CS 301
Project 2

In this project, you will use a dataset from one of the listed data portals to build at least two machine learning models to demonstrate your technical and analytical skills. Upon project completion, you are required to produce an 8-minute presentation (slide deck with voice overlay), summarizing key aspects and findings. You must also submit your Jupyter notebook.

You can use the following resources to collect the data for the project:

Kaggle	kaggle.com	Over 50,000 datasets on a broad range of subjects. Also provides Jupyter notebooks that analyze the datasets.
FiveThirtyEight	data.fivethirtyeight.com	Datasets on politics, sports, science, economics, health, and culture, initially developed to support FiveThirtyEight publications.
University of California Irvine Machine Learning Repository	archive.ics.uci.edu	622 datasets, primarily in science, engineering, and business.
Data.gov	data.gov	U.S. government datasets on agriculture, climate, energy, maritime, oceans, and health.
World Bank Open Data	data.worldbank.org	Global datasets on subjects such as health, education, agriculture, and economics.
Nasdaq Data Link	data.nasdaq.com	Financial and economic datasets.
NYC Open Data	opendata.cityofnewyork.us	NYC government services datasets.
US Federal Reserve	www.federalreserve.gov/data.htm	

You may use another public data source with the instructor's approval.

In your project, you will complete the following tasks:

Task 1: Exploratory Data Analysis
Select a dataset from one of the resources above, ensuring it comprises a minimum of 10 features, with at least one categorical and one numerical feature. First, explain the dataset by providing a detailed analysis of each feature, highlighting aspects such as distribution and statistics. In cases where the dataset comprises an extensive number of features, prioritize the most significant ones and present a concise summary for the remaining attributes. When demonstrating the insights of the data, you must use proper visualizations and maintain good quality and standard in your graphs.

After explaining the dataset, explain the most important features that you have selected to train the model. As a justification, you may use visualization or some metrics that quantify the relationship between the target variable and the features.

Next, you will explain the preprocessing steps undertaken, including handling null values, cleaning the data, and the encoding process applied to the categorical feature.

Task 2: Train Models
In this step, initiate the process by clearly defining the type of machine learning problem you are addressing, such as classification or regression. You will train two models using the same training data with a minimum of three features where one of these features must be categorical. Next, you will explain the optimization algorithm used during the training phase. In instances where multiple optimization techniques are available, endeavor to implement at least two, providing a comprehensive explanation based on your understanding. While an exhaustive understanding of optimization techniques is not obligatory, your efforts to comprehend and explain them to the best of your ability are encouraged.

Task 3: Test and Evaluate
In this step, you will test both models using the test data and explain the evaluation metric with mathematical equations for clarity. In instances where multiple evaluation metrics are available, endeavor to employ at least two and expound upon the distinctions between them.

Task 4: Make Comparison
Compare the outcomes of both models and engage in a discussion on the strengths and weaknesses of each. Delve into how each model navigated the provided data, and if performance was suboptimal, provide an insightful analysis of the reasons behind it. This discussion should encompass an evaluation of the models' effectiveness, shedding light on their respective advantages and disadvantages. For instance, certain models exhibit limited robustness in handling outliers, while others excel in managing such data anomalies effectively. If, during your comparison of two models, you notice that one performed well while the other did not, it's valuable to investigate this discrepancy with consideration to their inherent capabilities regarding outlier handling, drawing on your prior knowledge in this regard. This exploration can provide insights into the specific aspects of the data where one model may have failed compared to the other.