**IERG3050 Project Proposal**: "Predicting Student Success with Logistic Regression: A Simulated and Real-World Analysis"

**Team Members**

Leung, Chi Wa 1155176527

Tsoi, Ming Hon 1155175123

Zhang, Yi Yao 1155174982

**Project Overview**

This project leverages logistic regression to predict student success (pass/fail) based on factors such as study hours, sleep hours, and attendance. By combining simulated data with a real-world dataset, we aim to build a robust classification model, evaluate its performance, and explore practical insights. The project integrates simulation, statistical modelling, and data analysis to demonstrate a deep understanding of logistic regression and its applications.

**Objectives**

1. Simulate a dataset of student performance to test logistic regression under controlled conditions.
2. Apply logistic regression to a real-world student dataset to validate findings.
3. Assess model performance using multiple metrics and visualizations.
4. Investigate challenges like data imbalance and regularization to enhance model robustness.
5. Deliver a comprehensive analysis suitable for a 30% course project, showcasing teamwork and technical skills.

**Methodology**

1. Data Collection and Simulation

* Simulated Data: Generate 1,000 student records using Python (e.g., numpy.random). Features include study hours (0-10), sleep hours (0-10), and attendance (0-100%), with pass/fail (0/1) outcomes based on a logistic probability function plus noise.
* Real Data: Use an open-source dataset (e.g., "Student Performance" from UCI Machine Learning Repository), selecting relevant features like study time and attendance. Clean data as needed (e.g., handle missing values).

1. Model Development and Training

* Implement logistic regression using Python’s sklearn.linear\_model.LogisticRegression.
* Train two versions: (1) basic model, (2) regularized model (L2 penalty, tuning C parameter).
* Split data into 80% training and 20% testing sets to evaluate performance.

1. Evaluation and Visualization

* Compute metrics: accuracy, confusion matrix, F1-score, and ROC curve (AUC).
* Visualize results: scatter plots with decision boundaries (simulated data) and ROC curves (both datasets).
* Address challenges: Test class\_weight='balanced' for imbalanced data and compare results.

1. Analysis and Reporting

* Compare simulated vs. real-world model performance.
* Explore insights (e.g., “How much do sleep hours impact passing?”).
* Compile findings into a report or presentation with visuals, code snippets, and conclusions.

**Tools and Resources**

* Programming: Python (libraries: numpy, pandas, sklearn, matplotlib, seaborn)
* Datasets: Simulated data (self-generated); real data (e.g., UCI Student Performance)
* References: Course materials, The Elements of Statistical Learning (Ch. 4), online tutorials (e.g., Kaggle, sklearn docs)

**Deliverables**

* Code: Fully documented Python scripts for simulation, modeling, and visualization.
* Report/Presentation: 8-12 pages or 12-minute talk (depending on course requirements), covering introduction, methods, results, and discussion.
* Visuals: At least 3 plots (e.g., decision boundary, ROC curve, feature impact).

**Expected Outcomes**

* A working logistic regression model with >80% accuracy (or justified lower performance based on data).
* Clear insights into how features affect student success.
* A polished deliverable demonstrating simulation, stats, and teamwork—worthy of a top grade.