KNN Classification

Supervised Learning

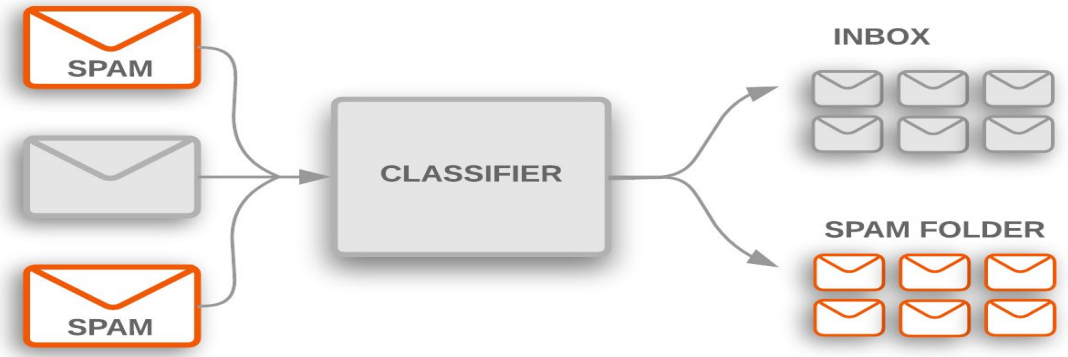
Supervised learning algorithms learns from a labelled set of training data.

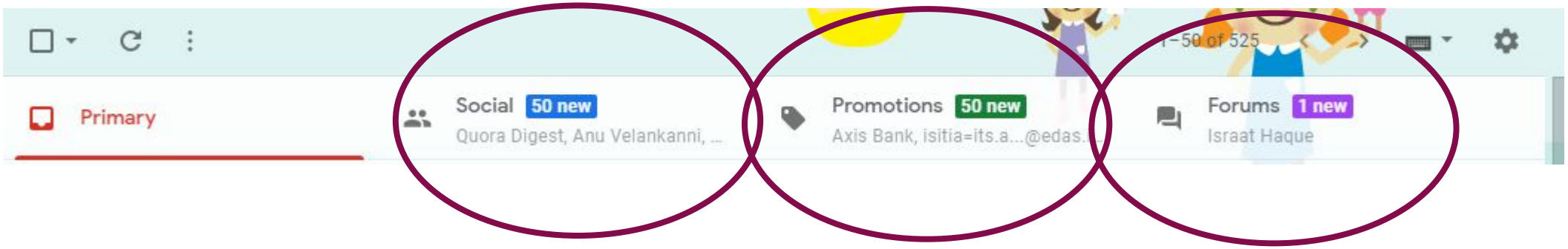
Supervised learning problems can be further grouped into **Regression** and **Classification** problems.

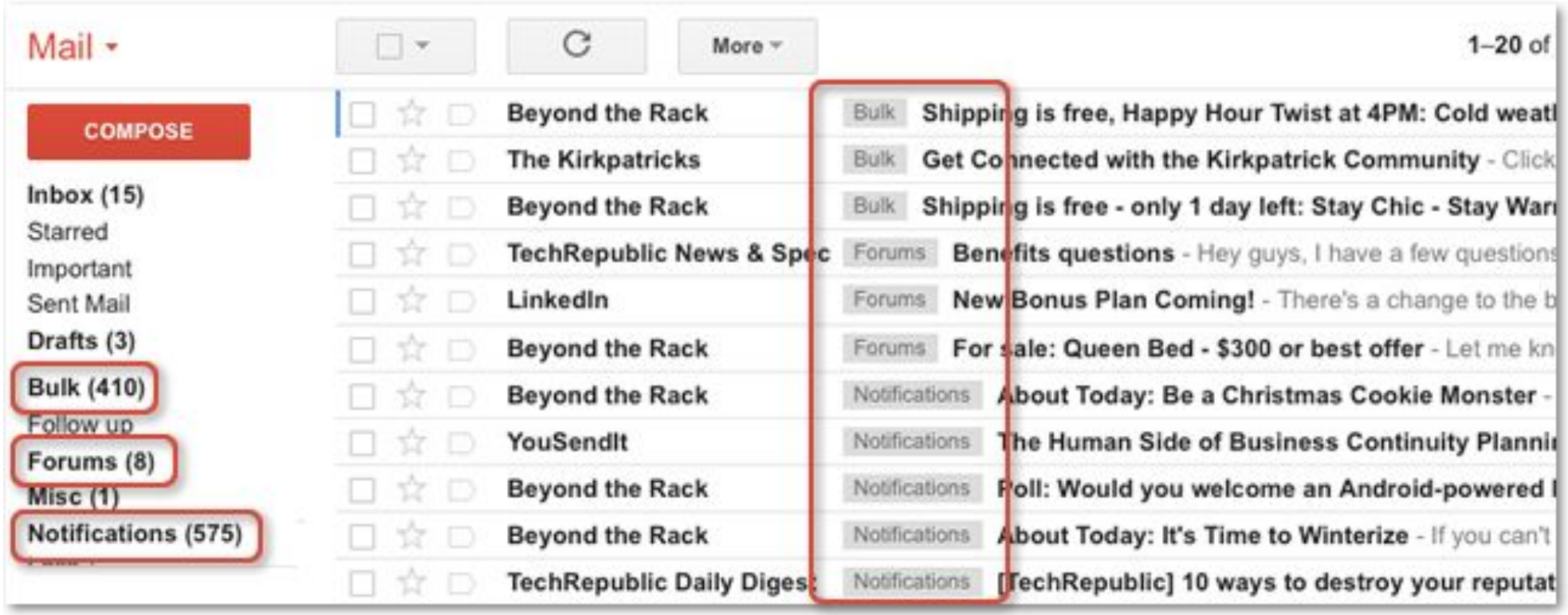
- Regression:
  - The target attribute is numerical, such as predict the weather level.
- Classification:
  - The target attribute is categorical, such as classify the image is dog or cat.

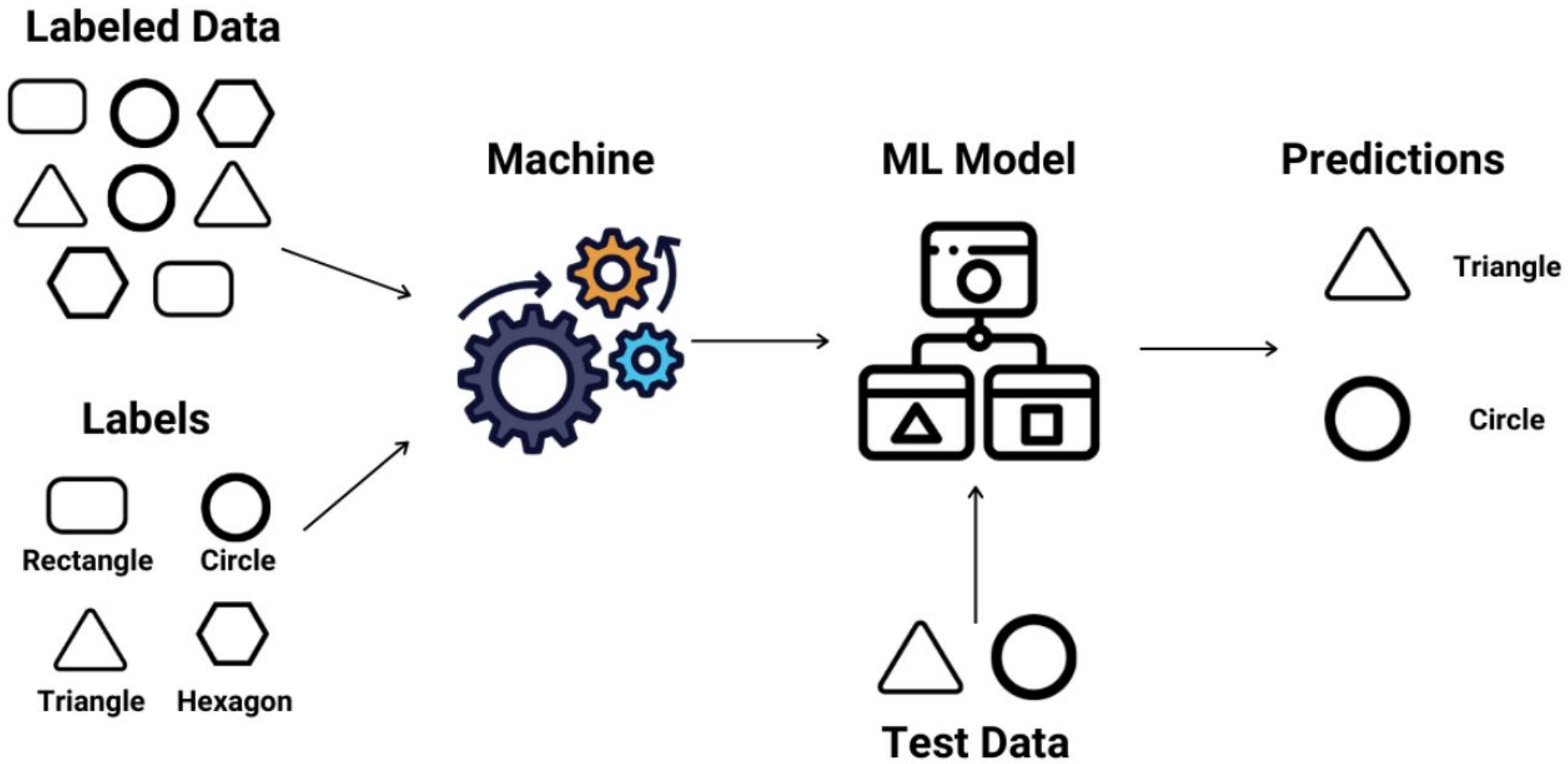
What is classification?

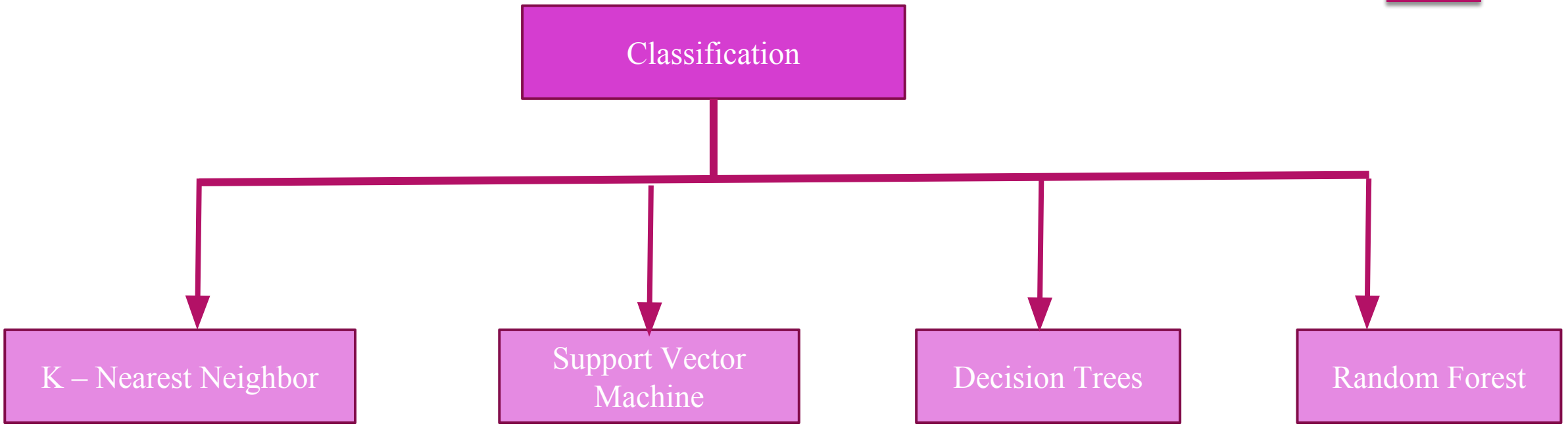
- Classification is the process of predicting the class of given data points. Classes are sometimes called as targets/ labels or categories.
- Classification predictive modeling is the task of approximating a mapping function (f) from input variables (X) to discrete output variables (y).




1–50 of 525
Primary
Social 50 new
Quora Digest, Anu Velankanni, ...
Promotions 50 new
Axis Bank, isitia=its.a...@edas.
Forums 1 new
Israat Haque


Mail
More
1–20 of
COMPOSE
Inbox (15)
Starred
Important
Sent Mail
Drafts (3)
Bulk (410)
Follow up
Forums (8)
Misc (1)
Notifications (575)
Beyond the Rack
Bulk
Shipping is free, Happy Hour Twist at 4PM: Cold weat
The Kirkpatricks
Bulk
Get Connected with the Kirkpatrick Community - Click
Beyond the Rack
Bulk
Shipping is free - only 1 day left: Stay Chic - Stay War
TechRepublic News & Spec
Forums
Benefits questions - Hey guys, I have a few questions
LinkedIn
Forums
New Bonus Plan Coming! - There's a change to the b
Beyond the Rack
Forums
For sale: Queen Bed - $300 or best offer - Let me kn
Beyond the Rack
Notifications
About Today: Be a Christmas Cookie Monster -
YouSendIt
Notifications
The Human Side of Business Continuity Planni
Beyond the Rack
Notifications
Poll: Would you welcome an Android-powered
Beyond the Rack
Notifications
About Today: It's Time to Winterize - If you can't
TechRepublic Daily Diges
Notifications
[TechRepublic] 10 ways to destroy your reputat


Labeled Data
Labels
Rectangle
Circle
Triangle
Hexagon
Machine
ML Model
Predictions
Triangle
Circle
Test Data

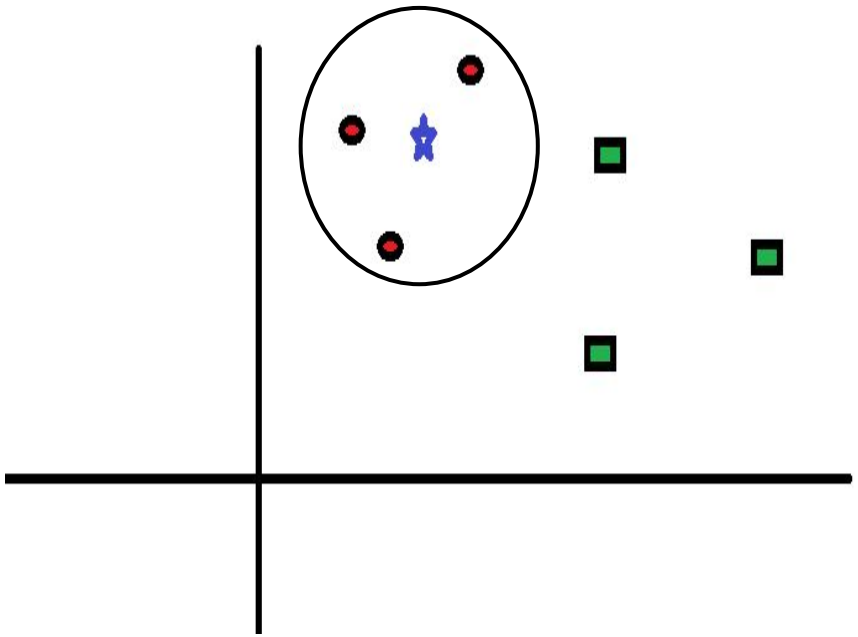

Classification
K – Nearest Neighbor
Support Vector Machine
Decision Trees
Random Forest

K-nearest neighbor

- Simple, easy-to-implement supervised machine learning algorithm
- KNN can be used for both classification and regression – mostly used in classification
- **Lazy learning algorithm** – it does not learn from the training phase, instead it memorize the training data set.
- **Non-parametric learning algorithm** - it doesn't take any assumptions about the data.

KNN classifier classifies the an unlabeled object into the target class depending on the similarity features of its neighbouring object

K-Nearest Neighbor Technique



- *k*-Nearest Neighbor is a lazy learning algorithm which stores all instances correspond to training data points in n-dimensional space.
- When an unknown discrete data is received, it analyzes the closest k number of instances saved (nearest neighbors)and returns the most common class as the prediction and for real-valued data it returns the mean of k nearest neighbors.
- KNN works based on the feature similarity to its neighbouring data points
- Similiarity is computer by calculating the distance b/w the unlabeled data point and labelled data point

KNN Algorithm

We can implement a KNN model by following the below steps:

- Load the data
- Initialise the value of k
- For getting the predicted class, iterate from 1 to total number of training data points
  - Calculate the distance between test data and each row of training data. Here we will use Euclidean distance as our distance metric since it's the most popular method. The other metrics that can be used are Chebyshev, cosine, etc.
  - Sort the calculated distances in ascending order based on distance values
  - Get top k rows from the sorted array
  - Get the most frequent class of these rows
  - Return the predicted class

KNN Classifier Algorithm

Distance Measures

$$Euclidean\ distance: \ d(x,y) = \sqrt{\sum (x_i - yi)^2}$$

$$Squared\ Euclidean\ distance: \ d(x,y) = \sum (x_i - yi)^2$$

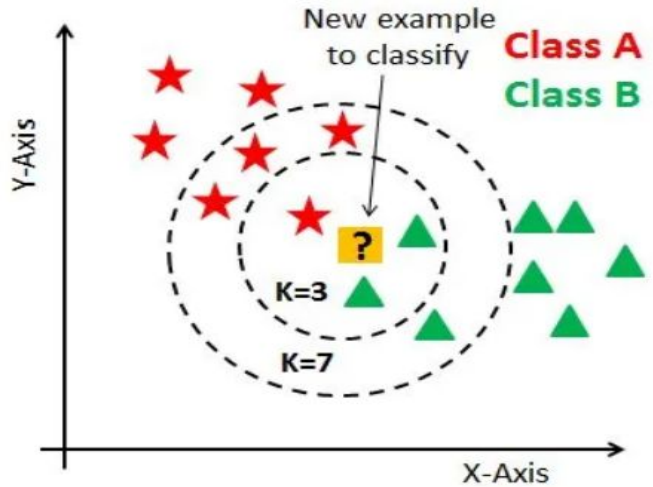
$$Manhattan\ distance: \ d(x,y) = \sum |(xi - yi)|$$

Which distance measure to use?

We use Euclidean Distance as it treats each feature as equally important.

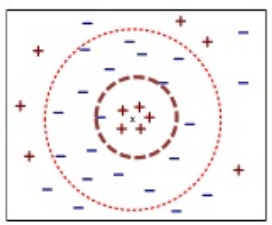
3-KNN: Example(1)

Customer	Age	Income	No. credit cards	Class	Distance from John
George	35	35K	3	No	sqrt $[(35-37)^2+(35-50)^2+(3-2)^2]=15.16$
Rachel	22	50K	2	Yes	sqrt $[(22-37)^2+(50-50)^2+(2-2)^2]=15$
Steve	63	200K	1	No	sqrt $[(63-37)^2+(200-50)^2+(1-2)^2]=152.23$
Tom	59	170K	1	No	sqrt $[(59-37)^2+(170-50)^2+(1-2)^2]=122$
Anne	25	40K	4	Yes	sqrt $[(25-37)^2+(40-50)^2+(4-2)^2]=15.74$
John	37	50K	2	YES	



How to choose K?

- If K is too small it is sensitive to noise points.
- Larger K works well. But too large K may include majority points from other classes.



- Rule of thumb is K < sqrt(n), n is number of examples.

X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	Y = Classification
7	7	Bad
7	4	Bad
3	4	Good
1	4	Good

Now the factory produces a new paper tissue that pass laboratory test with X1 = 3 and X2 = 7. Without another expensive survey, can we guess what the classification of this new tissue is?

X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	Square Distance to query instance (3, 7)	Rank minimum distance	Is it included in 3-Nearest neighbors?	**Y = Category of nearest Neighbor**
7	7	$(7-3)^2+(7-7)^2=16$	3	Yes	**Bad**
7	4	$(7-3)^2+(4-7)^2=25$	4	No	-
3	4	$(3-3)^2+(4-7)^2=9$	1	Yes	**Good**
1	4	$(1-3)^2+(4-7)^2=13$	2	Yes	**Good**

BMI	Age	Sugar
33.6	50	1
26.6	30	0
23.4	40	0
43.1	67	0
35.3	23	1
35.9	67	1
36.7	45	1
25.7	46	0
23.3	29	0
31	56	1

Test data : BMI =43.6 ,Age =40 ,Sugar =?

BMI	Age	Sugar	Distance	
33.6	50	1	$\sqrt{(43.6-33.6)^2+(40-50)^2}$	14.14
26.6	30	0	$\sqrt{(43.6-26.6)^2+(40-30)^2}$	19.72
23.4	40	0	$\sqrt{(43.6-23.4)^2+(40-40)^2}$	20.20
43.1	67	0	$\sqrt{(43.6-43.1)^2+(40-67)^2}$	27.00
35.3	23	1	$\sqrt{(43.6-35.3)^2+(40-23)^2}$	18.92
35.9	67	1	$\sqrt{(43.6-35.9)^2+(40-67)^2}$	28.08
36.7	45	1	$\sqrt{(43.6-36.7)^2+(40-45)^2}$	8.52
25.7	46	0	$\sqrt{(43.6-25.7)^2+(40-46)^2}$	18.88
23.3	29	0	$\sqrt{(43.6-23.3)^2+(40-29)^2}$	23.09
31	56	1	$\sqrt{(43.6-31)^2+(40-56)^2}$	20.37

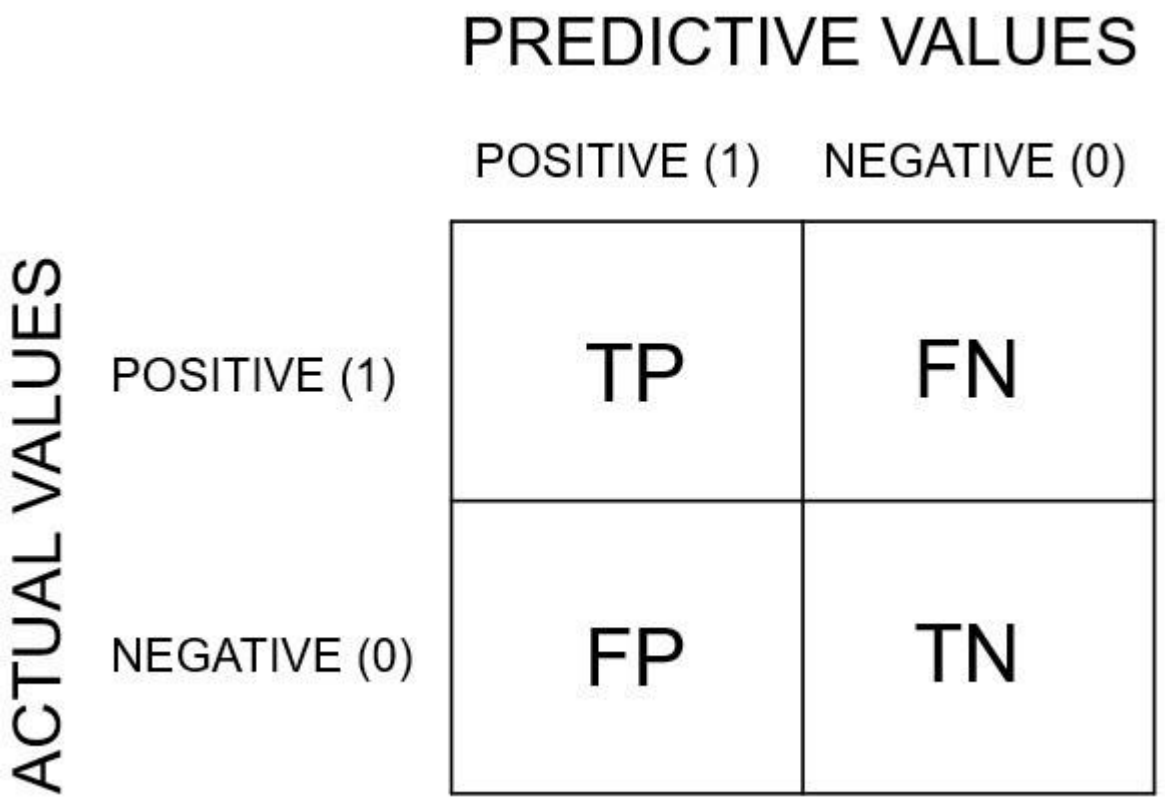
Problem 2 - K Nearest Neighbor (10pts)

- Please use KNN with Euclidean distance to predict the label for data sample Strawberry with K = 1, 3 and 5.
- Why KNN is called a lazy learner? What are the advantage and disarranges of KNN?

Fruit	Sweetness	Sourness	Fruit Type
Lemon	1	9	Sour
Grapfruit	2	8	Sour
Orange	3	7	Sour
Cherry	6	4	Sweet
Banana	9	1	Sweet
Grapes	8	2	Sweet
Strawberry	5	5	?

Metric to Evaluate the performance of classification Technique

Confusion Matrix



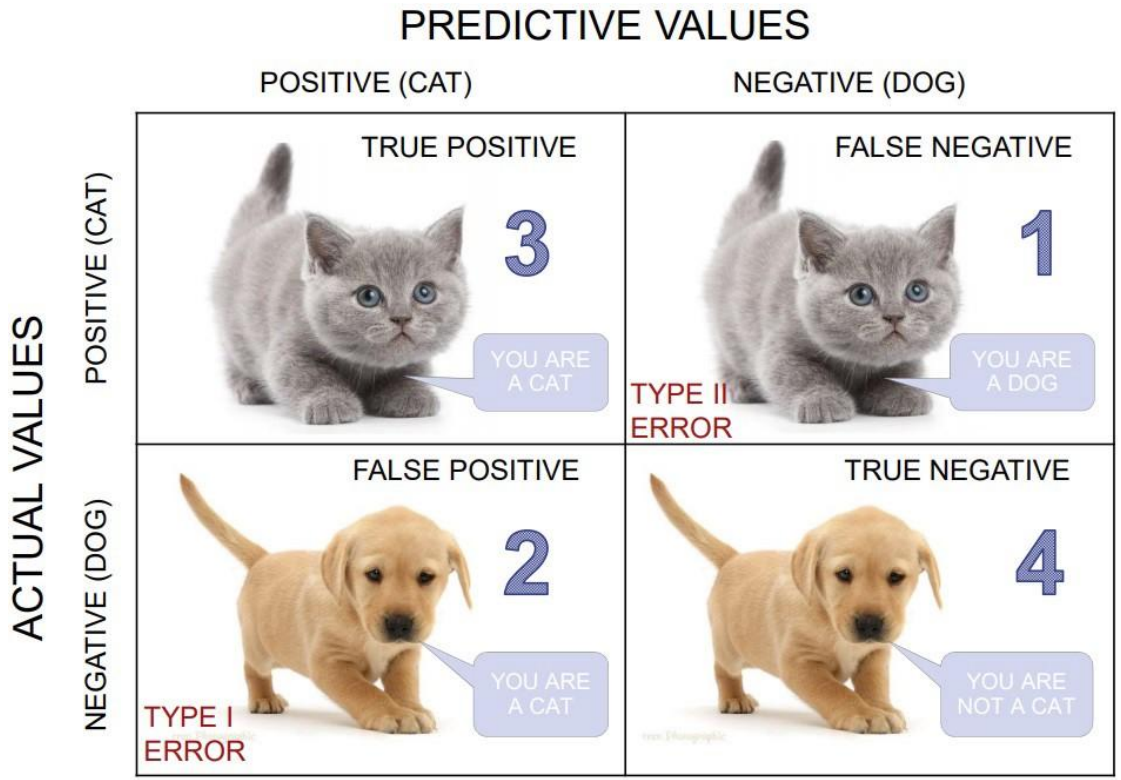
Definition of the Terms:

True Positive: You predicted positive and it's true.

True Negative: You predicted negative and it's true.

False Positive (Type 1 Error): You predicted positive and it's false.

False Negative (Type 2 Error): You predicted negative and it's false.



Classification Accuracy:

Classification Accuracy is given by the relation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Recall (aka Sensitivity):

Recall is defined as the ratio of the total number of correctly classified positive classes divide by the total number of positive classes.

$$Recall = \frac{TP}{TP + FN} \quad \text{or} \quad \frac{\text{True Positive}}{\text{Actual Results}}$$

Confusion Matrix

True Positive (TP)

- The predicted value matches the actual value
- The actual value was positive and the model predicted a positive value

True Negative (TN)

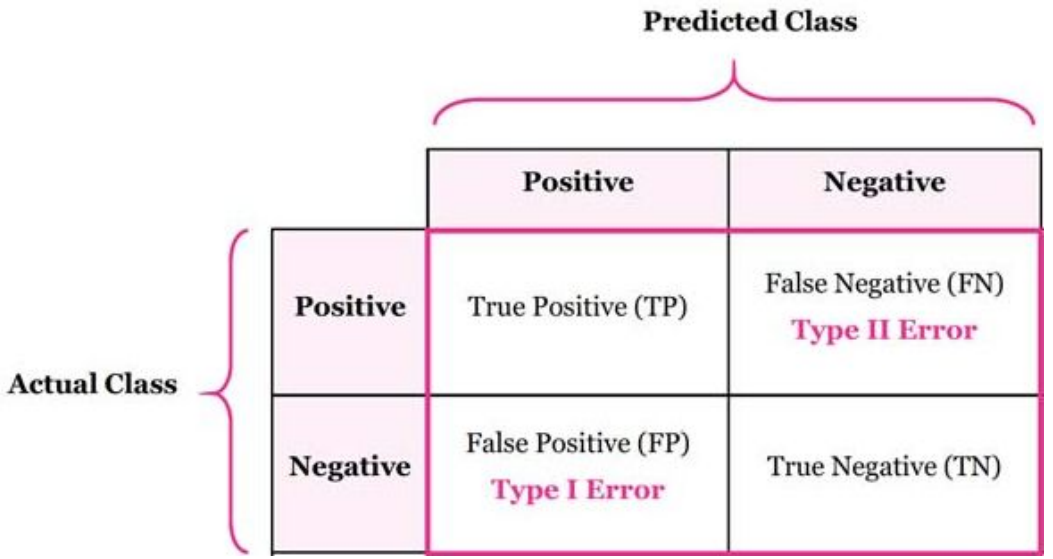
- The predicted value matches the actual value
- The actual value was negative and the model predicted a negative value

False Positive (FP) – Type 1 error

- The predicted value was falsely predicted
- The actual value was negative but the model predicted a positive value
- Also known as the **Type 1 error**

False Negative (FN) – Type 2 error

- The predicted value was falsely predicted
- The actual value was positive but the model predicted a negative value
- Also known as the **Type 2 error**



Precision:

Precision is defined as the ratio of the total number of correctly classified positive classes divided by the total number of predicted positive classes.

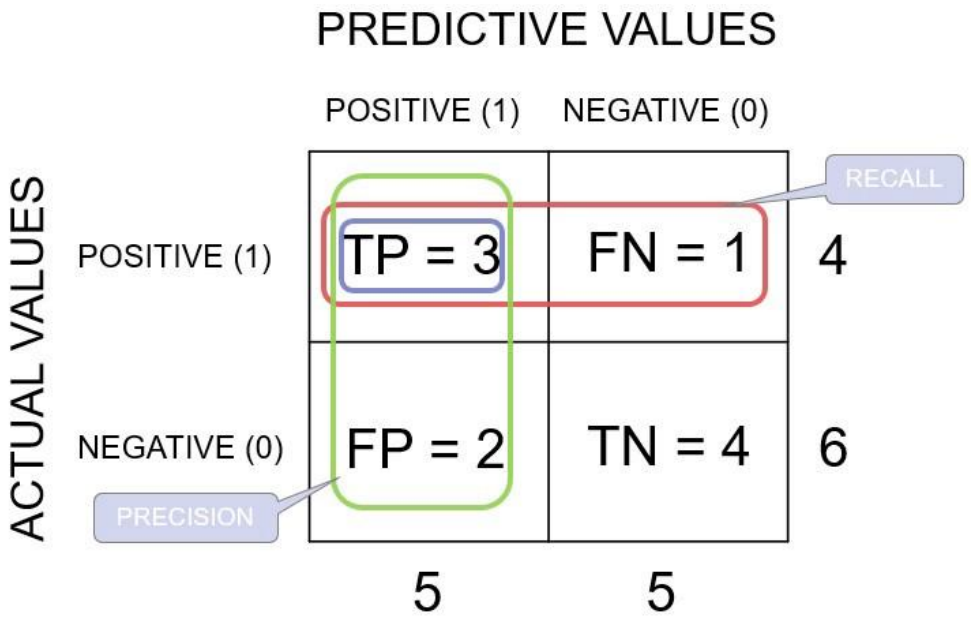
$$\textit{Precision} = \frac{TP}{TP + FP} \quad \text{or} \quad \frac{\text{True Positive}}{\text{Predictive Results}}$$

$$\textit{F - score} = \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}}$$

Specificity:

Specificity determines the proportion of actual negatives that are correctly identified.

$$\textit{Specificity} = \frac{TN}{TN + FP}$$



Classification Accuracy:

Accuracy = (TP + TN) / (TP + TN + FP + FN) =

(3+4)/(3+4+2+1) = 0.70

Recall: Recall gives us an idea about when it's actually yes, how often does it predict yes.

Recall = TP / (TP + FN) = 3/(3+1) = 0.75

Precision: Precsion tells us about when it predicts yes, how often is it correct.

Precision = TP / (TP + FP) = 3/(3+2) = 0.60

F-score:

F-score = (2 * Recall * Precision)/(Recall+Presision) =

(2 *0.75 *0.60)/(0.75+0.60) = 0.67

Specificity:

Specificity = TN / (TN + FP) = 4/(4+2) = 0.67

Strengths of KNN

- Very simple and intuitive.
- Can be applied to the data from any distribution.
- Good classification if the number of samples is large enough.

Weaknesses of KNN

- Takes more time to classify a new example.
  - need to calculate and compare distance from new example to all other examples.
- Choosing k may be tricky.
- Need large number of samples for accuracy.

Thank you