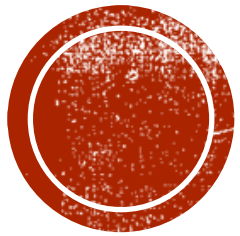
TechieNest
...Transforming Engineers to Technocrats



MACHINE LEARNING

(Dive Deep with Mr.Sanam)

TechieNest
...Transforming Engineers to Technocrats

INDEX – MACHINE LEARNING

INDEX – SUPERVISED LEARNING

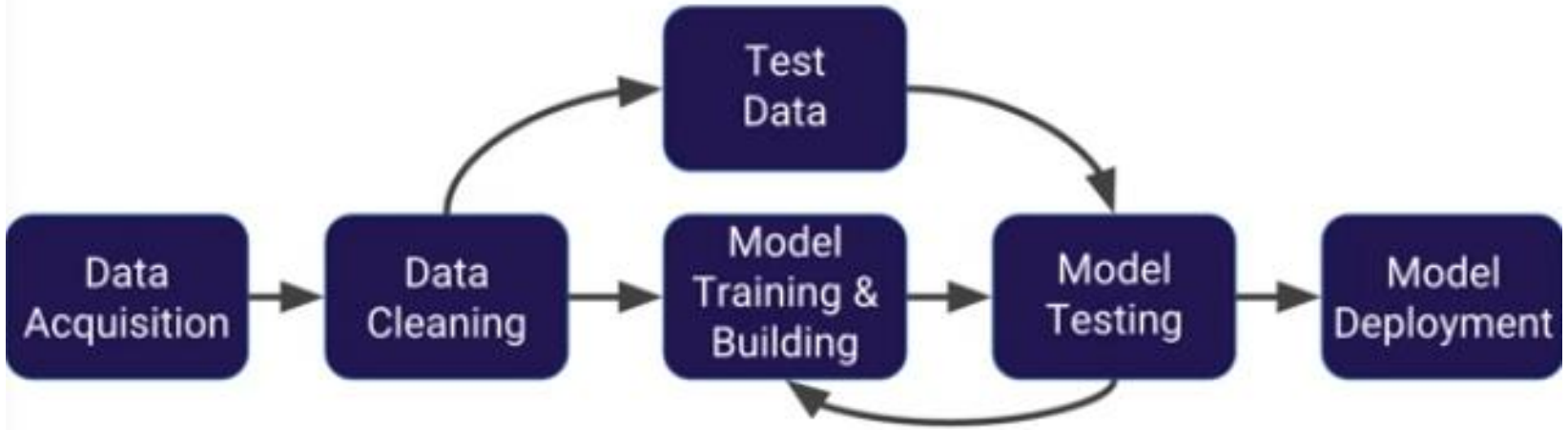
TechieNest
...Transforming Engineers to Technocrats

INDEX – UNSUPERVISED LEARNING

TechieNest
...Transforming Engineers to Technocrats

INDEX – REINFORCEMENT LEARNING

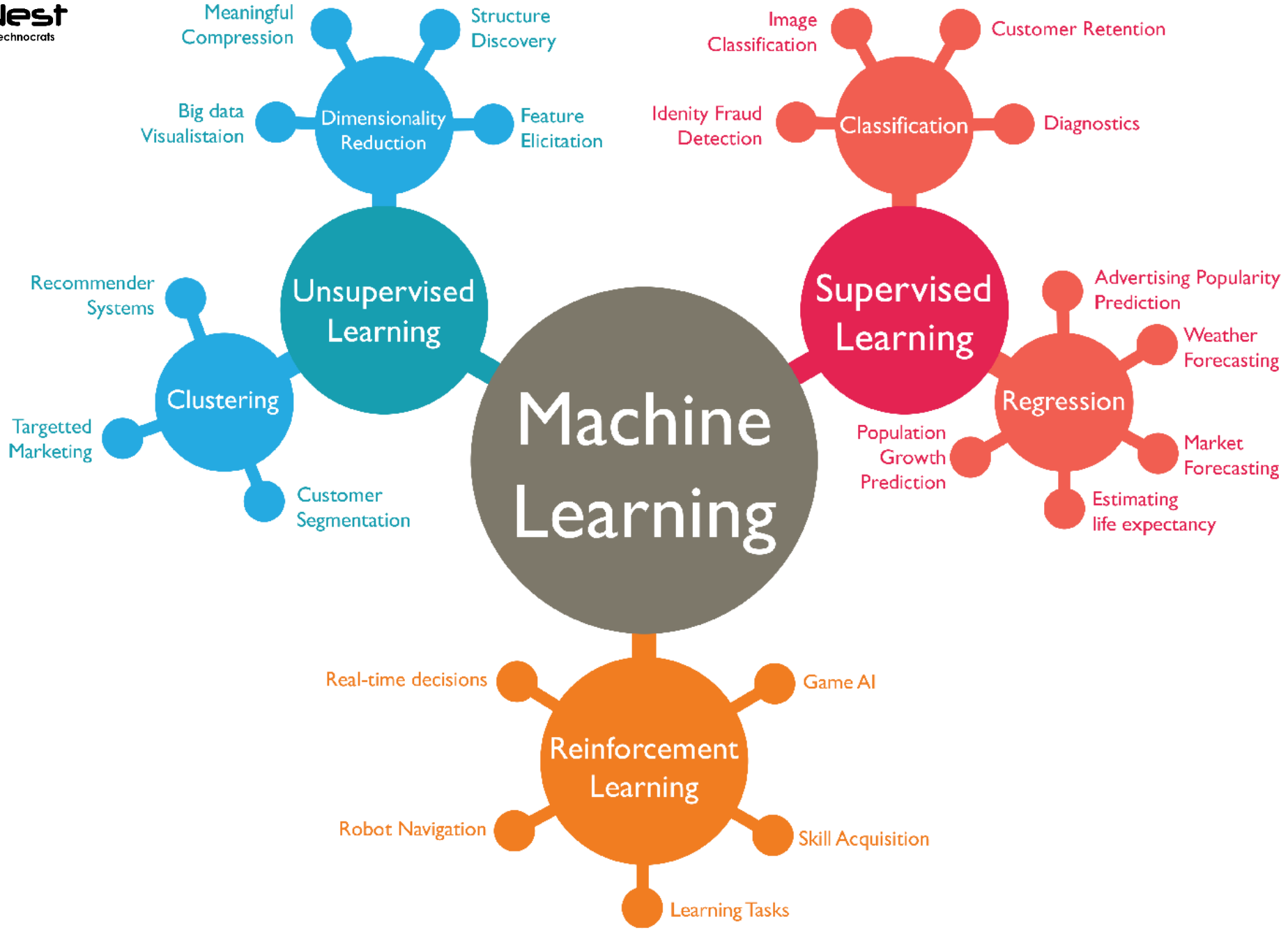
HOW MACHINE LEARNING WORKS



TYPES OF MACHINE LEARNING

- Supervised Learning
  - You have labeled data and are trying to predict a label based off of known features
- Unsupervised Learning
  - You have unlabeled data and are trying to group together similar data points based off of features
- Reinforcement Learning
  - Algorithm learns to perform an action from experience

TechieNest
...Transforming Engineers to Technocrats



SUPERVISED LEARNING

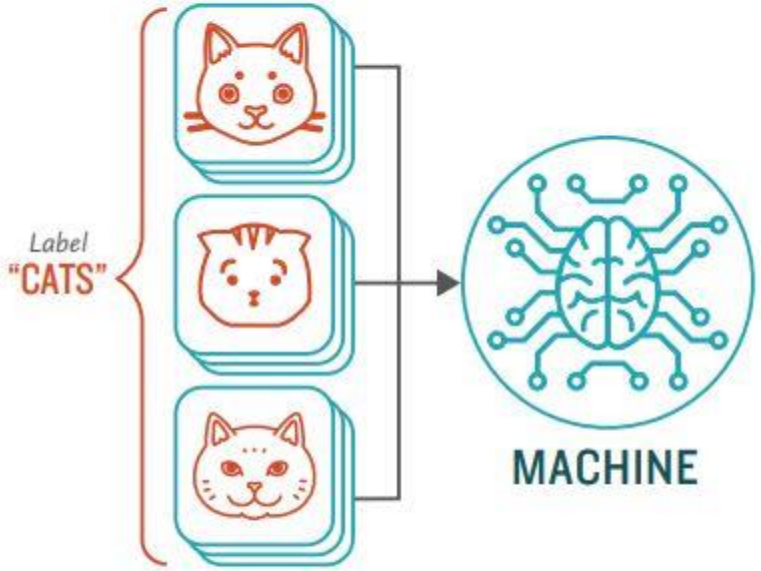
- Methods used like: classification, regression, prediction and gradient boosting
- Used where historical data predicts likely future events.
- For example, it can attempt to predict the price of a house based on different features for houses for which we have historical price data.

TechieNest
...Transforming Engineers to Technocrats

How Supervised Machine Learning Works

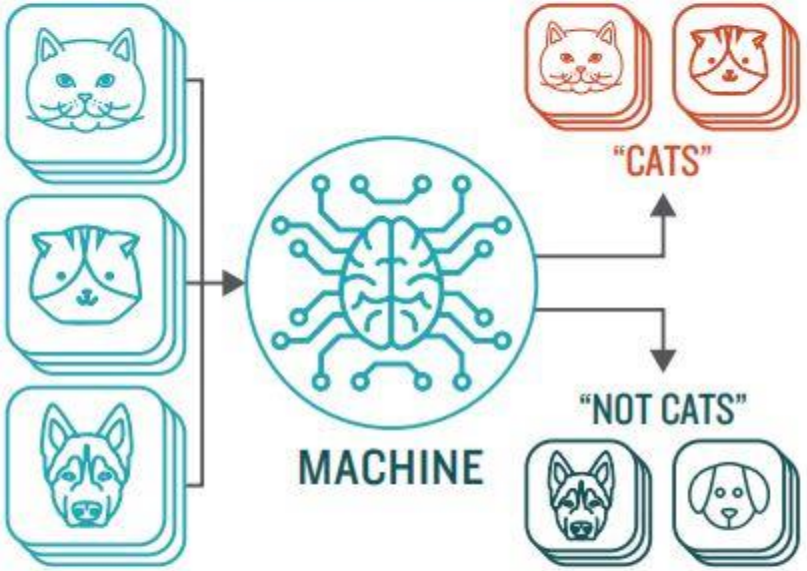
STEP 1

Provide the machine learning algorithm categorized or "labeled" input and output data from to learn

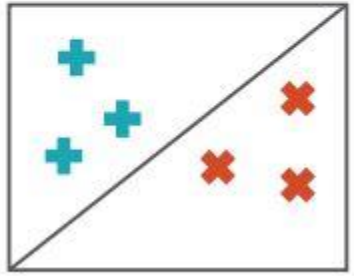


STEP 2

Feed the machine new, unlabeled information to see if it tags new data appropriately. If not, continue refining the algorithm

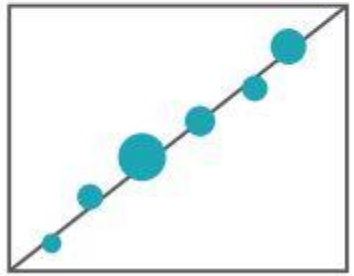


TYPES OF PROBLEMS TO WHICH IT'S SUITED



CLASSIFICATION

Sorting items into categories



REGRESSION

Identifying real values (dollars, weight, etc.)

UNSUPERVISED LEARNING

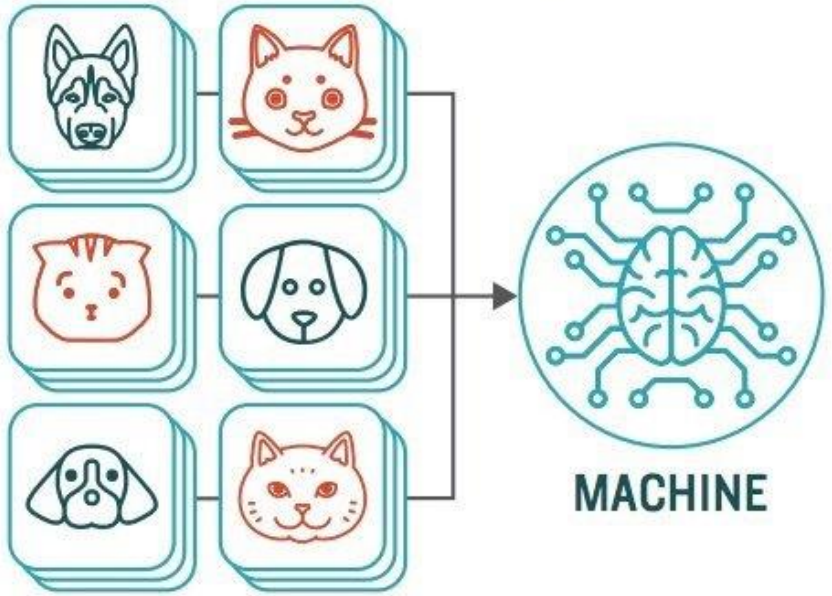
- Used against data that has no historical labels
- The goal is to explore the data and find some structure within
- It can find the main attributes that separate customer segments from each other
- Also used to segment text topics, recommend items and identify data outliers
- Methods used: self organizing maps, nearest-neighbor mapping, k-means clustering and single value decomposition

TechieNest
...Transforming Engineers to Technocrats

How Unsupervised Machine Learning Works

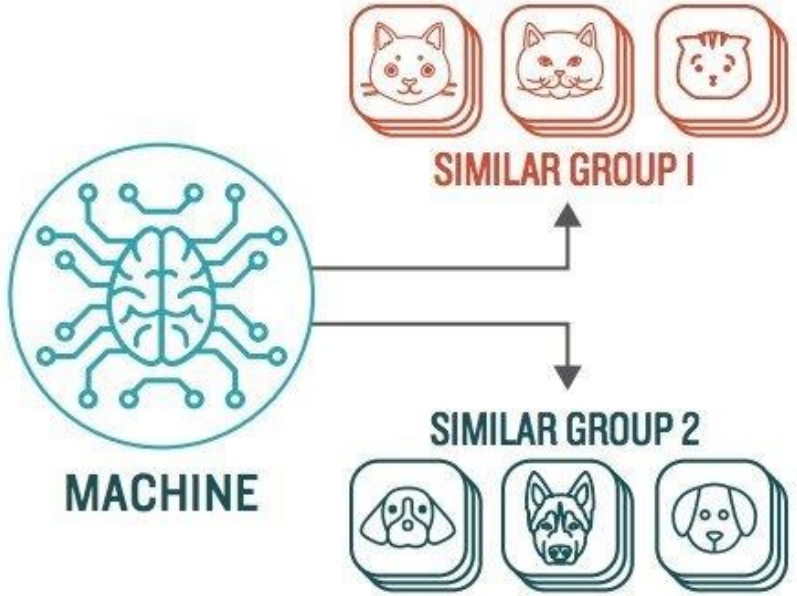
STEP 1

Provide the machine learning algorithm uncategorized, unlabeled input data to see what patterns it finds

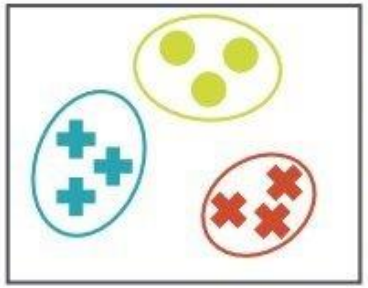


STEP 2

Observe and learn from the patterns the machine identifies



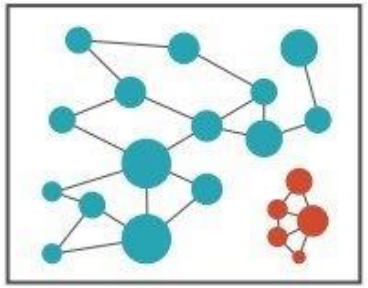
TYPES OF PROBLEMS TO WHICH IT'S SUITED



CLUSTERING

Identifying similarities in groups

For Example: Are there patterns in the data to indicate certain patients will respond better to this treatment than others?



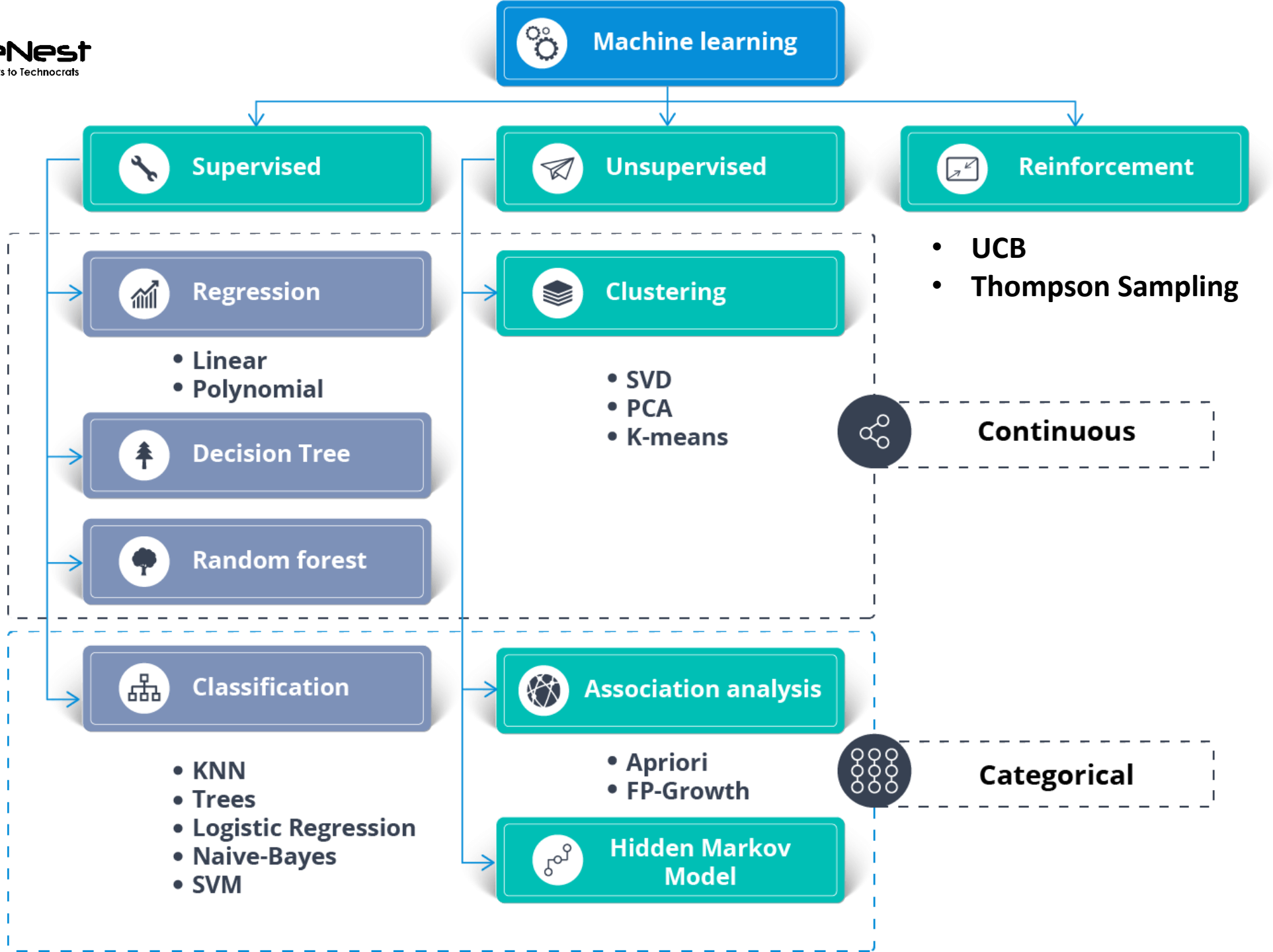
ANOMALY DETECTION

Identifying abnormalities in data

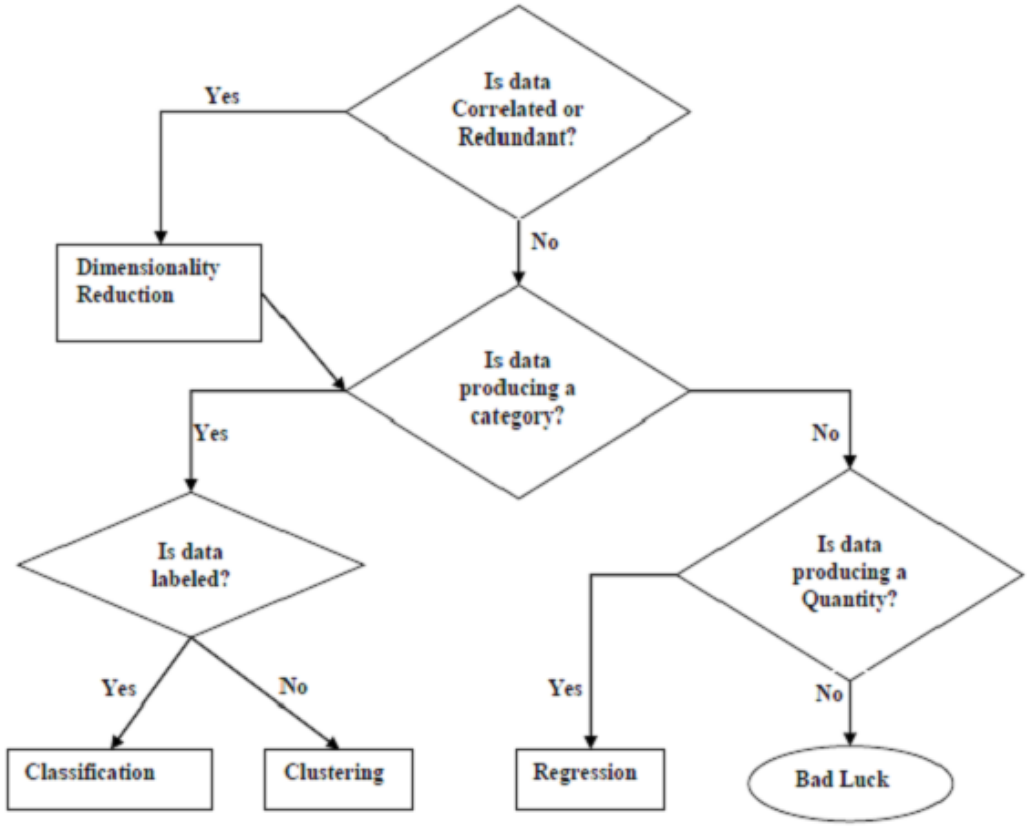
For Example: Is a hacker intruding in our network?

REINFORCEMENT LEARNING

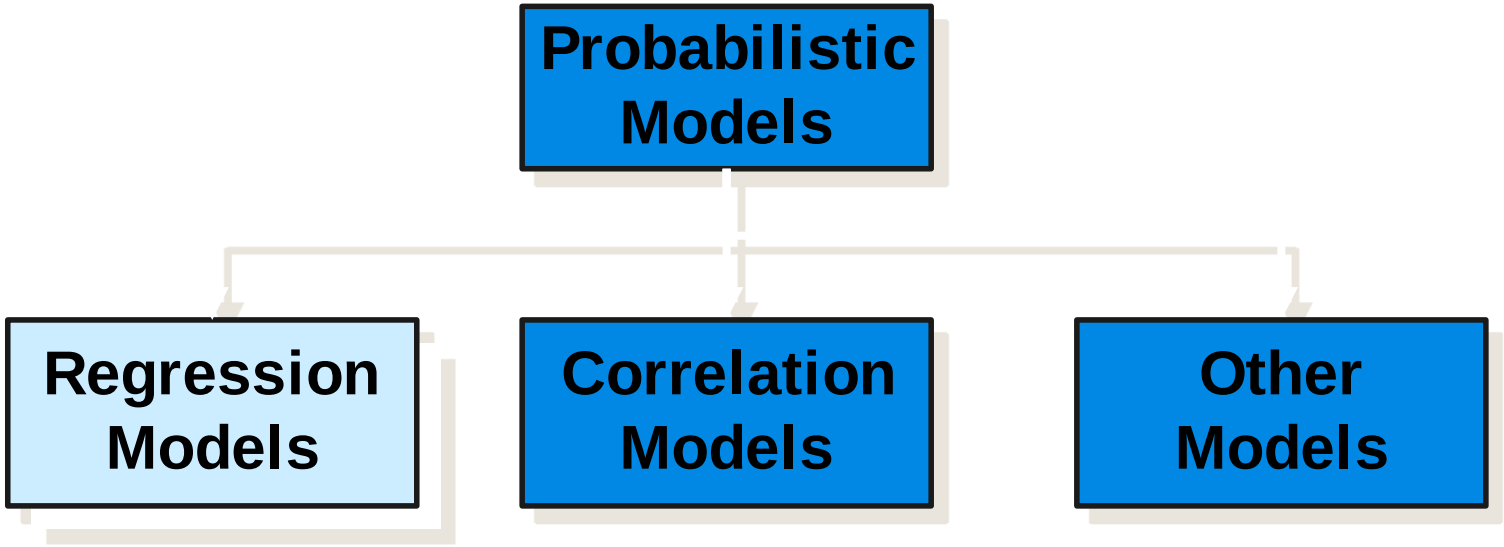
- Often used for robotics, gaming and navigation
- The algorithm discovers through trial and error which actions yield greatest rewards
- It has three primary components:
  - Agent (the learner or decision maker),
  - Environment (Everything the agent interacts with)
  - Actions (what agent can do)
- Objective is for the agent to choose actions that maximize the expected reward over a given amount of time
- The agent will reach the goal much faster by following a good policy


Machine learning
Supervised
Unsupervised
Reinforcement
Regression
Linear
Polynomial
Decision Tree
Random forest
Classification
KNN
Trees
Logistic Regression
Naive-Bayes
SVM
Clustering
SVD
PCA
K-means
Association analysis
Apriori
FP-Growth
Hidden Markov Model
UCB
Thompson Sampling
Continuous
Categorical

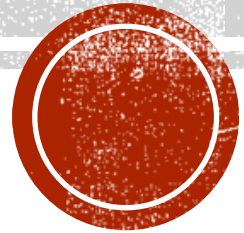
MACHINE LEARNING FLOW



MODELING

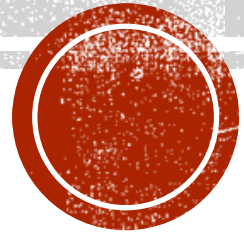

Probabilistic Models
Regression Models
Correlation Models
Other Models

SUPERVISED LEARNING

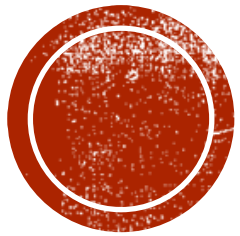


TechieNest
...Transforming Engineers to Technocrats

REGRESSION MODELLING



TechieNest
...Transforming Engineers to Technocrats



LINEAR REGRESSION

Regression Modelling

REGRESSION MODEL

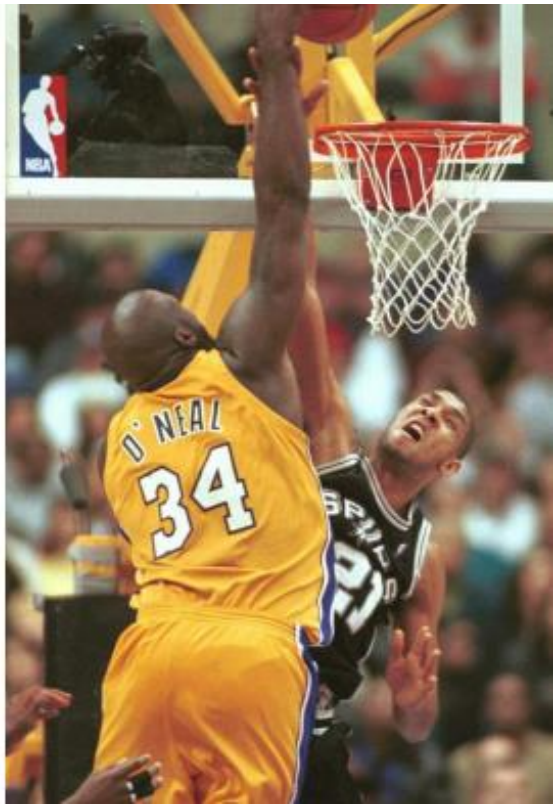
- Relation between variables where changes in some variables may "explain" or possibly "cause" changes in other variables.
- Explanatory variables are termed the independent variables and the variables to be explained are termed the dependent variables.
- Regression model estimates the nature of the relationship between the independent and dependent variables.
- Change in dependent variables that results from changes in independent variables, i.e. size of the relationship.
- Strength of the relationship.
- Statistical significance of the relationship.

HISTORY

- This all started in the 1800s with a guy named Francis Galton. Galton was studying the relationship between parents and their children. In particular, he investigated the relationship between the heights of fathers and their sons.
- What he discovered was that a man's son tended to be roughly as tall as his father. However Galton's breakthrough was that the son's height tended to be closer to the overall average height of all people

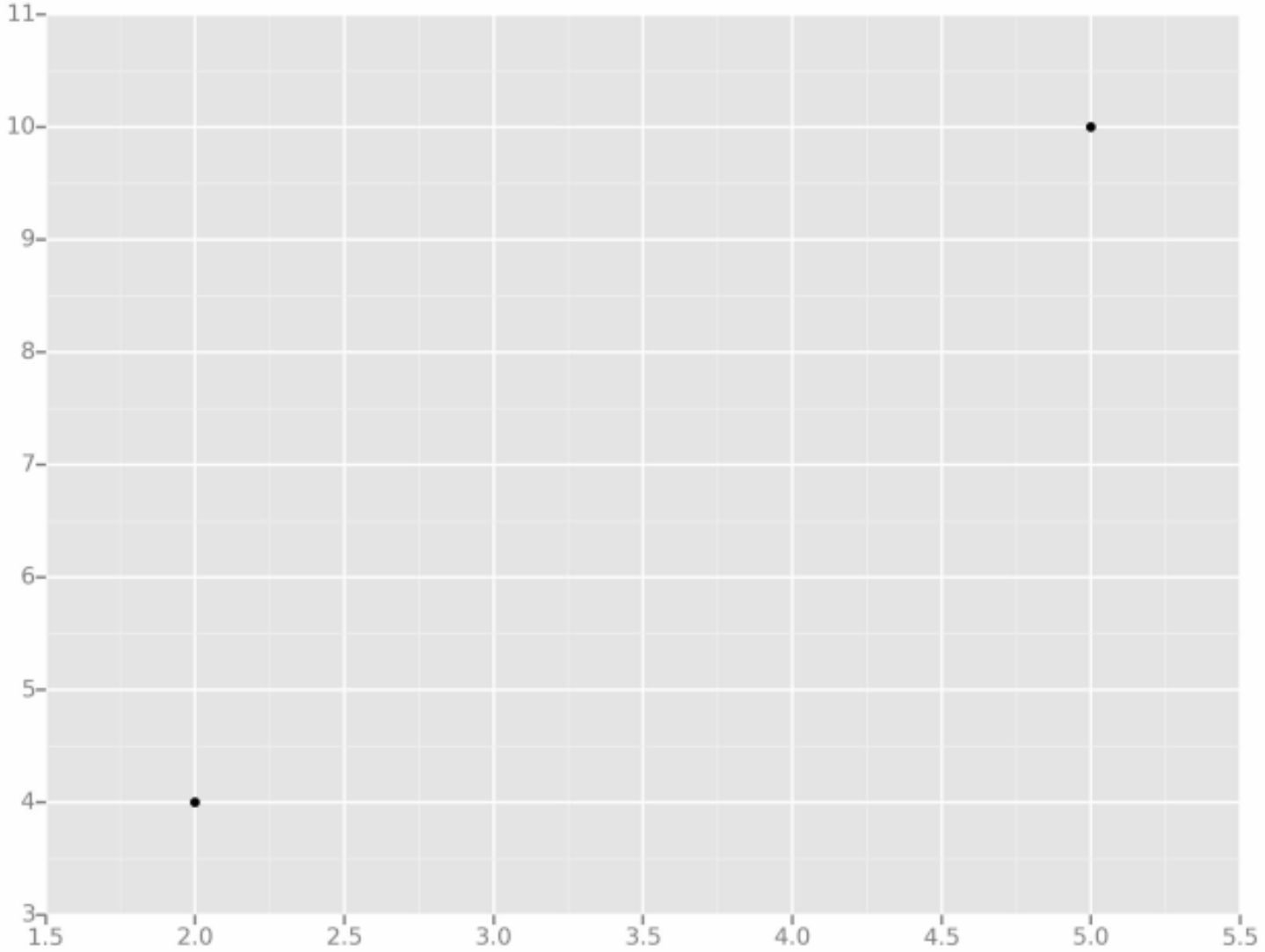
EXAMPLE

- Let's take O'neal example. He is 7ft in. He has a son with chances of him to be very tall too. With chances of his son not to be tall as well
- Although his son is also tall, around 6ft 7in, but surely not as tall as his dad.
- Galton this phenomenon as **"REGRESSION"**
- A Father's Son's height tends to regress (or drifts towards) the mean (average) height



EXAMPLE 2

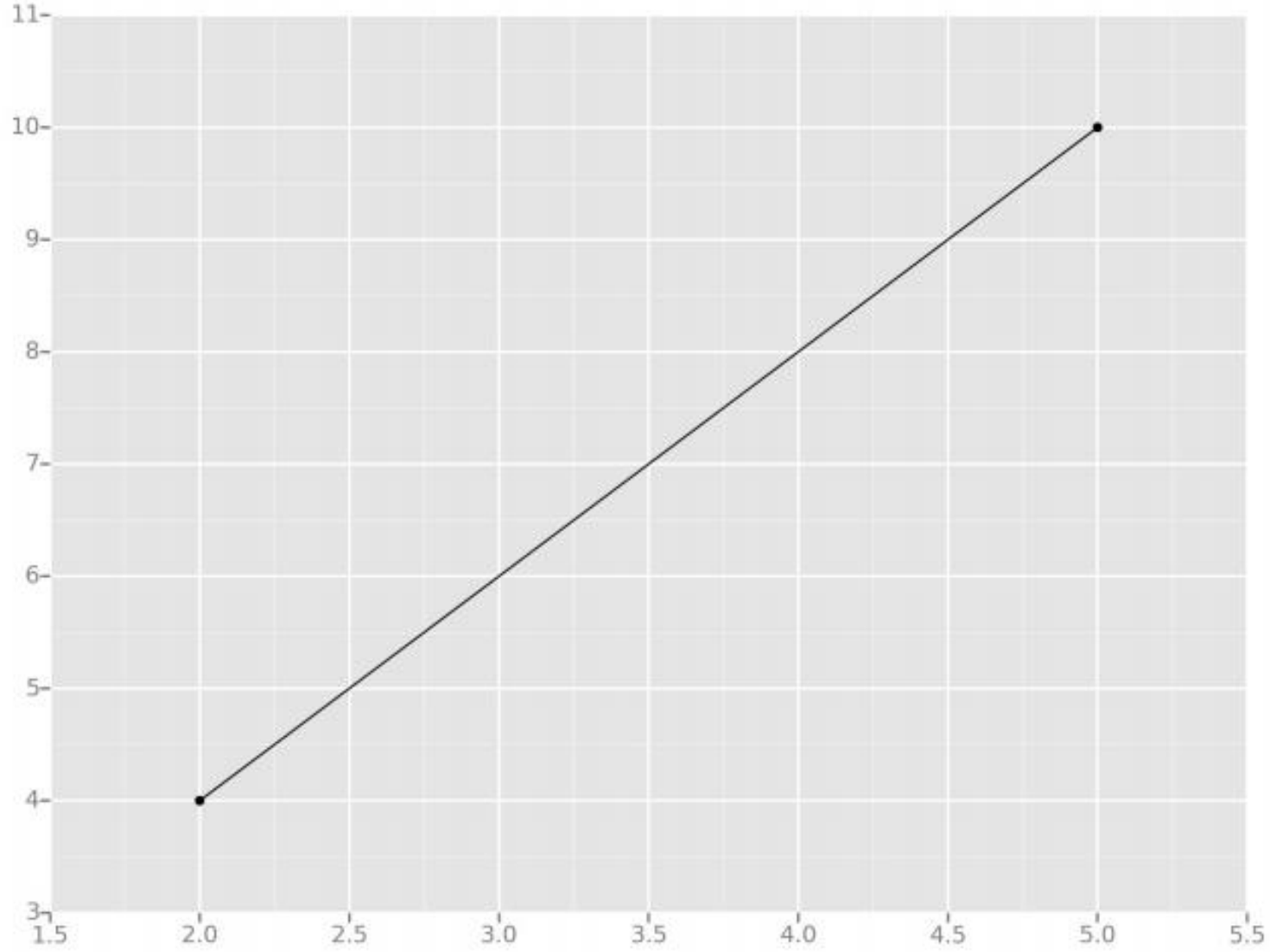
- Now consider only two data points



TechieNest
...Transforming Engineers to Technocrats

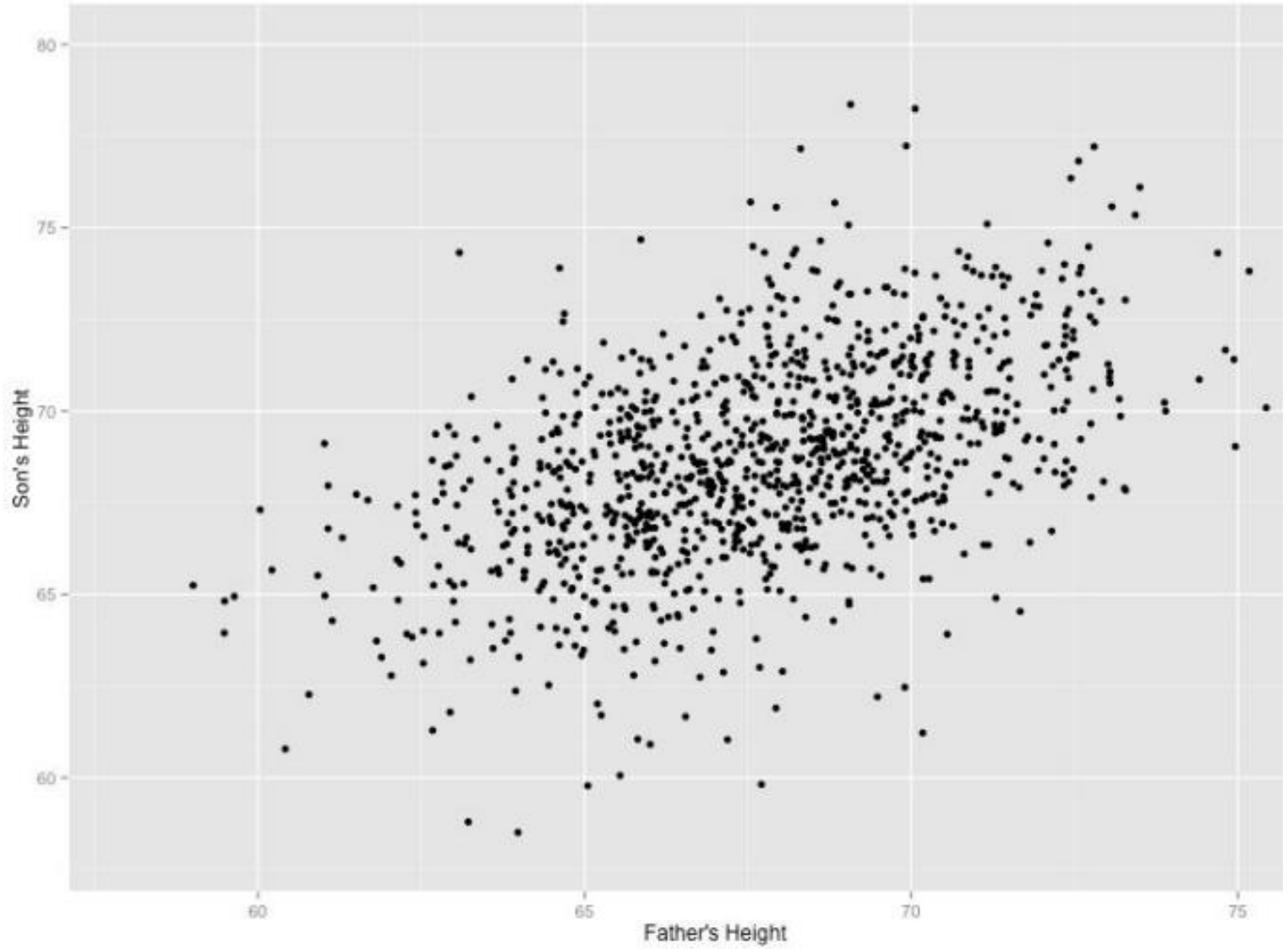
LINEAR REG

- All we're trying to do when we calculate our regression line is draw a line that's as close to every dot as possible
- For class Linear Regression, or Least Squares Method, you can only measure the closeness in the up and down direction



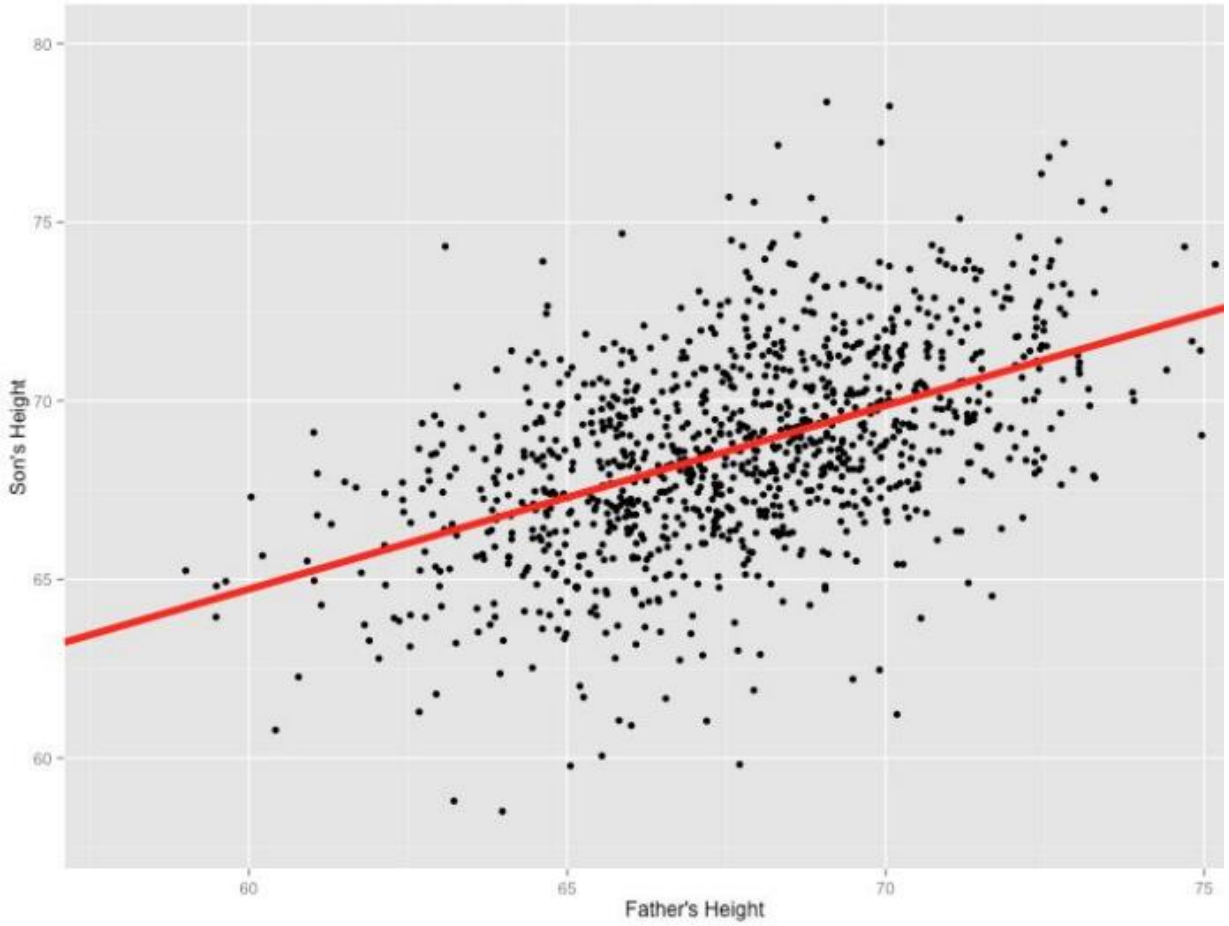
LINEAR REG

- Now applying this concept with multiple data points
- By doing this, we get multiple men and their son's heights and do things like tell a man how tall we expect his son to be, before he even has a son.



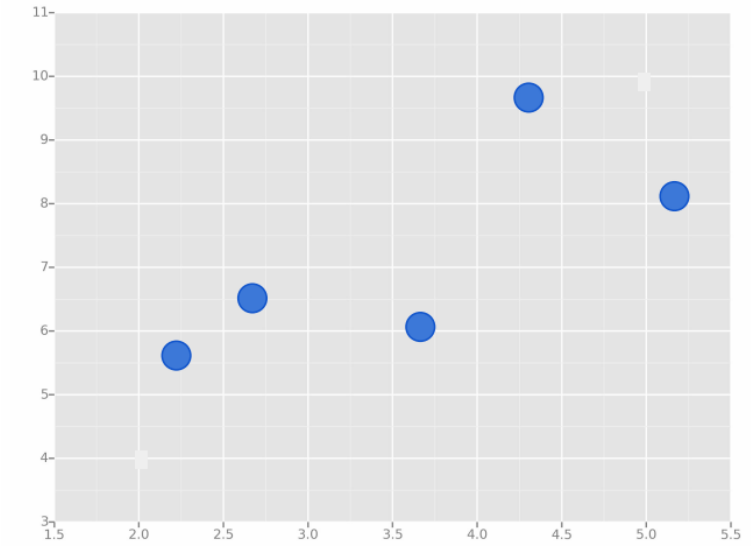
LINEAR REGRESSION

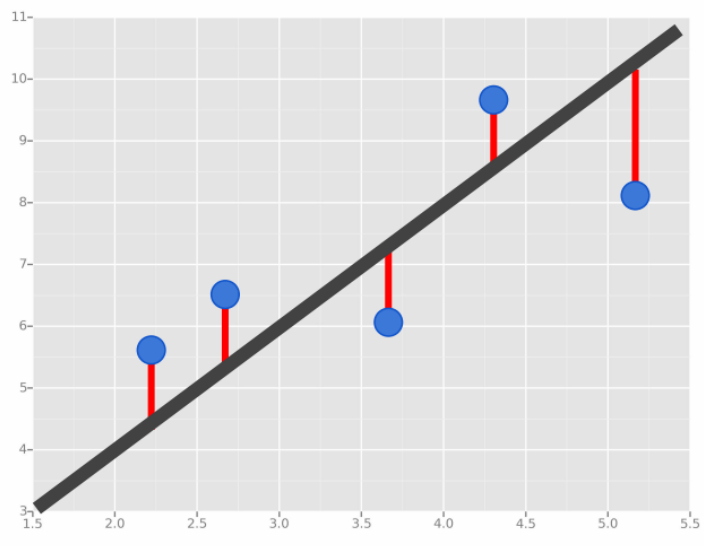
- The goal with Linear Regression is to minimize the vertical distance between all the data points and our line
- So in determining the best line, we are attempting to minimize the distance between all the points and their distance to our line by using some methods like, some of squared errors, some of absolute errors etc.



LINEAR REGRESSION

- We can also use Least Squared Error Method
- This method fits the data by Sum of Squares of Residuals (residual is difference between observed value and regression line)





STEPS

With linear regression, you are trying to reduce a set of data into a line. The equation for a line is → $y = a + bx$

Intercept: a — Slope: b

Slope: $b = r * \frac{s_y}{s_x}$

Pearson's Correlation: r — Sample Standard Deviation of y: s_y — Sample Standard Deviation of x: s_x

The first thing to calculate is the slope of the regression line. That value is the correlation of the two data sets, multiplied by the ratio of their standard deviations. Note that those are sample standard deviations, not population

Correlation shows how much two sets of data change together. Correlation is always between -1 and 1, and is unit-less. Correlation is frequently around the average x, average y, but if you want to force the line through a specific point, you can use that point instead of average x, y in all the

→ $r = \frac{\sum((x - \bar{x}) * (y - \bar{y}))}{(n - 1) * s_x * s_y}$

Pearson's Correlation: r — Sum Over All Data Points: $\sum$ — x & y values of each point minus x & y mean values: $(x - \bar{x}) * (y - \bar{y})$ — # of Data Points: n — Standard Deviation of x & y: $s_x * s_y$

TechieNest
...Transforming Engineers to Technocrats

STEPS

$$s_x = \sqrt{\frac{\sum (x - \bar{x})^2}{(n-1)}} \qquad s_y = \sqrt{\frac{\sum (y - \bar{y})^2}{(n-1)}}$$

The sample standard deviation of x and y measure how spread out the x and y values are around their mean. They have units

Once you have done those calculations you have the slope. With the slope and a point the line goes through you can calculate the intercept. If you used the average x, y before use it here again. Otherwise use the same x, y that you specified before

$$a = \bar{y} - b\bar{x}$$

Intercept, Slope, Y & X Average

When you have done the regression, one way of evaluating it's quality is R-squared. R-squared is a measure of the summed squared error in the regression vs the error if you didn't do a regression. Summed Squared error is shown below

$$R^2 = 1 - \frac{SS_{Regression}}{SS_{Total}}$$

Sum Squared Regression Error / Sum Squared Total Error

$$SS_{Total} = \sum (y_i - \bar{y})^2$$

Sum Squared Total Error; Sum Over All Data Points; Each Data Point; Mean Value; Square The Result

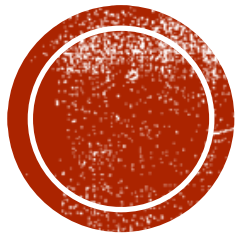
$$SS_{Regression} = \sum (y_i - y_{Regression})^2$$

Sum Squared Regression Error; Sum Over All Data Points; Each Data Point; Regression Value; Square The Result

ASSUMPTIONS OF LINEAR REGRESSION

- Linearity
- Homoscedasticity
- Multivariate Normality
- Independence of Errors
- Lack of Multicollinearity

TechieNest
...Transforming Engineers to Technocrats



PERFORMANCE EVALUATION

Regression Modelling

PERFORMANCE EVALUATION

- Mean Absolute Error

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

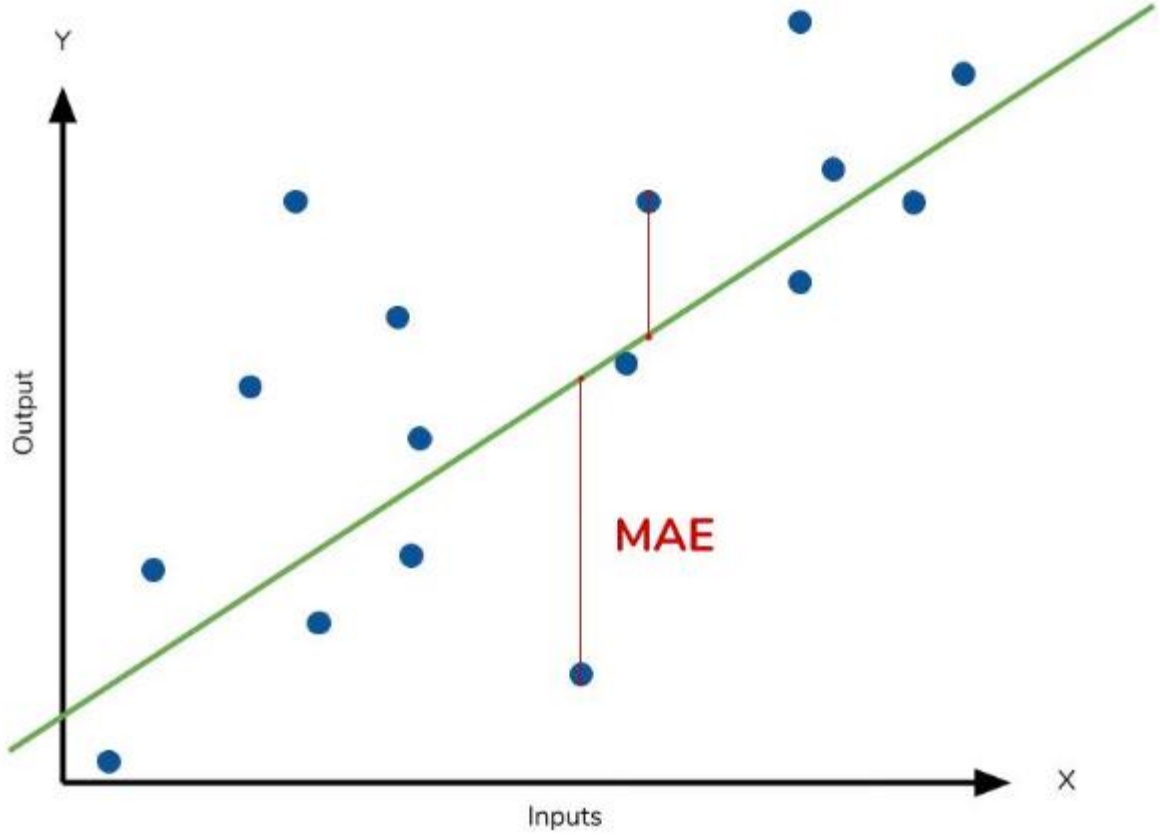
Divide by the total number of data points

Predicted output value

Actual output value

Sum of

The absolute value of the residual

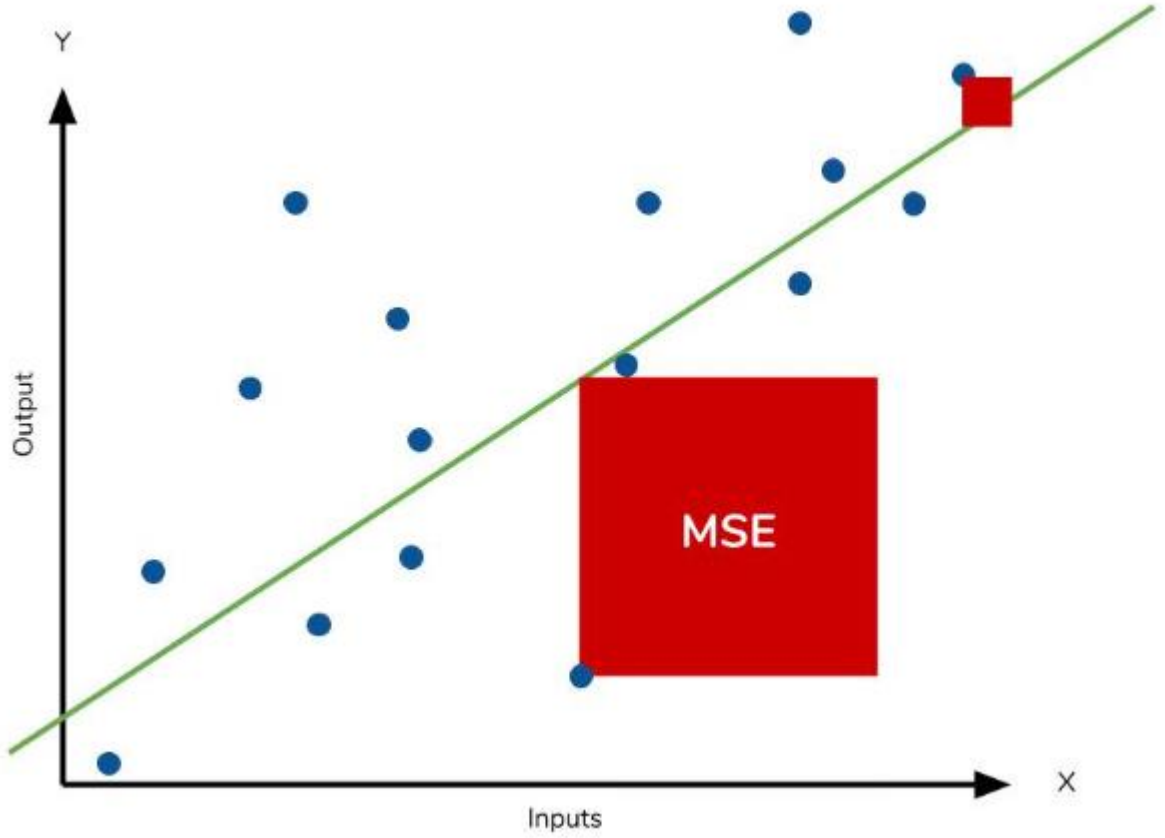


PERFORMANCE EVALUATION

- Mean Squared Error

$$MSE = \frac{1}{n}\Sigma\underbrace{\left(y - \hat{y}\right)}^{2}$$

The square of the difference between actual and predicted



PERFORMANCE EVALUATION

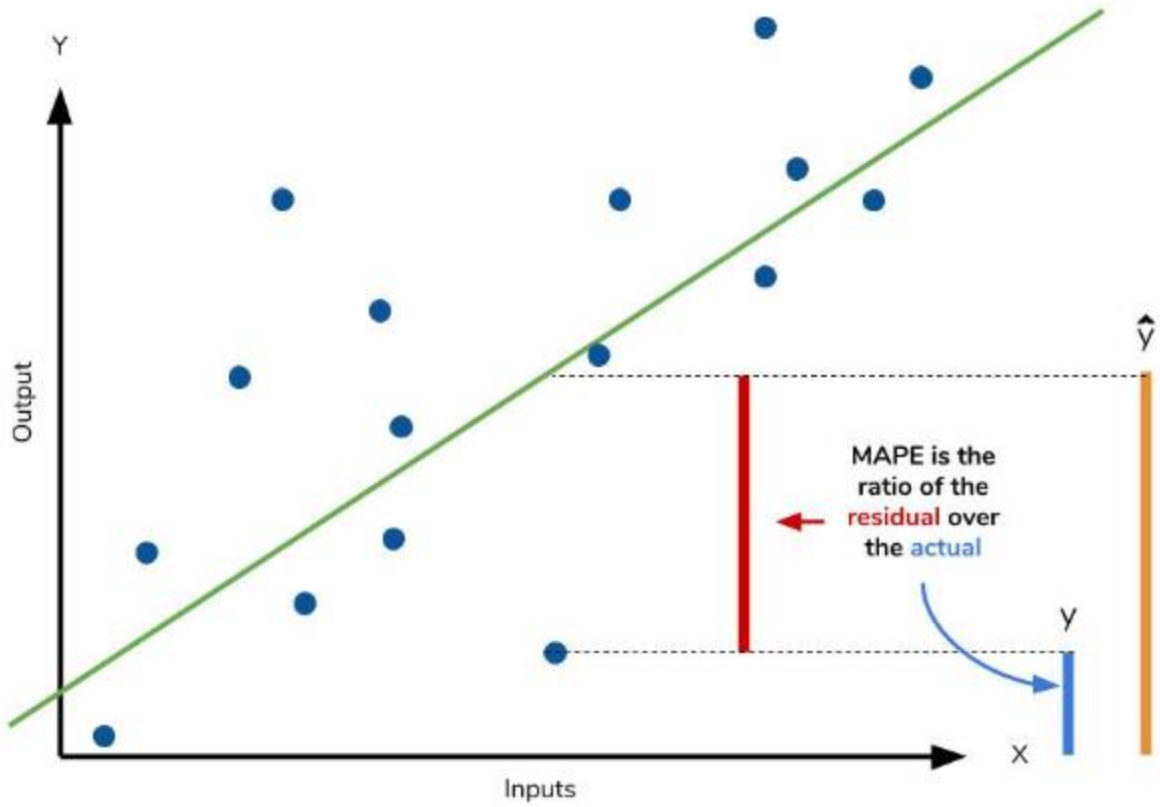
- Mean absolute percentage error

$$MAPE = \frac{100\%}{n} \sum \left| \frac{y - \hat{y}}{y} \right|$$

Multiplying by 100% converts to percentage

The residual

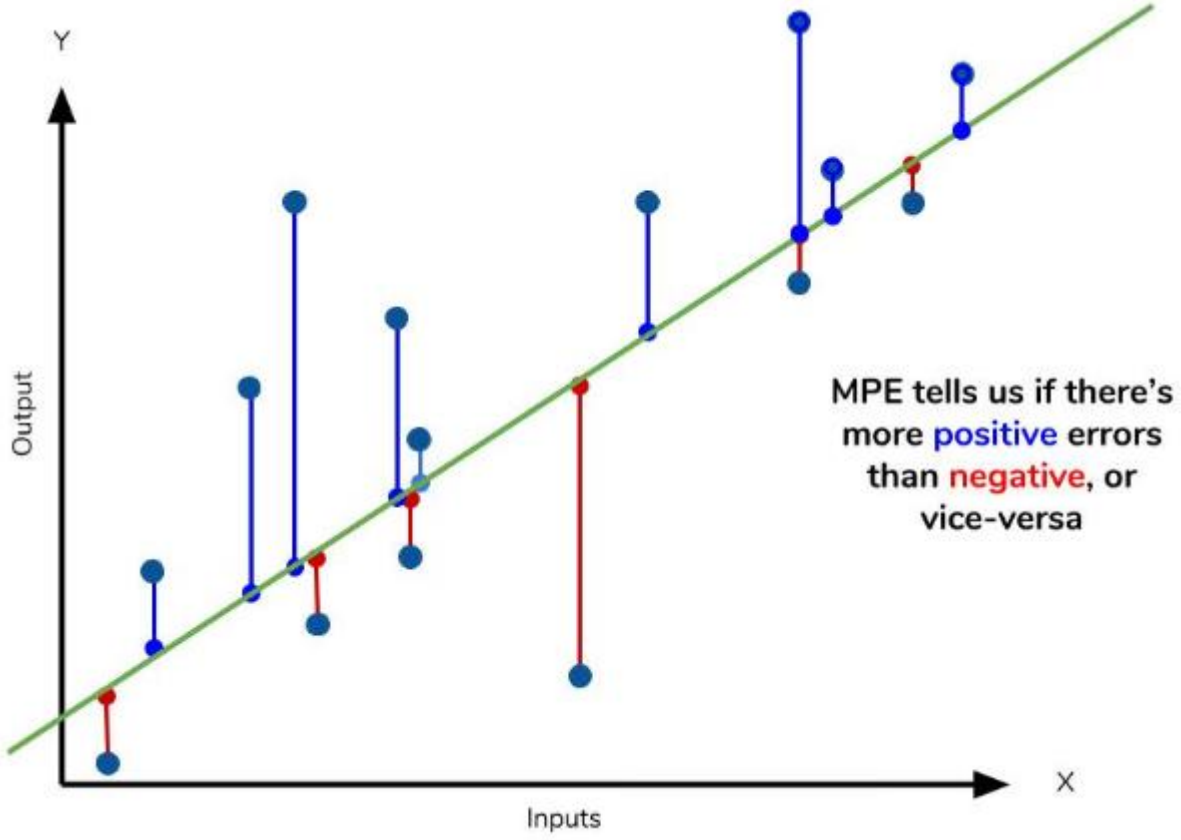
Each residual is scaled against the actual value



PERFORMANCE EVALUATION

- Mean percentage error

$$MPE = \frac{100\%}{n} \Sigma \left(\frac{y - \hat{y}}{y} \right)$$



PERFORMANCE EVALUATION

- Root Mean Squared Error

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

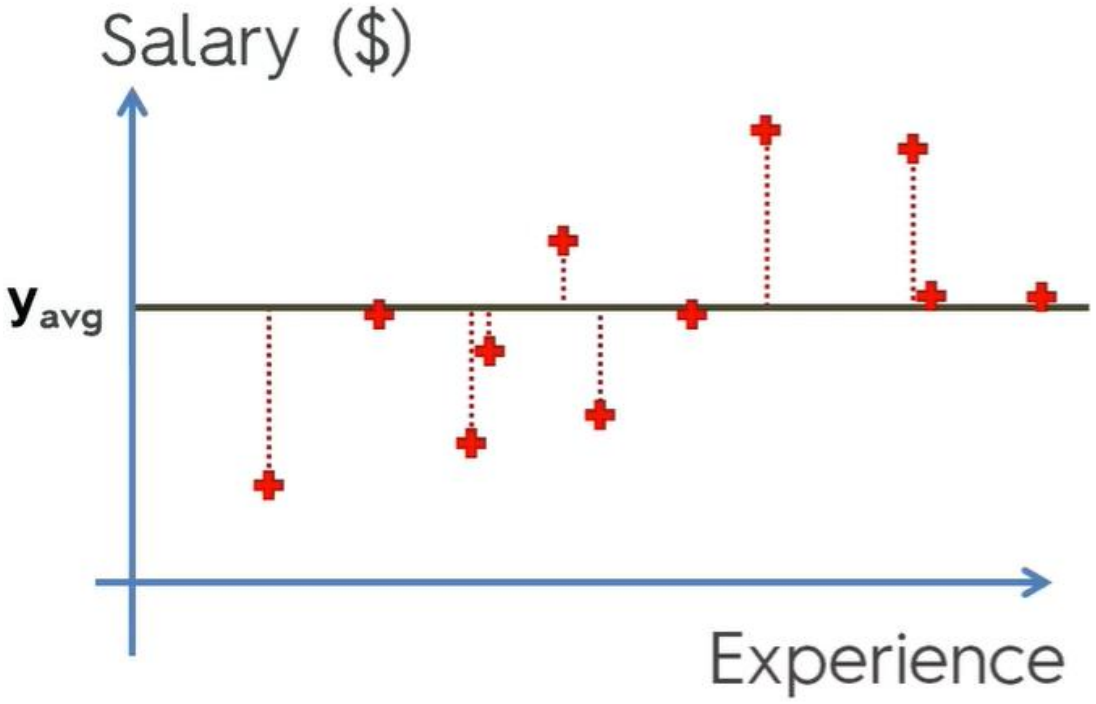
PERFORMANCE EVALUATION

- R^2 Error

$$R^2 = 1 - \frac{\text{MSE(model)}}{\text{MSE(baseline)}}$$

R SQUARED

- Simple Linear Regression



$$SS_{res} = SUM\ (y_i - \hat{y}_i)^2$$

$$SS_{tot} = SUM\ (y_i - y_{avg})^2$$

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

PERFORMANCE EVALUATION

- Adjusted R^2 Error

$$R_a^2 = 1 - \left[\left(\frac{n-1}{n-k-1}\right) \times (1 - R^2)\right]$$

where:

n = number of observations

k = number of independent variables

R_a^2 = adjusted R^2

ADJUSTED R-SQUARE

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

$$\text{Adj } R^2 = 1 - (1 - R^2)\frac{n - 1}{n - p - 1}$$

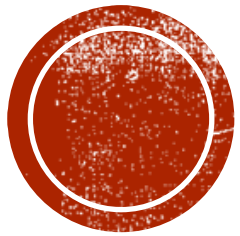
p - number of regressors

n - sample size

CONCLUSION

Acroynm	Full Name	Residual Operation?	Robust To Outliers?
MAE	Mean Absolute Error	Absolute Value	Yes
MSE	Mean Squared Error	Square	No
RMSE	Root Mean Squared Error	Square	No
MAPE	Mean Absolute Percentage Error	Absolute Value	Yes
MPE	Mean Percentage Error	N/A	Yes

TechieNest
...Transforming Engineers to Technocrats



POLYNOMIAL REGRESSION

Regression Modelling

REGRESSION

Simple Linear Regression

$$y = b_0 + b_1 x_1$$

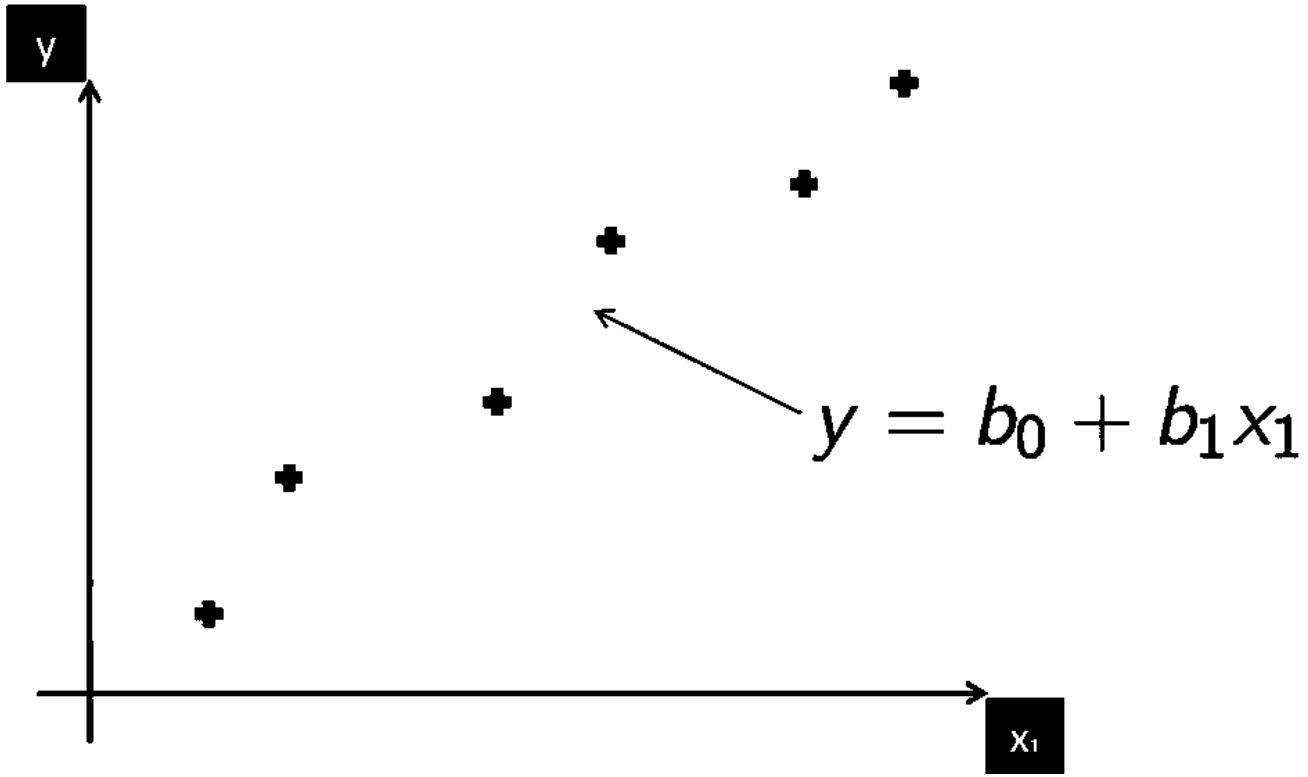
Multiple Linear Regression

$$y = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_n x_n$$

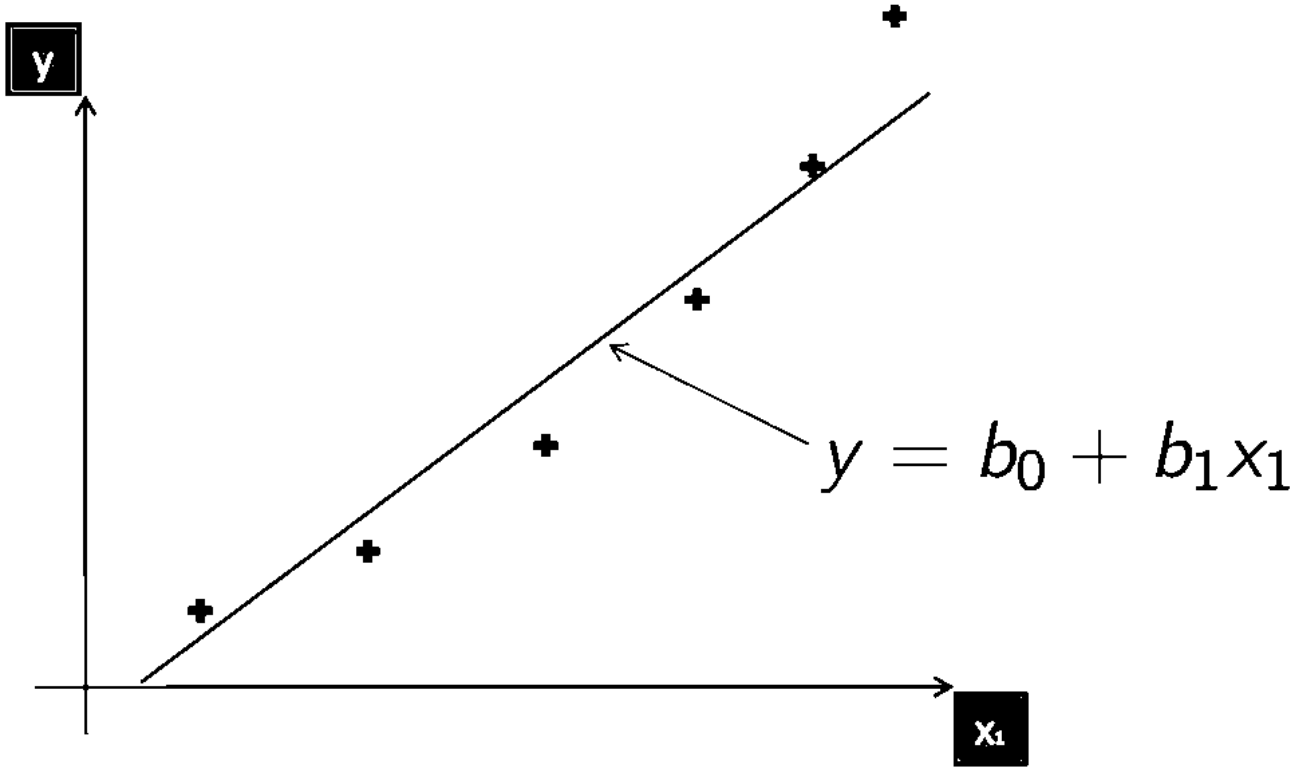
Polynomial Linear Regression

$$y = b_0 + b_1 x_1 + b_2 x_1^2 + \ldots + b_n x_1^n$$

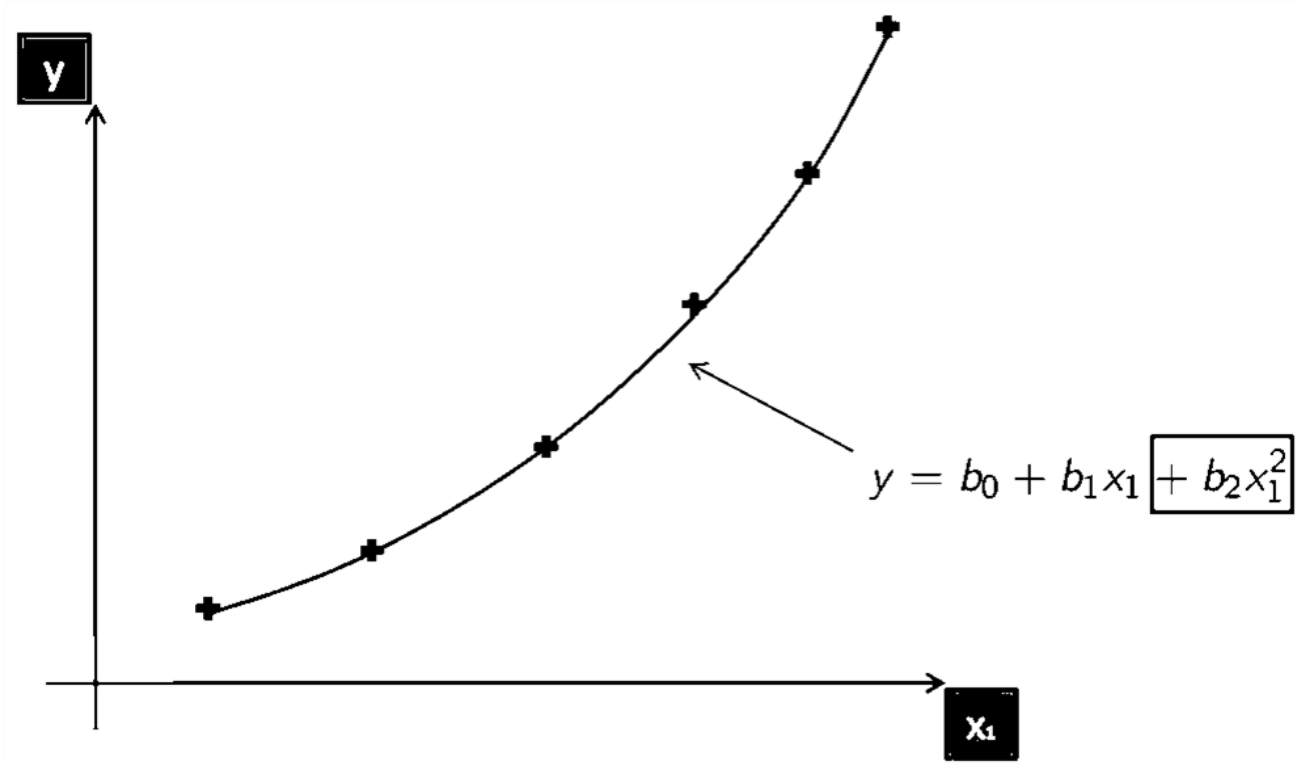
SIMPLE LINEAR REGRESSION



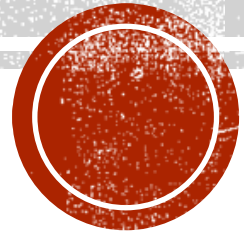
SIMPLE LINEAR REGRESSION



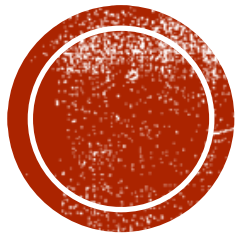
POLYNOMIAL REGRESSION



CLASSIFICATION



TechieNest
...Transforming Engineers to Technocrats



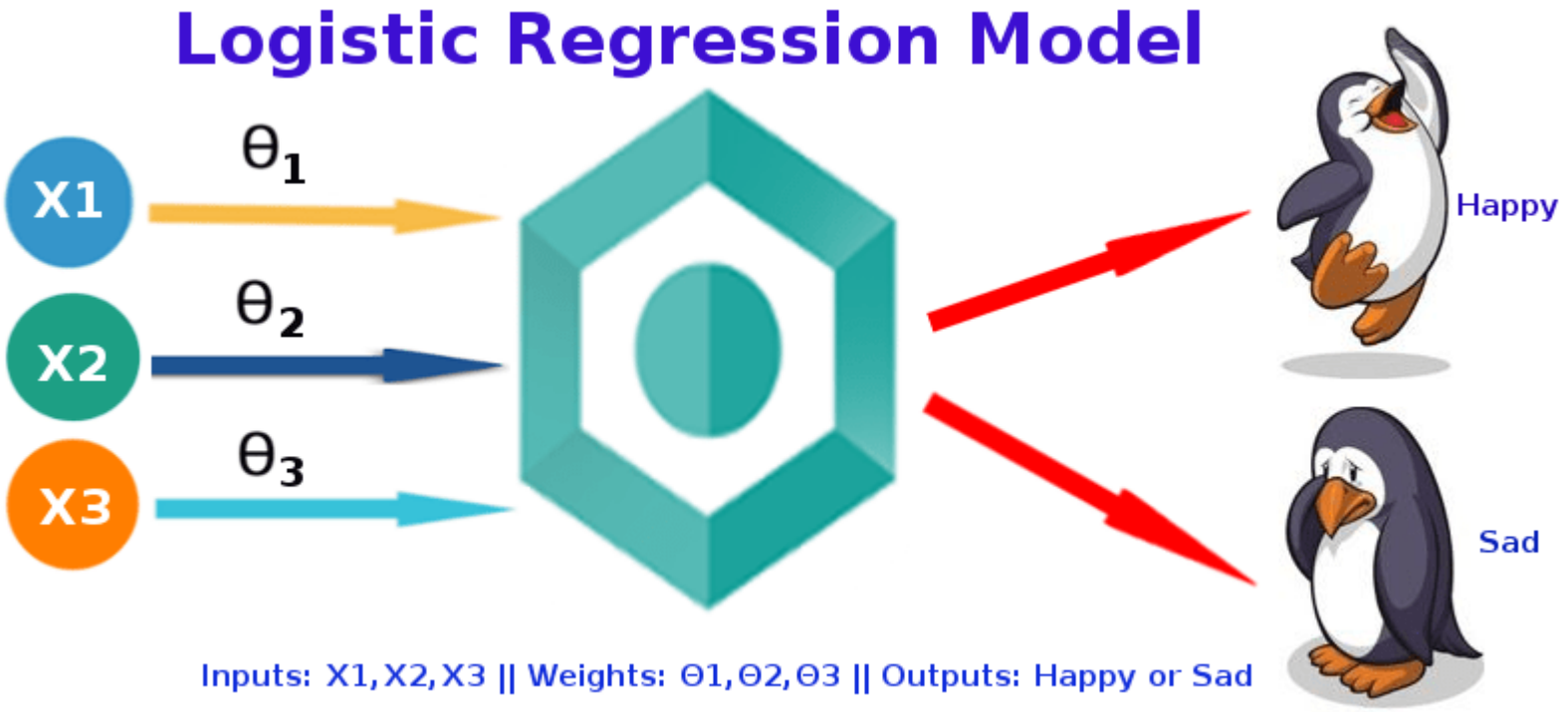
LOGISTIC REGRESSION

Classification with Regression

LOGISTIC REGRESSION

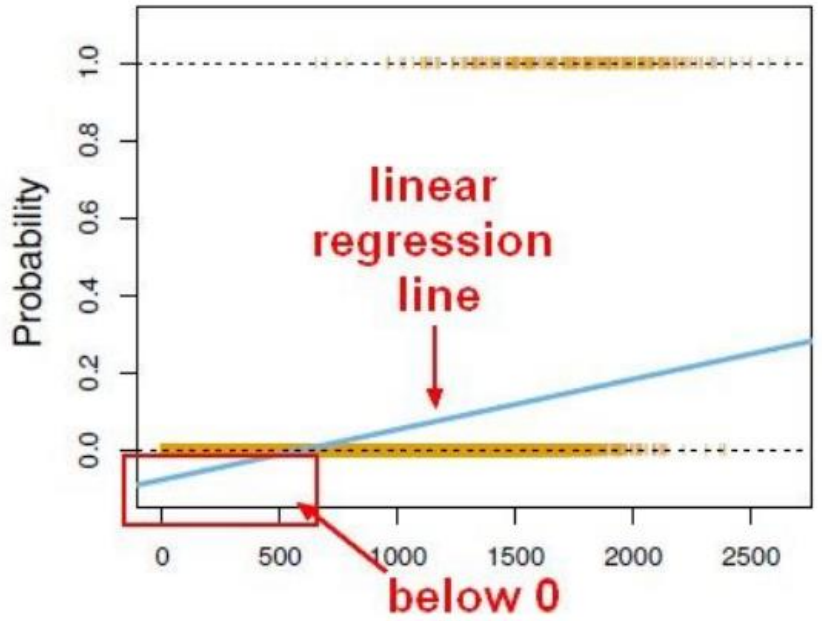
- Logistic Regression is used when the dependent variable(target) is categorical.
- For example,
  - To predict whether an email is spam (1) or (0)
  - Whether the tumor is malignant (1) or not (0)

LOGISTIC REGRESSION

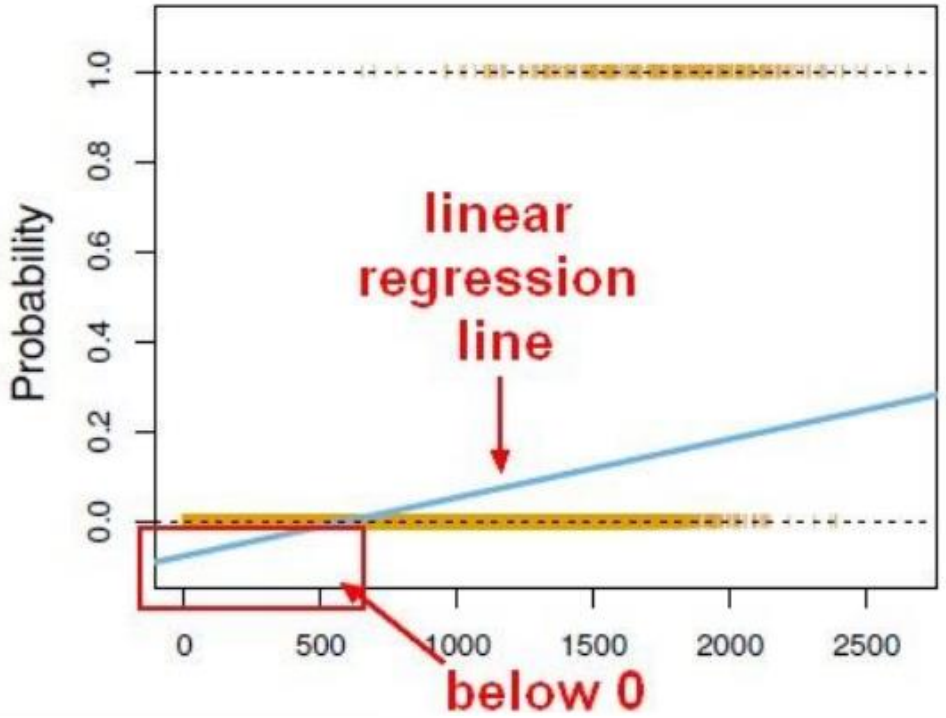


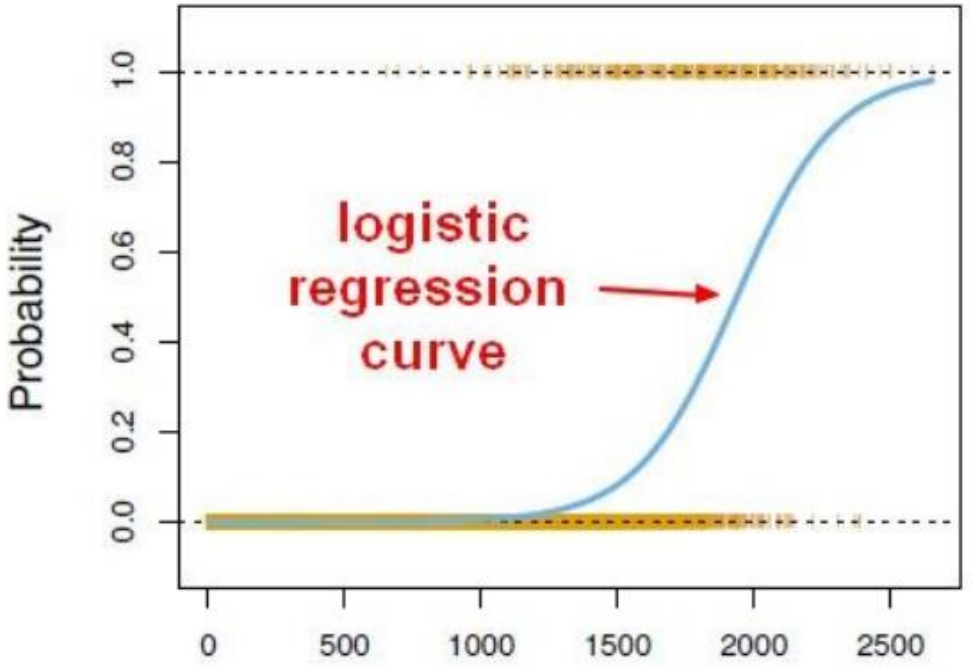
LOGISTIC REGRESSION

- We can't use a normal linear regression model on binary groups
- It won't lead to a good fit at all.



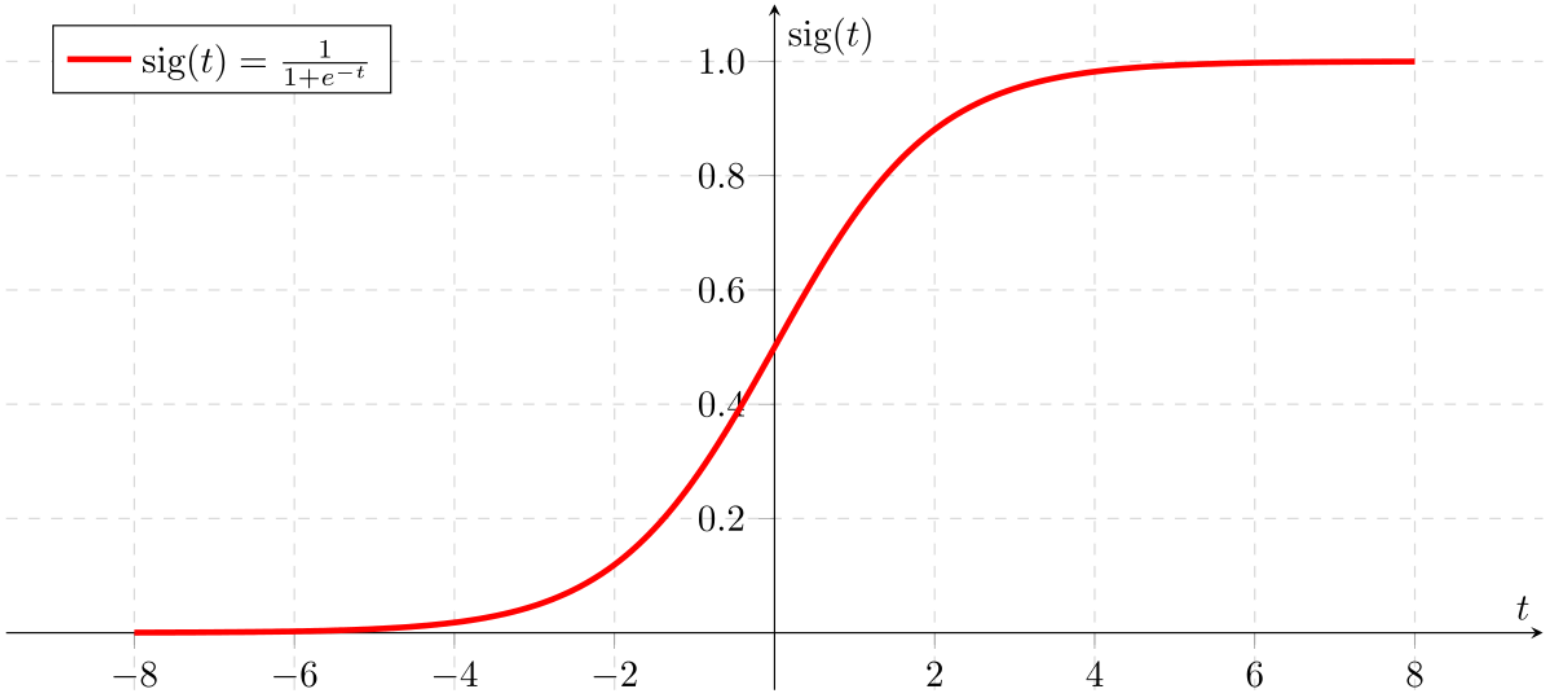
LOGISTIC REGRESSION





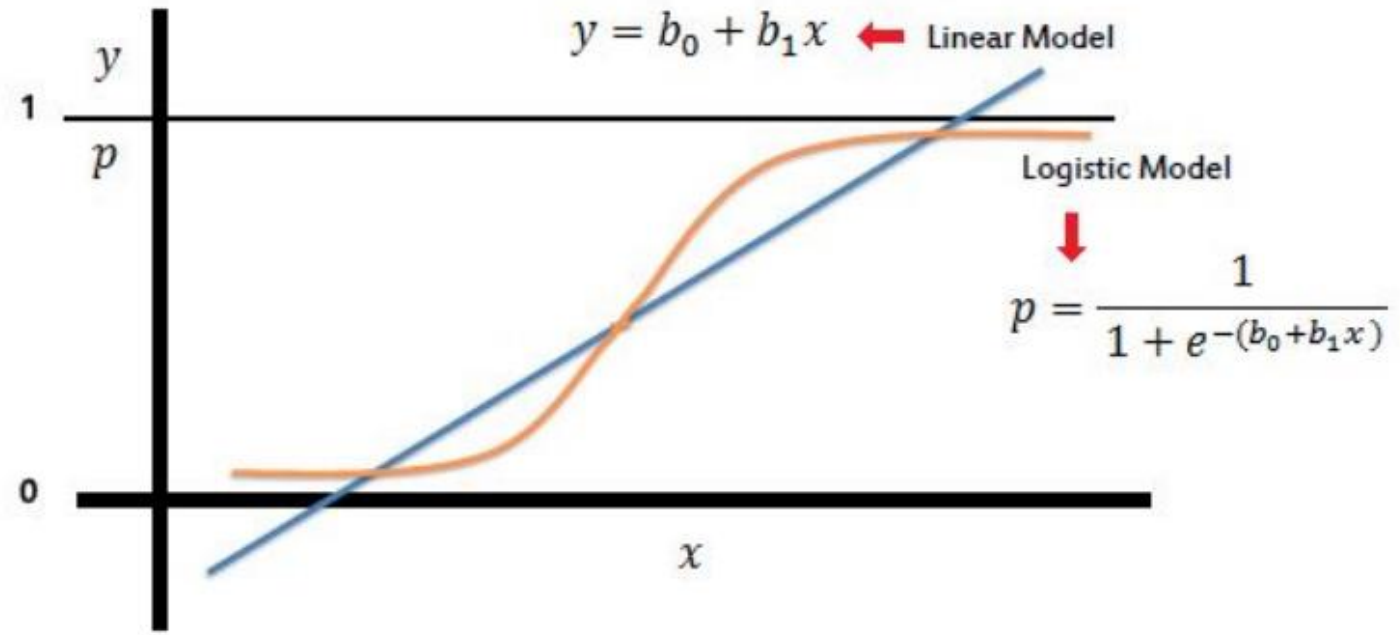
LOGISTIC REGRESSION

- This Sigmoid Function takes in any value and outputs it to be between 0 and 1



LOGISTIC REGRESSION

- This means we can take our Linear Regression Solution and place it into the sigmoid Function



TYPES OF LOGISTIC REGRESSION

- **Binary Logistic Regression**
  - The categorical response has only two 2 possible outcomes. Example: Spam or Not
- **Multinomial Logistic Regression**
  - Three or more categories without ordering. Example: Predicting which food is preferred more (Veg, Non-Veg, Vegan)
- **Ordinal Logistic Regression**
  - Three or more categories with ordering. Example: Movie rating from 1 to 5

TERMINOLOGY

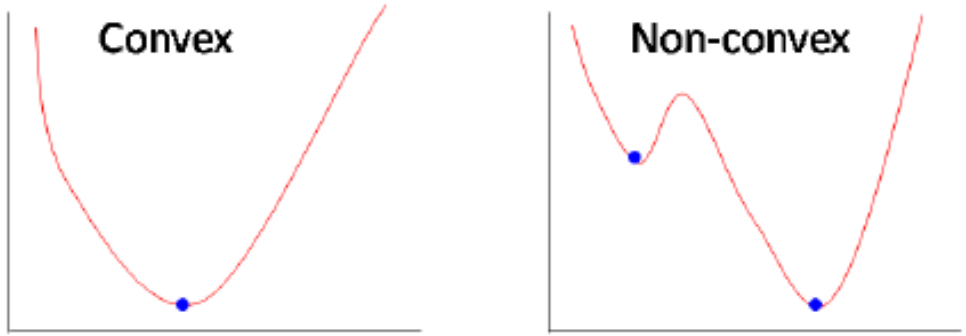
- ***Decision Boundary***
  - To predict which class a data belongs, a threshold can be set. Based upon this threshold, the obtained estimated probability is classified into classes.
  - Say, if predicted_value $\geq$ 0.5, then classify email as spam else as not spam.
  - Decision boundary can be linear or non-linear. Polynomial order can be increased to get complex decision boundary.

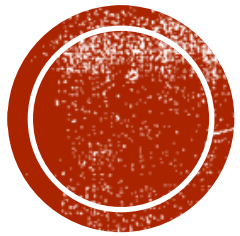
TERMINOLOGY

- **Cost Function**

$$\text{Cost}(h_\theta(x), Y(\text{actual})) = -\log(h_\theta(x)) \text{ if } y=1$$

$$-\log(1-h_\theta(x)) \text{ if } y=0$$





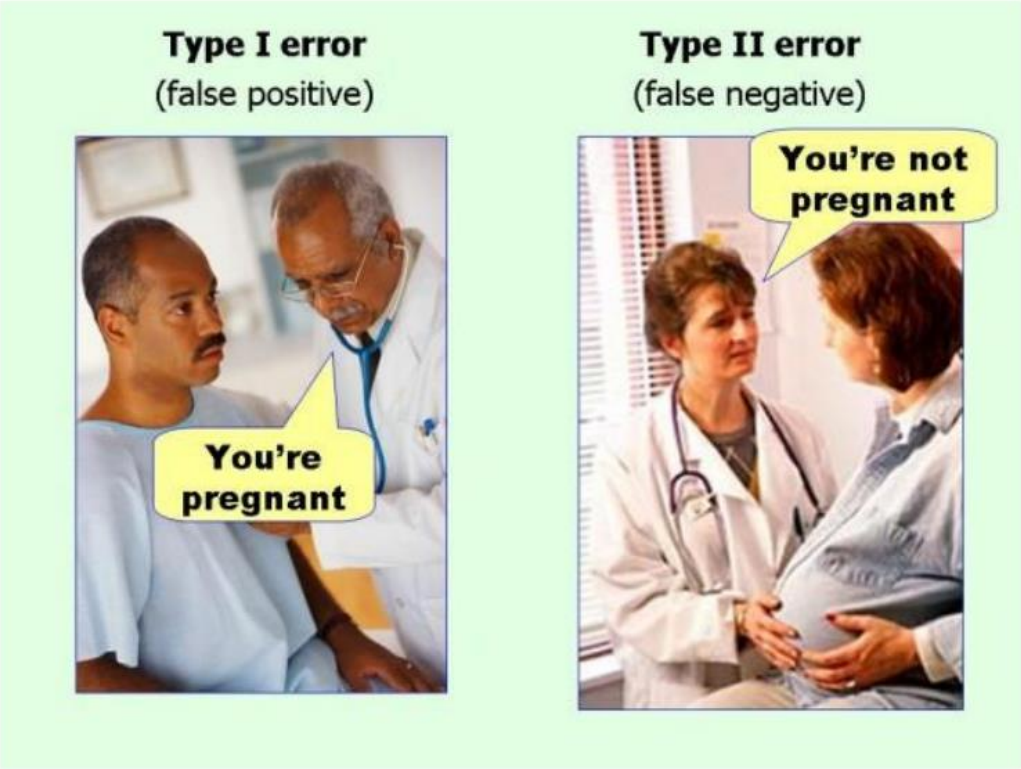
PERFORMANCE EVALUATION

Classification Modelling

PERFORMANCE EVALUATION

- Confusion Matrix
- Precision
- Recall
- F1-Score
- Model Score

HOW ERRORS CAN BE DIFFERENT



TechieNest
...Transforming Engineers to Technocrats

CONFUSION MATRIX

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

		predicted class	
		0	1
true class	0	True Positive (TP)	False Negative (FN)
	1	False Positive (FP)	True Negative (TN)

Basic Terminology:

- True Positives (TP)
- True Negatives (TN)
- False Positives (FP)
- False Negatives (FN)

		True condition			
	Total population	Condition positive	Condition negative	Prevalence = $\frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$	Accuracy (ACC) = $\frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$
Predicted condition	Predicted condition positive	**True positive**	**False positive**, Type I error	Positive predictive value (PPV), Precision = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Predicted condition positive}}$	False discovery rate (FDR) = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Predicted condition positive}}$
	Predicted condition negative	**False negative**, Type II error	**True negative**	False omission rate (FOR) = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Predicted condition negative}}$	Negative predictive value (NPV) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Predicted condition negative}}$
		True positive rate (TPR), Recall, Sensitivity, probability of detection, Power = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$	Positive likelihood ratio (LR+) = $\frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) = $\frac{\text{LR+}}{\text{LR−}}$	F_1 score = $2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$
		False negative rate (FNR), Miss rate = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$	Specificity (SPC), Selectivity, True negative rate (TNR) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$	Negative likelihood ratio (LR−) = $\frac{\text{FNR}}{\text{TNR}}$		

PERFORMANCE EVALUATION

- sensitivity, recall, hit rate, or true positive rate (TPR) $\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{P}} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = 1 - \mathrm{FNR}$
- specificity, selectivity or true negative rate (TNR) $\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{N}} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} = 1 - \mathrm{FPR}$
- precision or positive predictive value (PPV) $\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} = 1 - \mathrm{FDR}$
- negative predictive value (NPV) $\mathrm{NPV} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FN}} = 1 - \mathrm{FOR}$

PERFORMANCE EVALUATION

- miss rate or false negative rate (FNR) $$\mathrm{FNR} = \frac{\mathrm{FN}}{\mathrm{P}} = \frac{\mathrm{FN}}{\mathrm{FN} + \mathrm{TP}} = 1 - \mathrm{TPR}$$
- fall-out or false positive rate (FPR) $$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{N}} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} = 1 - \mathrm{TNR}$$
- false discovery rate (FDR) $$\mathrm{FDR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TP}} = 1 - \mathrm{PPV}$$
- false omission rate (FOR) $$\mathrm{FOR} = \frac{\mathrm{FN}}{\mathrm{FN} + \mathrm{TN}} = 1 - \mathrm{NPV}$$

PERFORMANCE EVALUATION

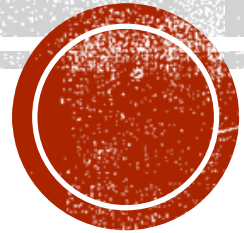
- Threat score (TS) or Critical Success Index (CSI) $$\mathrm{TS} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP}}$$
- accuracy (ACC) $$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{P} + \mathrm{N}} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$
- balanced accuracy (BA) $$\mathrm{BA} = \frac{TPR + TNR}{2}$$
- F1 score is the harmonic mean of precision and sensitivity

$$\mathrm{F_1} = 2 \cdot \frac{\mathrm{PPV} \cdot \mathrm{TPR}}{\mathrm{PPV} + \mathrm{TPR}} = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

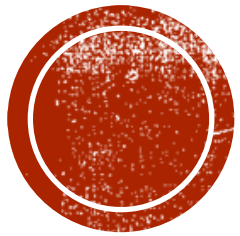
PERFORMANCE EVALUATION

- Matthews correlation coefficient (MCC) $$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}}$$
- informedness or bookmaker informedness (BM) $$\mathrm{BM} = \mathrm{TPR} + \mathrm{TNR} - 1$$
- markedness (MK) or deltaP $$\mathrm{MK} = \mathrm{PPV} + \mathrm{NPV} - 1$$

UNSUPERVISED LEARNING



TechieNest
...Transforming Engineers to Technocrats



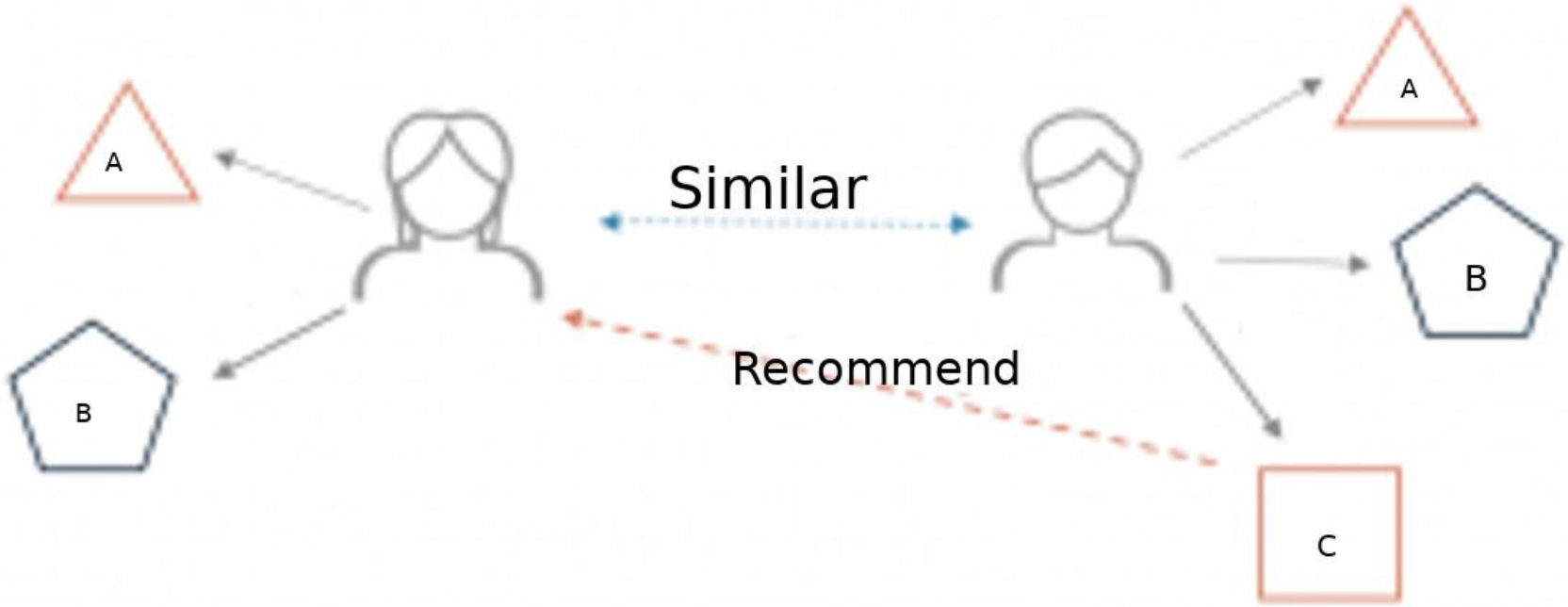
RECOMMENDATION SYSTEMS

Clustering

RECOMMENDATION SYSTEM

- A recommender system is a simple algorithm whose aim is to provide the most relevant information to a user by discovering patterns in a dataset.
- The algorithm rates the items and shows the user the items that they would rate highly.
- An example of recommendation in action is when you visit Amazon and you notice that some items are being recommended to you or when Netflix recommends certain movies to you. They are also used by Music streaming applications such as Spotify and Deezer to recommend music that you might like

RECOMMENDATION



SIMPLEST APPROACH

$$W = \frac{Rv + Cm}{v + m}$$

where:

W = Weighted Rating

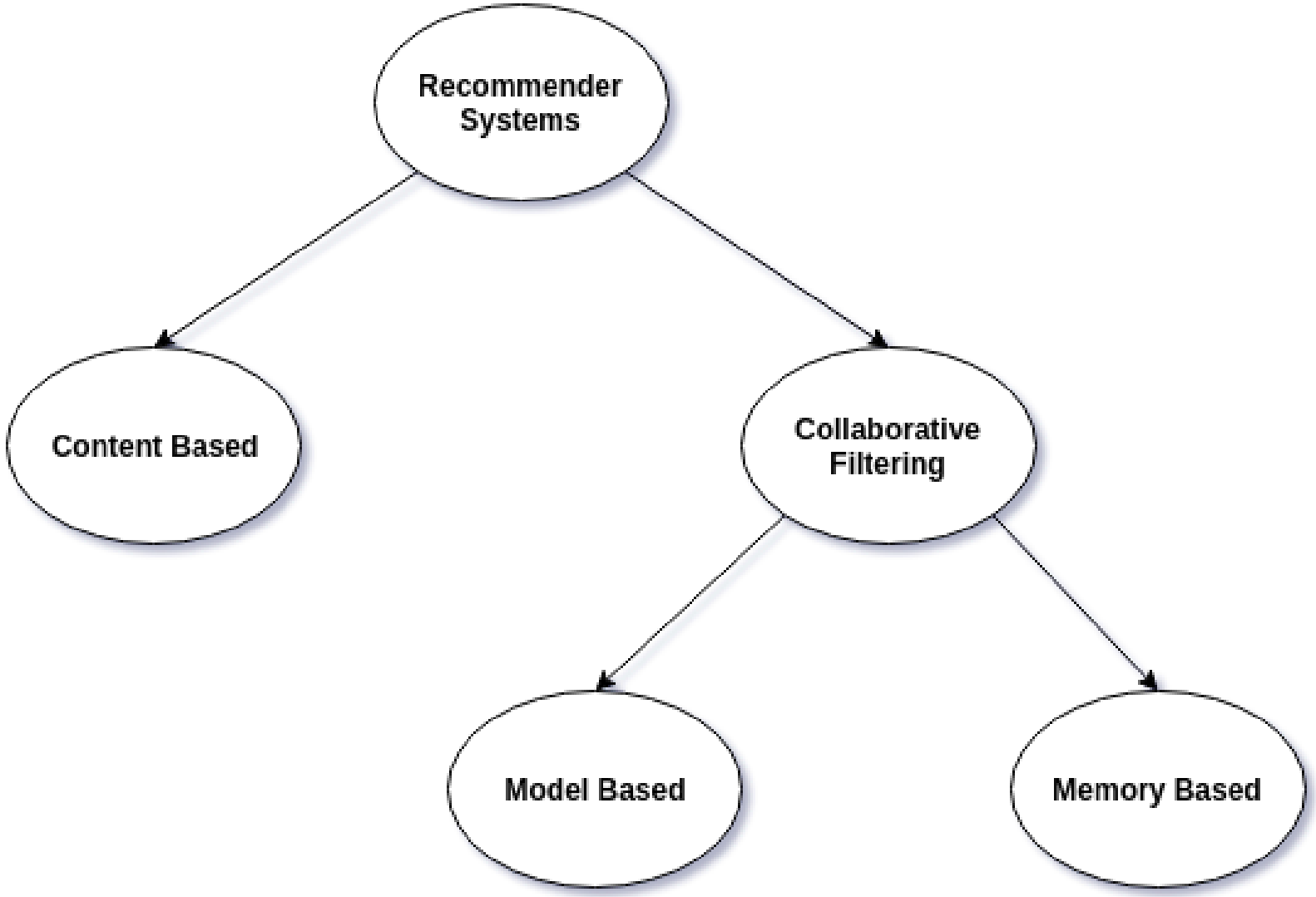
R = average for the movie as a number from 0 to 10 (mean) = (Rating)

v = number of votes for the movie = (votes)

m = minimum votes required to be listed in the Top 250 (currently 3000)

C = the mean vote across the whole report (currently 6.9)

TYPES


Recommender Systems
Content Based
Collaborative Filtering
Model Based
Memory Based

CONTENT BASED SYSTEMS

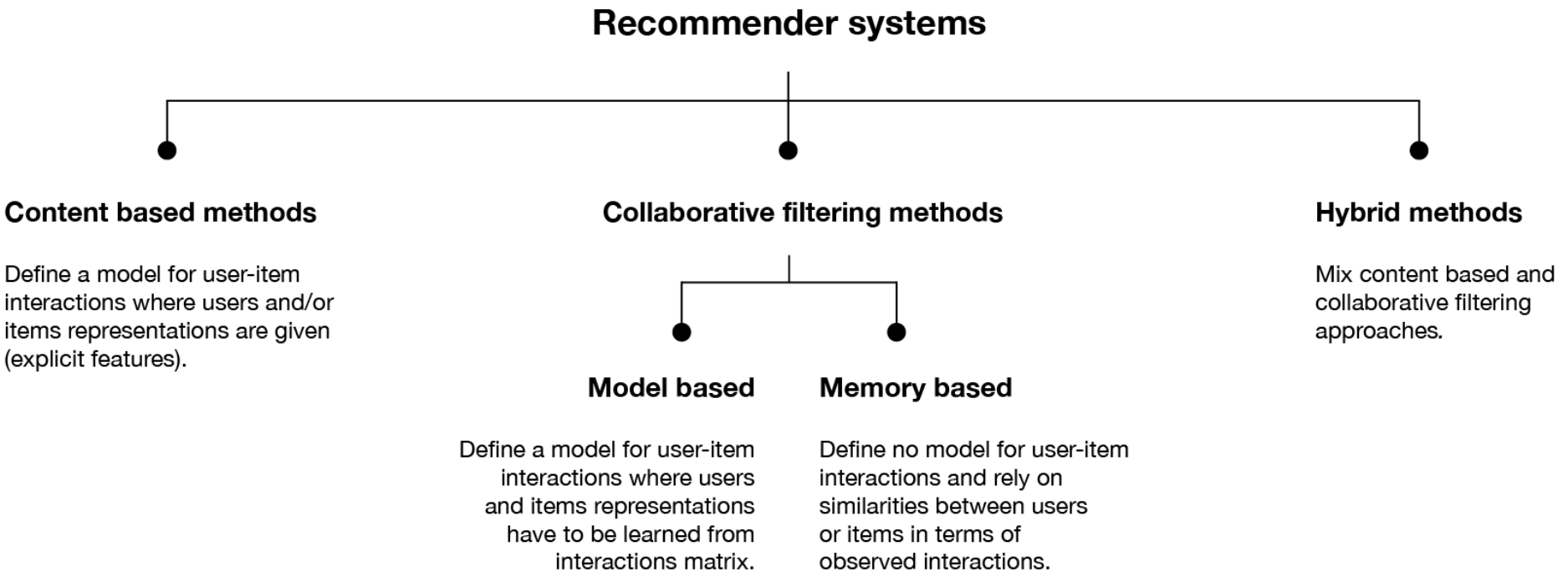
- They use meta data such as genre, producer, actor, musician to recommend items say movies or music.
- Such a recommendation would be for instance recommending Infinity War that featured Vin Disiel because someone watched and liked The Fast and the Furious.
- Similarly you can get music recommendations from certain artists because you liked their music.
- Content based systems are based on the idea that if you liked a certain item you are most likely to like something that is similar to it.

COLLABORATIVE FILTERING SYSTEMS

- The behavior of a group of users is used to make recommendations to other users. Recommendation is based on the preference of other users.
- A simple example would be recommending a movie to a user based on the fact that their friend liked the movie.
- There are two types of collaborative models
  - **Memory-based methods** - They are simple to implement and the resulting recommendations are often easy to explain. They are divided into two:
    - **User-based collaborative filtering**: In this model products are recommended to a user based on the fact that the products have been liked by users similar to the user. For example if Derrick and Dennis like the same movies and a new movie comes out that Derick likes, then we can recommend that movie to Dennis because Derrick and Dennis seem to like the same movies.
    - **Item-based collaborative filtering:** These systems identify similar items based on users' previous ratings. For example if users A,B and C gave a 5 star rating to books X and Y then when a user D buys book Y they also get a recommendation to purchase book X because the system identifies book X and Y as similar based on the ratings of users A,B and C.
  - **Model-based methods**
    - Model-based methods are based on matrix factorization and are better at dealing with sparsity. They are developed using data mining, machine learning algorithms to predict users' rating of unrated items. In this approach techniques such as dimensionality reduction are used to improve the accuracy. Examples of such model-based methods include decision trees, rule-based models, Bayesian methods and latent factor models.

TechieNest
...Transforming Engineers to Technocrats

SUMMARY



MEMORY BASED CF

- item-item CF: "Users who liked this item also liked..."
- user-item CF: "Users who are similar to you also like..."

MEMORY BASED CF - STEPS

- In both cases, we create a user-item matrix which built from the entire dataset. Since we have to split the data into testing and training, we will need to create two matrices (all users by all movies)
- After building the user-item matrix, we calculate the similarity and create a similarity matrix
- The similarity values between items in item-item CF are measured by observing all the users who have rated both items
- For User-Item CF the similarity values between users are measured by observing all the items that are rated by both users.
- A distance metric commonly used in Recommender systems is cosine similarity, where the ratings are seen as vectors in n-dimensional space and the similarity is calculated based on the angle between these vectors

MEMORY BASED CF - STEPS

- Next step is to make predictions. You have already created similarity matrices: user_similarity and item_similarity and therefore you can make a prediction by applying following formula for user-based CF:

$$\hat{x}_{k,m} = \bar{x}_k + \frac{\sum_{u_a} sim_u(u_k, u_a)(x_{a,m} - \bar{x_{u_a}})}{\sum_{u_a} |sim_u(u_k, u_a)|}$$

MEMORY BASED CF - STEPS

- You can look at the similarity between users *k* and *a* as weights that are multiplied by the ratings of a similar user *a* (corrected for the average rating of that user). You will need to normalize it so that the ratings stay between 1 and 5 and, as a final step, sum the average ratings for the user that you are trying to predict.
- The idea here is that some users may tend always to give high or low ratings to all movies. The relative difference in the ratings that these users give is more important than the absolute values. To give an example: suppose, user *k* gives 4 stars to his favourite movies and 3 stars to all other good movies. Suppose now that another user *t* rates movies that he/she likes with 5 stars, and the movies he/she fell asleep over with 3 stars. These two users could have a very similar taste but treat the rating system differently.
- When making a prediction for item-based CF you don't need to correct for users average rating since query user itself is used to do predictions.

MODEL BASED CF

- Model-based Collaborative Filtering is based on matrix factorization (MF) which has received greater exposure, mainly as an unsupervised learning method for latent variable decomposition and dimensionality reduction.
- Matrix factorization is widely used for recommender systems where it can deal better with scalability and sparsity than Memory-based CF.
- The goal of MF is to learn the latent preferences of users and the latent attributes of items from known ratings (learn features that describe the characteristics of ratings) to then predict the unknown ratings through the dot product of the latent features of users and items.
- When you have a very sparse matrix, with a lot of dimensions, by doing matrix factorization you can restructure the user-item matrix into low-rank structure, and you can represent the matrix by the multiplication of two low-rank matrices, where the rows contain the latent vector.
- You fit this matrix to approximate your original matrix, as closely as possible, by multiplying the low-rank matrices together, which fills in the entries missing in the original matrix

MODEL BASED CF – EXAMPLE

- To give an example of the learned latent preferences of the users and items: let's say for the MovieLens dataset you have the following information: (user id, age, location, gender, movie id, director, actor, language, year, rating).
- By applying matrix factorization the model learns that important user features are age group (under 10, 10-18, 18-30, 30-90), location and gender, and for movie features it learns that decade, director and actor are most important.
- Now if you look into the information you have stored, there is no such feature as the decade, but the model can learn on its own.
- The important aspect is that the CF model only uses data (user_id, movie_id, rating) to learn the latent features.
- If there is little data available model-based CF model will predict poorly, since it will be more difficult to learn the latent features.

MEMORY VERSUS MODEL BASED CF

- Memory-based algorithms are easy to implement and produce reasonable prediction quality.
- The drawback of memory-based CF is that it doesn't scale to real-world scenarios and doesn't address the well-known cold-start problem, that is when new user or new item enters the system.
- Model-based CF methods are scalable and can deal with higher sparsity level than memory-based models, but also suffer when new users or items that don't have any ratings enter the system.

HYBRID MODELS

- Models that use both ratings and content features are called Hybrid Recommender Systems where both Collaborative Filtering and Content-based Models are combined.
- Hybrid recommender systems usually show higher accuracy than Collaborative Filtering or Content-based Models on their own: they are capable to address the cold-start problem better since if you don't have any ratings for a user or an item you could use the metadata from the user or item to make a prediction.