Mining Causal Links Between Real-World Events and TV Content Viewing Patterns

Orlando Duarte Rodrigues Ferreira de Melo
Master's degree in Network and Information Systems Engineering
Computer Science Departament
2021

Supervisor
João Vinagre, Guest Assistant Professor, Faculty of Sciences of the University of Porto

Co-Supervisor
Jessica Condesso Delmoral, Researcher, NOS SGPS

U.PORTO

FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

Todas as correções determinadas pelo júri, e só essas, foram efetuadas.

O Presidente do Júri,

Porto, _____/_____/_____

Abstract

In recent decades, the offer of television content has seen an increase in viewing options, channels, and programs. This new reality has made this service's customers adopt increasingly diverse and personal usage patterns. This makes the task of TV content distributors more difficult since with the increase in the amount of television content it becomes difficult to anticipate customer preferences and to recommend the best possible content. Taking this environment into account, characterized by risk and uncertainty, it is necessary to adopt customer-focused strategies that enable a television content offer that is better adjusted to the needs of each consumer.

Integrated into a project of a large Portuguese telecommunications company, this dissertation aims to carry out an in-depth study of the impact that real-world events like characteristics of football tournaments, news, Google search interest and weather have on sports live television content viewing. To this end, data mining and causality techniques were applied to a dataset of television audiences from the telecommunications company, along with several external data sources referring to real-world events.

This dissertation studies the impact of external real-world events on sports TV audiences in three ways: (1) A study of external data related to audience behavior for a specific time; (2) a comparison of forecasting accuracy of a classic statistical approach - based on the past values of the volume of clients - and a machine learning approach - based on the past values of the volume of clients and sports related external real-world events; (3) a Granger causality analysis of the effect of external real-world events in volume of clients and viewing times.

The results show a clear influence of external events on sports TV volume of clients and viewing time. External factors such as tournaments characteristics, match popularity, match interest and the home team effect proved to be the most informative about the TV audiences.

Keywords: real-world data; data fusion; forecasting sportscasts; ensemble methods; Granger causality;

Resumo

Nas últimas décadas, a oferta de conteúdos televisivos tem registado um aumento de opções de visualização, canais e programas. Esta mudança de paradigma, fez com que os clientes deste serviço adotassem padrões de uso cada vez mais diversificados e pessoais. Isto torna a tarefa dos distribuidores de conteúdo televisivo mais difícil, pois com o aumento da quantidade de conteúdos torna-se difícil antecipar a preferência dos clientes para assim recomendar o melhor conteúdo possível. Tendo em conta este ambiente, caracterizado pelo risco e incerteza, é necessária a adoção de estratégias focadas no cliente que permitam uma oferta de conteúdos televisivos mais ajustada às necessidades de cada consumidor.

Integrada em um projeto de uma empresa de telecomunicações, esta dissertação tem como objetivo realizar um estudo aprofundado do impacto que eventos do mundo real como a dinâmica de torneios de futebol, notícias, interesse de pesquisa do Google ou clima têm na visualização de conteúdo desportivos televisivos em direto. Com esse objetivo, técnicas de data mining e causalidade foram aplicadas a um conjunto de dados de visualizações de televisão da empresa de telecomunicações, juntamente com várias fontes de dados externas referentes a eventos do mundo real.

Esta dissertação estuda o impacto de eventos externos do mundo real nas audiências de conteúdos desportivos televisivos de três maneiras: (1) Um estudo de dados externos relacionados ao comportamento do público em um momento específico; (2) uma comparação da precisão da previsão de uma abordagem estatística clássica - com base nos valores anteriores do volume de clientes - e uma abordagem de aprendizagem de máquina - com base nos valores anteriores e em eventos externos do mundo real relacionados; (3) uma análise de causalidade de Granger do efeito de eventos externos do mundo real no volume de e nos tempos de visualização.

Os resultados mostram uma clara influência de eventos externos no volume de clientes e no tempo de visualização de programas desportivos televisivos. Fatores externos como a dinâmica de campeonato, a popularidade dos jogo, o interesse gerado em volta do jogo e o afeto dos clientes com o jogo mostraram-se os mais informativos sobre as audiências televisivas.

Palavras-chave: variáveis do mundo real; fusão de dados; previsão transmissão desportivas em direto; métodos ensemble; causalidade de Granger;

Acknowledgements

This dissertation marks the endpoint of my journey at FCUP. A path made of constant learning and curiosity. For making this possible, my thanks to FCUP and all the people who are part of it.

Furthermore, this dissertation would not be possible without the contribution of several people.

First of all, and because they had a direct influence on the outcome of this work, I would like to thank my two supervisors. To Prof. João Vinagre for all his support, availability, and guidance and to Jessica Delmoral for all her patience, availability, and all the knowledge transmitted. For these reasons, my gratitude to both.

Would also like to thank my classmates and childhood friends for sharing this 5-year journey with me.

Finally, a special thanks to my parents and brothers for always believing in me and supporting me even in the most difficult times.

Contents

List of Tables

List of Figures

Listings

Acronyms

AI Artificial Intelligence

API Application Programming Interface

ARIMA AutoRegressive Integrated Moving Average

BA Business Analytics

CRISP-DM CRoss Industry Standard Process for Data Mining

EDA Exploratory Data Analysis

EPG Electronic Programming Guide

IBM International Business Machines Corporation

IDC International Data Corporation

KDD Knowledge Discovery from Databases

ML Machine Learning

RMSE Root Mean Square Error

SAS Statistical Analysis System

SHAP SHapley Additive exPlanations

SMAPE Symmetric Mean Absolute Percentage Error

SPI Soccer Power Index

XGBoost Extreme Gradient Boosting

Chapter 1

Introduction

The distribution of TV content, in recent years, has seen an increase in terms of diversification of channels, contents, and viewing options. Audiences are more and more fragmented, with people watching TV at different times on different platforms. As a result, it is more difficult to outline metrics about the profile of the users who consume this type of content.

At the same time, the fast development of networking, data storage, and data collection capacity has enabled new levels of scientific discovery and economic value. Through big data analytics, commercial enterprises can do better job at monitoring acceptance of products, providing personalized services able to adapt to individual needs, and understanding their business environment, potentially fueling competitive advantages and boosting the quality and effectiveness of decision making.

In the context of this rapidly evolving television landscape and big data era, we set out to examine high-dimensional data and gain insights into the relationship between real-world events and the TV watching patterns. This new approach to television audience promises more stability and predictability, for an industry typically characterized by risk and uncertainty.

1.1 Motivation

Although television consumption has decreased in recent years, sports TV live broadcasts are still today one of the most popular broadcasting media content [101]. Understanding the drivers that lead people to see sports content is, therefore, of great importance for a wide range of fields, such as economics [7], broadcasting management [95] and marketing [100].

The drivers that lead people to see certain content can be categorized into two types: individual variation or structural variation [106].

In the first case, the drivers of TV consumption are linked to individual characteristics. For example, people who have a more active social life are not expected to have the same viewing patterns as people who have fewer outside contacts. In the first case, people tend to be more

selective in the content they seek, while in the second case, people have more varied viewing patterns to get experiences they didn't have in their normal life [79].

Although this type of individual analysis is useful in controlled environments, as it allows explaining the consequences of exposure to television, it is not so useful when the objective is to unveil mass behavior, generally of greater value to the industry [106].

The second perspective ignores the individual characteristics and uses aggregated audience measurements (e.g., total viewing time of a certain program). This is an approach more focused on finding structural patterns about audience behavior (e.g., what impact does the scheduling of a program have on audience behavior).

In recent years, we have witnessed a growing interest in research that aims to develop tools for real-world event detection and characterization (e.g., weather, football tournaments, twitter trends). However, it is still not clear the effect that this type of events has on people's engagement in TV visualization (i.e., the structural patterns on audience behavior).

With the proliferation of social media usage by large portions of the population, coupled with recent advances in data collection, storage, and management, have made it possible for research organizations and data scientists to analyze massive amounts of data [55]. This knowledge can have several important implications. First, TV distributors may be willing to adjust the content and advertisements to the target audience. Second, TV recommender systems may leverage this knowledge and adapt their recommendations accordingly.

1.2 Objectives

With this dissertation, we intend to conceive and develop a solution that will investigate causal relation between external real-world events and live sports TV audiences, thus producing a more accurate model of the users' viewing patterns.

Therefore, the goal of our research was to address the following research questions:

- RQ1: What data sources will be useful for our problem?
- RQ2: What data analysis pipeline apply to our data?
- RQ3: What type of causal patterns they have?
- RQ4: How do these patterns affect the live sports TV audiences?

Having that in mind, for this work the main goals are:

- Investigate external behavioural/opinion related data that may take an effect on enterprise outcomes for a specific amount of time;
- Validate the effect taken, using data mining techniques and causality statistical analysis methods to extract meaningful cause-effect patterns;
- Measure quantitatively and qualitatively the desirable effects on the specific business outcome (TV volume of clients and viewing time);
- Present data-driven conclusions of the external factors/data that have the greatest impact in viewing times and volume of clients, being the ultimate goal to find patterns that maximize these parameters.

1.3 Contributions

This dissertation will have the following contributions:

- We research data sources and APIs that can have an impact on our business goal target and generate a dataset with standardized data;
- We developed a pipeline of methods for the analysis of this data;
- We Present data-driven conclusions of the impact of events occurring in external factors (such as the weather), in the TV service viewing times and volume of clients.

1.4 Organization

The remainder of the thesis is organized as follows. This section provides an introduction and motivation for this work. Chapter 2 provides the background required for the proper understanding of this work. Chapter 3 presents a literature review of forecasting in the Big data era and its application in predicting sports live TV audiences. As well as, an overview of causality analysis across different research areas, with special emphasis on the Granger causality framework. Chapter 4 compares the accuracy of forecasting sports live TV audiences with a simpler statistical method and a more complex machine learning method. The last, including several external factors as input features of the model. Chapter 5 presents a causality analysis of the impact of external factors on the service viewing time and volume of clients of a popular local tournament. Finally, in chapter 6 the conclusions and future work are presented.

Chapter 2

Background

Over the past few decades, the amount of data generated has grown exponentially. According to the International Data Corporation (IDC), the world's data will grow 5x from 33 Zettabytes (ZB) in 2018 to 175 ZB by 2025 [90].

With this increase of global data, a new term called big data has emerged. Big data, compared with traditional datasets, includes a large amount of unstructured data that needs more real-time analysis.

We can describe Big data as three main dimensions (The Three V's) [40]:

- **Volume.** Refers to the magnitude of data, generally, several terabytes and petabytes;
- **Variety.** Refers to the structural heterogeneity in a dataset (structured, semi-structured, and unstructured data);
- **Velocity.** Refers to the rate at which data are generated and the speed at which it should be analyzed and acted upon.

In addition to the three V's, other dimensions of big data have also been mentioned. These include [40]:

- **Veracity.** IBM designated veracity as the fourth V, which represents the uncertainty inherent in some data source;
- **Variability.** SAS introduced variability as the variation in the data flow rates;
- **Value.** According to Oracle, big data are often characterized by relatively "low-value density".

With the advance of science and technology, the developments in causality analysis have been heavily influenced by the Big Data era, with a number of studies emerging in the last decades

[42], such as education [26][49][51][62], medical science [73] [25], economics [56], epidemiology [50][91], meteorology [32], environmental health [66] and sports [30][28].

Answering causal questions with big data leads to some unique new problems. For example, public databases or data collected via web crawling or application program interfaces (APIs) are unprecedentedly large, we have little intuition about what types of bias a dataset can suffer from [42].

To uncover these hidden patterns in the data, the successful adoption of Data Mining techniques or a combination of techniques will be important to the success of causality studies [45].

The remainder of this chapter will introduce what new possibilities and challenges arise for learning about causality in the era of big data.

2.1 Data analytics

There are two common situations in which data analysis can help solve a certain problem or question. The first situation is when the problem is not new and historical data exists with the results achieved (e.g., poor customer performance, malfunction of parts, etc...), such historical data may be used to improve and optimize the presently used strategy to reach a decision. In a second case, a certain question arises for the first time, and only little experience is available (e.g., a new product, a large experiment). Here, the inclusion of data from similar problems can be useful to discover and gain insights about the new problem [13].

Business analytics (BA) is the iterative exploration of an organization's data to gain insight and drive business planning by applying statistical analysis techniques [9]. Depending on the purpose, we can arrange BA into 4 different types (Fig. 2.1): descriptive, diagnostic, predictive, or prescriptive.

- **Descriptive analytics.** Here the goal is to try to unravel *what happened* and alert about that fact. Describe the phenomenon through different means to capture the most important dimensions;
- **Diagnostic analytics.** In this case, *why* something happened is explored. You need to explore existing data or add data to get an answer. To find out the causes of the problem visualization techniques are used;
- **Predictive analytics.** Future business imperatives, potential future outcomes and drivers of observed phenomena are explored using statistical or data mining techniques. Some examples include predicting the outcome of future sales of a product and the behavior of a target customer segment;
- **Prescriptive analytics.** It goes a step beyond predictive analytics, combining decision

options with predicting future outcomes. To assess the best decision that can be taken to optimize business processes in the future, decision analysis tools such as optimization and simulation are employed.

In figure 2.1 we can see the 4 types of business analysis and the question that each tries to answer.

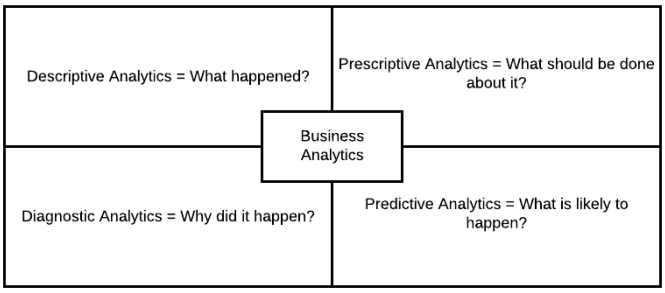


Figure 2.1: Types of Business Analytics [9].

2.1.1 Exploratory data analysis

We can look to practical data analysis as two wide phases: an exploratory phase and a confirmatory phase. Exploratory data analysis (EDA) is concerned with isolating patterns and features of the data and with revealing these to the analyst. Allowing a first touch with the data to effectively analyze it and produce the best model that fits the data. A characteristic of exploratory analysis is its flexibility, both in tailoring the analysis to the structure of the data and in responding to patterns that successive steps of analysis uncover [71].

We can divide an EDA into four main components [1]:

- **Univariate non-graphical.** This data analysis is the simplest of the four because consists of only one single variable. The main objective of a univariate analysis is to describe and find patterns within it;
- **Univariate graphical.** To have a better view of the data, graphical methods are used. Graphics such as: steam-and-leaf plots, histograms and box plots;
- **Multivariate nongraphical.** Multivariate non-graphical EDA techniques usually show the relationship between two or more variables of the data through cross-tabulation or statistics;
- **Multivariate graphical.** In the case of multivariate graphics visualization, some of the most used types of graphics include: bar plot, scatter plot, multivariate chart, run chart, bubble chart and heat map.

Once EDA is complete and insights are drawn, its features can then be used for more sophisticated data analysis or modeling, including machine learning.

Confirmatory data analysis is where you put your findings and arguments to trial. Traditional statistical tools such as significance, inference, and confidence are used. This phase also incorporates an analysis of another, closely related body of data and the step of validating a result by collecting and analyzing new data [71].

2.1.2 Data collection

It is well known that the majority of the time for running machine learning end-to-end is spent on preparing the data, which includes collecting, cleaning, analyzing, visualizing, and feature engineering [92].

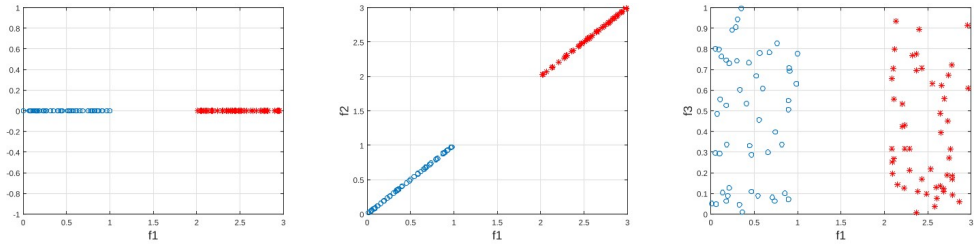
Generally, we can consider three types of data collection [92]: data acquisition, data labeling and existing data. Data acquisition techniques can be used to discover, augment, or generate new datasets. Data labeling can be applied when a dataset already exists however it is necessary to add labels to individual examples. Finally, we can improve the existing data or use trained models. Despite this distinction, these three methods can be used simultaneously.

2.1.3 Feature selection

Data of high dimensionality can significantly increase the memory storage requirements and computational costs for data analytics. The curse of dimensionaltiy is another problem that occurs in this type of data: when the volume of represented space increases, the data does not keep up and becomes sparse [104].

We can solve these problems using dimensionality reduction, a techniques that reduce the number of input variables in a dataset through feature extraction and feature selection. In the first case, the original high dimensional dataset is projected in a new low dimensional space. This type of approach is more suitable if the raw input data does not contain very comprehensive features for a specific machine learning algorithm. A disadvantage of this type of approach is that it creates a new set of features, making further analysis challenging as it does not retain the physical meaning of the original features. In the second case, a subset of the most relevant features for the model construction is select. This approach is best suited when real-world data contains a large number of irrelevant, redundant and noisy features (Fig. 2.2). Removing these variables by feature selection reduces storage and computational cost while avoiding significant loss of physical information or degradation of learning performance [67].

Selecting some subset of a learning algorithm's input variables, either using feature extraction or feature selection, have the advantage of improving computational efficiency, overcome the curse of dimensionality and building better generalization models that do not overfit to new data [67].



(a) relevant feature f_1 (b) redundant feature f_2 (c) irrelevant feature f_3

Figure 2.2: (a) Relevant, (b) redundant and (c) irrelevant features [67].

2.1.4 Data mining

Data mining is the central part of the process of Knowledge Discovery from Databases (KDD). We can break the KDD process, presented in Fayyad et al. [34], into the following steps: data selection, data cleaning, data transformation, data mining, pattern evaluation and interpretation.

Data mining is a process where the goal is to extract patterns and knowledge from large amounts of data. In other words, finding either unusually frequent or infrequent relationships between entries in a dataset.

2.1.4.1 Data mining methods

Four problems in data mining are considered fundamental to the mining process: clustering, classification, association pattern mining, and outlier detection. Following [3] relationships between data items can be classified in two different types:

- **Relationships between columns.** Here, the frequent or infrequent relationships between the values in a particular row are determined. This relationship is either a positive or a negative association pattern problem.

  Supervised learning is a particular case that tries to find patterns of association between columns. In this kind of data mining problem, a feature gains special importance (target feature), and the goal is to use the others input features to predict this special attribute. This problem is referred to as data classification or data regression;

- **Relationships between rows.** Here, the goal is to determine subsets of rows in which the values in the corresponding columns are related. This can be seen either as a clustering analysis if the goal is finding relationships where the subsets of rows are similar, or an outlier analysis if the goal is finding a row that is very different from other rows. In the last case, the outlier row can be also referred to as an unusual data point, or as an anomaly.

2.1.4.2 CRISP-DM

CRISP-DM stands for CRoss Industry Standard Process for Data Mining. It provides a uniform framework and guidelines for data miners, and can be seen as a six phase pipeline (Fig. 2.3) [5][108]:

- **Business understanding.** Is the first phase of CRISP-DM process and focuses on understanding the project objectives and requirements from a business perspective, allowing the definition of the concrete data mining problem and the plan to achieve the business objectives;
- **Data understanding.** The data understanding stage starts with an initial data collection and proceeds with the exploration of the data to get insights to form hypotheses. This phase mainly serves to familiarize with the data and identify possible quality problems that need to be resolved;
- **Data preparation.** The data preparation phase focuses on the selection and preparation of the final dataset. To obtain this dataset a set of tasks has to be performed. Tasks such as table, record, and attribute selection, data cleaning, construction of new attributes, and transformation of data for modeling tools;
- **Modeling.** In this phase, occurs the selection and application of various modeling techniques. Different parameters are set and different models are built for same data mining problem;
- **Evaluation.** Before proceeding to the final deployment of the model, it is important to evaluate it and review, through an in-depth analysis of each step executed to construct the model, whether the model achieves the business objectives properly or not;
- **Deployment.** Finally, in the deployment phase, the knowledge gained will need to be organized and presented in a way that the customer can use it. Being as simple as generating a report or as complex as implementing a repeatable data mining process.

2.1.4.3 Temporal data mining

Temporal data mining can be described as the extraction of information contained in large sequential databases. Where sequential databases correspond to data ordered by some index. These records are usually ordered by time. However, there are examples of sequential data without the notion of time as in the case of text documents, gene sequences, or football match tournament. Here, although the notion of time is not so present, the order of records is of great importance for the proper description and extraction of information from the data [64].

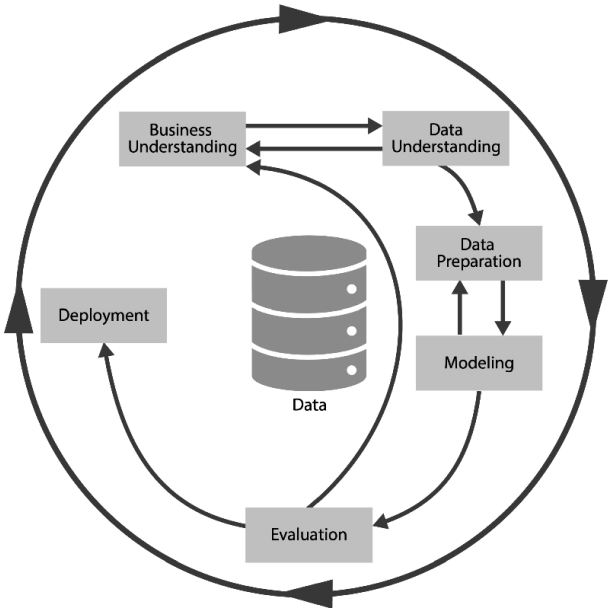


Figure 2.3: The CRISP-DM life cycle [60].

2.2 Time series analysis

A time series is a set of observations x_t, each one being recorded at a specific time t. We can divide time series into two different types: *discrete time series* and *Continuous time series*.

- A *discrete time series* is one in which the set T_0 of times at which observations are made is a discrete set:
  - A data-reporting interval that is infrequent (e.g., 1 point per minute) or irregular (e.g., whenever a user logs in);
  - Gaps where values are missing due to reporting interruptions (e.g., intermittent server or network downtime).
- A *Continuous time series* are obtained when observations are recorded continuously over some time interval, e.g., when $T_0 = [0,1]$.

2.2.1 Stationarity

A time series $\{X_t, t = 0, \pm 1, ...\}$ is considered stationarity if when compared to its shifted version $\{X_{t+h}, t = 0, \pm 1, ...\}$ for each integer h it has similar statistical properties.

Looking only at the first- and second-order moments of X_t, we can described a stationary time series as follows [19]:

Let $\{X_t\}$ be a time series with $E(X_t^2) < \infty$. The mean function of $\{X_t\}$ is

$$\mu_X(t) = E(X_t) \tag{2.1}$$

The covariance function of $\{X_t\}$ is,

$$\gamma_X(r,s) = Cov(X_r, X_s) = E[(X_r - \mu_X(r))(X_s - \mu(s))] \tag{2.2}$$

for all interegers r and s.

$\{X_T\}$ is (weakly) stationary if,

$$\gamma_X(t) \text{ is independent of t} \tag{2.3}$$

and

$$\gamma_X(t+h,t) \text{ is independent of t for each h.} \tag{2.4}$$

2.2.2 Components of a time series

We can classify the patterns in a time series into 4 components: trend, seasonal, cyclical and residual [53] [105].

- Trend($T[t]$): is a pattern with a long-term increase or decrease in the data and represents the mean rate of change with respect to time;
- Seasonality($S[t]$): is a periodical fluctuation where the same pattern occurs at a regular interval of time. Seasonality is always of a fixed and known frequency;
- Cyclical($C[t]$): fluctuations that are not of a fixed frequency. Compared to seasonality, the average length of cycles is longer than the length of a seasonal pattern, and the magnitudes of cycles tend to be more variable than the magnitudes of seasonal patterns;
- Residual($e[t]$): fluctuations that are purely random and irregular.

Figure 2.4 shows the trend, cyclical and residual components of a time series.

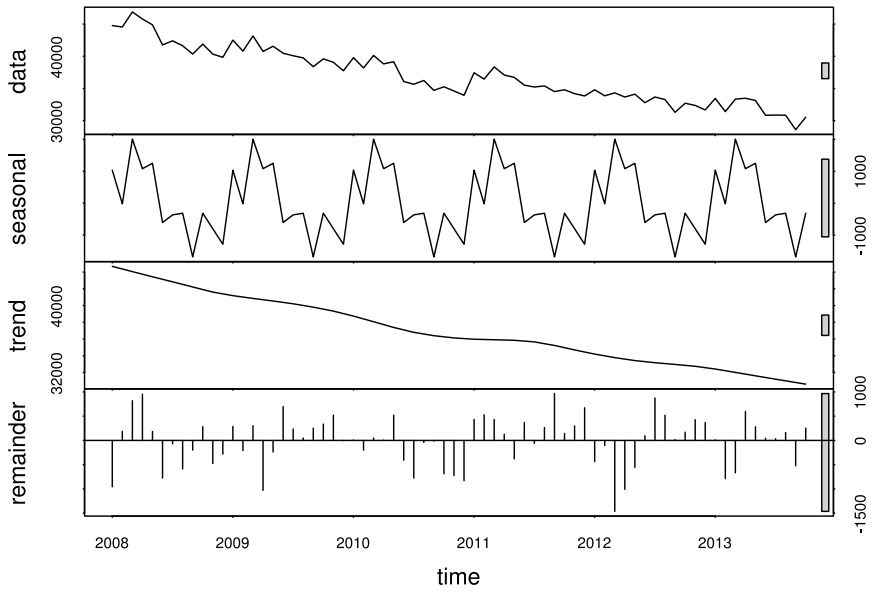


Figure 2.4: Time Series Components.

2.2.3 Time series decomposition

To better visualize the patterns, it is helpful to split a time series into is components such as trends, seasonality, cyclic variance, and residuals. When decomposition a time series usually the trend and cycle are combined into a single trend-cycle component (sometimes called a trend component for simplicity). Thus, during the decomposing, we can see the time series as being formed by three components: a trend-cycle component, a seasonal component and a residual component.

To decompose the data into its components there are basically two methods: additive decomposition and multiplicative decomposition. The additive decomposition is the most suitable if the magnitude of the seasonal fluctuations, or the variation around the trend-cycle, does not vary with the level of the time series. On the other hand, if the variation on the seasonal component, or the variation around the trend-cycle, appears to be proportional to the level of the time series then the multiplicative decomposition must be chosen [53].

The additive decomposition formula:

$$Y[t] = T[t] + S[t] + C[t] + e[t] \tag{2.5}$$

The multiplicative decomposition formula:

$$Y[t] = T[t] * S[t] * C[t] * e[t] \tag{2.6}$$

2.3 Time series forecasting

The forecasting process is used in a wide range of areas such as the stock market, weather, electricity demand and business. Forecasting time series data provides organizations with valuable information that allows anticipate business outcomes and make important decisions [72].

Forecasting methods may be broadly classified into three different types [22]:

- **Judgemental forecasts.** A forecast made on subjective information. Based essentially on intuition and knowledge about the company or market, although some quantitative information can also be included;
- **Univariate methods.** Where future values of a single time series are assumed to be based exclusively on past values, possibly modeled by a time function such as the linear trend;
- **Multivariate methods.** Here, in addition to the dependence on previous values, more additional time-series variables, called the predictor or explanatory variables, are used to forecast future values. A multivariate forecast may be composed of more than one equation if the variables are jointly dependent.

2.3.1 Univariate time series models

Univariate time series models are a class of specifications where a model tries to predict a variable using only information contained in their past values (possibly current and past values) of an error term [20]. An important class of time series models is the family of AutoRegressive Integrated Moving Average (ARIMA) models.

2.3.1.1 ARIMA

ARIMA is a univariate time series method which is based on the premise that information in the past values of the time series alone is enough to predict future values. Presented by Box et al. [16], is on of the most popular time series forecasting techniques. The ARIMA model is defined by 3 parameters (p, d, q) where p is the number of autoregressive terms(AR), d is the number of differences(I) and q is the number of moving averages(MA). An ARIMA model is one where the time series was differenced at least once to make it stationary and you combine the AR(p) and the MA(q) terms. ARIMA's forecast model is given by the following equation:

$$y_t = \theta_0 + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + ... + \varphi_p y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - ... - \theta_q \varepsilon_{t-q} \tag{2.7}$$

where, at time period t, y_t is the actual value and ε_t is the random error; $\varphi_i (i = 1, 2, ..., p)$ and $\theta_j (j = 0, 1, 2, ..., q)$ are the AR term and MA term, respectively. Random errors, ε_t, are

assumed to be independently and identically distributed with a mean of zero and a constant variance of σ^2 [109].

2.3.2 Multivariate time series model

One problem with the ARIMA model is that is mostly limited to linear univariate time series and do not scale well to multivariate time series. Other approaches use vector autoregression (VAR), a generalization of the AR-based models, that captures relationships between multiple time series. However, VAR models (as in the case of AR models) do not capture non-linearity relationships [99]. Machine Learning (ML) methods have been emerging in the literature as viable alternatives to univariate time series models that captures non-linearity relations [98]. In particular, ensemble methods stand out as one of the most used methods for time series forecasting [107] [65].

When we have a high-dimensional dataset extracted from multiple sources, where features have heterogeneous physical properties, a single regressor or classifier does not have the power to learn the information contained in the aggregated data. One solution is to use each data modality to train a different regressor, with the outputs of all models combined to get more accurate prediction results. Applications that take advantage of multiple data sources to make a more informed decision are called data fusion applications. On the other hand, solutions that combine the outcome of several different supervised learning algorithms are called ensemble methods [87].

2.3.2.1 Ensemble methods

The idea behind ensemble methods is to combine several simpler models in order to improve predictive performance over a single estimator [87]. Two major approaches can be used to combine the weak estimators:

- **Bagging.** Learns from weak prediction models independently and combines them following either the aggregation averages (regression analysis) or the majority vote (classification analysis) [17]; This type of technique is designed to improve the stability and accuracy of machine learning algorithms. It also reduces the variance and overfitting of the model.
- **Boosting.** Is an ensemble technique that creates a prediction model by joining the predictions of weak prediction models (such as decision trees or neural networks). These weak estimators are added sequentially to the collection with each of them trying to improve the general ensemble's performance. While boosting is a general algorithm for building an ensemble out of simpler models, it is more effectively applied to models with high bias and low variance [14];

Random forest

Random forest [18] is a bagging method that is fast, robust to noise, does not overfit and offers possibilities for explanation and visualization of its output [46]. The main idea of bagging is to average many noisy but roughly unbiased models in order to reduce the variance. Trees are simple models that do not capture complex interactions, so they are ideal candidates for bagging.

The combined regression formula for random forest can be expressed as follows:

$$\hat{F}(x) = \frac{1}{K} \sum_{k=1}^{K} t_i(x) \tag{2.8}$$

where $\hat{F}(x)$ represents the combined regression model, t_i is a single decision tree regression model, K is the number of regression trees.

Gradient boosting

Opposed to a random forest, whose final result is the mode or the average of the results obtained by the trees in the ensemble, where the trees are independent of each other, in the case of Gradient Boosting [38] it works sequentially where each new tree included in the model depends on previous trees. Random Forest makes a multitude of trees that try to capture uncorrelated features, while each tree learner's in Gradient Boosting depends on the output of the previous tree [14].

$$\hat{F}(x) = \sum_{i=1}^{K} \gamma_i h_i(x) + c \tag{2.9}$$

Where γ_i can be thought of as the learning rate, the c value is there to initialise the model and $h_i(x)$ is the decision trees we are trying to fit over our residuals.

$$h_i(x) = y - F_i(x) \tag{2.10}$$

Thus in the gradient boosting the final model is built as a series of subsequent models where each model is trying to fit over the residual values calculated by it's previous models. Compared to a random forest, being a boosting based method, gradient boosting is more successfully applied to models with high bias and low variance [14].

Extreme gradient boosting - XGBoost

Extreme Gradient Boosting (XGBoost) [23], is an advanced supervised algorithm designed as an optimized implementation of the Gradient Boosting framework, XGBoost's loss function adds a smoothing term, which helps to smooth out the final learned weights to avoid overfitting. In

addition, XGBoost also supports row and column sampling to solve the overfitting problem. It also uses first and second-order gradient statistics to optimize the loss function. It also has low runtimes, as parallel and distributed computing ensures faster learning [110].

The estimated output of a tree ensemble model can be expressed as the sum of the prediction score of all trees:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{2.11}$$

Where K represents the number of trees used in the model, F the space of regression trees, f_k represents the (k-th tree) and x_i represents the features corresponding to sample i.

Loss functions are the most basic expression in ML problems, and the augmentation process continues until the objective function can no longer be minimized [**?** **?**].

$$\phi = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \tag{2.12}$$

Where l is a loss function, Ω is a term for penalizing the complexity of the model.

The algorithm minimizes ϕ by iteratively introducing each f_k. Assuming that a ensemble currently contains K trees. We add a new tree f_{K+1} that minimizes

$$\sum_i l(\hat{y}_i, y_i + f_{K+1}(x_i)) + \sum_k \Omega(f_k) \tag{2.13}$$

2.4 Causality analysis

We can define causality as a generic relationship between an effect and a cause. Often we look at causality intuitively. (e.g.: *We had a bad grade on the exam because we didn't study enough*, *We get a sore throat because we do not wear enough clothes*, etc...) However, when we deal with causality data a difference between statistical correlation and causation needs to be considered [53]. For example, when there is a rise in temperatures, a beach resort owner may observe a high number of ice-cream sales and a high number of drownings. A strong correlation between ice-cream sales and the number of drownings exists, however, the first one was not the cause of the second one. They are both caused by a third variable - temperature. People eat more ice-creams on hot days when they are also more likely to go swimming.

Current machine learning systems work, largely, in a statistical, or model-free mode, theoretically limiting the power and performance of the system. The ability to learn causality, following a model of reality, is considered a significant component of human-level intelligence and can serve as the foundation of Artificial intelligence (AI) [83].

2.4.1 Overview

If we want to understand the causal relationships in the data, different questions must be taken into account. Each task assuming a different (related) question that we want to answer: (1) Which variables could change the value of another variable? and (2) What is the impact of changing the value of a specific variable in a different one?

The first is a causal discovery problem and the second a causal inference problem [42].

For the causal discovery problem, researchers attempt to determining whether there exists a causal relationship between a variable and another. In the example of the beach resort, answering the question *is the temperature rise responsible for the increase of drownings?*

For causal inference problem, researchers investigate to what extent manipulating the value of a potential cause would influence a possible effect. Still in the same example, answering the question *how much temperatures rise drownings?*

Data for learning causality can take three classes. Observational data, where data arise from observing a system in a 'steady state' without any interventions. Interventional data, that comes from (randomized) intervention experiments. Finally, a mixture of the two can also be used to study causality [48] [42].

2.4.2 Time series causality

Two important properties present in causality are [33]:

- Temporal precedence. Causes precede their effects.
- Physical influence. Manipulation of the cause changes the effects.

In time series analysis, most approaches to causality make use of the first aspect of temporal precedence. Among these approaches, the definition introduced by Granger [41] is probably the most prominent and most widely used concept [33].

2.4.2.1 Granger causality

Following Granger [41] causality relationship occurs on two principles:

1. The effect does not precede its cause in time;
2. The causal series contains unique information about the series being caused that is not available otherwise.

Let A_t be a stationary stochastic process, $\overline{A_t}$ the set of past values $(A_{t-1}, A_{t-2}, ..., A_{t-\infty})$ and $\overline{\overline{A_t}}$ the set of past and present values $(A_t, A_{t-1}, ..., A_{t-\infty})$. Denote the optimum, unbiased, least-squares predictor of A_t using the set of values B_t by $P_t(A|B)$. Consequently, $P_t(X|\overline{X})$ will be the optimum predictor of X_t using only past of X_t series. Denote the predictor error as $\varepsilon_t(A|B) = A_t - P_t(A|B)$ and the variance of the error as $\sigma^2(\varepsilon_t(A|B))$. Finally, let U_t be all the information in the universe accumulated since time t-1 and let $U_t - Y_t$ be all this information apart from the series Y_t [41].

Then, we say that X_t is causing Y_t, denoted by $X_t \Rightarrow Y_t$, if we are better able to predict Y_t using all available information than if the information apart from X_t had been used.

$$\sigma^2(Y|\overline{U}) < \sigma^2(Y|\overline{U - X}) \tag{2.14}$$

Furthermore, we say that feedback is occurring, which is denoted $Y_t \Leftrightarrow X_t$, i.e., feedback is said to occur when X_t is causing Y_t and also Y_t is causing X_t.

$$\begin{aligned} \sigma^2(X|\overline{U}) < \sigma^2(X|\overline{U - Y}) \\ \sigma^2(Y|\overline{U}) < \sigma^2(Y|\overline{U - Y}) \end{aligned} \tag{2.15}$$

Finally, we say that Instantaneous causality is occurring, denoted by $X_t \Rightarrow Y_t$, if the current value of Y_t is better *predicted* if the present value of X_t is also included.

$$\sigma^2(Y|\overline{U}, \overline{\overline{X}}) < \sigma^2(Y|\overline{U}) \tag{2.16}$$

Chapter 3

Literature review

The ability to predict future events is of real value for companies, as this knowledge allows them to make better enterprise decisions and therefore have a greater market success [37].

In the specific case of the TV content distribution, this specialized knowledge it has been extracted through a small audience sampling strategy. However, this type of strategy is not nearly as effective. With the increase of available content (TV channels, programs and viewing options), the television audience is increasingly fragmented [58]. As a result, there were changes in the profile of the audience and in the viewing patterns. Although there has been a great deal of research on modeling individual and group preferences [21, 86], the impact of real-world events on user preferences is still a poorly understood topic.

This chapter contains literature review related to forecasting using external sources (Section 3.1) and learning causal relations from Big data (Section 3.2)

3.1 Forecasting using external sources

Forecasting is the process of predicting future values of some continuous time-series data. Generally, the forecast is produced by projecting the identified trend and seasonal cycles of the data (see section 2.2.3) into the future and discarding the irregular component [78].

With the rise of the big data era, a wide range of data have been generated in various domains. This new reality created several opportunities for gains through forecasting with Big Data. The forecasting of time series data provides organizations useful information that is necessary for making important decisions [72]. At present, there is increased research into using Big Data for obtaining accurate weather forecasts and the results are promissing [102]. The energy sector has also taken advantage of this new reality. As is the case of Selim et al. [97] where the authors explore the impact that including external features can have in terms of prediction accuracy, namely, in short-term energy forecasting. For this purpose, four computation models have been tested: Long Short-Term Memory neural networks (LSTM), Support Vector Regression

(SVR), Gradient Boosted Trees, and Facebook Prophet. In addition, the predictive accuracy was compared using the Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE). The results showed that multivariate algorithms using external features outperform univariate algorithms, and that multivariate algorithms achieve reasonable accuracy even without using past step energy consumption as an input feature.

On the other hand, the appearance of social media provides researchers with a new source of easily accessible data about individuals, society and, potentially, the world in general [94]. Such open source indicators have been shown to be effective at monitoring disease emergence and progression. In Mesquita et al. [74], a methodology was developed that discriminates search patterns related to general information (media drive) and search patterns related to infection (disease drive). With this information, it was possible to improve the results of disease outbreaks forecasting. In addition, a Granger causality test found out whether the number of cases and media preceded the search for disease-related terms.

Other studies go further and also study the individual importance that external features have on forecasting, as in [81] where traffic accidents were detected using an XGBoost machine learning model and SHAP (SHapley Additive exPlanation) a game-theoretical approach to explain the results and analyze the importance of the individual features.

Finally, sports organizations have recently realized the science value available in external sources [63]. Predicting results of sports matches is one of the fields that has been using external data to improve the accuracy of its models.

3.1.1 Using external features to forecast TV viewership

Over time, TV ratings and viewership forecast accuracy have decreased due to the fragmentation of consumer behavior [77]. Understanding the behavior of TV viewership is essential for an accurately targeted recommendation of TV content and, in general, has a decision support system. A new area of research has emerged that improves forecasting accuracy combining external features to the EPG metadata [75]. For example in Nixon et al. [80] the authors improve the accuracy of the audience forecasting by (1) adding content categories extracted from the EPG metadata as new features and (2) collecting and adding to the learning model specific event occurrences (e.g. finale of a popular program) that link to audience outliers. The results of this study shown that content-based features have a greater impact on prediction than event-based features. This study manually associates the external events to TV programming in the EPG, resulting in only 5% of links between events and EPG metadata, meaning event features have less impact on the evaluating results. Compared to our solution, we use an automatic method that has a higher connection rate (we cover this topic in detail in section 4.2).

A slightly different approach was taken in Khryashchev et al. [59], where the authors selected 5 predictors to forecast aggregate TV viewership base on viewing behavior prior to the broadcast, and showed that combining multiple models through an ensemble method is the most accurate

method to predict tv viewership. An 11% improvement was achieved over the baseline model.

Although some studies use external real-world features to improve the accuracy of sports attendance [82] [76] prediction and ultimately study the impact these features have on viewing patterns. Few studies address this problem in the specific case of sports live TV broadcasting.

Our research adds to the literature on improving the accuracy of sports TV viewership forecasts. More specifically, it investigates the value of ML techniques and the inclusion of external factors to forecasting football live TV broadcasting.

3.2 Learning causal relations from Big Data

With the advance of science and technology, the developments on causality analysis have also been overwhelmingly influenced by the age of Big Data [45]. The availability of multiple heterogeneous datasets presents new opportunities to big data analytics because the knowledge that can be acquired from multiple data sources would not be possible from any individual source alone [10].

The ability to learn causality is considered a significant component of human-level intelligence can serve as a fundamental component for AI [42]. However, nowadays, a lot of learning machines are improved by optimizing parameters over a stream of observational inputs received from the environment. These systems cannot reason about interventions and retrospection and, therefore, cannot serve as the basis for strong AI [83].

Time series analysis, especially the Granger causality test [41], has gained increasing interest in the last few decades. It has been applied in a range of studies in which the goal is to discover the relationships between different variables, with special emphasis on economics [57]. These econometric methods have also been adapted to a large range of different areas as energy [24], environmental health [111], or companies decision support systems like in Lim and Tucker [68] where the authors propose a framework that identifies influential term groups having causal relationships with real-world enterprise outcomes from Twitter data. Besides that, appropriate time lags between the influential term groups and the enterprise outcome are also identified. To achieve this they exploited a co-occurrence network analysis model to discover influential term groups, two time series models and a Granger causality analysis model.

3.2.1 Causality analysis in sports

Causal relationships in team sports such as football, basketball or baseball have also attracted increasing interest in the last couple of decades, with a large number of studies developed [47] [54] [31]. Understand the determinants of demand for professional sports is an important research topic for a variety of stakeholders [15]. For example, in Karanfil [57] a study was conducted on the causal relationship that rivalry between two teams has on their performance. In other words, if there is a rivalry, the performance of a club maybe be expected to affect that of its rival.

For this purpose the rivals of the most competitive football competitions were extracted, and a granger causality framework was developed. The study show that only a small percentage of the competitions (11 out of 23) can be qualified as performance-based rivalries. Other example is Hsu et al. [52], where a statistical analysis found out the impact of competitive pressure on the performance of a kicker (a key player in the NFL) in decisive moments of a match. Particularly, this study explores the situational effect in natural-field setting contexts on pressure kicks in the NFL 2000–2017. Natural-field conditions such as temperature, wind speed, field environment, the pressure faced by the players, and offensive and defensive strategies at crucial moments of a match are studied. The results showed that psychological/situational variables could play a more important role in pressure kicks. Other example, in Lago-Peñas et al. [61], examine the relationship between the national teams' ELO rating (originally developed for rating chess players) and the number of migrating players in the "big-five" leagues. Finally, in Hall et al. [43] was explored the relationship that payroll has on performance, or vice versa, and it has been shown that the hypothesis that higher payrolls granger causes better performance cannot be rejected.

3.2.2 Causality in televised sports content

Sports broadcasts is known to have the largest share of TV audiences [35] and given the increasingly fragmented scenario of TV audiences, it has generated a growing interest in understanding the determinants of television demand. However, despite the importance of sportscasts, few studies investigate the factors of indirect demand for sports events, measured by TV viewership. The literature on television audience demand is still relatively underdeveloped compared to the literature analyzing live attendance [89] [6] [29]. Despite that, two studies stood out when compared to our research problem.

3.2.2.1 Determinants of demand for televised live football: features of german national football team

This study, presented by Feddersen and Rott [35], analyzes all the TV broadcasts of the German national football team from January 1993 to June 2008. The analysis is based on television ratings generated by the Growth from Knowledge (GfK). This data source estimates viewership from a representative panel of 5,640 households that contain approximately 13,000 people. Nonsporting determinants of viewing like weather conditions (temperature, precipitation, etc.), the broadcasting network, and student holidays were used. This study tries to build a bridge between determinants of demand from the sports economic perspective and determinants of demand from a classical critical success factor analysis, and from a media economics perspective. A regression analysis was applied, and with this, it has concluded that the demand for a sportscast depends mostly on the sporting competition of the match and its relevance within the context of the tournament. Moreover, viewers prefer a national team with more experienced players and matches with an opponent of high quality with a greater reputation. This study also showed

that some sport-unrelated factors have explanatory powers, such as kickoff time and weather conditions.

3.2.2.2 Determinants of football TV audience: the straight and ancillary affects of the presence of the local team on the FIFA world cup

In Uribe et al. [101] four FIFA World Cup competitions (2002, 2006, 2010, 2014) have been used to evaluate the determinants of their TV audience size. Through multiple regression analyses, the study tested the explanatory power of independent variables to predict TV audience size. The independent variables incorporated were:

- Home team effect (presence of the national team);
- Outcome uncertainty (competitive balance or symmetry among teams and fans' interest);
- Match quality (teams with higher reputation);
- Team familiarity:
  - The number of years that the team has been in competition;
  - The presence of a team that is geographically close;
- Scheduling (day and scheduling of the match).

The results showed that when the national team qualifies for the tournament, the home team effect is the most relevant predictor of audience size, followed by match quality and scheduling variables.

From these two studies we can concluded that the main determinants of sports TV viewership are elements associated with the attractiveness of the match, namely: the match quality and importance, the outcome uncertainty, and audience identification with a team. Besides that, sports indirect factors (e.g. weather, news or social media trends), which is the main focus of our study, showed to also have explanatory power on the number of television audiences.

The present study builds on, and fills some gaps in, the literature by (1) extracting data of traditional sports related explanatory variables – outcome uncertainty, match quality and match importance – and examining indirect factors related to football matches – news generated by a football match, twitter teams' popularity and meteorological factors –, (2) improving TV viewership prediction accuracy using real-world data and (3) using a causality analysis to identify influential real-world events on the volume and view patterns of the television clients. The pipeline of the proposed solution is presented in figure 3.1.

Although many existing expert and intelligent systems for determinants of sports TV viewership enable computers to analyze the correlation between real-world events and TV viewership, limited contributions have been made to analyze causal effects of real-world events

on TV viewership. Moreover, to the best of our knowledge, no studies have explored the causal relationships between real-world data and TV viewership in such a wide range of external events.

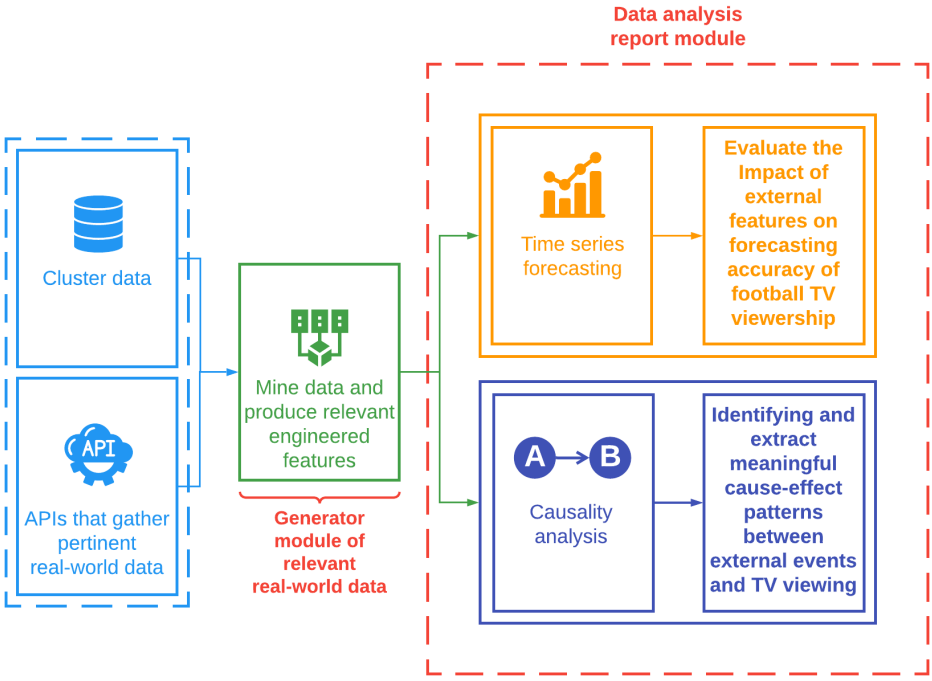


Figure 3.1: Proposed solution.

Chapter 4

Data description

For this work, one of the main goals is to perform a review on various web scraping APIs that gather pertinent data about real-world events. This *external* data, together with the measurement of television audience data provided by a telecommunication company, would allow us to study the impact of the external events on both the volume of clients and the viewing time of football live TV content.

Five types of data are employed in this work. First, the TV demand for football games is studied through the number of viewers and total viewing time of each game; second, the characteristics of the competition are explored through the outcome uncertainty and the match quality; third, the interest generated by a game is measured via the game's news count and google search popularity; fourth, the team's popularity is measured by the number of followers on one of the most popular social networks today; five, weather factors are taken into account, through several atmospheric quantities (e.g., precipitation, temperature, etc...).

Pooling data from such a wide range of external sources and joining them with the TV demand for football games leads to an exclusive data set, which is a key part and a major strength of this research.

The remainder of this chapter introduces the multidimensional dataset used for the forecasting and causality analysis and examines them by doing an exploratory data analysis for preliminary data understanding. It ends with a discussion about the challenges faced throughout the development, how they were overcome, and the main limitations of the built multidimensional dataset.

4.1 Data sources

The most valuable data for the goals and tasks of our project is in the form of timestamps and football match details. Taking this into account and inspired by prior theoretical constructs we collect external factors that could potentially affect football live TV broadcasts from four broad

categories, namely:

1. **The characteristics of football competition.** Following the discoveries in [85], [101], [35], and [4] where it was shown that factors inherent to competition (e.g., outcome uncertainty and match quality) can affect viewing patterns, we use data from a free football betting portal (*Football-Data*) providing historical results and odds. Although *Football-Data* portal has games for the main European leagues, other types of tournaments such as cups tournaments are not present. In order to obtain these missing tournaments, we use *Sports DB* data source, an open crowd-sourced database of sports artwork and metadata with a free API built by its users. It has historical and future data from several national/international tournaments. (e.g., Champions league, Europa league, etc...). Finally, we use ESPN's Soccer Power Index (SPI), an international and club rating system designed to be the best possible representation of a team's current overall skill level. It as rating data back to 1888 (from more than 550,000 different matches);

2. **The interest generated by a match.** Inspired by the findings of Mesquita et al. [74] were has been shown that the use Google search trends can provide relevant information about mass behaviors, we use the same Google tool to find the popularity of match-related search terms. In addition, we also explore the impact of the number of news in the viewing patterns through *Público* newspapers (one of the most popular local newspapers);

3. **Teams' popularity.** To measure the impact that teams' popularity has on television patterns, we use the Twitter API to extract the number of followers for each team;

4. **Meteorological factors.** Weather factors are also believed to have an impact on viewing patterns [35] [11], so the telecommunications company provided weather data for 675 different geographic points. As television data is related to local audiences, weather data is collected from local weather stations.

To support the development of the thesis research, the telecommunication company made available a television database containing TV programs records, with information about the number of viewers and total viewing time. As this is proprietary data, all customer counts and the sum of second values will be normalized between 0 and 1. Channel names will also be anonymized by generic labels (e.g., Channel 1, Channel 2, etc...)

Table 4.1 shows a summary of all the raw data sources used.

Finally, mention that the data source used are available in different ways. The television audience data is available through Apache Hive; the Público newspaper, Google trends, and Twitter data are available through a JSON API; the other sources are available in CSV (Comma-Separated Values) files.

Table 4.1: Raw Data Sources

	Description	External Data	Data Type
Football-Data	**i)** Historical football results and odds.	✓	Characteristics of football competition
Sports DB	**i)** Historical and future data from various national/international tournaments.	✓	
Soccer SPI	**i)** International football teams' and matches rating system.	✓	
Público	**i)** "Público" newspaper API; **ii)** Allows you to search for a specific topic, defining a time interval.	✓	Interest generated by a match
Google Trends	**i)** Analyzes the popularity search queries in Google Search and allows the user to compare the volume of searches between different terms.	✓	
Twitter	**i)** Lets you read and write Twitter data such as: tweets, users, direct messages, lists, trends, media, locations. **ii)**	✓	Teams' popularity
Company	**i)** Average weather forecasts for 675 geographical points; **ii)** This is a weather forecast with the generation of values by timestamp (every 3 hours): the forecast values are generated at 0h on D day for 8 timestamps on D + 1.	✓	Meteorological factors
Company	**i)** EPG metadata with total viewing time and volume of clients.		Television Audience

4.2 Data aggregation

This section presents the process that takes the raw data from the data sources and converts it to a more friendly format, ready to be used by forecasting and causality methods. We will start by detailing the process of extracting data from each external data source, then we will present the methodology used to merge the external data with television demand data.

4.2.1 Data sources extraction

Before we can do any kind of data processing or storage, we need to extract the data from the corresponding data sources. We start our extraction by gathering the television audience data as this would be a baseline to merge with the remaining datasets.

4.2.1.1 Television audience

The telecommunications company made the television data available through an Hadoop cluster. We use a Spark SQL script (also provided by the telecommunication company) to extract the television audience dataset (Listing 4.1).

```
spark_connection = spark_session()

query = '''
    SELECT
        *
    FROM cluster_table.television_audiences
    '''

sports_timeseries = spark_connection.sql(query)
sports_timeseries = sports_timeseries.toPandas()
```

Listing 4.1: Code snippet of the spark function used that establishes a connection to the cluster.

The dataset used comprises television programs from 40 different channels between March 2019 and March 2021. During this period, we obtained TV session data from around 1,000,000 distinct programs with information such as the start/end time of the program, the title and the volume and the total viewing time of the clients. Table 4.2 summarizes all features present in this dataset.

4.2.1.2 Characteristics of football competition

To study the impact of external events on television data, the selected football tournaments are of great importance, as tournaments with little representation in television data will not

Table 4.2: Description of the television audience dataset features.

Feature	Description	DType	Non-Null Count
ChannelName	TV channel name	object	1072435
Title	Program name	object	1072435
StartTime	Program start time	object	1072435
EndTime	Program end Time	object	1072435
Category	Category	object	1072435
Genre	Genre	object	1072435
TotalViewTime	Normalized total viewing time	int64	1072435
ClientsVolume	Normalized number of clients	int64	1072435

be the best indicator of the impact of external events on TV audiences. Thus, we decided to extract football matches from the most competitive local football tournaments: Liga NOS, LEDMAN LigaPro, Taça de Portugal and Taça da Liga. As well as football matches from the most competitive international football tournaments [57]: La Liga, Premier League, Bundesliga, Champions League, Serie A, Ligue 1, Europa League and International Champions Cup.

We start by extracting the characteristics of football competition data from the *Football-data* data source. This data source includes historical data from the major European leagues with information such as match results, match statistics, and match odds from several different bookmakers. The following following football tournaments were extracted:

- Liga NOS
- La Liga;
- Premier League;
- Bundesliga;
- Serie A;
- Ligue 1;

To extract the remaining data from the other football tournaments we use the *Sport DB* data source, this data source only has information about the result of the game. The following football tournaments were extracted:

- LEDMAN LigaPro
- Taça de Portugal
- Taça da Liga
- Champions League
- Serie A

- Ligue 1
- Europa League
- International Champions Cup

The data from each competition were extracted for three different seasons (2018/2019, 2019/2020 and 2020/2021), which comprises games in the television data period (from March 2019 to March 2021). Each football tournament for a given season has its own CSV file.

After extracting the different football tournaments datasets, we decided to concatenate them all. Furthermore, for a better understanding of the data, they were divided into three different datasests: one with the all the match results (table 4.3), another with all the match stats (table 4.6) and another with the all the match odds (table 4.5).

Table 4.3: Description of the results dataset features.

Feature	**Description**	**DType**	**Non-Null Count**
Date	Game day	object	8296
HomeTeam	Home team name	object	8296
AwayTeam	Away team name	object	8296
FTHG	Full time home team goals	float64	8169
FTAG	Full time away team goals	float64	8169
Div	Competition name	object	8296
Time	Game hour	object	3896
Round	Game round	object	2268

Table 4.4: Description of the match stats dataset features.

Feature	**Description**	**DType**	**Non-Null Count**
Date	Game day	object	6028
HomeTeam	Home team name	object	6028
AwayTeam	Away team name	object	6028
HS	Home team shots	int64	6028
AS	Away team shots	int64	6028
HST	Home team shots on target	int64	6028
AST	Away team shots on target	int64	6028
HC	Home team corners	int64	6028
AC	Home team corners	int64	6028
HF	Home team fouls Committed	int64	6028
AF	Away team fouls Committed	int64	6028
HY	Home team yellow Cards	int64	6028
AY	Away team yellow Cards	int64	6028
HR	Home team red Cards	int64	6028
AR	Away team red Cards	int64	6028
Referee	Match referee	object	1088

Table 4.5: Description of the odds dataset features.

Feature	Description	DType	Non-Null Count
Date	Game day	object	6028
HomeTeam	Home team name	object	6028
AwayTeam	Away team name	object	6028
BWH	Bet&Win home win odds	float64	6025
BWD	Bet&Win draw odds	float64	6025
BWA	Bet&Win away win odds	float64	6025
WHH	William Hill home win odds	float64	6024
WHD	William Hill draw odds	float64	6024
WHA	William Hill away win odds	float64	6024
VCH	VC Bet home win odds	float64	6024
VCD	VC Bet draw odds	float64	6024
VCA	VC Bet away win odds	float64	6024
IWH	Interwetten home win odds	float64	6024
IWD	Interwetten draw odds	float64	6024
IWA	Interwetten away win odds	float64	6024
B365H	Bet365 home win odds	float64	6024
B365D	Bet365 draw odds	float64	6024
B365A	Bet365 away win odds	float64	6024
PSH	PH = Pinnacle home win odds	float64	6010
PSD	PD = Pinnacle draw odds	float64	6010
PSA	PA = Pinnacle away win odds	float64	6010

Following BORLAND and MACDONALD [15] we decided that it was important to have the ranking of football matches. For each league to establish the classification of the clubs in each journey, the following rules for comparing teams are applied:

1. Highest number of points in the entire competition;
2. Greater difference between the number of goals scored and the number of goals conceded by clubs in matches played throughout the competition;
3. Highest number of victories in the entire competition;
4. Highest number of goals scored in the entire competition.

To determine the ranking in each league we use the individual competition datasets from *Football-Data* and *Sports DB*. In the case of knockout competitions, the ranking values are those of previous league matches. If the previous match is not related to a league competition, then the ranking value in that game is set to null. Table 4.6 summarizes the resulting features from this generated ranking dataset. Note that ranking features are pre-match features to prevent data leakage (i.e., predicting the output, using a feature that at the time of prediction cannot be available).

Table 4.6: Description of the ranking dataset features.

Feature	Description	DType	Non-Null Count
Date	Game Day	object	7388
HomeTeam	Home Team Name	object	7388
AwayTeam	Away Team Name	object	7388
PointsH	Home Team Points	float64	7123
PointsA	Away Team Points	float64	7132
GoalsForH	Goals For Home Team	float64	7123
GoalsForA	Goals For Away Team	float64	7132
GoalsAH	Goals Against Home Team	float64	7123
GoalsAA	Goals Against Away Team	float64	7132
GoalDiffH	Goal Difference For Home Team	float64	7123
GoalDiffA	Goal Difference For Away Team	float64	7132
WinsH	Home Team Wins	float64	7123
WinsA	Away Team Wins	float64	7132
DrawsH	Home Team Draws	float64	7123
DrawsA	Away Team Draws	float64	7132
LossesH	Home Team Losses	float64	7123
LossesA	Away Team Losses	float64	7132
RankH	Home Team Ranking	float64	7123
RankA	Away Team Ranking	float64	7132

Data collection referring to the characteristics of football competition is completed with data extraction of the teams' overall skill (before a match) for the selected matches.

For this purpose, we use a soccer system based on ESPN Soccer Power Index (SPI). In this system, every team has an offensive and defensive rating that expresses the number of goals it would be expected to score and concede, respectively, against an average team on a neutral field. The SPI rating is the combination of the two values (offensive and defensive rating) and represents the percentage of points a team would expect to take if it always played against the average team. In addition to the SPI rating, this data source also provides other projections, such as the result probabilities or the importance of a game for a team (e.g., whether it will be decisive to win the league or not). These two projections, as in the case of SPI rating, range between 0 and 100. Table 4.7 presents the features used from the *Soccer SPI* data source.

4.2.1.3 Interest generated by a match

Finding the popularity of search terms related to a football match can provide insight into the interest generated by a match. In this sense we collect the number of news related to a football match and also the popularity of Google search terms related to the football matches and the individual teams.

Thus, for each match, the number of news from 5 days before the start of the game until a day before was collected from a local newspaper (*Público*). A description of the dataset features is shown in Table 4.8. We use the Público JSON:API to get the news count. This returns all

Table 4.7: Description of the Soccer SPI dataset features.

Feature	Description	DType	Non-Null Count
Season	Season year	int64	28936
Date	Match day	object	28936
League	League name	object	28936
Team1	Home team name	object	28936
Team2	Away team name	object	28936
Spi1	Soccer Power Index - Home team overall strength	float64	28936
Spi2	Soccer Power Index - Away team overall strength	float64	28936
Prob1	Home team win probability	float64	28936
Prob2	Away team win probability	float64	28936
Probtie	Draw probability	float64	28936
ProjScore1	Home team goals projection	float64	28936
ProjScore2	Away team goals projection	float64	28936
Importance1	Home team match importance	float64	28936
Importance2	Away team match importance	float64	28936

news between the start and end time specified for the search query. Note that only 10 news items at most are returned, however this number is more than enough for the time period used.

Table 4.8: Description of the counted news dataset features.

Feature	Description	DType	Non-Null Count
Date	Match day	object	3642
HomeTeam	Home team name	object	3642
AwayTeam	Away team name	object	3642
CountedNews	Counted news before the match	int64	3642

In addition, the google search terms popularity referring to a match are also collected (Table 4.9). For this purpose, we use *Pytrends*, an Unofficial JSON API for Google Trends, which allows the download of Google Trends reports. Data was extracted only from Portugal. It provides a normalized number (between 0 and 100) indicating the search popularity of the term.

Table 4.9: Description of the Google trends dataset features.

Feature	Description	DType	Non-Null Count
StartTime	Start Time of the Match	object	3074
HomeTeam	Home team name	object	3074
AwayTeam	Away team name	object	3074
WeekInterest	Match week popularity	float64	3074
WeekInterestVs	Match week popularity	float64	3074
DayInterest	Match day popularity	float64	3074
DayInterestVs	Match day popularity	float64	3074
HourInterest	Match hour popularity	float64	2982
HourInterestVs	Match hour popularity	float64	2982

Finally, as the data extracted in Table 4.9 measures interest in the game as a whole, we decided to also extract Google search trends for each individual team.

Table 4.10: Description of the Google Trends teams dataset features.

Feature	Description	DType	Non-Null Count
StartTime	Start time of a match	object	3589
HomeTeam	Home team name	object	3589
AwayTeam	Away team name	object	3589
Lag0InterestHome	Home team popularity (Match Day)	float64	3516
Lag0InterestAway	Away team popularity (Match Day)	float64	3516
Lag1InterestHome	Home team popularity (1 Day Before)	float64	3516
Lag1InterestAway	Away team popularity (1 Day Before)	float64	3516
Lag2InterestHome	Home team popularity (2 Days Before)	float64	3516
Lag2InterestAway	Away team popularity (2 Days Before)	float64	3516
Lag3InterestHome	Home team popularity (3 Days Before)	float64	3516
Lag3InterestAway	Away team popularity (3 Days Before)	float64	3516

4.2.1.4 Teams' popularity

The popularity of a team, namely, the number of supporters can also be one of the impacting factors in the TV numbers. As we do not have access to the exact number of supporters for each team, we use the number of followers on twitter in order to simulate the popularity of each team. Twitter provides a search application program interface (API) for extracting accounts related to some search keyword. To obtain the data required to our study, we required the Twitter search API the accounts for all the teams' presented in our dataset. In order to avoid secondary accounts only verified Twitter accounts are considered. Table 4.11 shows the resulting dataset features. Team location information is also collected along with each team's followers count.

Table 4.11: Description of the Twitter dataset Features.

Feature	Description	DType	Non-Null Count
Team	Football team name	object	427
FollowersCount	Number of followers	float64	282
Location	Location of the team	object	282

4.2.1.5 Meteorological factors

As described in the section 3, meteorology is known to have an impact on TV patterns. So, in order to test the effect in our specific case of live broadcasts of football matches, weather forecasts from 675 different geographical points were aggregate by the average, resulting in more than 6000 weather forecasts (every 3 hours from January 2019 to January 2021). A summary description of the features is presented in Table 4.12.

Table 4.12: Description of the Weather dataset features.

Feature	Description	DType	Non-Null Count
Predictiondate	Weather forecast date	object	6000
WindSpeed	Atmospheric quantity	float64	6000
WindDirection	Atmospheric quantity	float64	6000
Temp	Atmospheric quantity	float64	6000
AirDensity	Atmospheric quantity	float64	6000
Pressure	Atmospheric quantity	float64	6000
Radiation	Atmospheric quantity	float64	6000
HR	Atmospheric quantity	float64	6000
Precipitation	Atmospheric quantity	float64	6000

Table 4.13 gives an overview of the resulting collected datasets detailing the datasets provided as input and the number of rows and columns resulting from the extraction.

Note that although we have introduced the features using camel case naming convention (to promote the understanding of the data), from now on we will refer to the features using only the snake case naming convention (e.g., *home_team*, *away_team*).

Table 4.13: Structured Data Sources

Dataset Name	Description	Input Dataset(s)	Shape (Rows, Columns)
results	Match results (e.g. goals, full time result, etc...)	Football-Data, The Sports DB	(8296, 10)
match_stats	Stats about the game (e.g. red/yellow cards)	Football-Data	(6028, 16)
odds	Football odds	Football-Data	(6028, 62)
rank	League ranking before a certain game for the home/away team	results	(7388, 21)
spi	Contains match-by-match SPI ratings (e.g. importance of a game)	Soccer SPI	(28936, 15)
counted_news	The number of news in the "Público" newspaper, related to a particular game, from 5 days before the game until the day before a match	results, Público	(3642, 4)
google_trends	Interest generated by a game in the week, day and hour before starting.	results, Google Trends	(3074, 9)
google_trends_teams	Interest generated by the teams' of a given game in each of the last four days.	results, Google Trends	(3589, 14)
twitter	Number of twitter followers for a given team.	Twitter	(427, 3)
meteo_mean	Average weather forecasts for 675 geographical points.	Weather Company	(6000, 9)
sports_timeseries	Program names and information about those programs.(watching time, volume of clients)	Telecommunication company	(1072435, 12)

4.2.2 Final dataset construction

After completing the selection of pertinent APIs and extract the raw data from the external sources, we begin piecing together the multiple datasets collected under heterogeneous conditions.

To merge external data and EPG data, some unique match features have to be taken into account, namely: the timestamp of the match, the home team name, and the away team name.

In cases where these features are not explicitly available, it is necessary to extract them from other features or sources. This is the case of television audience data, where the information regarding the names of the teams is contained in the title of the program.

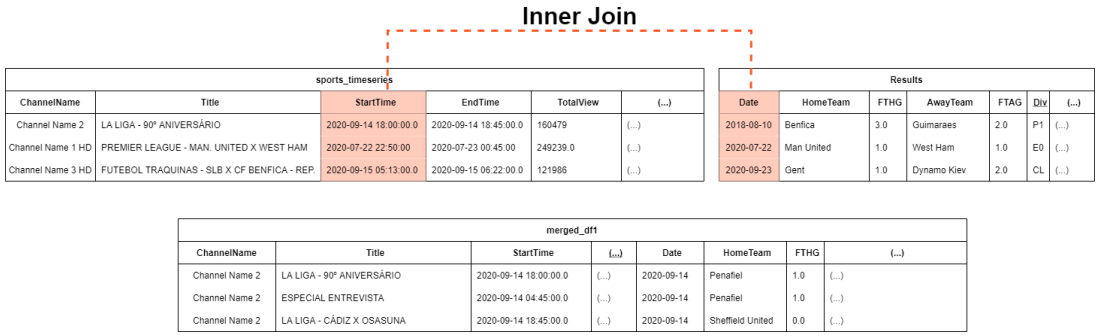


Figure 4.1: Merge datasets using date as key.

After we have explored the structure and organization of the platforms, our findings support that we should guide our implementation of the data fusion by two major tasks identified:

1. Merge the television audience data with the football matches information;
2. Merge the remaining external datasets.

In this chapter, we will only cover the data fusion of the results dataset with the EPG dataset. Once we have an association between the program title and the game identifier, the merge of the remaining tables becomes trivial.

4.2.2.1 Results and television audience

To merge the television audience data with the football matches, the following tasks were outlined:

1. Merge datasets taking into account the football match date;
2. Drop rows with no association between the program title and the tournament;
3. Drop rows with no association between the program title and the teams' names;
4. Filter live content related to football matches.

The data fusion starts by joining football matches and programs that took place on the same day, for that purpose an inner join was applied. Figure 4.1 shows the final result (table bellow) of the join of the Results with the Television Audience dataset (the two tables at the top).

After an analysis of the titles referring to football game programs, we verified that the competition appeared in all cases as a prefix of the program title. With this information, we created a dictionary that matches the competition at the beginning of the program title and the competition present in the Div feature (Listing 4.2). Figure 4.2 shows the final result of this filtering.

```
{'F1': ('LIGUE 1', 'LIGUE 1 2020', 'LIGA FRANCESA'),
 'CL': ('LIGA DOS CAMPEOES', 'CHAMPIONS LEAGUE', 'UEFA CHAMPIONS LEAGUE'),
 'EL': ('UEFA EUROPA LEAGUE', 'LIGA EUROPA', 'TACA UEFA'),
 'P1': ('LIGA NOS', 'LIGA NOS 2017/18 ', 'LIGA NOS 2018/19', 'LIGA NOS
    2016/17'),
 'D1': 'BUNDESLIGA',
 'E0': 'PREMIER LEAGUE',
 'I1': ('LIGA ITALIANA', 'LIGA ITALIANA 2020'),
 'SP1': ('LA LIGA', 'LIGA ESPANHOLA'),
 'TP': 'TACA DE PORTUGAL',
 'P2': 'SEGUNDA LIGA',
 'IC': ('INTERNATIONAL CHAMPIONS CUP', 'JOGO DE PREPARACAO', 'JOGO PARTICULAR'),
 'TL': 'TA A DA LIGA'}
```

Listing 4.2: League dictionary.

merged_df

ChanneName	Title	(...)	Div	(...)
Channel Name 2	LA LIGA - CÁDIZ X OSASUNA	(...)	P1	(...)
Channel Name 2	LA LIGA - CÁDIZ X OSASUNA	(...)	SP1	(...)
Channel Name 2	LIVERPOOL TV MAGAZINE	(...)	P2	(...)

merged_df2

Channel Name	Title	(...)	Div	(...)
Channel Name 3	SEGUNDA LIGA - BENFICA B X VILAFRANQUENSE - REP.	(...)	P2	(...)
Channel Name 1	PREMIER LEAGUE - RESUMO DA 1ª JORNADA	(...)	E0	(...)
Channel Name 1	PREMIER LEAGUE - RESUMO DA 1ª JORNADA	(...)	E0	(...)

Figure 4.2: Competition filtering.

After cleaning out football matches that didn't meet the competition in the title of EPG metadata, we started filtering the teams' names. First, we found that most program names that refer to live football broadcasts on TV have the following pattern:

$$Tournament - HomeTeam\ X\ AwayTeam - Moreinfo$$

Based on this knowledge, we use 5 different regex to extract each component of the program title (Listing 4.3). Final result is presented in figure 4.3.

```
television_df['comp_str'] =
    television_df['Title'].str.extract(r'^(?:[^-]*\-){0}([^-(]*)')
television_df['match_str'] =
    television_df['Title'].str.extract(r'^(?:[^-]*\-){1}([^-(]*)')
television_df['home_team_str'] =
    television_df['Title'].str.extract(r'^(?:[^-]*\-){1}([^-X]*)')
television_df['away_team_str'] =
    television_df['Title'].str.extract(r'^(?:[^X]*X){1}([^-(]*)')
television_df['more_info_str'] =
    television_df['Title'].str.extract(r'^(?:[^-]*\-){2}([^-(]*)')
```

Listing 4.3: Title extraction regex

merged_df3_regex

ChannelName	(...)	Title	comp_str	match_str	home_team_str	away_team_str	more_info_str
Channel Name 3	(...)	FUTEBOL FEMININO - ESTORIL X SLB - REP.	FUTEBOL FEMININO	ESTORIL X SLB	ESTORIL	SLB	REP.
Channel Name 2	(...)	LA LIGA - CÁDIZ X OSASUNA	LA LIGA	CÁDIZ X OSASUNA	CÁDIZ	OSASUNA	NaN
Channel Name 4	(...)	LIVERPOOL TV MAGAZINE	LIVERPOOL TV MAGAZINE	NaN	NaN	NaN	NaN

Figure 4.3: Extracting relevant labels from the program title (Regex).

After extracting the names of the teams from the EPG metadata, it was necessary to compare the *match_str* column in figure 4.3 with the game identifier columns. To this end, we start by creating a new column, with an identically format to *match_str* feature:

$$HomeTeam\ X\ AwayTeam$$

In figure 4.4 we can see the new *match* column created using the *home_team* and *away_team* features.

merged_df3_regex_match

ChannelName	(...)	HomeTeam	AwayTeam	Match
Channel Name 3	(...)	Penafiel	Covilha	PENAFIEL X COVILHA
Channel Name 1	(...)	Brighton	Chelsea	BRIGHTON X CHELSEA
Channel Name 1	(...)	Sheffield United	Wolves	SHEFFIELD UNITED X WOLVES

Figure 4.4: New feature *Match* with the name of the match in a identical format to the program title.

Once these two identical columns were created, a function was applied to measure the distance between the two string (Figure 4.5). For this purpose, SequenceMatcher function from the *difflib* package was used. This is based on the gestalt pattern matching algorithm, presented by Ratcliff and Metzener [88]:

$$D_{ro} = \frac{2K_m}{|S_1| + |S_2|} \tag{4.1}$$

where S_1 and S_2 are the strings we want to compute the similarity and K_m the number of matching characters. The similarity of the two string can assume values between 0 (no match of any letter) and 1 (a complete match).

merged_df3_regex_match_similarity

ChannelName	(...)	match_str	match	Similarity
Channel Name 3	(...)	BENFICA B X VILAFRANQUENSE	PENAFIEL X COVILHA	0.478261
Channel Name 1	(...)	SHEFFIELD UNITED X WOLVERHAMPTON	SHEFFIELD UNITED X WOLVES	0.813559
Channel Name 1	(...)	BRIGHTON & HOVE ALBION X CHELSEA	BRIGHTON X CHELSEA	0.692308

Figure 4.5: Similarity function applied to the string extracted from the title and the string created based on the teams' names features.

After measuring the similarity between the two match columns, for each unique program, we selected the highest similarity value.

Finally, a threshold is applied and small similarity values are removed (Figure 4.6).

merged_df3_regex_match_similarity

ChannelName	(...)	match_str	match	Similarity
Channel Name 3	(...)	BENFICA B X VILAFRANQUENSE	PENAFIEL X COVILHA	0.478261
Channel Name 1	(...)	SHEFFIELD UNITED X WOLVERHAMPTON	SHEFFIELD UNITED X WOLVES	0.813559
Channel Name 1	(...)	BRIGHTON & HOVE ALBION X CHELSEA	BRIGHTON X CHELSEA	0.692308

merged_df3_threshold

Channel Name	(...)	match_str	match	Similarity
Channel Name 1	(...)	SHEFFIELD UNITED X WOLVERHAMPTON	SHEFFIELD UNITED X WOLVES	0.813559
Channel Name 1	(...)	BRIGHTON & HOVE ALBION X CHELSEA	BRIGHTON X CHELSEA	0.692308
Channel Name 1 HD	(...)	SHEFFIELD UNITED X WOLVERHAMPTON	SHEFFIELD UNITED X WOLVES	0.813559

Figure 4.6: Rows with low similarity are removed

As a game may be broadcast live and deferred on the same day, we use a regex with specific channel information to filter live football matches (listing 4.4). The result is in figure (figure 4.7)

```
channel_1_live =
    channel_1[channel_1['program_title_dsc'].str.contains("DIRETO")]
channel_2_live =
    channel_2[channel_2['program_title_dsc'].str.contains("(DIRETO)")]
```

Listing 4.4: Live programs extraction regex

merged_df3_threshold				
ChannelName	**(...)**	**Start time**	**Title**	**more_info_str**
Channel Name 1	(...)	2019-11-29 20:30:00	LIGA NOS - SANTA CLARA X BOAVISTA (DIRETO)	NaN
Channel Name 1	(...)	2019-11-29 22:30:00	LIGA NOS - SANTA CLARA X BOAVISTA	NaN
Channel Name 2	(...)	2019-11-30 15:30:00	LIGA NOS - MOREIRENSE X DESP. AVES (DIRETO)	NaN

final_merged_df				
ChannelName	**(...)**	**Start time**	**Title**	**more_info_str**
Channel Name 1	(...)	2019-11-29 20:30:00	LIGA NOS - SANTA CLARA X BOAVISTA (DIRETO)	NaN
Channel Name 2	(...)	2019-11-30 15:30:00	LIGA NOS - MOREIRENSE X DESP. AVES (DIRETO)	NaN
Channel Name 1	(...)	2019-11-30 20:30:00	LIGA NOS - PORTIMONENSE X FAMALICÃO (DIRETO)	NaN

Figure 4.7: Deferred programs are removed

After having a connection between the programs and the football matches completed, the union of the other tables was more or less straightforward using the identifiers: timestamp, home team name, and away team name.

4.3 Exploratory data analysis

To understand patterns that can provide valuable information to test hypotheses and to check assumptions we apply an exploratory data analysis (EDA). This chapter describes the methods used for this purpose as well as the main results of the analysis.

4.3.1 Number of football matches used

We start our exploratory data analysis by doing a quantitative analysis of the framework used to merge the different tables.

In the figure 4.8 we have the number of rows in each of the data sources before the data fusion (horizontal bar chart on the left) and after the data fusion (horizontal bar char on the right). Here the reference point is the results table (orange horizontal bar), where all available football matches are. We can see that except for the *spi*, *odds*, *match_stats* (where only data for some tournaments are available) and *google_trends_teams*, *google_trends*, *twitter* (where rate limit and API specifications made it impossible to search for some matches) all the other data sources had a match utilization rate close to 100%.

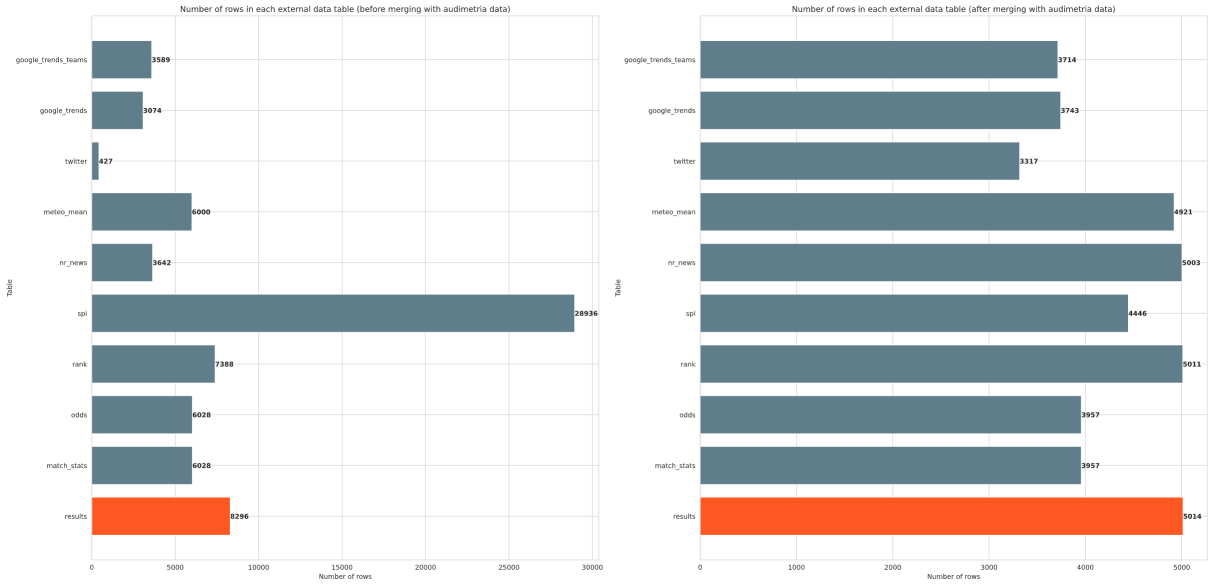


Figure 4.8: Total number of matches in each external dataset before the data fusion (left-hand side) and after the data fusion (right-hand side). Note that the number of rows on the right have duplicates (i.e., games broadcast in more than one channel) and are not directly comparable with the number of rows on the left (no duplicates).

We also plot the number of matches used from the original results table. For this purpose, we use a pie chart with the percentages of used data (matches that merged with the EPG data) and unused data. The final result is in figure 4.9.

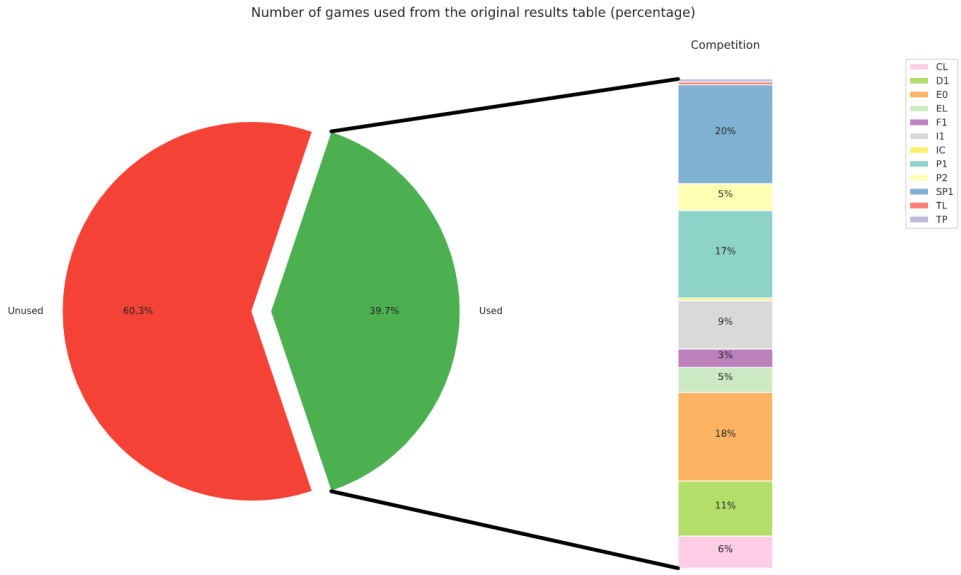


Figure 4.9: Number of games used from the original results table (percentage).

We can see through this visualization that 40% of the games were successfully joined to the EPG metadata. If we look even closer, we can see that 20% of these matches are related to *La*

Liga (SP1). This may be explained considering that it is one of the most prominent contents on *Sport Channel 2* (figure 4.10). After *La Liga*, we have the *Premier League (E0)* with the highest percentage (more precisely 18%) of matches in the data. Once again this makes sense based on the major content of this competition on *Sport Channel 1*. Next, we have the *Liga NOS (P1)* with 17% of the matches, which can be explained by a large amount of content both in the *Sport Channel 1* and *Sport Channel 3*. After that, we have *Bundesliga (D1)* with 11%, *Serie A (I1)* with 9%, and the remaining matches, except for the *International Cup (IC)*, *Taça da Liga (TL)* and *Taça de Portugal (TP)* that have an insignificant amount of game, are more or less distributed among the other competitions.

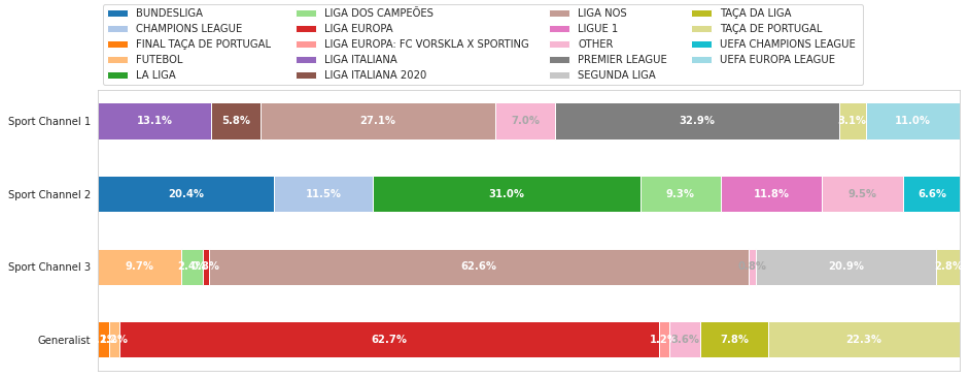


Figure 4.10: Number of rows for each competition (percentage).

Given the diversity of match competitions and the fact that we are fetching different competitions from different data sources, these are encouraging results. Even more, compared to other manual approaches in the literature, where only 5% of external data were successfully linked to EPG metadata [80]. The unused data can be explained by the following reasons:

- **Date out of range.** Games before March 2019 or after March 2021 are discarded because there is no television audience data for those periods;
- **Before similarity threshold.** Discarded games before the similarity threshold are a strong indication that they were not broadcast on television, as they did not even match a program on a given day;
- **After similarity threshold.** Games can be dropped after the similarity threshold if they do not occur in the television audience data or it can also be a mismatch;
- **Deferred television content.** Finally, it may happen that some matches are not broadcast live.

4.3.2 Number of live football broadcasts

In figures 4.11 and 4.12 we can see the distribution of programs in our dataset by channel and competition, respectively. At the channel level, we see a large amount of programs related to

Sports Channel 1 and *Sports Channel 2*.

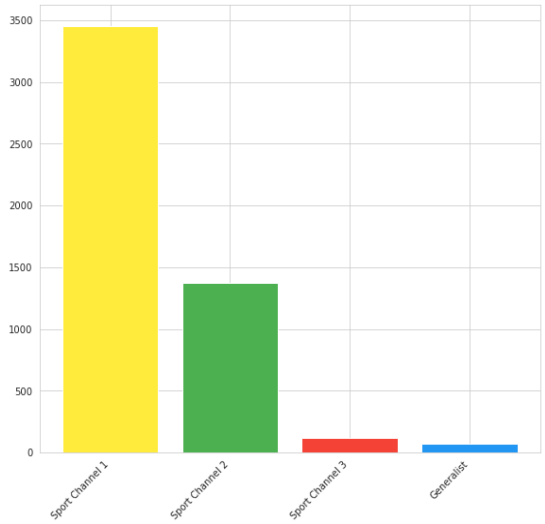


Figure 4.11: Bar plot - Number of live football broadcasts per channel.

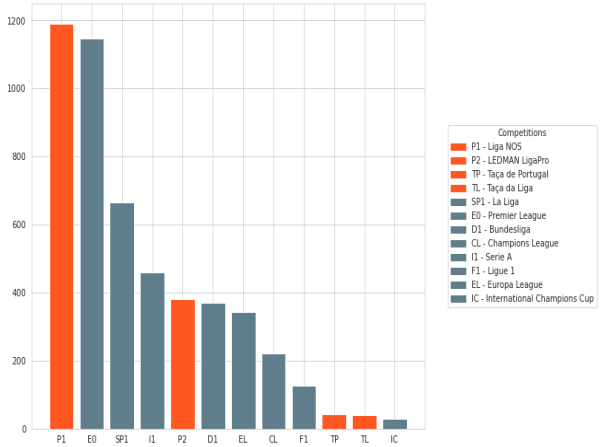


Figure 4.12: Bar plot - Number of live football broadcasts per competition.

At the competition level, a large amount of programs related to *Liga NOS* and *La Liga*. Which shows that although *La Liga* has more extracted games (fig. 4.9), *Liga NOS* has more live TV football broadcasts of the same game.

4.3.3 Distribution of Viewing Time and Clients Volume per Competition

In this visualization, the goal is to check which competitions have the largest audience in total and on average, in the last case normalized by the duration of the programs; In Figure 4.13 in the three-bar plots at the top, which represent the total viewing time, clients volume and total viewing time per client, we can see a clear dominance of the competition Liga NOS (P1). In the case of the three-bar plots at the bottom, which represent the average of the same features normalized by the duration of the program, we can see that the visualization patterns are more evenly distributed across all the competitions, with a slight dominance of views in the Taça da Liga (TL), Taça de Portugal (TP), and Liga NOS (P1).

To have a more refined view of the data variability or dispersion over the different football competitions ("minimum", first quartile (Q1), median, third quartile (Q3), "maximum", and outliers), we decided to apply two box plots (Figure 4.14 and 4.15);

In Figure 4.14 the values of each competition are spread out over the y-axis range counted clients and in Figure 4.15 over the y-axis range summed seconds;

When we look at the clients' count box plot (Figure 4.14) we observe that there is a greater variability for *P1*, *EL*, *TL* and *TP* counted clients. The medians (which generally will be close to the average) are all at the same level. However the box plots in these examples show very different distributions of counted clients. In the case of *TP* and *EL*, larger outliers are also

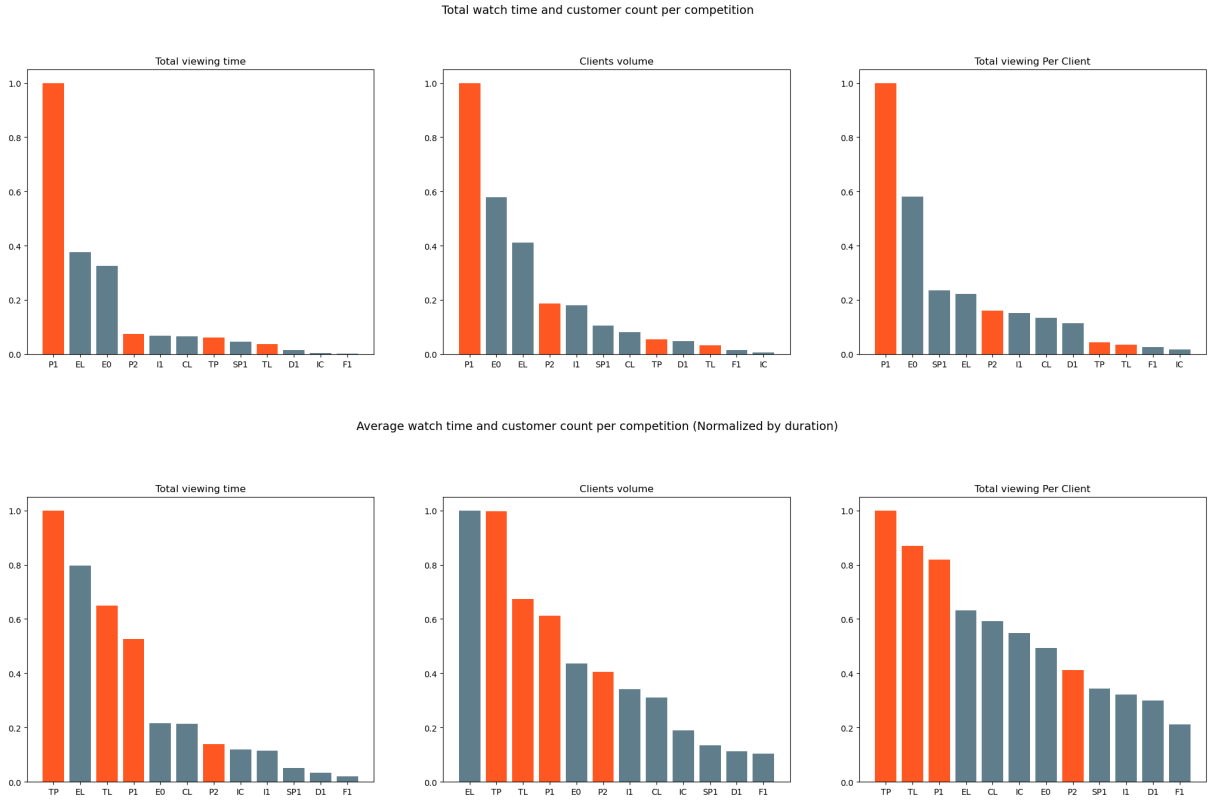


Figure 4.13: Bar plots - The three bar plots at the top represent the total of total viewing time, clients volume and total viewing time per client for each competition; the three bar plots at the bottom represent the average value of the same features, normalized by the program duration

observed.

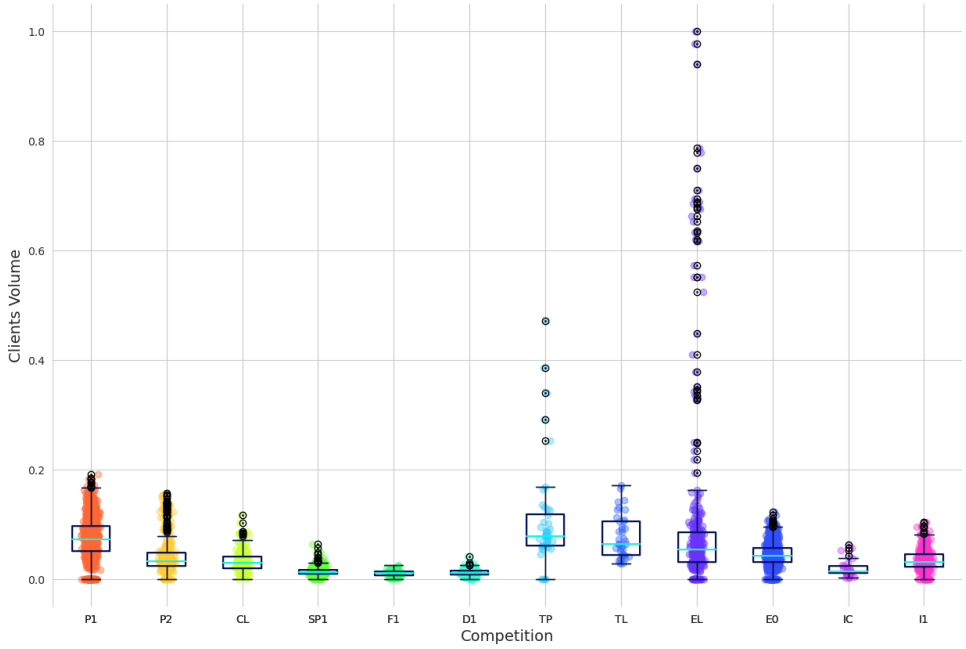


Figure 4.14: Box plot - Depicting groups of numerical data through their quartiles (clients volume)

On the other hand, when we look at the sum of seconds (Figure 4.15), we can observe that the competitions *P1*, *EL*, *TP*, and *TL* have more dispersed summed seconds points (longer boxes). This idea is confirmed when we look at the ranges of each competition (extreme values at the end of two whiskers). In this case, having *TP* and *TL* the longest ranges (two knockout competitions).

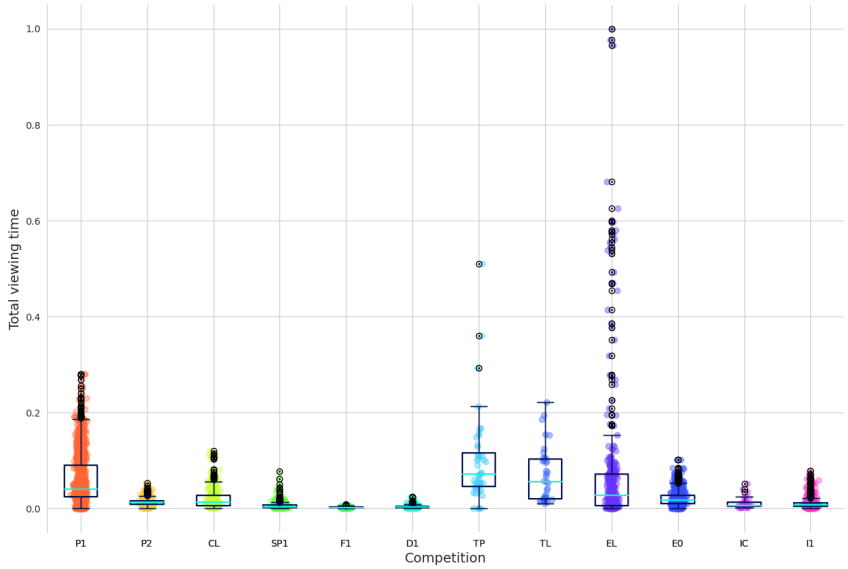


Figure 4.15: Box plot - Depicting groups of numerical data through their quartiles (total viewing time)

4.3.4 Correlation Analysis

Finally, we decided to analyze the correlation of the external features with the target variables counted clients and summed seconds. After doing some high-level analysis, we decided to go into more detail on the relationships that exist between the different features and apply a correlation matrix using the Pearson correlation method. The Pearson correlation r_{xy}, defined by F.R.S. [39], measures the the degree of linear relationship between two continuous variables in a sample and can be measure as follows:

$$r_{xy} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}} \tag{4.2}$$

Where n is the number of observations and x_i, y_i are the ith observation of the value of x and y, respectively. The r_{xy} assumes values between -1 and 1. A value of $\pm$ 1 indicates a perfect degree of association between the two variables. On the other hand, a value of 0 represents no linear relationship. In this case, a significant correlation coefficient is considered if its value is greater than or equal to 0.3. To compute the Pearson correlation we apply the `corr` function from the *python* package pandas.

Based on Uribe et al. [101] where it has been shown that public affection for competition is important in predicting television audiences, we applied Pearson's correlation to four different datasets:

- All competitions. The idea here is to assess the overall impact of external features on viewing time and customer count;
- Local competitions. To assess whether there is a greater association with regard to local competitions, we have gathered the games from the following local competitions: Liga NOS, ledman liga pro, Taça de Portugal and Taça da liga;
- International competitions. To confirm the *home team effect*, we used a dataset with data referring to international competitions: Serie A, La Liga, Bundesliga, Premier League, Uefa Champions League, Uefa Europa League and International Cup.

In figure 4.16 we have the resulting correlation matrix for all tournaments, from this we can draw the following conclusions:

1. It is only possible to observe an association between the number of news and the two target features (cells outlined in red);
2. A moderate negative correlation occurs between the SPI and ranking features. This makes sense, considering that a lower ranking corresponds to stronger teams;
3. The remaining correlations are between features that measure the same external factor (e.g. temperature, radiation)

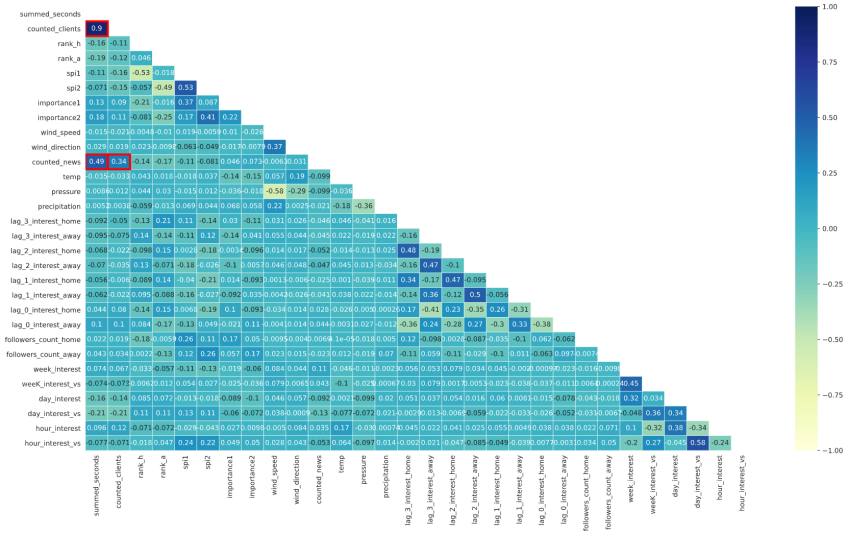


Figure 4.16: All competition correlation matrix.

In the specific case of local games (Figure 4.17), the number of relevant associations is higher (seven in total). Furthermore, the strength of correlation between the external features and the target features is greater in the case of the viewing time feature (*summed_seconds*). We can see that features like *spi*, *news count* and *followers count* have a moderate to strong correlation when compared to the data from all competitions.

In the specific case of followers count, this can be explained in part by the *home team effect* (audience connection with the team). In other words, the television audience sampling is best represented by the count of followers of local teams. Knowing the followers count of a football team in Russia (probably quite a few) won't tell you much about that team's popularity in Portugal. This feature also has a strong association with match quality and match interest features, which makes sense as it is an indirect driver of these variables (e.g., a game with a greater number of followers will likely also be a game with better teams)

Another relevant fact of this visualization is the negative correlation between the features of Google searches interest and the target variables. Indicating that on days with fewer Google searches, the TV audiences increase.

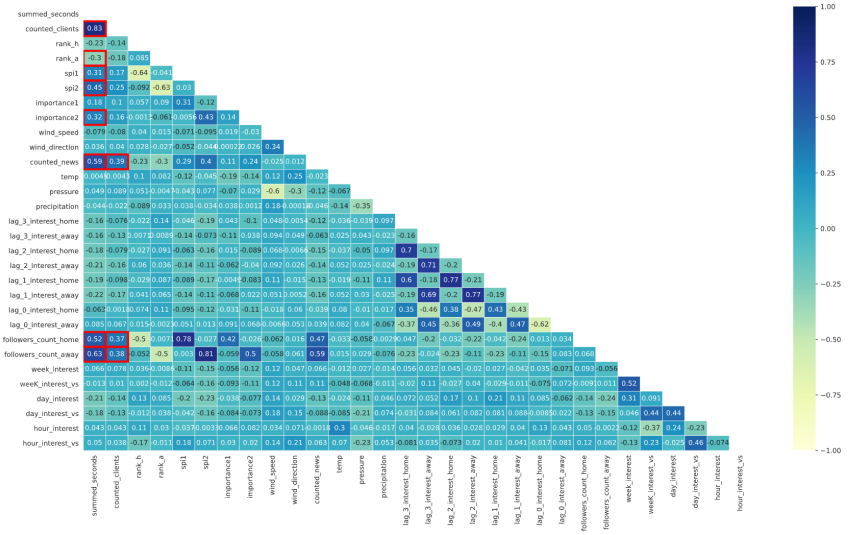


Figure 4.17: Local competition correlation matrix.

Finally, we look only for the international games. From Figure 4.18 it becomes clear that external events have a weak correlation with visualization patterns. This confirms that external events are best represented when looking at local games.

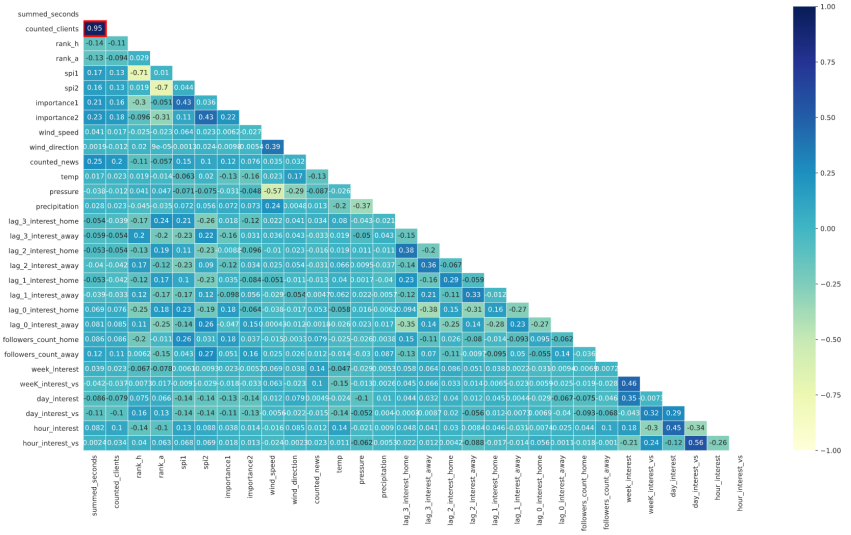


Figure 4.18: International competition correlation matrix.

4.4 Conclusion

In this chapter, we have presented all the external data sources used as well as the methodology used to join the external data sources with the EPG metadata. The data fusion methodology used achieves a usage rate of around 40%, taking into account that not all football matches are broadcast live on TV, these are encouraging results. At the same time, due to API request limits

and missing data tournaments, some football matches were left with a smaller set of information from external events. Despite this, the vast majority of matches have complete set of events data, thus forming a rich dataset of live football matches and external events, ready to be applied to confirmatory analysis.

We also found that, in terms of quantity of content, *Sports Channel 1* is the channel with the greatest number of *live TV broadcasts*, and *Liga NOS* the competition with the most content of this type. In addition, we also discovered that in average knockout tournaments have a higher audience.

This chapter also served to realize that competitions with little connection to the public are less affected by external events.

Chapter 5

Machine learning aproach to football TV forecasting

This chapter provides an overview of the methodology used to forecast football live broadcasts taking advantage of the rich dataset create on section 4.

Through an empirical study, a series of machine learning methods are compared. In the end, closely following methodology applied in Selim et al. [97] and in Neagu [78], the most accurate machine learning model forecasting the TV audiences is compared against a simpler statistical model, to check and validate the obtained results.

Comparing results with and without both content and external features allows us to verify whether (1) the model is flexible enough to use different kinds of attributes (2) how the different kinds of attributes affects the forecasting accuracy.

In addition, to see the impact that external features have on TV viewership, and inspired by Parsa et al. [81], Feddersen and Rott [35] and Uribe et al. [101], we apply a popular method of output model explainability.

How different data subsets affects the model accuracy and features importance is also tested using different tournaments splits.

5.1 Methodology

5.1.1 Sample

To test the effect of external features, we broke the data created in section 4 into two different datasets: a univariate time series of the volume of clients without external features and a multivariate time series dataset with all external features.

In addition, we split the data based on the football tournament region (local and international)

to test the accuracy of the forecasting models on different data subsets. This results in three different datasets: the original dataset with matches from all tournaments, a dataset with matches from local tournaments and a dataset with matches from international tournaments.

The predictor variables that we use have frequently been identified as relevant drivers of TV audience of football games and closely follow the variable specifications that are used by Uribe et al. [101], Nixon et al. [80] and [59]. Table 5.1 shows the top rows of the dataset used.

Table 5.1: Sample of the *all_data* dataframe.

channel_name	program_title_dsc	home_team	away_team	event_start_time_local	event_end_time_local	External Features	summed_seconds	counted_clients
channel_name_SP1CH2	PREMIER LEAGUE - CRYSTAL PALACE X BRIGHTON & H...	Crystal Palace	Brighton	2019-03-09 12:30:00	2019-03-09 14:30:00	...	0.009916	0.026400
channel_name_SP1CH2 HD	PREMIER LEAGUE - CRYSTAL PALACE X BRIGHTON & H...	Crystal Palace	Brighton	2019-03-09 12:30:00	2019-03-09 14:30:00	...	0.009129	0.021598
channel_name_SP1CH1 HD	LIGA NOS - MARÍTIMO X MOREIRENSE (DIRETO)	Maritimo	Moreirense	2019-03-09 15:30:00	2019-03-09 17:40:00	...	0.031389	0.049866
channel_name_SP1CH1	LIGA NOS - MARÍTIMO X MOREIRENSE (DIRETO)	Maritimo	Moreirense	2019-03-09 15:30:00	2019-03-09 17:40:00	...	0.041676	0.066752
channel_name_SP1CH2 HD	PREMIER LEAGUE - MAN. CITY X WATFORD (DIRETO)	Man City	Watford	2019-03-09 17:30:00	2019-03-09 19:30:00	...	0.017220	0.049345

Before embarking on model development, it is worth emphasizing that, in our forecasting context, the criterion for a good model is that it predicts well and has good explanatory power so that we can validate the effect of external events on TV audiences. Taking this requirements into account, 4 models were selected: a linear model (ARIMA) and 3 ensemble nonlinear models (Random forest, Gradient boosting and XGBoost).

5.1.2 ARIMA

In this research, an ARIMA methodology was conducted to compare its performance with the events-based machine learning model. To fit ARIMA to the available time series, the following steps were executed.

1. **Determine the right order of differencing(d):** To check stationariy we use the Augmented Dickey Fuller test from the statsmodels package. The null hypothesis of the ADF test is that the time series is non-stationary. So, if the p-value of the test is less than the significance level (0.05) we can reject with 95% of confidence that the time series is non-stationary;

2. **Find the right order of the AR term (p):** To find the number of AR terms we use a partial autocorrelation (PACF) plot, this type of plot evaluates the pure correlation between a lag and the series. We take the order of the AR term to be equal to as many lags that crosses the significance limit in the PACF plot;

3. **Find the right order of the MA term (q):** To find the number of MA terms we use a autocorrelation (ACF) plot, which is the autocorrelation between an observation and another observation at a prior time step that includes direct and indirect dependence information. As in PACF, we take the order of the MA term to be equal to as many lags that crosses the significance limit in the ACF plot. For the final model identification of the AR and MA terms, we use the specification in table 5.2;

4. **Evaluate ARIMA model using a *walk-forward* validation:** to see how effective our model really would have been in the past data, we use a *walk-forward* approach to predict the TV viewership.

Table 5.2: Table used for ARMA identification.

Conditional Mean Model	ACF Behavior	PACF Behavior
AR(p)	Tails off gradually	Cuts off after p lags
MA(q)	Cuts off after q lags	Tails off gradually
ARMA(p,q)	Tails off gradually	Tails off gradually

5.1.3 Random Forest, Gradient Boosting and XGBoost

To compare the performance of ML methods to traditional statistical ones in TV viewership forecasting, we apply 3 ensemble methods. We choose this three machine learning methods because they are a compromise between performance and explainability (see section 2).

To fit all the 3 ensemble methods, the following steps were executed.

1. **Feature selection:** To reduce computational costs and to avoid the curse of dimensionality we remove redundant features from the model. The following methods are used to feature selection:
   (a) Missing Values. Find any columns with a missing fraction greater than a specified threshold (features with a value greater than 0.30 missing values are removed);
   (b) Single Unique Values. Find any features that have only a single unique value;
   (c) Collinear Features. Collinear (highly correlated) features (features with a correlation magnitude greater than 0.97 are removed).
2. **Imputer:** for completing missing values we use an imputer.
3. **Normalize data:** To guarantee that our algorithm can generalize better on the test set we apply a normalization to our data.
4. **Add time variables:** in order to capture trends, season and cyclical patterns from the multivariate dataset we add the following time based features:
   - Month;
   - Year;
   - Week;
   - Day;
   - Dayofweek;

- Dayofyear;
- Is_month_end;
- Is_month_start;
- Is_quarter_end;
- Is_quarter_start;
- Is_year_end;
- Is_year_start;
- Elapsed.

5. **One hot encoding:** in order to assess the impact of some categorical external features we apply one hot encoding to the following features:

   - div;
   - result_home (Only used in lagged features to avoid data leakage);
   - result_away (Only used in lagged features to avoid data leakage);
   - channel_name;
   - home_team;
   - away_team.

6. **Add lagging variables:** As these three machine learning methods evaluate the data points without making connections to previous information (as opposed to linear models, for example), and following the implementation in Khryashchev et al. [59], we define the lags of a given football match broadcast live on a given channel as follows: the N home team football matches and N away team football matches, on the same channel as the actual live TV broadcast, that take place before the actual match;

7. **Backtesting:** To access the performance of the models on the historical data, two backtesting strategies (a special type of cross-validation applied to time series data) were addressed: sliding window and expanding window. This type of approach allows evaluating the ability of predictive models to generalize, avoiding at the same time overfitting [12]. Figure 5.1 illustrates the validation process of the sliding window strategy in n different data splits. New points are added at the front (gray window) and the older points are removed as the window moves ahead. In the case of expanding window (figure 5.2), the only difference is that the previous points are not removed and are used to predict the points ahead.

For model training, we used the *XGBoost* packages and the Scikit-learn library [84].

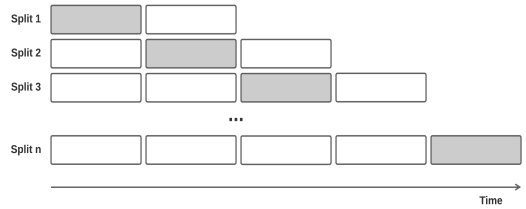


Figure 5.1: Sliding window.

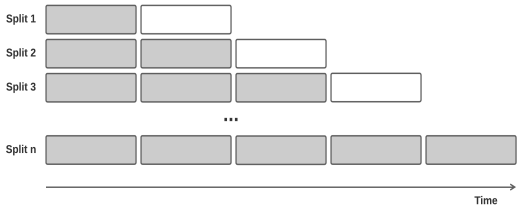


Figure 5.2: Expanding window.

5.1.4 Machine learning parameters optimization

To consider what is the best set of parameters for testing the impact of external events data on machine learning forecasting models, we have designed an exhaustive search test to assess which of the following parameters are a good match:

- **Model.** The goal here is to outline which machine learning model is most accurate in our specific forecasting problem. Three models were used for this purpose: a random forest, a gradient boosting and XGBoost model;
- **Cross Validation.** In this, we focused on verifying which backtesting approach is most appropriate for our irregular time series. For this purpose, we test two types of approaches: sliding window (more appropriate to test high-frequency data) and expanding window (more appropriate for time series with larger intervals between points);
- **Train size.** To check the best training window size (in the case of the sliding window) and initial starting window size (in the case of expanding window), we tested our models with four different train sizes: 100, 500, 1000, and 1500;
- **Test size.** To determine the best forecast window size to apply for each backtesting method, the following test sizes were selected: 200, 100, 50, 25 and 1;
- **Lags for the football live broadcasts.** Since there is a tradeoff between the number of lags and the number of predictions (the greater the number of lags, the more football live TV broadcasts are dropped), we also tested which is the most suitable choice for this parameter;
- **Normalizer.** Finally, we test whether normalizing our data to a smaller scale helps improving the accuracy of television audiences forecasting. For this purpose we tested the *StandardScaler* and the *MinMaxScaler* (We use the raw data as a baseline).

Note that we do not do any tuning to the models, we use the default configuration parameters for all three models, with the number of threads set to three (except for the gradient boosting model which currently does not allow parallelization). Future work should be done in this direction.

Table 5.1 shows the lists of manually defined values for the exhaustive search test.

Table 5.3: Set of values for each paramater used in the exhaustive search test.

Parameter	Range of values
Model	[RandomForestRegressor, GradientBoostingRegressor, XGBoost]
Cross validation	[Sliding Window, Expanding Window]
Train size	[100, 500, 1000, 1500]
Test size	[200, 100, 50, 25, 1]
Lags size	[1,2,3, 4, 5]
Normalizer	[Raw, StandardScaler, MinMaxScaler]

5.1.5 Model evaluation

Once a model has been generated and tested, its performance should be evaluated. In this study, three forecast error measures, namely, Root Mean Squared Error (RMSE), Normalized Root Mean Squared Error and Symmetric Mean Absolute Percentage Error (SMAPE) were employed for model evaluation and model comparison.

RMSE expresses the standard deviation of the residuals. The lower the RMSE, the better is the forecasting model. It can be represented by the following equation:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}(\hat{y}_t - y_t)}{n}}$$

where y_t is the actual observation, $\hat{y}_t$ is the forecast at period t, and n is the number of different predictions.

As we are testing models with possibly different scales (all tournaments data, local tournaments data and international tournaments data) we also use a normalized version of RMSE, given by the following equation.

$$NRMSE = \frac{RMSE}{y_{max} - y_{min}}$$

where y_{max} and y_{min} are the maximum and minimum value observed, respectively.

Finally, we use sMAPE (Symmetric Mean Absolute Percentage Error) as a relative error measure. This measure is easy to understand because it provides the error in terms of percentage. The SMAPE value can never be greater than 200%.

$$SMAPE = \frac{100\%}{n} \sum_{t=1}^{n} \frac{|\hat{y}_t - y_t|}{|\hat{y}_t| + |y_t|}$$

5.1.6 Model interpretation

To interpret the output of the model we use SHAP (SHapley Additive exPlanations), proposed by Lundberg and Lee [70], SHAP is based on game theory and it offers a means to estimate the contribution of each feature. Features with higher absolute shapely values are more important for the forecasting outcome. Shapely values are determined through:

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n - |S| - 1)!}{n!} [f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_s)] \tag{5.1}$$

Where ϕ_i is the shapley value for feature i, f the black box model and the N is a group of n features used to predict an output.

Lundberg and Lee [69] developed a practical package in Python that is able to calculate SHAP values for different techniques including XGBoost.

5.2 Results

5.2.1 Machine learning parameters optimization

Through an empirical comparison of three machine learning models (random forest, gradient boosting and an xgboost model) for predicting TV ratings, we decided to use XGBoost as our final evaluation model since it is the one with the most consistent results throughout the tests (table 5.4).

Table 5.4: Overview of the results across all the different models.

Model	# tests	Average RMSE	Average SMAPE	Average Total Time
Random Forest	84	2728.53	**37.80**	3824.03
Gradient Boosting	84	2737.12	44.55	2922.95
XGBoost	84	**2680.30**	39.15	**1121.70**

Moreover, we also concluded that for xgboost the best train/test split results are the 1000/1 and 1500/1 ratios. Finally, the best lag for split 1000/1 is two and for split 1500/1 is four. For our final evaluation, we choose to use the 1000/1 split with a match lag of two since would allow was to test more football matches live broadcasts. Despite that, in our final test, as we are using different data subsets, in some cases we choose to use a smaller train size (with the same train/size ration) and a smaller match lag size as well.

Table 5.5: The set of parameters for the best RMSE and the best SMAPE among all the different models.

Best RMSE								
Model	Backtesting	Normalizer	Lags	Train Size	Test Size	RMSE	SMAPE	total_time
Random Forest	GameBasedEWCV	StandardScaler	4	1500	1	**1806.30**	35.44	1132.41
Gradient Boosting	GameBasedEWCV	StandardScaler	4	1500	1	2020.84	45.40	1080.79
XGBoost	GameBasedSWCV	StandardScaler	4	1500	1	1907.23	41.15	**266.39**
Best SMAPE								
Random Forest	GameBasedEWCV	Raw	1	1500	1	2295.72	**33.43**	5545.69
Gradient Boosting	GameBasedSWCV	StandardScaler	3	1000	1	2189.65	40.39	4228.14
XGBoost	GameBasedEWCV	StandardScaler	2	1000	1	1936.83	34.17	1711.95

In this test, 2 different types of normalization were also tested, specifically: *StandardScaler* and *MinMaxScaler*. We use a *StandardScaler* normalization approach in our final evaluation since it was the normalizer with the best results (see appendix A for more details).

5.2.2 ARIMA - Univariate approach

After the calculation of the match lag for the machine learning models, some matches end up being removed. To have a more reliable comparison, between the different models, only the matches used in the machine learning models were considered in the ARIMA model.

As described in the previous section, we start by performing a stationarity check. In the different data aggregations, the null hypothesis that the series is non-stationary was rejected (p-value < 0.05). It was not necessary to apply no differentiation (d=0). To find de AR term (p) and de MA term (q) we applied the PACF and ACF plot, respectively. We verified, in all the cases, that the ACF cuts off after 1 lags, and PACF decays, so ARIMA(0,0,1) (or MA(1)) is the chosen model.

Figure 5.3 displays the results of the expanding window cross validation for the error measures presented in subsection 5.1.5 (The orange dot line plot represents the prediction value). In the all tournaments model, the accuracy is close to 0.07 in NRMSE and 59.01% in the SMAPE. In local tournaments, the accuracy is close to 0.08 in NRMSE and 40.39% in the SMAPE. Finally, in the international tournaments model, for the same error measures, the accuracy is 0.17 and 55.31%.

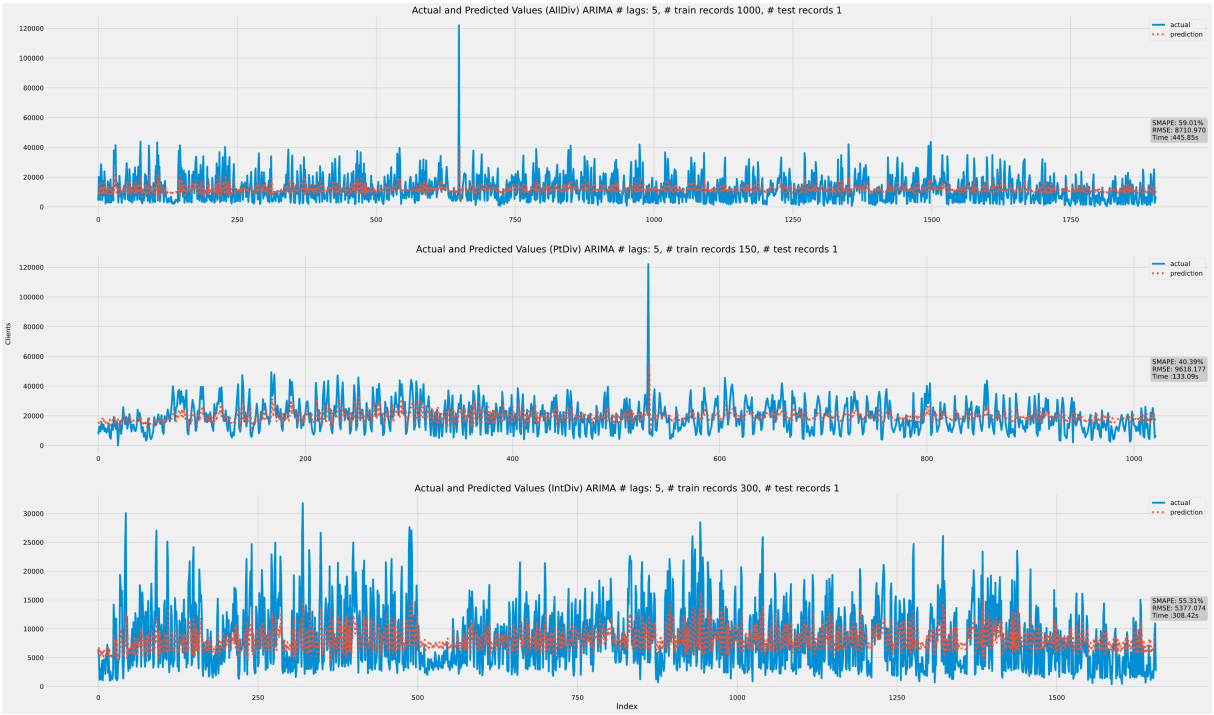


Figure 5.3: ARIMA walk forward results.

Visually we can notice that although there are certain differences among the tournament models, they all underfit the true values. We also verified that the prediction of games for local competitions had better results.

5.2.3 XGBoost - Multivariate approach

To assess the impact that external sources data have on TV viewership forecast, we start by applying the pipeline described in subsection 5.1.4. We use a training size of 150 and a lag of 1 in the local tournament dataset so that we can make a comparison against the results of the general model. For the same reason, we use a training size of 300 in the international tournament data, keeping the number of lags the same of the all tournaments dataset (2 lags). For the remaining parameters, we use the choices made in the subsection 5.2.1 .

Fig. 5.4 displays the results of the expanding window cross validation using the multivariate approach. In the all tournaments model, the accuracy is close to 0.02 in NRMSE and 21.30% in the SMAPE. In local tournaments, the accuracy is close to 0.03 in NRMSE and 19.11% in the SMAPE. Finally, in the international tournaments model, the accuracy is 0.06 and 24.01% for the NRMSE and SMAPE measures, respectively.

Figure 5.4: XGBoost expanding window results.

Comparing the two models, we can see that the univariate approach has more difficulty in tracking customer count variations over time. On the other hand, we noticed in both cases difficulty tracking outliers (approximately the game index 1600 and 680 in the first and second plot of figure 5.4, respectively), this may happen since we do not have all the external event information available. To get around this, a possible solution presented in Nixon et al. [80] would be to first look at all outliers and determine whether they report a specific TV event or not. Nevertheless, the results support the hypothesis that the inclusion of external features, especially when it is difficult to capture the TV seasonality and trend components, due to the irregularity of the time series, can bring advantages in terms of TV demand prediction.

Table 5.6 shows the RMSE, NRMSE and SMAPE values for each model across all data splits. The XGBoost model obtains the highest accuracy. On the other hand, the ARIMA model has a shorter execution time across all the splits.

Table 5.6: Summary of the accuracy results.

Method	**Tournament**	**RMSE**	**NRMSE**	**SMAPE(%)**	**Time(s)**
ARIMA	All	8710.97	0.07	59.01	**445.85**
ARIMA	Pt	9618.18	0.08	40.39	**133.092**
MoARIMA	Int	5377.07	0.17	55.31	**308.42**
XGBoost	All	**2210.03**	**0.02**	**21.30**	5116.18
XGBoost	Pt	**3366.78**	**0.03**	**19.11**	573.79
XGBoost	Int	**1839.56**	**0.06**	**24.01**	2671.74

5.2.3.1 Features importance analysis

In order to determine the impact of features on the model's predictions, we use the SHAP method (Section 5.1.6).

Figure 5.5 shows the *all tournaments* SHAP plot of the features that the model relies on most to make its predictions.

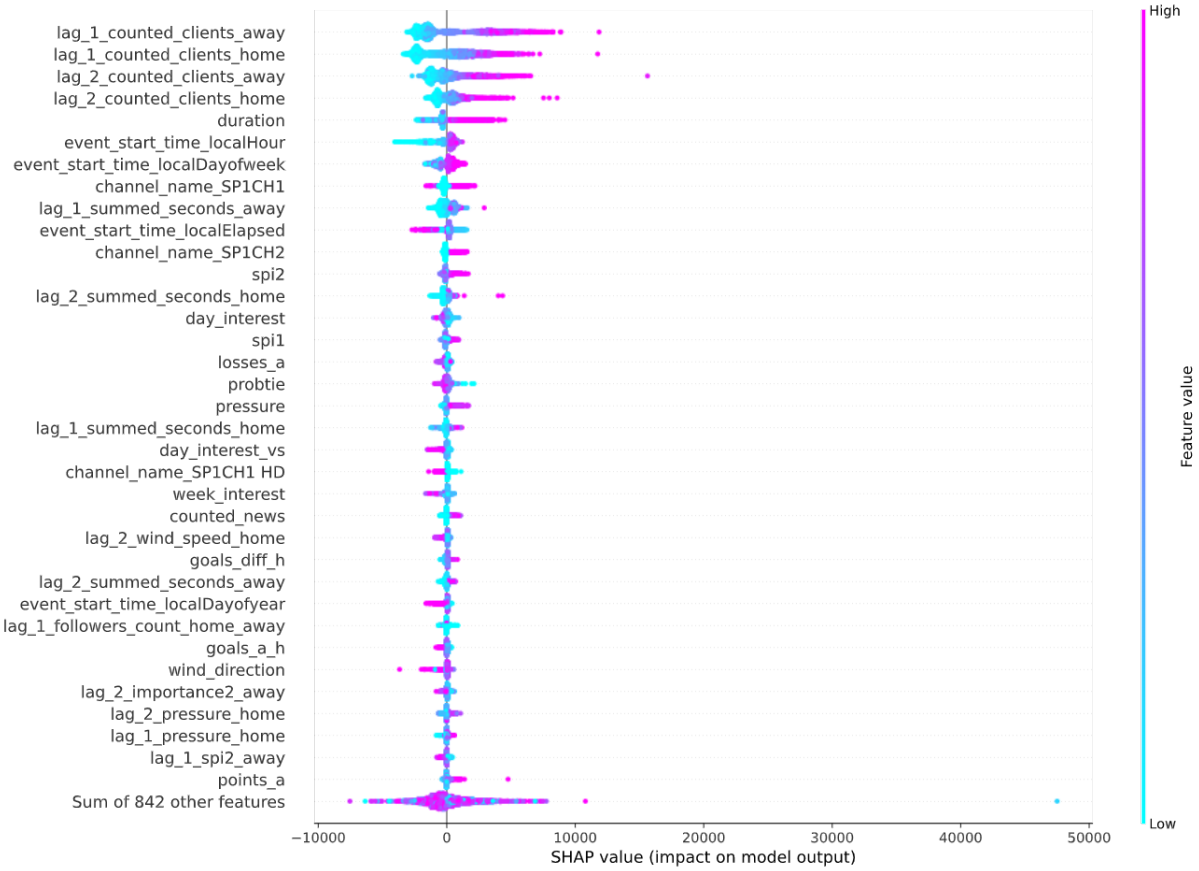


Figure 5.5: All competitions shap.

We can see that the lagged features of volume of clients are the most important features in the model. This makes sense considering that the customer count over the course of a tournament does not vary much for the same team matches (i.e., a team that had many viewers in the last games is expected to also have in the following ones). Particularly, when we compare the lagged values for the home and away team the customer count value of the away team has a greater impact on the model (i.e. *lag_1_counted_clients_away* and *lag_2_counted_clients_away*). We also found that lagged external features have an lower impact on the model, which makes sense considering that these are more distant events to the match time.

Duration and start time of a match are the next two most important features after the lagged *counted_clients*, and lower values of these features correspond to a lower customers count. The transmission channel also has an impact on customer count (*channel_name_Sports Channel 1*,

channel_name_Sports Channel 2.

In terms of external-based features we can see that, in general, higher values for match quality (*spi1*, *spi2*, *goals_diff_h*, *points_a*, *importance2*) and match interest features (*counted_news*) result in higher customer count values. On the other hand, higher outcome uncertainty values (*probtie*) result in lower customer count. Although logic tells us otherwise (i.e. more spectators are attracted to watch matches in which the outcome possibilities of the competing teams are equally balanced), Forrest and Simmons [36] showed that tournaments where matches with unbalanced teams are in bigger number, the impact of outcome uncertainty is usually negative (weak home team hosts a weak away side). Finally, The weather conditions have, as expected (sunnier and warmer weather decreases demand for television broadcasts [35]), influence on television customer count (*pressure*, *temp* and *precipitation*).

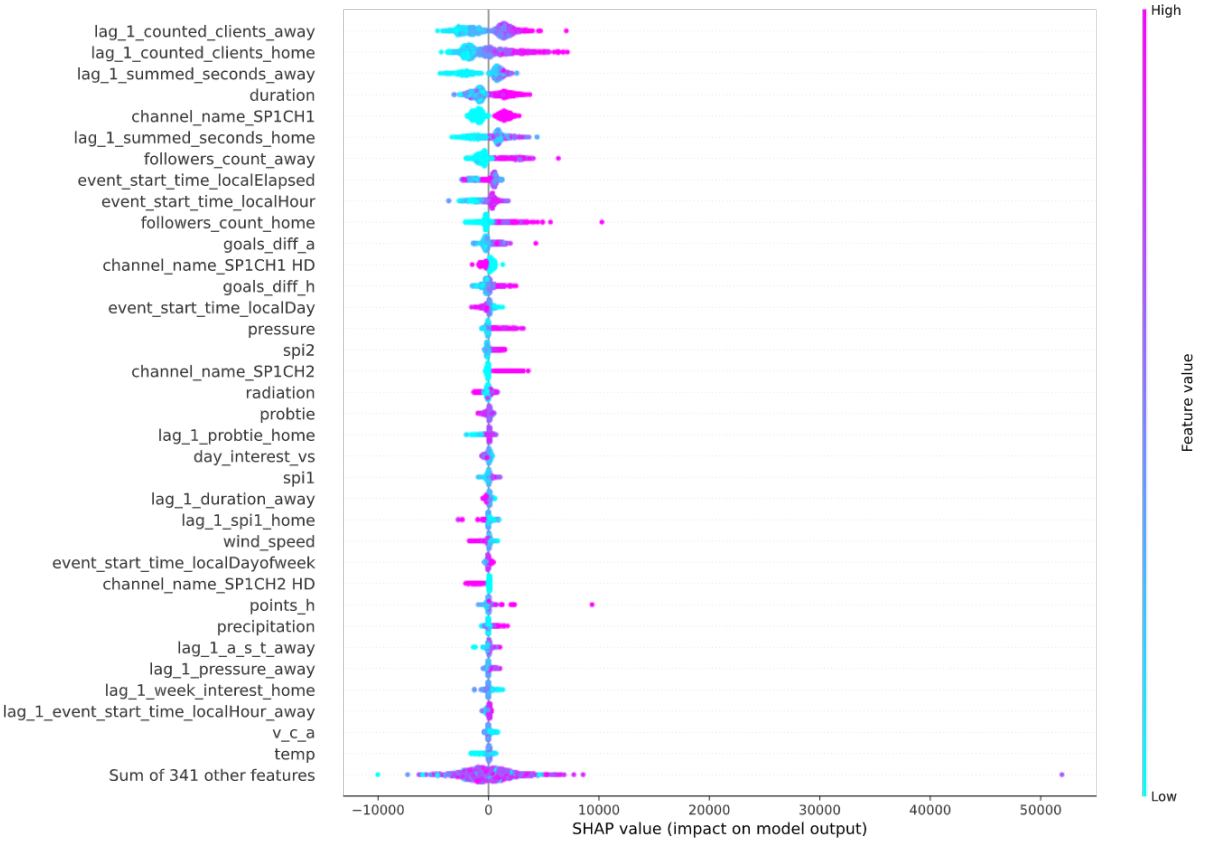


Figure 5.6: Portuguese competitions shap.

When we only consider local matches (Fig. 5.6) we see that the model relies more or less on the same content-related features (*lag_1_counted_clients*, *lag_1_counted_clients_home*, *duration*, *event_start_time_local* and *event_start_time_local*) to forecast customer count. In the case of event-related features, highlight the emergence of teams' popularity as one of the most important features (*followers_count_home* and *followers_count_away*). This fact confirms once again that the supporters' affection for local teams has an impact on the customer count, as shown in Uribe et al. [101].

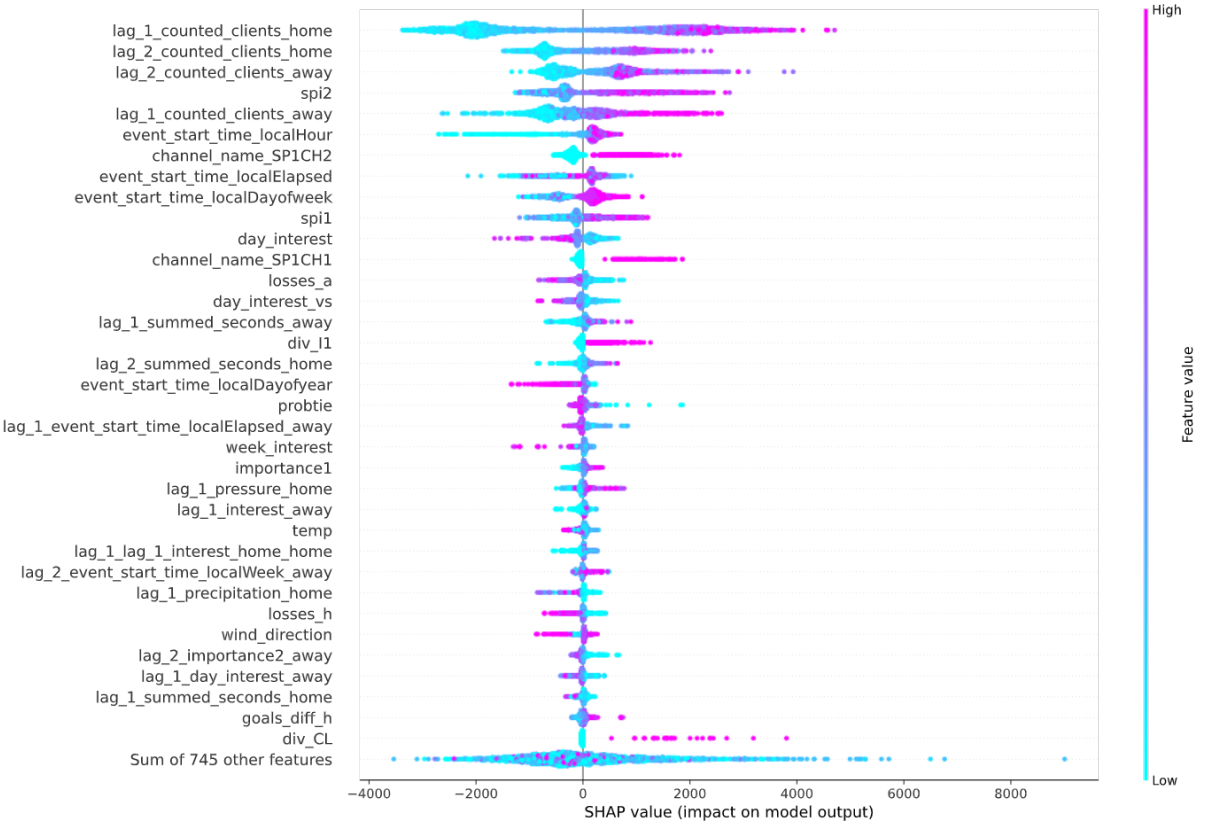


Figure 5.7: International competitions shap.

Finally, we conducted the same SHAP method in the dataset with international matches and the result is shown in figure 5.7. Here a great emphasis is given to features related to the match quality (*spi1* and *spi2*). This may indicate that when there is no local team present, fans are essentially looking for a high quality matches. Another curious fact about this visualization is the impact that different tournaments have on customer count. A positive shapley value in the case of Serie A and Champions League tournaments (*div_I1*, *div_CL*) and a negative shapley value in the case of Bundesliga e La Liga tournaments *div_D1* and *div_SP1*). This fact once again may be due to the *home team effect* as Serie A has a very famous local player (Cristiano Ronaldo) and the Champions League being a tournament open to teams from all over Europe also has, in some matches, the presence of local teams.

5.3 Conclusions

In this work, we compare four forecasting algorithms for time series - ARIMA, Random Forest, Gradient boosting and XGBoost for the problem of TV viewership forecasting. We show that the inclusion of content-based features and external-based features, using a multivariate approach, considerably increase the accuracy of the TV viewership forecast compared to the univariate base model that takes into account only previous television audiences counts. Considering

the general case (dataset from all tournaments), and taking the RMSE as the error measure, the inclusion of external features using a multivariate approach resulted in an improvement of roughly 70% in model accuracy. In addition, we apply SHAP, a method that estimates the importance of features through shapley values. From this analysis we conclude that, in general, the content-based features have a bigger impact on the model's predictions, closely followed by external-based features such as match quality and match interest. Finally, based on the different dataset aggregations, we conclude that people's affection to a competition (*home team effect*) also has an impact on TV customer count.

Chapter 6

Impact of external factors in the service viewing time and volume of clients

In order to go one step further in evaluating the impact of external events on TV viewership, and confirm the results obtained in section 5, we apply a causal relationship analysis framework between real-world data (weather, news, google trends) and TV sports viewership. To do so, we employ an econometric framework based on time series methods.

In this chapter, we will start describing the methodology used in the construction of the econometric framework and all the hypotheses formed. We will finish presenting a case study involving several external factors and a popular local football tournament.

6.1 Methodology

6.1.1 Sample

For this study, a general to specific procedure was used. In this sense, a set of case studies were applied. These case studies were used to explore the causal relationship between real-world data in TV viewership in complete season (19/20) for a specific football tournament (Liga NOS).

It should be noted that, despite having a dataset with a large number of football matches referring to several local and international competitions (see chapter 4), this study only considers the impact of external events on a local football tournament, more specifically, on the Liga NOS for the 19/20 season. The main reason is that it is very difficult to quantify the impact of external events in such a diverse environment of competition and competitive structures. For example, the spectators' affection for a local tournament is different from the affection for an international tournament [101] [8]. Also, tournaments with different competitive structures (i.e., cup and league

tournaments) cannot be compared directly, intuitively the impact of external events will have on a three-point rule tournament, where the most regular team wins the championship, is different from an elimination tournament, where fewer matches are played and the unpredictability of results prevails. However, this two assumption has not been investigated in this study. This can be tested in future research.

We chose the Liga NOS tournament because is the competition with the greatest impact in terms of television audience (see section 4.3.3), and the 19/20 season for being the only one available that brings together all the matches from an entire season. Table 6.1 shows the top rows of the sample used for the causality analysis.

Table 6.1: Sample of the Liga NOS 19/20 dataset.

event_start_time_local	home_team	away_team	spi_match	counted_news	precipitation	external_features	counted_clients_shift1	summed_seconds_shift1
2019-08-10 19:00:00	Gil Vicente	FC Porto	60.025	3.0	0.000000	(...)	0.391411	0.171781
2019-08-10 21:10:00	Benfica	F.C. Paços de Ferreira	59.440	1.0	0.000000	(...)	0.505247	0.378332
2019-08-11 18:30:00	Maritimo	Sporting CP	53.475	3.0	0.005786	(...)	0.353997	0.269769
2019-08-11 21:00:00	Sp Braga	Moreirense	51.335	3.0	0.000297	(...)	0.545166	0.404101
2019-08-12 20:15:00	Vitoria Setubal	Tondela	41.245	1.0	0.000000	(...)	0.343146	0.125612

6.1.2 Granger causality test

To establish whether real-world event data can affect and hence predict future TV viewership fluctuations, will use a Granger causality test [41]. The essential principle of Granger causality analysis is to test whether the past values of one variable X (the driving variable) help explain the current values of another variable Y (the response variable).

There are many ways to apply this Granger causality test, in our specific case we follow an approach in Hamilton [44] that uses a bivariate vector autoregression. This assumes a lag length p, and estimates the restricted and unrestricted equation by ordinary least squares(OLS).

$$y_t = c_0 + \sum_{i=1}^{p} \gamma_i y_{t-i} + e_t \tag{6.1}$$

$$y_t = c_1 + \sum_{i=1}^{p} \alpha_i y_{t-i} + \sum_{i=1}^{p} \beta_i x_{t-i} + u_t \tag{6.2}$$

Then an F test of the null hypothesis is conducted,

$$H_0 : \beta_1 = \beta_2 = ... = \beta_p = 0 \tag{6.3}$$

And the sum of squared residual from the restricted model [6.1] and the unrestricted model [6.2] are compared,

$$RSS_0 = \sum_{t=1}^{T} \hat{e}_t^2 \; RSS_1 = \sum_{t=1}^{T} \hat{u}_t^2 \tag{6.4}$$

The F-test is conducted by the following expression:

$$F = \frac{(RSS_0 - RSS_1)/p}{RSS_1/(T - 2p - 1)} \tag{6.5}$$

If the critical value of F at 95% probability level is lower than the observed value of F [6.5], we reject the null hypothesis(H_0).

To apply the Granger causality test, we use `grangercausalitytests` function from the package *statsmodels* [96]. The significance level we use is 5% and if the p-value of a pair of variables is smaller than 0.05, we could say with 95% confidence that a predictor x causes a response y. The null hypothesis test (H_0) is that the lagged values of x does not Granger cause y. For this problem, we run the Granger test with only one lag (p=1). This choice happens for two reasons. First, in this study, we are more interested in the short-term impact of events on viewing time and volume of clients that in long-term (as seen in section 5 the short-term impact events have a closer relationship with the TV audience). Second, a bigger lag than one, in the Liga NOS case study, corresponds to a different football match between two other teams (as shown in the table 6.1) and although the events that affected other matches (between two different teams) can also affect the next one, for the sake of simplicity, we chose to just study the impact of events before a match.

Although the Granger causality test used can provide support about a hypothesis, note that being a bivariate analyses, this test cannot account for indirect links or common drivers.

6.1.3 Unit root test

The Granger causality test assumes series stationary. So before conducting causality tests, in order not to have false estimates, we first test whether the series is stationary or not. For this purpose, we use the augmented Dickey Fuller Dickey and Fuller [27] unit root test.

In the case of non-stationarity, differentiation is applied to the series in order to make them stationary, thus removing trends and seasonality. The differenced series can be written as:

$$y_t^{'} = y_t - y_{t-1} \tag{6.6}$$

This results in $T - 1$ values, since the first value cannot be differentiated.

6.1.4 Variables

Based on prior knowledge and studies (see section 3), it was hypothesized that:

1. Match result uncertainty Granger causes TV viewership;

2. Participant teams' quality Granger causes TV viewership;
3. Participant teams' interest Granger causes TV viewership;
4. Participant teams' popularity Granger causes TV viewership;
5. Weather conditions Granger causes TV viewership;
6. Scheduling Granger causes TV viewership;
7. TV-network Granger causes TV viewership.

In order to make a better assessment of these hypotheses, a set of variables were selected within the scope of each external factor (see table 6.2).

Table 6.2: Summary of the external variables used for the causality analysis.

Factor	Variable	Description
Outcome uncertainty	'probtie'	Match tie probability
	'b365_d'	Match draw odds
Match quality	'goals_diff_match'	The average goals difference for the home and away team
	'goals_a_match'	The average goals against for the home and away team
	'losses_match'	The average number of losses for the home and away team
	'rank_match'	The average raking of the match
	'spi_match'	The average strength of the home and away team
	'importance_match'	The average importance of the home and away team
Match interest	'counted_news'	Number of news in the days before the match
	'week_interest'	Number of google searches in the week preceding the match
	'day_interest'	Number of google searches in the day preceding the match
	'hour_interest'	Number of google searches in the hour preceding the match
Match popularity	'followers_count_match'	Sum of the number of followers of the home and away team on twitter
Weather	'wind_speed'	Atmospheric quantity just before the start of the match
	'temp'	
	'precipitation'	
Scheduling	'day_of_week'	Day of the week
	'hour'	Hour
TV-network	'counted_channels'	Number of channels a match was broadcast live

The outcome uncertainty is referred to as one of the most important factors in public forecasting of sporting events [36] [95] [89] [85]. In this way, we measure the impact of this factor through a draw projection produced by the *SPI* and a draw odd on *b365* bookmakers.

The match quality and the match popularity of the game are other factors that possibly influence the number of television audiences [101] [29]. In order to obtain these match values, we follow the implementation in BORLAND and MACDONALD [15] and grouped the features referring to the home and away team into a single feature. In this study, n teams are considered. The ranking of each team based on performance is $\{T_1, T_2, T_3, ..., T_n\}$, where T_i identifies the ranking of the ith team i. Knowing that the success of competing teams can be measured by rank-oder of each team (e.g., T_i, T_j). The quality of the match can be expressed by the average rank-order of competing teams [15].

$$(T_i + T_j)/2 \tag{6.7}$$

Search patterns have also proved useful in give informing about mass behavior [74]. Therefore, we use Google search trends match features. In addition, the number of news in the 5 days preceding the game is also used as an interest factor.

Broadcast related factors are also considered to have an impact in terms of television audiences [103] [80]. Thus, we used the day of the week and the game time as scheduling factors and the number of channels that broadcast the game as a broadcast factor.

6.2 Results

6.2.1 Case study for Liga NOS 19/20

To verify the proposed methodology we use seven cases studies. On one hand, a general case study with all Liga NOS matches validates whether or not external events have predictive power on TV volume and viewing time of Liga NOS live content. On the other hand, six teams case studies evaluates the external events predictive power across different teams context, providing more accurate information about the Liga NOS TV audience behavior. For this purpose, we select two teams with big TV customer engagement (FC Porto and Sporting CP), two teams with medium customer engagement (Sp Braga and Famalicão FC) and two teams with low customer engagement (Rio Ave and CD Aves). The average viewing time and the volume of customers per team in Liga NOS in the 19/20 season is shown in figure 6.1.

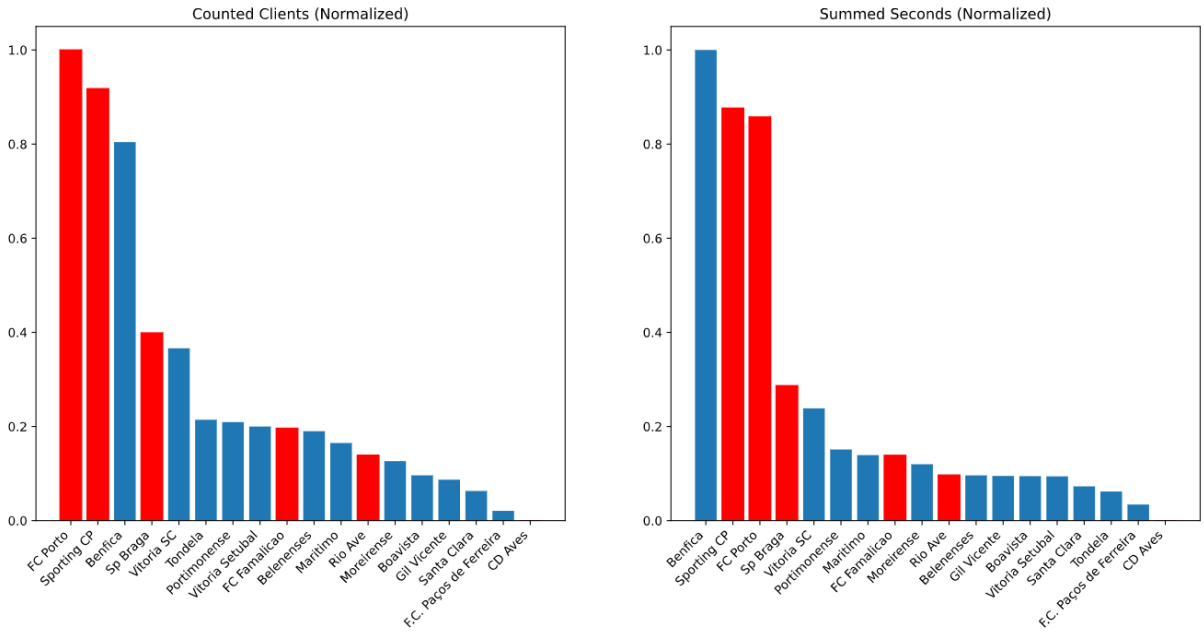


Figure 6.1: TV viewing time and client volume for Liga NOS 19/20 teams. The red bars represent the six teams' case studies selected.

The results of the Granger causality test applied to the different external factors on the

volume a viewing time are presented in table 6.3 and table 6.4, respectively. The results elucidate on two crucial issues. First, when we look at all NOS league matches, it appears that at the significance level of 5%, with the exception of precipitation and the day of the week, all the other external events have predictive power on customer volume and viewing time. Second, the number of external events that have predictive power on customer volume and viewing time is higher in teams with lower customer engagement.

Table 6.3: Granger causality tests (counted_clients).

	All		FC Porto		Sporting CP		Sp Braga		Famalicão FC		Rio Ave		CD Aves	
Null hypothesis	p-value	Result	p-value	Result	p-value	Result	p-value	Result	p-value	Result	p-value	Result	p-value	Result
'probtie' does not Granger cause 'counted_clients'	0.0000	Reject	0.0867	Accept	0.6143	Accept	0.8870	Accept	0.0034	Reject	0.1921	Accept	0.0004	Reject
'b365_d' does not Granger cause 'counted_clients'	0.0001	Reject	0.1024	Accept	0.2384	Accept	0.9503	Accept	0.0466	Reject	0.2165	Accept	0.0019	Reject
'goals_diff_match' does not Granger cause 'counted_clients'	0.0000	Reject	0.0295	Reject	0.1620	Accept	0.7904	Accept	0.0128	Reject	0.0006	Reject	0.0033	Reject
'goals_a_match' does not Granger cause 'counted_clients'	0.0000	Reject	0.7543	Accept	0.7006	Accept	0.8435	Accept	0.0662	Accept	0.0003	Reject	0.1893	Accept
'losses_match' does not Granger cause 'counted_clients'	0.0000	Reject	0.4064	Accept	0.6864	Accept	0.7200	Accept	0.0109	Reject	0.0049	Reject	0.0097	Reject
'rank_match' does not Granger cause 'counted_clients'	0.0000	Reject	0.2069	Accept	0.7721	Accept	0.5389	Accept	0.0755	Accept	0.0114	Reject	0.0004	Reject
'spi_match' does not Granger cause 'counted_clients'	0.0000	Reject	0.0189	Reject	0.1661	Accept	0.1338	Accept	0.0549	Accept	0.0001	Reject	0.0000	Reject
'importance_match' does not Granger cause 'counted_clients'	0.0000	Reject	0.4718	Accept	0.0616	Accept	0.2455	Accept	0.6542	Accept	0.0410	Reject	0.7404	Accept
'counted_news' does not Granger cause 'counted_clients'	0.0000	Reject	0.0712	Reject	0.8655	Accept	0.4417	Accept	0.0074	Reject	0.0101	Reject	0.0080	Reject
'week_interest' does not Granger cause 'counted_clients'	0.0174	Reject	0.0463	Reject	0.0294	Reject	0.4990	Accept	0.6846	Accept	0.1537	Accept	0.2734	Accept
'day_interest' does not Granger cause 'counted_clients'	0.0217	Reject	0.5412	Accept	0.3345	Accept	0.7482	Accept	0.6846	Accept	0.0044	Reject	0.1663	Accept
'hour_interest' does not Granger cause 'counted_clients'	0.0011	Reject	0.8177	Accept	0.1920	Accept	0.7616	Accept	0.6846	Accept	0.1719	Accept	0.8842	Accept
'followers_count_match' does not Granger cause 'counted_clients'	0.0000	Reject	0.9855	Accept	0.0015	Reject	0.0334	Reject	0.0092	Reject	0.0001	Reject	0.0000	Reject
'wind_speed' does not Granger cause 'counted_clients'	0.0406	Reject	0.6981	Accept	0.8237	Accept	0.8694	Accept	0.4784	Accept	0.7426	Accept	0.3271	Accept
'temp' does not Granger cause 'counted_clients'	0.0002	Reject	0.1423	Accept	0.4837	Accept	0.7689	Accept	0.3547	Accept	0.1547	Accept	0.0461	Reject
'precipitation' does not Granger cause 'counted_clients'	0.7665	Accept	0.2098	Accept	0.3977	Accept	0.9242	Accept	0.2605	Accept	0.0460	Reject	0.6049	Accept
'day_of_week' does not Granger cause 'counted_clients'	0.1271	Accept	0.7066	Accept	0.8005	Accept	0.8991	Accept	0.6788	Accept	0.5085	Accept	0.9602	Accept
'hour' does not Granger cause 'counted_clients'	0.0000	Reject	0.0439	Reject	0.4416	Accept	0.3463	Accept	0.9038	Accept	0.0679	Accept	0.0356	Reject
'counted_channels' does not Granger cause 'counted_clients'	0.0000	Reject	0.3661	Accept	0.0751	Accept	0.3118	Accept	0.2214	Accept	0.7996	Accept	0.1803	Accept

Table 6.4: Granger causality tests (summed_seconds).

	All		FC Porto		Sporting CP		Sp Braga		Famalicão FC		Rio Ave		CD Aves	
Null hypothesis	p-value	Result	p-value	Result	p-value	Result	p-value	Result	p-value	Result	p-value	Result	p-value	Result
'probtie' does not Granger cause 'summed_seconds'	0.0000	Reject	0.0182	Reject	0.3254	Accept	0.9719	Accept	0.0018	Reject	0.0317	Reject	0.0008	Reject
'b365_d' does not Granger cause 'summed_seconds'	0.0000	Reject	0.0229	Reject	0.6316	Accept	0.3151	Accept	0.0509	Accept	0.0602	Accept	0.0093	Reject
'goals_diff_match' does not Granger cause 'summed_seconds'	0.0000	Reject	0.0012	Reject	0.1322	Accept	0.0494	Reject	0.0006	Reject	0.0000	Reject	0.0015	Reject
'goals_a_match' does not Granger cause 'summed_seconds'	0.0000	Reject	0.5233	Accept	0.8175	Accept	0.2116	Accept	0.0060	Reject	0.0000	Reject	0.1572	Accept
'losses_match' does not Granger cause 'summed_seconds'	0.0000	Reject	0.2266	Accept	0.5867	Accept	0.1566	Accept	0.0006	Reject	0.0004	Reject	0.0014	Reject
'rank_match' does not Granger cause 'summed_seconds'	0.0000	Reject	0.0574	Accept	0.5626	Accept	0.0879	Accept	0.0031	Reject	0.0012	Reject	0.0007	Reject
'spi_match' does not Granger cause 'summed_seconds'	0.0000	Reject	0.0001	Reject	0.0677	Accept	0.0005	Reject	0.0009	Reject	0.0000	Reject	0.0001	Reject
'importance_match' does not Granger cause 'summed_seconds'	0.0000	Reject	0.4289	Accept	0.0207	Reject	0.0090	Reject	0.8476	Accept	0.0076	Reject	0.9655	Accept
'counted_news' does not Granger cause 'summed_seconds'	0.0000	Reject	0.2665	Accept	0.8224	Accept	0.2365	Accept	0.0048	Reject	0.0013	Reject	0.0266	Reject
'week_interest' does not Granger cause 'summed_seconds'	0.0018	Reject	0.0435	Reject	0.0019	Reject	0.8584	Accept	0.7787	Accept	0.0145	Reject	0.5736	Accept
'day_interest' does not Granger cause 'summed_seconds'	0.0005	Reject	0.2743	Accept	0.5419	Accept	0.7161	Accept	0.7787	Accept	0.0669	Accept	0.5039	Accept
'hour_interest' does not Granger cause 'summed_seconds'	0.0018	Reject	0.4230	Accept	0.3488	Accept	0.7097	Accept	0.7787	Accept	0.2556	Accept	0.8311	Accept
'followers_count_match' does not Granger cause 'summed_seconds'	0.0000	Reject	0.7152	Accept	0.0001	Reject	0.0000	Reject	0.0008	Reject	0.0000	Reject	0.0000	Reject
'wind_speed' does not Granger cause 'summed_seconds'	0.0661	Reject	0.5088	Accept	0.9282	Accept	0.6149	Accept	0.7900	Accept	0.2108	Accept	0.1525	Reject
'temp' does not Granger cause 'summed_seconds'	0.0004	Reject	0.1188	Accept	0.4082	Accept	0.5284	Accept	0.5312	Accept	0.3216	Accept	0.3206	Reject
'precipitation' does not Granger cause 'summed_seconds'	0.8115	Accept	0.0687	Accept	0.0645	Accept	0.9183	Accept	0.7712	Accept	0.0487	Reject	0.1143	Accept
'day_of_week' does not Granger cause 'summed_seconds'	0.0550	Accept	0.2471	Accept	0.7584	Accept	0.3185	Accept	0.5503	Accept	0.6911	Accept	0.3115	Accept
'hour' does not Granger cause 'summed_seconds'	0.0000	Reject	0.0894	Accept	0.2426	Accept	0.6625	Accept	1.0000	Accept	0.0726	Accept	0.1062	Accept
'counted_channels' does not Granger cause 'summed_seconds'	0.0005	Reject	0.9896	Accept	0.4560	Accept	0.8331	Accept	0.4655	Accept	0.8014	Accept	0.7287	Accept

Table 6.5 presents an overview of the causal analysis results: the number of matches analyzed, the number of external events related to TV audience and the factors that affected TV audience, in each case.

Looking at these results in more detail, if we consider the results across the different teams, it can be concluded that:

- **The outcome uncertainty (H1)** mostly affects the sum of seconds. This may indicate that the uncertainty measure has a bigger impact during the game's progression. For example, may be related to games that ended the first half with a tie or the existence of a

Table 6.5: Summary of live matches under analysis.

Teams	Number of live matches under analysis	Number of variables influencing counted clients	Number of variables influencing summed seconds	Factors influencing counted clients	Factors influencing summed seconds
All	283	17	17	outcome uncertainty(2), match quality(6), match interest(4), match popularity(1), scheduling(1), weather(5), TV-network(1)	outcome uncertainty(2), match quality(6), match interest(4), match popularity(1), scheduling(1), weather(4), TV-network(1)
FC Porto	33	5	5	match quality(2), match interest(1), scheduling(1)	outcome uncertainty(2), match quality(2), match interest(1)
Sporting CP	30	2	3	math interest(1), match popularity(1)	match quality(1), math interest(1), match popularity(1)
Sp Braga	27	1	4	match popularity(1)	match quality(3), match popularity(1)
FC Famalicão	18	6	8	outcome uncertainty(2), match quality(2), match interest(1), match popularity(1)	outcome uncertainty(1), match quality(5), match interest(1), match popularity(1)
Rio Ave	30	10	11	match quality(6), match interest(2), match popularity(1) weather(1)	outcome uncertainty(1), match quality(6), match interest(2), match popularity(1), weather(1)
CD Aves	27	10	10	outcome uncertainty(2), match quality(4), match interest(1), match popularity(1), weather(3), scheduling(1)	outcome uncertainty(2), match quality(4), match interest(1), match popularity(1), weather(1)

penalty shootout definition [2];

- **Match quality (H2)** proved to be one of the most important factors, only in the *Sporting CP* and *Sp Braga* customers count no external feature showed any causality effect;
- **Match interest (H3)**, mainly through news counting, has also show to have predictive power over television audiences.
- **Match popularity (H4)**, as in the case of match quality, it also proved to be one of the most important factors in television audiences. Only in the *FC Porto* case no effect was verified;
- **The weather factor (H5)** only in the case of teams with low customer engagement it has shown to have a predictive effect. This may suggest that fans of teams with less engagement do not have as stronger bond with their team and let themselves be carried away by external factors;
- **Scheduling (H6)** showed to have an impact only on the counting of customers of two teams (*FC Porto* and *CD Aves*). These results are not surprising given that we are only considering games from one competition, the game schedule is very similar throughout the league;
- **Channel counting (H7)** proved to be an irrelevant factor in predicting television

audiences. This suggests that when a game is played on more than one channel, people spread across different channels.

6.2.2 Conclusions

This causality analysis sought to examine the relevance of the presence of the external features as a predictor of audience size of sports TV viewership. For this purpose, seven different case studies from Liga NOS (19/20) were analyzed: a general case with all the teams, two big customer engagement teams, two medium customer engagement teams and two low customer engagement teams.

The results showed that, in general, all external events used in this study have a cause-effect relationship on TV viewership. Also, matches of teams with less customer engagement are more likely to be affected by external events. Which can be an indication that supporters of this type of teams do not have a stronger bond with their teams and are much more influenced by external events.

Furthermore, the match quality, match interest, match popularity and outcome uncertainty have shown to be the external factors most closely related to variations on television audiences.

Finally, the scheduling of a program and the channel counting did not produce relevant effects on television audiences.

Chapter 7

Conclusion

We present in this dissertation (1) a pipeline for the generation of external data related to live football matches, (2) a multivariate machine learning model that uses the external data as input to predict sportscast customer counts, and (3) a quantitative and qualitative characterizations (through a SHAP and Granger causality analysis) of the relationships between external events and television audiences.

The results showed a high connection rate of external data and EPG metadata data, a multivariate approach with better accuracy in sports broadcast prediction than classical approaches, and a large number of events detected as having a cause-and-effect relationship in sports television audiences. Our findings further support the usefulness of online data to understand television behaviors. Data sources such as news, twitter API and Google trends proved to be of great value in predicting television audiences. That said, we can affirm that the main goals proposed for this work were achieved.

7.1 Future work

Finally, some limitations should be noted, which also suggest future research directions. First, although a large amount of real-world events have been extracted, improvements are still possible. For example, anomaly detection can be used to determine whether a particular spike in data refers to a specific television event (e.g., a national holiday). This information can be included in the data to detect new peaks.

Second, our main focus is not to achieve the best possible model accuracy, but rather to analyze the effect of external features on forecast accuracy, for better results the application of hyper-parameter tuning can be of great value.

Third, and last, the causality analysis results are based on bivariate time series models. Future research should investigate causal relationships between external real-world events and TV viewing patterns within a multivariate Granger causality framework. In addition, the use

of a different causal discovery framework can bring more confidence in the obtained results. A possible alternative to Granger causality test is PCMCIplus [93], an conditional independence (CI) based method for linear and nonlinear, lagged and contemporaneous causal discovery from observational time series.

Appendix A

Appendix

A.1 Data description

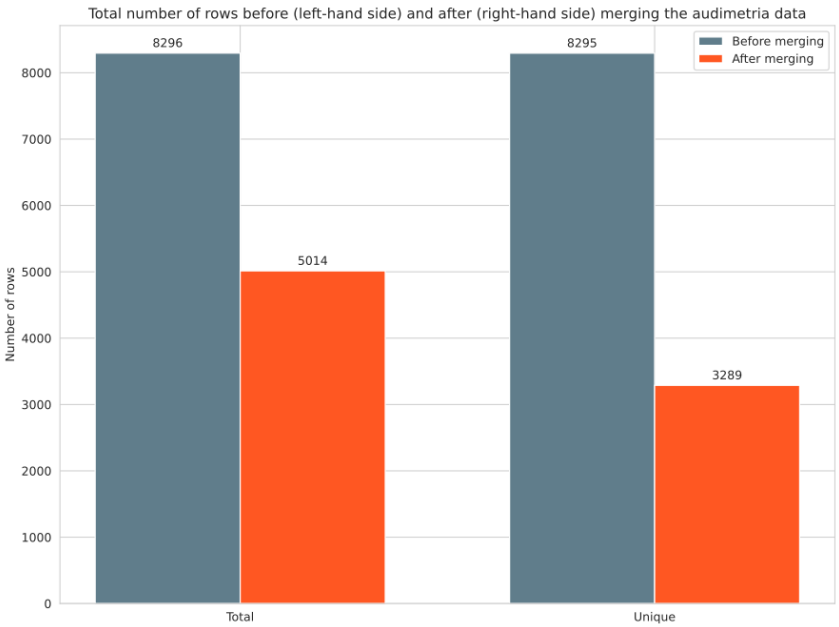


Figure A.1: Total number of matches before (left-hand side) and after (right-hand side) merging the TV data.

Figure A.2: Total number of matchs before (left-hand side) and after (right-hand side) merging the audimetria data.

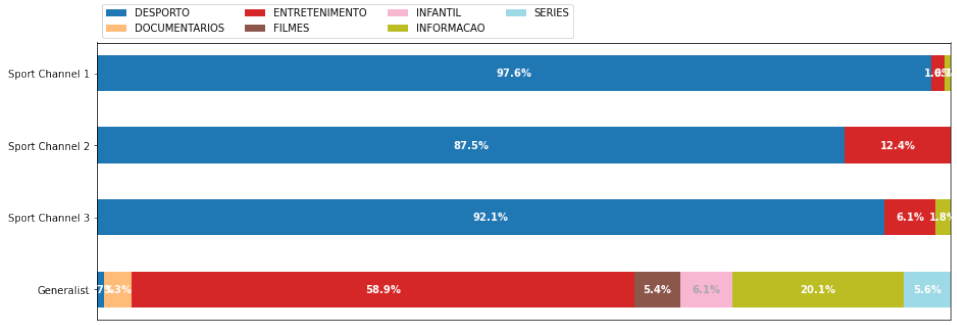


Figure A.3: Number of rows for each category (percentage).

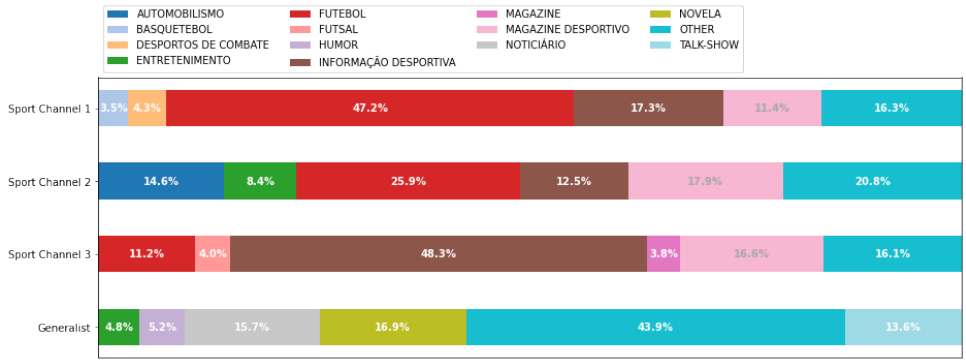


Figure A.4: Number of rows for each genre (percentage).

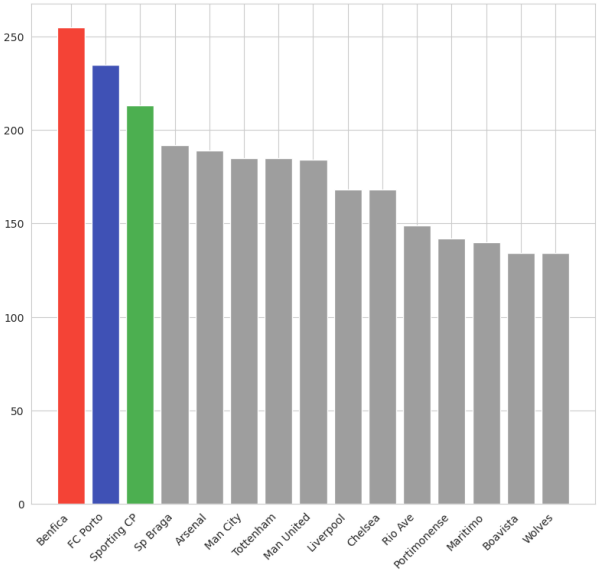


Figure A.5: Number of football matches per team.

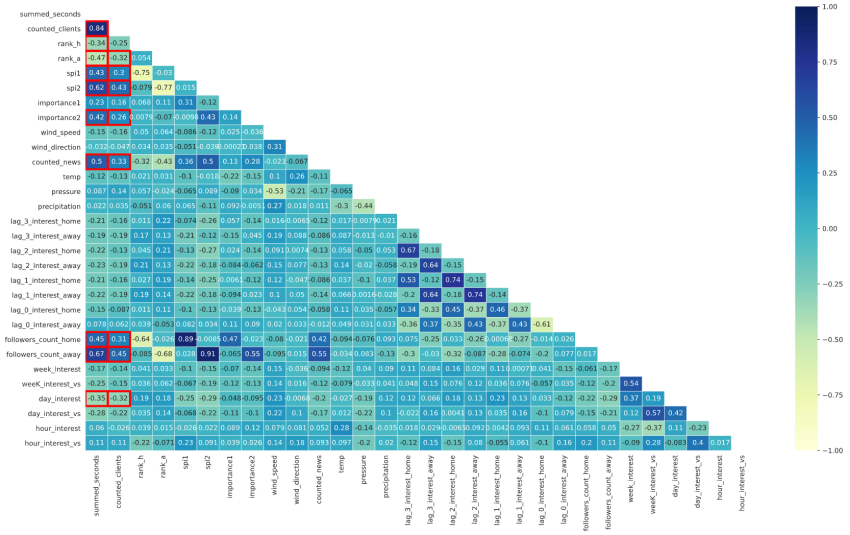


Figure A.6: Liga NOS correlation matrix.

A.2 Forecasting

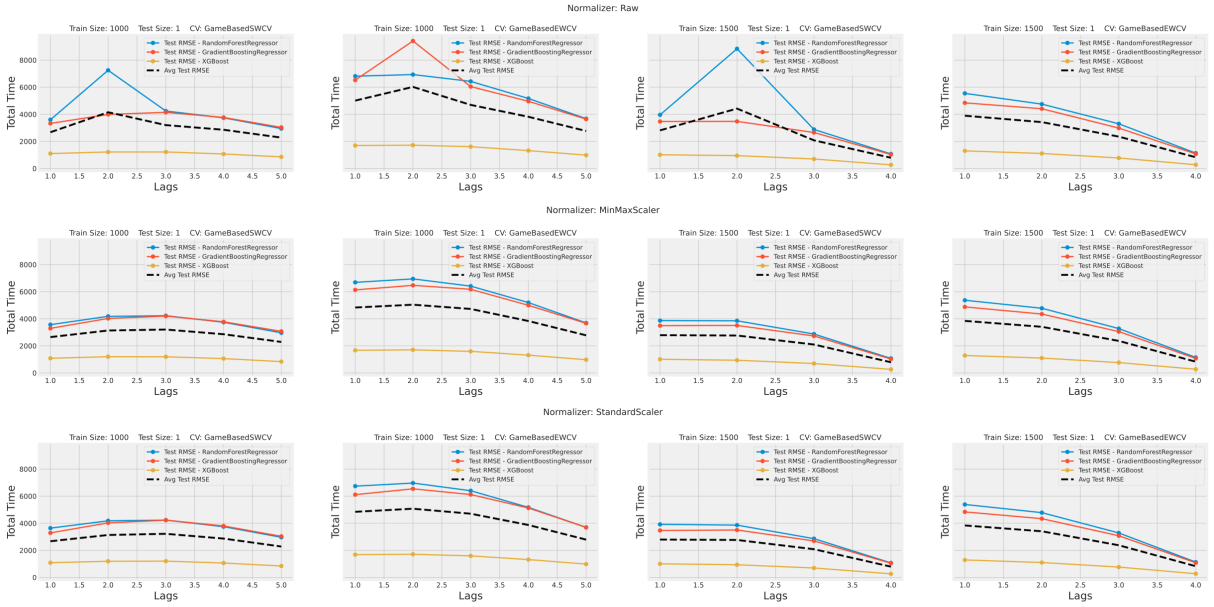


Figure A.7: Best Number of lags - Total time.

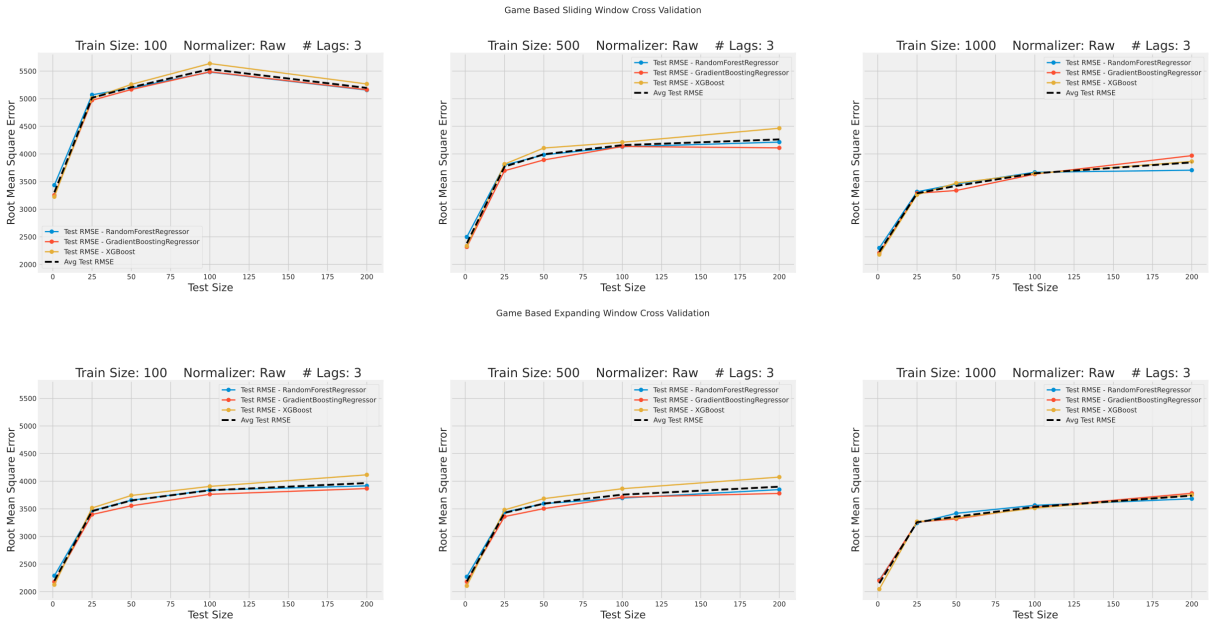


Figure A.8: Best test size - RMSE.

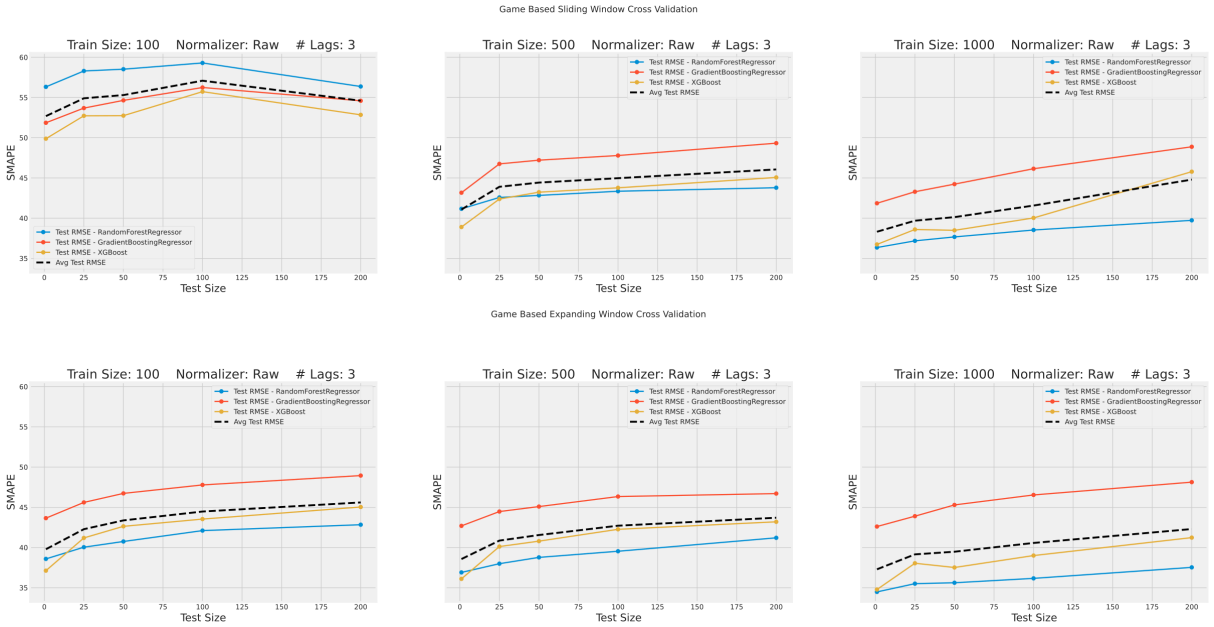


Figure A.9: Best test size - SMAPE.

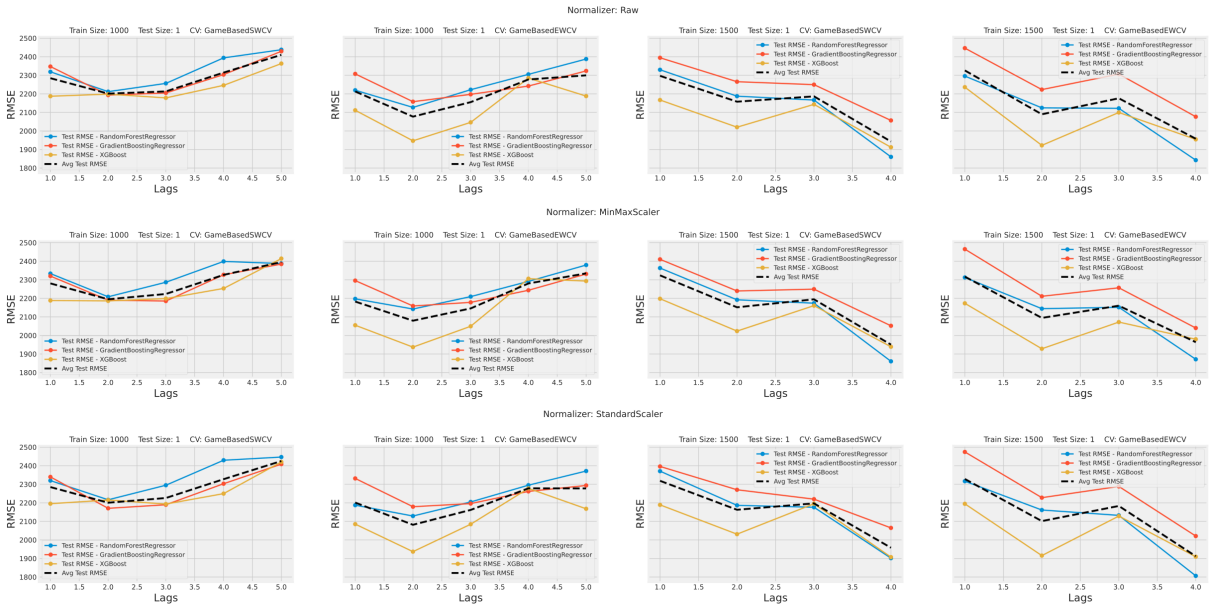


Figure A.10: Best number of lags - RMSE.

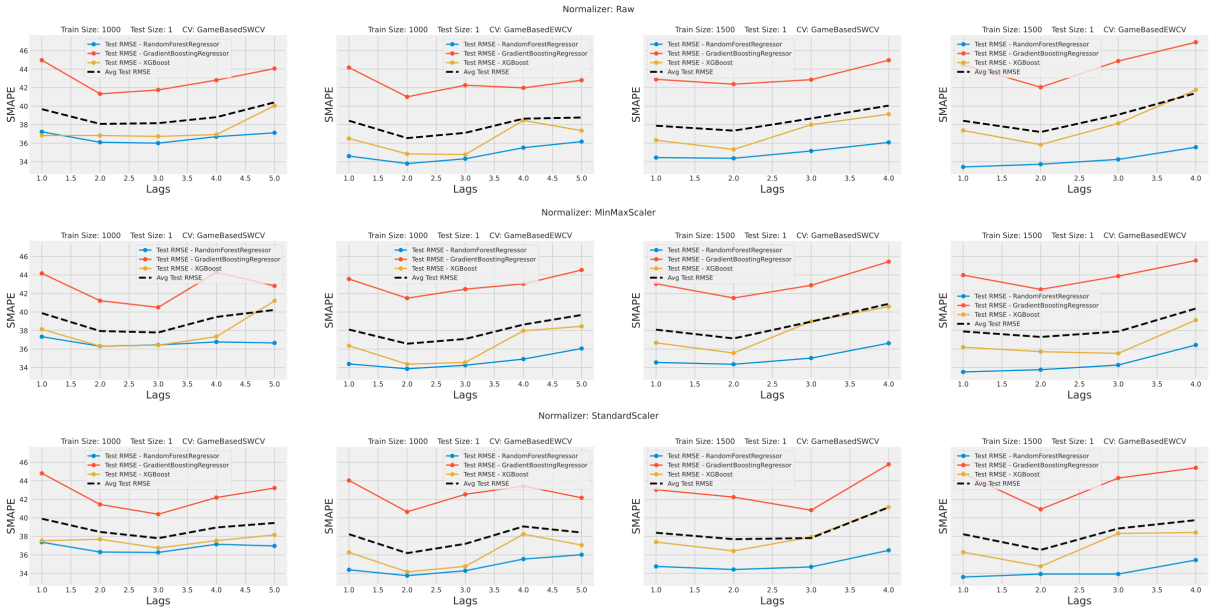


Figure A.11: Best number of lags - SMAPE.

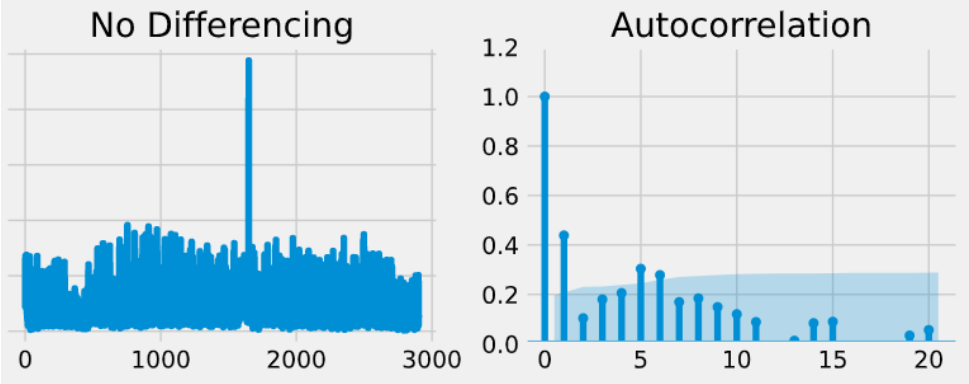


Figure A.12: ACF - All data tournaments.

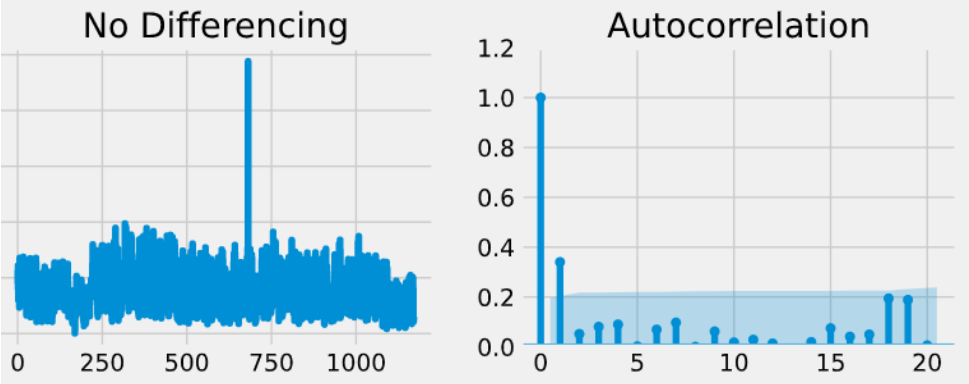


Figure A.13: ACF - PT data tournaments.

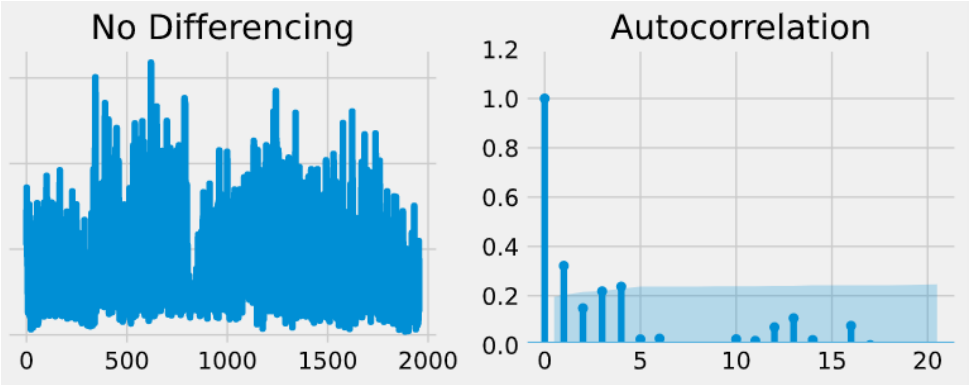


Figure A.14: ACF - INT data tournaments.

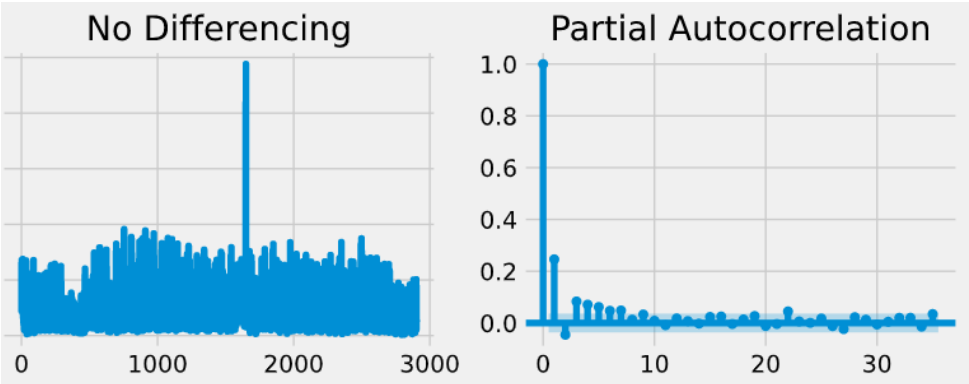


Figure A.15: PACF - All data tournaments.

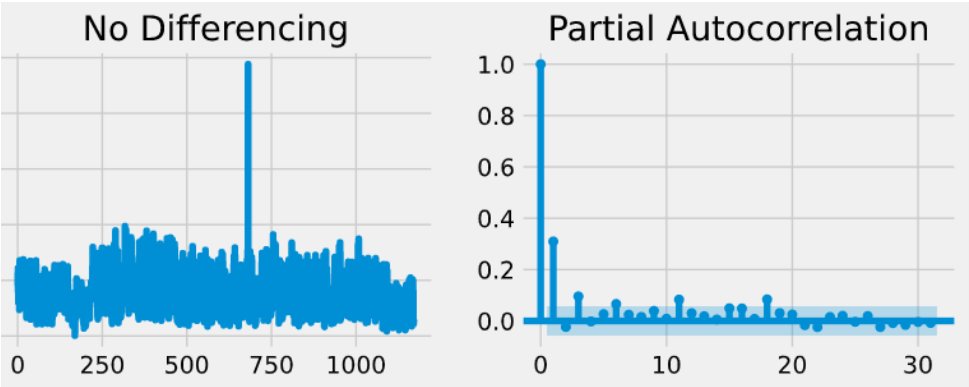


Figure A.16: PACF - PT data tournaments.

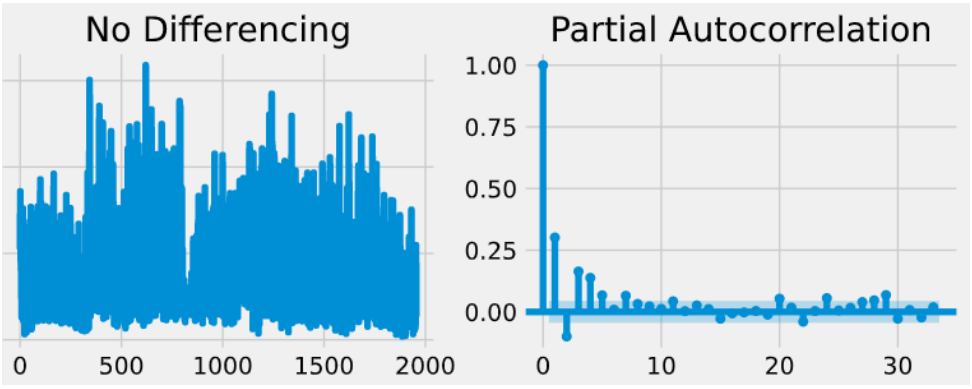


Figure A.17: PACF - INT data tournaments.

Table A.1: ML Otimization 1.

model	cv	normalizer	nr_lags	train_size	test_size	average_rmse	average_smape	total_time
RandomForestRegressor	GameBasedSWCV	Raw	3	1000	200	3706.0498960365385	39.7302459453925	54.11983680725098
RandomForestRegressor	GameBasedSWCV	Raw	3	1000	100	3668.289500891574	38.529127362601024	107.9681453704834
RandomForestRegressor	GameBasedSWCV	Raw	3	1000	50	3455.3675048554305	37.66949976425352	226.47514200210568
RandomForestRegressor	GameBasedSWCV	Raw	3	1000	25	3314.7309485633677	37.180082040711795	453.9058928489685
RandomForestRegressor	GameBasedSWCV	Raw	3	1000	1	2298.412108294931	36.34892079992363	12040.131406068802
RandomForestRegressor	GameBasedSWCV	Raw	3	500	200	4214.342340965056	43.78670715315725	33.9584527015686
RandomForestRegressor	GameBasedSWCV	Raw	3	500	100	4130.461484916516	43.34843046862709	75.33814311027527
RandomForestRegressor	GameBasedSWCV	Raw	3	500	50	3983.437215968483	42.83163736767815	155.1508195400238
RandomForestRegressor	GameBasedSWCV	Raw	3	500	25	3798.424670753127	42.575482792988	307.45371174812317
RandomForestRegressor	GameBasedSWCV	Raw	3	500	1	2497.724071637427	41.16600761063873	7765.464665889741
RandomForestRegressor	GameBasedSWCV	Raw	3	100	200	5155.891500504162	56.37158551192106	6.685309410095215
RandomForestRegressor	GameBasedSWCV	Raw	3	100	100	5481.380351708985	59.29821806863805	14.048025608062744
RandomForestRegressor	GameBasedSWCV	Raw	3	100	50	5194.77540564189	58.523197430937	28.86410927772522
RandomForestRegressor	GameBasedSWCV	Raw	3	100	25	5070.1456875768545	58.300544669834004	57.84924030303955
RandomForestRegressor	GameBasedSWCV	Raw	3	100	1	3434.845627828054	56.327596440576755	1464.6191947460175
RandomForestRegressor	GameBasedEWCV	Raw	3	1000	200	3682.667747373743	37.529730211629456	73.65422272682191
RandomForestRegressor	GameBasedEWCV	Raw	3	1000	100	3563.3199770099573	36.162704509895775	154.84128379821775
RandomForestRegressor	GameBasedEWCV	Raw	3	1000	50	3419.0352754836103	35.616715288141855	340.0500769615173
RandomForestRegressor	GameBasedEWCV	Raw	3	1000	25	3242.876275157204	35.50390997455579	683.9615490436554
RandomForestRegressor	GameBasedEWCV	Raw	3	1000	1	2211.0198847926267	34.48011240244559	18087.566040039066
RandomForestRegressor	GameBasedEWCV	Raw	3	500	200	3848.46834318046	41.202571895724475	80.06620025634766
RandomForestRegressor	GameBasedEWCV	Raw	3	500	100	3695.0292116667574	39.533228700747046	194.50869631767281
RandomForestRegressor	GameBasedEWCV	Raw	3	500	50	3594.558160509824	38.77050580304678	424.2103862762451
RandomForestRegressor	GameBasedEWCV	Raw	3	500	25	3439.316200412324	37.98811176078539	853.4885265827179
RandomForestRegressor	GameBasedEWCV	Raw	3	500	1	2270.289692982456	36.9013057562199	22294.21680045128
RandomForestRegressor	GameBasedEWCV	Raw	3	100	200	3916.1849803247374	42.833328120775136	83.77978825569153
RandomForestRegressor	GameBasedEWCV	Raw	3	100	100	3838.90443224637	42.100409770783756	203.3322696685791
RandomForestRegressor	GameBasedEWCV	Raw	3	100	50	3656.2712042808225	40.746542826326326	443.44845700263977
RandomForestRegressor	GameBasedEWCV	Raw	3	100	25	3464.1536898759755	40.0376511519279	904.1711683273317
RandomForestRegressor	GameBasedEWCV	Raw	3	100	1	2289.0508597285066	38.58246165403375	23445.678170681
GradientBoostingRegressor	GameBasedSWCV	Raw	3	1000	200	3969.4714822230817	48.87422717352109	18.988256931304928
GradientBoostingRegressor	GameBasedSWCV	Raw	3	1000	100	3634.3989220133744	46.14523452112621	38.002966642379754
GradientBoostingRegressor	GameBasedSWCV	Raw	3	1000	50	3338.643002314909	44.23184474900579	81.24786543846129
GradientBoostingRegressor	GameBasedSWCV	Raw	3	1000	25	3287.756011561851	43.284069777693325	161.205002784729
GradientBoostingRegressor	GameBasedSWCV	Raw	3	1000	1	2207.6149657455003	41.85411243160705	4148.813942909241
GradientBoostingRegressor	GameBasedSWCV	Raw	3	500	200	4109.774254891519	49.3225489291841	14.321879863739015
GradientBoostingRegressor	GameBasedSWCV	Raw	3	500	100	4135.166720691699	47.78845503919351	30.5623836517334
GradientBoostingRegressor	GameBasedSWCV	Raw	3	500	50	3891.294393952353	47.21014091141185	63.02297592163086
GradientBoostingRegressor	GameBasedSWCV	Raw	3	500	25	3696.5205322504994	46.747616703222675	127.48227405548096
GradientBoostingRegressor	GameBasedSWCV	Raw	3	500	1	2316.540069096487	43.16493120704754	3236.487483024597
GradientBoostingRegressor	GameBasedSWCV	Raw	3	100	200	5162.05857896951	54.594869545555184	4.075878381729127
GradientBoostingRegressor	GameBasedSWCV	Raw	3	100	100	5487.177830942675	56.24679141799687	8.58015775680542
GradientBoostingRegressor	GameBasedSWCV	Raw	3	100	50	5166.608052645603	54.64687550587631	17.390042781829838
GradientBoostingRegressor	GameBasedSWCV	Raw	3	100	25	4971.582061315213	53.687418995733005	34.39503502845764
GradientBoostingRegressor	GameBasedSWCV	Raw	3	100	1	3253.2599528825126	51.85948319098275	872.6290094852448
GradientBoostingRegressor	GameBasedEWCV	Raw	3	1000	200	3782.0349313916295	48.1281373399335	25.99597454071045
GradientBoostingRegressor	GameBasedEWCV	Raw	3	1000	100	3528.4028288563977	46.53507339455989	53.0957190990448
GradientBoostingRegressor	GameBasedEWCV	Raw	3	1000	50	3318.750223039333	45.29383126036858	115.60547280311584
GradientBoostingRegressor	GameBasedEWCV	Raw	3	1000	25	3262.1111990483405	43.893278190443105	234.07537865638733
GradientBoostingRegressor	GameBasedEWCV	Raw	3	1000	1	2200.3767162253635	42.60565773791	6009.481504201888
GradientBoostingRegressor	GameBasedEWCV	Raw	3	500	200	3779.614458802546	46.70463344300247	29.295054197311398
GradientBoostingRegressor	GameBasedEWCV	Raw	3	500	100	3710.57198888166	46.33627073515704	68.3689341545105
GradientBoostingRegressor	GameBasedEWCV	Raw	3	500	50	3504.093919835091	45.08854095428536	150.38038182258606
GradientBoostingRegressor	GameBasedEWCV	Raw	3	500	25	3359.868617121325	44.4696084076346	304.9062716960907
GradientBoostingRegressor	GameBasedEWCV	Raw	3	500	1	2180.957175569972	42.68831537140807	7836.913321495056
GradientBoostingRegressor	GameBasedEWCV	Raw	3	100	200	3867.2938443101643	48.941893852950066	31.029099464416504
GradientBoostingRegressor	GameBasedEWCV	Raw	3	100	100	3763.1039717501217	47.782485670381725	73.03226733207704
GradientBoostingRegressor	GameBasedEWCV	Raw	3	100	50	3555.836491293848	46.72806730480952	162.02217626571658
GradientBoostingRegressor	GameBasedEWCV	Raw	3	100	25	3397.379659507557	45.605712806854655	325.99882078170776
GradientBoostingRegressor	GameBasedEWCV	Raw	3	100	1	2177.9264688707026	43.647867586280384	8516.618428230286
XGBoost	GameBasedSWCV	Raw	3	1000	200	3863.781407756306	45.77917285069972	47.39369654655457
XGBoost	GameBasedSWCV	Raw	3	1000	100	3646.328722943777	40.03057807089064	55.62584376335144
XGBoost	GameBasedSWCV	Raw	3	1000	50	3472.332625401592	38.48680570838649	101.32467436790466
XGBoost	GameBasedSWCV	Raw	3	1000	25	3260.877691298597	38.593527872865636	168.47873330116272
XGBoost	GameBasedSWCV	Raw	3	1000	1	2178.2903594970708	36.735563208564095	4777.061422586441
XGBoost	GameBasedSWCV	Raw	3	500	200	4465.299498225081	45.06312735136294	32.457359790802
XGBoost	GameBasedSWCV	Raw	3	500	100	4211.573377353277	43.77383866950175	72.66784286499023
XGBoost	GameBasedSWCV	Raw	3	500	50	4107.02538430189	43.23099199540097	140.8762502670288
XGBoost	GameBasedSWCV	Raw	3	500	25	3815.934383136729	42.38201722923958	279.73216104507446
XGBoost	GameBasedSWCV	Raw	3	500	1	2333.6945932343688	38.89722492176225	6609.552387714386
XGBoost	GameBasedSWCV	Raw	3	100	200	5266.383388117785	52.84760417404097	16.23098063468933
XGBoost	GameBasedSWCV	Raw	3	100	100	5636.289448769456	55.731963166159105	36.9958176612854
XGBoost	GameBasedSWCV	Raw	3	100	50	5258.846832853888	52.74561811936102	74.95019102096558
XGBoost	GameBasedSWCV	Raw	3	100	25	5009.893012603822	52.72673479232557	140.69506287574768

Table A.2: ML Optimization 2.

model	cv	normalizer	nr_lags	train_size	test_size	average_rmse	average_smape	total_time
XGBoost	GameBasedSWCV	Raw	3	100	1	3225.2557044741247	49.8813203101933	3609.530901193619
XGBoost	GameBasedEWCV	Raw	3	1000	200	3750.850880119569	41.22885957451293	19.963659286499023
XGBoost	GameBasedEWCV	Raw	3	1000	100	3513.071201073665	38.99853499960307	39.423140287399285
XGBoost	GameBasedEWCV	Raw	3	1000	50	3338.017232706784	37.51394799087824	101.31896686553956
XGBoost	GameBasedEWCV	Raw	3	1000	25	3268.6800635056165	38.03934436475061	211.11594986915588
XGBoost	GameBasedEWCV	Raw	3	1000	1	2046.3976667020727	34.776526193821375	4894.0226640701285
XGBoost	GameBasedEWCV	Raw	3	500	200	4074.783518094183	43.193116720594006	27.89009380340576
XGBoost	GameBasedEWCV	Raw	3	500	100	3864.7653079431057	42.26130413769675	58.731938600540154
XGBoost	GameBasedEWCV	Raw	3	500	50	3684.710181737436	40.78930096286434	147.4188561439514
XGBoost	GameBasedEWCV	Raw	3	500	25	3481.731290619347	40.11742242476265	284.4610757827759
XGBoost	GameBasedEWCV	Raw	3	500	1	2106.1129453889807	36.10060906764353	7001.348330259322
XGBoost	GameBasedEWCV	Raw	3	100	200	4115.945705221503	45.032049762068176	38.45863986015321
XGBoost	GameBasedEWCV	Raw	3	100	100	3906.0669034433736	43.53242410007149	75.54423213005066
XGBoost	GameBasedEWCV	Raw	3	100	50	3742.093090125649	42.63148936548725	157.58381962776184
XGBoost	GameBasedEWCV	Raw	3	100	25	3516.117374850889	41.18967099228936	298.5614049434662
XGBoost	GameBasedEWCV	Raw	3	100	1	2126.0134331022987	37.119872747550176	7502.74641776085
RandomForestRegressor	GameBasedSWCV	Raw	1	1000	1	2318.844217894096	37.2273062053145	3599.034618377685
RandomForestRegressor	GameBasedSWCV	Raw	1	1500	1	2329.850944881889	34.447132580862224	3958.798049926758
RandomForestRegressor	GameBasedEWCV	Raw	1	1000	1	2219.4942544126598	34.60876288102262	6814.964159250258
RandomForestRegressor	GameBasedEWCV	Raw	1	1500	1	2295.717734033246	33.43139959524121	5545.685058116913
GradientBoostingRegressor	GameBasedSWCV	Raw	1	1000	1	2347.2675831492293	44.971272771051524	3327.034697532654
GradientBoostingRegressor	GameBasedSWCV	Raw	1	1500	1	2394.5458040694125	42.899642960455054	3469.250072479248
GradientBoostingRegressor	GameBasedEWCV	Raw	1	1000	1	2307.5718983600696	44.171913736336144	6525.2578592300415
GradientBoostingRegressor	GameBasedEWCV	Raw	1	1500	1	2446.438535680814	44.46339465974699	4844.091957569121
XGBoost	GameBasedSWCV	Raw	1	1000	1	2187.2308384000853	36.83687022532297	1100.5306572914124
XGBoost	GameBasedSWCV	Raw	1	1500	1	2166.830516559872	36.323230440918906	1014.6342906951904
XGBoost	GameBasedEWCV	Raw	1	1000	1	2111.5356690957638	36.50563714438999	1692.596221446991
XGBoost	GameBasedEWCV	Raw	1	1500	1	2236.150109639005	37.372751596176855	1299.4606153964994
RandomForestRegressor	GameBasedSWCV	Raw	2	1000	1	2211.926260575296	36.10074231119805	7248.769802570343
RandomForestRegressor	GameBasedSWCV	Raw	2	1500	1	2187.525410557185	34.37060319777106	8839.340185403824
RandomForestRegressor	GameBasedEWCV	Raw	2	1000	1	2126.9419543147205	33.80110821491891	6933.495932102203
RandomForestRegressor	GameBasedEWCV	Raw	2	1500	1	2124.5655131964813	33.731383603297736	4751.2728168964395
GradientBoostingRegressor	GameBasedSWCV	Raw	2	1000	1	2193.3741121203184	41.33259739817113	3989.8834688663483
GradientBoostingRegressor	GameBasedSWCV	Raw	2	1500	1	2265.4970370749465	42.377174429319524	3477.119590759277
GradientBoostingRegressor	GameBasedEWCV	Raw	2	1000	1	2157.8839560030774	41.01135284147993	9413.28144645691
GradientBoostingRegressor	GameBasedEWCV	Raw	2	1500	1	2223.035929953081	42.0492241505691	4409.2675235271445
XGBoost	GameBasedSWCV	Raw	2	1000	1	2198.4955064842948	36.83103099620195	1219.0861229896545
XGBoost	GameBasedSWCV	Raw	2	1500	1	2020.3374980062333	35.32573605401454	948.181412935257
XGBoost	GameBasedEWCV	Raw	2	1000	1	1946.8922367039472	34.84645443365595	1721.405157327652
XGBoost	GameBasedEWCV	Raw	2	1500	1	1922.389783880228	35.84476642921196	1108.5054922103882
RandomForestRegressor	GameBasedSWCV	Raw	3	1000	1	2256.4411405529954	36.006961952053	4241.555945873261
RandomForestRegressor	GameBasedSWCV	Raw	3	1500	1	2166.675434782609	35.160904785141	2880.6214141845703
RandomForestRegressor	GameBasedEWCV	Raw	3	1000	1	2222.4645276497695	34.31799211480069	6429.641725778579
RandomForestRegressor	GameBasedEWCV	Raw	3	1500	1	2122.082282608696	34.25178509360545	3298.3995435237885
GradientBoostingRegressor	GameBasedSWCV	Raw	3	1000	1	2204.8513407511123	41.75696280928624	4145.254520654677
GradientBoostingRegressor	GameBasedSWCV	Raw	3	1500	1	2249.6359791019345	42.86568590300865	2654.418081998825
GradientBoostingRegressor	GameBasedEWCV	Raw	3	1000	1	2197.382554649236	42.267875476786855	6045.91342806816
GradientBoostingRegressor	GameBasedEWCV	Raw	3	1500	1	2306.6717163864714	44.872385193910894	2983.467916965485
XGBoost	GameBasedSWCV	Raw	3	1000	1	2178.2903594970708	36.735563208564095	1219.3828699588776
XGBoost	GameBasedSWCV	Raw	3	1500	1	2143.3759700396786	37.99842260991666	699.8271613121033
XGBoost	GameBasedEWCV	Raw	3	1000	1	2046.3976667020727	34.776526193821375	1609.4011120796204
XGBoost	GameBasedEWCV	Raw	3	1500	1	2099.603337607954	38.13654983450874	774.7586960792543
RandomForestRegressor	GameBasedSWCV	Raw	4	1000	1	2394.067086743044	36.70503498907944	3737.586535215378
RandomForestRegressor	GameBasedSWCV	Raw	4	1500	1	1860.2328828828827	36.08702088445203	1072.405715227127
RandomForestRegressor	GameBasedEWCV	Raw	4	1000	1	2305.191129296236	35.51595954334872	5166.855430841446
RandomForestRegressor	GameBasedEWCV	Raw	4	1500	1	1842.665855855856	35.56622963980941	1132.2231962680814
GradientBoostingRegressor	GameBasedSWCV	Raw	4	1000	1	2303.9191024764123	42.81261906487772	3766.469014406204
GradientBoostingRegressor	GameBasedSWCV	Raw	4	1500	1	2056.719405699793	44.96703460393751	1033.7150120735166
GradientBoostingRegressor	GameBasedEWCV	Raw	4	1000	1	2242.132516534592	41.97885892575295	4959.297845363617
GradientBoostingRegressor	GameBasedEWCV	Raw	4	1500	1	2076.663494422083	46.90711585696148	1072.3307764530182
XGBoost	GameBasedSWCV	Raw	4	1000	1	2245.943138376976	36.92979079296786	1071.3803193569183
XGBoost	GameBasedSWCV	Raw	4	1500	1	1912.185145747554	39.139749063955215	266.98941540718084
XGBoost	GameBasedEWCV	Raw	4	1000	1	2285.180293051974	38.46170710124408	1320.623999834061
XGBoost	GameBasedEWCV	Raw	4	1500	1	1954.6809245960133	41.75096119059626	276.0710322856903
RandomForestRegressor	GameBasedSWCV	Raw	5	1000	1	2437.982375	37.1264738927781	2946.8899490833282
RandomForestRegressor	GameBasedEWCV	Raw	5	1000	1	2387.7054000000007	36.17327852215854	3676.536447763443
GradientBoostingRegressor	GameBasedSWCV	Raw	5	1000	1	2429.476208985436	44.06614984080062	3033.673850774765
GradientBoostingRegressor	GameBasedSWCV	Raw	5	1000	1	2323.7163581708296	42.803359257289685	3643.17679810524
XGBoost	GameBasedSWCV	Raw	5	1000	1	2363.262372816801	40.05258119977466	855.6241252422333
XGBoost	GameBasedEWCV	Raw	5	1000	1	2187.9537487339967	37.35658001819323	983.0889019966124
RandomForestRegressor	GameBasedSWCV	MinMaxScaler	1	1000	1	2333.4502678027998	37.33043393835542	3565.5427017211914
RandomForestRegressor	GameBasedSWCV	MinMaxScaler	1	1500	1	2363.243272090989	34.55554192081193	3866.550012588501
RandomForestRegressor	GameBasedEWCV	MinMaxScaler	1	1000	1	2197.15256238588	34.39212105092966	6690.454813957214
RandomForestRegressor	GameBasedEWCV	MinMaxScaler	1	1500	1	2312.47154855643	33.528978912846306	5367.857131481172
GradientBoostingRegressor	GameBasedSWCV	MinMaxScaler	1	1000	1	2319.926375156827	44.17171819382777	3293.7797474861145

Table A.3: ML Optimization 3.

model	cv	normalizer	nr_lags	train_size	test_size	average_rmse	average_smape	total_time
GradientBoostingRegressor	GameBasedSWCV	MinMaxScaler	1	1500	1	2410.319133840038	43.051812168078506	3492.2310972213745
GradientBoostingRegressor	GameBasedEWCV	MinMaxScaler	1	1000	1	2295.6200968466587	43.5658557794429	6132.707284688951
GradientBoostingRegressor	GameBasedEWCV	MinMaxScaler	1	1500	1	2465.1536171061207	43.984631269392004	4877.285618305206
XGBoost	GameBasedSWCV	MinMaxScaler	1	1000	1	2188.365427702139	38.13481893645784	1086.6356797218325
XGBoost	GameBasedSWCV	MinMaxScaler	1	1500	1	2197.8509407702604	36.67009210921329	1010.0395185947418
XGBoost	GameBasedEWCV	MinMaxScaler	1	1000	1	2055.019395512484	36.360139040105906	1673.4600839614868
XGBoost	GameBasedEWCV	MinMaxScaler	1	1500	1	2173.1308299412563	36.19949753930434	1289.5096921920774
RandomForestRegressor	GameBasedSWCV	MinMaxScaler	2	1000	1	2208.395592216583	36.30659997471644	4180.3569502830505
RandomForestRegressor	GameBasedSWCV	MinMaxScaler	2	1500	1	2192.0843695014664	34.353153113505805	3850.632691383362
RandomForestRegressor	GameBasedEWCV	MinMaxScaler	2	1000	1	2142.445456852792	33.86819900518506	6945.276827335358
RandomForestRegressor	GameBasedEWCV	MinMaxScaler	2	1500	1	2144.3209237536653	33.77007549880525	4776.2260060310355
GradientBoostingRegressor	GameBasedSWCV	MinMaxScaler	2	1000	1	2190.2985730895043	41.22053785727324	4025.312801599503
GradientBoostingRegressor	GameBasedSWCV	MinMaxScaler	2	1500	1	2239.7762034600737	41.51424071626386	3507.409679889679
GradientBoostingRegressor	GameBasedEWCV	MinMaxScaler	2	1000	1	2158.858222493764	41.4941544298386	6469.391637802124
GradientBoostingRegressor	GameBasedEWCV	MinMaxScaler	2	1500	1	2211.133484530295	42.44217680541336	4348.428189516068
XGBoost	GameBasedSWCV	MinMaxScaler	2	1000	1	2186.23327777188	36.31388937950688	1201.42777967453
XGBoost	GameBasedSWCV	MinMaxScaler	2	1500	1	2023.4619434921624	35.57158666175898	939.358984708786
XGBoost	GameBasedEWCV	MinMaxScaler	2	1000	1	1937.0982236870248	34.36719450733297	1706.4073662757874
XGBoost	GameBasedEWCV	MinMaxScaler	2	1500	1	1928.2078679328088	35.713966811267255	1099.8832330703738
RandomForestRegressor	GameBasedSWCV	MinMaxScaler	3	1000	1	2286.726900921659	36.446449935585974	4224.441247463225
RandomForestRegressor	GameBasedSWCV	MinMaxScaler	3	1500	1	2173.91597826087	35.021738360121304	2872.768575668335
RandomForestRegressor	GameBasedEWCV	MinMaxScaler	3	1000	1	2209.371947004608	34.25246592109258	6409.599452018738
RandomForestRegressor	GameBasedEWCV	MinMaxScaler	3	1500	1	2151.887092391304	34.27505434113574	3276.436587333679
GradientBoostingRegressor	GameBasedSWCV	MinMaxScaler	3	1000	1	2185.432998234849	40.50439412930528	4196.924350023271
GradientBoostingRegressor	GameBasedSWCV	MinMaxScaler	3	1500	1	2249.521232094019	42.88625435375951	2729.768505573273
GradientBoostingRegressor	GameBasedEWCV	MinMaxScaler	3	1000	1	2178.349593578136	42.471104178609494	6177.137480020522
GradientBoostingRegressor	GameBasedEWCV	MinMaxScaler	3	1500	1	2256.8825159549724	43.87691794526553	3053.8535475730896
XGBoost	GameBasedSWCV	MinMaxScaler	3	1000	1	2198.6965124044	36.413734465654436	1194.0440948009489
XGBoost	GameBasedSWCV	MinMaxScaler	3	1500	1	2160.749551814536	38.98033414045487	696.3106682300568
XGBoost	GameBasedEWCV	MinMaxScaler	3	1000	1	2049.434075110588	34.55500607634965	1594.839802980423
XGBoost	GameBasedEWCV	MinMaxScaler	3	1500	1	2072.0564904655125	35.53362286184222	766.2456450462341
RandomForestRegressor	GameBasedSWCV	MinMaxScaler	4	1000	1	2399.043371522095	36.76851510836936	3740.952360868454
RandomForestRegressor	GameBasedSWCV	MinMaxScaler	4	1500	1	1860.4732432432431	36.6309199865274	1074.9881489276886
RandomForestRegressor	GameBasedEWCV	MinMaxScaler	4	1000	1	2291.4388870703765	34.916471005105855	5195.764693737029
RandomForestRegressor	GameBasedEWCV	MinMaxScaler	4	1500	1	1871.4516216216214	36.43573040367989	1144.7403745651245
GradientBoostingRegressor	GameBasedSWCV	MinMaxScaler	4	1000	1	2327.011396016789	44.295160449955304	3777.786843776703
GradientBoostingRegressor	GameBasedSWCV	MinMaxScaler	4	1500	1	2052.022267017961	45.4360268284212	1028.492398262024
GradientBoostingRegressor	GameBasedEWCV	MinMaxScaler	4	1000	1	2243.572365627223	43.0442573619892	4993.198346138001
GradientBoostingRegressor	GameBasedEWCV	MinMaxScaler	4	1500	1	2040.077460845171 1	45.56226187576114	1073.3818678855896
XGBoost	GameBasedSWCV	MinMaxScaler	4	1000	1	2253.413465833898	37.33895437668914	1061.0233399868014
XGBoost	GameBasedSWCV	MinMaxScaler	4	1500	1	1938.910794679109	40.58635336564635	264.6457359790802
XGBoost	GameBasedEWCV	MinMaxScaler	4	1000	1	2305.8784161956337	37.978082592160895	1313.9850635528564
XGBoost	GameBasedEWCV	MinMaxScaler	4	1500	1	1979.4742423392638	39.12590338700152	272.5810582637787
RandomForestRegressor	GameBasedSWCV	MinMaxScaler	5	1000	1	2387.4708750000004	36.658675875906695	2963.280436754227
RandomForestRegressor	GameBasedEWCV	MinMaxScaler	5	1000	1	2379.60905	36.0573331651631	3692.1636216640472
GradientBoostingRegressor	GameBasedSWCV	MinMaxScaler	5	1000	1	2384.795294429161	42.823763334918105	3070.666026353836
GradientBoostingRegressor	GameBasedSWCV	MinMaxScaler	5	1000	1	2331.267483113555	44.53910036010842	3663.337689161301
XGBoost	GameBasedSWCV	MinMaxScaler	5	1000	1	2414.327536644936	41.20128397732574	836.9485085010529
XGBoost	GameBasedEWCV	MinMaxScaler	5	1000	1	2293.1688221740724	38.453825267140736	978.1710772514343
RandomForestRegressor	GameBasedSWCV	StandardScaler	1	1000	1	2319.795934266586	37.365128411698606	3641.300252199173
RandomForestRegressor	GameBasedSWCV	StandardScaler	1	1500	1	2370.695748031497	34.75326601026014	3917.67414522171
RandomForestRegressor	GameBasedEWCV	StandardScaler	1	1000	1	2187.2995313451	34.38861943578919	6741.242140054705
RandomForestRegressor	GameBasedEWCV	StandardScaler	1	1500	1	2316.577716535433	33.608865539207144	5387.13155412674
GradientBoostingRegressor	GameBasedSWCV	StandardScaler	1	1000	1	2339.6442874253808	44.81152478812725	3290.5484726428986
GradientBoostingRegressor	GameBasedSWCV	StandardScaler	1	1500	1	2396.075876024571	43.03433008023989	3468.1598551273346
GradientBoostingRegressor	GameBasedEWCV	StandardScaler	1	1000	1	2331.395010639295	44.039444350448456	6113.321495771407
GradientBoostingRegressor	GameBasedEWCV	StandardScaler	1	1500	1	2474.247051815736	44.7922750579721	4838.380365610124
XGBoost	GameBasedSWCV	StandardScaler	1	1000	1	2195.3223032123487	37.530431339125336	1086.6644024848938
XGBoost	GameBasedSWCV	StandardScaler	1	1500	1	2188.9759029006077	37.38099928908629	1005.9918255805968
XGBoost	GameBasedEWCV	StandardScaler	1	1000	1	2085.1458799416746	36.26730507608007	1683.3263010978699
XGBoost	GameBasedEWCV	StandardScaler	1	1500	1	2194.589288993561	36.29311999864078	1291.1696856021879
RandomForestRegressor	GameBasedSWCV	StandardScaler	2	1000	1	2216.328790186125	36.30725980335009	4181.044786930084
RandomForestRegressor	GameBasedSWCV	StandardScaler	2	1500	1	2186.391979472141	34.419234038826076	3862.7343163490295
RandomForestRegressor	GameBasedEWCV	StandardScaler	2	1000	1	2128.885922165821	33.75575730521062	6967.796255350114
RandomForestRegressor	GameBasedEWCV	StandardScaler	2	1500	1	2161.2294574780053	33.929855081869306	4783.639132261276
GradientBoostingRegressor	GameBasedSWCV	StandardScaler	2	1000	1	2170.3879967709295	41.44397899039246	4029.446592330933
GradientBoostingRegressor	GameBasedSWCV	StandardScaler	2	1500	1	2270.360715172616	42.23864947305904	3503.219154119492
GradientBoostingRegressor	GameBasedEWCV	StandardScaler	2	1000	1	2179.041904439825	40.649691779271215	6543.149601697924
GradientBoostingRegressor	GameBasedEWCV	StandardScaler	2	1500	1	2227.8134558390752	40.923084802300174	4338.5649394989005
XGBoost	GameBasedSWCV	StandardScaler	2	1000	1	2215.1034030381797	37.68532408144394	1195.0971713066099
XGBoost	GameBasedSWCV	StandardScaler	2	1500	1	2031.0318718180279	36.42439277076746	937.179381608963
XGBoost	GameBasedEWCV	StandardScaler	2	1000	1	1936.8250033867541	34.16630755827482	1711.952437877655
XGBoost	GameBasedEWCV	StandardScaler	2	1500	1	1915.526115585283	34.76286745858516	1102.7084839344022
RandomForestRegressor	GameBasedSWCV	StandardScaler	3	1000	1	2294.893870967742	36.26368352757309	4232.045913696289
RandomForestRegressor	GameBasedSWCV	StandardScaler	3	1500	1	2176.0991847826085	34.701492747117435	2863.2938861846924

Table A.4: ML Optimization 4.

model	cv	normalizer	nr_lags	train_size	test_size	average_rmse	average_smape	total_time
RandomForestRegressor	GameBasedEWCV	StandardScaler	3	1000	1	2204.7046428571425	34.287379660461674	6402.415554523468
RandomForestRegressor	GameBasedEWCV	StandardScaler	3	1500	1	2132.401440217392	33.933130682633696	3284.289610862732
GradientBoostingRegressor	GameBasedSWCV	StandardScaler	3	1000	1	2189.6472003102667	40.3888098889645	4228.144110202788
GradientBoostingRegressor	GameBasedSWCV	StandardScaler	3	1500	1	2219.767436960677	40.826844012638134	2696.5973682403564
GradientBoostingRegressor	GameBasedEWCV	StandardScaler	3	1000	1	2195.922797097557	42.540665147884546	6120.381167650225
GradientBoostingRegressor	GameBasedEWCV	StandardScaler	3	1500	1	2288.414561856821	44.30031705890666	3077.322019815445
XGBoost	GameBasedSWCV	StandardScaler	3	1000	1	2193.372548519741	36.75005463594066	1204.3154654502866
XGBoost	GameBasedSWCV	StandardScaler	3	1500	1	2195.0892051302867	37.93219541727224	697.434014081955
XGBoost	GameBasedEWCV	StandardScaler	3	1000	1	2084.584971047766	34.75744069704436	1594.775661945343
XGBoost	GameBasedEWCV	StandardScaler	3	1500	1	2128.6044655105343	38.31779679227408	766.4617712497711
RandomForestRegressor	GameBasedSWCV	StandardScaler	4	1000	1	2429.547381342062	37.14687381169315	3745.8853721618652
RandomForestRegressor	GameBasedSWCV	StandardScaler	4	1500	1	1901.6402702702703	36.49279415702447	1072.7347722053528
RandomForestRegressor	GameBasedEWCV	StandardScaler	4	1000	1	2295.1282160392802	35.547432707516194	5170.691815614699
RandomForestRegressor	GameBasedEWCV	StandardScaler	4	1500	1	1806.297207207208	35.43967816302241	1132.4108135700224
GradientBoostingRegressor	GameBasedSWCV	StandardScaler	4	1000	1	2303.018880562769	42.19342901734023	3798.990427732468
GradientBoostingRegressor	GameBasedSWCV	StandardScaler	4	1500	1	2064.837015984069	45.77768461606592	1048.15900182724
GradientBoostingRegressor	GameBasedEWCV	StandardScaler	4	1000	1	2261.552289409047	43.43654284260303	5125.759004831314
GradientBoostingRegressor	GameBasedEWCV	StandardScaler	4	1500	1	2020.8419898446411	45.40342299629504	1080.7909457683563
XGBoost	GameBasedSWCV	StandardScaler	4	1000	1	2249.6273933693938	37.535262457687104	1066.058252096176
XGBoost	GameBasedSWCV	StandardScaler	4	1500	1	1907.2297458476846	41.15325490045501	266.3869683742523
XGBoost	GameBasedEWCV	StandardScaler	4	1000	1	2277.6446705925878	38.23297341745223	1319.1018443107605
XGBoost	GameBasedEWCV	StandardScaler	4	1500	1	1909.5570430583787	38.41278527947679	273.60980558395386
RandomForestRegressor	GameBasedSWCV	StandardScaler	5	1000	1	2446.752575	36.96383171674293	2964.639957666397
RandomForestRegressor	GameBasedEWCV	StandardScaler	5	1000	1	2371.332425	36.0180220601322	3696.570827722549
GradientBoostingRegressor	GameBasedSWCV	StandardScaler	5	1000	1	2408.7536183793945	43.220644516087425	3035.998885154724
GradientBoostingRegressor	GameBasedEWCV	StandardScaler	5	1000	1	2293.577607085484	42.164086029421405	3692.504406929016
XGBoost	GameBasedSWCV	StandardScaler	5	1000	1	2420.3551104545595	38.15603942367722	844.6354706287384
XGBoost	GameBasedEWCV	StandardScaler	5	1000	1	2167.8439937114717	37.05300265547635	986.7595613002777

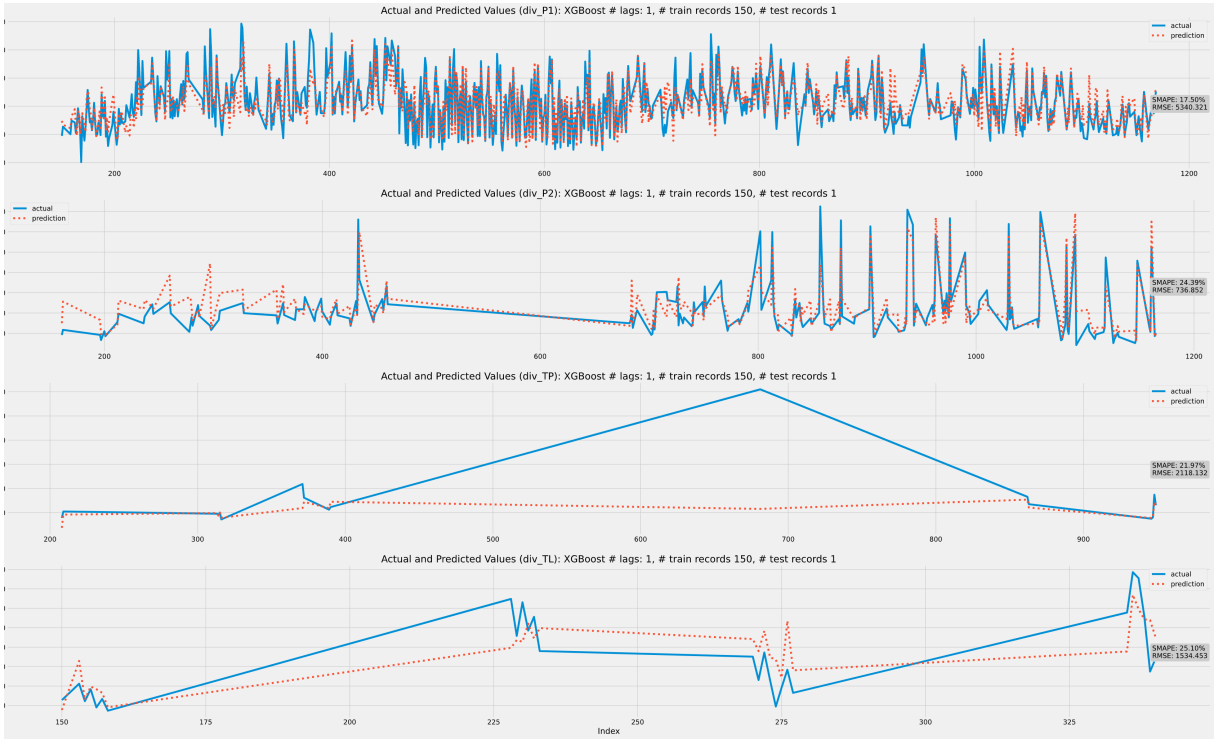


Figure A.18: Portuguese competitions prediction.

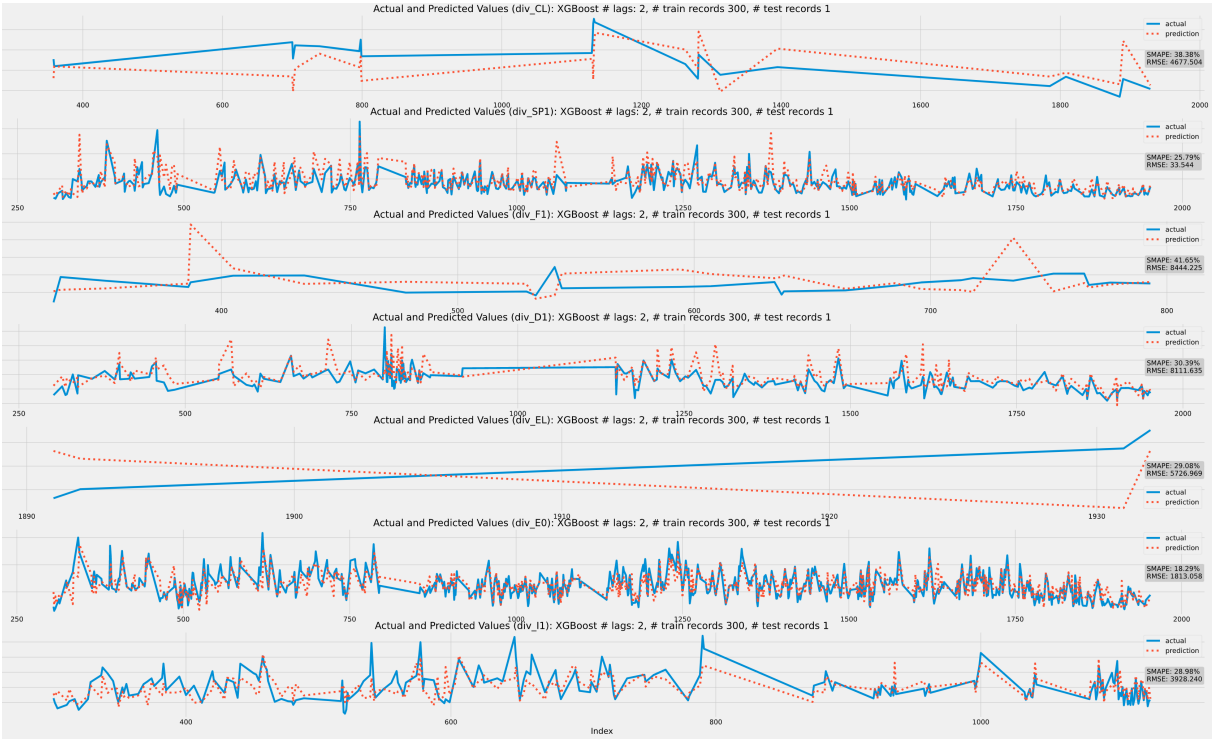


Figure A.19: International competitions prediction.

Figure A.20: Total time.

A.3 Case Study for Liga NOS.

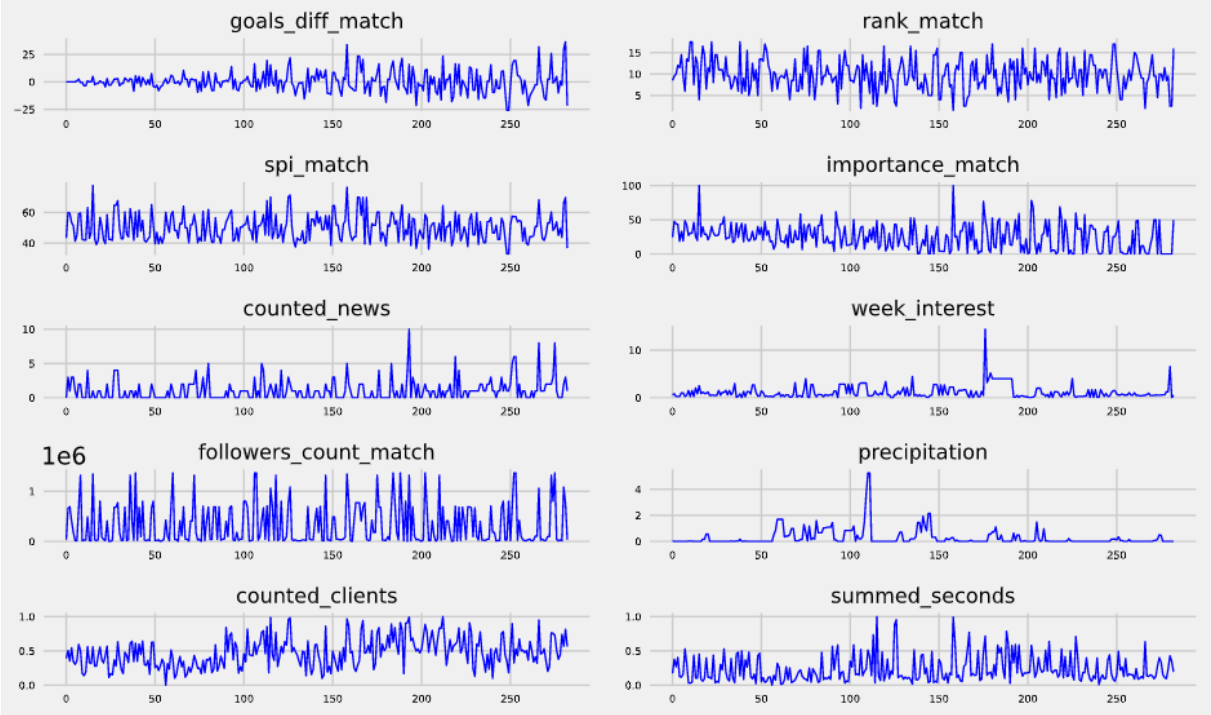


Figure A.21: Liga NOS time series plot.

A.4 Case Study for FC Porto

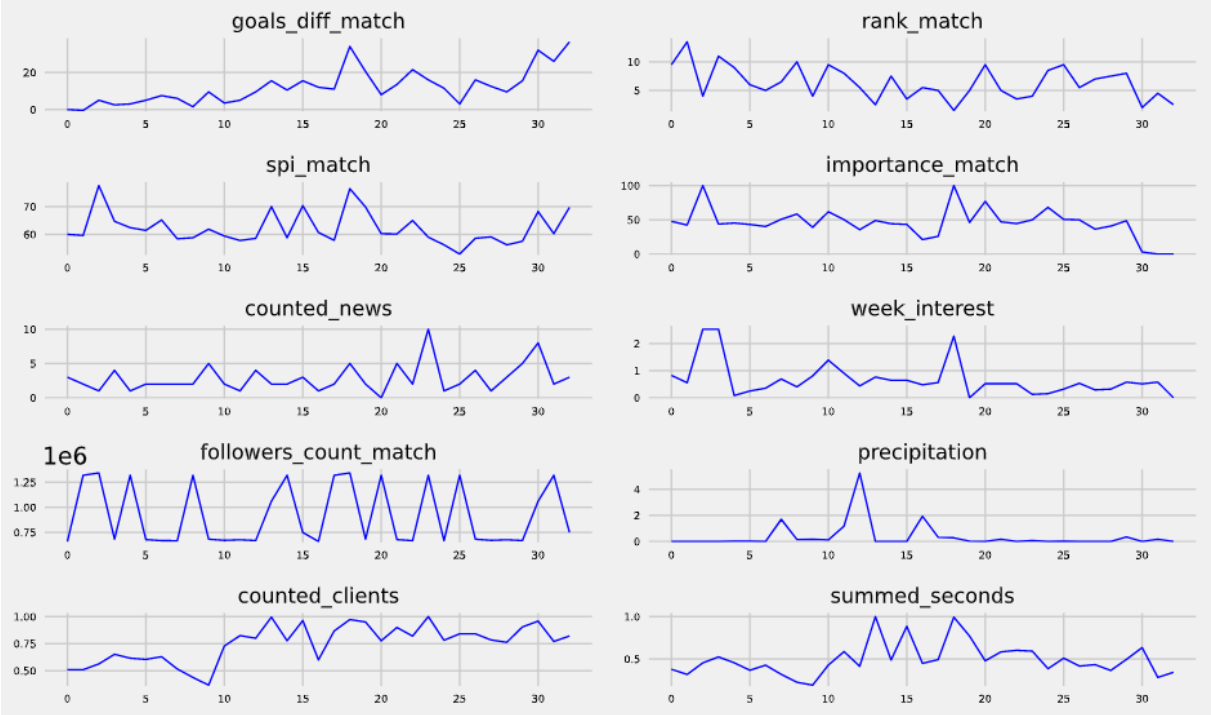


Figure A.22: FC Porto time series plot.

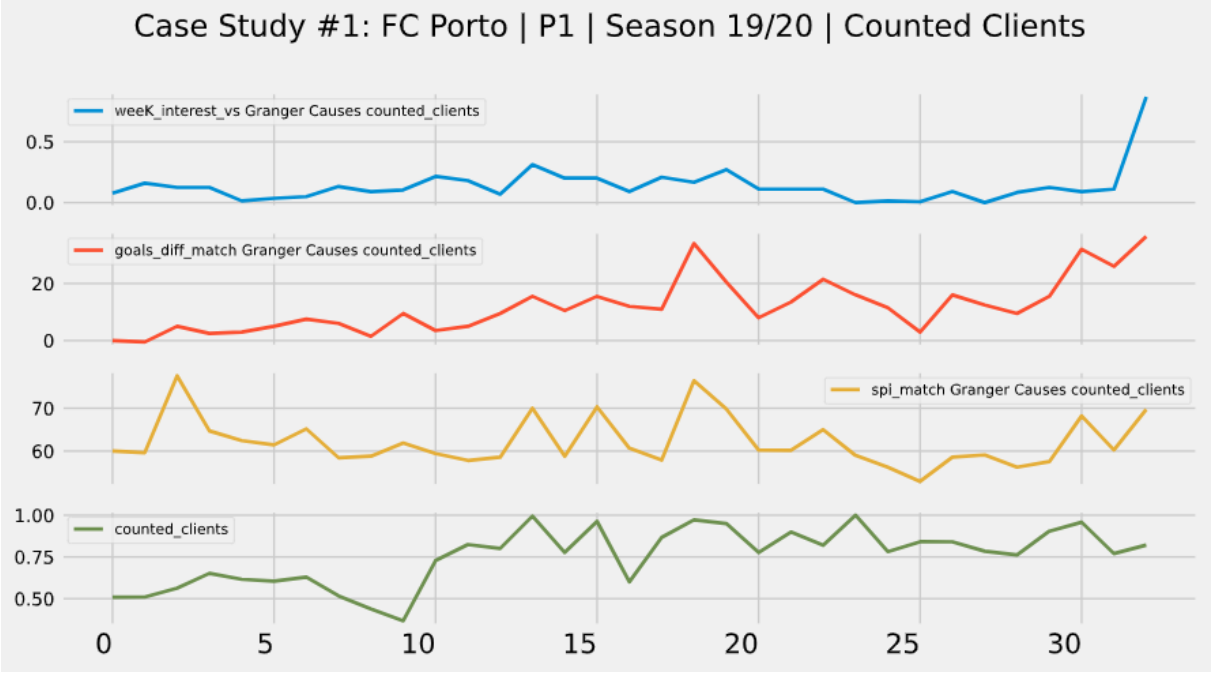


Figure A.23: Granger Causality FC Porto - Counted Clients.

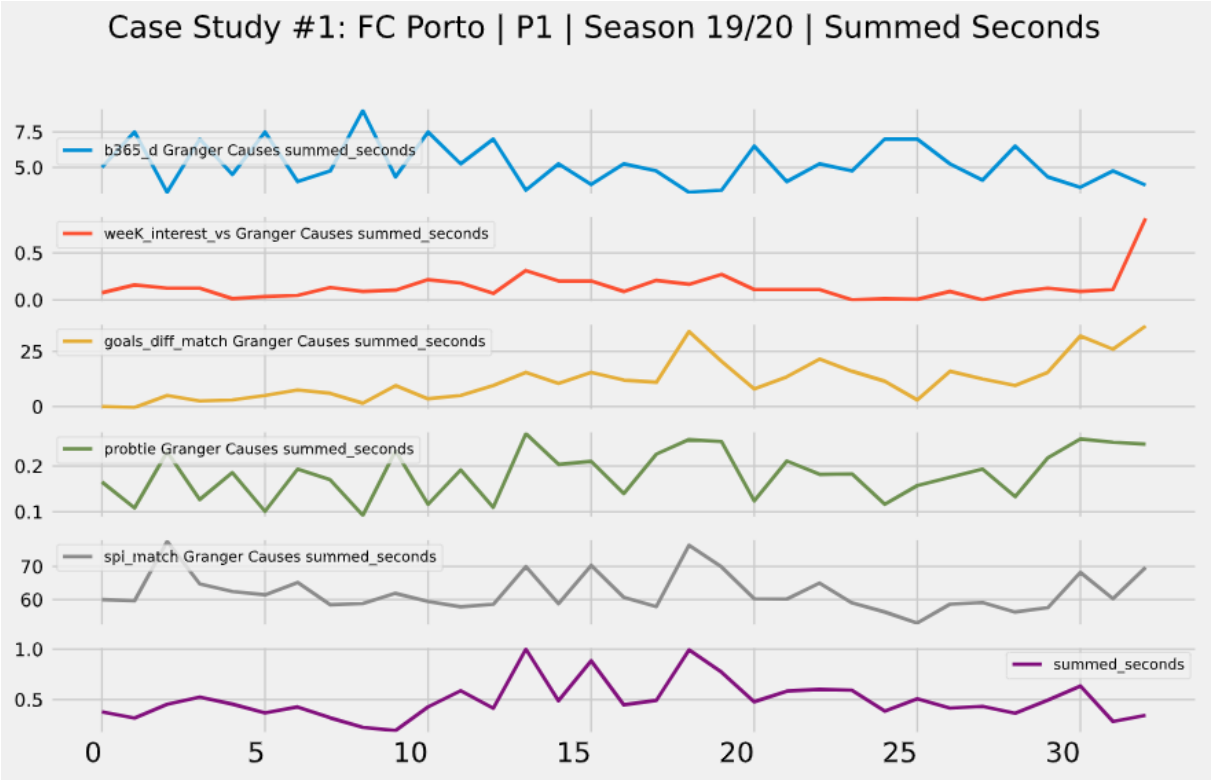


Figure A.24: Granger Causality FC Porto - Summed Seconds.

A.5 Case Study for Sporting CP

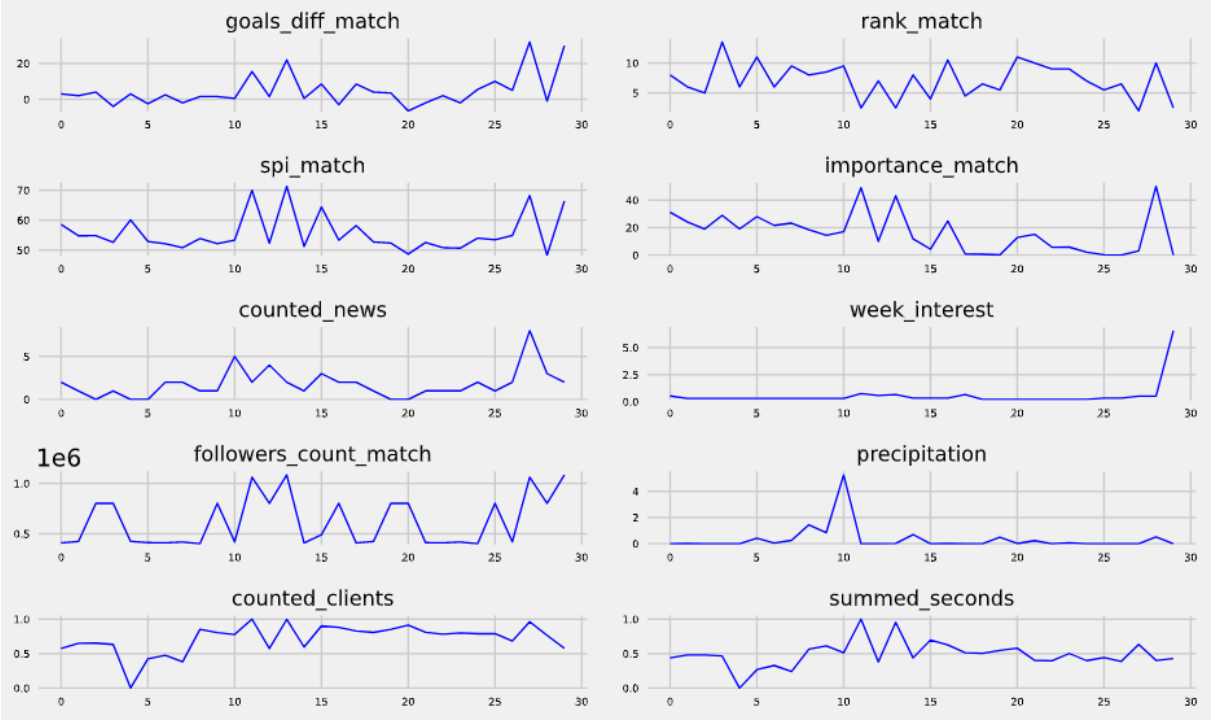


Figure A.25: Sporting CP time series plot.

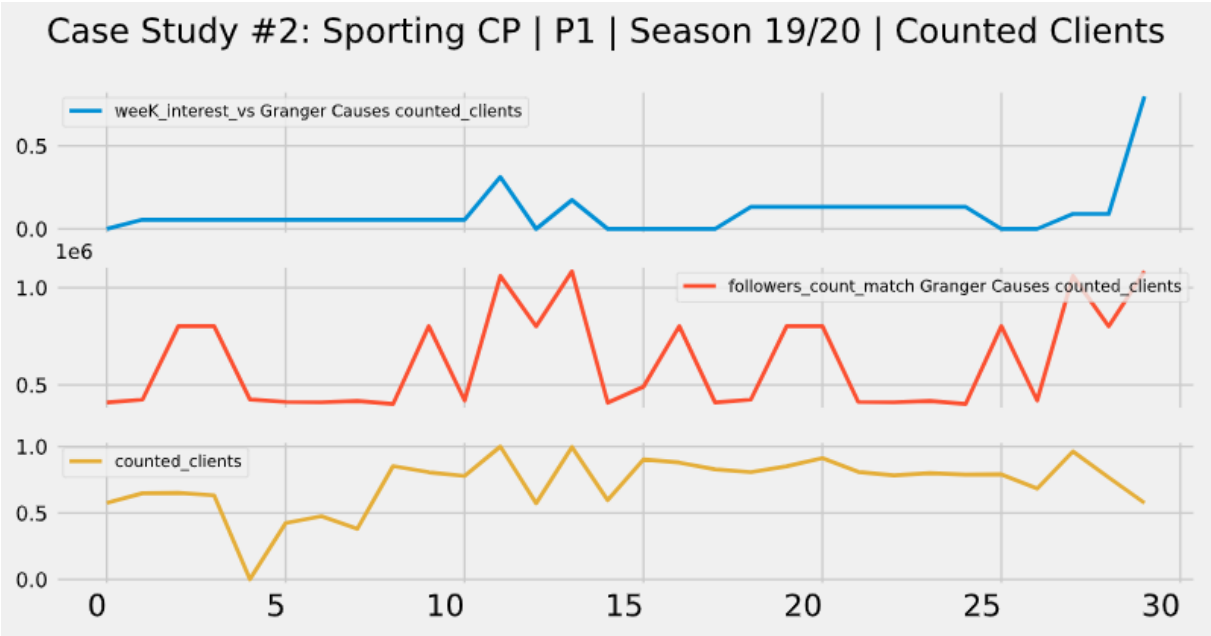


Figure A.26: Granger Causality Sporting CP - Counted Clients.

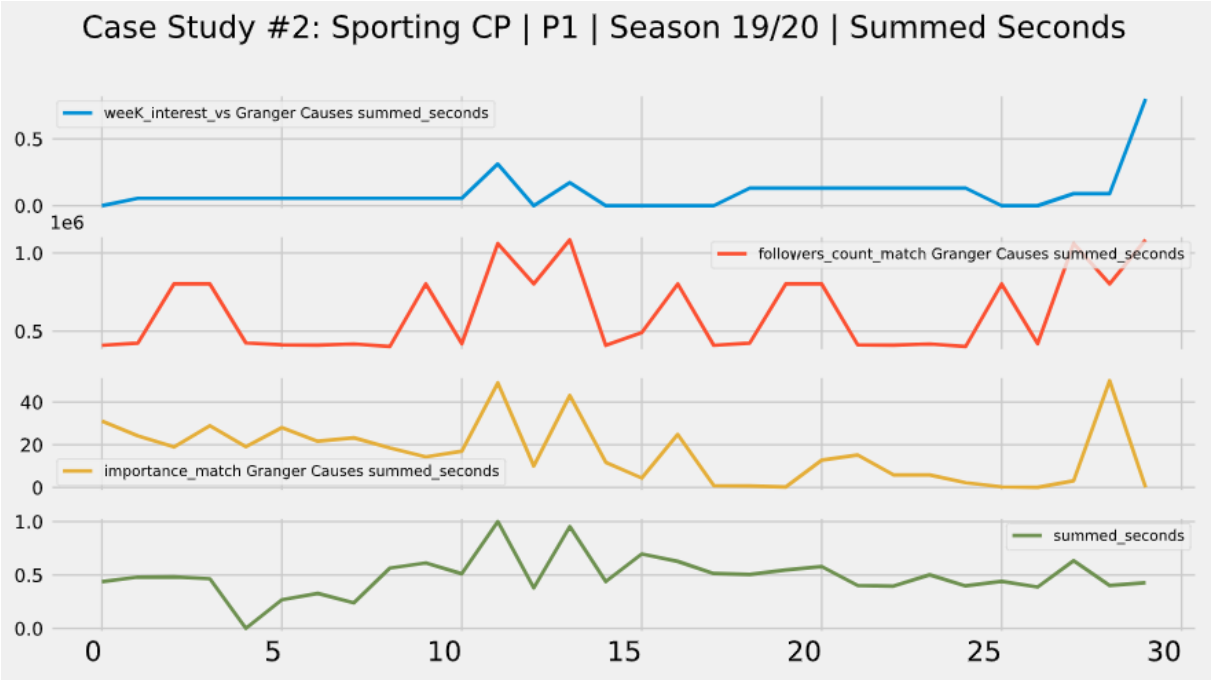


Figure A.27: Granger Causality Sporting CP - Summed Seconds.

A.6 Case study for Famalicão FC

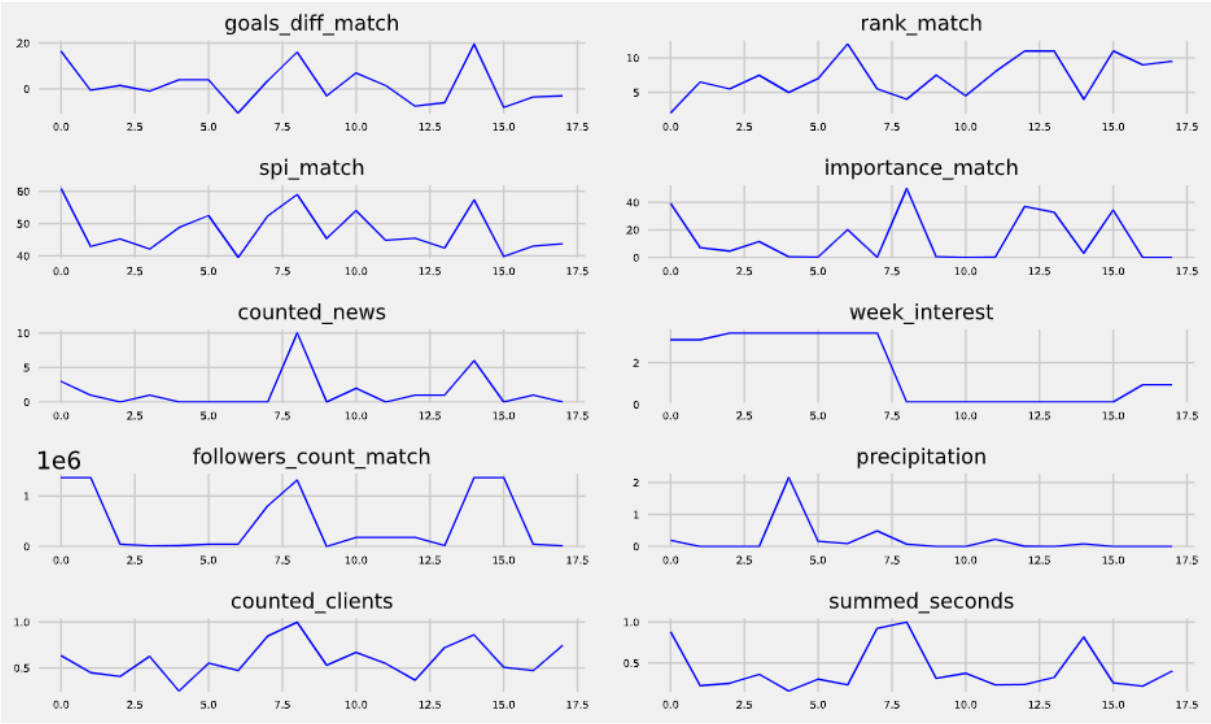


Figure A.28: Famalicão FC time series plot.

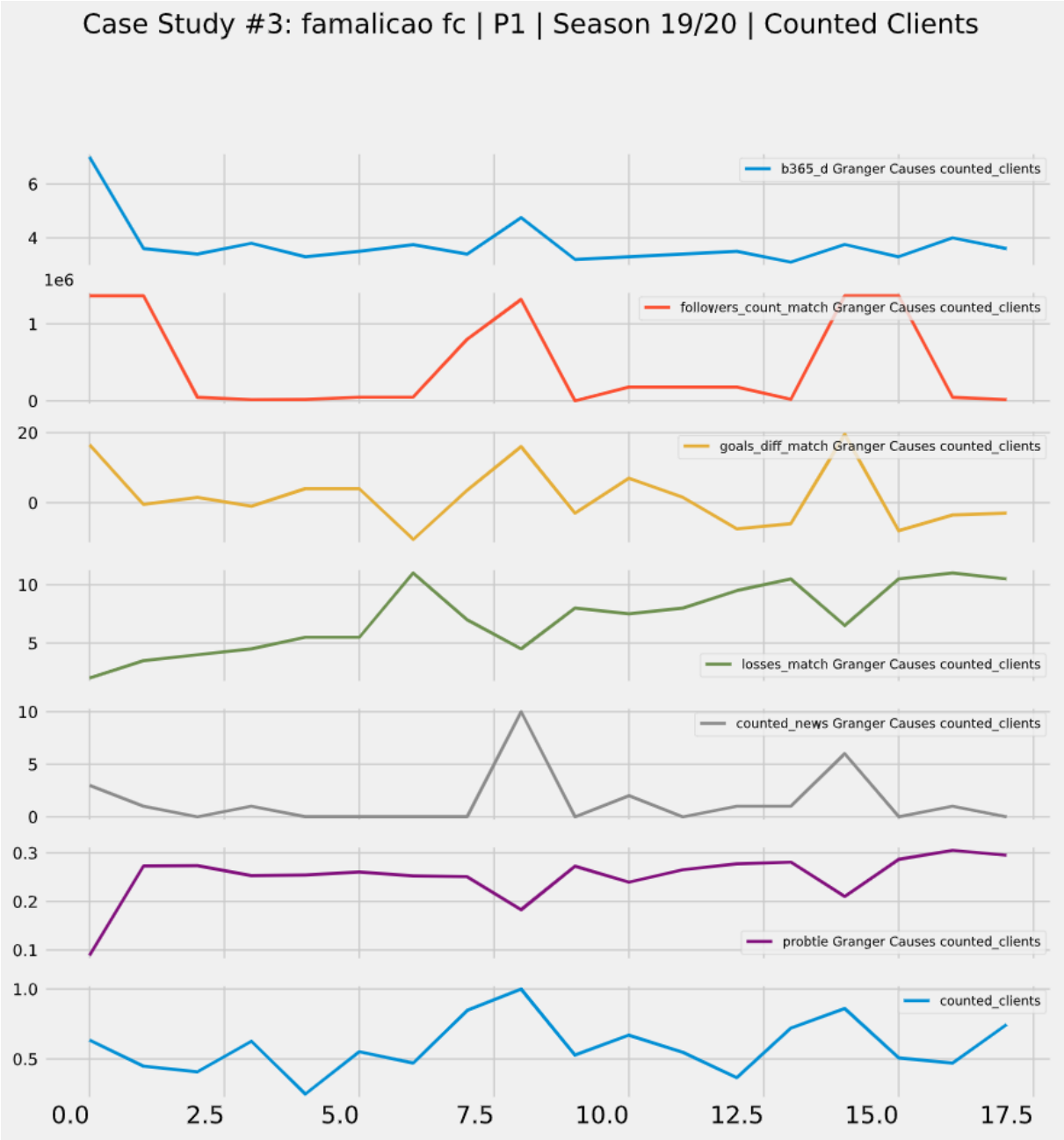


Figure A.29: Granger Causality Famalicão FC - Counted Clients.

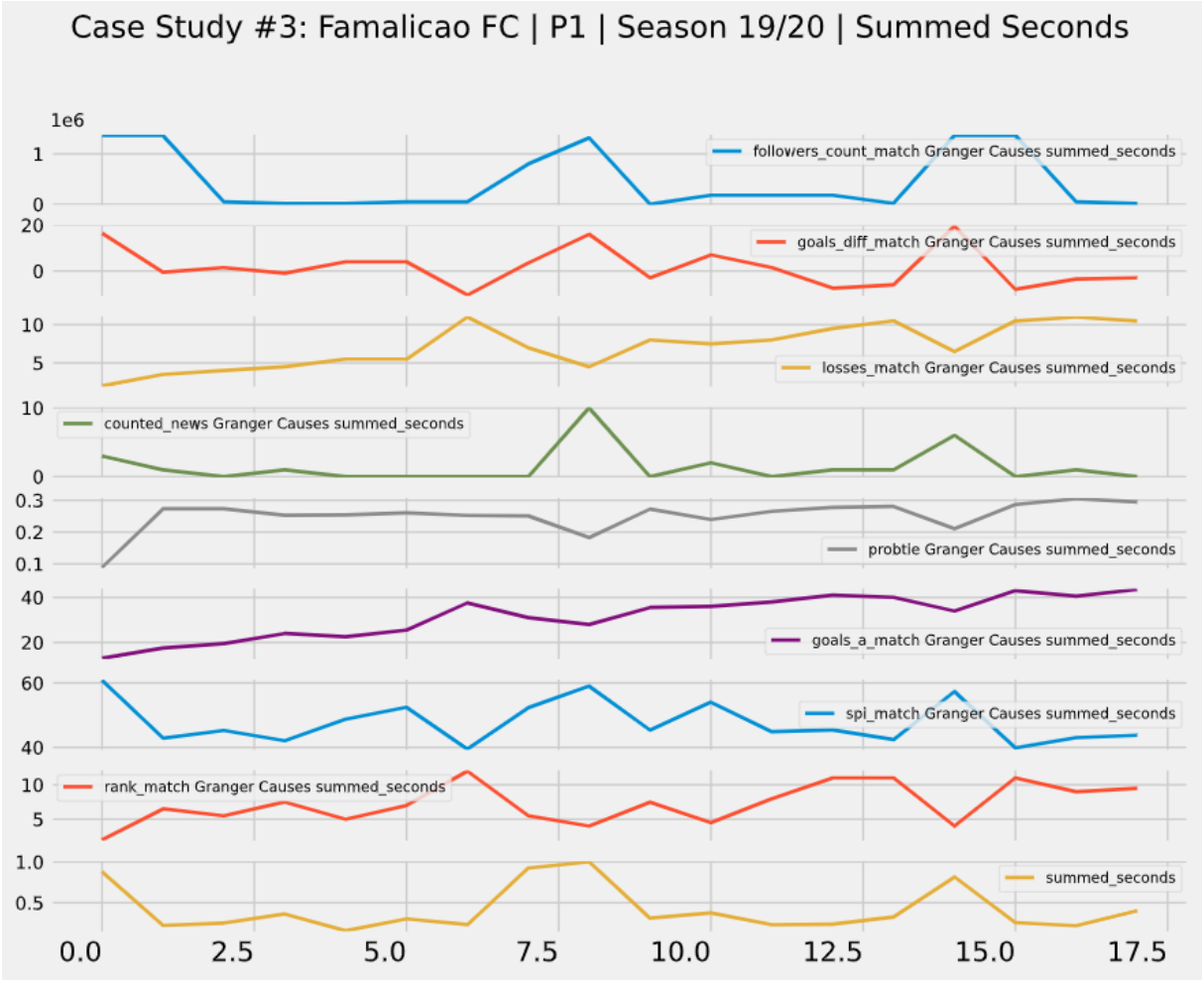


Figure A.30: Granger Causality Famalicão FC - Summed Seconds.

A.7 Case study for Sp Braga

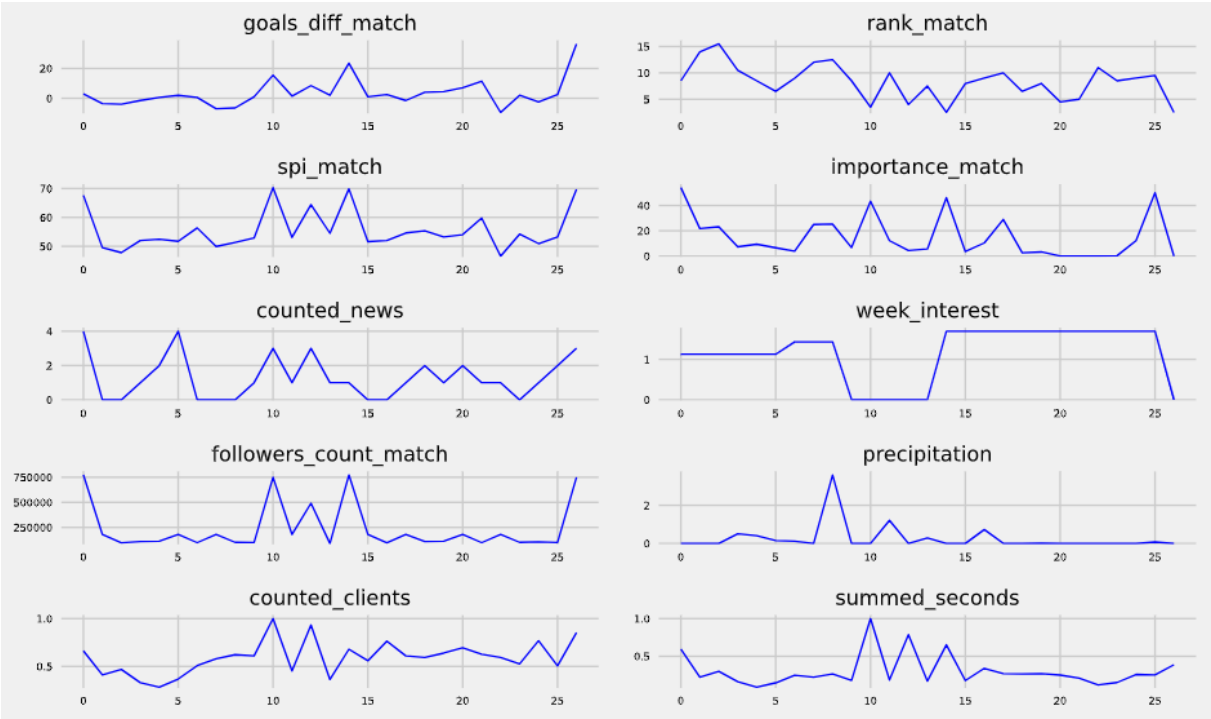


Figure A.31: Sp Braga time series plot.

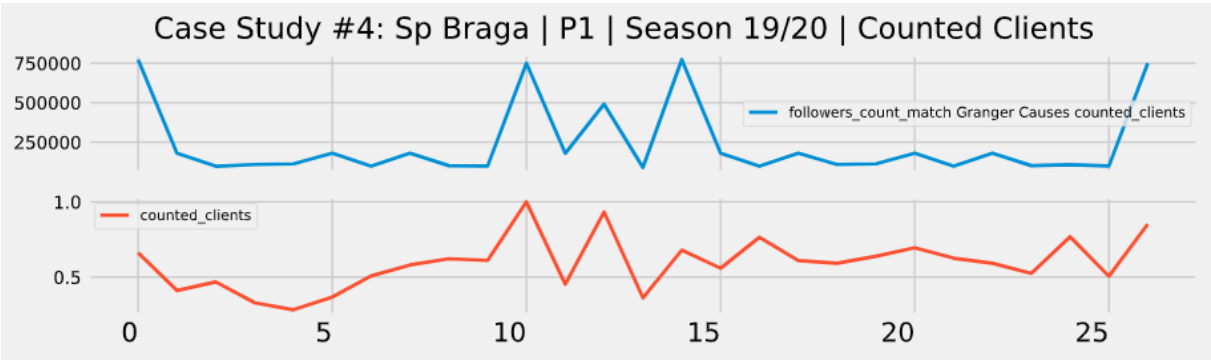


Figure A.32: Granger Causality Sp Braga - Counted Clients.

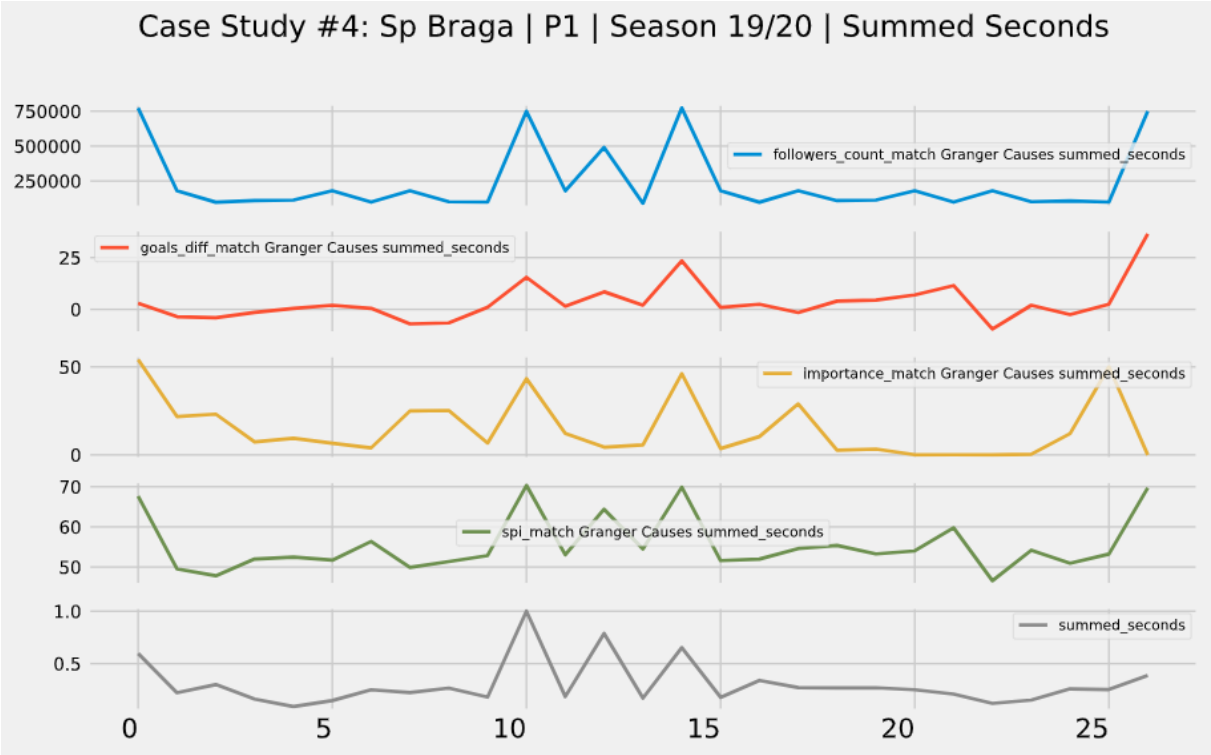


Figure A.33: Granger Causality Sp Braga - Summed Seconds.

A.8 Case study for CD Aves

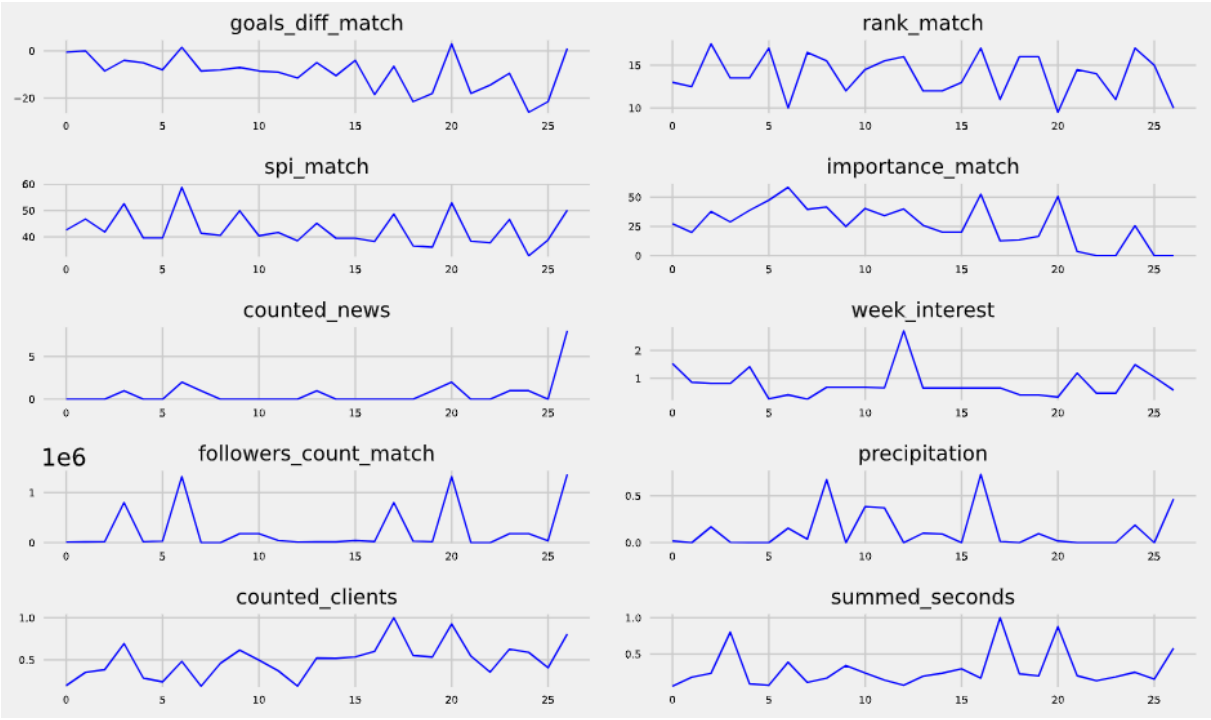


Figure A.34: CD Aves time series plot.

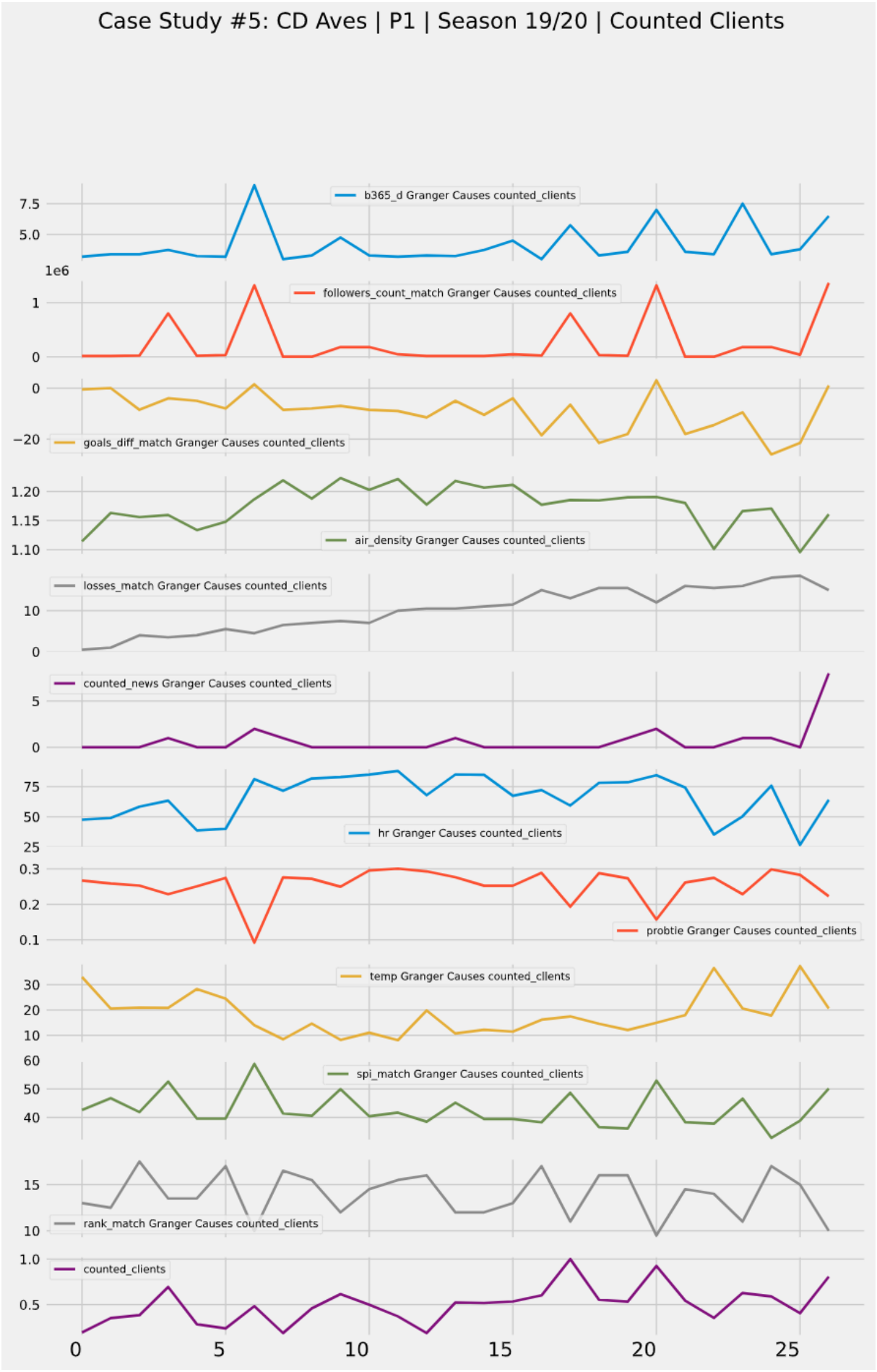


Figure A.35: Granger Causality CD Aves - Counted Clients.

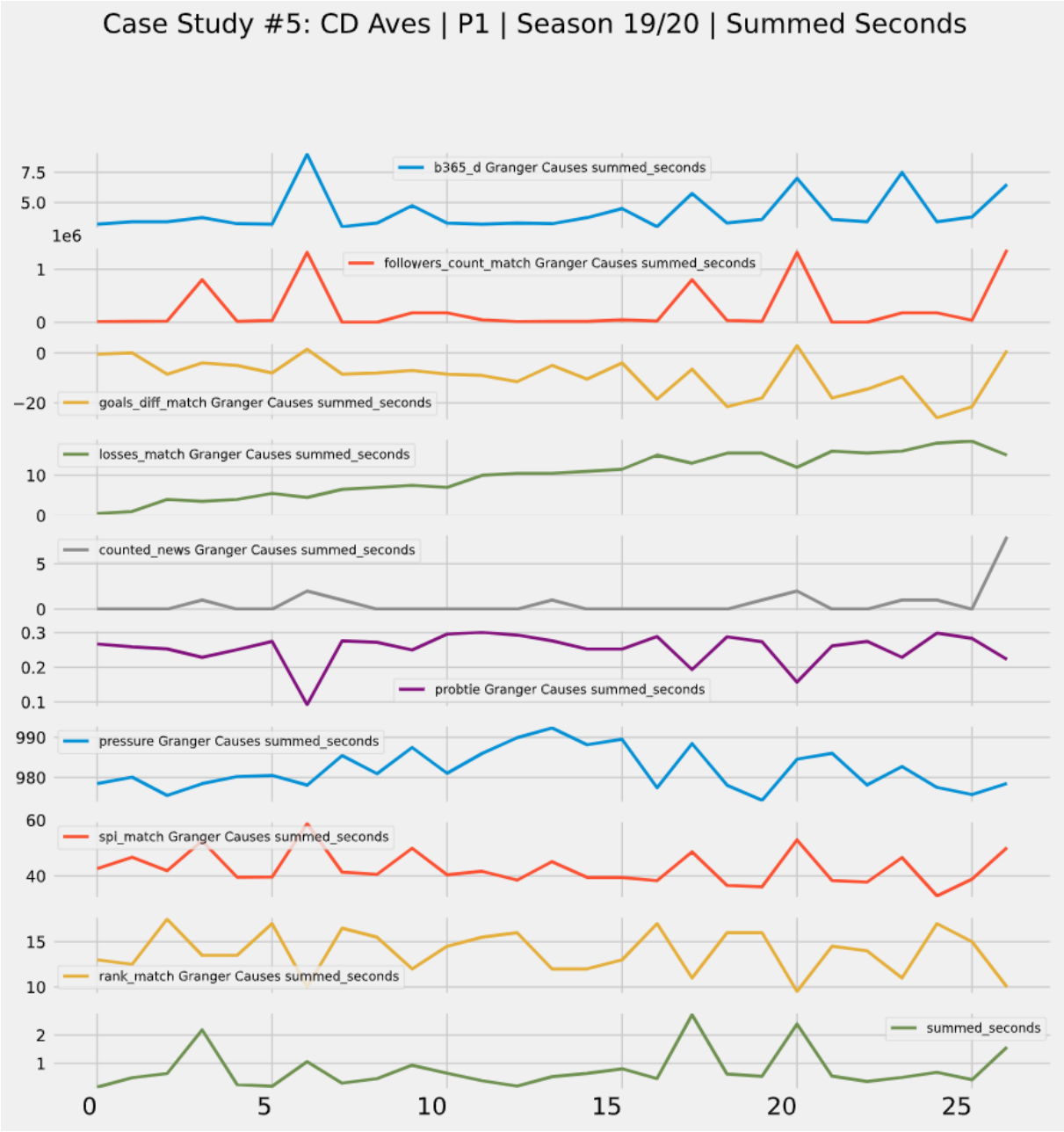


Figure A.36: Granger Causality CD Aves - Summed Seconds.

A.9 Case study for Rio Ave

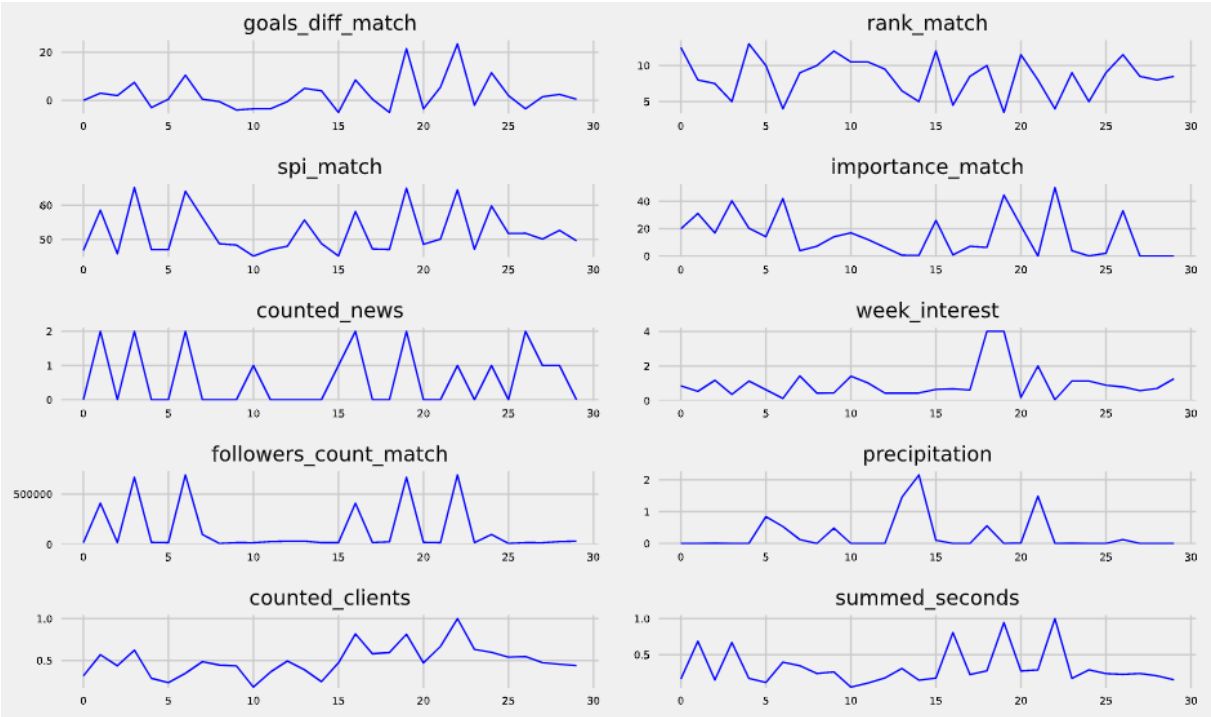


Figure A.37: Rio Ave time series plot.

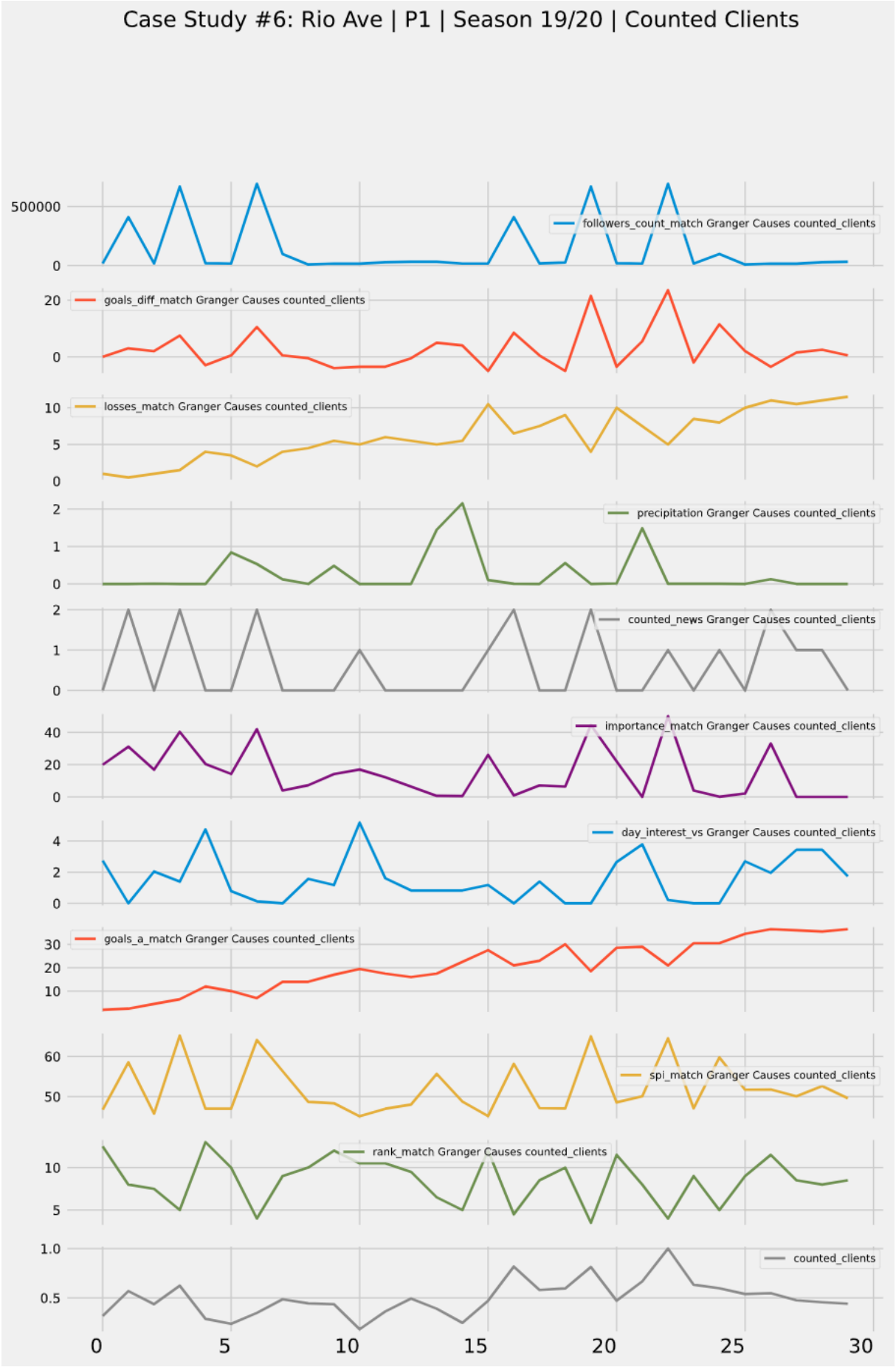


Figure A.38: Granger Causality Rio Ave - Counted Clients.

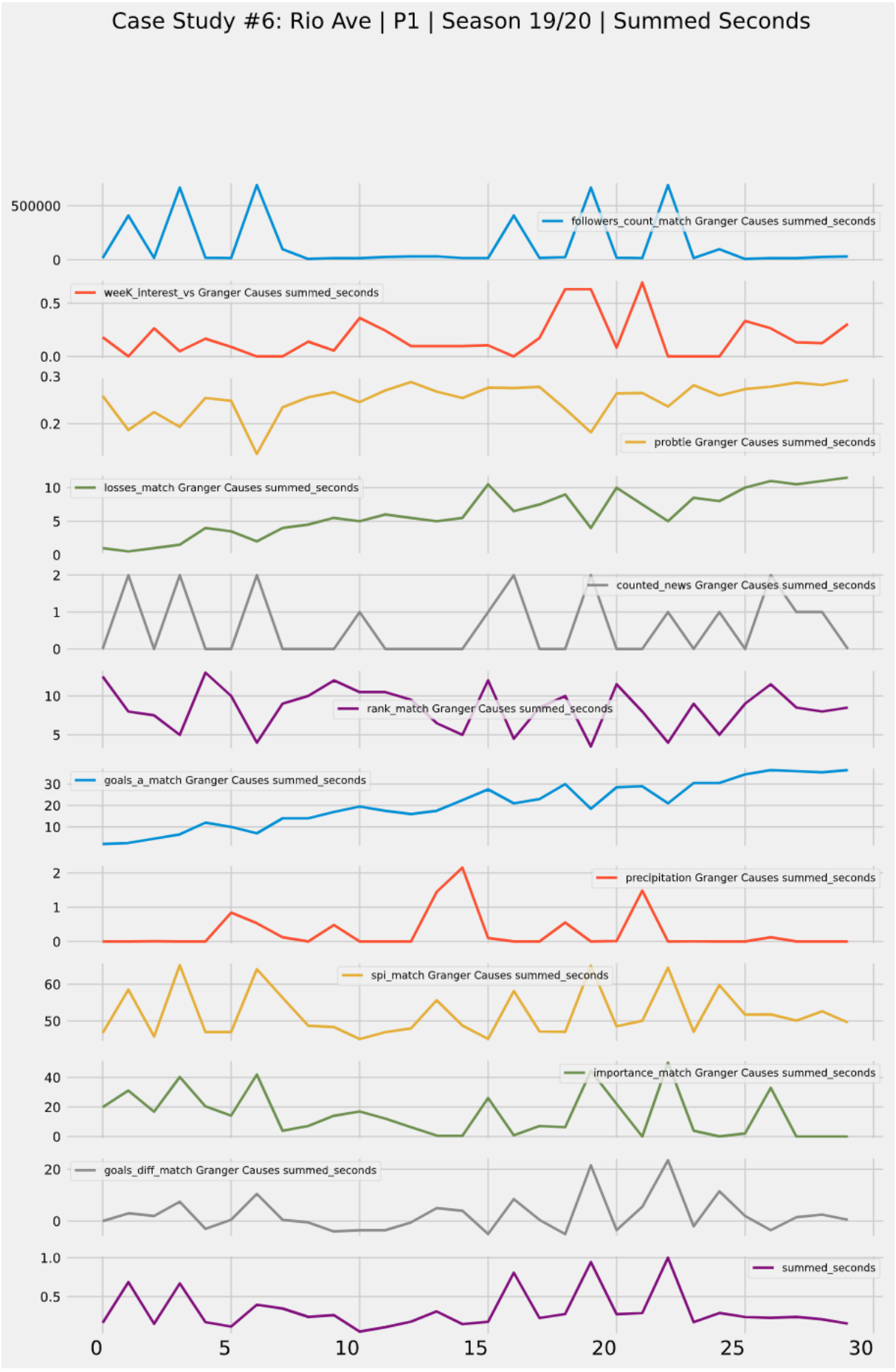


Figure A.39: Granger Causality Rio Ave - Summed Seconds.

Bibliography

[1] Exploratory data analysis. https://www.ibm.com/cloud/learn/exploratory-data-analysis. Accessed: 2021-01-05.

[2] Jackie Abell, Susan Condor, Robert D Lowe, Stephen Gibson, and Clifford Stevenson. Who ate all the pride? patriotic sentiment and english national football support. *Nations and nationalism*, 13(1):97–116, 2007.

[3] Charu C. Aggarwal. *Data mining: the textbook*. Springer, 2015.

[4] Kevin Alavy, Alison Gaskell, Stephanie Leach, and Stefan Szymanski. On the edge of your seat: Demand for football on television and the uncertainty of outcome hypothesis. *International Journal of Sport Finance*, 5(2):75, 2010.

[5] Ana Azevedo and Manuel Santos. Kdd, semma and crisp-dm: A parallel overview. pages 182–185, 01 2008.

[6] Mark Baimbridge, Samuel Cameron, and Peter Dawson. SATELLITE TELEVISION AND THE DEMAND FOR FOOTBALL: A WHOLE NEW BALL GAME? *Scottish Journal of Political Economy*, 43(3):317–333, aug 1996. doi:10.1111/j.1467-9485.1996.tb00848.x.

[7] Mark Baimbridge, Samuel Cameron, and Peter Dawson. Satellite television and the demand for football: A whole new ball game? *Scottish Journal of Political Economy*, 43(3):317–333, 1996.

[8] John Bale. Sport and national identity: a geographical view. *The International Journal of the History of Sport*, 3(1):18–41, 1986. doi:10.1080/02649378608713587.

[9] Arindam Banerjee, Tathagata Bandyopadhyay, and Prachi Acharya. Data analytics: Hyped up aspirations or true potential? *Vikalpa*, 38(4):1–12, 2013. doi:10.1177/0256090920130401.

[10] Elias Bareinboim and Judea Pearl. Causal inference and the data-fusion problem. *Proceedings of the National Academy of Sciences*, 113(27):7345–7352, 2016. ISSN: 0027-8424. doi:10.1073/pnas.1510507113.

[11] George A Barnett, Hsiu-Jung Chang, Edward L Fink, and William D Richards Jr. Seasonality in television viewing: A mathematical model of cultural processes. *Communication Research*, 18(6):755–772, 1991.

[12] Daniel Berrar. Cross-validation., 2019.

[13] Michael R. Berthold, Christian Borgelt, Frank Hppner, and Frank Klawonn. *Guide to Intelligent Data Analysis: How to Intelligently Make Sense of Real Data.* Springer Publishing Company, Incorporated, 1st edition, 2010. ISBN: 1848822596.

[14] Brad Boehmke and Brandon M Greenwell. *Hands-on machine learning with R.* CRC Press, 2019.

[15] JEFFERY BORLAND and ROBERT MACDONALD. Demand for sport. *Oxford Review of Economic Policy*, 19(4):478–502, 2003. ISSN: 0266903X, 14602121.

[16] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control.* John Wiley & Sons, 2015.

[17] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[18] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[19] Peter J Brockwell, Peter J Brockwell, Richard A Davis, and Richard A Davis. *Introduction to time series and forecasting.* Springer, 2016.

[20] Chris Brooks. *Univariate time series modelling and forecasting*, page 206–264. Cambridge University Press, 2 edition, 2008. doi:10.1017/CBO9780511841644.006.

[21] Allison J.B. Chaney, Mike Gartrell, Jake M. Hofman, John Guiver, Noam Koenigstein, Pushmeet Kohli, and Ulrich Paquet. A large-scale exploration of group viewing patterns. In *Proceedings of the ACM International Conference on Interactive Experiences for TV and Online Video*, TVX '14, page 31–38, New York, NY, USA, 2014. Association for Computing Machinery. ISBN: 9781450328388. doi:10.1145/2602299.2602309.

[22] Chris Chatfield. *Time-series forecasting.* CRC press, 2000.

[23] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[24] Song Zan Chiou-Wei, Ching-Fu Chen, and Zhen Zhu. Economic growth and energy consumption revisited — evidence from linear and nonlinear granger causality. *Energy Economics*, 30(6):3063–3076, 2008. ISSN: 0140-9883. Technological Change and the Environment. doi:https://doi.org/10.1016/j.eneco.2008.02.002.

[25] Cross-Disorder Consortium, Jordan Smoller, Craddock N, Kendler K, Phil Lee, Neale BM, John Nurnberger, Stephan Ripke, Susan Santangelo, Sullivan JW, Purcell SM, Anney R, Jan Buitelaar, Fanous A, Faraone SV, Witte Hoogendijk, Lesch KP, Levinson DF, Roy Perlis, and Sebastian Zöllner. Cross-disorder group of the psychiatric genomics c, genetic risk outcome of psychosis c. identification of risk loci with shared effects on five major psychiatric disorders: a genome-wide analysis. lancet 381: 1371-1379. *The Lancet*, 381: 1371–1379, 04 2013. doi:10.1016/S0140-6736(12)62129-1.

[26] Rajeev H. Dehejia and Sadek Wahba. Causal effects in nonexperimental studies: Reevaluating the evaluation of training programs. *Journal of the American Statistical Association*, 94(448):1053–1062, 1999. doi:10.1080/01621459.1999.10473858.

[27] David A. Dickey and Wayne A. Fuller. Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association*, 74(366a): 427–431, 1979. doi:10.1080/01621459.1979.10482531.

[28] Stephen M Dobson and John A Goddard. Performance and revenue in professional league football: evidence from granger causality tests. *Applied Economics*, 30(12):1641–1651, 1998.

[29] Paul Downward, Alistair Dawson, and Trudo Dejonghe. *The Demand for Professional Team Sports: Attendance and Broadcasting*, pages 261–300. 12 2009. ISBN: 9780750683548. doi:10.1016/B978-0-7506-8354-8.00010-7.

[30] Brendan Dwyer and James Weiner. Daily grind: A comparison of causality orientations, emotions, and fantasy sport participation. *Journal of gambling studies*, 34(1):1–20, 2018.

[31] Brendan Dwyer and James Weiner. Daily Grind: A Comparison of Causality Orientations, Emotions, and Fantasy Sport Participation. *Journal of Gambling Studies*, 34(1):1–20, March 2018. ISSN: 1573-3602. doi:10.1007/s10899-017-9684-4.

[32] Imme Ebert-Uphoff and Yi Deng. Causal discovery for climate research using graphical models. *Journal of Climate*, 25:5648–5665, 09 2012. doi:10.1175/JCLI-D-11-00387.1.

[33] Michael Eichler. *Causal Inference in Time Series Analysis*, chapter 22, pages 327–354. John Wiley Sons, Ltd. ISBN: 9781119945710. doi:https://doi.org/10.1002/9781119945710.ch22.

[34] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. Knowledge discovery and data mining: Towards a unifying framework. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 82–88. AAAI Press, 1996.

[35] Arne Feddersen and Armin Rott. Determinants of demand for televised live football: Features of the german national football team. *Journal of Sports Economics*, 12(3):352–369, 2011. doi:10.1177/1527002511404783.

[36] David Forrest and Robert Simmons. Outcome uncertainty and attendance demand in sport: The case of english soccer. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 51(2):229–241, 2002. ISSN: 00390526, 14679884.

[37] Louis A. Fourt and Joseph W. Woodlock. Early prediction of market success for new grocery products. *Journal of Marketing*, 25(2):31–38, 1960. doi:10.1177/002224296002500206.

[38] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[39] Karl Pearson F.R.S. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11): 559–572, 1901. doi:10.1080/14786440109462720.

[40] Amir Gandomi and Murtaza Haider. Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, 35(2):137 – 144, 2015. ISSN: 0268-4012. doi:https://doi.org/10.1016/j.ijinfomgt.2014.10.007.

[41] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438, 1969. ISSN: 00129682, 14680262.

[42] Ruocheng Guo, Lu Cheng, Jundong Li, P. Richard Hahn, and Huan Liu. A survey of learning causality with data: Problems and methods. *CoRR*, abs/1809.09337, 2018.

[43] Stephen Hall, Stefan Szymanski, and Andrew S. Zimbalist. Testing causality between team performance and payroll: The cases of major league baseball and english soccer. *Journal of Sports Economics*, 3(2):149–168, 2002. doi:10.1177/152700250200300204.

[44] James Douglas Hamilton. *Time series analysis*. Princeton university press, 2020.

[45] Hossein Hassani, Xu Huang, and Mansi Ghodsi. Big data and causality. *Annals of Data Science*, 5, 06 2018. doi:10.1007/s40745-017-0122-3.

[46] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *Random Forests*, pages 587–604. Springer New York, New York, NY, 2009. ISBN: 978-0-387-84858-7. doi:10.1007/978-0-387-84858-7$_1$5.

[47] Kjetil Haugen and Knut Heen. The competitive evolution of european top football – signs of danger [in european j. of sport studies]. *European Journal of Sport Science*, 04 2018.

[48] Alain Hauser and Peter Bühlmann. Jointly interventional and observational data: estimation of interventional markov equivalence classes of directed acyclic graphs. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77(1):291–318, 2015. doi:https://doi.org/10.1111/rssb.12071.

[49] David Heckerman, Christopher Meek, and Gregory Cooper. *A Bayesian Approach to Causal Discovery*, pages 1–28. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006. ISBN: 978-3-540-33486-6. doi:10.1007/3-540-33486-6$_1$.

[50] Miguel Hernán, Babette Brumback, and James Robins. Marginal structural models to estimate the causal effect of zidovudine on the survival of hiv-positive men. *Epidemiology (Cambridge, Mass.)*, 11:561–70, 10 2000. doi:10.1097/00001648-200009000-00012.

[51] Jennifer L. Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011. doi:10.1198/jcgs.2010.08162.

[52] Nai-Wei Hsu, Kai-Shuo Liu, and Shun-Chuan Chang. Choking under the pressure of competition: A complete statistical investigation of pressure kicks in the nfl, 2000–2017. *PLOS ONE*, 14(4): 1–18, 04 2019. doi:10.1371/journal.pone.0214096.

[53] Rob J. Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[54] Kota Itoda, Norifumi Watanabe, and Yoshiyasu Takefuji. Model-based behavioral causality analysis of handball with delayed transfer entropy. *Procedia Computer Science*, 71:85–91, 2015. ISSN: 1877-0509. 6th Annual International Conference on Biologically Inspired Cognitive Architectures, BICA 2015, 6-8 November Lyon, France. doi:https://doi.org/10.1016/j.procs.2015.12.210.

[55] Nikos Kalatzis, Ioanna Roussaki, Christos Matsoukas, Marios Paraskevopoulos, Symeon Papavassiliou, and Simona Tonoli. Social media and google trends in support of audience analytics: Methodology and architecture. *DATA ANALYTICS 2018*, page 49, 2018.

[56] Nathan Kallus, Aahlad Manas Puli, and Uri Shalit. Removing hidden confounding by experimental grounding. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, pages 10888–10897. Curran Associates, Inc., 2018.

[57] Fatih Karanfil. An empirical analysis of european football rivalries based on on-field performances. *Sport Management Review*, 20(5):468–482, 2017. ISSN: 1441-3523. doi:https://doi.org/10.1016/j.smr.2016.12.003.

[58] JP Kelly. Television by the numbers: The challenges of audience measurement in the age of big data. *Convergence*, 25(1):113–132, 2019. doi:10.1177/1354856517700854.

[59] Denis Khryashchev, Alexandru Papiu, Jiamin Xuan, Olivia Dinica, Kyle Hubert, and Vo Huy. *Who Watches What: Forecasting Viewership for the Top 100 TV Networks*, pages 163–174. 11 2019. ISBN: 978-3-030-34979-0. doi:10.1007/978-3-030-34980-6$_1$9.

[60] Eivind Kristoffersen, Oluseun Omotola Aremu, Fenna Blomsma, Patrick Mikalef, and Jingyue Li. Exploring the relationship between data science and circular economy: An enhanced crisp-dm process model. In Ilias O. Pappas, Patrick Mikalef, Yogesh K. Dwivedi, Letizia Jaccheri, John Krogstie, and Matti Mäntymäki, editors, *Digital Transformation for a Sustainable Society in the 21st Century*, pages 177–189, Cham, 2019. Springer International Publishing. ISBN: 978-3-030-29374-1.

[61] Santiago Lago-Peñas, Ignacio Lago, and Carlos Peñas. Player migration and soccer performance. *Frontiers in Psychology*, 10, 03 2019. doi:10.3389/fpsyg.2019.00616.

[62] Robert LaLonde. Evaluating the econometric evaluations of training programs with experimental data. *American Economic Review*, 76(4):604–20, 1986.

[63] Milad Keshtkar Langaroudi and Mohammadreza Yamaghani. Sports result prediction based on machine learning and computational intelligence approaches: A survey. 2019.

[64] Srivatsan Laxman and P Shanti Sastry. A survey of temporal data mining. *Sadhana*, 31(2): 173–198, 2006.

[65] Junho Lee, Wu Wang, Fouzi Harrou, and Ying Sun. Reliable solar irradiance prediction using ensemble learning-based models: A comparative study. *Energy Conversion and Management*, 208:112582, 2020. ISSN: 0196-8904. doi:https://doi.org/10.1016/j.enconman.2020.112582.

[66] Jundong Li, Osmar R. Zaïane, and Alvaro Osornio-Vargas. Discovering statistically significant co-location rules in datasets with extended spatial objects. In Ladjel Bellatreche and Mukesh K. Mohania, editors, *Data Warehousing and Knowledge Discovery*, pages 124–135, Cham, 2014. Springer International Publishing. ISBN: 978-3-319-10160-6.

[67] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, and Huan Liu. Feature Selection: A Data Perspective. *ACM Computing Surveys*, 50(6):1–45, January 2018. ISSN: 0360-0300, 1557-7341. arXiv: 1601.07996. doi:10.1145/3136625.

[68] Sunghoon Lim and Conrad S. Tucker. Mining twitter data for causal links between tweets and real-world outcomes. *Expert Systems with Applications: X*, 3:100007, 2019. ISSN: 2590-1885. doi:https://doi.org/10.1016/j.eswax.2019.100007.

[69] Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *arXiv preprint arXiv:1705.07874*, 2017.

[70] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[71] M. Eileen Magnello. *Weldon, Walter Frank Raphael*. American Cancer Society, 2014. ISBN: 9781118445112. doi:https://doi.org/10.1002/9781118445112.stat01312.

[72] G. Mahalakshmi, S. Sridevi, and S. Rajaram. A survey on forecasting of time series data. In *2016 International Conference on Computing Technologies and Intelligent Data Engineering (ICCTIDE'16)*, pages 1–8, 2016. doi:10.1109/ICCTIDE.2016.7725358.

[73] S. Mani and G. Cooper. Causal discovery from medical textual data. *Proceedings. AMIA Symposium*, pages 542–6, 2000.

[74] Sara Mesquita, Cláudio Haupt Vieira, Lília Perfeito, and Joana Gonçalves-Sá. Learning from pandemics: using extraordinary events can improve disease now-casting models. *CoRR*, abs/2101.06774, 2021.

[75] Denny Meyer and Rob Hyndman. The accuracy of television network rating forecasts: The effects of data aggregation and alternative models. *Model Assisted Statistics and Applications*, 1: 147–155, 11 2006. doi:10.3233/MAS-2006-1303.

[76] Steffen Q Mueller. Pre-and within-season attendance forecasting in major league baseball: a random forest approach. *Applied Economics*, 52(41):4512–4528, 2020.

[77] Philip M. Napoli. The unpredictable audience: An exploratory analysis of forecasting error for new prime-time network television programs. *Journal of Advertising*, 30(2):53–60, 2001. doi:10.1080/00913367.2001.10673637.

[78] Radu Neagu. Forecasting television viewership: a case study. *GE Global Research, 2003GRC039*, 2003.

[79] Kimberly Neuendorf, Leo W Jeffres, and David Atkin. The television of abundance arrives: cable choices and interest maximization. *Telematics and Informatics*, 17(3):169–197, 2000. ISSN: 0736-5853. doi:https://doi.org/10.1016/S0736-5853(00)00007-1.

[80] Lyndon Nixon, Krzysztof Ciesielski, and Basil Philipp. Ai for audience prediction and profiling to power innovative tv content recommendation services. In *Proceedings of the 1st International Workshop on AI for Smart TV Content Production, Access and Delivery*, AI4TV '19, page 42–48, New York, NY, USA, 2019. Association for Computing Machinery. ISBN: 9781450369176. doi:10.1145/3347449.3357485.

[81] Amir Bahador Parsa, Ali Movahedi, Homa Taghipour, Sybil Derrible, and Abolfazl (Kouros) Mohammadian. Toward safer highways, application of xgboost and shap for real-time accident detection and feature analysis. *Accident Analysis Prevention*, 136:105405, 2020. ISSN: 0001-4575. doi:https://doi.org/10.1016/j.aap.2019.105405.

[82] Rodney J Paul, Yoav Wachsman, and Andrew Weinbach. Measuring and forecasting fan interest in nfl football games. *Journal of Gambling Business & Economics*, 6(3), 2012.

[83] Judea Pearl. Theoretical impediments to machine learning with seven sparks from the causal revolution. *CoRR*, abs/1801.04016, 2018.

[84] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011.

[85] Leví Pérez Carcedo, Víctor Puente Robles, and Plácido Rodríguez Guerrero. Factors determining tv soccer viewing: Does uncertainty of outcome really matter? *International Journal of Sport Finance, 12*, 2017.

[86] Roger Cooper Ph.D. and Tang Tang Ph.D. Predicting audience exposure to television in today's media environment: An empirical integration of active-audience and structural theories. *Journal of Broadcasting & Electronic Media*, 53(3):400–418, 2009. doi:10.1080/08838150903102204.

[87] Robi Polikar. Ensemble based systems in decision making. *IEEE Circuits and systems magazine*, 6(3):21–45, 2006.

[88] John W Ratcliff and David E Metzener. Pattern-matching-the gestalt approach. *Dr Dobbs Journal*, 13(7):46, 1988.

[89] Daam Reeth. Television demand for the tour de france: the importance of outcome uncertainty, patriotism and doping. 01 2011.

[90] David Reinsel, John Gantz, and John Rydning. The digitization of the world from edge to core. Technical report, International Data Corporation, November 2018.

[91] JM Robins, MA Hernan, and Brumback B. Marginal structural models and causal inference in epidemiology. *Epidemiology*, 11:550–560, 01 2000.

[92] Yuji Roh, Geon Heo, and Steven Euijong Whang. A survey on data collection for machine learning: a big data - AI integration perspective. *CoRR*, abs/1811.03402, 2018.

[93] Jakob Runge. Discovering contemporaneous and lagged causal relations in autocorrelated nonlinear time series datasets. In *Conference on Uncertainty in Artificial Intelligence*, pages 1388–1397. PMLR, 2020.

[94] Harald Schoen, Daniel Gayo-Avello, Panagiotis Metaxas, Eni Mustafaraj, Markus Strohmaier, and Peter Gloor. The power of prediction with social media. *Internet Research: Electronic Networking Applications and Policy*, 23, 10 2013. doi:10.1108/IntR-06-2013-0115.

[95] Schreyer, Benno Torgler, and Sascha L. Schmidt. Game outcome uncertainty and television audience demand: New evidence from german football. *German Economic Review*, 19(2):140–161, 2018. doi:doi:10.1111/geer.12120.

[96] Skipper Seabold and Josef Perktold. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010.

[97] Maher Selim, Ryan Zhou, Wenying Feng, and Omar Alam. The impact of external features on prediction accuracy in short-term energy forecasting, 10 2020. doi:10.13140/RG.2.2.27398.40003.

[98] Shunrong Shen, Haomiao Jiang, and Tongda Zhang. Stock market forecasting using machine learning algorithms. *Department of Electrical Engineering, Stanford University, Stanford, CA*, pages 1–5, 2012.

[99] Shun-Yao Shih, Fan-Keng Sun, and Hung-yi Lee. Temporal pattern attention for multivariate time series forecasting. *Machine Learning*, 108(8):1421–1441, 2019.

[100] Aaron CT Smith and Bob Stewart. *Introduction to sport marketing*. Routledge, 2014.

[101] Rodrigo Uribe, Cristian Buzeta, Enrique Manzur, and Isabel Alvarez. Determinants of football tv audience: The straight and ancillary effects of the presence of the local team on the fifa world cup. *Journal of Business Research*, 127:454–463, 2021. ISSN: 0148-2963. doi:https://doi.org/10.1016/j.jbusres.2019.10.064.

[102] A. Vaccaro, P. Mercogliano, P. Schiano, and D. Villacci. An adaptive framework based on multi-model data fusion for one-day-ahead wind power forecasting. *Electric Power Systems Research*, 81(3):775–782, 2011. ISSN: 0378-7796. doi:https://doi.org/10.1016/j.epsr.2010.11.009.

[103] Daam Van Reeth. Forecasting tour de france tv audiences: A multi-country analysis. *International Journal of Forecasting*, 35(2):810–821, 2019. ISSN: 0169-2070. doi:https://doi.org/10.1016/j.ijforecast.2018.06.003.

[104] Michel Verleysen and Damien François. The curse of dimensionality in data mining and time series prediction. In *International work-conference on artificial neural networks*, pages 758–770. Springer, 2005.

[105] B Vishwas and ASHISH PATEL. *Hands-on Time Series Analysis with Python: From Basics to Bleeding Edge Techniques*. 01 2020. ISBN: 978-1-4842-5991-7. doi:10.1007/978-1-4842-5992-4.

[106] James G Webster and Ting-Yu Wang. Structural determinants of exposure to television: The case of repeat viewing. *Journal of Broadcasting & Electronic Media*, 36(2):125–136, 1992.

[107] Bin Weng, Lin Lu, Xing Wang, Fadel M. Megahed, and Waldyn Martinez. Predicting short-term stock prices using ensemble methods and online data sources. *Expert Systems with Applications*, 112:258–273, 2018. ISSN: 0957-4174. doi:https://doi.org/10.1016/j.eswa.2018.06.016.

[108] R. Wirth and Jochen Hipp. Crisp-dm: Towards a standard process model for data mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, 01 2000.

[109] G.Peter Zhang. Time series forecasting using a hybrid arima and neural network model. *Neurocomputing*, 50:159–175, 2003. ISSN: 0925-2312. doi:https://doi.org/10.1016/S0925-2312(01)00702-0.

[110] Wengang Zhang, Chongzhi Wu, Haiyi Zhong, Yongqin Li, and Lin Wang. Prediction of undrained shear strength using extreme gradient boosting and random forest based on bayesian optimization. *Geoscience Frontiers*, 12(1):469–477, 2021. ISSN: 1674-9871. doi:https://doi.org/10.1016/j.gsf.2020.03.007.

[111] Yongli Zhang. Dynamic effect analysis of meteorological conditions on air pollution: A case study from beijing. *Science of The Total Environment*, 684:178–185, 2019. ISSN: 0048-9697. doi:https://doi.org/10.1016/j.scitotenv.2019.05.360.