MACHINE LEARNING INTERN(PYTHON)

A COMPARATIVE STUDY ON MACHINE LEARNING MODELS USING RAINFALL PREDICTION

Submitted in partial fulfillment for the award of certificate of

BACHELOR OF TECHNOLOGY
IN COMPUTER SCIENCE AND ENGINEERING

By

A.HIMASREE(208T1A0511)

DHANEKULA INSTITUTE OF ENGINEERING & TECHNOLOGY

GANGURU, VIJAYAWADA - 521 139

Affiliated to JNTUK, Kakinada Approved By AICTE,

New Delhi Certified by ISO 9001-2015, Accredited By NBA

DHANEKULA INSTITUTE OF ENGINEERING&TECHNOLOGY

GANGURU, VIJAYAWADA - 521139

Affiliated to JNTUK,Kakinada Approved By AICTE,New Delhi Certified by ISO 9001-2015,

Accredited by NBA

Department of Computer Science & Engineering

CERTIFICATE

This is to certify that the Summer Internship work entitled "**TECHNOLOGY VIRTUAL EXPERIENCE[A Comparative study on machine learning models using rainfall prediction]"** is a bonafide record of internship work done by A.HIMASREE(208T1A0511) for the award of the Summer Internship in Computer Science and Engineering by Jawaharlal Nehru Technological University, Kakinada during the academic year 2022- 2023.

Head of Department:

Dr. K. SOWMYA

Professor, HOD CSE

EXTERNAL EXAMINER

MACHINE LEARNING INTERNSHIP CERTIFICATE

codegnan IT SOLUTIONS PVT LTD

+91 63013 41478
info@codegnan.com
www.codegnan.com

Id: IjuP-Ah73Xx2023

14th July 2023,
Vijayawada.

CERTIFICATE OF INTERNSHIP

This is to certify that the **Arekapudi himasree**, a student of **DHANEKULA COLLEGE ENGINEERING & TECHNOLOGY**, has successfully completed the Internship Program as a "**Machine Learning Intern**" under the guidance and supervision of **K.Saketh Reddy-Data Science Mentor,** Codegnan IT Solutions Pvt Ltd, Vijayawada, from **15th May 2023 to 8th July 2023.**

During the internship, various tasks related to Web Scraping, Exploratory Data Analysis, Web Frameworks, Machine Learning Model Training, Testing and AWS Deployment were undertaken. A project on **"A Comparative Study of Machine Learning Methods for Rainfall Prediction"** was also completed.The candidate demonstrated professionalism, knowledge, and a result-oriented mindset throughout the internship, showcasing a theoretical and practical understanding of design work requirements.

The candidate exhibited a friendly, outgoing personality and performed well both as an individual and as a team member, meeting the management's expectations. On behalf of the company, I take this opportunity to wish the candidate all the very best in their future career endeavour and have a smooth life.

For Codegnan IT Solutions Pvt Ltd.

T. Saikiran

T. Saikiran

HR Manager

40-5-19/16, Prasad Naidu Complex,
P.B.Siddhartha Busstop, Moghalrajpuram
Vijayawada-520010

DHANEKULA INSTITUTE OF ENGINEERING & TECHNOLOGY

Department of Computer Science & Engineering

VISION – MISSION - PEOs

Institute Vision	Pioneering Professional Education through Quality
Institute Mission	Providing Quality Education through state-of-art infrastructure, laboratories and committed staff. Molding Students as proficient, competent, and socially responsible engineering personnel with ingenious intellect. Involving faculty members and students in research and development works for the betterment of society.
Department Vision	To empower the budding talents and ensure them with probable employability skills in addition to human values by optimizing the resources.
Department Mission	* To encourage students to become pioneers in the global competition with problem-solving skills * To make students become innovative with potential skills to explore the employment opportunities and/or to become entrepreneurs * To promote Research environment and inculcate corporate social responsibility
Program Educational Objectives (PEOs)	Graduates of Computer Science & Engineering will: PEO1: Excel in problem solving and designing new products for a competitive and challenging business environment PEO2: Contribute to technological innovation, research and society through the application of information technology in a diversified world.

PROGRAM OUTCOMES(POs)

1. **Engineering knowledge**: apply the knowledge of mathematics, science, engineering fundamentals, and anengineering specialization to the solution of complex engineering problems.

2. **Problem Analysis**: identify, formulate, review research literature, and analyze complex engineering problems reaching sustained conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. **Design/Development Of Solutions**: design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. **Conduct Investigations Of Complex Problems**: use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. **Modern Tool Usage**: create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. **The Engineer And Society:** apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment And Sustainability**: understand the impact of the professional engineering solutions in societaland environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. **Ethics**: apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual And Team Work**: function effectively as an individual, and as a member or a leader in diverse teams, and in multidisciplinary settings.

10. **Communication:** communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project Management And Finance:** demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life- Long Learning**: recognize the need for, and have the preparation and ability to engage inindependent and life- long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1: Designing and developing Computer Science based systems with high professional skills.

PSO2: Qualify in national and international level competitive examinations for successful higher studies and get employment in Computer Science enabled industries

Internship Mappings

Project Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
Rainfall prediction [machine learning Using python]														

Mapping Level	Mapping Description
1	Low Level Mapping with PO & PSO
2	Moderate Mapping with PO & PSO
3	High Level Mapping with PO & PSO

A.HIMASREE
208T1A0511
4-1 Sem(A)

Contents

1. Internship carried out Company/Organization Details

CODEGNAN IT SOLUTIONS PRIVATE LIMITED VIJAYAWADA.

- **Codegnan** started with a mission to make our Andhra Pradesh a machine learning hub.

- To make this happen they have to start from very roots and hence training and tutoring students on edge technologies like python, data science and artificial intelligence.

- At the same time building strong foundation in technologies like Internet of Things, MERN stack etc
  .
- It is founded in the year 2018.

- Headquarters of this company is located in Vijayawada , Andhra Pradesh.

- **Speciality internships of this company are**
  1. Data Science Training
  2. Python Training
  3. Digital marketing
  4. Web Development
  5. Machine Learning Training
  6. MERN stack Training

2. Internship Log.

Log Book: 60-Day ML Internship

Day 1:

Introduction to Data Science and Artificial Intelligence

Importance of Data in the 21stCentury

Types of Data and its usage

Difference between Data Science and Artificial IntelligenceOverview

of Data Analysis and key steps involved

How does Machine Learning work?

Different stages of Machine Learning projects

Understanding the Importance of Path and Installation of Jupyter Notebook Modules

Day 2:

What is a Python Module? How is it different from a Python File? Creating Modules and Packages

Importing Functions, Variables from different modules

Python built-in modules

Working on math, time, random modules

Day 3:

Hands-On – Working on Python Built-in Modules and User-defined Custom Module

Understanding Importance of Data Analysis

Understanding Importance and Types of Data AnalysisUnderstanding

types of Data

Day 4:

Understanding Importance of Visualization and Types of Graphs Understanding the

Importance and Usage of Jupyter Notebook

Understanding and Working on NumPy
Introduction to NumPy
Advantages of NumPy over lists
Creating NumPy arrays - 1-D, 2-D, N-D arrays
Data types for ndarrays

Day 5:

Checking the attributes - shape, size, dimensions, dtype

NumPy Arithmetic Operations

NumPy universal functions Linear

Algebra using NumPy

Working on Appending and Concatenating

Day 6:

Pandas for Data Analysis

Getting Started with Pandas

Introduction to Pandas Data Structures - Series, DataFrames Checking Attributesand Description

Day 7:

Basic Essential functionality - Reindexing, Dropping entries from an axis Indexing, Selection,and Filtering

Working on loc and iloc FunctionalitiesData

Loading and Storage

Reading and writing different file types (.txt, .xlsx, .csv files)

Day 8:

Interacting with Web APIs

Accessing data from databases

Writing/Saving Files

Day 9:

Data Cleaning and Preparation

Handling Missing Data - Filtering out Missing Data, Filling in Missing data Removing Duplicates

Computing Indicator/Dummy Variables

Day 10:

Data Wrangling

Concatenation - Adding Rows, Adding Columns, Concatenation with Different Indices Merging DataFrames

Day 11:

Data Aggregation and Group Operations

Pivot Tables and Cross-Tabulation

Working on Time Series Data

Hands-on: Case Study Working on Titanic Dataset for Cleansing Data, HR Analytics Data

Day 12:

Data Visualization Using Matplotlib

Introduction to Matplotlib

Setting Labels, Titles, xticks, and yticks

Multiple Line Plots, adding legend

Bar charts - What are they, When to use itBar

chart for comparing categorical data

Histogram to check the distribution of numerical data

Day 13:

Scatter Plots and their UsagePie

charts and their usage Subplots

and their Usage

Hands-on: Case Study Working on Titanic Dataset for Visualization

Day 14:

Python Interactive Visualization using Plotly for Dashboards

Introduction to Plotly and Cufflinks

Loading Plotly and Cufflinks

Loading the Data

Day 15:

Quick Visualization with custom bar charts

Interactive Bubble charts

Understanding and Working on Choropleth Maps

Hands-on: Analyzing Gapminder dataset

Day 16:

Data - Wealth of the 21st Century - Web Scraping using Python Why Web Scraping and Understanding its importance

Installing BeautifulSoup

Understanding web structures

Scraping data from the web using Beautiful Soup - Static & Dynamic websites Performing Data Visualization over the scraped data

Day 17:

Machine Learning Fundamentals

Data Transformation and PreprocessingHandling

Numeric Features

Feature Scaling

Standardization and Normalization

Day 18:

Handling Categorical Features

One Hot Encoding, pandas get_dummies

Label Encoding
More on different encoding techniquesTrain,

Test and Validation Split

Simple Train and Test Split

Drawbacks of train and test splitK-

fold cross-validation

Time-based splitting

Day 19:

Overfitting And Underfitting

What is overfitting? What

causes overfitting?What is

Underfitting?

What causes underfitting? What

are bias and Variance?

How to overcome overfitting and underfitting problems? Day 20:

Supervised Machine Learning Algorithms

Regression and its Importance in real-world cases Introduction toLinear

Regression

Understanding How Linear Regression Works Day 21:Maths

behind Linear Regression

Ordinary Least Square
Gradient Descent
R - Square
Adjusted R-square

Day 22:

Polynomial Regression
Multiple Regression

Performance Measures - MSE, RMSE, MAEAssumption

of Linear Regression

Ridge and Lasso regression

Hands-on: Algorithm implementation with real use case datasets Day 23:

Building and Deployment of Machine learning model - Flask, Git, Github & PythonAnywhere

Understanding steps in end-to-end ML projects

Building a web service for Machine Learning ModelGit

Download and Github Usage

Deploying the Final Trained Model on PythonAnywhere

Day 24:

Understanding Classification Modelling Approach

Introduction to the Classification problem

Why the name Regression? and Implementation of the Sigmoid Function

Day 25:

Working on a dataset for Logistic Regression Performance

Metrics for Classification Algorithms

Accuracy Score Confusion Matrix, Precision-Recall F1-Score, ROC Curve and AUC, Log Loss

Day 26:

Decision Trees

Introduction to Decision Tree

Homogeneity and Entropy Gini

Index

Information Gain

Advantages of Decision Tree

Day 27:

Preventing Overfitting Plotting

Decision Trees Plotting feature

importance

Regression using Decision Trees

Hands-On - Decision Tree on US Adult income dataset

Day 28:

Ensemble Learning

Introduction to Ensemble Learning

Bagging (Bootstrap Aggregation)

Constructing random forests Runtime

Case study on Bagging

Day 29:

Tuning hyperparameters of random forest (GridSearch, RandomizedSearch) Measuring model performance

Day 30:

Boosting Gradient

Boosting

Adaboost and XGBoost

Case study on boosting trees

Day 31:

Hyperparameter tuning

Evaluating performance

Stacking Models

Hands-On - Talking Data Ad Tracking Fraud Detection case study

Day 32:

Naive Bayes

Refresher on conditional Probability

Bayes Theorem

Examples of Bayes theorem Exercise

problems on Naive BayesNaive Bayes

Algorithm

Day 33:

Assumptions of Naive Bayes Algorithm

Laplace Smoothing

Naive Bayes for Multiclass classification

Handling numeric features using Naive Bayes

Measuring performance of Naive Bayes

Hands-On - Working on Spam detection and Amazon Food Reviewdataset

Day 34:

Support Vector Machines

Introduction to SVM What

are hyperplanes? Geometric

intuition Maths behind

SVM

Loss Function

Kernel trick

Polynomial kernel, RBF, and linear kernels

Day 35:

SVM Regression

Tuning the parameter

GridSearch and RandomizedSearch

Hands-On - Case Study SVM on Social network Ads

Day 36:

K Nearest Neighbors

Introduction to KNN

Effectiveness of KNN

Distance Metrics

Accuracy of KNN

Day 37:

Effect of an outlier on KNN

Finding the k Value

KNN on regression

Where not to use KNN

Hands-On - Case Study on E-Commerce Recommendation

Day 38:

Unsupervised Machine Learning Algorithms

Introduction to Unsupervised LearningK

Means Geometric intuition

Maths Behind KMeans

Day 39:

Determining the right k

Evaluation metrics for KMeans

Case study on K Means

Introduction and Working on Hierarchical Clustering Day 40:

Dimensionality Reduction Techniques

What are the dimensions?

Why is high dimensionality a problem?

Introduction to MNIST dataset with (784 Dimensions)Into
Dimensionality reduction techniques

PCA (Principal Component Analysis) for dimensionality reduction Hands-on

Day 41:

Pythonanywhere Deployment

Day 42:

AWS Deployment

Day 43 - 60:

Project Development

3. Domain area of the Internship

Domain Area: Machine Learning Using Python

Project Title: A Comparative Study on machine learning models using rainfall prediction Or Rainfall forecast

Python

- Python is an Interpreted, object-oriented, high-level programming language with dynamic semantics.
- Its high-level built indata structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.
- Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance.
- Python supports modules and packages, which encourages program modularity and code reuse.
- The Python interpreter and theextensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Python libraries in machine learning used are

1.numpy

2 .pandas

3.matplot

4.sea born

Machine learning with python

- Machine Learning is making the computer learn from studying data and statistics.
- Machine Learning is a step into the directionof Machine learning is a program that analyzes data and learns to predict the outcome.
- Machine learning models used are:

    1. Random Forest
    2. Naïve Bayes
    3. Decision Trees
    4. Support Vector Machine
    5. K Nearest Neighbours

4. Project report

Project title: A Comparative Study on machine learning models using rainfall prediction.

Abstract:

Majority of Indian farmers depend on rainfall for Agriculture .Thus, in an agricultural country like India, rainfall prediction becomes very important. Rainfall causes natural disasters like flood and drought, which are encountered by people across the globe every year. Rainfall prediction over drought regions has a great importance for countries like India whose economy is largely dependent on agriculture. A sufficient data length can play an important role in a proper estimation of drought, leading to a better appraisal for drought risk reduction. Due to the dynamic nature of the atmosphere statistical techniques fail to provide good accuracy for rainfall prediction. So, we are going to use Machine Learning algorithms like Logistic Regression ,Random forest classifier, SVC,K-Nearest Neighbors, Decision trees classifier where different models are going to be trained using a training data set and tested using a testing data set. The dataset which we have collected has the rainfall data Nonlinearity of rainfall data makes Machine Learning algorithms a better technique. Comparison of different approaches and algorithms will increase an accuracy rate of predicting rainfall over drought regions. We are going to use Python to code for algorithms. Intention of this project is to say, which algorithm can be used to predict rainfall, in order to increase the countries socioeconomic status.

Random forest classifier.

- A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

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 Variables using a given set of independent variables.

Decision Trees classifier

- Decision Tree is a **Supervised learning technique** that can be used for both classification 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**

K -Nearest Neighbors

- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.

Support Vector Machine

- SVC works by mapping data points to a high-dimensional space

SAMPLE CODE

#Import the libraries and read the dataset

```
import numpy as np
 import pandas as pd
import matplotlib.pyplot as plt
#read the dataset
data = pd.read_csv('rainfalltrm.csv')
data
####splitting dependent and independent variables in X and y
X = data.drop('Trm Rain',axis=1)
y = data['Trm Rain']
 #######logistic Regression
from sklearn.linear_model import LogisticRegressionfrom
sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score,precision_score,recall_score,f1_scoreimport warnings
warnings.filterwarnings("ignore")
# Splitting the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=42)
 logreg.fit(X_train, y_train)
LogisticRegression()
O/P:
Accuracy: 0.8133333333333334
Precision: 0.825
Recall: 0.9705882352941176
F1_score: 0.8918918918918919

# Fit the model on the training data
SVC.fit(X_train,  y_train)
```

SVC()

O/P:
Accuracy: 0.7933333333333333

Precision: 0.7933333333333333

Recall: 1.0

F1 Score: 0.8847583643122676

 # Fitting the model with training data

knn.fit(X_train, y_train)

KNeighborsClassifier()

O\P:

Accuracy: 0.77

Precision: 0.8188679245283019

Recall: 0.9117647058823529

F1 Score: 0.8628230616302188

 # Fit the model with training data

clf.fit(X_train, y_train)

DecisionTreeClassifier()

O\P:

Accuracy: 0.7666666666666667

Precision: 0.8529411764705882

Recall: 0.8529411764705882

F1-score: 0.85294117647058

 # Fit the model with the training data

rf_classifier.fit(X_train, y_train)

RandomForestClassifier()

O/P:

Accuracy: 0.7933333333333333

Precision: 0.8410852713178295

Recall: 0.9117647058823529

F1 Score: 0.87500000000

Outcome:

The **best model** is logistic regression.

```
#####Making a rainfall predictive system
input_data=(2,26.840000000000032,27.700000000000045,275,4.03,82,1003,100,27.700000000000045,1)
###changing the input_data to the numpy array
input_data_as_numpy_array=np.asarray(input_data)
###reshape the array as we are predicting for one instance
input_data_reshaped=input_data_as_numpy_array.reshape(1,-1)
###standardise the input data
std_data=scaler.transform(input_data_reshaped)
print(std_data)
prediction=logreg.predict(std_data)
print(prediction)
```

Output:

```
[[-1.44417822 -0.62050338 -0.41506295  0.53885306 -0.29042928  1.14149113
  -0.28390341  0.43220388 -0.41287291  1.1771516 ]]
[1]
if prediction[0]==1:
    print("it will rain tomorrow")
else:
    print("it will not rain tomorrow")
```

Output:

it will rain tomorrow.

```
input_data=(118,25.960000000000036,25.960000000000036,274,7.77,81,1005,100,25.960000000000036,0)
###changing the input_data to the numpy array
input_data_as_numpy_array=np.asarray(input_data)
###reshape the array as we are predicting for one instance
input_data_reshaped=input_data_as_numpy_array.reshape(1,-1)
###standardise the input data
std_data=scaler.transform(input_data_reshaped)
print(std_data)
prediction=logreg.predict(std_data)
print(prediction)
```

Output:

```
[[11.95779564 -0.85825448 -0.87843363  0.52398917  1.70102495  1.07603553
  0.34992746  0.43220388 -0.88009382 -0.84950826]]
```

```
[0]
```

```
if prediction[0]==1:
    print("it will rain tomorrow")
else:
    print("it will not rain tomorrow")
```

Output:

```
it will not rain tomorrow
```