Assignment 2 - Classification

Instructions:

For the Second Assignment, we are going to be implementing the classification algorithms: Logistic Regression, Multinomial Naive Bayes, and SVM.

Each task is associated with a different set of classification methods. Each task will use its own dataset that is already written in the cell. Each Task is presenting a challenge.

Task 1: Logistic Regression on Bank Data Bank client data:

- age (numeric)
- job: type of job (categorical: "admin.", "unknown", "unemployed", "management", "housemaid", "entrepreneur", "student", "blue-collar", "self-employed", "retired", "technician", "services")
- marital: marital status (categorical: "married", "divorced", "single"; note: "divorced" means divorced or widowed)
- education (categorical: "unknown", "secondary", "primary", "tertiary")
- default: has credit in default? (binary: "yes", "no")
- balance: average yearly balance, in euros (numeric)
- housing: has a housing loan? (binary: "yes", "no")
- loan: has a personal loan? (binary: "yes", "no")

Related to the last contact of the current campaign:

- contact: contact communication type (categorical: "unknown", "telephone", "cellular")
- day: last contact day of the month (numeric)
- month: last contact month of the year (categorical: "jan", "feb", "mar", ..., "nov", "dec")
- duration: last contact duration, in seconds (numeric)

Other attributes:

- campaign: number of contacts performed during this campaign and for this client (numeric, includes the last contact)
- pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric, -1 means the client was not previously contacted)
- previous: number of contacts performed before this campaign and for this client (numeric)
- poutcome: outcome of the previous marketing campaign (categorical: "unknown", "other", "failure", "success")

Output variable (desired target):

- y: has the client subscribed to a term deposit? (binary: "yes", "no")

Task 1A: 5 points

- Is there any need to convert columns based on their Dtype? Check details about the data.
- Check if there are any missing values. Handle the missing values if any.
- Comment your actions if any.

Task 1B: 5 points

- Split the data into training/testing with an 80-20 ratio. Explain why we need to do this.
- Map the target variable.
- Use stratify to ensure an equal percentage of class samples in both subsamples.

Task 1C: 10 points

- Define a class 'convert_cat' that will:
  - Map the binary categorical values.
  - Determine whether it is feasible to create dummy variables from the 'month' categorical variable. If not, suggest using frequency encoding.
  - Create dummy variables for the rest of the categorical variables.
- Call the function for the training and testing data.
- Store the column names into a variable for later use.
- Reindex the dummied test set variables to ensure all feature columns in the train set are also available in the test set.

Task 1D: 5 points

- Scale the training and testing data using the StandardScaler method.
- Tip: Only transform the testing data.

Task 1E: 10 points

- Define Logistic Regression without fitting the model.
- Use class_weight with 'balanced' to penalize the false positives more as the class is imbalanced.
- Use Repeated Stratified K Fold method with 5 splits, 3 repeats, and roc_auc scoring.
- Print the mean of roc_auc scores.
- Fit the training data.

Task 1F: 5 points

- Predict the class for the testing data.

- Create another dataframe that contains actual y values and y test probability values from the model.
- Note: y test probability values are for the event (i.e., 1, Spam).

Task 1G: 10 points

- Define a threshold of 0.5 and add a column for y test prediction based on the probabilities predicted for the thresholds.
- Print the confusion matrix.
- Plot the ROC_AUC curve.
- Print the AUC_ROC score for actual and predicted y.
- Comment your interpretations here.

Task 2: NLP on Email Data

- This task involves applying NLP techniques to the data, which consists of text from the email sent and their classification (Spam or Ham).

Task 2A: 7 points

- Map the 'Category' variable to 1 for spam and 0 for ham.
- Display a pie chart for the distribution of the two categories of emails.
- The pie chart should have a title, the name and percentage of each category, and a start angle equal to 90.
- Use the explode feature for the spam category.

Task 2B: 10 points

- Clean the text by removing special characters, stop words, lemmatizing the words, and keeping only alphabetic text with a length greater than 2.
- Lowercase the text before applying the cleaning techniques.

Task 2C: 5 points

- Split the data into training/testing with an 80-20 ratio and stratify the data.
- Define stratified k-fold with 3 splits.

Task 2D: 5 points

- Use the Bag of Words technique with bigrams and apply it to the training and testing data.
- Use the TF-IDF technique with bigrams and apply it to the training and testing data.
- Store the modified data into different variables.
- Note: Only transform the testing data.

Task 2E: 15 points

- Apply machine learning algorithms (Multinomial Naive Bayes, Logistic Regression, and SVM) to the cleaned and transformed data.
- Use Roc-Auc and precision as evaluation metrics.
- Display the scores in a tabular format.
- Perform hyper-parameter tuning for each algorithm with grid search and use the best estimated model for each type to test the data further.
- Display the confusion matrix.
- Note: Look at the model performance while selecting the Bag of Words and TF-IDF techniques for each algorithm.

Task 2F: 8 points

- Display a bar chart for the precision scores for each model.
- Display the percentages on the bar.
- Apply opacity to the bars according to the scores (1 being the highest).
- Display a proper title and axis names.
- Interpret the results and provide conclusions.

Programming Assignment Details:

- You can use the following libraries: Numpy, Pandas, Scikit-Learn, Matplotlib, and Seaborn.
- Explain each task and its actions in Markdown format.
- Comment your code.
- If using any external resources (books, internet), cite them within the cell.
- Do not rename the dataset files.

Submission Details:

- Fill in the name and ID of each group member in the Jupyter notebook in the provided format.
- Name your submission files using the following format: yourLastName_Last4digitsofyourID.ipynb (e.g., smith_1234_assignment2.ipynb).
- Only one team member should submit the file.