beingdatum
the data society

Machine Learning

BeingDatum.com
contact@beingdatum.com

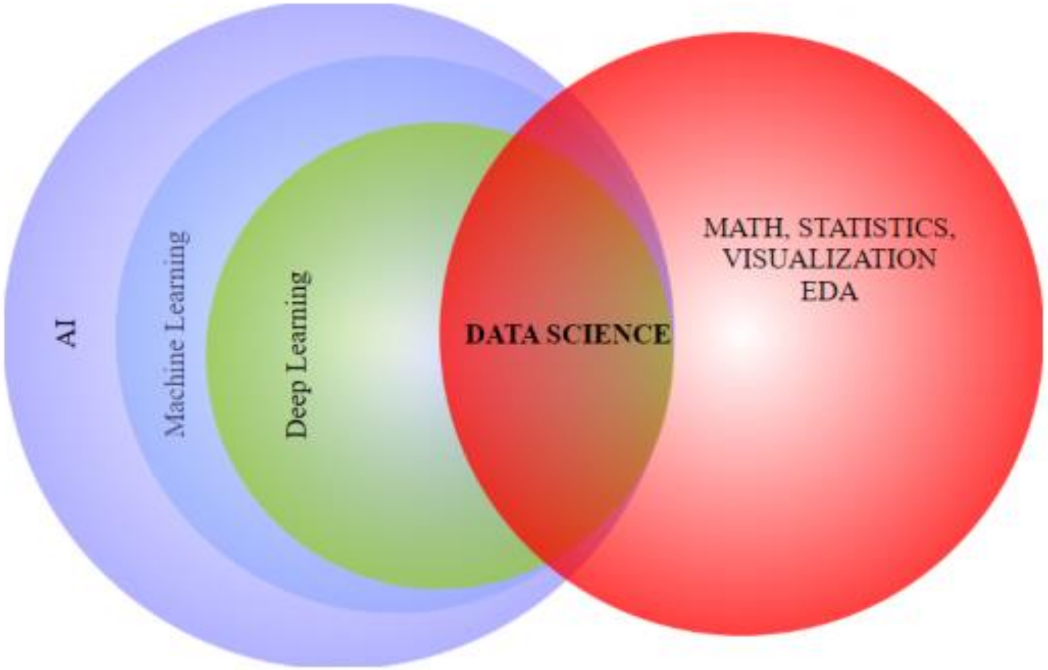
Agenda

- Introduction
- Use Cases
- Essential libraries
- Types of Learning
- Feature Scaling
- Classification Algos
- Regression Algos
- Clustering Algos
- Association Rule Learning
- Ensemble Techniques
- Time Series Analysis
- Dimensionality Reduction

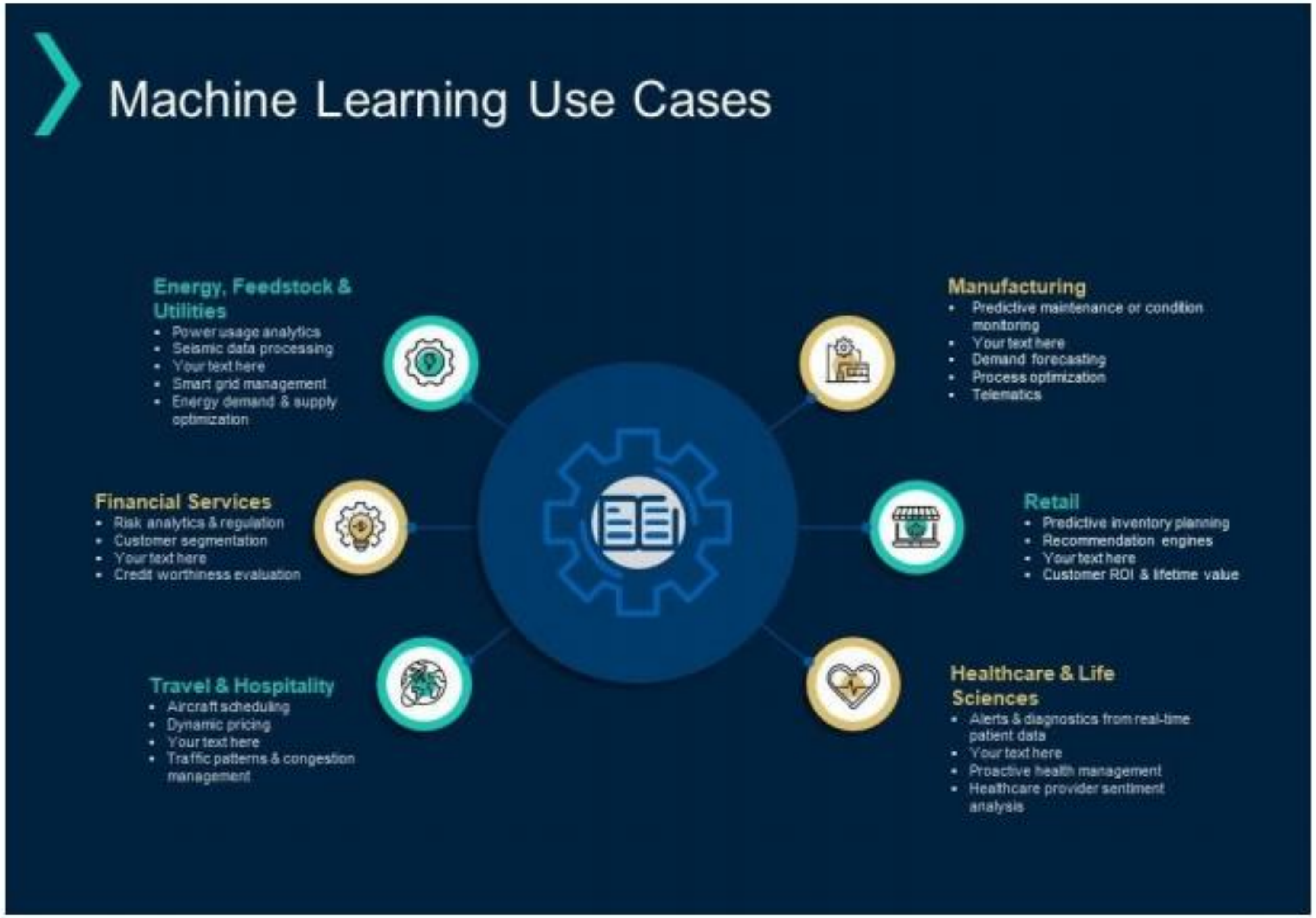
Intro

ML is a subset of AI.

Name derived from the concept that it deals with “construction & study of systems that can learn from data”



Use Cases



Machine Learning Use Cases

Energy, Feedstock & Utilities
- Power usage analytics
- Seismic data processing
- Your text here
- Smart grid management
- Energy demand & supply optimization

Financial Services
- Risk analytics & regulation
- Customer segmentation
- Your text here
- Credit worthiness evaluation

Travel & Hospitality
- Aircraft scheduling
- Dynamic pricing
- Your text here
- Traffic patterns & congestion management

Manufacturing
- Predictive maintenance or condition monitoring
- Your text here
- Demand forecasting
- Process optimization
- Telematics

Retail
- Predictive inventory planning
- Recommendation engines
- Your text here
- Customer ROI & lifetime value

Healthcare & Life Sciences
- Alerts & diagnostics from real-time patient data
- Your text here
- Proactive health management
- Healthcare provider sentiment analysis

Essential Libraries

1. **numpy: The matrix / numerical analysis layer at the bottom**
2. **scipy: Scientific computing utilities (linalg, FFT, signal/image processing...)**
3. **sklearn: Machine learning (our focus here)**
4. **matplotlib: Plotting and visualization**
5. **opencv: Computer vision**
6. **pandas: Data analysis**
7. **caffe, theano, minerva: Deep neural networks**
8. **spyder: The front end (Scientific Python Development Environment)**

Types of Learning

- **Supervised-**Predict unknown data attributes(outcomes), based on known attributes (predictors), Model built on training data set (has both predictors and outcomes).

Types – Regression (continuous outcome), Classification (classes)

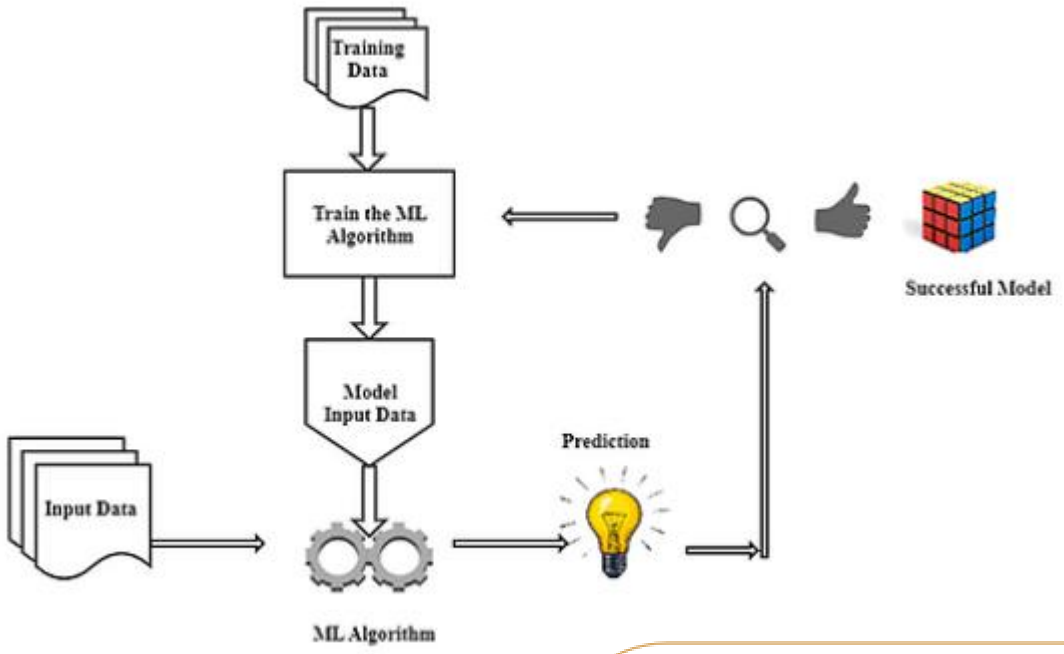
- **Unsupervised** – similarity / grouping entities.

Types- Clustering, Associative Rule Mining, Collaborative filtering

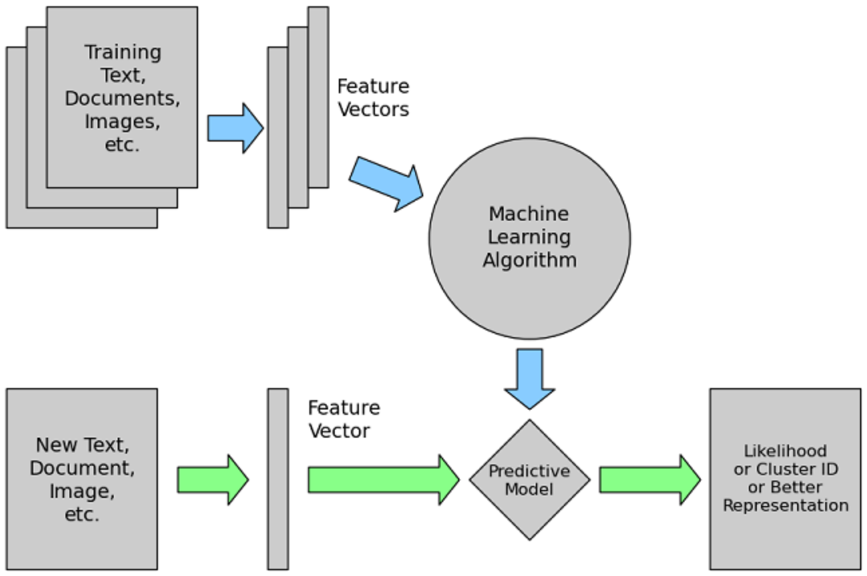
- **Reinforcement-** learn from feedback

Types- Deep approach, Inverse approach, apprentice approach.

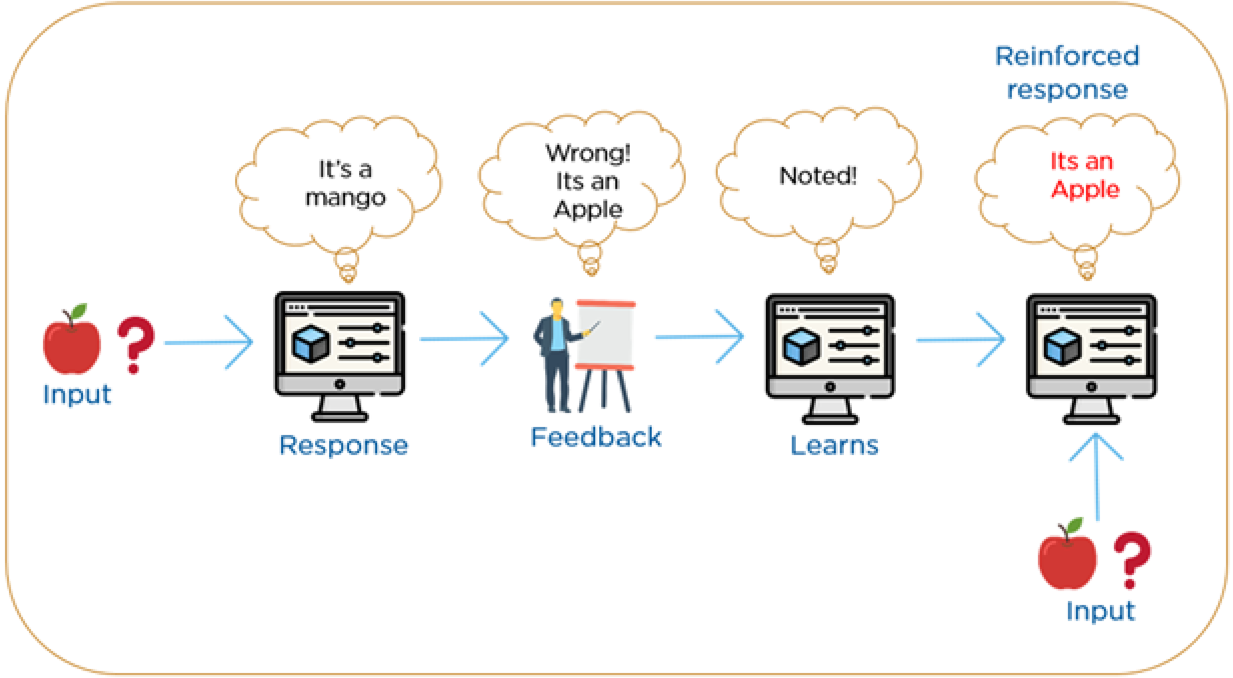
SUPERVISED



UN-SUPERVISED


Training Text, Documents, Images, etc.
Feature Vectors
Machine Learning Algorithm
New Text, Document, Image, etc.
Feature Vector
Predictive Model
Likelihood or Cluster ID or Better Representation

RE-INFORCEMENT



Additional Concepts

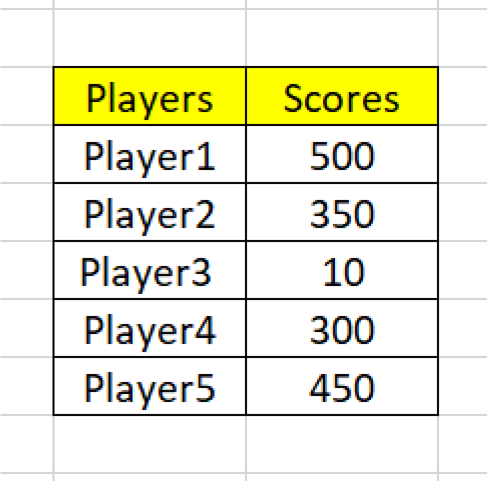
- **Data preprocessing-**Machine only understand numbers, e.g. text data is converted to numerical factors i.e. Document term Matrix
- **Metrics**- Confusion matrix(based on prediction types) , ROC AUC etc.
- **Errors** – In sample, Out of sample - Over fitting, MAE,MASE etc
- **Algorithm Tuning-** Accuracy, Sensitivity, Specificity, Precision etc, Hyperparameter Tuning, Cross validation

Data Preprocessing

Outlier identification

The difference between a good and an average machine learning model is often its ability to clean data. One of the biggest challenges in data cleaning is the identification and treatment of outliers.

In simple terms, outliers are observations that are significantly different from other data points. Even the best machine learning algorithms will underperform if outliers are not cleaned from the data because outliers can adversely affect the training process of a machine learning algorithm, resulting in a loss of accuracy.



Players	Scores
Player1	500
Player2	350
Player3	10
Player4	300
Player5	450

Hands On- Boston House Pricing Dataset which is included in the sklearn dataset API

Discretization/Binning

Converting numeric to categorical

- **Manual**

If age is less than 40 then 1,

If between 40 to 60 then 2,

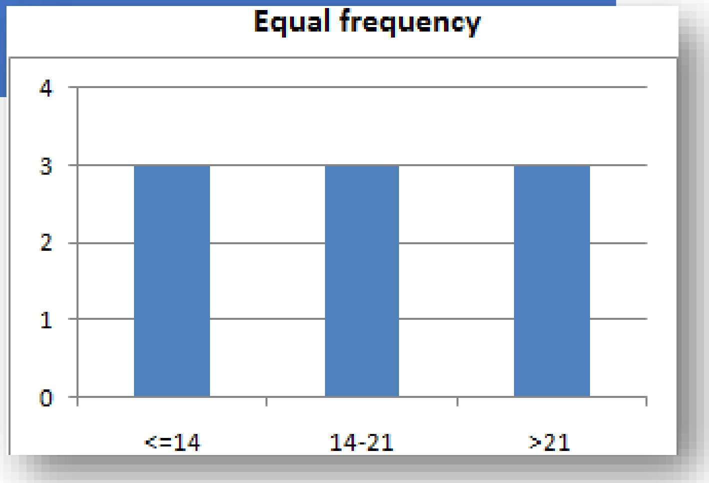
If more than 60 then 3

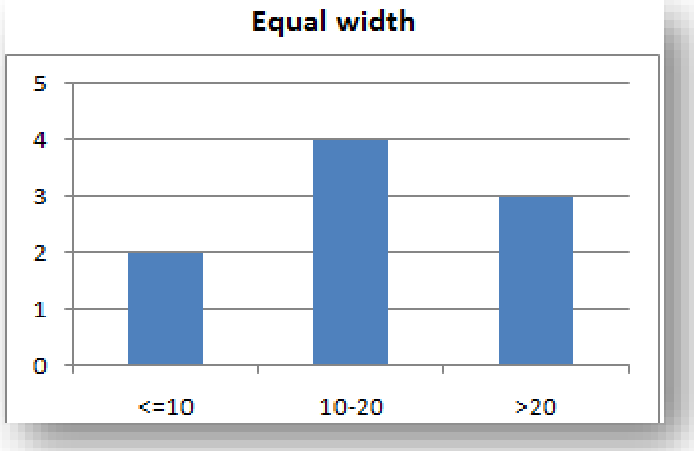
- Equal width

Interval is same, data is divided into *k* groups where each group has approximately the same value .

- Equal frequency

Number of sample in each bin is same, data is divided into *k* intervals of equal size.





Handling missing values

Data is not always fully available. e.g., many tuples have no recorded value for several attributes, such as customer income in sales data

- Missing data may be due to equipment malfunction
  - E.g., system failure
- data not entered due to misunderstanding
  - E.g., cities/zipcodes
- certain data may not be considered important at the time of entry
  - E.g., age of customers
- no registered history or changes of the data
  - E.g., employee promotion or change in designation

Hence, missing data may need to be inferred.

Data Cleansing: action

- Fill in missing values:
  - Ignore the tuple
  - Fill in the missing values manually: tedious + infeasible?
  - Use a global constant to fill in the missing value: e.g., “unknown”, a new class?!
  - Central imputation
    - Use the attribute mean (or majority nominal value) to fill in the missing value.
  - kNN imputation
    - Imputation using k-nearest neighbors. For each record, identify missing features. For each missing feature find the k nearest neighbors which have that feature. Impute the missing value using the imputation function on the k-length vector of values found from the neighbors.

    - Scenarios of NULL -> 0

      ""

      " "

      NA

      " "

      NULL

Central imputation

Stock	Price
Day1	22.4
Day2	20
Day3	19
Day4	
Day5	22.7
Day6	18.5

- Suppose the price value for day 4 is missing.
  - Compute the mean/median price for the given data
  - Substitute the value

- Average price = (22.4 + 20 + 19 + 22.7 + 18.5)/5
- = 20.52
- Median price = 20

- Hence, the Price for Day 4 = 20.5 or 20

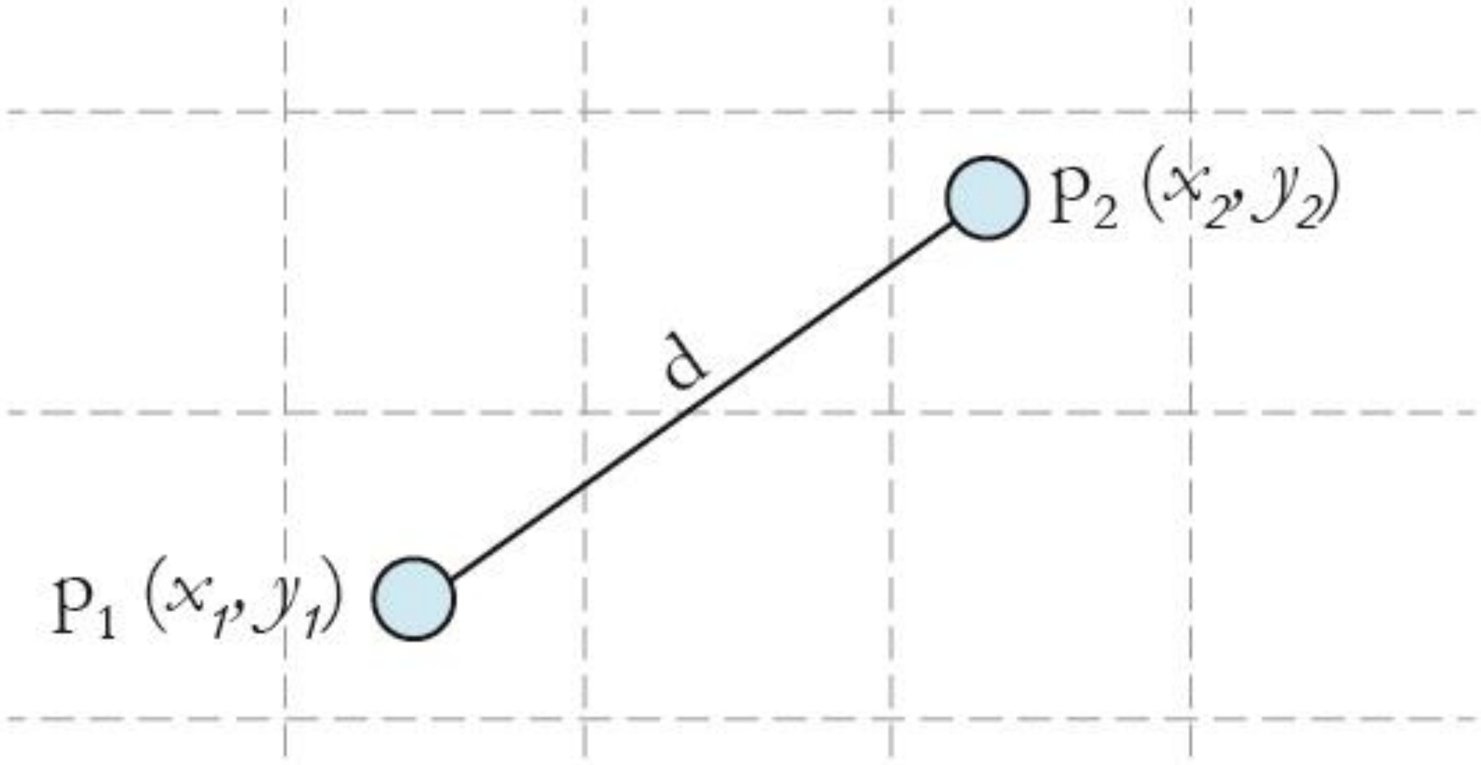
Central imputation

ToyCategory	Price
Infant	46
Infant	20
1 year	19
	45
2-3 years	22.7
4+	18.5

In the case, when the attribute is categorical then substitute with mode

Mode = Infant

Euclidean distance



$$\text{Euclidean distance (d)} = \sqrt{(x_2\text{-}x_1)^2 + (y_2\text{-}y_1)^2}$$

Feature Scaling

Sepal Length, Sepal Width → What is the magnitude & Units.

Most ML’s model use Euclidean distance, hence without feature scaling, most of the algo’s neglect the Units and focus on Magnitude, in that case the Euclidean distance may vary significantly → Hence, o/p will be impacted.

For e.g., the variable Age that ranges between 0 – 100 outweighs the Income that ranges between 10,000 – 50,000.

Age	Income (£)
24	15000
30	12000
28	30000

Income dominates completely!

Feature Scaling techniques

- Standard Scaler
- Min-Max Scaling
- Normalization
- Others..

Scaling techniques- Min-Max

- **Min-Max scaling/0-1**: each variable in the data set is recalculated as

$$(V - \min V)/(\max V - \min V)$$

where V represents the value of the variable in the original data set.

This method allows variables to have differing means and standard deviations but equal ranges.

In this case, there is at least one observed value at the 0 and 1 endpoints.

Example

Age	Income (£)	New value
24	15000	(15000 – 12000)/18000 = 0.16667
30	12000	(12000 – 12000)/18000 =0
28	30000	(30000 – 12000)/18000 =1

Income_Minimum = 12000
Income_Maximum = 30000
(Max – min) = (30000 – 12000) = 18000

Please note, the new values have
Minimum = 0
Maximum = 1

Hence, we have converted the income values between 0 and 1.

Scaling techniques- Standard

- **Standard/ Z-score scaling**:

variables recalculated as (*V* - mean of *V*)/s,

where "s" is the standard deviation. As a result, all variables in the data set have equal means (0) and standard deviations (1) but different ranges.

Example

Age	Income (£)	New value
24	15000	(15000 - 19000)/9643.65 = -0.4147
30	12000	(12000 - 19000)/9643.65 = -0.7258
28	30000	(30000-19000)/9643.65 = 1.1406

Average = (15000 + 12000 + 30000)/3 = 19000
Standard deviation = 9643.65

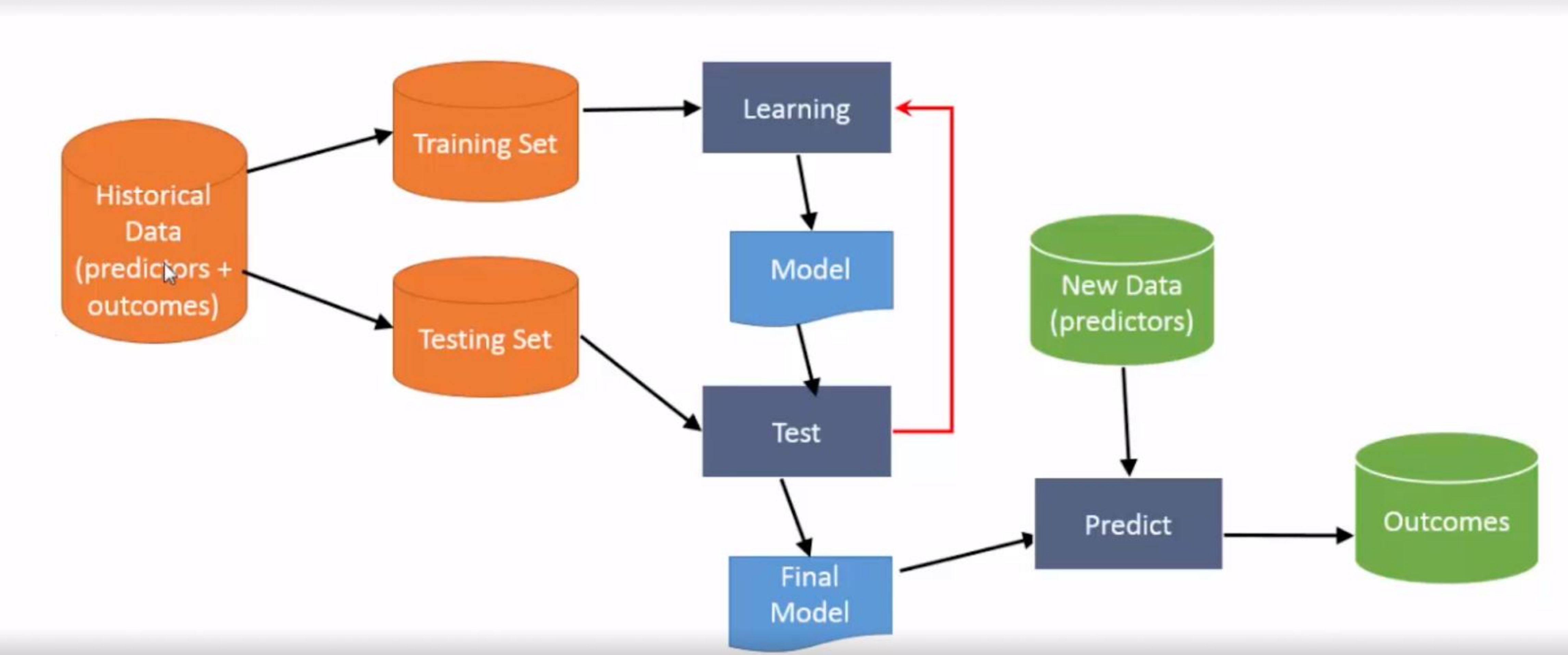
Hence, we have converted the income values to lower values using the z-score method.

x = c(-0.4147, -0.7258, 1.1406)
mean(x) = -0.000003 ~ 0
var(x) = 0.999 ~1

X	Income (£)
10000	15000
3	12000
28	30000

Age	Income
24	-0.4147
30	-0.7258
28	1.1406

Supervised Learning



Historical Data (predictors + outcomes)

Training Set

Testing Set

Learning

Model

Test

Final Model

New Data (predictors)

Predict

Outcomes

Training and Testing Data

- Historical Data contains both predictors and outcomes
- Split as training and testing data
- Training data is used to build the model
- Testing data is used to test the model
- Apply model on testing data
- Predict the outcome
- Compare the outcome with the actual value
- Measure accuracy
- Training and Test fit best practices
- 70-30 split
- Random selection of records. Should maintain data spread in both datasets

Regression

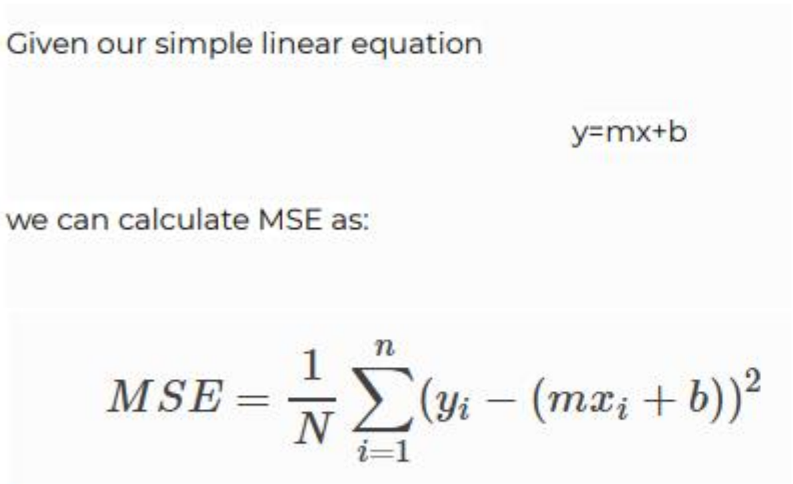
Regression analysis is a form of predictive modelling technique which investigates the relationship between a dependent (target) and independent variable (s) (predictor). This technique is used for forecasting, time series modelling and finding the causal effect relationship between the variables.

Let’s go through various regression algorithms:

1. Linear Regression
2. Multiple Linear Regression
3. Polynomial Regression
4. Stepwise Regression

Linear Regression

Simple linear regression is a statistical method that enables users to summarise and study the relationships between two continuous (quantitative) variables. Linear regression is a linear model wherein a model that assumes a linear relationship between the input variables (x) and the single output variable (y). Here, y can be calculated from a linear combination of the input variables (x). When there is a single input variable (x), the method is called a simple linear regression. When there are multiple input variables, the procedure is referred to as multiple linear regression.

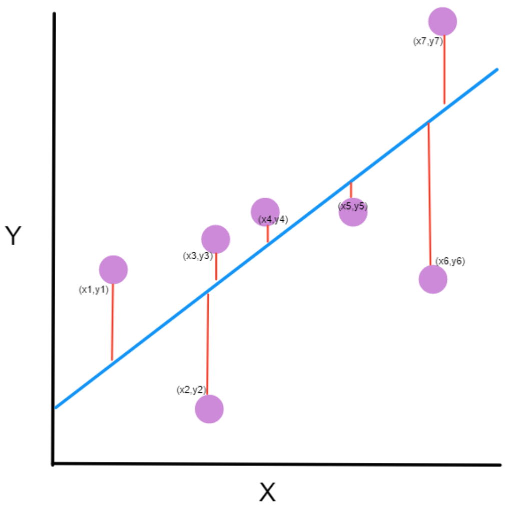


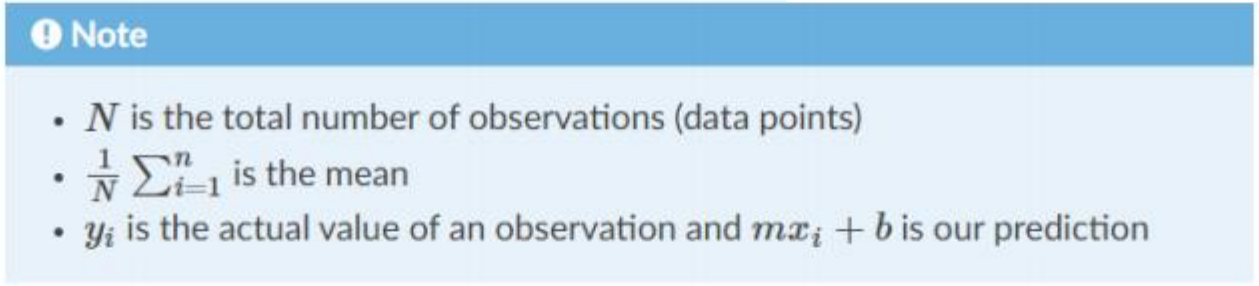
Given our simple linear equation

y=mx+b

we can calculate MSE as:

$$MSE = \frac{1}{N}\sum_{i=1}^{n}(y_i - (mx_i + b))^2$$





Note

- N is the total number of observations (data points)
- $\frac{1}{N}\sum_{i=1}^{n}$ is the mean
- y_i is the actual value of an observation and $mx_i + b$ is our prediction

Multiple Linear Regression

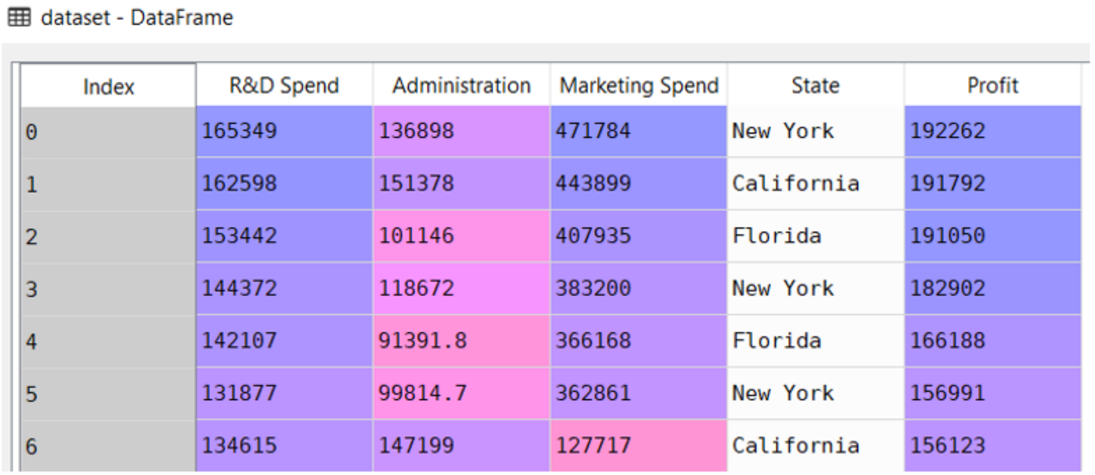
The difference between simple linear regression and multiple linear regression, multiple linear regression has (>1) independent variables, whereas simple linear regression has only 1 independent variable.

Simple Linear Regression → y = bo + b1*x1

Multiple Linear Regression → y = bo + b1*x1 + b2*x2 + …. + bn*xn

Where, y → Dependent variable

x1, x2, ….xn → Independent variables



dataset - DataFrame

Index	R&D Spend	Administration	Marketing Spend	State	Profit
0	165349	136898	471784	New York	192262
1	162598	151378	443899	California	191792
2	153442	101146	407935	Florida	191050
3	144372	118672	383200	New York	182902
4	142107	91391.8	366168	Florida	166188
5	131877	99814.7	362861	New York	156991
6	134615	147199	127717	California	156123

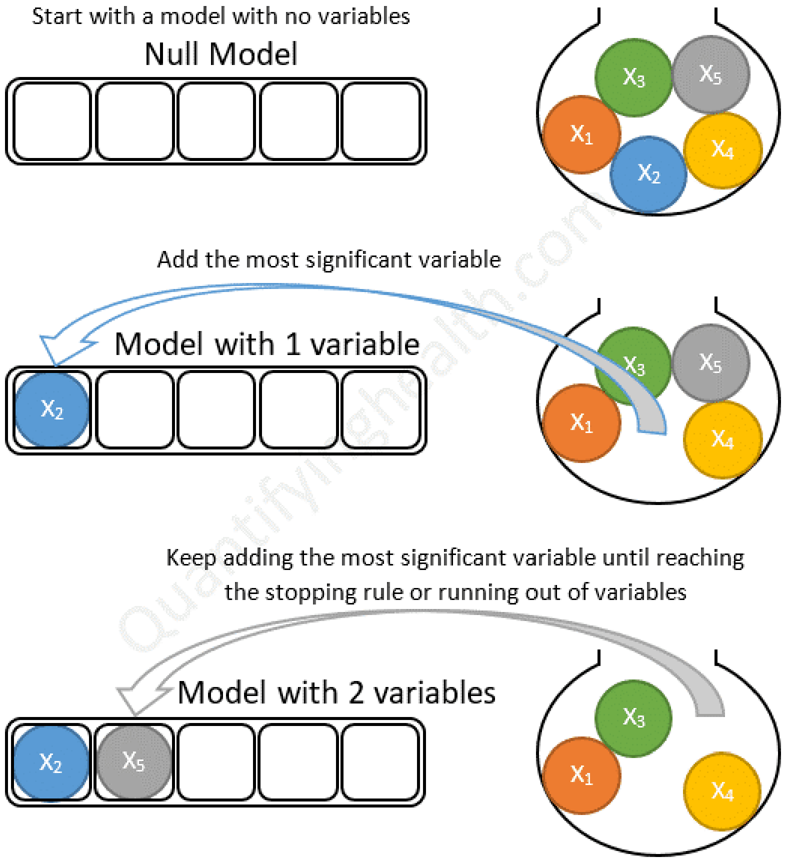
Multiple Linear Regression

5 methods of building models:

1. All-in
2. Backward Elimination → Stepwise
3. Forward Selection → Stepwise
4. Bidirectional Elimination → Stepwise
5. Score Comparison

All in cases: 2^n-1 = 2^10-1 = 1023 models..

Forward stepwise selection example with 5 variables:



Start with a model with no variables

Null Model

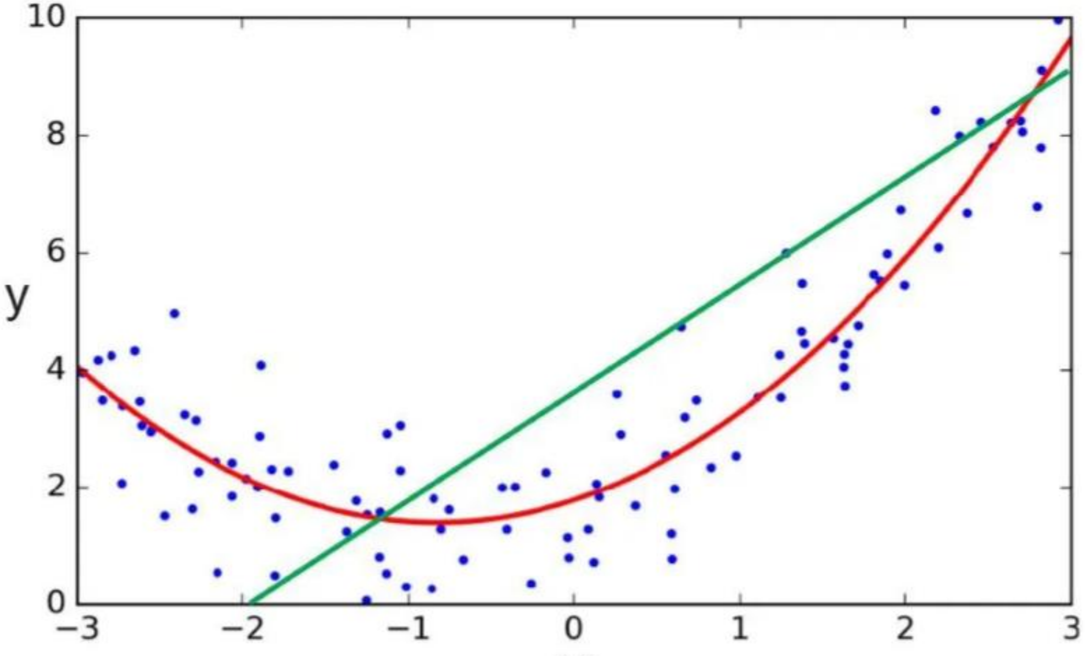
Add the most significant variable

Model with 1 variable

Keep adding the most significant variable until reaching the stopping rule or running out of variables

Model with 2 variables

Polynomial Regression



In this type of data, linear equation might not be a good fit as seen in the green line, hence we use Polynomial equation i.e. red line, which completely fits in with the data points and comes with an equation of

$$y = b_0 + b_1x_1 + b_2x_1^2$$

The equation of Polynomial Regression is:

Simple Linear Regression	$y = b_0 + b_1x_1$
Multiple Linear Regression	$y = b_0 + b_1x_1 + b_2x_2 + \dots + b_nx_n$
Polynomial Linear Regression	$y = b_0 + b_1x_1 + b_2x_1^2 + \dots + b_nx_1^n$

Covariance

The **covariance** of two variables *x* and *y* in a data set measures how the two are linearly related. A positive covariance would indicate a positive linear relationship between the variables, and a negative covariance would indicate the opposite.

Similarly, the **covariance** is defined in terms of the mean as:

$$s_{xy} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

However, covariance can be a very large number. It is best to express it as a normalized number between -1 and 1 to understand the relation between both quantities. This is achieved by normalizing covariance with standard deviations of both variables.

Correlation

- **Correlation** : a mutual relationship or connection between two or more things

•Interdependence & Correlation between 2 sets of data ar

•**Example** : Age and Blood Pressure

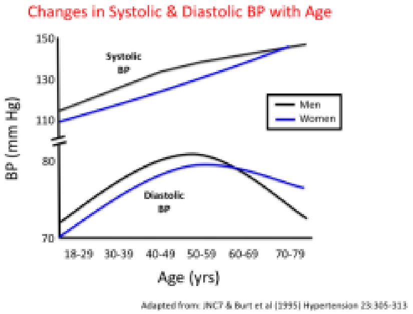
•Pearson’s Correlation co-efficient -1 to +1

•**Causation** : The reason for a change in value

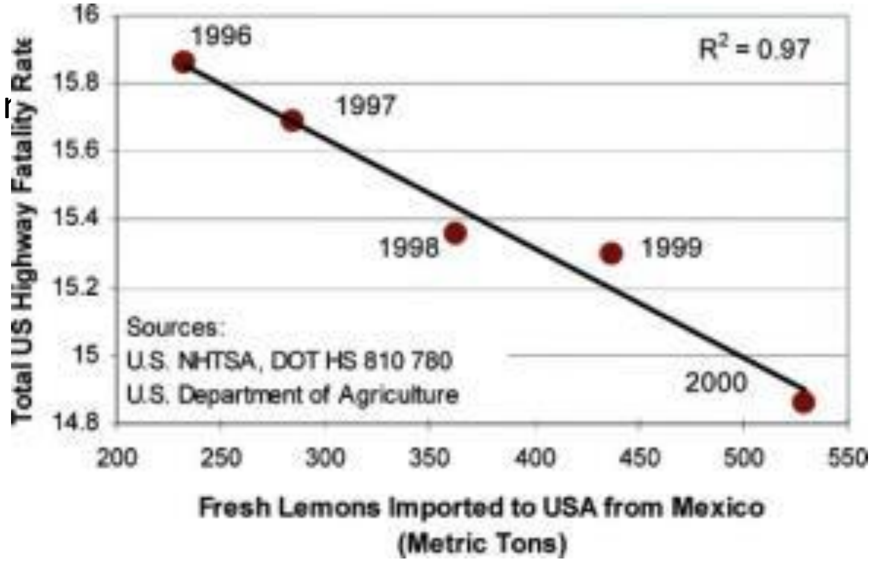
•Correlation does not imply causation

•Correlation might be **due** to –

Causation, Common cause, Incidental

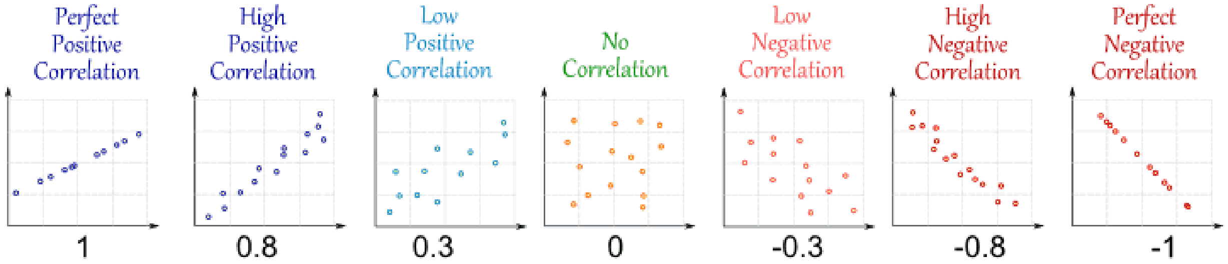


Adapted from: JNC7 & Burt et al (1995) Hypertension 25:305-313



Sources:
U.S. NHTSA, DOT HS 810 780
U.S. Department of Agriculture

$$Cor(x,y) = \rho_{xy}/\sigma_x\sigma_y$$



Correlation Coefficient

Correlation Coefficient, r, gives the strength and direction of the relationship between two variables.

$r = \frac{bs_x}{s_y} = \frac{s_{xy}}{s_x s_y}$ where b is the slope of the line of best fit, s_x is the standard deviation of the x values in the sample, s_y is the standard deviation of the y values in the sample and s_{xy} is the covariance between x and y.

Coefficient of Determination

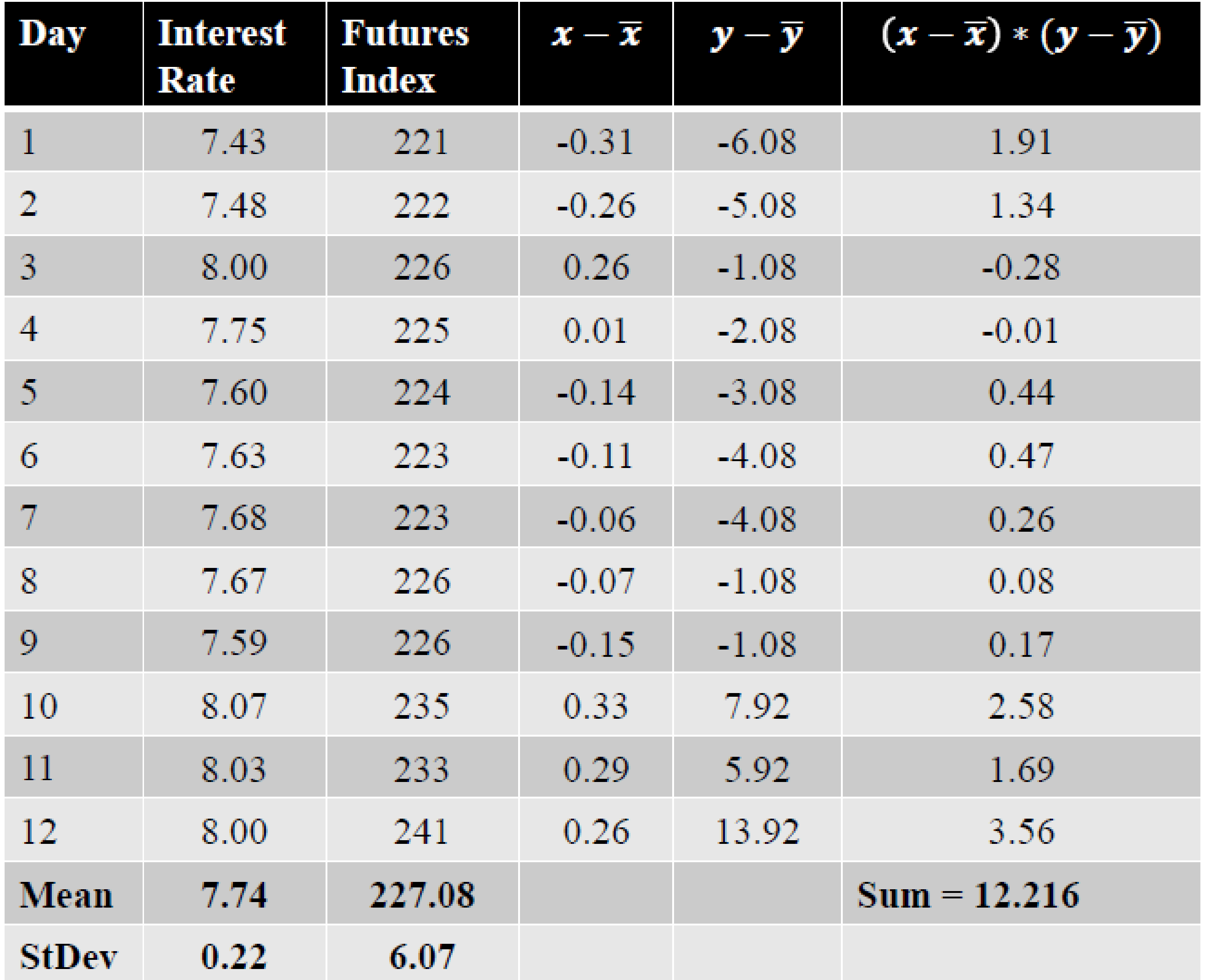
The coefficient of determination is given by r^2 or R^2. It is the percentage of variation in the *y* variable that is explainable by the *x* variable.

If $r^2 = 0$, it means you can't predict the *y* value from the *x* value.

If $r^2 = 1$, it means you can predict the *y* value from the *x* value without any errors.

Usually, r^2 is between these two extremes.

Covariance, Correlation and R^2



Day	Interest Rate	Futures Index	$x-\bar{x}$	$y-\bar{y}$	$(x-\bar{x})*(y-\bar{y})$
1	7.43	221	-0.31	-6.08	1.91
2	7.48	222	-0.26	-5.08	1.34
3	8.00	226	0.26	-1.08	-0.28
4	7.75	225	0.01	-2.08	-0.01
5	7.60	224	-0.14	-3.08	0.44
6	7.63	223	-0.11	-4.08	0.47
7	7.68	223	-0.06	-4.08	0.26
8	7.67	226	-0.07	-1.08	0.08
9	7.59	226	-0.15	-1.08	0.17
10	8.07	235	0.33	7.92	2.58
11	8.03	233	0.29	5.92	1.69
12	8.00	241	0.26	13.92	3.56
Mean	**7.74**	**227.08**			**Sum = 12.216**
StDev	**0.22**	**6.07**			

$$Cov = \frac{12.216}{11} = 1.111$$

$$r = \frac{1.111}{0.22 * 6.07} = 0.815$$

$$R^2 = 0.815^2 = 0.665$$

“p-values” and Significance Levels

- p-value suggests the probabiity of our experiment results(stats params with sample population) happening, p-value less than alpha value means there is less chance of experiment results happening
- In Linear Regression, the Null Hypothesis is that the coefficients associated with the variables is equal to zero
- So, p-value of <0.05 means the we can reject null hypothesis that the coefficients associated with the variables is equal to zero. Hence, the variables are contributing to the model.

Regression Error Metrics

MAE, MSE, RMSE, MAPE etc.

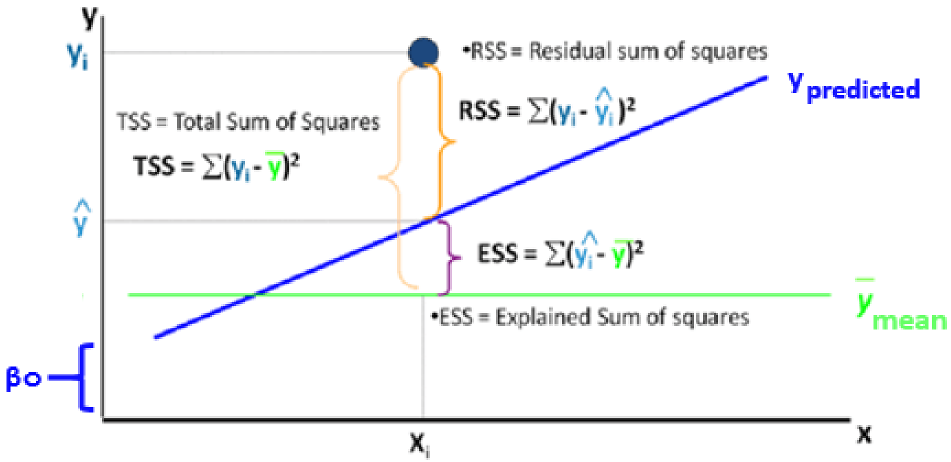
MSE (Mean Squared Rrror)

RMSE (Root Mean Squared Error)

MAE (Mean Absolute Error)

MAPE (Mean Absolute Percentage Error)

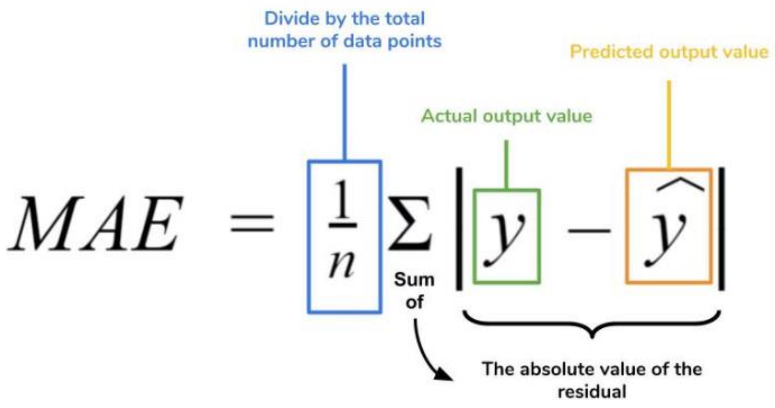
Anatomy of Regression Errors


y
y_i
•RSS = Residual sum of squares
$y_{predicted}$
RSS = $\sum(y_i - \hat{y}_i)^2$
TSS = Total Sum of Squares
TSS = $\sum(y_i - \bar{y})^2$
$\hat{y}$
ESS = $\sum(\hat{y}_i - \bar{y})^2$
$\bar{y}_{mean}$
•ESS = Explained Sum of squares
βo
X_i
x

Mean Absolute Error

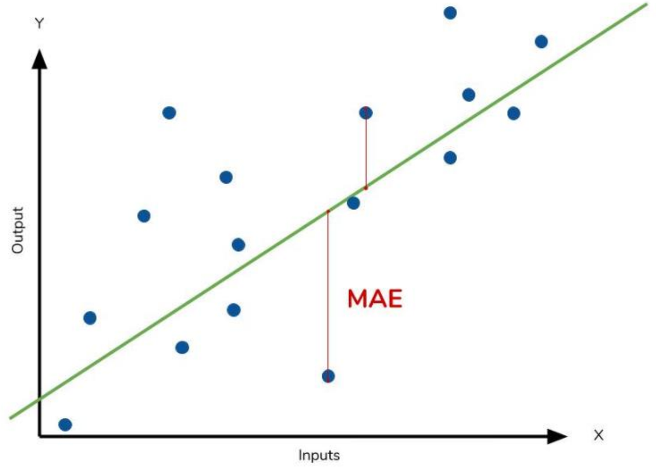
MAE: We just look at the absolute difference between data and model’s predictions

Lower MAE → Better model



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

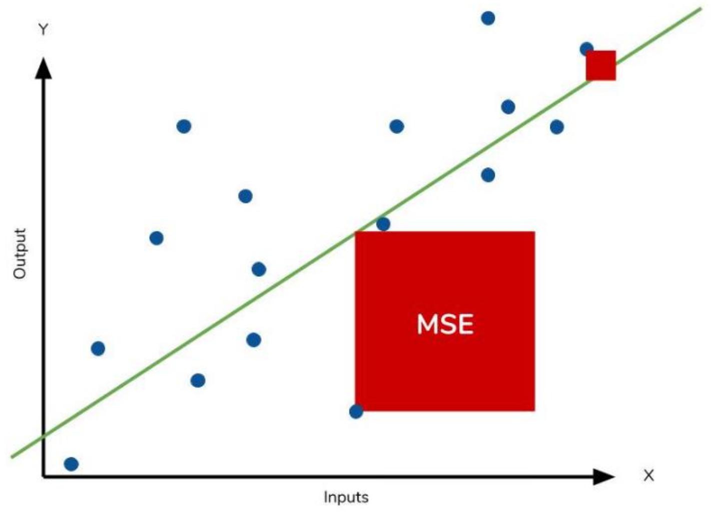
Divide by the total number of data points
Predicted output value
Actual output value
Sum of
The absolute value of the residual



Mean Squared Error

MSE is going to be a huge number because of squaring, hence, we can't compare it with MAE. This ultimately means that outliers in our data will contribute to much higher total error in the MSE than they would the MAE. Similarly, our model will be penalized more for making predictions that differ greatly from the corresponding actual value. This is to say that large differences between actual and predicted are punished more in MSE than in MAE.

$$MSE = \frac{1}{n}\Sigma\underbrace{\left(y - \widehat{y}\right)^{2}}_{\text{The square of the difference}}$$



beingdatum
the data society

Root Mean Squared Error

RMSE = Square root(MSE)

As the name suggests, it is the square root of the MSE. Because the MSE is squared, its units do not match that of the original output. Researchers will often use RMSE to convert the error metric back into similar units, making interpretation easier. Since the MSE and RMSE both square the residual, they are similarly affected by outliers.

AIC and BIC

- Akaike information criterion (AIC) (Akaike, 1974) is a fined technique based on in-sample fit to estimate the likelihood of a model to predict/estimate the future values.
- Bayesian information criterion (BIC) (Stone, 1979) is another criteria for model selection that measures the trade-off between model fit and complexity of the model. A lower AIC or BIC value indicates a better fit.
- AIC=-2*lnL+2*k
- BIC=-2*lnL+2*lnN*k
- where L is the value of the likelihood, N is the number of recorded measurements, and k is the number of estimated parameters.

Improve Accuracy

How can you check and improve the accuracy of a regression model ?

- You can do variable selection based on **p values**. If a variable shows p value > 0.05, we can remove that variable from model since at p> 0.05, we'll always fail to reject null hypothesis. Try for Stepwise Regression.
- Higher R squared
- Error metrics- As low as possible
- Do as much Pre-processing as possible.

Regularization

- Ridge and Lasso work by penalizing the magnitude of coefficients of features along with minimizing the error between predicted and actual observations. These are called ‘regularization’ techniques. The key difference is in how they assign penalty to the coefficients:

Ridge Regression:

- Performs L2 regularization, i.e. adds penalty equivalent to square of the magnitude of coefficients
- Minimization objective = OLS Obj + a * (sum of square of coefficients)

Lasso Regression:

- Performs L1 regularization, i.e. adds penalty equivalent to absolute value of the magnitude of coefficients
- Minimization objective = OLS Obj + a * (sum of absolute value of coefficients)
- Note that here ‘OLS Obj’ refers to ‘ordinary least squares objective’, i.e. the linear regression objective without regularization.

Classification

Predicting category of new observations

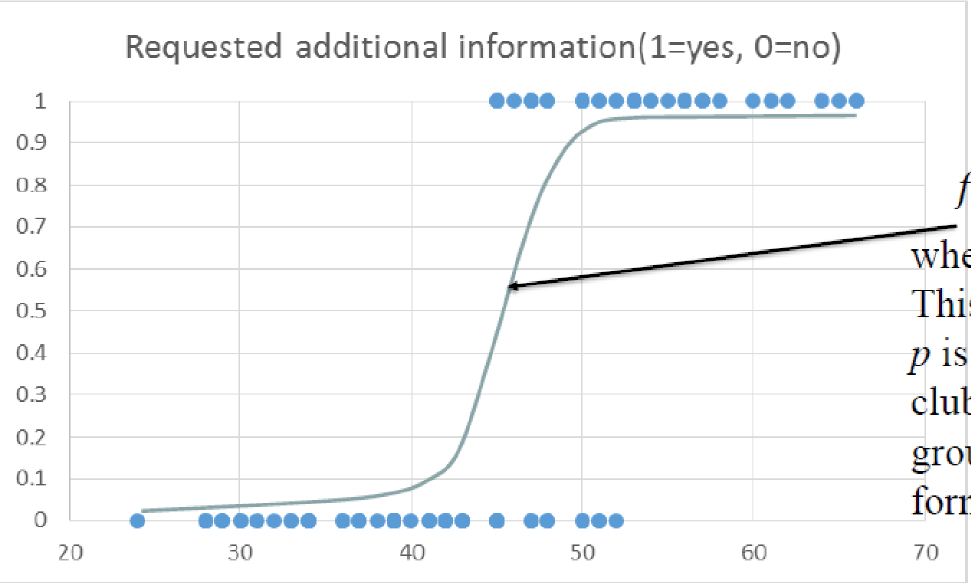
Let’s go through various classification algorithms:

1. k-nn (K- nearest neighbours)
2. Decision Trees
3. Random Forest
4. Naive Bayes
5. Support Vector Machines
6. Logistic Regression

Logistic Regression

- Logistic regression is a special case of linear regression where we only predict the outcome in a categorical variable. It predicts the probability of the event using the log functior
- We use the **Sigmoid function/curve** to predic the categorical value. The threshold value decides the outcome(win/lose).
- Linear regression equation: **y = β0 + β1X1 + β2X2 …. + βnXn**
- Sigmoid function: $\boldsymbol{p = 1 / 1 + e^{-y} = e^{y} /1 + e^{y}}$
- Logistic Regression equation:

$$\boldsymbol{p = 1 / 1 + e^{-(\beta 0 + \beta 1X1 + \beta 2X2 .... + \beta nXn)}}$$



$$f(x) = p = \frac{e^{\mu}}{1 + e^{\mu}}$$

where $\mu = \beta_0 + \beta_1 x_1$
This is a logistic model.
p is the probability that a club member fits into group 1 (returns the form; success).

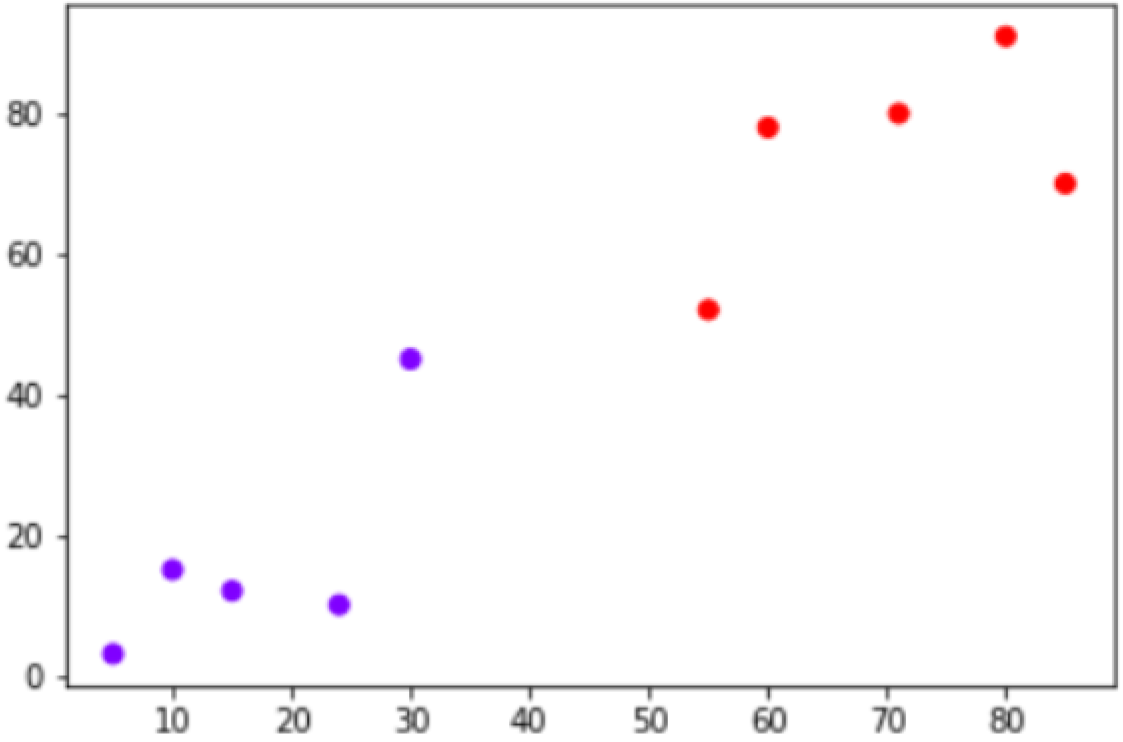
K-NN (K-Nearest Neighbors)

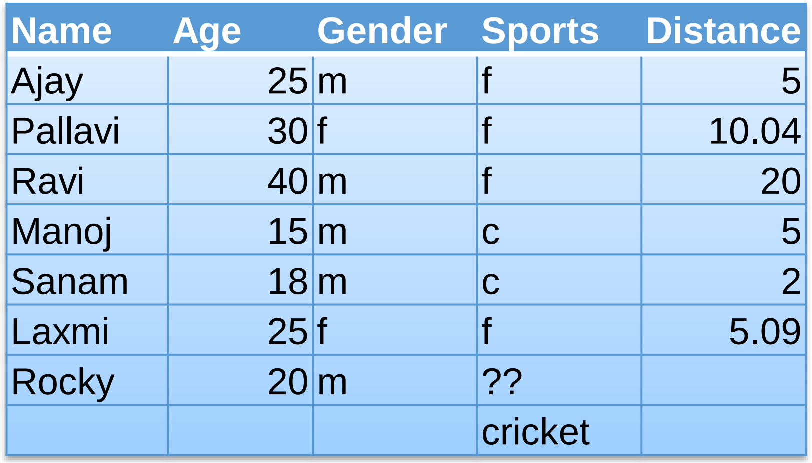
K-Nearest Neighbors.

Let say, we have a point at (40,60)

As we have 2 classes here, let’s assume k = 3 (k can be any value as multiple of number of classes + 1)

As, we see 2 reds and 1 blue nearby the (40,60) mark, we can assume the new datapoint as red class.





Name	Age	Gender	Sports	Distance
Ajay	25	m	f	5
Pallavi	30	f	f	10.04
Ravi	40	m	f	20
Manoj	15	m	c	5
Sanam	18	m	c	2
Laxmi	25	f	f	5.09
Rocky	20	m	??	
			cricket	

Let say, Male = 1, Female = 0

Euclidean distance = $\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$

For Ajay →

$\sqrt{(25-20)^2+(1-1)^2}$

$= 5$

As, M=1, F=0

For Pallavi →

$\sqrt{(30-20)^2+(0-1)^2}$

$= \sqrt{101} = 10.04$

- And similarly, all the distance is calculated to Rocky, and we found that the 3 nearest neighbors are 5, 5, 2 i.e. Ajay, Manoj & Sanam and these 3 plays (1 Football, 2 Cricket) -->
- So based on the distance, 2 of the nearest neighbors are Cricket, so Rocky's game is **Cricket.**

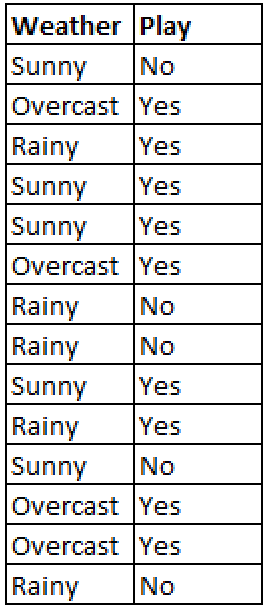
- **You can also use Manhattan distance by passing p=1, the formula as below:**

$$|x_1 - x_2| + |y_1 - y_2|.$$

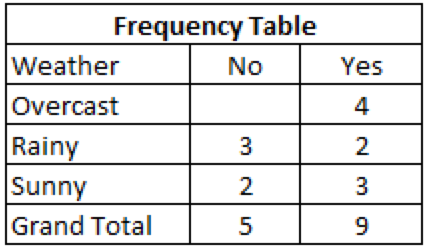
Naive Bayes

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

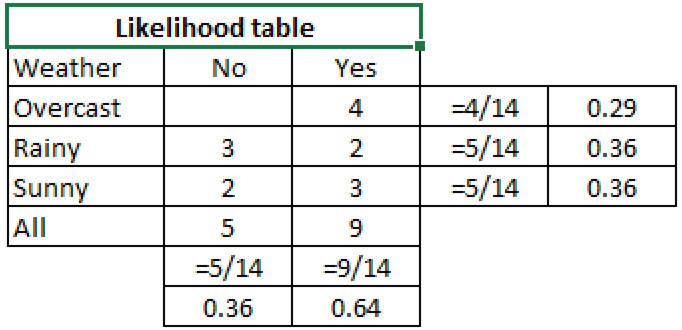
Let's understand it using an example. Below I have a training data set of weather and corresponding target variable 'Play' (suggesting possibilities of playing). Now, we need to classify whether players will play or not based on weather condition. Let's follow the below steps to perform it



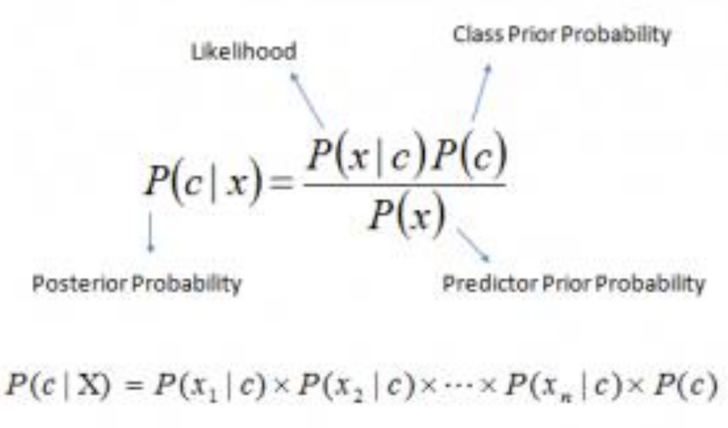
Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No



Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9



Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		



$$P(c \mid x) = \frac{P(x \mid c) P(c)}{P(x)}$$

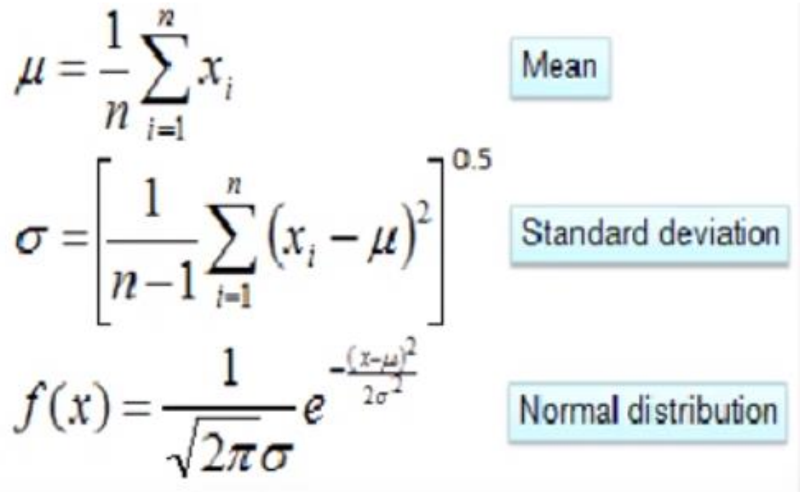
$$P(c \mid \mathbf{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Problem: Players will play if weather is sunny. Is this statement is correct?
We can solve it using above discussed method of posterior probability.
P(Yes | Sunny) = P(Sunny | Yes) * P(Yes) / P (Sunny)
Here we have P (Sunny |Yes) = 3/9 = 0.33, P(Sunny) = 5/14 = 0.36, P(Yes)= 9/14 = 0.64
Now, P (Yes | Sunny) = 0.33 * 0.64 / 0.36 = 0.60, which has higher probability.

Naive Bayes - Numerical Columns

If we have numerical data, just convert it to categorical. “OR”

Assume normal distribution for numerical variables.



$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

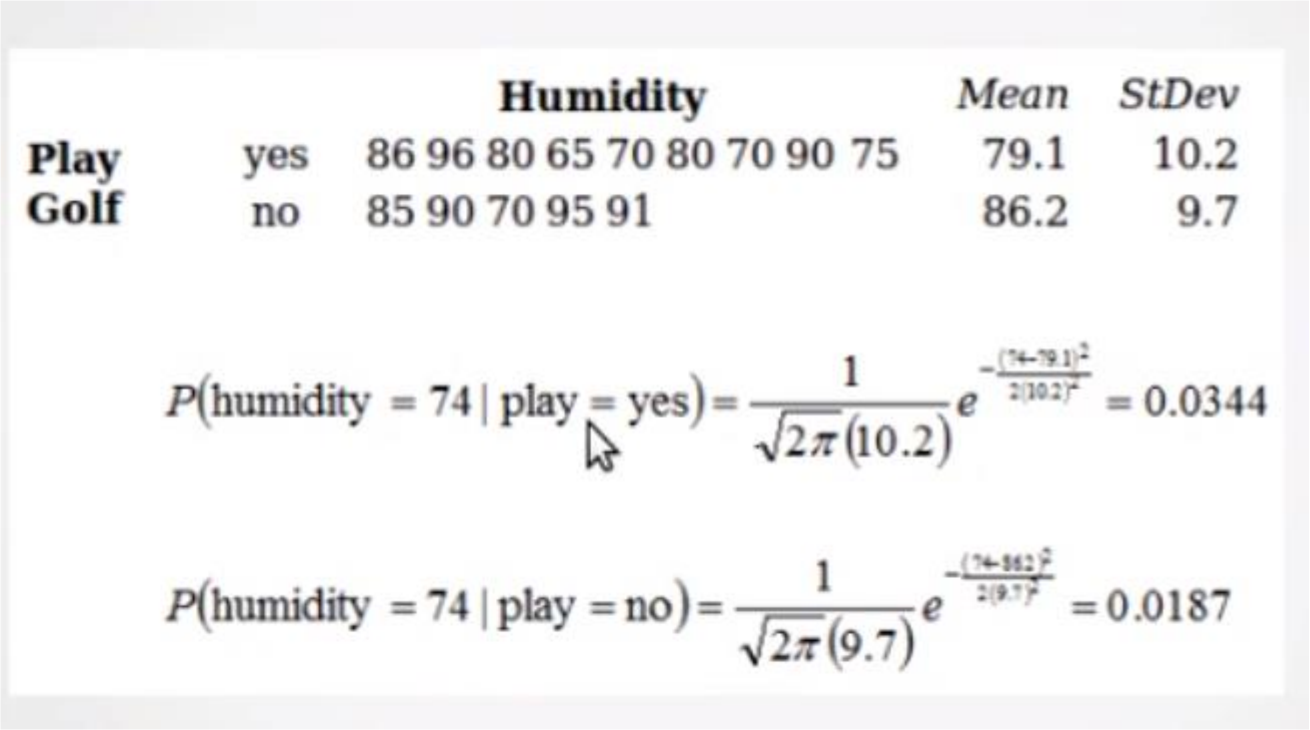
Mean

$$\sigma = \left[\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu)^2\right]^{0.5}$$

Standard deviation

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Normal distribution



		Humidity	*Mean*	*StDev*
Play Golf	yes	86 96 80 65 70 80 70 90 75	79.1	10.2
	no	85 90 70 95 91	86.2	9.7

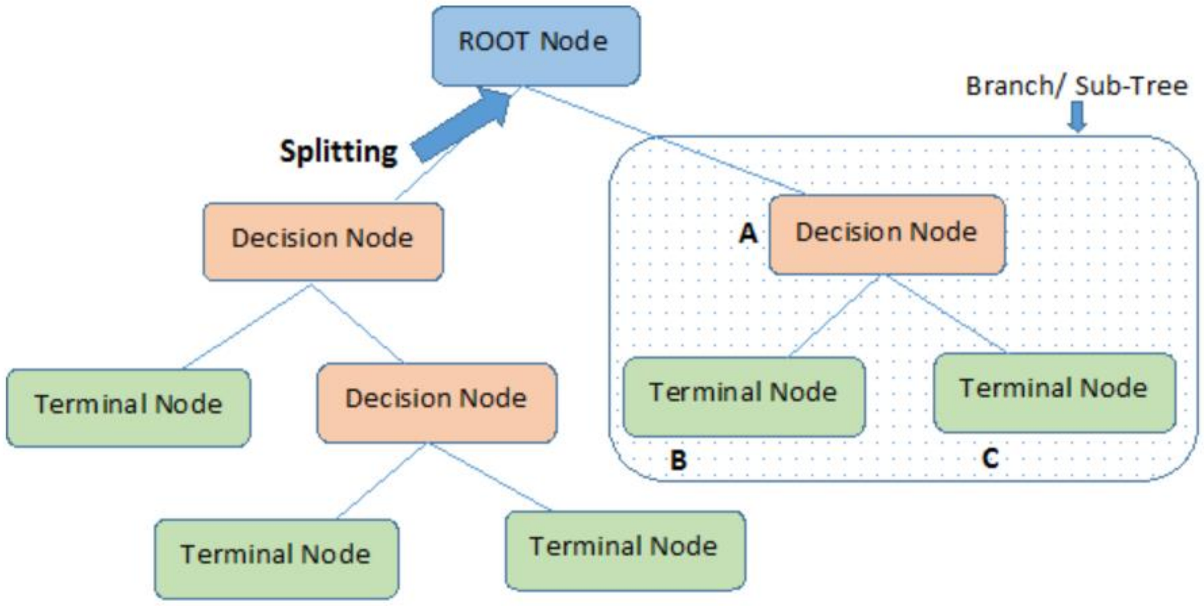
$$P(\text{humidity} = 74 \mid \text{play} = \text{yes}) = \frac{1}{\sqrt{2\pi}(10.2)} e^{-\frac{(74-79.1)^2}{2(10.2)^2}} = 0.0344$$

$$P(\text{humidity} = 74 \mid \text{play} = \text{no}) = \frac{1}{\sqrt{2\pi}(9.7)} e^{-\frac{(74-86.2)^2}{2(9.7)^2}} = 0.0187$$

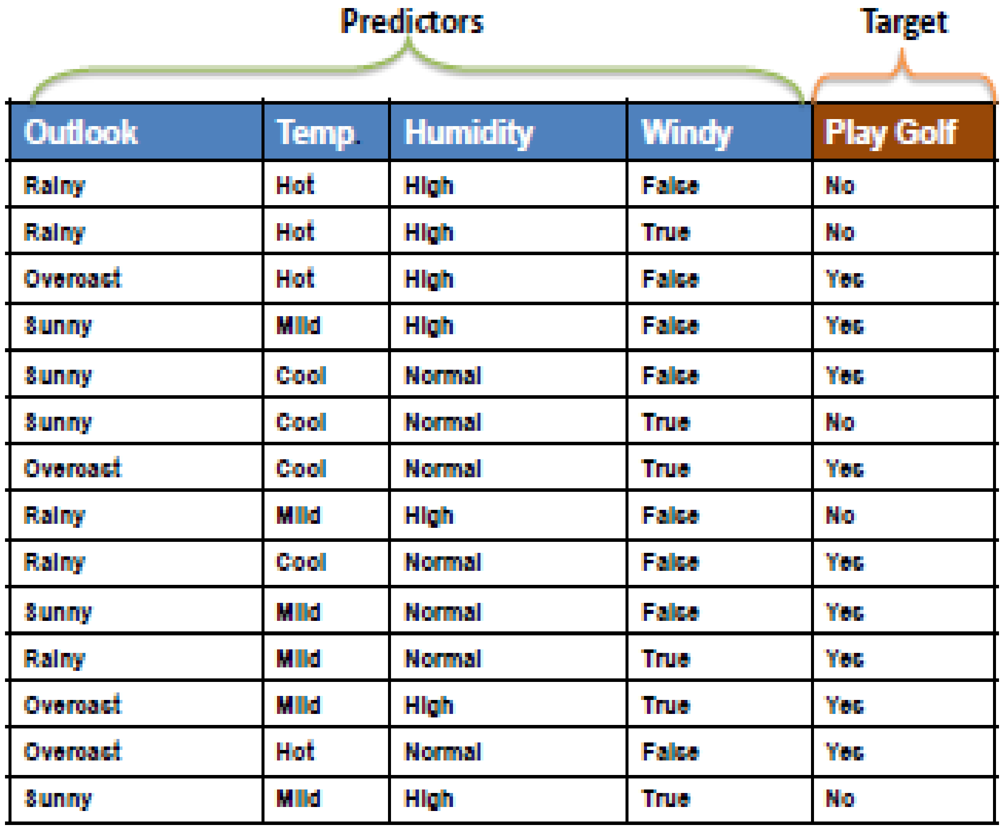
beingdatum
the data society

Decision Tree

It works on the principle of identifying the root node, and creates a tree to traverse from top-bottom approach to identify the right classes.

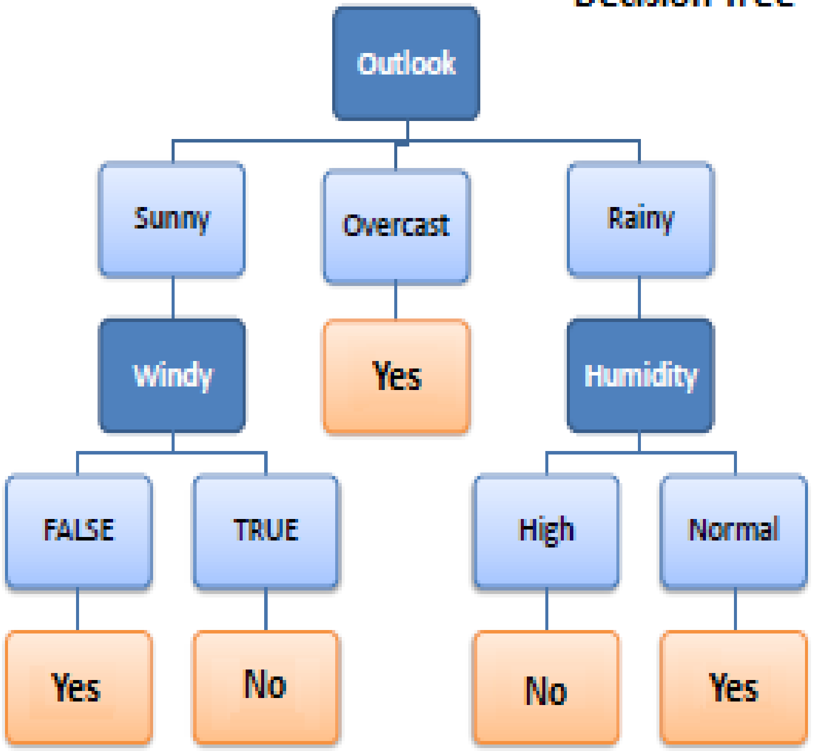


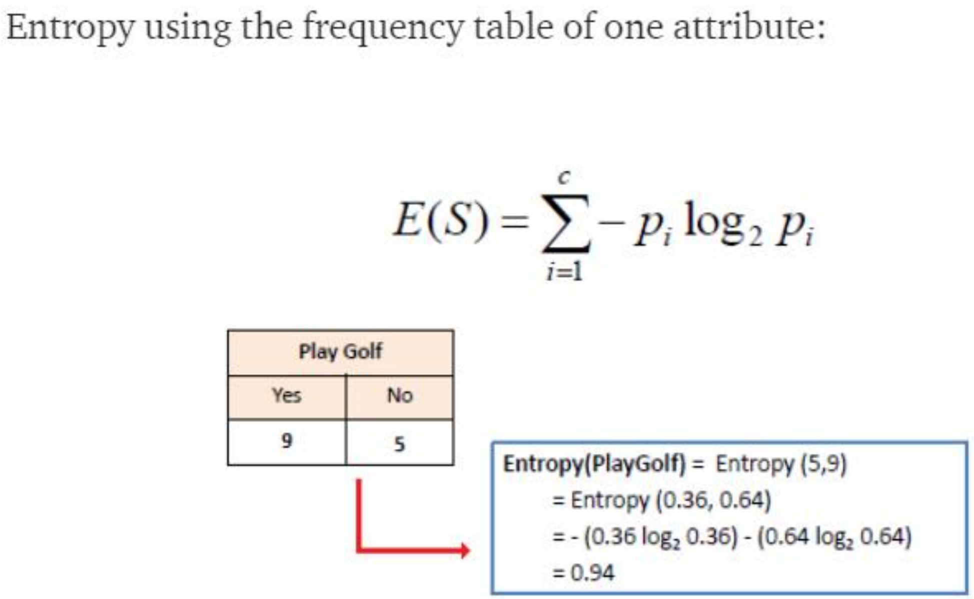
Note:- A is parent node of B and C.


Predictors
Target

Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Decision Tree





Entropy using the frequency table of one attribute:

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

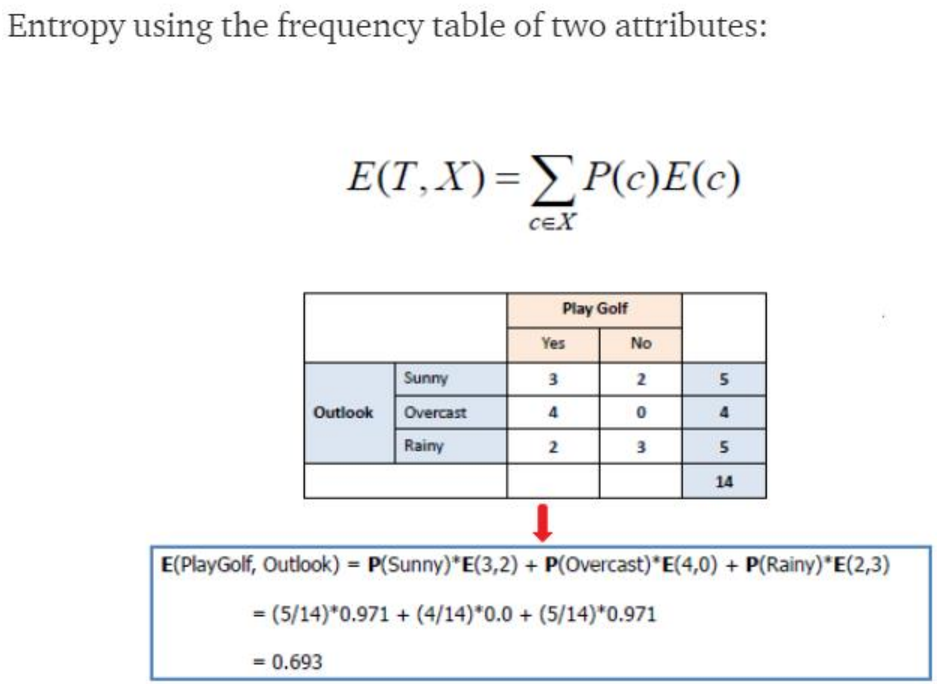
Play Golf	
Yes	No
9	5

Entropy(PlayGolf) = Entropy (5,9)

= Entropy (0.36, 0.64)

= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)

= 0.94



Entropy using the frequency table of two attributes:

$$E(T,X) = \sum_{c \in X} P(c)E(c)$$

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

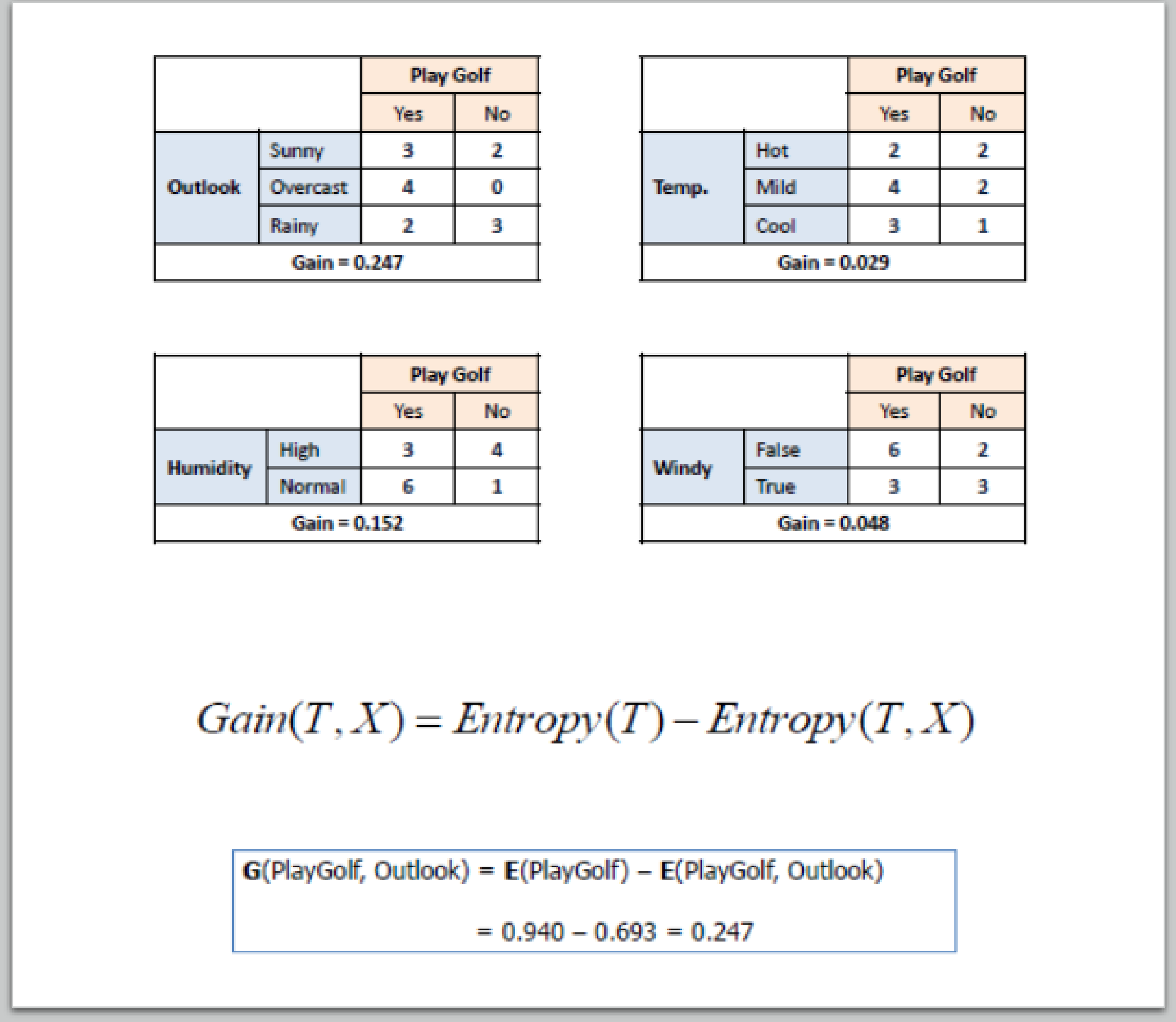
E(PlayGolf, Outlook) = **P**(Sunny)***E**(3,2) + **P**(Overcast)***E**(4,0) + **P**(Rainy)***E**(2,3)

= (5/14)*0.971 + (4/14)*0.0 + (5/14)*0.971

= 0.693

Entropy- A measure of Randomness

- **Information Gain:**The information gain is based on the decrease in entropy after a data-set is split on an attribute. Constructing a decision tree is all about finding attribute that returns the highest information gain (i.e., the most homogeneous branches). Ode split will take place on where entropy is less and Information gain is high.



		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1
Gain = 0.029			

		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1
Gain = 0.152			

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3
Gain = 0.048			

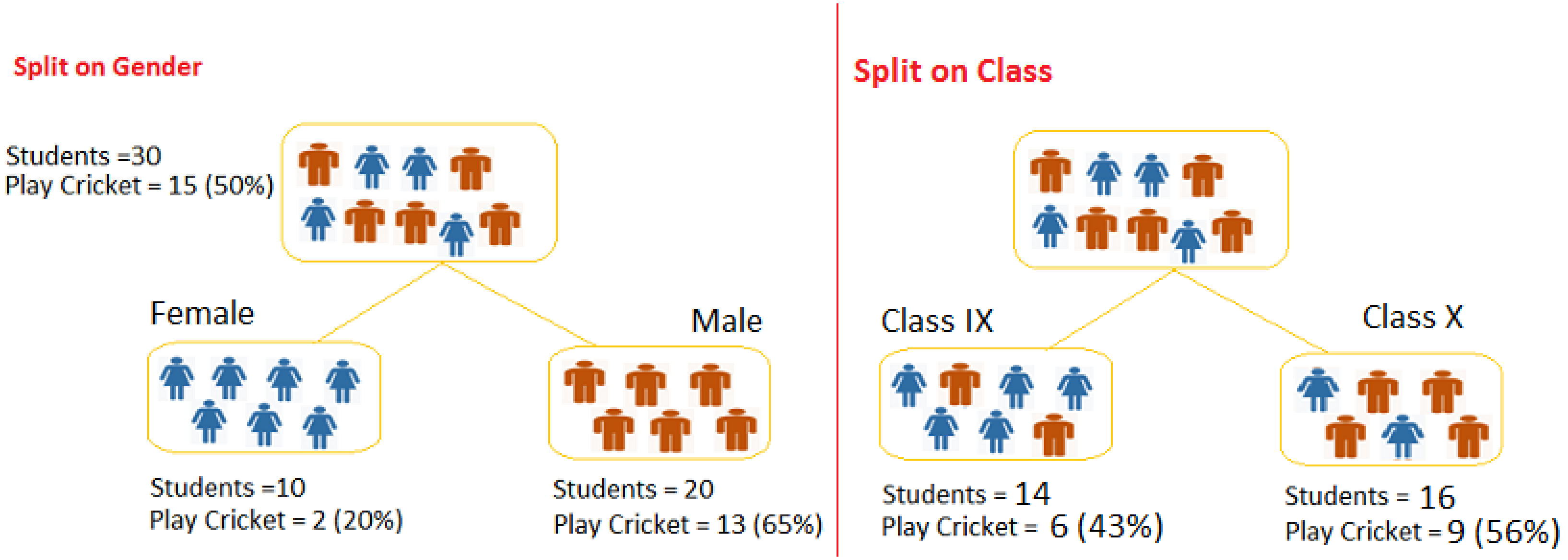
$$Gain(T,X) = Entropy(T) - Entropy(T,X)$$

G(PlayGolf, Outlook) = **E**(PlayGolf) – **E**(PlayGolf, Outlook)

= 0.940 – 0.693 = 0.247

Gini Index

- Calculates the homogeneity of a binary class
- Calculate Gini for sub-nodes, using formula sum of square of probability for success and failure (p^2+q^2).
- Weighted Gini for split using weighted Gini score of each node of that split
- **Example: —** Referring to example where we want to segregate the students based on target variable (playing cricket or not). In the snapshot below, we split the population using two input variables Gender and Class. Now, I want to identify which split is producing more homogeneous sub-nodes using Gini index.



- **Split on Gender:**
- Gini for sub-node Female = (0.2)*(0.2)+(0.8)*(0.8)=0.68
- Gini for sub-node Male = (0.65)*(0.65)+(0.35)*(0.35)=0.55
- Weighted Gini for Split Gender = (10/30)*0.68+(20/30)*0.55 = **0.59**
- **Similar for Split on Class:**
- Gini for sub-node Class IX = (0.43)*(0.43)+(0.57)*(0.57)=0.51
- Gini for sub-node Class X = (0.56)*(0.56)+(0.44)*(0.44)=0.51
- Weighted Gini for Split Class = (14/30)*0.51+(16/30)*0.51 = **0.51**
- Above, you can see that Gini score for *Split on Gender* is higher than *Split on Class,* hence, the node split will take place on Gender.

Setting Constraints on tree based algorithms

- **Minimum samples for a node split:** Defines the minimum number of samples (or observations) which are required in a node to be considered for splitting.

- **Minimum samples for a terminal node (leaf):** Defines the minimum samples (or observations) required in a terminal node or leaf.

Used to control over-fitting similar to min_samples_split.

- **Maximum depth of tree (vertical depth):** The maximum depth of a tree.

Used to control over-fitting as higher depth will allow model to learn relations very specific to a particular sample.

- **Maximum number of terminal nodes:** The maximum number of terminal nodes or leaves in a tree.

Can be defined in place of max_depth. Since binary trees are created, a depth of ‘n’ would produce a maximum of 2^n leaves.

- **Maximum features to consider for split:** The number of features to consider while searching for a best split. These will be randomly selected.

As a thumb-rule, square root of the total number of features works great but we should check upto 30-40% of the total number of features.

Higher values can lead to over-fitting but depends on case to case.

Pruning in tree based algorithms

We first make the decision tree to a large depth.
Then we start at the bottom and start removing leaves which are giving us negative returns when compared from the top.
Suppose a split is giving us a gain of say -10 (loss of 10) and then the next split on that gives us a gain of 20. A simple decision tree will stop at step 1 but in pruning, we will see that the overall gain is +10 and keep both leaves.

Note that sklearn's decision tree classifier does not currently support pruning. Advanced packages like xgboost have adopted tree pruning in their implementation. But the library *rpart* in R, provides a function to prune. Good for R users!

Classification Metrics

1. Classification Accuracy
2. Log Loss
3. Area under ROC
4. Confusion Matrix
5. Classification Report

Classification Accuracy: Number of correct predictions made as a ratio of all predictions made.

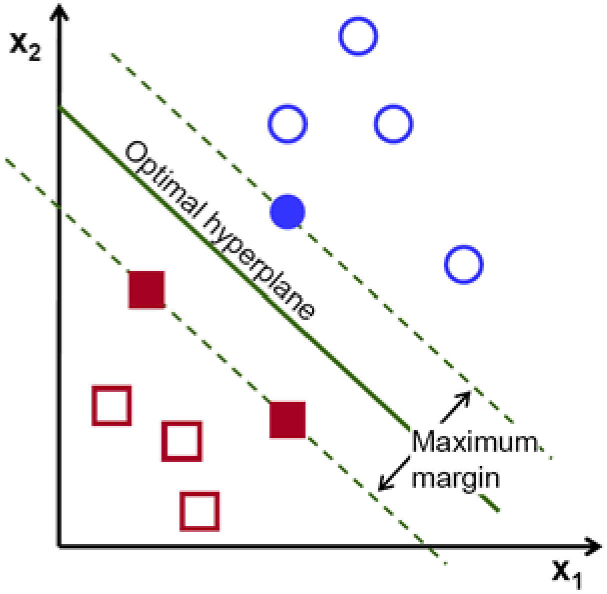
Log Loss: Evaluating the predictions of probabilities of membership to a given class.

Area under ROC Curve(ROC AUC): Used for binary classification problems. It’s a plot of true positive rate and true negative rate.

Confusion Matrix: It’s a handy presentation of the accuracy of a model with two or more classes.

Support Vector Machines

An SVM model is basically a representation of different classes in a hyperplane in multidimensional space. The hyperplane will be generated in an iterative manner by SVM so that the error can be minimized. The goal of SVM is to divide the datasets into classes to find a maximum marginal hyperplane (MMH).



Support Vectors

- Support vectors are the data points, which are closest to the hyperplane. These points will define the separating line better by calculating margins. These points are more relevant to the construction of the classifier.

Hyperplane

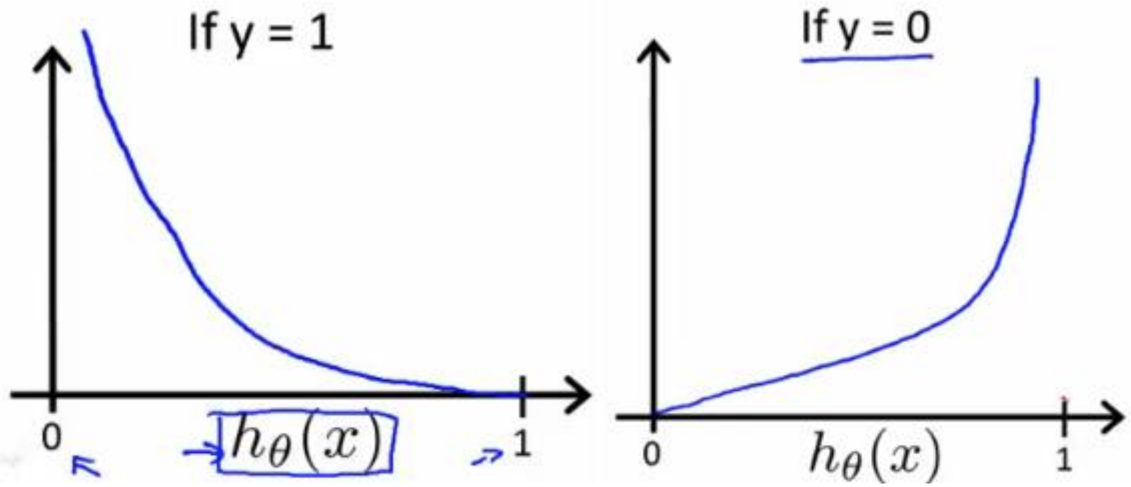
- A hyperplane is a decision plane which separates between a set of objects having different class memberships.

Margin

- A margin is a gap between the two lines on the closest class points. This is calculated as the perpendicular distance from the line to support vectors or closest points. If the margin is larger in between the classes, then it is considered a good margin, a smaller margin is a bad margin.

Classification Metrics - Log Loss

- Each predicted probability is compared to the actual class output value (0 or 1) and a score is calculated that penalizes the probability based on the distance from the expected value. The penalty is logarithmic, offering a small score for small differences (0.1 or 0.2) and enormous score for a large difference (0.9 or 1.0).
- A model with perfect skill has a log loss score of 0.0.(for y=1)
- Cost = -(yact) ln (ypred) - (1-yact) ln (1 - ypred)
- Cost = -ln ypred, where yact = 1
- Cost = -ln (1-ypred), where yact = 0



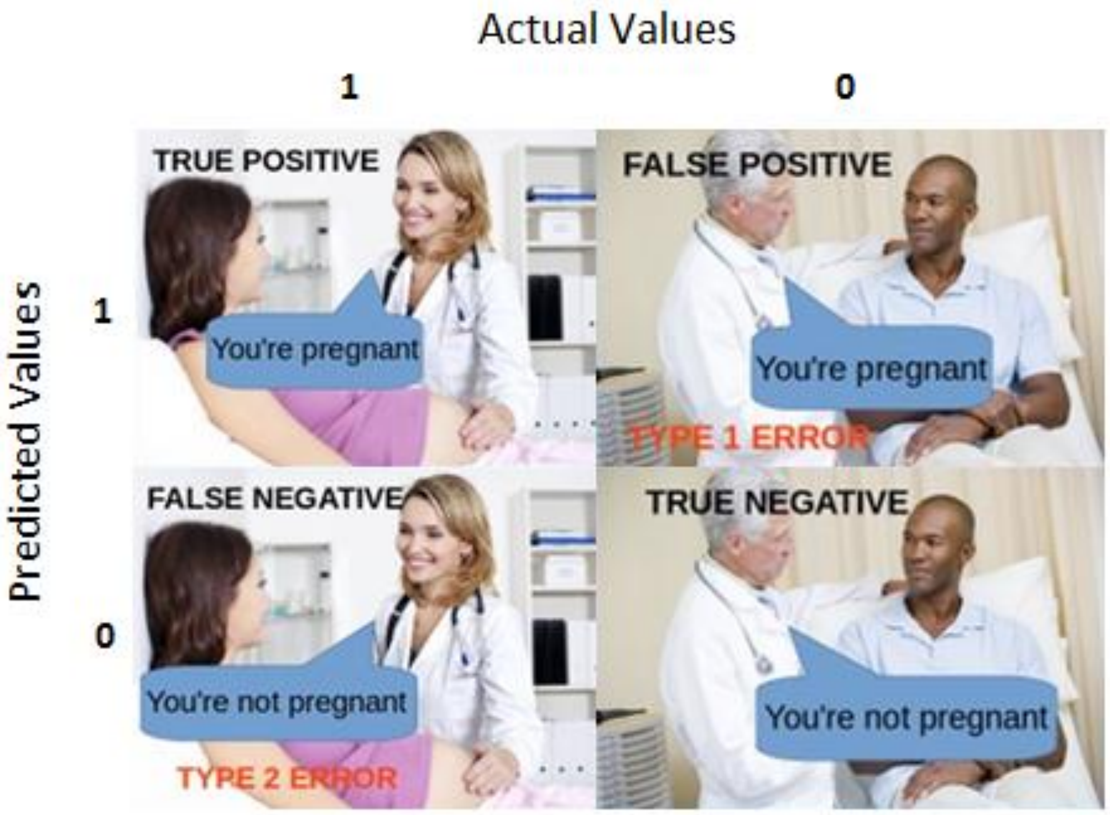
beingdatum
the data society

Confusion Matrix

Useful for measuring recall, precision, specificity, accuracy & AUC-ROC Curve.

False Positive: Type I Error

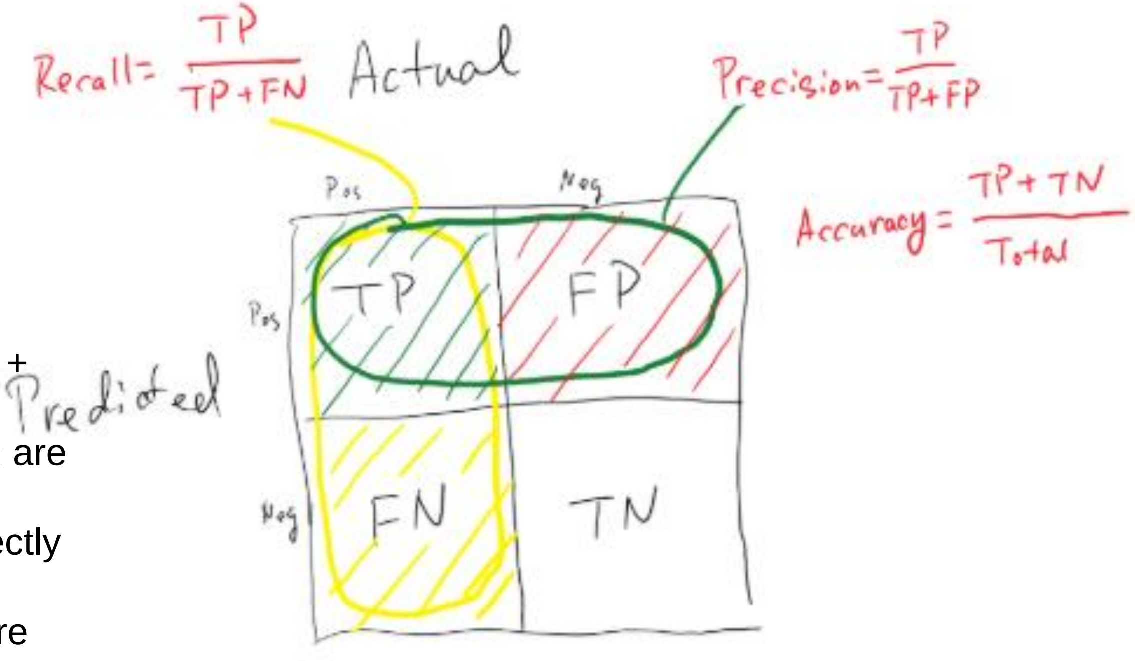
False Negative: Type II Error

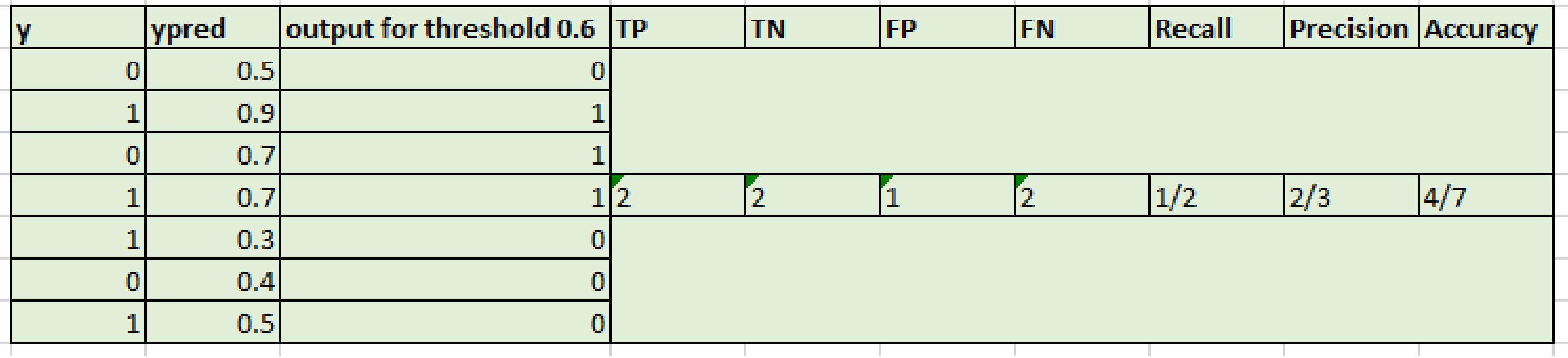


Confusion Matrix

Confusion Matrix metrics

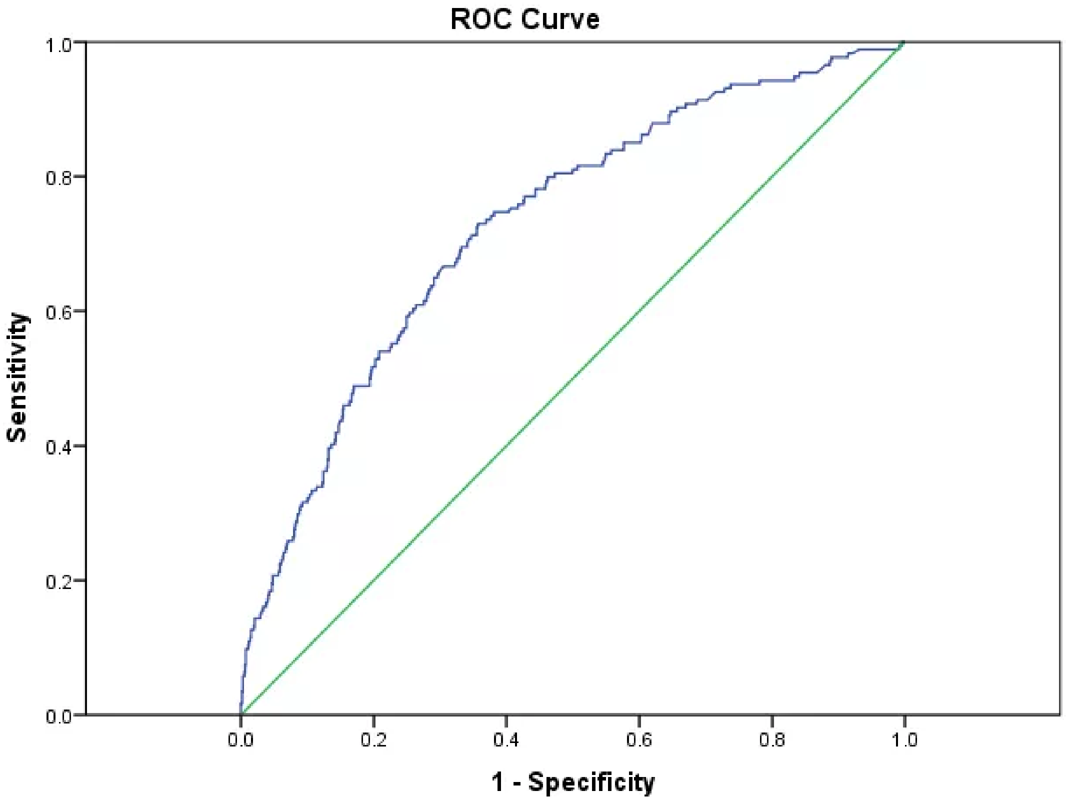
- **Accuracy**-Measures the accuracy of the prediction = (TP + TN) / (TP + TN + FP + FN)
- **Sensitivity**-R-Hit rate or recall(the proportion of actual positives which are correctly identified) = TP / (TP + FN)
- **Specificity**-True negative rate(proportion of negatives which are correctly identified as such) = TN / (TN + FP)
- **Precision** P = TP / (TP + FP) (How many of my predicted positives are actually positives)
- $F1 Statistic = Metric = 2PR/P+R$





y	ypred	output for threshold 0.6	TP	TN	FP	FN	Recall	Precision	Accuracy
0	0.5	0							
1	0.9	1							
0	0.7	1							
1	0.7	1	2	2	1	2	1/2	2/3	4/7
1	0.3	0							
0	0.4	0							
1	0.5	0							

Classification Metrics - Area Under ROC



Diagonal segments are produced by ties.

beingdatum
the data society

Issues in Classification

Bias and Variance

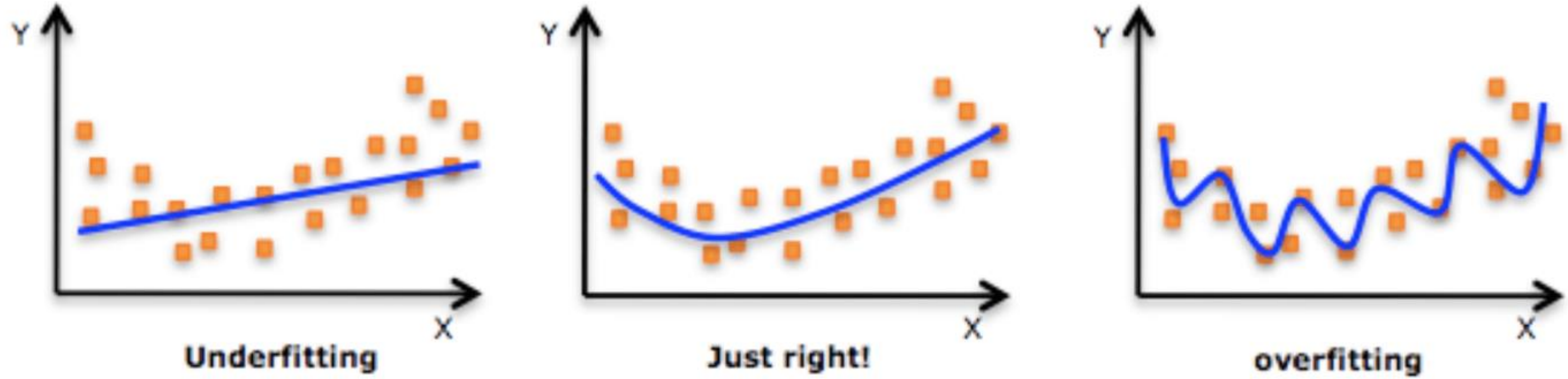
- Bias happens when the model “skews” itself to certain aspects of the predictors, while ignoring others. It is the error between prediction and actuals. Underfitting.
- Variance refers to the stability of a model –Keep predicting consistently for new data sets. It is the variance between predictions for different data sets. Overfitting.

Types of Errors

- In-Sample error is the prediction error when the model is used to predict on the training data set it is built upon.
- Out-of-sample error is the prediction error when the model is used to predict on a new data set.
- Over fitting refers to the situation where the model has very low in-sample error, but very high out-of-sample error. The model has “over fit” itself to the training data.

Class Imbalance Problems

- This is a scenario where the number of observations belonging to one class is significantly lower than those belonging to the other classes.



Underfitting

Just right!

overfitting

Thank you!

QnA

beingdatum
the data society