Machine Learning Lab Assignment

Vedhesh Dhinakaran
Department of Computer Science and Engineering
Amrita School of Computing, Amrita Vishwa Vidyapeetham
Bengaluru, 560035, India
bl.en.u4cse23257@bl.students.amrita.edu

***Abstract*—This report summarizes Machine Learning (ML) assignment designed to provide hands-on experience with key ML tasks. Using real-world Excel datasets like Purchase Data, IRCTC Stock Prices, and thyroid0387_UCI, applied concepts like matrix operations, similarity measures (Jaccard Coefficient, Simple Matching Coefficient, Cosine Similarity), imputation, and normalization. Python libraries such as pandas, NumPy, seaborn, matplotlib and scikit-learn were used throughout. The assignment aimed to strengthen practical understanding of ML workflows, from data handling to model development and evaluation.**

***Index Terms*—Machine Learning, Data imputation, Data normalization**

I. Introduction

The advancements in technology has paved way for Artificial Intelligence (AI) and the broad domain of Machine Learning (ML). With the help of ML models, many tasks such as regression, classification and clustering can be performed on datasets which are predominantly excel or csv files. With ML as a core subject for the 5th semester, students were assigned with assignment to have a better understanding on how to create a model and train it for using for tasks which humans often tend to do mistakes.

To understand how ML models are trained and how tasks such as regression, classification and clustering are done with the help of these models by giving excel files as the dataset, there was as assignment assigned to students to enhance their understanding on the subject of ML. The assignment consisted of 9 questions in total, which included the process of reading the data from excel sheets and to perform tasks such as classification. Firstly, we had to segregate the data from the Purchase data excel file, into 2 matrices A and C. Then to perform some mathematical operations such as finding the dimensionality of vector space, the rank of the matrix and the pseudo-inverse of the matrix. Next was to develop a classifier which would classify customers from the same dataset as Rich or Poor based on the purchase value. Then, from the IRCTC stock price dataset, we had to calculate the mean and variance, mean of price data on wednesdays, mean of price data for the month of April, Probability of making loss over stock, probability of making profit on stock if it was a Wednesday and to plot a scatter plot of the charge percentage against the day of the week. With the thyroid0387UCI dataset, task of data exploration was to be done. Where the study of attributes and its associated values was done, to find encoding scheme to deploy label and one-hot encoding, to study the sata range, the missing values, presence of outliers and finally to calculate mean and variance for numeric features. To understand the concept of similarity measure, the first 2 observation vectors from the dataset, which are binary were chosen to calculate the Jaccard coefficient (JC) and Simple Matching Coefficient (SMC) between the document vectors. For the concept of cosine similarity measure, the complete vectors from the dataset were taken to calculate the Cosine similarity (COS) between the documents by considering the second feature vector for each document. Based on the first 20 observation vectors, we had to calculate the JC, SMC and COS between pairs of vectors for 20 vectors. Finally to employ a heatmap plot to visualize the similarities. To understand the concept of data imputation, to fill the missing values in data variables, we could mean when there is no outliers, median when data is numeric and outliers are present and mode for categorical attributes. Finally, for data normalization and scaling, identification of attributes which may need normalization was done. And to the identified data, normalization techniques were to be applied.

The literature Survey section delves more into the technologies and libraries used to solve these ML assignment questions.

II. Literature Survey

As the technology advances, so does the need for computation increases. To aid better computation, to make computers solve repetitive tasks which humans cannot solve, Artificial Intelligence (AI) evolved to be the best tool to make computers mimic human intelligence and to perform complex tasks and for better efficiency, is prone to less errors. One such domain under AI is the Machine Learning (ML). In simple words, ML is to make computers learn patterns and relationships among the data which is fed to the model, the model after understanding the key factors and understands the desired output based on the inputs [1], helps in classification, regression and clustering based on the problem statement. The data which is fed into the ML model could be any text files, csv or excel files and other types of data. ML majorly revolves around tabular textual data and consists of two main sub branches which are supervised learning and un-supervised learning. In supervised learning, the model is fed with labeled data, the ground truth or the output class is labeled, this helps the model in regression and classification tasks [2]. In regression, the ML model predicts based on continuous numerical data [3] whereas in

classification, the model predicts for categorical labels [4]. In case of un-supervised learning, there is no labels on the data. The ML model, understands the patterns, relationships and clusters similar data vectors together [5]. Un-supervised learning is predominantly used for clustering where it groups data points into clusters based on the similarities without any predefined labels in the dataset [6]. Jaccard Coefficient (JC) measures the similarity between 2 sets or binary vectors by comparing the number of common 1s (intersection) and the total number of 1s (union) [7]. The formula used to compute JC has been mentioned in eq. 1.

$$\text{Jaccard}(X, Y) = \frac{f_{11}}{f_{01} + f_{10} + f_{11}} \tag{1}$$

Here f11 is the number of attributes where both A and B are 1, f01 is where A is 0 and B is 1, and f10 is where A is 1 and B is 0. JC ignores the vectors where both have 0s. Simple Matching Coefficient (SMC) measures the similarity by comparing both matches of 1s and 0s between 2 binary vectors [8]. The formula used to compute SMC is mentioned in eq. 2.

$$\text{SMC}(X, Y) = \frac{f_{00} + f_{11}}{f_{00} + f_{01} + f_{10} + f_{11}} \tag{2}$$

Where, f11 is where both the vectors have 1s, f00 is where both have 0s and f01 and f10 are the mismatches vectors with 0s and 1s. Cosine Similarity Measure is a metric used to measure how similar 2 vectors are, regardless of their magnitude [9]. It is often used for text analysis, recommendation systems, clustering and ML to determine the similarity between the feature vectors. It calculates the cosine angle between the 2 non-zero vectors in a multi dimensional space. The formula used to calculate cosine similarity is mentioned in eq. 3.

$$\text{Cosine}(\vec{A}, \vec{B}) = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\|\|\vec{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{3}$$

Where A.B is the dot product of vectors A and B, $\|A\|$ and $\|B\|$ is the euclidean norm (magnitude) of vectors A and B respectively. Heatmaps are data visualization techniques which uses color gradients to represent the magnitude of values in a matrix or 2D array. In case of this assignment, the heatmap generated displays the cosine similarity values between pairs of vectors from the dataset. Darker shades in the heatmaps represents the presence of outliers in the dataset and the varied color distribution helps to understand whether the data is homogeneous with similar vectors or diverse with many dissimilar vectors. Data imputation involves filling missing values using the mean when data is normally distributed without outliers, the median when numeric data has outliers, and the mode for categorical attributes [10]. Data normalization transforms data to a common scale using min-max normalization (scaling values between 0 and 1) [11] or z-score normalization (standardizing data based on mean and standard deviation) [12]. Figure 1. depicts the flowchart which outlines the key stages of a typical machine learning pipeline, from data collection to model deployment.

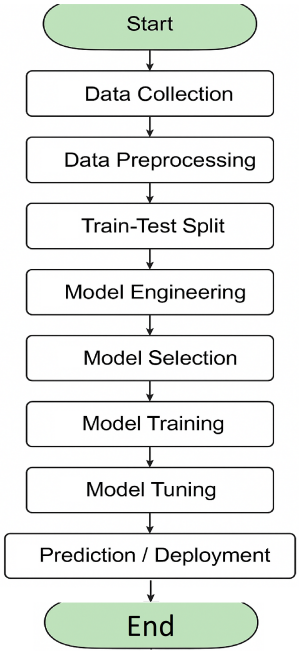

Fig. 1. Generic Machine Learning Workflow from Data Collection to Deployment

To solve the questions in the assignment, python was used as the base programming language and Google Colab as the platform where these codes were run to get outputs. Python has innumerable libraries for working with excel files and for training and testign ML models. The libraries which were used in this assignment are : pandas,a numpy, statistics, matplotlib, seaborn and sklearn. Pandas is a robust library in python which is used for data analysis and manipulation for labeled data structures. Numpy is a core library for numerical computing with a large support for multi-dimensional arrays and mathematical operations. Statistics is a built in module which provides functions for statistical computations liek mean, variance and standard deviation. Matplotlib is a plotting library used for creating static, interactive and animated visuals. Seaborn is a high level interface built on top of matplotlib for creating informative ad attractive statistical graphs. sklearn is a comprehensive ML library which offers tools for model building, evaluation and for data processing.

III. Methodology

The assignment consisted of 9 questions which was explained in the introduction section of this article. The dataset consisted of an excel sheet with 4 sheets of Purchase data, which was the dataset of customers and their purchase on products. Next sheet was IRCTC Stock Price, which had the data on the stock price on date, days and months for IRCTC along with volume and charges. Third sheet was the thyroid0387UCI datasheet, which consisted of records of patients along with features such as age and other medical ailments. Last sheet was the marketingcampaign, which consisted data on adults, kids and teens with other marketing

products.

At the start of the first question, pandas and numpy libraries were imported. Once imported, using pandas.read_excel() the Purchase data excel sheet was read. Next the model was tuned to consider specific columns for better performance. Then, the columns were divided to 2 matrices A and C with A having all the features and C having the class label or ground truth of the purchase value. Once the matrices were created, the dimensions of the matrices was found using numpy.shape() which returns the dimensions of the matrix. The number of vectors was computed using the len() function. The rank was calculated using numpy.linalg.matrix_rank() function. Finally, the pseudo-inverse of the matrix was computed using numpy.linalg.pinv() function, using this, the cost for each feature was calculated. With the help of these functions, the required results were obtained.

To solve the second question, pandas and numpy were imported, then the Purchase data excel sheet was read using the pandas.read_excel() function. Once read, the required columns were specified. Once the dataset was loaded, then we had to classify customers based on their purchase as "Rich" or "Poor" if their purchase was above 200 respectively. In order to solve this, a lambda function was used with the help of .apply() function where it applied this lambda function to the vectors in the Customer data based on the Price feature. Additionally, it added an extra column of Customer type which specifies if the customer is Rich or Poor, by using the .to_excel() function which writes this data to the excel file.

For the third question, pandas, statistics and matplotlib libraries were imported. Then IRCTC Stock Price dataset was loaded and the required columns were specified. Using the statistics.mean() and statistics.variance() functions, the mean and variance of the Price feature in the dataset was computed. Next, the vectors only for Wednesday were segregated and the same mean and variance function was applied for the Price feature and the results were obtained. Similarly, the vectors for the month of April were collected and the same mean and variance functions were used to compute the required results. In order to find the probability of loss over stock, the data in Chg% feature were changed by removing the % sign, which aided in easy computation. The fields where ti was a loss were segregated and the probability was computed for loss over stock. Similarly with the data fetched for wednesday, the profit days were segregated and the probability of profit on stock if it was a wednesday was computed. Finally using matplotlib, the stock percentage change was plotted against the week of the day and the required plot obtained has been depicted in figure 1.

The fourth question was solved by importing pandas, matplotlib, seaborn and sklearn libraries at the start. Then the thyroid0387_UCI data sheet was loaded usinf pandas.read_excel() function. .info() function was used to understand the attributes and its associated values. .isnull().sum() was used to compute the number of missing data in the dataset. For the normalization of the dataset for label and one-hot encoding, At the start, for label encoding, the missing fields were filled with the NaN, with the help of .replace('?',pandas.Na) and .fillna('Missing') functions. Then using the LabelEncoder() function, label encoding was performed. pandas.get_dummies() function was used for one-hot encoding. To find out the data range, data.describe() was used. In order to find out the outliers present in the dataset, a function was defined which used Interquartile range (IQR) technique to filter out the outliers. Finally a dictionary with the number of outliers in the row and the row indices where the outliers are present , was returned as the result. The mean, variance and the standard deviation for all the numeric columns were computed using statistics.mean(), statistics.variance() and statistics.stdev() functions respectively.

The fifth question was solved firstly by importing pandas library. Using pandas.read_excel() function, thyroid0387_UCI dataset was loaded. Once the dataset was loaded, the numerical columns were specified. Since the dataset has t for true and f for false. The t's were replaced with 1s and the f's were replaced with 0s with the help of .replace() function. Two vectors were selected from the dataset. To compute the Jaccard Coefficient, the required values were computed by selecting the appropriate values. Similarly the Jaccard Coefficient and Simple Matching Coefficient were computed and the results were obtained.

To solve the sixth question, pandas and sklearn.metrics.pairwise libraries were imported. All the sheets from the dataset were loaded, once loaded, numeric features were selected and other non-numeric features were discarded, two vectors were selected and reshaped to [1,-1]. The cosine similarity was computed using the cosine_similarity() function from sklearn.metrics.pairwise library and the results were displayed.

The seventh question focused on the plot for heatmaps. First, pandas, seaborn, sklearn and matplotlib libraries were imported. Then, the data sheets were loaded and the numeric features were considered and the non-numeric features were discarded. The first 20 vectors were selected and the cosine similarity for these vectors was computed using cosine_similarity() function from sklearn.metrics.pairwise, and the results were displayed. With the .figure() function from matplotlib, the plot was generated. Using seaborn.heatmap() function, the heatmap for all 4 data sheets was generated and diaplayed as the output which is depicted with figures. 2-4. Each cell (i,j) in the heatmap shows the cosine similarity between vector i and j. The color intensity indicates on how similar the two vectors are. If the cell is yellowish, then it has high similarity close to 1. If it is dark purple or blue then it has low similarity close to 0.

The eight question is on the data imputation using mean, mode and median imputation. First the pandas, sklearn.impute and numpy libraries were imported, then the dataset was loaded using pandas.read_excel() function, then the numeric features were selected. The "?" in the dataset were replaced with numpy.nan. Once the dataset has been pre-processed, using sklearn.impute,

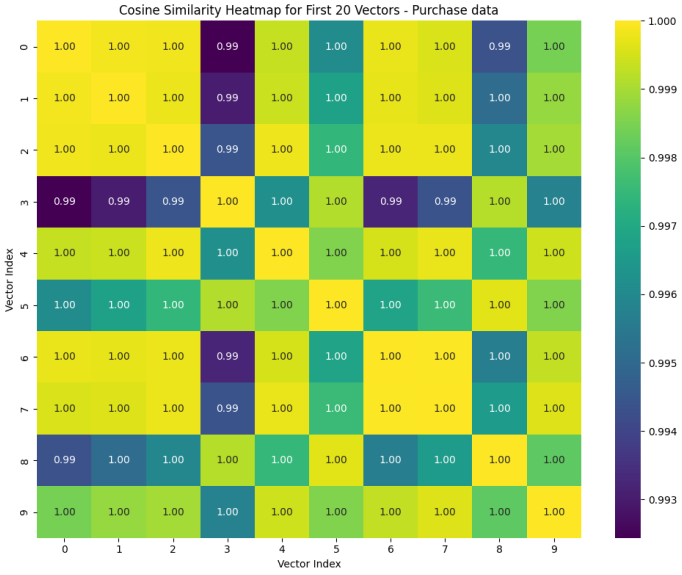

Fig. 2. Cosine Similarity Heatmap for first 20 vectors - Purchase data

Fig. 3. Cosine Similarity Heatmap for first 20 vectors - IRCTC Stock Price

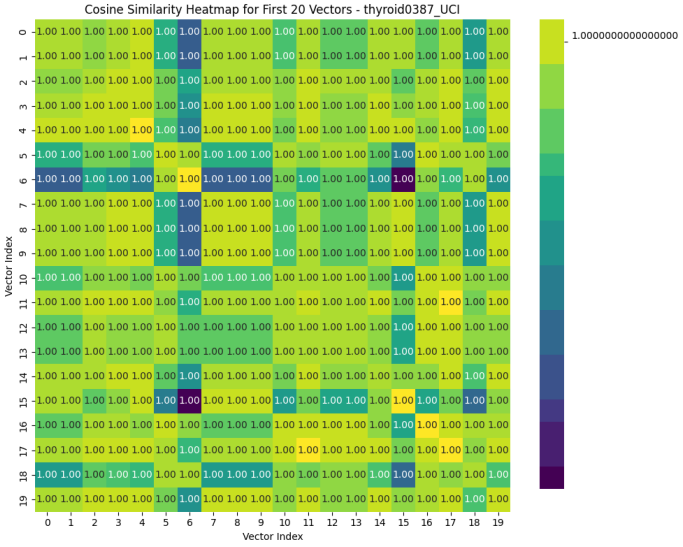

Fig. 4. Cosine Similarity Heatmap for first 20 vectors - thyroid0387_UCI

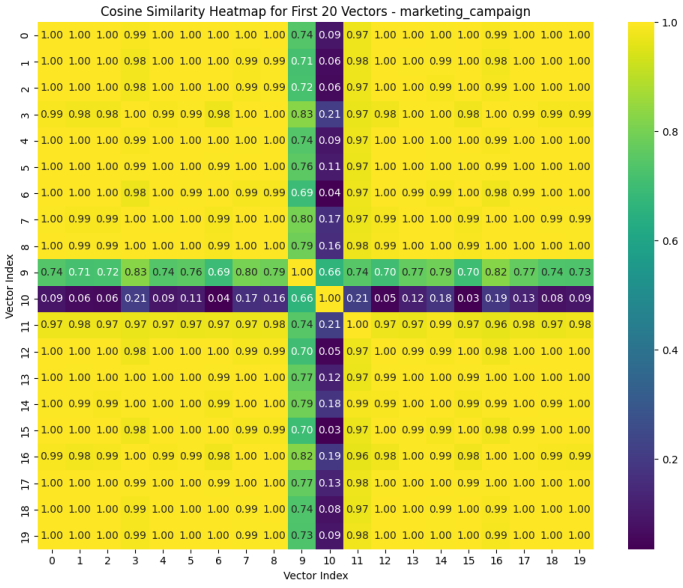

Fig. 5. Cosine Similarity Heatmap for first 20 vectors - marketing_campaign

mean_imputer.fit_transform(), median_imputer.fit_transform() and mode_imputer.fit_transform() functions were used and the mean, median and mode imputation was performed on the data and the sample dataset was displayed as the output. The ninth question revolves around data normalization. At the start pandas, sklearn.preprocessing libraries were imported, then the dataset was loaded using pandas.read_excel() function. Once dataset is loaded, the "?" were replaced with Nan using the .replace() function. Then Min-max normalization was applied to the numeric features of the dataset and the sample data was displayed. Then Z-Score normalization was applied to the numeric features of the dataset and the sample data was displayed. Finally both the normalization techniques were applied and studied the difference in the way it is done in different methods.

IV. Results Analysis and Discussions

One key aspect we explored in this assignment was the rank of an observation matrix and its importance in model building for classification. The rank tells us how many independent features or dimensions exist in our dataset. If the rank is full (i.e., equal to the number of columns), it means all features are unique and contribute meaningful information to the model. However, if the rank is low, it suggests some features may be linearly dependent or redundant, which can confuse the model and affect its ability to accurately classify data.

In Task A2, we focused on regression—predicting a continuous value, like how much a customer would spend. On the other hand, Task A3 involved classification—categorizing customers into ”Rich” or ”Poor” based on their purchase amount. While both tasks involve using input features to make predictions, regression gives us a number, and classification gives us a label. This distinction is important when choosing the right algorithm and evaluation metrics, as regression focuses on minimizing error, whereas classification emphasizes accuracy and the correct identification of classes.

based on the IRCTC stock price data, there is potential to build a predictive system for future prices and percentage changes. To do this effectively, we could use machine learning models like linear regression, decision trees, or even time-series models like LSTM (Long Short-Term Memory networks) that are good at handling sequential data. Features like the day of the week, past price trends, and trading volume can be very helpful. Also, ensuring the data is clean—by handling missing values and normalizing the values—would lead to more accurate predictions. Such a system could help users or investors make smarter decisions based on expected price movements.

REFERENCES

[1] S. Cohen, “The evolution of machine learning: Past, present, and future,” Elsevier, 2024, pp. 3–14. doi: https://doi.org/10.1016/B978-0-323-95359-7.00001-7.

[2] L.-Z. Guo, L.-H. Jia, J.-J. Shao, and Y.-F. Li, “Robust semi-supervised learning in open environments,” Frontiers of Computer Science, vol. 19, no. 8, Jan. 2025, doi: https://doi.org/10.1007/s11704-024-40646-w.

[3] H. Lee and S. Chen, ”Systematic Bias of Machine Learning Regression Models and Correction,” in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 47, no. 6, pp. 4974-4983, June 2025, doi: 10.1109/TPAMI.2025.3552368.

[4] Field Cady, ”Machine-Learning Classification,” in The Data Science Handbook , Wiley, 2025, pp.89-107, doi: 10.1002/9781394234523.ch8.

[5] S. Perumal, P. Kola Sujatha, Krishnaa S, and M. Krishnan, “Clusters in chaos: A deep unsupervised learning paradigm for network anomaly detection,” Journal of Network and Computer Applications, pp. 104083–104083, Dec. 2024, doi: https://doi.org/10.1016/j.jnca.2024.104083.

[6] P. Prasetyaningrum, P. Purwanto, and A. Rochim, “Consumer Behavior Analysis in Gamified Mobile Banking: Clustering and Classifier Evaluation,” Online) Journal of System and Management Sciences, vol. 15, no. 2, pp. 290–308, 2025, doi: https://doi.org/10.33168/JSMS.2025.0218.

[7] S. Preti, “Machine Learning for Link Prediction: Exploring Collaborative Dynamics in Research Organizations,” pp. 227–233, Jan. 2025, doi: https://doi.org/10.1007/978-3-031-96033-8_38.

[8] Y. Li, E. J. Michaud, D. D. Baek, J. Engels, X. Sun, and M. Tegmark, “The Geometry of Concepts: Sparse Autoencoder Feature Structure,” Entropy, vol. 27, no. 4, p. 344, Mar. 2025, doi: https://doi.org/10.3390/e27040344.

[9] I. D. Stanciu and N. Nistor, “Doctoral capstone theories as indicators of university rankings: Insights from a machine learning approach,” Computers in Human Behavior, vol. 164, p. 108504, Mar. 2025, doi: https://doi.org/10.1016/j.chb.2024.108504.

[10] P. H. T. Gama et al., “Imputation in well log data: A benchmark for machine learning methods,” Computers & Geosciences, vol. 196, p. 105789, Nov. 2024, doi: https://doi.org/10.1016/j.cageo.2024.105789.

[11] D. Waema, W. Mwangi and P. Muriithi, ”A Min-Max Based Data Normalization and Maximum Pooling Approach for Improved Maize Leaf Disease Detection,” 2025 IST-Africa Conference (IST-Africa), Nairobi, Kenya, 2025, pp. 1-10, doi: 10.23919/IST-Africa67297.2025.11060470.

[12] S. Omoyajowo2, “Z-Score Normalized Machine Learning Approach for Predictive Maintenance Optimization in Gas Treatment Plants,” Corrosion Management ISSN:1355-5243, vol. 35, no. 2, pp. 1–12, 2025, doi: https://doi.org/10.3390/4g4kxt27.