Report: Assignment 1 - Basic

Abhishek Iyer[2724035], Don Mani[2693434], and Azhar Shaikh[2701842]

Vrije Universiteit Amsterdam `Group 100`

1 Task 1: Explore a small data set

1.1 Exploration

The data set ODI-2022.csv contains student responses gathered during the first Data Mining lecture. There are 304 records, with 17 attributes, the types of the attributes can be found in Table 1. The attribute 'timestamp' was removed, as it did not contribute any meaningful information.

Table 1: Distribution and types of Attributes

Attribute	Type
Study Programmes	Nominal
Courses on ML, IR, Stats and DB	Categorical Variable
Gender	Nominal
Chocolate	Nominal
Birthday	Date
Number of neighbours	Integer
Stand up	Bool
Stress level	Integer
Competition question	Integer
Random number	Real
Bedtime	Time
Good day(1 and 2)	String

We observed that the data set was corrupt i.e., it had a lot of values which weren't useful for interpretation and that all the questions about the courses had the same ternary options but those options had different representation, so our first task was to convert it all into a uniform representation. For that we chose the values to be yes, no and unknown. We dug deep into the attributes we found interesting and challenging and they are listed below.

Programme Enrolment: We observed that a lot of the values in the 'Programme Enrolled in' column were values that referred to the same course. For example, we had a few values like MSc AI, Ai, artificial intelligence, etc. that all referred to AI and the same for CS and other fields. So, here we basically made a dictionary with all the unique ways in which the courses were referred to and searched those values and replaced them using python. We also dropped the rows

where we encountered garbage values and we made a value called Other which encompasses all the courses which had less than 3 students enrolled in it. From the barplot in Figure 1, we can say, a diverse group of students are taking the data mining course. Among them, the maximum number of students are from Artificial Intelligence background (114 students), followed by Computer Science (41) students. This is explained by the fact that Data Mining is a constrained elective for AI and CS students.

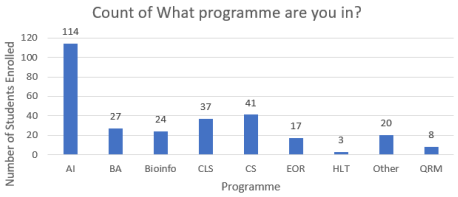


Fig. 1: Distribution of students from different study programmes

Birthday and the birthday paradox: The 'date of birth' feature cleaning was challenging as some of the entries were not in the dd-mm-yyyy format. So we used a mix of excel as well as some of pandas' cleaning methods to get the date in dd-mm-yyyy format. For further processing, wherever we found an incomplete birth date or garbage values, we converted those entries into null value and in the end, we simply dropped all the rows that had null values and in the end we were left with 214 entries. We also pondered upon the birthday paradox [8]. The probability of two people having the same birthday in a group of k people is

$$p(k) = 1 - 365!/365^k(365 - k!)$$

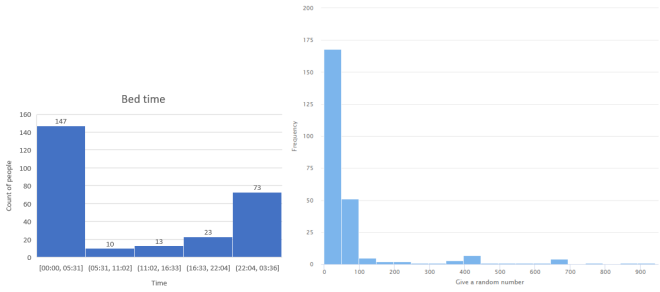
In our case we have a group of 214 people, that probability amounts to $1-2.84168037427282e-33$. We found out that 3 people share the same birthday, so using Poisson approximation [8],

$$p(k, m) = 1 - exp(-\binom{k}{m}/365^{m-1})$$

We see that probability of 3 (m=3) people sharing the same birthday in a group of 214 people (k=214) = 99.999437 percent which is approximately the same as the probability given by the birthday paradox [7].

Sleeping Time: The 'bedtime' feature cleaning was challenging and we had to resort to some manual formatting and cleaning for some parts of it. The rows with invalid times were dropped and times in AM or PM format converted to 24 hour format. Separators other than ':' were converted into ':'. Figure 2a shows the bed time distribution of students.

Random Number: To clean this attribute, we first converted all floating points to their nearest integers and then took absolute values i.e., we converted



(a) (b)

Fig. 2: (a)Bed times of students, (b)Distribution of Random numbers

all the negative integers into positive values. We also categorize any number having 4 digits or more as INF(infinity). Figure 2b shows distribution of random numbers made by the students. Most students (168) chose a number between 0 and 50.

1.2 Basic Classification

Dataset: We use the Statlog Heart Dataset [6] downloaded from the UCI data repository. The task defined for the dataset is predicting presence of heart disease based on statlog data of patients. This task can be framed as a classification task as the target is a categorical variable i.e (presence of disease = 1, disease not present = 2). The dataset has 270 samples with 13 attributes such as age, sex, blood pressure, cholesterol etc.

Motivation: The dataset is interesting as it can help in early detection and diagnosis of heart disease. Furthermore, from the perspective of a data scientist, the dataset is interesting as it has a mix of nominal as well as continuous attributes.

Experiment Design: We divide the dataset into a 80-20 train test split. We do not use a separate validation set due to the small size of the dataset but instead do 5-fold cross-validation. We select the model with the best accuracy out of the 5-folds to evaluate on the test set. When evaluating the best model on the test set, we record precision, recall, F1 score and accuracy. We experiment with two classification models: Logistic Regression and Decision Tree in our analysis.

Decision Tree Classifier: Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

Hyper-parameters: max_depth: tree grown until leaves are pure or until all leaves contain less than two samples. min_sample_leaf: 1, min_samples_split: 2.

Logistic Regression: Logistic regression is a linear model for classification. The probabilities describing the possible outcomes of a sample are modeled using a logistic/sigmoid function.

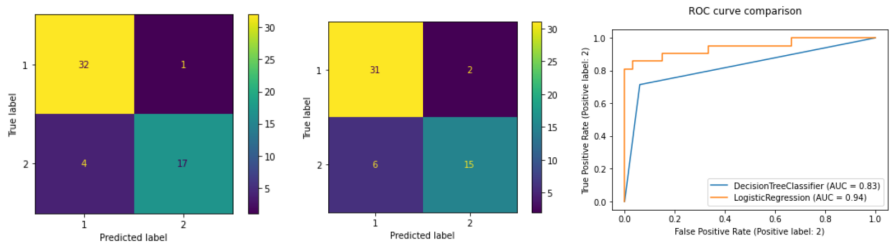
Hyper-parameters: solver: liblinear, penalty:l2, max iteration: 100

Rationale Behind Models Chosen: We decided on Decision Trees and Logistic regression as they are simple yet powerful classification algorithms. Furthermore we also wanted to study the differences between a non-parametric(Decision Tree) and parametric(Logistic Regression) classification models.

Comparison From our experiments we find that Logistic regression performs better than decision tree. Logistic regression has bigger F1 score, precision, recall and accuracy values compared to decision tree as seen in table 3. Furthermore, area under the ROC curve is also greater for logistic regression model than the decision tree model as seen in Figure 3c. Although the logistic regression model outperforms decision tree for this dataset it still has certain benefits such as no need of standardising and normalising the data and interpretability.

Table 2: Model Evaluation

Method	Precision	Recall	F1	Accuracy
Decision Tree	0.86	0.83	0.84	0.85
Logistic Regression	0.92	0.89	0.90	0.91



(a) (b) (c)

Fig. 3: (a)Confusion matrix for logistic regressor, (b)Confusion matrix for decision tree (c)Comparison of ROC curves of the models

2 Task 2: Kaggle Titanic Competition

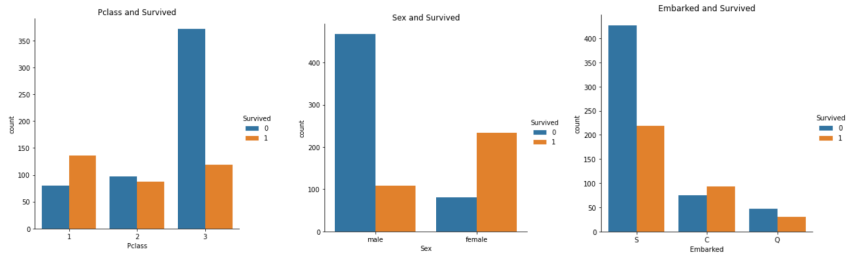
The aim of this competition is to use machine learning to predict who among the Titanic passengers were more likely to survive or not. Kaggle provides a labelled train data set and an unlabelled test set. Our goal is to predict which passenger from the test data set survived the sinking of the Titanic.

2.1 Preparation

The Titanic training set has 891 rows and 11 columns. Survived column is the target variable and the rest of the columns make up the original features. 38.38% of passengers in the training set survived the sinking of the Titanic. Table 3 describes the attributes and their types and provides a brief explanation of the attribute. About 68% of the the total data is in the training set. All the 891 names are unique. There are 314 females which is 35.24% of the training data and 577 males (64.75%) of training data. Another interesting finding is that Cabin values have several duplicates across samples, that is, several passengers shared a cabin. Embarked takes three possible values (C for Cherbourg, Q for Queenstown, S for Southampton). Southamption port was used by most passengers (644). Ticket feature has high ratio (22%) of duplicate values (unique=681).

Table 3: Attributes of the Titanic Training Set

Attribute	Type	Description
PassengerID	Integer	Key
Survived	Integer	Target Variable. 0 = No, 1 = Yes
Pclass	Integer	Passenger class
Name	String	Name of passenger
Sex	String	Sex of passenger
Age	Float	Age of passenger
SibSp	Integer	Number of siblings/spouses on board.
Parch	Integer	Number of parents/children aboard the titanic.
Ticket	String	Ticket number
Fare	Float	Fare paid by passenger
Cabin	String	Cabin of passenger
Embarked	String	Port of embarkation



(a) (b) (c)

Fig. 4: (a)Pclass is correlated with Survived, (b)Sex is correlated with Survived (c)Embarked is correlated with Survived

Analysing Correlations: Pclass, Sex and Embarked features seem to be correlated to the passenger's survival, as shown in Figure 4. However, the Embarked-Survived correlation is a spurious one as Embarked is correlated to Pclass, and probably Pclass is the underlying driving force behind the correlation. Furthermore, feature correlations were also analysed by pivoting features against each other. This was done after imputing the missing values and was done only for features which were categorical (Sex and Embarked), ordinal (Pclass) and discrete (Sibsp and Parch). There is significant correlation (¿0.5) among Pclass=1 and Survived. Similarly, passengers of female sex had very high survival rate of 74%. Hence, it is important to include Sex and Pclass in our model. In contrast, SibSp and Parch features had 0 correlation with Survived for few values. Hence, it was decided it is better to derive a feature from these two features.
Data Cleaning: The columns Age, Cabin and Embarked had missing values in the training data set. Embarked column contains 2 missing values in the training set and none in the test set. The missing data corresponds to Martha Evelyn and her maid Amelie Icard. From domain knowledge, we were able to gather that both of them embarked from Southamption. [1] Age contained 177 missing values. The mean of the Age feature is greater than the median, hence the distribution is positively skewed. We use median ages of the Pclass groups. Median age of Pclass group is better as it has higher correlation with Age than any other feature. Furthermore, we used Sex as a secondary groupby feature. Finally, about 77% of the values in Cabin feature were missing. We cannot ignore the feature completely as many cabins had high survival rates. The missing values were imputed with 'U' Cabin.
Feature Engineering: As part of feature engineering, we created some new attributes, which will help in increasing accuracy of the model. [4] Firstly, Title is created by extracting the prefix before Name feature. Family Size feature is created by adding SibSp, Parch and 1 (for the current passenger). Those columns are added in order to find the total size of families. We also created an 'IsAlone' feature to find out if the passenger is alone or with family. This is done by checking whether the passenger's Family Size is equal to 1 or not. Through domain knowledge and EDA, we understand that the first letter of the Cabin feature contains information about the deck. We extract only that information and reassign it into the Cabin feature. Another feature we created was the 'IsMarried' feature which looks at the passengers with Title as 'Mrs'. Finally, we also created a 'IsWomanOrBoy' feature which is set to true if either the passenger is female or if he has the title 'Master'. Our rationale for this, is that woman and boys are more likely to have survived the sinking of the Titanic.
Data Binning: To reduce the effect of minor observation errors and to deal with continous data, we utilise panda's qcut function to create binned features 'AgeBand' and 'FareBand' from Age and Fare features respectively. [3]

2.2 Classification and Evaluation

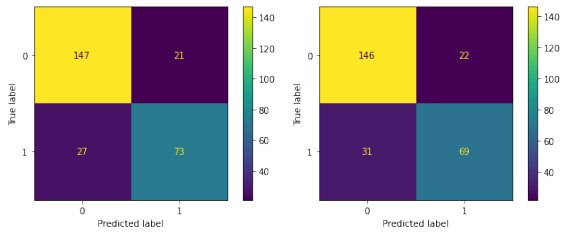
Creating a setup: We use sklearn's train_test_split to split Kaggle's training set further into training set (70%) and validation data (30%). For evaluation,

accuracy metric was used primarily.

Classification Algorithms: We use two classification algorithms: Logistic Regression and Random Forest Classifier. Both classifiers were trained and tested with the same set of data. Table 4 shows the precision, recall, F1 and accuracy obtained on testing against the validation data using both classifiers. Figure 5 displays the summary of prediction results of both classifiers on the validation set. Logistic Regression slightly outperforms Random Forest Classifier on the validation set. Random Forest Classifier is a classifier 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. [2] We achieved an accuracy of 80.22% when using Random Forest Classifier on the validation data. Logistic Regression classifier is a predictive analysis which estimates the parameters of logistic model. We achieve an accuracy of 82.08% when using the Logistic Regression classifier on the validation data.

Table 4: Model Evaluation on Titanic Validation Set

Classification Algorithm	Precision	Recall	F1	Accuracy
Logistic Regression	0.81	0.80	0.81	0.82
Random Forest Classifier	0.79	0.78	0.78	0.80



(a) (b)

Fig. 5: (a)Confusion matrix for Logistic Regression model, (b)Confusion matrix for Random Forest Classifier model

Kaggle Submission: For Kaggle competition, we decided to use Random Forest Classifier, even though Logistic Regression was better on the validation set as it provided the better accuracy than Logistic Regression model against the Kaggle test set. Logistic Regression model over-fitted on the the training data and performed worse on the test set. After submission, we received best score of 0.81339, which puts us in the top 1.6% on the leaderboard. This is a very good score as the top models in the competition are in the 82% accuracy range. This is in line with our expectations as the the model performed quite

well on the validation set also and was feature engineered quite extensively. We were also lucky that the Random Forest Classifier got higher accuracy against the test set than the validation set.

3 Task 3: Research and Theory

3.1 Research - State of the art solutions

Description of the competition : The data mining competition that we have chosen to look at, is the KDD Cup 2014. In this competition, the participants were asked to identify projects that are, from a business point of view, exceptionally exciting, for DonorsChoose.org (it is an online charity dedicated to helping in need students via school donation and when a project's funding goals are met, they ship out materials required to complete the project). Total of six **datasets** were used for this task. Namely, donations.csv - contained information about the donations to each project (only provided for projects in training set), essays.csv - contained project text posted by the teachers, projects.csv - contained information about each project, resources.csv - contained information about the resources requested for each project, outcomes.csv - contained information about the outcomes of projects in the training set, sampleSubmission.csv - contained the project ids of the test set and it had the submission format for the competition. Submissions were **evaluated** on area under the ROC curve between the predicted probability that a project is exciting and the observed outcomes. This competition describes the following criteria as necessary for a project to be considered exciting: 1) was fully funded, 2) had at least one teacher-acquired donor, 3) has a higher than average percentage of donors leaving an original message, 4) has at least one "green" donation, 5) has one or more of - (a) donations from three or more non teacher-acquired donors, (b) one non teacher-acquired donor gave more than 100 dollars, (c) the project received a donation from a "thoughtful donor".
Winner and the winning approach : The winners of this competition were Yoon Kim and Peng Liu. They used two ensembled models (features for both were almost similar). The **first model** they used was an ensemble of 4 Gradient Boost Models and 1 Extra Trees model. In this approach, they considered historical variables (like how many exciting projects did the school have in the past, etc.) as an important feature and they used (credibility-adjusted) historical rates for some categorical variables (e.g. historical exciting rate for teacher, pulled towards population mean, as opposed to actual counts), for text they created feature using logistic regression with tf-idf on title/essay using a leave-10 percent out scheme to make sure that the response variables were not used twice and the predictions from this model were used as features in a Gradient Boost Model. The **second model** they used was an ensemble of Gradient Boost Model, ExtraTrees, Random Forests, and Elastic Net. The final model was simply an average of the two models with discounting applied after averaging.
What makes the approach stand out? : The thing that makes this approach stand out is their usage of out-of-time validation sets and also the fact that they

considered historical variables and used historical features and data to having a better prediction accuracy.

3.2 Research - MSE vs MAE

Below are the formulae for the Mean Absolute and Mean squared error loss

$$MAE = \frac{1}{n}\sum_{k=1}^{n} Y_k - \hat{Y_k} = \frac{1}{n}\overrightarrow{1}^T r \tag{1}$$

$$MSE = \frac{1}{n}\sum_{k=1}^{n} (Y_k - \hat{Y_k})^2 = \frac{1}{n} r^T r \tag{2}$$

$\hat{Y_k}$ is the model prediction for sample k, Y_k is the target value for sample k. r is a nx1 vector of residuals for the n samples in the dataset and $\overrightarrow{1}$ is a nx1 vector of ones.

Discuss MSE vs MAE: MAE gives equal weight to all errors while large errors are emphasized greatly and small errors not emphasized at all when using MSE. It is this behaviour of MSE that makes it sensitive to outliers therefore MAE error is usually employed as an alternative to the MSE when a dataset has a large number of outlier cases.

Case when MSE and MAE are same: Setting the vector form of the MSE and MAE equal and rearranging gives the below equation:

$$(r - \overrightarrow{1})r = 0 \tag{3}$$

Therefore MSE equals MAE when all residuals are 0s or 1s.

Dataset and Experiment: We perform regression on the Dubai Housing prices prediction dataset [5]. A linear regression and SVM regression model are evaluated on the dataset with MSE and MAE as the performance metrics. This dataset is chosen as it has outliers and therefore could help test the hypothesis about the effect of outliers on the MSE and MAE metrics. The data has 1905 samples and 36 attributes like neighborhood, latitude, longitude, no of bedrooms, no of bath rooms, size in sqft, price per sqft etc, with price being the target variable.

Analysis: We observe that the Linear Regression model has (MAE=4.55, MSE=927.52) while the SVM model has (MAE=2.15, MSE=130.88) the larger value for MSE can be attributed to the outliers present in the data. When we perform a log transformation on the price to remove the effect of outliers and then do evaluation we observe that Linear Regression has (MAE=0.14, MSE=0.035) while the SVM model has (MAE=0.101, MSE=0.016) i.e MSE now has a lower value than MAE confirming our hypothesis.

3.3 Theory - Analyze a less obvious dataset

Text Classfication: Since the dataset is full of text, and, the target variable label is a categorical variable, this problem is an example of a text classification

problem and to tackle such problems, we will be first perform some meaningful EDA on the dataset, then select a few NLP pre processing techniques and then we will transform the text variables into numerical ones using TF-IDF vectorization (which takes into account the importance of each term to document). Then we will use SVM model to perform the binary classification on this text dataset and analyze the results.

Description and EDA of the dataset: Here, we are working with a text message dataset SmsCollection.csv, where messages are classified as Ham or Spam. On taking a closer look at the dataset, we find that out of the 5574 messages, 4487 are labelled as Ham and 746 are labelled as Spam, this means that the dataset is highly imbalanced. We also create two additional dataframes, one which contains only the messages labelled as Ham and the other one containing all the Spam messages, then we create two more dataframes with one of it containing the text length of all Ham labelled messages and the other one containing the text lenght of all Spam labelled messages, on visualizing this, we come to the conclusion that most of the Ham text messages have string lengths between 10-30.

NLP Preprocessing: For this dataset, the nlp pre processing techniques that we found useful and meaningful are: 1) Converting all the characters into lower case, 2) Tokenization, i.e., we store each word into a different token, 3) Removal of special characters (as in this dataset we see that a lot of the messages contain special characters and they add no semantic value), 4) Stop word removal, we do this to remove the most common stop words in the English language as they provide little to no meaning, 5) We applied stemming using nltk's PorterStemmer(). We also converted the values in the label column from Ham and Spam to 1 and 0 respectively, then we created a dataframe where we stored all preprocessed text messages.

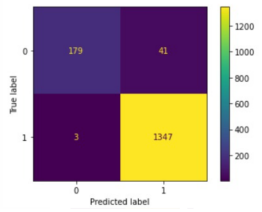


Fig. 6: Result of SVM Classifier

Result of SVM: From Figure 6 and we can see that using SVM gives us a good accuracy of 0.97 and there's very little mislabelling, however, the overall accuracy can be improved by using Word Embeddings and using a specialized NLP model like RESCAL for classification.

References

1. Encyclopedia Titanica Martha Evelyn Stone, https://www.encyclopedia-titanica.org/titanic-survivor/martha-evelyn-stone.html. 24th August 2018
2. Random Forest Algorithm: A Complete Guide, https://builtin.com/data-science/random-forest-algorithm. 22nd July 2021
3. Data Preprocessing with Python Pandas — Binning, https://towardsdatascience.com/data-preprocessing-with-python-pandas-part-5-binning-c5bd5fd1b950 23rd December 2020
4. What Is Feature Engineering for Machine Learning?, https://medium.com/mindorks/what-is-feature-engineering-for-machine-learning-d8ba3158d97a 14th February 2018
5. Dubai Properties - Apartments, https://www.kaggle.com/datasets/dataregress/dubai-properties-dataset
6. Statlog (Heart) Data Set, https://archive.ics.uci.edu/ml/datasets/Heart+Disease
7. DasGupta, A. (2005) The Matching, Birthday and the Strong Birthday Problem: A Contemporary Review. Journal of Statistical Planning and Inference, 130, 377-389.
8. Understanding the birthday paradox https://betterexplained.com/articles/understanding-the-birthday-paradox/