Kaggle #03

QUEEN MARY MACHINE
LEARNING SOCIETY

Agenda:

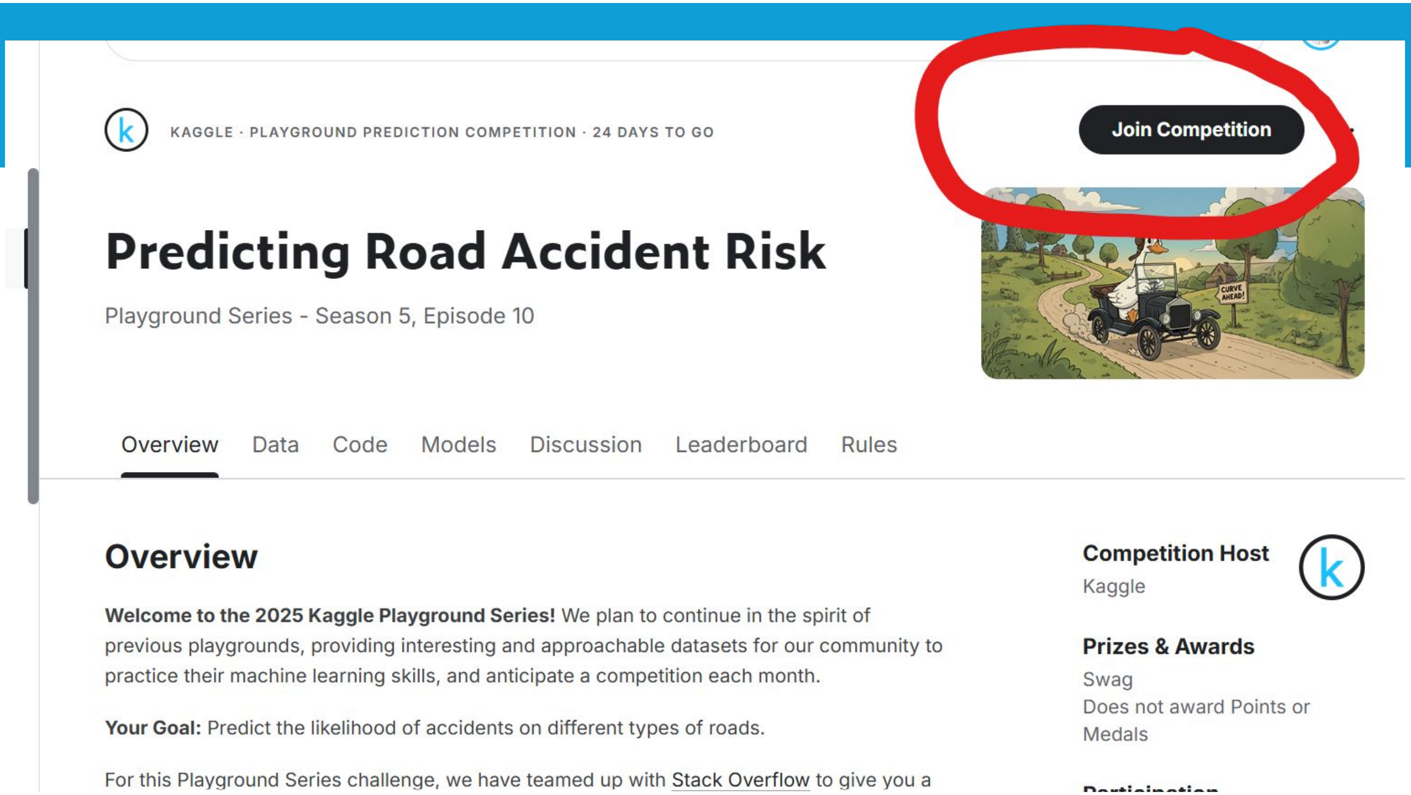
1. What is machine learning?

2. Kahoot

3. Continue work on the kaggle submission

Reminder: If you do want to participate in the Kaggle competitions and/or in the Quant Trading, you should be purchasing the £5.99 membership from QMSU

Recap: Playground Series



KAGGLE · PLAYGROUND PREDICTION COMPETITION · 24 DAYS TO GO

Predicting Road Accident Risk

Playground Series - Season 5, Episode 10

Overview

Welcome to the 2025 Kaggle Playground Series! We plan to continue in the spirit of previous playgrounds, providing interesting and approachable datasets for our community to practice their machine learning skills, and anticipate a competition each month.

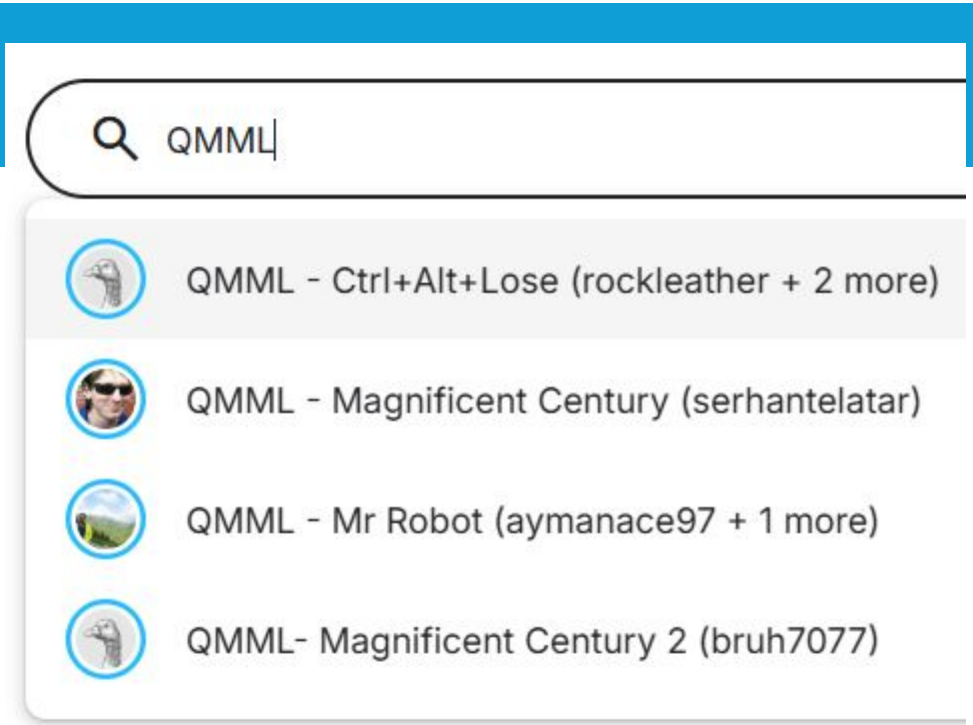
Your Goal: Predict the likelihood of accidents on different types of roads.

For this Playground Series challenge, we have teamed up with Stack Overflow to give you a

Competition Host
Kaggle

Prizes & Awards
Swag
Does not award Points or Medals

Participation



QMML

- QMML - Ctrl+Alt+Lose (rockleather + 2 more)
- QMML - Magnificent Century (serhantelatar)
- QMML - Mr Robot (aymanace97 + 1 more)
- QMML- Magnificent Century 2 (bruh7077)

Link to dataset: https://www.kaggle.com/competitions/playground-series-s5e10/overview
Due: October 31 2025 11:59 PM UTC

Kaggle Teams

- Open the spreadsheet from the QR link
- Pick a row and Enter your Team name + Your names
- Get into teams of 3 people
- Come up to us at the end if you don’t have a team!

https://docs.google.com/spreadsheets/d/1HNIhKU8CkD-5e0b14W2KVPZ2It
B7bTK2hT_NKgNyQxI/edit?usp=drivesdk

What is Machine Learning

A field of study in AI concerned with the development and study of statistical algorithms that can learn from data and generalise with unseen data.

Simple definition: ML is about teaching computers to learn patterns from data rather than being explicitly programmed.

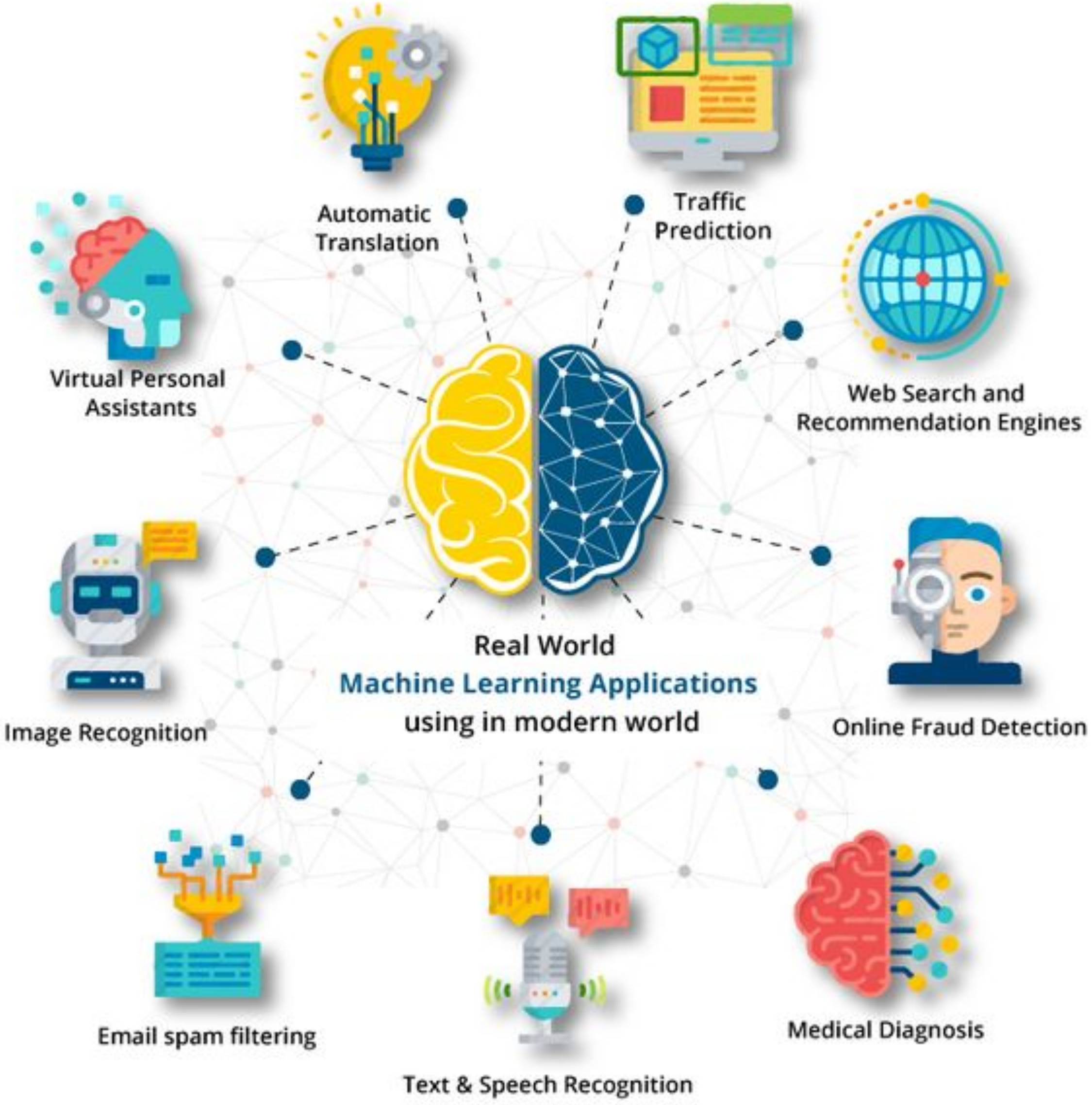
Under the hood:

Lots of fancy maths!

Use cases

Use Cases:

- Retail: Recommender system use ML algorithms to recommend you items you would buy on an ecommerce website
- Finance: Predict stock prices, algorithmic trading
- Healthcare: Predict disease outbreaks/ People at risk of mental illness
- etc.



A Beginner’s Guide to The Machine Learning Workflow

datacamp

1 Project setup

1. Understand the business goals

Speak with your stakeholders and deeply understand the business goal behind the model being proposed. A deep understanding of your business goals will help you scope the necessary technical solution, data sources to be collected, how to evaluate model performance, and more.

2. Choose the solution to your problem

Once you have a deep understanding of your problem—focus on which category of models drives the highest impact. See this **Machine Learning Cheat Sheet** for more information.

2 Data preparation

1. Data collection

Collect all the data you need for your models, whether from your own organization, public or paid sources.

2. Data cleaning

Turn the messy raw data into clean, tidy data ready for analysis. Check out this **data cleaning checklist** for a primer on data cleaning.

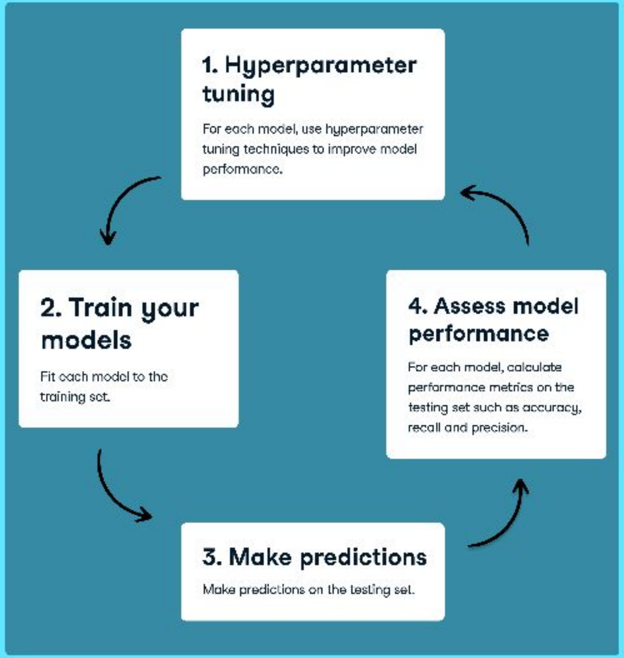
3. Split the data

Randomly divide the records in the dataset into a training set and a testing set. For a more reliable assessment of model performance, generate multiple training and testing sets using cross-validation.

4. Feature engineering

Manipulate the datasets to create variables (features) that improve your model’s prediction accuracy. Create the same features in both the training set and the testing set.

3 Modeling



1. Hyperparameter tuning

For each model, use hyperparameter tuning techniques to improve model performance.

2. Train your models

Fit each model to the training set.

3. Make predictions

Make predictions on the testing set.

4. Assess model performance

For each model, calculate performance metrics on the testing set such as accuracy, recall and precision.

4 Deployment

1. Deploy the model

Embed the model you chose in dashboards, applications, or wherever you need it.

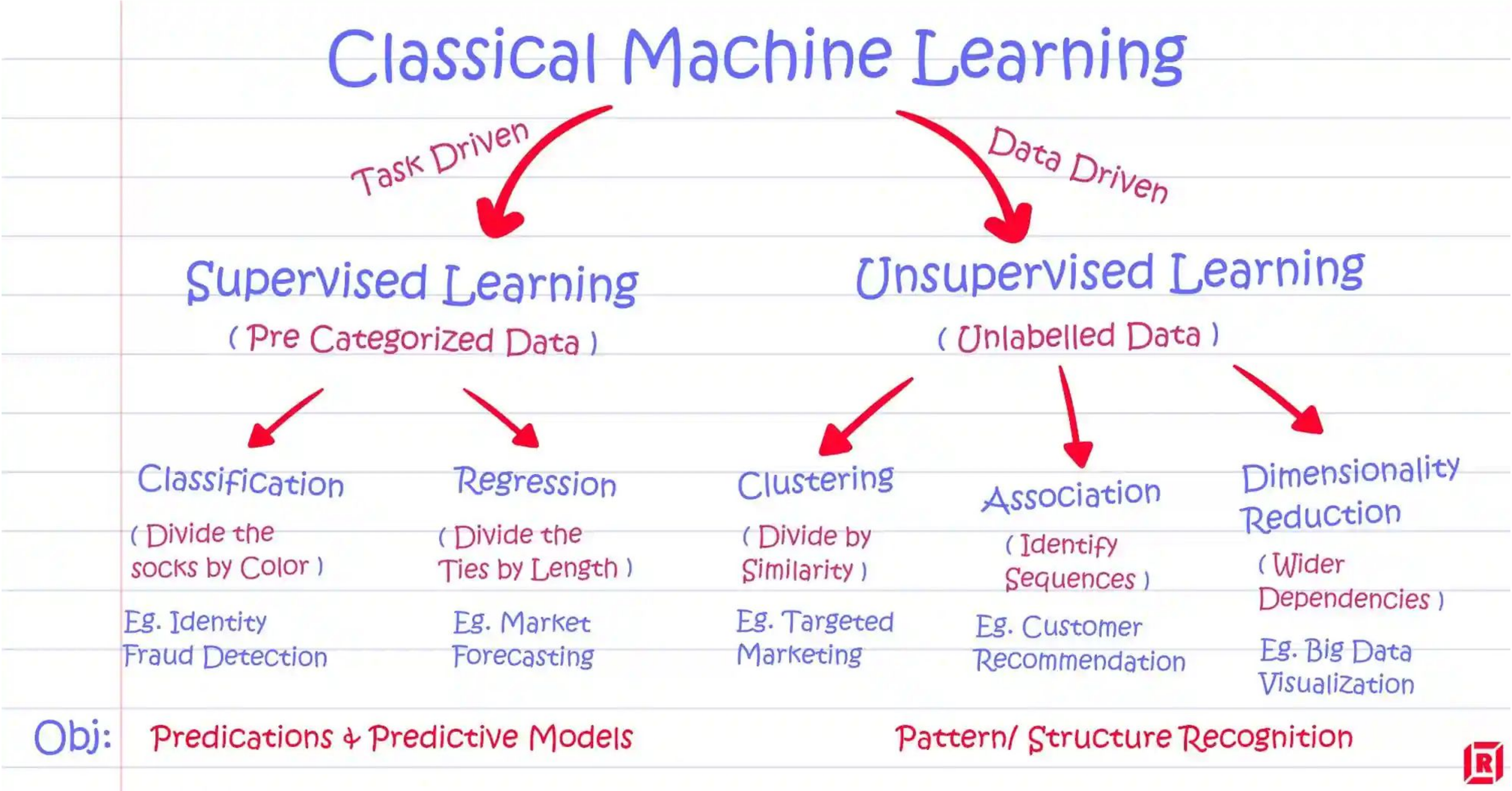
2. Monitor model performance

Regularly test the performance of your model as your data changes to avoid model drift.

3. Improve your model

Continously iterate and improve your model post-deployment. Replace your model with an updated version to improve performance.

Supervised and Unsupervised Learning



Classical Machine Learning

Task Driven

Data Driven

Supervised Learning

(Pre Categorized Data)

Unsupervised Learning

(Unlabelled Data)

Classification

(Divide the socks by Color)

Eg. Identity Fraud Detection

Regression

(Divide the Ties by Length)

Eg. Market Forecasting

Clustering

(Divide by Similarity)

Eg. Targeted Marketing

Association

(Identify Sequences)

Eg. Customer Recommendation

Dimensionality Reduction

(Wider Dependencies)

Eg. Big Data Visualization

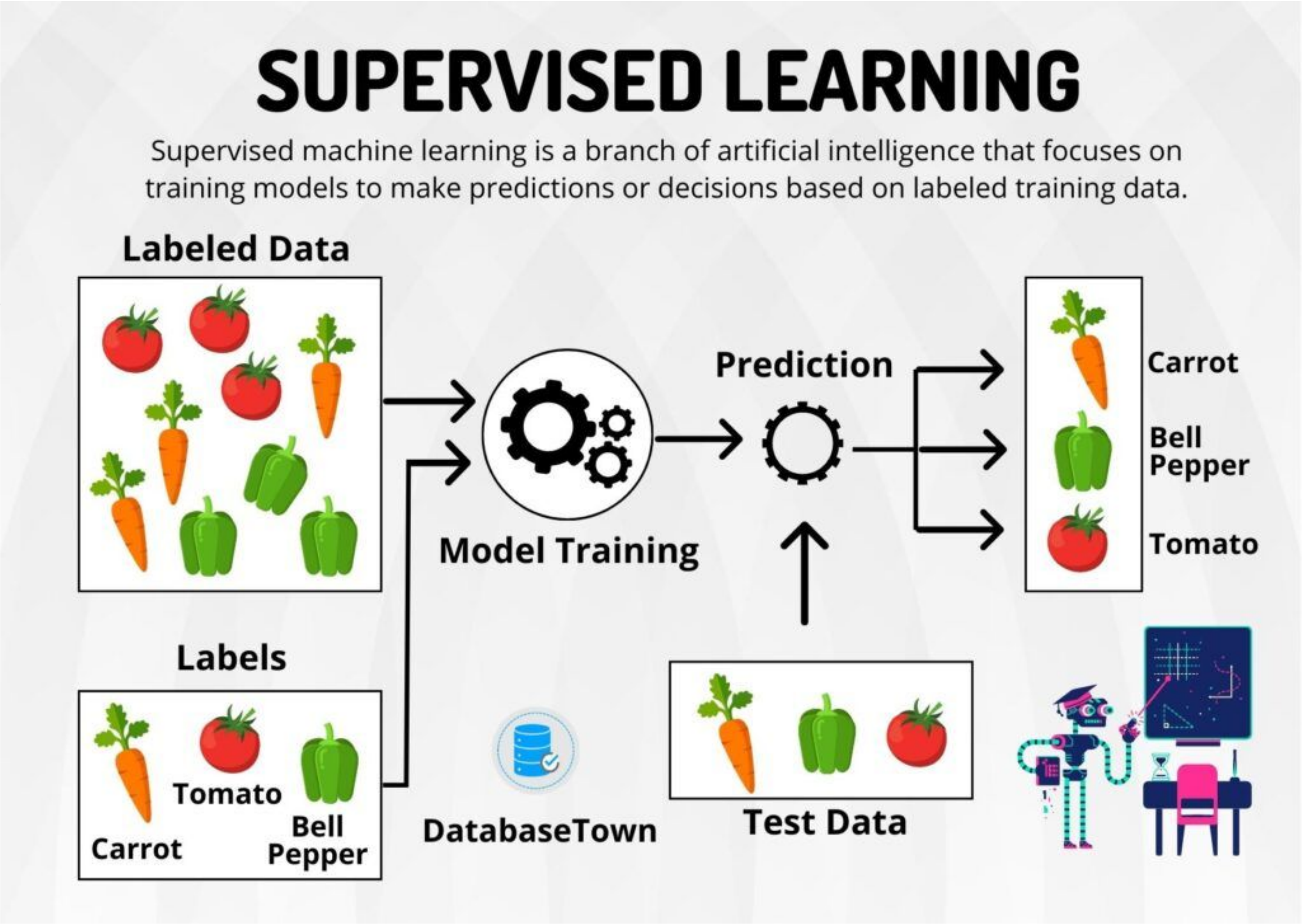
Obj: Predications & Predictive Models

Pattern/ Structure Recognition

Supervised

Uses labelled data, where the model learns from inputs and their corresponding outputs.

For Kaggle Competitions - We will only look at Supervised algorithms

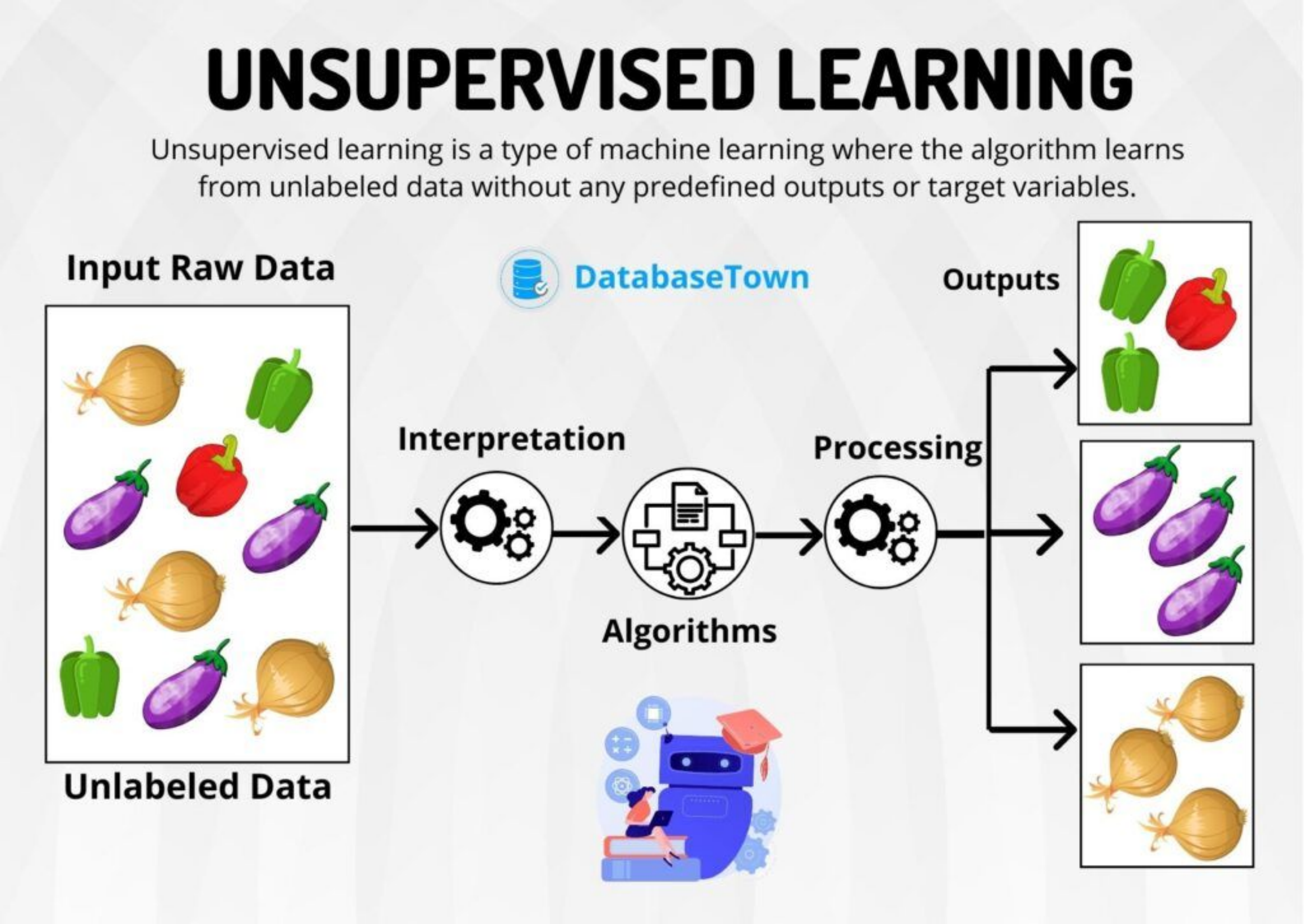


SUPERVISED LEARNING

Supervised machine learning is a branch of artificial intelligence that focuses on training models to make predictions or decisions based on labeled training data.

Unsupervised

Works with unlabelled data and tries to uncover hidden patterns or groupings, such as clustering customers into market segments or reducing complex data into simpler forms

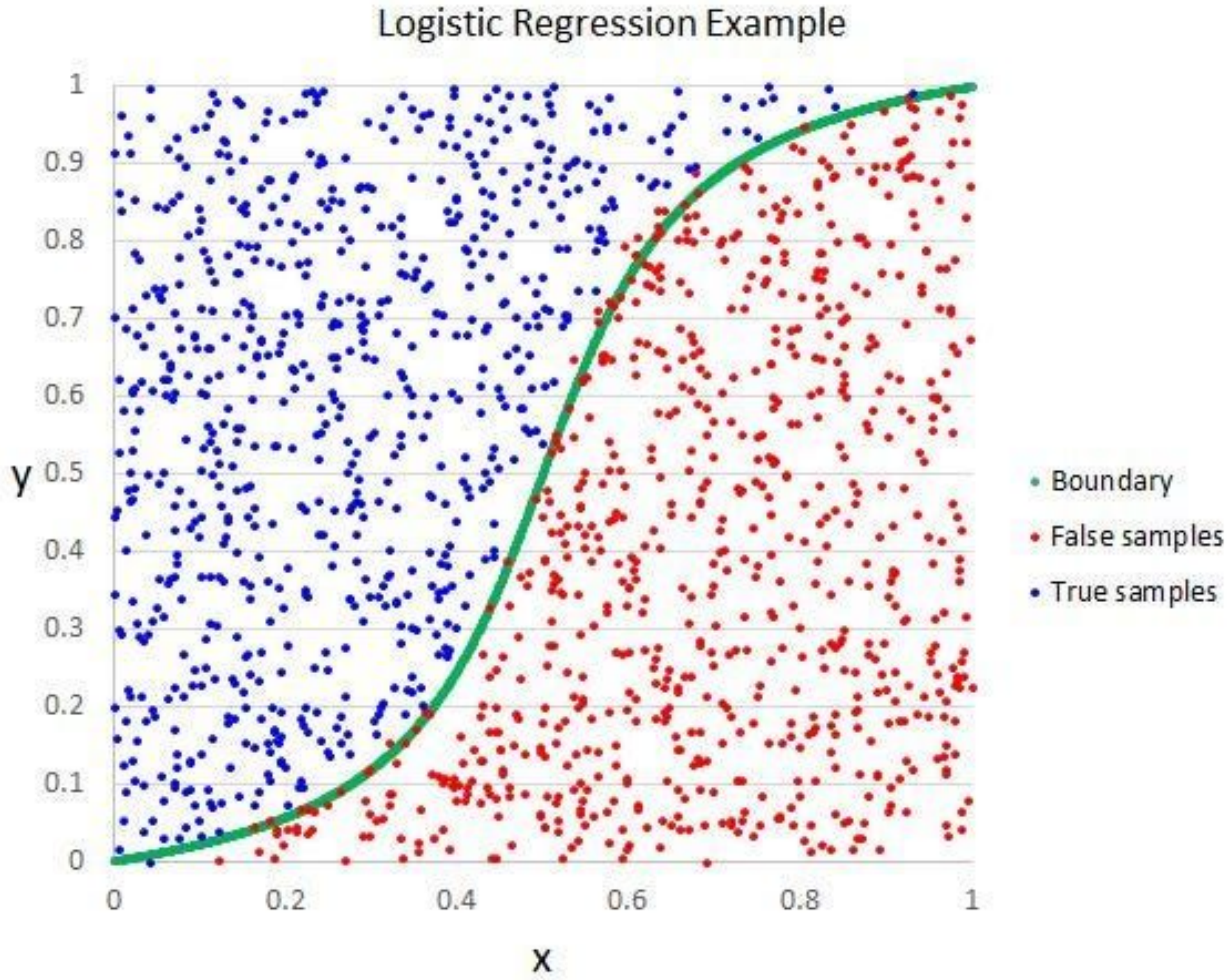


UNSUPERVISED LEARNING

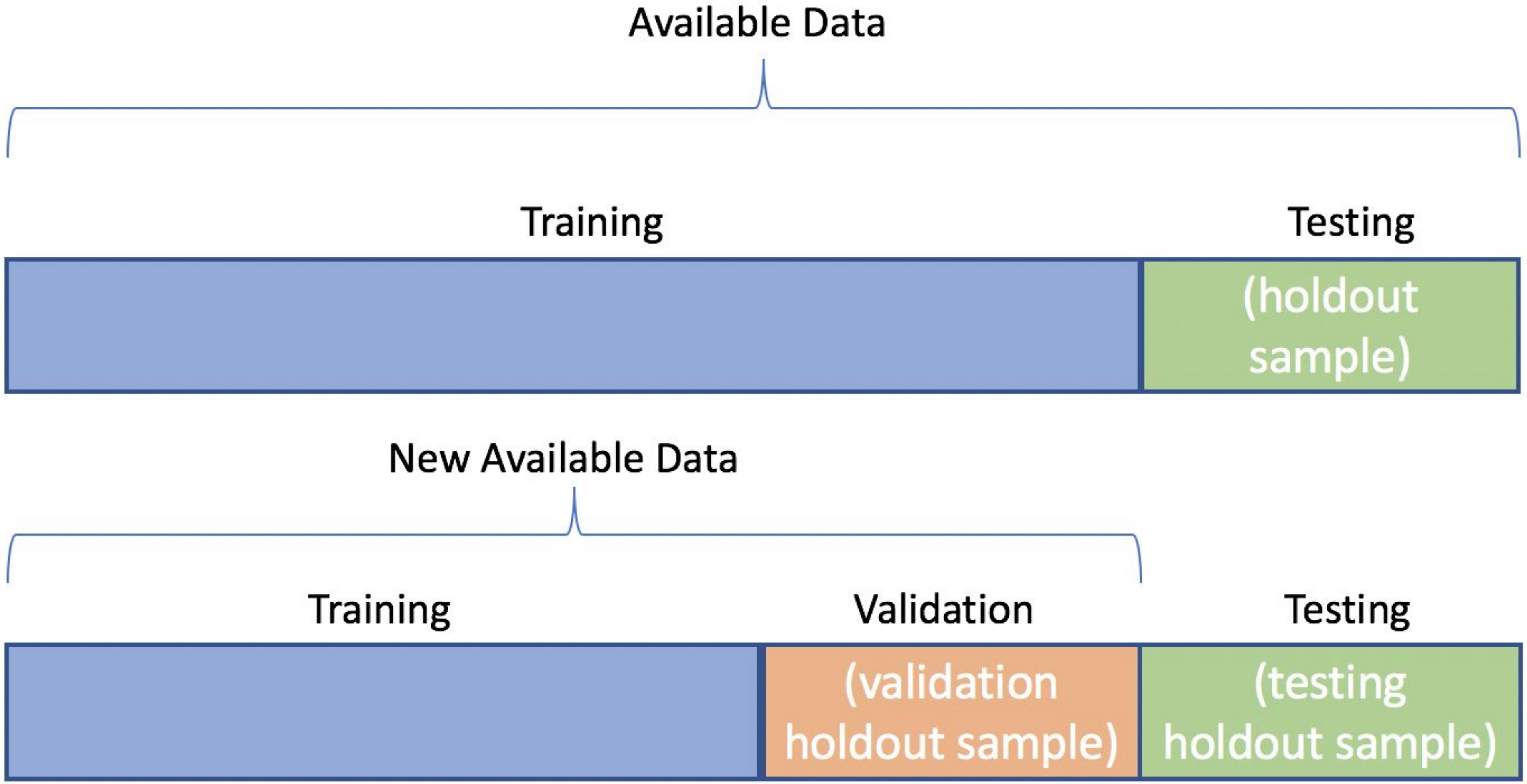
Unsupervised learning is a type of machine learning where the algorithm learns from unlabeled data without any predefined outputs or target variables.

Some Common machine learning algorithms

- **Linear regression** - one of the simplest, predicting continuous values such as house prices.
- **Logistic regression** - is used for classification tasks such as spam detection
- **Decision Trees** - split data into branches to make predictions, often in a way that is easy to interpret
- **Support Vector Machines** - tries to find the best boundary that separates different classes


Logistic Regression Example
y
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.2
0.4
0.6
0.8
1
x
● Boundary
● False samples
● True samples

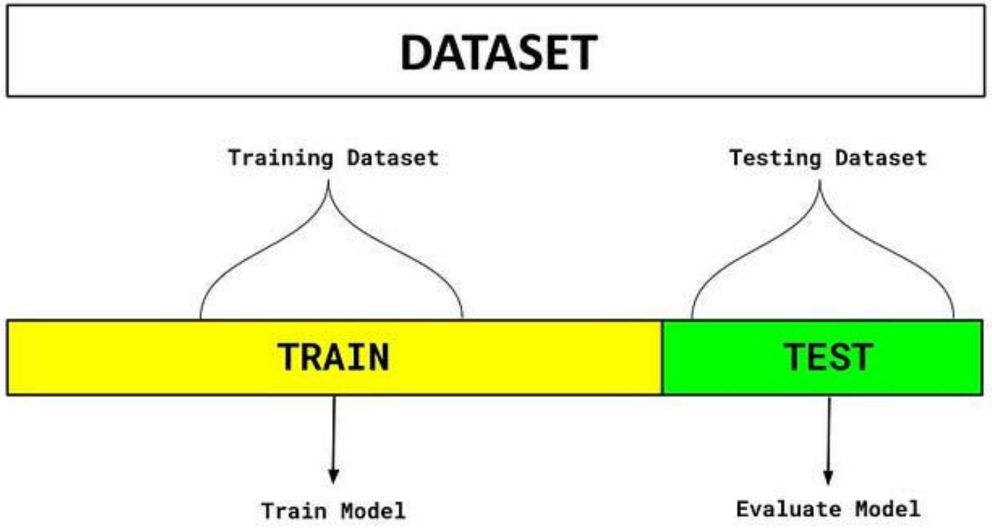
Train-Test Validation split



Note there are 2 datasets in kaggle competitions

Training data: helps the model learn

Testing data: evaluates the model's performance on new, unseen data

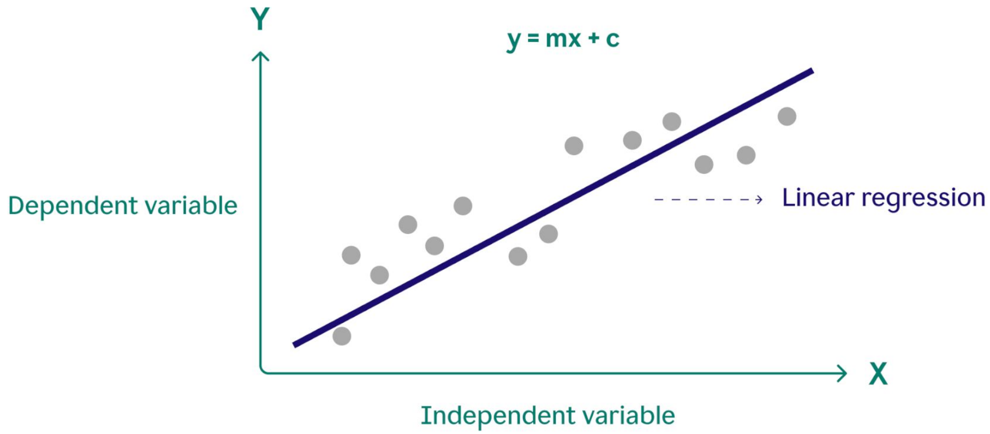


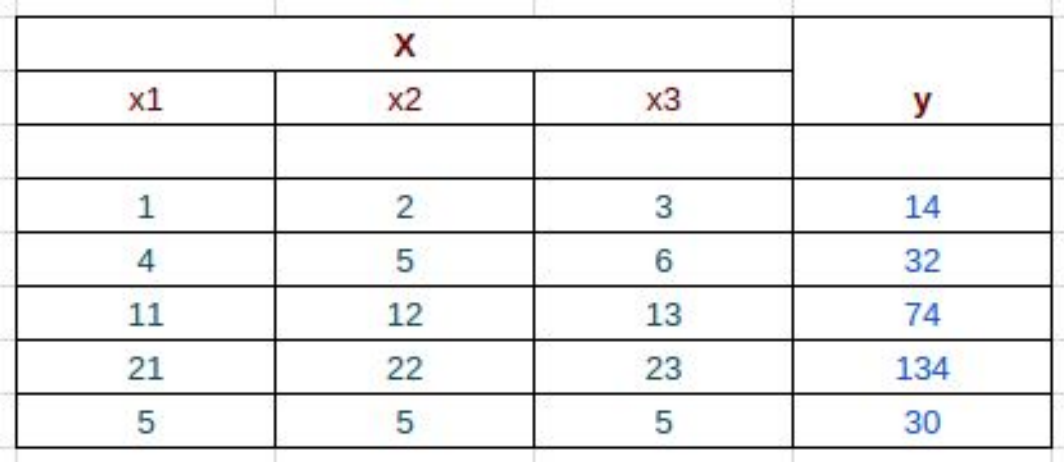
Submission.csv -> where you save your results to submit for the competition

What is X and y?

X = the input features (independent variables) used to train a model

Y = the target variable (dependent variable or output) that the model is trying to predict.





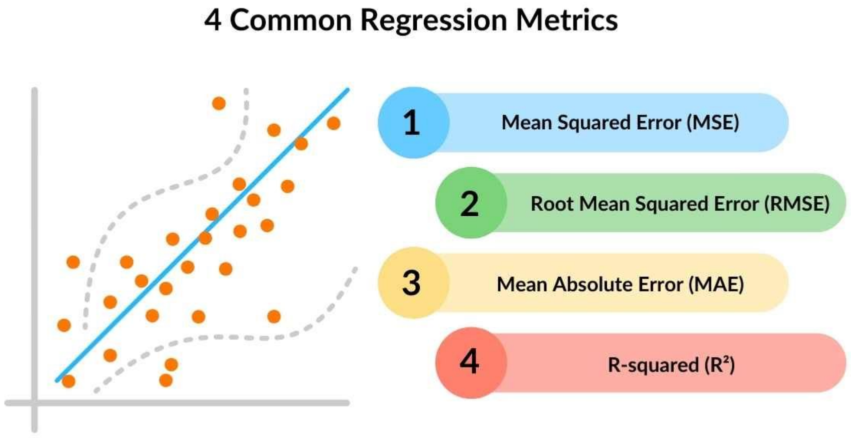
X			y
x1	x2	x3	
1	2	3	14
4	5	6	32
11	12	13	74
21	22	23	134
5	5	5	30

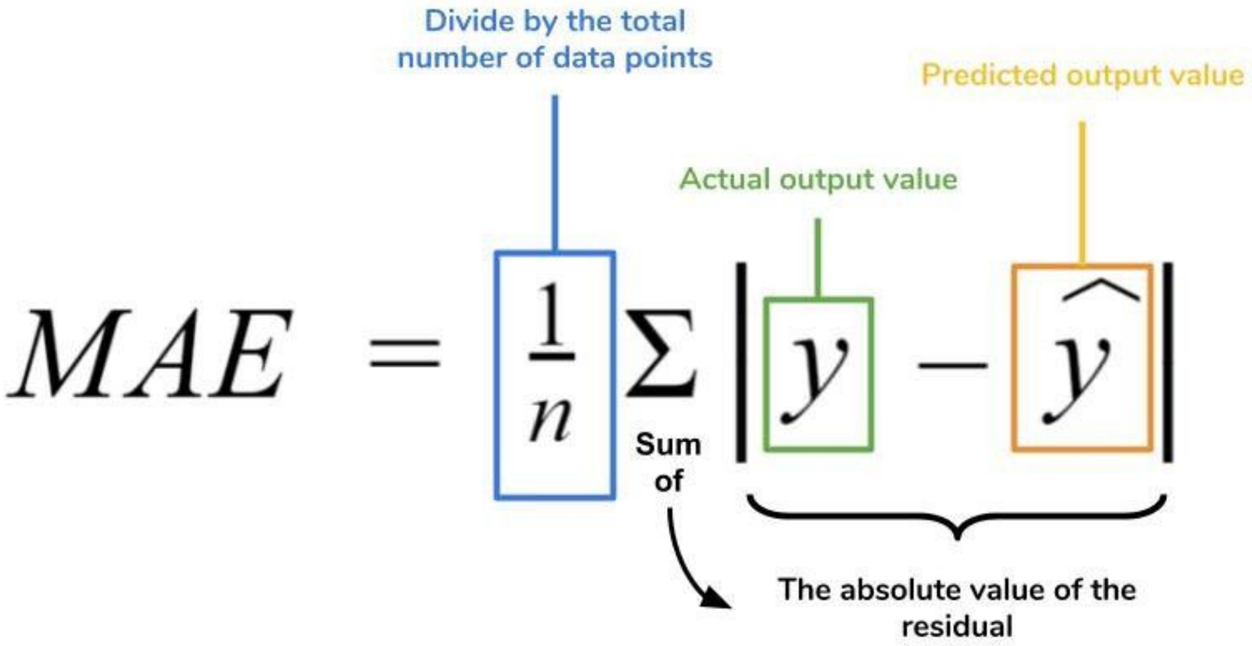
Challenges & Limitations

- **Data Quality** - If the input data is biased or incomplete, the model will produce poor predictions.
- **Overfitting/Underfitting** - we are looking for **generalisation**.
- **Bias and Fairness** - in case the models are trained on biased data

Common Regression Metrics

- **MAE:** Takes the average of all absolute errors.
    - On average, how far are the predictions from the actual values?
- **MSE:** Squares the errors before averaging.
    - Similar to MAE but big mistakes are punished much more.
- **RMSE:** Square root of MSE.
    - Same as MSE but brought back to original units of the target.
- **R^2:** percentage of variance explained by the model.


4 Common Regression Metrics
1 Mean Squared Error (MSE)
2 Root Mean Squared Error (RMSE)
3 Mean Absolute Error (MAE)
4 R-squared (R^2)



$$MAE = \boxed{\frac{1}{n}} \sum \left| \boxed{y} - \boxed{\hat{y}} \right|$$

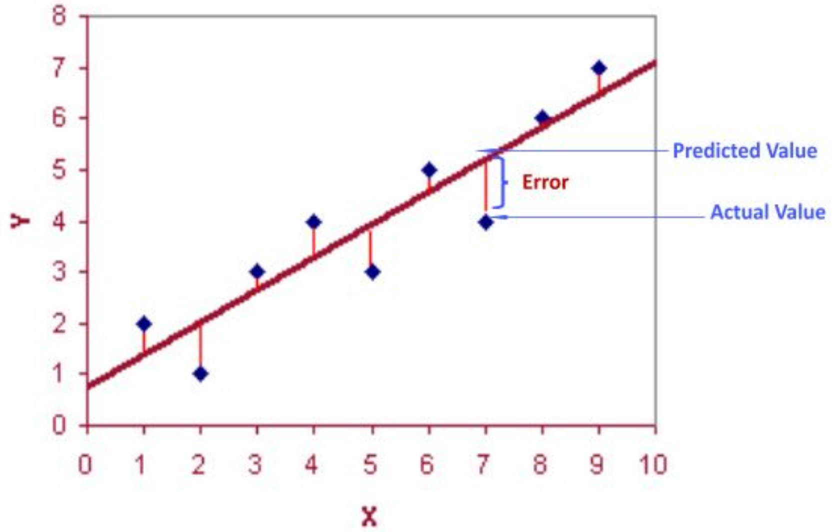
Divide by the total number of data points

Predicted output value

Actual output value

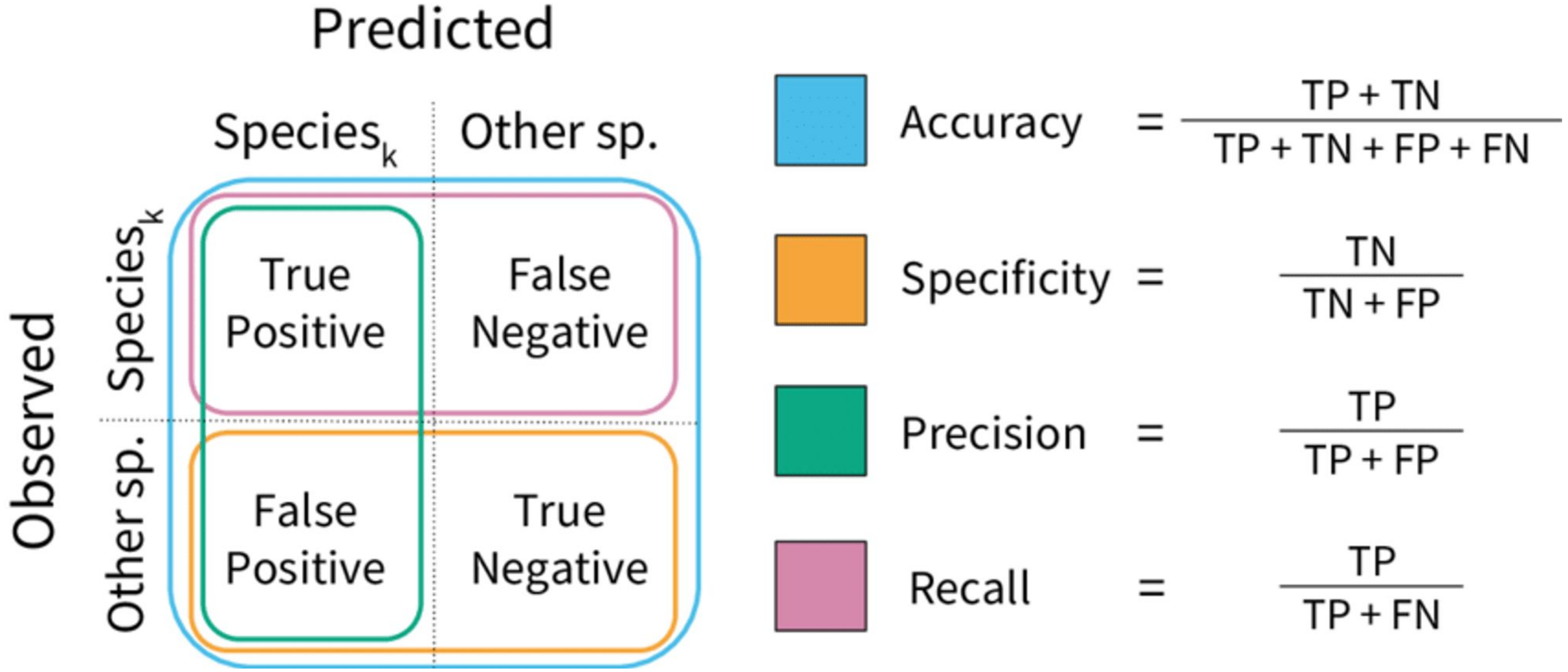
Sum of

The absolute value of the residual



Classification metrics

- **Accuracy:** overall correctness
-
- **Precision :** “When I predict positive, how often am I right?”
-
- **Recall:** “How many real positives did I catch?”
-
- **F1-score:** balance between precision & recall
-
- **Confusion Matrix:** full picture of mistakes



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

🟧 Specificity $= \frac{TN}{TN + FP}$

🟩 Precision $= \frac{TP}{TP + FP}$

🟪 Recall $= \frac{TP}{TP + FN}$

- **False Negative (FN) :** patient has cancer but test says “No cancer”. Very dangerous, because the disease is missed.

- **False Positive (FP) :** important email marked as spam.You miss an important job offer or bank notification.

But there are so many algorithms to learn!

For these Kaggle competitions we will only look at tree based models for now.