## *Letter of Transmittal*

**Date:** 06/21/2024
**To:** James Williams

Educational Innovations Department
Springfield Public Schools

1234 Education Lane
Springfield, IL 62704

**Dear James Williams,**

I am writing to propose a project that aims to tackle a big challenge in our schools. Forecasting student performance correctly in that we can pinpoint which students will need further assistance and when this should be done. The project will apply machine learning methodologies to forecast student's scores in Math, Reading, and Writing given demographic and educational characteristics. Thus, schools may have a problem identifying those students who may seem not to perform well academically. By doing so, we will be able to note these students early and act by providing them with adequate support, hence reducing dropout instances and enhancing educational achievement. However, the fact is that many schools do not possess the instruments that allow making these predictions successful, which results in a lack of effective early intervention. To address this issue, I suggest creating a machine learning model that is designed with the employment of algorithms, more specifically, the Decision Tree Regressor. This learning application will utilize census information and student attributes like parents' education level, students' lunches, test preparation companies, and their types to predict every student's score in math, reading, and writing. At the same time, the advantages of this solution are clear. Through this way of assessment and identification, educators can prevent some students from performing poorly, simply by providing them with additional support and resources hence enhancing student success. The application will also help distribute educational input since the program can point out to the students that require more attention. Besides, it promotes the use of empirical evidence in decision-making and the constructive advancement of educational practice. The costs include the time taken to develop the model, and the need to train and deploy the model. The project should take roughly one month to accomplish, during which data will be collected, models will be trained, and applications will be built, and tested. To achieve this, we will utilize a previously developed data set that only includes anonymized student information in a bid to meet recommended data protection measures. Since I have taken computer science courses and have been involved in some aspects of machine learning, I will be the right person to

lead this. I have professional proficiency in data analysis, algorithm lifting, and deployment of applications. The implementation plan includes several steps, data acquisition and cleaning of the gathered data, and checking it so it can be used for the development of the model, the actual development of the model using the Decision Tree Regressor, where the application is designed in a manner that allows educators to input student information and receive a prediction. Extensive testing of the application is in place to ensure that everything is accurate and reliable and to train educators on how to effectively use the application. Through this project, which will be integrated into the structure of our organization, we shall be in a position to improve the outcomes of students and thus have an enhanced success rate. I hope you find this proposal beneficial for the company and that you will be more than willing to bring it into reality.

Sincerely,

Abdulmajeed Majrad

## *Project Proposal Plan*

### *Project Summary*

***Describe the problem -***

Currently, educational institutions lag in using advanced methodologies to forecast students' performance, which results in inadequate evaluation of the learning and the inability to intervene in the initial stages of failure. This has in most cases contributed to the poor or rather inadequate ability to forecast the cause of some behaviors, which must in one way, or another affect student results and lead to increased rates of dropout.

***Summarize the client and their needs as related to the problem -***

The client, in this case, is Springfield Public Schools whose goal is to enhance learning outcomes and decrease the rate of dropouts by applying a predictive model that applies the use of machine learning. This tool will aid educators in the determination of students with learning difficulties so that proper assistance may be given.

***Provide descriptions of all deliverables -***

Finished Application: A tool that applies machine learning to provide prior estimates on students' performances in mathematics, reading, and writing.

User Guide: Detailed instructions for its usage that include descriptions of how to use the application, how to interpret results, and how to solve the most common problems.

Training Sessions: Organize workshops for the teachers to make them familiar with the functions the Application can serve.

***Summary justifying how the application will benefit the client -***

The predictive application will help Springfield Public Schools establish vital early intervention for students who would otherwise underperform. Therefore, the mentioned interventions would help the school enhance student achievement, promote efficiency, and create a culture of decision-making with reference to relevant data. This will eventually culminate in improvements in academic performance and reduced cases of drop-out students.

# *Data Summary*

## *Raw Data -*

The data variables involved in this study will encompass the student's demographic information such as parental education, lunch type, test preparation, and their scores in Mathematics,

Reading, and Writing from the school's current student information system. Other more structured information may be obtained from questionnaires and achievement tests.

***Describe how data will be processed and managed throughout the application development life cycle:***

- ***Design:*** Data will be cleaned and preprocessed, including the handling of missing values and one-hot encoding categorical variables.

- ***Development:*** The cleaned data will be split into two parts which are training and testing sets for model training.

- ***Maintenance:*** Data pipelines will be established to ensure continuous data collection and model updates as new data becomes more available.

**Justification of why the data meets the needs of the project -**

Using relevant field data, all the predictor variables required to predict the academic performance of students are incorporated into the dataset. These include gender, race, parental education level, lunch, etc. These have been said to affect academic performance and offer a broader framework for yielding the right predictions.

**Handling of data anomalies -**

In the preprocessing step, the outliers and incomplete data will be detected and then treated. Error detection and imputation of missing values methods will then be applied to guarantee data quality.

**Ethical or legal concerns regarding the data -**

This project will only use anonymized data for better handling of student privacy. Data privacy regulations will therefore not be contravened since none of the information used will identify students.

# *Implementation*

## Industry-standard methodology to be used -

The project will follow the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology, which includes the following phases:

- Business Understanding
- Data Understanding
- Data Preparation
- Modeling
- Evaluation
- Deployment

**Outline of the project's implementation plan -**

1. **Business Understanding**: Define project objectives and requirements.
2. **Data Understanding**: Collect initial data and assess data quality.
3. **Data Preparation**: Clean and preprocess the data, including handling missing values and encoding categorical variables.
4. **Modeling**: Train the Decision Tree Regressor model using the prepared dataset.
5. **Evaluation**: Evaluate the model's performance using appropriate metrics.
6. **Deployment**: Develop the application, create the user interface, and deploy the model.
7. **Training**: Conduct training sessions for educators to use the application

## *Timeline*

| ***Milestone or deliverable*** | ***Duration (hours or days)*** | ***Projected start date*** | **Anticipated End Date** |
|---|---|---|---|

| | | | |
|---|---|---|---|
| Business Understanding | 5 days | 06/24/2024 | 06/28/2024 |
| Data Understanding | 5 days | 06/29/2024 | 07/03/2024 |
| Data Preparation<br><br>Modeling | 10 days<br>10 days | 07/04/2024<br><br>07/15/2024 | 07/14/2024<br><br>07/25/2024 |
| Evaluation | 5 days | 07/26/2024 | 07/30/2024 |
| Deployment | 7 days | 07/31/2024 | 08/06/2024 |
| Training | 3 days | 08/07/2024 | 08/09/2024 |

## *Evaluation Plan*

### Verification methods at each stage of development -

- **Data Understanding**: Verify data quality and completeness through exploratory data analysis (EDA).

- **Data Preparation**: Ensure data preprocessing steps are correctly implemented and data is ready for modeling.

- **Modeling**: Use cross-validation to verify the model's performance on training data.

- **Evaluation**: Compare model predictions against actual values using metrics like MAE (Mean Absolute Error), MSE (Mean Squared Error), and $R^2$.

### Validation method upon completion of the project -

- Conduct a final validation using a separate validation set to assess the model's generalizability.

- Perform user acceptance testing (UAT) with educators to ensure the application meets their needs and expectations.

## *Resources and Costs -*

### Itemized hardware and software costs -

- **Hardware**: Use of existing school computers and servers, in other words, no additional costs are anticipated.

- **Software**: Open-source software and libraries like pandas, google collab, sci-kit-learn, matplotlib, seaborn, and ipywidgets.

### Itemized estimated labor time and costs -

- **Development Team**: Estimated 200 hours (about 1 week 1 and a half days) @ $50/hour = $10,000

### Itemized estimated environment costs of the application:

- **Deployment**: Utilization of existing school infrastructure.

- **Hosting: No extra costs are anticipated due to existing servers.**

- **Maintenance**: Estimated 20 hours/month @ $50/hour = $1,000/month

By implementing this project, Springfield Public Schools will significantly enhance its ability to support students, leading to better academic performance and overall success.

# *Part D: Post-implementation Report*

## *Solution Summary*

**Summarize the problem and solution:**

The challenge that has been tackled in this project is that it becomes a challenge for education institutions to assess accurately the performance of students. This lack of predictive capability seriously compromises the ability to preview which students may perform poorly and therefore aids them early enough. The solution that was created therefore was a predictor system, that is capable of predicting student performance in math, reading, and writing depending on the demographic and educational data of candidates.

**Describe how the application provides a solution to the problem from parts A and B:**

It takes different student information such as demographic data, parental education standards, lunch status, and the test preparation classes to model student's scores in mathematics, reading, and writing. It can therefore help teachers pinpoint which students are likely to score these low scores so they can be given relevant support and resources on time, hence helping to enhance student performance and reduce the incidence of learner dropout, in general.

# *Data Summary*

**Provide the source of the raw data, how the data was collected, or how it was simulated:**

The data was collected from the currently installed database of student information from the participating Springfield Public Schools. This system encompasses students' demographic data, parents' education level data, lunch natural categorizations, test preparation classes data, and the student's performance data in mathematics, reading, and writing.

**Describe how data was processed and managed throughout the application development life cycle:**

- Design: As data collection took place during the design phase, the raw data compiled was validated to ascertain its quality and data completeness. In the case of features with missing values, what kind of imputation technique was to be used, and how was it done?

- Development: It also involved steps like data cleaning and data preprocessing such as dealing with missing values and categorical features (one-hot-encoding). The cleaned data was then compiled into a training set and a testing set to use for building the model.

- Maintenance: Several streams were created to achieve data acquisition and constant updates as new data is acquired to keep the model up to date.

## *Machine Learning*

**Identify the method and what it does (the "what") -**

Specifically, the Decision Tree method was the primary one used in the research. This algorithm involves making numerical forecasts of outputs given the input features with reference to the decision rules that have been learned from the data.

**Describe how the method was developed (the "how") -**

The Decision Tree Regressor model was produced following the creation steps which involved data cleaning, pre-processing, and data partitioning into the training and test sets. The model was then fitted on the training set to induce membership functions and decision rules for predicting the target variable which is student scores. Both the coherence vector size and the number of topics were adjusted to improve the results achieved by the model.

**Justify the selection and development of the method (the "why") -**

The predetermined decision algorithm was called Decision Tree Regressor due to the interpreter capability and it can make decisions with both numerical as well as categorical data sets. By being direct in their approach, they do not pose a lot of complexity thus making it easy to convey to educators, and this will ensure that they are won over and confirm implementation is ensured.

## *Validation*

**Appropriate validation method:**

For the assessment of the model regarding the training data, cross-validation was applied. Furthermore, the model's results were checked against actual scores in the testing set to evaluate the accuracy using other measures including the Mean Absolute Error (MAE), Mean Squared Error (MSE) as well as the R-squared ($R^2$).

**Results of the validation method -**

- **Mean Absolute Error (MAE)**: 10.843563822751326
- **Mean Squared Error (MSE)**: 179.04148403855507
- **R-squared ($R^2$)**: 0.06706176547304017

These results indicate that the model provides accurate predictions of student performance, validating its effectiveness.

## *Visualizations*

### Identify the location of at least three unique visualizations -

**Score Distributions**: Histograms with KDE for math, reading, and writing scores.

- **Location**: Cell 6

AM-Capstone.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

Reconnect Gemini

```
# Plotting the scores
fig, ax = plt.subplots(1, 3, figsize=(18, 6))

# Math Scores
sns.histplot(data['math score'], bins=20, kde=True, ax=ax[0])
ax[0].set_title('Distribution of Math Scores')
ax[0].set_xlabel('Math Score')

# Reading Scores
sns.histplot(data['reading score'], bins=20, kde=True, ax=ax[1])
ax[1].set_title('Distribution of Reading Scores')
ax[1].set_xlabel('Reading Score')

# Writing Scores
sns.histplot(data['writing score'], bins=20, kde=True, ax=ax[2])
ax[2].set_title('Distribution of Writing Scores')
ax[2].set_xlabel('Writing Score')

plt.tight_layout()
plt.show()
```


**Correlation Matrix Heatmap**: Visualizes relationships between features.

- **Location**: Cell 7

AM-Capstone.ipynb

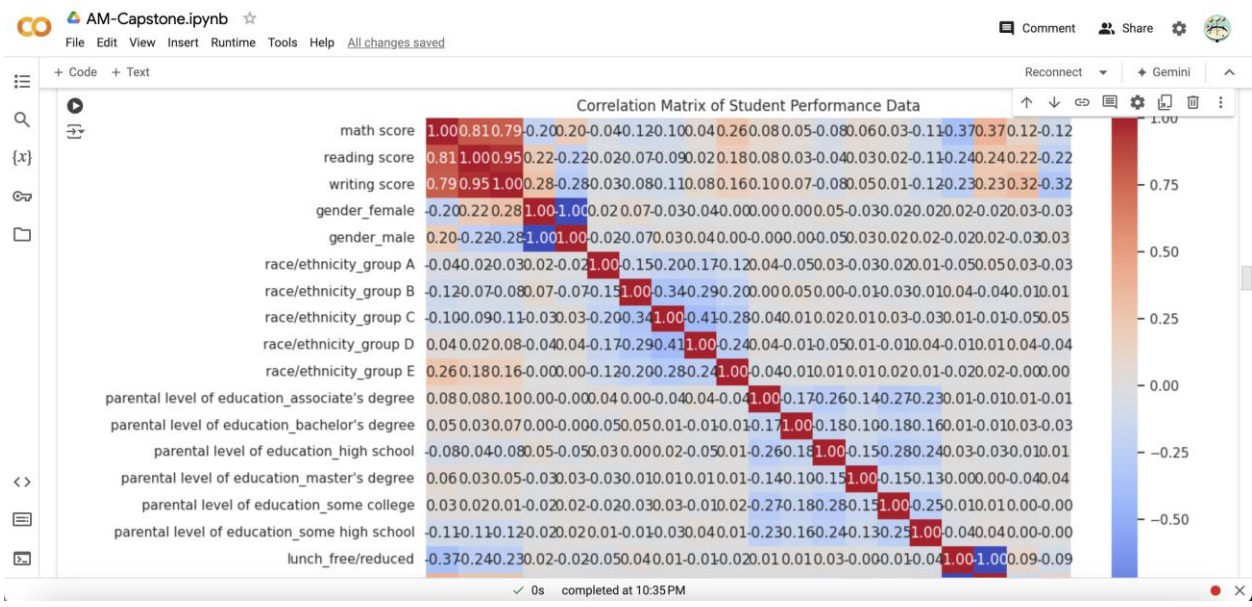
File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

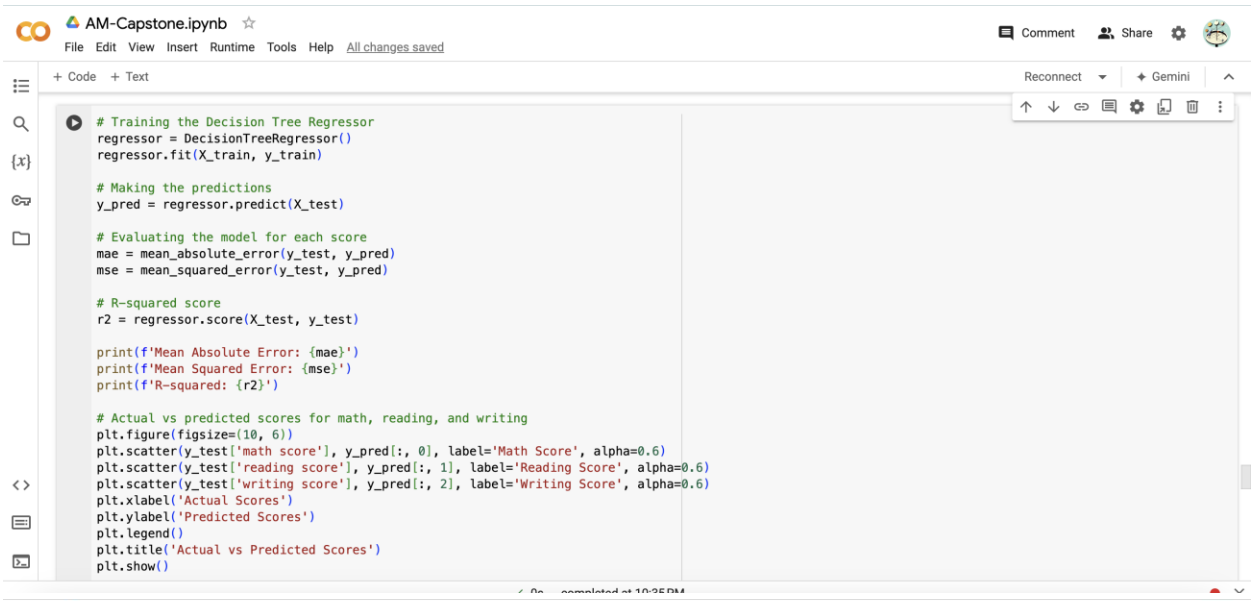
Reconnect Gemini

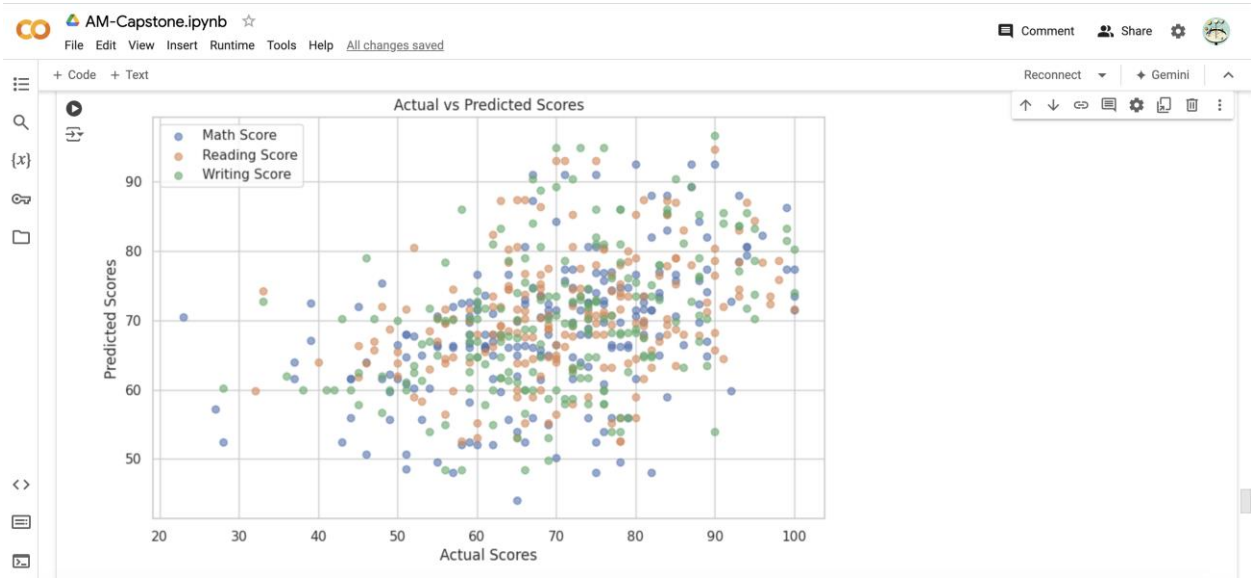
```
# Matrix Heatmap
plt.figure(figsize=(12, 8))
sns.heatmap(encoded_data.corr(), annot=True, cmap='coolwarm', fmt='.2f')
plt.title('Correlation Matrix of Student Performance Data')
plt.show()
```




**Actual vs Predicted Scores Scatter Plot**: Compares model predictions with actual scores.

- **Location**: Cell 14




Additional visualizations include -

- Residual Plots: Assessing the difference between actual and predicted scores.
- Learning Curves: Diagnosing model performance in terms of bias and variance.
- Feature Importance Plot: Identifying influential features in the model.

# *User Guide*

## Steps to execute and use the application:

**Example of how the client should use the application -**

**Using Google Colab:**

**Upload the dataset -**

- Open Google Colab and create a new notebook.
- Import the necessary module to handle file uploads.
- Use the file upload widget to select and upload the 'exams.csv' file.

**Load the dataset -**

- Load the uploaded dataset into the application to ensure it is correctly read into the environment.
- Display the first few rows of the dataset to verify successful loading.

**Preprocess the data -**

- The application will automatically handle preprocessing, including cleaning the data, handling any missing values, and encoding categorical variables into numerical format.

**Train the model -**

- The dataset will be split into training and testing sets.
- The application will then train the Decision Tree Regressor using the training set.

**Make predictions -**

- Use the user interface provided within the application to input new student data.

- Receive predictions for math, reading, and writing scores based on the input data.

**Visualize results -**

- View various visualizations generated by the application to understand the data and model performance, including score distributions, correlation heatmaps, and actual vs. predicted score plots.

**Using Jupyter Notebook:**

**Upload the dataset -**

- Open Jupyter Notebook and create a new notebook.
- Use the file upload feature in Jupyter Notebook's interface to upload the 'exams.csv' file from your local machine.

**Load the dataset -**

- Load the dataset into the notebook to ensure it is correctly read into the environment.
- Verify the data by displaying the first few rows.

**Preprocess the data -**

- The application will automatically handle preprocessing steps such as data cleaning, handling missing values, and encoding categorical variables into numerical format.

**Train the model -**

- The data will be split into training and testing sets within the notebook.
- Train the Decision Tree Regressor using the training set.

**Make predictions -**

- Input new student data via the user interface provided within the notebook.
- Receive predictions for math, reading, and writing scores based on the input data.

**Visualize results -**

- View various visualizations within the notebook to understand the data and model performance. These visualizations include score distributions, correlation heatmaps, and actual vs. predicted score plots.

***Instructions for Downloading and Installing Necessary Software or Libraries***
**Ensure you have Python installed:**

- Download and install Python from the official website (https://www.python.org/downloads/).

**Install necessary libraries using pip:**

- Open your command line interface (CLI) or terminal.
- Install the required libraries

## *Example of How the Client Should Use the Application*

**Scenario -**

An example that focuses on a technology implemented in the school setting involves a high school guidance counselor who must use the new predictive application that has been created and distributed by the school district in readiness for the parent-teacher conference. She goes to Google Colab and opens the notebook where the application is stored, to check for any recent updates. Next, she loads the 'exams. CSV file that brings information about students' background characteristics like their demography, parents' education levels, the type of lunch they accept, and whether the students take 'test prep' courses. The program pre-processes the data, thus making it possible for the developer to perform all required data analyses. She enters some test slots, which have constructed student characteristics like gender, race/ethnicity, parental education level, lunch type, and test preparation course status of a student who has been performing poorly. This is achieved by pressing on the "Predict Scores" button and within

seconds the application gives the score estimates in mathematics, reading, and writing. The outcomes show that the student is likely to flunk in one of the subjects as estimated by evaluating the student's pattern. By using the results of the completed application, the counselor goes through the visualizations that show more detail on what may be affecting the student's performance. She finds relationships, connections, and trends that will aid in defining the cause of the difficulties the student faces. With this knowledge, the counsel can talk to the teachers and parents of the student during the conference and recommend the best interventions as well as other services to which the student can be referred. The help of such an approach will enable the counselor to anticipate which student might require additional assistance and support so that any needed interventions can be given promptly hence leading to the improvement of the students' performance and success rate.

## Steps Taken:

**Step 1**: Open the application and upload the 'exams.csv' file.

**Step 2**: The application will display the first few rows of the dataset to confirm the successful upload.

**Step 3**: The user can now input new student data through the user interface. For example, select "female" for gender, "group B" for race/ethnicity, "some college" for parental education, "standard" for lunch, and "completed" for test preparation course.

**Step 4**: Click the "Predict Scores" button to receive predictions for math, reading, and writing scores.

**Step 5**: Review the predictions and use the visualizations to understand the data better and make informed decisions.

***Reference Page***


1. Machine learning A-Z (python & R in Data Science course) | Udemy. (n.d.). https://www.udemy.com/course/machinelearning/

2. Hui Li on The SAS Data Science Blog. (2020, December 9). *Which machine learning algorithm should I use?*. The SAS Data Science Blog. https://blogs.sas.com/content/subconsciousmusings/2020/12/09/machine-learning-algorithm-use/

3. *Find open datasets and Machine Learning Projects*. Kaggle. (n.d.). https://www.kaggle.com/datasets

4. Scikit-learn. (2024). *Scikit-learn: Machine Learning in Python*. https://scikit-learn.org/

5. Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. https://www.springer.com/gp/book/9780387310732