STTHK 3013 (PATTERN RECOGNITION & ANALYSIS)

A241 – Individual Assignment #I (10%)

Instructor: Azizi Ab Aziz, *PhD*
Submission date: 10th Nov 2024 (before 11.59 pm) via UUM Learning Portal

"Be a lamp, or a lifeboat, or a ladder. Help someone's soul heal. Walk out of your house like a shepherd." [*Rumi*]

CASE STUDY: LEARNING ANALYTICS (PREDICTING STUDENT PERFORMANCE)

In educational research and analytics, understanding the drivers of student success can lead to better support strategies for both students and institutions. With more insights into how study habits, attendance, and prior performance impact exam outcomes, educators can identify students at risk of underperforming and develop personalized interventions.

Scenario:
A university's academic analytics team wants to improve their understanding of factors that contribute to students' final exam scores. They have collected data on a range of behavioural and performance indicators over the semester, including study habits, attendance, prior grades, and extracurricular participation. The team's goal is to develop a model to predict final exam scores and to identify the most impactful factors on student performance. By analyzing this dataset, students can engage in a realistic simulation of a common academic problem, learning how to apply feature engineering and regularized regression methods to generate insights.

Information about Dataset
The data is generated based on the following features:

Feature / Item	Description
final_exam_score	Final exam score, the target variable (0 to 100).
study_hours_per_week	Weekly study hours, randomly distributed (5-30 hours). This feature measures the average number of hours a student studies weekly. More time spent studying generally correlates with better performance, but there may be diminishing returns or other non-linear effects.
attendance_rate	Attendance percentage (0 to 100). Higher attendance might reflect students' commitment and discipline, often leading to better performance in exams.
previous_exam_scores	Average score on previous exams (0 to 100). This represents the average score from earlier exams in the semester. It gives an idea of students' baseline academic ability or consistency.
assignments_completed	Number of assignments completed, ranging from 0 to 20. The number of assignments submitted during the

	semester, capturing the effort and consistency of students in meeting academic requirements.
extracurricular_participation	Number of extracurricular activities (0 to 5).

In addition, feature engineering has been added for the following:

Feature / Item	**Description**
study_attendance_interaction	Interaction terms of study hours and attendance rate. This feature reflects how students who study more and also attend classes regularly may benefit more than those with lower engagement in either.
assignments_per_week	The ratio of assignments completed per week, calculated as assignments_completed / (study_hours_per_week + 1) (to avoid division by zero). It reflects a student's efficiency in completing assignments relative to time spent studying.
study_hours_per_week_squared	Squared terms for study_hours_per_week. *
attendance_rate_squared	Squared terms for attendance_rate. *

*study_hours_per_week_squared and attendance_rate_squared capture potential non-linear relationships, such as diminishing returns on studying and the possibility that extreme attendance percentages (very high or very low) might affect scores differently.

Tasks:
Based on this case study, perform the given tasks and answer the following questions.

- You are required to submit your code as a Jupyter Notebook / Python script.
- Write a brief report summarizing your approach to handling all questions.
- Discuss any challenges you encountered and how you addressed them.

1. Load and Explore the Dataset:
    a. Use Python libraries such as pandas to load and inspect the dataset (filename: assgmt01_student_performance_dataset.xlsx)
    b. Check for missing values and get a sense of the distribution of each feature.
    c. Replace possible missing values with your preferred methods.

2. Visualize the Data:
    a. Create visualizations such as distribution, 3D scatterplots, box plots, and correlation heatmaps to better understand the relationships between features.
    b. Explain at least THREE (3) features based on these methods
    c. Based on correlation analysis, identify and discuss which features may be most useful in predicting the final exam score.

3. Multicollinearity & Heteroscedasticity Evaluation:
    a. By using the statsmodels library, do the following:
        i. Identify possible features that may cause problems in terms of multicollinearity (if any, what should we do with those features?)

ii. Detect possible cases of heteroskedasticity based on the residual plot approach (if yes – why?).
iii. Suggest a strategy that could help mitigate the impact of multicollinearity and heteroskedasticity in this model.

4. Model Development:
    a. Split the Data: Split the dataset into training and testing sets (e.g., 60:40, 80:20, 90:10 split). Describe the importance of this split for model evaluation.
    b. Baseline Model: Train a simple linear regression model using only the original features (without the engineered features) and evaluate its performance on the test set.
    c. Lasso Regression: Implement Lasso regression on the training data and tune the regularization parameter (α) using cross-validation.
        i. Identify which features have non-zero coefficients in the final model.
    d. Ridge Regression: Implement Ridge regression, tuning the regularization parameter (α) through cross-validation.

5. Model Evaluation:
    a. Compare the performance (e.g., mean squared error, R^2 score) of:
        i. Baseline Linear Regression,
        ii. Lasso Regression,
        iii. Ridge Regression.
    b. Discuss which model performs best and why.
    c. Feature Importance: Analyse the Lasso regression coefficients to determine which features are most influential. Explain how Lasso's feature selection helps interpret the model.
    d. Regularization Effects: Discuss the effects of Lasso and Ridge regularization on the model, particularly in terms of feature selection and coefficient shrinkage.

6. Build a working application with a simple interface to deploy this solution (by using the best developed model).

Policy:

All grading of deliverables will be based on standards indicated for each deliverable. Deliverables may not be turned in late, and no cheating! For this class, cheating will include plagiarism (using the writings of another without proper citation), copying of another (either current or past student's work), working with another on individually assigned work, or in any other way presenting as one's work that which is not entirely one's work. The occurrence of plagiarism will result in removal from the course with a failing grade.