Code ▾

K Means Clustering on diabetes dataset

Install and load libraries : mlr

Hide

```
# install.packages("mlr", dependencies = TRUE)
```

Install and load libraries : tidyverse and mclust

Hide

```
# install.packages("tidyverse")
# install.packages("mclust")
```

Load the diabetes data :

Hide

```
data(diabetes, package = "mclust")
```

Use the installed libs :

Hide

```
library(mlr)
library(tidyverse)
```

Format and display data :

Hide

```
diabetesTib <- as_tibble(diabetes)
diabetesTib
```

class	glucose	insulin	sspg
<fctr>	<dbl>	<dbl>	<dbl>
Normal	80	356	124
Normal	97	289	117
Normal	105	319	143
Normal	90	356	199
Normal	90	323	240
Normal	86	381	157
Normal	100	350	221
Normal	85	301	186
Normal	97	379	142

class <fctr>	glucose <dbl>	insulin <dbl>	sspg <dbl>
Normal	97	296	131

1-10 of 145 rows Previous **1** 2 3 4 5 6 ... 15 Next

Get some stats of the data :

Hide

```
summary(diabetesTib)
```

```
      class       glucose        insulin            sspg      
 Chemical:36   Min.   : 70   Min.   :  45.0   Min.   : 10.0  
 Normal  :76   1st Qu.: 90   1st Qu.: 352.0   1st Qu.:118.0  
 Overt   :33   Median : 97   Median : 403.0   Median :156.0  
               Mean   :122   Mean   : 540.8   Mean   :186.1  
               3rd Qu.:112   3rd Qu.: 558.0   3rd Qu.:221.0  
               Max.   :353   Max.   :1568.0   Max.   :748.0  
```

Keep the classes for visualisation purposes, remove them for the clustering part (unsupervised)

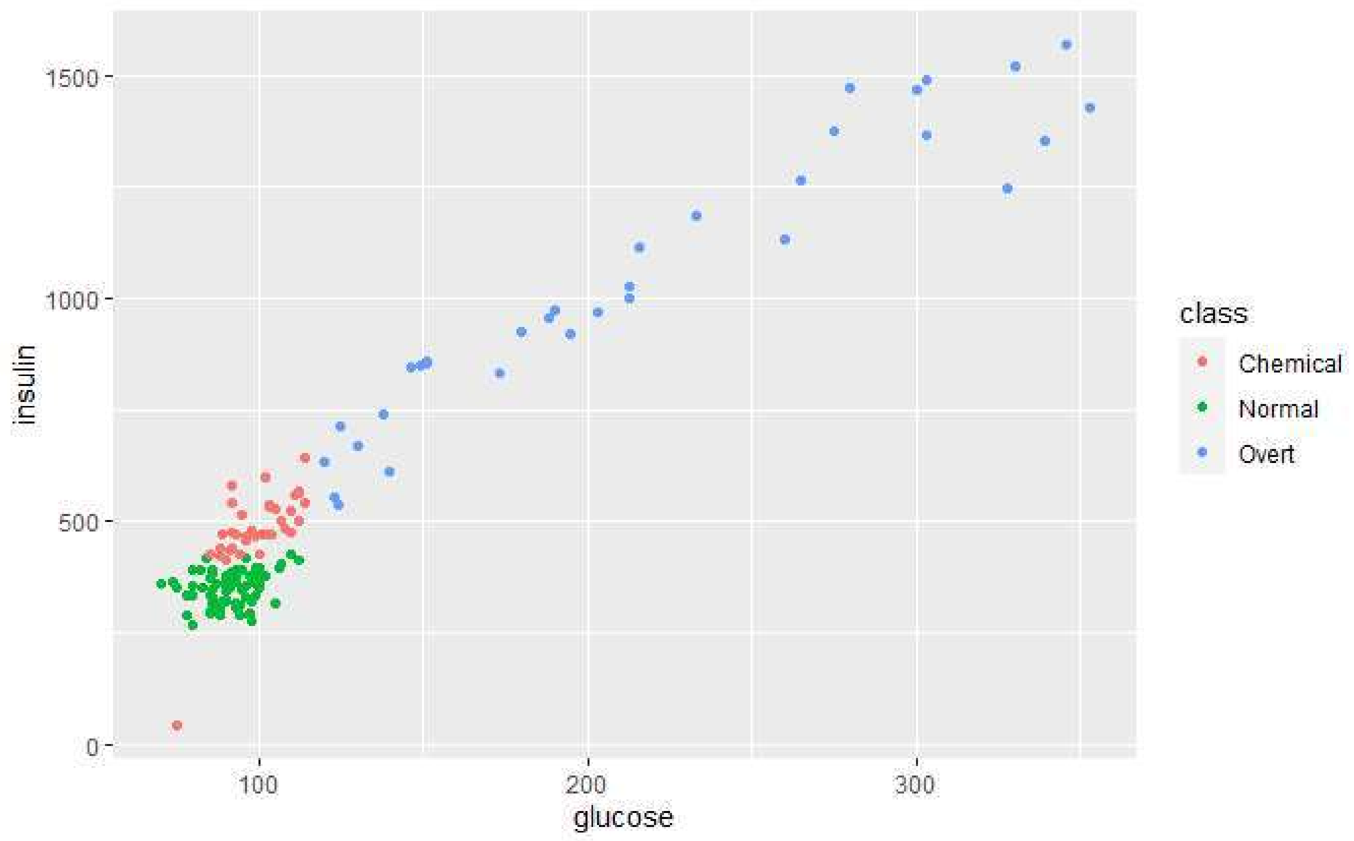
Hide

```
diabetesTib_without_class <- select(diabetesTib, -class)
```

Plot the data around some of its classes :

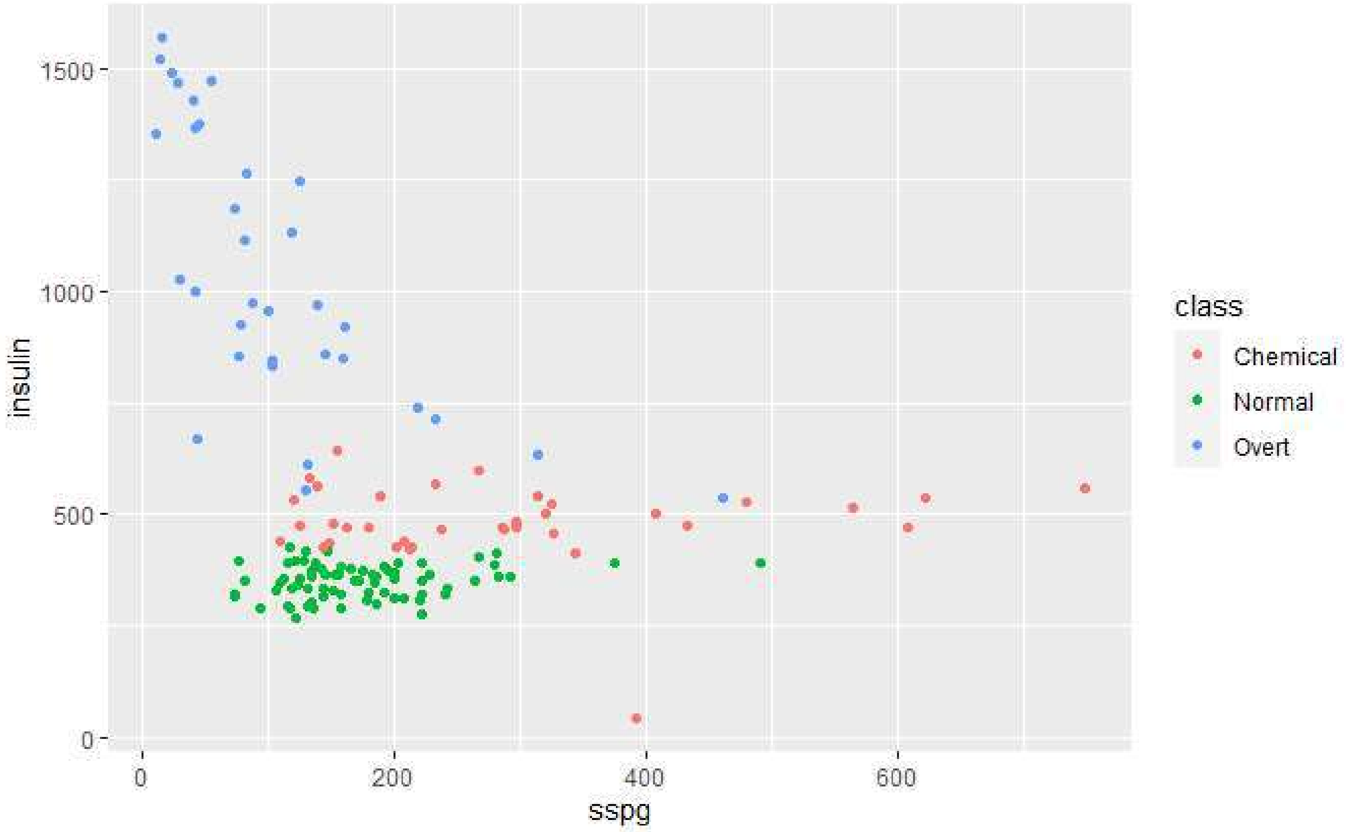
Hide

```
ggplot(diabetesTib, aes(glucose, insulin, col = class)) +
  geom_point()
```



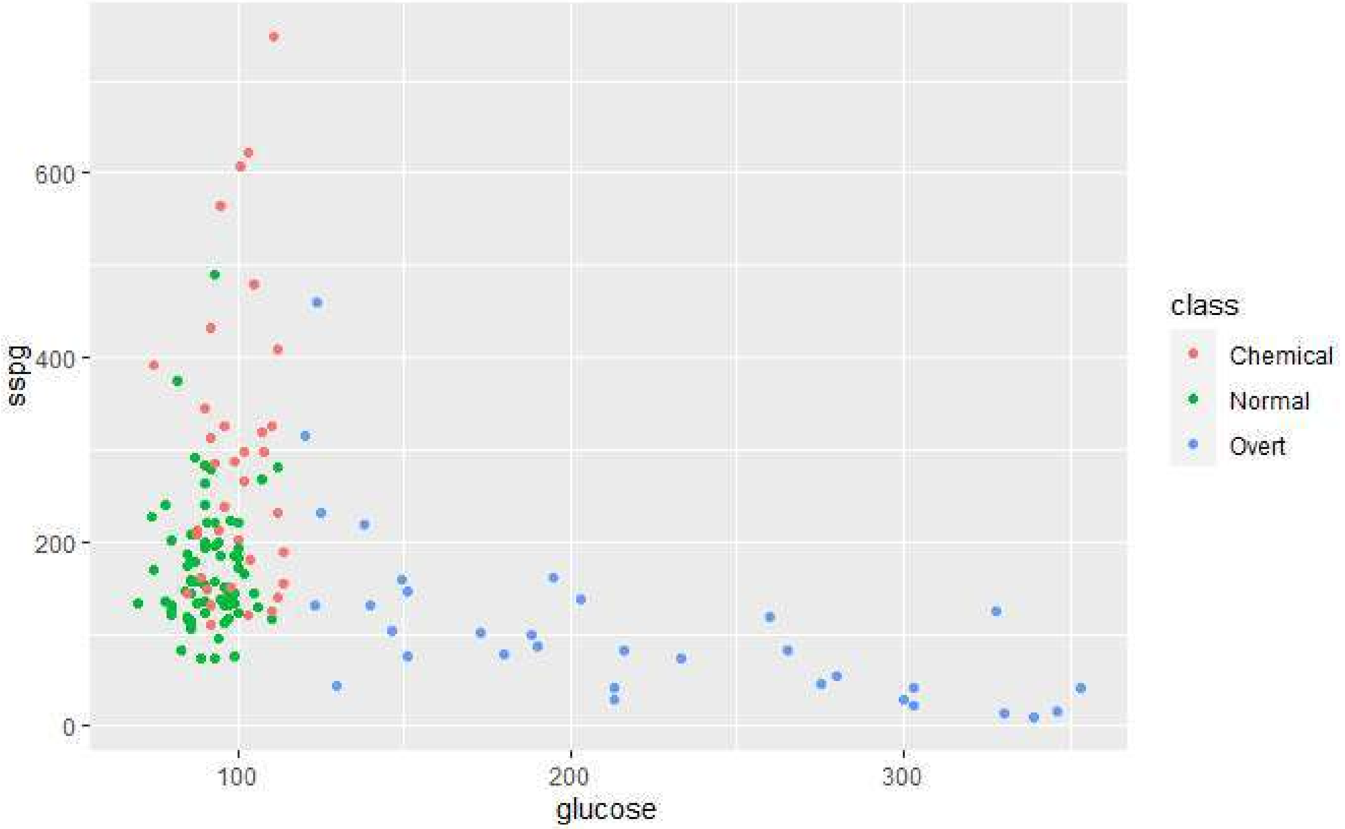
Hide

```
ggplot(diabetesTib, aes(sspg, insulin, col = class)) +
  geom_point()
```



Hide

```
ggplot(diabetesTib, aes(glucose, sspg, col = class)) +
  geom_point()
```



Define a Task :

Hide

```
diabetesTask <- makeClusterTask(data = diabetesTib_without_class)
```

```
Provided data is not a pure data.frame but from class tbl_df, hence it will be converted.
```

Hide

```
diabetesTask
```

```
Unsupervised task: diabetesTib_without_class
Type: cluster
Observations: 145
Features:
   numerics     factors     ordered functionals 
          3           0           0           0 
Missings: FALSE
Has weights: FALSE
Has blocking: FALSE
Has coordinates: FALSE
```

Define a learner : K Means Clustering

Hide

```
knn <- makeLearner("cluster.kmeans", centers = 3)
```

Train the model :

Hide

```
knnModel <- train(knn, diabetesTask)
```

Get the predictions :

Hide

```
pred <- predict(knnModel, task = diabetesTask)
```

Get some performance score results : we select these from the list below

“db” is Davies-Bouldin cluster separation measure “G1” is Calinski-Harabasz pseudo F statistic “G2” is Baker and Hubert adaptation of Goodman-Kruskal’s gamma statistic “Silhouette” is Rousseeuw’s silhouette internal cluster quality index

Hide

```
listMeasures("cluster")
```

```
[1] "featperc"    "db"          "timeboth"    "timetrain"   "timepredict" "silhouette"  "G1"         
[8] "G2"
```

Compute the scores from the predictions

Hide

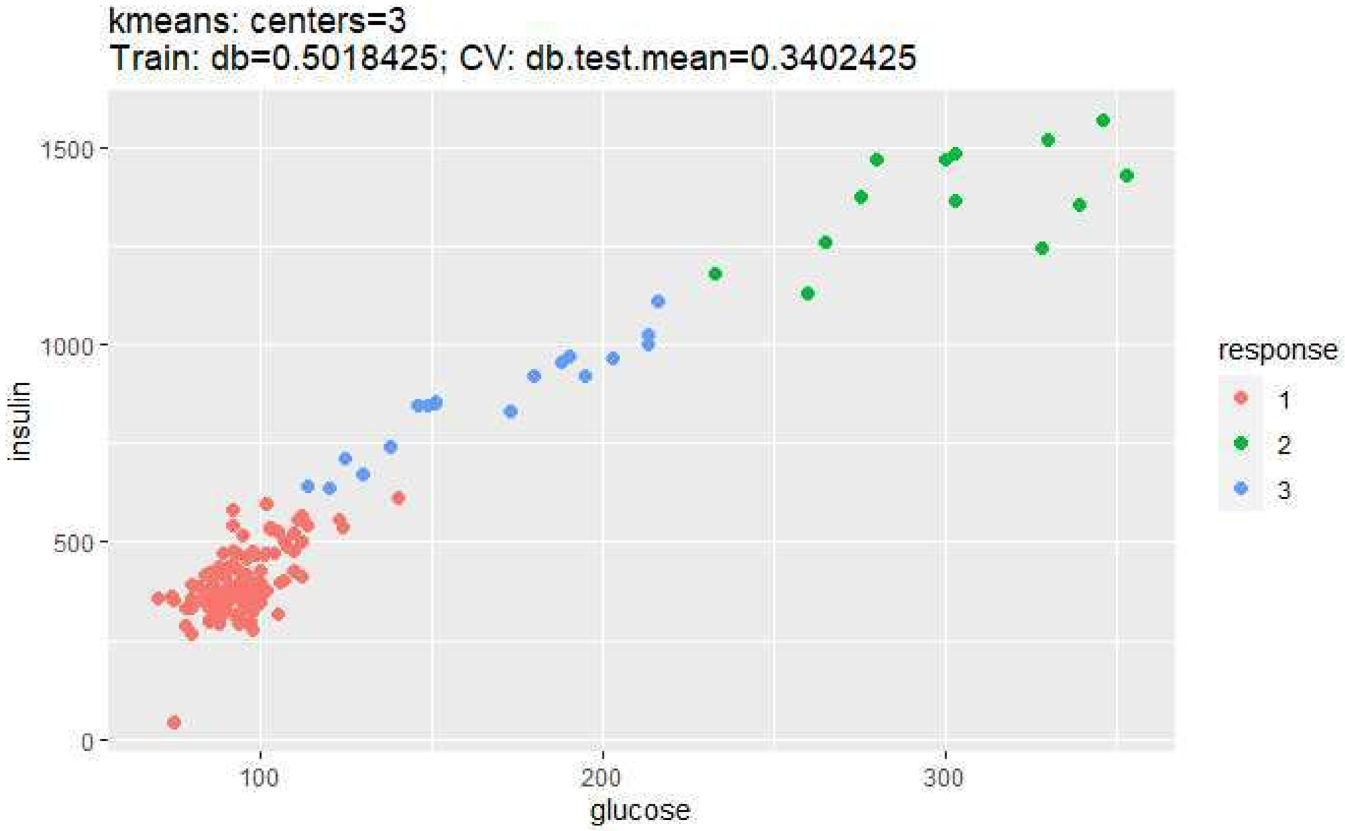
```
performance(pred, measures = list(db, G1, G2, silhouette), task = diabetesTask)
```

```
       db          G1          G2  silhouette 
0.9617377 293.5111025   0.9230416   0.7158229 
```

We can plot the clustering model :

Hide

```
plotLearnerPrediction(learner = knn, task = diabetesTask)
```


kmeans: centers=3
Train: db=0.5018425; CV: db.test.mean=0.3402425
insulin
glucose
response
● 1
● 2
● 3