2  49       160         180   156
3  37       130         283    98
4  48       138         214   108
5  54       150         195   122
6  39       120         339   170
```

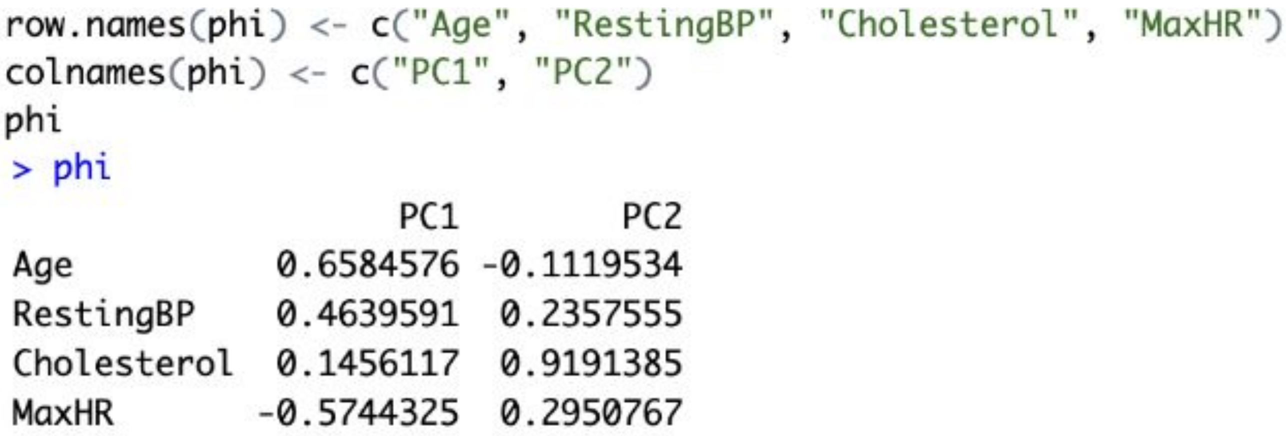
>>>



```
> head(scaled_df)
             Age  RestingBP Cholesterol      MaxHR
[1,] -1.43235901  0.4106850  0.82462075  1.3821748
[2,] -0.47822290  1.4909396 -0.17186736  0.7537463
[3,] -1.75040438 -0.1294423  0.76976820 -1.5243071
[4,] -0.58423803  0.3026596  0.13896379 -1.1315393
[5,]  0.05185271  0.9508123 -0.03473597 -0.5816643
[6,] -1.53837413 -0.6695696  1.28172539  1.3036212
```

Principal Component Analysis

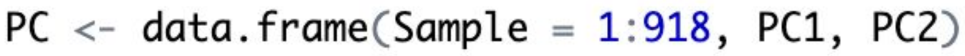
- eigenvalues & eigenvectors
- loadings



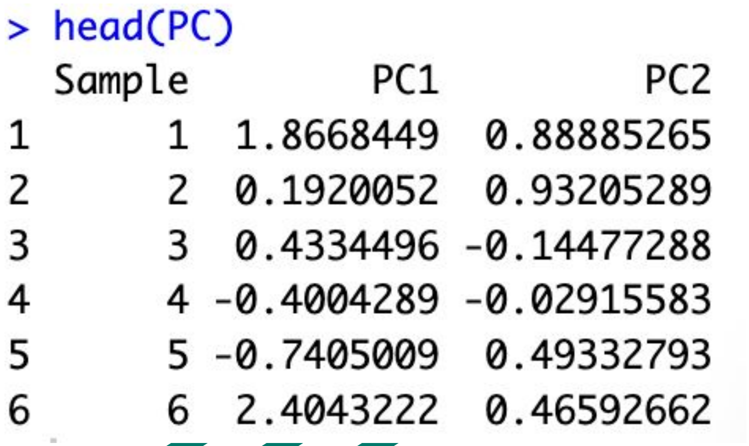
```
row.names(phi) <- c("Age", "RestingBP", "Cholesterol", "MaxHR")
colnames(phi) <- c("PC1", "PC2")
phi
> phi
                   PC1        PC2
Age          0.6584576 -0.1119534
RestingBP    0.4639591  0.2357555
Cholesterol  0.1456117  0.9191385
MaxHR       -0.5744325  0.2950767
```

Principal Component Analysis

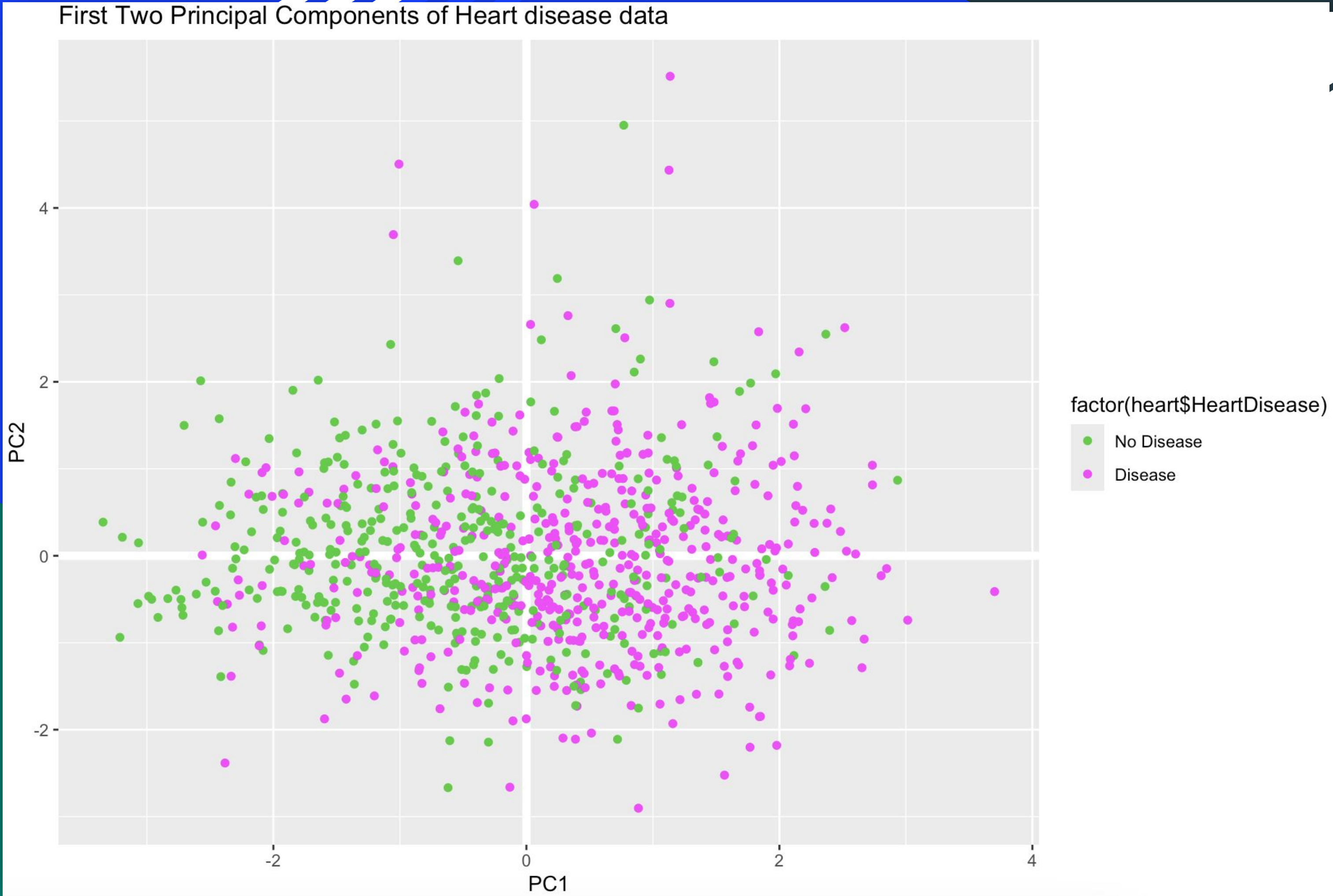
- PC for the data frame



```
PC <- data.frame(Sample = 1:918, PC1, PC2)
```

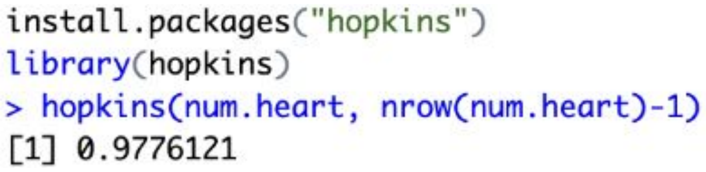


```
> head(PC)
  Sample        PC1         PC2
1      1  1.8668449  0.88885265
2      2  0.1920052  0.93205289
3      3  0.4334496 -0.14477288
4      4 -0.4004289 -0.02915583
5      5 -0.7405009  0.49332793
6      6  2.4043222  0.46592662
```

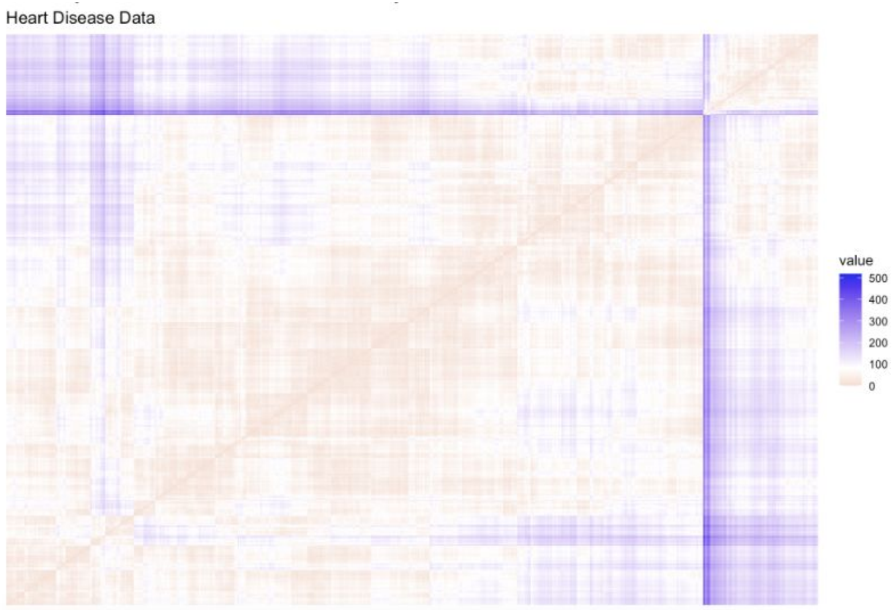

First Two Principal Components of Heart disease data
PC2
4
2
0
-2
PC1
-2
0
2
4
factor(heart$HeartDisease)
● No Disease
● Disease

Clustering Analysis

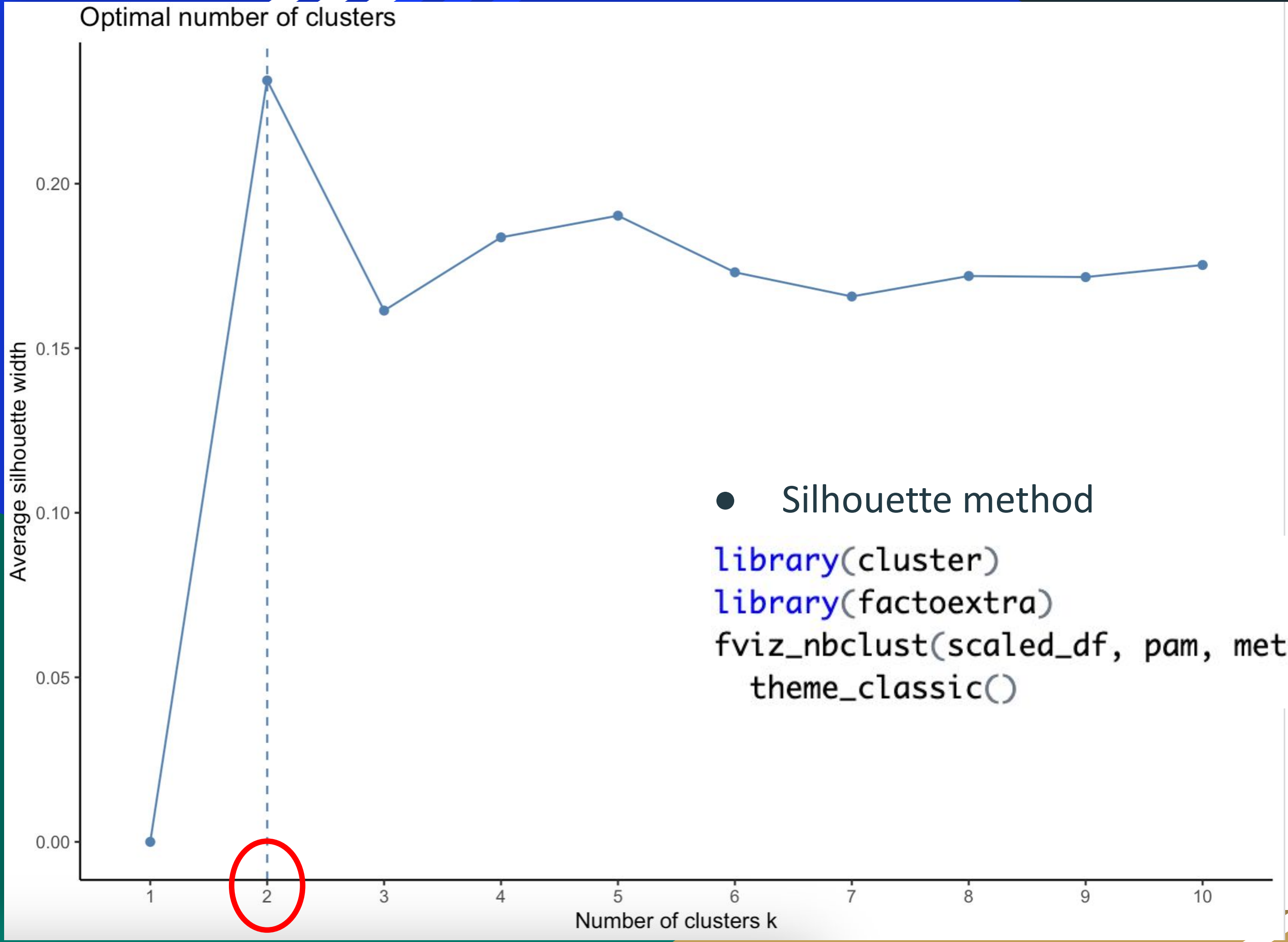
- Hopkins and VAT



```
install.packages("hopkins")
library(hopkins)
> hopkins(num.heart, nrow(num.heart)-1)
[1] 0.9776121
```



Heart Disease Data

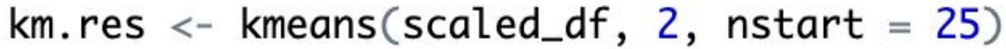


- Silhouette method

```
library(cluster)
library(factoextra)
fviz_nbclust(scaled_df, pam, method = "silhouette") +
  theme_classic()
```

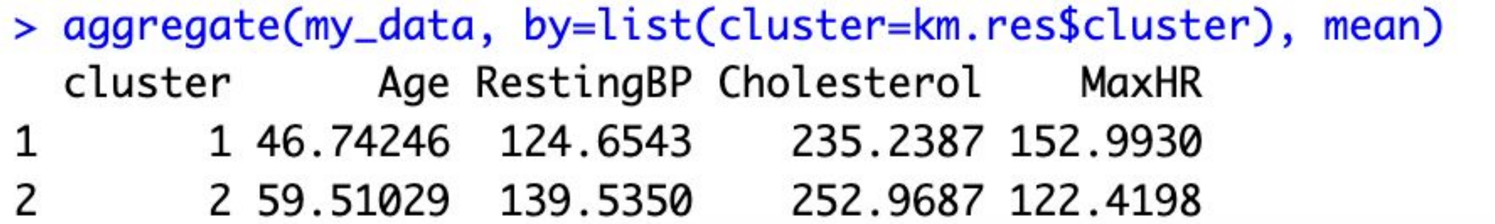
Clustering Analysis

- k-means(k = 2)

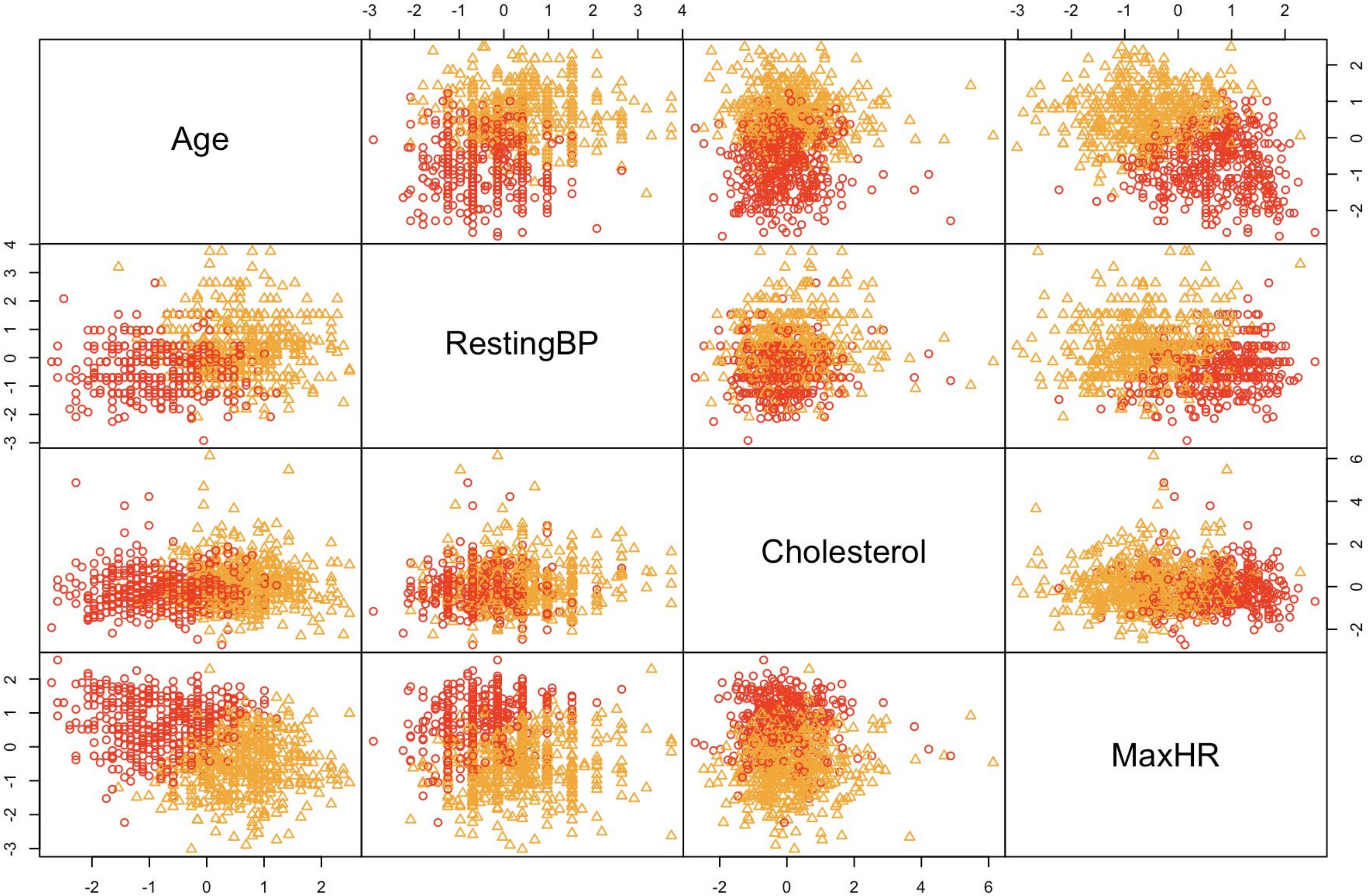


```
km.res <- kmeans(scaled_df, 2, nstart = 25)
```

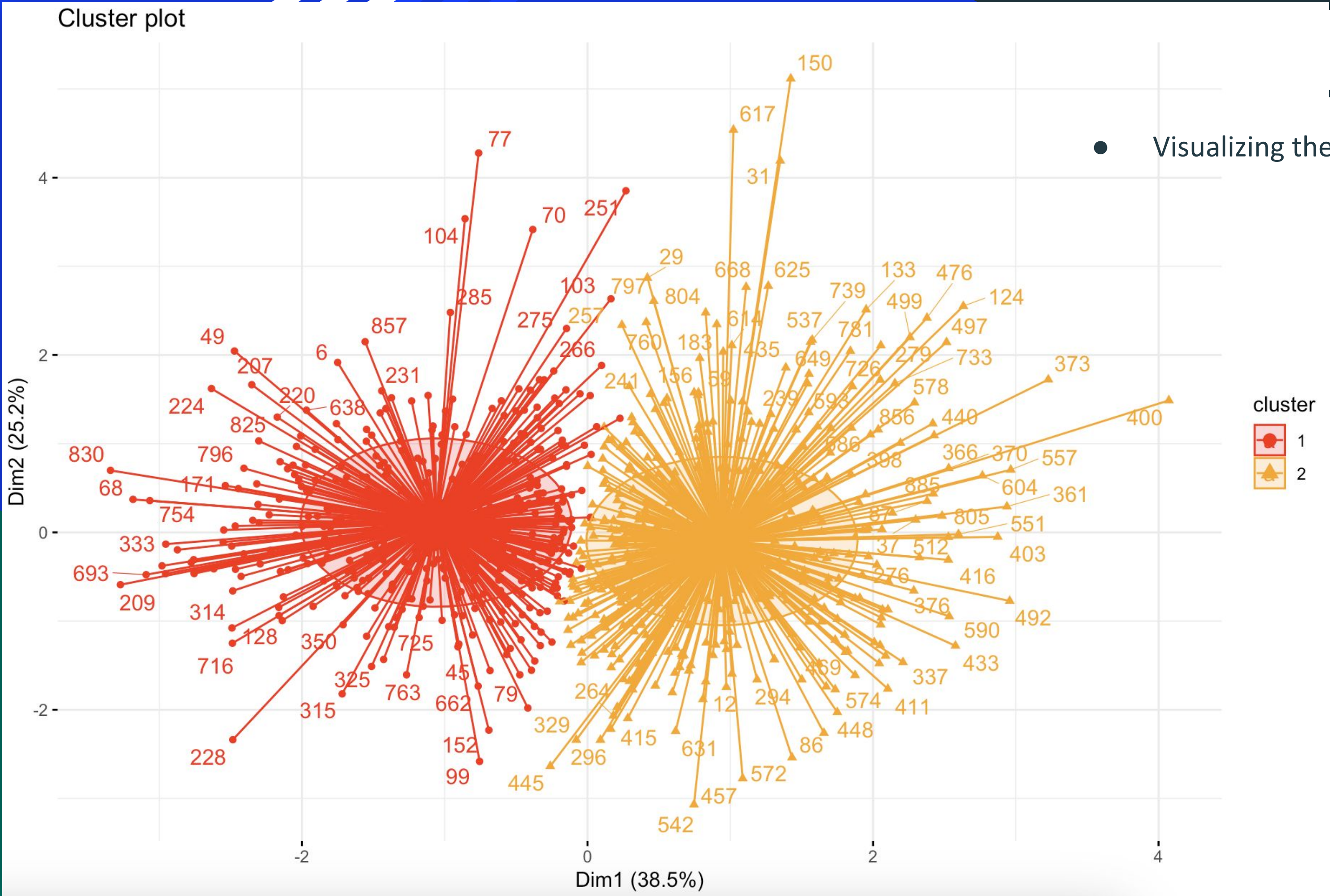
- means of these 2 clusters



```
> aggregate(my_data, by=list(cluster=km.res$cluster), mean)
  cluster      Age RestingBP Cholesterol    MaxHR
1       1 46.74246  124.6543    235.2387 152.9930
2       2 59.51029  139.5350    252.9687 122.4198
```



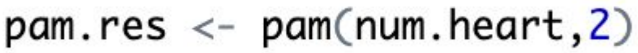
- apply the 2 clusters on the pairplot



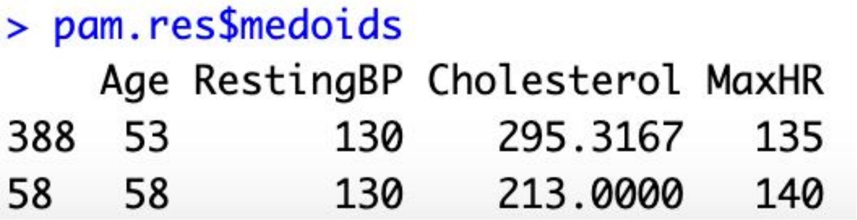
- Visualizing the 2-means on PC

Clustering Analysis

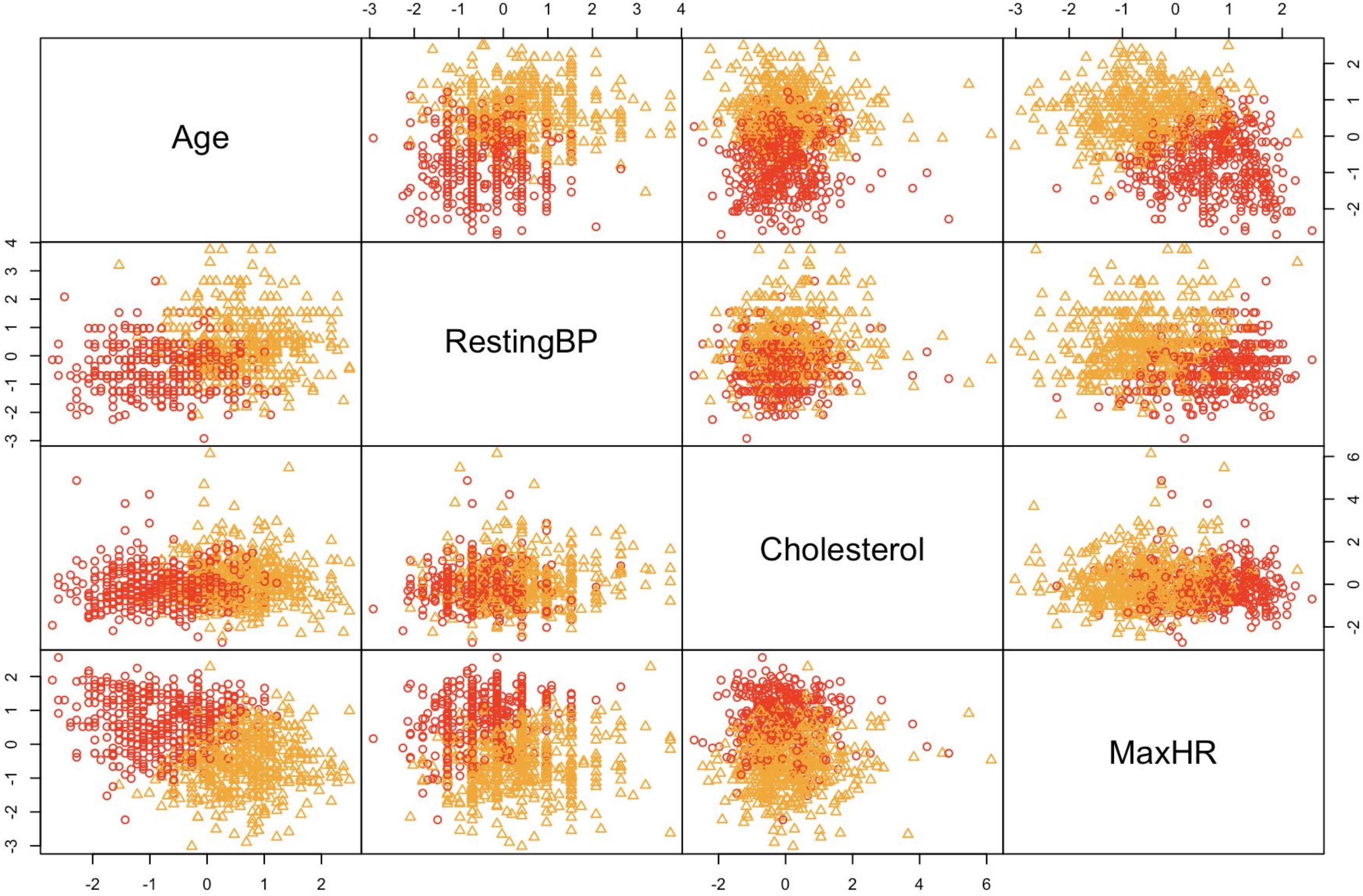
- k-medoids(k = 2)
- computing pam clustering



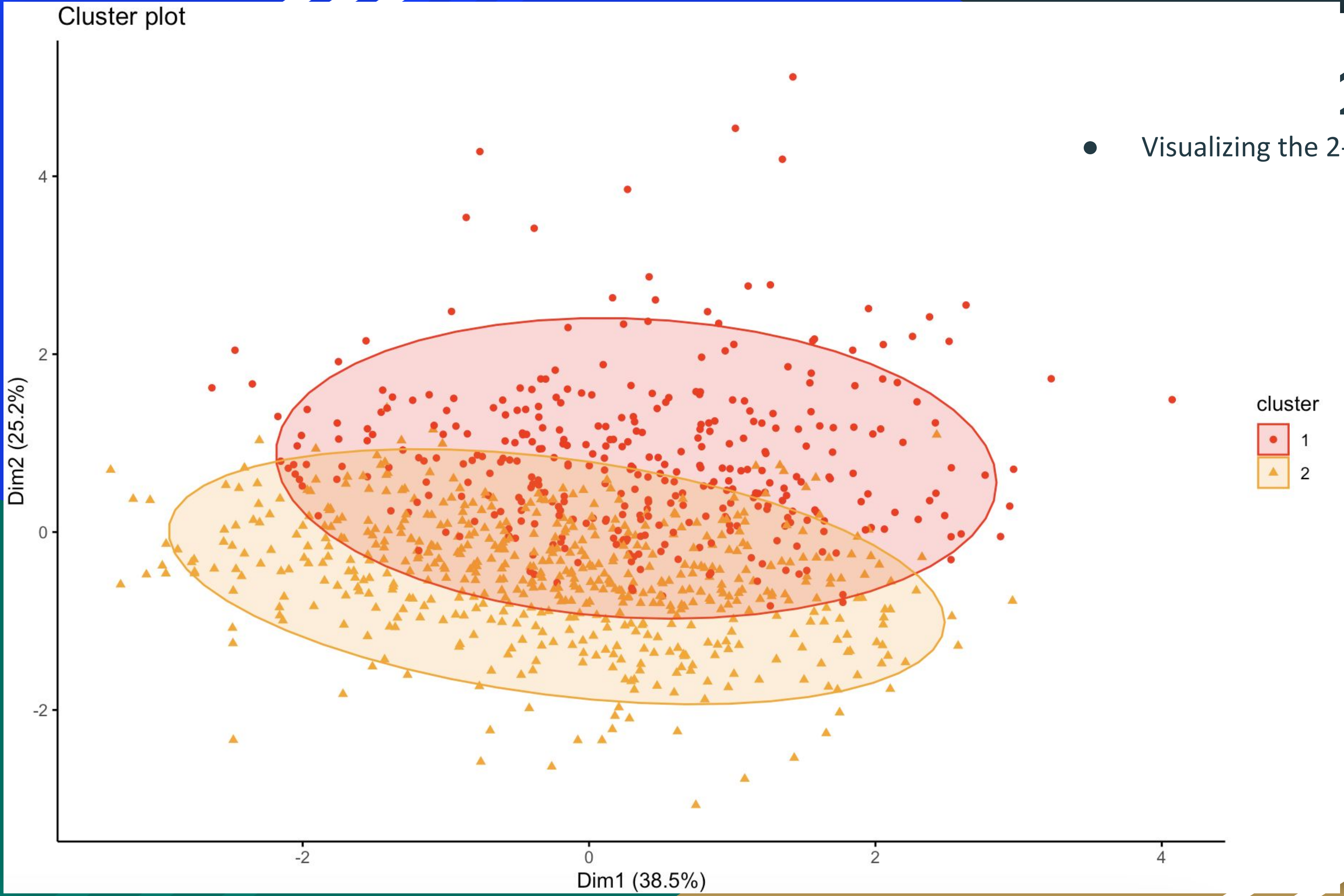
```
pam.res <- pam(num.heart,2)
```



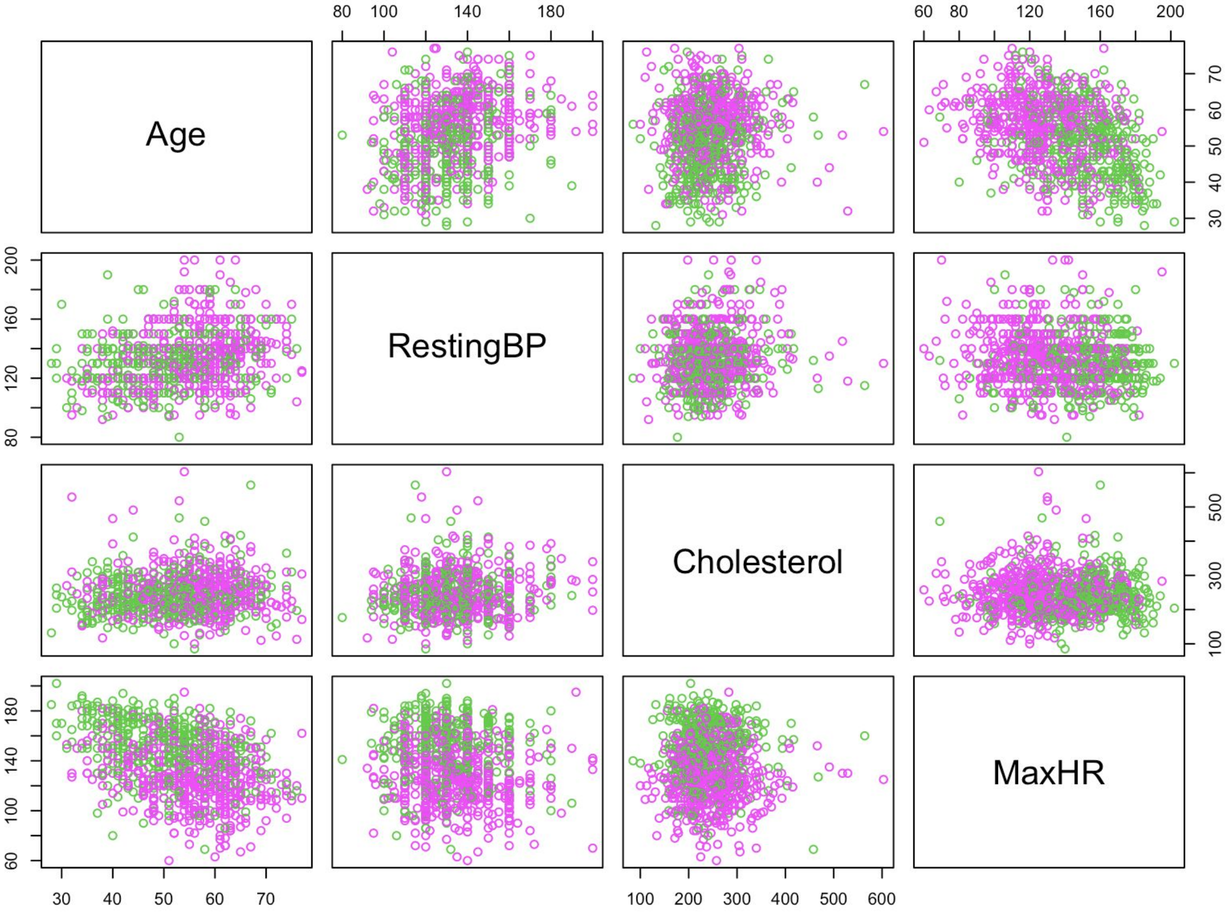
```
> pam.res$medoids
     Age RestingBP Cholesterol MaxHR
388   53       130    295.3167   135
58    58       130    213.0000   140
```



apply the 2 clusters on the pairplot



- Visualizing the 2-medoids on PC



Pair plot using true data group

● No Disease

● Disease

Conclusions

- **Logistic Regression:**
  - Most important variables are: age, sex, chestPainType, FastingBP,, MaxHR, ExerciseAngina, Oldpeak, and ST_slope (ST-segment).
  - 80% accuracy
  - R^2 = 58.4%
- **Clustering:**
  - Using different kinds of ways to find out the result.
  - The result of k-means(69.5%) are well-fitted to the real data with k-medoids(45.8%)

From: Phuong Huynh, Bufan Zhou, and Phuc Thinh Nguyen
To: Everyone

Thank you everyone for your close attention to the presentation.
Thank you Professor Antonio Punzo for giving us amazing lectures.