A Project Report

on

Employee Churn Model

Submitted in partial fulfillment of the requirement for the award of the degree of

Bachelor of Technology in Computer Science and Engineering

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Dr. Kavita Saini
Associate Professor
Department of Computer Science and Engineering

Submitted By

18SCSE1180037 - Shivansh Sharma
18SCSE1180041 - Utsav Sarkar

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA, INDIA

DECEMBER - 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

We hereby certify that the work which is being presented in the project, entitled **"EMPLOYEE CHURN MODEL"** in partial fulfillment of the requirements for the award of the **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING** submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of JULY-2021 to DECEMBER-2021, under the supervision of Dr. Kavita Saini, Associate Professor, Department of Computer Science and Engineering of School of Computing Science and Engineering , Galgotias University, Greater Noida .

The matter presented in the project has not been submitted by us for the award of any other degree of this or any other places.

18SCSE1180037 – Shivansh Sharma
18SCSE1180041 – Utsav Sarkar

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor
Dr. Kavita Saini, Associate Professor

CERTIFICATE

The Final Project Viva-Voce examination of 18SCSE1180037 – Shivansh Sharma 18SCSE1180041 – Utsav Sarkar has been held on **21ST December 2021** and their work is recommended for the award of BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING.

Signature of Examiner(s)

Signature of Supervisor(s)

Signature of Dean

Dean School of Computing Science
& Engineering
GALGOTIAS UNIVERSITY
* UTTAR PRADESH *

Signature of Project Coordinator

Date: December, 2021

Place: Galgotias University

Abstract

Nowadays companies want a machine learning model that can predict their employee retentions based on the HR inputs. Large companies contain thousands of employees working for them so taking care of their needs and satisfaction of each employee is a challenging task for the HR department.

This paves way for the departure of talented and important employees leaving the company without giving general reasons. To overcome this the companies, need an end-to-end machine learning model to predict the retention of employees within an organization.

For this we use the data of previous employees who have worked for the company and, by finding a pattern in it we predict the retentions in the form of yes or no.

There are a few data input points like satisfaction, last evaluation, number of projects done by the employee, work accidents, promotion in the last five years and salary which we will use to train the model.

The essential goal of this project is to anticipate the odds of leaving the company by an employee. This project center's around dissecting the churn forecast methods to distinguish the churn conduct and approve the explanations behind employee churn advantages.

Table of Contents

List of Table

List of Figures

Acronyms

ANN	Artificial Neural Networks
BRNN	Bidirectional Recurrent Neural Network
CNN	Convolutional Neural Networks
ELU	Exponential Linear Unit
GAP	Global Average Pooling
HMM	Hidden Markov Models
POS	Part-of-Speech Tagging
RF	Random Forest
RL	Reinforcement Learning
ReLU	Rectified Linear Unit
SVM	Support Vector Machine
NLP	Natural Language Processing
PCA	Principal Component Analysis

CHAPTER-1

Introduction

Churn model is a mathematical representation of how churn impacts your business. Churn calculations are built on existing data (the number of employees who left your service during a given time period). A predictive churn model extrapolates on this data to show future potential churn rates.

Employee churn (also known as employee attrition) refers to when an employee (player, subscriber, user, etc.) ceases his or her relationship with a company. Online businesses typically treat an employee as churned once a particular amount of time has elapsed since the employee's last interaction with the site or service. The full cost of churn includes both lost revenue and the marketing costs involved with replacing those employees with new ones. Reducing churn is a key business goal of every online business.

In order to succeed at retaining employees who would otherwise abandon the business, marketers and retention experts must be able to-

(a) predict in advance which employees are going to churn through churn analysis

(b) know which marketing actions will have the greatest retention impact on each particular employee. Armed with this knowledge, a large proportion of employee churn can be eliminated.

While simple, in theory, the realities involved with achieving this "proactive retention" goal are extremely challenging.

Client churn prediction has accumulated a more noteworthy premium in business particularly in companies. Numerous creators have introduced various forms of churn forecast models enormously dependent on the information mining ideas utilizing AI and meta-heuristic calculations. The point of this paper is to concentrate on the absolute most significant churn forecast methods created over the new year's. The essential goal of this venture is to foresee the odds of leaving the company by a client. This paper centers around breaking down the churn forecast methods to distinguish the churn conduct and approve the explanations behind client churn. This paper sums up the churn forecast procedures to have a more profound comprehension of the client churn and it shows that the most exact churn expectation is given by the cross-breed models as opposed to single calculations so telecom enterprises become mindful of the necessities of high danger clients and improve their administrations to upset the churn choice.

1.2 Formulation of Problem

The ability to predict that a particular employee is at a high risk of churning, while there is still time to do something about it, represents a huge additional potential revenue source for every online business. Besides the direct loss of revenue that results from an employee abandoning the business, the costs of initially acquiring that employee may not have already been covered by the employee's spending to date. (In other words, acquiring that employee may have actually been a losing investment.) Furthermore, it is always more difficult and expensive to acquire a new employee than it is to retain a current paying employee.

Unfortunately, most of the churn prediction modelling methods rely on quantifying risk based on static data and metrics, i.e., information about the employee as he or she exists right now. The most common churn prediction models are based on older statistical and data-mining methods, such as logistic regression and other binary modelling techniques. These approaches offer some value and can identify a certain percentage of at-risk employees, but they are relatively inaccurate and end up leaving money on the table.

Problem Statement

In companies, there are a lot of data in their database. So, in some interval of time, they have to increase the size of the database.
But the problem is that many people may leave the company and still their data would be present in the database. It's just a waste of memory.
Churn is defined as the movement of the employee from one company to other. The reason can be: -

1. Services Offered
2. Pay Range
3. Employee-friendly staff
4. Location(where the company is situated)
5. Technologies Available

Solution

With the help of machine learning models, we can easily analyze the data and further we can predict the probability of an employee leaving the company.

In this problem we can use supervised learning techniques like regression, classification etc.

1.2.1 Tools and Technology Used

We have used several technologies and made use of libraries like Tensorflow, Pandas, Numpy and Seaborn and also by using various machine learning platforms like Kaggle, we were easily able to compute whatever was needed and will be needed in the future for this project.

Using various machine learning models available in the market we could accurately predict the probability when an employee would leave the company.

Our machine learning algorithm's space efficiency would lessen calculations and decrease the AI node probability run-time.

An artificial neural network (ANN) is the piece of a figuring structure planned to duplicate the way in which the human frontal cortex separates and measures information. It is the foundation of man-made awareness (AI) and handles gives that would show inconceivable or problematic human or real standards. ANNs cause them to learn capacities that enable them to convey better results as more data opens up. Artificial neural networks are gathered like the human cerebrum, with neuron centres interconnected like a web[1]. The human brain has a large number of cells called neurons. Each neuron is made out of a cell body that is liable for dealing with information by means of passing on data towards (inputs) and away (yields) from the brain. During the planning and regulatory stage, the ANN is determined what to search for and what its yield ought to be, utilizing yes/no-solicitation types with twofold numbers.

CHAPTER-2

Literature Survey

M.A.H. Farquad [1] proposed a hybrid approach to overcome the drawbacks of general SVM model which generates a black box model (i.e., it does not reveal the knowledge gained during training in human understandable form). The hybrid approach contains three phases: In the first phase, SVM-RFE (SVM-recursive feature elimination) is employed to reduce the feature set. In the second phase, dataset with reduced features is then used to obtain SVM model and support vectors are extracted. In the final phase, rules are then generated using Naive Bayes Tree (NBTree which is combination of Decision tree with naive Bayesian Classifier). The dataset used here is company credit card employee dataset (Business Intelligence Cup 2004) which is highly unbalanced with 93.24% loyal and 6.76% churned employees. The experimental showed that the model does not scalable to large datsets. Chih-Fong Tsai [5] introduced the hybrid neural networks techniques to predict the employee churners in a CRM dataset provided by American telecom companies. Here, they built two hybrid models by combining two different neural network techniques like back-propagation artificial neural networks (ANN) and self organizing maps (SOM) for churn prediction.

Ssu-Han Chen [2] used a novel mechanism based on the gamma Cumulative SUM (CUSUM) chart in which the gamma CUSUM chart monitors individual employee's Inter Arrival Time (IAT) by introducing a finite mixture model to design the reference value and decision interval of the chart and used a hierarchical Bayesian model to capture the heterogeneity of employees. Recency, another time interval variable which is complementary to IAT, is combined into the model and tracks the recent status of the login behavior. In addition, benefits from the basic nature of control charts, the graphical interface for each employee is an additional advantage of the proposed method. The results showed that the accuracy rate (ACC) for gamma CUSUM chart is 5.2% higher than exponential CUSUM and the Average Time to Signal (ATS) is about two days longer than required for exponential CUSUM.

So, in general, the performance of rotation-based ensemble classifier depends upon: (i) the performance criteria used to measure classification performance and (ii) the implemented feature extraction algorithm.

Chapter 3

Functionality/Working of Project

Implementation

Model Selection -

For the second part of our project, we chose 2 models which are-

1. **XGBoost**

This library is laser-focused on computational speed and model performance, as such there are few frills. Nevertheless, it does offer a number of advanced features.

The implementation of the model supports the features of the scikit-learn and R implementations, with new additions like regularization. Three main forms of gradient boosting are supported:

Gradient Boosting algorithm also called gradient boosting machine including the learning rate.

Stochastic Gradient Boosting with sub-sampling at the row, column and column per split levels.

Regularized Gradient Boosting with both L1 and L2 regularization.

The two reasons to use XGBoost are also the two goals of the project:

1.Execution Speed

2.Model Performance.

1. Random Forest Classifier

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is controlled with the max_samples parameter if bootstrap=True (default), otherwise the whole dataset is used to build each tree.

Random forest is a supervised learning algorithm. The "forest" it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models increases the overall result.

Put simply: random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.

One big advantage of random forest is that it can be used for both classification and regression problems, which form the majority of current machine learning systems.

Random forest has nearly the same hyperparameters as a decision tree or a bagging classifier. Fortunately, there's no need to combine a decision tree with a bagging classifier because you can easily use the classifier-class of random forest. With random forest, you can also deal with regression tasks by using the algorithm's regressor.

Random forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity that generally results in a better model.

Therefore, in random forest, only a random subset of the features is taken into consideration by the algorithm for splitting a node. You can even make trees more random by additionally using random thresholds for each feature rather than searching for the best possible thresholds.

Table classifying different supervised machine learning algorithms based on accuracy

Algorithm	Time	Correctly Classified	Incorrectly Classified
SVM	0.09 sec	77.34%	22.66%
Naïve Bayes	0.03 sec	76.30%	23.70%
Neural Networks (Perceptron)	0.81 sec	75.13%	24.87%
Random Forest	0.55 sec	74.74%	25.26%
JRip	0.19 sec	74.48%	25.52%
Decision Tree (J48)	0.14 sec	73.83%	26.17%
Decision Table	0.23 sec	72.40%	27.60%

The Importance of Predicting Employee Churn

The ability to predict that a particular employee is at a high risk of churning, while there is still time to do something about it, represents a huge additional potential revenue source for every online business. Besides the direct loss of revenue that results from an employee abandoning the business, the costs of initially acquiring that employee may not have already been covered by the employee's spending to date. (In other words, acquiring that employee may have actually been a losing investment.) Furthermore, it is always more difficult and expensive to acquire a new employee than it is to retain a current paying employee.

Reducing Employee Churn with Targeted Proactive Retention

In order to succeed at retaining employees who would otherwise abandon the business, marketers and retention experts must be able to

(a) Predict in advance which employees are going to churn through churn analysis.

(b) Know which marketing actions will have the greatest retention impact on each particular employee. Armed with this knowledge, a large proportion of employee churn can be eliminated.

While simple, in theory, the realities involved with achieving this "proactive retention" goal are extremely challenging.

The Difficulty of Predicting Churn

Churn prediction modelling techniques attempt to understand the precise employee behaviors and attributes that signal the risk and timing of employee churn. The accuracy of the technique used is obviously critical to the success of any proactive retention efforts. After all, if the marketer is unaware of an employee about to churn, no action will be taken for that employee. Additionally, special retention-focused offers or incentives may be inadvertently provided to happy, active employees, resulting in reduced revenues for no good reason.

Unfortunately, most of the churn prediction modelling methods rely on quantifying risk based on static data and metrics, i.e., information about the employee as he or she exists right now. The most common churn prediction models are based on older statistical and data-mining methods, such as logistic regression and other binary modelling techniques. These approaches offer some value and can identify a certain percentage of at-risk employees, but they are relatively inaccurate and end up leaving money on the table.

A Better Churn Prediction Model

Optimove uses a newer and far more accurate approach to employee churn prediction: at the core of Optimove's ability to accurately predict which employees will churn is a unique method of calculating employee lifetime value for each and every employee. The LTV forecasting technology built into Optimove is based on advanced academic research and was further developed and improved over a number of years by a team of first-rate PhDs and software developers. This method is battle-tested and proven as an accurate and effective approach in a wide range of industries and employee scenarios.

Without revealing too much about the "secret sauce" of Optimove's employee churn prediction technology, the approach combines continual dynamic micro-segmentation and a unique, mathematically intensive predictive behaviour modelling system. The former intelligently and automatically segments the entire employee base into a hierarchical structure of ever-smaller behaviouraldemographic segments. This segmentation is dynamic and updated continually based on changes in the data. The latter is based on the fact that the behaviour patterns of individual employees frequently change over time. In other words, the “segment route history” of each employee is an extremely important factor in determining when and why the employee may churn.

By merging the most exacting micro-segmentation available anywhere with a deep understanding of how employees move from one micro-segment to another over time – including the ability to predict those moves before they occur – an unprecedented degree of churn analysis accuracy is attainable.

REQUIREMENT ANALYSIS

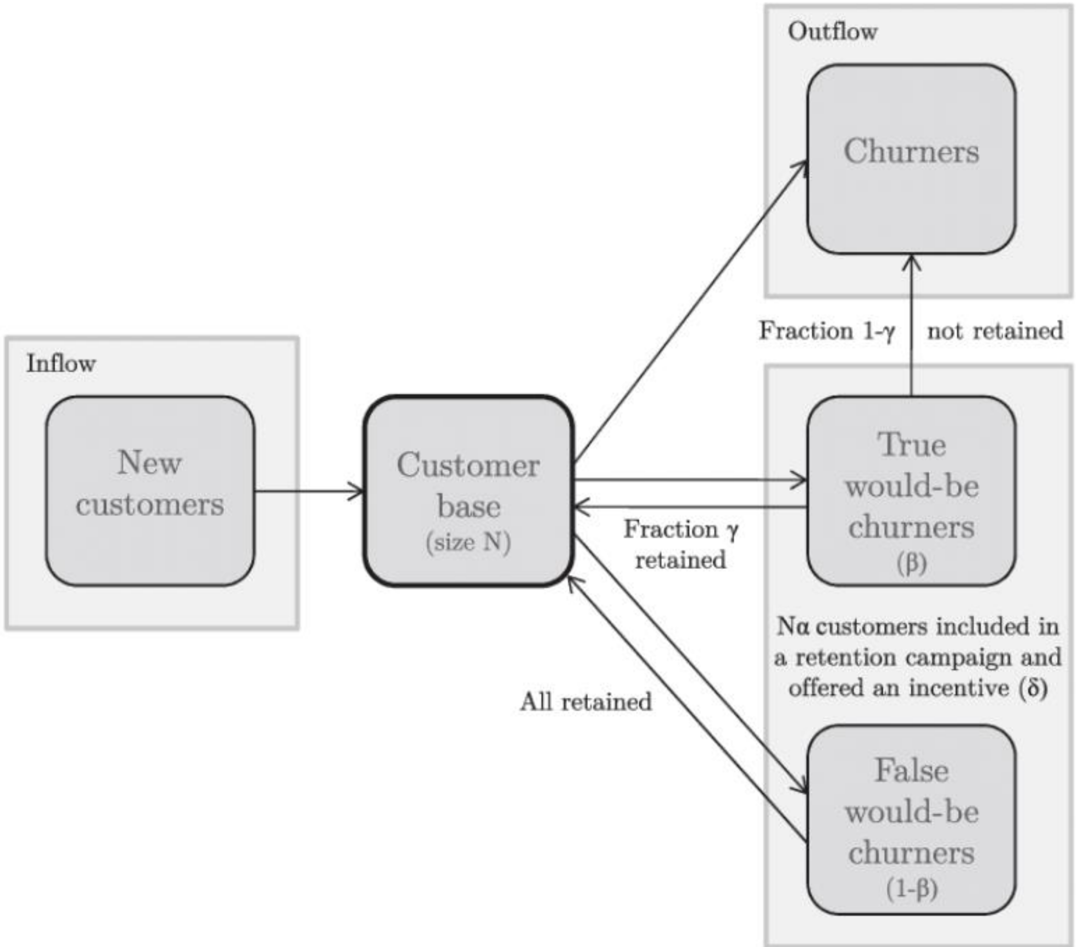
Hardware:-

1. Keyboard
2. Monitor
3. RAM(min 8GB)

Software:-

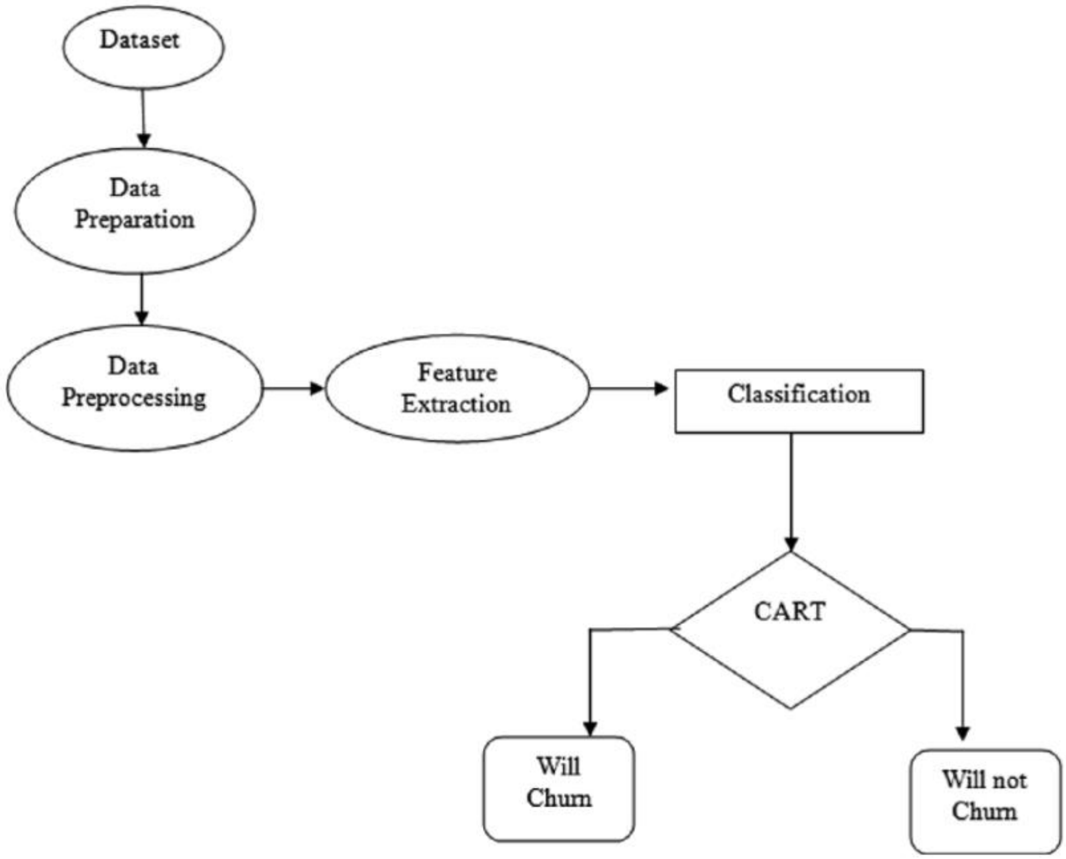
1. Tensorflow.
2. Pandas
3. Numpy
4. Sklearn

UML Diagram



Outflow
Churners
Fraction 1-γ
not retained
Inflow
New customers
Customer base
(size N)
True would-be churners
(β)
Fraction γ retained
Nα customers included in a retention campaign and offered an incentive (δ)
All retained
False would-be churners
(1-β)

Data Flow Diagram



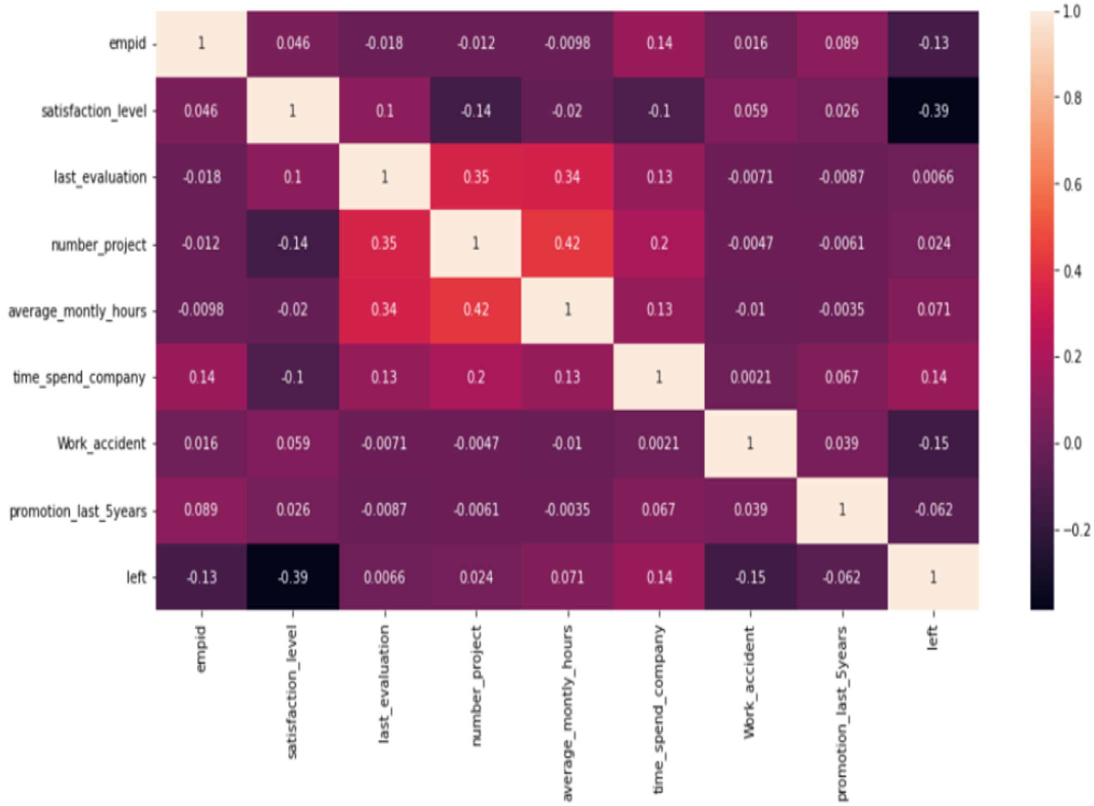
Parameters in the churn model

Following are the parameters:-

1. Satisfaction Level
2. Number of Projects Assigned
3. Last Evaluation
4. Average Monthly Hours
5. Work Accidents
6. Promotion in last 5 years
7. Time spent in the company
8. Salary

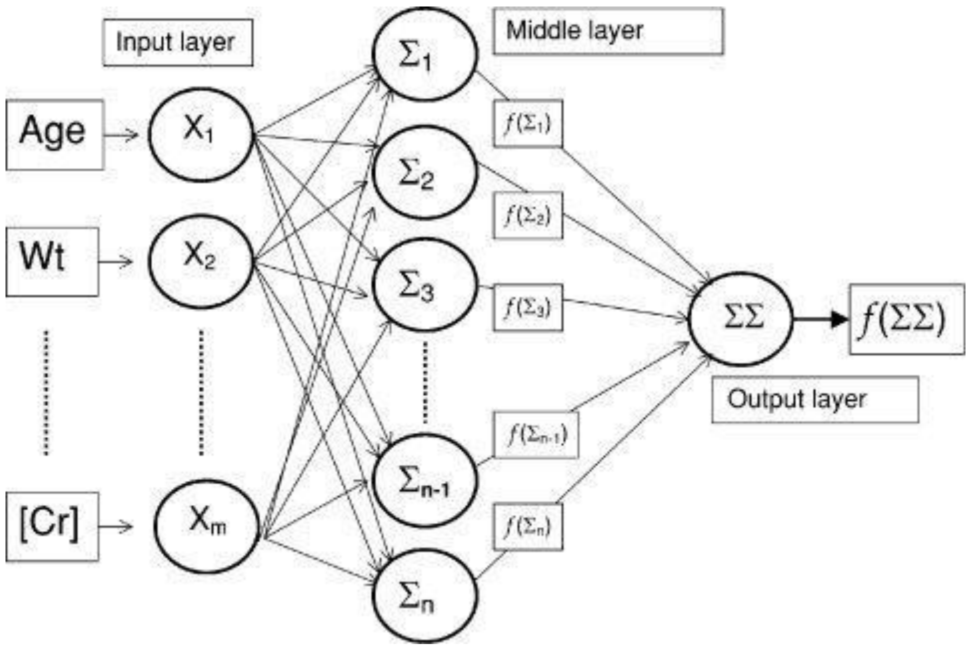
9. Employee ID

Heatmap for the above mentioned parameters given by our model.



SYSTEM ARCHITECTURE

Basic architecture of an **Artificial Neural Network**



1. Artificial neural network (ANN) architecture. ANNs consist of artificial neurons.

2. Each artificial neuron has a processing node ('body') represented by circles in the figure as well as connections from ('dendrites') and connections to ('axons') other neurons which are represented as arrows in the figure.

3. In a commonly used ANN architecture, the multilayer perceptron, the neurons are arranged in layers. An ordered set (a vector) of predictor variables is presented to the input layer.

4. Each neuron of the input layer distributes its value to all of the neurons in the middle layer. Along with each connection between input and middle neurons there is a connection weight so that the middle neuron receives the product of the value from the input neuron and the connection weight.

5. Each neuron in the middle layer takes the sum of its weighted inputs and then applies a non-linear (usually logistic) function to the sum. The result of the function then becomes the output from that particular middle neuron.

6. Each middle neuron is connected to the output neuron. Along with each connection between a middle neuron and the output neuron, there is a connection weight.

7. In the final step, the output neuron takes the weighted sum of its inputs and applies the non-linear function to the weighted sum.

8. The result of this function becomes the output for the entire ANN. More details are provided in the appendix.

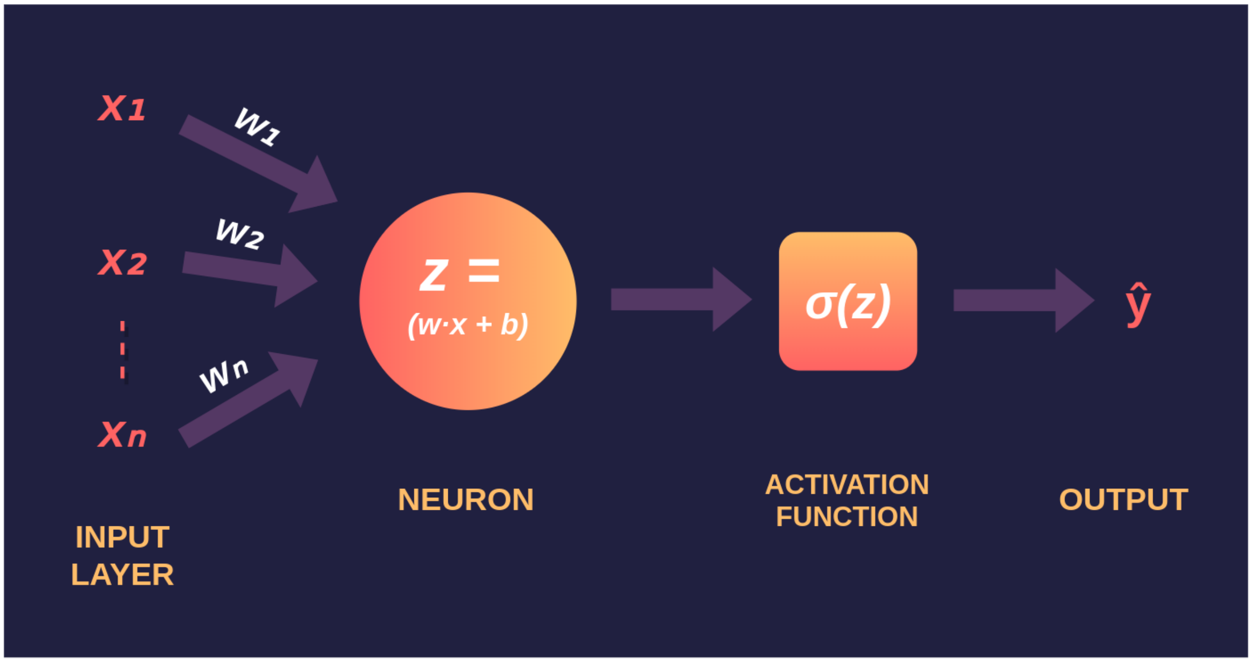
IMPLEMENTATION

We are going to use artificial neural networks to train our model.
The details filled in by the user is used as inputs for training the model.
Based on the user inputs we are going to predict the employee leaves the company or not.

What is ANN?

An artificial neural network is an information processing model that is inspired by the way the biological nervous system such as the brain process information.

An artificial neural network is based on a collection of connected units or nodes called an artificial neuron, which loosely model the neurons in a biological brain.

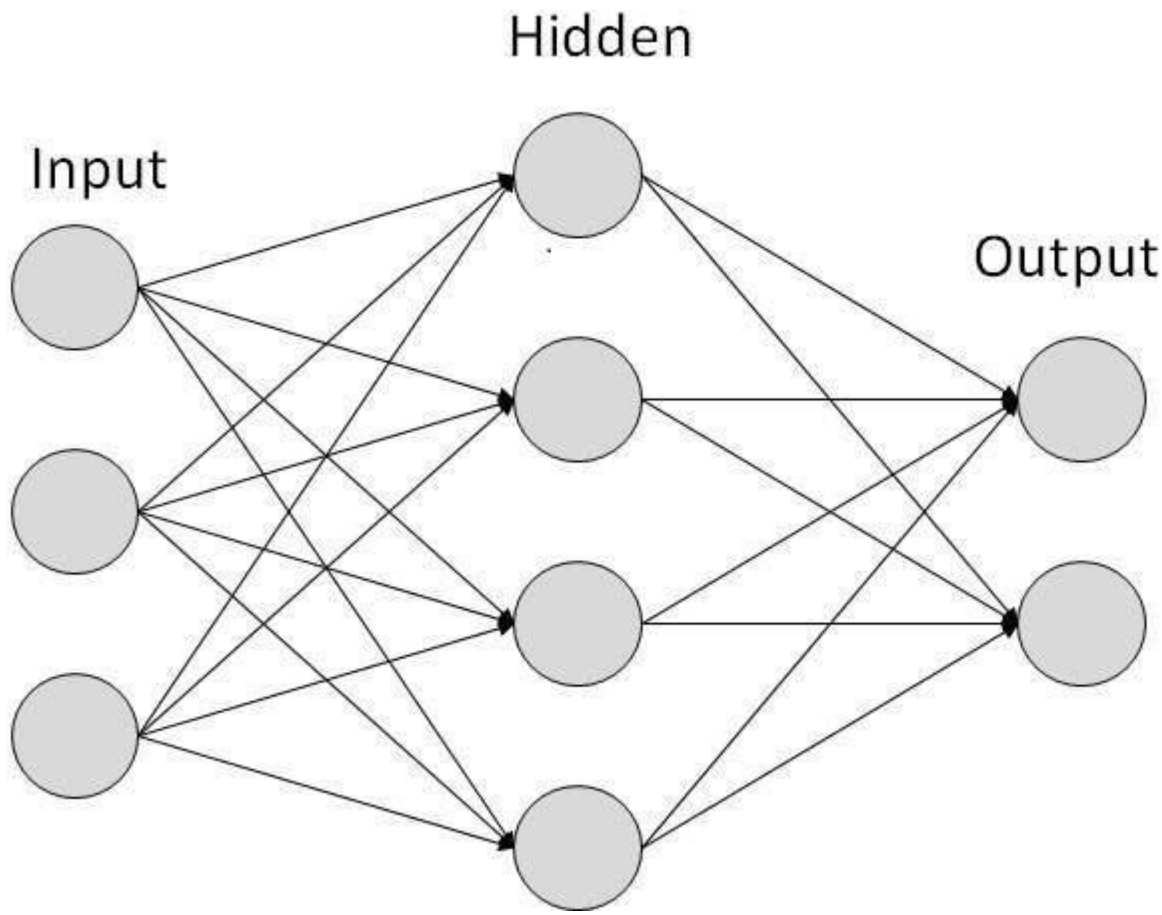


For classification purpose we are using the **sigmoid** function.
As it can change any value between the range 0 to 1.

Structure of ANN

The possibility of ANNs depends on the conviction that the working of the human mind by creating the correct associations can be imitated utilizing silicon and wires as living Neurons and Dendrites [1].
ANNs are made out of numerous nodes, which mimic biological neurons of the human cerebrum. The neurons are associated by connections and they collaborate with one another. The nodes can take input data and perform the direct procedure on the data. The outcome of these exercises is passed to various neurons. The output at every node is called its activation or node value [6].

Each connection is related to weight. ANNs are equipped for realization, which happens by changing weight esteems. The accompanying outline shows a straightforward ANN –



Types of ANN

There are two Artificial Neural Network topologies − Feedforward and Feedback.

FeedForward ANN:

In this ANN, the data stream is unidirectional. A unit sends data to another unit from which it doesn't get any data. There are no criticism circles. They areutilizedInpatterngeneration/recognition/classification. They have fixed information sources and outputs [7].

FeedBack ANN:-

Here, feedback loops are allowed. They are used in content-addressable memories.

Working of ANNs:

In the topology diagrams shown, every bolt addresses a connection between two neurons and demonstrates the pathway for the progression of data. Every connection has a weight, a whole number that controls the sign between the two neurons.

Computational Model of Neuron

For the computational part, we take x[i] value for the input nodes and its corresponding weight w[i] which is associated between input and hidden layer.

1. We take the summation of all nodes and its corresponding weight.

$\sum = x[1].w[1] + x[2].w[2] +.. + x[i].w[i]$

2. We add a bias in the hidden layer for setting up the default values.

$z = x.w + b$ (b is bias)

3. Now we will pass this z value to an activation function. For this project, we use the sigmoid activation function.

4. After giving it to the activation function we get the predicted value, so we check it with the actual output and reset the weights of this neural network so that the efficiency improved.

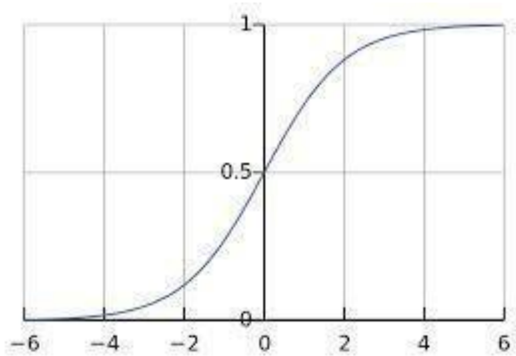
The Sigmoid Function

This is an S-like structured function. It is monotonic in nature. The sigmoid function is characterized as:

The value of this function ranges from 0 to 1.

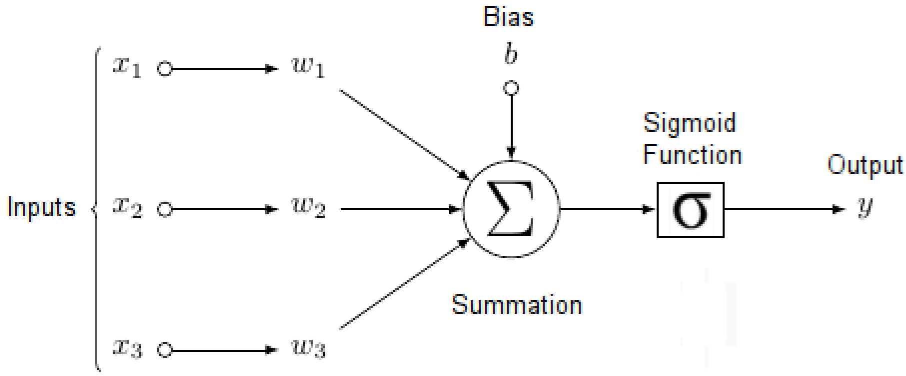
It's a nonlinear curve. The main benefit of using this is because it ranges between 0 to 1. So it is easily classifiable.

Below is the curve for the sigmoid function.



As we compute the value of z in the computation part of ANN, it gives us a floating-point number, so it is really hard for us to analyse. So we give the value of z to the sigmoid function, this function return in the form of 0 or 1, so we can easily classify the inputs.

Below is the figure of the full process:-

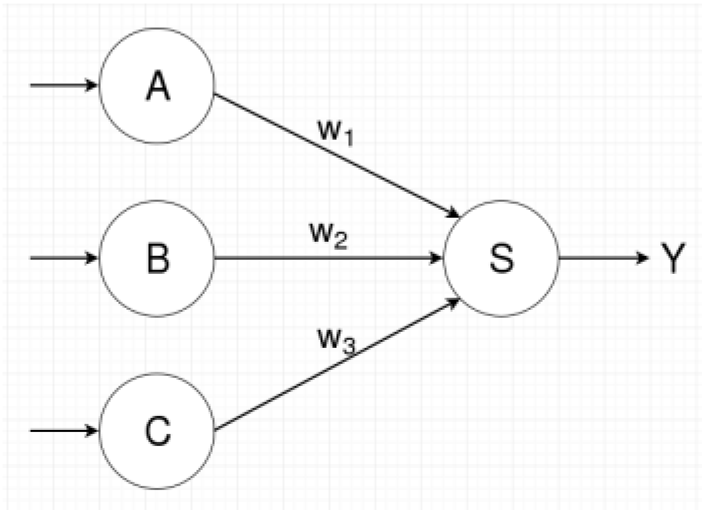


ANNs as Graphs

A typical neural network contains 3 layers, in our case, it is:-

1. **Input Layer:-** In our dataset, various parameters are there which affect the output, so we have taken all such parameters as input(salary, promotion, no. of projects, location, etc).

2. **Hidden layer:-** The main mathematics happens here. Each input node has a relation with the node of the hidden layer. We give weights to this connecting input and hidden nodes. The weights are important for our calculation. After getting the output we reset the weights so that it gets more and more efficient. Below is the figure:-



3. **Output layer:-** In between the hidden layer and output layer our activation function is also present which basically scales down the values to some range so that they can be analysed easily. In the output part, we get predicted output. In the training phase, we can check for the error percentage in the output using techniques like mean squared error and resetting the value of the weights.

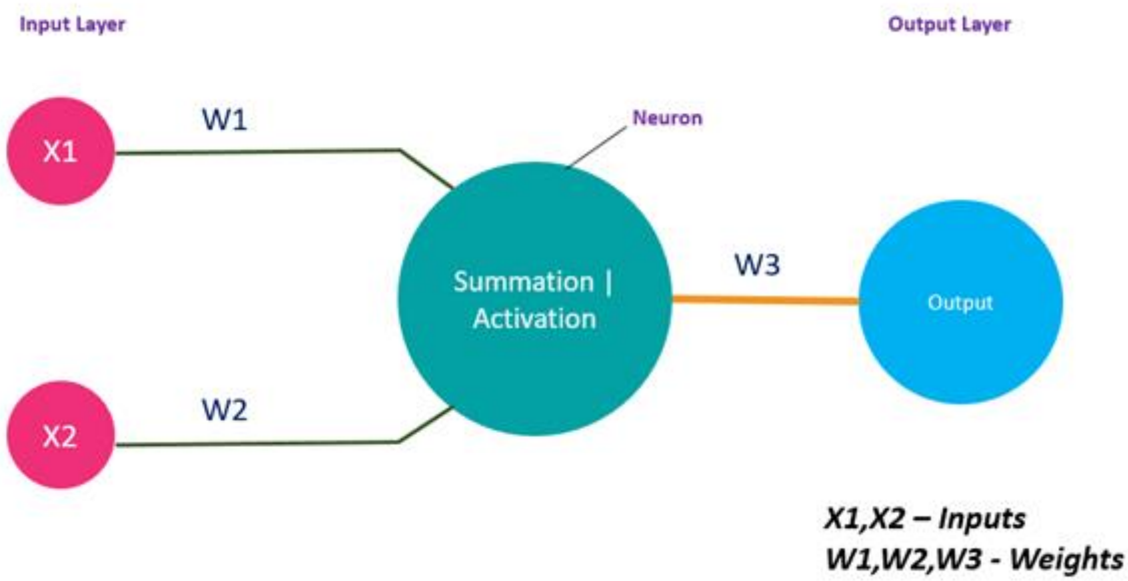
Perceptron

A single artificial neuron is called a perceptron.

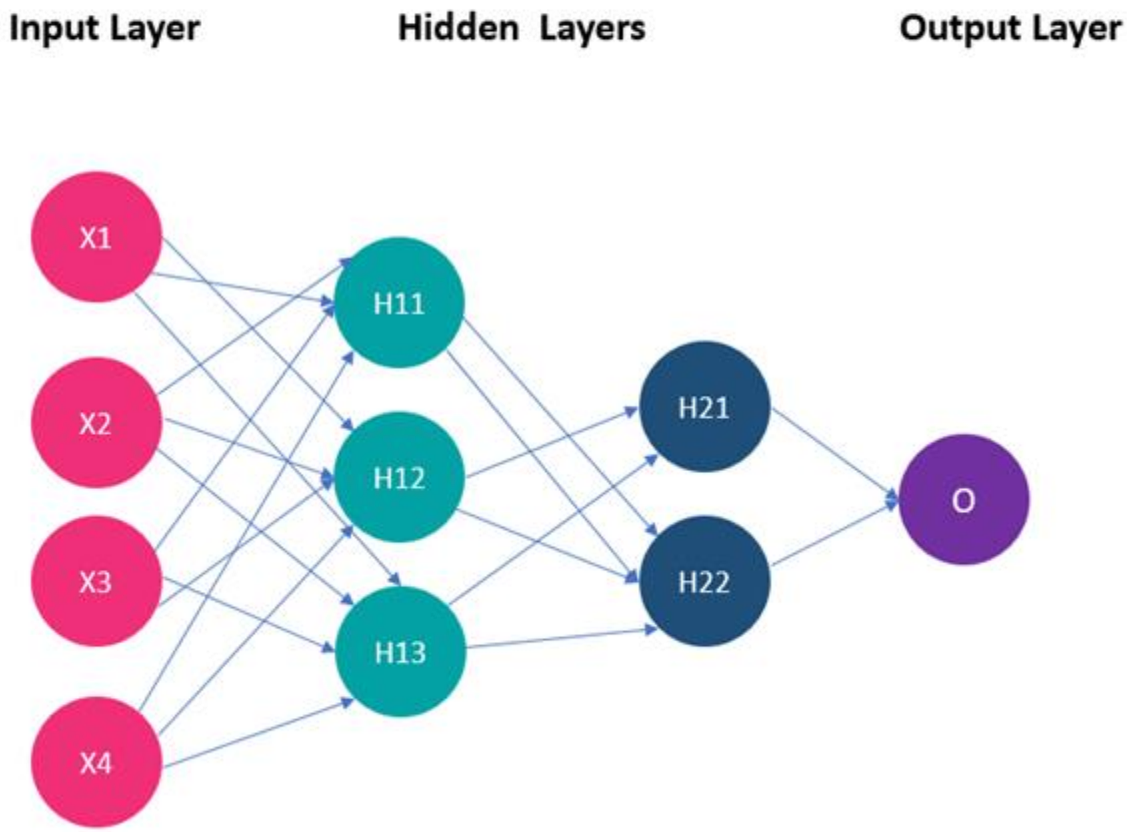
What does this Neuron contain?

1. Summation function
2. Activation function

The data sources given to a perceptron are handled by a Summation function and followed by an actuation function to get the ideal yield.



This is a basic perceptron, yet imagine a scenario in which we have numerous inputs and immense data; a solitary perceptron isn't sufficient. We should continue expanding the neurons. Also, here is the fundamental neural network having an input layer, hidden layer and output layer.



We ought to consistently recall that a neural network has a solitary input layer, output layer however it can have different hidden layers. In the above fig, we can see the example of a neural network with one input layer, two hidden layers, and one output layer.

Activation Function

Any activation function ought to be differentiable since we utilize a back proliferation system to decrease the blunder and update the weights likewise.

Stage 1: For each input we take the value of the node and multiply with weight of the edge and then take summation of all.

$\sum = (x1*w1) + (x2*w2)+\ldots\ldots+(xn*wn)$

$X = [x1,x2,\ldots.,xn]$

$W = [w1,w2\ldots,wn]$

$\sum = x.w$

Stage 2: After taking the summation of all we add the basis.

$Z = x.w + b$

Stage 3: Pass the value of Z in the activation function. we also called $\sigma(x)$ as $\hat{y}$, in this case, $\sigma(x) = \hat{y}$.

And we also know that,

where σ denotes the sigmoid activation function and the output we get after the forward prorogation is known as the predicted value $\hat{y}$ [4].

ANNs use activation functions (AFs) to perform complex computations in the hidden layers and then transfer the result to the output layer. They convert the linear input signals of a node into non-linear output signals to facilitate the learning of high order polynomials that go beyond one degree for deep networks.

Types of Activation Functions

1. Sigmoid Function

In an ANN, the sigmoid function is a non-linear AF used primarily in feedforward neural networks. It is a differentiable real function, defined for real input values, and containing positive derivatives everywhere with a specific degree of smoothness.

The sigmoid function appears in the output layer of the deep learning models and is used for predicting probability-based outputs. The sigmoid function is represented Generally, the derivatives of the sigmoid function are applied to learning algorithms. The graph of the sigmoid function is _S‘ shaped.

Some of the major drawbacks of the sigmoid function include gradient saturation, slow convergence, sharp damp gradients during backpropagation from within deeper hidden layers to the input layers, and non-zero centered output that causes the gradient updates to propagate in varying directions.

Some of the major drawbacks of the sigmoid function include gradient saturation, slow convergence, sharp damp gradients during backpropagation from within deeper hidden layers to the input layers, and non-zero centered output that causes the gradient updates to propagate in varying directions.

$$f(x) = \left(\frac{1}{(1 + exp^{-x})} \right) - (1.4)$$

2. Hyperbolic Tangent Function (Tanh)

The hyperbolic tangent function, a.k.a., the tanh function, is another type of AF. It is a smoother, zero-centered function having a range between -1 to 1.

The tanh function is much more extensively used than the sigmoid function since it delivers better training performance for multilayer neural networks. The biggest advantage of the tanh function is that it produces a zerocentered output, thereby supporting the backpropagation process. The tanh function has been mostly used in recurrent neural networks for natural language processing and speech recognition tasks.

However, the tanh function, too, has a limitation – just like the sigmoid function, it cannot solve the vanishing gradient problem.

$$f(x_i) = \frac{exp\ (x_i)}{\sum_j exp\,(x_j)} \quad -(1.12)$$

3. Softmax Function

The softmax function is another type of AF used in neural networks to compute probability distribution from a vector of real numbers. This function generates an output that ranges between values 0 and 1 and with the sum of the probabilities being equal to 1.

This function is mainly used in multi-class models where it returns probabilities of each class, with the target class having the highest probability. It appears in almost all the output layers of the DL architecture where they are used.

$$f(x_i) = \frac{exp\ (x_i)}{\sum_j exp\,(x_j)} \quad -(1.12)$$

4. Softsign Function

The softsign function is another AF that is used in neural network computing. Although it is primarily in regression computation problems, nowadays it is also used in DL based text-to-speech applications. It is a quadratic polynomial.

The main difference between the softsign function and the tanh function is that unlike the tanh function that converges exponentially, the softsign function converges in a polynomial form.

$$\psi(z) = z/(1 + |z|)$$

5. Rectified Linear Unit (ReLU) Function

The rectified linear unit (ReLU) function, is a fast-learning AF that promises to deliver state-of-the-art performance with stellar results. Compared to other AFs like the sigmoid and tanh functions, the ReLU function offers much better performance and generalization in deep learning. The function is a nearly linear function that retains the properties of linear models, which makes them easy to optimize with gradient-descent methods.

The ReLU function performs a threshold operation on each input element where all values less than zero are set to zero.

The most significant advantage of using the ReLU function in computation is that it guarantees faster computation.

Another critical aspect of the ReLU function is that it introduces sparsity in the hidden units by squishing the values between zero to maximum.

6. Exponential Linear Units (ELUs) Function

The exponential linear units (ELUs) function is an AF that is also used to speed up the training of neural networks (just like ReLU function). The biggest advantage of the ELU function is that it can eliminate the vanishing gradient problem by using identity for positive values and by improving the learning characteristics of the model.

Real-World Applications

In companies and in other institutions, there are lots of valuable data in their databases so after some period of time they would have to change the data or would have to increase the size of the database. The main problem would arise when people leave the company and still their data would be present in the database, it would be a waste of memory and resources. The movement from one company to another by an employee may be due to various factors such as-

1. Services Offered
2. Pay Range
3. Employee-friendly staff
4. Location(where the company is situated)
5. Technologies Available

Using various machine learning models available in the market we could accurately predict the probability that when an employee would leave the company. We have used artificial neural networks and machine learning to predict the probability of a employee transfer using data and resources efficiently. This would also help various companies to see how employees use their services and for how long do they use it, so that they can accurately identify which group of individuals they should target to while giving their positions as to maximize their profits.

Benefits and Involvement of AI

AI is now being used all over the world for various purposes to automate things in the industry. With the help of AI and machine learning models, we have analyzed data accurately so we can predict the probability of the churn. For this, we have used supervised learning techniques like classification and regression. The details filled in by the user are used as inputs for the training model. So, based on these user inputs, we can predict whether the employee leaves the membership or not.

We have also used various libraries like Tensorflow, Pandas, Numpy and Sklearn for this project. To train our model we have used Artificial Neural Networks. The churn calculations are built on existing data like the number of users who left the given service in a given amount of time. The predictive churn model speculates on this data and tells us the probable churn values for the oncoming future. This would help the companies as they won't have to spend huge amounts of money on database shifting and changing. Our machine learning algorithm’s space efficiency would lessen calculations and decrease the AI node probability run-time.

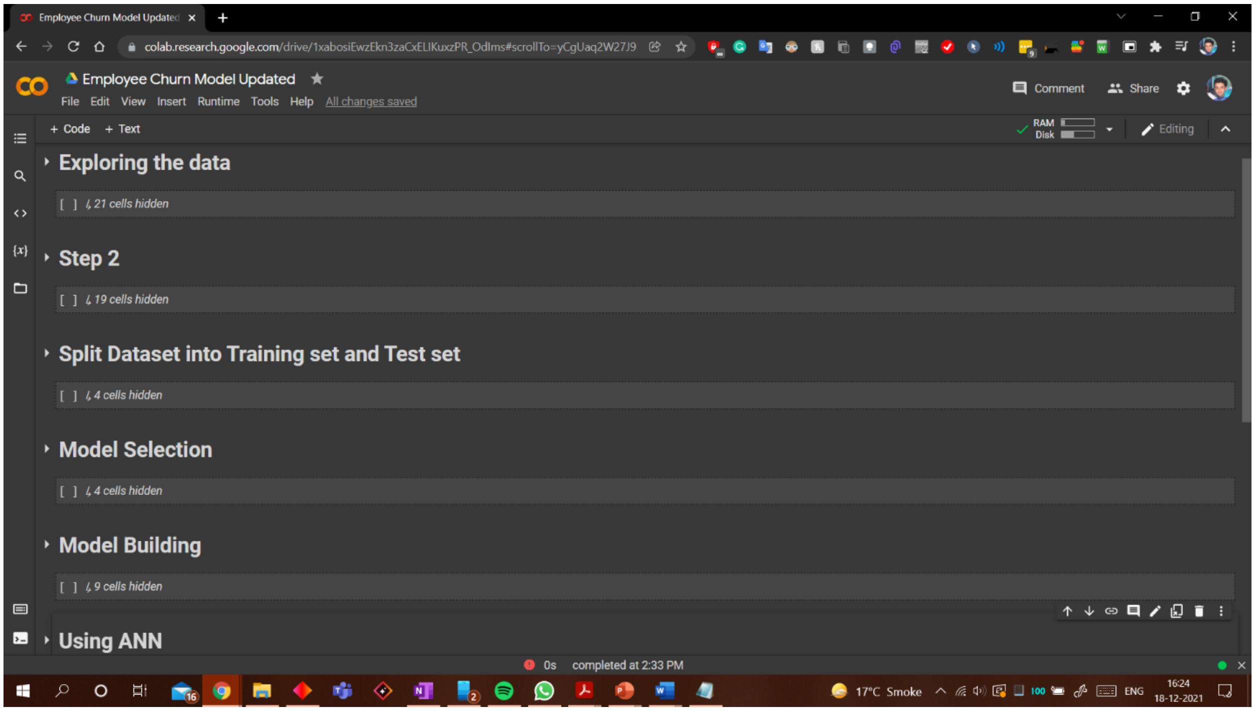
Churn could happen due to many different reasons and churn analysis helps to identify the cause (and timing) of this churn opening up opportunities to implement effective retention strategies.

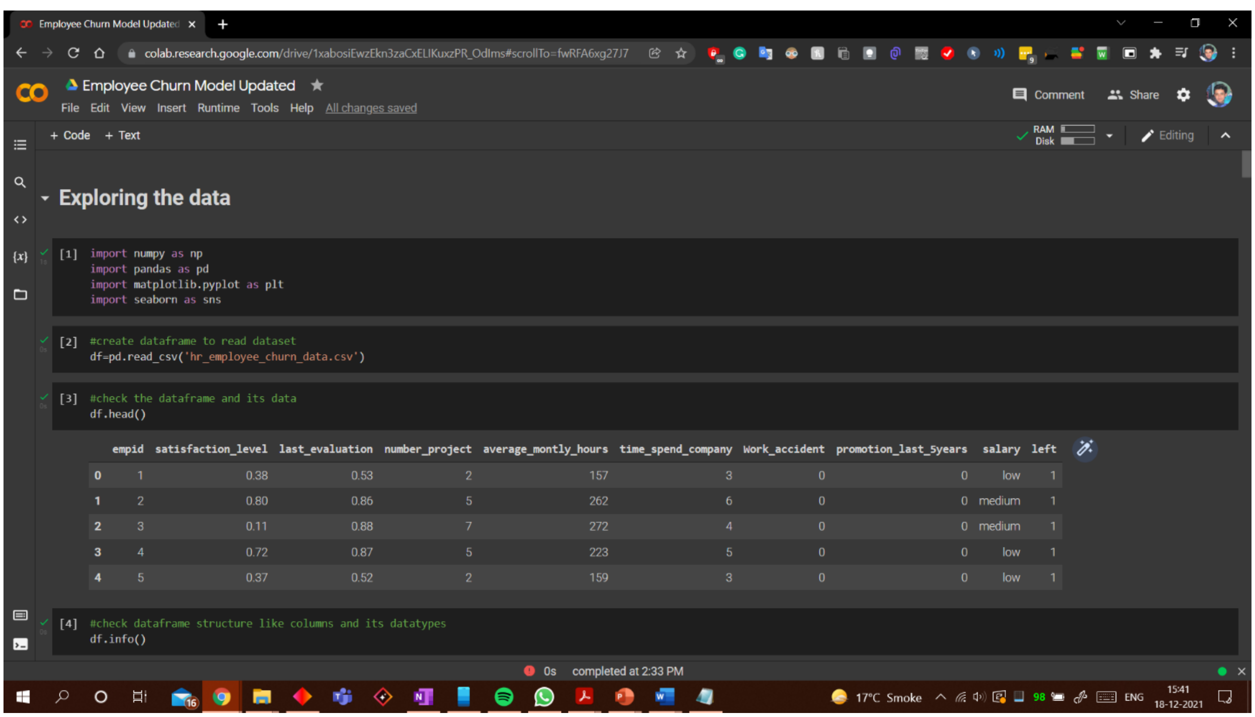
Here are 6 time-tested steps to make sure you are focusing on retaining your employees — we are going to focus only on step 2 and parts of step 3 for this. While at this, remember that this is not about blaming the product or employee success group for the churn but to create a strategy to improve employee retention.

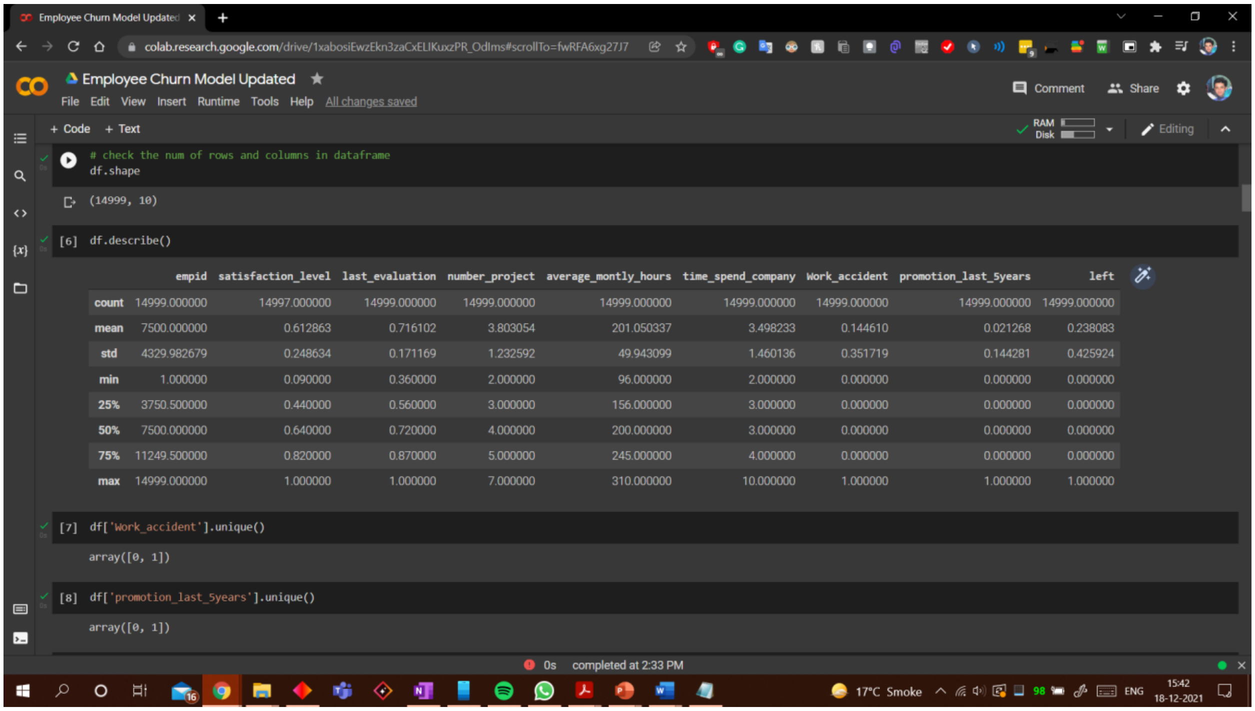
1. Gather available employee behavior, transactions, demographics data and usage patterns
2. Utilize these data points to predict employee segments who are likely to churn
3. Create a model to pattern the risk tolerance of the business with respect to churn probability.
4. Design an intervention model to consider how the level of intervention could affect the churn percentages and employee lifetime value (CLV)
5. Implement effective experimentation across multiple employee segments for reducing churn and promoting retention.
6. Rinse and Repeat from Step 1 (cognitive churn management is a continuous process and not once a year exercise).

Chapter 4

Results and Discussions and Code







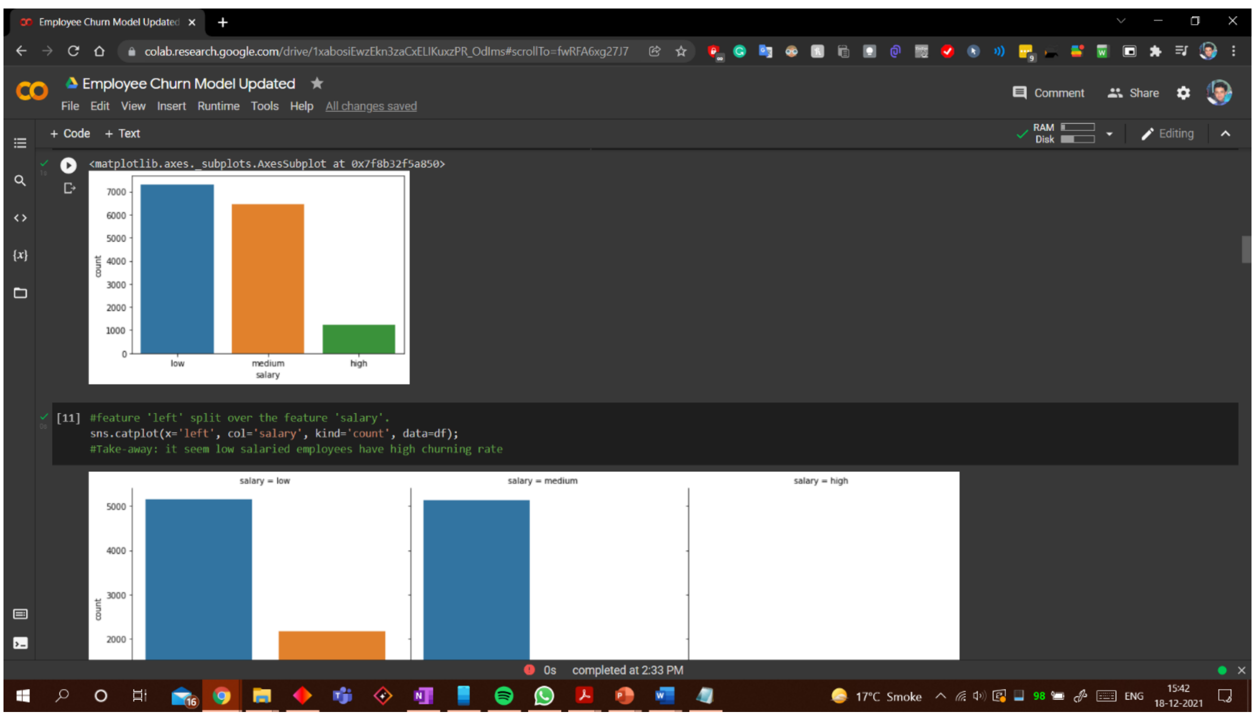
```
# check the num of rows and columns in dataframe
df.shape
```

```
(14999, 10)
```

```
[6] df.describe()
```

	empid	satisfaction_level	last_evaluation	number_project	average_montly_hours	time_spend_company	Work_accident	promotion_last_5years	left
count	14999.000000	14997.000000	14999.000000	14999.000000	14999.000000	14999.000000	14999.000000	14999.000000	14999.000000
mean	7500.000000	0.612863	0.716102	3.803054	201.050337	3.498233	0.144610	0.021268	0.238083
std	4329.982679	0.248634	0.171169	1.232592	49.943099	1.460136	0.351719	0.144281	0.425924
min	1.000000	0.090000	0.360000	2.000000	96.000000	2.000000	0.000000	0.000000	0.000000
25%	3750.500000	0.440000	0.560000	3.000000	156.000000	3.000000	0.000000	0.000000	0.000000
50%	7500.000000	0.640000	0.720000	4.000000	200.000000	3.000000	0.000000	0.000000	0.000000
75%	11249.500000	0.820000	0.870000	5.000000	245.000000	4.000000	0.000000	0.000000	0.000000
max	14999.000000	1.000000	1.000000	7.000000	310.000000	10.000000	1.000000	1.000000	1.000000

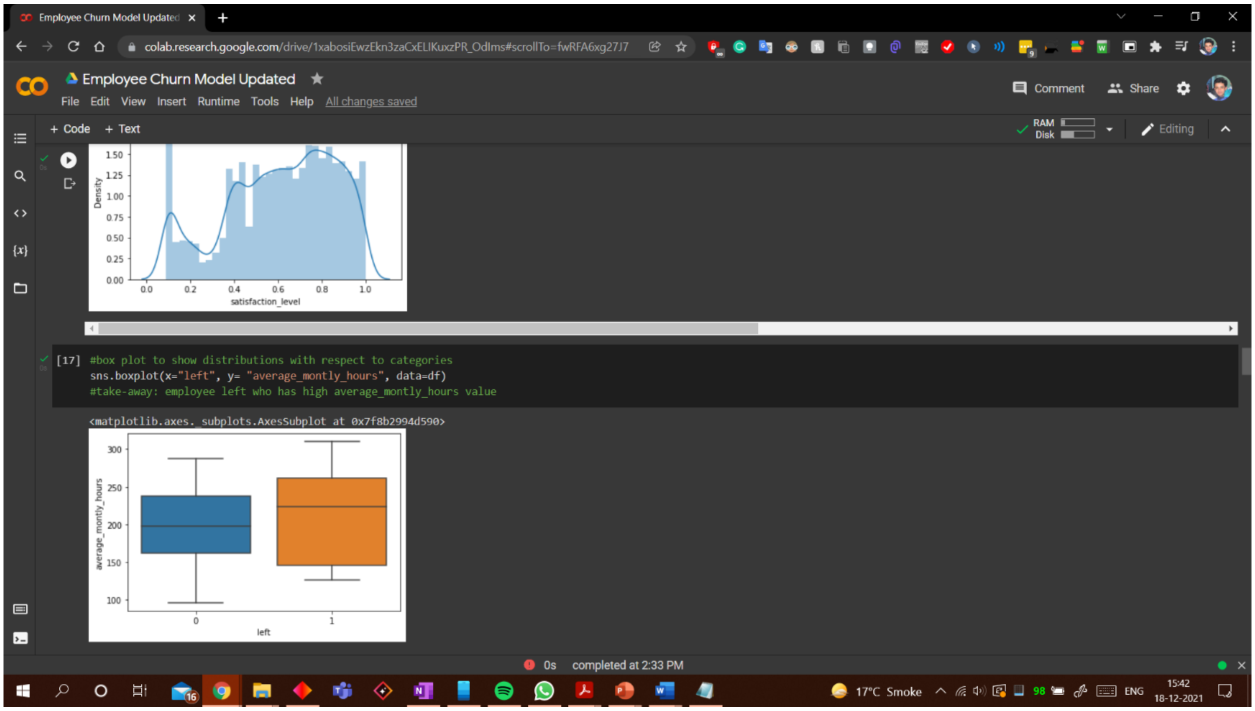
```
[7] df['Work_accident'].unique()
array([0, 1])

[8] df['promotion_last_5years'].unique()
array([0, 1])
```

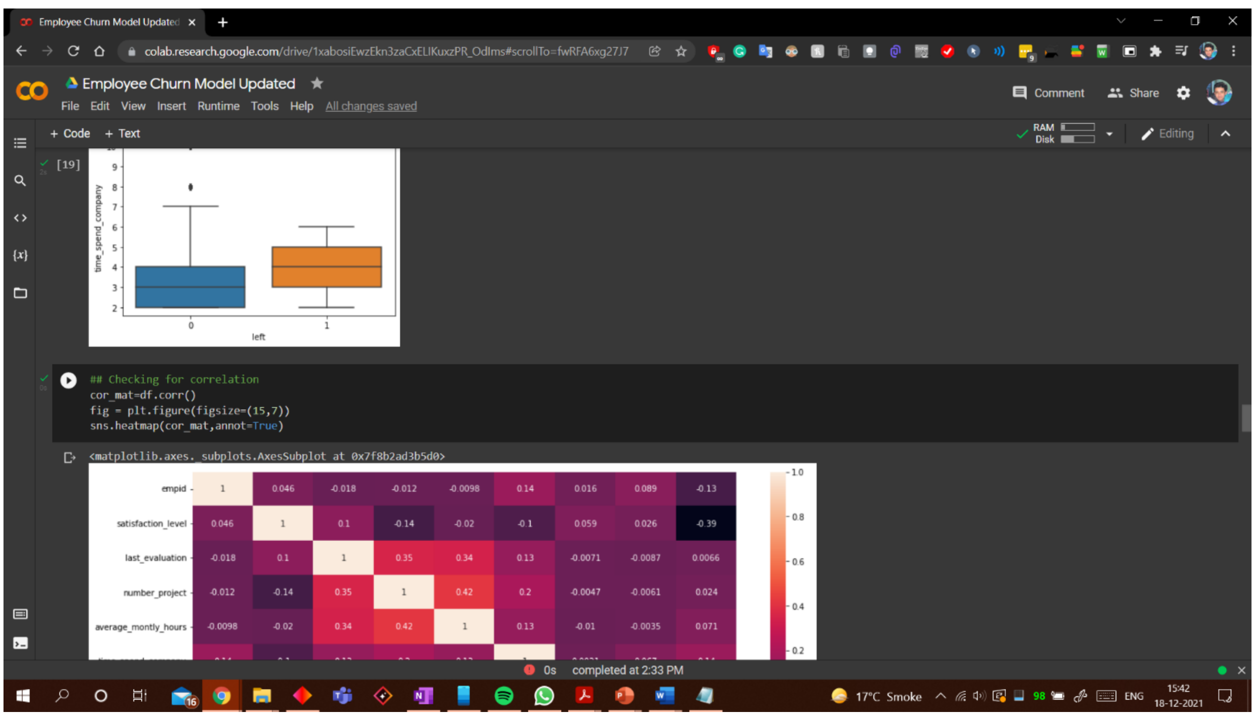


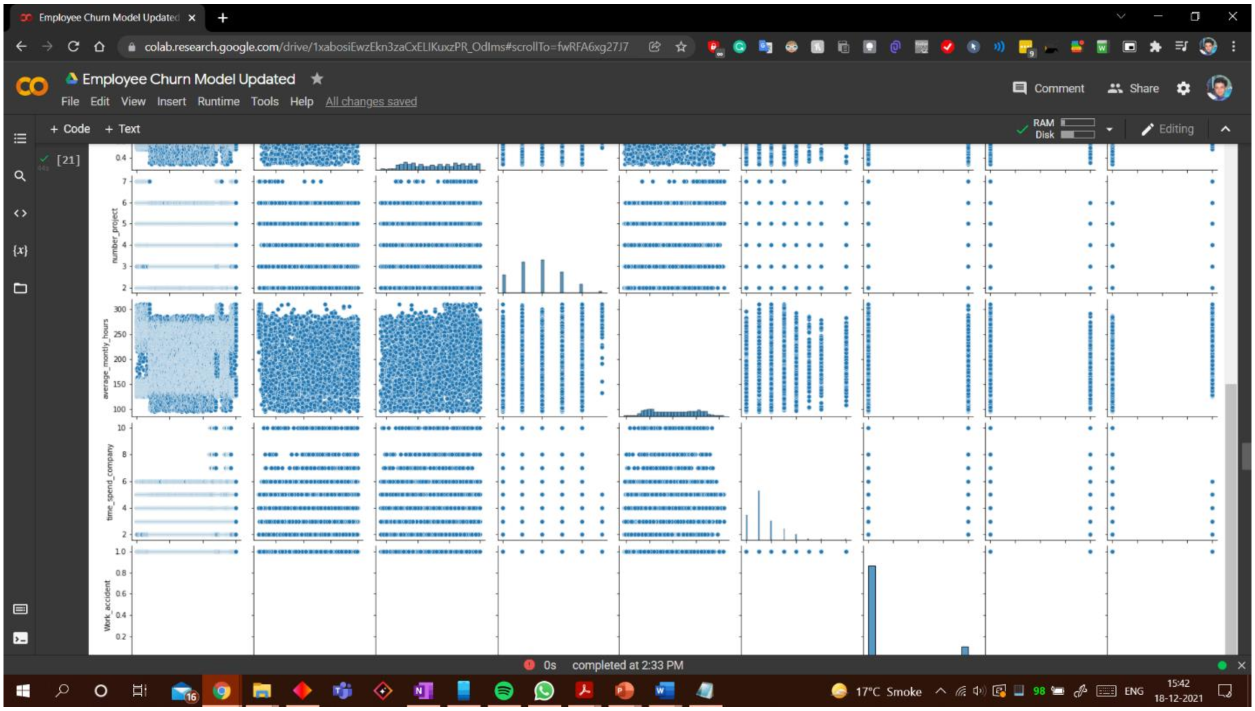
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8b32f5a850>
```

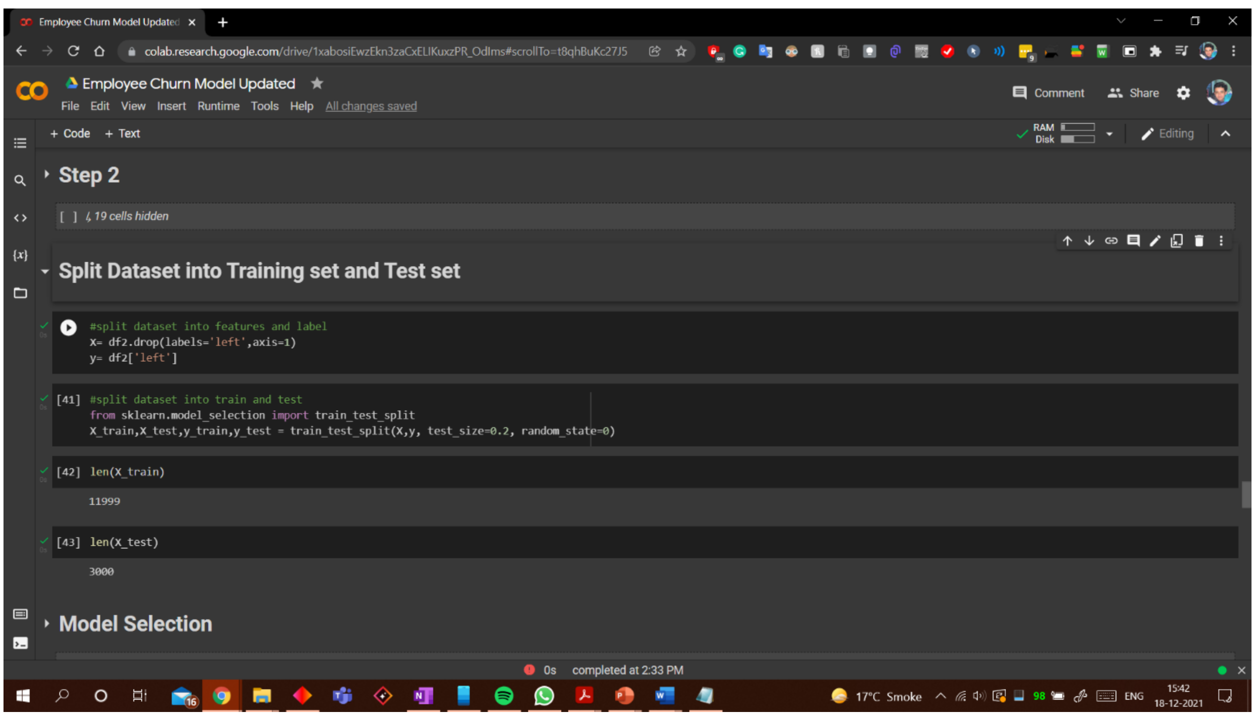


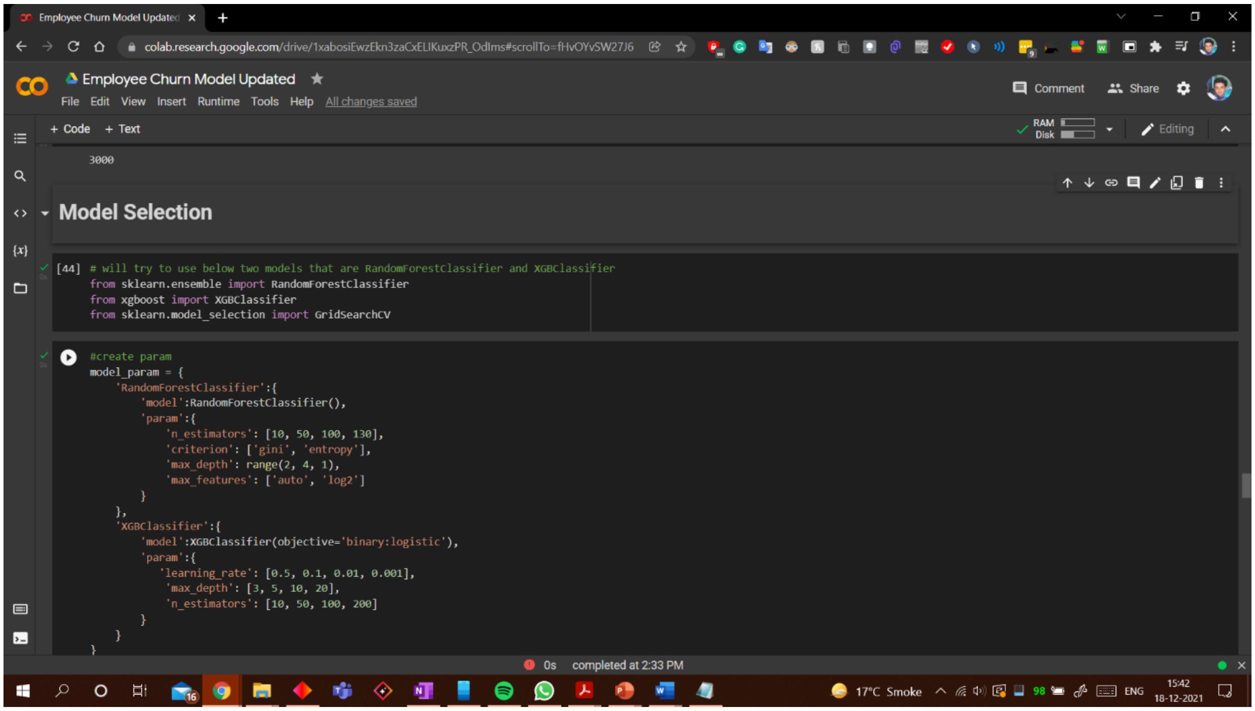
```
[11] #feature 'left' split over the feature 'salary'.
sns.catplot(x='left', col='salary', kind='count', data=df);
#Take-away: it seem low salaried employees have high churning rate
```

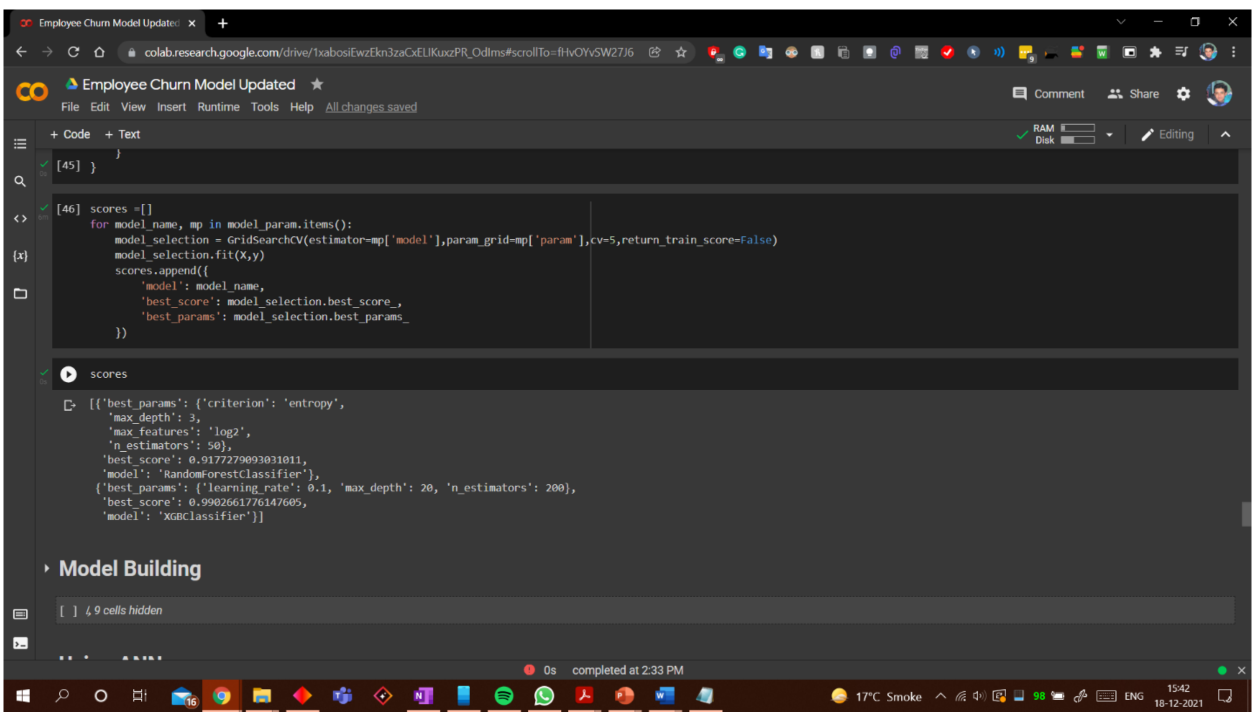

Employee Churn Model Updated
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
Editing
Density
satisfaction_level
[17]
#box plot to show distributions with respect to categories
sns.boxplot(x="left", y= "average_montly_hours", data=df)
#take-away: employee left who has high average_montly_hours value
<matplotlib.axes._subplots.AxesSubplot at 0x7f8b2994d590>
average_montly_hours
left
completed at 2:33 PM


Employee Churn Model Updated
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
Editing
[19]
time_spend_company
left
Checking for correlation
cor_mat=df.corr()
fig = plt.figure(figsize=(15,7))
sns.heatmap(cor_mat,annot=True)
<matplotlib.axes._subplots.AxesSubplot at 0x7f8b2ad3b5d0>
empid
satisfaction_level
last_evaluation
number_project
average_montly_hours
completed at 2:33 PM


Employee Churn Model Updated
File Edit View Insert Runtime Tools Help All changes saved
[21]
number_project
average_montly_hours
time_spend_company
Work_accident
0s completed at 2:33 PM


Employee Churn Model Updated
File Edit View Insert Runtime Tools Help All changes saved
Step 2
[ ] 19 cells hidden
Split Dataset into Training set and Test set
#split dataset into features and label
X= df2.drop(labels='left',axis=1)
y= df2['left']
[41] #split dataset into train and test
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test = train_test_split(X,y, test_size=0.2, random_state=0)
[42] len(X_train)
11999
[43] len(X_test)
3000
Model Selection
0s completed at 2:33 PM


Employee Churn Model Updated
File Edit View Insert Runtime Tools Help All changes saved
3000
Model Selection
[44] # will try to use below two models that are RandomForestClassifier and XGBClassifier
from sklearn.ensemble import RandomForestClassifier
from xgboost import XGBClassifier
from sklearn.model_selection import GridSearchCV
#create param
model_param = {
'RandomForestClassifier':{
'model':RandomForestClassifier(),
'param':{
'n_estimators': [10, 50, 100, 130],
'criterion': ['gini', 'entropy'],
'max_depth': range(2, 4, 1),
'max_features': ['auto', 'log2']
}
},
'XGBClassifier':{
'model':XGBClassifier(objective='binary:logistic'),
'param':{
'learning_rate': [0.5, 0.1, 0.01, 0.001],
'max_depth': [3, 5, 10, 20],
'n_estimators': [10, 50, 100, 200]
}
}
}
0s completed at 2:33 PM


Employee Churn Model Updated
File Edit View Insert Runtime Tools Help All changes saved
[45] }
[46] scores =[]
for model_name, mp in model_param.items():
model_selection = GridSearchCV(estimator=mp['model'],param_grid=mp['param'],cv=5,return_train_score=False)
model_selection.fit(X,y)
scores.append({
'model': model_name,
'best_score': model_selection.best_score_,
'best_params': model_selection.best_params_
})
scores
[{'best_params': {'criterion': 'entropy',
'max_depth': 3,
'max_features': 'log2',
'n_estimators': 50},
'best_score': 0.9177279093031011,
'model': 'RandomForestClassifier'},
{'best_params': {'learning_rate': 0.1, 'max_depth': 20, 'n_estimators': 200},
'best_score': 0.9902661776147605,
'model': 'XGBClassifier'}]
Model Building
[ ] ↳ 9 cells hidden
0s completed at 2:33 PM

Model Building

```
[48] #as per above results, xgboost gives best result and hence selecting same to model building...
     model_xgb = XGBClassifier(objective='binary:logistic',learning_rate=0.1,max_depth=20,n_estimators=200)
```

```
[49] model_xgb.fit(X_train,y_train)
```

```
XGBClassifier(max_depth=20, n_estimators=200)
```

```
model_xgb.score(X_test,y_test)
```

```
0.9923333333333333
```

```
[51] X_test.head()
```

	satisfaction_level	last_evaluation	number_project	average_montly_hours	time_spend_company	Work_accident	promotion_last_5years	low	medium
1670	0.44	0.57	2	141	3	0	0	0	1
13378	0.55	0.96	3	194	3	0	0	0	1
10233	0.72	0.67	5	210	2	0	0	0	1
4719	0.96	0.75	4	177	2	0	0	1	0
7003	0.96	0.54	3	198	3	0	0	1	0

```
[52] X_test[:1]
```

0 1
Predicted

Using ANN

```
[57] dataset = pd.read_csv('hr_employee_churn_data.csv')
     dataset.head()
```

	empid	satisfaction_level	last_evaluation	number_project	average_montly_hours	time_spend_company	Work_accident	promotion_last_5years	salary	left
0	1	0.38	0.53	2	157	3	0	0	low	1
1	2	0.80	0.86	5	262	6	0	0	medium	1
2	3	0.11	0.88	7	272	4	0	0	medium	1
3	4	0.72	0.87	5	223	5	0	0	low	1
4	5	0.37	0.52	2	159	3	0	0	low	1

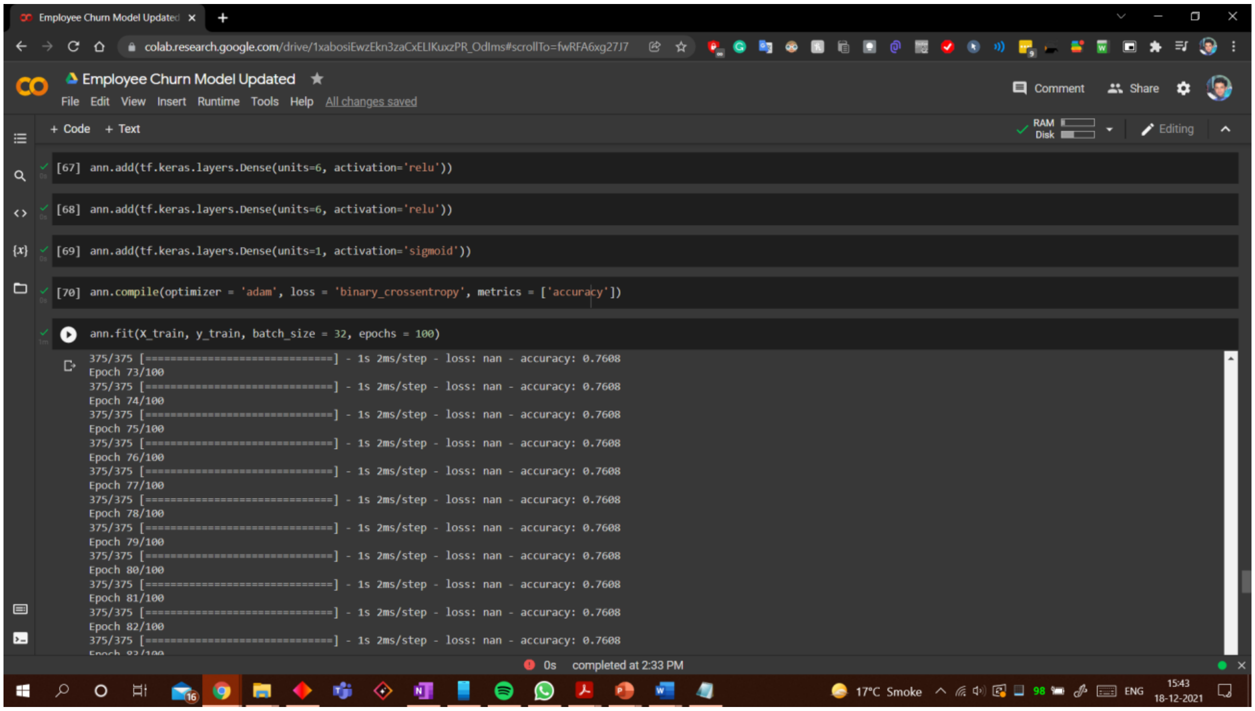
```
X = dataset.iloc[:, 1:-1].values
y = dataset.iloc[:, -1].values
```

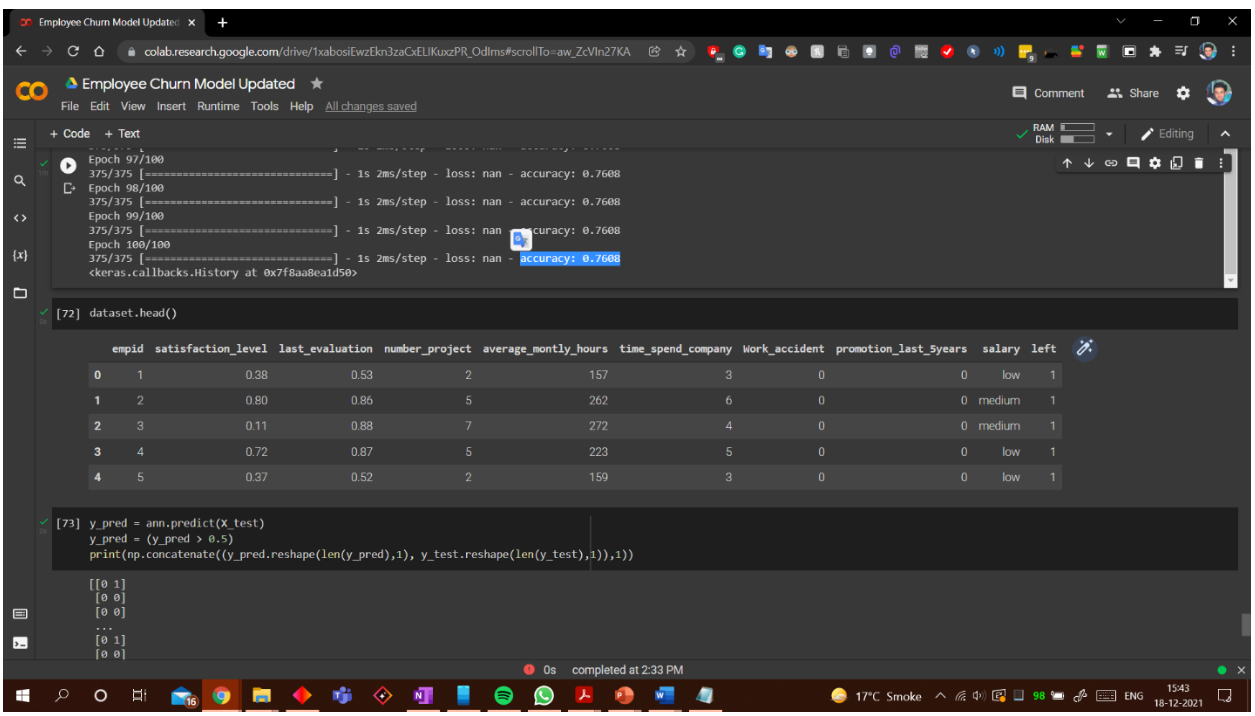
```
[59] print(X[0])
```

```
[0.38 0.53 2 157 3 0 0 'low']
```

```
[60] print(y)
```

```
[1 1 1 ... 1 1 1]
```





Code link –

https://colab.research.google.com/drive/1xabosiEwzEkn3zaCxELIKuxzPR_OdIms?usp=sharing

Chapter 5

Conclusion and Future Scope

Employee Churn is a term every business dreads, especially those which rely heavily on employee experience. Today's customers are well informed and armed with multiple options. They want competitive pricing, value for money and, above all, high quality service and won't hesitate to switch providers if they dont find what they're looking for. As such, it becomes crucial to put in place a sustainable and robust strategy for employee retention to preserve employee lifetime value. Businesses that have a high acquisition low margin model are highly affected by employee churn and need to ensure quick real time decisions to lower the impact.

Current churn analysis techniques rely heavily on Employee Lifetime Value calculations, which are based on certain fixed metrics such as average monthly transactions, average gross margin, monthly retention rate, and so on. All of these metrics are highly number driven and not behavior based.

Neural networks are the best as they are very effective in the classification of inputs and also for a very large dataset it works the best. Neural networks do not take any parameters as in the case of regression models, so it is convenient for analysis. In the case of classification, no weights are there, so the output result varies very much but in the case of the neural network for each error, we reset the value of weights, which in short give the best accuracy. Nevertheless, neural networks were regularly operative for foreseeing time arrangement. These are most useful especially when there are no other portrayals of the perceived arrangement.

A well-structured Big Data analytics model will help re-define existing predictive churn analytics techniques. Businesses can now tap into non-traditional sources such as social media data analytics, employee touch point feedback, call center feedback and many more to create a holistic analytics model that deals with, not only the monetary value but also the behavioral patterns of the customer. This will allow the businesses to identify potential churning customers and act on preventing the churn. This proactive strategy can deliver better results than waiting for a trigger that indicates an inevitable churn and then trying to re-capture the customer. As such, any analytics model should have:

1. The capacity to accumulate vast unstructured data from both traditional and non-traditional sources
2. The ability to superimpose intelligent filters to reduce noise and false alerts
3. Business specific analytics engines to co-relate data, detect patterns, and generate real-time business insights
4. Agility, through flexible tuning of rules and models for near real time streaming of data

Future Scope

An intelligent predictive churn analytics model, powered by Big Data analytics will allow businesses to process, analyze, and co-relate traditional and non-traditional metrics to achieve a holistic employee blueprint and effective insights that can trigger an alarm way before real damage is done. A simple example can be that of personalized retention incentives. Businesses can combine the insights from traditional churn analytics models such as average transaction value, monthly discount values, last transaction date, and so on with data from non-traditional sources such as brand or product sentiment on social media, number of complaints in the last month to the call center, competitor offers, and others. They can use this, to predict the churning intent of the said employee and quickly create a customized offer to try and retain them.

Predictive churn analytics is a small step towards automated personalization, which will be a critical business differentiator delivered by full-fledged Big Data adoption. However, businesses will have to start from small use cases, strategizing progressively to encompass complex multi-level use cases to realize the full potential of Big Data analytics. Businesses have to gear up and ensure that they can manage the speed and complexity of Big Data, establish well defined data points, and equip employees with enough training to handle the process complexities. Most importantly, businesses will have to shift the traditional Business Intelligence mindset of reporting and adapt to the real-time action mindset to successfully decipher the holistic employee insights that Big Data analytics is capable of providing.

References

[1] Supervised Learning in Artificial Neural Networks, Dr Rajeshwari S. Mathad, Volume 5, Issue 3, March (2014), pp. 208-215, International Journal of Advanced Research in Engineering and Technology.

[2] Nancy J and Mirunalini V, Artificial Neural Networks In Optimizing Methane Production From Domestic Waste Digestion, International Journal of Civil Engineering and Technology, 8(4), 2017, pp.1279-1286

[3] Artificial neural networks applications, http://www.alyuda.com/products/forecaster/neuralnetwork-applications.htm, retrieved 2016.

[4] Kiran Sharma, Ankit Naik, Purushottam Patel, "Study of Artificial Neural Network", International Journal of Advanced Research Trends in Engineering and Technology (IJARTET) Vol. 2, Issue 4, April 2015, pp. 46-48.

[5] “Neural Networks” https://cs.stanford.edu/people/eroberts/courses/soco/projects/neuralnetworks/index.html, retrieved 2016

[6] Ahmad Baylari and Montazer, Gh. A., Design a personalized E-Learning system based on item response theory and artificial neural network approach, Expert Systems with Applications, Vol. 36, 4, 2009, pp. 8013-8021

[7] M.R.M. VeeraManickam, Mohanapriya, Bishwajeet K Pandey, Gajanan P Arsalwad, Senthil Kumar Janahan, Vigneshwar .M, “Dragonfly-Artificial Neural Network Model For ELearning Data Analyses: Is Future Generation

Communication Model -Smart E-Learning System” 3rd ICGCET 08-10 August 2017, Ireland.