

https://dspace.org/

Monochromatic Arithmetic Progressions: A Machine Learning Approach

Malatsi, Tebogo

2025
University of the Witwatersrand, Johannesburg
https://hdl.handle.net/10539/50273
Malatsi, Tebogo. (2025). Monochromatic Arithmetic Progressions: A Machine Learning Approach [Master's dissertation, University of the Witwatersrand, Johannesburg]. WIReDSpace. https://hdl.handle.net/10539/50273

Downloaded from a DSpace repository. https://dspace.org/

Monochromatic arithmetic progressions: A Machine learning approach

School of Computer Science & Applied Mathematics
University of the Witwatersrand

Tebogo Malatsi
1672358

Supervised by Dr. Jonathan Kariv and Dr. Pravesh Ranchod

August 21, 2025



A research report submitted to the Faculty of Science, University of the Witwatersrand, Johannesburg, in partial fulfilment of the requirements for the degree of Master of Science in e-Science

Abstract

Arithmetic progressions, fundamental in both number theory and combinatorics, have consistently fascinated mathematicians. This thesis explores the application of machine learning models to predict the fraction of monochromatic arithmetic progressions (APs) in randomly colored integer sequences, demonstrating the potential of Artificial Intelligence (AI) to contribute to the field of pure mathematics, particularly extremal combinatorics. Specifically, we investigated the use of Linear Regression, Random Forest, and Multi-Layer Perceptron models to predict these fractions for varying AP lengths. We examined two distinct approaches: training individual models for specific AP lengths, and training combined models that learn across multiple lengths to enable generalization. Our results demonstrated the capability of machine learning to capture complex relationships within colored integer sequences. While models trained on individual AP lengths effectively predicted fractions for those specific lengths, the Multi-Layer Perceptron trained on combined lengths successfully generalized to unseen AP lengths, outperforming simpler models and showcasing its ability to capture underlying patterns. This work strengthened the intersection of AI methods with problems in pure mathematics, particularly extremal combinatorics, and laid the groundwork for future investigations.

Declaration

I, Tebogo Malatsi, hereby declare the contents of this research report to be my own work. This research report is submitted for the degree of Master of Science in e-Science at the University of the Witwatersrand. This work has not been submitted to any other university, or for any other degree.

Acknowledgements

I wish to express my heartfelt appreciation to my supervisors Dr. Jonathan Kariv and Dr. Pravesh Ranchod for their unwavering guidance and support throughout this research. Their combined expertise, insightful feedback, and continuous encouragement have been instrumental in shaping this work. Their dedication and mentorship have refined my ideas, strengthened my research approach, and provided critical direction at every stage. I am truly thankful for their support.

Additionally, I extend my sincere gratitude to DSI-NICIS for their generous funding, without which this research would not have been possible. Their support has been essential in facilitating this study, and for that, I am deeply appreciative.

Contents

Preface
Abstract i
Declaration ii
Acknowledgements iii
Table of Contents iv
List of Figures vi
List of Tables viii

1 Introduction **1**

2 Related Work and Research Background **4**
2.1 Introduction 4
2.2 Related Work 4
2.3 Linear Regression 6
2.3.1 Ordinary Least Squares (OLS) 7
2.4 Random Forest 7
2.4.1 Decision Tree Construction 8
2.5 Artificial Neural Networks 9
2.5.1 Feedforward Neural Network 9
2.5.2 Backpropagation Algorithm 13
2.6 Conclusion 14

3 Research Methodology **15**
3.1 Introduction 15
3.2 Problem Statement 15
3.3 Data Generation 16
3.4 Feature Generation 17
3.5 Testing and Evaluation 21
3.5.1 Evaluation Metrics 21
3.5.2 Feature Importance Analysis 22
3.5.3 Generalization Efforts 22
3.6 Baseline Model 22
3.6.1 Baseline model for $r = 2$ (2 colors) 23
3.6.2 Generalization to r Colors 24
3.7 Study Limitations 25

3.7.1 Illustrative Example: Growth in the Number of Arithmetic Progressions with N . . . 25
3.8 Conclusion . . . 27

4 Monochromatic Arithmetic Progressions: Single Length Models 28
4.1 Linear Regression Models . . . 28
4.1.1 RMSE Performance by Sequence Length . . . 29
4.1.2 Feature Importance . . . 29
4.1.3 Prediction Scatterplots . . . 29
4.2 Random Forest Models . . . 30
4.2.1 Performance by Sequence Length . . . 30
4.2.2 Feature Importance . . . 31
4.2.3 Prediction Scatterplots . . . 31
4.3 Multilayer Perceptron Models . . . 32
4.3.1 Performance by Sequence Length . . . 32
4.3.2 Feature Importance . . . 32
4.3.3 Prediction Scatterplots . . . 33
4.4 Model Performance Evaluation . . . 33
4.4.1 Overall Model Comparison by RMSE . . . 34
4.4.2 Feature Importance Analysis . . . 34
4.4.3 Scatterplot Analysis . . . 34
4.4.4 Discussion . . . 35
4.5 Conclusion . . . 35

5 Monochromatic Arithmetic Progressions: Combined Length Models 36
5.1 Evaluation of Model Performance and Generalization . . . 36
5.2 Results and Discussion . . . 37
5.2.1 Evaluation by RMSE . . . 37
5.2.2 Analysis of Feature Importance . . . 38
5.2.3 Scatter Plots: Predicted vs. Actual Values . . . 40
5.3 Conclusion . . . 42

6 Overall Discussion of Results and Conclusion 43

A Code Repository 46

References 50

List of Figures

2.1 Example of monochromatic arithmetic progressions for $r = 2$, $N = 15$, and $\kappa = 3$. 5
2.2 A single perceptron. 10
2.3 A NN with multiple intermediate layers [Hachimi *et al.* 2020]. 11
2.4 ReLU. 12
2.5 Tanh. 13

3.1 Growth in the number of k-term arithmetic progressions as N increases, for various values of k. 27

4.1 $k = 3$ 29
4.2 $k = 4$ 29
4.3 $k = 5$ 29
4.4 LR feature importances across sequence lengths $k = 3$, $k = 4$, and $k = 5$. 29
4.5 $k = 3$ 30
4.6 $k = 4$ 30
4.7 $k = 5$ 30
4.8 LR prediction vs actual values across sequence lengths $k = 3$, $k = 4$, and $k = 5$ 30
4.9 $k = 3$ 31
4.10 $k = 4$ 31
4.11 $k = 5$ 31
4.12 RF feature importances across sequence lengths $k = 3$, $k = 4$, and $k = 5$. 31
4.13 $k = 3$ 31
4.14 $k = 4$ 31
4.15 $k = 5$ 31
4.16 RF prediction vs actual values across sequence lengths $k = 3$, $k = 4$, and $k = 5$ 31
4.17 $k = 3$ 33
4.18 $k = 4$ 33
4.19 $k = 5$ 33
4.20 MLP feature importances across sequence lengths $k = 3$, $k = 4$, and $k = 5$ 33
4.21 $k = 3$ 33
4.22 $k = 4$ 33
4.23 $k = 5$ 33

4.24 MLP prediction vs actual values across sequence lengths $k = 3$, $k = 4$, and $k = 5$ 33

5.1 Overall Analysis of Feature Importance Distributions for Linear Regression across $k = 3, 4, 5, 6,$ and 7 39
5.2 $k = 3$ 39
5.3 $k = 4$ 39
5.4 $k = 5$ 39
5.5 $k = 6$ 39
5.6 $k = 7$ 39
5.7 Overall Analysis of Feature Importance Distributions for Random Forest across $k = 3, 4, 5, 6,$ and 7 39
5.8 $k = 3$ 39
5.9 $k = 4$ 39
5.10 $k = 5$ 39
5.11 $k = 6$ 39
5.12 $k = 7$ 39
5.13 Overall Analysis of Feature Importance Distributions for Multi Layer Perceptron across $k = 3, 4, 5, 6,$ and 7 40
5.14 $k = 3$ 40
5.15 $k = 4$ 40
5.16 $k = 5$ 40
5.17 $k = 6$ 40
5.18 $k = 7$ 40
5.19 Overall Comparison of Linear Regression Predictions vs. Actual Values for $k = 3, 4, 5, 6,$ and 7 40
5.20 $k = 3$ 40
5.21 $k = 4$ 40
5.22 $k = 5$ 40
5.23 $k = 6$ 40
5.24 $k = 7$ 40
5.25 Overall Comparison of Random Forest Predictions vs. Actual Values for $k = 3, 4, 5, 6,$ and 7 41
5.26 $k = 3$ 41
5.27 $k = 4$ 41
5.28 $k = 5$ 41
5.29 $k = 6$ 41
5.30 $k = 7$ 41
5.31 Overall Comparison of Multi Layer Perceptron Predictions vs. Actual Values for $k = 3, 4, 5, 6,$ and 7 41
5.32 $k = 3$ 41
5.33 $k = 4$ 41
5.34 $k = 5$ 41
5.35 $k = 6$ 41
5.36 $k = 7$ 41

List of Tables

4.1 Linear Regression performance across sequence lengths 29
4.2 Random Forest performance across sequence lengths 30
4.3 MLP performance across sequence lengths 32

5.1 Overall RMSE Comparison for Linear Regression, Random Forest, and MLP Models 37

Chapter 1

Introduction

The field of extremal combinatorics is a significant and vibrant area of mathematical research, focusing on understanding the maximum or minimum size of a collection of finite objects that meet certain criteria. This discipline, which deals with problems related to determining the extremal structure of sets, graphs, and other combinatorial configurations, has profound connections to various branches of mathematics, including number theory, geometry, and computer science [Alon 2020].

One of the central themes in extremal combinatorics is to explore how large or small a structure can be while satisfying given properties. Classic problems in this field include finding the largest subset of integers with no three-term AP [Dybizbański 2012] or finding the greatest number of edges in a graph with n-vertices that does not include a complete subgraph (clique) of a certain size [Turán 1941]. These problems not only advance theoretical understanding but also have practical implications in areas such as coding theory, network design, and algorithm optimisation.

The study of extremal combinatorics is essential for several reasons. Firstly, it advances the frontiers of pure mathematical research, resulting in the discovery of new principles and methods. For instance, results from extremal combinatorics have been instrumental in solving longstanding problems in graph theory and set theory, providing deep insights into the structure and behavior of complex systems [Turán 1941]. Second, it intersects with probabilistic combinatorics, where randomness and probability distributions on discrete structures offer a powerful toolkit for solving deterministic problems [Alon and Spencer 2016].

Artificial Intelligence (AI) has recently emerged as a transformative tool in many scientific domains, including mathematics. [Charton *et al.* 2024] introduced PatternBoost, a novel method combining classical search algorithms with transformer neural networks to find interesting constructions in mathematics. Their approach has yielded impressive results in extremal combinatorics, including the solution to long-standing open problems and the construction of counterexamples to conjectures that had remained open for decades. Furthermore, [Mundinger *et al.* 2025] showcased the power of neural networks in driving mathematical discovery by tackling the Hadwiger-Nelson problem, a long-standing open problem in discrete geometry and combinatorics. Their

approach, reformulating the problem as an optimization task with a differentiable loss function, led to the discovery of novel colorings, demonstrating the potential of AI to provide new insights into challenging mathematical questions. The application of AI to pure mathematical problems is a burgeoning area of research that holds great promise [Davies *et al.* 2021]. While machine learning has traditionally been applied to data-rich areas such as image recognition and natural language processing, its potential in pure mathematics, particularly in combinatorial optimization and extremal problems, is increasingly being recognized. AI can assist in pattern recognition and conjecture formulation, thus providing a new lens through which to tackle complex mathematical challenges [Davies *et al.* 2021].

Despite the powerful capabilities of machine learning, its application to extremal combinatorics or pure mathematics remains relatively unexplored. This gap presents a unique opportunity to leverage AI for advancing mathematical research. By applying machine learning techniques to extremal combinatorial problems, researchers can uncover new patterns, generate novel conjectures, and potentially discover new extremal structures that were previously unknown [Parczyk *et al.* 2022].

A key theoretical result that motivates this study is Van der Waerden's theorem, which guarantees that for any integers r and k, there exists an integer N such that every r-coloring of the set $[1, 2, \cdots, N]$ contains a monochromatic arithmetic progression of length k [van 1927]. This foundational result illustrates the inevitability of structure within colorings and underpins the study of monochromatic patterns in extremal combinatorics. A more detailed discussion is presented in Chapter 2.

This research investigates whether machine learning models can predict the proportion of monochromatic k-term arithmetic progressions (k-APs) in randomly colored sequences of integers. Specifically, given a binary coloring (red/blue) of the integers $[1, 2, \cdots, N]$, can machine learning models accurately estimate the fraction of k-APs that are monochromatic, using features that capture the structural patterns of the coloring?

The proportion of monochromatic k-APs refers to the ratio between the number of k-term arithmetic progressions in a given coloring where all terms share the same color, and the total number of k-term arithmetic progressions in that sequence. In this study, the total sequence length N is fixed at 100, and the coloring uses two colors (red and blue). For each $k \in \{3, 4, 5, 6, 7\}$, the set of all k-APs within $[1, \cdots, N]$ is fixed and deterministic. The variation arises from the random colorings, which affect how many of those k-APs are monochromatic. Therefore, for each sample, the proportion value lies in $[0, 1]$ and serves as the target in the regression task. This is explained in detail in (3.4).

This thesis explores the following core questions:

- Can machine learning models accurately predict the proportion of monochromatic k-APs in randomly generated 2-colorings?
- Which hand-crafted features (e.g., transition frequency, popular color proportion, anti-symmetry) contribute most to prediction performance?

- How do different regression models: Linear Regression, Random Forests, and Multi-Layer Perceptrons, compare in effectiveness?
- Can a model trained on certain progression lengths generalize to unseen progression lengths?

To answer these questions, the study generates a large synthetic dataset of random 2-colorings of the integer interval $[1, \cdots, N]$. For each coloring, it computes interpretable features (e.g., transition counts, symmetry measures) and the exact proportion of monochromatic k-APs. Supervised regression models are then trained to predict this proportion. The models are evaluated in two settings:

- *Single-length modeling*: models are trained and tested on a fixed value of k, such as 3, 4, and 5.
- *Cross-length generalization*: models are trained on multiple/combined k-values (e.g., $3, 4, 5, 7$) and tested on an unseen value (e.g., 6).

This research makes the following contributions:

- A novel application of supervised machine learning to the study of monochromatic arithmetic progressions in extremal combinatorics.
- A set of interpretable, hand-engineered features based on the structural properties of colorings.
- An empirical comparison of three ML models across multiple progression lengths and configurations.
- A generalization experiment to assess whether models trained on known k-values can predict unseen progression lengths.

The structure of this research is as follows: Chapter 2 presents the background and relevant studies, including an introduction to monochromatic arithmetic progressions (APs) and the application of Linear Regression (LR), Random Forest (RF), and Multi Layer Perceptrons (MLP) in predictive modeling. Chapter 3 outlines the methodology, detailing data generation, feature extraction, and model evaluation. Chapter 4 examines the performance of single-length models for various lengths and compares their predictive accuracy to a baseline model. Chapter 5 integrates the different lengths into a single model to evaluate its ability to generalize to new data. Lastly, Chapter 6 concludes the report with a summary of the main findings and discusses the research's significance and impact.

Chapter 2

Related Work and Research Background

2.1 Introduction

This chapter delves into the convergence of extremal combinatorics with machine and deep learning, exploring their combined impact on mathematical research. We also introduce essential machine learning and deep learning algorithms, such as LR, RF, ANN, and review related studies and research contributions in these fields.

2.2 Related Work

Monochromatic arithmetic progressions (APs) are a significant area of study within combinatorics and Ramsey theory. Ramsey theory, named after the British mathematician and philosopher Frank P. Ramsey, is a branch of the mathematical field of combinatorics that focuses on the emergence of ordered substructures in systems of a given size. Problems in Ramsey theory typically ask a question of the form: "how large must a structure be to guarantee a specific property?" [Graham and Butler 2015] This principle underlies seminal theorems like Van der Waerden's on arithmetic progressions.

According to Van der Waerden's theorem, for any positive integers k and r, there exists a number N such that every r-coloring of $[N] := \{1, 2, \ldots, N\}$ has a monochromatic k-AP [van 1927]. This means that we are coloring each element of the set $[N]$ using r different colors. An r-coloring is simply a function that assigns one of the r colors to each element of the set. An AP of length κ is a sequence that follows the format,

$$a, a + d, a + 2d, \ldots, a + (k - 1)d, \tag{2.1}$$

where a and d are integers and d not equal to zero. This is called a monochromatic progression if all the elements of the progression are the same color. This theorem laid the groundwork for subsequent studies into the properties and bounds of such progressions. Figure 2.1 present an example for the case where the coloring $r = 2$

(with red and blue colors) and $N = 15$ (indicating that the integers range from 1 to 15), with $k = 3$ representing monochromatic arithmetic triples.

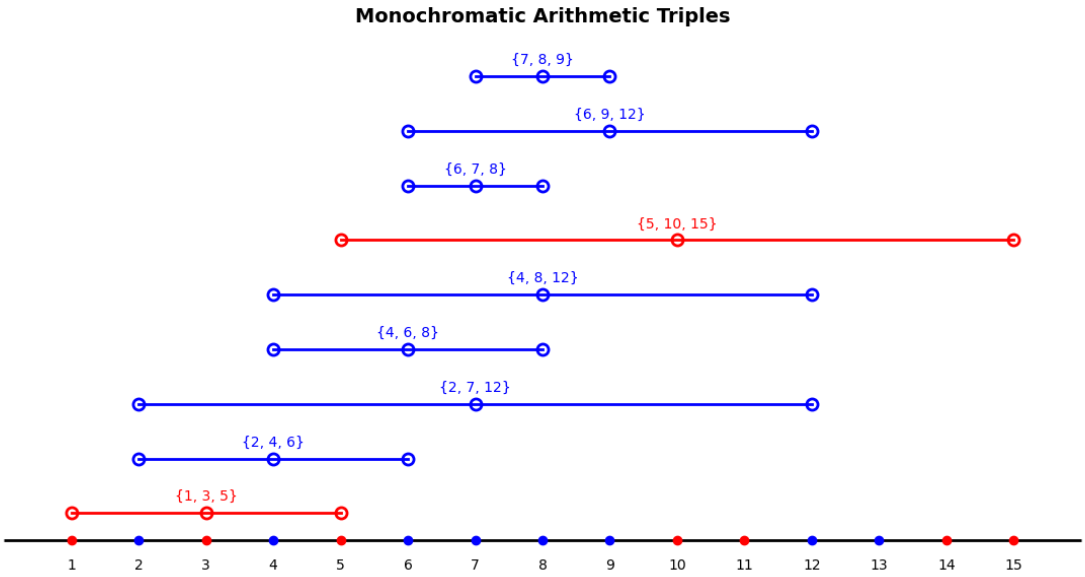


Figure 2.1: Example of monochromatic arithmetic progressions for $r = 2$, $N = 15$, and $\kappa = 3$.

Szemerédi's theorem extends Van der Waerden's theorem by asserting that any subset of the integers with positive upper density includes arbitrarily long APs [Vijay 2012]. Closely related to Szemerédi's theorem is the Erdős-Turán conjecture, which posits that any subset of the natural numbers with positive density contains infinitely many arithmetic progressions of any given length. Although still unresolved in its full generality, it has driven considerable research efforts in the field [Erdös and Turán 1936]. The authors in [Greenwood *et al.* 2024] demonstrated that a specific 2-coloring of the integers $[N]$ effectively minimizes the number of monochromatic 3-term APs under certain conditions. In their research, [Aedo *et al.* 2022] investigated the occurrence of long monochromatic arithmetic progressions within the Thue-Morse word and a class of generalized Thue-Morse words. They provided exact values or upper bounds for the lengths of such progressions with fixed differences, connecting the existence of these long progressions to the structure of the underlying substitution. For instance, they showed that in the Thue-Morse word, which is generated by a simple substitution rule, there exist arbitrarily long monochromatic APs with specific differences. Their work contributes to the broader understanding of how specific colorings, even those generated by simple rules, influence the distribution of monochromatic APs, a central theme in extremal combinatorics.

Traditionally, pure machine learning techniques have been underutilized in solving pure mathematics problems. However, recent studies indicate a growing interest in this interdisciplinary approach, highlighting its potential to revolutionize mathematical research. Machine learning can be applied to complex mathematical challenges through various methods, authors in [Parczyk *et al.* 2022] examined the problem of determin-

ing constructive bounds by framing it as a discrete optimization problem. They evaluated the effectiveness of various methods, including metaheuristics and reinforcement learning, in solving these types of problems. In their study, [Bello *et al.* 2016] introduced a framework leveraging neural networks and reinforcement learning to address combinatorial optimization problems. Specifically, they targeted the traveling salesman problem (TSP). By training a RNN, their approach predicts the distribution of various city permutations based on the input set of city coordinates. In a notable study [Wagner 2021], researchers demonstrated the potential of AI in uncovering counterexamples to open conjectures. They utilized a reinforcement learning algorithm, specifically the deep cross-entropy method, the study showcased how AI can be effectively employed to generate explicit constructions and identify counterexamples in the realms of extremal combinatorics and graph theory. This approach highlights the capability of advanced AI techniques to address complex mathematical challenges by providing novel solutions and insights into longstanding conjectures.

2.3 Linear Regression

LR is a cornerstone of statistical modeling and machine learning, tracing its origins back to the early 19th century [Stigler 1990; Montgomery and Runger 2010]. The method provides a strong tool for understanding and predicting relationships between variables. Its widespread use across various fields, from economics [Woolridge 2009] to engineering [Montgomery and Runger 2010] and social sciences, underscores its importance.

The development of LR can be attributed to several pioneering figures in statistics and mathematics. The concept of the "least squares" method, which is fundamental to linear regression, was first established by Adrien-Marie Legendre in 1805 and later independently by Carl Friedrich Gauss in 1809 [Farebrother 2018; Legendre 1806; Gauss 1877]. In order to determine the best-fitting line across a set of data points, Legendre's technique concentrated on minimising the sum of the squares of the residuals.

Karl Pearson further developed the theory of linear regression and correlation in the late 19th and early 20th centuries. His work laid the groundwork for modern statistical techniques and provided a formal framework for regression analysis[Porter 2010].

The basic form of a LR model is:

$$y = \eta_0 + \eta_1 z_1 + \eta_2 z_2 + \ldots \eta_p z_p + \varepsilon. \tag{2.2}$$

In this context:

- y denotes the response variable,
- $z_1, z_2, \ldots, z_p$ represent the predictor variables,
- $\eta_0, \eta_1, \ldots, \eta_p$ represent the regression coefficients,
- ε denotes the residual term, capturing the disparity between observed and predicted values.

The objective of LR is to find the coefficients (η) that best describe the interaction between predictors and the variable being predicted. The OLS approach, which minimizes the sum of squared residuals, is commonly used to do this. [Weisberg 2005].

Linear regression analysis relies on several key assumptions to guarantee the model's validity and reliability:

- **Linearity:** The connection between the response and predictor variables follows a linear pattern.
- **Independence:** The observations are unrelated to one another.
- **Homoscedasticity:** The variability of residuals remains consistent across all levels of the predictor variables.
- **Normality:** The model's errors, or residuals, have a normal distribution.

Breach of these presumptions may result in skewed or ineffective approximations, hence affecting the predictive capacity of the model.

2.3.1 Ordinary Least Squares (OLS)

The most popular method for estimating a LR model's parameters is the OLS approach. The OLS method aims to reduce the sum of squared discrepancies between observed and predicted values [James *et al.* 2013; Montgomery and Runger 2010]. The function we aim to minimize is:

$$\sum_{i=1}^{n}(q_i - \hat{q}_i)^2, \tag{2.3}$$

where q_i is the observed value and $\hat{q}_i$ is the predicted value for the i-ith observation. In matrix notation, the LR model can be expressed as:

$$\mathbf{q} = \mathbf{X}\eta + \varepsilon, \tag{2.4}$$

where $\mathbf{q}$ is the vector of observed values, $\mathbf{X}$ represents the matrix of independent variables, and η denotes the regression coefficients vector. The OLS estimate of the regression coefficients is given by

$$\hat{\eta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{q}. \tag{2.5}$$

Extensions of LR, such as multiple regression, polynomial regression, and regularized regression methods (e.g., Ridge and Lasso), enhance its utility by addressing more complex scenarios and mitigating issues like over-fitting[Hastie *et al.* 2009].

2.4 Random Forest

RF is an ensemble learning approach primarily applied to both classification and regression tasks. Introduced by Leo Breiman in 2001 [Breiman 2001], Random Forest trains numerous decision trees and blends them to get more precise and reliable forecasts. This method leverages bagging (Bootstrap Aggregating) and randomness in feature selection, making it a powerful tool in machine learning. Random Forests has several

advantages, including the ability to handle categorical and missing data, high accuracy facilitated by its capacity to process nonlinear and high-dimensional data without requiring feature selection or standardization.

The development of Random Forests can be traced back to decision trees and ensemble methods. Decision trees, a fundamental component of Random Forest, were popularized by J. Ross Quinlan in the 1980s through his work on the ID3 algorithm [Quinlan 1986]. However, the concept of combining multiple models to improve performance was pioneered earlier by statisticians such as Bradley Efron with the bootstrap method in 1979 [Efron 1979].

The Random Forest algorithm, introduced by Leo Breiman and Adele Cutler in their seminal paper "Random Forests" in 2001 [Breiman 2001], relies on combining multiple decision trees. Breiman demonstrated how this technique improves prediction accuracy while reducing the likelihood of overfitting.

2.4.1 Decision Tree Construction

A decision tree is built by recursively partitioning the data according to feature values to minimize a certain criterion. For regression trees, the criterion often used is the MSE:

$$MSE = \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} (q_i - \hat{q}_i)^2. \tag{2.6}$$

The goal is to select splits that reduce the MSE the most. Let S be the set of data points and S_L and S_R be the subsets resulting from a split. The optimal split minimizes the sum of the weighted MSE of the resulting subsets:

$$\textit{Split criterion} = \frac{|S_L|}{|S|} MSE(S_L) + \frac{|S_R|}{|S|} MSE(S_R). \tag{2.7}$$

At each node, a decision tree chooses a random selection of features, which are then used to find the best split. This randomness ensures that the trees are different and lowers the correlation among individual trees, resulting in a stronger ensemble.

Bootstrap Sampling

Bootstrap sampling involves generating multiple datasets by sampling with replacement from the original dataset. If the original dataset has N observations, each bootstrap sample will also have N observations, but some observations may be repeated while others may be omitted. This process introduces variability into the training data for each tree. Given the original dataset $D = \{(x_i, q_i\}_{i=1}^{\mathcal{N}}$, a bootstrap sample D^* is created by drawing N observations from D with replacement.

Aggregation of Predictions

In regression problems, the average of all individual trees' predictions yields the final prediction of a Random Forest. For example, if there are M trees in the forest, the

ultimate prediction for a specific input is provided by:

$$\hat{q} = \frac{1}{\mathcal{M}} \sum_{m=1}^{\mathcal{M}} \hat{q}^m(x) \tag{2.8}$$

where $\hat{q}^m(x)$ is the prediction for the m-th tree.

Model Evaluation

Metrics like R^2 or MSE can be used to assess how well a Random Forest regressor performs. One of the advantages of Random Forests is that it also provides a mechanism for assessing the importance of each feature. One common method is to measure the total decrease in node impurity (e.g., MSE for regression) brought by a feature over all trees.

2.5 Artificial Neural Networks

ANNs function as information processing systems that emulate biological nervous systems and brain structure [Bahrammirzaee 2010]. Unlike traditional linear models, ANNs are nonlinear models that can predict continuous output values by modeling complex relationships between inputs and outputs. This enables them to capture and predict intricate patterns within the data, making them particularly effective for regression tasks where there are non-linear correlations between the variables[Malhotra and Malhotra 2003]. ANNs' ability to model these complex relationships allows them to provide more accurate and robust predictions for a wide range of regression problems.

2.5.1 Feedforward Neural Network

Perceptron

One kind of artificial neural network used for regression is the single-layer perceptron, first introduced by [Rosenblatt 1958]. It consists of a layer of input nodes connected to an output node. The output node applies an activation function after calculating the weighted sum of the inputs, typically a linear or continuous function, to produce a continuous output value. This output represents the predicted value of the dependent variable in a regression problem. Diagrammatically, a simple perceptron used for regression can be represented as follows (2.2):

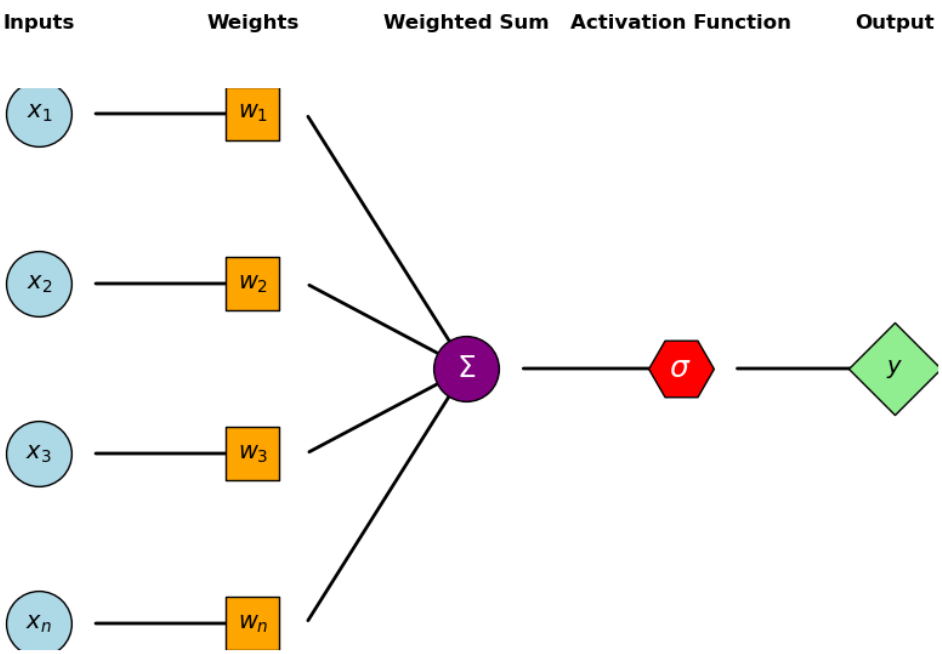


Figure 2.2: A single perceptron.

Weighted sum for the output node:

The inputs to the output node are summed together, with each input being assigned a specific weight

$$z = \sum_{i=1}^{N} w_i x_i + b \tag{2.9}$$

where:
- The weight that links the ith input node to the output node is represented by w_i.
- The value of the ith input node is x_i
- b is the bias term for the output node.

Output node activation function:

The output of the perceptron, y, is obtained by applying the activation function f to the weighted sum z:

$$y = f(z) \tag{2.10}$$

The function $f(z)$ is defined as follows: it equals one if z is greater than θ and equals zero if z is less than or equal to θ.

Multilayer Perceptron

An input layer, one or more hidden layers, and an output layer make up a multi-layer perceptron. Neurons within each layer receive inputs, calculate a weighted sum, apply

an activation function, and transmit the outcome to the subsequent layer. Figure 2.3 depicts a network with multiple layers.

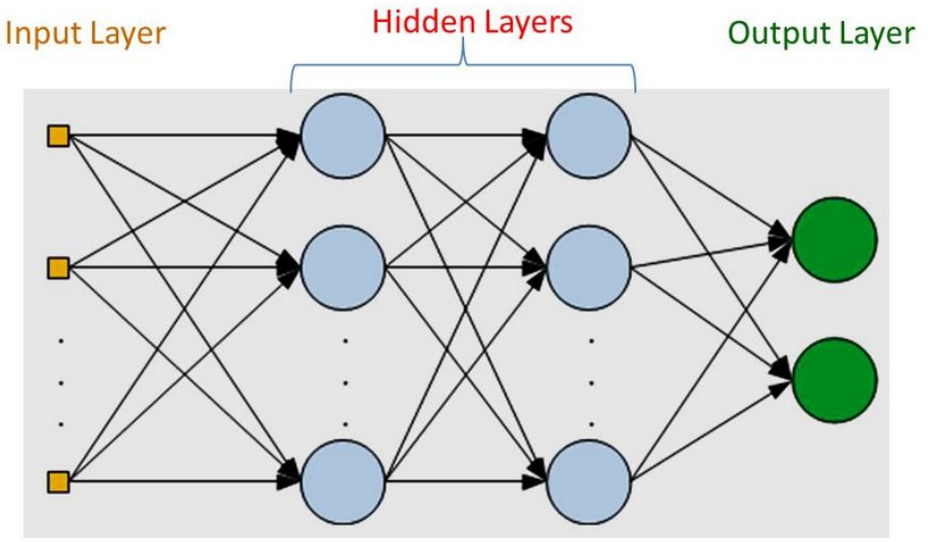


Figure 2.3: A NN with multiple intermediate layers [Hachimi *et al.* 2020].

Neuron's Weighted Sum (for layer l):

For a neuron j in layer l, the weighted sum $z_j^{(l)}$ of its inputs is given by:

$$z_j^{(l)} = \sum_{i=1}^{N^{(l-1)}} w_{ji}^{(l)} a_i^{(l-1)} + b_j^{(l)} \tag{2.11}$$

where:
-$w_{ji}^{(l)}$ is the weight between neuron i in layer $l-1$ and neuron j in layer l.

- The activation value of neuron i in layer $l-1$ is $a_i^{(l-1)}$.
- $b_j^{(l)}$ represents the bias term for neuron j in layer l.

Activation Function (for layer l):

The activation value $a_j^{(l)}$ for neuron j in layer l is obtained by applying the activation function $f^{(l)}$ to the weighted sum $z_j^{(l)}$:

$$a_j^{(l)} = f^{(l)}(z_j^{(l)}) \tag{2.12}$$

Output Layer (Layer L):

To determine the output y_k for a certain regression value, the activation function $f^{(L)}$ is applied to the weighted sum $z_k^{(L)}$ of the output layer neurons in the output layer (layer L):

$$y_k = f^{(L)}(z_k^{(L)}). \tag{2.13}$$

Activation Functions

In ANNs, activation functions are pivotal as they bring about essential nonlinear transformations to the output of neurons or layers. These functions enable neural networks to model complex relationships in data, facilitating the solving of non-linear problems through the network's ability to learn intricate patterns, making them a fundamental component in the learning process of modern machine learning models. In this project, our attention will be directed toward two activation functions, namely, the Tanh and ReLU activation functions.

ReLU:

The ReLU function is characterized by its piecewise linear structure, directly yielding the input if positive, and otherwise producing an output of zero [Agarap 2018; Dubey *et al.* 2022]. In neural networks, especially CNNs and Multilayer Perceptrons, it is the most often employed activation function[Agarap 2018]. This makes it computationally efficient, which can accelerate the training process. Additionally, ReLU helps to mitigate the vanishing gradient problem. This, in DNNS, may be a major problem. [Glorot *et al.* 2011]. Mathematically, it can be described as:

$$f(x) = \max(0, x). \tag{2.14}$$

When x is non-negative, the function outputs x; if x is negative, the function yields 0. The ReLU activation function is shown in Figure 2.4.

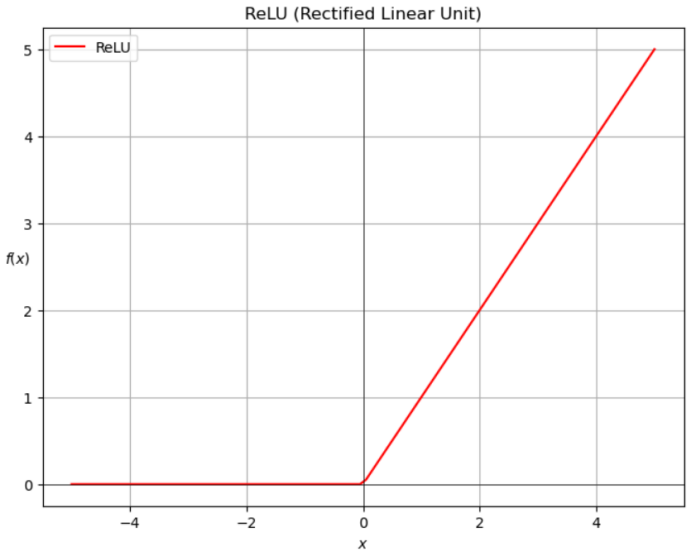


Figure 2.4: ReLU.

Tanh:

The Tanh function is an activation function that scales the input to a range between minus one and one. This can be particularly useful when modeling data that exhibits a non-linear relationship [Ma and Tang 2021]. Furthermore, like ReLu, Tanh also introduces non-linearity into the model, which can boost the model's performance.

Figure 2.5 depicts the Tanh activation function.

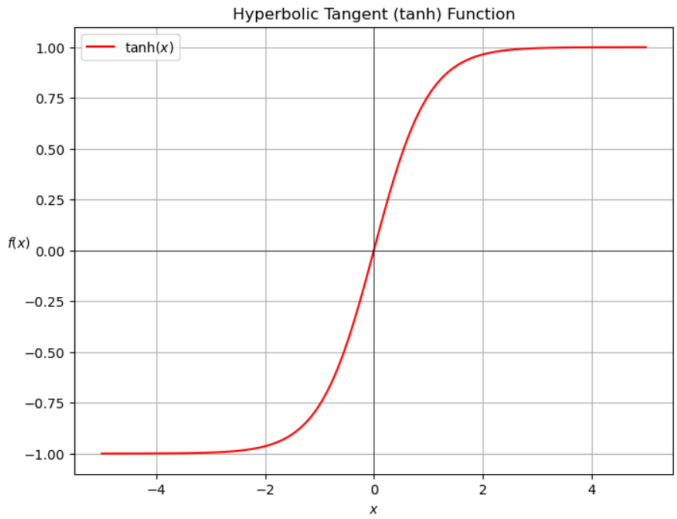


Figure 2.5: Tanh.

2.5.2 Backpropagation Algorithm

The backpropagation algorithm is used for training feedforward neural networks [Rumelhart *et al.* 1986]. It aims to minimize a cost function C by adjusting the weights and biases. We assume that the cost function is the MSE (2.6) between the predicted outputs and the target outputs for illustrative purposes. The key equations in backpropagation are as follows:

Error (Output Layer):

The error E_k at the output layer for neuron k is computed as the the predicted output y_k minus the target t_k:

$$E_k = y_k - t_k. \tag{2.15}$$

Error (Hidden Layers):

For hidden layers l, the error $E_j^{(l)}$ for neuron j is computed by propagating the error from the next layer:

$$E_j^{(l)} = (\sum_k w_{kj}^{(l+1)} E_k^{(l+1)}) \cdot f'^{(l)}(z_j^{(l)}) \tag{2.16}$$

where $f'^{(l)}(z_j^{(l)})$ is the derivative of the activation function in layer l with respect to $z_j^{(l)}$.

Weight and Bias Updates:

The biases and weights are subject to updating using the calculated errors and the learning rate η:

$$w_{ji}^{(l)} \leftarrow w_{ji}^{(l)} - \eta \cdot \frac{\partial C}{\partial w_{ji}^{(l)}} \tag{2.17}$$

$$b_{j}^{(l)} \leftarrow b_{j}^{(l)} - \eta \cdot \frac{\partial C}{\partial b_{j}^{(l)}} \tag{2.18}$$

The derivatives are computed using the chain rule.

These equations form the basis of the backpropagation algorithm, which iteratively adjusts the network's weights and biases to minimize the cost function C.

2.6 Conclusion

This chapter thoroughly explored the foundational theories and current research landscape of monochromatic arithmetic progressions within combinatorics and Ramsey theory. This included delving into fundamental theorems such as Van der Waerden's and Szemerédi's, which form the bedrock of understanding for the existence and properties of these progressions in colored integer sets.

Additionally, we delved into the application of advanced computational techniques, specifically machine and deep learning methodologies like LR, RF, and ANN. We discussed the background and basic principles of these algorithms, highlighting their utility in modeling relationships between variables and in constructing robust predictive models.

Chapter 3

Research Methodology

3.1 Introduction

This segment scrutinizes the methodologies and processes employed in generating and analyzing the data for this study. It begins with a detailed explanation of how the dataset of randomly colored integers is generated, ensuring a robust foundation for subsequent analysis. The focus then shifts to the feature generation process, where various properties of these colorings are identified and extracted to serve as inputs for predictive modeling. The chapter proceeds with the testing and evaluation phase, describing how the predictive capabilities of the machine learning models are assessed. Finally, the limitations of the study are discussed, providing a comprehensive outline of the potential challenges and constraints that might be encountered during the research.

3.2 Problem Statement

The goal of this research is to explore the application of machine learning (ML) and deep learning (DL) algorithms, specifically LR, RF, and MLP, to predict the proportions of monochromatic arithmetic progressions that emerge from randomly coloring the interval $\{1, 2, \ldots, N\}$.

While traditional combinatorial number theory approaches, such as exhaustive enumeration and probabilistic methods, are commonly employed to study mathematical phenomena like monochromatic arithmetic progressions, they often fall short when analyzing complex or large-scale problems. Recently, there has been an increasing interest in using AI techniques to identify and model intricate patterns within mathematical structures [He 2024]. However, the application of these methods to pure mathematics, particularly in the context of combinatorial number theory, remains underexplored.

This research seeks to answer whether ML and DL models can be effectively used to predict these proportions. Specifically, it aims to investigate whether LR, RF, and ANN can accurately predict the occurrence of monochromatic arithmetic progressions in randomly colored intervals and whether specific properties of the interval coloring can be

identified and quantified as significant factors in these predictions. By bridging the gap between theoretical combinatorics and predictive modeling, this study tries to demonstrate the potential of AI in advancing the understanding of combinatorial problems in pure mathematics.

AI has revolutionized fields such as computer vision, natural language processing, and data analytics, demonstrating its predictive capabilities in various domains [LeCun *et al.* 2015; Goodfellow *et al.* 2016]. However, within pure mathematics, and specifically combinatorial number theory, these tools have not been fully utilized. Studies have shown that AI can be applied to analyze mathematical conjectures and predict potential outcomes, highlighting its adaptability in theoretical contexts [Davies *et al.* 2021; Raayoni *et al.* 2021]. By applying machine learning algorithms to predict monochromatic arithmetic progressions, this research seeks to integrate AI techniques into the study of combinatorial number theory.

3.3 Data Generation

To generate the dataset for this study, a systematic approach is employed to create random colorings of integers within a specified range. The process begins by setting a seed value to ensure reproducibility. Each integer within the range from 1 to N is randomly assigned one of two colors, typically represented as "blue" or "red". This random assignment is conducted independently for each integer, ensuring that each coloring is unique and reflects a random distribution of colors across the interval.

For arbitrary integers and colors, the method involves iterating through each integer up to N and assigning a color randomly selected from a predefined set of colors. This process is repeated to generate the desired number of samples for the dataset. Each sample represents a complete coloring of the integers from 1 to n, encapsulated in a structured format suitable for subsequent analysis.

The dataset generation is designed to handle large-scale data requirements by batching the creation of samples. This approach ensures efficiency and scalability, facilitating the generation of a substantial dataset suitable for training and evaluating machine learning and deep learning models.

Each sample in the dataset corresponds to a complete coloring of the integers from 1 to $N = 100$. For every coloring, a set of handcrafted structural features is computed, which form the input to the machine learning models.

For the *single-length models*, the input vector consists of the following five features, which are explained in detail in the following subsection:

- Proportion of blue and red integers
- Blockiness values
- Transition frequency
- Most popular color proportion

- Anti-symmetry

For the *combined-length models*, the same five features are used, with the addition of:

- Progression length k
- Transition frequency deviation
- Anti-symmetry deviation

Each input is thus represented as a feature vector of shape $(5,)$ for single-length models, or $(8,)$ for combined-length models. The target variable is a single scalar value between 0 and 1, representing the proportion of monochromatic k-term arithmetic progressions in the given coloring.

An example input-target pair for a single-length model might look like:

$$\text{Input} = [0.51, 0.63, 0.24, 0.51, 0.17], \quad \text{Target} = 0.076$$

All samples are stored in structured arrays or tabular format, where each row contains the input features and the corresponding target value.

3.4 Feature Generation

In the feature generation section of this research, the focus is on identifying predictive properties within the randomly generated colorings of integers from 1 to n. The objective is to explore various characteristics of these colorings that could potentially correlate with the proportion of monochromatic arithmetic progressions, which serves as the primary outcome measure or "payoff" in this study. Below is a description of the features that were generated for the purpose of this study.

Anti-Symmetry [Greenwood *et al.* 2024]

Let $S = \{s_1, s_2, \ldots, s_n\}$ be a sequence of colors, where each s_i is an element of the sequence (representing a color). The *opposite elements* in the sequence are pairs of elements such that one element is at index i and the other at index $n-1-i$ where i is an index in the first half of the sequence, and $n-1-i$ is the corresponding opposite index in the second half of the sequence.

For a sequence of length n, there are $\frac{n}{2}$ opposite pairs if n is even (if n is odd, the middle element doesn't form an opposite pair) .

Given the total opposite pairs, the antisymmetry feature can be defined as the fraction of those pairs where the colors at the opposite indices are different. Specifically, for each pair $(i, n-1-i)$, if the color at index i is different from the color at index $n-1-i$ (i.e., $s_i \neq s_{n-1-i}$), then it is counted as an antisymmetric pair.

Let A represent the number of opposite pairs where the colors are different. The antisymmetry feature is then the ratio of these antisymmetric pairs to the total number of

opposite pairs:

$$f_{anti_symmetry} = \frac{A}{\frac{n}{2}} = \frac{A}{Total\ opposite\ pairs} \tag{3.1}$$

where A is the number of opposite pairs with different colors and $\frac{n}{2}$ is the total number of opposite pairs in the sequence.

Proportion of Most Popular Color

Let $S = \{s_1, s_2, \ldots, s_n\}$ be a sequence of colors, where each s_i represents a color from a set of possible colors. The goal is to identify the most frequently occurring color (the most popular color) in the sequence and calculate its proportion.

The frequency of each color in the sequence is given by the number of times that color appears in the sequence. Mathematically, for a color c, its frequency $f(c)$ is:

$$f(c) = count\ of\ color\ c\ in\ the\ sequence. \tag{3.2}$$

The proportion of the more popular color is the ratio of the frequency of the most popular color to the total number of elements in the sequence. That is

$$p_{max} = \frac{f(c_{max})}{n} \tag{3.3}$$

where $f(c_{max})$ is the frequency of the most popular color and n is the total number of elements in the sequence.

Example: Consider the sequence $S = \{red, blue, red, red\}$. The frequencies of each color are $red = 3$ and $blue = 1$. The most popular color is "red", with a frequency of 3. The total number of elements is 4, so the proportion of the more popular color is:

$$p_{max} = \frac{3}{4}.$$

Color Transition Frequency

Let $S = \{s_1, s_2, \ldots, s_n\}$ be a sequence of colors, where each s_i represents a color at position i in the sequence. A color transition occurs between two consecutive elements s_i and s_{i+1} if the colors are different, i.e., $s_i \neq s_{i+1}$.

The total number of transitions is simply the count of consecutive pairs (s_i, s_{i+1}) for which the colors differ. Mathematically, the number of transitions T can be defined as:

$$T = \sum_{i=1}^{n-1} \mathcal{I}(s_i \neq s_{i+1}), \tag{3.4}$$

where $\mathcal{I}$ is the indicator function that is 1 when the condition $s_i \neq s_{i+1}$ is true (i.e., the colors differ), and 0 otherwise.

The color transition frequency is the proportion of transitions in the sequence, relative to the total number of consecutive pairs in the sequence. Given that there are $n - 1$

possible consecutive pairs in a sequence of length n, the color transition frequency $f_{transition}$ is:

$$f_{transition} = \frac{T}{n-1}, \tag{3.5}$$

where T is the total number of transitions, and $n-1$ is the total number of consecutive pairs.

Example: Consider the sequence $S = \{red, blue, blue, red, red\}$. The transitions are as follows:

- From "red" to "blue" (transition),
- From "blue" to "blue" (no transition),
- From "blue" to "red" (transition),
- From "red" to "red" (no transition).

There are 2 transitions in total and the sequence has 5 elements, so the total number of consecutive pairs is $n-1 = 5-1 = 4$. Thus, the color transition frequency is:

$$f_{transition} = \frac{2}{4} = \frac{1}{2}.$$

Blockiness

Let $S = \{s_1, s_2, \ldots, s_n\}$ be a sequence of colors, where each s_i represents a color at position i in the sequence. A block in this context is defined as a contiguous segment of the sequence where all the colors are the same. For example, in the sequence $\{red, red, blue, blue, blue\}$, there are two blocks: one of "red" and one of "blue". A block starts when the color changes from one element to the next and the length of each block is the number of consecutive elements with the same color.

Mathematically, a block is defined as a maximal contiguous subsequence of the form:

$$B = \{s_i, s_{i+1}, \ldots, s_{i+k-1}\} \tag{3.6}$$

where $s_i = s_{i+1} = \cdots = s_{i+k-1}$. The block count b is the number of such blocks in the sequence.

The maximum block length is the longest length of any contiguous block of the same color in the sequence. If there are multiple blocks with the same maximum length, it is still considered as a single value, which is the length of the longest block. The maximum block length L_{max} is:

$$L_{max} = \max_{B} L(B) \tag{3.7}$$

where B represents all blocks in the sequence.

The blockiness of the sequence can be defined as the ratio of the length of the longest block to the total number of elements in the sequence. This ratio gives an indication

of how "concentrated" the sequence is in terms of color repetition. The formula for blockiness $f_{blockiness}$ is:

$$f_{blockiness} = \frac{L_{max}}{n} \tag{3.8}$$

where L_{max} is the length of the longest block and n is the total number of elements in the sequence.

Color Proportion

Let $S = \{s_1, s_2, \ldots, s_n\}$ be a sequence of colors, where each s_i represents a color at position i in the sequence. The proportion of a color (for example, red or blue) is the fraction of elements in the sequence that are of the specific color. This can be represented as

$$p(c) = \frac{count(c)}{n} \tag{3.9}$$

where $count(c)$ is the number of occurrences of color c in the sequence S and n is the total number of elements in the sequence S.

The proportion of blue in the sequence is calculated as the number of "blue" elements in the sequence divided by the total number of elements in the sequence:

$$p(c) = \frac{count(blue)}{n} \tag{3.10}$$

where $count(blue)$ is the number of occurrences of color $blue$ in the sequence S.

Similarly, the proportion of red in the sequence is:

$$p(c) = \frac{count(red)}{n} \tag{3.11}$$

where $count(red)$ is the number of occurrences of color red in the sequence S.

Fraction of monochromatic arithmetic progressions

The fraction of monochromatic arithmetic progressions of various lengths is the target variable we seek to predict. This variable quantifies how frequently k consecutive elements are all the same color in a progression of randomly colored integers. Our goal is to develop predictive models that can accurately estimate these fractions based on the features extracted from the color sequences.

An arithmetic progression (AP) of length k is a sequence of k indices such that the difference between consecutive indices is constant. For example, for a progression of length 3 (triplets), the indices i, j, κ must satisfy:

$$j - i = \kappa - j = \text{constant difference.}$$

Generalizing this, for any progression of length k, $i_1, i_2, \ldots, i_k$ should satisfy the condition:

$$i_2 - i_1 = i_3 - i_2 = \cdots = i_k - i_{k-1} = d$$

Where d is the common difference between consecutive indices, and each i_k valid index in the sequence.

A monochromatic arithmetic progression of length k is a progression where all the colors at the indices $i_1, i_2, \ldots, i_k$ are the same. Specifically, for a sequence of colors $color[i_1], color[i_2], \ldots, color[i_k]$, a monochromatic progression occurs if:

$$color[i_1] = color[i_2] = \cdots = color[i_k].$$

The fraction of monochromatic APs of length k is the ratio of the number of monochromatic progressions of length k to the total number of possible progressions of length k. That is,

$$f_{monochromatic}(k) = \frac{Number\ of\ monochromatic\ APs\ of\ length\ k}{Total\ number\ of\ APs\ of\ length\ k}. \tag{3.12}$$

The ultimate goal is to identify features that exhibit significant predictive power regarding the emergence of monochromatic arithmetic progressions of varying lengths. These features will be systematically analyzed and selected based on their potential to capture underlying structures and correlations in the data.

Machine learning and deep learning models, such as LR, RF, and ANN, will be employed to train on these extracted features. The models will be evaluated to assess their capability to predict the proportion of monochromatic arithmetic progressions accurately for different colorings and various lengths of arithmetic progressions.

3.5 Testing and Evaluation

The testing and evaluation phase of this research involves a structured approach to validate the predictive capacity of the models developed from the generated features. The primary focus is on assessing how well these models can predict the proportions of monochromatic arithmetic progressions (k-APs) for different lengths and color configurations. We will begin by evaluating the predictive ability of individual models for monochromatic arithmetic progressions (k-APs) of various colors. Following this, we will train a combined model on the whole dataset, which includes different lengths of progressions, with length added as an extra feature. Dataset generated from progressions of length six will be used to assess the model. This will help us determine if the model can generalize effectively when tested on progression lengths it has not previously encountered.

3.5.1 Evaluation Metrics

The primary metric for evaluating the models will be the Root Mean Squared Error (RMSE). This metric measures the average magnitude of the prediction error, providing an indication of how well the model is able to predict the proportions of monochromatic k-APs. The RMSE is calculated as the square root of the average squared differences between the predicted values and the actual values, where a lower RMSE indicates better predictive accuracy.

The performance of the model will be compared to that of a baseline model, and improvements in RMSE over the baseline will be used as a key indicator of the model's effectiveness. This approach will help assess how much better the machine learning models (LR, RF, and ANN) are at predicting the proportions of monochromatic arithmetic progressions compared to a simple baseline model.

3.5.2 Feature Importance Analysis

To understand the contribution of each feature to the model's predictions, we evaluate two types of feature importance.

(1) Permutation Feature Importance: This method measures how much a model's performance degrades when a single feature's values are randomly shuffled. Specifically, we compute the root mean squared error (RMSE) on the validation set, then permute one feature at a time and recalculate the RMSE. The increase in RMSE reflects the importance of that feature. This method is model-agnostic and captures how the model actually uses each feature.

(2) Random Forest Feature Importance: For the Random Forest model, we also report the built-in feature importance based on the mean decrease in impurity (MDI). At each split in each tree, the algorithm records how much the split on a given feature reduces the variance (or Gini impurity for classification). The final importance score is the average of these reductions across all trees and splits where the feature is used.

3.5.3 Generalization Efforts

After evaluating the models on specific k-AP lengths and coloring configurations, we will attempt to generalize our findings to arbitrary lengths. This involves extending our analysis to k-APs of various other lengths, beyond those initially tested. The goal is to determine if the identified features and their importance remain consistent across different scenarios, thereby establishing broader applicability of our predictive models.

3.6 Baseline Model

In this section, we derive and explain the baseline prediction for the fraction of monochromatic k-term APs in a random two-coloring of integers. This baseline serves as the theoretical benchmark against which we evaluate the performance of our predictive models, including LR, RF, and MLP. By comparing the RMSE of the baseline to that of these models, we can assess the models' ability to predict the fraction of monochromatic k-APs.

The baseline prediction is the expected fraction of k-APs that are monochromatic (all integers in the AP having the same color) under a completely random two-coloring of integers. We consider a sequence of integers and analyze the probability of a given k-term AP within that sequence being monochromatic.

3.6.1 Baseline model for $r = 2$ (2 colors)

Let M_k be the total number of k-term arithmetic progressions in the integer sequence. This M_k is fixed, determined by the length of the sequence N and the value of k.

For each i from 1 to M_k, let X_i be a random variable defined as follows:

$$X_i = \begin{cases} 1, & \text{if the } i\text{-th } k\text{-term AP is monochromatic} \\ 0, & \text{if the } i\text{-th } k\text{-term AP is not monochromatic} \end{cases} \tag{3.13}$$

Each integer in the sequence can be assigned one of two colors: red (R) or blue (B). The coloring is random and independent, with each integer having an equal probability of being either red or blue:

$$P(\text{red}) = P(\text{blue}) = 0.5 \tag{3.14}$$

For an AP of length k (a single k-term AP), each integer can independently be either red or blue. Thus, the total number of possible color configurations for the k-term AP is:

$$\text{Total Configurations} = 2^k \tag{3.15}$$

A k-term AP is considered monochromatic if all integers in the sequence are of the same color. There are two possible monochromatic configurations:

- All integers are red: $(R, R, \ldots, R)$
- All integers are blue: $(B, B, \ldots, B)$

Hence, the number of monochromatic configurations for a single k-term AP is:

$$\text{Number of Monochromatic Configurations} = 2. \tag{3.16}$$

The probability of a single k-term AP being monochromatic is the ratio of monochromatic configurations to the total number of configurations:

$$P(X_i = 1) = \frac{\text{Number of Monochromatic Configurations}}{\text{Total Configurations}} = \frac{2}{2^k} = \frac{1}{2^{k-1}}. \tag{3.17}$$

This is also the expected value of the random variable X_i:

$$E(X_i) = (1) \times P(X_i = 1) + (0) \times P(X_i = 0) = \frac{1}{2^{k-1}}. \tag{3.18}$$

The total number of monochromatic k-APs is given by the sum of the individual indicator random variables:

$$\text{Total Monochromatic } k\text{-APs} = \sum_{i=1}^{M_k} X_i. \tag{3.19}$$

By the linearity of expectation, the expected number of monochromatic k-APs is:

$$E(\text{Total Monochromatic } k\text{-APs}) = E\left(\sum_{i=1}^{M_k} X_i\right) = \sum_{i=1}^{M_k} E(X_i) = M_k \cdot \frac{1}{2^{k-1}}. \tag{3.20}$$

Finally, the expected fraction of monochromatic k-APs is the expected number of monochromatic k-APs divided by the total number of k-APs:

$$\begin{aligned} \text{Expected Fraction of Monochromatic } k\text{-APs} &= \frac{E(\text{Total Monochromatic } k\text{-APs})}{M_k} \\ &= \frac{M_k \cdot \frac{1}{2^{k-1}}}{M_k} \\ &= \frac{1}{2^{k-1}}. \end{aligned} \tag{3.21}$$

3.6.2 Generalization to r Colors

The preceding example considered the special case of $r = 2$ colors (e.g., red and blue). This can be extended to the case where each element in the sequence is randomly assigned one of r colors independently and uniformly.

The total number of possible colorings for a k-term arithmetic progression is:

$$\text{Total Configurations} = r^k \tag{3.22}$$

A monochromatic coloring occurs when all k elements are the same color. Since there are r colors, there are exactly r monochromatic configurations:

$$\text{Monochromatic Configurations} = r. \tag{3.23}$$

Hence, the probability that a single k-AP is monochromatic is:

$$P(X_i = 1) = \frac{r}{r^k} = \frac{1}{r^{k-1}}. \tag{3.24}$$

Therefore, the expected value of the indicator variable X_i remains:

$$E(X_i) = \frac{1}{r^{k-1}}. \tag{3.25}$$

If there are M_k total k-term arithmetic progressions in the sequence, the expected number of monochromatic progressions is:

$$E\left(\sum_{i=1}^{M_k} X_i\right) = M_k \cdot \frac{1}{r^{k-1}}. \tag{3.26}$$

Finally, the expected proportion of monochromatic k-APs is:

$$\frac{E\left(\sum_{i=1}^{M_k} X_i\right)}{M_k} = \frac{1}{r^{k-1}}. \tag{3.27}$$

This general expression forms the analytical baseline used to evaluate the performance of machine learning models across different values of r and k.

The baseline serves as a theoretical benchmark that assumes no information about the data beyond random coloring. We compare the performance of our predictive models (LR, RF, and MLP) to this baseline using the RMSE:

- The RMSE is calculated assuming the predicted fraction of monochromatic k-APs is always equal to $\frac{1}{2^{k-1}}$, regardless of the data.
- The RMSEs of the LR, RF, and MLP models are calculated based on their predictions for the same target values.

3.7 Study Limitations

This study faces several limitations that could impact its outcomes. The accuracy of our forecasts rely on the quality and representativeness of the generated colorings; if the dataset does not capture a wide variety of scenarios, the models may not generalize well to new data. Additionally, as N and the number of colorings increase, the computational requirements for generating data and training models also rise, potentially limiting the feasibility of testing very large datasets without substantial computational resources. While machine learning models can identify important features, understanding and interpreting these features in a mathematical context can be challenging, as important features might not always have clear mathematical significance.

3.7.1 Illustrative Example: Growth in the Number of Arithmetic Progressions with N

To illustrate how the number of arithmetic progressions grows with the sequence length N, we consider the number of k-term arithmetic progressions (k-APs) in the set $[1, N]$. A k-term arithmetic progression is defined by:

$$a,\ a+d,\ a+2d,\ \ldots,\ a+(k-1)d$$

where $a \in \mathbb{N}$, $d \in \mathbb{N}$, and the largest term must satisfy:

$$a + (k-1)d \leq N$$

This implies:

$$a \leq N - (k-1)d$$

To ensure at least one valid value for a, the maximum valid d must satisfy:

$$(k-1)d \leq N-1 \quad \Rightarrow \quad d \leq \left\lfloor \frac{N-1}{k-1} \right\rfloor$$

Let $M = \left\lfloor \frac{N-1}{k-1} \right\rfloor$ denote the number of valid values for d.

For each valid $d \in \{1, 2, \ldots, M\}$, the number of valid starting values a is $N - (k-1)d$.

Hence, the total number of k-APs is:

$$\sum_{d=1}^{M} (N - (k-1)d) = NM - (k-1)\sum_{d=1}^{M} d = NM - (k-1) \cdot \frac{M(M+1)}{2}$$

Example: Let $N = 100$, $k = 3$. Then:

$$M = \left\lfloor \frac{100-1}{3-1} \right\rfloor = \left\lfloor \frac{99}{2} \right\rfloor = 49$$

So the number of 3-APs is:

$$100 \cdot 49 - 2 \cdot \frac{49 \cdot 50}{2} = 4900 - 2450 = 2450$$

Thus, there are exactly 2450 3-term arithmetic progressions in the set $[1, 2, \ldots, 100]$. This number grows rapidly with N, and illustrates the increasing computational cost of evaluating each sample as N becomes larger.

The plot below shows how the number of k-term arithmetic progressions increases as N increases, for different values of k from 3 to 7. For each value of k, the number of k-APs grows exponentially with N. The steeper curves for smaller values of k indicate that there are more progressions for smaller k for a given N.

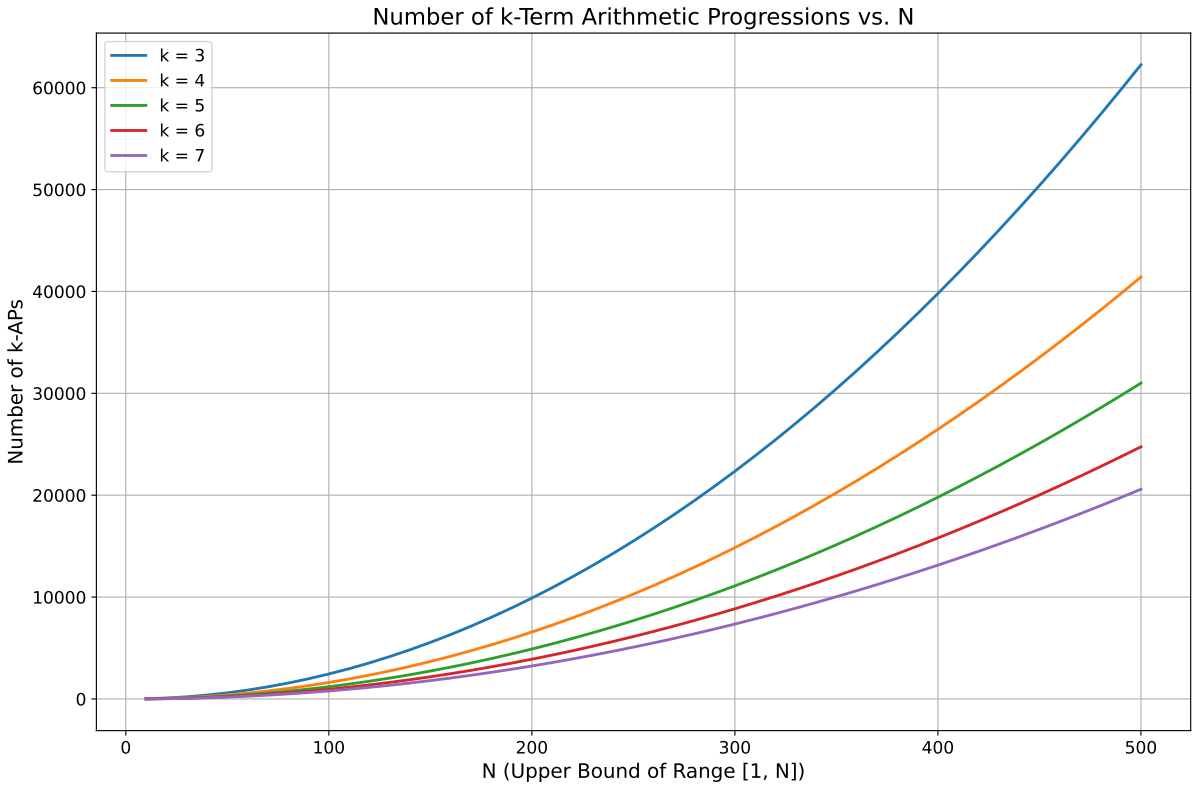


Figure 3.1: Growth in the number of k-term arithmetic progressions as N increases, for various values of k.

3.8 Conclusion

This chapter outlined the methodologies and processes employed to generate and analyze the data for the study. It began with an explanation of how the dataset of randomly colored integers was generated, establishing a strong foundation for further analysis. The focus then shifted to the feature generation process, where various characteristics of these colorings were identified and extracted as inputs for predictive modeling. The chapter continued with a description of the testing and evaluation phase, detailing how the predictive capabilities of the machine learning models will be assessed. Finally, limitations were discussed, providing insight into potential challenges and constraints encountered during the research.

Chapter 4

Monochromatic Arithmetic Progressions: Single Length Models

This chapter evaluates the performance of LR, RF, and MLP models in predicting the fractions of monochromatic APs of varying lengths ($k = 3, 4, 5$) within a random coloring of the integers from 1 to 100. A monochromatic arithmetic progression refers to a sequence of integers where the difference between consecutive terms is constant, and all terms are colored the same. The models are compared against a baseline prediction, which assumes a constant fraction of $\frac{1}{2^{k-1}}$, reflecting the theoretical expectation for random monochromatic progressions. This baseline is derived from the probability that all k integers in the progression are assigned the same color in a random coloring, where each integer has a 50% chance of being either red or blue. Specifically, for $k = 3$ (triples), the probability is $\frac{1}{4}$, for $k = 4$ (quadruples) it is $\frac{1}{8}$, and for $k = 5$ (quintuples) it is $\frac{1}{16}$. The aim is to assess how well the models predict these fractions and compare their performance to the baseline, using RMSE as the primary evaluation metric.

4.1 Linear Regression Models

The LR models were trained on 100,000 samples using six features:

- Proportion of blue and red integers
- Blockiness values
- Transition frequency
- Most popular color proportion
- Anti-symmetry

The dataset was split into 80% training and 20% testing set.

4.1.1 RMSE Performance by Sequence Length

Length (k)	Baseline RMSE	LR RMSE	Improvement
3	0.0126	0.0076	39.7%
4	0.0143	0.0099	30.8%
5	0.0135	0.0110	18.5%

Table 4.1: Linear Regression performance across sequence lengths

For $k = 3$ (arithmetic triples), the LR model achieved a 39.7% lower RMSE than the baseline (0.0076 vs 0.0126), demonstrating strong predictive capability for triplets. For $k = 4$ (arithmetic quadruples), the model maintained strong performance with a 30.8% RMSE reduction (0.0099 vs 0.0143). For $k = 5$ (arithmetic quintuples), the LR model still outperformed the baseline with an 18.5% improvement, although the margin was smaller. This decline in relative performance reflects the exponential increase in the number of k-APs as k grows, which makes accurate prediction of longer arithmetic progressions inherently more challenging.

4.1.2 Feature Importance

The permutation feature importance analysis for the LR model consistently highlighted proportion_blue and proportion_red as the dominant predictors, exhibiting extremely high importance values across all k values, indicating the primary influence of color distribution on the model's predictions. popular_color_prop consistently showed moderate importance, while blockiness_values, anti_symmetry, and transition_frequency consistently had low importance, suggesting minimal impact on the LR model's predictive power for these isolated settings. The model's heavy reliance on color proportions implies a direct linear relationship with the target variable, and the consistent pattern across different k values suggests the model captures a simple relationship, primarily driven by color proportions.

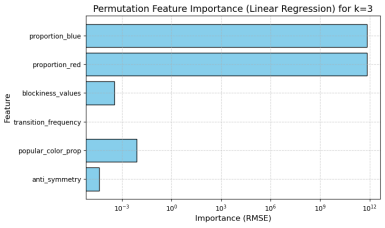


Figure 4.1: $k = 3$

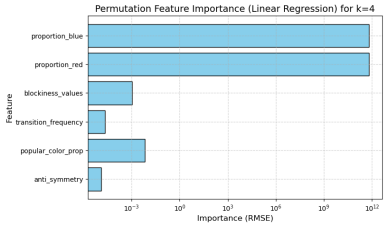


Figure 4.2: $k = 4$

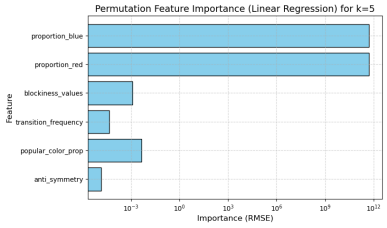


Figure 4.3: $k = 5$

Figure 4.4: LR feature importances across sequence lengths $k = 3$, $k = 4$, and $k = 5$

4.1.3 Prediction Scatterplots

The scatterplots for the LR model consistently demonstrated a strong linear correlation between predicted and actual values. In each case, data points were tightly clustered

around the y=x line, indicating a good fit and minimal scatter, suggesting accurate predictions. The range of predictions varied across different k values, reflecting the changes in the target variable's distribution, but the model's reliability and good predictive performance remained consistent.

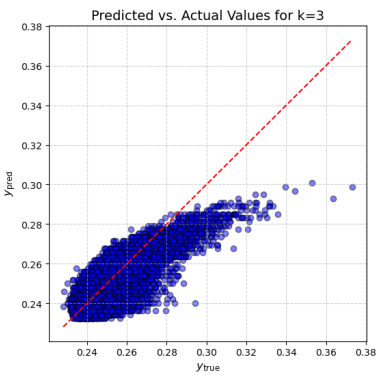


Figure 4.5: $k = 3$

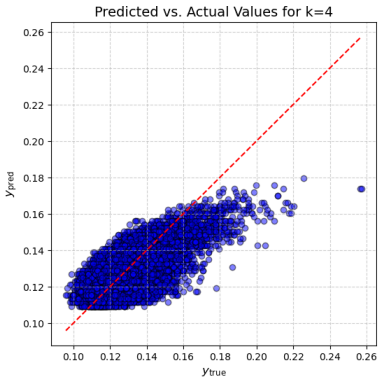


Figure 4.6: $k = 4$

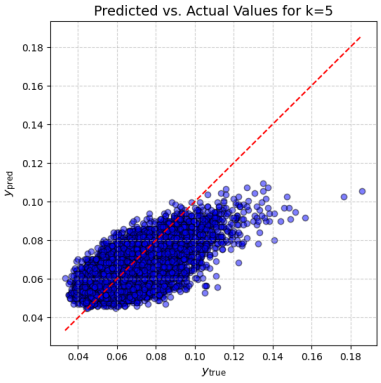


Figure 4.7: $k = 5$

Figure 4.8: LR prediction vs actual values across sequence lengths $k = 3$, $k = 4$, and $k = 5$

4.2 Random Forest Models

A Random Forest with n_estimators set to 200 and random_state set to 42 was implemented using the same dataset as in the LR case. The same data split of 80% and 20% was applied.

4.2.1 Performance by Sequence Length

RF achieved the strongest performance for arithmetic triples ($k = 3$), with a 42.1% improvement over the baseline, slightly outperforming linear regression. For arithmetic quadruples ($k = 4$), the model maintained good performance with a 29.4% reduction in RMSE, though it exhibited slightly higher error than LR for this progression length. As the sequence length increased to arithmetic quintuples ($k = 5$), RF demonstrated better scalability compared to LR, achieving a 22.2% improvement over the baseline, indicating its robustness in capturing patterns in longer progressions.

Length (k)	Baseline RMSE	RF RMSE	Improvement
3	0.0126	0.0073	42.1%
4	0.0143	0.0101	29.4%
5	0.0135	0.0105	22.2%

Table 4.2: Random Forest performance across sequence lengths

4.2.2 Feature Importance

The Random Forest model was used to assess feature importance across different values of k. The results indicate that the most influential feature across all progression lengths was the proportion of the most popular color. This feature consistently held the highest importance score, followed by transition frequency, anti-symmetry, and blockiness values. The proportions of blue and red were the least influential features. A general trend observed was that the influence of transition frequency and anti-symmetry increased as k grew. This shift suggests that for longer progressions, structural properties of the coloring (such as how often color transitions occur and the level of symmetry) play a more important role in determining the fraction of monochromatic APs.

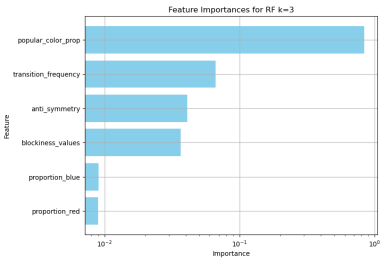


Figure 4.9: $k = 3$

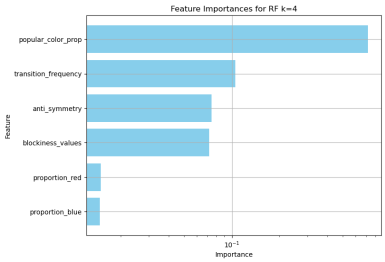


Figure 4.10: $k = 4$

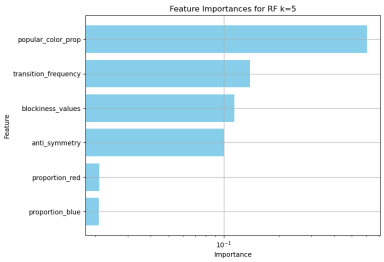


Figure 4.11: $k = 5$

Figure 4.12: RF feature importances across sequence lengths $k = 3$, $k = 4$, and $k = 5$

4.2.3 Prediction Scatterplots

For $k = 3$, the scatterplot shows a significant spread of points along the diagonal line, indicating strong model accuracy. However, as k increases, the spread of the points grows wider, reflecting greater prediction error and increased variability in the fraction of monochromatic APs. The spread also suggests that while the models still outperform the baseline, their confidence in predictions decreases for longer progressions.

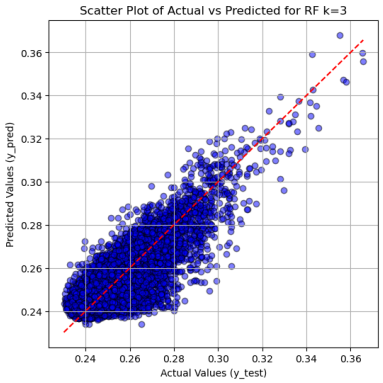


Figure 4.13: $k = 3$

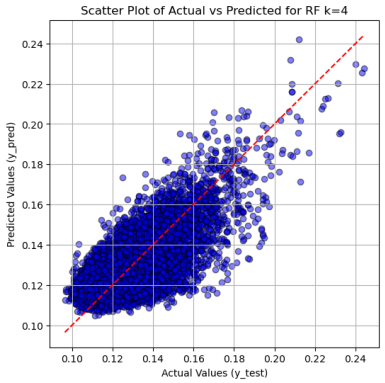


Figure 4.14: $k = 4$

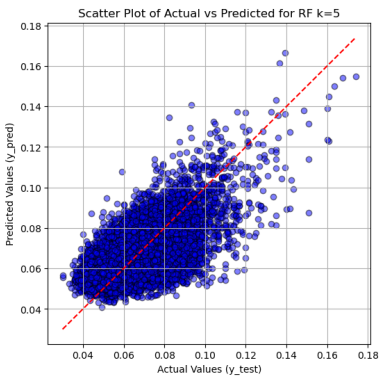


Figure 4.15: $k = 5$

Figure 4.16: RF prediction vs actual values across sequence lengths $k = 3$, $k = 4$, and $k = 5$

4.3 Multilayer Perceptron Models

A Multilayer Perceptron (MLP) model was trained on a dataset of 100,000 observations, split into training (80%) and testing (20%) sets. The MLP architecture consisted of two hidden layers: the first layer contained 200 neurons, while the second hidden layer had 180 neurons. The ReLU activation function was applied to both hidden layers.

4.3.1 Performance by Sequence Length

MLP achieved the best overall performance for arithmetic triples ($k = 3$), with a 43.7% improvement over the baseline, suggesting its superior handling of non-linear relationships. For arithmetic quadruples ($k = 4$), the model maintained strong performance, achieving a 35.7% reduction in RMSE and outperforming both LR and RF for this length. As the sequence length increased to arithmetic quintuples ($k = 5$), MLP demonstrated the best scalability, achieving a 28.1% improvement over the baseline, significantly outperforming the other models and highlighting its effectiveness in capturing complex patterns in longer progressions.

Length (k)	Baseline RMSE	MLP RMSE	Improvement
3	0.0126	0.0071	43.7%
4	0.0143	0.0092	35.7%
5	0.0135	0.0097	28.1%

Table 4.3: MLP performance across sequence lengths

4.3.2 Feature Importance

The permutation feature importance analysis for the MLP model, when trained and tested separately for k=3, 4, and 5, revealed a dynamic pattern of feature importance that evolved with increasing AP length. Consistently, popular_color_prop, proportion_blue, and proportion_red ranked among the most important features, highlighting the MLP's recognition of color distribution significance, though with a more nuanced approach than LR. Notably, blockiness_values gained importance with increasing k, suggesting the model adapted to emphasize sequence blockiness for longer APs. transition_frequency showed variable importance, becoming more relevant for k=5, indicating the model's adaptability to AP length. anti_symmetry consistently remained less influential. Unlike Linear Regression, the MLP captured complex, non-linear relationships, evident in the dynamic feature relevance across different k values, showcasing its adaptive feature utilization.

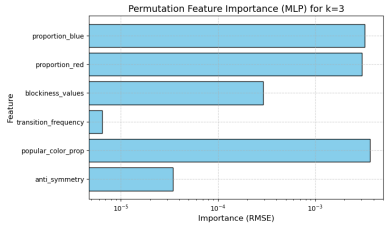


Figure 4.17: $k = 3$

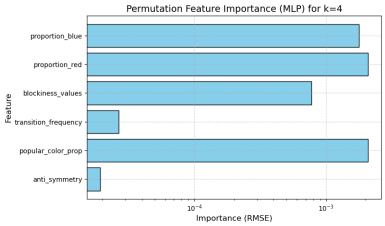


Figure 4.18: $k = 4$

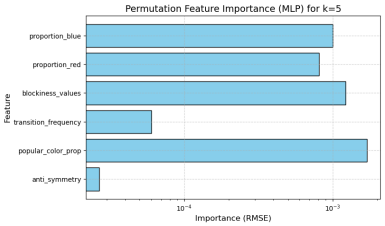


Figure 4.19: $k = 5$

Figure 4.20: MLP feature importances across sequence lengths $k = 3$, $k = 4$, and $k = 5$

4.3.3 Prediction Scatterplots

The scatterplots for the Multi-Layer Perceptron (MLP) model, when trained and tested separately for $k = 3, 4,$ and 5, consistently demonstrated a strong correlation between predicted and actual values. In each case, data points were generally clustered tightly around the $y = x$ line, indicating a good fit and minimal scatter, suggesting accurate predictions. The range of predictions varied across different k values, reflecting the changes in the target variable's distribution, but the model's reliability and good predictive performance remained consistent. Notably, compared to the LR and RF models, the MLP scatterplots showed slightly tighter clustering around the $y = x$ line, particularly in the lower ranges, suggesting that the MLP model might be capturing some nonlinear patterns that the linear and tree models misses, potentially leading to slightly more accurate predictions.

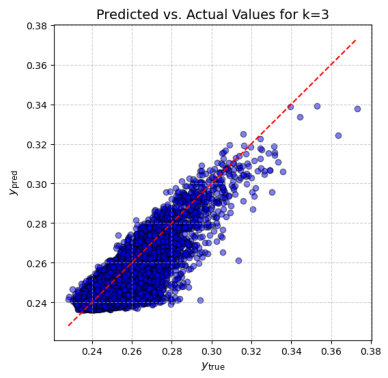


Figure 4.21: $k = 3$

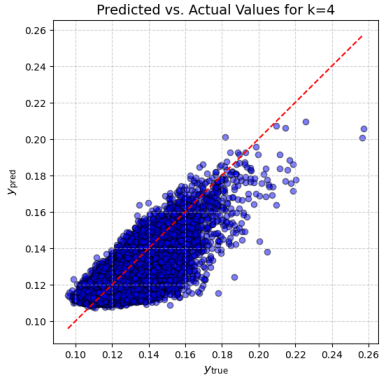


Figure 4.22: $k = 4$

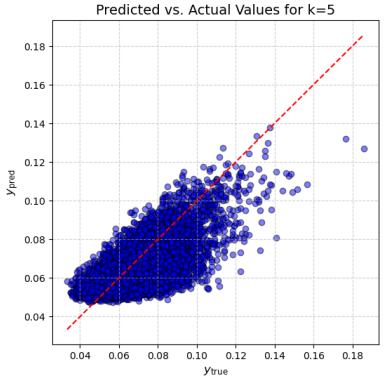


Figure 4.23: $k = 5$

Figure 4.24: MLP prediction vs actual values across sequence lengths $k = 3$, $k = 4$, and $k = 5$

4.4 Model Performance Evaluation

This chapter evaluated the performance of LR, RF, and MLP models in predicting the fractions of monochromatic arithmetic progressions (APs) of lengths $k = 3, 4, 5$ within randomly colored integer sequences. The models were compared against a baseline

prediction of $\frac{1}{2^{k-1}}$, representing the theoretical expectation under random coloring. The aim was to assess how well the models predict these fractions and compare their performance to the baseline, using RMSE as the primary evaluation metric.

4.4.1 Overall Model Comparison by RMSE

Across all progression lengths, the models outperformed the baseline, indicating their ability to capture patterns beyond random chance. The MLP model demonstrated the best overall performance, achieving the lowest RMSE values and the highest percentage improvements over the baseline for all k. Specifically, MLP achieved improvements of 43.7%, 35.7%, and 28.1% for $k = 3, 4, 5$ respectively. LR and RF also showed significant improvements, but to a lesser extent, with RF showing better scalability for longer progressions compared to LR. A noticeable trend in the results is that the RMSE for all models increases as k grows. This outcome aligns with the expectation that predicting longer APs is inherently more difficult, as the fraction of monochromatic APs for higher k exhibits more variance and is harder to capture using available features. The increasing RMSE suggests that while the models successfully learn predictive patterns for shorter progressions, their effectiveness diminishes as the complexity of the problem increases.

4.4.2 Feature Importance Analysis

For the LR model, the permutation feature importance analysis consistently highlighted proportion_blue and proportion_red as the dominant predictors, indicating the primary influence of color distribution. The MLP model, however, revealed a dynamic pattern of feature importance that evolved with increasing AP length. While color proportions remained important, popular_color_prop and blockiness_values gained significance, suggesting the model's ability to capture complex, non-linear relationships and adapt to different progression lengths. RF showed popular_color_prop to be the most influential feature across all progression lengths, with increasing importance of transition_frequency and anti_symmetry for longer APs. This suggests that structural properties of the coloring become more significant for longer progressions.

4.4.3 Scatterplot Analysis

The scatterplots for all models consistently demonstrated a strong correlation between predicted and actual values, with data points tightly clustered around the $y = x$ line. However, the MLP scatterplots showed slightly tighter clustering compared to the LR and RF models, particularly in the lower ranges. This suggests that the MLP model might be capturing some non-linear patterns that the linear and tree-based models miss, potentially leading to slightly more accurate predictions. The range of predictions varied across different k values, reflecting the changes in the target variable's distribution, but the models' reliability and good predictive performance remained consistent.

4.4.4 Discussion

The results indicate that all three models were able to predict the fractions of monochromatic APs with lower errors compared to a simple baseline model that predicts the mean proportion. The MLP model demonstrated superior performance, likely due to its ability to capture complex, non-linear relationships in the data. Feature importance analysis revealed that the models utilized different features and adapted their feature utilization strategies based on the progression length. Scatterplot analysis further illustrated the models' predictive performance, with the MLP model showing a slight advantage in capturing subtle patterns. While formal statistical significance tests were not performed, the reported error metrics suggest that all models consistently outperformed the baseline. Overall, these results highlight the practical effectiveness of machine learning models in predicting fractions of monochromatic APs and provide insights into feature importance and model performance across different progression lengths.

4.5 Conclusion

This chapter evaluated the predictive performance of LR, RF, and MLP models in estimating the fraction of monochromatic arithmetic progressions (APs) of varying lengths. The models were compared against a baseline prediction, which represents the theoretical probability of observing a monochromatic AP in a random two-coloring of integers. The RMSE was used as the primary evaluation metric to measure the deviation of predicted values from actual values.

Across all tested values of k, the machine learning models outperformed the baseline, demonstrating their ability to capture underlying patterns in the data. The MLP model consistently achieved the lowest RMSE, followed by RF and then LR. As the progression length k increased, the RMSE for all models showed a general upward trend, indicating a growing difficulty in accurately predicting the fractions of monochromatic APs for larger k.

Chapter 5

Monochromatic Arithmetic Progressions: Combined Length Models

5.1 Evaluation of Model Performance and Generalization

This chapter assesses the performance of LR, RF, and MLP models trained on data encompassing multiple progression lengths ($k = 3, 4, 5, 7$) to predict the fractions of monochromatic arithmetic progressions (APs) in a randomly colored integer sequence. A central objective of this analysis is to evaluate the extent to which these models can generalize beyond the training data by testing their predictive capabilities on both the seen progression lengths ($k = 3, 4, 5, 7$) and the completely unseen case of $k = 6$.

To benchmark the models' effectiveness, their predictions are compared against a theoretical baseline that assumes a constant fraction of $\frac{1}{2^{k-1}}$, which represents the expected proportion of monochromatic progressions in a purely random coloring. This comparison is essential for determining whether the models successfully capture meaningful structures within the data or merely approximate the expected behavior of random assignments.

The primary evaluation metric is the RMSE, enabling a detailed assessment of predictive performance across different values of k. The results obtained for $k = 6$ serve as a key indicator of whether the models can extrapolate learned patterns to a previously unseen progression length, thereby testing their ability to generalize beyond the training distribution.

To further enhance the predictive capability of the combined length model, the progression length k was explicitly included as a feature, allowing the model to directly learn relationships between progression length and the fraction of monochromatic APs. In addition to this, two supplementary features, **transition frequency deviation** and

anti-symmetry deviation, were introduced to assess their influence on the model's generalization across different progression lengths.

The **transition frequency deviation** measures how much the transition frequency of a given coloring deviates from the average transition frequency in the dataset. This feature provides insight into structural variations within the coloring process. Similarly, the **anti-symmetry deviation** quantifies departures from symmetry in the coloring pattern, potentially influencing the formation of monochromatic progressions. By incorporating these additional features, we aim to determine whether they contribute meaningfully to improving model generalization, particularly in predicting the unseen case of $k = 6$.

5.2 Results and Discussion

This chapter presents the results of training and evaluating LR, RF, and MLP models to predict the fraction of monochromatic arithmetic progressions (APs) in randomly colored integer sequences. The models were trained on data from progression lengths k = 3, 4, 5, and 7, and their performance was assessed on both these lengths and the unseen length k = 6. The primary metric for evaluation was Root Mean Squared Error (RMSE), compared against a baseline RMSE derived from the theoretical probability of random coloring $(1/2^{k-1})$. The inclusion of k as a feature, along with transition frequency deviation and anti-symmetry deviation, aimed to enhance generalization capabilities, particularly to the unseen $k = 6$

5.2.1 Evaluation by RMSE

Table 5.1: Overall RMSE Comparison for Linear Regression, Random Forest, and MLP Models

k	Linear Regression	Random Forest	MLP	Baseline RMSE
3	0.297066	0.007251	0.007061	0.012583
4	0.229654	0.009995	0.009297	0.014384
5	0.010545	0.010425	0.009639	0.013430
6	0.009669	0.032812	0.010008	0.011307
7	0.327274	0.007480	0.006983	0.008847

The LR model demonstrated inconsistent performance across different values of k. While it showed improvement over the baseline for $k = 5$ and the unseen $k = 6$, it performed significantly worse than the baseline for $k = 3, 4$, and 7. The high RMSE values for these lengths suggest that the linear nature of the LR model is insufficient to capture the underlying complexities of the relationship between k and the fraction of monochromatic APs. Although the model showed some ability to generalize to $k = 6$, the overall inconsistency raises serious concerns about its reliability for this task. The poor performance strongly indicates the presence of non-linear relationships in the data that a simple linear model cannot effectively capture. The fact that it surpasses

the baseline for $k = 6$ might be due to the specific characteristics of the data at that point, but the overall inconsistency makes it an unreliable model.

The RF model exhibited a more nuanced performance. For the seen values of k (3, 4, 5, and 7), the RF model consistently outperformed the baseline, demonstrating its ability to learn predictive patterns from the training data. However, a key finding was the model's struggle to generalize to the unseen k = 6. While it performed well on the seen data, the RMSE for k = 6 was significantly higher than the baseline, indicating a breakdown in predictive accuracy for the unseen progression length.

The MLP model provided the most consistent and promising results. It outperformed the baseline for all values of k, including the unseen k = 6. While the performance on k = 6 was slightly weaker than on the seen lengths, the fact that it still surpassed the baseline suggests that the MLP was better able to generalize than the LR or RF models. The consistent performance across different k values indicates that the MLP effectively learned the relationship between k and the fraction of monochromatic APs. The lower RMSE for k = 7 compared to other seen values suggests that the model may have captured some specific characteristics of longer progressions.

5.2.2 Analysis of Feature Importance

Linear Regression

The permutation feature importance analysis for the LR model revealed a consistent pattern across all values of k. The features anti_symmetry and anti_symmetry_deviation, which capture the degree of symmetry in the colored integer sequences, were consistently the most important predictors of the fraction of monochromatic APs. This suggests that symmetry plays a crucial role in determining the prevalence of monochromatic APs. The features transition_frequency and transition_frequency_deviation, which measure the frequency of color transitions in the sequences, were also consistently important, although to a lesser extent. Other features, such as popular_proportions and blockiness_values, showed moderate importance for some k values but were generally less influential. The features proportion_red and proportion_blue consistently had very low importance, suggesting redundancy or that their information is already captured by other features. Overall, the consistent dominance of anti_symmetry and deviation across all k values highlights the importance of symmetry in predicting monochromatic APs using a linear model and suggests that these features capture a fundamental aspect of the underlying relationship, even though the LR model itself is not a good fit for the data overall.

Figure 5.1: Overall Analysis of Feature Importance Distributions for Linear Regression across $k = 3, 4, 5, 6,$ and 7

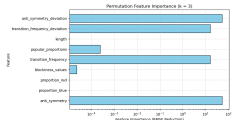


Figure 5.2: $k = 3$

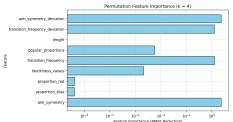


Figure 5.3: $k = 4$

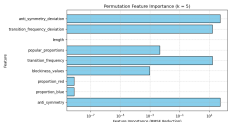


Figure 5.4: $k = 5$

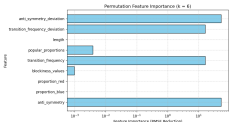


Figure 5.5: $k = 6$

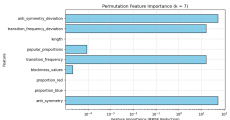


Figure 5.6: $k = 7$

Random Forests

Analysis of feature importance using RF (5.7) revealed that the progression length (k) was by far the most dominant feature, accounting for over 98% of the model's decision-making. This strongly suggests that the model primarily relied on the value of k itself for its predictions, rather than effectively utilizing the other features (transition frequency deviation and anti-symmetry deviation) to generalize.

Figure 5.7: Overall Analysis of Feature Importance Distributions for Random Forest across $k = 3, 4, 5, 6,$ and 7

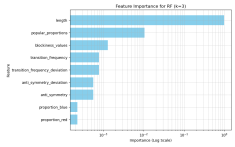


Figure 5.8: $k = 3$

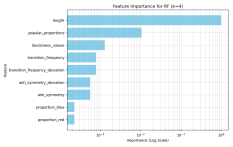


Figure 5.9: $k = 4$

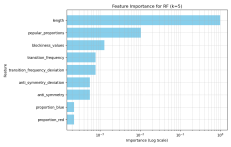


Figure 5.10: $k = 5$

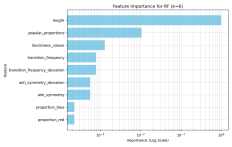


Figure 5.11: $k = 6$

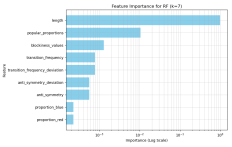


Figure 5.12: $k = 7$

Multi Layer Perceptron

The permutation feature importance analysis for the Multi-Layer Perceptron (MLP) (5.13) model revealed a dynamic interplay of features contributing to the prediction of monochromatic arithmetic progressions (APs). While popular_proportions and blockiness_values consistently ranked among the most important, their relative influence varied across different k values. For shorter APs (k = 3, 4, 5), popular_proportions dominated, suggesting its key role in predicting monochromatic AP occurrences. However, as k increased to 6 and 7, blockiness_values, capturing the blockiness of the sequences, emerged as the most influential feature, indicating its heightened relevance for predicting longer APs. Color proportions, represented by proportion_red and proportion_blue, were consistently significant across all k values, highlighting their importance in the MLP model's predictions. Symmetry features, anti_symmetry and anti_symmetry_deviation, maintained moderate importance throughout, suggesting their relevance, although less dominant than in the simpler linear model. This dynamic pattern of feature importance underscores the MLP model's ability to capture complex, non-linear relationships and utilize a wider range of features to make accurate predictions across different AP lengths.

Figure 5.13: Overall Analysis of Feature Importance Distributions for Multi Layer Perceptron across $k = 3, 4, 5, 6,$ and 7

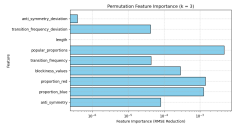


Figure 5.14: $k = 3$

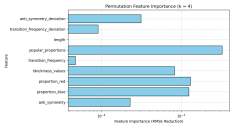


Figure 5.15: $k = 4$

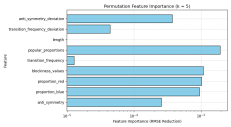


Figure 5.16: $k = 5$

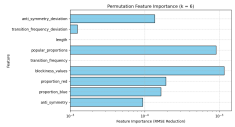


Figure 5.17: $k = 6$

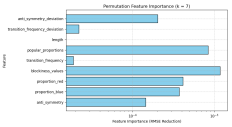


Figure 5.18: $k = 7$

5.2.3 Scatter Plots: Predicted vs. Actual Values

Linear Regression

The Linear Regression (LR) model generally demonstrated poor predictive accuracy for the fraction of monochromatic arithmetic progressions (APs), particularly for k=3, k=4, and k=7. However, it surprisingly outperformed the baseline model for k=5 and k=6, achieving lower RMSE values despite showing a poor fit in the scatterplots. This suggests that while a linear model is not suitable for capturing the complex relationships in the data, it might sometimes produce reasonable predictions due to random chance or data distribution. The scatterplots consistently revealed significant deviations from the ideal fit line, clustering of predicted values, and varying biases across different k values. These findings collectively highlight the limitations of a linear model for this prediction task and emphasize the need for more sophisticated models capable of capturing non-linear relationships.

Figure 5.19: Overall Comparison of Linear Regression Predictions vs. Actual Values for $k = 3, 4, 5, 6,$ and 7

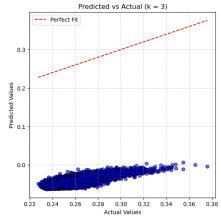


Figure 5.20: $k = 3$

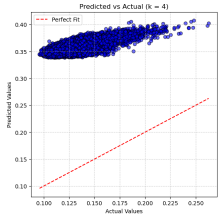


Figure 5.21: $k = 4$

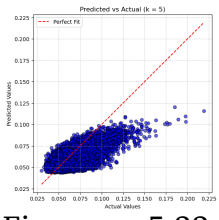


Figure 5.22: $k = 5$

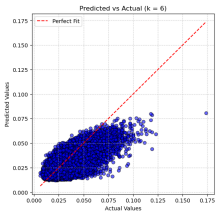


Figure 5.23: $k = 6$

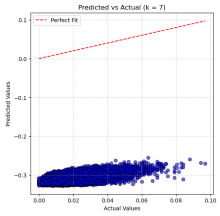


Figure 5.24: $k = 7$

Random Forests

For seen k values, the predictions clustered reasonably well around the diagonal, although some deviations, particularly at higher fraction values, were observed. For k = 6, however, the scatter plot clearly showed the model's difficulty in capturing the true distribution of fractions, with a tendency to underestimate larger values and overestimate smaller ones. This reinforces the conclusion that the RF model, despite its good performance on seen data, struggled to extrapolate to unseen k values.

Figure 5.25: Overall Comparison of Random Forest Predictions vs. Actual Values for $k = 3, 4, 5, 6,$ and 7

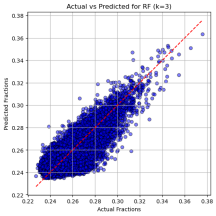


Figure 5.26: $k = 3$

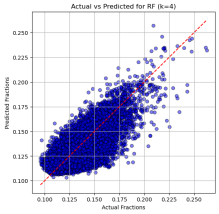


Figure 5.27: $k = 4$

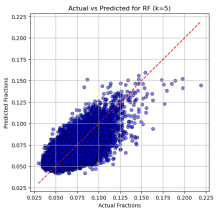


Figure 5.28: $k = 5$

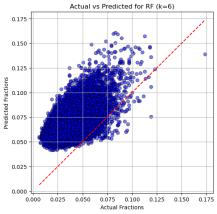


Figure 5.29: $k = 6$

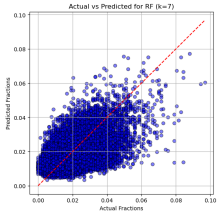


Figure 5.30: $k = 7$

Multi Layer Perceptron

The MLP model demonstrated good predictive accuracy for the fraction of monochromatic arithmetic progressions (APs) across all values of k (AP length). The scatterplots consistently showed data points tightly clustered around the ideal fit line, indicating a strong correlation. This suggests that the MLP model was able to effectively capture the complex, non-linear relationships between the features and the target variable. The MLP model also exhibited minimal bias, with data points distributed relatively symmetrically around the perfect fit line. While some outliers were observed, particularly at higher actual values, the overall performance of the MLP model was consistently strong across different k values. This highlights the suitability of the MLP model for this prediction task and its ability to capture the underlying patterns in the data, even for longer and less frequent APs.

Figure 5.31: Overall Comparison of Multi Layer Perceptron Predictions vs. Actual Values for $k = 3, 4, 5, 6,$ and 7

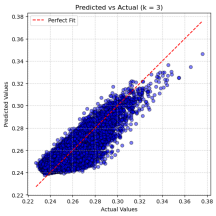


Figure 5.32: $k = 3$

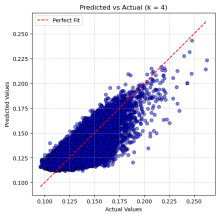


Figure 5.33: $k = 4$

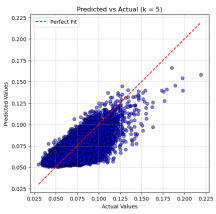


Figure 5.34: $k = 5$

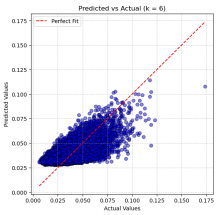


Figure 5.35: $k = 6$

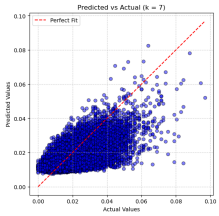


Figure 5.36: $k = 7$

Overall, the MLP model demonstrated the strongest performance and the best generalization capabilities. While the RF model performed well on seen data, its inability to generalize to k = 6 highlights the limitations of relying heavily on the k value itself as a predictor. The LR model's inconsistent performance underscores the need for models capable of capturing non-linear relationships in the data. The results suggest that the MLP, with its ability to learn complex non-linear functions, is better suited for this prediction task. Future work could explore further optimization of the MLP architecture

and investigate the potential of incorporating additional features to improve generalization, particularly for unseen progression lengths. The fact that even the best model (MLP) showed some degradation in performance for the unseen k = 6 highlights the inherent difficulty of extrapolating in this domain.

5.3 Conclusion

In this chapter, we investigated the performance of LR, RF, and MLP models in predicting the fraction of monochromatic APs across multiple progression lengths (k = 3, 4, 5, 7), with a critical focus on their ability to generalize to an unseen length, k = 6. The primary metric for evaluation was the RMSE, comparing model predictions against a theoretical baseline. The MLP model emerged as the most consistent and effective, demonstrating superior performance across all k values, including the unseen k = 6. While its accuracy slightly decreased for k = 6, it still outperformed the baseline, indicating successful generalization. This highlighted the MLP's capacity to learn complex, non-linear relationships inherent in the data.

Conversely, the RF model, while performing well on seen k values, struggled to generalize to k = 6, with its RMSE exceeding the baseline. This failure underscores the RF's reliance on memorizing training data and its limited ability to extrapolate beyond the learned patterns. Feature importance analysis revealed that the RF model heavily relied on the progression length (k) itself as a predictor, neglecting other potentially informative features. The LR model exhibited inconsistent performance, performing poorly on most k values except for k = 5 and k = 6, where it surprisingly outperformed the baseline. However, the scatterplots revealed a poor fit, indicating that the LR model's occasional success might be due to specific data characteristics rather than genuine learning of underlying relationships.

Feature importance analysis further illuminated the models' varying approaches. The LR model predominantly relied on symmetry-related features, while the MLP dynamically adapted feature importance based on k, emphasizing popular color proportions for shorter APs and blockiness for longer ones. The scatterplots corroborated the RMSE results, showing tight clustering of MLP predictions around the ideal fit line, indicating high accuracy, while the RF and LR scatterplots revealed significant deviations and biases.

In conclusion, the MLP model demonstrated the strongest performance and the best generalization capabilities, highlighting its suitability for predicting monochromatic AP fractions across varying progression lengths. The RF model's over-reliance on the k value and the LR model's inability to capture non-linear relationships underscored the importance of model selection and feature engineering for this complex task. The slight degradation in MLP performance at k = 6 emphasizes the inherent difficulty of extrapolation in this domain and suggests avenues for future research, such as exploring enhanced feature engineering to further improve generalization.

Chapter 6

Overall Discussion of Results and Conclusion

This thesis aimed to explore the efficacy of machine learning models in predicting the fraction of monochromatic arithmetic progressions (APs) within randomly colored integer sequences, bridging AI methodologies with problems in pure mathematics, particularly extremal combinatorics. Two distinct paradigms were explored: the development of individual models for specific AP lengths (k) in Chapter 4, and the creation of combined models capable of learning across multiple k values in Chapter 5. This dual approach facilitated a thorough evaluation of the models' predictive accuracy and their capacity to generalize to unseen progression lengths, a critical aspect of mathematical inquiry.

In Chapter 4, separate LR, RF, and MLP models were trained for each AP length (k = 3, 4, 5). Notably, all models consistently outperformed the baseline prediction, which was based on the theoretical probability of random coloring, for their respective k values. This outcome underscored the predictive power of the selected features: proportion of blue/red integers, blockiness values, transition frequency, proportion of the most popular color, and anti_symmetry, demonstrating AI's potential to uncover insights into combinatorial structures. The MLP model consistently achieved the lowest RMSE, followed by RF and LR, indicating that more complex models better capture the inherent relationships for specific k values.

However, these single-length models were inherently limited by their lack of generalization. Trained independently for each k, they could not predict monochromatic AP fractions for unseen lengths. The observed increase in RMSE with higher k values across these models also highlighted the growing complexity of predicting longer progressions.

To address the generalization challenge, Chapter 5 explored models trained on data encompassing multiple AP lengths (k = 3, 4, 5, 7), evaluating their performance on both seen and unseen lengths (k = 6). This approach aimed to leverage shared information across different k values, enhancing predictive accuracy and enabling generalization, a vital capability for AI applications in mathematics.

The results presented a more nuanced picture. The MLP model demonstrated the most promising performance, successfully generalizing to the unseen k = 6, albeit with a slight reduction in accuracy compared to seen lengths. This success highlighted the MLP's ability to learn complex non-linear relationships, allowing it to extract generalizable patterns and extrapolate to unseen progression lengths.

In contrast, the Random Forest (RF) model, despite performing well on seen k values, failed to generalize to k = 6, performing worse than the baseline. This failure underscored the RF's tendency to rely heavily on the k value itself as a predictor, rather than learning generalizable patterns from other features, as evidenced by the feature importance analysis. The Linear Regression (LR) model showed inconsistent performance, likely due to its inability to capture the inherent non-linearities in the data.

Overall Discussion:

This thesis has demonstrated the potential of machine learning models to predict monochromatic AP fractions and contribute to extremal combinatorics. Single-length models are effective for capturing patterns in specific k values, but they lack generalization. Combined models, especially the MLP, offer a more promising avenue by enabling generalization to unseen lengths. The MLP's capacity to learn complex non-linear relationships makes it particularly well-suited for this task, especially when generalization to unseen k values is desired.

The slight performance degradation of the MLP at k = 6, however, underscores the challenges of extrapolation in this domain and the importance of careful feature engineering when applying AI to mathematical problems. The feature importance analyses across models also revealed crucial insights. In single-length models, features like color proportions and blockiness exhibited significant predictive power. In combined models, the MLP effectively utilized a broader range of features, adapting their importance based on k, while the RF's over-reliance on k highlighted the risk of overfitting to specific training data.

Future Research Directions:

To further advance this research, several avenues could be explored:

- Investigate the incorporation of additional features that capture more subtle aspects of coloring patterns, potentially improving model generalization.
- Explore alternative machine learning models, including deep learning architectures and graph neural networks, which might be better suited for capturing complex combinatorial relationships.
- Foster a deeper theoretical understanding of the relationship between k and monochromatic AP fractions, which could guide the development of more robust and generalizable predictive models.
- Extend the analysis to larger datasets and higher k values to test the scalability and robustness of the models.

This research serves as a valuable step in bridging AI methodologies with problems in pure mathematics, particularly in extremal combinatorics, and paves the way for future investigations that could lead to new mathematical insights and discoveries.

Appendix A

Code Repository

The code for this research can be found at: GitHub Repository

References

[Aedo *et al.* 2022] Ibai Aedo, Uwe Grimm, Yasushi Nagai, and Petra Staynova. Monochromatic arithmetic progressions in binary thue–morse-like words. *Theoretical Computer Science*, 934:65–80, 2022.

[Agarap 2018] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

[Alon and Spencer 2016] Noga Alon and Joel H Spencer. *The probabilistic method*. John Wiley & Sons, 2016.

[Alon 2020] Noga Alon. Problems and results in extremal combinatorics–iv. *arXiv preprint arXiv:2009.12692*, 2020.

[Ardal *et al.* 2005] Hayri Ardal, Tom C Brown, and Peter AB Pleasants. Almost disjoint families of 3-term arithmetic progressions. *Journal of Combinatorial Theory, Series A*, 109(1):75–90, 2005.

[Bahrammirzaee 2010] Arash Bahrammirzaee. A comparative survey of artificial intelligence applications in finance: artificial neural networks, expert system and hybrid intelligent systems. *Neural Computing and Applications*, 19(8):1165–1195, 2010.

[Bello *et al.* 2016] Irwan Bello, Hieu Pham, Quoc V Le, Mohammad Norouzi, and Samy Bengio. Neural combinatorial optimization with reinforcement learning. *arXiv preprint arXiv:1611.09940*, 2016.

[Breiman 2001] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[Charton *et al.* 2024] François Charton, Jordan S Ellenberg, Adam Zsolt Wagner, and Geordie Williamson. Patternboost: Constructions in mathematics with a little help from ai. *arXiv preprint arXiv:2411.00566*, 2024.

[Davies *et al.* 2021] Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Battaglia, Charles Blundell, András Juhász, et al. Advancing mathematics by guiding human intuition with ai. *Nature*, 600(7887):70–74, 2021.

[Dubey *et al.* 2022] Shiv Ram Dubey, Satish Kumar Singh, and Bidyut Baran Chaudhuri. Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing*, 2022.

[Dybizbański 2012] Janusz Dybizbański. Sequences containing no 3-term arithmetic progressions. *the electronic journal of combinatorics*, pages P15–P15, 2012.

[Efron 1979] Bradley Efron. Computers and the theory of statistics: thinking the unthinkable. *SIAM review*, 21(4):460–480, 1979.

[Erdös and Turán 1936] Paul Erdös and Paul Turán. On some sequences of integers. *Journal of the London Mathematical Society*, 1(4):261–264, 1936.

[Farebrother 2018] Richard William Farebrother. *Linear least squares computations*. Routledge, 2018.

[Gauss 1877] Carl Friedrich Gauss. *Theoria motus corporum coelestium in sectionibus conicis solem ambientium*, volume 7. FA Perthes, 1877.

[Glorot *et al.* 2011] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings, 2011.

[Goodfellow *et al.* 2016] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[Graham and Butler 2015] Ron Graham and Steve Butler. *Rudiments of Ramsey theory*, volume 123. American Mathematical Soc., 2015.

[Greenwood *et al.* 2024] Torin Greenwood, Jonathan Kariv, and Noah Williams. From discrete to continuous: Monochromatic 3-term arithmetic progressions. *Mathematics of Computation*, 2024.

[Hachimi *et al.* 2020] Marouane Hachimi, Georges Kaddoum, Ghyslain Gagnon, and Poulmanogo Illy. Multi-stage jamming attacks detection using deep learning combined with kernelized support vector machine in 5g cloud radio access networks. In *2020 international symposium on networks, computers and communications (ISNCC)*, pages 1–5. IEEE, 2020.

[Hastie *et al.* 2009] 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.

[He 2024] Yang-Hui He. Ai-driven research in pure mathematics and theoretical physics. *Nature Reviews Physics*, 6(9):546–553, 2024.

[Hornik *et al.* 1989] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

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

[Jukna 2011] Stasys Jukna. *Extremal combinatorics: with applications in computer science*, volume 571. Springer, 2011.

[LeCun *et al.* 2015] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[Legendre 1806] Adrien Marie Legendre. *Nouvelles méthodes pour la détermination des orbites des comètes: avec un supplément contenant divers perfectionnemens de ces méthodes et leur application aux deux comètes de 1805*. Courcier, 1806.

[Ma and Tang 2021] Yao Ma and Jiliang Tang. *Deep learning on graphs*. Cambridge University Press, 2021.

[Malhotra and Malhotra 2003] Rashmi Malhotra and Davinder K Malhotra. Evaluating consumer loans using neural networks. *Omega*, 31(2):83–96, 2003.

[Montgomery and Runger 2010] Douglas C Montgomery and George C Runger. *Applied statistics and probability for engineers*. John wiley & sons, 2010.

[Montgomery *et al.* 2021] Douglas C Montgomery, Elizabeth A Peck, and G Geoffrey Vining. *Introduction to linear regression analysis*. John Wiley & Sons, 2021.

[Mundinger *et al.* 2025] Konrad Mundinger, Max Zimmer, Aldo Kiem, Christoph Spiegel, and Sebastian Pokutta. Neural discovery in mathematics: Do machines dream of colored planes? *arXiv preprint arXiv:2501.18527*, 2025.

[Parczyk *et al.* 2022] Olaf Parczyk, Sebastian Pokutta, Christoph Spiegel, and Tibor Szabó. New ramsey multiplicity bounds and search heuristics. *Discrete Mathematics Days 2022*, 263:224, 2022.

[Porter 2010] Theodore M Porter. *Karl Pearson: The scientific life in a statistical age*. Princeton University Press, 2010.

[Quinlan 1986] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1:81–106, 1986.

[Raayoni *et al.* 2021] Gal Raayoni, Shahar Gottlieb, Yahel Manor, George Pisha, Yoav Harris, Uri Mendlovic, Doron Haviv, Yaron Hadad, and Ido Kaminer. Generating conjectures on fundamental constants with the ramanujan machine. *Nature*, 590(7844):67–73, 2021.

[Rosenblatt 1958] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[Rumelhart *et al.* 1986] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[Stanton 2001] Jeffrey M Stanton. Galton, pearson, and the peas: A brief history of linear regression for statistics instructors. *Journal of Statistics Education*, 9(3), 2001.

[Starship Knowledge 2023] Starship Knowledge. *Neural Networks: Perceptrons*, 2023.

[Stigler 1990] Stephen M Stigler. *The history of statistics: The measurement of uncertainty before 1900*. Harvard University Press, 1990.

[Turán 1941] Paul Turán. On an external problem in graph theory. *Mat. Fiz. Lapok*, 48:436–452, 1941.

[van 1927] Beweis einer baudetschen vermutung. *Nieuw Arch. Wiskunde*, 15:212–216, 1927.

[Vijay 2012] Sujith Vijay. Monochromatic progressions in random colorings. *Journal of Combinatorial Theory, Series A*, 119(5):1078–1080, 2012.

[Wagner 2021] Adam Zsolt Wagner. Constructions in combinatorics via neural networks. *arXiv preprint arXiv:2104.14516*, 2021.

[Weisberg 2005] Sanford Weisberg. *Applied linear regression*, volume 528. John Wiley & Sons, 2005.

[Woolridge 2009] Jeffrey M Woolridge. Introductory econometrics: A modern approach. *Mason, OH: South Western Cengage Learning*, 2009.