CAP5771 - Spring 2025 - Class Project

Description

In this project, you will have the opportunity to go over the whole data science process (refer to the CRISP-DM[1] methodology) to generate a tool to help your customers

[1] Cross-Industry Standard Process for Data Mining is used to data intensive projects

improve their processes. This project can be done individually or in pairs and has three milestones before the final submission. This document introduces the breadth and expectations of the semester project. The goal is to provide a guided data science experience and allow the students to demonstrate their knowledge of the principles of data science.

Team

- **Individual:** If working individually, you will be solely responsible for all aspects of the project.
- **Two-person:** If working in a team of two, you will need to collaborate and divide the workload between yourselves. Clearly define the specific tasks and responsibilities each team member will undertake to justify the need for a two-person team. We expect that the amount of effort each member puts forward will be equal to the effort of an individual working alone.

Data

To ensure the project maintains a certain level of complexity, students are required to identify an application that uses **three or more** of the approved datasets.

Upon the selection of three or more datasets, students can then proceed to develop a tool that effectively utilizes the aforementioned data. Part of the assignment is to develop a method of combining or integrating the data to support your tool.

Datasets: Identify 3 or more datasets from any of the following sources. You can select other data sources, if discussed and approved by your corresponding instructor.

UF Hosted Datasets https://help.rc.ufl.edu/doc/AI_Reference_Datasets

Kaggle Datasets https://www.kaggle.com/datasets

Huggingface Datasets https://huggingface.co/datasets

Source: Clearly mention the sources of the datasets, including the URLs if available and their properties, including the sizes and attributes. The sources available at these locations should also be licensed to permit academic use.

Tools

I. **Language:** Python

II. **Libraries and Frameworks:**Identify and justify the non-standard libraries that you will use.

Below is a list of libraries we may discuss in class.

A. Data storage: SQLite, DuckDB, PostgreSQL
B. Data manipulation: Pandas, NumPy
C. Visualization: Matplotlib, Seaborn
D. Machine learning: Scikit-Learn, Pytorch
E. Dashboarding: Streamlit, Plotly Dash
F. Conversational agents: Rasa, Dialogflow, LangChain, your preferred LLM APIs
G. PDF reporting: ReportLab, FPDF

Type of project

Select the type of data science tool you will create. Your selection should fall in one of the categories below. These are examples, please produce more.

A. **Interactive dashboard**: Create an interactive dashboard to visualize patterns, geospatial data, time series forecasting, and key statistics from your selected datasets.

B. **Conversational agent**: Develop an interactive chatbot tool that supports querying over your datasets.
C. **Recommendation Engine:** Build a predictive model that recommends personalized items based on user behavior and historical data from three different sources.

The type of project indicates the way the user will interact with your tool. However, all of these projects will include a modelling and prediction component.

Methodology

Your methodology should follow the Cross Industry Standard Process for Data Mining (CRISP-DM) methodology or similar life cycle. You should be as detailed as possible when describing your steps. Below is an example outline.

1. **Data Collection and Preprocessing:**
    - Collect the data from the specified source.
    - Describe the dataset, including its dimensions (number of rows and columns), the variables it contains, and their data types.
    - Clean and preprocess the data, handling missing values, outliers, and inconsistencies.
2. **Exploratory Data Analysis (EDA):**
    - Perform exploratory data analysis to gain insights into the data.
    - Use descriptive statistics and visualizations to identify patterns, trends, and relationships.
3. **Feature Engineering and Selection:**
    - Engineer new features from existing ones to improve model performance.
    - Select the most relevant features for the chosen machine learning task.
4. **Data Modeling**

    - Select appropriate machine learning models or statistical techniques for the task.
    - Train and evaluate the models on the dataset.
    - Fine-tune the model parameters to optimize performance.
5. **Evaluation and Interpretation:**
    - Evaluate the model's performance using appropriate metrics.
    - Interpret the model's output and draw conclusions from the analysis.
6. **Tool Development:**
    - Develop a tool (dashboard, conversational agent, or recommendation engine) to showcase the project findings and insights.
    - Ensure the tool is user-friendly and effectively communicates the results.

Project Timeline

Below is a brief list of the dates and items required for each milestone. We will give more detailed information concerning the submission for each as they come.

Milestone 1: Data Collection, Preprocessing, and Exploratory Data Analysis (EDA)

Timeline: February 5, 2025 - February 21, 2025 (2 weeks)

Tasks

- **Data Collection**
    - Identify and acquire the dataset from a reliable source (e.g., Kaggle, AI Reference Datasets).
    - Verify dataset accessibility and ensure compliance with licensing or usage restrictions.
    - Document the dataset source, dimensions, and variable descriptions.

- **Data Preprocessing**
    - Handle missing data through imputation or removal (e.g., mean imputation, mode imputation, dropping rows).
    - Address outliers using statistical methods or domain knowledge (e.g., z-score thresholding, interquartile range).
    - Normalize or scale features as needed for analysis (e.g., Min-Max scaling, standardization).
- **Exploratory Data Analysis (EDA)**
    - Use descriptive statistics (e.g., mean, median, standard deviation) to summarize the dataset.
    - Create visualizations (e.g., histograms, scatter plots, box plots) to identify patterns, trends, and relationships.
    - Identify potential issues such as multicollinearity or skewed distributions using correlation analysis and statistical tests.

Deliverables

- Report containing: Objective of the project, indicating type of tool, data to be used, tech stack, and project timeline with specific dates for each of the future tasks, and EDA report with key insights (charts, tables, visualizations, and observations). The repository must contain a folder, and a file called **Reports/Milestone1.pdf** containing the outlined information.
- Any .ipynb and .py files used for the EDA
- The project should be fully contained in a private GitHub Repo with the name format **https://github.com/<username>/cap5771sp25-project** . Where <username> is the GitHub Id of one person in the project group. The other student may fork the repo to also keep a copy.

- Invite the following project staff as project collaborators: TA Jimmy [@JimmyRaoUF], Grader Daniyal [@abbasidaniyal], Dr. Cruz [@lcruzcas], and Dr. Grant [@cegme] to the project GitHub repo.
- Use the grade assignment to link your repository. Gradescope will import all the necessary files from your repository for the submission.

Example of project structure, including all milestones, however up until this submission you should only include the Milestone1.pdf

Note: Your Readme.md file should contain all contain all the necessary instructions required to reproduce your finding.

```
cap5771sp25-project/
│
├── Readme.md
│
├── Data/
│   ├── dataset.csv
│   └── data_access_info.txt
│
├── Report/
│   ├── Milestone1.pdf
│   ├── Milestone2.pdf
│   └── Milestone3.pdf
│
└── Scripts/
    ├── main.py
    ├── analysis.py
    ├── visualization.py
    └── requirements.txt
```

Milestone 2: Feature Engineering, Feature Selection, and Data Modeling Timeline

February 21, 2025 - March 26, 2025 (5 weeks)

Tasks

- **Feature Engineering**
  - Create new features from existing ones to improve model performance (e.g., interaction terms, aggregations, time-based features).
  - Encode categorical variables using techniques such as one-hot encoding or label encoding.
- **Feature Selection**
  - Evaluate feature importance using methods like correlation analysis, chi-square test, or tree-based feature importance.
  - Reduce dimensionality using techniques like PCA or LASSO if necessary.
- **Data Modeling**
  - Split the dataset into training, validation, and testing sets.
  - Train at least three machine learning models to accomplish your objective (e.g., logistic regression, decision trees, random forests, support vector machines, neural networks).
  - Evaluate and compare each model's performance using appropriate metrics (e.g., accuracy, precision, recall, F1-score, ROC curve).

Deliverables

- Selected features with justification for inclusion/exclusion.
- Trained models with performance metrics on validation data.
- Analysis of model performance and comparison of different models.

- Updated repositories with all current information.

Submission

Report containing: Objective of the project, indicating type of tool, data to be used, tech stack, and project timeline with specific dates for each of the future tasks, and EDA report with key insights (charts, tables, visualizations, and observations), feature engineering, feature selection, data modeling. The repository must contain a folder and a file called **Reports/Milestone2.pdf** containing the outlined information.

- Any .ipynb and .py files used for in your development
- The project should be fully contained in a private GitHub Repo with the name format **https://github.com/<username>/cap5771sp25-project** . Where <username> is the Github Id of one person in the project group. The other student may fork the repo to also keep a copy.
    - Invite the following project staff as project collaborators: TA Jimmy [@JimmyRaoUF], Grader Daniyal [@abbasidaniyal], Dr. Cruz [@lcruzcas], and Dr. Grant [@cegme] to the project GitHub repo.
    - Use the grade assignment to link your repository. Gradescope will import all the necessary files from your repository for the submission.

Milestone 3: Evaluation, Interpretation, Tool Development, and Presentation

Timeline: March 24, 2025 - April 23, 2025 (5 weeks)

Tasks

- **Evaluation and Interpretation**
    - Evaluate model performance on the test set using the same metrics as in training.
    - Interpret model outputs to derive actionable insights and explain predictions.
    - Identify potential biases or limitations in the model.
- **Tool**
    - Develop an interactive dashboard using Streamlit or Plotly Dash to visualize key findings and KPIs. OR
    - Optionally create a conversational agent (e.g., Rasa) for answering project-related questions. OR
    - Implement an automatic PDF reporting tool to summarize findings.

Deliverables

- Finalized report in PDF format.
- The project should be fully contained in a private GitHub Repo with the name format **https://github.com/<username>/cap5771sp25-project** . Where <username> is the Github Id of one person in the project group. The other student may fork the repo to also keep a copy.

    - Invite the following project staff as project collaborators: TA Jimmy [@JimmyRaoUF], Grader Daniyal [@abbasidaniyal], Dr. Cruz [@lcruzcas], and Dr. Grant [@cegme] to the project GitHub repo.
    - Use the grade assignment to link your repository. Gradescope will import all the necessary files from your repository for the submission.
- Tool demo (A four minute short video showcasing your tool) - Also add the video of demo to the project README.md.
- **4-minute presentation:** Summarize the project, methodology, findings, and tool demonstration in a powerpoint presentation format.

Presentation

More details to come in the future.

The project will be presented and include:

- **4-minute presentation:** Summarize the project, methodology, findings, and tool demonstration in a powerpoint presentation format.
- **4-minute demo:** Showcase the functionality and features of the developed tool.

Regulations on Using LLMs

The use of Large Language Models (LLMs) such as ChatGPT is permitted for this project, but with the following regulations:

1. **Declaration:** You must clearly declare the specific parts of the project where you used an LLM and how it helped you. This includes providing the prompts you used and explaining how the LLM's output was incorporated into your project.
2. **Transparency:** The use of LLMs should not be hidden or misrepresented as your own work. Be transparent about how the LLM contributed to your project.

3. **Understanding:** You are responsible for understanding the LLM's output and ensuring that it is accurate, relevant, and appropriate for your project. Do not blindly copy and paste LLM-generated content without critical evaluation.
4. **Originality:** LLMs should be used as a tool to assist you in your project, not to replace your own original thinking and analysis. The majority of the project should be your own work.

Examples of Unacceptable LLM Use

- **Generating entire sections of your report:** You should not use an LLM to generate entire sections of your report, such as the introduction or discussion.
- **Copying and pasting code without understanding:** You should not copy and paste code generated by an LLM without understanding how it works and how it fits into your project.
- **Fabricating results:** You should not use an LLM to fabricate results or to make your analysis appear more sophisticated than it is.

Consequences of Violating LLM Regulations

- **Where:** The violations will be identified during the grading process, specifically when reviewing the code, analysis, and report.
- **How:** The instructor will assess the extent of the violation, considering factors such as the degree of unoriginality, the impact on the project's integrity, and the level of transparency in declaring LLM usage.
- **Requirements for Submission:** You are required to submit your code, accompanying documentation, and a clear declaration of any LLM assistance, including prompts and explanations of how the output was utilized.

Any violation of these regulations may result in a reduction of your project grade or other disciplinary action https://sccr.dso.ufl.edu/process/student-honor-code/.

Additional Notes

- Ensure that you adhere to all ethical guidelines and privacy considerations when working with the data.
- Cite all relevant sources and references throughout your report and presentation.
- Consult the TAs and instructors if you are unsure about any stage of the process.
- If you have a multi-person team, record each person's individual contribution. We recommend keeping a log of each component or tasks contributed and for inclusion to the final project submission.

Pro Tips

Extract a small subset of data to work it while developing your project. You should investigate and develop first and scale second.

If your model is performing poorly or producing unexpected results, double-check your data preprocessing steps. Ensure that you have handled missing values appropriately, scaled or normalized features if necessary, and encoded categorical variables correctly. Further, *before* you run any function you should have an estimate of the result.

If you encounter errors related to data types or shapes, carefully examine the data structures you are using to store and manipulate your data. Make sure that the data types are compatible with the operations you are performing and that the dimensions of your arrays or matrices are correct.

If your model is overfitting (performing well on the training data but poorly on unseen data), consider using regularization techniques or increasing the amount of training data. You can also try simplifying your model architecture or reducing the number of features.

When visualizing your data, choose the appropriate chart type for the insights you want to convey. Use clear and concise labels and annotations to make your visualizations easy to understand.

If you are building a machine learning model, start with a simple baseline model and gradually increase complexity. This will help you understand the impact of different model architectures and hyperparameters on performance.

Keep track of your experiments and results using a tool like MLflow or Weights & Biases. These tools allow you to log your hyperparameters, metrics, and code versions, making it easier to reproduce and compare experiments.

When presenting your project, focus on the key insights and takeaways. Use clear and concise language and avoid technical jargon that your audience may not understand.