Path of Exile: A Time Series Analysis and Prediction

Dylan Kline
dkline4@u.rochester.edu
Computer Science Department,
University of Rochester Hajim School of Engineering

Abstract

As virtual economies grow in prevalence, understanding their unique market dynamics becomes crucial for effective forecasting. This research focuses on the game Path of Exile, specifically aiming to predict the prices of 'Exalted Orbs' up to five days ahead using its league-based economy as a model. Employing a dataset with over 2 million data points spanning 2016 to 2024, the study explores various machine learning techniques, including neural networks and regressor chains. The models are assessed using mean squared error (MSE) and mean absolute percentage error (MAPE). Initial results indicate that while neural networks provide a robust baseline, regressor chains—particularly those utilizing XGBoost—demonstrate potential for more accurate predictions by effectively capturing the temporal correlations inherent in this complex market.

1 Introduction

Among virtual economies, Path of Exile stands out as it features a distinct league system for its player driven economy. The league system of Path of Exile (POE) consists of leagues which are generally three-month periods where new content is added to the game and the virtual market starts fresh. Thus, creating an equal starting point for each player, leading to prior knowledge being the only differing point between players.

Furthermore, unlike typical game economies that use a standard currency like gold or some kind of coin, POE operates entirely on a player-driven barter economy, where every item could in theory be exchanged for any other. Though, players have opted to designate a few currencies as the main currencies due to inventory limits and ease of transaction needs, as each transaction is directly player-to-player rather than through an exchange medium. The designated currencies are 'Exalted Orbs' and 'Chaos Orbs', though as 'Chaos Orbs' are higher in supply and lower in rarity they are often used as a basis for pricing. Therefore, throughout the analysis and prediction process to follow 'Value' is interchangeable with 'Chaos Orbs'.

The objective of this time series analysis and forecasting project is to predict the price of 'Exalted Orbs' 5 days into the future provided POE economic data sourced from *poe.ninja [1]*.

The initial data set from *poe.ninja* [1] consisted of over 2,000,000 data points describing the daily closing prices of various in-game items and currencies spanning from 2016 to 2024. Of which only a small subset of the data was for 'Exalted Orbs' in specific was selected as the training and testing data. The training data set consisted of 2564-2584 examples with 6 features, namely Date, League, Get, Pay, Value, and Confidence. While the test data set was chosen to be a single league's data, thus ranged in size from 70-116 total observations depending on league length in days.

Model performance for the predictive models used were measured both in mean squared error (MSE) and mean absolute percent error (MAPE). The latter of which was chosen for its ability to provide a give or take error metric for the predicted prices vs. the actual prices.

The overview of the approach used was as follows:

- *Exploratory Data Analysis*: Directed to gain an understanding of the structure of the data, and distribution of initial variables.
- *Data Preprocessing*: Nominal and Ordinal features were encoded, Numeric features were normalized to a range of 0-1, and extensive feature engineering was performed.
- *Analysis and Modeling*: A variety of regression techniques were used, and enhancements to these regression techniques were explored.

2 Exploratory Data Analysis

The first step of the time series analysis and forecasting process was to perform exploratory data analysis. This step aimed to gain a greater understanding of the characteristics of the data provided initially. The Python libraries pandas, numpy, and matplotlib were used for this phase.

The exploratory data analysis consisted of three components:

- *Data exploration (Summary Statistics and Autocorrelation analysis)*
- *Distribution of Numeric Features*
- *Trend Analysis*

2.1 Data Exploration

2.1.1 Data Type: From initial data visualization and exploration, a low number of features compared to data examples was exhibited seen in Figure 1, thus signaling the need for significant feature engineering.

Figure 1: Distribution of Features by Data Type

It was noted that 4 of the 5 initial features in the data set were non-numeric and would need to be encoded or converted for the regression models.

2.1.2 Autocorrelation Analysis: Given the nature of the objective, to predict continuous prices of 'Exalted Orbs' 5 days into the future, there was a need to understand the autocorrelation between past values and current values.

From Figure 2, values of 'Exalted Orbs' are seen to be highly positively autocorrelated to previous values. Therefore, a proportional increase or decrease in current values based on previous values is expected.

Further, distinct increases and decreases in autocorrelation between values is depicted in Figure 2. This pattern is hypothesized to be related to the league system of Path of Exile. As a new league begins, the autocorrelation between past and present values rises until it peaks at around the middle of the league, and as the league gets closer to its end, the autocorrelation dips, presumably due to the price stabilizing towards the normal economic market value. The normal economic market being where items and currencies are transferred to after each league ends.

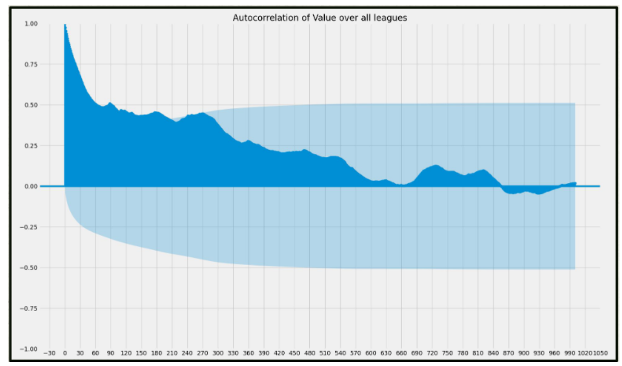

Figure 2: Autocorrelation of 'Exalted Orbs' over all leagues

2.2 Distribution of Numeric Features

The distribution of the only numeric feature in the data set was a multi-modal normal distribution, with a slight skew to the left. This skew can be attributed to the general nature of the league system, where prices tend to hover at the same value near the middle of each league.

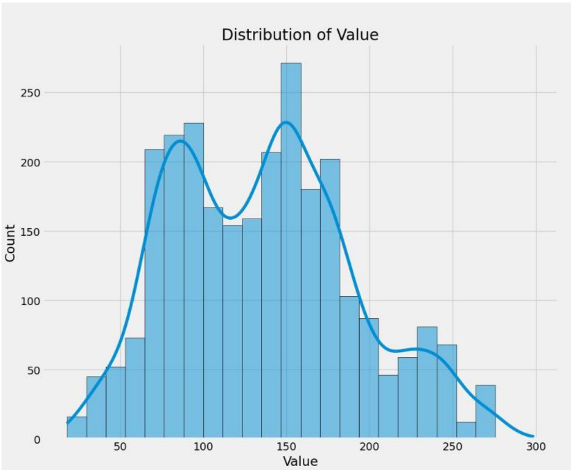

Figure 3: Distribution of Numeric Features

2.3 Trend Analysis

Given the nature of the Path of Exile economy, it was hypothesized that there would be a general distinct trend for the value of 'Exalted Orbs' . As seen in Figures 4 and 5, a distinctive sharp increase in value was seen within the beginning period of each league. This increase in value can be attributed to the player count reaching its peak at the start of the league and the short supply of both 'Exalted Orbs' and 'Chaos Orbs' due to the fresh economy.

A few other noticeable trends are depicted in the two Figures 4 and 5, namely at the middle and end of each league. It is seen that the value of the currency being analyzed stabilizes following the start of each league, before eventually shifting to a lower value as the end of the league draws nearer. These key features of the data suggest a correlation between player count and the value of 'Exalted Orbs', in addition to a correlation between end of league price and the normal market price. These insights will be pivotal for feature engineering in the later process of this study.

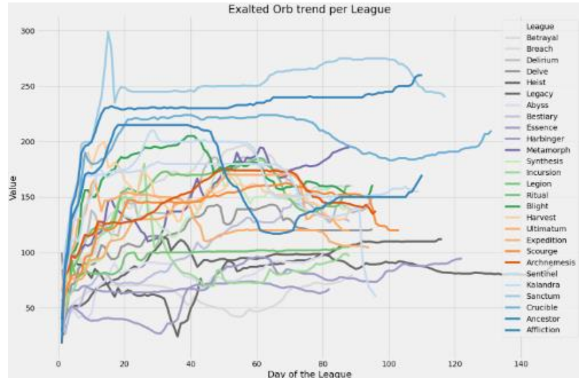

Figure 4: Value trend of 'Exalted Orbs' per league

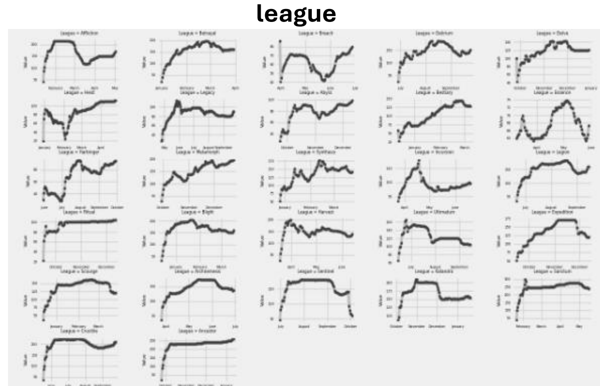

Figure 5: Separated Value trends of 'Exalted Orbs' for each league

3 Data Preprocessing

One way to make up for the low number of initial features in the data set is to clean the data effectively and perform extensive feature engineering [3]. As most models have underlying assumptions about the data, preprocessing the data to fit these assumptions is key to improving modelling results. Therefore, much of the research time was dedicated to preprocessing and engineering the data.

3.1 Data Cleaning

3.1.1 Outlier Detection: The first step performed in the data cleaning process was the detection of outliers in the training dataset.

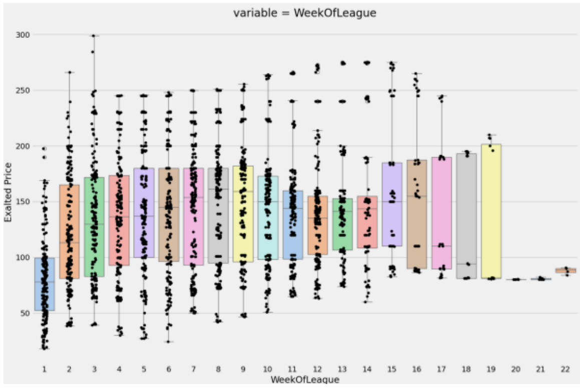

Figure 6: Boxplots of 'Exalted Orbs' Value for each week of the leagues.

To perform the outlier detection boxplots for the 'Exalted Orbs' value per week of the league were used, Figure 6. From which, outliers were detected within the first week of leagues and the 11th, 12th, 13th, and 14th weeks. These were hypothesized to have occurred due to outliers in the new content added during specific leagues. Since the objective of this study was to potentially predict these occurrences, the outliers were not removed.

3.1.2 Conversion of Data Types: Predictive models such as the various regression techniques used in this study, tend to require numerical features to perform prediction. Thus, any non-numerical features were converted to either one-hot encodings or ordinal (sequential) encodings.

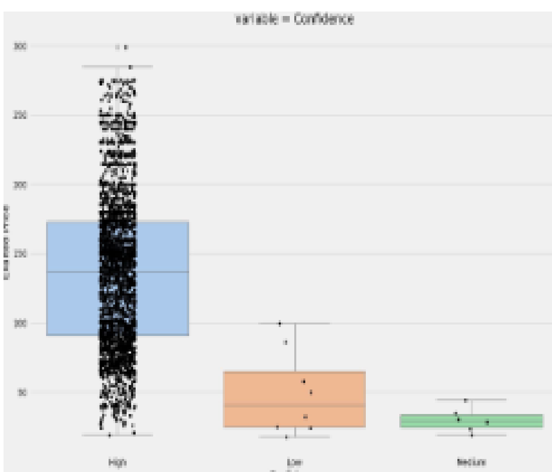

Figure 7: Boxplots of Exalted Price vs. Confidence

The feature 'Confidence' for instance was noticed to have a distinct order from low to high, thus ordinal encoding was applied.

While for the date it was opted to be split into three parts, namely, day of the league, week of the league, and month of the league. It was predicted based upon experience with the game of Path of Exile and from the former data exploration, that the value at any given time is highly correlated with the relative time within the data point's league.

3.2 Feature Engineering:

Substantial effort was dedicated to enhancing the feature engineering for the dataset, as the initial features proved inadequate for capturing the intricate dynamics observed in the Path of Exile (POE) economy. In virtual markets such as those in POE, there is a pronounced correlation between the interest in the game's current league and the values of items and currencies.

Based upon this, a critical feature to be engineered was the relationship between the generation of the base currency, 'Chaos Orbs', and the current player count at time *t*. This feature is pivotal because it directly links economic activity within the game to player engagement, potentially offering insights into market behaviors and price fluctuations.

However, due to the lack of direct data on the currency generation rate, an approximation method was necessary. This approach would involve estimating the Chaos Orbs generation based on available proxies or derived metrics, which include player counts, and prior knowledge of POE.

3.2.1 Chaos Engineering: Based upon prior knowledge about the game's marketplace mechanics and estimations about the 'Chaos Orb' generation rate, it was concluded that a logarithmic growth function could sufficiently map the change in 'Chaos Orb' generation rate as a league progresses. This conclusion was derived from the idea that due to the complete fresh start in each league, players will at first only slowly gain 'Chaos Orbs' due to player weakness, and as player strength grows so too does the generation rate. However, given the limit placed on player strength a need for a limit on the generation rate was needed.

$$\text{chaos\_rate} = \frac{\text{max\_rate}}{1 + e^{-\text{growth\_rate}\cdot(\text{day}+t_0)}}$$

Equation 1: Chaos creation rate logistic growth equation.

$$t_0 = -\frac{\ln\left(\frac{\text{max\_rate}}{\text{base\_rate}} - 1\right)}{\text{growth\_rate}}$$

Equation 1.1: Shift equation for Equation 1.

A similar idea led to the creation of an estimation function for the player count at a given time. As can be seen in Figure 8, a notable pattern is exhibited in the player count over time. Distinct peaks can be seen, each of which corresponds to a new league's starting point. Thus, the player count is at its max for any given league at the start, which makes sense given interest in the new game content should be at its highest when it is first released. Following the start of each league the player count decays exponentially as interest wanes. Equations 2 and 2.1 show this relationship.

$$\text{player\_count} = (\text{initial\_player\_count} - \text{end\_player\_count}) \cdot e^{-\text{decay\_rate}\cdot(\text{day}-1)} + \text{end\_player\_count}$$

Equation 2: Exponential decay of player count over a league.

Utilizing the approximation functions for chaos generation rate (equation 1) and player count (equation 2), the chaos created per day feature was derived, equation 3. Additionally, a running total of 'Chaos Orbs' in the market was kept as a feature for each league. The purpose of which was to simulate the saturation of the market.

$$\text{chaos\_for\_day} = \text{player\_count} \cdot \text{chaos\_rate}$$

Equation 3: Chaos Orbs generated per day.

Figure 8: Path of Exile player count chart from 2016-2024 *(source: Steam Charts)*

3.2.2 Time Series Features: The primary objective of this study is to make predictions based on a time series dataset. To achieve this, it was essential to engineer key time series features such as rolling, lagged, and return features. These features enrich the predictive models with additional historical context, thereby enhancing prediction accuracy.

The first time series features to be engineered were the rolling features. Two different window frames were utilized, one a local rolling window containing 5 data points, and the other a more extensive general rolling window consisting of 15 data points. For each window, the rolling minimum, maximum, and mean values were calculated. These metrics provide insights into the short-term fluctuations and longer-term trends in the data.

Following the creation of rolling features, lagged features were introduced to address the significant autocorrelation observed between current values and their historical counterparts. By incorporating the values from one day, three days, and one week ago into the dataset, the models were equipped with the necessary temporal context to effectively capture and leverage the dynamics of the series.

To further provide features for the models to capture these dynamics of the series, daily return and percentage returns were incorporated. These aimed to highlight the volatility of 'Exalted Orb' value from day to day.

4 Analysis and Modeling

Before commencing the analysis and modeling, two final processes needed to be performed. The first was to split the data set into training and test sets. Around 96.5% of the data set was designated as the training set. While a single league's worth of data was decided as the test set (~3.5% of the total data), which would allow for simulation of the predictive models on a new league start.

The second process was to perform min-max normalization on the training and test data, to prevent overvaluation of features due to possessing on average larger values, such as between total chaos supply, which has a large range of values from 0 to 10,000+, and daily return which ranges between values less than 100.

4.1 Baseline Model: Regression using a Multi-layered Perceptron

As the objective was to forecast the value of a continuous feature multiple days into the future based on observations, a simple multi-layered perceptron (MLP) was selected as the baseline model. MLPs with relu units provide a flexible architecture capable of modeling non-linear relationships inherent in complex time series data like those found in financial markets or virtual economies in this case.

The results from the test data seem to suggest that the MLP regression model does a decent job at fitting the data towards the middle of the league, at least with sufficient hidden nodes. Which was to be expected given the relative stability of value during that period in each league. However, the model failed to capture the complex nature of the value at both the start and end of the test league. This suggests that the model is unable to inherently account for the temporal context and lacks the ability to generalize well to other leagues, given seasonality adjustments or new pattern emergence caused by specific game content being added to POE. The results can be seen in Figures 9 and 10.

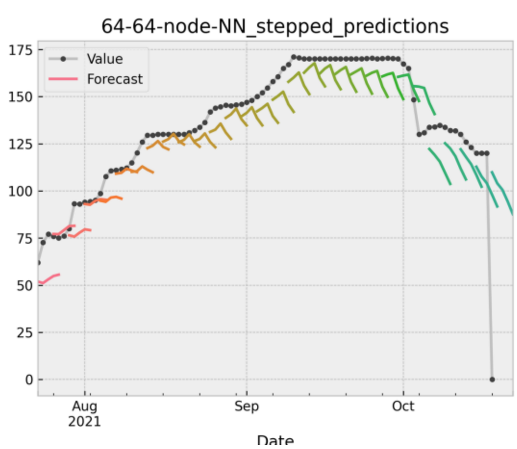

Figure 9: 64x64 2-layer MLP's stepped predictions plot over a test league.

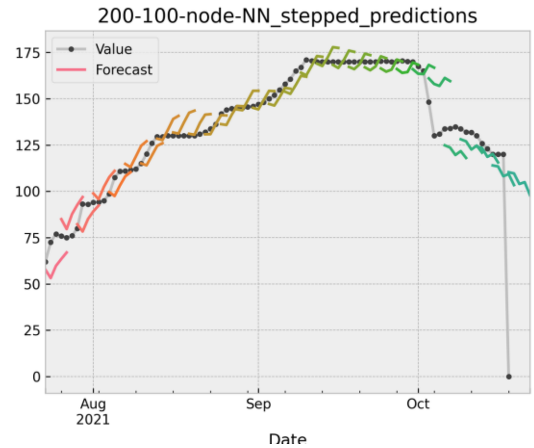

Figure 10: 200x100 2-layer MLP's stepped predictions plot over a test league.

4.2 Advanced Models: Regressor Chains (LASSO, RandomForest, XGBoost)

As highlighted previously, the baseline model struggled with capturing the temporal patterns and context essential for accurate time series forecasting. To overcome this limitation, the study proceeded to explore the use of regressor chains. Distinct from MLPs, regressor chains are designed to account for the autocorrelation between past and present values, thus effectively incorporating temporal context into the analysis. This method enables more precise modeling of time-dependent relationships, which is crucial for enhancing the predictive accuracy of the models. In this phase, various regressor chain configurations employing LASSO, RandomForest, and XGBoost were evaluated to determine their efficacy in capturing complex temporal dynamics.

4.2.1 Lasso Regressor Chains: The first regressor chain configuration to be explored utilized Lasso regression. Lasso minimizes the residual sum of squares with a penalty imposed on the number of coefficients in the model (L1). It has the property that if a regularization constant is sufficiently large, some of the coefficients are driven to zero [2].

This technique is effective for sparse data sets as features that don't meaningfully contribute to the prediction are driven out of the model.

The stepped prediction results of the model, seen in Figure 11, show an improvement over the baseline model in predicting the values at the beginning of the test league. However, a decrease in performance is noted towards the middle-end of the league. This suggests further improvement is needed to learn the complex end of league pattern.

From the feature importance plots in Figures 12.1-12.5, a high importance is placed on the previous regressor's prediction, with a small importance placed on the day of the league. This highlights a pattern between the temporal space and the value.

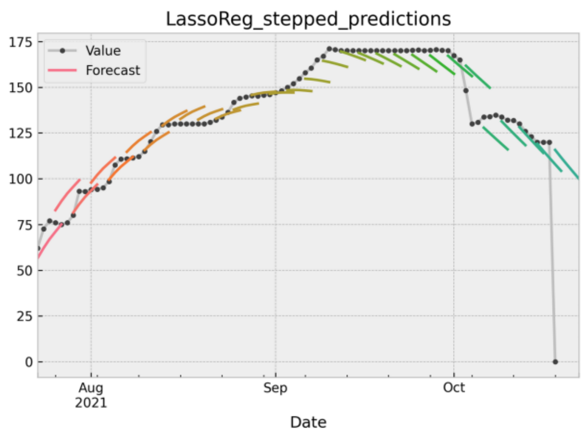

Figure 11: Results of stepped prediction over a test league with Lasso regressor chains.

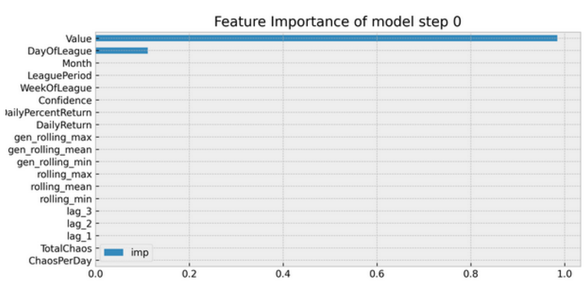

Figure 12.1: Feature importance at step 0 of the Lasso model.

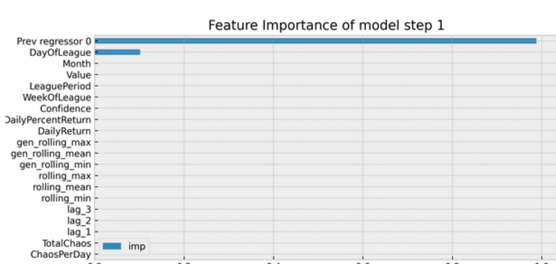

Figure 12.2: Feature importance at step 1 of the Lasso model.

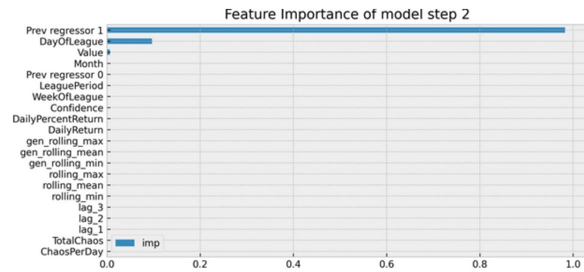

Figure 12.3: Feature importance at step 2 of the Lasso model.

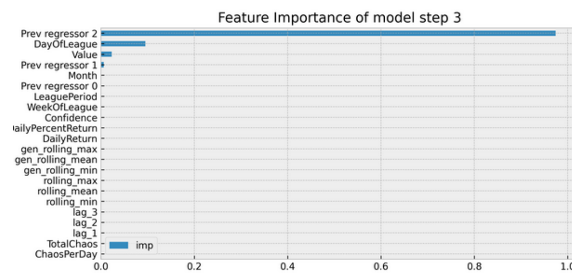

Figure 12.4: Feature importance at step 3 of the Lasso model.

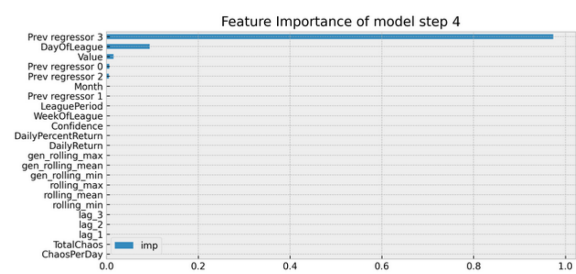

Figure 12.5: Feature importance at step 4 of the Lasso model.

4.2.2 Random Forest Regressor Chains: The next configuration to be explored within the regressor chains framework involved using Random Forest regression. Random Forest constructs a multitude of decision trees during training and outputs the mean prediction of the individual trees. This ensemble method is effective at reducing overfitting common with high-dimensional data.

This technique was chosen due to its property to capture non-linear relationships in complex datasets, as it can handle many features without variable deletion unlike the Lasso model.

Overall, the Random Forest model performed better than the Lasso model in predicting values near the middle and end of the test league. Which suggests that it was able to learn the complex patterns more effectively.

The feature importance plots, detailed in Figures 14.1-14.5, reveal that previous regressors' predictions hold significant weight, while the specific day of the league plays a lesser role. This underscores the influence of historical data and the interdependencies between sequential predictions in capturing the dynamics of the league's progression.

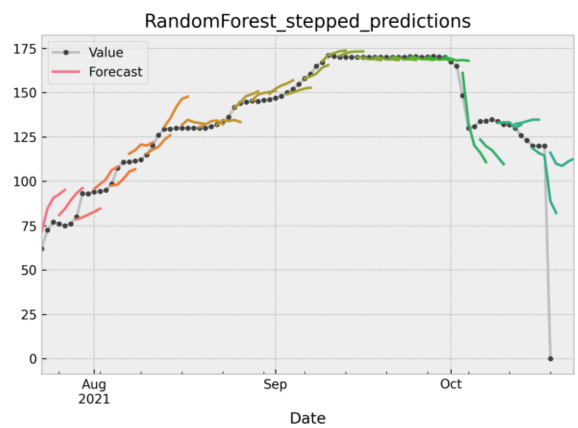

Figure 13: Stepped predictions over a test league with the Random Forest Regressor chain model.

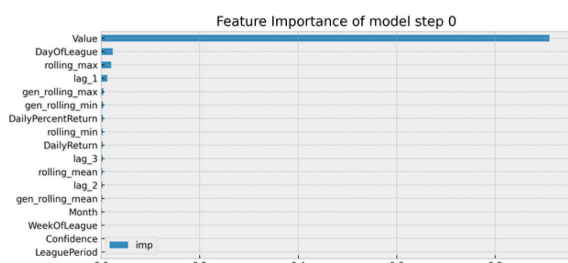

Figure 14.1: Feature importance at step 0 of the Random Forest regressor chain model.

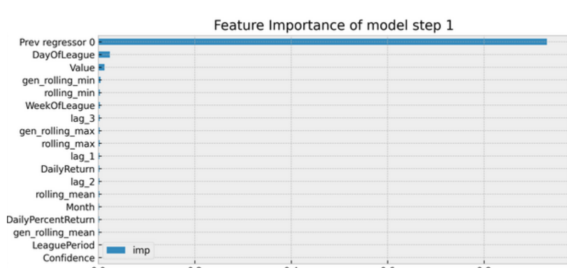

Figure 14.2: Feature importance at step 1 of the Random Forest regressor chain model.

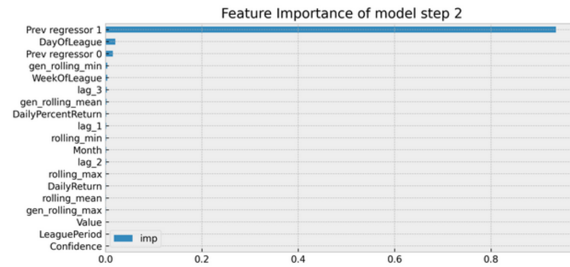

Figure 14.3: Feature importance at step 2 of the Random Forest regressor chain model.

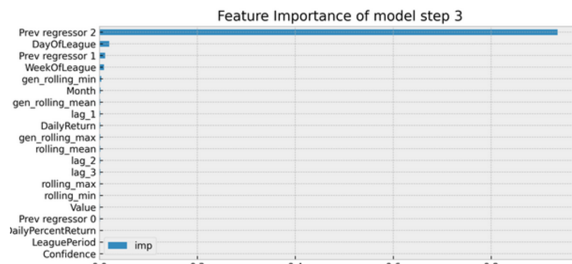

Figure 14.4: Feature importance at step 3 of the Random Forest regressor chain model.

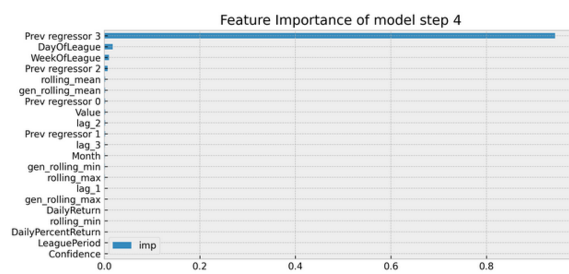

Figure 14.5: Feature importance at step 4 of the Random Forest regressor chain model.

4.2.3 XGBoost Regressor Chains: The final configuration explored was the XGBoost Regressor Chains. XGBoost, or Extreme Gradient Boosting, like Random Forest employs an ensemble technique that builds decision trees incrementally. Each new tree corrects errors made by previously trained trees, making it a powerful tool in complex predictive modeling scenarios.

XGBoost is particularly effective due to its ability to handle various types of data structures and distributions, including outliers and non-linearities. It also includes built-in regularization

(L1 and L2), which helps prevent overfitting by penalizing complex models. The stepped prediction results, as depicted in Figure 15, show a notable improvement over the baseline model in predicting start and middle patterns, however it performs similarly to the Random Forest model in that it fails to capture end of league value trends.

Remarkably, XGBoost attributed significant importance to the 'ChaosPerDay' feature, alongside the predictions from previous regressors, as seen in Figures 16.1-16.5. This indicates that the model effectively captured the interplay between player interest, league mechanics, and the autocorrelation of values. In future studies, it could be beneficial to incorporate a normal market valuation of 'Exalted Orbs' as the league progresses, to reflect the intricate valuation patterns more accurately at the league's conclusion.

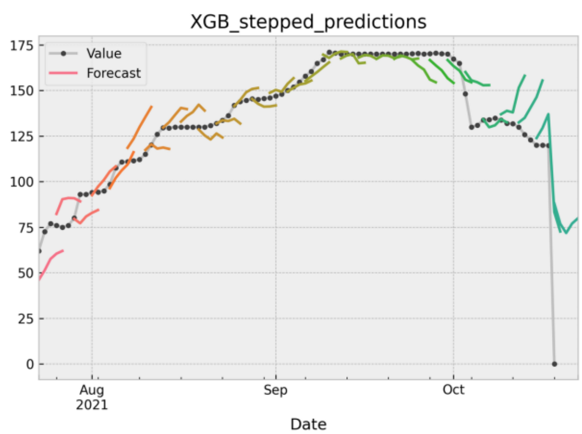

Figure 15: XGBoost Regressor chain stepped predictions over a test league.

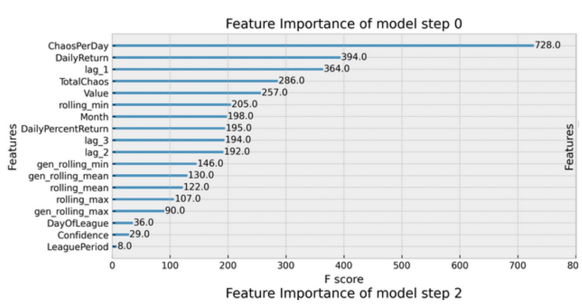

Figure 16.1: XGBoost feature importance plot at step 0.

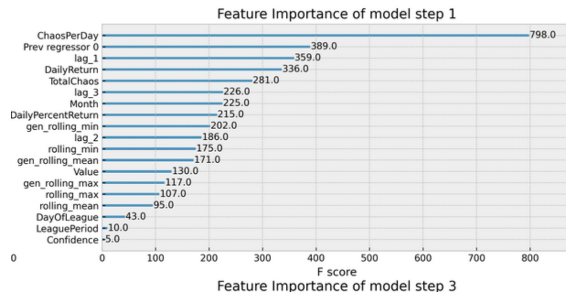

Figure 16.2: XGBoost feature importance plot at step 1.

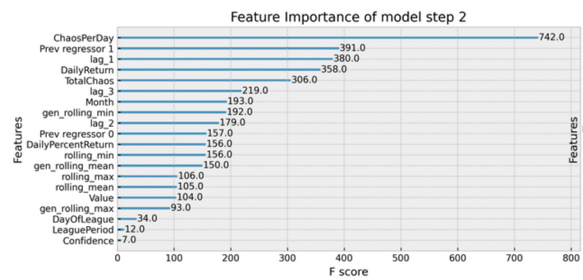

Figure 16.3: XGBoost feature importance plot at step 2.

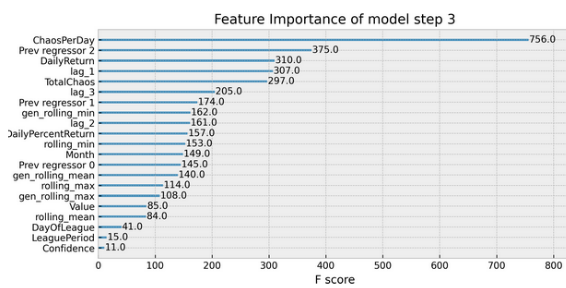

Figure 16.4: XGBoost feature importance plot at step 3.

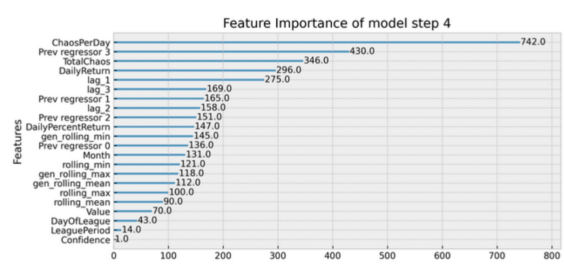

Figure 16.5: XGBoost feature importance plot at step 4.

Regressor Chain Results

Model	MSE (Train)	MSE (Test)	MAPE (Train)	MAPE (Test)
Lasso	642.678	441.252	0.116	0.076
Random Forest	251.484	391.038	0.91	0.079
XGBoost	44.649	267.916	0.062	0.080

Figure 17: Regressor Chain Results.

5 Conclusions

This time series analysis delved into various regression techniques with the goal of developing a model capable of predicting the value of 'Exalted Orbs' in Path of Exile up to five days in advance. Among the challenges encountered was the limited number of features—initially only five—relative to the total number of observations. Additionally, it was challenging to delineate the relationship between player interest in a POE league and the value of 'Exalted Orbs', which is essential for capturing the complex dynamics of Path of Exile's virtual economy, including the distinct phases of each league: start, middle, and end.

Overall, the Lasso model emerged as the top performer based on the mean absolute percentage error (MAPE) value shown in Figure 17. However, XGBoost displayed considerable potential for future research. It achieved the lowest MAPE value for the training dataset and, although its MAPE was slightly higher for the test set compared to the Lasso and Random Forest models, this discrepancy could be attributed to the small size of the test dataset, which might skew results given that each league exhibits unique value patterns. As previously suggested, incorporating a feature that captures the standard market valuation of 'Exalted Orbs' might enable the XGBoost model to distinguish itself further from its competitors.

6 Acknowledgements

Sincere acknowledgement must be given to Dr. Pawlicki for his impeccable teaching, both easy to understand and thought provoking. Thanks for being a true teacher. In addition, high recognition must be given to Saon Pal for his brilliant insights on virtual economies and pattern recognition in the game Path of Exile.

7 References

[1] Path of Exile. 2024. Daily Prices of Exalted Orbs. poe.ninja. https://poe.ninja/data
[2] Bishop, Christopher M. 2006. Pattern Recognition and Machine Learning. Springer.
[3] Jiawei Han, Jian Pei, and Micheline Kamber. 2011. Data mining: concepts and techniques. Elsevier.