Data Mining Final Project Report

Team 4:
Amey Borkar
Shivam Joshi
Lolyna de la Fuente
Can Demirel
Joseph Ng

Executive Summary

The goal of this dataset was a classification task, as most real-world examples are, which involves being able to classify how well a marketing tactic was for a banking company by determining if a customer would give money.

We used several different classification models to determine the best algorithm to use for this dataset. Logistic regression performed the best with an accuracy of 91%, signifying the dataset has a linear classification boundary as logistic regression is a linear classifier.

For the future directions of this dataset, being able to use sophisticated visualization should be done, as visualizations for this dataset were difficult due to the many types of data types, and the high dimensionality of the dataset made it challenging to take all the attributes and visualize them in one summarizing graph.

Another future direction would be to perform more complex grid search, feature selection, and feature engineering to help improve each model's accuracy and to fine-tune the model to improve its accuracy. Also, being able to run some of the models on a GPU or a high-performance computer would be beneficial, as the SVC model is a model that takes a considerable amount of time on a large dataset, which forced us to sample the dataset for this model. Overall, the high accuracy and multi-algorithm testing created a thorough check that the best model was chosen and is a suitable algorithm for the context of this dataset.

Main Report

Problem Statement

The project aimed to predict the success of direct marketing campaigns conducted via phone calls by a Portuguese banking institution. Specifically, the goal was to develop a predictive model to determine whether clients would subscribe to a term deposit ('y'). This task involved analyzing a dataset enriched with social and economic features alongside traditional bank client demographics and campaign interaction data.

Dataset

We primarily utilized the "bank-additional-full.csv" dataset for our project. This comprehensive dataset comprises 41,188 examples, each with 20 input variables. The data spans from May 2008 to November 2010, ordered chronologically.

This dataset was chosen due to its extensive coverage, providing a rich source of information for our analysis of direct marketing campaigns conducted by a Portuguese banking institution.

Input variables:
1: age (numeric)
2: job: type of job (categorical: "admin,", "blue-collar", "entrepreneur", "housemaid", "management", "retired", "self-employed", "services", "student", "technician", "unemployed", "unknown")
3: marital status (categorical: "divorced,married,single,unknown"; note: "divorced" means divorced or widowed)
4: education (categorical: "basic.4y", "basic.6y", "basic.9y", "highe", "university.degree", "unknown")
5: default: has credit in default? (categorical: "no","yes","unknown")

6: housing: has housing loan? (categorical: "no","yes","unknown")
7: Loan: has personal loan? (categorical: "no","yes","unknown")

Related to the last contact of the current campaign:
8: contact: contact communication type (categorical: "cellular","telephone")
9: last contact month of year (categorical: "jan", "feb", "mar", ..., "nov", "dec")
10: day_of_week: last contact day of the week (categorical: "mon, tue, wed, thu, fri")
11: duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration = 0, then y = no"). Yet, the duration is not known before a call is performed. Also, after the end of the call, y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.

Other attributes:
12: campaign: number of contacts performed during this campaign and for this client (numeric, including last contact)
13: days: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)
14: previous: number of contacts performed before this campaign and for this client (numeric)
15: poutcome: outcome of the previous marketing campaign (categorical: "failure,nonexistent,success")

Social and economic context attributes:
16: emp.var.rate: employment variation rate, quarterly indicator (numeric)
17: cons.price.idx: consumer price index, monthly indicator (numeric)
18: cons.conf.idx: consumer confidence index, monthly indicator (numeric)
19: euribor3m: euribor 3 month rate, daily indicator (numeric)

20: nr.employed: number of employees; quarterly indicator (numeric)

Output variable (desired target):
21: y - has the client subscribed a term deposit? (binary: "yes","no")

Preprocessing the Dataset

Preprocessing the dataset involved several systematic steps to prepare it for analysis and modeling. Initially, the necessary libraries were imported, and the dataset was read using Pandas, ensuring accessibility to data manipulation tools. To enhance ease of access, columns containing periods (.) were renamed with underscores (_) for uniformity. Numerical and categorical features were identified through predefined data types, laying the groundwork for tailored preprocessing steps. Mitigating multicollinearity risks, highly correlated numerical features were prudently removed from consideration.

Separate pipelines were constructed for numerical and categorical features, enabling focused preprocessing strategies. Missing values within numerical features were addressed using SimpleImputer, followed by scaling with StandardScaler to standardize the data distribution. Categorical features were encoded into numerical representations using OneHotEncoder, facilitating compatibility with machine learning algorithms.

Subsequently, the dataset was partitioned into features (X) and the target variable (y), with the target variable 'y' encoded into binary values (0 and 1) for classification tasks. Additionally, categorical features underwent transformation into numerical representations using LabelEncoder, ensuring uniformity in data format.

Data Organization

Regarding data organization, meticulous categorization and structuring were employed to streamline the preprocessing workflow. Numerical features, discerned from predefined numeric data types, underwent further refinement through the identification and subsequent removal of highly correlated features. Conversely, categorical features, recognized through predefined object, category, and bool data types, were systematically processed alongside numerical features.

The preprocessing steps were methodically organized into distinct pipelines for numerical and categorical features, encompassing handling missing values, scaling, and one-hot encoding. This systematic approach ensured clarity and coherence throughout the preprocessing journey. Following the completion of preprocessing, the dataset was partitioned into features and the target variable, paving the way for model training and evaluation through a well-structured train-test split.

Results

Classifier	**Accuracy**	**TP**	**FP**	**FN**	**TN**
KMeans	89%	7181	122	745	190
KNearestNeighbors	90%	7179	124	739	196
RandomForest	89%	7056	247	670	265
DecisionTree	83%	6549	754	613	322
LogisticRegression	91%	7105	198	532	403
NaiveBayes	81%	6070	1233	313	622
SupportVectorC	88%	709	11	89	15
GradientB	88%	707	13	84	20

As we can see from the results, the best classifier for predicting the target class 1, or ‘yes’, is the Naive Bayes classifier, with 622 values as true negative. K-Means was the best classifier for recalling class 0 or ‘no’ with 7181 values as true positive. Amongst them all, comparing both their classification matrices and classification reports, we conclude that the Logistic Regression Classifier is the best for our dataset, as it not only has the best accuracy at 91%, but for both classes (0 and 1), its f1-score proved to be superior.

Classifiers Performance

All classifiers were compared using their confusion matrix and their classification report results, which contain metrics such as precision, recall, f1-score, and accuracy, among others. The metrics we focused on were the confusion matrix, accuracy, and f1-score, paying special attention to class 1, which is the one we are interested in predicting.

Conclusion

We evaluated the performance of eight different classifiers on our dataset. Through our analysis, we gained insights into the strengths and weaknesses of each algorithm.

Logistic Regression achieved the highest accuracy of 91%, indicating that it correctly classified the most instances overall.

KNearest Neighbors also performed well, with an accuracy of 90%.

Decision tree and random forest both achieved an accuracy of 89%, indicating strong predictive performance.

Preprocessing techniques played an important role in enhancing classifier performance. One-hot encoding and handling of missing values improved the models' ability to capture relevant patterns in the data.

We used a range of evaluation metrics, including accuracy, precision, recall, and ROC curves, to assess classifier performance. These metrics provided valuable insights into the classifiers' effectiveness.

Based on our findings, we recommend the following areas for future research:

- Explore the effectiveness of methods for improving classification performance.
- Investigate advanced feature engineering techniques to extract more informative features from the dataset.
- Develop strategies to address class imbalance issues, such as oversampling.

Our analysis underscores the importance of selecting appropriate classifiers and preprocessing techniques for effective classification. By leveraging a diverse set of machine learning algorithms and rigorous evaluation methods, we can develop robust models that provide valuable insights for decision-making in various domains.