# Applications of Mathematics in Machine Learning

By

Alinda Rolland Mucunguzi (ralinda@aims.edu.gh)

June 2024

*AN ESSAY PRESENTED TO AIMS-GHANA IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF MASTER OF SCIENCE IN MATHEMATICS*

AIMS African Institute for Mathematical Sciences GHANA

# DECLARATION

This work was carried out at AIMS-Ghana in partial fulfilment of the requirements for a Master of Science Degree.

I hereby declare that except where due acknowledgement is made, this work has never been presented wholly or in part for the award of a degree at AIMS-Ghana or any other University.

Student: Alinda Mucunguzi Rolland

Scan your signature

Supervisor: Dr. Laure Gouba

# ACKNOWLEDGEMENTS

I convey my utmost gratitude to The Pre-eminent God, the giver of all wisdom, knowledge and understanding for His endless unconditional love and blessings.

I am deeply grateful to The African Institute For Mathematical Sciences(Ghana) for the fully funded opportunity to study mathematical sciences at an advanced level. In addition to this, sincere appreciation goes to all of the AIMS-Ghana founders, funders and administration who made all this masters journey possible.

I extend heartfelt appreciation to my project supervisor, Dr. Laure Gouba, her due guidance, support and encouragement were of vital role in this endeavor. Furthermore, her expertise was indispensable in shaping the direction of this research.

Special thanks to my dedicated tutor, Dr. Sara Abdelazeem Hassan Abaass, her constant guidance and insights in the research process were of huge impact.

Lastly, endless thanks to my family and friends for the steadfast support and encouragement in this journey. Their unreserved love has sustained me all the way.

# DEDICATION

I dedicate this piece of work to my father, Mr. Patrick Musambya, my mother, Mrs. Grace Birungi, and my brother, Mr. Don Kasaija. Their steadfast support, motivation and love are always my strong driving force.

Lastly, I would like to dedicate this piece work to all the scholars and researchers who have come before me whose work has laid a firm ground for my own. May this work contribute to the collective knowledge and inspire future generations.

# Abstract

This work explores integral applications of mathematics in development and usage of machine learning algorithms with a strong focus on linear regression models. The study begins with an overview of machine learning, highlighting its fundamental concepts and significance in various fields. Subsequently, we look at the mathematical foundations essential for machine learning, including; linear algebra, probability theory, calculus, optimization, statistics, graph theory and geometry. For a concrete illustration of the applications of mathematics in machine learning, this work employs simple and multiple linear regression models using data that is about pH of pure water. Through these examples, the thesis demonstrates how mathematical techniques are applied in formulating, estimating and evaluating linear regression models. Key processes such as least squares estimation and statistical inference are highlighted to show their critical application in parameter estimation and model validation. The findings underscore the importance of the mathematical rigor in ensuring accuracy and interpretability of machine learning models. By bridging theoretical principles with practical applications, this thesis emphasizes the applications of mathematics as a foundational tool that drives the machine learning process. The study concludes that as machine learning continues to advance, the integration of sophisticated mathematical methods will be pivotal in the development of more powerful and effective machine learning algorithms.

# Contents

# 1. Introduction

Machine learning involves designing algorithms that automatically extract useful information such patterns and structures from data without explicit programming. When we talk about machine learning, three concepts come into play; data, a model and learning. Machine learning is a data driven process and machine learning models are trained using data that is relevant for the task at hand. Learning takes place when a machine learning model finds useful structures and patterns in the data through optimization of model parameters. A machine learning process is considered successful when the trained model can correctly generalize and make accurate predictions on data it has not seen before.

Systems developed on a machine learning foundation have registered enormous success in different domains of our lives such as agriculture, health, education, trade, transport and more. It is notable that behind every machine learning performance, there are principles of mathematics in play. It is these principles of mathematics that pave way to understanding the fundamental principles upon which machine learning algorithms thrive from the most basic of algorithms to the highly sophisticated ones. Acquaintance with these deeply embedded principles of machine learning and how they are powered by mathematics is key in; understanding how the existing algorithms work, developing and innovating new ones, debugging code when things go wrong and understanding what we can and cannot do with currently existing machine learning tools.

This work highlights what machine learning embodies, demonstrates the connection between machine learning and mathematics, then shows how some mathematical topics link to machine learning. Furthermore, we use data on the variation of pH of pure water with different predictors to develop two linear regression models and we use these models to demonstrate how mathematics is applied in linear regression machine learning tasks.

## 1.1 The Problem Statement

As an emerging, exciting and rapidly evolving technology of the modern era of computers and artificial intelligence, machine learning is playing a vital role in shaping our present times and potentially the future. Machine learning models are powered by machine learning algorithms that enable computerized systems to learn from given data and this learning is independent of explicit programming. These machine learning algorithms come to life with utilization of various mathematical principles and frameworks. An understanding of how we apply the principles and frameworks of mathematics in the development of machine learning algorithms and subsequently the process of machine learning is at the core of this research work.

### 1.1.1 Aim.

This research work is aimed at providing a firm understanding of how mathematics is applied in the process of machine learning.

### 1.1.2 Objectives.

The objectives of this research work are:

- To give a precise description of what machine learning entails.
- To briefly demonstrate how some topics in mathematics are applied in machine learning.
- To develop a simple and a multiple linear regression machine learning model.
- To demonstrate how mathematics is applied in linear regression machine learning tasks.

## 1.2 Significance of the Research

Machine learning and artificial intelligence are rapidly changing our world with applications in transport such as automated vehicles, e-commerce such as AI-powered chatbots, health such as detection of malignant tumors, education such as simulation of biochemical processes and more. Artificially intelligent systems use machine learning algorithms and models, the better the algorithm, the more powerful is the artificially intelligent system at executing tasks provided using the underlying model. In the development and usage of machine learning algorithms and models, different tools and concepts from mathematics are integrated. This research work demonstrates how the tools and concepts from mathematics are applied in machine learning and inherently stresses why a good understanding of mathematics is important in the development, use and improvement of intelligent systems that are powered by machine learning algorithms and models.

## 1.3 Scope of the Research

In this research work, we look at the applications of mathematics in machine learning. We start with a general overview of key ideas in machine learning, after which, we look briefly at some key topics in mathematics that play a vital role in machine learning. We crown the work with simple and multiple linear regression models where we look at examples of linear regression tasks along with the mathematics involved in these linear regression machine learning endeavors. It is notable that sophisticated machine learning models such as gradient descent and deep learning neural networks are looked at in concisely brief details in this work.

## 1.4 Research Structure

This research work is organized along six chapters. Chapter 1 introduces and lays ground for the research work. Chapter 2 of this work gives an overview of the key ideas in machine learning. Chapter 3 gives a highlight of the mathematics applied in machine learning. Chapter 4 is about simple linear regression and the mathematics applied in a simple linear regression task. Chapter 5 is about multiple linear regression and the mathematics applied in a multiple linear regression task. Chapter 6 crowns the work with a conclusion on results from the linear regression models and applications of mathematics in machine learning.

# 2. Overview of Machine Learning

## 2.1 Machine Learning

Machine learning refers to the development and use of computer algorithms that power models capable of learning and adapting from data without being explicitly given instructions. Machine learning is a branch of computer science and artificial intelligence that mainly focuses on using algorithms to detect patterns in given datasets with gradual improvements towards accuracy depicting the way humans learn. A computer program is said to learn from a given experience with respect to some particular task and its performance, if the performance of the task improves with experience [11]. The novel role of machine learning lies in developing and implementing algorithms that power models which facilitate prediction and decision making processes.

**An example of a machine learning performance.**
When you give 500 images of cats and dogs to the computer and you would like the computer to distinguish cats and dogs, then, the machine learning algorithm figures out a common patterns and creates a way of distinguishing cats from dogs using their features. Over time, as the algorithm processes more and more images, it gets better and better at distinguishing cats from dogs such that even when provided with an image it has never seen before, it is able to categorize it as a cat or dog.

**2.1.1 Remark.** Machine learning involves extracting valuable information from given datasets. This is a huge part of our daily life where websites such as Twitter, Amazon, Instagram, You-tube and Facebook employ robust machine learning algorithms to optimize and automate their online operations using data on user activity.

A distinction between traditional programming and machine learning is shown in the Table 2.1;

| **Traditional Programming** | **Machine Learning** |
|---|---|
| Predefined instructions to perform a task are provided to a computer. | A dataset and a task to perform are provided to a computer. |
| Rule based and fully dependent on the intelligence of a programmer. | Can find patterns in robust datasets that could be difficult for human to discover. |

Table 2.1: A comparison between traditional programming and machine learning.

### 2.1.2 Why Machine Learning.

Machine learning is an improvement in computational intelligence that offers us the ability to derive meaningful insights from data, make predictions and automate complicated decision making processes in different aspects of our lives. Machine learning is vital due to its enormous ability in obtaining valuable configurations from datasets. This task is inherently challenging when we employ simple rule-based systems that involve hand coding. Machine learning employs techniques from mathematics and related computational resources to create models that learn from experience, adapt to new data provided by a user and generalize patterns after a successful training process. Machine learning helps us handle problems that are very complicated and else poorly addressed through hand coded programming, for example, in facial recognition systems the task would require hand coding details of each individual person's facial details and yet in machine learning, an algorithm extracts pixels from a large dataset containing images of human faces and comes up with a good set of rules to describe and distinguish all human faces in a digital format.

## 2.2 Real-world Applications of Machine Learning

Some real-world applications of machine learning include:

**Computer vision.** Here, computers are able to generate meaningful information from visual inputs such as images and videos provided by the user and then they perform appropriate actions. Computer vision has applications in health such as in radiology, automotive industry like in self-driving cars and more [30].

**Speech recognition.** This involves translation of human speech into a written format with the use of natural language processing technology. Speech recognition enables computers to process, interpret and comprehend human language in different formats like written, spoken or even scribbled. A good example of a natural language processing model is ChatGPT.

**Customer service.** Chatbots in a form of virtual assistants on different websites and social media platforms engage in a very productive way with customers. Chatbots answer frequently asked questions, provide personalized advice, suggest commodities to consumers and the like.

**Automated stock trading.** This involves the use of machine learning algorithms to optimize stock portfolios. World over, artificially intelligent trading platforms make thousands to millions of trades per day with no human involvement [21].

**Recommendation engines.** These use data about the past consumer behavior and discover patterns that can be used to come up with more effective selling and advertisement strategies. Recommendation engines are used by online retailers to make desirable recommendations of products to customers.

**Fraud detection.** Banking and related financial institutions use tools built with aid of machine learning to identify potential fraudulent transactions. A great technique here is anomaly detection where financial institutions can identify transactions that look atypical of a particular account holder and deserve investigation before a transaction is made.

**Robotic process automation.** This is software robotics. Robotic process automation employs the use of artificially intelligent automation technologies to carry out repetitive tasks that could otherwise be manually performed by humans and else at times could be dangerous such as handling radioactive materials [24].

## 2.3 The Process of Machine Learning

We would like to address the question, *"How does machine learning work?"* The process of machine learning involves the following stages [27]:

1. **Data collection:** This is the first stage in machines learning and it is important to know that machines learn from data we avail them with. Here, structured or unstructured data that is relevant to a problem at hand is obtained from a credible source. The quantity and quality of collected data greatly affects the efficiency of the subsequent machine learning model. The data collected should be comprehensive, relevant and representative of the problem at hand. Furthermore, data privacy, ethical concerns and regulatory compliance should be adhered to in order to maintain integrity and trustworthiness.

2. **Data preprocessing:** This step involves cleaning and preparing collected data to make it appropriate for analysis. Here, we handle outliers, missing values, inconsistencies and noise within a dataset. We employ techniques such as normalization and feature scaling with an intent of improving the quality of data and promoting its consistency. The end goal of this step is to optimize the data for model training such that we enhance the accuracy, efficiency and interpretability of the resulting machine learning model.

3. **Choosing a model:** When choosing a model, we select an algorithm or a combination of algorithms that can capture the underlying patterns within the preprocessed data. This step requires us to have a solid understanding of both the problem at hand and the characteristics of the machine learning models. The selection process involves a good balance of model complexity, interpretability and performance metric tailored to a specific task. Other considerations when selecting a machine learning model include; computational resources available, ease of implementation of the model and model scalability.

4. **Training:** This is the most important phase in any machine learning process within which models evolve and refine their predictive power. Training is an iterative process that involves optimizing model parameters to reduce disparity between predicted outcomes and ground truths. If we take a case of training a model that employs gradient descent algorithm, the model will iteratively update its internal parameters in the direction that minimizes a predefined loss function and convergence of this optimization signifies attainment of a model that effectively captures the hidden pattern in the training dataset. Factors such as quality of the training dataset, representativeness of the training dataset, choice of algorithms and optimization strategy employed affect the efficacy of the training phase. Here, care should be taken to avoid over-fitting and under-fitting of the trained model.

5. **Model evaluation:** During model evaluation, performance and efficiency of the trained model are assessed using evaluation metrics and validation techniques to ascertain the predictive power, accuracy, robustness and generalisation capacity of the model [8]. We usually begin by testing the model using a validation dataset that the model has not seen during training to check how it performs on generalization. Other measures of performance include the following metrics; accuracy, confusion matrix, precision, recall, $F1$ score, mean squared error, mean absolute error, $R^2$ value and precision-recall curve.

6. **Hyperparameter tuning and optimization:** Hyperparameters are parameters set before the learning process begins. These dictate the overall behavior and performance of the machine learning algorithm and consequently the resulting model [22]. They include; the learning rate and batch size in gradient descent, regularization parameter that controls degree of over-fitting, number of hidden layers in neural networks, number of trees and tree depth in random forest, kernel parameters in support vector machine, number of epochs and others. Our intension at this stage is to examine whether our model accuracy can be enhanced in any way once we have constructed and tested it.This is accomplished by fine-tuning where small adjustments in the hyperparameters are made to achieve the best model performance.

7. **Prediction and deployment:** The first aspect of this stage involves the trained model making predictions on a new dataset. Once predictions are generated, they are used for the intended purpose, for example, in healthcare, the predictions from a diagnostic model can assist in making informed treatment decisions. Deployment of the model requires care over model scalability, reliability and security. The model should be integrated into existing workflows in a way that ensures minimal disruptions to operations [2]. Moreover, other considerations such as privacy and ethical implications need to be addressed throughout the deployment process to ensure reliable and ethical use of machine learning technologies.

## 2.4 Types of Machine Learning

We have three main types of machine learning, that is; supervised, unsupervised and reinforcement machine learning.

### 2.4.1 Supervised Machine Learning.

A supervised machine learning scenario is depicted by the concept of a teacher whose main task is to give an agent an exact measure of its error [25]. In supervised machine learning, a model is trained using a labeled dataset. The model learns a mapping between specific features and their labels.

**Assumption:**
Given a dataset, $(x_i, y_i)$, there exists a unique hidden mapping

$$f: \; X \longrightarrow Y.$$

Thus, in a supervised learning situation, given a training dataset, $(x_i, y_i)$, with $i \in \{1, ......, n\}$, we find $\hat{f}: \; X \longrightarrow Y$ which is a good approximation of $f$.

**Regression Versus Classification**
In machine learning, we have two kinds of variables, that is, qualitative variables which are also known as discrete or categorical variables and quantitative variables which are also known as continuous variables. Tasks that utilize data with qualitative variables such as deciding whether a gender is male or female, classifying whether a tumor is malignant or benign, filtering emails as spam or not are termed as classification problems whereas those that utilize data with quantitative

variables such as prices of stocks in a market, size of fish in a pond are termed as regression problems [17]. Common algorithms for these tasks are highlighted in Figure 2.1

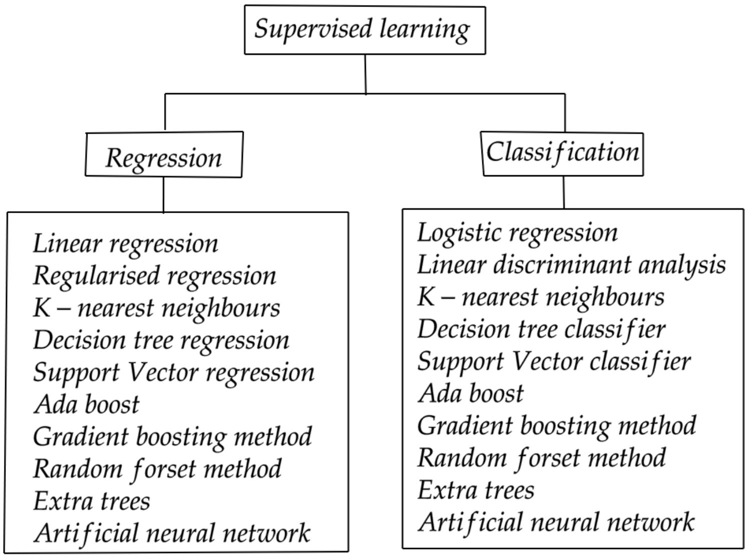

Figure 2.1: Algorithms for regression and classification tasks.

Supervised learning involves both regression and classification tasks. Common applications of supervised learning include pattern recognition, spam detection, natural language processing, automatic image classification, sentiment analysis and automatic sequence processing.

### 2.4.2 Unsupervised Machine Learning.

During unsupervised machine learning, an algorithm is tasked with discovering patterns within a dataset which has no labels. Unsupervised learning is useful when it is essential to learn how a set of elements can be categorized according to similarities between them. Common unsupervised machine learning tasks include clustering, dimensionality reduction and density estimation [29]. The common unsupervised machine learning applications of include recommendation systems, anomaly detection, image clustering and text clustering.

### 2.4.3 Reinforcement Machine Learning.

Reinforcement machine learning is a science of decision making that combines machine learning and optimal control oriented to learning an optimal behavior in a dynamic environment in order to maximize reward. Reinforcement learning is based on feedback received through trial and error as an agent explores and interacts with the environment. Reinforcement learning algorithms aim at an optimal policy that can maximize the cumulative reward obtained by the agent over time [26]. Some reinforcement learning algorithms include Markov decision process, dynamic programming and value iteration.

# 3. Mathematics in Machine Learning

## 3.1 Why the Interest in the Link Between Mathematics and Machine Learning?

The interest in the connection between machine learning and mathematics follows the machine learning process in Section 2.3 within whose stages different mathematical tools and principles are applied. Some of the reasons for the interest in the connection between mathematics and machine learning this are highlighted below;

- Mathematics provides a theoretical foundation upon which machine learning algorithms are built. Concepts from linear algebra, calculus, optimization, statistics, probability theory, graph theory and geometry are essential for understanding the inner operations of algorithms.
- Selecting and customizing a given model. Mathematics plays a vital role in choosing a model with an algorithm that can handle a unique problem at hand.
- Debugging code when things go wrong. Mathematics helps us understand errors and diagnose them effectively. Here, it is like being a detective in the world of code.
- Knowledge of mathematics enables us effectively interpret the predictions made by a given model that we choose to implement.
- Mathematics offers us rigorous tools for analyzing the performance, behavior and limitations of machine learning models. Through mathematical analysis, we gain insights into why certain algorithms work well in specific scenarios.
- Innovation and research. Creating new machine learning algorithms and improving existing ones require a great deal of mathematical insights and principles.

In subsequent sections of this chapter, we look briefly at different areas of mathematics and their applications in machine learning;

## 3.2 Linear Algebra

Linear algebra deals with the study of vectors, vector spaces and mappings that are required to perform linear transformations between given vector spaces. Linear algebra provides tools needed to effectively represent, modify, draw insights and improve machine learning models. A lot of potential in machine learning is put forth using linear algebra objects such as vectors, tensors, matrices and operations like addition, multiplication, inversion, transposition and decomposition of matrices [4].

### 3.2.1 An Understanding of Linear Algebra.

Various tools are unveiled to us through the knowledge of linear algebra;

1. Vectors: Vectors are elements of a vector space. Vectors are generally objects with magnitude and direction. In the light of machine learning we use vectors to represent data points and features of data. Addition, subtraction and scalar multiplication are all possible operations with vectors.

2. Matrices and tensors: Matrices are rectangular arrays of objects in rows and columns. Tensors are multidimensional arrays of data. We organize tabular data by way of matrices or tensors and various operations come to life when we organize data using matrices or tensors. Matrices can undergo scaling, addition, rotation, translation and other operations. Tensors can undergo tensor contraction, reshaping, broadcasting and other operations.

3. Vector spaces: A vector space can be visualized as a set of vectors along with vector addition and scalar multiplication satisfying a given set of axioms. Vectors spaces provide us a foundation for understanding characteristics and connections between vectors.

4. Linear transformations: These transfer vectors between vector spaces taking into account the underlying properties of the vector spaces. In machine learning, linear transformations give us ability to reshape and alter data in appropriate ways.

### 3.2.2 Linear Algebra and Machine Learning Algorithms.

Some algorithms in machine learning that utilize various principles from linear algebra include;

- **Linear regression**. We employ linear regression in creating a linear connection between a predictor and a target variables. For example, matrix operations are employed in solving systems of linear equations which enables determination of ideal coefficients that minimize error and thus allow us obtain the best linear fit for data.

- **Principle components analysis(PCA)**. In machine learning, principle component analysis is used to decrease dimensionality of high-dimensional datasets. Using PCA, we can transform a high dimensional feature space into a lower dimensional one while retaining the most relevant aspects of a dataset. Here, we use the analysis of eigenvalues and eigenvectors to dissect a given covariance matrix.

- **Support vector machine(SVM)**. For the determination of an ideal hyperplane that categorizes data points into multiple classes, support vector machines heavily rely on linear algebra. SVMs are able to work with complicated decision boundaries and classify new instances by representing data points as vectors and employing dot products coupled with matrix operations for their classification tasks. The Figure 3.1 shows an example of a support vector machine classifier for two classes.

- **Neural networks**. At the core of deep learning are neural networks that heavily employ linear algebra computations. Matrix multiplication and activation functions are employed in forward and backward propagation in neural networks to handle weights and biases in the network.

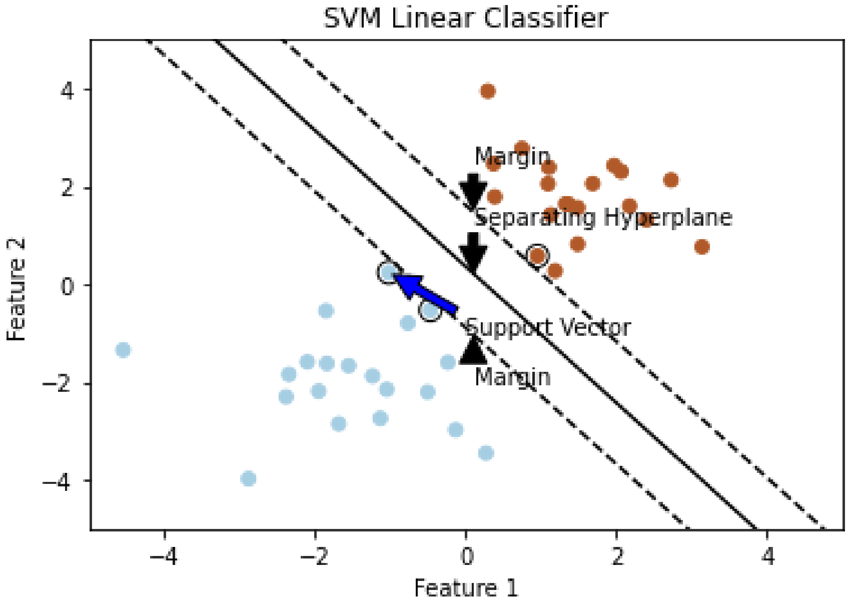

Figure 3.1: Shows a separating linear classifier for two data classes.

## 3.3 Calculus

Calculus forms a mathematical foundation for several machine learning algorithms and models. Calculus has two branches, that is; integral calculus and differential calculus. Integral calculus focuses on the concept of integration, which is essentially the process of finding the accumulated total of quantities. Differential calculus puts emphasis on creating an understanding of rates of change and steepness of curves [5]. Machine learning models that employ vast concepts from calculus include;

- **Gradient descent**. This is a first order optimization algorithm employed when finding the minimum point of a differentiable function. Gradient descent algorithm repeatedly changes the parameter of a function being optimized in a direction opposite to the slope of the function and continuously reduces the value of the function. The gradient descent algorithm iteratively takes steps in the direction opposite to that of the gradient until convergence. The update rule for gradient descent is:

$$x_{i+1} = x_i - \eta((\nabla f)(x_i))^T, \tag{3.3.1}$$

  with $\eta$ the step size, $x_i$ a data point and $\nabla f$ the gradient vector.

- **Convolutional neural networks**. Here, we observe calculus in deep learning. Calculus enables convolution and pooling operations in covolutional neural networks. During convolution, we apply filters to extract meaningful information from data. During pooling, we employ calculus to decrease feature space dimensionality with minimal information loss. The Figure 3.2 shows an illustration of what happen in these kinds of neural networks. These convolutional neural networks are employed in computer vision to perform certain tasks such as image recognition.

- **Recurrent neural networks**. This is still calculus in deep learning. Recurrent neural networks are used in the analysis of sequential data. These networks employ gradients to learn over time enabling them detect patterns and make predictions about sequences.

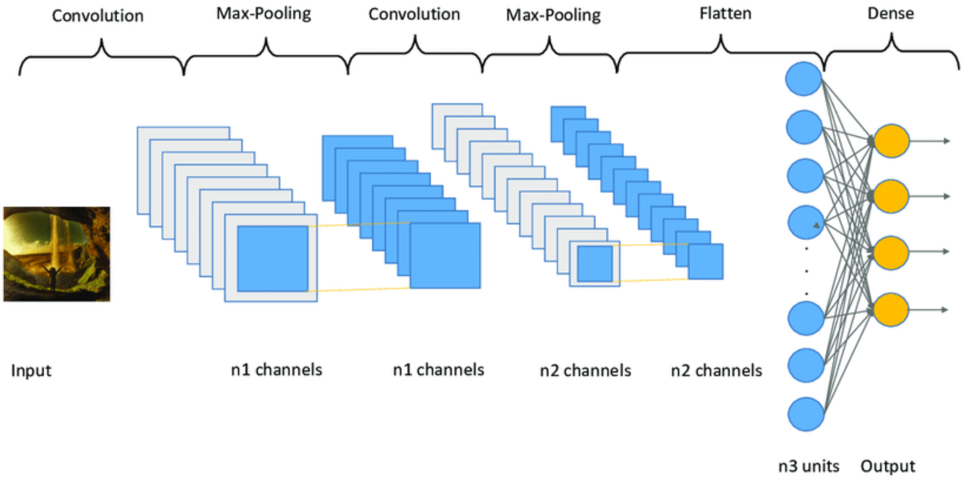

Figure 3.2: Shows a vanilla convolutional neural network representation [10].

### 3.3.1 Calculus and Regularization.

Regularization techniques are used to prevent the model from over-fitting a given dataset. During regularization, extra terms are added to a loss function. The extra terms serve as penalties that encourage the model to find solutions that are not too complicated and can easily generalize on new datasets. For $\lambda$, $\lambda_1$, $\lambda_2$ as scalars and $\beta_i$ as model coefficients, some regularization techniques are;

- Lasso regularization with a penalty term: $\lambda \sum_{i=1}^{n} |\beta_i|$.
- Ridge regularization with a penalty term: $\lambda \sum_{i=1}^{n} {\beta_i}^2$
- Elastic net regularization with a penalty term: $\lambda_1 \sum_{i=1}^{n} |\beta_i| + \lambda_2 \sum_{i=1}^{n} {\beta_i}^2$

## 3.4 Optimization

Optimization is the process of finding the best solution from a set of feasible solutions. There are two kinds of optimization, continuous optimization and combinatorial optimization. When we use computers to implement machine learning algorithms, then, inevitably mathematical formulations are expressed as numerical optimization methods. In machine learning, the process of optimization involves obtaining an ideal set of model parameters that minimize a given cost function. By convention, most machine learning optimization problems are handled as minimization problems.

### 3.4.1 Cost Functions.

A cost function calculates the discrepancy between the predicted output values and the actual output values. In an optimization lens, training a machine learning model boils down to simply finding a good set of model parameters. If we have $n$ data points, $y_i$ as the actual value and $\hat{y}_i$ as the predicted value, some common cost functions can be defined as;

- **Mean squared error(MSE).** This is usually used in regression tasks,

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2.$$

- **Mean absolute error(MAE).** This is also used in regression tasks,

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|.$$

- **Cross-entropy loss.** This is also known as the log loss function. It is used in binary classification tasks where it measures the performance of a classification model whose output is a probability between 0 and 1.

$$LogLoss = -\frac{1}{n}\sum_{i=1}^{n}[y_i log(\hat{y}_i) + (1 - y_i)log(1 - \hat{y}_i)].$$

- **Cosine similarity loss.**

$$\ell(\theta) = 1 - cos(\theta) = 1 - \frac{A \cdot B}{||A||||B||},$$

  with $A$ as the true vector and $B$ as the predicted vector.

### 3.4.2 Optimization and Machine Learning Algorithms.

In machine learning to find the minimum of a cost function, we use optimization algorithms which iteratively modify model parameters until a desired estimate is obtained. Examples of these algorithms include gradient descent, stochastic gradient descent and Adam.

### 3.4.3 Convex and Non-convex Optimization.

A real-valued function is said to be convex if the line segment joining any two distinct points on its graph lies above the graph between the two points. Convex optimization involves convex objective functions with global minima. Many of the machine learning problems are non-convex resulting in the occurrence of multiple minima. Techniques for convex optimization such as, quadratic programming and Langrage multipliers provide foundational tools for the more complex non-convex optimization problems.

## 3.5 Probability Theory

In machine learning, we have probabilistic models and a probabilistic framework. The rigorous probabilistic framework describes how to represent and manipulate uncertainty in models and their predictions. Here, learning can be seen as an attempt to make predictions about future data and the consequences of decisions taken. Observed data can have consistency with different models and thus an appropriate model is uncertain. Further still, predictions about future data and the consequences of future actions are uncertain.

### 3.5.1 The Probabilistic Framework.

In a probabilistic framework for supervised machine learning, the goal is still one of approximating the relationship between input features,$X$ and labels,$Y$. For regression tasks, we assume that $Y \in [0,1]$ and we assume that $Y \in \{0,1\}$ for classification tasks. The relationship between $X$ and $Y$ is encapsulated by an unknown probability measure $\mu$ on $Z = X \times Y$. The elements $(x,y) \in Z$ can be called labeled examples. Learning takes place when we present such labeled data of the form $z \in Z^m$ to a machine learning algorithm which must return $h : X \to [0,1]$ for a fixed set of possible hypothesis. The level at which the output hypothesis $\mathcal{A}(z)$ correctly represents the hidden hypothesis $\mu$ is quantified with aid of a loss function $\ell : [0,1] \to [0,1]$. We always aim at having an $\mathcal{A}(z)$ with a small value of the loss [1].

### 3.5.2 What are Probabilistic Models.

These are models that quantify the inherent uncertainty in data and integrate it into their predictions. These models are applied in speech and image recognition systems, recommendation systems, natural language processing and more.

### 3.5.3 Categories of Probabilistic Models.

- **Generative models**. These aim at modeling the joint distribution of the predictor and the target variable. Generative models create new datasets by utilizing the probability distribution of the original dataset. Examples of these models include; Bayesian networks, variational auto-encoders, pixel recurrent neural networks, Markov chains and diffusion models. A good application of these models is Chat Generative Pre-trained Transformer(ChatGPT) which is a type of a large language model that uses probabilistic methods to generate text.

- **Discrimination models**. These aim at discriminating the conditional distribution of the target variables given the predictor variables. Discrimination models learn an appropriate decision boundaries which separates different classes of target variables. Examples of these models include support vector machines, logistic regression and some neural networks.

- **Graphical models**. Graphical probabilistic machine learning models determine the conditional dependence between variables using graphical representations. They include Bayesian networks which are basically directed graphical models and Markov networks which utilize undirected graphs.

## 3.6 Statistics

Different machine learning models fall in a class of statistical learning models in which we employ vast statistical techniques to develop models that have ability to learn from data and make appropriate predictions. Statistics also provides us powerful tools in a form of descriptive statistics that enable us visualize data, summarize data, identify patterns and also detect outliers in a dataset. Furthermore, the principles of statistics form pillars upon which we construct machine learning models, interpret results, validate models.

### 3.6.1 Various Statistical Tools in Machine Learning.

These include;

- Measures of central tendency.
- Measures of spread.
- Sampling.
- Estimation.
- Hypothesis testing.
- Cross validation.

Machine learning models that employ vast concepts from statistics include;

1. **Linear regression.** This is a supervised machine learning algorithm we employ to establish the relationship between predictor variables and target variables. More about linear regression and its utility of statistical tools is elaborated in Chapter 4 and Chapter 5.

2. **Logistic regression**. This employs the logistic function to estimate the probability of a categorical outcome basing on the input variables. Logistic regression performs binary classification using the logistic function also known as the sigmoid function to map outcomes to their probabilities. The logistic function is defined as;

$$\sigma(z) = \frac{1}{1 + e^{-z}},$$

   where $z$ is a linear combination of input features.

3. **Decision tree**. This employs statistical tools to split data based on the features in the dataset creating a tree-like structure. For regression tasks, we can employ a loss function whereas for classification tasks, we can use the Gini index and the entropy function to show how "pure" leaf nodes are. The Gini index and the entropy function are respectively given by;

$$G(t) = 1 - \sum_{i=1}^{c} {p_i}^2 \quad \text{and} \quad E = -\sum_{i=1}^{n} p_i log(p_i).$$

In the Gini index expression, $t$ is a node with $n_t$ data points and $p_i$ is the proportion of data points in the node $t$ that belongs to a class $i$. In the entropy formula, $E$ is entropy over all classes, $p_i$ is the proportion of training data points with a class label $i$.

4. **Random forest**. This is an ensemble machine learning model which improves the accuracy of predictions by using several decision trees. Random forests sample randomly selected subsets of features to build trees. Each tree in a decision forest makes a forecast for an output and the final prediction is obtained from the majority vote for all the trees in the forest.

$$Random\ forest = Decision\ trees + Bagging + Bootstraping + Random\ split.$$

  - Bagging is a learning method that employs several different models built by training them on different sample subsets then after we either aggregate, average or take majority of the predictions.
  - Bootstrapping is a resampling technique used to estimate a model's parameters by repeatedly sampling with replacement from a training dataset.

**3.6.2 Remark.** We have a special class of trees called boosted trees. The process of boosting involves combining several poorly performing classifiers to obtain a better classifier. For example, instead of using an equal probability for all items in a dataset, we can give a higher weight to items that have been misclassified by the model. XGBoost which stands for extreme gradient boosting is one of the models that employs boosted trees in its algorithm.

**3.6.3 Remark.** Majority of the statistical machine learning models such as decision tree, random forest and logistic regression incorporate vast ideas from probability theory. At this point, we can also observe that a single machine learning algorithm can employ concepts from different areas of mathematics.

## 3.7 Geometry

Geometry in mathematics deals with shapes and structures. Other than the usual Euclidean geometry, other geometries are also relevant in machine learning such as the elliptic geometry of Riemann. However, non-Euclidean data suffers the curse of dimensionality, we need an exponential number of samples to approximate even a basic multidimensional function. Principal component analysis as a technique for dimensionality reduction was described in Subsection 3.2.2, however, sometimes losing important information about the data is inevitable. Thus, we employ the geometric structure of the input data to overcome this problem and this is formalized as geometric priors. For example, in geometric deep learning, functions used must respect two priors;

- Symmetry: This is respected by functions that leave an object invariant. These functions must be composable, invertible and their collection should contain the identity map.
- Scale preparation: This deals with stability of a function under slight deformation of its domain.

## 3.8 Graph Theory

Graphs are one of the prime objects in discrete mathematics. A graph,$G$, is an ordered pair $G = (V, E)$ where $V$ are vertices and $E$ are edges. A common application of graph theory is graph clustering in unsupervised learning where we partition vertices in a graph into groups based on their similar characteristics. Graph clustering algorithms are employed in social network analysis, biological network analysis, recommendation systems, security and fraud detection [23]. The Figure 3.3 shows network clusters for brain functionality that was simulated with Matlab using Lancichinetti-Fortunato-Radicchi benchmark algorithm.

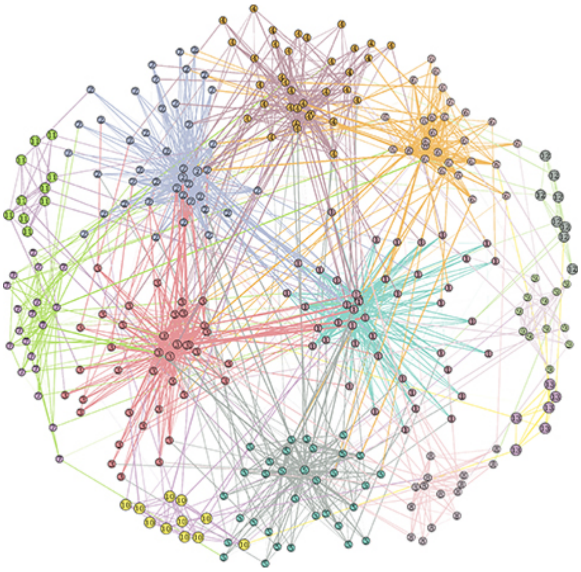

Figure 3.3: Shows network clusters for brain functional connectivity [3].

### 3.8.1 Graph Clustering Algorithms.

These include;

- **$k$-means clustering**. This partitions a graph into $k$ distinct clusters based on $k$ centroids that are initialized by the user. Given a similarity measure, $k$-means clustering recomputes the centroids until convergence.

- **Node embedding**. This is a technique in machine learning that represents graph nodes as vectors in a low-dimensional space. This exploits the fact that there are several mathematical techniques that can be employed to cluster similar vectors. In this algorithm, similar nodes get the same node embedding.

- **Label propagation**. Here, we label each node and iteratively pass the labels between neighboring nodes. Labels are associated with probabilities and a node is assigned the most probable label. If a node is highly representative of its cluster, its label propagates through and wins over all of the other labels.

# 4. Simple Linear Regression

Linear regression is a fundamental supervised learning algorithm that is employed on data with continuous input and output variables. In linear regression, we seek to obtain a combination of inputs that best explains the output by assuming a linear connection between the input and output variables [28]. Simple linear regression models involve one independent variable and one dependent variable. The words; independent variable, predictor and explanatory variable can be used interchangeably with input variable. Also the words; labels, dependent variable and target variable are used interchangeably with output variable. The simple linear regression model estimates the intercept and the slope of a line of best fit between the input and output variables.

## 4.1 A Simple Linear Regression Task

We begin with an example to motivate an understanding of simple linear regression. In chemistry and various industries, pH measurements play a key role in success of experiments and formation of desired products. Various factors affect the pH of a given solution. The data in Table 4.1 shows how pH of pure water varies with increasing temperature.

| Temperature($^{\circ}C$) | 0 | 10 | 15 | 20 | 25 | 30 | 40 | 50 | 100 |
|---|---|---|---|---|---|---|---|---|---|
| pH | 7.47 | 7.27 | 7.17 | 7.08 | 7.00 | 6.92 | 6.77 | 6.63 | 6.14 |

Table 4.1: Shows how pH of pure water varies with changes in temperature.

**Aim**: To predict the pH of pure water when given a particular temperature.

### 4.1.1 The Approach.

Using matplotlib.pyplot library in python, we create a scatter plot for visualization of the trend of the data using 8 of the data points leaving out the third data point and this is as shown in the first subplot in Figure 4.1. We fit a simple linear regression model using LinearRegression model from sklearn.linear_model provided by the scikit-learn library in python. This process fits a model to the data and we obtain the optimal estimates of the intercept and the slope. The second subplot of the Figure 4.1 shows the fitted linear regression model trend line. Generally, the equation of a straight line is $y = mx + c$ which we use in this work as $y = \beta_1 x + \beta_0$, where $m = \beta_1$ is the gradient and $c = \beta_0$ the $y$-intercept.

Thus, the values of parameters are $\hat{\beta}_0 = 7.3585$ and $\hat{\beta}_1 = -0.01305$. The equation of model is

$$y = 7.3585 - 0.01305x. \tag{4.1.1}$$

**Prediction:** The model was tested on its capacity to generalize using $x = 15^{\circ}C$ and the corresponding prediction was a pH value, $\hat{y} = 7.16279$ while the actual value was $y = 7.17$.

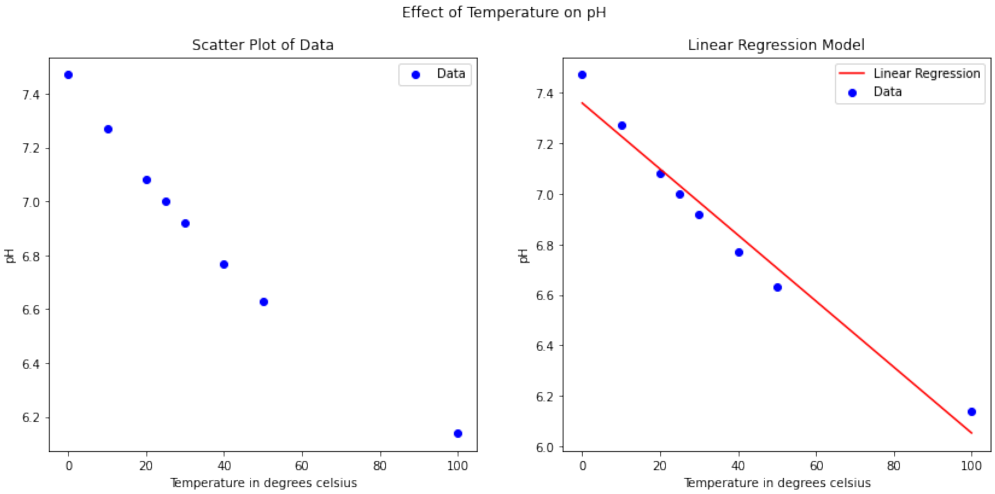

Figure 4.1: The left plot shows a scatter plot of the data points. The right plot shows the linear regression model line fitted to the data.

### 4.1.2 Interpretation: A Chemistry Perspective.

When temperature increases, pH reduces, however, this does not necessarily imply that a given sample of pure water has become acidic. Increasing temperature increases molecular vibrations within water molecules causing more dissociation of the water molecules thus yielding more hydroxyl and hydrogen ions. The increased hydrogen ions cause a decrease in the pH of water as pH is simply a measure of the hydrogen ion concentration of a solution. However, the hydroxyl and hydrogen ions are liberated in equimolar proportions and this ensures that the pure water remains neutral.

**Implication:** When one sets out to measure the pH of pure water or any other solution, the temperature too should be measured. Notably, a pH value without a corresponding temperature is incoherent. Using a pH sensor that has an automatic temperature compensation stands a viable alternative.

**4.1.3 Remark.** We would like to know how accurate the values of the model parameters obtained are, gauge the closeness of the predicted to the actual values and also we want to know about the general performance of the entire model. This calls for rigorous mathematical formulations which are covered in the proceeding sections of this chapter.

## 4.2 The Least Squares Regression Approach

We consider a case where we have a single real-valued feature as input and a real-valued output. Let us denote the input-output pairs as $(x_i, y_i)$ with $x_i, y_i \in \mathbb{R}$ and $i \in \{1, 2, ....., n\}$. The model here is from a family of straight line functions mapping from $\mathbb{R}$ to $\mathbb{R}$. As earlier noted, we use the equation of a straight line as $y = \beta_1 x + \beta_0$ and in linear regression, the task is to find the best combination of $\beta_0$ and $\beta_1$ that best describe the trend observed in the data [13].

To obtain a straight line that best describes the trend in a dataset, we use a loss function that tells us how far off we are from the true value $y$ when we use its approximation $\hat{y}$. We denote the loss function as $\ell(y, \hat{y})$ such that $\ell : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$. There are different kinds of loss functions, we use the quadratic loss function also known as the squared loss function defined as;

$$\ell(y, \hat{y}) = (y - \hat{y})^2.$$

The cost obtained using the squared loss function is non-negative and it increases quadratically, for example every time we double $y - \hat{y}$, the cost increases by a factor of four. The Figure 4.2 shows the distances we wish to minimize on the left subplot and how the losses vary quadratically when using the squared loss function on the right subplot for an arbitrary dataset.

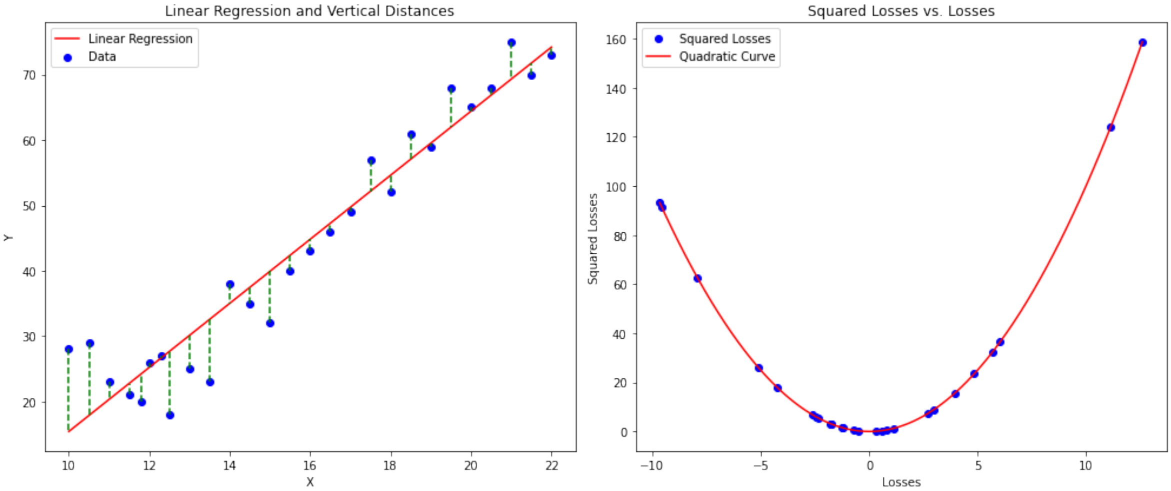

Figure 4.2: The first subplot shows vertical distances from the data points to the regression line. The second subplot shows how losses are varying with the squared loss function.

Using the equation of a straight line and working with the data points $(x_i, y_i)$, we have;

$$\hat{y}_i = \beta_1 x_i + \beta_0.$$

The left subplot of Figure 4.2 can aid in the visualization of labels predicted by the fitted regression line and the true labels as by the data. Thus, for any datum, we have the squared loss as;

$$\ell(y_i, \hat{y}_i) = (y_i - \beta_1 x_i - \beta_0)^2.$$

If we take $n$ data points which are usually the number of rows in a dataset and consider the average loss also known as the empirical loss, then, we have that;

$$\begin{aligned} \mathscr{L}(\beta_0, \beta_1) &= \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \hat{y}_i), \\ &= \frac{1}{n} \sum_{i=1}^{n} (y_i - \beta_1 x_i - \beta_0)^2, \\ \mathscr{L}(\beta_0, \beta_1) &= \frac{1}{n} \sum_{i=1}^{n} (\beta_1 x_i + \beta_0 - y_i)^2. \end{aligned} \tag{4.2.1}$$

The empirical loss function,$\mathscr{L}(\beta_0, \beta_1)$ can be minimized with respect to the parameters $\beta_0$ and $\beta_1$ in order to obtain optimal estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize it;

$$\hat{\beta}_0, \hat{\beta}_1 = arg \min_{\beta_0,\beta_1} \mathscr{L}(\beta_0, \beta_1),$$

$$\hat{\beta}_0, \hat{\beta}_1 = arg \min_{\beta_0,\beta_1} \frac{1}{n} \sum_{i=1}^{n} (\beta_1 x_i + \beta_0 - y_i)^2. \tag{4.2.2}$$

Arguments that minimize the empirical loss function or empirical risk function,$\mathscr{L}(\beta_0, \beta_1)$ are denoted in Equation (4.2.2) as "*arg min*". The problem of obtaining estimates of $\beta_0$ and $\beta_1$ that minimize $\mathscr{L}(\beta_0, \beta_1)$ is the *empirical risk minimization* and we accomplish it with the use of partial derivatives of the empirical risk function with respect to the parameters $\beta_0$ and $\beta_1$.

Taking the partial derivative of $\mathscr{L}(\beta_0, \beta_1)$ with respect to $\beta_0$, we have that;

$$\begin{aligned}
\frac{\partial}{\partial \beta_0} \mathscr{L}(\beta_0, \beta_1) &= \frac{\partial}{\partial \beta_0} \left( \frac{1}{n} \sum_{i=1}^{n} (\beta_1 x_i + \beta_0 - y_i)^2 \right), \\
&= \frac{1}{n} \sum_{i=1}^{n} \frac{\partial}{\partial \beta_0} (\beta_1 x_i + \beta_0 - y_i)^2, \\
&= \frac{2}{n} \sum_{i=1}^{n} (\beta_1 x_i + \beta_0 - y_i), \\
\frac{\partial}{\partial \beta_0} \mathscr{L}(\beta_0, \beta_1) &= 2\beta_1 \left( \frac{1}{n} \sum_{i=1}^{n} x_i \right) + 2\beta_0 - 2 \left( \frac{1}{n} \sum_{i=1}^{n} y_i \right).
\end{aligned} \tag{4.2.3}$$

Also, taking the partial derivative of $\mathscr{L}(\beta_0, \beta_1)$ with respect to $\beta_1$, we have that;

$$\begin{aligned}
\frac{\partial}{\partial \beta_1} \mathscr{L}(\beta_0, \beta_1) &= \frac{\partial}{\partial \beta_1} \left( \frac{1}{n} \sum_{i=1}^{n} (\beta_1 x_i + \beta_0 - y_i)^2 \right), \\
&= \frac{1}{n} \sum_{i=1}^{n} \frac{\partial}{\partial \beta_1} (\beta_1 x_i + \beta_0 - y_i)^2, \\
&= \frac{2}{n} \sum_{i=1}^{n} x_i (\beta_1 x_i + \beta_0 - y_i), \\
\frac{\partial}{\partial \beta_1} \mathscr{L}(\beta_0, \beta_1) &= 2\beta_1 \left( \frac{1}{n} \sum_{i=1}^{n} {x_i}^2 \right) + 2\beta_0 \left( \frac{1}{n} \sum_{i=1}^{n} x_i \right) - 2 (\frac{1}{n} \sum_{i=1}^{n} x_i y_i).
\end{aligned} \tag{4.2.4}$$

The expressions in Equations (4.2.3) and (4.2.4) involve four averages and for convenience, these can be simplified as;

$$\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i \quad , \quad \bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i,$$

$$\lambda = \frac{1}{n} \sum_{i=1}^{n} {x_i}^2 \quad , \quad \alpha = \frac{1}{n} \sum_{i=1}^{n} x_i y_i.$$

For minimum values, we set the partial derivatives in Equations (4.2.3) and (4.2.4) to zero. We obtain a system of two equations in two unknowns $\hat{\beta}_0$ and $\hat{\beta}_1$;

$$\begin{aligned}\frac{\partial}{\partial\beta_0}\mathscr{L}(\beta_0,\beta_1) &= 0,\\ 2(\hat{\beta}_1\bar{x}+\hat{\beta}_0-\bar{y}) &= 0,\\ \hat{\beta}_1\bar{x}+\hat{\beta}_0-\bar{y} &= 0, \qquad (4.2.5)\end{aligned}$$

Also, we have;

$$\begin{aligned}\frac{\partial}{\partial\beta_1}\mathscr{L}(\beta_0,\beta_1) &= 0,\\ 2(\hat{\beta}_1\lambda+\hat{\beta}_0\bar{x}-\alpha) &= 0,\\ \hat{\beta}_1\lambda+\hat{\beta}_0\bar{x}-\alpha &= 0, \qquad (4.2.6)\end{aligned}$$

We multiply the Equation (4.2.5) by $\bar{x}$ and then we subtract it from Equation (4.2.6). After such an operation, we obtain that;

$$\hat{\beta}_1 = \frac{\alpha-\bar{x}\bar{y}}{\lambda-\bar{x}^2} = \frac{\frac{1}{n}\sum_{i=1}^{n}x_iy_i-\left[\left(\frac{1}{n}\sum_{i=1}^{n}x_i\right)\left(\frac{1}{n}\sum_{i=1}^{n}y_i\right)\right]}{\frac{1}{n}\sum_{i=1}^{n}{x_i}^2-\left(\frac{1}{n}\sum_{i=1}^{n}x_i\right)^2}, \qquad (4.2.7)$$

$$\hat{\beta}_0 = \bar{y}-\hat{\beta}_1\bar{x} = \frac{1}{n}\sum_{i=1}^{n}y_i-\frac{\hat{\beta}_1}{n}\sum_{i=1}^{n}x_i. \qquad (4.2.8)$$

If $\hat{y}_i = \hat{\beta}_1x_i+\hat{\beta}_0$ is the predicted value of $y$ based on some $ith$ value of $x$, then, $e_i = y_i-\hat{y}_i$ is the value of the $ith$ residual. Thus, $e_i$ is the difference between the value of the $ith$ observed response value and the value of the $ith$ prediction made by the linear regression model. The $residual\ sum\ of\ squares(RSS)$ is defined as;

$$RSS = {e_1}^2+{e_2}^2+\cdots+{e_n}^2 = \sum_{i=1}^{n}{e_i}^2. \qquad (4.2.9)$$

This can equivalently be represented as;

$$\begin{aligned}RSS &= (y_1-\hat{\beta}_1x_1-\hat{\beta}_0)^2+(y_2-\hat{\beta}_1x_2-\hat{\beta}_0)^2+\cdots+(y_n-\hat{\beta}_1x_n-\hat{\beta}_0)^2,\\ RSS &= \sum_{i=1}^{n}(y_i-\hat{\beta}_1x_i-\hat{\beta}_0)^2. \qquad (4.2.10)\end{aligned}$$

## 4.3 Assessing the Accuracy of Coefficient Estimates

When we have data about the input and the output variables, the true relationship between the two is unknown. We assume that the true relationship is of the form $y = f(x)+\epsilon$ with $f$ some an unknown function and $\epsilon$ is a normally distributed independent error term that is a catch all for what we miss in the model [20]. We can approximate $f$ by a linear function between $x$ and $y$ as;

$$y = \beta_1 x + \beta_0 + \epsilon. \tag{4.3.1}$$

The Equation (4.3.1) provides us with the best linear approximation to the true relationship between variables $x$ and $y$ which is also known as the population regression line. The least squares regression coefficient estimates in Equations (4.2.7) and (4.2.8) are characteristic of the least squares regression line.The distinction is visualized by the Figure 4.3 using a randomly generated set of 20 data points for a hypothetical model $y = 3 + 4x + \epsilon$ with epsilon randomly generated from a normal distribution.

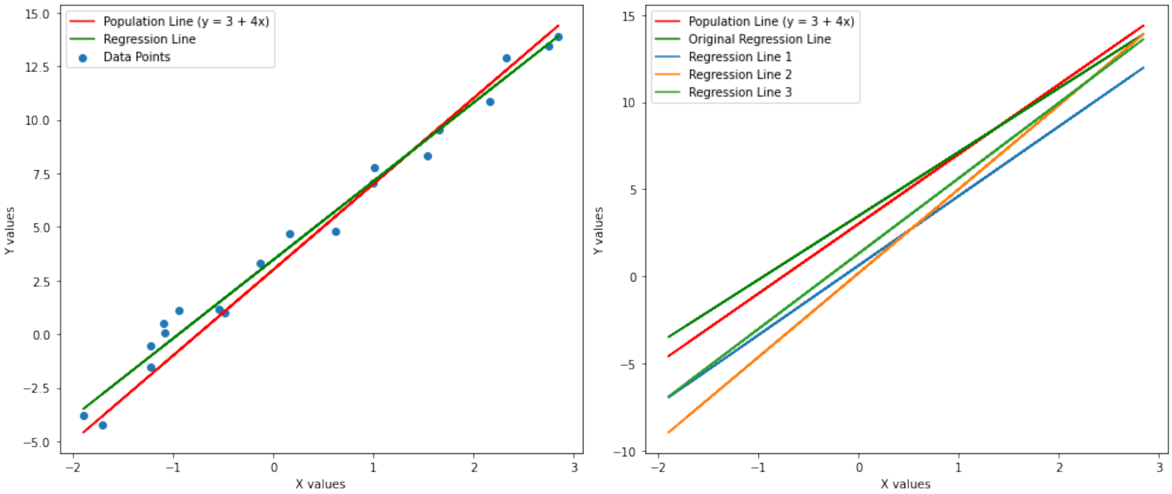

Figure 4.3: In the first subplot, the green line represents the linear regression line and the red line the population regression line. The second subplot includes three more regression lines for more random sets of observations.

The red line in the Figure 4.3 shows the true relationship $f(x) = 3 + 4x$ whereas the green line shows the least squares estimate based on the input data. In real life applications, we are given a set of data from which we compute the least squares regression coefficients, however, the population regression lines remains unknown.

We are interested in assessing the accuracy of the least squares regression coefficients, $\hat{\beta}_0$ and $\hat{\beta}_1$. A good analogy is the sample mean and the population mean for $n$ observations, $Z = z_1, z_2, ...., z_n$. An unbiased estimate for the population mean,$\mu$ is the sample mean,$\hat{\mu}$ with

$$\hat{\mu} = \bar{z} = \frac{1}{n}\sum_{i=1}^{n} z_i. \tag{4.3.2}$$

In a similar manner, the true population coefficients $\beta_0$ and $\beta_1$ remain unknown but, $\hat{\beta}_0$ and $\hat{\beta}_1$ provide their unbiased estimates and this is proved in Section 5.4. If we separately average the estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ over a large number of datasets, then, the average of the estimates would be very close or even equal to the actual $\beta_0$ and $\beta_1$ of the population.

### 4.3.1 How close $\hat{\beta}_0$ and $\hat{\beta}_1$ are to $\beta_0$ and $\beta_1$.

We again uses the $\mu$ and $\hat{\mu}$ analogy. A single $\hat{\mu}$ can underestimate or overestimate $\mu$. Thus, the standard error,$SE(\hat{\mu})$ of a single $\hat{\mu}$ can be obtained from its variance that is given by;

$$var(\hat{\mu}) = SE(\hat{\mu})^2 = \frac{\sigma^2}{n}. \tag{4.3.3}$$

Equation (4.3.3) tells us how a single estimate $\hat{\mu}$ on average differs from $\mu$ and how the variance shrinks with respect to $n$. For least squares regression, the estimate of the standard error is;

$$SE(\hat{\beta}_i) = RSE = \sqrt{\frac{RSS}{n-2}},$$

where $RSE$ is the residual standard error and the denominator is $n-2$ because we would have already estimated $\beta_0$ and $\beta_1$. The estimators $\hat{\beta}_0$ and $\hat{\beta}_1$ follow the distributions;

$$\hat{\beta}_0 \sim N\left(\beta_0, \sigma^2\left[\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}\right]\right), \qquad \hat{\beta}_1 \sim N\left(\beta_1, \frac{\sigma^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}\right). \tag{4.3.4}$$

Therefore, the standard errors $SE(\hat{\beta}_0)$ and $SE(\hat{\beta}_1)$ are;

$$SE(\hat{\beta}_0) = \sqrt{\sigma^2\left[\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}\right]}, \qquad SE(\hat{\beta_1}) = \sqrt{\frac{\sigma^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}}, \tag{4.3.5}$$

where $\sigma^2$ is $Var(\epsilon)$.

### 4.3.2 Confidence Intervals.

The standard errors in Equations (4.3.5) can be used to compute confidence intervals within which we can capture the unknown values of the population parameters $\beta_0$ and $\beta_1$. A $95\%$ confidence interval in simple linear regression takes the form;

$$\hat{\beta}_1 \pm 2SE(\hat{\beta}_1).$$

This is to say that we have $95\%$ chance that the interval, $[\hat{\beta}_1 - 2SE\hat{\beta}_1, \hat{\beta}_1 + 2SE\hat{\beta}_1]$ captures the true value of $\beta_1$. A similar reasoning applies to $\hat{\beta}_0$.

### 4.3.3 Hypothesis Testing.

In simple linear regression, standard errors are also used for hypothesis testing on coefficients. For a predictor,$x$ and a tareget variable $y$, let us have a null hypothesis,$H_0$ and an alternative hypothesis,$H_a$ as;

| Null hypothesis,$H_0$ | Alternative hypothesis,$H_a$ |
|---|---|
| There is no relationship between $x$ and $y$. | There is a relationship between $x$ and $y$. |

Which corresponds to testing; $H_0 : \beta_1 = 0$ versus $H_a : \beta_1 \neq 0.$

If $\beta_1 = 0$, then we have that $y = \beta_0 + \epsilon$ and this gives an implication that $x$ is not associated with $y$. To test the null hypothesis, we need to determine if $\hat{\beta}_1$ is sufficiently large and far from zero so that we are sure $\beta_1$ is not zero. If $SE(\hat{\beta}_1)$ has a small value, then, even relatively small values of $\hat{\beta}_1$ provide sufficient evidence that $\beta_1$ is not zero. If $SE(\hat{\beta}_1)$ has a large value, then, $\hat{\beta}_1$ should to be large enough such that $\beta_1$ is not zero [15].
In practice, we use the *t-statistic* from the *t-distribution* to measure the number of standard deviations that $\hat{\beta}_1$ is away with from zero. The *t-statistic* is defined as;

$$t = \frac{\hat{\beta}_1}{SE(\hat{\beta}_1)}.$$

We compute the probability of having any number say $q \in \mathbb{R}$ such that $q \geq |t|$ assuming $\beta_1 = 0$. This probability is what we call the *p-value*. If the *p-value* is small, then we can conclude that there is an association between $x$ and $y$ then we reject the null hypothesis. Using the data on pH of pure water in Table 4.1, we have the following results;

| | *coefficient* | *standard error* | *t-statistic* | *p-value* |
|---|---|---|---|---|
| Intercept | 7.35850 | 0.04223 | 174.21542 | 2.41301$e$-12 |
| Temperature | -0.01305 | 0.00094 | -13.86829 | 8.75248$e$-06 |

Table 4.2: The table shows the coefficients from the simple linear model in Equation 4.1.1, the *standard errors*, *t-values* and *p-values*.

- The intercept,$\hat{\beta}_0$ is a positive coefficient and has a very high *t-value*, indicating that it is significantly different from zero.
- The temperature coefficient,$\hat{\beta}_1$ is negative with a big negative t-value indicating that there is a significant inverse relationship between temperature and pH. The associated *p-value* is very low, confirming the statistical significance of this inverse relationship.

## 4.4 Assessing the Accuracy of the Model

When we reject the null hypothesis and accept the alternative hypothesis, then we would like to know the extent to which the model fits the data. We assess the quality of the simple linear regression fit using the *residual standard error(RSE)* and the $R^2$-*statistic* [7].

### 4.4.1 The Residual Standard Error(RSE).

By the Equation (4.3.1), the presence of $\epsilon$ tells us that we are not able to accurately predict the output,$y$ when we have the input feature,$x$. The *residual standard error* is an estimate of the standard deviation of $\epsilon$ which is typically the average amount by which the estimates $\hat{y}'s$ deviate from the true regression line of the population. It is given by the formula;

$$RSE = \sqrt{\frac{1}{n-2} RSS}.$$

As we saw in Section 4.2, the $residual\ sum\ of\ squares, RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ and thus;

$$RSE = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}. \tag{4.4.1}$$

- $Residual\ standard\ error (RSE)$: This is a measure of the lack of fit of a given linear regression model to the data. For a model that fits the data well, the $RSE$ value is small and for a model that does not fit the data well, the $RSE$ value is large indicating that the estimates $\hat{y}_i's$ are far from the ground truths $y_i's$.
- The $R^2$-$statistic$: This is a measure of fit that quantifies the proportion of variance in the response variable,$y$ that is explained by the predictor variable,$x$. The $R^2$-$value$ is calculated by the formula;
$$R^2 = \frac{TSS - RSS}{TSS} = 1 - \frac{RSS}{TSS}.$$
where total sum of squares, $TSS = \sum_{i=1}^{n}(y_i - \bar{y})^2$ and $RSS$ is as in Equation (4.2.9),

Now, using the data on the variation of pH of pure water with temperature in Table 4.1, we can observe the following about the entire constructed model;

| Statistical Quantity | Value |
|---|---|
| $RSE$ | 0.0768 |
| $R^2$ | 0.9697 |

Table 4.3: The table shows the $residual\ standard\ error$ and the $R^2$ value based on the data.

- The residual standard error,$RSE$ tells us that, on average, the predicted pH values deviate from the actual pH values by about 0.07862 units and this is a small $residual\ standard\ error$ indicating a better fit of the model to the data.
- The $R^2$-$value$ of approximately, 0.9697 indicates that about 96.97% of the variability in pH of pure water can be explained by the linear relationship with temperature. This tells us that the model provides a good fit to the observed trend between pH values of pure water and temperature.

# 5. Multiple Linear Regression

## 5.1 The Foundation of Multiple Linear Regression

In practice, we usually have more than one predictor for the output variable(s). Instead of fitting separate linear regression models for each predictor, we extend the model to take in more predictors [19]. The multiple linear regression model takes the form;

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_m x_m + \epsilon, \tag{5.1.1}$$

where each $\beta_i$ represents an association between some predictor,$x_i$ and the response,$y$, $m$ is the total number of features and $\epsilon$ is the error term.
Using the data in the Table 5.1, we can have the following multiple linear regression model;

$$pH = \beta_0 + \beta_1(Temperature) + \beta_2(\delta_w) + \beta_3(K_w).$$

We can have some of the predictors in the model as; powers of the original predictor, interactions of the predictors or else as a mixture of powers and interactions of the predictors;

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_m x^m + \epsilon, \tag{5.1.2}$$
$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{12} x_1 x_2 + \cdots + \beta_m x_m + \epsilon, \tag{5.1.3}$$
$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} {x_1}^2 + \beta_{12} x_1 x_2 + \cdots + \beta_m x_m + \epsilon. \tag{5.1.4}$$

We hold the following assumptions which apply apply to both multiple and simple linear regression;

$$Cov(\epsilon_i, \epsilon_j) = 0, \quad \text{for} \quad i \neq j, \quad \text{with} \quad i, j \in \{0, 1, 2, ..., n\}.$$
$$\mathbb{E}(\epsilon_i) = 0 \quad \text{and} \quad Var(\epsilon_i) = \sigma^2 \quad \forall i \quad \text{with} \quad i \in \{0, 1, 2, ..., n\}.$$

The three Equations (5.1.2), (5.1.3) and (5.1.4) are all multiple linear regression models, here, linearity is in terms of the regression coefficients. If we consider a sample of size $n$, then the sample version of the Equation (5.1.1) is;

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_m x_{im} + \epsilon_i, \quad \text{with} \quad i \in \{0, 1, 2, ..., n\}. \tag{5.1.5}$$

For each of the $n$ observations from a given dataset, we have the following system of equations;

$$\begin{aligned} y_1 &= \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \cdots + \beta_m x_{1m} + \epsilon_1, \\ y_2 &= \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \cdots + \beta_m x_{2m} + \epsilon_2, \\ \vdots & \qquad \vdots \\ y_n &= \beta_0 + \beta_1 x_{n1} + \beta_2 x_{n2} + \cdots + \beta_m x_{nm} + \epsilon_n. \end{aligned}$$

Expressing the above system of equations in matrix form yields;

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1m} \\ 1 & x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_m \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}.$$

The above matrix form can be condensed to;

$$\underbrace{y}_{n\times 1} = \underbrace{X}_{n\times p} \cdot \underbrace{\beta}_{p\times 1} + \underbrace{\epsilon}_{n\times 1}, \tag{5.1.6}$$

with $p = m + 1$.

## 5.2 A Multiple Linear Regression Task

We extend the Table 4.1 to include two more features that is; conductivity values of pure water,$\delta_w$ in $\mu S/cm \times 10^{-2}$ and dissociation constants of pure water,$K_w$ in $mol^2 dm^{-6} \times 10^{-14}$.

| Temperature(C) | $\delta_w(\mu S/cm)$ | $K_w(mol^2dm^{-6})$ | pH value |
|---|---|---|---|
| 0 | 0.1162 | 0.114 | 7.47 |
| 10 | 2.312 | 0.293 | 7.27 |
| 20 | 4.205 | 0.681 | 7.08 |
| 25 | 5.512 | 1.008 | 7.00 |
| 30 | 7.105 | 1.471 | 6.92 |
| 40 | 11.298 | 2.916 | 6.77 |
| 50 | 17.071 | 5.476 | 6.63 |
| 100 | 77.697 | 51.300 | 6.14 |

Table 5.1: Shows temperature, conductivity values, dissociation constants and pH values of pure water.

For the data in the Table 5.1, the multiple linear regression line is of the form;

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3, \tag{5.2.1}$$

where; $\hat{\beta}_1$,$\hat{\beta}_2$ and $\hat{\beta}_3$ are gradients corresponding to the features; temperature($x_1$), electric conductivity($x_2$), dissociation constant($x_3$) respectively and $\hat{\beta}_0$ is a constant. The dependent variable estimate,$\hat{y}$ denotes the pH value of pure water depending on the three input features, $x_1, x_2$ and $x_3$.
The Equation (5.2.1) can be rewritten as;

$$\hat{y} = \sum_{i=0}^{3} \hat{\beta}_i x_i = \hat{\beta}_0 x_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3, \tag{5.2.2}$$

with $x_0 = 1$.

We convert the Equation (5.2.2) to matrix form. Then insert feature values and output values;

$$y = \begin{bmatrix} x_0 & x_1 & x_2 & x_3 \end{bmatrix} \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \hat{\beta}_2 \\ \hat{\beta}_3 \end{bmatrix},$$

$$\underbrace{\begin{bmatrix} 7.47 \\ 7.27 \\ 7.08 \\ 7.00 \\ 6.92 \\ 6.77 \\ 6.63 \\ 6.14 \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} 1 & 0 & 0.1162 & 0.114 \\ 1 & 10 & 2.312 & 0.293 \\ 1 & 20 & 4.205 & 0.681 \\ 1 & 25 & 5.512 & 1.008 \\ 1 & 30 & 7.105 & 1.471 \\ 1 & 40 & 11.298 & 2.916 \\ 1 & 50 & 17.071 & 5.476 \\ 1 & 100 & 77.697 & 51.300 \end{bmatrix}}_{X} \underbrace{\begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \hat{\beta}_2 \\ \hat{\beta}_3 \end{bmatrix}}_{\hat{\beta}}.$$

Given the feature matrix,$X$, we can find the values of parameters $\hat{\beta}_i$ for $i \in \{0, 1, 2, 3\}$ as;

$$\hat{\beta} = (X^T X)^{-1} X^T y, \tag{5.2.3}$$

where $\hat{\beta} = [\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3]^T$ with $T$ for transpose. Theorem 5.3.1 shows us how we can arrive at the Equation (5.2.3). Using a python library,NumPy, we utilize the NumPy arrays to handle the data which is in two dimensions. If we had a large dataset, we would split it into training and validation sets. Then, we implement multiple linear regression using the *LinearRegression* model from the *sklearn.linear_model* which is provided by the *scikit-learn* library in python. This process fits a multiple linear regression model to data and yields the optimal parameters as;

$$\hat{\beta}_0 = 7.464836, \quad \hat{\beta}_1 = -0.02479, \quad \hat{\beta}_2 = 0.23045, \quad \hat{\beta}_3 = 0.019051,$$

Thus, the equation of the model becomes;

$$y = 7.464836 - 0.02479 x_1 + 0.23045 x_2 + 0.019051 x_3. \tag{5.2.4}$$

**Prediction:** After creating the model, we test its power to generalize using data it has not seen before. New $[x_0, x_1, x_2, x_3] = [1, 15, 4.833, 0.895]$. The algorithm predicted a pH value, $\hat{y} = 7.2213$.

## 5.3 The Least Squares Estimation

We can still employ the least squares approach to fit a model to data with multiple predictors. The coefficients $\beta_0$, $\beta_1$, $\cdots$, $\beta_m$ are unknown and we would like to estimate them in such a way that they minimize the squared loss function,

$$\min_{\hat{\beta}} \; \ell(\hat{\beta}) = \min_{\hat{\beta}} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \min_{\hat{\beta}} \sum_{i=1}^{n} {e_i}^2 \quad \text{with} \quad i \in \{0, 1, 2, ..., n\}.$$

If we let $e = e_i \in \mathbb{R}^n$ and $\hat{y} = \hat{y}_i = X\hat{\beta} \in \mathbb{R}^n$, then, $e$ is obtained as $e = y - \hat{y}$. By the Equation (5.1.6), the minimization problem becomes;

$$\min_{\hat{\beta}} \ \ell(\hat{\beta}) = \min_{\hat{\beta}} ||e||^2 = \min_{\hat{\beta}} ||y - X\hat{\beta}||^2. \tag{5.3.1}$$

**5.3.1 Theorem.** *If $X^TX$ is nonsingular, then the least square estimate of $\beta$ is [9]*

$$\hat{\beta} = (X^TX)^{-1}X^Ty. \tag{5.3.2}$$

*Proof.* For the proof of this theorem, the following ideas about gradient of a function of multiple variables are required;

$$\begin{aligned} \frac{\partial}{\partial X}(X^Ta) &= \frac{\partial}{\partial X}(a^TX) = a, \\ \frac{\partial}{\partial X}(||X||^2) &= \frac{\partial}{\partial X}(X^TX) = 2X, \\ \frac{\partial}{\partial X}(X^TAX) &= 2AX, \\ \frac{\partial}{\partial X}(||AX||^2) &= \frac{\partial}{\partial X}(X^TA^TAX) = 2A^TAX, \end{aligned}$$

where $X$ and $a$ are vectors of same dimension,$n$ and $A$ is an $n \times n$ symmetric matrix.
For the proof of the Theorem 5.3.1, we use the identity $||U - V||^2 = ||U||^2 + ||V||^2 - 2U^TV$. Using Equation (5.3.1) and the stated identity, we have that;

$$\begin{aligned} \ell(\hat{\beta}) &= ||y||^2 + ||X\hat{\beta}||^2 - 2(X\hat{\beta})^Ty, \\ &= y^Ty + \hat{\beta}^TX^TX\hat{\beta} - 2\hat{\beta}^TX^Ty. \end{aligned}$$

We compute the derivative of the loss function with respect to $\hat{\beta}$ utilizing the ideas in the equations stated after the theorem statement;

$$\frac{\partial \ell}{\partial \hat{\beta}} = 0 + 2X^TX\hat{\beta} - 2X^Ty.$$

We set the gradient function, $\dfrac{\partial \ell}{\partial \hat{\beta}}$ to zero and this yields the least squares normal equations as;

$$X^TX\hat{\beta} = X^Ty. \tag{5.3.3}$$

Multiplying both sides of Equation (5.3.3) by $(X^TX)^{-1}$, we obtain $\hat{\beta}$ as;

$$\hat{\beta} = (X^TX)^{-1}X^Ty. \tag{5.3.4}$$

□

**5.3.2 Remark.** The very first normal equation of $X^TX\hat{\beta} = X^Ty$ is;

$$n\hat{\beta}_0 + \hat{\beta}_1 \sum_{i=1}^{n} x_{i1} + \hat{\beta}_2 \sum_{i=1}^{n} x_{i2} + \cdots + \hat{\beta}_m \sum_{i=1}^{n} x_{im} = \sum_{i=1}^{n} y_i.$$

Dividing through by $n$ yields;

$$\hat{\beta}_0 + \hat{\beta}_1\bar{x}_1 + \hat{\beta}_2\bar{x}_2 + \cdots + \hat{\beta}_m\bar{x}_m = \bar{y}.$$

This shows us that the centroid of the data is on the least squares regression plane. This signifies that the plane is a good representation of the central tendency of the data.

**5.3.3 Remark.** The fitted values of the least squares model are;

$$\hat{y} = X\hat{\beta} = \underbrace{X(X^TX)^{-1}X^T}_{H} y = Hy. \tag{5.3.5}$$

By the Equation (5.3.1), the residuals can be visualized as;

$$e = y - \hat{y} = y - Hy = (I - H)y. \tag{5.3.6}$$

The matrix, $H \in \mathbb{R}^{n\times n}$ is known as the *hat matrix* and it has the following characteristics;

- It is symmetric, that is; $H^T = H$.
- It is idempotent, that is; $H^2 = H$.
- $H(I - H) = O$ where $O$ is the zero matrix.

## 5.4 Point Estimation in Multiple Linear regression

The least squares estimator, $\hat{\beta}$ is an unbiased linear estimator for $\beta$. This still holds true for a simple linear regression case.

**5.4.1 Theorem.** *Under the assumptions of multiple linear regression*

$$\mathbb{E}(\hat{\beta}) = \beta. \tag{5.4.1}$$

*That is, $\hat{\beta}$ is a component-wise unbiased estimator for $\beta$; $\mathbb{E}(\hat{\beta}_i) = \beta_i \quad \forall i \in \{0, 1, 2, ..., m\}$.*

*Proof.* By the Theorem 5.3.1, we have that;

$$\hat{\beta} = (X^TX)^{-1}X^Ty.$$

However, $y = X\beta + \epsilon$, thus we have that;

$$\begin{aligned} \hat{\beta} &= (X^T X)^{-1} X^T \cdot (X\beta + \epsilon), \\ &= (X^T X)^{-1} X^T \cdot X\beta + (X^T X)^{-1} X^T \cdot \epsilon, \\ &= (X^T X)^{-1} (X^T X)\beta + (X^T X)^{-1} X^T \cdot \epsilon, \\ &= I\beta + (X^T X)^{-1} X^T \cdot \epsilon, \\ &= \beta + (X^T X)^{-1} X^T \cdot \epsilon. \end{aligned} \tag{5.4.2}$$

Taking the expectation,$\mathbb{E}$ on both sides of Equation (5.4.2) yields;

$$\begin{aligned} \mathbb{E}(\hat{\beta}) &= \mathbb{E}(\beta + (X^T X)^{-1} X^T \cdot \epsilon), \\ &= \beta + (X^T X)^{-1} X^T \underbrace{\mathbb{E}(\epsilon)}_{0}, \\ \mathbb{E}(\hat{\beta}) &= \beta. \end{aligned}$$

□

**5.4.2 Remark.** Similar to the simple linear regression case, in multiple linear regression we can; assess the accuracy of coefficient estimates, estimate confidence intervals, carry out hypothesis testing and also assess the accuracy of the entire model.

**5.4.3 Remark.** When we fit a linear regression model to a given dataset, some problems may arise and these need to be handled carefully. Some of these problems can be dealt with during the data pre-processing stage of machine learning. They include[14];

- Outliers.
- High leverage points.
- Collinearity of predictors.
- Correlation of error terms.
- Non-constant variance of error terms.
- Non-linearity of the response-predictor relationship.

# 6. Conclusion

In this work, we have explored the profound interconnection between mathematics and machine learning, specifically through the lens of linear regression models. We began the study with a comprehensive overview of machine learning and how it is important to our world today. We looked at various mathematical topics that underpin machine learning algorithms and we established a foundation for understanding how mathematical principles are indispensable to the field of machine learning. Both simple and multiple linear regression, served as a focal point to aid us illustrate the theoretical as well as practical applications of mathematical principles and concepts in machine learning. Using data on the pH of pure water, we demonstrated how machine learning models are employed to derive meaningful insights and predictions from data with vast utility of mathematical frameworks. Through the examples provided, we showed how linear algebra facilitates the representation and manipulation of data, how calculus is used to optimize model parameters and how statistics provides the framework for making inferences about data. In our detailed examination of simple and multiple linear regression, we showcased the step-by-step mathematical processes involved in; model formulation, parameter estimation, model evaluation and interpretation of results. Through these examples, we emphasized that mathematics is not just a theoretical foundation but also a practical tool that drives the development and refinement of machine learning models. The mathematical rigor involved in linear regression models ensures that the predictions are not only accurate but also interpretable, which is essential for scientific and industrial applications.

## Future Work

The applications of mathematics in machine learning extend far beyond the examples covered in this work. From the simplest of algorithms to the most complex neural networks, mathematics forms the backbone of machine learning enabling the creation of models that can learn from data and make useful predictions. As the field of machine learning continues to evolve, the integration of advanced mathematical techniques will undoubtedly lead to more sophisticated powerful algorithms and this is a possible orientation for future studies.

# References

[1] Martin Anthony. Aspects of discrete mathematics and probability in the theory of machine learning. *Discrete applied mathematics*, 156(6):883–902, 2008.

[2] Armando D Bedoya, Nicoleta J Economou-Zavlanos, Benjamin A Goldstein, Allison Young, J Eric Jelovsek, Cara O'Brien, Amanda B Parrish, Scott Elengold, Kay Lytle, Suresh Balu, et al. A framework for the oversight and local deployment of safe and high-quality prediction models. *Journal of the American Medical Informatics Association*, 29(9):1631–1636, 2022.

[3] Cécile Bordier, Carlo Nicolini, and Angelo Bifone. Graph analysis and modularity of brain functional connectivity networks: searching for the optimal threshold. *Frontiers in neuroscience*, 11:265813, 2017.

[4] Jason Brownlee. Basics of linear algebra for machine learning. *Machine Learning Mastery*.

[5] Jason Brownlee, Stefania Cristina, and Mehreen Saeed. *Calculus for machine learning*. Machine Learning Mastery, 2022.

[6] Rishabh Choudhary and Hemant Kumar Gianey. Comprehensive review on supervised machine learning algorithms. In *2017 International Conference on Machine Learning and Data Science (MLDS)*, pages 37–43. IEEE, 2017.

[7] Frank Emmert-Streib and Matthias Dehmer. Evaluation of regression models: Model assessment, model selection and generalization error. *Machine learning and knowledge extraction*, 1(1):521–551, 2019.

[8] Bradley J Erickson and Felipe Kitamura. Magician's corner: 9. performance metrics for machine learning models, 2021.

[9] David A Freedman. *Statistical models: theory and practice*. cambridge university press, 2009.

[10] María Teresa García-Ordás, José Alberto Benítez-Andrades, Isaías García-Rodríguez, Carmen Benavides, and Héctor Alaiz-Moretón. Detecting respiratory pathologies using convolutional neural networks and variational autoencoders for unbalancing data. *Sensors*, 20(4):1214, 2020.

[11] Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. " O'Reilly Media, Inc.", 2022.

[12] Bonaccorso Giuseppe. Machine learning algorithms: popular algorithms for data science and machine learning, 2018.

[13] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

[14] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, et al. *An introduction to statistical learning*, volume 112. Springer, 2013.

[15] Adel Javanmard and Andrea Montanari. Confidence intervals and hypothesis testing for high-dimensional regression. *The Journal of Machine Learning Research*, 15(1):2869–2909, 2014.

[16] Hai-Tao Jin, Fei Wang, Wen Zhang, Qi-Lin Liu, Jing-Lan Zhang, Miao Yu, Zhen-Zhen Guo, Wei Pan, et al. Linear regression analysis of sleep quality in people with insomnia in wuhan city during the covid-19 pandemic. *International Journal of Clinical Practice*, 2023, 2023.

[17] Jens Kirchner, Andreas Heberle, and Welf Löwe. Classification vs. regression-machine learning approaches for service recommendation based on measured consumer experiences. In *2015 IEEE World Congress on Services*, pages 278–285. IEEE, 2015.

[18] Xiao-Yang Liu, Hongyang Yang, Qian Chen, Runjia Zhang, Liuqing Yang, Bowen Xiao, and Christina Dan Wang. Finrl: A deep reinforcement learning library for automated stock trading in quantitative finance. *arXiv preprint arXiv:2011.09607*, 2020.

[19] Dastan Maulud and Adnan M Abdulazeez. A review on linear regression comprehensive in machine learning. *Journal of Applied Science and Technology Trends*, 1(2):140–147, 2020.

[20] Bruce D McCullough. Assessing the reliability of statistical software: Part i. *The American Statistician*, 52(4):358–366, 1998.

[21] Vishal Parikh and Parth Shah. Stock prediction and automated trading system. *IJCS*, 6: 104–111, 2015.

[22] Sofia K Pillai, MM Raghuwanshi, and M Gaikwad. Hyperparameter tuning and optimization in machine learning for species identification system. In *Proceedings of International Conference on IoT Inclusive Life (ICIIL 2019), NITTTR Chandigarh, India*, pages 235–241. Springer, 2020.

[23] Anandhan Prathik, K Uma, and J Anuradha. An overview of application of graph theory. *International Journal of ChemTech Research*, 9(2):242–248, 2016.

[24] Jorge Ribeiro, Rui Lima, Tiago Eckhardt, and Sara Paiva. Robotic process automation and artificial intelligence in industry 4.0–a literature review. *Procedia Computer Science*, 181: 51–58, 2021.

[25] Shruthi H Shetty, Sumiksha Shetty, Chandra Singh, and Ashwath Rao. Supervised machine learning: algorithms and applications. *Fundamentals and methods of machine and deep learning: algorithms, tools and applications*, pages 1–16, 2022.

[26] Richard S Sutton, Andrew G Barto, et al. Reinforcement learning. *Journal of Cognitive Neuroscience*, 11(1):126–134, 1999.

[27] Manohar Swamynathan. *Mastering machine learning with python in six steps: A practical implementation guide to predictive data analytics using python*. Springer, 2017.

[28] Hasan Halit Tali and Ceren Çelti. An approach towards the least-squares method for simple linear regression. *Advances in Artificial Intelligence Research*, 2(2):38–44, 2022.

[29] Muhammad Usama, Junaid Qadir, Aunn Raza, Hunain Arif, Kok-Lim Alvin Yau, Yehia Elkhatib, Amir Hussain, and Ala Al-Fuqaha. Unsupervised machine learning for networking: Techniques, applications and research challenges. *IEEE access*, 7:65579–65615, 2019.

[30] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018, 2018.

[31] Yilu Wu. Linear regression in machine learning. In *International Conference on Statistics, Applied Mathematics, and Computing Science (CSAMCS 2021)*, volume 12163, pages 1253–1264. SPIE, 2022.

# APPENDICES

## Python Codes Used In The Work

The python codes used in this work were saved in a repository on GitHub labeled Project_Codes_AIMS-Ghana and can be accessed via the link: https://github.com/RAM456-star/The-Project-Codes.git.