October 14, 2023

C964: Computer Science Capstone Template

Task 2 parts A, B, C and D

[Part A: Letter of Transmittal 1](#_Toc98598250)

[Letter of Transmittal Requirements 2](#_Toc1738085866)

[Letter Template 2](#_Toc1133266111)

[Part B: Project Proposal Plan 3](#_Toc1370766476)

[Project Summary 4](#_Toc904507251)

[Data Summary 4](#_Toc1736393957)

[Implementation 4](#_Toc1241988654)

[Timeline 4](#_Toc1357178365)

[Evaluation Plan 5](#_Toc623361460)

[Resources and Costs 5](#_Toc1538507987)

[Part C: Application 5](#_Toc1471073175)

[Part D: Post-implementation Report 6](#_Toc651895932)

[Solution Summary 7](#_Toc1134136520)

[Data Summary 7](#_Toc182221765)

[Machine Learning 7](#_Toc1505466430)

[Validation 7](#_Toc391434166)

[Visualizations 7](#_Toc201059345)

[User Guide 7](#_Toc1365484010)

[Reference Page 8](#_Toc1702353417)

# Part A: Letter of Transmittal

January 31, 2024

Bob Robertson

The Motorcycle Co.

42 Wallaby Way, Sydney, NSW

Dear Mr. Roberston,

It has come to my attention (through numerous customer complaints) that The Motorcycle Co’s website does not have any system in place to categorize motorcycles. This issue is driving away many customers due to their frustration with the website. Many customers begin shopping for a motorcycle with an idea of what category of motorcycle they want. The lack of categorization on the website causes frustration because, for example, if Timmy grew up watching MotoGP and is dead set on purchasing a sport bike, he has no way to view the sport bikes for sale. The Motorcycle Co. has thousands upon thousands of listings, so rather than sorting through all of these listings, Timmy will just go somewhere that makes it easy to look at the bikes he wants to purchase. Unfortunately, due to the absolutely confounding number of listings on the website, it is completely impractical to manually research and input the category of each individual motorcycle on the website, and would take thousands of man-hours.

The solution to this issue is a machine learning program. This program will be capable of determining a motorcycle’s categorization based on its weight and engine displacement. Using this algorithm, all listings on the website can be updated automatically without any research or wasted man-hours. This will save the company massive amounts of time and money, and will largely increase customer satisfaction with the website, thus leading to more sales.

The only cost that will be associated with this program is the cost of employing the software developer to create this program, which will be substantially less than the cost of someone researching and manually entering the categorization of every single listing. The estimated cost of the software developer is $30,000, at a rate of $15,000/month for two months. The estimated timeline for this program’s completion is approximately two months. The first month would revolve around the development of the program, while the second month would revolve around implementing the program into the website. Once the project is complete, there is no maintenance or upkeep cost, as any further listings added to the website will have the categorization entered manually. Due to the nature and source of this data, no personal information is collected or utilized, and there are no ethical concerns with this project. The selected developer has ten years of experience in building similar machine-learning algorithms and has a Master’s degree in Software Development.

Thank you for taking the time to read this proposal. Please feel free to reach out if you have any questions at my listed phone number or email.

Sincerely,

Nicolas Hardin

Nicolas Hardin, Student

# Part B: Project Proposal Plan

## Project Summary

The problem that this proposal aims to solve is the currently absent categorization system on The Motorcycle Co.’s website. Currently, when a customer wishes to use the company’s website to view motorcycles for sale, they are forced to wade through thousands of listings without the ability to search for specific types of motorcycles by category. For example, if Steve has decided that he is getting too old for his sport bike and wants a more comfortable adventure bike, he is unable to filter out unwanted motorcycle categories, such as enduros, sport bikes, minibikes, etc. that he does not wish to see. This leads to a large amount of frustration and customers being turned away due to the non-user-friendly nature of the website.

The Motorcycle Co. desperately needs a categorization system to be implemented on the website. This will drastically increase customer satisfaction and sales. Unfortunately, due to the massive number of listings on the website, it would take an incredibly large amount of time to manually research and enter in the category of every motorcycle listed on the website.

This project aims to solve the time dilemma by using a machine-learning program to automatically determine and enter the categorization of the motorcycles listed on the website. The deliverables in this proposal include a program that will be able to determine the categorization of a motorcycle by comparing its weight and engine displacement and a guide on how to utilize that application.

This application will make it significantly faster and more cost effective to implement categorization filters onto The Motorcycle Co.’s website. It will completely remove the need to manually research and enter in categorizations for thousands and thousands of listings. Additionally, this will largely increase customer satisfaction and increase sales, since customers will have a much easier time sifting through the motorcycles listed and viewing the specific category of motorcycles that they are searching for.

## Data Summary

The data for this program will be found on and downloaded from kaggle.com. Kaggle.com is an artificial-intelligence and machine-learning focused website that contains a large amount of datasets with various purposes. There is a dataset on this website named “Motorcycle Specifications Dataset” that seems ideal for this project and is sourced from bikez.com. This dataset contains a large number of motorcycles with information about them, including their categorization, engine displacement, and weight.

In order to properly utilize this data, the motorcycle classifications listed in the dataset will have to match the classifications needed on the website. When this data is being processed, all empty values will have to be removed, all unnecessary data will have to be removed, any categories of motorcycles not sold by The Motorcycle Co. will have to be removed, and some specific categories will have to be renamed. This can all be done via basic CSV editing tools provided in Microsoft Excel. Once the machine-learning program has been trained, the data will not need to be managed as any further listings posted on the website will have the category manually entered.

This data will meet the needs of the project because it includes all relevant data required by this proposal. The data needed to create this project includes motorcycle category, engine displacement, and weight, all of which are included in the dataset. Any data that does not have all three of these characteristics will have to be removed. All other characteristics of the motorcycles listed in the dataset such as make, model, year, etc. are irrelevant to the program and can be removed. Finally, any categories listed in the dataset not sold by The Motorcycle Co. such as ‘ATV,’ ‘Scooter,’ and ‘Prototype’ will be completely removed.

This project has no ethical or legal concerns. Since the data being utilized is strictly numerical data about motorcycles, there is no personal or branded information being used. As previously stated, the make and model of all bikes will be removed. Since this program is simply using numerical figures about a motorcycle to determine the motorcycle’s categorization, there are no ethical concerns.

## Implementation

The industry standard methodology that will be used for this project is the Waterfall method. The Waterfall Method is appropriate for this project due to its linear nature. All requirements for this project will be known in advance and one singular developer will be working on it. The sequential flow will guide the single developer through the process of developing the product. The Agile method does not suit this project because it is not necessary to go back and alter things that have already been completed.

1. Requirement Analysis
   1. Meeting between the software developer and the end user so that the exact specifications of what the program should do can be discussed.
   2. Determine requirements for the project and determine the scope of the project
2. System Design
   1. Analyze the requirements and scope determined in the previous step.
   2. Create a system design that explains how the program will be structured, methods needed in the program and what they will do, and the steps that need to be taken to deliver a complete program.
3. Implementation
   1. Begin creating the program based on the previously created system design.
   2. Write and test the code for each step of the program and each method outlined in the previous step
4. Integration and Testing
   1. Integrated all created modules and functions created in the previous step into one, cohesive program
   2. As each piece is implemented into the main program, perform integration tests to ensure that the added piece still functions with the rest of the program
   3. Test the program as a whole to ensure that all parts still work as intended after integration has been completed
5. Deployment
   1. Fix any bugs discovered in testing
   2. Allow end users to interact with the software to ensure that it is user-friendly
   3. Deliver the software to the client for their use
6. Maintenance
   1. Resolve any issues with the software reported by the client
   2. Since this program is a one-time-use program, it will not require maintenance after it has been successfully utilized by the client

## Timeline

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone or deliverable | Duration  (hours or days) | Projected start date | Anticipated end date |
| Design plan completion | 5 days | 02/01/2024 | 02/06/2024 |
| Program completion | 14 days | 02/07/2024 | 02/21/2024 |
| Program implementation and testing completion | 6 days | 02/22/2024 | 02/29/2024 |
| Program delivery | 6 days | 03/01/2024 | 03/07/2024 |

## Evaluation Plan

The verification methods that will be used at each stage of development are various tests. As the program is developed, each individual module and method created will be subject to unit tests involving a variety of different values. Once all parts have been created, they will be implemented one at a time, and every time a new part is integrated into the program, the program should be subject to tests involving a variety of different values. Once all parts have been successfully integrated into the program, all parts should be tested together in a system test. During this system test, the tester should be entering a variety of weights and displacements into the program and verifying that the output is an expected value. Once the program has passed these system tests, it should be given to a subset of end users so that they can test it and find any errors that have been missed and give feedback on the program’s usability.

Once the program has been completed and passed all tests, the accuracy metric will be evaluated. This accuracy measurement will be implemented into the program to measure its accuracy. This test will function by going through a large number of tests and comparing the values that the program outputs against the correct value. This test will then output an accuracy percentage that is calculated by dividing correct predictions by total predictions. Some errors are okay to be manually corrected, but a large number of errors will make the program pointless. This program needs to have an accuracy of ninety percent or higher to be considered successful.

## 

## Resources and Costs

* Software developer: $15,000/month for two months at a total of **$30,000**
* High-end laptop (including Windows) for software developer – **$1,500**
* 1 year of Microsoft 365 Business Standard – **$150**
* Programming language: Python – **FREE**
* Programming environment: PyCharm – **FREE**
* Libraries: Scikit learn, pandas, pyplot – **FREE**
* Kaggle.com data – **FREE**
* Since this program only needs to be run once to perform its function, it is not necessary to pay running costs such as server hosting, maintenance, etc.

# Part C: Application

The application is attached.

# Part D: Post-implementation Report

## Solution Summary

The problem listed in this proposal is that the listings on The Motorcycle Co.’s website do not contain categorizations, and due to the vast inventory of The Motorcycle Co., it is impractical to manually research and enter every motorcycle’s category. The solution to this problem is to train a machine learning algorithm to accurately guess the categorization of the motorcycle based on its weight and engine displacement.

The solution proposed in this document is to train and utilize a machine learning algorithm that can guess the categorization of a motorcycle based on its engine displacement and weight. This constitutes a solution to the listed problem because this algorithm should be able to automatically enter the categorizations of the motorcycles without doing any research thus saving countless man-hours of research and data entry.

## Data Summary

The raw data utilized to train the machine learning algorithm was collected from kaggle.com. The data downloaded from kaggle.com was originally sourced from bikez.com. This data was downloaded from kaggle.com as a CSV file utilizing the existing download feature of the website.

Once the data was downloaded, it was processed into the necessary data for the machine learning algorithm. First, all columns were removed from the CSV except for the data needed to train the algorithm – classification, weight, and engine displacement in this case. Next, all entries with missing data in any column were removed by using built-in Microsoft Excel tools. After that, the ‘ATV’, ‘Scooter’, and ‘Prototype’ categories and all corresponding data was removed since The Motorcycle Co. does not sell any of those products. Finally, some category names were altered to make the output easier to read and to remove special characters, and the data was compiled into a ‘pandas’ data frame in the program.

## Machine Learning

This program employed machine learning methods to predict the classification of a motorcycle based on its weight and engine displacement. In this program, the machine learning algorithm analyzed a dataset containing classifications of existing motorcycles along with their weight and engine displacement and used this data to predict the classification of the motorcycle entered into the program. The machine learning method used in this project was the logistic regression method from the ‘sickit learn’ library utilizing the ‘liblinear’ solver. This method is used for “classification and predictive analysis” and it “estimates the probability of an event occurring…based on a given dataset of independent variables” (*What Is Logistic Regression? | IBM*, n.d.). Additionally, this program used the ‘pyplot’ library to create visualizations and the ‘pandas’ library for data management.

This method was developed by first discovering and selecting data and then importing that data and converting it to a usable data frame using the aforementioned ‘pandas’ library. After that, the logistic regression algorithm was selected as the optimal algorithm based on the dataset’s size and contents. Next, a solver was selected from the ‘scikit-learn’ library to use with the linear regression algorithm. After testing various solvers, the ‘liblinear’ solver was selected due to a combination of factors including the lowest number of iterations, fastest speed, and highest accuracy. This algorithm was then applied to the data frame to train the machine learning algorithm and tested to ensure functionality. Finally, the program was converted to take user input and output human-readable data.

For this program, a supervised learning method was ideal because the data was labeled and contained the output. The logistic regression algorithm was used as it was the most appropriate choice based on the size and type of data that was being analyzed, and the ‘liblinear’ solver was utilized due to needing the lowest number of iterations, having the fastest speed, and having the highest accuracy rating.

## Validation

The accuracy of this program was validated using the built-in ‘train\_test\_split’ method from the ‘scikit learn’ library which splits data into random training subsets and testing subsets to measure the algorithm's accuracy. The percentage is then calculated by using the ratio of the correct predictions divided by the total predictions, which is then multiplied by one hundred to achieve a percentage-based result.

Since there are fourteen categories that the machine learning algorithm can choose from, the accuracy rating of the program if it were randomly guessing would be one out of fourteen, or roughly seven percent. The results of this validation method return a roughly thirty-seven percent accuracy rating. While this is significantly better than randomly guessing, it is still relatively low. This is likely due to an abnormal distribution of data, large outliers, and some categories having a vastly larger range in engine displacement than other categories. A larger dataset containing more specific categorizations and a higher number of categorizations would likely yield a higher accuracy rating.

## Visualizations

Once all steps of the user guide are completed, the visualizations will be located under the program’s output in the Notebook.

## User Guide

1. Download and install Python 3.12
2. Install Jupyter Notebook using the following command in command prompt:
   1. py -m pip install notebook
3. Install ‘matplotlib’ using the following command in command prompt:
   1. py -m pip install matplotlib
4. Install ‘scikit-learn’ using the following command in command prompt:
   1. py -m pip install scikit-learn
5. Install ‘pandas’ using the following command in command prompt:
   1. py -m pip install pandas
6. Run Jupyter Notebook using the following command in command prompt:
   1. py -m jupyter notebook

NOTE: If Jupyter Notebook does not automatically open in a browser, copy and paste the link listed under the section labeled “This page should redirect you to a Jupyter application” into a browser

1. Navigate to where the program was downloaded to in the Jupyter interface
2. Open “Capstone Jupyter.ipynb”
3. Select the “Run” tab at the top of the notebook, and select “Run all cells”
4. Enter the information into the prompts asking for the weight (in kg) and engine displacement (in cc) of the motorcycle

# Reference Page

*What is Logistic regression? | IBM*. (n.d.). https://www.ibm.com/topics/logistic-regression