KAVYA MP

by Kavya .

Submission date: 29-Jan-2022 05:10PM (UTC+0530)
Submission ID: 1750557460
File name: Kavya_mp_report.pdf (1.63M)
Word count: 4600
Character count: 24038

ABSTRACT

This project is a modular application of HR system software. As an HR system deals with all the tasks handled by a manual HR of a company such as leave management, HR metrics, Recruitment, Employee Relations, Satisfaction, Communication, Retention, Performance Management. Similarly, my project is a smaller-scale instance of the same where it handles one part of the whole system which is performance review but instead helps by predicting the review beforehand. It predicts the attrition and retention rate of any company making the whole system more efficient as it's an automatic software implementation instead of humans governing everything manually

The entire program has been developed in Python and uses the Eclipse IDE for running the python application.

The mini-project is completely based on the high-level language, Python and uses GUI programming to provide a simple and easy to understand platform for the users.

CHAPTER 1

INTRODUCTION

1.1 PROBLEM DEFINITION

This project is an instance of HR system software with modular implementation of performance evaluation system modified according to specific requirements of the company. This project will be customized according to the needs of a company as many such software exists for a hefty price but they're not customized according to the requirements of the company. What makes this project better is implementation of machine learning on the idea. This will help us predict the performance of an employee better

1.1.1 OBJECTIVES

Once developed, the application will provide services to companies which will help manual HR to take some load off as they wont have to manually rate each and every employee but the program will automatically do it for them.

- More efficient implementation
- Works as appraisal system
- Helps improve workforce efficiency by analyzing weaknesses
- Predicts attrition and retention rate
- Improves outcome of company
- Helps in more value for money per employee

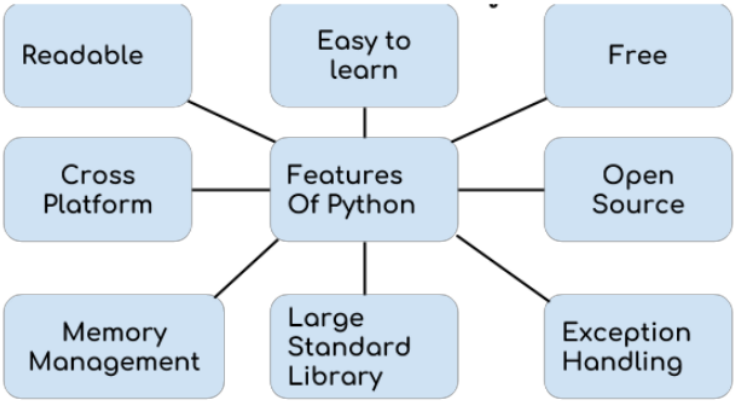
CHAPTER 2

FUNDAMENTALS OF PYTHON

2.1 INTRODUCTION TO PYTHON

Python is a regular and widely utilized broadly useful, significant-level programming language. Guido van Rossum in 1991 was the author of Python and was subsequently evolved by Python Software Foundation. It was basically intended to underscore code meaningfulness, and its linguistic structure permits software engineers to communicate thoughts in a couple of lines of code.

Fig 2.1: features of python



Python applications:

1. Web improvement - Web system like Django and Flask depends on Python. They assist you with composing server-side code which assists you with overseeing the information base, composing backend programming rationale, planning URLs and so forth

2. AI - There are many AI applications written in Python. AI is a method for composing a rationale so a machine can learn and take care of a specific issue all alone. For instance, an items suggestion in sites like Amazon, Flipkart, eBay and so forth is an AI calculation that recognises the client's advantage. Face acknowledgement and Voice acknowledgement in your telephone is one more illustration of AI.

3. Information Analysis - Data investigation and information perception in the type of graphs can likewise be created utilizing Python.

4. Prearranging - Scripting is composing little projects to robotize basic errands, for example, sending computerized reaction messages and so forth Such kinds of utilizations can likewise be written in Python programming language.

5. Game turn of events - You can foster games utilizing Python.

6. You can foster Embedded applications in Python.

7. Work area applications - You can foster work area applications in Python utilizing libraries like TKinter or QT.

What is a Data Structure?

Data structures are used in computer programming to simplify a complex data arrangement. By storing data into data structures, it can be easily accessed and utilized. A data structure is a conceptual model for representing, storing and organizing information in a computer system.

Built-in Data Structures

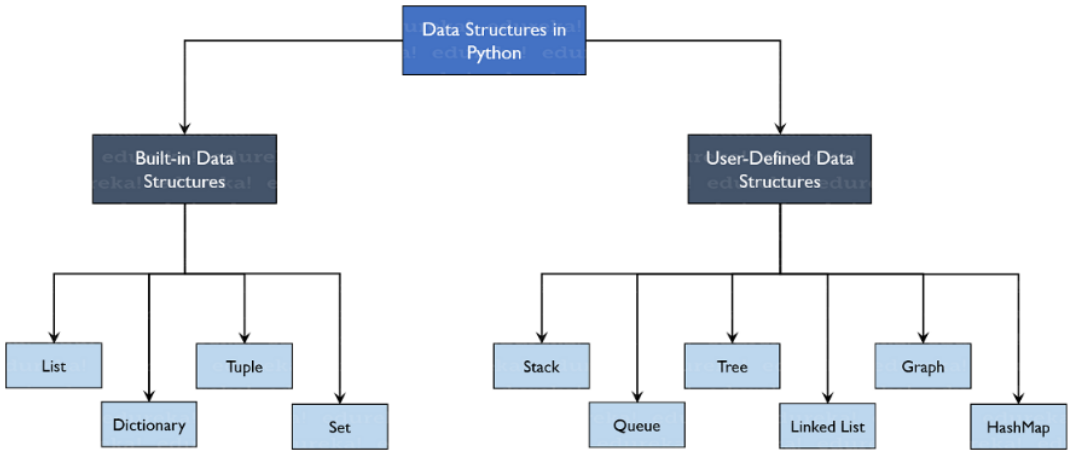


Fig 2.2: DS in python

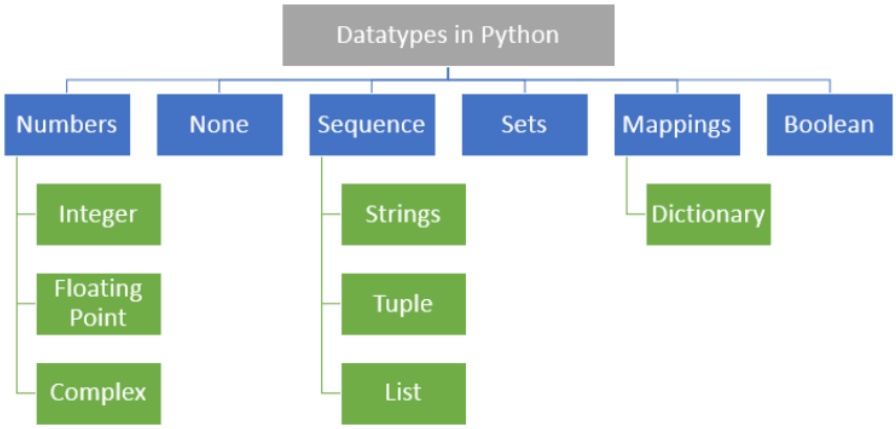


Fig 2.3: data types in python

Lists

Lists are utilized to store information of various information types in a successive way. There are addresses allocated to each component of the list, which is called as Index. The record esteem begins from 0 and continues until the last component called the positive list. There is likewise regrettable ordering what begins from - 1 empowering you to get to components from the last to first. Allow us now to comprehend lists better with the assistance of a model program.

Dictionary

Dictionaries are utilized to store key-esteem sets. To see better, think of a telephone index where hundreds and thousands of names and their corresponding numbers have been added. Presently the consistent qualities here are Name and the Phone Numbers which are called as the keys. What's more the different names and telephone numbers are the qualities that have been taken care of to the keys. Assuming you access the upsides of the keys, you will obtain every one of the names and telephone numbers. So that is what a key-esteem pair is. What's more in Python, this construction is put away using Dictionaries. Allow us to comprehend this better with a model program.

Tuple

Tuples are equivalent to lists are with the exemption that the data once went into the tuple can't be changed regardless. The main special case is the point at which the data inside the tuple is alterable, really at that time the tuple data can be changed. The model program will assist you with understanding better.

Sets

Sets are an assortment of unordered components that are novel. Meaning that regardless of whether the data is rehashed more than one time, it would be gone into the set just a single time. It looks like the sets that you have learnt in number-crunching.

-

User-Defined Data Structures

Arrays versus Lists

Arrays and lists are a similar design with one distinction. Lists permit heterogeneous data component stockpiling while Arrays permit just homogenous components to be put away within them.

Stack

Stacks are linear Data Structures which depend on the principle of Last-In-First-Out (LIFO) where data which is entered last will be the first to get gotten to. It is fabricated using the array structure and has activities to be specific, pushing (adding) components, popping (deleting) components and accessing components just from one point in the stack called as the TOP. This TOP is the pointer to the current place of the stack. Stacks are prominently utilized in applications like Recursive Programming, reversing words, fix instruments in word editors, etc.

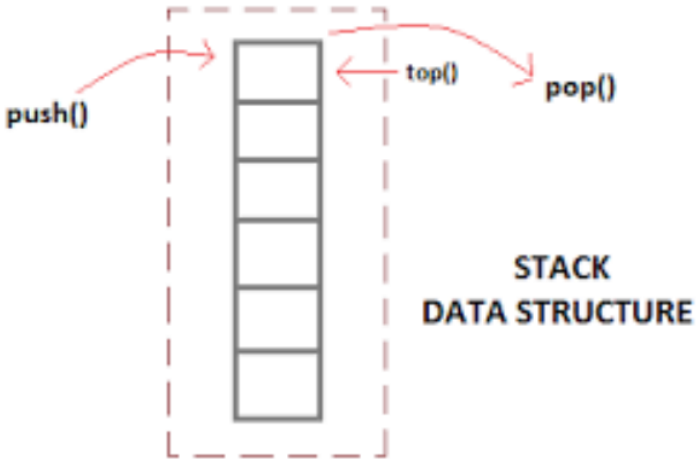


Fig 2.4: stack

Queue

A queue is additionally a linear data structure which depends on the principle of First-In-First-Out (FIFO) where the data entered first will be gotten to first. It is assembled using the array structure and has activities which can be performed from the two closures of the Queue, that is, head-tail or front-back. Tasks, for example, adding and deleting components are called En-Queue and De-Queue and accessing the components can be performed. Queues are utilized as Network Buffers for gridlock the executives, utilized in Operating Systems for Job Scheduling and some more.

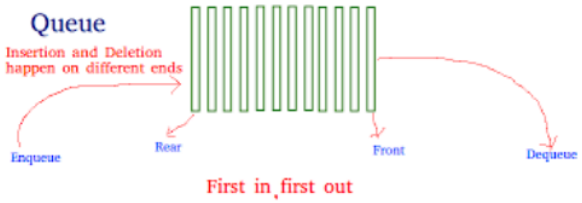


Fig 2.5: queue

Tree

A tree is a connected graph of nodes and edges which does not have a loop. It is a data structure for organizing data in computer science that uses information about the relationships between different pieces of data, usually by recursively relating smaller pieces to larger ones.

A node is a data object that holds information and may also contain references to other nodes. An edge represents a relationship or connection between nodes.

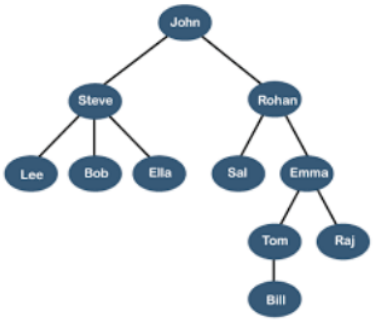


Fig 2.6: tree

Linked List

Linked lists are linear Data Structures which are not put away subsequently however are linked with one another using pointers. The node of a linked list is made out of data and a pointer called straightaway. These structures are most generally utilized in picture viewing applications, music player applications, etc.

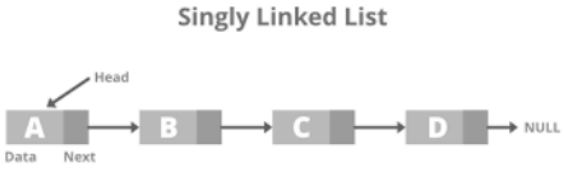


Fig 2.7: linked list

Graph

Graphs are utilized to store information assortment of points called vertices (nodes) and (edges). Graphs can be called as the most dependable portrayal of a certifiable map. They are utilized to find the different expense to-separate between the different information points called as the nodes and thus find the least way. Numerous applications like Google Maps, Uber, and a lot more use Graphs to find the least distance and increase benefits in the most effective ways.

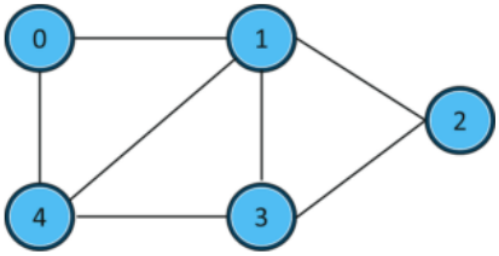


Fig 2.8: graphs

HashMaps

HashMaps are equivalent to what word references are in Python. They can be utilized to execute applications like phonebooks, populate information according to the rundowns and significantly more.

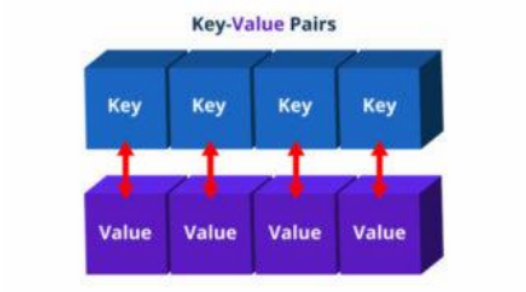


Fig 2.9: HashMaps

CHAPTER 3

3.1 Machine learning:

Machine learning is a piece of computerized reasoning (AI) and PC programming which bases on the utilization of information and calculations to mirror the way that people learn, reliably working on its precision.

AI is the field of study that gives PCs the ability to learn without being expressly customized. ML is one of the most thrilling advances that one would have at any point run over. As it is obvious from the name, it gives the PC that makes it more like people: The capacity to learn. AI is effectively being utilized today, maybe in a lot a bigger number of spots than one would anticipate.

Machine Learning(ML) can be clarified as mechanizing and further developing the learning system of PCs in light of their encounters without being really customized for example with no human help. The cycle begins with taking care of good quality information and afterwards preparing our machines(computers) by building AI models utilizing the information and various calculations. The selection of calculations relies upon what sort of information do we have and what sort of assignment we are attempting to robotize.

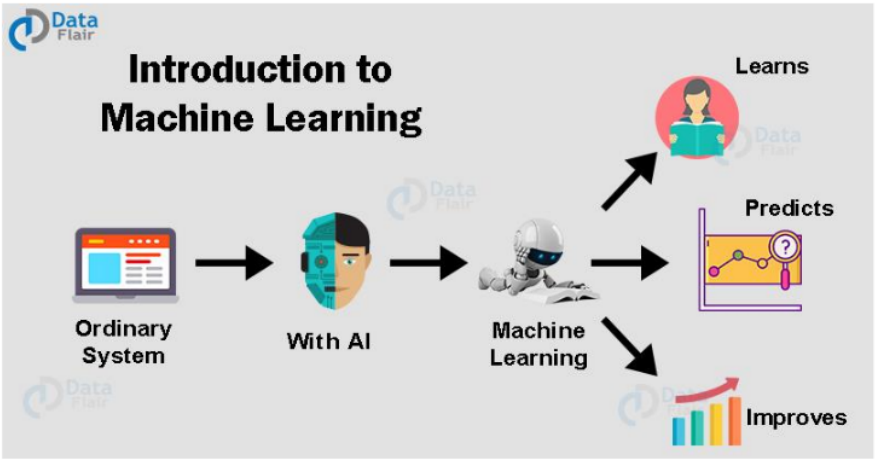


Fig 3.2: introduction to ML

Example of how ML work

We use SIRI to ask questions like “hi Siri how far is the nearest Subway”, a powerful speech recognition kicks off and converts the audio into its corresponding textual form which is then sent to the Apple service for the processing then neural network processing algorithms are run to understand the user's intent and finally Siri tells the answer. well, this is what machine learning is all about making machines learn and act like humans by feeding them with data and information without being explicitly programmed as we saw in the previous example when the data counting machines immediately start analysing data and eventually gets trained on it and learn it now when you date a point comes in machine accurately makes prediction and decisions based on the past data now that you know what is machine learning let's talk about supervised and unsupervised learning.

Supervised learning

Supervised learning is a machine learning task in which the algorithm learns from input with labels. The algorithm analyzes input data to make predictions and then an error is calculated to measure accuracy. A supervised machine learning algorithm can be trained on a training set of examples, each of which has a known label, being either right or wrong.

So supervised learning can be further divided into **classifications and regression.**

It is a classification problem when the output variable is categorical such as red or blue disease or no disease male-female there as its regression model when the output variable is it real or continuous for example salary based work experience weight based on height should create a predictive model showing trends and data.

Example 1: If I say will I get a salary raise or not it's classification and if I ask how much salary raise will I get that's regression. But if I say how much salary raise will I get that is regression.

Unsupervised learning

In unsupervised learning there is no super mention that is no training will be given to the machine allowing it to act on the data which is not labelled hence machine tries to identify patterns and get the response take a similar example as before this time we do not tell the machine whether it's Pune and if the machine identifies patterns for the given set and groups them based on the patterns similarities again and supervised learning can be for the grouped into clustering and association clustering is basically by the machine forms groups based on the behaviour of the data secondly association it is a rule-based machine learning to discover interesting relation between variables in large data sets for example which customer made similar product purchases is clustering there as association is which products were purchased together now let understand clustering with the help of an example to reduce the churn rate at telecom companies studies the behaviour of the customers based on average quality ration and internet usage and observes that while some customers call duration is quite high others have heavy internet usage the customers are grouped based on the observed behaviour analyst strategies adopted to minimise generate and maximize profit by suitable promotions campaigns as you can see the chart and the right inside customers in group is more data and also have high quality ration Ruby customers are heavy internet users by group c customers have high quality ration so group b will be given more data benefit plans by group c will be given cheaper call rates to buy their loyalty so this was the example of clustering now let understand association with another example let's say one goes to supermarket and buy this product se bread milk fruits feet and customer to NGOs and wise bread milk rice and better now when customer 3 goes in b gread it is highly likely that you will also find it hence relationship is established based on customer behaviour and recommendations Amit now let's look at some real life applications of unsupervised learning market basket analysis in machine learning model based on the algorithm that if you buy a certain group of items you are less more likely to buy another group of items cement plastering cement similar words share similar contacts people post queries on website On based clustering groups all responses in a cluster with same meaning to ensure that the customer find the information they want quickly and easily it plays an important role in information retrieval good browsing experience in comprehension delivery store optimisation machine learning models are used to predict the demand and keep up with the supply also to open stores when demand is more and

optimising rules for more efficient delivery going to pass data and behaviour can also use unsupervised machine learning models to identify accident prone areas based on the intensity of those accidents and the area in order to introduce safety measures by now I hope you want to supervised and unsupervised learning for a week and lets you are few differences between the two the most fundamental difference is that supervised learning uses known as legal data and unsupervised learning uses and label it has the input secondary supervised learning follows feedback mechanism and unsupervised learning does not also the most commonly used algorithms in supervised learning at decision tree logistic regression support vector machine accept and unsupervised learning k means clustering hierarchical clustering algorithm.

3. Reinforcement Learning:

How it functions: Using this algo, the machine is prepared to settle on explicit choices. It works thusly: the machine is presented to a climate where it trains itself constantly utilizing experimentation. This machine gains from previous experience and attempts to catch the most ideal information to settle on precise business choices. Illustration of Reinforcement Learning: Markov Decision Process

Working of Machine Learning processes:

1. Decision Process: by and large, AI calculations are utilized to make a figure or characterization. Considering a few information, which can be named or unlabeled, your calculation will make a check about a model in the information.

2. An Error Function: A mistake work serves to study the figure of the model. In the event that there are known models, a screw-up cutoff can make a relationship with surveying the precision of the model.

3. A Model Optimization Process: If the model fits better to the server farms in the arranging set, then, at that point, loads are acclimated to lessen the goof between the known model and the model check. The calculation will go over this assessment

and further foster measure, resuscitating troubles freely until an edge of precision has been met. develop a measure, reviving burdens independently until an edge of exactness has been met.

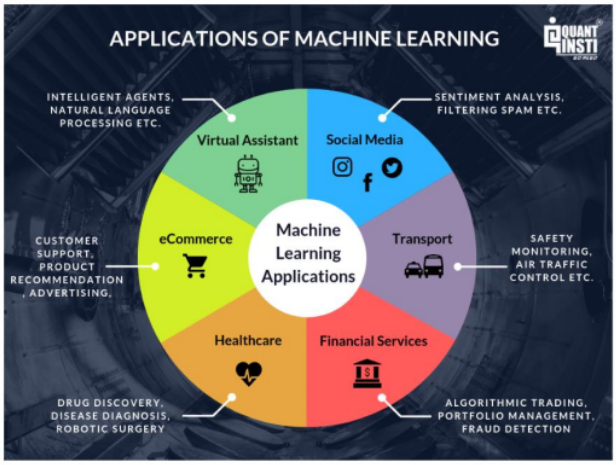


Fig 3.3: ML applications

Data and collection of data

Data are facts or information usually numeric that are collected through to making observations.
It is a set of values of quantitative or qualitative variables about one or more person objects used for discussion, calculation or reasoning

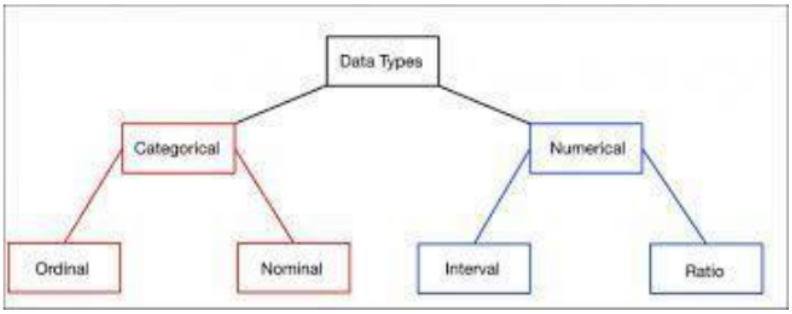


Fig 3.4: DATA TYPES

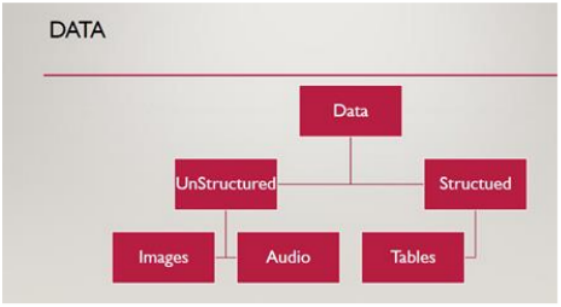


Fig 3.5: TYPES OF DATA

Data

Qualitative/Categorical

- Nominal
- Ordinal

Quantitative / Numerical

- Discrete
- Continuous

BASICS STEPS OF ML MODEL

1. Collect Data

2. Train Model

    • Iterate many times until it is good enough

3. Deploy Model

    Get data back and remodel

DATA IS MESSY

1. Garbage in, garbage out

2. Data Problems:

    - Incorrect Values
    - Missing values

EXPLORATORY DATA ANALYSIS

•In this stage, information engineers have a few inquiries close by and attempt to approve those inquiries by performing EDA.
•Exploratory Data Analysis, or EDA, is basically a kind of narrating for analysts.
•It permits us to reveal examples and experiences, frequently with visual techniques, inside information. EDA is frequently the initial step of the information demonstrating process
•Nonetheless, it's not really truly challenging to play out an EDA.
•EDA might sound colourful on the off chance that you are new to the universe of measurements.

EVALUATING REGRESSION MODELS

1. Mean Absolute Error
2. Mean Squared Error
3.R-Squared
4. Adjusted R-Squared

1) R-SQUARED METHOD

R Squared is an estimation that lets you to know degree the extent of difference in the reliant variable clarified by the fluctuation in the autonomous factors. In more straightforward terms, while the coefficients gauge patterns, R-squared addresses the spread around the line of best fit.

•For instance, in the event that the R^2 is 0.80, 80% of the variety can be clarified by the model's bits of feedbacks.

•Assuming the R^2 is 1.0 or 100 percent, that implies that all developments of the reliant variable can be totally clarified by the developments of the free factors.

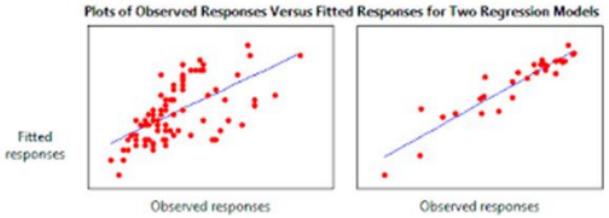


Fig 3.6: Comparision of the model with low r^2 and high r^2

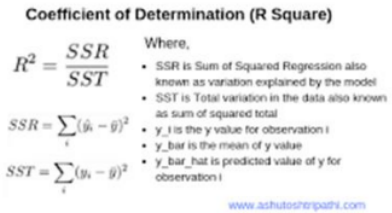


Fig 3.7: r squared formula

2) ADJUSTED R-SQUARED METHOD

Each extra autonomous variable added to a model generally builds the R^2 esteem - accordingly, a model with a few free factors might appear to be a superior fit regardless of whether it isn't.

•This is the place where Adjusted R^2 comes in.

•The changed R^2 makes up for each extra autonomous variable and possibly increments if each given variable works on the model above what is conceivable by likelihood.

$$Adjusted\ R^2 = 1 - \frac{(1 - R^2)(N - 1)}{N - p - 1}$$

Where

R^2 Sample R-Squared

N Total Sample Size

p Number of independent variable

Fig 3.8: adjusted r square formula

3) MEAN ABSOLUTE ERROR(MAE)

The absolute errorr is the contrast between the actual and predicted val. Consequently, the MAE is the normal of the absolute error.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\, x_i - x \,|$$

Fig 3.9: MAE formula

4) MEAN SQUARED ERROR(MSE)

The mean squared errorr or MSE is like the MAE, with the exception of you take the normal of the squared contrasts between the actual and predicted val.

Since the distinctions are squared, bigger mistakes are weighted all the more exceptionally, thus this ought to be utilized over the MAE when you need to limit large errors. The following is the condition for MSE, just as the code.

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

The square of the difference between actual and predicted

Fig 3.10: MSE formula

ASSESSING CLASSIFICATION MODELS

1.AUC-ROC Curve
2.Confusion Matrix and related metrics
3.F1 Score

CONFUSION MATRIX:

It is a matrix with 2 major classification i.e, positive and negative and further division true and false so : TP, FP, TN, FN

It is very valuable for estimating Accuracy, Specificity, Precision, Recall and in particular AUC-ROC bends.

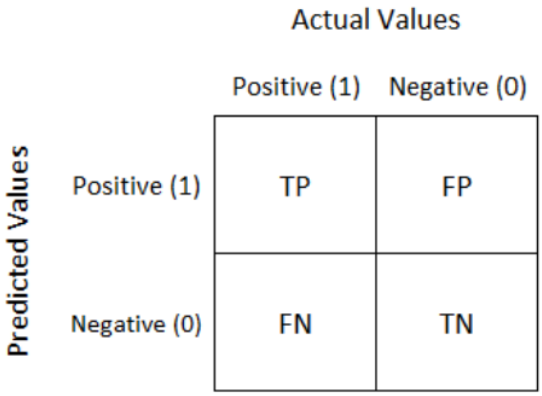


Fig 3.11: confusion matrix

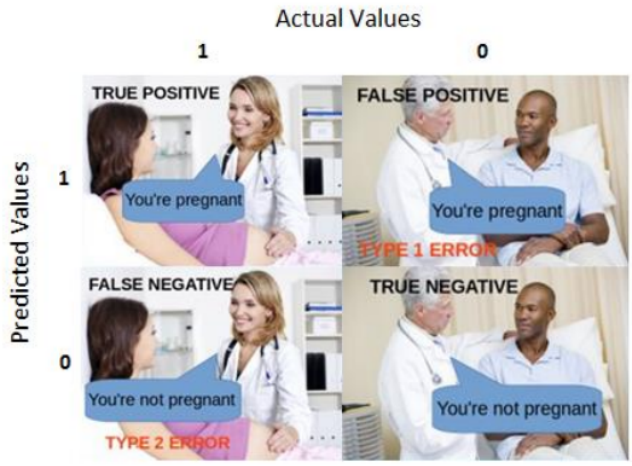


Fig 3.12: example to understand confusion matrix

- True Positive:

Understanding: we calculated output to be positive and the result was the same(T).

- True Negative:

Understanding: we calculated output to be negative and the result was not the same(T).

.False Positive: (Type 1 Error)

Understanding: we calculated output to be positive and the result was not the same (F).

- False Negative: (Type 2 Error)

Translation: we calculated output to be negative and the result was the same (F).

Remember,

Positive = true

Negative = false

F-score

The F1 score is a proportion of a test's exactness - it is the harmonic mean of recall & precision. It can attain the greatest score of 1 (perfect recall & precision) and at least 0. Generally, it is a proportion of the accuracy and vigor of your model.

$$F_1 = \frac{2}{\frac{1}{\text{recall}} \times \frac{1}{\text{precision}}} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$
$$= \frac{\text{tp}}{\text{tp} + \frac{1}{2}(\text{fp} + \text{fn})}$$

Fig 3.13: f score

Instances of machine learning applications:

Speech Recognition: It is a limit which uses standard language taking care of (NLP) to manage human talk into a created design. Various cells solidify talk affirmation into their structures to lead voice look for instance Siri-or give more noteworthy accessibility around informing.

Client care: Online chatbots are superseding human experts along the customer adventure. They answer sometimes presented requests around subjects, like transportation, or give modified appeal, decisively pitching things or proposing sizes for customers, changing the way wherein we consider customer responsibility across destinations and online media stages. Models informed bots for online business objections with virtual subject matter experts, illuminating applications, similar to Slack and Facebook Messenger, and tasks regularly done by modest partners and voice partners.

Mechanized stock trading: AI-driven high-repeat , Designed to smooth out stock portfolios, trading stages make thousands or even extraordinary many trades every day without human intervention.

Here is the rundown of generally utilized AI algos. These calculations practically used for:

- Gradient Boosting algorithms
    1. XGBoost
    2. CatBoost
    3. GBM
    4. LightGBM
- Random Forest
- Decision Tree
- SVM

- Naive Bayes
- Linear Regression
- K-Means
- Logistic Regression
- Dimensionality Reduction Algorithms
- kNN

3.2 Decision Tree:

Decision tree is highly noteworthy aslo notable instrument for forecast and classification. Its is a flowchart like tree structure, where each internal node demonstrates a test on a property, each branch tends to a consequence of the test, and each leaf node holds a class mark.

If all else fails, decision trees are managed an algorithmic way of thinking that recognizes ways of managing segment an informational record subject to various conditions. It is perhaps the most completely utilized and supportive framework for coordinated learning. Choice Trees are a non-parametric composed learning approach utilized for both characterization and relapse issues. The objective is to make a model that predicts the worth of an objective variable by taking in clear choice guidelines got from the information highlights.

Fig 3.7: decision tree

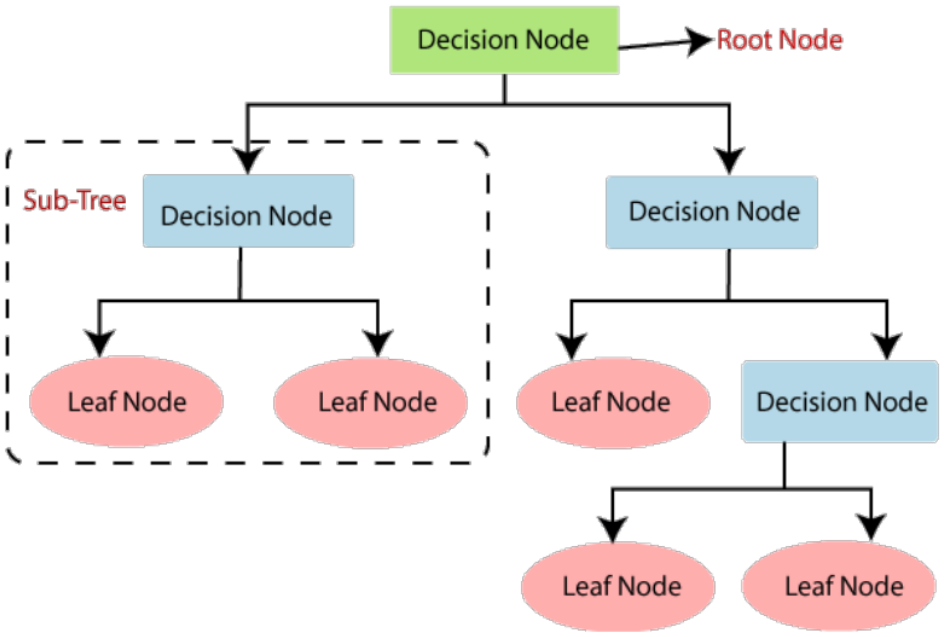


Fig 3.8: decision node

3.3Random Forest Classifier:

Random forest is an overseen learning algo. It very well may be utilized both for backslide and classification. It is additionally the most adaptable and simple to utilize assessment. A forest is incorporated trees. It is said that the more trees it has, the heartier a forest is. Random forest makes choice trees on arbitrarily picked information tests, gets presumption from each tree and picks the best course of action through projecting a surveying structure. It besides gives an amazingly decent pointer of the part importance. The underneath chart clarifies the working of the Random Forest calculation:

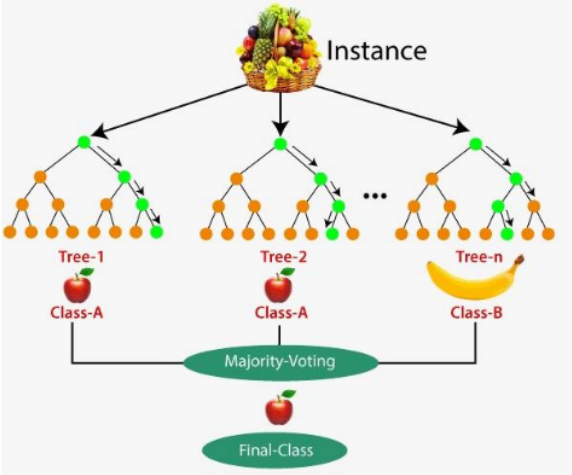


Fig 3.9: decision tree

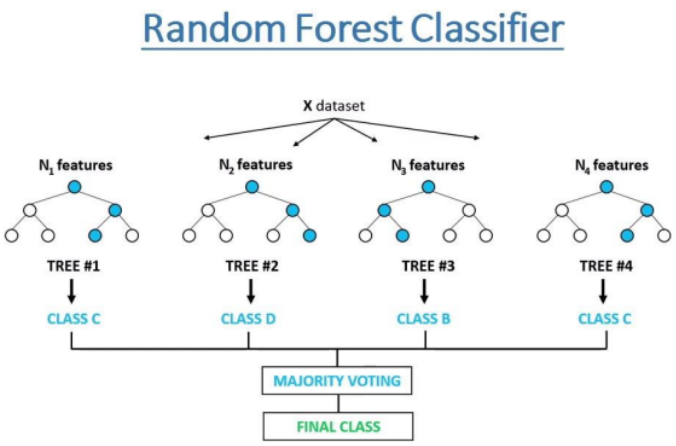


Fig 3.10: random forest classifier

Random forests have a combination of employments, similar to proposition engines, picture classification, and part assurance. It might be used to classify enduring progressed up-and-comers, identify tricky activity and expect ailments. It lies at the establishment of the Boruta computation, which picks significant arrangements in a dataset.

It really is an outfit method (in light of the gap and-overcome approach) of decision trees created on a randomly split dataset. This arrangement of decision tree classifiers is generally called the forest. The singular decision trees are made using a characteristic decision pointer, for instance, information obtain, procure extent, and Gini record for every quality. Each tree depends upon a free random model. In a classification issue, each tree votes, and the most standard class is picked as the finished result. By virtue of backsliding, the ordinary of all the tree yields is considered as the final result. It is more clear and even more exceptional diverged from the other non-direct classification estimations.

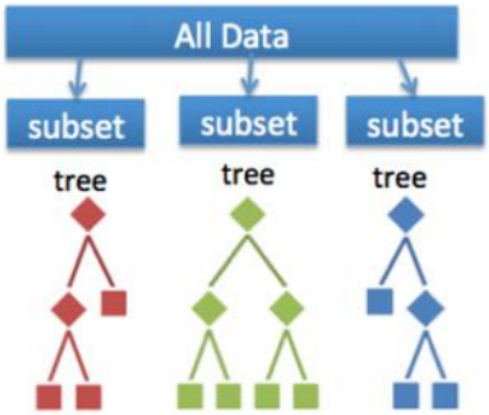


Fig 3.11: the creation of trees

Utilizations of Random Forest

Let's talk about fields in which RF is mostly utilized:

Promoting: the patterns of marketing can be classified.

Use of land: we can compare big /huge areas of land.

Medication: using this method we can easily identify patterns of sickness.

Banking: it is highly used in banking to foresee any credential hazards.

Impediments of Random Forest

Albeit random forest can be utilized for both arrangement and relapse assignments, it isn't more reasonable for Regression undertakings.

Python Implementation of RFA

Presently we will use python to execute the RFA tree. For this, we will utilize the equivalent dataset "user_data.csv", which has been used in past grouping models. By utilizing the equivalent dataset, we can contrast the RF classifier and other characterization models, for example, KNN, Decision tree Classifier, SVM, Logistic Regression, and so forth

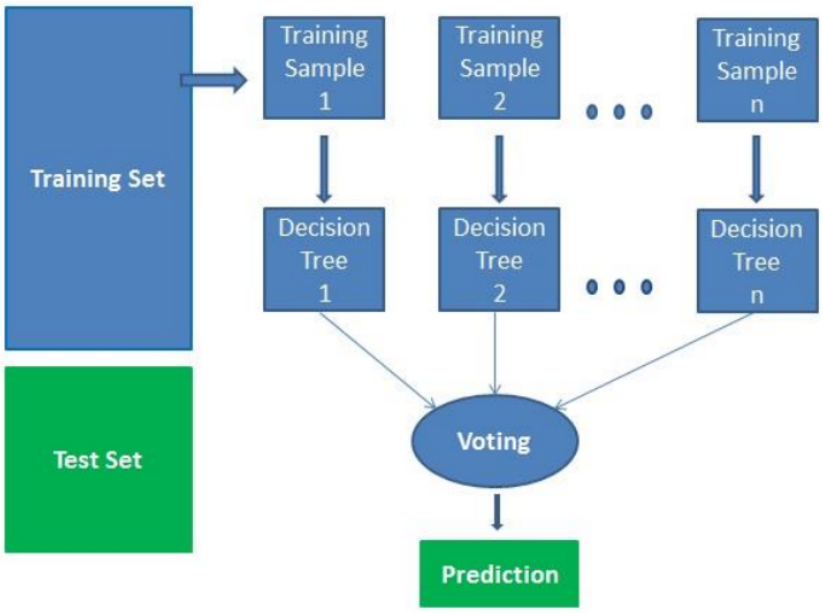


Fig 3.12: training and testing set

Advantages:

1. Random forest is considered an extraordinarily careful and strong methodology by the excellence of how much choice trees taking an interest in the meantime.
2. It doesn't experience the malicious effects of the overfitting issue. The focal explanation is that it takes the convenience of the enormous number of suspicions, which balances propensities.
3. The computation can be used in both arrangements and backslide issues.
4. Random forest can in addition oversee missing qualities. There are two distinct ways to deal with these: utilizing focus qualities to supersede unlimited factors, and taking care of the closeness weighted common of missing attributes.
5. You can get the general part importance, which helps in picking the most contributing plans for the classifier.

IMPLEMENTATION

In my project the attributes initially on which data will be evaluated and a prediction would be made are:
EmpNumber,
Age,
Gender,
EducationBackground,
MaritalStatus,
EmpDepartment,
EmpJobRole,
BusinessTravelFrequency,
DistanceFromHome,
EmpEducationLevel,
EmpEnvironmentSatisfaction,
EmpHourlyRate,
EmpJobInvolvement,
EmpJobLevel,
EmpJobSatisfaction,
NumCompaniesWorked,
OverTime,
EmpLastSalaryHikePercent,
EmpRelationshipSatisfaction,
TotalWorkExperienceInYears,
TrainingTimesLastYear,
EmpWorkLifeBalance,
ExperienceYearsAtThisCompany,
ExperienceYearsInCurrentRole,
YearsSinceLastPromotion,
YearsWithCurrManager,
Attrition,
PerformanceRating

How the data looks:

Out[14]:

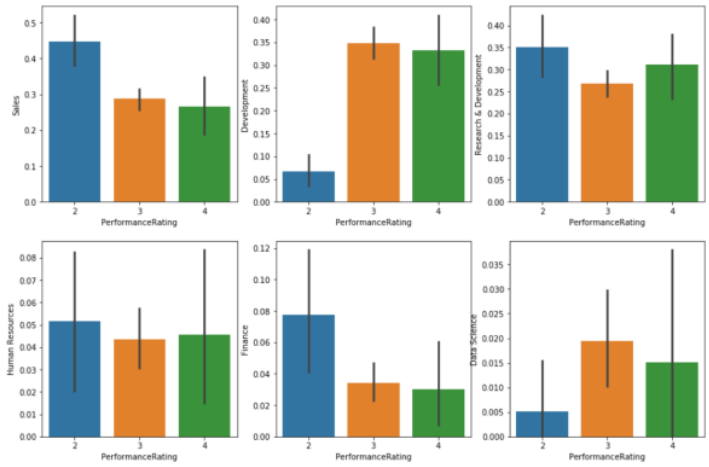
	EmpNumber	Age	Gender	EducationBackground	MaritalStatus	EmpDepartment	EmpJobRole	BusinessTravelFrequency	DistanceFromHome	EmpEducationLevel	...	EmpRelatio
0	E1001000	32	1	2	2	5	13	2	10	3	...	
1	E1001006	47	1	2	2	5	13	2	14	4	...	
2	E1001007	40	1	1	1	5	13	1	5	4	...	
3	E1001009	41	1	0	0	3	8	2	10	4	...	
4	E1001010	60	1	2	2	5	13	2	16	4	...	

5 rows × 28 columns

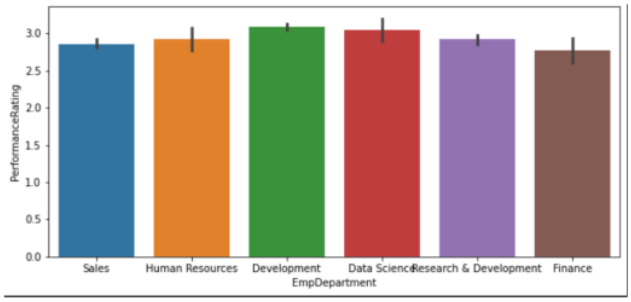
Out[14]:

ningTimesLastYear	EmpWorkLifeBalance	ExperienceYearsAtThisCompany	ExperienceYearsInCurrentRole	YearsSinceLastPromotion	YearsWithCurrManager	Attrition	PerformanceRating
2	2	10	7	0	8	0	3
2	3	7	7	1	7	0	3
2	3	18	13	1	12	0	4
2	2	21	6	12	6	0	3
1	3	2	2	2	2	0	3

Plotting a separate bar graph for performance of each department using seaborn



Department wise performance analysis



Out[15]:

	Age	Gender	EducationBackground	MaritalStatus	EmpDepartment	EmpJobRole	BusinessTravelFrequency	DistanceFromHome	EmpEducationLe
Age	1.000000	-0.040107	-0.055905	-0.098368	-0.000104	-0.037665	0.040579	0.020937	0.207
Gender	-0.040107	1.000000	0.009922	-0.042169	-0.010925	0.011332	-0.043608	-0.001507	-0.022
EducationBackground	-0.055905	0.009922	1.000000	-0.001097	-0.026874	-0.012325	0.012382	-0.013919	-0.047
MaritalStatus	-0.098368	-0.042169	-0.001097	1.000000	0.067272	0.038023	0.028520	-0.019148	0.026
EmpDepartment	-0.000104	-0.010925	-0.026874	0.067272	1.000000	0.568973	-0.045233	0.007707	0.019
EmpJobRole	-0.037665	0.011332	-0.012325	0.038023	0.568973	1.000000	-0.086251	0.022939	-0.016
BusinessTravelFrequency	0.040579	-0.043608	0.012382	0.028520	-0.045233	-0.086251	1.000000	-0.020935	0.002
DistanceFromHome	0.020937	-0.001507	-0.013919	-0.019148	0.007707	0.022939	-0.020935	1.000000	0.045
EmpEducationLevel	0.207313	-0.022960	-0.047978	0.026737	0.019175	-0.016792	0.002064	0.045856	1.000
EmpEnvironmentSatisfaction	0.013814	0.000033	0.045028	-0.032467	-0.019237	0.044612	0.012267	-0.017719	-0.037
EmpHourlyRate	0.062867	0.002218	-0.030234	-0.013540	0.003957	-0.016179	0.025400	0.013730	0.014
EmpJobInvolvement	0.027216	0.010949	-0.025505	-0.043355	-0.076988	-0.008034	0.016652	0.003231	0.027
EmpJobLevel	0.509139	-0.050685	-0.056338	-0.087359	0.100526	0.004406	0.036360	0.017270	0.100
EmpJobSatisfaction	-0.002436	0.024680	-0.030977	0.044593	0.007150	0.032916	-0.031236	-0.003036	0.000
NumCompaniesWorked	0.284408	-0.036675	-0.032879	-0.030095	-0.033950	-0.009111	0.021476	-0.021411	0.128
OverTime	0.051910	-0.038410	0.007046	-0.022833	-0.026841	0.015075	0.032229	0.024940	-0.021
EmpLastSalaryHikePercent	-0.006105	-0.005319	-0.009788	0.010128	-0.012661	0.005735	-0.041946	0.044974	0.002
EmpRelationshipSatisfaction	0.049749	0.030707	0.005652	0.026410	-0.050286	-0.043067	-0.032705	-0.009509	-0.016
TotalWorkExperienceInYears	0.680886	-0.061055	-0.027929	-0.093537	0.016065	-0.049529	0.042736	0.027306	0.151
TrainingTimesLastYear	-0.016053	-0.057654	0.051596	0.026045	0.016438	0.004452	0.006720	-0.032082	-0.013
EmpWorkLifeBalance	-0.019563	0.015793	0.022890	0.014154	0.068875	-0.007519	-0.040969	-0.044788	0.010
xperienceYearsAtThisCompany	0.318852	-0.030392	-0.009887	-0.075728	0.047677	-0.009047	-0.015029	0.021908	0.076
ExperienceYearsInCurrentRole	0.217163	-0.031823	-0.003215	-0.076663	0.069602	0.019383	-0.006541	0.019898	0.066
YearsSinceLastPromotion	0.228199	-0.021575	0.014277	-0.052951	0.052315	0.012190	-0.020824	0.013246	0.054

Feature selection:

There are a ton of segments in the indicator variable. In this way, the connection coeffecient is determined to see which of them are significant and these are then utilized for preparing techniques. From that point, we additionally get the top variables which influence execution. We can see that the main elements selected were Department , Last Salary Hike Percent, Work Life Balance, Job Role, Environment Satisafaction , Experience Years At This Company, Experience Years In Currrent Role, Years Since Last Promotion, Years With Current Manager. These were chosen on the grounds that their connection co-effecient with Performance Rating was more than 0.1.

Normalization and Label Encoding was additionally utilized for include change.

A different consideration thinking about every one of the indicators was done however it brought about diminishing the exactness. Essentially, Principal Component Analysis likewise diminishes the exactness.

Top 3 variables which influence the representative presentation are 1. Employee Environment Satisfaction, 2. Employee Last Salary Hike Percent and 3. Years Since Last Promotion.

After selecting the attributes which actually helps us predict performance of an employee after applying feature selection we have :

EmpDepartment,
EmpJobRole,
EmpEnvironmentSatisfaction,
EmpLastSalaryHike,

EmpWorkLifeBalance,
ExperienceYearsAtThisCompany,
ExperienceYearsInCurrentRole,
YearsSinceLastPromotion,
YearsWithCurrManager,

I trained my model on 70% data and saved 30% for testing. The model is trained on random forest algorithm. I tried training on other algorithms as well but with the amount of data (size of data) and the type of data, random forest performed best with highest accuracy among all with an accuracy of 90.5%.

CHAPTER 4

FUNDAMENTALS OF UI using tkinter

4.1 DESIGN GOALS

This mini project has ensured that the user has an interactive and explorable environment. The interface is user friendly, simple to understand and has tried to ensure that there are no bugs.

Tkinter is a module that is constructed in python which is used to create GUI applications. It is one of the most widely/popularly used modules for developing GUI applications in python as it is simple and easy to work with.

Programmers can create any GUI applications that they desire with the use of Tkinter.

Figure 4.1: flow of creating GUI

Widget is a component of Graphical User Interface (GUI) that presentations/delineates data or gives a way for the client to connect with the OS. In Tkinter , Widgets are oobjects ; occasions of classes that address buttons, outlines, etc.

Layout managers additionally are alluded to as geometry managers. They are utilized forsituating, organizing and enlisting gadgets on tkinter window. Python offers 3 design/calculation supervisors.

Tk empowers 3 sorts of geometry managers : placer, packer, & grid.

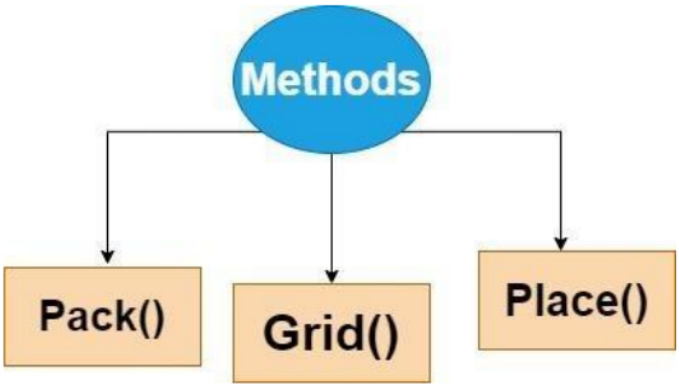


Figure 4.2: methods in Tkinter

Packer:

Pack is the ideal to apply of the 3geometrymanagers of Tk and Tkinter. Rather than getting to guarantee precisely in which a gadget need to appear at the show screen, we will guarantee the places of gadgets with the % order comparative with each other. The % order takes care of the subtleties. However the % order is less hard to apply, this organization administrators is obliged in its chances in contrast with the lattice and district troughs.

Placer:

The Place unadulterated arithmetic boss permits you explicitly set the arrangement and length of a window, each in outright terms, or comparative with each unique window. The placement supervisor is likewise gotten to by means of the situation system. it will be distributed to any or every single boundless gadget.

Grid:

Grid is in masses of times the awe inspiring decision for best in class use. While rate is sometimes now presently not adequate for changing measurements with inside the design,

region gives you entire control of situating every component, except this makes it masses more noteworthy complex than rate and Grid.

LABELS

Tkinter Label is a gadget that is utilized to carry out show boxes where you can put text or pictures. The text showed by this gadget can be changed by the designer at any time you need. It is likewise used to perform assignments, for example, to underline the piece of the text and length the text across various lines. It is essential to take note of that a name can utilize just a single textual style at a time to display text.

Figure 4.3: lables

BUTTON

The Button gadget is a customary Tkinter gadget, which is utilized for different sorts of buttons. A button is a gadget which is intended for the client to collaborate with, for example assuming the button is squeezed by mouse click some activity may be begun. They can likewise contain text and pictures like marks. While marks can show text in different textual styles, a button can show text in a solitary textual style. The text of a button can traverse more than one line.

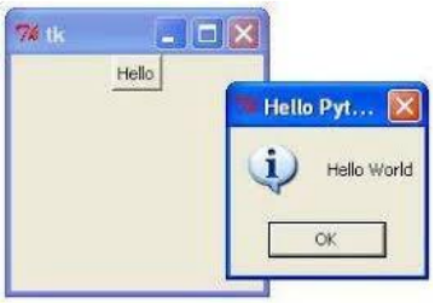


Figure 4.4: buttons

COMBOBOX

Combobox is a blend of Listbox and a entry field. It is one of the Tkinter gadgets where it contains a down bolt to choose from a rundown of choices. It helps the clients to select as indicated by the rundown of choices showed. At the point when the client taps on the drop-down bolt on the passage field, a spring up of the looked over Listbox is shown down the section field. The chose choice will be shown in the passage field just when a choice from the Listbox is chosen.

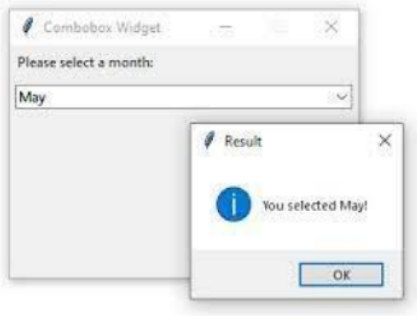


Figure 4.5: combobox

FRAME

An frame is a rectangular locale on the screen. An edge can likewise be utilized as an establishment class to execute complex gadgets. It is utilized to coordinate a gathering of gadgets.

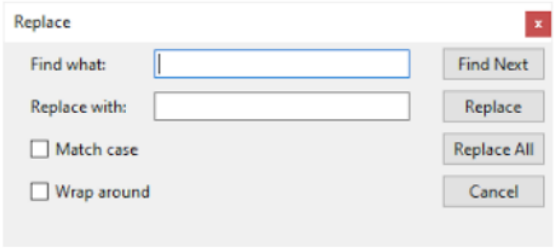


Figure 4.6: frame

Each different gadget is a Python object. While making a gadget, you should pass its parent as a boundary to the gadget creation work. The main exemption is the "root" window, which is the high level window that will contain all the other things and it doesn't have a parent.

WIDGETS	DESCRIPTION
Label	This widget is used to display text or image on the window/frame
Button	This widget is used to add buttons to the user interface
Canvas	This widget allows one to draw pictures and different types of layouts like texts, graphics etc.
Entry	This widget is used to take as input, a single line text entry from user
Frame	This widget is used as box or container. It holds and organizes the widgets in an orderly fashion
SpinBox	This widget allows users to select from a given number of values
ComboBox	This widget contains a down arrow to select from a list of options
CheckButton	This widget displays a number toggle buttons which represent various options from which user can select any number of options.
RadioButton	This widget is similar to the CheckButton but allows only one option to be selected
Scale	This widget is used to provide a slider which allows the user to select any value from the scale

Table 4.7: Various widgets available in Tkinter

CHAPTER 8

RESULTS

8.1 Launch Screen

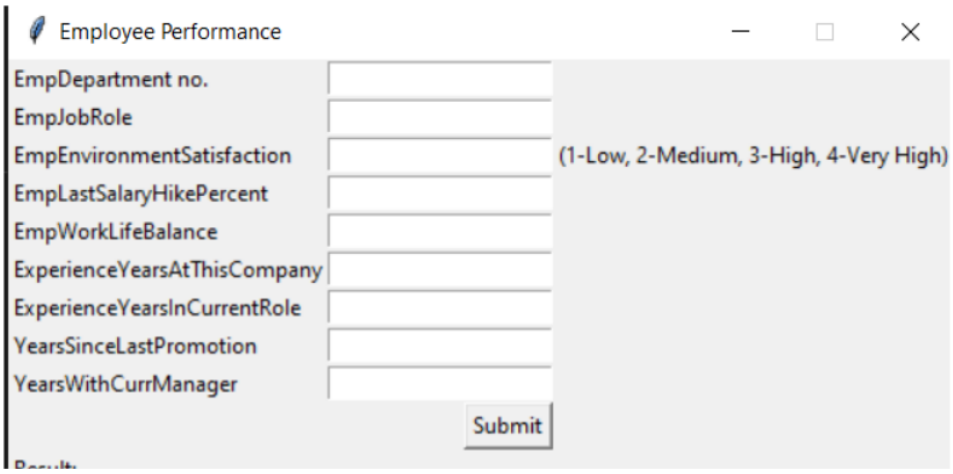
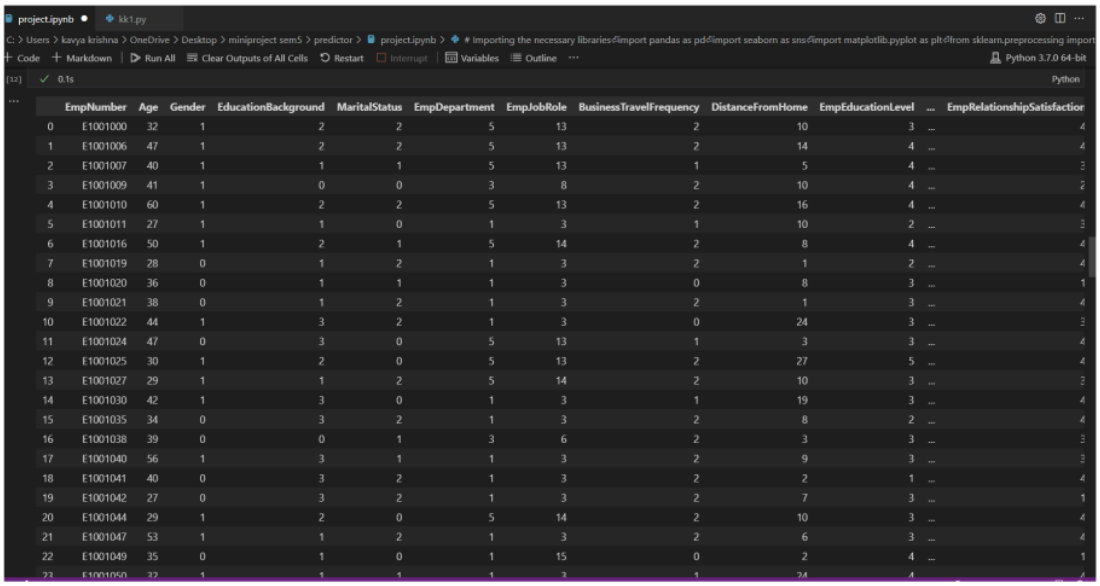


Figure 8.1: Screen Shot of Launch Screen

8.2 Screenshot of data set:



	EmpNumber	Age	Gender	EducationBackground	MaritalStatus	EmpDepartment	EmpJobRole	BusinessTravelFrequency	DistanceFromHome	EmpEducationLevel	...	EmpRelationshipSatisfactior
0	E1001000	32	1	2	2	5	13	2	10	3	...	4
1	E1001006	47	1	2	2	5	13	2	14	4	...	4
2	E1001007	40	1	1	1	5	13	1	5	4	...	3
3	E1001009	41	1	0	0	3	8	2	10	4	...	2
4	E1001010	60	1	2	2	5	13	2	16	4	...	4
5	E1001011	27	1	1	0	1	3	1	10	2	...	3
6	E1001016	50	1	2	1	5	14	2	8	4	...	4
7	E1001019	28	0	1	2	1	3	2	1	2	...	4
8	E1001020	36	0	1	1	1	3	0	8	3	...	1
9	E1001021	38	0	1	2	1	3	2	1	3	...	4
10	E1001022	44	1	3	2	1	3	0	24	3	...	3
11	E1001024	47	0	3	0	5	13	1	3	3	...	4
12	E1001025	30	1	2	0	5	13	2	27	5	...	4
13	E1001027	29	1	1	2	5	14	2	10	3	...	3
14	E1001030	42	1	3	0	1	3	1	19	3	...	4
15	E1001035	34	0	3	2	1	3	2	8	2	...	4
16	E1001038	39	0	0	1	3	6	2	3	3	...	3
17	E1001040	56	1	3	1	1	3	2	9	3	...	3
18	E1001041	40	0	3	2	1	3	2	2	1	...	4
19	E1001042	27	0	3	2	1	3	2	7	3	...	1
20	E1001044	29	1	2	0	5	14	2	10	3	...	4
21	E1001047	53	1	1	2	1	3	2	6	3	...	4
22	E1001049	35	0	1	0	1	15	0	2	4	...	1

Figure 8.2: data set

8.3 Screenshot of confusion matrix and accuracy:

```
# Training the model
from sklearn.tree import DecisionTreeClassifier

classifier_dtg=DecisionTreeClassifier(random_state=42,splitter='best')
parameters=[{'min_samples_split':[2,3,4,5],'criterion':['gini']},{'min_samples_split':[2,3,4,5],'criterion':['entropy']}]

model_griddtree=GridSearchCV(estimator=classifier_dtg, param_grid=parameters, scoring='accuracy',cv=10)
model_griddtree.fit(X_train,y_train)

model_griddtree.best_params_

# Predicting the model
y_predict_dtree = model_griddtree.predict(X_test)

# Finding accuracy
print(accuracy_score(y_test,y_predict_dtree))
print(classification_report(y_test,y_predict_dtree))
```

[25] ✓ 0.8s

```
0.9055555555555556
              precision    recall  f1-score   support

           2       0.85      0.83      0.84        63
           3       0.94      0.95      0.94       264
           4       0.75      0.73      0.74        33

    accuracy                           0.91       360
   macro avg       0.85      0.83      0.84       360
weighted avg       0.90      0.91      0.90       360
```

Figure 8.3: Output

8.4 OUTPUT

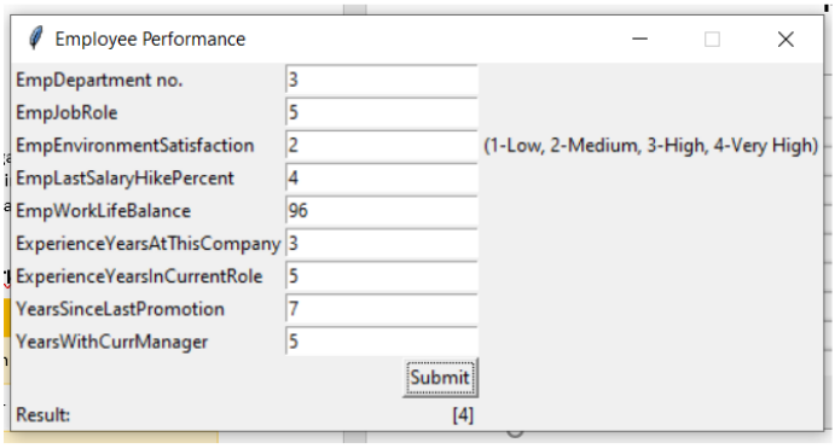


Figure 8.4: Output

CHAPTER 9

CONCLUSION

The project is successfully completed fulfilling all the requirements of the problem statements. The model works accurately for the provided data and has been trained using random forest algorithm while trying other algorithms as well which really helped me understand machine learning even better. This project has been designed keeping in mind the pace at which whole industry is being automated and this one section, i.e, HR system, which can be automated as well instead of continuing with the manual approach. This helps us save time and utilize it towards more tech evolution.

REFERENCES

[1] https://www.javatpoint.com/ (example for website referred)

{2] https://docs.python.org/

[3] A.Conci, J. E. R. de Carvalho, T. W. Rauber, A Complete System for Vehicle Plate Localization, Segmentation and Recognition in Real Life Scene, IEEE LATIN AMERICA TRANSACTIONS, VOL. 7, NO. 5, September 2009. (Example for paper referred)

[4] Joseph Yiu, The Definitive Guide to ARM Cortex-M3 and Cortex M4 Processor, 3rd Edition, Newness Publication (example for book referred)

KAVYA MP

ORIGINALITY REPORT

1%	%	1%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Nahid Sami, Asfia Aziz. "Machine Learning and Big Data: An Approach Toward Better Healthcare Services", Wiley, 2021 Publication	1%
2	Kesheng Wang, Quan Yu. "Chapter 22 Product Quality Inspection Combining with Structure Light System, Data Mining and RFID Technology", Springer Science and Business Media LLC, 2013 Publication	1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		