**PROJECT DOCUMENT**

ARTIFICIAL NEURAL NETWORK

(ANN)

TITLE: **PURCHASING INTENTION PREDICTION**

BY

ANUSHA KP

**AGENDA**

1. INTRODUCTION……………………………………………… 01
2. ANN ARCHITECTURE ……………………………………….. 02
3. ARTIFICIAL NEURONS VS BIOLOGICAL NEURONS……...………………………………………………..03
4. HOW DO ANN LEARNS………………………….……………04
5. TYPES OF ANN………………………...……………………….05
6. ANN PROJECT…………………………………………………..06
7. DIFFERENT PARAMETERS OF PROJECT……………………07
8. SUMMARY………………………………………………………08
9. CONCLUSION…………………………………………………..09
10. REFERENCE……………………………………………………10

**Introduction:**

Artificial Neural Networks (ANN)are the basic algorithms and also simplified methods used in Deep Learning (DL) approach. We have come across more complicated and high-end models in the DL approach. However, ANN is a vital element of the progressive procedure and is the first stage in the DL algorithm.

* Artificial Neural Networks contain artificial neurons which are called ***units***. These units are arranged in ***a series of layers that together constitute the whole Artificial Neural Network in a system.***
* A layer can have only a dozen units or millions of units as this depends on how the complex neural networks will be required to learn the hidden patterns in the dataset. Commonly, Artificial Neural Network has an input layer, an output layer as well as hidden layers.
* The input layer receives data from the outside world which the neural network needs to analyze or learn about.
* Then this data passes through one or multiple hidden layers that transform the input into data that is valuable for the output layer.
* Finally, the output layer provides an output in the form of a response of the Artificial Neural Networks to input data provided.
* In the majority of neural networks, units are interconnected from one layer to another. Each of these connections has weights that determine the influence of one unit on another unit.
* As the data transfers from one unit to another, the neural network learns more and more about the data which eventually results in an output from the output layer.



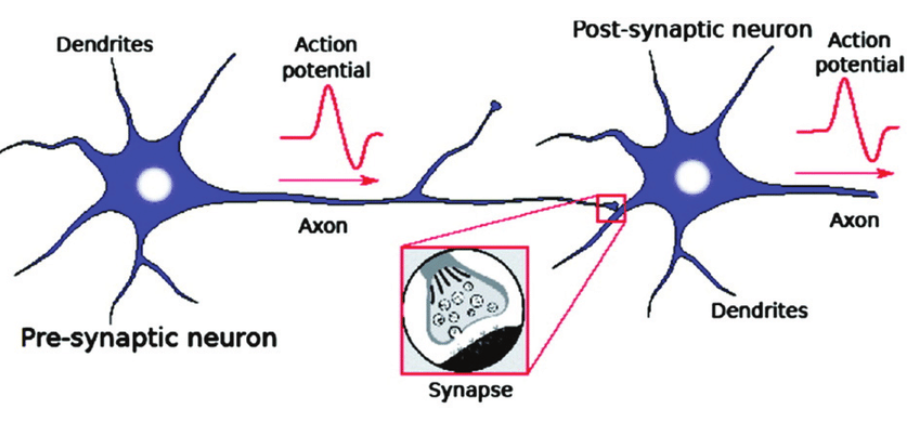
* The structures and operations of human neurons serve as the basis for artificial neural networks. It is also known as neural networks or neural nets.
* The input layer of an artificial neural network is the first layer, and it receives input from external sources and releases it to the hidden layer, which is the second layer.
* In the hidden layer, each neuron receives input from the previous layer neurons, computes the weighted sum, and sends it to the neurons in the next layer.
* These connections are weighted means effects of the inputs from the previous layer are optimized more or less by assigning different-different weights to each input and it is adjusted during the training process by optimizing these weights for improved model performance.

**Artificial Neurons vs Biological Neurons:**

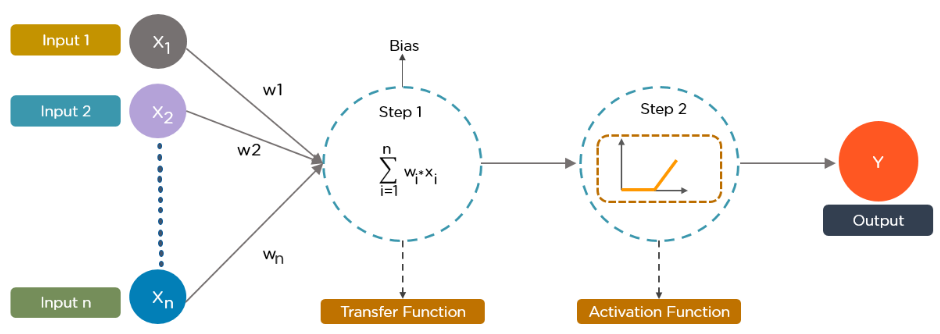
The concept of artificial neural networks comes from biological neurons found in animal brains So they share a lot of similarities in structure and function wise.

* **Structure :** The structure of artificial neural networks is inspired by biological neurons. A biological neuron has a cell body or soma to process the impulses, dendrites to receive them, and an axon that transfers them to other neurons.  The input nodes of artificial neural networks receive input signals, the hidden layer nodes compute these input signals, and the output layer nodes compute the final output by processing the hidden layer’s results using **activation functions.**
* **Synapses**: Synapses are the links between biological neurons that enable the transmission of impulses from dendrites to the cell body. Synapses are the weights that join the one-layer nodes to the next-layer nodes in artificial neurons. The strength of the links is determined by the weight value.
* **Learning**: In biological neurons, learning happens in the cell body nucleus or soma, which has a nucleus that helps to process the impulses. An action potential is produced and travels through the axons if the impulses are powerful enough to reach the threshold. This becomes possible by synaptic plasticity, which represents the ability of synapses to become stronger or weaker over time in reaction to changes in their activity. In artificial neural networks, backpropagation is a technique used for learning, which adjusts the weights between nodes according to the error or differences between predicted and actual outcomes.
* **Activation**: In biological neurons, activation is the firing rate of the neuron which happens when the impulses are strong enough to reach the threshold. In artificial neural networks, A mathematical function known as an activation function maps the input to the output, and executes activations.

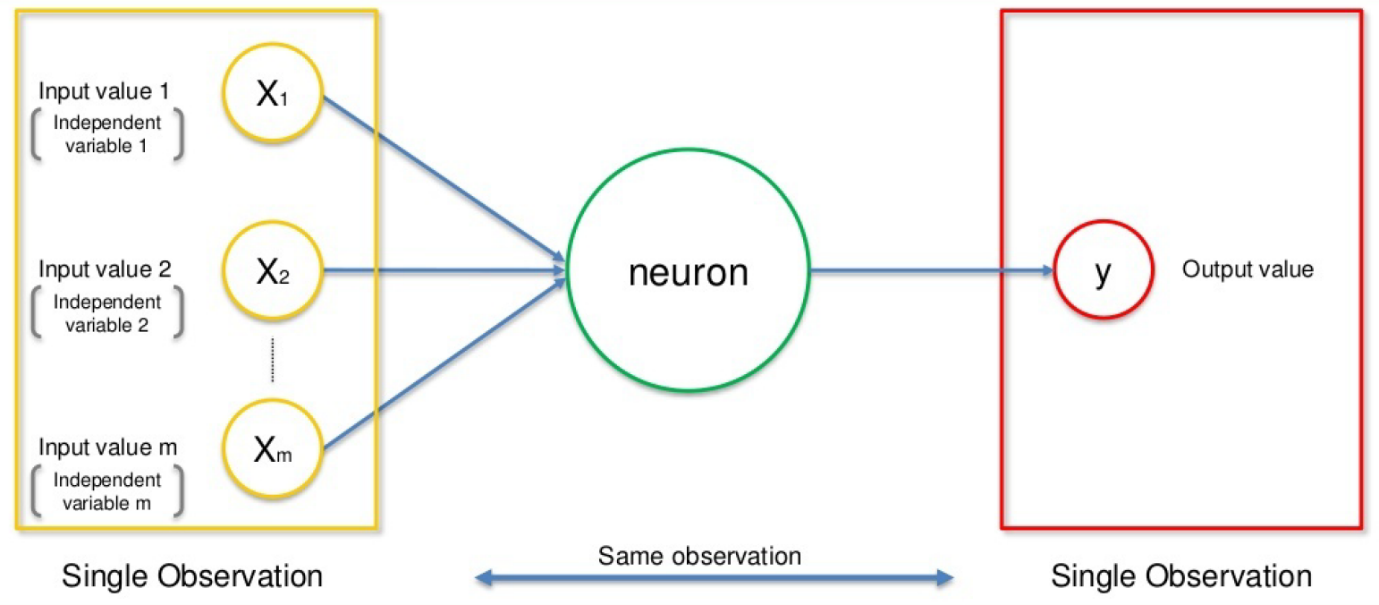
|  |  |
| --- | --- |
| **BIOLOGICAL NEURON** | **ARTIFICIAL NEURON** |
| Dendrite | Inputs |
| Cell nucleus or Soma | Nodes |
| Synapses | Weights |
| Axon | Output |
| Synaptic Plasticity | Backpropagations |

****

**Neural Network – Basic foundation to understand:**

****

Now biological notions, as demonstrated in the previous subject, become a technological approach in this graphical representation above. So let’s assume, x1, x2…xn is considered as multiple neurons (presumed to be multiple sensory) transmitting signal to a signal neuron. Let us assume a scenario for the above diagram, you are going to watch a horror movie with your friends in a theatre, so there is a particular scene where the whole theatre gets erupted for a horror scene and you are having horror phobia (fear of watching horror scene), so for that scene what you do? assume x1 – eyes, x2 – ears, and x3 – mouth are some of the inputs, then what about your outputs be like? You will automatically close your eyes, close your ears and scream more (as like the voice in the movie is not audible) as per the scenario. In short, it can be picturized as,



We have terms like w1, w2, w3 which means weights: Weights are crucial for ANN function because weights are how neural networks (NN) learn, by modifying the weights, the NN decides which signal is significant in each case and which signal is not relevant for neurons, or which signal is passed or which signal is not passed or to which intensity signal is passed, they are the things that can be adapted through the learning method, like when you are training ANN, you are basically adjusting all the weights, and that’s where Gradient  
Descent (GD) and Back Propagation (BP) comes into play, so for the same scenario, you will be having the priority right which part has the highest weightage to process – here by default we close our eyes right. Finally, the dotted line (blue lines) here in the above diagram means Synapse, communication between neurons.

To summarize in short, how neural network takes place:

1. Signals from multiple neurons like x1, x2, x3… xn, along with associated weights w1, w2, w3… wn are transmitted to the successive neurons. (Input layers)

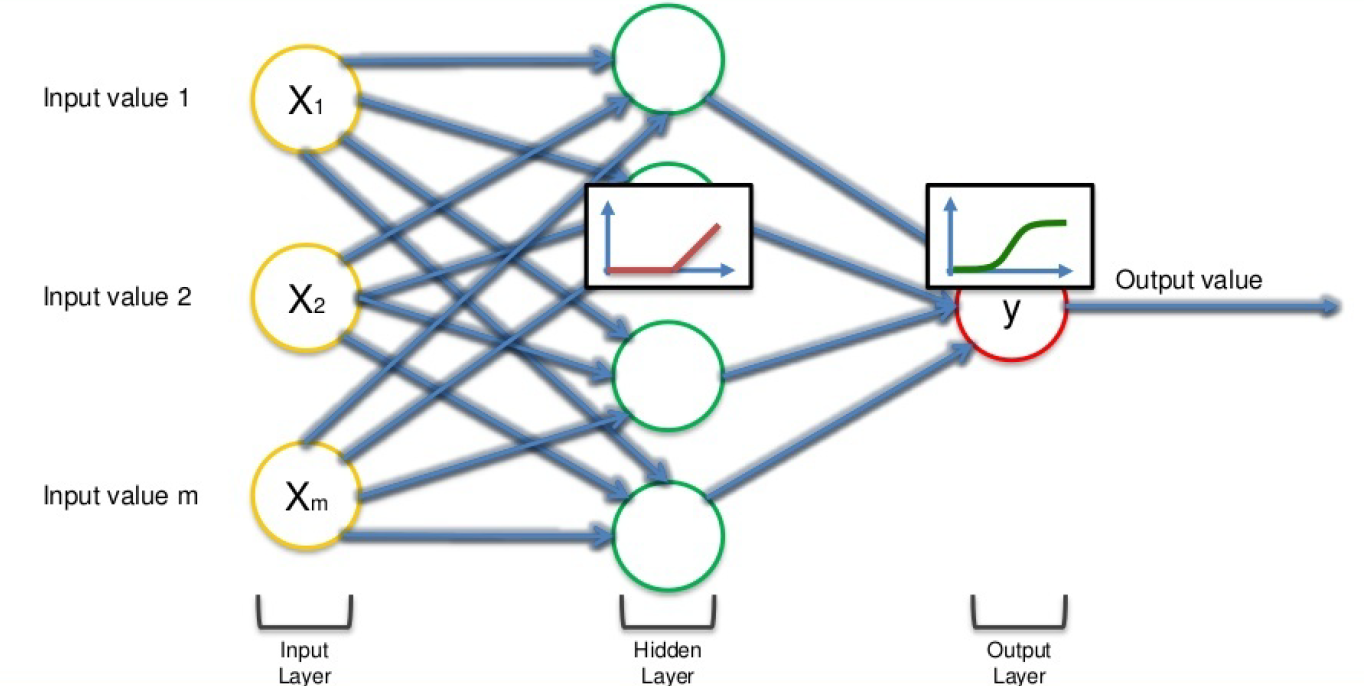
2. So after we are receiving multiple signals (here consider it as neurons), with weights associated with it, we will be summing up all the signals with weights, and that’s our step 1 as shown in the above figure (Hidden layers).

3. Then it applies the activation function as step 2 as shown in the above figure (Hidden layers).

4. From the above step neuron understand if it needs to pass the signal or not.

5. If the neuron passes the signal (that’s to another neuron), that’s our output (Output layers).

6. The above process keeps on repeating until the last neuron or the part of our body here.



Finally, NN is a sequence of algorithms that imitate human brain functions to recognize connections between huge volumes of data. This is the basis for the DL approach, and NN can be categorized as stated below based on the architecture.

**How do Artificial Neural Networks learn?**

* Artificial neural networks are trained using a training set. For example, suppose you want to teach an ANN to recognize a cat.
* Then it is shown thousands of different images of cats so that the network can learn to identify a cat.
* Once the neural network has been trained enough using images of cats, then you need to check if it can identify cat images correctly.
* This is done by making the ANN classify the images it is provided by deciding whether they are cat images or not.
* The output obtained by the ANN is corroborated by a human-provided description of whether the image is a cat image or not.
* If the ANN identifies incorrectly then backpropagation is used to adjust whatever it has learned during training.
* Backpropagation is done by fine-tuning the weights of the connections in ANN units based on the error rate obtained.
* This process continues until the artificial neural network can correctly recognize a cat in an image with minimal possible error rates.

**Types of Artificial Neural Networks:**

**Feedforward Neural Network :**

* The feedforward neural network is one of the most basic artificial neural networks. In this ANN, the data or the input provided travels in a single direction.
* It enters into the ANN through the input layer and exits through the output layer while hidden layers may or may not exist. So the feedforward neural network has a front-propagated wave only and usually does not have backpropagation.

**Convolutional Neural Network :**

* A Convolutional neural network has some similarities to the feed-forward neural network, where the connections between units have weights that determine the influence of one unit on another unit.
* But a CNN has one or more than one convolutional layer that uses a convolution operation on the input and then passes the result obtained in the form of output to the next layer.
* CNN has applications in speech and image processing which is particularly useful in computer vision.

**Modular Neural Network:**

* A Modular Neural Network contains a collection of different neural networks that work independently towards obtaining the output with no interaction between them.
* Each of the different neural networks performs a different sub-task by obtaining unique inputs compared to other networks.
* The advantage of this modular neural network is that it breaks down a large and complex computational process into smaller components, thus decreasing its complexity while still obtaining the required output.

**Radial basis function Neural Network:**

* Radial basis functions are those functions that consider the distance of a point concerning the center.
* RBF functions have two layers. In the first layer, the input is mapped into all the Radial basis functions in the hidden layer and then the output layer computes the output in the next step. Radial basis function nets are normally used to model the data that represents any underlying trend or function.

**Recurrent Neural Network:**

* The Recurrent Neural Network saves the output of a layer and feeds this output back to the input to better predict the outcome of the layer.
* The first layer in the RNN is quite similar to the feed-forward neural network and the recurrent neural network starts once the output of the first layer is computed.
* After this layer, each unit will remember some information from the previous step so that it can act as a memory cell in performing computations.

**Artificial Neural Network project (Purchasing Intention Prediction)**

* The online shoppers intention Prediction dataset is given by a famous online shopping platform to predict whether the visitor of the page will make a purchase or not.
* They faced a problem that most of the people are visiting the page but no proper purchase is made
* The company was worried about it and the marketing team of that platform has introduced, Coupons and discount patterns for the visitors.
* This neural network project aims to predict whether a visitor will make a purchase based on their browsing behaviour and other characteristics. The project uses a neural network algorithm and features such as time spent on page, number of pages viewed, and device type to make predictions.
* In this project, We will explore the Online Shoppers Purchasing Intention. The aim of this project is to understand the data and predict whether a visitor will make a purchase or not.

**2. Objectives**

* Analyze the Online Shoppers Purchasing Intention dataset.
* Identify key features that influence purchasing decisions.
* Build an ANN model to predict purchase likelihood.
* Evaluate model performance and improve accuracy.

**3. Dataset Description** The dataset, "Online Shoppers Intention," contains records of user interactions on an e-commerce platform. The key attributes include:

* **Administrative, Administrative\_Duration**: Number of administrative pages visited and time spent.
* **Informational, Informational\_Duration:** Number of informational pages visited and time spent.
* **ProductRelated, ProductRelated\_Duration:** Number of product-related pages visited and time spent.
* **Bounce Rates, Exit Rates, Page Values:** Website engagement metrics.
* **Special Day:** Indicates proximity to special occasions.
* **Month:** The month of the visit.
* **Operating Systems, Browser, Region, Traffic Type:** Technical user details.
* **Visitor Type:** New or returning visitor.
* **Weekend:** Whether the session occurred on a weekend.
* **Revenue (Target Variable):** Indicates whether a purchase was made (1) or not (0).

**4. Methodology**

* **Data Preprocessing**: Handling missing values, encode categorical features, and normalize numerical data.
* **Exploratory Data Analysis (EDA)**: Identifying trends, correlations, and feature importance.
* **Model Development**:
  + Defining an ANN architecture using TensorFlow/Keras.
  + Using appropriate activation functions, layers, and optimization techniques.
* **Model Training & Evaluation**:
  + Split data into training and testing sets.
  + Train the ANN model using cross-validation.
  + Evaluate performance using accuracy

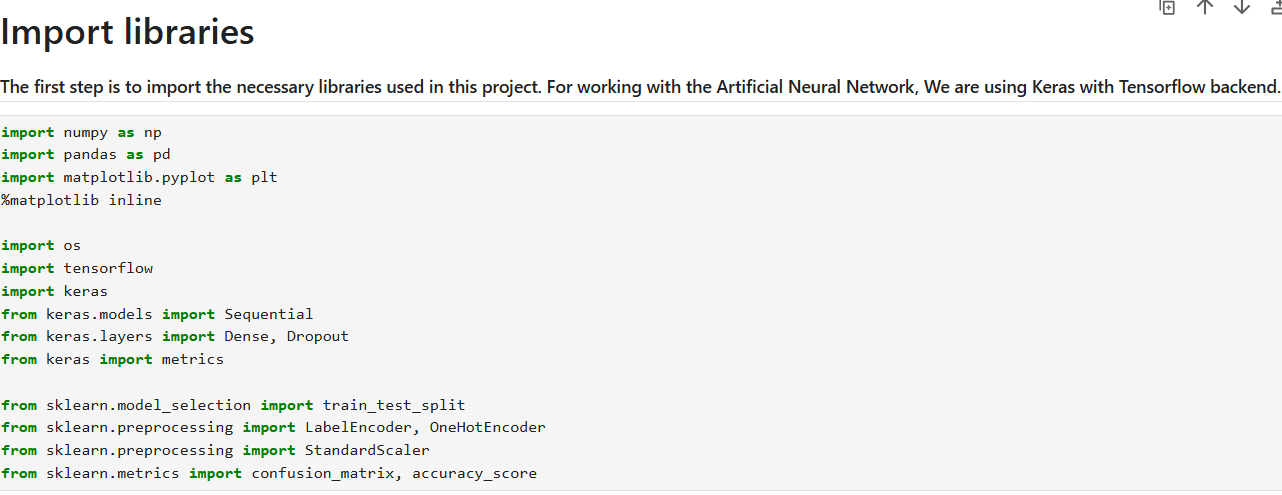
**6. Expected Outcome**

* A well-trained ANN model capable of accurately predicting whether a user will make a purchase.
* Insights into key factors influencing online shopping decisions.

**STEP BY STEP ANALYSIS OF THE PROJECT:**

**IMPORTING ALL THE NECESSARY LIBRARIES**

In this step we are going to import all the necessary libraries which is required.

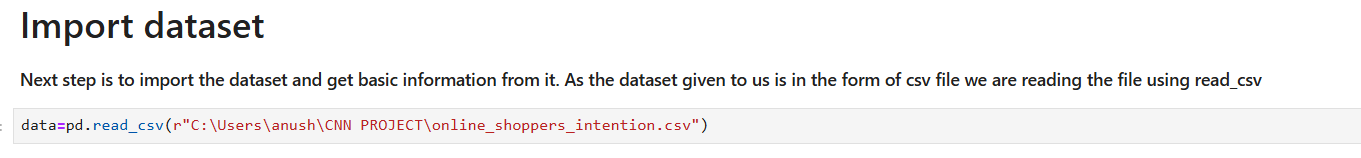


**IMPORTING DATASET:**

In this step we are going to import the dataset. The dataset is given in the form of csv file

1. Dataset Overview

* Total rows: 12,330
* Total columns: 18
* Target Variable: Revenue (Boolean → True = Purchase, False = No Purchase)
* Features include:
  + Numerical: Page views, durations, bounce rates, exit rates, etc.
  + Categorical: Month, visitor type, weekend

**Why ANN is Suitable:**

* The dataset has multiple features influencing purchasing behavior, which can be complex and nonlinear.
* Continuous numerical features like Bounce Rates, Exit Rates, Page Values are well-suited for ANN learning.
* Categorical features like Month, Visitor Type can be one-hot encoded for ANN input.

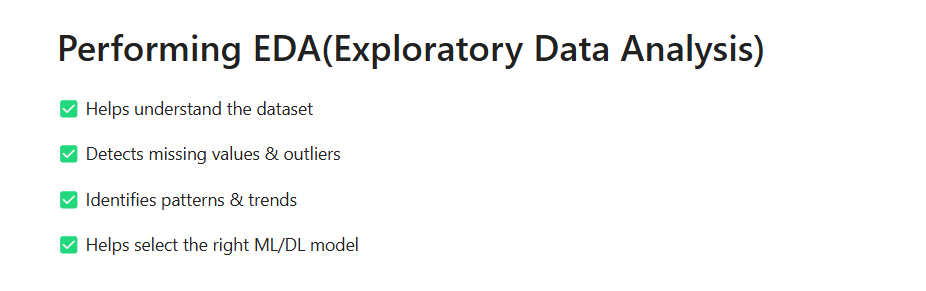
**Capturing Complex Customer Behavior**

* Customer purchasing decisions are influenced by many nonlinear factors (e.g., time spent on pages, bounce rates, past visits).
* ANN can learn hidden patterns and relationships that traditional models like Logistic Regression might miss.

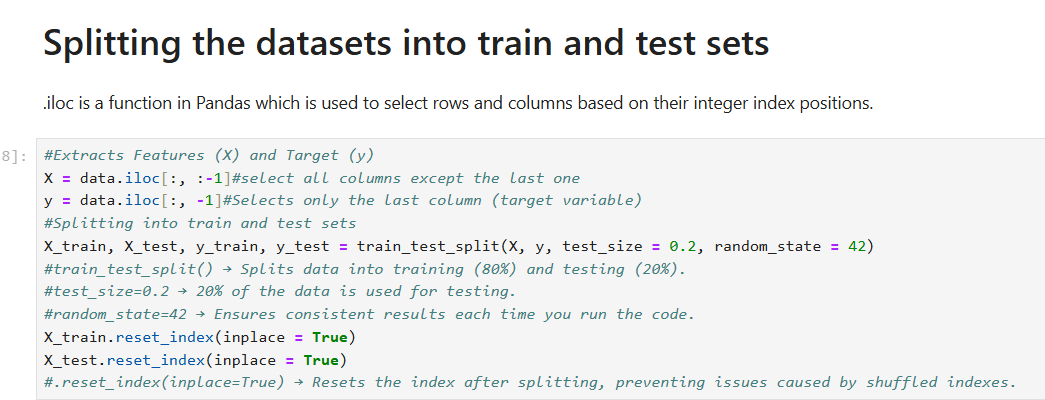
Example:

* A customer with low bounce rate & high page value may be likely to purchase.
* But a simple model might not see that a user with many short visits but high page value later also buys.

**PERFORMING EDA (EXPLORATORY DATA ANALYSIS)**



**SPLITTING DATA SETS INTO TRAIN AND TEST SETS:**

****

**DATA ANALYSIS AND VISUALISATION :**

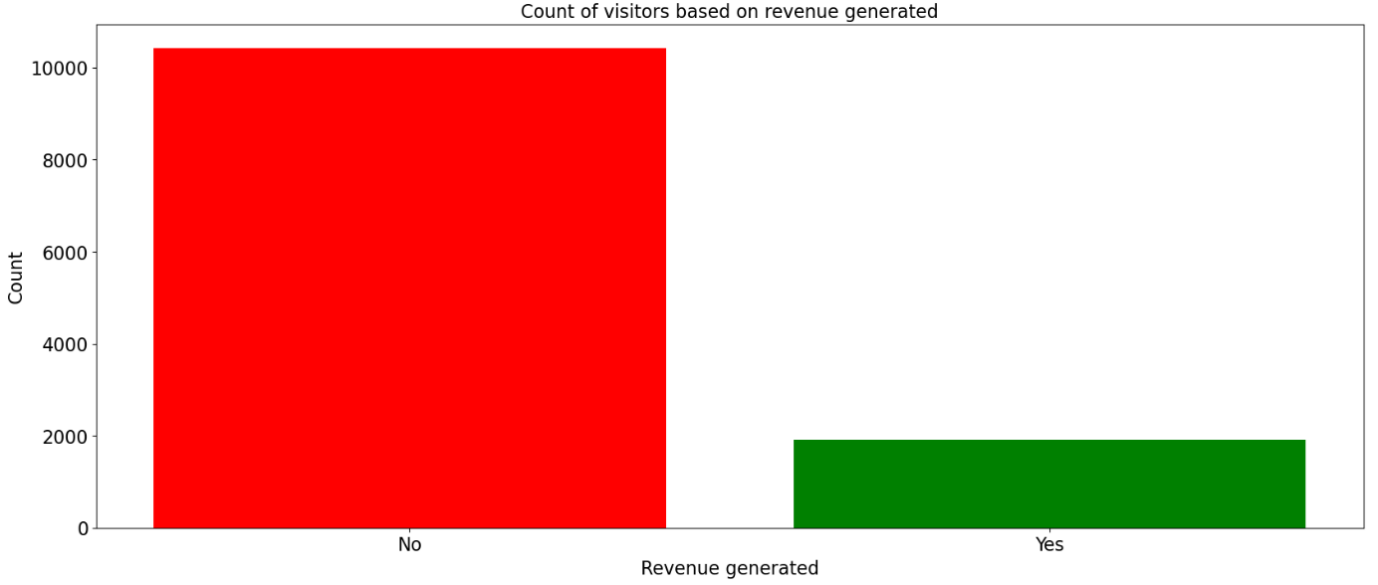
\*The first and foremost step is to look at the distribution of data between the two classes

\* As is clear from the bar plot below, the dataset includes majority of feature values that resulted in no revenue generation.

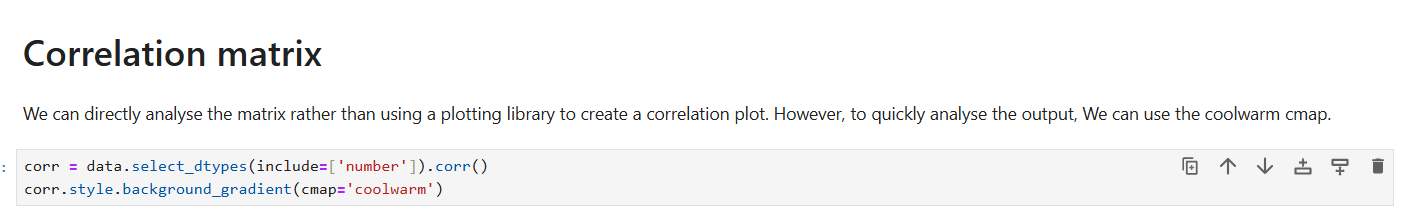
\* The dataset is highly imbalanced and this is where we'd have to create an efficient model that can still classify between the two classes.

\* We are going to use SMOTE for balancing the classes before building the model.

\* Baed on the target variable revenue we are classifying the customers into two revenue generated and non revenue generated

****

**CORRELATION MATRIX:**

****

**Understanding Relationships Between Variables**

* The correlation matrix helps identify how strongly and in what direction (positive or negative) two variables are related.
* Correlation values range from -1 to 1:
  + 1: Perfect positive correlation (as one increases, the other increases).
  + 0: No correlation (no relationship between the variables).
  + -1: Perfect negative correlation (as one increases, the other decreases).

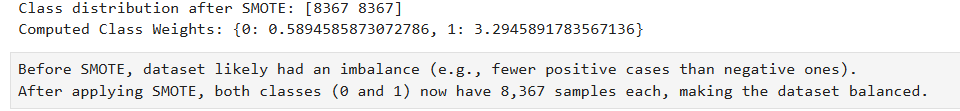
**This helps in:**

* Identifying relationships between features.
* Detecting multicollinearity (highly correlated features).
* Deciding which features to keep or remove in machine learning models.
* **Dark Red (Near +1.0):** Strong positive correlation (both variables increase together).
* **Dark Blue (Near -1.0):** Strong negative correlation (one increases while the other decreases).
* **Near 0.0:** Weak or no correlation.

**DATA ENGINEERING:**

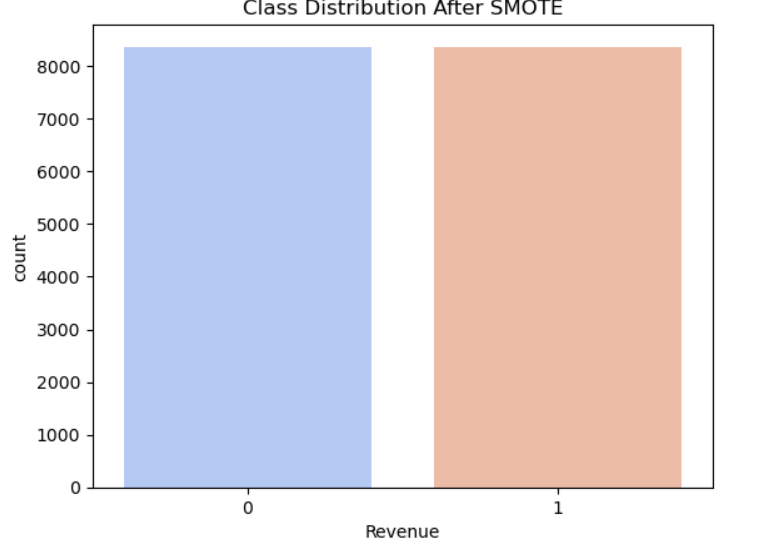
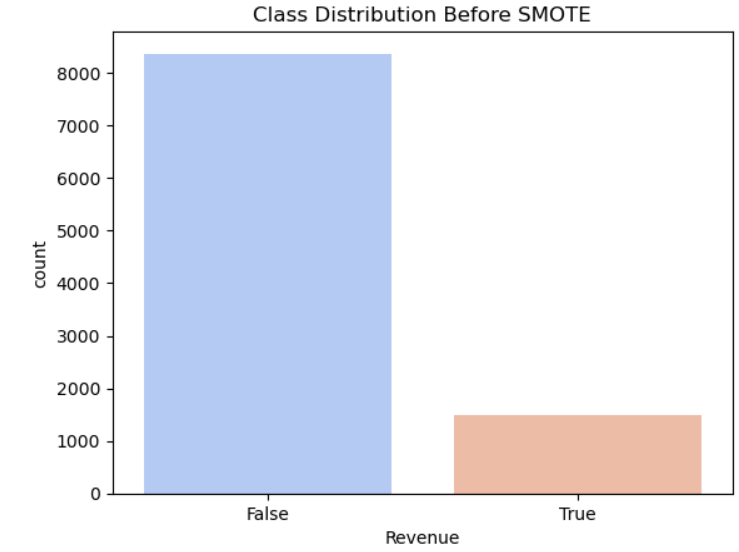
* We can remove columns that appear to be redundant and provide no useful information towards the target class. We will begin by dropping the count of each page visited.
* Next, We can use Label Encoder and OneHotEncoder on categorical columns FOR ENCODING
* Once the labels are encoded, We can use OneHotEncoder to create separate columns for each class in the columns then drop the first column for each class and append the new columns to train and test data, once the original columns are removed.
* After removing some columns, We can now scale the columns to have mean 0.
* Standard Scala**r** to scale the training and test datasets.

**APPLYING SMOTE FOR CLASS BALANCING:**



Before SMOTE, dataset likely had an imbalance (e.g., fewer positive cases than negative ones).

After applying SMOTE, both classes (0 and 1) now have 8,367 samples each, making the dataset balanced.



**Model Generation**

Creating the Artificial Neural Network now. network is created with many layers.

| **Layer Type** | **Output Shape** | **Parameters** | **Explanation** |
| --- | --- | --- | --- |
| **Dense (128 neurons)** | (None, 128) | **4,224** | First hidden layer with 128 neurons, likely using ReLU activation. |
| **Dropout (0.?)** | (None, 128) | **0** | Prevents overfitting by randomly dropping neurons during training. |
| **Dense (128 neurons)** | (None, 128) | **16,512** | Second hidden layer, possibly ReLU activation. |
| **Dropout (0.?)** | (None, 128) | **0** | Again, dropout for regularization. |
| **Dense (256 neurons)** | (None, 256) | **33,024** | Third hidden layer with 256 neurons, increasing network complexity. |
| **Dropout (0.?)** | (None, 256) | **0** | Regularization layer. |
| **Dense (128 neurons)** | (None, 128) | **32,896** | Fourth hidden layer, reducing dimensions again. |
| **Dropout (0.?)** | (None, 128) | **0** | Regularization. |
| **Dense (1 neuron, output layer)** | (None, 1) | **129** | **Final output layer**, likely using **sigmoid activation** for binary classification. |

**Training and Evaluating the Model:**

We can now train the model with the train data and the validation data as 10% of the train data.

We are using 30 epochs as the dataset is the medium one

**Detecting the confusion matrix :**

**Confusion matrix:**

**[[1980 75]**

**[ 200 211]]**

**True Positives (1980):**

* model correctly predicts most of the buyers.

**True Negatives (75):**

* The model struggles to correctly classify non-buyers.

**False Positives (200):**

* 200 users were predicted to buy but didn't.

**False Negatives (211):**

* 211 users actually bought but were predicted not to buy.

**Accuracy turns out to be 88.85%**

**Result analysis**[**¶**](http://localhost:8888/notebooks/Downloads/Capstone_Purchasing_Intention_Prediction.ipynb#Result-analysis)

The confusion matrix reveals that we can able to identify both the type of visitors, visitors who are going to generate revenue and visitors who are not going to generate revenue. We can use this information as follows:

1. Once we can able to identify that the visitors are going to generate revenue, then we do not want to provide any coupons, special prizes and all rather we can give the **visitors special points** **which they can use the next time when they visit.** (Using the special points they will get best offers which makes them to stay in the loop of buying products from us)
2. The visitors that are unlikely to make a purchase can be provided **with discount** coupons so that they are more likely to make a purchase.

**SUMMARY:**

**What is ANN:**

An Artificial Neural Network (ANN) is a machine learning model inspired by the way the human brain processes information. It consists of layers of neurons (nodes) that learn to recognize patterns in data.

**ANN Architecture**

An ANN typically consists of three main types of layers:

1. **Input Layer**

* **Receives raw data** (e.g., images, text, numbers).
* Each **neuron** in this layer represents a feature (e.g., pixel values for images).
* **No computations** happen here, it only **passes data** to the next layer.

1. **Hidden Layers**

* Perform mathematical **transformations on data** using **weights** and **activation functions**.
* Can have **one or more** hidden layers. More layers = **deeper network** (**Deep Learning**).
* Activation functions like **ReLU, Sigmoid, and Tanh** introduce **non-linearity**.

1. **Output Layer**

* Generates the final **prediction** or classification.
* Uses **activation functions** based on the task:
  + **Sigmoid** → For **binary classification** (e.g., Spam or Not Spam).
  + **Softmax** → For **multi-class classification** (e.g., Dog, Cat, Bird).

**How ANN works:**

* **Input Data** → Passes through the network.
* **Weights & Biases Apply Transformations** → Each neuron computes weighted sums.
* **Activation Function** → Decides whether neurons "fire" or stay inactive.
* **Forward Propagation** → Data moves forward through layers.
* **Loss Calculation** → Measures how far the output is from the correct answer.
* B**ackpropagation** → Adjusts weights using **gradient descent** to reduce errors.
* **Repeat Training** → Network learns over many iterations (**epochs**) until optimal performance.

**Where are ANNs Used?**

✅ **Image Recognition** (Face detection, medical imaging)

✅ **Speech Recognition** (Alexa, Siri, Google Assistant)

✅ **NLP (Natural Language Processing)** (Chatbots, text classification)

✅ **Stock Market Prediction** (Financial forecasting)

✅ **Self-Driving Cars** (Object detection, navigation)

**Why Do We Need ANNs?**

✅ **Solves complex problems** → Traditional algorithms struggle with unstructured data like images, text, and speech.

✅ **Learns from data** → Unlike rule-based systems, ANNs learn patterns automatically.

✅ **Generalizes well** → Can make predictions on unseen data after training.

**REFERENCES:**

[**https://archive.ics.uci.edu/dataset/468/online+shoppers+purchasing+intention+dataset-**](https://archive.ics.uci.edu/dataset/468/online+shoppers+purchasing+intention+dataset-) **dataset , Online purchase intention prediction**

[**https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/**](https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/) **Understanding artificial neural network and its applications**

[**https://www.sciencedirect.com/topics/biochemistry-genetics-and-molecular-biology/artificial-neural-network**](https://www.sciencedirect.com/topics/biochemistry-genetics-and-molecular-biology/artificial-neural-network) **- deep understanding of ANN**

[**https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/**](https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/)

**smote analysis for class imbalance**

[**https://medium.com/@hadican/how-to-build-a-simple-artificial-neural-network-ann-a064939f940b**](https://medium.com/@hadican/how-to-build-a-simple-artificial-neural-network-ann-a064939f940b) **How to build an ann model**