

CHAPTER 1

COMPANY PROFILE

It is pleasure in introducing “Karunadu Technologies Private Limited” as a leading IT software solutions and services industry focusing on quality standards and customer values. It is also a leading Skills and Talent Development company that is building a manpower pool for global industry requirements.

1.1 Profile



Fig 1.1 Company Logo

The company offers broad range of customized software applications powered by concrete technology and industry expertise. It also offers end to end embedded solutions and services. They deal with broad range of product development along with customized features ensuring at most customer satisfaction and also empower individual with knowledge, skills and competencies that assist them to escalate as integrated individuals with a sense of commitment and dedication.

1.1.1 Vision

To Empower Unskilled Individual with knowledge, skills and technical competencies in the field of Information Technology and Embedded engineering which assist them to escalate as integrated individuals contributing to company's and Nation's growth.

1.1.2 Mission

- Provide cost effective and reliable solutions to customers across various latest technologies.
- Offer scalable end-to-end application development and management solutions.
- Provide cost effective highly scalable products for varied verticals.
- Focus on creating sustainable value growth through innovative solutions and unique partnerships.
- Create design and deliver business solutions with high value and innovation by leveraging technology expertise and innovative business models to address long-term business objectives.
- Keep our products and services updated with the latest innovations in the respective requirement and technology.

1.1.3 Objectives

- To develop software and Embedded solutions and services focussing on quality standards and customer values.
- Offer end to end embedded solutions which ensure the best customer satisfaction.
- To build Skilled and Talented manpower pool for global industry requirements.
- To develop software and embedded products which are globally recognized .
- To become a global leader in Offering Scalable and cost effective Software solutions and services across various domains like E-commerce, Banking, Finance, Healthcare and much more.
- To generate employment for the skilled and highly talented youth of our Country INDIA

1.2 Company Products and Services Offered

1.2.1 Products

1 KECMS – Karunadu Enterprise Content Management System

Karunadu Enterprise Content Management System is a one stop solution for all our enterprise content management System relating to digital asset management, document imaging, workflow systems and records management systems. Increasing digitalization has led to an exponential growth in business content and managing this sea of unstructured data is tedious work. KECMS enables you to create, capture, manage, distribute, archive different forms of content, and has much more features.

2 KEMS – Karunadu Education Management System

Manage diversified data relating to education management on cloud. Educational data including students and staff is gathered over years which contain information from admission/appointment until leaving the Education. Statistical reports for the College/school can be generated along with admission Tracking and result analysis to keep track of progressive improvements of both student and staff.

3 KASS – Karunadu Advanced Security System

A Complete one stop embedded solution for large apartments. Security system which monitors door breakage, window breakage, gas leakage, motion detection and various other features which can be operated and maintained by centralized monitored system. This Embedded solution enhances the security measures of apartment/building and enhances the security of individuals may be from unintended intervention or from unauthorized access.

4 IT Solutions and Services

Karunadu Technologies is a Bangalore based IT Training and Software Development center with exclusive expertise in the area of IT Services and Solutions. Karunadu Technologies Pvt. Ltd. is also expertise in Web Designing and Consulting Services.

5 Embedded Design and Development

Karunadu Technologies Pvt. Ltd. has expertise in Design and development of embedded products and offers solutions and services in field of Electronics.

6 Academic Projects

Karunadu Technologies Pvt. Ltd. helps students in their academics by imparting industrial experience into projects to strive excellence of students. Karunadu Technologies Pvt. Ltd. encourages students to implement their own ideas to projects keeping in mind "A small seed sown upfront will be nourished to become a large tree one day", thereby focusing the future entrepreneurs. They have a wide range of IEEE projects for B.E, MTech, MCA, BCA, DIPLOMA students for all branches in each domain.

7 In plant Training

Karunadu Technologies Pvt. Ltd. provides Implant training for students according to the interest of students keeping in mind the current technology and academic benefit one obtains after completing the training. Students will be nourished and will be trained throughout with practical experience. Students will be exposed to industrial standards which boost their carrier. Students will become Acquaint to various structural partitions such as labs, workshops, assembly units, stores, and administrative unit and machinery units. They help students to understand their functions, applications and maintenance. Students will be trained from initial stage that is from collection of Project Requirements, Project Planning, Designing, implementation, testing, deployment and maintenance there by helping to understand the business model of the industry. Entire project life cycle will be demonstrated with hands on experience. Students will also be trained about management skills and team building activities. They assure that by end of implant training students will Enhance communication skills and acquire technical skills, employability skills, start-up skills, and will be aware of risks in industry, management skills and many other skills which are helpful to professional engagement.

8 Software Courses

Karunadu Technologies Pvt. Ltd. provides courses for students according to the interest of students keeping in mind the current technology and assist them for their further Employment. Company

provides various courses such as C, C++, VB, DBMS, Dot Net, Core Java and J2EE along with live projects.

1.3 Contact Details



#17, ATK complex, 4th Floor, Acharya College Main Road, Beside KarurVysya Bank,
Guttebasaveshwaranagar, Chikkabanvara, Bengaluru, Karnataka- 560090



support@karunadutechnologies.com

CHAPTER 2

DEPARTMENT PROFILE

Karunadu Technologies is a trailblazing technology solutions provider known for its unwavering commitment to innovation and talent development. With a rich legacy of expertise in cutting-edge technologies, Karunadu Technologies has emerged as a leading force in the industry, delivering state-of-the-art solutions and fostering the growth of aspiring professionals.

SKILL DEVELOPMENT AND INTERNSHIP

1. Identifying and training of manpower pool for global industry requirements.
2. Training and development of talents on all leading technologies.
3. Internship is provided for all academic courses to encourage upcoming global talents.
4. Conduct wide range of Training programs which includes courses, workshops, internship, Industrial talks for students and professionals on all latest technologies and trends.

REVEIW PAPER WRITING

Help in thesis consultation along with paper reviewing and organize points and relate currentwork of literature to the thesis.

Publish paper on internationally famed journals which include IEEE, SCI, Scopus, Springer,Elsevier, Taylor & Francis, Inder Science, Wiley and so on.

RESEARCH PROPOSAL

Help in research proposals during the development process in collaboration with the researcherand their requirements. We can be of great help to you while we develop the proposal in collaboration with you according to your requirements.

CHAPTER 3

BASICS OF PYTHON

3.1 INTRODUCTION

- Python is a general-purpose high-level programming language.
- Python was developed by Guido Van Rossum in 1989 while working at National Research Institute at Netherlands.
- But officially Python was made available to public in 1993. The official Date of Birth for Python is: Feb 20th 1993.
- Python is recommended as the first programming language for beginners.

Ex: In C:

```
#include<stdio.h>

void main()

{

    print("Hello world");

}
```

Output: Hello world

In Python:

```
Print("Hello world");
```

Output: Hello world

3.2 Use of Python

We can use it everywhere. The most common important application areas are:

- For developing Desktop Applications
- For developing web Applications
- For developing database Applications
- For Network Programming
- For developing games
- For Data Analysis Applications
- For Machine Learning and IOT
- For developing Artificial Intelligence Applications etc.....,

3.3 Features of Python

Simple and easy to learn.

- Python is a simple programming language. When we read Python program, we can feel like reading English statements.
- The syntaxes are very simple and only 30+ keywords are available.
- When compared with other languages, we can write programs with very less number of lines. Hence more readability and simplicity.
- We can reduce the development and cost of the project.

Freeware and Open Source

- We can use Python software without any license, and it is freeware.
- Its source code is open, so that we can customize based on our requirement.
- **Eg:** Python is customized version of Python to work with Java Applications.

High Level Programming language

- Python is high level programming language and hence it is a programmer friendly language.
- Being a programmer, we are not required to concentrate on low level activities like memory management and security etc.

Platform Independent

- Once we write a Python program, it can run on any platform without rewriting once again.
- Internally PVM is responsible for converting into machine understandable form.

Portability

- Python programs are portable. i.e. we can migrate from one platform to another platform very easily. Python programs will provide same results on any platform.

3.4 Limitations of Python

- Performance wise not up to the mark because it is interpreted language.
- Not using for mobile Applications

CHAPTER 4

MACHINE LEARNING

Machine Learning is a system that can learn from example through self-improvement and without being explicitly coded by programmer. The breakthrough comes with the idea that a machine can singularly learn from the data (i.e., example) to produce accurate results.

4.1 What is Machine Learning?

Machine learning combines data with statistical tools to predict an output. This output is then used by corporate to make actionable insights. Machine learning is closely related to data mining and Bayesian predictive modeling. The machine receives data as input, use an algorithm to formulate answers.

A typical machine learning task is to provide a recommendation. For those who have a Netflix account, all recommendations of movies or series are based on the user's historical data. Tech companies are using unsupervised learning to improve the user experience with personalizing recommendations. Machine learning is also used for a variety of task like fraud detection, predictive maintenance, portfolio optimization, automatize task and so on.

4.2 How does Machine learning work?

Machine learning is the brain where all the learning takes place. The way the machine learns is similar to the human being. Humans learn from experience. The more we know, the more easily we can predict. By analogy, when we face an unknown situation, the likelihood of success is lower than the known situation.

Machines are trained the same. To make an accurate prediction, the machine sees an example. When we give the machine a similar example, it can figure out the outcome. However, like a human, if it feed a previously unseen example, the machine has difficulties to predict.

4.3 Why Machine Learning?

The world today is evolving and so are the needs and requirements of people. Furthermore, we are witnessing a fourth industrial revolution of data. In order to derive meaningful insights from this data and learn from the way in which people and the system interface with the data, we need computational algorithms that can churn the data and provide us with results that would benefit us in various ways.

Machine Learning has revolutionized industries like medicine, healthcare, manufacturing, banking, and several other industries. Therefore, Machine Learning has become an essential part of modern industry.

Furthermore, machine learning has facilitated the automation of redundant tasks that have taken away the need for manual labour. All of this is possible due to the massive amount of data that you generate on a daily basis. Machine Learning facilitates several methodologies to make sense of this data and provide you with steadfast and accurate results. Different Types of Machine Learning

1. Supervised Machine Learning
2. Unsupervised Machine Learning
3. Reinforcement Machine Learning
4. Semi Supervised Machine Learning

CHAPTER 5

ALGORITHMS

5.1 Linear Regression

Definition

Linear Regression establishes a relationship between dependent variable (Y) and one or more independent variables (X) using a best fit straight line (also known as regression line).

It is represented by an equation $Y = a + b * X + e$, where a is intercept, b is slope of the line and e is error term. This equation can be used to predict the value of target variable based on given predictor variable(s).

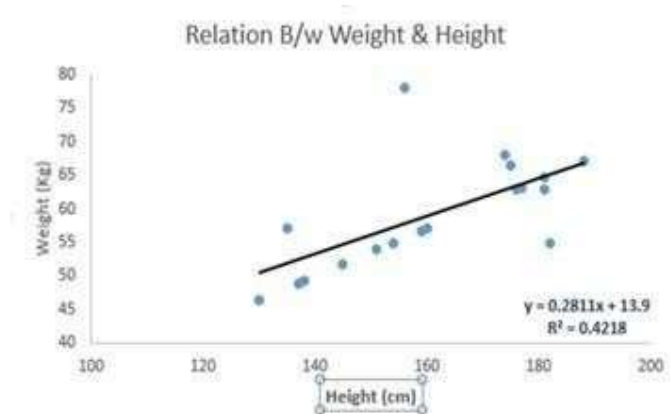


Figure 5.1 Regression analysis How to obtain best fitline

It calculates the best-fit line for the observed data by minimizing the sum of the squares of the vertical deviations from each data point to the line. Because the deviations are first squared, when added, there is no cancelling out between positive and negative values.

Advantages

Linear regression is an extremely simple method. It is very easy and intuitive to use and understand. A person with only the knowledge of high school mathematics can understand and use it. In addition, it works in most of the cases. Even when it doesn't fit the data exactly, we can use it to find the nature of the relationship between the two variables.

Disadvantage

- By its definition, linear regression only models relationships between dependent and independent variables that are linear. It assumes there is a straight-line relationship between them which is incorrect sometimes.
- Take for example most of your data lies in the range 0-10. If due to any reason only one of the data items comes out of the range, say for example 15, this significantly influences the

regression coefficients.

- Another disadvantage is that if we have a number of parameters than the number of samples available then the model starts to model the noise rather than the relationship between the variables.

5.2 Multiple Linear Regression

Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression (MLR) is to model the linear relationship between the explanatory (independent) variables and response (dependent) variable.

A simple linear regression is a function that allows an analyst or statistician to make predictions about one variable based on the information that is known about another variable. Linear regression can only be used when one has two continuous variables an independent variable and a dependent variable. The independent variable is the parameter that is used to calculate the dependent variable or outcome.

Definition

In many cases, there may be possibilities of dealing with more than one predictor variable for finding out the value of the response variable. Therefore, the simple linear models cannot be utilized as there is a need for undertaking Multiple Linear Regression for analyzing the predictor variables. Using the two explanatory variables, we can delineate the equation of Multiple Linear Regression as follows:

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \epsilon_i$$

The two explanatory variables x_{1i} and x_{2i} , determine y_i , for the i th data point. Furthermore, the predictor variables are also determined by the three parameters β_0 , β_1 , and β_2 of the model, and by the residual ϵ_i of the point i from the fitted surface.

General Multiple regression models can be represented as:

$$y_i = \sum \beta_j x_{ji} + \epsilon_i$$

Advantages of multiple regression

- Any disadvantage of using a multiple regression model usually comes down to the data being used. Two examples of this are using incomplete data and falsely concluding that a correlation is causation.
- When reviewing the price of homes, for example, suppose the real estate agent looked at only 10 homes, seven of which were purchased by young parents. In this case, the relationship between the proximity of schools may lead her to believe that this had an effect on the sale price for all

homes being sold in the community.

Disadvantages of multiple regression

- Multiple regression model usually comes down to the data being used. Two examples of this are using incomplete data and falsely concluding that a correlation is causation. When reviewing the price of homes, for example, suppose the real estate agent looked at only 10 homes, seven of which were purchased by young parents.
- In this case, the relationship between the proximity of schools may lead her to believe that this had an effect on the sale price for all homes being sold in the community.
- This illustrates the pitfalls of incomplete data. Had she used a larger sample, she could have found that, out of 100 homes sold, only ten percent of the home values were related to a school's proximity.

5.3 Logistic Regression

- Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.
- Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, **it gives the probabilistic values which lie between 0 and 1.**
- Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas **Logistic regression is used for solving the classification problems.**
- In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).
- The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.
- Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

- Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification. The below image is showing the logistic function:

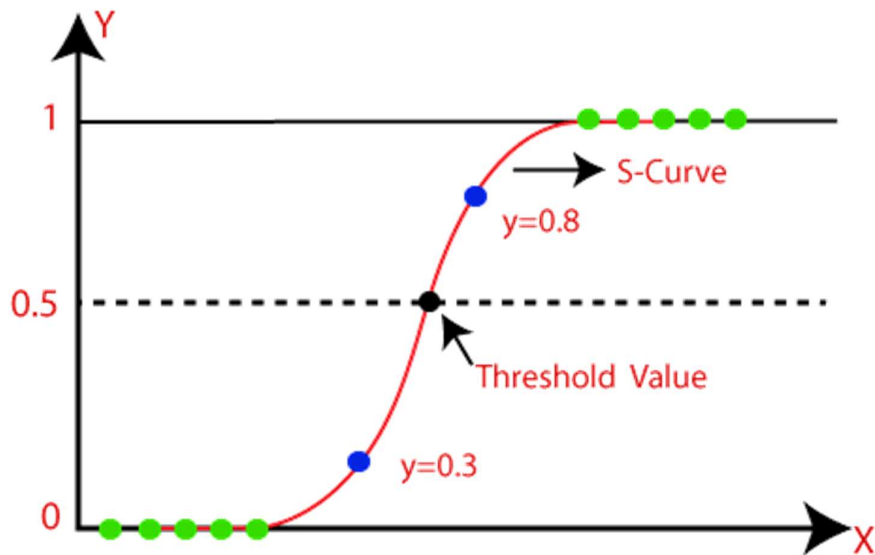


Figure 5.2: Logistic Regression Graph

5.4 KNN Algorithm

What is KNN Algorithm?

K nearest neighbour or KNN Algorithm is a simple algorithm which uses the entire dataset in its training phase. Whenever a prediction is required for an unseen data instance, it searches through the entire training dataset for k-most similar instances and the data with the most similar instance is finally returned as the prediction.

KNN is often used in search applications where you are looking for similar items, like find items similar to this one.

Advantage of KNN algorithm

- The algorithm is simple and easy to implement.
- There's no need to build a model, tune several parameters, or make additional assumptions.
- The algorithm is versatile. It can be used for classification, regression, and search.
- The training phase of K-nearest neighbour classification is much faster compared to other classification algorithms. There is no need to train a model for generalization that is why KNN is known as the simple and instance-based learning algorithm.
- KNN can be useful in case of nonlinear data. It can be used with the regression problem.

Output value for the object is computed by the average of k closest neighbours' value.

Disadvantage of KNN

- The algorithm gets significantly slower as the number of examples and/or predictors/independent variables increase.
- The testing phase of K-nearest neighbour classification is slower and costlier in terms of time and memory. It requires large memory for storing the entire training data set for prediction.
- KNN requires scaling of data because KNN uses the Euclidean distance between two data points to find nearest neighbours. Euclidean distance is sensitive to magnitudes. The features with high magnitudes will weigh more than features with low magnitudes.
- KNN also not suitable for large dimensional data.

5.5 Support Vector Machines (SVM)

Generally, Support Vector Machines is considered to be a classification approach, it but can be employed in both types of classification and regression problems. It can easily handle multiple continuous and categorical variables. SVM constructs a hyperplane in multidimensional space to separate different classes. SVM generates optimal hyperplane in an iterative manner, which is used to minimize an error. The core idea of SVM is to find a maximum marginal hyperplane (MMH) that best divides the dataset into classes.

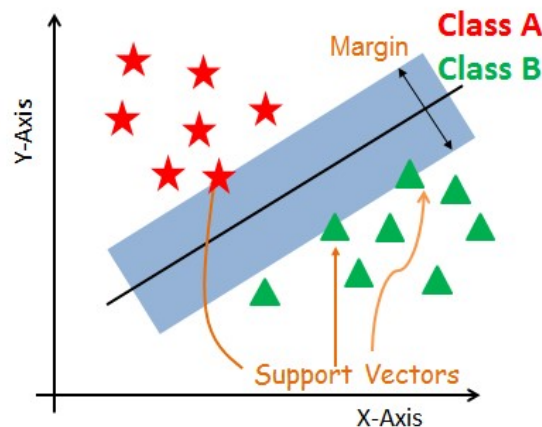


Figure 5.3: Support Vectors

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

5.6 Decision Tree Algorithm

- Decision Tree is a **Supervised learning technique** that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-

structured classifier, where **internal nodes** represent the features of a dataset, **branches** represent the decision rules and each **leaf node** represents the outcome.

- In a Decision tree, there are two nodes, which are the **Decision Node** and **Leaf Node**. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.
- The decisions or the test are performed on the basis of features of the given dataset.
- *It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.*
- It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.
- In order to build a tree, we use the **CART algorithm**, which stands for **Classification and Regression Tree algorithm**.
- A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.
- Below diagram explains the general structure of a decision tree:

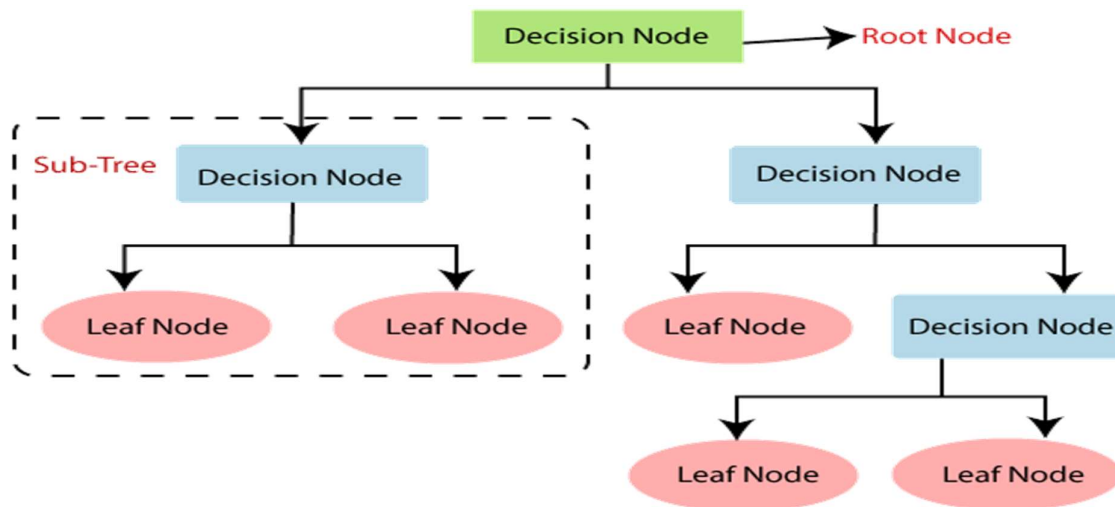


Figure 5.4: Decision Tree

How does the Decision Tree algorithm Work?

In a decision tree, for predicting the class of the given dataset, the algorithm starts from the root node of the tree. This algorithm compares the values of root attribute with the record (real dataset) attribute and, based on the comparison, follows the branch and jumps to the next node.

For the next node, the algorithm again compares the attribute value with the other sub-nodes and move further. It continues the process until it reaches the leaf node of the tree. The complete process can be better understood using the below algorithm:

- **Step-1:** Begin the tree with the root node, says S, which contains the complete dataset.

- **Step-2:** Find the best attribute in the dataset using **Attribute Selection Measure (ASM)**.
- **Step-3:** Divide the S into subsets that contains possible values for the best attributes.
- **Step-4:** Generate the decision tree node, which contains the best attribute.
- **Step-5:** Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

Advantages of the Decision Tree

- It is simple to understand as it follows the same process which a human follow while making any decision in real-life.
- It can be very useful for solving decision-related problems.
- It helps to think about all the possible outcomes for a problem.
- There is less requirement of data cleaning compared to other algorithms.

Disadvantages of the Decision Tree

- The decision tree contains lots of layers, which makes it complex.
- It may have an overfitting issue, which can be resolved using the **Random Forest algorithm**.
- For more class labels, the computational complexity of the decision tree may increase.

CHAPTER 6

TASKS ASSIGNED

6.1 INTRODUCTION

The objective of the internship is to apply theoretical knowledge of “Machine Learning using Python” to solve real time complex problems, in order to achieve these following basic concepts were learned:

- Python
- Machine Learning

Based on the concepts learned, projects were assigned.

6.2 PROJECT DESCRIPTION

The project was done using Python with a Django framework. The projects done were:

- **Heart Attack prediction using Random Forest algorithm:** Create a machine learning model to predict whether a person will have a heart attack or not.
- **Classification of Pulsar Waves using Logistic Regression:** Create a machine learning model to classify the Pulsar Waves based on the input data.

6.3 PROGRAMMING STEPS

- Import the required library (here pandas,sklearn,numpy).
- Import train_test_split from sklearn.model_selection
- Import StandardScaler from sklearn.preprocessing
- Import the required algorithm from the library
- Import confusion_matrix from sklearn.metrics
- Import accuracy_score from sklearn.metrics
- Provide the path of data file so that it can be included in our program
- Analyse the given data set
- Drop the columns and rows which are not needed
- Print the inputs and outputs as per the problem statement
- Create a model and train the model
- Scale the model (x_train and x_test)
- Provide the input and let the model predict the output
- Calculate the accuracy of the model using confusion matrix or by using accuracy_score()
- Also create the front-end using HTML and Django, with bootstrap for framework

6.4. Heart Attack prediction using Random Forest algorithm

The objective is to create a machine learning model to predict whether a patient will have Heart Attack or not.

Dataset

Age	Age of the patient.
Sex	The gender of the patient.
Chest Pain Type:	The type of chest pain experienced by the patient.
Resting Blood Pressure:	The resting blood pressure of the patient.
Cholesterol Levels	The cholesterol levels of the patient
Fasting Blood Sugar	Whether the patient's fasting blood sugar is greater than 120 mg/dL
Resting ECG Results	The results of the resting ECG.
Maximum Heart Rate	The maximum heart rate during the exercise test.
Exercise-Induced Angina	Whether the patient experienced angina during exercise.
ST depression (Old Peak)	The ST depression induced by exercise relative to rest.
Slope of ST Segment:	The slope of the ST segment during peak exercise
Number of Major Vessels	The number of major vessels (0-3) colored by fluoroscopy.
Thallium Stress Test:	The result of the thallium stress test.

Algorithm –Random Forest

age	sex	cp	trestbps	chol	fb	restecg	thalach	exang	oldpeak	slope	ca	thal	target
63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
56	0	1	140	294	0	0	153	0	1.3	1	0	2	1
44	1	1	120	263	0	1	173	0	0	2	0	3	1
52	1	2	172	199	1	1	162	0	0.5	2	0	3	1
57	1	2	150	168	0	1	174	0	1.6	2	0	2	1
54	1	0	140	239	0	1	160	0	1.2	2	0	2	1
48	0	2	130	275	0	1	139	0	0.2	2	0	2	1
49	1	1	130	266	0	1	171	0	0.6	2	0	2	1
64	1	3	110	211	0	0	144	1	1.8	1	0	2	1
58	0	3	150	283	1	0	162	0	1	2	0	2	1
50	0	2	120	219	0	1	158	0	1.6	1	0	2	1
58	0	2	120	340	0	1	172	0	0	2	0	2	1
66	0	3	150	226	0	1	114	0	2.6	0	0	2	1
43	1	0	150	247	0	1	171	0	1.5	2	0	2	1

Fig 6.1 Overview of Dataset

```

1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from sklearn.preprocessing import StandardScaler
4 from sklearn.ensemble import RandomForestClassifier
5 import numpy as np
6 from sklearn.metrics import accuracy_score
7
8 data=pd.read_csv("D:\\PROJECT\\Internship\\Project\\24_heartattackprediction/heart .csv")
9 print(data.info())
10 print(data.isnull().sum())
11
12 inputs=data.drop('target', 'columns')
13 outputs=data['target']
14
15 x_train,x_test,y_train,y_test=train_test_split(inputs,outputs,test_size=0.2)
16
17 sc=StandardScaler()
18 x_train= sc.fit_transform(x_train)
19 x_test=sc.fit_transform(x_test)
20
21 model=RandomForestClassifier()
22 model.fit(x_train,y_train)
23
24 y_pred=model.predict(x_test)
25
26 print("Accuracy: ",accuracy_score(y_test,y_pred)*100)
27
28 newinputs=np.array([[63,1,3,145,233,1,0,150,0,2.3,0,0,1]])
29 newinputs=sc.transform(newinputs)
30 print(model.predict(newinputs))

```

Fig 6.2 Code for Heart Attack prediction

```

def HeartAttack(request):

    data=pd.read_csv("others/heart .csv")
    inputs=data.drop('target', 'columns')
    outputs=data['target']
    x_train,x_test,y_train,y_test=train_test_split(inputs,outputs,test_size=0.2)

    sc=StandardScaler()
    x_train= sc.fit_transform(x_train)
    x_test=sc.fit_transform(x_test)

    model=RandomForestClassifier()
    model.fit(x_train,y_train)

    y_pred=model.predict(x_test)

    Accuracy="Accuracy: "+str(accuracy_score(y_test,y_pred)*100)

    submit=request.POST
    if 'submit' in submit:
        try:
            newinputs=np.array([[int(submit.get('txt1')),int(submit.get('txt2')),int(submit.get('txt3')),int(submit.get('txt4')),int(submit.get('txt5')),int(submit.get('txt6')),int(submit.get('txt7')),int(submit.get('txt8')),int(submit.get('txt9')),float(submit.get('txt10')),int(submit.get('txt11')),int(submit.get('txt12')),int(submit.get('txt13'))]])
            newinputs=sc.transform(newinputs)
            result=model.predict(newinputs)

            if int(result[0]):
                result="Higher Chance of Heart Attack"
            else:
                result= "Lower Chance of Heart Attack"

            return render(request,'HeartAttack.html',context={'result': result,'Accuracy':Accuracy})

        except:
            return render(request,'HeartAttack.html',context={'result': 'Please fill all the blocks' })
    return render(request,'HeartAttack.html')

```

Fig 6.3 Implementing Heart Attack prediction in Django

6.5 Pulsar Classification for Class Prediction using Logistic Regression

17898 entries

Data can be useful for prediction models of classification.

COLUMNS:

Based on Integrated Profile of Observation

Dataset

Mean Integrated	The mean value of a set of observations.
SD (Standard Deviation)	A measure of the amount of variation or dispersion in a set of data.
EK (Excess Kurtosis)	A measure of the shape of the probability distribution of a random variable.
Skewness	A measure of the asymmetry of the probability distribution of a real-valued random variable about its mean.
Mean_DMSNR_Curve	The mean value of DM SNR CURVE observations.
SD_DMSNR_Curve	The standard deviation of DM SNR CURVE observations.
EK_DMSNR_Curve	The excess kurtosis of DM SNR CURVE observations.
Skewness_DMSNR_Curve	The skewness of DM SNR CURVE observations.
Class	A binary category indicating class membership (0 or 1).

Algorithm – **Logistic Regression**

	A	B	C	D	E	F	G	H	I
1	Mean_Integrated	SD	EK	Skewness	Mean_DMSNR_Curve	SD_DMSNR_Curve	EK_DMSNR_Curve	Skewness_DMSNR_Curve	Class
2	140.5625	55.68378214	-0.234571412	-0.699648398	3.199832776	19.11042633	7.975531794	74.24222492	0
3	102.5078125	58.88243001	0.465318154	-0.515087909	1.677257525	14.86014572	10.57648674	127.3935796	0
4	103.015625	39.34164944	0.323328365	1.051164429	3.121237458	21.74466875	7.735822015	63.17190911	0
5	136.75	57.17844874	-0.068414638	-0.636238369	3.642976589	20.9592803	6.89649891	53.59366067	0
6	88.7265625	40.67222541	0.600866079	1.123491692	1.178929766	11.4687196	14.26957284	252.5673058	0
7	93.5703125	46.69811352	0.53190485	0.416721117	1.636287625	14.54507425	10.6217484	131.3940043	0
8	119.484375	48.76505927	0.03146022	-0.112167573	0.99916388	9.279612239	19.20623018	479.7565669	0
9	130.3828125	39.84405561	-0.158322759	0.389540448	1.220735786	14.37894124	13.53945602	198.2364565	0
10	107.25	52.62707834	0.452688025	0.170347382	2.331939799	14.48685311	9.001004441	107.9725056	0
11	107.2578125	39.49648839	0.465881961	1.162877124	4.079431438	24.98041798	7.397079948	57.78473789	0
12	142.078125	45.28807262	-0.320328426	0.283952506	5.376254181	29.00989748	6.076265849	37.83139335	0
13	133.2578125	44.05824378	-0.081059862	0.115361506	1.632107023	12.00780568	11.97206663	195.5434476	0
14	134.9609375	49.55432662	-0.135303833	-0.080469602	10.69648829	41.34204361	3.893934139	14.13120625	0
15	117.9453125	45.50657724	0.325437564	0.661459458	2.836120401	23.11834971	8.943211912	82.47559187	0
16	138.1796875	51.5244835	-0.031852329	0.046797173	6.330267559	31.57634673	5.155939859	26.14331017	0
17	114.3671875	51.94571552	-0.094498904	-0.287984087	2.738294314	17.19189079	9.050612454	96.61190318	0
18	109.640625	49.01765217	0.13763583	-0.256699775	1.508361204	12.07290134	13.36792556	223.4384192	0
19	100.8515625	51.74352161	0.393836792	-0.011240741	2.841137124	21.63577754	8.302241891	71.58436903	0
20	136.09375	51.69100464	-0.045908926	-0.271816393	9.342809365	38.09639955	4.345438138	18.67364854	0

Fig 6.4 Overview of Dataset

```

1 import pandas as pd
2 data=pd.read_csv("D:\\PROJECT\\Internship\\Project\\42_pulsarclassification/Pulsar.csv")
3 print(data.info())
4 print(data.isnull().sum())
5 inputs=data.drop('Class', 'columns')
6 output=data['Class']
7
8 from sklearn.model_selection import train_test_split
9 x_train,x_test,y_train,y_test=train_test_split(inputs,output,test_size=0.2)
10
11 from sklearn.linear_model import LogisticRegression
12 model=LogisticRegression()
13
14 model.fit(x_train,y_train)
15 y_pred=model.predict(x_test)
16 print(model.predict([[14.22,153.1,45.13,458.22,451,14,52,15]]))
17
18 from sklearn.metrics import accuracy_score
19 print("Accuracy: ",accuracy_score(y_test,y_pred)*100)

```

Fig 6.5 Code for Pulsar Waves Classification

```
def PulsarClassificaton(request):

    data=pd.read_csv("others\\Pulsar.csv")
    inputs=data.drop('Class','columns')
    output=data['Class']
    x_train,x_test,y_train,y_test=train_test_split(inputs,output,test_size=0.2)

    model=LogisticRegression()
    model.fit(x_train,y_train)
    y_pred=model.predict(x_test)
    cm=confusion_matrix(y_test,y_pred)
    accuracy="Accuracy: "+str((cm[0][0]+cm[1][1])/(cm[0][0]+cm[0][1]+cm[1][0]+cm[1][1])*100)

    submit=request.POST
    if 'submit' in submit:

        try:
            result=model.predict([[float(submit.get('txtmiv')),float(submit.get('txtsd')),float(submit.get('txtek')),float(submit.get('txtSkew')),
            float(submit.get('txtmdc')),float(submit.get('txtsdc')),float(submit.get('txtedc')),float(submit.get('txtSkewdc'))]])

            return render(request,'PulsarClassification.html',context={'result': 'Prediction Class: '+str(result[0]),'accuracy':accuracy})

        except:
            return render(request,'PulsarClassification.html',context={'result': 'Please fill all the blocks' })

    return render(request,'PulsarClassification.html')
```

Fig 6.6 Implementing Pulsar Waves Classification in Django

CHAPTER 7

REFLECTION NOTES

As per our experience during the internship, Karunadu technologies follows a good work culture and they have friendly employees, starting from the staff level to the management level.

I was assigned various tasks that involved working with different machine-learning algorithms and components. Throughout the internship, I gained knowledge and hands-on experience in Python programming language for Machine Learning so as to apply the theoretical knowledge to solve real-time and complex problems. The internship helped to find appropriate prediction models for the problems by applying suitable learning algorithms that can be used in the future. The internship project assigned by the company helped to improve my programming skills and to implement basic knowledge for solving real-world problems.

The trainers are well versed in their area and they treat everyone equally. There is no distinguishing between fresher graduates and corporates, and everyone is respected equally. There is a lot of teamwork followed in a task, be it hard or easy and there is a very calm and friendly atmosphere maintained at all times.

The projects I worked on during my internship also taught me the importance of planning and execution. We had to carefully plan out each project, taking into account the requirements, timelines, and resources available, before executing it. This helped us to avoid errors and complete the projects on time and within budget.

There is a lot of scope for self-improvement due to the great communication and support that can be found. Interns have been treated and taught well and all our doubts and concerns regarding the training or the companies have been properly answered.

All in all, Karunadu Technologies was a great place for a fresher to start a career and also for a corporate to boost his/her career. It has been a great experience to be an intern in such a reputed organization.

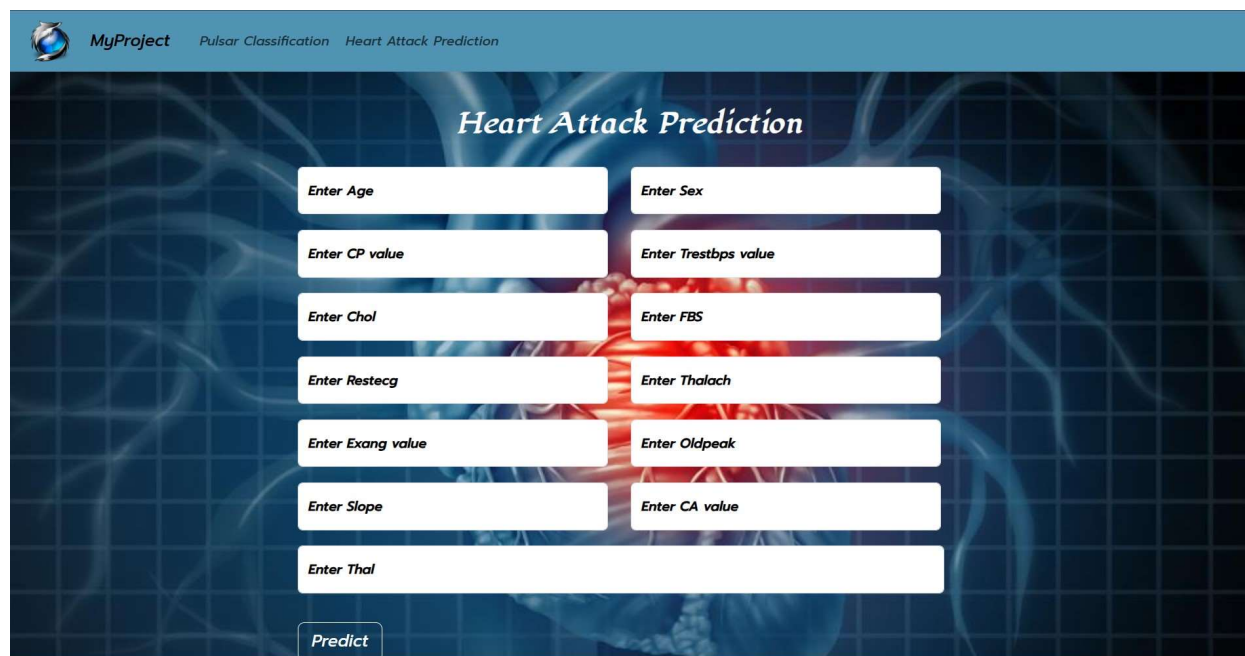
CHAPTER 8

RESULTS


8.1 HEART ATTACK PREDICTION

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   age         303 non-null   int64
1   sex         303 non-null   int64
2   cp          303 non-null   int64
3   trestbps    303 non-null   int64
4   chol        303 non-null   int64
5   fbs         303 non-null   int64
6   restecg     303 non-null   int64
7   thalach     303 non-null   int64
8   exang       303 non-null   int64
9   oldpeak     303 non-null   float64
10  slope       303 non-null   int64
11  ca          303 non-null   int64
12  thal        303 non-null   int64
13  target      303 non-null   int64
dtypes: float64(1), int64(13)
memory usage: 33.3 KB
None
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
Accuracy: 86.88524590163934
[1]
```

Fig 8.1 Output of Heart Attack Prediction



The image shows a web interface for a heart attack prediction model. At the top, there is a header with a logo and the text "MyProject Pulsar Classification Heart Attack Prediction". The main title "Heart Attack Prediction" is centered at the top of the form area. Below the title, there are ten input fields arranged in two columns. The left column contains: "Enter Age", "Enter CP value", "Enter Chol", "Enter Restecg", "Enter Exang value", "Enter Slope", and "Enter Thal". The right column contains: "Enter Sex", "Enter Trestbps value", "Enter FBS", "Enter Thalach", "Enter Oldpeak", and "Enter CA value". At the bottom left of the form, there is a "Predict" button.



This image shows the same web interface as the previous one, but with the output of the prediction. The input fields are still present, but the "Predict" button is now disabled. Below the input fields, the text "Lower Chance of Heart Attack" is displayed, followed by the accuracy value "Accuracy: 95.08196721311475".

Fig 8.2 Output of Heart Attack Prediction web interface

8.2 CLASSIFICATION OF PULSAR WAVES

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17898 entries, 0 to 17897
Data columns (total 9 columns):
#   Column                      Non-Null Count  Dtype
---  ---
0   Mean_Integrated             17898 non-null  float64
1   SD                          17898 non-null  float64
2   EK                          17898 non-null  float64
3   Skewness                    17898 non-null  float64
4   Mean_DMSNR_Curve           17898 non-null  float64
5   SD_DMSNR_Curve              17898 non-null  float64
6   EK_DMSNR_Curve              17898 non-null  float64
7   Skewness_DMSNR_Curve        17898 non-null  float64
8   Class                       17898 non-null  int64
dtypes: float64(8), int64(1)
memory usage: 1.2 MB
None
Mean_Integrated      0
SD                    0
EK                    0
Skewness              0
Mean_DMSNR_Curve     0
SD_DMSNR_Curve       0
EK_DMSNR_Curve       0
Skewness_DMSNR_Curve 0
Class                 0
dtype: int64
[0]
Accuracy: 98.10055865921787
```

Fig 8.3 Output of Classification of Waves

MyProject Pulsar Classification Heart Attack Prediction

Pulsar Classification

Enter Mean Integrated value

Enter SD value

Enter EK value

Enter Mean Skewness value

Enter Mean DMSNR Curve value

Enter SD DMSNR Curve value

Enter EK DMSNR Curve value

Enter Skewness DMSNR Curve value

Predict

Prediction Class: 1

Accuracy: 97.87709497206704

Fig 8.4 Django output for Pulsar Waves Classification

CHAPTER 9

CONCLUSION

Machine learning, with its ability to analyze vast datasets and make predictions or decisions without explicit programming, is at the forefront of technological innovation. It has a profound impact on diverse fields, from healthcare and finance to autonomous vehicles and e-commerce. The ability to harness the power of machine learning is not just a competitive advantage; it has become a necessity for organizations seeking to thrive in the modern landscape.

Python, as the programming language of choice for many machine learning applications, plays a pivotal role in this transformative process. Its simplicity, versatility, and extensive libraries make it an ideal tool for developing machine-learning algorithms and models. Python has emerged as the lingua franca of data science and artificial intelligence, enabling professionals to turn complex ideas into practical solutions efficiently.

In this era of rapid technological advancement, where data is the new currency and automation is redefining industries, my internship experience has underscored the importance of machine learning and Python in shaping our world. It has reinforced the notion that staying current with these technologies is not just an option but a fundamental requirement for anyone looking to make a meaningful impact in today's ever-evolving global landscape. As I move forward in my career, I am excited to continue exploring the limitless possibilities that machine learning and Python offer, confident in their significance in shaping the future.

REFERENCES

- www.karunadutechnologies.com
- Taiwo Ayodele, “Types of Machine learning algorithms”, 2018
- [https://www.geeksforgeeks.org/random-forest /](https://www.geeksforgeeks.org/random-forest/)
- [https://www.geeksforgeeks.org/logistic-regression /](https://www.geeksforgeeks.org/logistic-regression/)