April 05, 2023

**C964: Computer Science Capstone**

By: Austin Kim

Task 2 – Parts A, B, C, D, E

[Part A: Project Proposal for Business Executives 3](#_Toc131625851)

[Letter of Transmittal 3](#_Toc131625852)

[Project Recommendation 3](#_Toc131625853)

[Problem Summary 3](#_Toc131625854)

[Application Benefits 3](#_Toc131625855)

[Application Description 3](#_Toc131625856)

[Data Description 3](#_Toc131625857)

[Objectives and Hypothesis 4](#_Toc131625858)

[Methodology 4](#_Toc131625859)

[Funding Requirements 4](#_Toc131625860)

[Data Precautions 4](#_Toc131625861)

[Developer’s Expertise 4](#_Toc131625862)

[Part B: Project Proposal 5](#_Toc131625863)

[Problem Statement 5](#_Toc131625864)

[Customer Summary 5](#_Toc131625865)

[Existing System Analysis 5](#_Toc131625866)

[Data 5](#_Toc131625867)

[Project Methodology 5](#_Toc131625868)

[Project Outcomes 5](#_Toc131625869)

[Implementation Plan 6](#_Toc131625870)

[Evaluation Plan 6](#_Toc131625871)

[Resources and Costs 6](#_Toc131625872)

[Timeline and Milestones 6](#_Toc131625873)

[Part C: Application 7](#_Toc131625874)

[Part D: Post-implementation Report 8](#_Toc131625875)

[A Business (or Organization) Vision 8](#_Toc131625876)

[Datasets 8](#_Toc131625877)

[Data Product Code 9](#_Toc131625878)

[Objective (or Hypothesis) Verification 11](#_Toc131625879)

[Effective Visualization and Reporting 11](#_Toc131625880)

[Accuracy Analysis 11](#_Toc131625881)

[Application Testing 11](#_Toc131625882)

[Application Files 11](#_Toc131625883)

[User Guide 12](#_Toc131625884)

[Summation of Learning Experience 12](#_Toc131625885)

[Part E: Sources 12](#_Toc131625886)

# Part A: Project Proposal for Business Executives

## Letter of Transmittal

Write a single-page cover letter to the organization’s senior leadership. The letter should be concise and target a non-technical audience. Include the following:

* Problem summary.
* Recommendation for a solution centering around your application (called a *data product* in the task directions).
* Describe how the proposed solution benefits the organization.
* Provide an estimate of the total cost (this should match the total given in *Funding Requirements* of part A).
* Expertise and experience qualify you to develop the solution.
* Include all artifacts typical of a professional (business) letter, e.g., subject line, date, greeting, signature, etc.

## Project Recommendation

Write a follow-up proposal to the letter of transmittal providing more details on how your project meets their organizational need(s). Again, the target audience is the same non-technical senior leadership from the *Letter of Transmittal*. Typically, this section is 2-3 pages; **write everything in the future tense.**

### Problem Summary

* Summarize the project.
* Describe the setting and why the project is needed.
* Briefly describe how the project meets the business’s (or organization’s) needs.
* Describe what will be delivered and achieved.

### Application Benefits

* Describe (in more detail than above) how the project meets the business’s (or organization’s) needs.
* Describe how the business (or organization) will benefit from implementing the proposed solution.

### Application Description

* Provide technical details on how the application will solve the problem.

### Data Description

* Identify the origin of the raw data.
* Describe the type (nominal, quantitative, etc.) and data structure.
* Identify dependent and independent variables.
* Describe any anomalies (e.g., outliers) and limitations.

### Objectives and Hypothesis

* Identify and describe desired outcomes of the project.
* If applicable, state a hypothesis.
* If applicable, state the desired prediction accuracy.

### Methodology

* Identify the methodology, e.g., waterfall, agile, etc., used to develop and implement the project.
* Describe why the chosen methodology is appropriate for the project.
* Provide an outline of the project methodology describing each phase, e.g., Design, Implementaion, etc.

### Funding Requirements

* Describe the project’s funding requirements, including environment, personnel, licensing, and tools.
* The funding amount should match the letter of transmittal.

### Data Precautions

* Identify any sensitive or protected data.
* If applicable, review the general guidelines for working with that data.
* If applicable, describe necessary precautions which will be taken.
* If either of the above is not applicable, explain why (public datasets, such as those from Kaggle.com, have no such restrictions).

### Developer’s Expertise

* Describe the developer’s (you) qualifications, e.g., academic training, professional expertise, experience, etc. Using future qualifications, such as your WGU degree in Computer Science, is acceptable.
* Relate the listed qualifications to the needs of the project.

# Part B: Project Proposal

The project proposal should target your client’s technically savvy IT (Information Technology) professional leadership. Use appropriate industry jargon and sufficient technical details to describe the proposed project and its application. Remember, you’re establishing the technical context for your project and what it will accomplish for the client. Typically, this section is 8 – 10 pages. **Write everything in the future tense.**

## Problem Statement

* Describe the problem.

## Customer Summary

* Describe the client (or customers).
* Describe why your proposed *application* (a *data product* in the task directions) will resolve the problem successfully.

## Existing System Analysis

* Describe (if any) what application(s) or tool(s) the client currently uses.
* Describe the shortcomings of this current technological environment, i.e., why your solution is needed.

## Data

* This section should include (where applicable) descriptions of:
  + The raw data set.
  + How data will be collected, processed, and managed throughout the application development life cycle: design, development, maintenance, or others.
  + How data anomalies, e.g., outliers, incomplete data, etc., will be handled.

## Project Methodology

* Describe an industry-standard methodology to be used to develop and (if applicable) deploy your application.
* Describe the planned development of your application in each phase of the methodology, e.g., analysis, design, etc.

## Project Outcomes

* Provide descriptions of all deliverables. For example:
  + The finished application.
  + A user guide.

## Implementation Plan

* Provide an outline of how the project will be implemented. This description might include the following:
  + General strategy.
  + Phases of the rollout.
  + Dependencies.
  + Details for testing and distribution.

## Evaluation Plan

* Describe the verification method(s) to be used at each stage of development.
* Describe the validation method to be used upon completion of the project.

## Resources and Costs

1. Itemize hardware and software costs.
2. Itemize estimated labor time and costs.
3. Itemize estimated environment costs of the application, e.g., deployment, hosting, maintenance, etc.

## Timeline and Milestones

* Provide a projected timeline, including start dates and end dates for each milestone (a table is acceptable).

# Part C: Application

Submitted Files:

* The project files do not require any links. All necessary files are located in the fish-weight-estimation.zip file. After unzipping the file, the folder will contain all project files, documentation, writeups, visuals, etc.
* The two files necessary to run the application are the main.py file and the fish\_data\_processed.csv file. The environment and additional software will be the responsibility of the user. Directions will be provided in the “User Guide” section of Part D.

Components of the Project:

* The project contains three visualizations that are located in this document, in the visuals folder, and also inside the application. The application is able to generate the three visuals.
* The three descriptive methods in the project include the histogram, the scatter plot, and the graph displaying the prediction error of the regression model.
* The one non-descriptive method in the project is the linear regression algorithm that is included in the project’s code in the main.py file.
* The application of machine learning is seen through the non-descriptive method. The linear regression model is trained using the data set in order to improve its ability to predict the weight of a fish using its dimensions.
* An interactive dashboard is provided through the console in the IDE. Using text, the user interface mimics a navigatable menu using commands such as “A”, “B”, and “end”. The user is able to interact with the machine learning model by inputting fish dimensions to receive a predicted fish weight.
* The user interface is user-friendly because the “User Guide” section from Part D explains 23 steps to installing and using the application. There are images provided in almost all 23 steps to provide further clarification. There are links provided that give information about the software used in situations requiring troubleshooting.
* The application is able to run using only two files: the csv file and the python file. Because of this, all operations are done locally and do not require a constant network connection. The offline nature of the application protects it from external attacks because it will never require any contact with an outside source after it has been installed on a device. Thus, the security of the application remains strong.

# Part D: Post-implementation Report

Create a post-implementation as outlined below. Provide sufficient detail so that a reader knowledgeable in computer science but unfamiliar with your project can understand what you have accomplished. Using examples and visualizations (including screenshots) beyond the three required is highly recommended. **Write everything in the past tense.**

## A Business (or Organization) Vision

Description of Company:

* TastyFish is a company that catches and sells fish for consumption. Due to laws and regulations for commercial fishing, the fishers are required to measure various lengths of each fish they catch. If the dimensions fall below a certain threshold, the fish must be released to ensure the younger population can breed.

The Problem:

* The problem was that due to the unstable motion of fishing boats, fishers were unable to weigh each fish individually. The sea’s waves prevented accurate weight measurements. The company’s previous method of weighing fish was letting it dry and placing it on a scale. This proved to be inefficient because the shelf life of the fish was lowered when waiting for it to dry and also during the weighing process.

How the Application Solved the Problem:

* The application solved the problem by using machine learning to predict the weight of each fish. A linear regression algorithm was applied to the machine learning model and by training the model on existing fish data, the model was able to learn how to predict the weight of a fish given its vertical length, diagonal length, cross length, height, and diagonal width.

Use of Application to Solve the Problem:

* A user can use the application to predict the weight of a fish without weighing it. If the user possesses the five dimensions of the fish, the user can input those dimensions into the trained linear regression model to predict the weight of the fish in grams. An example would be if the user inputted the string “23.2, 25.4, 30, 11.52, 4.02”, then the model would provide a predicted weight of 325.23 grams.

Screenshot of Application Console Output

Graphical user interface, text, application

Description automatically generated

## Datasets

The Raw and Processed Data

* The raw data is a csv file that contains seven columns. The columns consist of the fish species, weight, vertical length, diagonal length, cross length, height, and diagonal width.
* The processed data is a csv file that contains six columns. The columns are the same as the raw data except the first column for fish species is removed.

Processing of Raw Data

* The raw data was processed by removing the first column for fish species. Little processing was needed because the raw data was minimalistic and contained mostly necessary columns. The data in its raw form was already accessible to the algorithm but processing the raw data removed the need for the dataframe object to iterate over an unused index in the array. Thus, a miniscule amount of processing power was saved by removing the first index of the arrays in the csv file. The algorithm would have returned the same results if the raw data was used with the appropriate array indexes changed.

Examples of the Raw and Processed Data

* Below is the first three rows from the raw data file named “fish\_data\_raw.csv”
  + Species,Weight,Length1,Length2,Length3,Height,Width
  + Bream,242,23.2,25.4,30,11.52,4.02
  + Bream,290,24,26.3,31.2,12.48,4.3056
* Below is the first three rows from the processes data file named “fish\_data\_processed.csv”
  + Weight,Length1,Length2,Length3,Height,Width
  + 242,23.2,25.4,30,11.52,4.02
  + 290,24,26.3,31.2,12.48,4.3056

Access to Datasets:

* The original dataset is available for download from this link:
  + <https://www.kaggle.com/datasets/aungpyaeap/fish-market>
* The dataset contained a single csv file.
* The “fish\_data\_raw.csv” file is the exact dataset that was downloaded from the link.
* The “fish\_data\_processed.csv” file is the only dataset that was used throughout the project.

## Data Product Code

Review of Code Functionality:

* In the analysis and development of the application, the code was used to execute a variety of functions.
* Overall, the product does exactly as it is supposed to do. It takes in five numbers that represent various measurements of a fish and predicts the weight of the fish using a trained linear regression model.
* Because the raw data was preprocessed manually, the code never touched the raw data. Only the processed data was touched by the code. This is because the processing only involved the deletion of one column in a csv file. This saved the code from having to iterate over that one column, which is an insignificant saving of processing power. The processing of data allowed the code to use all columns of the data.
* The code provided visualizations of data using imported libraries. These visualizations were localized to the “visuals()” function in the “main.py” file. The project used the “matplotlib” library to allow all three visualizations to appear on the screen in their individual windows. These three visualizations were the histogram, scatter plot, and graph displaying the prediction error of the regression model. The “metrics” module from the “scikit-learn” library was used to create the prediction error graph. The “pandas” library was used to create the histogram and scatter plot.
* The code aided in the development of the descriptive methods because as mentioned earlier, the the histogram, scatter plot, and graph displaying the prediction error of the regression model, were all genereated using Python libraries. A descriptive method is anything that describes data. The histogram described the distribution of each variable in the csv file. The data showed that there were certain numbers more prevelant than others in each variable. The scatter plot showed the correlation of each variable in the csv file. As each variable’s measurement increased, the scatter plot showed a positive correlation with all other measurements. The diagonal across the scatter plot displayed the histogram data. The graph displaying the prediction error of the regression model showed the difference between the predicted weight values and the actual weight values. It showed that the extreme ends of the weight predictions were more prone to error than the predictions of weight values near 200 to 700 grams.
* The code aided in the development of the non-descriptive methods because the “scikit-learn” library provided many machine learning modules to train a linear regression model. The “pandas” library was used to create the data structure for the model. After that, the “scikit-learn” library had modules to create linear regression models, train the models with data, and provide predictions with new data.
* The data analysis done through the visuals and reviewing of the csv file helped created correlations between the independent variables that lead to the prediction of the dependent variable. Seeing that there was a positive correlation between the independent variables and the dependent variable, it helped choose the scatter plot because the data points allowed one to realize that as the independent variables increased, the dependent variable increased as well.
* The analysis of the data lead to the creation and improvement of the histogram. Because of the data reviews, it was seen that the model could predict values more accurately when there was more existing data numerically similar to an input.
* This leads to the creation and improvement of the graph that displays the prediction error of the regression model. Due to the histogram showing trends in the existing data, the prediction error followed the histogram. This means that as the variables increased in quantity around a certain numerical range, the prediction error was decreased. This proves the use of the prediction error in showing that the histogram data was relevant in prediciting the accuracy of the model.

Non-Descriptive Method:

* The non-descriptive method of the project was the machine learning algorithm. The project used a supervised learning algorithm. Instead of a classification algorithm, a regression algorithm was utilized because a linear regression algorithm was used to predict the weights.
* The linear regression model was developed by using a Python library called “scikit-learn”. It was also improved using methods within that Python library.
* This use is justified because the independent variables and dependent variable were all numerical. With the five independent variables and one dependent variable, a numerical prediction called for the use of the linear algorithm. Because it was numerical and not qualitative, a logistitic regression algorithm was not used as the prediction was not a binary choice.
* The linear regression model was trained using the dataset and also tested using the dataset. The “test\_size” variable was set to 0.33 which meant ~33% of the dataset was used to train the model and ~66% of the dataset was used to test the model. Using the existing variable data, the model was trained to predict the weight of the fish given five other measurements.
* The training process was appropriate because the default “test\_size” variable is 0.25 and in any machine learning model, you need data to both train and test the prediction algorithm. Thus, the process of the algorithm development followed a reasonable plan with adequate justification.

The “test\_size” Variable



Project Source Code:

* Because the project is a standalone app that is run in an IDE, the project only requires the “main.py” file and the “fish\_data\_processed.csv” file in order to provide full functionality. The submission of this project will include these files, which should be run in an IDE (ideally PyCharm).

The 2 Most Important Project Files



## Objective (or Hypothesis) Verification

Project Objective:

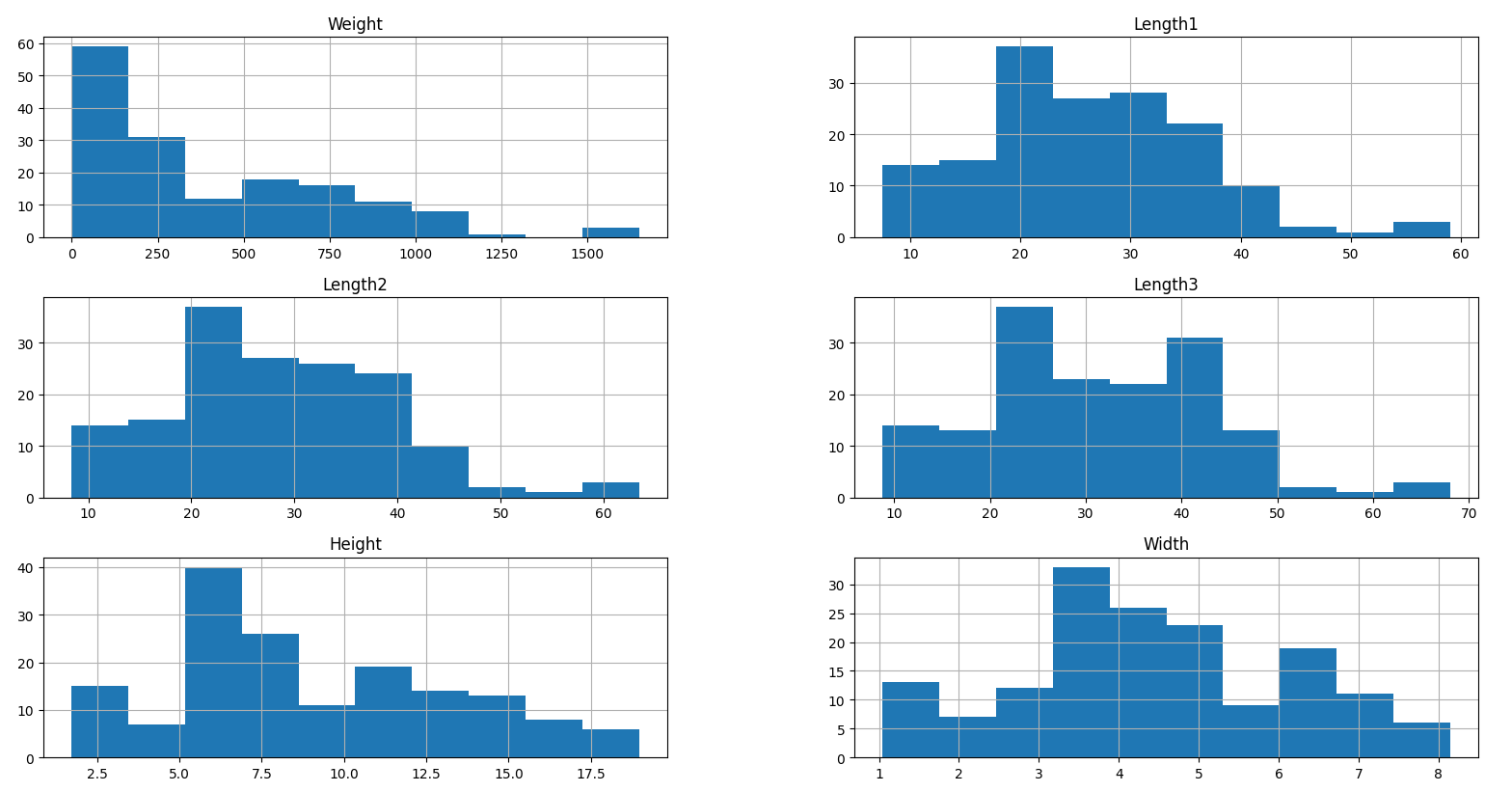
* The objective of the project was to create a machine learning model that would learn from existing fish data and be able to predict the weight of a fish given its five independent measurement variables.
* The objective was met because the resulting project was a linear regression model that used supervised learning to train from the fish dataset. As of April 5, 2023, the R-squared value of the linear regression model was 0.91. Thus, the linear regression model is able to predict the weight of a fish with some margin of error.
* “R-Squared (R² or the coefficient of determination) is a statistical measure in a regression model that determines the proportion of variance in the dependent variable that can be explained by the independent variable. In other words, r-squared shows how well the data fit the regression model (the goodness of fit)” (Taylor).

## Effective Visualization and Reporting

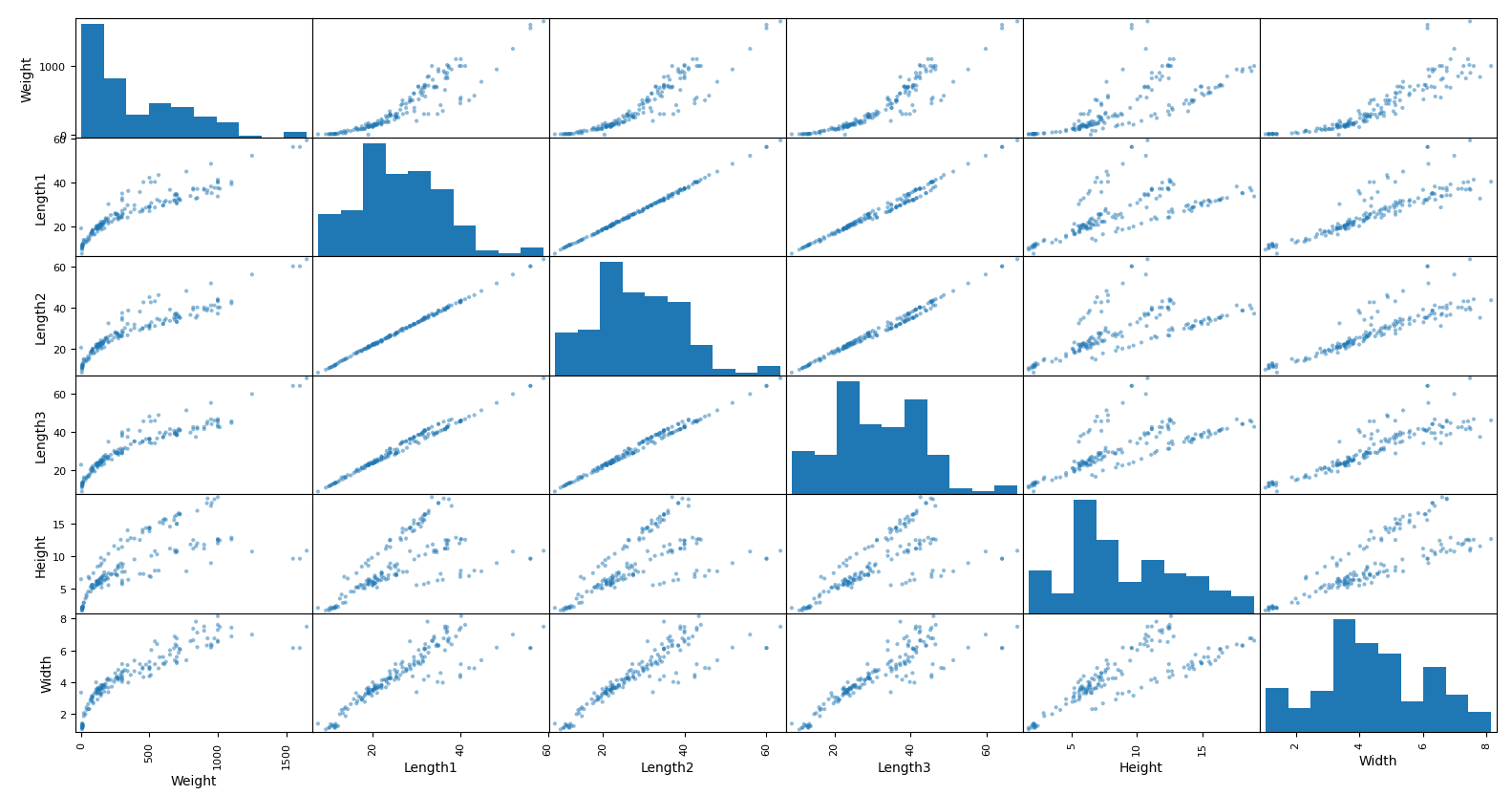
Non-Descriptive Development Process:

* The development process behind creating and training the linear regression model was supported by the visualizations and descriptive methods. At the most basic level, the development process by first exploring the data. The dataset was explored using various visualization tools to identify trends. Once it was observed that the weight and all five other measurements of the fish has a positive correlation, a linear regression model was seen as an appropriate approach. During the data exploration, it was clear that the five independent variables positively influenced the dependent variable.
* During the analysis of the data, it was clear that the linear regression model would be better suited when using measurements that were close to the mode (highest frequency) of the data. This helped when developing the non-descriptive method (the linear regression model), because it identified that altough the R-squared value might be 0.91, there was a higher prediction error in the upper and lower ranges of the weight predictions.
* The data summaries included the three visualizations and the R-squared value. These four data summaries provided a better insight into how the linear regression model would perform in certain situations.
* The three visualizations are shown below. For a clearer image, refer to the “visuals” folder in the project directory for the image files.

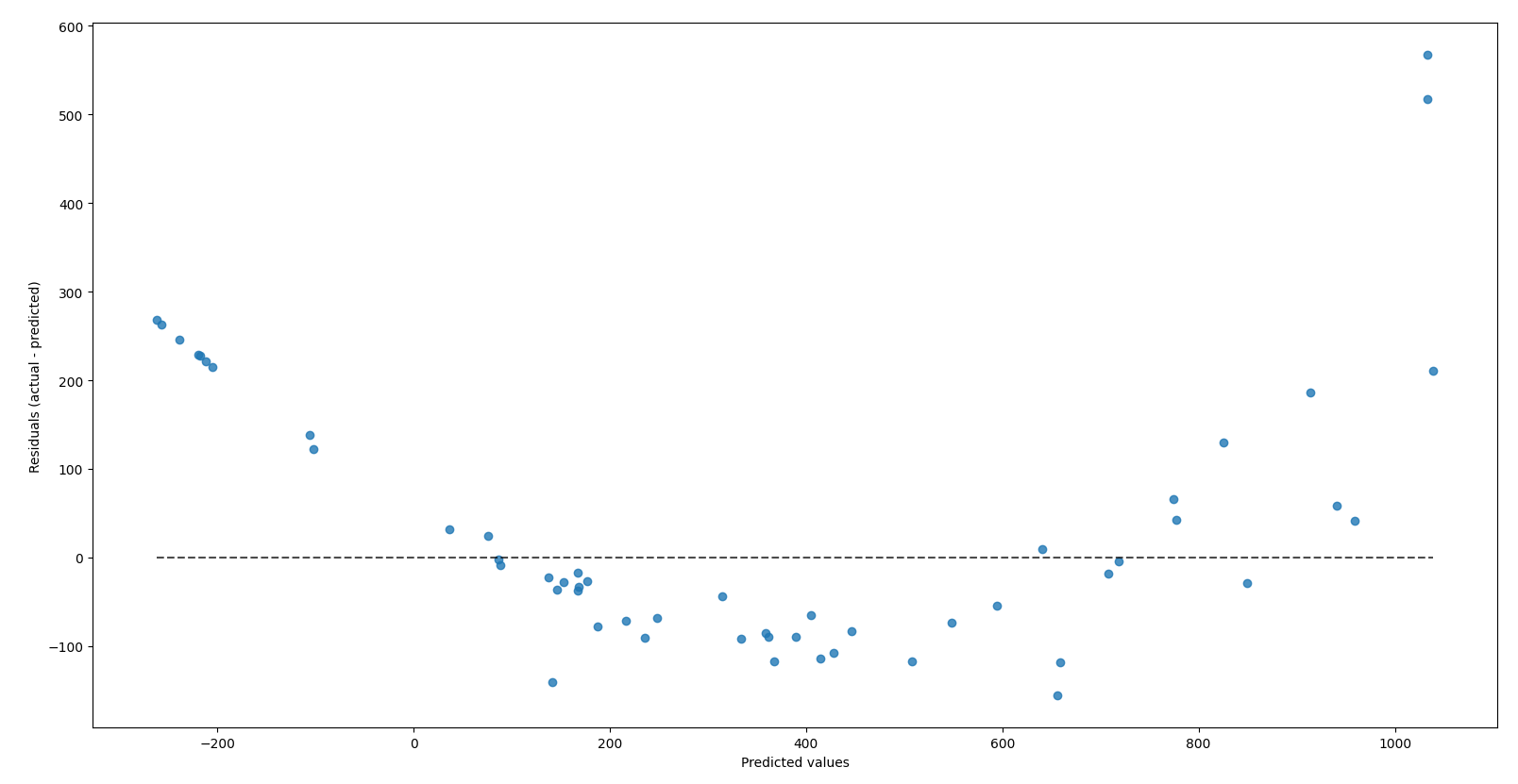
Histograms



Scatter Plots



Prediction Error of Regression Model



## Accuracy Analysis

The Model’s Accuracy Metric:

* The metric used to assess the accuracy of the linear regression model was a R-squared value.
* “R-Squared (R² or the coefficient of determination) is a statistical measure in a regression model that determines the proportion of variance in the dependent variable that can be explained by the independent variable. In other words, r-squared shows how well the data fit the regression model (the goodness of fit)” (Taylor).
* The R-squared value is a decimal number that ranges from 0.0 to 1.0. Our product returned a R-squared value of 0.91.
* Below is an example of the R-square value calculation in the project code. The calculation began with the non-descriptive method (linear regression model) predicting the weights of the fish using the test data of five measurements. Then, the predicted weight values were compared against the correct weight values. This variance was the R-squared value and illustrated the variance in the regression model versus the actual data.
* This metric was very accurate in proving the linear regression model’s accuracy because it calculated the variance of the true data against the predicted data. When dealing with numerical variables like in this project, it was the only way to display how accurate the regression model was.

Code Screenshot: Calculation of R-squared Value

Text, letter

Description automatically generated

## Application Testing

Testing Methods, Process, and Results:

* The application was tested only by the developer. The testing process included using a variety of valid and invalid inputs for the linear regression algorithm. In addition, the menu navigation in the console was also tested to account for various errors and situations.
* The results of the testing were signficant because it led to many changes in the code that improved the final product. The first tests involved inputting various combinations of the five numerical inputs for the linear regression algorithm. The outputs were compared to the actual data to identify similarities and to ensure that the final prediction was not outlandishly inaccurate.
* The testing of the application’s user interface also brought about many changes. By testing the application in the end user’s perspective. The graphical interface was altered to improve visiblity. The print statements were altered to inclue newline operators. If the linear regression algorithm was fed an invalid input, the appropriate error message was given. By testing the variety of inputs, the code was able to improve by including “try except” statements. These prevented errors from the algorithm receiving too few or too many numbers. In addition, if any of the inputs included a non-numerical value, a corresponding error was displayed.
* The use of the “A”, “B”, and “end” inputs were added after testing showed that it was not possible to maneuver between the reporting section and the calculation section of the application. This led to the improvement of the code where the aforementioned three strings allowed the user to navigate between the program’s different functionalities without having to terminate the program and start over again.
* There were various parts of the application that displayed error messages if the program was terminated while waiting for an input. To prevent this, a “ try except” statement included a “KeyboardInterrupt” error case where the program would quit instead of displaying an unnecessary error.

## Application Files

Location of Files:

* The 8 files mentioned below will all be located in the provided “zip” file. All files are located directly inside the directory.

Required Files:

* The project requires only two files to execute properly. Because the entire project and associated documentations are located inside the project folder, everything mention in this section can be accessed through the provided “zip” folder. The necessary files to execute the application are the following:
  + main.py (run this file in PyCharm IDE to access the console UI)
  + fish\_data\_processed.csv (place this file with “main.py” for regression model training)

Not Required Files:

* There are supplementary files that are not necessary to access the application. These files only serve to improve the viewer’s understanding of the project. They are not needed to execute the program. These files are the ones mentioned below:
  + “visuals” folder
  + task\_1\_writeup.docx
  + task\_2\_documentation.docx
  + fish\_data\_raw.csv
  + README.md

## User Guide

Installation and Use Guide:

* If any of the following software is already installed and configured, you can skip that step.
* The following directions apply to a Windows 10 machine.
  + - 1. Download the fish-weight-estimation.zip file onto your computer.



* + - 1. Unzip the fish-weight-estimation.zip file. You can use this link (<https://www.7-zip.org>) to install a file unarchiver tool to unzip it. Refer to this link (<https://www.7-zip.org/faq.html>) if any issues arise during the installation of 7-Zip.

Shape

Description automatically generated with medium confidence

* + - 1. Download and install PyCharm Community Edition from this link (<https://www.jetbrains.com/pycharm/download>). Refer to this link (<https://www.jetbrains.com/help/pycharm/installation-guide.html>) if any issues arise during installation of PyCharm.

Graphical user interface, text, application

Description automatically generated

* + - 1. Download and install Python from this link (<https://www.python.org/downloads>). Refer to this link (<https://wiki.python.org/moin/BeginnersGuide/Download>) if any issues arise during installation of Python.

Graphical user interface

Description automatically generated

* + - 1. Open PyCharm.



* + - 1. In PyCharm, go to File -> Open . . . -> select the unzipped fish-weight-estimation folder. Click “OK”.

Graphical user interface, application

Description automatically generated

* + - 1. With the project opened in PyCharm, open the main.py file in the IDE.

Graphical user interface, text, application

Description automatically generated

* + - 1. In PyCharm, configure the Python interpreter for this project using the Python installation from step 4. Use this link (<https://www.jetbrains.com/help/pycharm/configuring-python-interpreter.html>) if any issues arise during interpreter configuration.

Graphical user interface, table

Description automatically generated

* + - 1. In the main.py file, ensure lines 1 – 4 are not showing errors. If they are showing errors, it means that the required Python libraries are not installed. On each error line, right-click the problematic import and click “Show Context Actions”. Choose the option to install the corresponding Python library. Repeat this for all four lines of imports

Graphical user interface, text, application

Description automatically generated

* + - 1. After the libraries are installed, run the main.py file.

Text

Description automatically generated with medium confidence

* + - 1. The console should appear now. In the console, type in “B” and press ENTER.

Graphical user interface

Description automatically generated with medium confidence

* + - 1. The console will generate the R-squared value of the linear regression model.

A picture containing chart

Description automatically generated

* + - 1. Three windows will appear showing the prediction error of the regression model, a histogram, and a scatter plot. These three visuals will be discussed in this document. Enlarged versions are available in the “visuals” folder and also in the “Effective Visualization and Reporting” section of Part D.

Chart

Description automatically generatedChart

Description automatically generatedChart, scatter chart

Description automatically generated

* + - 1. After closing all three windows, the console will automatically return to the main menu.

Graphical user interface, text, application

Description automatically generated

* + - 1. In the console, type in “A” and press ENTER.

Graphical user interface, text, application, Word

Description automatically generated

* + - 1. The console will generate the menu to use the trained linear regression model that predicts the weight of a fish given five numers (the vertical length, diagonal length, cross length, height, and diagonal width).

Graphical user interface, text, application, email

Description automatically generated

* + - 1. In the console, type in “23.2, 25.4, 30, 11.52, 4.02” and press ENTER to test the model.

Graphical user interface, text, application

Description automatically generated

* + - 1. The output will return the predicted weight of the fish in grams.

Graphical user interface, text, application, email

Description automatically generated

* + - 1. Use this menu to repeatedly test variations of fish measurements.
      2. When want to close the program, type “end” in the console and press ENTER.

Text

Description automatically generated

* + - 1. Again, in the console type in “end” and press ENTER.

Graphical user interface, text, application, Word

Description automatically generated

* + - 1. The program is now terminated.
      2. Return to step 10 to start the program again.

## Summation of Learning Experience

Application of Previous Knowledge and Experience:

* This project was made easier because of my previous experience in Western Governors University’s (WGU) coursework. During my experience earning my Bachelor of Science in Computer Science degree, I learned many hard and soft skills. I learned how to better use various IDEs like PyCharm and IntelliJ. The practical experience using these IDEs in previous projects brought me the knowledge of various aspects. Things like configuring the project’s interpreter, installing the appropriate compilers, and creating a proper project structure were all thing that I learned through WGU.
* The use of version control using Git was also integral in this projects development. Ensuring that there were safe backups of my work in the cloud brought an extra layer of confidence to my work. In addition, the ability to rollback any changes brough comfort when changing and improving my code. WGU helped me develop my skills in Git because of the many projects that need to be developed during the coursework. In addition, working with the virtual machines in previous projects required me to be competent in Git.
* Previous projects that included developing code and solving a business need helped me understand the steps I needed to take to solve a solution with software. Using Python in this project was easy since I already had experience using Java in courses like “Software I” and “Software II”.

Application of New Knowledge:

* Additional knowledge was definitely needed to complete this project. The topic of machine learning was explored in my previous courses but never practically used to the extent of this project.
* I needed to learn various topics like supervised learning, unsupervised learning, reinforced learning, regression algorithms, and classification algorithms. These topics of machine learning were important in understanding the final product and how I was going to achieve it.
* I also needed to learn how to use various Python libraries such as the “scikit-learn” library in order to implement linear regression into my data. I needed to learn how to use the “pandas” library to implement “dataframe” structures that help csv data. I also needed to learn the “matplotlib” library to be able to create visuals from my data and model.
* I learned more about the application of visuals in data sets and regression models to create the visuals.
* I needed to expand my knowledge about the”try except” statement and the keyboard input error when creating my user interface.
* I needed to learn about descriptive methods and non-descriptive methods so that I could properly implement a way to describe my data and also infer new knowledge from it.

Project Contribution to Lifelong Learning:

* My definition of lifelong learning is “the continual pursuit of new knowledge that will improve one’s expertise in a subject matter”.
* When I analyze this experience for its contributions to my lifelong learning, I believe it has opened up a new pathway for me to develop a better understanding of machine learning. Prior to this project, I had no ambition for machine learning and the concept was very foreign to me. However, this project has caused me to become interested in the topic. This will lead me to pursue more opportunities and learning moments where I can expand my knowledge on machine learning.

# Part E: Sources

Taylor, Sebastion. “R-Squared**”.** Corporate Finance Institute, 4 Mar. 2023,

<https://corporatefinanceinstitute.com/resources/data-science/r-squared>.