Intro to Machine Learning Workshop

Day 2

Dr.Reem Alotaibi

February 27th 2022

Outline

Outline

Revision

Summary

- Machine learning process (get data, pre-processing, train the model, test the model, refine the model).
- Supervised learning vs unsupervised learning.
- Training vs test data.
- Train-test vs k-cross validation.

Outline

1 Day 1 Revision

2 Supervised Learning

3 Unsupervised Learning

4 Summary

Supervised Learning
Classification

Input space
Features, attributes, variables, covariate

Training set

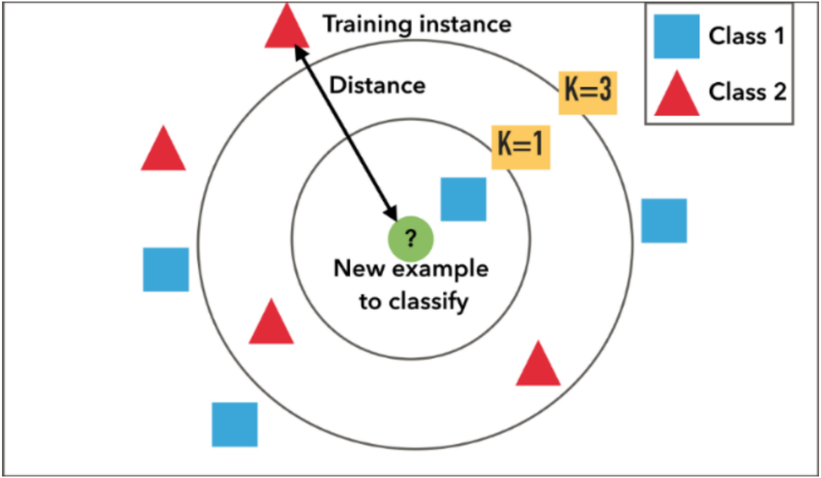
	Age	Gender	Glucose	Diabetics
1	65	male	120	Yes
2	35	female	180	Yes
3	55	female	150	Yes
4	37	male	90	No
5	25	female	105	No
6	56	male	125	?

Output space
Class, label, target, dependent variable

Classification: Categorical or discrete target

Classification

K-Nearest Neighbour (KNN) Algorithm



Classification

K-Nearest Neighbour (KNN) Algorithm

Detecting Prostate Cancer

- The dataset consists of 100 observations and 10 variables.
- 8 numeric variables and one categorical variable(ID).
- The task is to classify the tumour into benign (B) and malignant (M).

	id	diagnosis_result	radius	texture	perimeter	area	smoothness	compactness	symmetry	fractal_dimension
1	1	M	23	12	151	954	0.143	0.278	0.242	0.079
2	2	B	9	13	133	1326	0.143	0.079	0.181	0.057
3	3	M	21	27	130	1203	0.125	0.160	0.207	0.060
4	4	M	14	16	78	386	0.070	0.284	0.260	0.097
5	5	M	9	19	135	1297	0.141	0.133	0.181	0.059

Classification

K-Nearest Neighbour (KNN) Algorithm

Learn the KNN model with k=5

```
library(class)
model = knn(train_split, test_split , train_target,  k=5)
```

```
> table(test_target,model)
           model
test_target  B  M
          B  8  1
          M  4 11
```

Classification

K-Nearest Neighbour (KNN) Algorithm

```
             | model
test_target  |         B |         M | Row Total |
-------------|-----------|-----------|-----------|
           B |         8 |         1 |         9 |
             |     2.722 |     2.722 |           |
             |     0.889 |     0.111 |     0.375 |
             |     0.667 |     0.083 |           |
             |     0.333 |     0.042 |           |
-------------|-----------|-----------|-----------|
           M |         4 |        11 |        15 |
             |     1.633 |     1.633 |           |
             |     0.267 |     0.733 |     0.625 |
             |     0.333 |     0.917 |           |
             |     0.167 |     0.458 |           |
-------------|-----------|-----------|-----------|
Column Total |        12 |        12 |        24 |
             |     0.500 |     0.500 |           |
-------------|-----------|-----------|-----------|
```

Classification

K-Nearest Neighbour (KNN) Algorithm

Tips

- KNN performs much better when variables are normalized because higher range variables can bias it.
- Missing data will mean that the distance between samples can not be calculated. These samples could be excluded or the missing values could be imputed.
- The k parameter is often an odd number to avoid ties in the voting scores.
- How to select appropriate k value? sqrt(n)?
- KNN is suited for lower dimensional data.

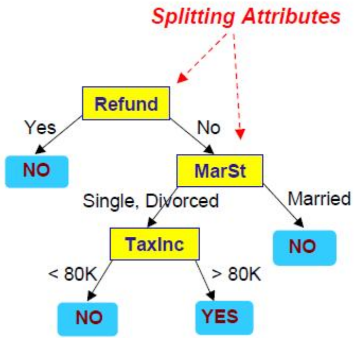
Classification

Decision Trees Algorithm

categorical categorical continuous class

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Training Data



Model: Decision Tree

Classification

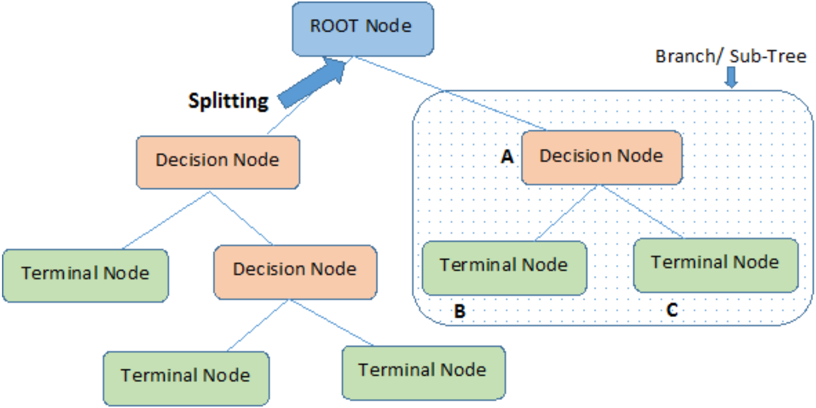
Decision Trees Algorithm

Important Terminology

- **Root Node:** it represents entire population or sample and this further gets divided into two or more homogeneous sets.
- **Splitting:** the process of dividing a node into two or more sub-nodes.
- **Decision/Internal Node:** when a sub-node splits into further sub-nodes, then it is called decision node.
- **Leaf/Terminal Node:** nodes do not split is called Leaf or Terminal node.

Classification

Decision Trees Algorithm-Important Terminology



Note:- A is parent node of B and C.

Classification

Decision Trees Algorithm

Fit the tree for the Prostate Cancer dataset

- Use rpart package
- "class" for classification, "anova" for regression.
- Fit the model using rpart.
- Print the summary.

```
library(rpart)
fit <- rpart(diagnosis_result ~ . ,train_split,  method="class")
```

Classification

Decision Trees Algorithm

Tips

- Overfitting
  - Pruning.
  - Setting constraints on tree size (minimum number of observations at leaf, maximum depth) .
- Splitting criteria (gini index, information gain, chi square).
- Random forest algorithm.

Supervised Learning

Regression

Input space

Features, attributes, variables, covariate

Training set

	Month	Temp C	Ice cream sales SAR/Day
1	Jan	15	501
2	Feb	20	550
3	March	25	600
4	April	27	900
5	May	30	1050
	June	35	?

Output space

Target

dependent variable

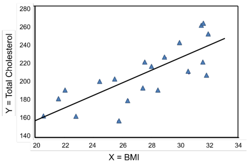
Regression: Numerical or continues target

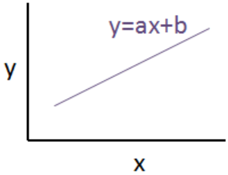
Regression

Linear Regression

Problem

- Help you to predict the dependent variable y using independent variable x.
- $y = ax + b$
  - y is the dependent variable.
  - x is the independent variable.
  - a and b are constant. a is the slope and b is the intercept.





Regression

Linear Regression

Example

- Predict the Ozone using the solar radiation.
- Create a relationship model using lm() function.
- Print the summary.

```
model1<-lm(Ozone~Solar.R, data=airquality)
summary(model1)
```

Regression

Linear Regression

Example

- Find the coefficients from the model.
- Ozone= 0.12717 Solar.R + 18.59873

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 18.59873    6.74790   2.756 0.006856 **
Solar.R      0.12717    0.03278   3.880 0.000179 ***
```

Regression

Linear Regression

Example

- To predict the Ozone, use the predict() function in R.

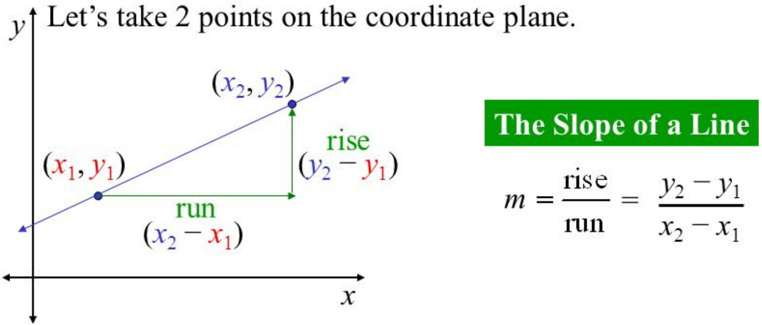
```

Solar.R=185.93
new_data=data.frame(Solar.R)
pred_oz=predict(model1,new_data)
pred_oz
```

Regression

Linear Regression

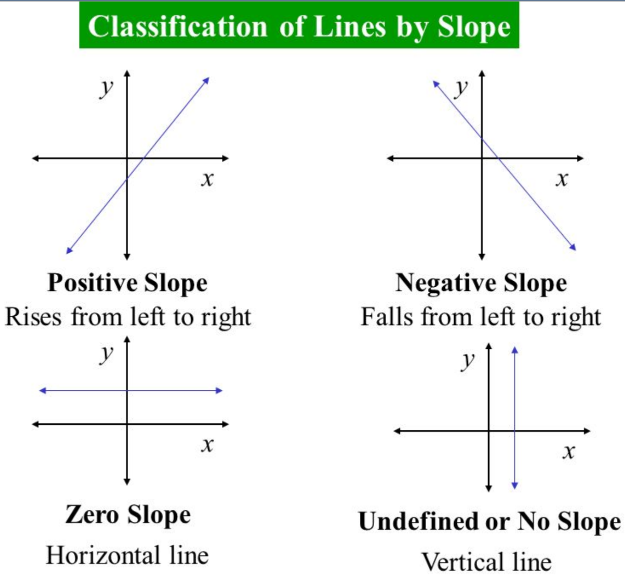
Slope

- slope $= \frac{y_2 - y_1}{x_2 - x_1}$



Regression

Linear Regression



Regression

Linear Regression

Tips

- Use regression lines when there is a significant correlation to predict values.
- Stay within the range of the data. Do not extrapolate!! For example, if the data is from 10 to 60, do not predict a value for 400.

Outline

Unsupervised Learning

Clustering

Input space
attributes, variables, covariate

Training set

	Student ID	Level	GPA
1	14002	9	4.55
2	14050	9	3.51
3	16007	7	4.92
4	16431	7	3.97
5	16001	7	4.70

Clustering: Hidden target

Unsupervised Learning

Clustering

Input space

attributes, variables, covariate

Training set

	Student ID	Level	GPA	
1	14002	9	4.55	1
2	14050	9	3.51	
3	16007	7	4.92	2
4	16431	7	3.97	
5	16001	7	4.70	

Unsupervised Learning
Clustering

Input space
attributes, variables, covariate

Training set

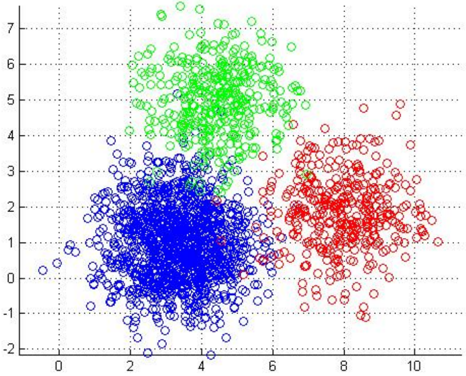
	Student ID	Level	GPA	
1	14002	9	4.55	1
2	14050	9	3.51	
3	16007	7	4.92	
4	16431	7	3.97	2
5	16001	7	4.70	

Unsupervised Learning

Clustering

K-means Algorithm

- K Means Clustering is an unsupervised learning algorithm that tries to cluster data based on their similarity.



Unsupervised Learning

Clustering

K-means Algorithm

1. Select K centroids randomly.
2. Assign each data point to its closest centroid.
3. Recalculate the centroids as the average of all data points in a cluster.
4. Continue steps 2 and 3 until the observations are not reassigned or the maximum number of iterations (R uses 10 as a default) is reached.

Clustering

K-means Algorithm

Example

- We would like to cluster the attitude dataset with the responses from 30 departments.
- Use kmeans() function.

```
data(attitude)
set.seed(7)
km1 = kmeans(attitude, 2)
km1
```

Clustering

K-means Algorithm

Example

- View the clusters.

```
K-means clustering with 2 clusters of sizes 14, 16

Cluster means:
   rating complaints privileges learning   raises critical  advance
1 74.0000   77.78571   60.14286 65.28571 71.71429 76.85714 47.71429
2 56.4375   56.81250   47.00000 48.56250 58.43750 72.93750 38.75000

Clustering vector:
 [1] 2 2 1 2 1 2 2 1 1 2 2 2 2 1 1 1 1 1 1 2 2 2 2 2 2 1 1 2 1 1
```

Clustering

K-means Algorithm

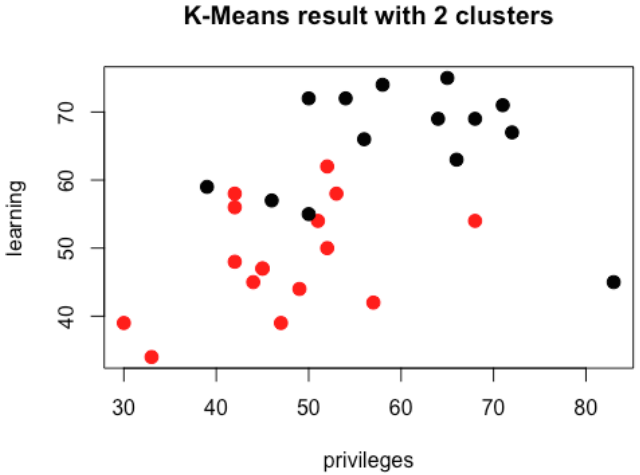
Example

- Plot subset of the data.

```
plot(attitude[,c(3,4)], col =(km1$cluster) ,
     main="K-Means result with 2 clusters", pch=20, cex=2)
```

Clustering

K-means Algorithm



Clustering

k-means Algorithm

Tips

- K can be assigned by experts.
- Clusters can be evaluated using:
  - Elbow method.
  - Silhouette analysis.

Clustering

K-means Algorithm

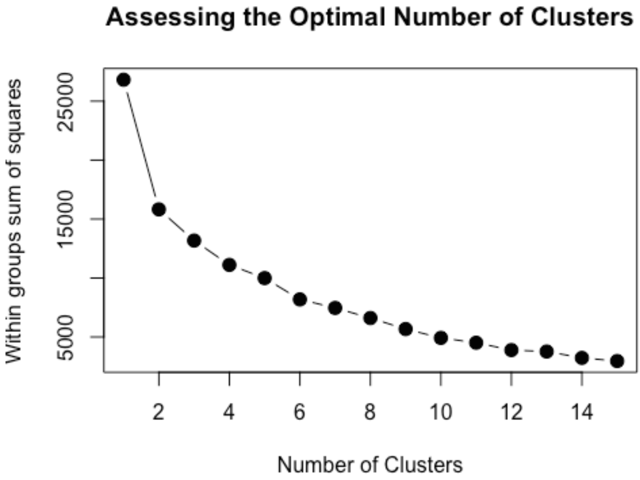
Elbow method

- Check for the optimal number of clusters given the data.

```
# Check for the optimal number of clusters given the data
for (i in 1:15)
  wss[i] <- sum(kmeans(attitude,centers=i)$withinss)

plot(1:15, wss, type="b", xlab="Number of Clusters",
     ylab="Within groups sum of squares",
     main="Assessing the Optimal Number of Clusters",pch=20, cex=2)
```

Clustering

K-means Algorithm



Outline

Summary

Wrap-up

- KNN and decision trees algorithms can be used for both classification regression tasks.
- K-means is a simple clustering algorithm but it is sensitive to outliers.
- K-medoids instead of K-means.

Day 3

- Try some real-world problems.