# Artificial Intelligence (AI)

Select one or more choices from the list of common Machine Learning Algorithms, do some investigations and write me a short summary. I am looking for the following:

* Linear Regression
* Logistic Regression
* Decision Tree
* SVM (Support Vector Machine)
* Naive Bayes
* KNN (K- Nearest Neighbours)
* K-Means
* Random Forest

**Your task is to (max 500 words):**

* Is it Supervised/Unsupervised/Reinforcement learning?
* What does the algorithm do?
* In which situations will it be most useful?
* (Optional) Can you find any examples of where this algorithm has been used?

Extra:

From this website select follow and complete the Linear Regression Model.

<https://stackabuse.com/linear-regression-in-python-with-scikit-learn/>

# Linear Regression

* Is it Supervised/Unsupervised/Reinforcement learning? and what does the algorithm do?
* In which situations will it be most useful?
* (Optional) Can you find any examples of where this algorithm has been used?

# What is Linear Regression and What Does the Algorithm Do?

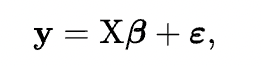
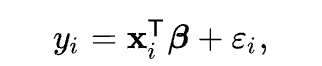
It is a supervised learning method for finding the coefficients to a linear model, i.e., output or y is a linear combination of input variables or X.

The most common algorithm used is ordinary least squares (OLE) or Gradient Descent in machine learning.

## Using Ordinary Least Squares to calculate Coefficients (betas)

Given a regression line, we can calculate the distance of each data point to the regression line (we call this the error/residuals/epsilons), square the value, then sum all of these errors. The aim is to minimise this value for different coefficients that give different regression lines.

We can use linear algebra to do this, but you would call a procedure rather than implement this in code yourself.



The equations for linear regression line in vector (left) and matrix notation (right)

## 

## Gradient Descent (GD)

An analogy, you are trying to descend a mountain in fog, you would intuitively, move in the direction that was the least steep and make a step forward and continue until you eventually reach the bottom.

In this analogy, the person represents the algorithm, and the path taken down the mountain represents the sequence of parameter settings that the algorithm will explore. The steepness of the hill represents the slope of the error surface at that point. differentiation (the slope of the error surface can be calculated by taking the derivative of the squared error function at that point). The direction they choose to travel in aligns with the gradient of the error surface at that point. The amount of time they travel before taking another measurement is the step size.

GD starts by using random values for each coefficient. The sum of the squared errors are calculated for each pair of input and output values. A learning rate is used as a scale factor and the coefficients are updated in the direction towards minimising the error. The process is repeated until a minimum sum squared error is achieved or no further improvement is possible. In addition, you must select a learning rate (alpha) parameter that determines the size of the improvement step to take on each iteration of the procedure.

# In which situations will it be most useful?

There are generally four assumptions associated with linear regression models:

* Linear relationship: There's a linear relationship between the independent variable x and the dependent variable y.
* Independence: The residuals are independent. There's no correlation between consecutive residuals in time-series data.
* Homoscedasticity: The residuals have equal variance at all levels.
* Normality: The residuals are normally distributed.

Here are some ways to account for incomplete data and create a more reliable prediction model.

* Linear regression thinks that the predictor and response variables aren't noisy. Due to this, removing noise with several data clearing operations is crucial. If possible, you should remove the outliers in the output variable.
* If the input and output variables have Gaussian distribution, linear regression will make better predictions.
* If you rescale input variables using normalisation or standardisation, linear regression will generally make better predictions.
* If there are many attributes, you need to transform the data to have a linear relationship.
* If the input variables are highly correlated, then linear regression will overfit the data. In such cases, remove collinearity.

Here are some notable advantages of linear regression:

* It's a go-to algorithm because of its simplicity.
* Although it's susceptible to overfitting, it can be avoided with the help of dimensionality reduction techniques.
* It has good interpretability.
* It performs well on linearly separable datasets.
* Its space complexity is low; therefore, it's a high latency algorithm

# Real Life Uses

1. Medical researchers often use linear regression to understand the relationship between drug dosage and blood pressure of patients.
2. Agricultural scientists often use linear regression to measure the effect of fertiliser and water on crop yields.
3. Businesses often use linear regression to understand the relationship between advertising spending and revenue.