AI and ML for Cybersecurity

Midterm Exam Retake

30 points, 3 hours

Nino Bukuri

Rules

- The document is private, do not share it with others.
- Searching on the internet and using AI is allowed and encouraged during the exam. If you're unsure about any technical concepts, you can use these resources to quickly fill knowledge gaps.
- Communication with other people is prohibited during the exam. The same code or text within two or more reports will lead to leaving the exam without evaluation for all such reports.
- The exam will not be evaluated in case of violation of the rules and cheating.
- Create one, new, public GitHub repository "`aimlmid2026_n_bukuri25`". In response to the exam, create the appropriate files, upload them to the repository, and place a report there in the form of a README.md file. Other formats are not accepted. The report should be easy to read. It should be clear to the reader where the answer to which question is given. The reader of your report should be able to easily reproduce the work you have completed. The program code should be written in Python.
- Send me the GitHub repository link by email at "`p.gogishvili@sangu.edu.ge`" as your exam response.
- The GitHub repository must be live until the feedback and evaluation of the exam is submitted.
- The email should be sent before exam deadline. Late answers (GitHub file upload dates will be checked) will not be evaluated.

Assignment

Finding the logistic regression model coefficients

10 points

Find the data at the following address "`max.ge/aiml_midterm/11221924_html`". On the given online graph, the data is displayed with color dots. When hovering the mouse over the data, the coordinates of the data point are displayed on the screen.
Create logistic regression model (in Python) for claasification given points into the following classes:

- All points below the line – class 1;
- Purple points – class 2;
- Blue points – class 3;

Find the model coefficients and describe the process in your report. (5 points).
The report must also include a relevant graph for visualization. (5 points).

Spam email detection

20 points

Given the data file at the following address "`max.ge/aiml_midterm/11221924_csv`" with email features and its classes (spam or legitimate). The main goal of this task is to develop one Python console application for email classification within spam and legitimate classes. Your program should do the actions described below. You should provide the corresponding data in the report as described below:

1. Upload the provided data file to your repository and provide a link to the uploaded file in your report. (1 point).
2. Your application should create and train a logistic regression model on 70% of this data (2 points). Provide a link to the appropriate source code(s) in the report (1 point). In the report, provide and describe the data loading and processing code (2 points), the model used (with the code) for logistic regression (1 point), and provide the coefficients found by your model (1 point).
3. Within your application validate (find the Confusion Matrix and Accuracy) your model on the data that you have not used for training (1 point). Present the Confusion Matrix and Accuracy in the report and describe the code for finding them. (2 points).
4. Your application should have ability to check email text i.e. parse it, extract the same features that you have in provided data file and evaluate it for spam using your model. (3 points).
5. Compose an email text (manually) that will be classified by your model as spam. Provide the composed email and appropriate explanation (how it was created to be the spam) in the report. (1 point).
6. Compose an email text (manually) that will be classified by your model as legitimate email. Provide the composed email and appropriate explanation (how it was created to be legitimate) in the report. (1 point).
7. Generate and include them in your report two (2) distinct visualizations using libraries like matplotlib or seaborn to provide insights into your data and model. Requirements for each visualization: Include the Python code used to generate the graph; The graph must have a title, axis labels, and a legend (if applicable); Provide a 2-3 sentence explanation of what the graph reveals about the dataset or the model's performance. (4 points).

Visualizations can be selected from the following options or you can create other, meaningfull ones:

A: Class Distribution Study – Create a Bar Chart or Pie Chart showing the ratio of Spam vs. Legitimate emails in the dataset. This helps evaluate if the dataset is imbalanced.

B: Confusion Matrix Heatmap – Instead of raw numbers, provide a graphical heatmap of your Confusion Matrix. This must include labels for "`Predicted`" and "`Actual`" classes.

C: Feature Importance/Correlation – A Heatmap showing the correlation between different features, or a Bar Chart showing which features (words/attributes) had the highest coefficients in your Logistic Regression model.