Cardiovascular Disease Prediction

Data reading:

The data required is readily available in the open UCI Machine learning repository https://archive.ics.uci.edu/ml/datasets/heart+disease. But it is split up based on location and is in a custom format with the extension `.data`. We can read it as table and merge it. The column names are available in the file `heart-disease.names`.

Read the data

```
# Read all processed data

cleaveland_data <- read.table("./data/processed.cleveland.data", fileEncoding = "UTF-8", sep = ",")
hungarian_data <- read.table("./data/processed.hungarian.data", fileEncoding = "UTF-8", sep = ",")
switzerland_data <- read.table("./data/processed.switzerland.data", fileEncoding = "UTF-8", sep = ",")
va_data <- read.table("./data/processed.va.data", fileEncoding = "UTF-8", sep = ",")
```

Print the dimensions of read data:

```
print("Dimensions of individual datasets :")
```

```
## [1] "Dimensions of individual datasets :"
```

```
print(dim(cleaveland_data))
```

```
## [1] 303  14
```

```
print(dim(hungarian_data))
```

```
## [1] 294  14
```

```
print(dim(switzerland_data))
```

```
## [1] 123  14
```

```
print(dim(va_data))
```

```
## [1] 200  14
```

Concatinate the data and assign column names:

```
# Concat all the datasets
tmp1 <- rbind(cleaveland_data, hungarian_data)
tmp2 <- rbind(switzerland_data, va_data)
heart_data <- rbind(tmp1, tmp2)

# Column names from heart-disease.names file
colnames(heart_data) <- c("age","sex","cp","trestbps","chol","fbs","restecg","thalach","exang","oldpeak
summary(heart_data)
```

```
##       age             sex               cp         trestbps
##  Min.   :28.00   Min.   :0.0000   Min.   :1.00   Length:920
##  1st Qu.:47.00   1st Qu.:1.0000   1st Qu.:3.00   Class :character
##  Median :54.00   Median :1.0000   Median :4.00   Mode  :character
##  Mean   :53.51   Mean   :0.7891   Mean   :3.25
##  3rd Qu.:60.00   3rd Qu.:1.0000   3rd Qu.:4.00
##  Max.   :77.00   Max.   :1.0000   Max.   :4.00
##      chol               fbs             restecg           thalach
##  Length:920         Length:920         Length:920         Length:920
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##     exang             oldpeak            slope               ca
##  Length:920         Length:920         Length:920         Length:920
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##      thal               goal
##  Length:920         Min.   :0.0000
##  Class :character   1st Qu.:0.0000
##  Mode  :character   Median :1.0000
##                     Mean   :0.9957
##                     3rd Qu.:2.0000
##                     Max.   :4.0000
```

```
print("Dimensions of combined data :")
```

```
## [1] "Dimensions of combined data :"
```

```
print(dim(heart_data))
```

```
## [1] 920  14
```

Remove all unnecessary columns with ? or :

```
heart_data [heart_data == "?"] <- NA
heart_data <- drop_na(heart_data)

# After removal of all the data we are down to 299 rows
dim(heart_data)
```

```
## [1] 299  14
```

See if the data types are okay

```
# print the data types
print(sapply(heart_data, class))
```

```
##         age         sex          cp    trestbps        chol         fbs
##   "numeric"   "numeric"   "numeric" "character" "character" "character"
##     restecg     thalach       exang     oldpeak       slope          ca
## "character" "character" "character" "character" "character" "character"
##        thal        goal
## "character"   "integer"
```

Fix the data types

```
# Data types are wrong, should update it based on data available from heart-disease.names
# Age should be a number
heart_data$age <- as.numeric(heart_data$age)

# Sex should be a factor (1 = male; 0 = female)
heart_data$sex <- as.factor(heart_data$sex)

# cp - chest pain should be a factor
# Value 1: typical angina
# Value 2: atypical angina
# Value 3: non-anginal pain
# Value 4: asymptomatic
heart_data$cp <- as.factor(heart_data$cp)

# trestbps - resting blood pressure
heart_data$trestbps <- as.numeric(heart_data$trestbps)

# chol - serum cholestoral in mg/dl
heart_data$chol <- as.numeric(heart_data$chol)

# fbs - If fasting blood sugar > 120 mg/dl,   (1 = true; 0 = false)
heart_data$fbs <- as.factor(heart_data$fbs)

# restecg - resting electrocardiographic results
# Value 0: normal
# Value 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 
```

```
# Value 2: showing probable or definite left ventricular hypertrophy by Estes' criteria
heart_data$restecg <- as.factor(heart_data$restecg)

# thalach: maximum heart rate achieved
heart_data$thalach <- as.numeric(heart_data$thalach)

# exang: exercise induced angina (1 = yes; 0 = no)
heart_data$exang <- as.factor(heart_data$exang)

# oldpeak = ST depression induced by exercise relative to rest
heart_data$oldpeak <- as.numeric(heart_data$oldpeak)

# slope: the slope of the peak exercise ST segment
# Value 1: upsloping
# Value 2: flat
# Value 3: downsloping
heart_data$slope <- as.factor(heart_data$slope)

# ca: number of major vessels (0-3) colored by flourosopy
heart_data$ca <- as.numeric(heart_data$ca)

# thal: 3 = normal; 6 = fixed defect; 7 = reversable defect
heart_data$thal <- as.factor(as.integer(heart_data$thal))

# goal: It distinguish presence (values 1,2,3,4) from absence (value 0)
heart_data$goal <- as.factor(heart_data$goal)

print("After manual updates of datatype")
```

```
## [1] "After manual updates of datatype"
```

```
# print the data types
print(sapply(heart_data, class))
```

```
##       age       sex        cp  trestbps      chol       fbs   restecg   thalach
## "numeric"  "factor"  "factor" "numeric" "numeric"  "factor"  "factor" "numeric"
##     exang   oldpeak     slope        ca      thal      goal
##  "factor" "numeric"  "factor" "numeric"  "factor"  "factor"
```

```
summary(heart_data)
```

```
##       age        sex     cp         trestbps          chol       fbs
##  Min.   :29.00   0: 96   1: 23   Min.   : 94.0   Min.   :100.0   0:256
##  1st Qu.:48.00   1:203   2: 49   1st Qu.:120.0   1st Qu.:211.0   1: 43
##  Median :56.00           3: 83   Median :130.0   Median :242.0
##  Mean   :54.52           4:144   Mean   :131.7   Mean   :246.8
##  3rd Qu.:61.00                   3rd Qu.:140.0   3rd Qu.:275.5
##  Max.   :77.00                   Max.   :200.0   Max.   :564.0
##  restecg    thalach      exang      oldpeak      slope         ca
##  0:149   Min.   : 71.0   0:200   Min.   :0.000   1:139   Min.   :0.0000
##  1:  4   1st Qu.:132.5   1: 99   1st Qu.:0.000   2:139   1st Qu.:0.0000
##  2:146   Median :152.0           Median :0.800   3: 21   Median :0.0000
```

```
##           Mean   :149.3          Mean   :1.059          Mean   :0.6722
##           3rd Qu.:165.5          3rd Qu.:1.600          3rd Qu.:1.0000
##           Max.   :202.0          Max.   :6.200          Max.   :3.0000
##  thal     goal
##  3:164    0:160
##  6: 18    1: 56
##  7:117    2: 35
##           3: 35
##           4: 13
##
```

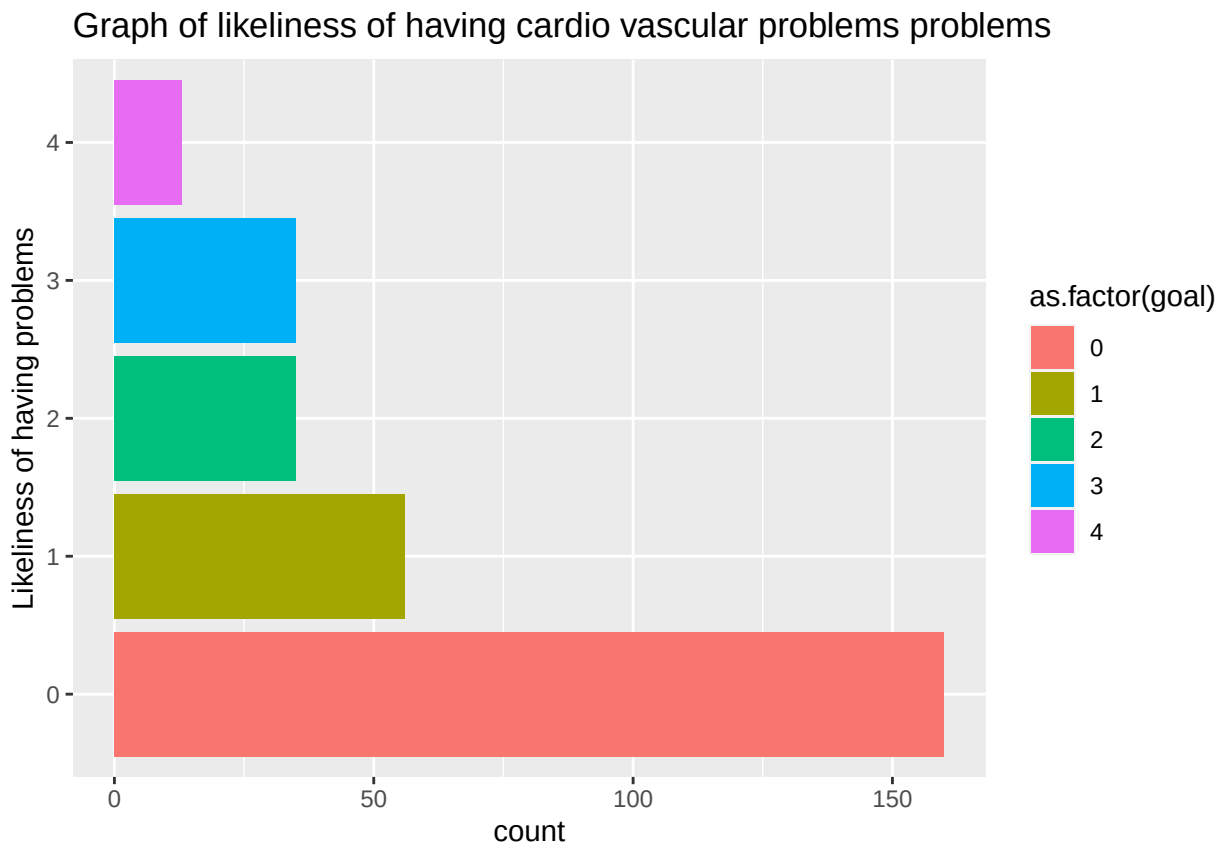
Write the data as csv to local machine:

```
write.csv(heart_data,"./data/heart_data.csv", row.names = FALSE)
```

Data exploration:

Plot a graph on likeliness of people having a cardio-vascular problems with 0 as abscense and (1,2,3,4) having problems.

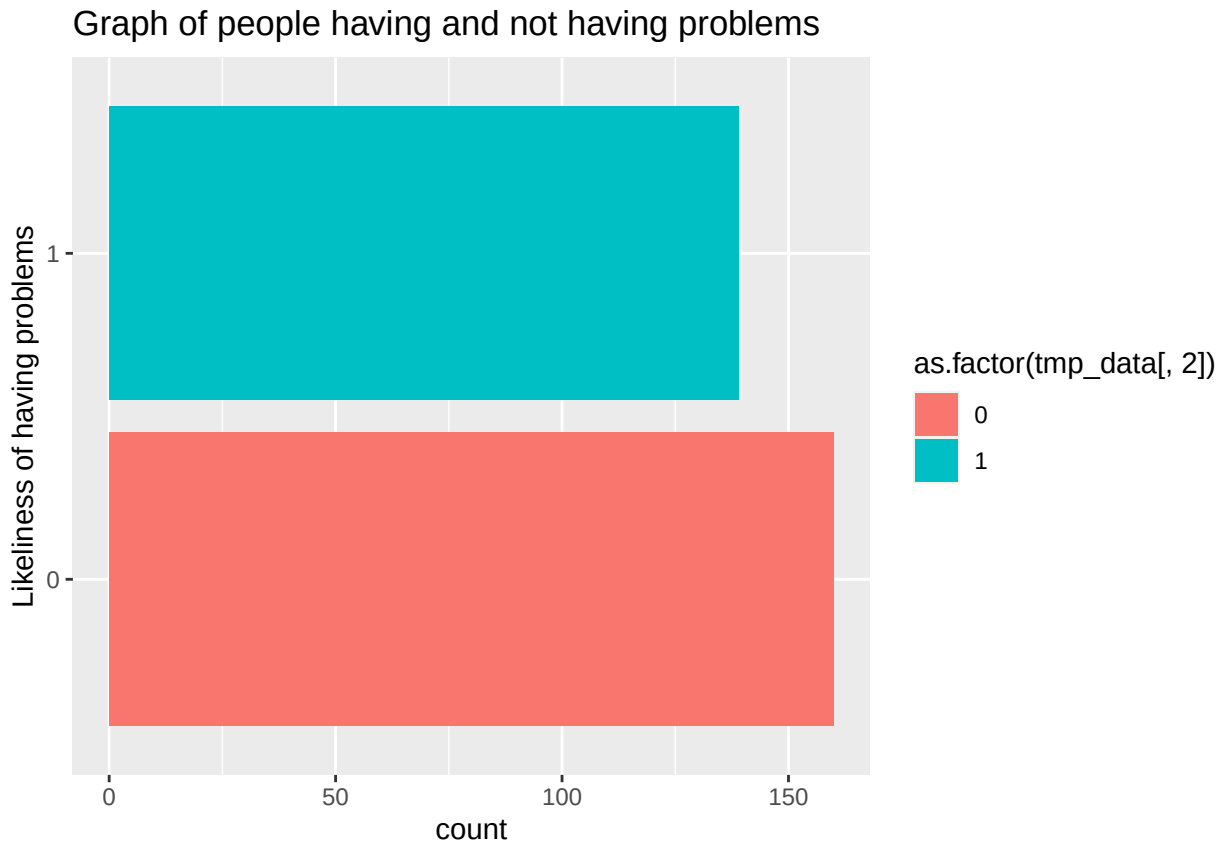
```
ggplot(heart_data, aes(x=as.factor(goal), fill=as.factor(goal) )) +
  geom_bar() +
  xlab("Likeliness of having problems") +
  ggtitle("Graph of likeliness of having cardio vascular problems problems") +
  coord_flip()
```



Plot a graph on likeliness of people having a cardio-vascular problems with 0 as abscense and 1 as having problems.

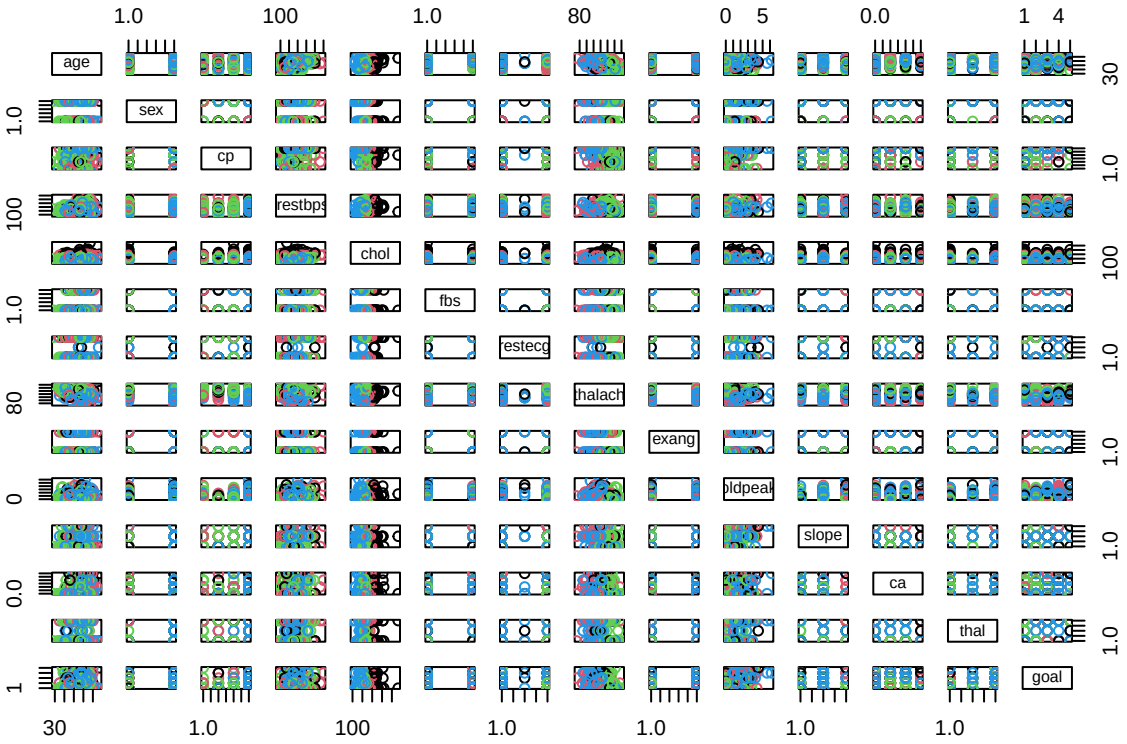
```
tmp_data <- cbind(heart_data$goal, ifelse(heart_data$goal!= 0, 1, 0))

ggplot(heart_data, aes(x=as.factor(tmp_data[, 2]), fill=as.factor(tmp_data[, 2]) )) +
  geom_bar() +
  xlab("Likeliness of having problems") +
  ggtitle("Graph of people having and not having problems") +
  coord_flip()
```



Apply kmeans to see how the clusters are formed

```
kmeans.res <- kmeans(heart_data, 4)
plot(heart_data, col=kmeans.res$cluster)
```


age
sex
cp
restbps
chol
fbs
restecg
thalach
exang
oldpeak
slope
ca
thal
goal