## THOMPSON RIVERS UNIVERSITY

Collectible Asset Valuation and Forecasting - Insights from Magic: The Gathering

By

Roberto Primo Curti Sanches

## A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Science in Data Science

#### KAMLOOPS, BRITISH COLUMBIA

August, 2024

#### SUPERVISORS

Dr. Erfanul Hoque Dr. Sean Hellingman

© Roberto Primo Curti Sanches, 2024

#### Abstract

Magic: The Gathering (MtG) represents a significant and dynamic market for collectible trading cards, characterized by fluctuating prices driven by tournament results, player demand, and card rarity. This thesis explores time series forecasting techniques applied to the MtG card market, focusing on forecasting card prices using statistical and machine learning models. Specifically, the research compares the performance of traditional methods such as ARIMA, Random Walk, and NNETAR models to a proposed forecast combination neural network model.

A comprehensive database was created, combining price, tournament, and card attribute data, and feature engineering was employed to enhance the predictive power of the models. The methodology incorporates advanced statistical techniques and machine learning to build a more accurate and robust forecasting system. The results indicate that the proposed neural network model outperforms traditional methods in forecasting accuracy. This project also presents the ts.shiny application, an interactive tool which offers an accessible platform for visualizing and analyzing time series data.

The research concludes with insights into the factors driving MtG card prices and suggestions for improving forecasting models and applications in the future.

**Keywords:** ARIMA, Collectible Market, Forecast Combination, Machine Learning, Magic: The Gathering, Neural Networks, Time series forecasting.

# Acknowledgements

I would like to express my gratitude to Dr. Erfanul Hoque for his invaluable guidance on this project and his patience in dealing with his hard-headed student. A heartfelt thanks to Dr. Sean Hellingman for his constant support along the path that led to this thesis and for his enthusiasm in discussing and analyzing materials from a game he encountered for the first time. Without them, this work would not exist.

Thanks to my family, who encouraged me in this endeavour and supported me from thousands of kilometres away, and to Dr. Richard Garfield for creating a masterpiece that epitomizes a love-hate relationship.

"I must not fear. Fear is the mind-killer. Fear is the little-death that brings total obliteration..." — Litany Against Fear, Frank Herbert

# Contents

1	Intr	oducti	ion	1
	1.1	Backg	round of Study	2
	1.2	Motiv	ation	5
	1.3	Proble	em Statement	6
	1.4	Objec	tives of Study	8
		1.4.1	Creation of a Consolidated Database for Study	8
		1.4.2	Performance Evaluation of Current Techniques	8
		1.4.3	Comparison of Proposed vs. Generic Models	9
		1.4.4	Development of Data Visualization Application	9
	1.5	Study	Outline	10
2	Lite	erature	Review	11
	2.1	Applie	eations of Time Series Forecasting	11
		2.1.1	Time Series Forecasting in Financial Markets	13
		2.1.2	Forecasting in collectibles and Non-financial Assets	14
	2.2	Time	Series Forecasting for MtG	16
		2.2.1	Review of Previous Research	17

3	Dat	tabase Creation	19
	3.1	Data Sources	 20
		3.1.1 Price Data	 20
		3.1.2 Tournament Data	 21
		3.1.3 Card Related Data	 23
	3.2	Preprocessing	 24
		3.2.1 Data Cleaning	 25
		3.2.2 Feature Engineering	 26
		3.2.3 Data Transformation	 32
4	Dat	ta	34
1	4 1	Card Selection Methods	34
	7.1		 94
		4.1.1 By Card Supply $\ldots$ $\ldots$ $\ldots$	 35
		4.1.2 By Card Demand	 37
	4.2	Data Selection	 39
		4.2.1 Selections for Group 1	 39
		4.2.2 Selections for Group 2	 41
	4.3	Data Summary	 47
F	Ма	thedelowy	40
J	me	thodology	49
	5.1	Statistical Models for Forecasting	 50
		5.1.1 Autoregressive Model	 50
		5.1.2 Random Walk $\ldots$ $\ldots$ $\ldots$ $\ldots$	 54
	5.2	Machine Learning Models for Forecasting	 56

		5.2.1	Neural Networks Models	57
		5.2.2	Forecast Combination	62
	5.3	Propo	sed Model: Forecast Combination Based Neural Network	63
		5.3.1	Establishment of Base Models	63
		5.3.2	Determination of Weights	64
		5.3.3	Combination of Point Forecasts	65
		5.3.4	A data-driven Forecast combination based Neural Network Model	67
6	Exp	erime	nts and Evaluation	70
	6.1	Analy	sis	70
		6.1.1	Experimental Settings	71
		6.1.2	Evaluation Metrics	72
		6.1.3	Evaluation Methods	73
	6.2	Result	S	76
		6.2.1	Group Comparison	76
		6.2.2	Forecast combination based Neural Network	78
7	Ts.s	shiny:	Visualization Using Interactive Graphics	84
	7.1	Ts.shi	ny	85
		7.1.1	Data Democracy	85
		7.1.2	Data Agnosticism	86
		7.1.3	Similar Tools	86
	7.2	Archit	ecture	87

		7.2.1	Data Module
		7.2.2	EDA Module
		7.2.3	Forecasting Module
		7.2.4	Integration and Visualization
8	Con	clusio	n and Discussion 94
	8.1	Summ	ary of Findings
		8.1.1	Database Creation
		8.1.2	Proposed Model
		8.1.3	Model Comparison
		8.1.4	ts.shiny
	8.2	Future	e Research Directions
		8.2.1	Improve Model Application
		8.2.2	Different Modeling Engineering
		8.2.3	Increased Base Model Selection
		8.2.4	Continuous Model Improvement
		8.2.5	ts.shiny Improvement

### A Card Prices Line Plots

110

# List of Tables

4.1	Card Selection for Group 1	41
4.2	Card Selection for Group 2	46
6.1	RMSEs for point forecasting results in Group 1 $\ldots$	80
6.2	MAPEs for point for ecasting results in Group 1 $\ldots$	80
6.3	RMSEs for point forecasting results in Group 2 $\ldots \ldots$	83
6.4	MAPEs for point forecasting results in Group 2	83

# List of Algorithms

1	Clustering Algorithm															44
2	Cluster Optimization		•		•	•	•	•		•		•				46

# List of Figures

1.1	Comparison of 'Reserved List' returns to other financial indices	3	
1.2	Play Booster Content Breakdown Graphic. From "Introduc- ing Play Boosters: The Best of Two Worlds Combined," by Wizards of the Coast, 2024, https://wpn.wizards.com/en/ news/introducing-play-boosters-the-best-of-two-worlds	-combined.	4
1.3	Market value price of competitive decks	5	
3.1	Plot showing the card price and tournament use side by side	23	
3.2	MtG card with its cost and effect highlighted	30	
3.3	Tournament Frequency Schedule	33	
4.1	Heatmap of commercial MtG set releases	35	
4.2	Set price distribution by decade	36	
4.3	Diagram explaining game formats and their card pools	37	
4.4	Group 2 representation of card clusters	38	
4.5	Groups 1 and 2 Venn Diagram	38	
4.6	Price distributions of Group $1 \setminus \text{Group } 2 \times \text{Group } 1 \cap \text{Group } 2$ .	39	
4.7	Group 1 final selection diagram	40	
4.8	Sequence of states for the clustering process	43	

4.9	K-means result for Modern in 2017	45
5.1	Representation of an Artificial Neural Network Diagram	57
5.2	Representation of a NNETAR(4,7) Diagram	60
5.3	Forecast combination based ANN proposed model diagram	69
6.1	Rolling window cross-validation	75
6.2	Heatmap of model accuracy ranking for cards of Group 1. $\ .$ .	77
6.3	Heatmap of model accuracy ranking for cards of Group 2. $$ .	77
6.4	Average of MAPEs across all models for Group 1	78
6.5	X2ED.233 Forecast results for best models	79
6.6	Average of MAPEs across all models for Group 2	81
6.7	LRW.145 Forecast results for best models	82
6.8	IKO.67 Forecast results for best models	82
7.1	ts.shiny System Architecture	88
7.2	Exploratory Data Analysis interactive elements	90
7.3	Forecasting Analysis interactive elements	91
7.4	ts.shiny application, showcasing the modularity of the visu- alizations.	92

# Listings

7.1	Selector for visualization container				•		•					92
7.2	Visualization container $\ldots$ .		•	•	•		•	•	•	•		93

# Chapter 1

# Introduction

Throughout history, humans have constantly attempted to predict and explain the future, from seers and prophets of classical times to modern statisticians and data scientists. The methods and approaches have evolved, but the quest remains the same. Regardless of prophetic interpretations of natural events, the deterministic belief of gods and dice, or butterfly flapping wings, our trust that the 'lessons of days gone by teach us what will come to pass' continues.

The utilization of time series and forecasting statistical inference pervades an extensive array of phenomena, from natural occurrences such as weather and earthquakes to health-related pathologies. It extends to routine corporate and financial operations for sales forecasting or asset price predictions. Its ubiquitous applications across disciplines such as economics, finance, engineering, and environmental sciences underscore the indispensable role of time series analysis in revealing underlying patterns, trends, and interdependencies embedded within temporal data. This chapter introduces 'Magic: The Gathering' (MtG) and identifies the shortcomings in the current forecasting methods when applied to such collectible markets. Based on this, the scope and the aims of the Thesis are identified.

### 1.1 Background of Study

Magic: The Gathering, created in 1993 by Richard Garfield, Ph.D., is the world's first and most famous trading card game, credited as the origin of modern trading card games. It is a fantasy-themed trading card game owned by Wizards of the Coast (WotC), with an estimated 50 million players worldwide over its history [Hasbro, 2023], MtG has grown to be one of the largest and most popular card games globally. It features over 25,000 unique cards, with new editions or sets released every few months, each containing a mix of new cards and reprints of older ones.

The game involves players using a carefully constructed set of cards to strategically outmaneuver their opponents while attempting to achieve one of the conditions for victory. Players build decks ranging from 40 to 250 cards, from over 25,000 unique cards, employing various strategies and playstyles, ranging from aggressive approaches to control and adaptable midrange tactics.

Hasbro, the parent company of WotC, generates approximately \$1.1 billion of its \$5 billion annual earnings from MtG card sales [Schmidt, 2023]. The game's appeal lies in its complexity and adaptability, allowing players to build unique decks from its extensive card pool. This customization and the strategic depth of optimizing resources to control or defeat opponents attract players to local, national, and international tournaments. The value of individual MtG cards depends on their utility and power within popular decks, similar to the valuation of sports cards based on player performance. Certain cards, either by high desirability or low demand, present a price increase several times higher than established indices in the financial market (Figure 1.1).



Figure 1.1: Comparison of 'Reserved List' returns to other financial indices

The primary market for MtG involves WotC selling physical products to distributors that, in turn, supply specialized stores, game shops, and bookstores. Unlike most products, MtG cards are typically sold in randomized packs, meaning players cannot directly purchase specific cards. Still, they must rely on chance or the secondary market, which, when applied to the MtG community, defines the trade of single cards by players or game stores to acquire desired cards. This robust secondary market provides thousands of jobs annually for store owners, investors, and traders who buy and sell individual cards and sealed products.



Figure 1.2: Play Booster Content Breakdown Graphic. From "Introducing Play Boosters: The Best of Two Worlds Combined," by Wizards of the Coast, 2024, https://wpn.wizards.com/en/news/ introducing-play-boosters-the-best-of-two-worlds-combined.

MtG cards are sold in booster packs containing a random assortment of cards (Figure 1.2). While each set includes a few highly valuable cards, most cards in a pack are of lower value, mirroring the dynamics seen in sports card collecting. The expected value of the contents in most sealed products is often less than the retail price, introducing an element of chance similar to gambling. This randomness, combined with the game's strategic elements, contributes to the vibrant secondary market and the ongoing popularity of MtG. Through this secondary market, players spend hundreds or thousands of dollars to build a single deck to play the game (Figure 1.3). An expenditure that often repeats as the sets rotate out of use, new strategies become dominant, or the player delves into a different game format. Analyzing these cards' relationships and market trends presents a fascinating opportunity for understanding and predicting card values in this unique collectible market.



Figure 1.3: Market value price of competitive decks

MtG's competitive scene includes organized international tournaments, with a total prize pool of millions of dollars, and a significant secondary market for trading cards, where some individual cards can sell for thousands of dollars. This market is dynamic, driven by the limited print runs of cards, new releases, and the evolving strategies of the game. The substantial secondary market for MtG cards draws comparisons to stock markets, presenting opportunities for investment and speculation.

### 1.2 Motivation

Deckbuilding in MtG is a critical aspect of gameplay, leading to extensive discussion, analysis, and testing by players. Numerous websites and online forums are dedicated to cataloguing, showcasing, and rating tournament decks. The complexity of deckbuilding arises from the intricate interactions between cards [Ward et al., 2021]. MtG cards possess a range of attributes, including both numerical values and descriptive text detailing various effects and abilities. The strength of a card is not always immediately apparent; a seemingly underwhelming card can exhibit strong synergies with specific other cards [Alvin et al., 2021]. Identifying these synergies is a significant part of the game. Experienced players develop a keen ability to recognize positive synergies, evaluate card potential, and anticipate opponents' choices.

This dynamic creates a unique interplay between the game and its financial value. Unlike other collectibles, MtG cards possess intrinsic deterministic values governed by clear and established rules. This allows players to differentiate between valuable and less valuable assets within a closed system. The game's competitive nature drives this system, where the emergence of dominant strategies leads to increased demand for certain cards, subsequently raising their value and price. In contrast, the demand and price for others decline. With over 30 years of comprehensive data, these interactions can be analyzed in detail.

This research, therefore, proposes that forecasting the prices of Magic: The Gathering cards is an intriguing problem worthy of exploration.

### 1.3 Problem Statement

Time series analysis is a powerful statistical tool for understanding data collection patterns, predicting trends, and making informed decisions. However, using historical data to forecast future outcomes is complex, raising questions about stationarity, linearity, and the need for exogenous or endogenous variables. Real-world data, such as the market for MtG cards, often defies simple theoretical classifications, appearing convoluted and multifaceted.

Traditional methods like autoregressive (ARMA and ARIMA) and expo-

nential smoothing models rely on strict assumptions [Box and Jenkins, 1968], which may not always hold in the context of the MtG market [Di Napoli, 2017]. For example, the randomness of booster packs, which indirectly affects the supply of rare cards, and the periodic introduction of new sets create non-stationary data patterns that these traditional models may struggle to handle. Dynamic or state-space models offer greater flexibility by incorporating external variables, such as market trends and player strategies, capturing system dynamics more precisely.

Effective time series analysis requires adherence to two core principles: stationarity, where statistical properties remain constant over time, and identifying trends over extended periods. However, the MtG market often exhibits non-stationarity, with properties changing due to trends, seasonal variations, or random processes. Ignoring these shifts can result in inaccurate forecasts.

External factors, such as market rules, regulatory changes, and socioeconomic influences, further complicate the analysis by altering time series trends. MtG's competitive nature, with its tournaments and evolving strategies, drives demand for specific cards, influencing their market value. The emergence of dominant strategies increases demand for counter-strategy cards while others decline in value.

Therefore, traditional time series analysis methods may fall short in the MtG context. This research aims to enhance the reliability and effectiveness of time series analysis by addressing these challenges and expanding its applications to understand better and predict the values of MtG cards in the secondary market. This involves capitalizing on comprehensive historical data and advanced statistical models to capture the intricate dynamics of this unique collectible market.

### 1.4 Objectives of Study

This research aims to deepen the understanding of external influences on time series data and propose a model for forecasting MtG card prices. Focusing on the collectible market, particularly MtG, this study will critically decompose predictor features, dissecting the card price time series into its fundamental components and isolating the impacts of these external variables. The comprehensive analysis proposed here is expected to significantly enhance forecasting capabilities, yielding more robust, accurate, and actionable predictions.

#### 1.4.1 Creation of a Consolidated Database for Study

Despite over 30 years of history, there is no single comprehensive source for data on MtG prices and influencing variables. Tournament results, price data, and card statistics are scattered across different sources. The first objective of this research is to create a unified database encompassing all relevant information regarding Magic: The Gathering.

#### **1.4.2** Performance Evaluation of Current Techniques

The second objective focuses on evaluating the performance of existing time series analysis techniques under the influence of strong external factors present in the MtG market. This inquiry seeks to determine whether current techniques perform differently under such conditions, assessing their accuracy and reliability. By scrutinizing these methods, the research aims to identify limitations in current practices and inform the development of more effective techniques.

#### **1.4.3** Comparison of Proposed vs. Generic Models

The third objective explores the effectiveness of a neural network-based forecast combination model compared to established models, such as ARIMA, RW and NNETAR. This study posits whether a data-driven model tailored to a specific time series would outperform a generic model in the context of the MtG market. While generic models offer a universally applicable approach, their efficacy may diminish when dealing with time series data influenced by persistent external factors. A comparative study will be conducted to test this hypothesis, contributing to the development of optimized models for time series analysis.

#### 1.4.4 Development of Data Visualization Application

The final objective concerns the development of an interactive data visualization tool. Both time series forecasting and MtG are highly specialized areas that rarely overlap. Introducing an interactive tool allows MtG domain experts to better understand statistical concepts and applications while allowing time series experts to explore better data in a foreign domain.

### 1.5 Study Outline

The remaining seven chapters of this project are structured as follows: Chapter 2 discusses the structured literature review, findings, and interpretations; Chapter 3 presents the consolidated database and its creation process; Chapter 4 details the process of selecting cards for use in the modelling; Chapter 5 outlines the implementation of the modelling and the respective methodology; Chapter 6 covers the analysis discussed in the previous chapter and presents the results; development and description of a ready-to-use data visualization application is discussed in Chapter 7; finally, Chapter 8 concludes the thesis by summarizing the results and suggesting future work.

# Chapter 2

# Literature Review

This chapter presents a brief background on asset price changes and forecasting. First, it demonstrates its application to the financial market and how the methods changed over time to reach the current "state of the art" of the field. Then, it explains the initial conceptualization of performing an analysis of non-financial assets, such as art or collectibles, and how this field has grown over the past couple of decades. Finally, it discusses previous works on MtG and how the game has drawn research interest.

### 2.1 Applications of Time Series Forecasting

The application of time series forecasting spans multiple fields, each benefiting from the method's ability to predict future trends based on historical data. First published in 1922, one of the earliest applications was meteorology. Lewis Fry Richardson's work [2007] laid the groundwork for numerical weather prediction by pioneering mathematical models to forecast weather patterns.

Time series forecasting has played a crucial role in monitoring and predicting disease outbreaks in healthcare and epidemiology. The methods introduced by Box and Tiao [1975] have been instrumental in analyzing interventions in time series data, for applying stochastic models to represent the noise and intervention effects, enhancing the accuracy of estimating the magnitude and nature of changes caused by interventions. More recently, machine learning models have been applied to epidemiological forecasting application of non-parametric methods for influenza prediction, as demonstrated by Viboud et al. [2003], who utilized the method of analogues to effectively predict influenza activity across national and regional levels, outperforming traditional autoregressive models in terms of accuracy and forecasting horizon.

In retail and inventory management, time series forecasting is crucial for optimizing stock levels and meeting customer demand. Building upon the foundational methods introduced by Brown [1959] and Holt [2004], which established sound mathematical theories for computing the average rate of demand and the maximum reasonable demand during a lead time, Winters [1960] extended these concepts by introducing a seasonal adjustment component into exponential smoothing models, making them more effective for forecasting sales patterns that exhibit both trend and seasonal variations.

Adopting machine learning techniques has revolutionized time series forecasting across multiple domains. The exploration of Support Vector Machines (SVM) by Vapnik [1995] paved the way for practical applications in various fields by introducing a rigorous theoretical foundation for maximizing the margin between data classes, which has been instrumental in improving the generalization performance of models on unseen data.

#### 2.1.1 Time Series Forecasting in Financial Markets

The application of time series forecasting to financial markets began with Alfred Cowles [1933], who examined the predictive accuracy of financial analysts and demonstrated that their forecasts often performed no better than random chance. This early work laid the groundwork for understanding the complexities of financial market prediction, further explored by Kendall's [1953] analysis of time series, revealing the stochastic nature of economic data and challenging the assumption of predictable trends in stock prices.

A significant shift occurred in 1970 with Eugene Fama's Efficient Market Hypothesis (EMH) [1970]. Fama's EMH posited that stock prices are unpredictable, challenging the premise of time series forecasting in financial markets. Despite this, the work of Box, Jenkins, and Reinsel [1968] provided a robust framework for modelling financial time series with their ARIMA models, which allowed for systematic analysis and forecasting of economic data.

The evolution of time series forecasting in financial markets continued with the introduction of econometric models. Robert Engle's [1982] development of the Autoregressive Conditional Heteroskedasticity (ARCH) model enabled the modelling of time-varying volatility in financial data by allowing for a conditional variance to change over time as a function of past errors, thereby capturing the clustering of volatility often observed in financial markets. Tim Bollerslev [1986] expanded this with the Generalized ARCH (GARCH) model, which generalized the ARCH model to include past conditional variances in the current variance equation, providing a more flexible framework for modelling and forecasting volatility in financial markets.

In recent years, the adoption of machine learning has significantly advanced financial time series forecasting. The exploration of Support Vector Machines (SVM) by Vapnik and colleagues in the 1990s paved the way for practical applications in market prediction. Building on this, deep learning models, particularly Long Short-Term Memory (LSTM) networks, have emerged as state-of-the-art techniques. The application of LSTM networks in financial forecasting has been demonstrated by researchers such as Fischer and Krauss [2018] in their paper "Deep learning with long short-term memory networks for financial market predictions," highlighting the enhanced accuracy and capability of these models in capturing complex patterns in financial data due to their ability to effectively model temporal dependencies and learn from sequences, leading to superior predictive performance compared to traditional machine learning methods.

## 2.1.2 Forecasting in collectibles and Non-financial Assets

Research on forecasting in collectibles is relatively limited compared to financial markets. The initial work often focused on art markets and antiques. One of the earliest studies was by Baumol [1985] in "Unnatural Value: Or Art Investment as a Floating Crap Game," which analyzed the volatility and returns in art markets and concluded that art investment carries high risk with returns that may not justify the investment compared to other financial assets. This foundational work set the stage for further exploration into the predictability of prices in the collectibles market.

The development of hedonic pricing models significantly advanced the field. Chanel, Gérard-Varet, and Ginsburgh [1996] in "The Relevance of Hedonic Price Indices" explored the use of these models to forecast prices in art and antique markets, providing a systematic approach to understanding price determinants and highlighting the importance of quality adjustments in pricing data. Goetzmann [1993] in "Accounting for Taste: Art and the Financial Markets over Three Centuries" provided an extensive historical analysis of art prices and their predictability, finding a strong correlation between art prices and broader financial market trends, which suggested that art prices tend to follow economic cycles and are influenced by the wealth of investors.

Recent advancements have seen the incorporation of machine learning techniques. Ashenfelter and Graddy [2003] in "Auctions and the Price of Art" discussed the application of econometric and machine learning models to forecast auction prices, demonstrating that these models could improve prediction accuracy by accounting for complex variables such as bidder behaviour and auction dynamics. The predictability of returns on niche collectibles like stamps or coins has also been studied, with Burton and Jacobsen [1999] examining these markets in "Measuring Returns on Investment in Collectibles," where they explored the investment potential of collectibles and emphasized the importance of market-specific factors in forecasting returns.

Additionally, niche collectibles such as sports cards and comic books have been subjects of academic inquiry. In "An Introduction to the Collectible Sportscard Market," O'Brien, Gramling and Rodriguez [1995] explored the investment potential while considering different portfolio strategies, illustrating these markets' volatility and unique risks. Wyburn and Roach [2012] in "A hedonic analysis of American collectible comic book prices" examined factors affecting comic book prices, providing insights into how variables such as condition, rarity, and demand contribute to price fluctuations and their implications for forecasting in these specialized markets.

Despite the diverse applications illustrated by these authors, financial analysis for non-financial assets is still a relatively unexplored field. Literature on this still focuses on demonstrating its similarity to traditional investment assets, presenting non-financial assets as a portfolio strategy for risk management, or identifying valuable attributes within a niche. Still, due to their low volatility, no study focused on forecasting future prices. However, this trait is not shared by MtG singles, which show extreme volatility and trade volume in short periods compared to other collectibles.

### 2.2 Time Series Forecasting for MtG

Research on MtG is relatively sparse, even within niche collectibles. Despite its significant presence in modern popular culture and the active engagement of its community, discussions on MtG are seldom recognized as academic or scientific evidence.

#### 2.2.1 Review of Previous Research

Despite limited work on time series analysis in MtG, some studies have explored its potential as an investment asset. Langelett and Wang [2023] compare the returns of MtG's sealed products to other financial indexes, discussing their inclusion in traditional portfolios for diversification, and find that these products offer competitive returns with unique diversification benefits within an investment portfolio.

Di Napoli delved deeper into investment strategies in his thesis, "Multiasset Trading with Reinforcement Learning: An Application to Magic the Gathering Online" [Di Napoli, 2017]. He proposed a machine learning-based trading strategy for Magic: The Gathering Online (MTGO), the digital counterpart of the tabletop game, which has its virtual economy, demonstrating the viability of reinforcement learning in optimizing multi-asset portfolios within a complex, dynamic market.

Studies on MtG span various topics beyond time series forecasting. Sajaki [2019] applied regression models to predict card prices based on card attributes, providing insights into how specific features of cards, such as rarity and set, influence their market value. The computational aspects of MtG have garnered significant attention, with research ranging from developing autonomous agents to play the game, as proposed by Ward [2021] and Alvin [2021], who both worked on creating competitive AI frameworks that can effectively strategize and make in-game decisions based on the complex rule set of MtG.

Further, studies by Bjørke and Fludal [2017], and Tieber and Felfernig [2021] explored modelling card selection and deck building, illustrating how computational methods can be employed to optimize deck configurations for various game formats. Additionally, Cowling's research [2009] examined the application of Monte Carlo search techniques to card selection in MtG, demonstrating that these techniques could significantly enhance the decisionmaking process by simulating numerous game outcomes to choose optimal plays under uncertainty, even with the game's inherent complexity. Churchill [2019] also demonstrated the game's Turing completeness, highlighting the computational depth and potential for AI research within the MtG domain.

These studies illustrate the diverse applications and evolving methodologies. Although some of these are related to financial analysis, such as the work from Di Napoli, literature review showed a clear gap regarding card price forecasting through time series analysis.

# Chapter 3

# **Database Creation**

+

This chapter delves into the data sources and preprocessing steps essential for analyzing and forecasting MtG card prices. The fragmented nature of the data necessitates a meticulous approach to data collection and preparation. By sourcing data from various platforms, such as MTGGoldfish, MTGTop8, and MTGJSON, and applying preprocessing techniques, a comprehensive dataset was constructed to support the development of accurate forecasting models. The following sections outline the specific data sources and the preprocessing methods employed to ensure consistency, reliability, and the effective integration of these diverse data streams.

### 3.1 Data Sources

The analysis and forecasting of MtG card prices rely on diverse and fragmented data sources, reflecting the nascent stage of academic research in this area. Unlike more established fields, consolidated academic-level datasets are scarce for MtG card price forecasting. Consequently, this research data collection process integrates information from multiple public and proprietary sources, each offering unique insights and challenges.

Given the fragmented nature of the data, meticulous curating and preprocessing of the datasets is essential to ensure consistency and reliability. The data can be broadly categorized into three main types: price data, tournament data, and card-related data. Each type provides a distinct perspective on the factors influencing card prices and contributes to the robustness of our forecasting models.

By synthesizing these diverse data sources, constructing a comprehensive dataset that captures the multifaceted nature of MtG card prices is possible, facilitating more accurate and reliable forecasting.

#### 3.1.1 Price Data

The price data for MtG cards used in this study are sourced from MTGGoldfish [n.d.], one of the largest online databases for MtG decks. MTGGoldfish hosts comprehensive card information and aggregates the latest decks, individual card details, and recent listings from significant e-commerce marketplaces, including eBay. MTGGoldfish compiles a price history for each card based on daily prices from TCGPlayer, the largest MtG marketplace in North America and owned by eBay [TCGplayer, 2022]. This aggregated price history is essential for conducting time series analysis.

Although MTGGoldfish does not provide an API or direct method for requesting this data, it makes each time series available for download on individual card sub-pages for subscribers. A web crawler bot was developed in Python to collect this extensive dataset to retrieve price data for over 40,000 cards individually. During this process, non-commercial sets (a classification applied to any set not designed and destined for tournament play) were excluded from the data selection to avoid overwhelming the site with unnecessary requests and focus computational resources on relevant data collection.

Despite MtG being a game with over 30 years of history, reliable price data with fine temporal granularity has only been consistently collected since 2011. Before this, historical price information was sporadically available in monthly card game magazines such as InQuest and Duelist. However, not all cards were listed regularly, making constructing a consistent historical database impossible. Even post-2011, there has been some inconsistency in price data for low-demand cards during the early months of data collection. Therefore, this study restricts the data collection window from September 2013 to September 2023 to ensure the reliability and consistency of the dataset.

#### 3.1.2 Tournament Data

Tournament data provides valuable insights into the competitive landscape of Magic: The Gathering (MtG), reflecting the popularity and effectiveness of specific cards in high-stakes play. This study's tournament data was sourced from MTGTop8 [n.d.].

MTGTop8 is a comprehensive database that tracks top-performing decks from various competitive events, including the professional tournament circuit (Grand Prix, Pro Tours, and Magic Online Championships). This platform offers detailed information on deck compositions, individual card usage, and tournament outcomes, which is crucial for understanding the competitive meta and card performance over time.

The collected tournament data spans a significant period, covering events from September 2013 to September 2023. This period was chosen to align with the price data collection window, ensuring consistency across datasets. As for the tournament selection, MtG has thousands of tournaments each year, ranging from casual to the highest level of play. Only the professional tournament circuit and the qualifiers that lead directly to them were considered for this research.

In addition to the tournament usage of cards, an important concept considered in this study is the legality of cards in each game format. Official MtG tournaments are governed by WotC, which regularly updates the rules to maintain a stable and healthy competitive environment. One of these updates may involve banning a card, rendering it illegal in decks while the ban is active. These updates, decided by the parent company and communicated through their website [of the Coast, n.d.], were also scraped and included in the database.

Integrating tournament data with price data can help better understand the factors driving card prices. For instance, cards frequently appearing in



Figure 3.1: Plot showing the card price and tournament use side by side.

top-performing decks tend to experience price increases due to higher demand from competitive players during periods around the professional tournaments and their qualifiers (Figure 3.1). Conversely, cards that fall out of favour in the meta may see price declines. This correlation between tournament performance and market value underscores the importance of including tournament data in our analysis and forecasting models.

#### 3.1.3 Card Related Data

Card-related data encompass various attributes and characteristics of MtG cards, providing essential context that enhances the accuracy of our forecasting models. This study primarily utilizes data from MTGJSON [n.d.], a comprehensive and easy-to-use resource aggregating card data from multiple sources.

MTGJSON offers a well-structured API that facilitates efficient data retrieval, making it the preferred choice for this study. The data includes
detailed information on card attributes such as card name, set, rarity, colour, mana cost, type, and abilities (terms that will be explained when discussing feature selection). This rich dataset is crucial for understanding each card's intrinsic value and appeal and identifying potential factors influencing card prices.

Although other sources, such as Wizards of the Coast's official database and Scryfall, provide valuable card information, MTGJSON was selected for its ease of use and the comprehensiveness of its API. MTGJSON aggregates data from these sources, ensuring high accuracy and consistency in the information provided.

Integrating card-related data with price and tournament data allows a better understanding of the factors influencing MtG card prices. Attributes such as rarity and card abilities can significantly impact a card's desirability and market value. For instance, rare cards with powerful abilities are often in higher demand and with increased prices. Understanding these relationships is critical for building accurate and robust forecasting models.

## 3.2 Preprocessing

A comprehensive preprocessing step ensures the collected data is suitable for analysis and forecasting. This step is crucial for maintaining the dataset's integrity, consistency, and reliability.

### 3.2.1 Data Cleaning

Most preprocessing focused on standardizing card information across the three data sources, as each had its system. Additionally, addressing missing values was a significant concern.

#### Missing Values

The initial step in the preprocessing stage involved carefully identifying and addressing any missing values within the dataset. MTGJSON provides a complete and well-structured dataset, minimizing the prevalence of missing values. Data scraped from MTGTOP8 are generally well-structured and of high integrity. Most data issues arise from non-professional tournaments or missing decks from certain events. Although missing decks from tournaments are not ideal, the overall impact on the dataset is minimal since this issue does not affect the top eight or sixteen decks of each tournament, and nonprofessional tournaments are excluded from the analysis.

As previously mentioned, price data gathered from MTGGoldfish may occasionally have gaps for low-demand cards during the early months of data collection. To address these gaps, the first few months of price data were excluded from the analysis, only including data from September 2013 onward, where coverage is more reliable and consistent. Additionally, linear interpolation was applied to estimate missing prices for individual cards over short periods within the defined data collection window. Missing data for card prices were only observed for a few cards with low demand and comprised less than 1% of the dataset before interpolation.

## 3.2.2 Feature Engineering

As part of this study, in addition to those scraped from the previous sources, a series of features were engineered from the raw data to support the modelling process.

#### **Tournament-Based Features**

Tournament data was leveraged to create features that reflect cards' competitive performance and popularity.

**Card Appearances:** This feature counts the number of decks that include the card. It was engineered separately for the five main tournament formats (Legacy, Modern, Pioneer, Standard, and Vintage) and the combined values across all formats. The calculation is based on the card's presence in the decklist -the official list players must fill out with the cards they are playing for that tournament- in each format.

$$X_{\rm app}^{(i,j,t)} = \sum_{k=1}^{N^{(j,t)}} D_k^{(j,t)}.$$
(3.1)

Where  $X_{app}^{(I,j,t)}$  is the card *i* appearances feature for format *j* and time *t*, and  $N^{(j,t)}$  is the total number of decks. The variable  $D_k^{(j,t)}$  is an indicator for deck *k* in format *j* and time *t*, where  $D_k^{(j,t)} = 1$  if the deck includes card *i*, and  $D_k^{(j,i)} = 0$  otherwise. **Card Count:** This feature counts the total number of copies of the card across all tournament decks. Like Card Appearances, it was engineered for each main format individually and for their combined values. The calculation considers the number of times a card appears in a decklist.

$$X_{\text{count}}^{(i,j,t)} = \sum_{k=1}^{N^{(j,t)}} C_k^{(i,j,t)}.$$
(3.2)

Where  $X_{\text{count}}^{(i,j,t)}$  is the card *i* count feature for format j and time t,  $N^{(j,t)}$  is the total number of decks. The variable  $C_k^{(i,j,t)}$  is the number of copies of card *i* in deck *k* in format *j* and time *t*.

**Card Legality:** This feature is a boolean flag indicating whether the card is legal in a specific format. It was engineered once for each format, based on the official rules and updates provided by WotC.

$$X_{\text{leg}}^{(i,j,t)} = \begin{cases} 1 & \text{if the card } i \text{ is legal in format } j \text{ at time } t, \\ 0 & \text{if the card } i \text{ is banned in format } j \text{ at time } t. \end{cases}$$
(3.3)

#### Card Meta Features

Features based on card attributes were engineered to capture the intrinsic qualities of each card:

**Card Age:** This feature represents the number of days since the card's release. It captures the idea that older cards may become less available over time.

$$X_{\text{age}}^{(i,t)} = t - \text{Release Date of Card } i. \tag{3.4}$$

Where  $X_{age}^{(i,t)}$  is the age of card *i* at time *t* in days, and both *t* and "Release Date of Card *i*" are the dates expressed in a consistent format.

**Card Rotation Age:** This feature calculates the days until the card is removed from the Standard format, which rotates its legal sets yearly. This metric reflects the potential decline in demand from its removal of the Standard format pool as a card approaches its rotation date. Despite being only one of five tournament legal formats analyzed, Standard represents just under half of the tournament data.

$$X_{\text{rot\_age}}^{(i,t)} = \text{Rotation Date of } \text{Card}i - t$$
(3.5)

Where  $X_{\text{rot}\_age}^{(i,t)}$  is the rotation age of card *i* at time *t* in days, and the "Rotation Date of Card i" is the scheduled date when the card will no longer be legal in the Standard format.

**Print Count:** This feature counts the number of times the card has been reprinted. Since reprinting increases the card's availability, this feature captures the impact of supply on the card's market dynamics.

$$X_{\text{print\_count}}^{(i,t)} = \text{Number of Reprints of Card } i \tag{3.6}$$

Where  $X_{\text{print-count}}^{(i,t)}$  represents the total number of times card *i* has been reprinted across different MtG sets at time *t*.

#### **Card Attribute Features**

MtG is a game of resource management at its core, and a card's playability is always considered in a cost-efficiency ratio. A card's cost is represented by its mana cost, indicated by a series of numbers and symbols often positioned at the card's upper right corner (Figure 3.2). As a card becomes more expensive or restrictive to play compared to others, it becomes less desirable to players, which may affect its demand.

**Colour Count:** This feature counts the number of colours associated with a card. A player must generate resources, or mana, of the corresponding colour to play a card. A player's ability to play a card in MtG is inversely proportional to the number of colours it is associated with.

$$X_{\text{col_count}}^{(i)} = \text{Number of colours associated with card } i$$
 (3.7)

Where  $X_{\text{col.count}}^{(i)}$  is the colour count feature for card *i*.

Mana Pips: This feature counts the number of colour-specific resources required to play a card. A player must use generic or colour-specific re-



Figure 3.2: MtG card with its cost and effect highlighted

sources when paying for a card's cost, representing the colour-specific cost requirement of a card.

$$X_{\text{mana-pips}}^{(i)} =$$
Number of colour-specific mana symbols in the cost of card  $i$ 
(3.8)

Where  $X_{\text{mana-pips}}^{(i)}$  is the number of mana pips for card *i*.

**Total Mana Cost:** This feature represents the overall cost of playing a card. In MtG, maintaining cost efficiency is one of the player's main concerns when building their decks; overcosted cards, those that outperform the average for their cost, tend to be ignored, while undercosted cards, those

that underperform the average for their cost, are prioritized.

$$X_{\rm cmc}^{(i)} = \text{Total mana cost of card } i. \tag{3.9}$$

Where  $X_{\rm cmc}^{(i)}$  is the total mana cost for card *i*.

**Oracle Count:** A card's efficiency is also based on its impact in the game, which is related to its rules and static values, represented by the text in the rules box and, for some card types, the values in the lower right corner of the card. Magic is governed by a core set of game rules, modified or circumvented by a card's specific rules text. The more a card modifies the game's basic rules, the more powerful it becomes. Therefore, this simplified approach to the rules was modelled as follows:

$$X_{\text{oracle}\_\text{count}}^{(i)} = \text{Number of characters in the rules text of card } i, (3.10)$$

where  $X_{\text{oracle}\_\text{count}}^{(i)}$  is the oracle count feature for card *i*, representing the number of characters in its rules text. Although a word count is not directly correlated with card strength, there is a correlation between the design need to explain rules specifications and the length of the rules of a card text [Rosewater, 2002].

**Static Features:** The features described under Section 3.2.2 do not change over time. Once a card is printed, its mechanical attributes remain the same.

Including such features in a time series model is inappropriate due to their zero variance and static values. These features were further transformed to address this issue. A card's efficiency is not measured individually but compared to possible replacements at a given time. It is not about how good a card is but how much better it is compared to others. To capture this comparison, the difference between a card's static value and the average of all cards printed at that particular time was utilized:

$$X_{\rm a}^{(i,t)} = X_{\rm a}^{(i)} - \left(\frac{\sum_{n=1}^{N_t} X_{{\rm a},n,t}}{N_t}\right)$$
(3.11)

Where  $X_{a}^{i}$  is feature *a* of card *i*,  $N_{t}$  is the number of cards at time *t*, and  $X_{a,n,t}$  is feature *a* of card *i* at time *t*.

Combining these engineered features aims to provide the forecasting models with a rich and nuanced dataset that captures the multifaceted nature of MtG card prices. This approach allows a better understanding and prediction of price movements based on comprehensive predictors.

### 3.2.3 Data Transformation

While MtG card prices share some traits with stocks and other traditional financial assets, they exhibit less volatility and predictable patterns due to lower demand and trade volume. The data frequency was adjusted from daily to weekly to avoid long periods with minimal price changes.

This transformation helps smooth out low-magnitude price fluctuations and concentrates the signal in the data, enhancing the effectiveness of the forecasting models. Furthermore, this weekly aggregation aligns with the frequency of MtG tournaments (Figure 3.3), which typically occur weekly, and allows for a significant number of observations between major tournaments held every three to four months.



Figure 3.3: Tournament Frequency Schedule

Similarly, tournament data was consolidated weekly to align with the adjusted price data. Price data was aggregated using the mean, while tournament data was based on the week's total value.

# Chapter 4

# Data

After the database was created, it contained information on over twenty thousand different cards. However, many cards in the MtG card pool are irrelevant for price analysis or have almost zero variance in price. As such, selecting a group of relevant cards upon which the modelling will be done is necessary.

# 4.1 Card Selection Methods

As with any other market, the price of a card is ultimately determined by the number of copies available in the market (supply) and the number of players or collectors interested in acquiring said cards (demand).

# 4.1.1 By Card Supply

WotC often adheres to a schedule when producing new sets, releasing four main sets every year (Figure 4.1). The releases of additional non-commercial sets have no definite pattern. These extra sets are called specialty sets and are designed to cater to specific niches within the MtG community. All main sets, however, are designed to consider tournament play across all game formats. This strategy sets the overall tone for the supply of MtG cards.



Figure 4.1: Heatmap of commercial MtG set releases

Not all printings of a set are equal. As MtG gained popularity, the print numbers grew significantly—from Alpha's initial 2.5 million in 1993 to Fourth Edition's 600 million in 1995 [de Laval, 2020, DeLaney]. Today, a single set generates over one hundred million dollars in profit for WotC [Evans, 2023, DeLaney].

Set releases are not the only concern regarding card availability. As explained in the previous chapter, a card from an earlier set can be reprinted in new sets, increasing its overall availability and possibly impacting its prices. However, a small selection of cards is exempt from possible reprints, such as the Reserved List. Initially created in 1996 to appease the worries of collectors that their cards could drop in value in case of future reprints [Langelett and Wang, 2023], the current format of the Reserved List is that some cards released before October 1999 would no longer be considered for reprinting. As of 2024-08-05, this list is comprised of 565 cards.

When analyzing the price of a card based (Figure 4.2, collected data shows a different distribution pattern based on their release data. Cards printed before 2000, the cutoff date for the Reserved List, present both the highest price and price range, while newer cards have a narrower price distribution.



Figure 4.2: Set price distribution by decade

Considering this, this research will define cards that are part of the Re-

served List, to be described as Group 1. This will allow all Group 1 cards to possess similar supply levels within the MtG secondary market.

### 4.1.2 By Card Demand

While the number of cards printed in MtG history exceeds 45,000 across all sets and nearly 30,000 in main sets, only a fraction of these cards attract player interest. In a single set, most of its cards are, as referred to by members of the MtG community, classified as 'chaff' or 'bulk.'

As a method of selection to exclude these 'sub-prime cards,' since they provide little value to our analysis, we will limit our selection only to cards used in tournaments during our previously defined time window. This is not a perfect subset since there are indeed cards that have little tournament value but are desired as collectibles or for non-sanctioned play. However, with no access to an integrated sales database for a major retailer, this is the best approximation we can make to filter our data.



Figure 4.3: Diagram explaining game formats and their card pools.

Through this filter, we can reduce the list of cards to 4,297, including cards often used in tournaments, called 'staples' by the players, to cards with a single use registered in ten years. Similar to Group 1, which was defined as the cards grouped by supply, these cards will be described as Group 2 for the remainder of this research.



Figure 4.4: Group 2 representation of card clusters

It is important to note that these groups are not exhaustive, as some cards are simultaneously from the Reserved List and used in tournament decks. As such, Group 1 and Group 2 are not mutually exclusive.



Figure 4.5: Groups 1 and 2 Venn Diagram

## 4.2 Data Selection

After the database was created, it contained information on over twenty thousand different cards. To ensure computational efficiency and focus on the most relevant cards for the study, a methodology was developed to filter this extensive dataset down to a select few.

## 4.2.1 Selections for Group 1

Cards on the reserved list represent a unique value proposition within the MtG economy. The company's promise to never reprint such cards increases their collectible appeal compared to other cards. It makes these assets susceptible to their cost-efficiency value, the possible financial gains, and enjoyment that collectors can derive from these assets [Mcinish and Srivastava, 1982, Kleine et al., 2020].





The outcome of the above motivations is a wide difference in price among the reserved list cards. The cards that have no collectible appeal or tournament value are priced under a dollar, while cards that are both iconic to the game's history and desirable for tournament play can be traded for thousands of dollars (Figure 4.6),

While it is impossible to measure collectible interest for specific cards with the data previously collected, tournament data were used to filter the selection for Group 1. As Group 1 and Group 2 are not mutually exclusive, using the intersection of both groups to reduce the asset list allowed the selection of a subset of cards from Group 1 that are more susceptible to external factors in the database. This reduced the initial list of 571 cards to 80 cards.



Figure 4.7: Group 1 final selection diagram

Aligning with the intent of proposing a general model that can be applied to any MtG card, the final list of Group 1 was created to include the most used tournament card, the most iconic card in MtG history, and eight cards picked randomly from the remaining ones. This was defined as Group 1.

Card Code	$X_{\mathbf{app}}$	$X_{\mathbf{count}}$	D%	$y_1$	$y_t$	σ
X2ED.233	257	257	0.8924	11,985.02	16,995.00	5,879.24
X2ED.48	182	182	0.6319	$3,\!299.99$	$5,\!195.00$	1,044.99
X2ED.84	180	180	0.6250	$3,\!519.99$	4,999.99	$1,\!896.28$
MIR.307	73	247	0.2535	611.86	465.98	61.86
X2ED.275	51	51	0.1771	$1,\!172.50$	$1,\!259.99$	359.12
UDS.135	15	60	0.0133	112.97	120.07	21.97
EXO.72	4	4	0.0035	34.55	40.47	14.75
UDS.1	3	4	0.0027	80.32	68.76	23.55
MIR.241	3	3	0.0027	4.60	8.86	8.67
LEG.113	2	2	0.0018	730.49	675.03	403.08

Table 4.1: Card Selection for Group 1

**Notes:** D% represents the percentage of decks that had the card in their legal pool and used it, while  $\sigma$  is the standard deviation of the series.

#### 4.2.2 Selections for Group 2

A different approach was taken when defining the selection criteria for Group 2. Unlike Group 1 cards, which are legal in just two non-rotating formats, Group 2 includes cards used across all formats—both eternal and rotating. This makes Group 2 more sensitive to time, as cards with brief periods of heavy play shouldn't be compared to those with lower but sustained play based solely on card count or deck appearances.

To address this issue, the focus was shifted to understanding why a card is effective during tournament play. From a competitive play perspective, the appeal of a card lies in its overall power compared to its cost. This power can be categorized as individual power, the card's inherent strength, or synergistic power, which is how well it pairs with others in a game strategy [Bjørke and Fludal, 2017, Alvin et al., 2021, Ward et al., 2021]. Several approaches have already been explored to model synergistic relationships between MtG cards. Graph theory allows for a stochastic metric to represent deck synergy [Chodoriwsky, 2006, Alvin et al., 2021] through edges and nodes of the deck network. Another approach previously used involved the frequency of card pairs selected by a player and using the probability of each pair being selected to determine the relationship [Ward et al., 2021]. However, neither approach is a perfect fit for this research.

The synergy metric developed by Chodoriwsky allows for comparisons between decks. Still, it cannot create a more extensive network of all cards in a meta-game or identify the boundaries that define each game strategy. On the other hand, Ward's approach enabled the identification of relationships within a single edition set, each comprised of approximately two hundred cards, with hundreds or thousands of data points for each set. The data in this research requires a solution applicable to a dimension of 7,359 different decks that, collectively, use 4,297 cards in total.

Since the goal is not to model the relationship of all cards in the tournament data but to select a few cards that could represent this group, a focused approach was decided upon. By clustering the individual cards from the tournament data, each cluster would represent an overarching game strategy; since distance is the relevant metric used, the most central points of each cluster would represent the core cards for each game strategy.

#### Clustering

The entire process is represented in Figure 4.8. The first stages were dedicated to preprocessing prior to the clustering algorithm. To cluster the tournament data and visualize this multidimensionality, a matrix  $M_{m \times n}$  was constructed, where *m* represents the list of cards, *n* the list of decks, and (m, n) is the number of copies of that card in that deck.



Figure 4.8: Sequence of states for the clustering process.

A K-means cluster algorithm (Algorithm 1) was then used to create the individual clusters within the data. This clustering process used the overall distance to each centroid to identify the best number of clusters. A method of vector quantization, it aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, or centroid [Hastie et al., 2009].

Given a set of observations  $(x_1, x_2, \ldots, x_n)$ , where each observation is a d dimensional real vector, k-means clustering aims to partition the n observations into  $k \leq n$  set  $\mathbf{B} = \{S_1, S_2, \ldots, S_k\}$  so as to minimize the withincluster sum of squares (WCSS). Formally, the objective is to find:

$$\arg\min_{S} \sum_{i=1}^{k} \sum_{x \in S_{i}} \|x - \mu_{i}\|^{2} = \arg\min_{S} \sum_{i=1}^{k} |S_{i}| \operatorname{Var}S_{i},$$
(4.1)

where  $\mu_i$  is the mean, also called centroid, of points in  $S_i$ ,

$$\mu_i = \frac{1}{|S_i|} \sum_{x \in S_i} x, \tag{4.2}$$

 $|S_i|$  is the size of  $S_i$ , and ||.|| is the usual  $L^2$  norm.

Algor	<b>ithm 1</b> Clustering Algorithm	
Requ	ire: $D = \{x_1, x_2, \dots, x_n\}$	$\triangleright$ Set of n data items
	k clusters	$\triangleright$ Number of clusters
1: <b>fu</b>	$\mathbf{nction} \ \mathrm{K}_{-}\mathrm{MEANS}(D)$	
2:	Arbitrarily select $k$ data points as	initial centroids $\mu_1, \mu_2, \ldots, \mu_k$
3:	repeat	$\triangleright$ Iterate over l iterations
4:	for each data point $x_n$ do	$\triangleright$ Assign each x to nearest k
5:	for each cluster $k$ do	
6:	if $k == \arg \min_k   x_n - \mu $	$\ u_k^{l-1}\ ^2$ then
7:	$r_{nk} = 1$	$\triangleright$ Assign $x_n$ to cluster $S_k$
8:	else	
9:	$r_{nk} = 0$	
10:	for each cluster $k$ do	
11:	$\mu_k^l = \frac{1}{\sum r_{nk}} \sum_{x \in S_k} r_{nk} x$	$\triangleright$ Update $\mu_k$ as the mean of $S_k$
12:	$\mathbf{until} \ r_{nk}^l == r_{nk}^{l-1}$	
13:	return $(\{S_1, S_2, \ldots, S_k\}, \{\mu_1, \mu_2, \ldots, S_k\})$	$(\ldots, \mu_k\})$

While it is possible to define k prior to performing Equation 4.1 optimization, it is also possible to automate this process to obtain the optimal number of clusters (Algorithm 2). By defining a select metric, the k-means optimization is iterated over until the selected metric itself is minimized. This research performed this method for two different metrics: average within-cluster distance ( $\delta$ ) and maximum inter-cluster distance ( $\eta$ ).

$$\delta^{(i)} = \frac{1}{|S_i|} \sum_{x \in S_i} ||x - \mu_i||, \qquad (4.3)$$



$$\eta = \max_{i,j} \ \|\mu_i - \mu_j\|.$$
(4.4)

Figure 4.9: K-means result for Modern in 2017

Each metric resulted in a different number of clusters, as exemplified in Figure 4.9. After analyzing the results individually for each subset of year and game format, the average within-cluster distance presented the number of clusters more accurately by domain knowledge. Hence, it was the metric chosen. Five cards closest to each centroid were selected once the optimal number of clusters was determined.

After this process, the final selection included a list of 98 cards. Similar to the approach used in Group 1, the card with the most tournament uses was automatically selected, and nine other cards were randomly chosen.

Algorithm 2 Cluster Optimization

1: function OPTM\_K( $M^{j,t}$ )  $\triangleright M_{m \times n}$  matrix for format j and year t 2:  $\triangleleft$ k = 33:  $\begin{array}{l} k_{\rm opt} = k \\ \Delta_{\rm min}^0 = \infty \end{array}$ 4: 5:repeat 6:  $\begin{array}{l} \mathbf{K}\_\mathrm{MEANS}(D = M^{j,t}, k) & \triangleright \ Run \ K\text{-means for } k \ clusters \\ \delta^{(i)} = \frac{1}{|S_i|} \sum_{x \in S_i} \|x - \mu_i\| & \triangleright \ Where \ \delta^{(i)} \ is \ average \ distance \ for \ S_i \end{array}$ 7: 8:  $\Delta^l = \frac{1}{k} \sum_{i=1}^k \delta^{(i)}$  $\triangleright$  Where  $\Delta^l$  is average distance for all clusters 9:  $egin{array}{lll} extsf{if} & \Delta^l < \Delta_{\min} extsf{then} \ & \Delta_{\min} = \Delta^l \end{array}$ 10:11:  $k_{\rm opt} = k$ 12: $\overline{k} = k + 1$ 13:until  $\Delta^l \ge \Delta^{l-1}$ 14:  $C_i = \{x : x \in S_i, \|x - \mu_i\|\}$  for  $i = 1, \dots, k_{\text{opt}}$ 15:Select  $\{x_{(1)}, x_{(2)}, \dots, x_{(5)}\} \subseteq C_i$ 16:return  $\{x_{(1)}, x_{(2)}, \dots, x_{(5)}\}$   $\forall i = 1, \dots, k_{\text{opt}}$ 17:

Table $4.2$ :	Card	Selection	for	Group 2	
---------------	------	-----------	-----	---------	--

Card Code	$X_{\mathbf{app}}$	$X_{\mathbf{count}}$	D%	$y_1$	$y_t$	$\sigma$
LRW.145	1341	3696	0.3674	19.17	21.78	1.26
KDL.110	468	883	0.1244	6.82	2.94	1.04
IKO.67	439	1230	0.1156	8.34	8.24	1.35
KDL.234	300	412	0.0797	7.68	2.92	1.29
GPT.52	271	908	0.0744	9.37	9.13	0.21
IKO.88	187	359	0.0492	1.48	1.06	0.22
DTK.150	175	367	0.0485	2.47	2.87	0.38
THS.180	162	558	0.0444	5.07	3.96	0.48
VOW.63	108	144	0.0309	2.78	3.54	0.74
INV.226	48	156	0.0128	9.25	5.14	1.01

**Notes:** D% represents the percentage of decks that had the card in their legal pool and used it, while  $\sigma$  is the standard deviation of the series.

## 4.3 Data Summary

This study focuses on analyzing two distinct groups of MtG cards selected from a larger pool of over 25,000. These groups were chosen based on supply constraints (Group 1) and demand characteristics (Group 2), allowing for a focused examination of card price dynamics.

**Group 1 (Supply-Based Selection):** Group 1 consists of cards from the Reserved List, a selection of cards that will never be reprinted, thereby limiting their supply. This group's initial size was 571 cards, which were then filtered to 80 cards based on their tournament appearances. From this filtered set, the final 10 cards were chosen, including the most-used tournament card, the most iconic card in MtG history, and eight randomly selected cards from the remaining pool. These cards represent assets that are influenced primarily by collectible value and constrained supply.

Group 2 (Demand-Based Selection): Group 2 focuses on cards frequently used in tournaments, capturing demand-driven price changes. Starting with 4,297 cards, a clustering method was applied to group cards based on their synergy and usage in various decks. After clustering, 98 cards remained, from which a final selection of 10 cards was made. As in Group 1, the card with the highest tournament usage was automatically selected, and nine additional cards were chosen randomly from the remaining clustered cards.

The final analysis will be based on a total of 20 cards, with 10 cards from Group 1 and 10 from Group 2. Both groups represent critical aspects of the MtG secondary market, from the limited supply of collectible cards to the fluctuating demand for tournament staples. Line plots for each card price series are included in Appendix A.

**Chosen Cards:** For **Group 1**, the cards selected are highly representative of the Reserved List's collectible and iconic status. For example, the card **X2ED.233** appears in 257 decks, making it one of the most desirable due to its competitive utility, while **LEG.113**, with only 2 appearances, reflects a rarity-driven value with minimal tournament play.

For Group 2, cards like LRW.145 and KDL.110 reflect the highest tournament usage, with appearances in hundreds of decks, while others like VOW.63 or INV.226 reflect cards that play niche roles in certain game strategies. Each card provides a glimpse into the demand dynamics across different formats and game strategies.

This dual approach to card selection ensures a comprehensive examination of the MtG secondary market, combining both the scarcity of collectible assets and the practical utility of frequently used cards.

# Chapter 5

# Methodology

The methodology chapter outlines this research's various approaches and techniques to forecast time series data. This chapter is divided into three primary sections: Statistical Models for Forecasting; Machine Learning Models for Forecasting; and Proposed Model. The Statistical Models for Forecasting section explores traditional methods that rely on well-established statistical models to identify and predict patterns within data. The Machine Learning Models for Forecasting section introduces advanced techniques that use the power of neural networks to capture complex, non-linear relationships in the data. By combining these approaches, a Proposed Model for Forecast Combination Based Neural Network aims to develop a robust and accurate forecasting model that integrates the strengths of both statistical and machine learning methods. Each section will detail the underlying theories, specific models, and optimization strategies used to enhance forecast accuracy.

## 5.1 Statistical Models for Forecasting

This section covers common algorithms used for forecasting in statistics. The code for implementing these models was written in **R** using the *fable* package. This package provides a comprehensive suite of tools for time series analysis, including functions for estimating and forecasting with ARIMA and RW models.

### 5.1.1 Autoregressive Model

Regression analysis, a concept rooted in statistical modelling, is a set of processes used to explore the relationships between a dependent variable and one or more independent variables. George Box and Gwilym Jenkins [1968] introduced the idea of an autoregressive model, where a time series—consisting of equally spaced values—can be connected to its past values (known as an autoregressive process) or to the differences between these values (referred to as a moving average process). These approaches are integrated within the ARIMA framework and designed to capture the ongoing trends in the time series and the random fluctuations that may occur.

#### Autoregressive Integrated Moving Average (ARIMA)

Consider a time series y with equally spaced observations at times  $t, t - 1, \ldots, t - n$ , denoted as  $y_t, y_{t-1}, \ldots, y_{t-n}$ . Let  $a_t, a_{t-1}, \ldots, a_{t-n}$  represent a "white noise" series, where these values are random, uncorrelated, and have a constant variance  $\sigma_a^2$  with a mean of zero. Box and Jenkins developed the autoregressive model by relating the deviation from the mean,  $\dot{w}_t = w_t - \mu$ ,

to previous deviations and the white noise term  $a_t$ . For instance:

$$\begin{cases}
A.R. 1: \quad \dot{y}_t = \phi_1 \dot{y}_{t-1} + a_t, \\
A.R. 2: \quad \dot{y}_t = \phi_1 \dot{y}_{t-1} + \phi_2 \dot{y}_{t-2} + a_t,
\end{cases}$$
(5.1)

represent autoregressive models of order 1 and 2, respectively. The coefficients  $\phi_1$  and  $\phi_2$  represent the weights applied to the previous observations in the time series. These coefficients determine how much influence past values have on the current value. A positive  $\phi_n$  value suggests that an increase in the previous value leads to an increase in the current value, while a negative  $\phi_n$  value indicates an inverse relationship. The magnitude of these coefficients indicates the strength of the relationship between past and present values.

It is important to note that the time series should be stationary for an autoregressive model to be valid. Its statistical properties, such as mean and variance, remain constant over time. This stationarity assumption ensures that the relationship between past and present values remains consistent.

Similarly,  $\dot{y}_t$  can be linked to the white noise series and its past values, leading to moving average models:

$$\begin{cases} M.A. 1: \quad \dot{y}_t = a_t + \theta_1 a_{t-1}, \\ M.A. 2: \quad \dot{y}_t = a_t + \theta_1 a_{t-1} + \theta_2 a_{t-2}, \end{cases}$$
(5.2)

represent moving average models of order 1 and 2, respectively. The coefficients  $\theta_1$  and  $\theta_2$  represent the weights applied to the past white noise terms in

the moving average models. These coefficients determine how past random shocks or errors influence the current value of the series. A positive  $\theta_n$  value means that a positive shock in the past will increase the current value, while a negative  $\theta_n$  value means that a positive shock in the past will decrease the current value. The magnitude of these coefficients reflects the degree of impact that past shocks have on the current observation.

In addition to the autoregressive (AR) and moving average (MA) components, ARIMA models also include an integration (I) step to account for trends in the data. This involves differencing the series to make it stationary, meaning the statistical properties remain constant over time. For example, taking the difference between consecutive values in the series yields:

$$\begin{cases} I. 1: \quad \dot{y}_t = y_t - y_{t-1}, \\ I. 2: \quad \dot{y}_t = y_t - 2 \times y_{t-1} + y_{t-2}, \end{cases}$$
(5.3)

representing integration of order 1 and 2, respectively.

Combining these three components—autoregression, moving average, and integration—creates a generalized model that effectively captures both shortterm dependencies and long-term trends in the data. The resulting ARIMA model can be expressed as:

$$\dot{y}_t = \mu + \phi_1 \dot{y}_{t-1} + \dots + \phi_p \dot{y}_{t-p} - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q} + a_t, \tag{5.4}$$

where  $\phi_1, \ldots, \phi_p$  are the coefficients of the autoregressive terms,  $a_t$  is the

white noise error term,  $\theta_1, \ldots, \theta_q$  are the coefficients of the moving average terms, and  $\mu$  is the mean of the series.

This formula represents an ARIMA(p, d, q) model, where p is the order of the autoregressive component, d is the order of differencing, and q is the order of the moving average component. The integration step (differencing) is applied before fitting the AR and MA components. This research used the function ARIMA from the fable package to select these parameters.

## Autoregressive Integrated Moving Average with Predictor Features (ARIMAX)

The ARIMAX (Autoregressive Integrated Moving Average with Predictor Features) model extends the ARIMA framework by incorporating external variables that may influence the time series. While ARIMA models focus solely on the internal structure of the series—specifically its autoregressive (AR), moving average (MA), and integration (I) components—the ARIMAX model adds layer by considering the impact of other explanatory variables (often referred to as "predictor" features) [Box and Tiao, 1975].

In an ARIMAX model, the external variables are included to account for factors outside the time series that could affect its behaviour. These predictor features are typically denoted as  $X_t, X_{t-1}, \ldots, X_{t-n}$ , where  $X_t$  represents the value of an external variable at time t.

The general form of the ARIMAX model is similar to that of ARIMA but with the addition of a term that includes the predictor features. For example, an ARIMAX model of order (p, d, q) can be written as:

$$\dot{y}_{t} = \phi_{1}\dot{y}_{t-1} + \dots + \phi_{p}\dot{y}_{t-p} + \theta_{1}a_{t-1} + \dots + \theta_{q}a_{t-q} + \beta_{1}X_{t} + \dots + \beta_{m}X_{t-m} + a_{t},$$
(5.5)

here,  $\beta_1, \ldots, \beta_m$  are coefficients corresponding to the predictor features  $X_t, \ldots, X_{t-m}$ .

Incorporating external variables allows the ARIMAX model to capture additional influences on the time series, providing a more comprehensive understanding and more accurate forecasting when these external factors are significant.

While ARIMA models are particularly effective when the time series is driven primarily by its internal dynamics, ARIMAX models are helpful when external factors are known to substantially impact the analyzed series. Like with traditional ARIMA, the function *ARIMA* from the *fable* package was used, but this time, including the predictor features as the parameters in the formula.

### 5.1.2 Random Walk

Popularized by Fama [1970] in his review of efficient market theory, the term Random Walk represents the concept that the current price of an asset perfectly reflects its available information. Under this assumption, price changes are independent and successive changes are identically distributed. Combined, these two hypotheses constitute the Random Walk model, which can be mathematically expressed as:

$$y_t = y_{t-1} + a_t, (5.6)$$

where  $y_t$  is the value of the time series at time t,  $y_{t-1}$  is the value at time t-1, and  $a_t$  is a white noise error term with mean zero and constant variance.

The Random Walk model assumes that the current value is the best predictor for the future, with only the random error term accounting for changes. This means that the series does not tend to return to a long-term mean, making it non-stationary. Despite its simplicity, the Random Walk model is widely used in financial modelling and other fields where the future is highly uncertain and only past values are considered in predictions. The function RW from the *fable* package was used for this.

#### Random Walk with Drift

A Random Walk with Drift extends the basic Random Walk model by introducing a constant term known as the "drift," which accounts for a consistent trend in the data, either upward or downward [Nelson and Plosser, 1982]. The model can be mathematically expressed as:

$$y_t = \rho \times y_{t-1} + \mu + a_t,$$
 (5.7)

where  $\mu$  is the drift parameter, indicating the average change in the time series at each step,  $y_{t-1}$  is the previous value, and  $a_t$  is the white noise error term. The parameter  $\rho$  represents the coefficient of the previous value, typically set to 1 in a pure Random Walk model, indicating that the series follows a path dependent on its past value plus any drift and random noise.

The inclusion of  $\mu$  allows the model to capture trends in the data, making it more adaptable to real-world scenarios where such trends are often observed. This adjustment makes the Random Walk with Drift a more versatile model, capable of describing series that exhibit random fluctuations and an underlying directional movement. For this calculation, the drift was used as part of the formula for the *RW* function from the *fable* package.

**Relation to ARIMA:** It is important to note that a Random Walk can be seen as a particular case of the ARIMA model. Precisely, a Random Walk corresponds to an ARIMA(0, 1, 0) model, where there are no autoregressive terms (p = 0), the series is differenced once to achieve stationarity (d = 1), and there are no moving average terms (q = 0). Similarly, a Random Walk with Drift can represent an ARIMA(0, 1, 0) with a non-zero mean. This connection highlights the Random Walk's simplicity as a foundational model within the broader ARIMA framework.

## 5.2 Machine Learning Models for Forecasting

The advent of machine learning has introduced new methods for time series forecasting, allowing for the modelling of complex, non-linear relationships in data. These approaches can complement traditional statistical models, offering alternative and often more powerful tools for forecasting when dealing with large datasets or when the data-generating process is not fully understood.

## 5.2.1 Neural Networks Models

Initially proposed by McCulloch and Pitts [1990], the concept of how neurons might work together to perform complex tasks laid the foundation for the computational model that would one day evolve into the field of artificial neural networks (ANNs). Based on this biological neuron concept, ANNs originated with the mathematical modelling that, after receiving an input signal, the neuron is activated once these signals reach a threshold, sending a signal further down the chain.



Figure 5.1: Representation of an Artificial Neural Network Diagram.

Rumelhart et al. expanded this framework by describing a backpropagation algorithm capable of adjusting the weights of connections between neurons to minimize the difference between the actual and desired output vector of a neural network Rumelhart et al. [1986]. The learning process proposed by Rumelhart et al. considers a set of input-output functions represented in the paper as:

$$\mathbf{x}_j = \sum_i \mathbf{y}_j w_{j,i},\tag{5.8}$$

$$\mathbf{y}_j = \frac{1}{1 + e^{-\mathbf{x}_j}},\tag{5.9}$$

where  $x_j$  is the weighted sum of the inputs for neuron j,  $y_j$  is the output of neuron j,  $w_{j,i}$  is the weight between neuron j and its input i, and e is the base of the natural logarithm.

Combining (5.8) and (5.9), it is possible to calculate the exact values and weights at any point in the neural network:

$$\mathbf{x}_{j}^{(L)} = \sum_{l=1}^{L} \sum_{i} \mathbf{y}_{i}^{(l-1)} w_{i,j}^{(l)}.$$
 (5.10)

Here  $\mathbf{x}_{j}^{(L)}$  is the cumulative weighted sum for neuron j at layer L, l represents each layer leading up to L,  $y_{i}^{(l-1)}$  is the output from the previous layer (l-1), and  $w_{i,j}^{(l)}$  is the weight between neuron i in layer (l-1) and neuron j in layer l.

This learning process then allows for a set of weights that ensures that each generated output vector is as close to the desired vector as possible. The total error is computed by calculating every case's actual and desired output vector for a finite, fixed set of input-output cases. The total error (E) then is defined as

$$\mathbf{E} = \frac{1}{2} \sum_{c} \sum_{j} (\mathbf{y}_{(j,c)} - \mathbf{b}_{(j,c)})^2, \qquad (5.11)$$

where E is the total error across all cases, c represents each case in the dataset, j represents each output neuron,  $y_{(j,c)}$  is the actual output of neuron j for case c, and  $b_{(j,c)}$  is the desired output of neuron j for case c. The error term  $(y_{(j,c)}-b_{(j,c)})^2$  measures the squared difference between the actual and desired outputs and the factor of  $\frac{1}{2}$  is included to simplify the derivative calculation during backpropagation.

#### Neural Network Autoregression (NNETAR)

The NNETAR model extends the standard ANN specifically designed for time series forecasting. It stands for Neural Network Autoregression and integrates the autoregressive (AR) component, commonly used in time series analysis, into the structure of an ANN. The NNETAR model utilizes lagged observations of the target variable as inputs to the ANN, making it particularly effective at capturing complex nonlinear relationships within the time series data.

In an NNETAR model, the input layer consists of the lagged values of the target variable,  $y_{t-1}, y_{t-2}, \ldots, y_{t-p}$ , where p is the number of lags selected. The hidden layers process these inputs using a set of weights and activation functions, similar to the general structure of ANNs described earlier. The output layer provides the forecasted value, denoted as  $\hat{y}_t$ , for the target variable at time t.

In this context, the NNETAR model is often referred to using the no-
tation NNETAR(p, k), where p is the number of lagged inputs and k is the number of neurons in the hidden layer. For example, an NNETAR(4,7) (Figure 5.2) model uses the last four observations  $(y_{t-1}, y_{t-2}, \ldots, y_{t-7})$  as inputs to predict  $y_t$ , with seven neurons in the hidden layer. When k = 0, an NNETAR(p, 0) model is equivalent to an ARIMA(p, 0, 0) model without parameter restrictions to ensure stationarity.



Figure 5.2: Representation of a NNETAR(4,7) Diagram.

The **NNETAR** function in *fable* estimates an NNETAR $(p, P, k)_m$  model, where p refers to the number of non-seasonal lags, P represents the seasonal lags, and k indicates the number of neurons in the hidden layer. For nonseasonal data, p is automatically selected as the optimal number of lags based on the AIC of a linear AR(p) model. For seasonal data, the default value for P is 1, while p is determined using the optimal linear model applied to the seasonally adjusted series. If k is unspecified, it is calculated as  $k = \frac{p+P+1}{2}$ , rounded to the nearest integer [Hyndman and Athanasopoulos, 2021].

The learning process in NNETAR involves adjusting the weights through backpropagation to minimize the forecast error over the training period. The model is compelling for its ability to capture both linear and nonlinear patterns in the data, making it a versatile tool for various forecasting applications.

#### Neural Network Autoregression with Predictor Inputs (NNETARX)

The NNETARX model extends the NNETAR framework by incorporating predictor inputs (additional variables) into the neural network structure. These predictor features denoted as  $x_{t,1}, x_{t,2}, \ldots, x_{t,n}$  represent external factors that might influence the target variable, adding further predictive power to the model.

In NNETARX, the input layer includes both the lagged values of the target variable,  $y_{t-1}, y_{t-2}, \ldots, y_{t-p}$ , and the current or lagged values of the predictor features. The network processes these combined inputs through its hidden layers, allowing it to model the relationship between the target variable and both its past values and the external factors.

The output layer of the NNETARX model provides the forecasted value  $\hat{y}_t$  at time t, while the learning process adjusts the weights for both the autoregressive and predictor inputs. Including predictor features allows the NNETARX model to account for additional influences on the target variable, often leading to more accurate forecasts when such relationships exist. The *NBETAR* function from the *fable* package was used while including the

predictor features in the formula parameter.

### 5.2.2 Forecast Combination

In many practical applications, combining multiple forecasts has improved accuracy compared to relying on a single model. Forecast combination involves aggregating the predictions of several models to create a more robust and reliable forecast, leveraging the strengths of different methodologies. Over the past fifty years, the forecast combination has evolved from a popular approach to a well-established concept in the field of forecasting, supported by a rich body of literature. The foundational work by Bates and Granger ([1969]) laid the groundwork for this approach, demonstrating its effectiveness in enhancing forecast accuracy.

This approach leverages the strengths of different models, reducing the impact of any single model's weaknesses. One straightforward technique is simple averaging, where the forecasts from various models are averaged to generate a combined forecast. This method assumes that each model contributes equally to the final prediction.

Alternatively, weighted averaging assigns a weight to each model's forecast, reflecting its relative importance or performance. In this approach, the combined forecast is a weighted sum of the individual model forecasts, with the weights typically summing to one. This method allows for more influence from models that have historically performed better, potentially improving the overall accuracy of the combined forecast.

# 5.3 Proposed Model: Forecast Combination Based Neural Network

This research's proposed model is a forecast combination-based neural network model. This approach combines the forecast of selected base models, through weighted average, into a new series of time-based observations such as  $\tilde{y} = (y_1, y_2, \ldots, y_{n_1}, \hat{y}_{n_1+1}, \ldots, \hat{y}_{n_1+n_2})$ , where  $(y_1, y_2, \ldots, y_{n_1})$  is the  $n_1$  actual values of the time series,  $(\hat{y}_{n_1+1}, \ldots, \hat{y}_{n_1+n_2})$  is the  $n_2$  point forecasts values calculated from a single-model or forecast combination model. Then, the new series  $\tilde{y}$  is used to build the proposed neural network model. The detailed process for this combination is described as follows.

### 5.3.1 Establishment of Base Models

As a first step in combining forecasts, it is essential to establish the base models that will contribute to the final prediction. In this process, m different forecasting models, denoted as  $M_j$  (j = 1, 2, ..., m), are developed using the available training data. These models can vary in their underlying methodology, such as autoregressive integrated moving average (ARIMA), neural networks, or other machine learning approaches. For this research, the methods chosen are the ones previously discussed (ARIMA, ARIMAX, Random Walk, Random Walk with Drift, NNETAR, and NNETARX).

For each model  $M_j$ , a series of point forecasts  $\hat{y}_t^{(j)}$ , (j = 1, 2, ..., m) is generated for each time period t in the forecasting horizon. These forecasts are based on the model's specific algorithm and are intended to capture the underlying patterns in the data. The accuracy of each base model is evaluated using historical data, allowing for the assessment of the strengths and weaknesses of each model in isolation.

### 5.3.2 Determination of Weights

The following step is to use these forecasts to determine the weight for this specific combination. Determining weights is a critical step in combining forecasts, especially when using weighted averaging. The goal is to assign weights  $w_j$  to each model  $M_j$  to maximize the accuracy of the combined forecast.

Weights can be determined through various methods, such as minimizing the combined forecast's mean squared error (MSE) on a validation dataset. This research uses a Genetic Algorithm (GA) to determine the optimal weights for combining forecasts before applying stacking, as proposed by Yi-Chung Hu [2021]. Still, rather than optimizing for Coverage Width Criterion (CWC) and performing interval forecasting, this study will focus on optimizing the weights for Root Mean Square Error (RMSE) and performing point forecasting. This approach is designed to achieve a more precise and effective forecast by integrating the benefits of both GA and stacking methodologies.

GAs use natural selection principles to iteratively search for the optimal solution, with the fitness function typically defined as forecasting accuracy. The GA package in  $\mathbf{R}$  was employed to discover the optimal weights that minimized the RMSE of the combined forecast. For every model  $M_j$  of the selected combination, its forecasted series  $\hat{y}^{(j)}$  was used as an input parameter to minimize the error (RMSE) to the actual value y. Various parameters, such as population size, mutation rates, and crossover methods, can influence the performance of the GA. In this implementation, the total number of generations was set to 1000, with a population size of 150. The mutation rate was set to 0.1, and the *gareal\_blxCrossover* method was selected for crossover. These parameters were chosen based on experimentation to balance the exploration and exploitation capabilities of the algorithm. Upon completion, the GA algorithm identified the optimal weight for the selected combination,

RMSE = 
$$\sqrt{\frac{1}{T} \sum_{t=1}^{T} (\sum_{j=1}^{m} w_j \times \hat{y}_t^{(j)} - y_t)^2}$$
 (5.12)

where  $w_j$  is the weight given to model  $M_j$ ,  $\hat{y}_t^{(j)}$  is the base forecast value of model  $M_j$  and  $y_t$  is the actual value for time t. Equation (5.12) is then calculated a number of times equal to the selected population over the number of specified generations, improving  $w_j$  at each new generation until the optimal value is found.

### 5.3.3 Combination of Point Forecasts

Once the base models have been established and the weights defined, the next step is to combine their point forecasts to produce a single, more accurate forecast. The combination of point forecasts involves aggregating the predictions from each base model  $M_j$ , where j = 1, 2, ..., m.

$$\hat{y}_t^{\text{com}} = \sum_{j=1}^m \frac{w_j \hat{y}_t^{(j)}}{\sum_{j=1}^m w_j}.$$
(5.13)

This method assigns a specific weight  $w_j$  to each model  $M_j$ , reflecting its reliability or historical accuracy. The weights  $w_j$  are chosen such that  $0 \le w_j \le 1$  for all j, and  $\sum_{j=1}^m w_j = 1$ , ensuring that they are relative rather than absolute. This allows for greater influence from models that have demonstrated superior forecasting ability. The weighted average approach can be particularly effective when some models consistently outperform others under specific conditions.

Assume that, we have relative weight  $w_j$  of  $M_j$  (j = 1, 2) models. Then, for example, the point forecast based on a two-model combination can be defined as:

$$\hat{y}_t^{\text{com}} = \frac{w_1}{w_1 + w_2} \hat{y}_t^{(1)} + \frac{w_2}{w_1 + w_2} \hat{y}_t^{(2)}.$$
(5.14)

Where  $n_1 + 1 < t < n_1 + n_2$ . Similarly, the point forecast based on a three-model combination where  $M_j$  (j = 1, 2, 3) can be obtained as:

$$\hat{y}_t^{\text{com}} = \frac{w_1}{w_1 + w_2 + w_3} \hat{y}_t^{(1)} + \frac{w_2}{w_1 + w_2 + w_3} \hat{y}_t^{(2)} + \frac{w_3}{w_1 + w_2 + w_3} \hat{y}_t^{(3)}.$$
 (5.15)

Similarly, four or more than four models can be constructed following equation (5.15). Then a new sequence can be constructed as  $\tilde{y}_t = (y_1, \ldots, y_{n_1}, \hat{y}_{n_1+1}^{\text{com}}, \ldots, \hat{y}_{n_1+n_2}^{\text{com}})$  which is a combined sequence of  $n_1$  original samples and  $n_2$  combined forecasts and can be used to construct the proposed model.

## 5.3.4 A data-driven Forecast combination based Neural Network Model

The proposed method (Figure 5.3) is a data-driven approach to selecting the best model combination for a specific time series. Although the first step of generating the base forecast values, is consistent across all-time series, the subsequent steps are tailored to optimize the combination for each unique series.

After generating the base forecasts, the selected combination of models, as determined by the Genetic Algorithm (GA), creates a new time series of combined forecasts,  $\tilde{y}_t$ . This combined series serves as an input for the Neural Network, which is specifically trained to improve the forecast by leveraging the combined strengths of the selected models.

The ANN is structured to take both the combined forecast series  $\tilde{y}_t$  and the original time series data  $y_t$  as inputs. This allows the ANN to capture the underlying patterns in the data while also adjusting for any potential biases or errors present in the combined forecasts. The ANN then outputs a refined forecast value that is expected to outperform the individual base models and their weighted combination.

The architecture (Figure 5.3) is designed to be flexible, with the ability to adjust the number of layers and neurons depending on the complexity of the time series data. During the training phase, the ANN iteratively adjusts its internal weights to minimize the forecast error, typically measured by metrics such as Root Mean Squared Error (RMSE) or Mean Absolute Percentage Error (MAPE). RMSE measures the square root of the average squared differences between the predicted and actual values, while MAPE represents the average of the absolute percentage differences between the predicted and actual values. The final model is then evaluated on a validation set to ensure its generalizability and robustness.

This data-driven approach allows for a dynamic and adaptable forecasting model that can respond to the specific characteristics of each time series, ultimately providing more accurate and reliable forecasts.



Figure 5.3: Forecast combination based ANN proposed model diagram.

# Chapter 6

# **Experiments and Evaluation**

The preceding chapters detailed the data collection process and provided the theoretical foundation for this research. This chapter presents the analysis of the selected dataset and the results of the proposed methods. It is structured into two parts: the first part focuses on the experimental setting conducted on the selected dataset, and the second part presents the results from the modelling and experiments.

## 6.1 Analysis

This section delves into the initial preparation of the dataset, highlighting the essential preprocessing steps undertaken to ensure its suitability for predictive modelling.

### 6.1.1 Experimental Settings

Before proceeding with the modelling phase, several preprocessing steps were applied to the dataset to ensure data quality and model performance, following best practices in the literature.

Missing values in key features related to legality, restrictions, and card rotation age were treated by imputing zeroes. This approach is beneficial when the absence of data can be interpreted as a meaningful zero, as it preserves the information that these features might contribute to the model. According to Little and Rubin [2019], imputing missing values with zero is an appropriate strategy when the absence is informative and not simply missing at random. The previous observations, despite their representation as NA in the database, are not real missing values but were stored as such for dates when either the card or the format weren't officially released yet.

Features with no variability (constant values) were excluded, as they do not provide any discriminatory power for the model. Including such features can inflate the dimensionality without contributing to predictive accuracy, a concept well-documented in the feature selection literature [Guyon and Elisseeff, 2003].

To address multicollinearity, features exhibiting perfect multicollinearity were identified and removed using a stringent correlation threshold of 0.999. Multicollinearity can significantly distort the interpretation of coefficients in regression models and lead to inflated standard errors, making it crucial to eliminate perfectly correlated features [Kutner et al., 2004]. A high correlation cutoff (close to 1) is supported by Patel et al. [2011], who recommend this approach to maintain model stability and interpretability. Finally, all features discussed in Card Attribute Features (Section 3.2.2) were scaled to standardize their values, ensuring each feature contributes equally to the model. This step is crucial when using models sensitive to the scale of input data, as it prevents features with larger ranges from dominating the model's learning process [Kuhn and Johnson, 2018].

These steps collectively ensured that the dataset was comprehensive and optimized for accurate predictive modelling, aligning with best practices in data preprocessing.

### 6.1.2 Evaluation Metrics

Evaluation metrics are essential for quantifying the quality of forecasting models. In this research, comparisons and evaluations of different models within the same time series are conducted using absolute metrics. Among the scale-dependent measures, Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) are popular choices. While some authors argue against using RMSE for forecast accuracy evaluation in favour of MAE [Hyndman and Koehler, 2006], RMSE's ability to penalize large errors makes it particularly suitable for short-term financial forecasting, where large errors can lead to significant losses.

RMSE = 
$$\sqrt{\frac{1}{T} \sum_{t=1}^{T} (\hat{y}_t - y_t)^2},$$
 (6.1)

where  $\hat{y}_t$  is the forecast value and  $y_t$  is the actual value at time t.

When comparing and evaluating different time series, percentage-based

metrics are more appropriate. Absolute metrics, while effective for a single time series, may not be as suitable when evaluating multiple time series with different magnitudes. To assess models across different time series with varying scales, relative metrics are preferred [Hyndman and Koehler, 2006]. Although relative metrics can be infinite or undefined if  $y_t = 0$  for any t, and they assume a meaningful "zero," these concerns are not applicable in this research. In the context of monetary values, a "zero" is meaningful, and no forecast should yield a value of "zero," as this would be an unusual result in a financial model.

MAPE = 
$$\frac{100}{T} \sum_{t=1}^{T} \left| \frac{\hat{y}_t - y_t}{y_t} \right|,$$
 (6.2)

these measures evaluate how well the combined forecast aligns with the observed data, guiding the selection and adjustment of the combination method.

### 6.1.3 Evaluation Methods

Several methods have been proposed to estimate predictive performance metrics reliably in time series forecasting. The primary approaches for performance estimation are out-of-sample and cross-validation [Hyndman and Athanasopoulos, 2021].

**Out-of-Sample Method:** The out-of-sample method evaluates a model using data not included in the training process, reserving a separate dataset exclusively for testing. This approach involves dividing the dataset into two parts: one for training the model and the other for testing. The test set, often referred to as the "hold-out" set, is used to estimate the model's predictive loss. For time series data, it is essential to ensure that the test set follows the training set in temporal order, preserving the sequence of the observations.

The last twelve observations in this research are reserved as the "holdout" set. Given the weekly frequency of the data, this corresponds to three months of observations, which aligns with the average period between major MtG set releases and tournaments—a decision grounded in domain knowledge. It is important to note that this division is absolute, not relative. For example, if card A has 520 observations while card B has only 127, both will have the same twelve observations reserved as the "hold-out" set.

**Cross-Validation Method:** Cross-validation is widely used for independent and identically distributed data. In k-fold cross-validation, the data is randomly shuffled and split into k folds, where each fold contains  $\frac{n}{k}$  observations, with n being the total number of observations and k representing the total number of folds. Although this method efficiently uses the data, it has limitations for time series analysis because it does not maintain the temporal order of the data [Bergmeir and Benítez, 2012]. To address this, Hyndman [2021] presents an alternative to traditional k-fold cross-validation for time series, where each fold is a "window" of training data followed by a series of sequential test data that "rolls" forward by a fixed number of observations. Each new window can either include only the new data ("fixed origin") or exclude the same number of older observations as well ("rolling origin").

In this research, the "rolling window" approach is adopted due to its consistency in comparing models with the same number of observations in their training data. Each window maintains a 3 : 1 ratio between the train



Figure 6.1: Rolling window cross-validation.

and test sets. For example, for a time series with 520 observations, there will be an out-of-sample "hold-out" set of the last 12 observations of the complete series, and each window will have 381 observations for the train set. The window will roll forward by one observation a total of 127 times during crossvalidation. The forecast range for each window will also be 12 observations, matching the out-of-sample forecast range to increase consistency.

A key distinction between out-of-sample and CV is that the out-of-sample set is never included in the CV windows [Bergmeir and Benítez, 2012, Tashman, 2000].

## 6.2 Results

This section discusses and compares the experimental results. As outlined in Experimental Settings (Section 6.1.1) and throughout Chapters 5 and 4, two experiments were conducted. The first experiment divided the cards into two groups: one with limited supply, representing older MtG settings with low print runs, and the other with an optimal "cost-effect" ratio, making them desirable for tournament play. This was aimed to determine if the best-fitting models differ between cards constrained by supply and those limited by demand—for example, whether models without predictor features better explain Group 1. At the same time, Group 2 relies more on such variables. The second experiment proposed an improved forecasting method based on forecast combination, as detailed in Proposed Model: Forecast Combination Based Neural Network (Section 5.3).

### 6.2.1 Group Comparison

Each base model's performance for Group 1 is shown in 6.2. For any card, its worst performer was always one of ARIMAX, NNETAR, or NNETARX. ARIMA presented the best results in five of the ten cards, followed by RW and NNETARX with two each and NNETAR with one. However, the subset where RW performed best comprises the group's lowest-valued cards. Conversely, the subset of the group's older cards and those with the lowest availability was where RW and NNETAR had the best results. This indicates a trade-off between either the trade volume of a card or its price and the model selection.



Figure 6.2: Heatmap of model accuracy ranking for cards of Group 1.

Results for Group 2 (Figure 6.3) followed a similar pattern. Models incorporating predictor features had lower accuracy performance than models based only on **Y**. As with Group 1, RW had the best overall performance and was the model with the lowest accuracy for most cards, followed by RWD and ARIMA. Once more, NNETARX and ARIMAX had the worst result. As such, for the feature engineering and modelling done in this research, there is no evidence of difference of appropriate models for Group 1 and Group 2.



Figure 6.3: Heatmap of model accuracy ranking for cards of Group 2.

### 6.2.2 Forecast combination based Neural Network

To evaluate the forecasting performance of the proposed models that combine point forecasts, the models outlined in Chapter 5 were employed. Tables 6.1 and 6.2 present the results, which were assessed using the Mean Absolute Percentage Error (MAPE) and Root Mean-Squared Error (RMSE) for the forecasting models applied to cards in Group 1.

To measure overall prediction performance, the average MAPE of the base models was calculated across all ten cards from Group 1. This average was then compared with the average of the best-performing models for the single-model, two-model, and three-model neural network-based combinations (Figure 6.4).



Figure 6.4: Average of MAPEs across all models for Group 1. **Notes:** Values are for each combination's best base model and best model.

The accuracy for single-model and two-model neural network-based combating outperformed, on average, even the best-performing base model RW, 8.23%, 8.92% and 9.88%, respectively. The three-model neural networkbased combination was unable to outperform the best base model but was able to present better results than the second-best base model ARIMA, 10.2% and 10.3%, respectively.

For half of the selected cards, the proposed neural network-based models achieved lower MAPEs and RMSEs than the base models. The most significant improvements were observed in models that had previously underperformed before being incorporated into the single-model neural network. Card **X2ED.233** presented the best improvement, moving from a base model average MAPE of 33.58% to an NN-based single model average of 22.5%.



Figure 6.5: X2ED.233 Forecast results for best models.

However, when comparing the same model across different cards, the overall average performance of the single-model neural network was slightly worse than that of the base models, with averages of 18.95% and 16.34%, respectively. In contrast, the two-model and three-model neural network combinations demonstrated better performance, with average accuracy of 8.92% and 10.16%, respectively, lower than all base models and single-model neural network. These findings suggest that the proposed neural network-

based model, particularly in the two and three-model combinations, effectively leverages the strengths of different models. This supports the concept of model combination, as discussed by [Wang et al., 2023], and highlights the potential of meta-models like the neural network-based combination to enhance forecasting accuracy [Hu, 2021].

X2ED.48 X2ED.84 X2ED.233 X2ED.275 LEG.113 MIR.241 MIR.307 UDS.1 UDS.135EXO.72 Base model ARIMA 386.7299.624760.16 276.771.2636.590.86 5.7810.9 ARIMAX 541.51 638.02 322.83 84.97 12635.454.95.0224.7312.952.07883.35 790.98 5631.23 491.9 101.28 25.88 16.24 14.24 NNETAR 8.82 8.44 NNETARX 741.27 1192.23 2565.05 172.83 123.09 3.231.562.1714.44 26.75RW 351.0829.714436306.5868.291.334.83 0.966.08 11.16RWD 392.51 106.24 4775.37329.78 90.25 1.6443.951.8 7.81 13.83NN1 ARIMA 614 01 2095 35 3956.82 207 13 52.672.95 37.2 2 8.74 17 29 ARIMAX 6940.95 293.48126.444.81 379.362224.44 3.7812.3813.581183.39 1739.21 124.29 51.36 NNETAR 452.32783.42 5.616 16.490.95NNETARX 12.87 127.42 821.83 1580.534142.09 234.62 108.22.24 36.740.27 RW 549.3 1097.18 3175.05 231.5345.88 2.93 32.14 8.84 17.34 2.19RWD 535.72691.763456.35233.2261.843.0537.572.859.83 19.13NN2 409.01 151.34 2974.39 208.79 60.9 1.6531.171.976.01 12.73NN3 207.66 61.86 1.94 467.47 565.99 3047.3 1.98 28.57.56 10.65

Table 6.1: RMSEs for point forecasting results in Group 1

 Table 6.2: MAPEs for point forecasting results in Group 1

	A2LD.40	A2DD.04	M2DD.200	M2DD.210	EEG.110	101110.241	101110.001	1.10.12	000.1	000.100	model morage
Base model											
ARIMA	5.8	5.82	27.79	20.34	7.88	11.69	7.5	1.93	6.14	7.97	10.29
ARIMAX	8.71	1.52	72.8	45.26	40.72	19.16	11.21	11.33	29.84	9.41	25
NNETAR	15.62	15.67	32.9	35.76	12.53	84.11	5.36	38.57	10.57	11.02	26.21
NNETARX	12.67	22.96	14.27	12.01	15.1	26.16	6.27	4.67	16.45	18.42	14.9
RW	5.38	0.51	25.87	22.33	8.48	12.09	7.16	2.2	6.54	8.19	9.88
RWD	5.91	1.89	27.84	23.88	11.17	15.25	8.96	4.16	8.62	10.09	11.78
NN1											
ARIMA	9.9	39.85	22.93	15.23	6.72	29.84	7.11	4.81	10.38	13.23	16
ARIMAX	5.87	39.75	40.63	20.77	17.31	36.19		11.9	15.22	10.06	21.97
NNETAR	7.35	21.81	9.2	56.84	15.14	7.22	10.83	13.68	6.56	12.62	16.13
NNETARX	13.89	29.86	23.97	17.67	14.74	20.07	7.45	100.01	15.19	99.97	34.28
RW	8.68	18.67	18.29	15.73	5.87	29.64	5.48	5.35	10.55	13.4	13.17
RWD	8.55	11.97	19.96	16.36	8.11	30.83	6.52	7.01	12.02	14.84	13.62
NN2	6.7	2.39	17.12	14.82	7.22	15.03	5.44	4.73	6.59	9.16	8.92
NN3	7.4	9.64	17.52	14.89	7.31	19.54	4.99	4.64	8.66	6.99	10.16

For Group 2, the proposed model's average accuracy did not surpass the best base model for any of the selected cards. However, it outperformed the ARIMAX, NNETAR, and NNETARX models. The MAPEs for ARIMA models (10.8%, 12.1%, and 14.2%) were better than those for NN1 (17.8%), NN2 (14.8%), and NN3 (16.2%). In this group, NN2 and NN3 both outper-



formed NN1, which had the best accuracy in Group 1.

Figure 6.6: Average of MAPEs across all models for Group 2. **Notes:** Values are for each combination's best base model and best model.

When analyzing individual cards, the performance of NN-based single models was slightly worse, with improvements seen in only a quarter of cases compared to their respective base models. However, NN2 and NN3 continued to perform well, surpassing the average base model results in 9 out of 10 cards for NN2 and 8 out of 10 for NN3. The accuracy of the NN-based two-model approach in Group 2 was better than in Group 1, with three cards achieving the lowest MAPE with this model. Notably, Card **LRW.145**, which had the best accuracy among the base models, showed the most significant improvement in the proposed models, with its MAPE decreasing from an average of 9.31% to just 1.06% using NN2.

Overall, the average performance of NN2 and NN3 was substantially better than the base models, with respective averages of 14.76%, 16.19%, and 44.62%. In contrast, NN1 had a MAPE of 625.73%, significantly worse



Figure 6.7: LRW.145 Forecast results for best models.

than the base models. Even after excluding the ARIMAX result for **IKO.67**, which was a notable outlier, NN1 could not outperform the average of the base models. These results suggest that the proposed neural network-based models provide a meaningful improvement in forecasting accuracy over the base models.



Figure 6.8: IKO.67 Forecast results for best models.

	111 0.220	GF 1.52	LUM 1149	1113.100	D1K.150	<b>KLD.110</b>	KLD.234	110.07	11.0.00	V O W .03
Base model										
ARIMA	0.64	0.12	0.41	0.63	0.37	0.33	1.15	1.81	0.15	0.28
ARIMAX	1.17	3.34	0.53	0.51	0.65	1.42	1.09	365.53	1.1	2.48
NNETAR	0.96	0.49	3.64	0.91	1.12	5.47	0.67	3.88	0.94	0.95
NNETARX	4.33	18.91	8.5	1.36	1.33	5.82	1.19	2.8	0.35	0.64
RW	0.62	0.12	0.42	0.7	0.32	0.75	1.37	1.95	0.14	0.85
RWD	0.68	0.18	0.38	0.71	0.35	0.21	1.37	2.01	0.2	0.38
NN1										
ARIMA	3.78	0.12	0.42	3.59	1.07	1.14	1.42	1.79	0.35	1.24
ARIMAX	6.95	0.18	0.36	3.43	2.09	1.25	1.47	1.78	0.41	0.73
NNETAR	4.43	0.13	0.4	1.65	2.48	1.03	1.53	1.79	0.41	1.47
NNETARX	4.29	1.56	0.47	39.06	3.97	1.67	3.39	3701.93	7.44	1.7
RW	1.09	0.49	1.71	0.84	2.97	1.39	1.75	1.73	0.44	1.66
RWD	1.23	8.48	9.83	1.58	3.21	2.38	2.34	2.52	0.37	0.49
NN2	1.34	0.13	0.35	0.66	0.38	0.76	1.39	1.72	0.36	0.46
NN3	1.24	0.12	0.36	0.86	0.31	0.89	1.41	1.71	0.37	0.63

 Table 6.3: RMSEs for point forecasting results in Group 2

 INV.226
 GPT.52
 LBW.145
 THS.180
 DTK.150
 KLD.110
 KLD.234
 IKO.67
 IKO.88
 VOW.63

Table 6.4: MAPEs for point for ecasting results in Group 2

	INV.226	GPT.52	LRW.145	THS.180	DTK.150	KLD.110	KLD.234	IKO.67	IKO.88	VOW.63	Model average
Base model											
ARIMA	10.21	0.89	1.62	13.52	9.76	8.2	29.79	15.77	11.11	6.75	10.762
ARIMAX	17.5	26.31	1.59	11.26	17.18	33.03	28.48	1032.4	85.25	54.55	130.755
NNETAR	15.44	4.1	13.68	18.75	34.27	147.26	13.92	35.48	64.58	26.06	37.354
NNETARX	74.69	183.42	35.9	30.38	40.93	164.79	27.48	27.35	25.08	15.41	62.543
RW	9.77	0.89	1.69	15.37	8.39	19.53	35.9	16.95	10.77	22.71	14.197
RWD	10.72	1.38	1.36	15.43	9.26	5.15	35.93	17.51	14.92	9.26	12.092
NN-based single model											
ARIMA	57.75	0.89	1.7	68.21	30.04	31.24	30.14	15.67	28.02	34.04	29.77
ARIMAX	107.27	1.42	1.16	64.49	55.87	34.59	32.01	15.9	33.01	18.89	36.461
NNETAR	67.66	1	1.62	36.31	55.02	28.53	38.1	16.34	33.31	37.69	31.558
NNETARX	73.45	16.41	1.96	710.41	84.2	47.17	68.41	34262.51	479.25	46.75	3579.052
RW	16.73	5	7.6	16.9	83.83	38.69	46.31	14.57	36.02	45.68	31.133
RWD	19.38	79.02	42.63	36.16	93.38	67.42	64.05	22.33	29.18	10.54	46.409
NN-based two-model	21.8	0.96	1.06	10.13	9.99	19.66	29.66	15.19	28.74	10.36	14.755
NN-based three-model	19.54	0.89	1.38	18.26	8.52	23.86	30.25	15.51	29.18	14.54	16.193

## Chapter 7

# Ts.shiny: Visualization Using Interactive Graphics

This chapter presents the various aspects of the implementation of ts.shiny, an **R**-based Shiny framework developed to facilitate exploratory data analysis (EDA) and basic forecasting. The following sections detail the motivation behind the creation of **ts.shiny** and provide a comprehensive explanation of its core functionalities.

Following this, the architecture of the system is discussed, explaining the technical components that make **ts.shiny** a flexible and scalable platform. The architecture discussion is divided into two main areas: the front-end design, which emphasizes user interaction and data visualization, and the back-end processes, which focus on data manipulation, model fitting, and the generation of forecasts. Together, these components enable **ts.shiny** to deliver a robust and user-friendly experience for analyzing and forecasting time series data.

## 7.1 Ts.shiny

**ts.shiny** is an R-based Shiny framework application developed to support time series analysis and initial forecasting. It provides users with a readyto-use tool designed to handle user-submitted datasets, offering basic exploratory data analysis (EDA) and forecasting capabilities. The goal of **ts.shiny** is to make the process of time series analysis accessible to a broader audience, removing the dependency on extensive programming skills, while still providing a flexible, data-driven platform for both domain experts and data scientists.

The application was created with three key motivations in mind: data democracy, data agnosticism, and a focus on overcoming the limitations of similar tools.

### 7.1.1 Data Democracy

One of the core motivations behind the development of **ts.shiny** is to democratize time series analysis. The tool allows domain experts—those without coding or programming expertise—to contribute their insights into the data. This approach empowers users to visually explore and interact with the dataset without needing to write or understand complex code.

By providing a graphical user interface (GUI) that facilitates interaction, ts.shiny bridges the gap between data scientists and other team members. It allows domain experts to explore the data visually, detect patterns, and engage with different forecasting models. At the same time, it enables data scientists to visually demonstrate the effects of varying model parameters and assumptions in real time. This collaborative approach enhances the overall decision-making process and ensures that all team members, regardless of their technical background, can contribute meaningfully to the analysis.

### 7.1.2 Data Agnosticism

**Ts.shiny** was designed to be data agnostic, capable of handling a wide variety of time series datasets from different domains. While it is highly flexible and can accommodate many types of data, some preprocessing may be required to ensure that the datasets conform to certain standards (e.g., time-indexed data). This flexibility sets **ts.shiny** apart from many commercial tools, which often impose strict requirements on the structure and format of the data.

The agnostic nature of the tool allows it to be used in the early stages of a project, where the dataset may still be evolving, as well as in the final stages, where it can serve as a showcase for complete project results. It supports a wide range of exploratory tasks, including initial data analysis, trend identification, and even model testing, making it a valuable tool for both prototyping and presenting results.

### 7.1.3 Similar Tools

Compared to other data visualization and analysis tools, **ts.shiny** offers a distinct advantage by being both flexible and feature-rich. Many popular visualization tools are limited in their analytical capabilities and require highly structured datasets to function properly. **ts.shiny** overcomes these limitations by providing a flexible interface that can adapt to different data structures and analysis needs.

On the other hand, traditional code-based tools leave the source code open to modifications, which could introduce errors or lead to unintended consequences. **ts.shiny** strikes a balance by allowing users to interact with the tool in a visual, user-friendly manner, while keeping the underlying code protected and stable. This ensures that non-technical users can safely explore the data without risking the integrity of the analysis pipeline.

Overall, **ts.shiny** serves as a proof-of-concept for how the R Shiny framework can be applied to handle complex and lengthy tasks, offering a powerful solution for interactive time series analysis. It is designed to be both accessible and robust, making it suitable for a variety of users and applications.

## 7.2 Architecture

The architecture of **ts.shiny** is designed to ensure modularity and flexibility, allowing users to perform exploratory data analysis (EDA) and forecasting on time series datasets through a simple and intuitive interface. The system is organized into three main modules: **Data Module**, **EDA Module**, and **Forecasting Module**. Each module plays a critical role in the overall functionality of the application, and they work together to deliver a seamless experience for users. The diagram below illustrates the overall architecture of the system.



Figure 7.1: ts.shiny System Architecture

### 7.2.1 Data Module

The **Data Module** is responsible for handling the user's data throughout the lifecycle of the application. It manages three key stages: loading, transforming, and storing data.

- Loaded Data: The application accepts user-submitted datasets, which are uploaded through the Data Module. These datasets must contain time-indexed data suitable for time series analysis. Upon upload, the system validates the structure of the data, ensuring it conforms to the minimum requirements for processing.
- **Transformed Data**: After loading the data, users can apply selected transformations at any point during the application lifecycle. Available transformations include adding Moving Average series, transforming the index frequency, or excluding certain features from the dataset.

Each time a transformation is applied, the transformed data is stored within the **Transformed Data** submodule. This ensures that the dataset is prepared in a consistent and reliable format for subsequent analysis and forecasting phases without destroying the original data.

• Stored Data: The Data Module also stores results from analyses and forecasting models, ensuring that users can revisit their work without needing to start over. This feature supports the tool's flexibility, allowing users to continue their analysis or refine at any stage. Data transformations, model outputs, and forecasting results can be accessed or updated as needed, ensuring a seamless workflow.

The **Data Module** feeds the prepared and transformed data into the **Current Data** container, which serves as the central data structure for the application. This container is accessed by subsequent modules for visualization and analysis, ensuring smooth and continuous interaction between different application components.

### 7.2.2 EDA Module

The **EDA Module** facilitates exploratory data analysis by providing various interactive visualizations. These visualizations help users understand the underlying patterns, trends, and distributions within their time series data. The module includes several submodules, each responsible for a different type of plot or visualization. Subdividing the modules allows for easy inclusion of new visualizations, removal of existing ones, and troubleshooting any problems in the current application without having to go through hundreds of lines of code. Each visualization can also have selectors to allow the user to better customize the output (Figure 7.2).

	g Data Processing	Data Analysis				
ataset Summ	nary					
Metric	Y		card.age	copies.complete		Select Feature:
ount	520		520	520		Y
A	0		0	Ō		Y card ane
tin	3.631429		5465.00	0		copies.complete
a	6.880714		6371.25	0.00		
dean	15.97368		7279.501	0.01730769		
edian	10.515		7279.50	0.00		
2	21.54464		8187.75	0.00		
tax	65.65857		9094.50	4		
lange	62.02714		3629.50	4		
QR	14.66393		1816.50	0.00		
utliers	24		0	5		
eature Visual	ization					
eature Visual lect Plot Type: .ind .olumn listogram .catter .e	ization					M
eature Visual elect Plot Type: Line] Column Histogram Kater 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	ization	28		201		
eature Visual leet Piot Type:		20	9	219	next 200	M
eature Visual siect Piot Type: Line Column Histogram Socier 20 21 22 23 24 eature Trend	Analysis	29	8	2019	NOR ZD	M

Figure 7.2: Exploratory Data Analysis interactive elements.

These visualizations are dynamically generated and displayed in a Visualization Container, which serves as the primary interface for users to interact with their data. Users can toggle between different visualizations and adjust parameters to tailor the analysis to their needs.

### 7.2.3 Forecasting Module

The **Forecasting Module** provides users with access to various forecasting techniques, allowing them to generate predictive models based on their time

series data. The module supports a range of popular forecasting methods and the proposed model developed in this research.

Users can select from these models and apply them directly to the **Current Data**. The forecasts are then displayed in the **Visualization Container**, allowing for easy comparison between the actual values and the predicted values. Additionally, users can adjust model parameters, such as the number of lags or the length of the forecasting horizon, to customize the forecasts according to their specific use cases. As with the EDA module, the forecasting module also allows user interaction for better model parameter tuning (Figure 7.3).



Figure 7.3: Forecasting Analysis interactive elements.

After each forecasting operation, the data is saved using the **Stored Data** submodule, ensuring ease of access for future sessions. The system currently supports storing forecasts from multiple models at once, though it limits each model to a single forecast per session. This design enhances flexibility while maintaining the organization and accessibility of the forecasting results.

### 7.2.4 Integration and Visualization

At the heart of **ts.shiny** is the **Visualization Container**, which integrates the outputs from both the **EDA Module** and the **Forecasting Module**. This container module allows users to customize their current GUI by displaying only the desired plots, while moving any unwanted visualizations to a hidden sidebar. The **Visualization Container** is a generic container capable of holding all available visualizations, and users can select and add multiple instances of the same or different visualizations to their application (Figure 7.4).



Figure 7.4: **ts.shiny** application, showcasing the modularity of the visualizations.

```
f_switch <- function(viz, df) {</pre>
2
    switch(
3
      viz,
4
       "_time-series_" = ts_viz(df),
5
       "_pacf_" = pacf_viz(df),
6
       "_acf_" = acf_viz(df),
7
       "_time-series-ma_" = ts_smooth(df)
8
    )
9
11 }
```

Listing 7.1: Selector for visualization container

Through these containers, users can interact with their data in real-time, switching between visualizations, adjusting model parameters, and instantly viewing the results of their changes. The interactive nature of the tool empowers users to explore the effects of different forecasting methods and assumptions without the need for complex coding, enhancing the flexibility and usability of the application.

```
1 dash_analytics_viz <- function (class_specific, header,</pre>
      unique_id, viz, df) {
    # Set default column class
2
    column_class <- "col-md-6 col-lg-6 col-sm-12"</pre>
3
4
    # Update column class if "time-series" is part of the viz
5
     ID
    if (grepl("time-series", viz)) {
6
       column_class <- "col-md-12 col-lg-12 col-sm-12"</pre>
7
    }
8
9
    div(
10
      class = class_specific,
11
      div(
12
         class = column_class,
13
14
         div(
           class = "panel panel-default",
15
           div(
16
             class = "panel-heading clearfix",
17
             tags$h2(header, class = "pull-left panel-title"),
18
             div(
19
                class = "pull-right",
20
               shiny::actionButton(
21
                  inputId = unique_id,
22
                  label = '',
23
                  class = "btn-danger delete",
24
25
                  icon = shiny::icon("minus")
               )
26
             )
27
           ),
28
           div(
29
             class = "panel-body",
30
             f_switch(viz, df)
31
           )
32
         )
33
      )
34
    )
35
36 }
```

Listing 7.2: Visualization container

# Chapter 8

# **Conclusion and Discussion**

This chapter provides a comprehensive overview of the research conducted in this study, highlighting the key findings, implications, limitations, and potential directions for future work. The analysis focused on forecasting the prices of Magic: The Gathering cards, utilizing a combination of traditional statistical models and advanced machine learning techniques.

## 8.1 Summary of Findings

The primary objectives of this research were twofold: first, to create a consolidated, consistent database that meets the standards of analytical frameworks and is scalable enough to accommodate new cards and additional exogenous features; second, to propose a data-driven forecast combination method to improve forecast accuracy. Additionally, this research aimed to compare base forecasting methods to determine if their performance varies across different groups of cards and to develop a Shiny-based application to assist in data analysis.

### 8.1.1 Database Creation

The first objective of this research was to create a consolidated and consistent database to facilitate the analysis and forecasting of Magic: The Gathering card prices. Given the fragmented nature of available data, this task involved meticulous data collection from various sources, including MTG-Goldfish, MTGTop8, and MTGJSON. These sources provided diverse yet complementary datasets, encompassing price histories, tournament results, and card attributes.

The preprocessing stage included rigorous data cleaning and feature engineering to ensure the dataset's reliability and scalability. Missing price values were addressed through interpolation, while various features, such as card appearances, card count, and legality, were engineered to capture the nuanced dynamics of the MtG market. By integrating these datasets, a robust and scalable database was constructed, capable of accommodating new cards and additional exogenous features as the game evolves. This database served as the foundation for the subsequent forecasting models, providing a rich and comprehensive dataset that enhanced the accuracy and relevance of the forecasts.
#### 8.1.2 Proposed Model

The second objective was to develop a data-driven forecast combination model designed to improve the accuracy of price predictions for MtG cards. Traditional forecasting methods, such as ARIMA, have limitations when applied to the complex, non-linear nature of MtG card prices. To address this, a forecast combination based Neural Network model was proposed. This model leverages the strengths of individual base models, including ARIMA, Random Walk, and NNETAR, by combining their forecasts into a single, more accurate prediction.

The model operates by first establishing base forecasts using the individual models. These forecasts are then combined using a weighted approach, which determines the weights through optimization techniques that minimize forecast errors using a genetic algorithm. The combined series, together with the actual values, are then used as inputs in a new neural network architecture. The resulting model is data-driven, allowing it to adapt to different market conditions and external influences.

This approach is not only applicable to MtG forecasting but can also be generalized to other domains. The proposed architecture is generic and not dependent on MtG data, making it a versatile tool for improving forecast accuracy in various fields.

The results indicated that the forecast combination model generally outperformed the individual base models, particularly for cards in Group 1, which are more susceptible to external factors such as market speculation and collector interest. For Group 2, the model's performance was also superior, though the difference was less pronounced due to the less stable and predictable nature of these cards' market values.

#### 8.1.3 Model Comparison

The third objective involved comparing the performance of traditional forecasting methods across different groups of MtG cards. The research categorized cards into two main groups: those on the Reserved List (Group 1) and those frequently used in tournaments (Group 2). The comparison aimed to determine whether the performance of these models varies significantly between these groups, given their distinct characteristics in terms of supply and demand dynamics.

This comparative analysis failed to highlight any difference in model preference between the two groups for the models selected and engineered features.

#### 8.1.4 ts.shiny

The final component of this research involved the development of an R Shiny application named **ts.shiny**. Designed to facilitate out-of-the-box exploratory data analysis (EDA) and basic forecasting for time series data, this application was created with the intention of providing an intuitive, userfriendly tool that enables users to explore and analyze time series datasets interactively.

ts.shiny integrates several key functionalities:

• Interactive Data Visualization: Users can upload their datasets

and instantly visualize the time series through various plots, including line plots, histograms, and box plots. This feature allows users to quickly identify trends, seasonal patterns, and potential outliers in the data.

- Summary Statistics and Descriptive Analysis: The application provides summary statistics for the time series, such as mean, median, variance, and autocorrelation. These statistics are crucial for understanding the underlying distribution and characteristics of the data before moving on to more complex analyses.
- **Basic Forecasting**: The application includes built-in models for basic time series forecasting, such as ARIMA, Neural Network-based models, as well as the proposed model of this research. Users can easily apply these models to their data and visualize the resulting forecasts. This feature is particularly useful for quickly generating baseline forecasts and comparing different modelling approaches.
- User Customization and Flexibility: While ts.shiny offers a range of default settings for ease of use, it also allows for user customization. Users can adjust parameters for their analyses, select different models, and choose specific time ranges for forecasting. This flexibility ensures that the tool can be adapted to various datasets and analytical needs.

The **ts.shiny** application represents a significant contribution to the project by providing a practical, accessible platform for both novice and experienced users to perform EDA and basic forecasting. By streamlining these processes into an interactive tool, **ts.shiny** enhances the accessibility of time series analysis, making it easier for users to derive insights and make

informed decisions based on their data.

### 8.2 Future Research Directions

The results of this research open several avenues for future work, focusing on expanding the applicability and robustness of the proposed models and tools.

#### 8.2.1 Improve Model Application

While the current system was used for Magic: The Gathering (MtG) card price forecasting, its framework could be extended to other collectible games or markets with similarly robust secondary markets. Future research could explore how the model can be adapted and fine-tuned to different types of collectible assets, taking into account unique market dynamics and demand patterns inherent to each.

#### 8.2.2 Different Modeling Engineering

Modelling a card's cost-effectiveness information is a challenging task that requires substantial research. The methods presented in this research can be potentially extended to improve exogenous feature modelling. Leveraging Natural Language Processing (NLP) or other advanced approaches could enhance the accuracy of feature extraction and the overall predictive power of the model. This would be particularly useful for capturing the complex interactions between a card's textual attributes and market value.

#### 8.2.3 Increased Base Model Selection

The proposed method was based on popular forecasting methods, such as ARIMA, Random Walk, and NNETAR. Future research could explore the inclusion of additional base models to further enhance forecasting accuracy, particularly for cards with fewer observations or those in more volatile market segments. By diversifying the base models, the forecast combination approach could be optimized to handle a broader range of scenarios and data characteristics, potentially reducing performance loss in less stable datasets.

#### 8.2.4 Continuous Model Improvement

As forecasting techniques evolve, ongoing opportunities exist to refine and enhance the proposed model. In addition to incorporating new base models, future research could focus on developing approaches for interval forecasting and volatility modelling, which are crucial for dealing with uncertainty and market fluctuations. Integrating these techniques would improve the model's robustness, particularly in scenarios where accurate prediction ranges and risk assessments are critical.

#### 8.2.5 ts.shiny Improvement

The ts.shiny application provides a foundation for out-of-the-box exploratory data analysis (EDA) and basic forecasting. However, there is significant potential for expanding its capabilities. Future enhancements could include integrating more advanced forecasting models, a more robust framework for handling large datasets, and additional customization options for users to tailor their analyses more precisely. These improvements would position ts.shiny as a final product capable of filling a niche currently underserved by existing tools and software, making it a valuable resource for novice and expert users in time series analysis.

## Bibliography

- Chris Alvin, Michael Bowling, Shaelyn Rivers-Green, Deric Siglin, and Lori Alvin. Toward a competitive agent framework for magic: The gathering. *The International FLAIRS Conference Proceedings*, 34, Apr. 2021. doi: 10.32473/flairs.v34i1.128416.
- Orley Ashenfelter and Kathryn Graddy. Auctions and the price of art. Journal of Economic Literature, 41(3):763–787, 2003. doi: 10.1257/ 002205103322436188.
- J. M. Bates and C. W. J. Granger. The combination of forecasts. OR, 20 (4):451, dec 1969. doi: 10.2307/3008764.
- William J. Baumol. Unnatural value: Or art investment as floating crap game. Journal of Arts Management and Law, 15(3):47–60, 1985. doi: 10.1080/07335113.1985.9942162.
- Christoph Bergmeir and José M. Benítez. On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191:192–213, 2012. doi: https://doi.org/10.1016/j.ins.2011.12.028.
- Sverre Johann Bjørke and Knut Aron Fludal. Deckbuilding in magic: The gathering using a genetic algorithm. Master's thesis, Norwegian University

of Science and Technology, Trondheim, Norway, 2017. URL http://hdl. handle.net/11250/2462429.

- Tim Bollerslev. Generalized autoregressive conditional heteroskedasticity. Journal of Econometrics, 31(3):307–327, 1986. doi: 10.1016/0304-4076(86) 90063-1.
- G. E. P. Box and G. M. Jenkins. Some recent advances in forecasting and control. *Applied Statistics*, 17(2):91, 1968. doi: 10.2307/2985674.
- G. E. P. Box and G. C. Tiao. Intervention analysis with applications to economic and environmental problems. *Journal of the American Statistical Association*, 70(349):70–79, 1975. doi: 10.1080/01621459.1975.10480264.
- R. G. Brown. Statistical Forecasting for Inventory Control. McGraw-Hill, 1959.
- Benjamin J. Burton and Joyce P. Jacobsen. Measuring returns on investments in collectibles. *Journal of Economic Perspectives*, 13(4):193–212, 1999. doi: 10.1257/jep.13.4.193.
- Olivier Chanel, Louis-André Gérard-Varet, and Victor Ginsburgh. The relevance of hedonic price indices: The case of paintings. Journal of Cultural Economics, 20(1):1–24, 1996. doi: 10.1007/s10824-005-1024-3.
- Jacob Chodoriwsky. Synergy in magic: The gathering: An application of graph theory, 2006.
- Alex Churchill, Stella Biderman, and Austin Herrick. Magic: The gathering is turing complete, 2019. URL https://arxiv.org/abs/1904.09828.

- Alfred Cowles. Can stock market forecasters forecast? Econometrica: Journal of the Econometric Society, pages 309–324, 1933. doi: 10.2307/ 1907042.
- Magnus de Laval. Deconstructing print runs, 2020. URL https://oldschool-mtg.blogspot.com/2020/12/deconstructing-print-runs. html. Accessed: 2024-08-18.
- David DeLaney. Magic the gathering print runs. URL https://www. mtginformation.com/print-runs. Accessed: 2024-08-18.
- Matteo Di Napoli. Multi-asset trading with reinforcement learning: an application to magic the gathering online. Master's thesis, Politecnico di Milano, Milan, Italy, 2017. URL https://www.politesi.polimi.it/handle/10589/140266.
- Robert F. Engle. Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica: Journal* of the Econometric Society, pages 987–1007, 1982. doi: 10.2307/1912773.
- Alex Evans. Mtg lord of the rings is magic's second best selling set ever, 2023. URL https://www.wargamer.com/magic-the-gathering/ mtg-lord-of-the-rings-second-best-selling-set. Accessed: 2024-08-18.
- Eugene F. Fama. Efficient capital markets: A review of theory and empirical work. The Journal of Finance, 25(2):383, 1970. doi: 10.2307/2325486.
- Thomas Fischer and Christopher Krauss. Deep learning with long shortterm memory networks for financial market predictions. *European Journal* of Operational Research, 270(2):654–669, 2018. doi: 10.1016/j.ejor.2017. 11.054.

- William N. Goetzmann. Accounting for taste: Art and the financial markets over three centuries. *The American Economic Review*, 83(5):1370–1376, 1993. ISSN 00028282.
- Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. J. Mach. Learn. Res., 3:1157-1182, 2003. URL http://jmlr. org/papers/v3/guyon03a.html.
- Hasbro. Magic: The gathering overview, 2023. URL https://investor. hasbro.com/magic-gathering#:~:text=Created%20in%201993%2C% 20MAGIC%3A%20THE,digital%20players%20with%20MAGIC%20ARENA. Accessed: 2024-08-18.
- Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, volume 2. Springer, 2009.
- Charles C. Holt. Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, 20(1):5–10, 2004. doi: https://doi.org/10.1016/j.ijforecast.2003.09.015.
- Yi-Chung Hu. Forecasting the demand for tourism using combinations of forecasts by neural network-based interval grey prediction models. Asia Pacific Journal of Tourism Research, 26(12):1350–1363, 2021. doi: 10. 1080/10941665.2021.1983623.
- Rob J. Hyndman and George Athanasopoulos. Forecasting: Principles and Practice. OTexts, Melbourne, Australia, 3rd edition, 2021. URL https: //otexts.com/fpp3. Accessed on 12 August 2024.
- Rob J. Hyndman and Anne B. Koehler. Another look at measures of forecast

accuracy. International Journal of Forecasting, 22(4):679–688, 2006. doi: https://doi.org/10.1016/j.ijforecast.2006.03.001.

- M. G. Kendall and A. Bradford Hill. The analysis of economic time-seriespart i: Prices. Journal of the Royal Statistical Society. Series A (General), 116(1):11, 1953. doi: 10.2307/2980947.
- Jens Kleine, Thomas Peschke, and Niklas Wagner. Rich men's hobby or question of personality: Who considers collectibles as alternative investment? *Finance Research Letters*, 35:101307, 2020. doi: 10.1016/j.frl.2019.101307.
- Max Kuhn and Kjell Johnson. *Applied Predictive Modeling*. Springer, 2018. ISBN 1461468485,9781461468486,9781461468493.
- Michael H. Kutner, Christopher J. Nachtsheim, and John Neter. *Applied Linear Regression Models*. Ingram, 2004. ISBN 0073013447,9780073013442.
- George Langelett and Zhiguang Wang. Sealed collectible card game product as standalone investment and portfolio diversifier. *Global Journal of Accounting & Finance (GJAF)*, 7(1), 2023.
- Roderick J. A. Little and Donald B. Rubin. Statistical Analysis with Missing Data. Wiley Series in Probability and Statistics. Wiley, 2019. ISBN 0470526793,9781118596012,9781118595695,9780470526798.
- Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biology, 52:99–115, 1990. doi: 10.1016/S0092-8240(05)80006-0.
- Thomas H. Mcinish and Rajendra K. Srivastava. The determinants of investment in collectibles: A probit analysis. *Journal of Behavioral Economics*, 11(2):123–134, 1982. doi: https://doi.org/10.1016/0090-5720(82)90019-5.

- MTGGoldfish. Mtggoldfish, n.d. URL https://www.mtggoldfish.com/. Accessed: 2024-08-04.
- MTGJSON. Mtgjson, n.d. URL https://mtgjson.com/. Accessed: 2024-08-04.
- MTGTop8. Mtgtop8, n.d. URL https://www.mtgtop8.com/. Accessed: 2024-08-04.
- Charles R. Nelson and Charles R. Plosser. Trends and random walks in macroeconomic time series. *Journal of Monetary Economics*, 10(2): 139–162, 1982. doi: 10.1016/0304-3932(82)90012-5.
- Wizards of the Coast. Magic: The gathering, n.d. URL https://magic. wizards.com/en. Accessed: 2024-08-04.
- Thomas J. O'Brien, Lawrence J. Gramling, and Mauricio Rodriguez. An introduction to the collectible sportscard market. *Managerial Finance*, 21 (6):47–63, 1995. doi: 10.1108/eb018524.
- Nitin R. Patel, Peter C. Bruce, and Galit Shmueli. Data Mining for Business Intelligence: Concepts, Techniques, and Applications in Microsoft Office Excel (r) with XLMiner (r). John Wiley & Sons, 2nd ed edition, 2011. ISBN 9781118211397,1118211391.
- Lewis Fry Richardson. Weather Prediction by Numerical Process. Cambridge Mathematical Library. Cambridge University Press, 2 edition, 2007. doi: 10.1017/CBO9780511618291.
- Mark Rosewater. Rare, well done, 2002. URL https://magic.wizards. com/en/news/making-magic/rare-well-done-2002-02-25-0. Accessed: 2024-08-18.

- David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986. doi: 10.1038/323533a0.
- Hiroki Sakaji, Akio Kobayashi, Masaki Kohana, Yasunao Takano, and Kiyoshi Izumi. Card price prediction of trading cards using machine learning methods. In International Conference on Network-Based Information Systems, 2019. URL https://api.semanticscholar.org/CorpusID: 201109427.
- Gregory Schmidt. Magic: The gathering becomes a billion-dollar brand for toymaker hasbro, 2023. URL https://www.nytimes.com/2023/02/16/ business/magic-the-gathering-hasbro.html. Accessed: 2024-08-18.
- Leonard J. Tashman. Out-of-sample tests of forecasting accuracy: An analysis and review. International Journal of Forecasting, 16(4):437–450, 2000. doi: https://doi.org/10.1016/S0169-2070(00)00065-0.
- TCGplayer. ebay acquires tcgplayer, 2022. URL https://seller. tcgplayer.com/about/press-center/ebay-acquires-tcgplayer/. Accessed: 2024-08-18.
- Robert Tieber and Alexander Felfernig. A knowledge-based configurator for building magic: The gathering card decks. *CEUR Workshop Proceedings*, 2945:55–57, sep 2021. ISSN 1613-0073.
- Vladimir N. Vapnik. The Nature of Statistical Learning Theory. Springer New York, 1995. ISBN 9781475724400. doi: 10.1007/978-1-4757-2440-0.
- Cécile Viboud, Pierre-Yves Boëlle, Fabrice Carrat, Alain-Jacques Valleron, and Antoine Flahault. Prediction of the spread of influenza epidemics by

the method of analogues. *American Journal of Epidemiology*, 158(10): 996–1006, 11 2003. doi: 10.1093/aje/kwg239.

- Xiaoqian Wang, Rob J. Hyndman, Feng Li, and Yanfei Kang. Forecast combinations: An over 50-year review. *International Journal of Forecasting*, 39(4):1518–1547, 2023. doi: 10.1016/j.ijforecast.2022.11.005.
- Colin D. Ward and Peter I. Cowling. Monte carlo search applied to card selection in magic: The gathering. In 2009 IEEE Symposium on Computational Intelligence and Games, pages 9–16. IEEE, 2009.
- Henry N. Ward, Bobby Mills, Daniel J. Brooks, Dan Troha, and Arseny S. Khakhalin. Ai solutions for drafting in magic: The gathering. In 2021 IEEE Conference on Games (CoG). IEEE, aug 2021. doi: 10.48550/arXiv. 2009.00655.
- Peter R. Winters. Forecasting sales by exponentially weighted moving averages. *Management Science*, 6(3):324–342, 1960. doi: 10.1287/mnsc.6.3.324.
- John Wyburn and Paul Alun Roach. An hedonic analysis of american collectable comic-book prices. Journal of Cultural Economics, 36(4):309–326, 2012. doi: 10.1007/s10824-012-9166-6.

# Appendix A

## Card Prices Line Plots

## Group 1



### Group 2

