M. Ramani Priya
2451-18-733-001

ASSIGNMENF - V

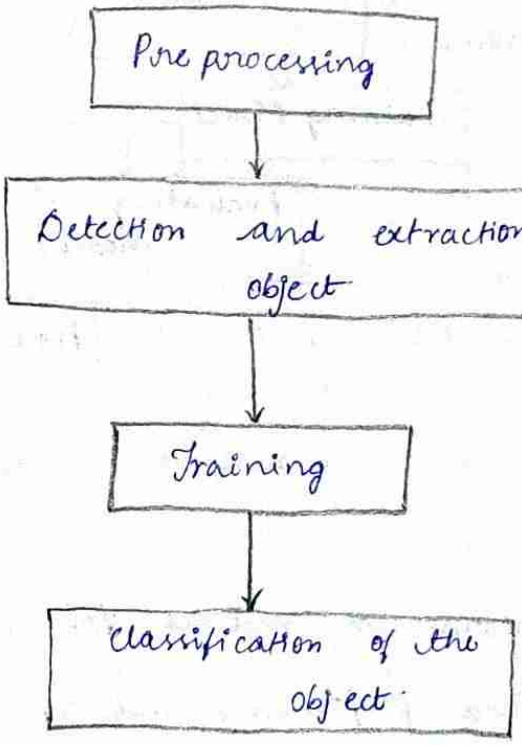
1. What is classification ? Draw and explain classification process.

Classification : It is a process of predicting a categorical label of a data object based on its features & properties.

→ In classification, we locate identifiers or boundary conditions that correspond to a particular label or category.

→ We then try to place various unknown objects into those categories, by using the identifiers.

Classification process.


Pre processing
Detection and extraction of an object
Training
Classification of the object

1. Pre processing : atmospheric correction, noise removal, image transformation, main component analysis etc.

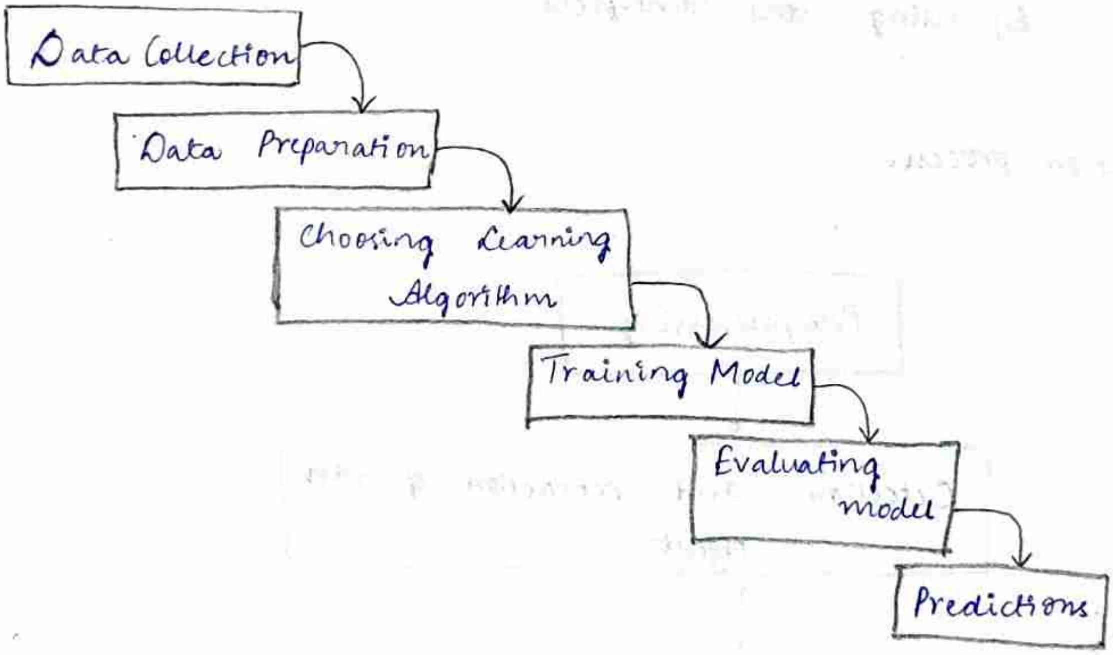
2. Detection and extraction of an object, including detection of position & other characteristics of a moving object image obtained by a camera, while in extraction, estimating the

trajectory of the detected object in the image plane.

3. Training - selection of the particular attribute which best describes the pattern.

4. Classification of the object - This step categorizes detected objects into predefined classes by using a suitable method that compares the image patterns with the target pattern.

Learning process.



2. List out various performance metrics for classification.

The most commonly used performance metrics for classification problem are as follows,

1. Accuracy.
2. Confusion matrix.
3. Precision, Recall & F1 score.
4. ROC AUC
5. Log loss.

1. Accuracy:

Accuracy is a simple ratio between the number of correctly classified points to the total number of points.

2. Confusion matrix:

It is a summary of predicted results in specific table layout that allows visualization of the performance measure of the machine learning model for a binary classification problem or multiclass classification problem.

Actual values

		Positive (1)	Negative (0)
Predicted values	Positive (1)	TP	FP
	Negative (0)	FN	TN

TP: True Positive

FP: False Positive

FN: False Negative

TN: True negative.

3. Precision, Recall & F-1 Score.

(i) Precision: it is a fraction of the correctly classified instances from the total classified instances (ii) Recall is the fraction of the correctly classified instances from the total classified instances.

(iii) F1-Score:

F1 score is the harmonic mean of precision & recall.

$$F1\ score = \frac{2 * Precision \times Recall}{Precision + Recall}$$

4. Log Loss.

Logarithmic loss (or log loss) measures the performance of a classification model where the prediction is a probability value between 0 & 1.

Log loss increases as the predicted probability diverge from the actual label.

$$log\text{-}loss = -\frac{1}{N}\sum_{i=1}^{N} y_i \log p_i + (1-y_i)\log(1-p_i).$$

5. ROC AUC

↓

Receiver Operating Characteristic curve is created by plating the True Positive (TP) against false Positive (FP) at various threshold setting.

ROC curve is generated by by plotting the cumulative distribution function of the true positive (TP) in the y axis versus the cumulative destination function of the false positive on x axis.

3. Define clustering. List out the applications of clustering technique?

Clustering is basically a type of unsupervised learning model An unsupervised learning method in which we draw references from datasets consisting of input data without labelled responses. Generally it is used as a process to find meaningful structure, explainatory underlying processes, generative features & grouping inherent in a set of examples. Clustering is the task of dividing the population or data points into a number of groups such that data points in the same group ~~as~~ are more Similar to other data points in the same group & dissimilar to the data points in other groups. It is basically a collection of objects on the basis of similarity & dissimilarity between them.

Applications of clustering.

clustering technique can be used in various areas or fields of real life examples such as data mining, web cluster engines, academics, bioinformatics, image processing & transformation, many more & emerged as an effective solution to above mentioned areas.

Some common application platform where clustering as a tool can be implemented are.,

1. Recomendation engines.

The recommendation engine is a widely used method for providing automated personalised suggestions about products, services & information where collaborative filtering is one of the famous recommendation system & techniques.

2. Market & customer segmentation.

A process of splitting market into smaller & more defined categories is known as market segmentation. This segments customers/ audiences into groups of similar characteristics (needs, location interests or demographics) where target & personalization, under it is an immense buissiness

3. Social Network Analysis (SNA)

It is the process of examining qualitative & quantitative social structures by utilizing Graph theory (a major branch of discrete mathematics) & networks. Here the mapping of social networks structure is arranged in terms of nodes (individual personality, people or other entity inside the network) & the edges or links (relationships, interaction or communication) that connect them.

4. Search Result Clustering:

You must have encountered similar results obtained while Searching something particular at Google, these results are a mixture of the similar matches of your original query.

5. Biological Data Analysis, Medical Imaging - Analysis & Identification of cancer cells.

One of the main means to connect analytical tools with biological content is biological content analysis for a cheamp & extended understanding of the relationships identified as to be linked with experimental observations.

4. Explain about evaluating clustering models.

Clustering evaluation strategies:

Three important strategies by which clustering can be evaluated are:

a) Clustering tendency

b) Number of clusters, k

c) Clustering quality

a) Clustering tendency:

Before evaluating the clustering performance, making sure that data set we are working has clustering tendency & doesnot contain clustering uniformly distributed points is very important. If ~~it~~ data doesnot contain clustering tendency, then cluster identified by any state of the art clustering algorithms may be irrelevant. Non uniform distribution of points in data set becomes important in clustering.

To solve this, Hopkins test, a statistical test for spatial ~~render~~ randomness of a variable, can be used to measure the probability of data points generated by uniform data distribution.

1. Null Hypothesis (Ho): Data points are generated by uniform distribution (implying no meaningful clusters).

2. Alternate Hypothesis (Ha): Data points are generated by random data point (presence of clusters).

If H>0.5, null hypothesis can be rejected & it is very much likely that data contains clusters. If H is more close to 0, the data set doesn't have clustering tendency.

b) ~~Opti~~ Number of optimal clusters, k

Some of the clustering algorithms need like kmeans require number of clusters k, as clustering parameters. Getting the optimal number of clusters is very significant in the analysis. If k is too high, each point will broadly start representing a cluster & if k is too low, then data points are incorrectly clustered. Find the optimal number of clusters leads to granularity in clustering.

c) Clustering quality.

Once clustering is done, how well the clustering has performed can be quantified by a number of metrics. Ideal clustering is characterized by minimal intra cluster distance & ~~minimal~~ maximal inter cluster distance. There are majorly 2 types of measures to assess the clustering performance.

i) Extrinsic measures which require ground truth labels. Examples are adjusted Rank index, Mallows scores, Mutual information based scores, homogeneity, completeness & V-measure.

ii) Intrinsic measures that doesnot require ground truth labels, some of the clustering measures are Silhouette coefficient, Calinski-Harobasz Index, Davies Bouldin Index etc.

5. Explain K-Nearest Neighbours Algorithm & its implementation in R programming language.

K-Nearest Neighbour (KNN): It is one of simplest Machine-Learning Algorithms based on Supervised learning technique.

→ KNN algorithm assumes the similarity between the new case/data & availiable cases & put the new case into the category that is most similar to the availiable categaries.

→ KNN can be used for classification & regression.

→ It is a non parametric algorithm, which means it doesnot make any assumption on underlying data

→ It is also called lazy learner algorithm because doesnot learn from the training set immediately instead it stores the dataset & at time of classification, it performs ~~on the~~ an action on the dataset

→ KNN algorithm at training phase just stores the dataset & when it gets new data, then it classifies that data into a category that is much similar to the new data.

The KNN can be explained on basis of following algorithm:

1. Select the number k of neighbours.
2. Calculate the Euclidean distance of k number of neighbours
3. Take the k nearest neighbours as per the calculated Euclidean distance.
4. Among these k neighbours, count the number of the data points in each category.
5. Assign the new data points to that ~~co~~ category for which the number of the neighbour is maximum.
6. Our model is ready.

KNN implementation in R programming language.

The Dataset

Iris dataset consists of 50 samples from each of 3 species of Iris (Iris setosa, Iris virginica, Iris versicolor) & a multivariate dataset.

```
#loading data
  data(iris)

#structure
  str(iris)
```

Performing Knn on Dataset

Using KNN algorithm on the dataset which includes 11 persons & 6 variables or attributes.

```
 install.packages("C1071")
 install.packages("caTools")
 install.packages("class")

# Loading package
  library(C1071)
  library(caTools)
  library(class)

# Loading data
  data(iris)
  head(iris)

# splitting data into train & test data
 split ← sample.split(iris, splitRatio = 0.7)
 train_cl ← subset(iris, split= "TRUE")
 test_cl ← subset(iris, split = "FALSE")

# feature scaling
 train_scale ← scale(train_cl[, 1:4])
 test_scale ← scale(test_cl[, 1:4])

# fitting KNN model to training dataset
 classifier_knn ← Knn(train = train_scale, test= test_scale,
                      cl = train_cl$species, k = 1)
```

```
# Confusion matrix
  Cm ← table (test_cl $ Species, classifier_knn)

# Model Evaluation - Choosing K
# Calculate out of sample error

   misClassError ← mean (classifier_knn != test_cl $Species)
   print (paste ('Accuracy = ', 1- misClassError))

# K = 7

  classifier_knn ← knn (train = train_scale, test = test_scale,
                        cl = train_cl $Species, k = 7)

  misClassError ← mean (classifier_knn != test_cl $Species)
   print (paste ('Accuracy = ', 1- misClassError))
```

Output :

Model classifier_knn (k = 1)

The KNN model is fitted with a train, test & k value. Also, the classifier species feature is fitted in model.

Confusion Matrix

Cm

	Setosa	versicolor	virginica
Setosa	20	0	0
versicolor	0	17	3
Virginica	0	3	17

Model evaluation

k = 1

Accuracy = 0.9

6. Explain K-Means Algorithm & its implementation in R programming language.

K Means Algorithm:

K Means clustering is an unsupervised learning algorithm, which groups the unlabelled dataset into different clusters. Here k defines the number of predefined clusters that need to be ~~treated~~ created in the process, as if k=2, there will be 2 clusters.

→ It is a centroid based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point & their corresponding clusters

→ This algorithm takes the unlabeled dataset as input, divides the dataset into k number of clusters & repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithms

→ The working of k means algorithms is explained in the below steps:

1. Select the number k to decide the number of clusters.

2. Select random k points or centroids (It can be other from the input dataset).

3. Assign each data point to their closest centroid, which will form the predefined k clusters.

4. Calculate the variance & place a new centroid of each cluster.

5. Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.

6. If any reassignment occurs then go to step 4 else go to finish.

7. The model is ready.

K. Means Implementation in R 2451-18-733-001

Using k-means clustering on the dataset which includes 11 persons & 6 variables or attributes.

```
# Installing Packages
   install.packages ("ClusterR")
   install packages ("cluster")

# Loading Package
   library (ClusterR)
   library (cluster)

# Removing initial label of Species from original dataset
   iris_1 <- iris [, -5]

# Fitting k-means clustering model to training dataset.
   set.seed (240) # setting seed
   kmeans.re <- kmeans( iris_1, centers = 3, nstart = 20)

# Cluster identification for each observation
   kmeans.re $ cluster

# Confusion matrix
   cm <- table ( iris $ Species, kmeans.re $ cluster)

# Model evaluation & visualization
  plot (iris_1 [c("Sepal.Length", "Sepal.Width")])
  plot (iris_1 [c("Sepal.Length", "Sepal.Width")],
                 col = kmeans.re $ cluster)

  plot (iris_1 [c ("Sepal.Length", "Sepal.Width")],
                 col = kmeans.re $ cluster,
                 main = "k-means with 3 clusters")
```

```
# visualizing clusters
y_kmeans <- kmeans.re $ cluster
clusplot (iris-1 [ , c("sepal. Length", "sepal. width")],
          y_kmeans,
          lines =0,
          shade =TRUE,
          color = TRUE,
          labels =2,
          plotchar = FALSE,
          span =TRUE,
          main = paste ("Cluster iris"),
          xlab = "Sepal. Length",
          ylab = "Sepal. Width")
```

Output:

Model kmeans.re :

The 3 clusters are made which one of 50, 62 & 38 sizes respectively, within the cluster, the sum of squares is 88.4%

The model acheived an accuracy of 100% with pvalue of less than 1 this indicates the model is good.

Confusion matrix

```
cm
              1    2    3
  setosa     50    0    0
  versicolor  0   48    2
  virginica   0   14   36
```