Assignment 1 – DATA 622

Exploratory Data Analysis Assignment

Feature Review

The goal of this assignment is to use Exploratory Data Analysis (EDA) to understand the Bank Marketing dataset and outline how to properly clean, prepare, and model the dataset to predict help the bank find potential long term deposit customers. The raw dataset contains 45,211 rows and 17 different columns that are both mixed numeric and categorical features. The target, "y", is binary (yes/no) with an imbalanced count. When looking at this imbalanced count, there are about 12% yes values and 88% no values.

Looking at the predictive features, there are several that have null values. They are as follows:

- The "job" column, which is categorical, has a total of 288 null values. This is ~0.64% of the rows.
- The "education" column, which is categorical, has a total of 1,857 null values. This is 4.11% of the rows.
- The "poutcome" column, which is categorical, has a total of 36,959 null values. This is 81.75% of the rows.
- The "contact" column, which is categorical, has a total of 13,020 null values. This is 28.8% of the rows.

Further information on each of these features, the characteristics of the column and more can be seen in the table below.

TABLE 1 – Raw Feature Information

Feature	Type	Notes
Age	Continuous	Slightly right-skewed; most are between 30–50 with a longer tail after 60; Range: 18–95; Average: 40.94;
Job	Categorical	Largest counts are blue-collar, management, and technician; long tail of other categories.
Marital	Categorical	Married is most common, single second, divorced third.
Education	Categorical	Secondary is most common, followed by tertiary, then primary
Default	Binary	Imbalanced; 815 yes (1.8%) vs 44,396 no (98.2%)

Balance	Continuous	Extremely right-skewed with upper outliers and some negative (overdrawn) values; Range: −8,819 to 102,127; Average: 1,362.27; may need a transform.
Housing	Binary	5,130 yes (55.58%) with a housing loan vs 20,081 no (44.42%).
Loan	Binary	Imbalanced, 7,244 yes (16.02%) vs 37,967 no (83.98%)
Contact	Categorical	Mostly cellular; Many nulls.
Duration	Continuous	Strong right skew with a long upper tail; Range: 0–4,918; Average: 258.16; not known before the call, so not a good predictor.
Campaign	Continuous	Strong right skew with many upper outliers; most around 1–3 contacts; Range: 1–63; Average: 2.76.
pdays	Continuous	Heavily around zero/low digits with a long upper tail; −1 means not previously contacted (partially categorical).
Previous	Continuous	Most have 0 (clustered at zero) with extreme right skew; Range: 0–275; Average: 0.58.
Poutcome	Categorical	Outcome of previous campaign; needs processing/regrouping (e.g., success/failure/other).

Processing

In processing and cleaning the data, the first task at hand is the missing values for the features outlined previously: job, education, contact, and poutcome. A new column named “missing_count” was created in order to sum up the null values in each row. The rows that contain 3+ nulls were dropped from the data. This resulted in a dropping of a total of 1.75% of the data. The remaining nulls were associated with rows that had 2 or less null values. These nulls were imputed. The methods of imputation were simple, the null categories placed into their own “unknown” category, or in the case of the poutcome column, the nulls were placed into a “no_previous_contact” category.

Other columns were dropped after some feature engineering or after the determination that they were not useful predictors, based on what information the feature was conveying. For instance, the duration column was dropped because it measured the length of campaign call, which is not a good predictor as it isn’t known before the call. Another example would be the dropping of the pdays column after using that column, along with the previous column to make a binary “previously_contacted” flag feature.

Additionally, other categorical features that had many categories were bucketed further to limit the number of values. For instance, for the job column the top four categories by count were kept, while the longer tail of other job categories were placed into an "other" category.

Lastly, while not all fully executed via code in the accompanying notebook, there should be scaling and normalization of the continuous variables before modeling takes place. The categorical values need to be encoded to numeric categories as well. The outcome values are also imbalanced, so when creating the model's training data a sampling technique, or otherwise, should be used in order to generate additional "yes" values in order to prevent the model from always predicting "no".

Model Selection

For model selection, based on what we covered in class, my primary two selections are Logistic Regression and Random Forest. The other models don't seem to be ideal options for this data set. KNN is not a good choice because of the high number of features and the mixed variable types it handles poorly, and the imbalanced target would further hurts KNN's performance. Discriminant Analysis (LDA/QDA) is also not ideal. While QDA might accommodate differing covariance in the data set, both LDA & QDA approaches assume normally distributed continuous variables. This is not the case in this data set. Additionally, they do not handle categorical data well. The Naïve Bayes (NB) modeling technique would also not be good here. This model assumes all of the features are independent, which is not the case. Jobs will be related to education, and the age of a call recipient has a direct relationship with the balance of the account. Just by process of elimination the two selected models are the most ideal.

Overall, while my ideal pick would be random forest, the logistic regression model would be a decent choice to have an easily explainable baseline model. While it assumes linear relationships and would need scaling/ transforming of the data, it is better than the other models outlined.

When it comes to Random Forest modeling, this model handles a mix of categorical and continuous data well. It also allows for nonlinear relationships and skew within the continuous features, this implies less processing needed. The one drawback would be a less amount of explainability.

Lastly, to answer the question: "Would your choice of algorithm change if there were fewer than 1,000 data records, and why?" If the dataset were under 1,000 rows, I think logistic regression would be a better option. The smaller size of the dataset would mean that the ideal would be a generalized type of insight that is easily explainable. Using Logistic Regression would allow for an easily explainable, simple model that provides general insights for the data. Random Forest would then be overfit to the data with just 1,000 rows.