Instituto Tecnologico y de Estudios Superiores de Monterrey

Monterrey Campus

School of Engineering and Sciences

Mining the SCOPUS Database to Identify Potential Academic Rising Stars

A thesis presented by

Jorge Antonio Ayala Urbina

Submitted to the
School of Engineering and Sciences
in partial fulfillment of the requirements for the degree of

Master of Science

in

Computer Science

Monterrey, Nuevo León, June, 2021

Instituto Tecnologico y de Estudios Superiores de Monterrey

Campus Monterrey

The committee members, hereby, certify that have read the thesis presented by Jorge Antonio Ayala Urbina and that it is fully adequate in scope and quality as a partial requirement for the degree of Master of Science in Computer Sciences.

Dr. Héctor Gibrán Ceballos Cancino
Tecnológico de Monterrey
Principal Advisor

Dr. Neil Hernández Gress
Tecnológico de Monterrey
Committee Member

Dr. Francisco J. Cantú Ortiz
Tecnológico de Monterrey
Committee Member

Dr. Juan Pablo García Vázquez
Universidad Autónoma de Baja California
Committee Member

Dr. Rubén Morales Menéndez
Associate Dean of Graduate Studies
School of Engineering and Sciences

Monterrey, Nuevo León, June, 2021

Declaration of Authorship

I, Jorge Antonio Ayala Urbina, declare that this thesis titled, Mining the SCOPUS Database to Identify Potential Academic Rising Stars and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Jorge Antonio Ayala Urbina
Monterrey, Nuevo León, June, 2021

Dedication

I would like to dedicate this thesis project to my parents, first of all, for always being there for me, and for being two outstanding role models. And also for listening to my unintelligible rants about the roadblocks I faced during the development of this thesis project. Also, I dedicate this thesis project to my two brothers, Santiago and Eduardo, for always being there when I have needed them the most. To my grandparents, and specially to my late grandfather Óscar, wish you were here to witness this milestone with me.
Additionally, I dedicate this thesis to my close friends who motivated me to pursue a Master's degree and have in some way or another being there for me. María, Demian, Checo, Carlos, Patricio, Adair, Emiliano, Liz, Martín, Ian, Lewis, Aizar, and Paulo, without your encouragement this would not have been possible.

To all of you, my deepest and sincerest gratitude.

Acknowledgements

First, I would like to express my deepest gratitude to my thesis supervisor, Dr. Héctor Ceballos, for being an excellent mentor and guiding me during this last two years. Without your comprehension, patience and knowledge, this thesis project could not have come to fruition. Thank you for sharing your knowledge and experience with me, and above all, thank you for trusting in me the day I stepped in your office asking you to being my thesis supervisor.

Also, I would like to thank my professors, Dr. Francisco Cantu, Dr. Hugo Terashima, Dr. Santiago Conant, Dr. Miguel Angel Medina, Dr. Olivia Carrillo, Dr. Miguel González, Dr. Leonardo Chang, and Dr. Neil Hernández for sharing with me their invaluable knowledge with me. In one way or another, your teachings have made their way into this thesis project, and for that I am infinitely grateful.

Finally, this thesis project and my Master's studies would not have been possible without Tecnológico de Monterrey paying for my tuition. I could not be more grateful with the institution and Dr. Hugo Terashima for accepting my into the Master's program. Thank you for trusting in me. Also, without the support and scholarship from CONACyT I would have never being able to fully dedicate myself to my master's studies.

Mining the SCOPUS Database to Identify Potential Academic Rising Stars

by

Jorge Antonio Ayala Urbina

Abstract

Academic Rising Stars are often defined as authors in the earlier years of their scientific careers who have the potential to become impactful authors in the future. Universities and research institutions would benefit greatly from identifying these Academic Rising Stars and convince them to join their research teams, because if the potential of these authors is fulfilled these could benefit the institution in terms of scientific prestige and impactful scientific production. This thesis project aims to prove if it is possible to identify these Academic Rising Stars using Machine Learning classifiers and the data that is available through Elsevier's Scopus and SciVal APIs. Conducting a case study in the field of *Clustering*, it was shown that it is possible to identify these authors using the average metrics from their first five years of scientific publications, with acceptable precision and accuracy. It was shown that the best attribute to label top authors is the h5-index and the classifier which can achieve the best result is the Support Vector Machine with a radial basis function kernel. The developed methodology provides a solid framework from which research institutions can identify Academic Rising Stars in the fields they are interested in.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

1.1 Scientometrics, Machine Learning and Academic Rising Stars

The field of scientometrics has greatly benefited from the advances in machine learning and data science. This has opened the possibility of predicting with varying grades of success the future scientific success of a researcher. One of the earliest models used to make these predictions is based on the number of articles written, current h-index, years since the publication of the researcher's first article, number of distinct journals published, and number of articles in high impact journals[1]. Since then, other alternative approaches have been proposed, such as approaches based on factors such as a researcher's professional network[27].

Early efforts to predict future scientific success have been based on citations, publications, and h-index[1, 2]. However, more comprehensive approaches have been developed, for example, some of them involve the quality of the research network the author collaborates with. These methods can be based on the temporal impact of the author, their co-author network, and the venue in which they publish [39].

The ability to evaluate and predict scientific impact has prompted a search for **Academic Rising Stars**. There is not a universal consensus on what an Academic Rising Start is, however one of the most straightforward definitions states that Academic Rising Stars are scholars in the beginning stage of their careers and as such are not outstanding among peers or have a low research profile, but they tend to become influential in their academic field[37]. Aiming to identify Academic Rising Stars various methods have been developed, and these will be further discussed in Chapter2.

But why is it important to search for Academic Rising Stars? Having a reliable and consistent way to predict if a researcher is going to become outstanding in their field of research is of interest for universities and research institutions as they constantly strive to improve their scientific output and prestige. Identifying who may become a prominent scientist in the future also allows institutions to reinforce the strength of their internal research network. Additionally, the factors which drive these predictions are of interest to young researchers as they can provide guidance on which aspects of their scientific career they should improve if they aspire to become influential in their field of research. In other words, the search for rising stars provides a university with an answer to the question: "Who should I hire?" and provide a young

researcher with an answer to the question: "Which of my indicators should I improve to stand among my peers?".

The availability of data enables this search. The biggest and most complete database for scientific publications is the Scopus database, and it has still not been exploited to predict rising stars, to the best of our knowledge. The Scopus database is a citation and abstract database which has more than 69 million records[10]. These records include more than 36,000 articles from more than 11,000 publishers. Scopus provides an API that enables a user to make certain queries to the database and get information in return. The API allows retrieving affiliation data, which is the data related to an academic institution, and author data which includes papers, affiliations, and h-index. Additionally, a traditional Scopus search using keywords can be conducted. Using this API, it can be possible to implement a machine learning classifier that gathers Scopus data and identifies Academic Rising Stars in any field of knowledge.

1.2 Finding Academic Rising Stars

Measuring and evaluating the scientific impact of a researcher has been of interest to the scientific community. Currently, the most accepted and widely used solution is the h-index. The h-index is calculated by counting the number of publications for which an author has been cited by other authors at least that same number of times[17]. Although it has come under criticism[8] it is widely accepted and useful to measure the productivity and impact of a scientist. To tackle the problem of finding Academic Rising Stars, using only one metric such as the h-index results inefficient. This mainly has to do with the young researcher having very little time in the scientific field, and as such, the number of papers, citations, and h-index alone are insufficient to describe the impact and potential of the young researcher. Methods to predict academic rising stars have been proposed. Two notable cases, *CocaRank* and *ScholarRank* are heavily inspired by *PageRank*[37, 38, 3]. *PageRank* is an algorithm developed by Larry Page and Sergey Brin, and it can be used by Google to rank web pages. It works by counting the number and quality of links to a page to determine a rough estimate of how important the web page is. To determine how important a web page is, a rank is assigned based on how many other web pages have a direct link to it. So this applied to the search for academic rising stars is used to assess the relevancy of young researchers works in within a research network. These algorithms also weigh in the influence of the citation network and the co-author network.

The latest approaches to identify Academic Rising Stars have been based on machine learning techniques. A non-iterative hierarchical weighted model employed to detect academic rising stars uses author, social, venue, and temporal features to make predictions[26]. These models require evaluating author features, such as how many papers an author has, how many of those have been published in the last five years, the average time between paper publications, and the average citation count. They also evaluate social features, such as the number of co-authors, the citation number of the researcher's co-authors, and the average citations of all its co-authors, this should speak about the quality of the researcher's network. The venue features include the number of citations by journal level, the number of papers in each level, the number of venues the author has published in, and the average number of publications on

each level in the last five years. And finally, the temporal features comprehend the author's citation increment in one year, two years, and the average, and the same increments in their co-author network. These features are the most likely to be used for this thesis project. As they are very comprehensive features and take into account the quality of the researchers, the young researcher works with. Additionally, taking into account the temporal features helps to weigh in the improvement of a young researcher over time.

An alternative approach has used scientometric indicators such as the researcher productivity level, scientific impact level, value of productivity, citations per publication, contribution impact, international collaboration, research area relevancy, and venue reputation[3]. The use of these scientometric indicators can also be evaluated and tested to see if that approach presents many advantages over the method which employs feature evaluation. This last method does not explore the impact that each indicator has, and exploring that would also be valuable for the thesis project. It is not enough to make predictions with a reasonable and reliable level of accuracy, explanations as to why the features and indicators employed in the predictions are impactful and relevant would also enrich the scientific contribution of the project. It is also important to mention that the last method that employs scientometric indicators, was tested using the *Web of Science (WoS)* database, which is similar to the Scopus database. The approach that this method uses could be more relevant to the scope of this project. However, Scopus is more strict on its records and that is why it is considered more reliable.

1.3 This Thesis Project's Goals

Currently, to the best of our knowledge, no one has attempted to predict which researchers are considered Academic Rising Stars using data gathered from the Scopus database. As previously stated, the Scopus database is regarded as the best of its kind; and given the quality of the data, the quality of the predictions should also be more robust compared to the ones previously done. Furthermore, the predictions done using scientometric indicators and the one that uses different features[3, 26], have been done with different databases and in limited scopes such as only for certain academic institutions[3], or very broad domains such as the whole computer science field[26]. In contrast, this thesis project aims at predicting Academic Rising Stars from a specific domain, in this case, *Clustering*. As such, the results should provide evidence that it is possible to predict Academic Rising Stars using a specific area, which is not done in most similar Academic Rising Star prediction exercises[3, 26].

Since this thesis project aims at predicting the Academic Rising Stars of a specific field, it is necessary to specify how these datasets are built in the context of the use of the Scopus database, which to the best of our knowledge has not been documented in previous research papers regarding Academic Rising Stars prediction. Retrieving data using the Scopus API has its limitations in terms of download speed and the amount of data that can be downloaded. Therefore, this project has the additional objective of presenting the method used to acquire, preprocess and enrich the necessary data for the Academic Rising Star prediction task. Once the data is ready, different types of classifiers (Logistic Regression and Support Vector Machine) are trained using different parameters, sets of features, and labels to verify if it is possible to predict Academic Rising Stars using Scopus data.

The successful prediction of Academic Rising Stars can have commercialization potential. It is in the best interest of universities and research institutions to identify and attract the most promising young researchers to their ranks. Therefore, providing these institutions with information on potentially outstanding researchers that suit their research areas can be something they are prepared to pay for. Additionally, the resulting methodology of this thesis project could be integrated into a service that aids the institutions in their researchers' hiring processes in a more comprehensive fashion.

All in all, the aim of this thesis project is to build a relevant data set and then select author features available in Scopus databases to identify Academic Rising Stars. The final goal of the project is to choose a field of knowledge as input and then implement a methodology that builds a data set through Scopus and then returns which young researchers adjust better to the description of an Academic Rising Star.

This thesis project starts by presenting how can researchers' data can be retrieved from the Scopus API in a feasible way, in terms of time and sticking to a reasonable download quota. Once the dataset is built, a thorough Exploratory Data Analysis (EDA) of the dataset will be presented explaining the particularities of the data and how is it processed to make it possible to predict if a researcher will achieve the Academic Rising Star status. Then, the different classification methods and their particularities will be presented. Assessing with different metrics the performance of the different classifiers, it will be possible to conclude if these Academic Rising Star identification exercises have been successful and if so, to what extent. Finally, the results will be discussed and if further experimentation can be conducted to increase the success of the identification process.

1.3.1 Hypothesis, Objectives and Research Questions

Finally, to be as specific as possible in the goals and purpose of this thesis project, in this subsection, the hypothesis, the objectives, and the research questions will be presented. First of all, the hypothesis of this project is:

Is it possible to predict if an author will become an Academic Rising Star in the next five years starting from the fifth year since the publication of its first indexed article using the data readily available in Scopus databases?

In other words, this thesis aims to prove that given a set of data retrieved from Scopus it is possible to establish criteria and a methodology to identify Academic Rising Stars. And that same hypothesis leads us to the following objectives:

- **Objective 1.** Establish a method to acquire the necessary data from the Scopus databases while sticking to the constraints and quotas inherent to the use of these databases.
- **Objective 2.** Find the best labeling criteria to establish which authors are considered as top researchers ten years after the publication of its first document in Scopus.
- **Objective 3.** Find the best sets of parameters and features to train a classifier that can predict the top researcher label using only features belonging to the first five years of the author since their first document in Scopus.

- **Objective 4.** Compare our methodology and results with the ones from two of the most recent and similar Academic Rising Star identification efforts[3, 26].

From the previously presented objective, a set of research questions arises. These questions are presented according to the objective they belong to.

- Objective 1:
  - **Q1.** Is it possible to conduct a Scopus query such that it only returns the relevant documents for a particular area of research, use those documents to identify the relevant authors, and gather individual metrics for those authors?
- Objective 2:
  - **Q2.** Which of the available author metrics works best to label the top authors ten years from the publication of its first document in Scopus, and which metric threshold should be used?
- Objective 3:
  - **Q3.** Which combination of labeling, data and parameters yield the best classification performance?
- Objective 4:
  - **Q4.** How does our research compares to the ones by Bin-Obaidellah et. al.[3] and the one proposed by Nie et. al.[26]?

Chapter 2

Background

This chapter is going to be divided into three sections. In the first section, we are going to present the work that has been carried out to predict Academic Rising Stars or some other variation of scientific success. Then, in the second section, the author metrics (or features) retrieved from Elsevier's databases used for this thesis project are going to be presented. In the third section of this chapter, the relevant theoretical framework for this thesis project will be presented, including but not limited to the machine learning classifiers and the metrics used to evaluate these classifiers.

2.1 Previous work

2.1.1 First attempts to predict future scientific impact

One notable effort in the prediction of scientific success, which is related to the identification of Academic Rising Stars was done by Acuna, Allesina, and Kording[1]. The authors intended to use Hirsch's h-index[17] and an additional set of features, to predict future scientific success. To do this, they used linear regression with elastic net regularization[41]. The authors achieved an $R^2 = 0.52$ for predictions ten years from the time frame used to train their models. It is also notable that they used scientists which were 3 to 15 years into their scientific careers. This work is relevant because it showed that the h-index can be predicted to an acceptable extent using a set of established authors. While these results cast a reasonable suspicion that identifying Academic Rising Stars may be possible, it still needs to be tested using the data from scientists in the earliest years of their scientific career. Additionally, identifying Academic Rising Stars is a classification task instead of a regression one. A similar approach was taken by Ayaz et al.[2]. The features used in this work were average citations per paper, number of coauthors, years since publishing their first article, number of publications, number of impact factor publications, and number of publications in distinct journals. The data they used was gathered from the Arnetminer dataset, using publications in the field of Computer Science. While they do not specify which type of regression they used, they achieve an $R^2 = 0.92$ for next year predictions and $R^2 = 0.82$ for predictions in 5 years. Nonetheless, they found out that this method yielded a low R^2 for authors with less than 5 years of experience. This implies that this method does not apply to the identification of Academic Rising Stars, however, it points out the potential of the h-index to be used as a

target variable for the prediction of scientific impact. Notably, as early as 2007, Hirsch had already discussed the potential predictive power of the h-index[18]. Even, to this day h-index is still being used not only to predict but to describe the scientific impact of researchers. For example, the list of Top Scientist by h-index in Computer Science and Electronics by Guide 2 Research[15].

2.1.2 Predicting Academic Rising Stars using bibliographic networks

To the best of our knowledge, the first effort to define and find Academic Rising Stars came in 2009 in Searching for Rising Stars in Bibliography Networks[23]. In this article, a Rising Star is defined as an author which has a low research profile at the beginning of their career but may become a prominent contributor in the future. Their proposed algorithm to detect Rising Stars is called PubRank. PubRank works by mining evolving links in a social network of researchers modeled by a bibliography network. In this work, every researcher is modeled as a node and the links represent a collaborative relationship (in other words, a joint publication). Then, the algorithm evaluates the mutual influence among the researcher in the network, the track record of the researcher, and the chronological changes of the network. Then, the algorithm assigns a PubRank value to each node (or author), and when a researcher shows an above-average increment it is considered a Rising Star. It is important to mention that the PubRank algorithm is based on PageRank[4]. PageRank is a webpage ranking algorithm, which works by ranking webpages based on the hyperlinks among the webpages, or in other words, how relevant is the webpage. The main difference between PageRank and PubRank is that while the PageRank of a node is dependent on the nodes that link to it, the PubRank is dependant on the nodes to which it links to. So, in simpler terms, PubRank ranks higher the nodes (authors) which influence other nodes. In 2013, in Finding Rising Stars in Social Networks[7] improvements were proposed to PubRank. In these improvements, the author's contribution-based mutual influence and dynamic publication venue scores are included in the calculation of the PubRank value. The authors named the resulting algorithm StarRank.

Then in CocaRank: A Collaboration Caliber-based Method for Finding Academic Rising Stars[38]. One of the main flaws of PubRank and StarRank is that there is no way to tell if young authors were being compared to more senior researchers. CocaRank implements the publishing year of the first paper as the start of the Rising Star Evaluation. CocaRank is calculated in three parts. In the first part, the collaboration caliber is computed, which is based on an entropy calculation that has the purpose of representing the ability of an author to collaborate with other authors. Then, it calculates the PageRank value in a heterogeneous academic network. Contrary to PubRank and StarRank which are based on a bibliographic network, CocaRank a three sub-network architecture comprised of a citation network, a paper-author network, and a paper-journal network. Finally, these results are used to calculate the final CocaRank. To validate the results of CocaRank, the future citation counts of the authors with the highest CocaRank (thus, being considered Academic Rising Stars) are compared to the future citation counts of the authors with the highest StarRank. The authors that CocaRank identified as Academic Rising Stars showed a significantly greater number of citation counts in the future than those authors identified by StarRank, thus taking a significant step forward in the identification of Academic Rising Stars. Further along the road, ScholarRank[37] was

introduced, building on the heterogeneous network proposed by CocaRank. The only difference between this approach and CocaRank is that the heterogeneous network is evaluated using a mutual reinforcement process of the sub-networks. This method achieved an improved number of future citation counts compared to CocaRank.

2.1.3 Predicting Academic Rising Stars in the Machine Learning era

One of the first efforts to predict Academic Rising Stars using some kind of machine learning method is found on Social Gene - A New Method to Find Rising Stars[27]. In this article, the authors propose the use of 14 different features to calculate a series of underlying author's parameters known as "social genes", associated with the "talents" of a given author. First, these features are fed to an Analytic Hierarchical Process, and then the weights are determined using a neural network with sigmoid functions as activation functions. Finally, the ranks are determined using the factor weights. All in all, while this first attempt was innovative at the time, it was only compared with PubRank, and while Social Gene did outperform it, it was not compared to the state of the art Academic Rising Star identification methods of the time.

However, one of the greatest breakthroughs in the use of machine learning to predict Academic Rising Stars is found in Academic Rising Star Prediction Via Scholar's Evaluation Model and Machine Learning Techniques[26]. In this article, for the first time, the prediction of Academic Rising Stars is not completely treated as a matter of future citation counts. Instead, it is considered a classification task. The data used in this project comes from the Arnetminer database. In a classification task, the positive causes need to be labeled before the classifier can be trained. In this case, the authors labeled as positive cases are the ones considered outstanding or influential at present. The approach that this work uses to label the positive cases is based on the increment in academic impact scores in a time window of five years, starting on the fifth year since the publication of its first article. This impact score is based on their papers' scores, the scores of the papers that cite the author's papers, and a score based on the author's contribution to each published paper based on the s-index[31]. Once the authors are labeled, the classifiers are trained using features from the first five years since the publication of the author's first article. These features comprehend author, social, venue, and temporal characteristics. The classifiers used in this work were k-Nearest Neighbor, Random Forest, Support Vector Machine, Gradient Boosting, and XGBoost. The best F1 scores achieved in this work ranged from 0.784 to 0.795.

Another notable work in the identification of Academic Rising Stars using machine learning came in Scientometric Indicators and Machine Learning-Based Models for Predicting Academic Rising Stars in Academia[3]. In this article, data from Web of Science is used to predict Academic Rising Stars, and again the prediction of Academic Rising Stars is treated as a classification task. To label the positive cases for the classification task, the authors choose the top 30% of the available authors according to their last year increment to their InCites Ranking, which ranks the authors according to the times they are cited and the number of documents indexed in Web of Science. Then, 8 different scientometric indicators are calculated for each author and two classifiers are trained, Support Vector Machine and k-Nearest Neighbors. This work achieved an AUC of 0.96 with a Support Vector Machine.

Finally, this thesis project will build on these two previously presented Academic Rising Star Prediction works. The methodology that will distinguish this research project from these

two previous machine learning-based works will be discussed in detail in Chapter 3. Additionally, the difference in results and their explanation from our work with previous work will be thoroughly discussed in Chapter 5.

2.2 Elsevier Retrieved Metrics

For this Academic Rising Stars identification project, several author metrics will be used. In the section, these metrics will be explained in detail according to the data provided by Elsevier's Research Metrics Guidebook[11]. In this document additionally to author metrics, other Elsevier's available metrics are also presented, journal metrics, article-level metrics, and institutional metrics. All of the discussed metrics are retrieved through the SciVal API for this thesis project.

1) Publications is a metric that shows how many publications does an author has indexed in Scopus. In the SciVal API, this metric is retrieved as Scholarly Output, but for the rest of the project, this metric will be referred to as *Publications*. One of the limitations of these metrics comes from comparing authors with different scientific career lengths, as more established authors are more likely to publish more documents than their peers who are in the earlier stages of their careers. This metric is retrieved on a per-year basis, which means that the metric shows how many documents an author published per year. Although it is possible to filter the types of documents counted towards the metric, in this case, we choose not to do so, as we considered important this metric to represent every type of document that the authors produced.

2) h5-index is a metric based on the original h-index[17], however, it only evaluates a 5-year publication and citation window on the calculation of the h-index. For example, the h5-index of an author in 2018 takes only into account the publications made from 2014 to 2018 and the citations received by those same publications from 2014 to 2018. While some published works have acknowledged the shortcomings of the h-index[9, 30], earlier research[1, 2], has shown that it is a good metric to represent scientific impact. By using the h5-index, we have the advantage of removing the possibility of the metric being biased by old publications and citations.

3) Citation Count is a metric that describes the total number of citations the author's publications have received. This metric is obtained on a per-year basis. However, it is very important to take into account that this metric refers to the total citations the publications in a determined year have received so far. In other words, this metric does not reflect the year in which these citations were received. Thus, the usefulness of this metric may be limited, for example, the Citation Count of an author in 2010 may be influenced by citations received years after 2010. Nonetheless, the metric is integrated into this project as every author in the dataset will be affected in the same way, so the effect of this phenomenon will be reasonably uniform across the retrieved dataset.

4) Field-Weighted Citation Impact (FWCI) indicates how the number of citations received by an author compares with the average number of citations received by all other similar publications. Therefore an FWCI of 1 indicates that the author's publications have received the average number of citations it is expected from them. An FWCI above 1 indicates that the publications have received more than the average number of citations it is expected from

them, while an FWCI below 1 indicates the contrary. For example, an FWCI of 1.5 indicates that the author's publications have received 50% more citations than the world average, while an FWCI of 0.25 indicates that the author's publications have received 75% fewer citations than the world average. Scopus assigns each publication in its database to one or more classification sub-categories. To avoid that the FWCI calculation is biased in favor of publications with multiple categories, it spreads symmetrically the publication's citations among the subcategories it belongs to. As with Citation Count, the yearly representation of this metric refer to the years in which the publications were published and not to the year where the citations were received. The FWCI formula is presented in equation 2.1, where N is the number of publications, c_i is the citations received by publication i, and e_i is the expected number of citations received by all similar publications in the publication year and the following 3 years.

$$FWCI = \frac{1}{N} \sum_{i=1}^{N} \frac{c_i}{e_i} \tag{2.1}$$

5) Outputs in Top Citation Percentiles (OTCP) is a metric that represents the percentage of an author's publications are present in the top most-cited publications. This metric is retrieved for the top 1%, 5%, 10%, and 25% most-cited publications. To define these top most-cited threshold Scopus calculates the percentiles of the number of citations that the publications have received, thus more than 10% of the total publications in the Scopus database can be in the top 10% most-cited publications, for example. Additionally, this metric is calculated per publication type. As with the previous metrics, the year of the metric accounts for the citations that a publication published in a certain year has received so far, not the year the citations were received.

5) Publications in Top Journal Percentiles (TJCP represents the percentage of an author's publications in the most-cited journals in Scopus' data universe. This metric represents how many publications does an author has in the top 1%, 5%, 10%, and 25% most-cited journals indexed in Scopus. The method used in this thesis project to define the most-cited journals is the CiteScore[21]. Since Scopus did not calculate CiteScore before 2011, this metric is calculated with the CiteScore of 2011 for publications published before 2011. It is important to point out that a publication that did not receive any citations may be counted towards this metric as long as it is published in one of the top most-cited journals.

6) Citations per Publication measures the average number of citations that the author's publications have received. It can be considered that this metric indicates the average citation impact of the author. As with the previous metrics, the year of the metric accounts for the citations that a publication published in a certain year has received so far, not the year the citations were received. It is calculated by adding all of the citations certain year publications have received and diving that number by the total number of documents published in that certain year.

7) Cited Publications measures the citability of an author. In other words, this metric shows the percentage of publications that have received at least one citation. As with the previous metrics, the year of the metric takes into account if the publication has received any citations regardless of the year the citation has been received.

2.3 Machine Learning Background

2.3.1 Normalization Methods

Before data is fed to a machine learning algorithm, in this case, classifiers, it may be convenient to normalize the data. In this thesis project, three normalization methods are used. These three methods are MinMax Scaling, Standard Scaling, and Robust Scaling. The implementation used is the one provided by Scikit-Learn[28].

MinMax Scaling

Min-Max scaling[32] replace every value in the feature being transformed with a new value. This value is found between 0.0 and 1.0, and the formula to calculate the new value is found in Equation 2.2, where m is the new value, x is the actual value of the feature, x_{min} is the minimum value in the dataset for that feature, and x_{max} is the maximum value in the dataset for that feature.

$$m = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{2.2}$$

This normalization method places every value within the 0 to 1 range. This causes that the weight of the outlier values in the normalized feature is taken away to a certain extent. This may end up hurting the performance of the model in some cases.

Standard Scaling

Standard Scaling, also known as Z-score standardization[32] transforms every value in the feature with its z-score. This means that the features are rescaled to have a mean of zero and a standard deviation of one, resulting in all of our features having a uniform mean and variance. The formula to calculate the new value, or z-score is found in Equation 2.3, where z is the new value, x is the original value of the feature, μ is the mean of the feature, and σ is the standard deviation of the feature.

$$z = \frac{x - \mu}{\sigma} \tag{2.3}$$

Robust Scaling

The Robust Scaling[28] works somehow like the Standard Scaling, however, instead of using the mean and the standard deviation, it uses the median and the interquartile range of the features. The formula is shown in equation 2.4, where r is the new value, x is the original value of the feature M is the median of the feature and IQR is the interquartile range, which is defined as the 75 percentile minus the 25 percentile.

$$r = \frac{x - M}{IQR} \tag{2.4}$$

2.3.2 Recursive Feature Elimination

Recursive Feature Elimination[22] is a backward selection of the predictors. The goal of this technique is to remove the least important predictors to then train the model. This technique works by building a model, then the least important feature is removed, and this is repeated recursively until the desired number of features is reached. For this method to work, an estimator model needs to be selected, so the Recursive Feature Elimination can use the feature coefficients calculated by the estimator model.

2.3.3 Support Vector Machine

One of the two main classifiers used in this thesis project is the Support Vector Machine (SVM) with a Radial Basis Function[5] (rbf) kernel. Support Vector Machines were developed as support vector networks by Cortes and Vapnik[6]. The idea behind support vector machines is that input vectors are non-linearly mapped to a high-dimensional feature space where a linear decision surface is constructed. In this case, the decision surface is used to classify data into two classes. So the purpose of the support vector machine is to optimize this decision surface in such a way that the widest marginal possible divides the two classes[35].

However, in some classification cases, the points in a dataset are not linearly separable, that is why SVM uses the kernel trick technique, which transforms the data into a higher-dimensional space, so this transformation can provide a dividing margin between the classes[35]. The kernel used in this project, as previously mentioned is the rbf kernel, which gives a value to each data point based on its distance from a fixed center, on a Euclidean space[35]. SVM results in a very convenient classifier for this classification task as it works well with a high number of features and the decision surface is only affected by the support vectors (which means that the outliers have a lesser impact).

2.3.4 Logistic Regression

Logistic Regression[14] is commonly used for the classification of observations, and the purposes of this analysis aim to predict the effect of a set of variables on a binary response variable. Or in other terms, it classifies the observations in a specific category based on the probability estimation. One important concept in logistic regression is the odds ratio, which denotes how the odds for one independent variable with the increase of a unit in that variable, while the rest are kept constant. So in simpler terms, the odds ratio is how likely is a specific result in the dependant variable if one of the independent variables changes.

However, it is important to mention that the Logistic Regression classifier makes some assumptions. The first assumption is that there should not be any outliers in the data, because the models are very sensitive to outliers in the data. The second assumption is that there should not be a high correlation between the independent variables. This assumption can be considered met if the correlation coefficients between the independent variables are less than 0.9[34]. Additionally, it is recommended[33] having 20 observations per independent variable. However, it is also argued[19] that 10 observations per independent variable may still be appropriate.

	Actual Positive Class	**Actual Negative Class**
Predicted Positive Class	True Positive (tp)	False Negative (fn)
Predicted Negative Class	False Positive (fp)	True Negative (tn)

Table 2.1: Confusion Matrix for Binary Classification

2.3.5 Classification Metrics

To assess the performance of the trained classifiers, a set of metrics is going to be used. All of the presented metrics are calculated using their respective implementations in scikit-learn[28]. For this project's binary classification task the first metric that we calculate is the *Confusion Matrix*[20]. This matrix shows how many True Positives (tp), False Negatives (fn), False Positives (fp), and True Negatives (tn) has the trained classifier predicted. True Positives and True Negatives refer to the positives and negatives that the classifier has correctly classified, while the False Positives and False Negatives refer to the positives and negatives that the classifier has incorrectly classified. In Table 2.1, a representation of the Confusion Matrix is shown.

From these four metrics in the Confusion Matrix, it is possible to derive other metrics. While not every one of them is presented in this thesis project, they are still used to derive even more metrics in some cases. The first of these metrics is the *Accuracy*[20] which measures the ratio of correct prediction over the total number of instances evaluated. The formula for Accuracy is shown in Equation 2.5. In cases where the classes are not balanced, this metric can be deceiving, as it is possible to have a high Accuracy if the model is very good at classifying the negative cases but not the positive cases when the number of negatives cases is overwhelmingly greater than the number of positive cases.

$$Accuracy = \frac{tp + tn}{tp + fp + tn + fn} \tag{2.5}$$

The *Precision*[20] measures the positive cases which are correctly predicted from the total predicted cases in the positive class. In simpler terms, this metric indicates the proportion of true positives in the positive class predicted by the classifier. The formula for Precision is shown in Equation 2.6.

$$Precision = \frac{tp}{tp + fp} \tag{2.6}$$

The *Recall*[20] is used to measure the fraction of positive cases from the total positive cases in the positive class. The formula for Recall is shown in Equation 2.7.

$$Recall = \frac{tp}{tp + tn} \tag{2.7}$$

The *F1 score* or F-measure[29], is defined as the harmonic mean of the Precision and the Recall. The formula for the F1 score is shown in Equation 2.8. This is a very useful metric because it presents a good balance between how good is a classifier at detecting the available positive cases and how precise it is at doing so.

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{2.8}$$

The *Average Precision Score*[40] is a metric that summarizes the Precision-Recall curve in a single value, specifically the area under the Precision-Recall curve. What this curve represents is the precision of the model at different classification thresholds. This metric is specifically useful for class imbalance. The highest the metric, the greater the area under the Precision-Recall curve, and thus it can be concluded that the better the model is at identifying a good proportion of the positive cases with good precision.

The *Area Under the Receiver Operating Characteristic Curve (AUC)*[12] summarizes the ROC curve in the form of the area underneath it. The ROC curve plots the false positive rate versus the true positive rate. The true positive rate can be considered as a measurement of how good is the model at predicting a positive case as positive. The false positive rate represents how often a negative case is predicted as negative. The ROC curve represents the trade-off between the true positive rate and the false positive rate at different decision thresholds. The highest the AUC, the better is the model at detecting true positives while detecting as few false positives as possible.

2.3.6 Post Hoc Tests

As part of this thesis project, we are to determine which parameters have a significant impact on the performance of the trained classifiers. To find if the variation in the value of these parameters brings a statistically significant difference among the classifiers trained with the different values in a specific parameter, the Autorank[16] library will be used. What Autorank does is running post hoc tests which then compare the central tendencies of the different classifier groups as paired samples. Fortunately, Autorank automatically decides which statistical test should be carried out depending on the characteristics of the classifier groups.

When comparing a parameter with only two possible values, then two groups of classifiers are compared. In this case, Autorank chooses to compare the two groups using Wilcoxon's signed-rank sum test[36]. This is a non-parametric test (the compared population does not need to be normally distributed) that is used to test the hypothesis that the probability distribution of the first population is equal to the probability distribution of the second population. If this null hypothesis is rejected, then it can be concluded that the two populations are different. And thus, for this thesis project, the variations in the values of the parameter have a significant effect on the performance of the classifier.

The other relevant case for this thesis project is comparing more than two populations that do not have a normal distribution. This is done for parameters with more than two possible values. In this case, Autorank chooses to compare these groups with a non-parametric Friedman test[13] and a Nemenyi post hoc test[25]. The non-parametric Friedman test is used to determine if there are significant differences between the median values of the populations. The null hypothesis of the non-parametric Friedman test states that there is no difference in the central tendencies of all of the populations being compared. When the null hypothesis is rejected, it is because at least two of the populations being compared have a significant difference in their medians. Then, the Nemenyi test established a critical distance between the mean ranks of the populations that are being compared. If the absolute difference between

the mean ranks of two or more populations is smaller than the critical distance provided by the Nemenyi test, then these populations are not significantly different. In the context of this thesis project, this test indicates to us which values for the different parameters being tested are not significantly different from each other. Additionally, the results of these post hoc tests are summarized in a convenient table. These table shows the Mean Rank (MR), the Median (MED), the Median Absolute Deviation (MAD), the Median Confidence Interval (CI), the effect size in comparison to the highest ranking approach using Akishin's gamma (γ), and the interpretation of the γ value in the Magnitude column.

Chapter 3

Methodology

3.1 General Methodology

This thesis project aims to utilize Machine Learning to identify Academic Rising Stars. Carrying out this project comprehends from acquiring the relevant data, processing it, classify it, and finally assess the extent to which the classification was successful. More specifically the methodology steps are the following:

- **Building the Dataset:** Gathering information on the relevant authors from the Scopus database
- **Preparing the Data:** Who will be labeled as a top researcher?
- **Train the Classifiers:** Attempt to predict with data from early stages in an author's career if it becomes a top researcher.
- **Evaluate the Classifiers:** To what extent were the classifiers able to predict if an author becomes a top researcher? (Thus, testing the classifiers ability to predict the Academic Rising Star status)

In the following sections of this chapter, the intricacies of each step will be approached in detail. Nonetheless, in this section, the process will be outlined in general, and some of the most important details will be mentioned. This to present the reader with a general and clear overview of the methodology, while providing information on the complexity involved in each step. First, building the dataset has its complications, since the data is retrieved through an API that limits the amount of information that can be downloaded, and also the speed at which this information can be downloaded. So, to gather the necessary information to build the dataset, a sensible and measured approach to this task had to be taken. In this process, it is important to make the correct queries to the API, so as not to waste the limited amount of data that can be retrieved. The gathering process starts with retrieving the metadata of all of the documents in the domain or field we are interested in finding the Academic Rising Starts. Then, from the metadata retrieve which authors publish significantly in this domain or field, and retrieve the relevant metrics for each of these authors.

Once the data on the relevant authors is acquired, then the data needs to be prepared. This is important, since classifiers to be trained, need to know which authors are the ones it

is looking for. In this case, the authors who excel among their peers in certain metrics are the ones labeled as top researchers, and as such are the ones considered as the positive class. Different criteria are used to label these top researchers, and in the following sections, we will explain in a detailed fashion which metrics were the ones chosen to label data and which was criteria used on these metrics. However, several datasets are generated, with the only difference being how the authors are labeled. This is to test which metrics or attributes are the most best suited to point out who is more likely to become an Academic Rising Star. It is also important to clarify that this labeling is done, based on the latest year metrics. Additionally, to ensure we are ”comparing apples to apples”, only authors whose first publication was done 10 years ago are taken into account for this labeling process, as done in previous Academic Rising Star identification works[26, 38].

When the datasets are labeled, then it is possible to train the classifiers under different parameters. Since this Academic Rising Star identification problem involves a heavy class imbalance (which is obvious, as it is not possible to have the same number of top researchers and non-top researchers, the fact that these are top researchers imply that they are scarce), one of the various parameters that are tweaked is the class weight. Where a greater weight is assigned to the positive class during the training of the model. As previously mentioned, different labeling sets are also used. This results in a great number of trained classifiers with different parameters and different data. Additionally, the features used to train these classifiers only comprehend the metrics of the first five years from the year of the author's first indexed publication. Since we are interested in finding out which authors have the potential to become top researchers in the following five years.

Finally, the results of these classifiers are analyzed. In this case, the classifier must be able to identify as many top researchers as it possibly can while keeping the number of false positives at a low and reasonable number. The two main metrics that will be used to evaluate these models are the Area Under the ROC Curve (AUC) and Average Precision Score. It is also important to mention that more than one classifier will be used. This Academic Rising Start identification exercise will be done using the Logistic Regression Classifier and Support Vector Machine.

3.2 Building the Dataset using the Scopus API

The identification of Academic Rising Stars has to start somewhere, and in this case, the first step is to know what kind of Academic Rising Star we are looking for. If we were a university that wants to hire a new researcher for their Computer Science department in the area of Clustering, then it would be convenient to know who is publishing in the Clustering area. Then, we should retrieve the documents from Scopus which deal with Clustering, through the Scopus API. The number of documents retrieved from Scopus depends largely on how specific is the query being made. Searching for the documents in the Computer Science area from the last 10 years would result in millions of documents that would have to be downloaded. It can be done, however, Scopus restricts the number of documents that any user can download. As such, it is recommended to delimit the areas to something more specific. In this case, the Scopus query that was conducted in the Study Case presented in Chapter 4 is related to the Clustering Area and it resulted in more than 50,000 documents.

Scopus ID	Title	Publication Name	ISSN
ISBN	EISSN	Volume	Page Range
Cover Date	DOI	Citation Count	Affiliation
Aggregation Type	Subtype Description	Authors	Full Text

Table 3.1: Retrieved Features per document from a Scopus Query.

The retrieval of documents was done through the *pybliometrics* library. This library allows the user to retrieve the documents in a very convenient JSON format, which can be then easily converted into a CSV database. The features in this document database are shown in Table 3.1. In this particular case, the feature we are interested in is the Authors feature. This feature represents a list of Scopus authors' IDs. Then, the unique authors in these lists are retrieved, and the number of times their ID appeared in the documents is paired with their ID. This allows us to know how many documents have that specific author published in our Query. Additionally, this Scopus author ID will allow us to retrieve the metrics related to the author. However, it was previously mentioned that it is important to conduct the retrieval of information in the most efficient way possible. In this case, it means not retrieving the metrics for those authors who are not relevant to the area of the query. Not relevant authors are considered in this case the authors that have two or fewer documents in the document query. Trimming these authors from the subsequent analysis saves a significant amount of time and avoids the query quota to be wasted on authors who are not even publishing constantly in the area of interest.

Then, for the remaining authors, their metrics are retrieved. These metrics are retrieved using the Scopus API and the SciVal API. Since we want to keep the retrieval of metrics as efficient as possible, the author's IDs are bundled into groups of 100 authors, since Scopus and SciVal allow us to retrieve the metrics of 100 authors in a single query, which is faster and more efficient than retrieving the metrics of each author one by one. However, one query is still needed per type of metric. The only author information retrieved for the authors from the Scopus API was the publication range. This reports the year in which the author's first indexed publication was made and when was the last one made. All of the other author metrics were retrieved using the SciVal API. The retrieved author metrics are shown in Table 3.2. All of the retrieved metrics were retrieved per year from 2010 to 2019. Additionally, it is important to note that one of the limitations of the SciVal API is that it only allows the retrieval of metrics from the last ten years. It is not possible to retrieve any metric from previous years, which limits this ten-year analysis to the 2010-2019 time frame exclusively. If it was possible to retrieve earlier metrics, it would be possible to have even more time frames (2009-2018, for example). Another important consideration is that, although SciVal offers the option to exclude self-citation from the calculation of each one of the metrics, it was chosen not to exclude self-citation. The reasoning behind this is that while self-citation may inflate to an extent the metrics of authors with a considerable scientific career, it may not be the case for authors in the first years of their scientific career.

All of these metrics were retrieved in JSON format, which was then easily used to build the authors' dataset in CSV format. This is the data that will be then used to train the different classifiers. However, the data still has to be processed, to be used in the training of the

Metric Abbreviation	Full Metric Name
fwci	Field Weighted Citation Impact
citCount	Citation Count
citPP	Citations per Paper
citedPub	Cited Publications
publications	Publications
OTCP01	Outputs in the Top 1% Citation Percentiles
OTCP05	Outputs in the Top 5% Citation Percentiles
OTCP10	Outputs in the Top 10% Citation Percentiles
OTCP25	Outputs in the Top 25% Citation Percentiles
TJCP01	Publications in the Top 1% Journal Percentiles
TJCP05	Publications in the Top 5% Journal Percentiles
TJCP10	Publications in the Top 10% Journal Percentiles
TJCP25	Publications in the Top 25% Journal Percentiles
h5index	h5-index

Table 3.2: Metrics retrieved per author using the SciVal API

classifiers.

3.3 Data Preparation

Once we have the metrics for each of the authors, the next step is to prepare the data for the classifiers. The first step is ”comparing apples to apples”. Since we are targeting authors whose first publication came 10 years ago to make predictions based on the first five years since that publication is made, the only time frame that can be used given the limitations of the SciVal API is 2010-2019. That is why, the first step is to separate those authors who started 10 years ago, from the rest of the authors, and these are the ones that are going to be used for the classification task.

Another important part of preparing the data is the labeling of the data. In simple terms, we want to point out who is a top researcher on the present (2019), and then use their metrics for the first five years since the first publication was made (2010-2019). This is one notable difference this Academic Rising Star identification effort has with other machine learning-based efforts. We want to make sure that the authors that we identify as Academic Rising Stars are still at an early stage in their careers. For example, in Scientometric Indicators and Machine Learning-Based Models for Predicting Rising Stars in Academia[3], takes a very different approach to identifying Academic Rising Stars, where they take a five-year time frame, and then try to predict if the author the next year after those five years. Additionally, the top 30% of authors according to their InCites rank (which is provided by Clarivate Analytics) are labeled as Academic Rising Stars. Which in contrast to our proposed methodology, does not ensure that the predicted Academic Rising Star is in an early stage of its scientific career. Moreover, our proposed methodology aims to predict if a young author became a top researcher in the next five years, in contrast to the next year approach, more like the approach

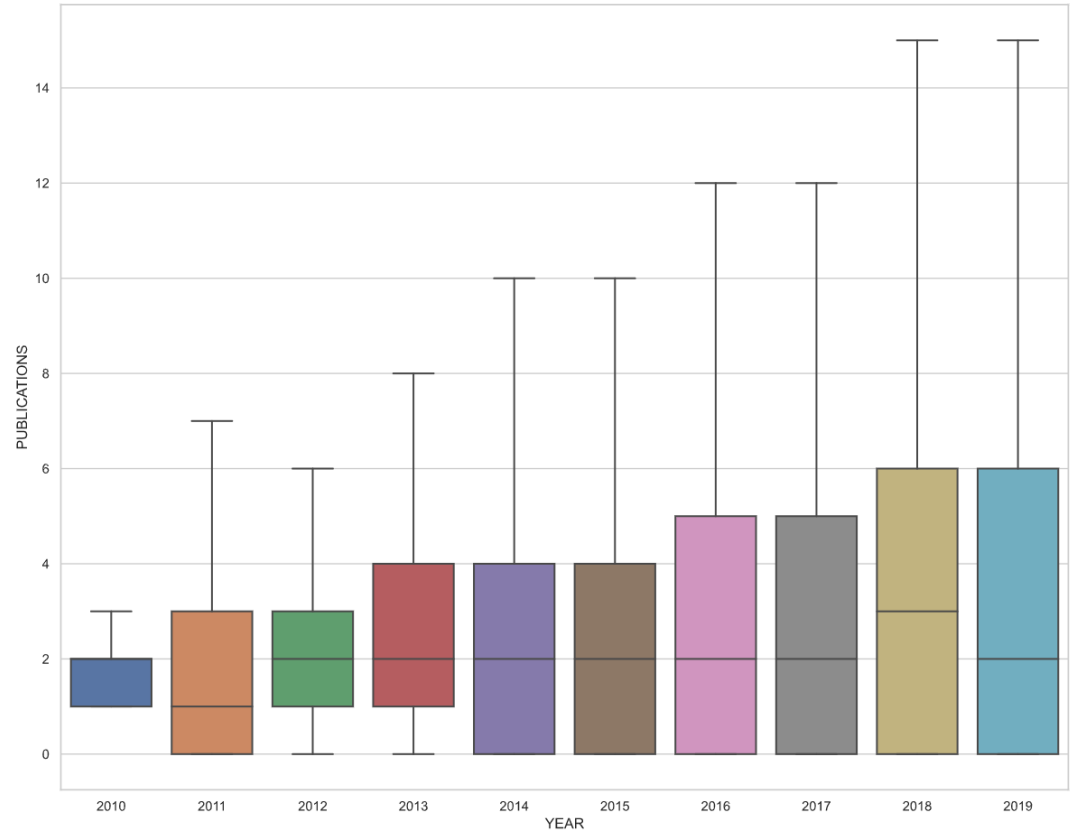


Figure 3.1: Yearly distribution of published articles per year by the authors whose first publication came in 2010.

taken by Nie. et. al.[26]. This aims to make our models more flexible and ensure that our identified Academic Rising Stars are at an early stage in their careers, and as such, they are easier to hire by the interested academic institutions. Additionally, to confirm that the year of the first publication of the author is a viable indicator of the starting point of the scientific career of a researcher, the number of publications per year for all of the authors who started in 2010 was plotted in boxplots in Figure 3.1. By looking at the boxplots it shows that while there are authors who do not publish in certain years, the median stays at two articles per year from 2012 onward. Therefore, the boxplots support our assumption of making the year of first publication the year where we assume that the author's scientific career started. It is important to note that the boxplots in Figure 3.1, exclude outliers, to make the boxplots easier to visualize.

Another important part of data preparation is the criteria we use to define who is a top researcher. Four metrics were chosen to label the top authors, publications, citCount, fwci, and h5index, all of these in 2019. In contrast to the approach taken by Nie et. al.[26] to label the data, we simplify this process by not creating additional metrics to measure who is a top author. However, we still need to define a threshold in these metrics, so authors whose metrics exceed these thresholds are labeled as top researchers. Two approaches were taken for this task. First, the outlier approach, which consists on calculating the median, the third quartile, and the interquartile range (which is $Q3 - Q1$), to define the top researcher threshold at the

outlier point of the metric. The equation to define the outlier point is the following:

$$Threshold = Q3 + 1.5(Q3 - Q1)$$

So in this case, if an author's metric is greater or equal to the threshold, then it is labeled as a top researcher, and the datasets labeled with this approach were referred to as the Outlier group. However, it was observed that in some cases there were authors whose metrics were considerably well beyond the outlier threshold. To deal with these extremely outlying authors, a second approach was developed. This second approach consists of calculating the outlier threshold for the metric, just as before, but the authors who are going to be labeled as top authors are those whose metrics are within a range of $\pm 35\%$, of that threshold. This labeling group is referred to as the Custom group. With this approach, authors who are beyond the outlier threshold are still labeled as top authors, and some authors who are reasonably above the median of the metric, but not above the outlier threshold are still labeled as top authors. The $\pm 35\%$ range can be tweaked, but we found that it worked well. So the result of this labeling criteria is 8 differently labeled datasets, two for each of the chosen metrics, one labeling the outlier group and the other the custom group. The range of $\pm 35\%$ for the custom group was empirically obtained from the case study presented in Chapter 4, as the aim was to classify as positive cases as close as possible to 10% of the authors.

Furthermore, each author has 70 metric features per year, only taking into account the metrics from 2010 to 2014. Training classifiers using these 70 features could make it hard to explain the models. That is why, as part of the data preparation two additional sets of features are added to the datasets. These two sets are the Average (AVG) feature set and the Median (MED) feature sets. These two sets reduce the number of attributes from 70 to 14, by calculating the average and the median of each type of metric from 2010 to 2014. This has the added benefit of resulting in models with fewer features and thus, easier to explain. The feature set with the 70 features is referred to as the ALL feature set.

Finally, the last step of the data preparation is data normalization. From all of the existing datasets up to this point, three additional variations of each of these datasets are created, one for each of the normalization methods. These methods are MinMax normalization, Standard normalization, and Robust normalization. All of the features, except the class feature, are normalized. This is done to find out which type of normalization helps the classifiers to perform better.

3.4 Classification

Once the data is prepared, all there is left to do is train the different classifiers. In these cases, two classifiers are going to be used Logistic Regression and Support Vector Machine. The Logistic Regression will use *liblinear* as its solver because it tends to perform better in small datasets, although it can be slow in greater datasets. Since we do not expect to deal with huge amounts of data, liblinear is the ideal choice. On the other hand, for the SVM classifier, as kernel, the Radial Basis Function Kernel (rbf) will be used, as it performs well in non-linear relationships. These two classifiers are chosen for their robustness in binary classification tasks. Since Logistic Regression is fitted with the logarithmic odds of each feature, it can model each feature as a probabilistic contributing factor to the positive or negative prediction

of the Academic Rising Star status. Nonetheless, Logistic Regression has some limitations in terms of the amount of data it is trained with, as explained in Chapter 2. For this reason, the Support Vector Machine with the rbf kernel is also implemented. This kernel is robust against the presence of outliers in the data and is also well-suited to work with a limited amount of data. Since the time it takes to train the number of models for each classifier is considerable, it was decided to limit the classification efforts to these two classifiers.

One additional step before training the classifiers is the Recursive Feature Elimination. There may still be some colinearity among the features, and this can impact negatively the performance of the classifiers. That is why before training the classifiers, three types of Recursive Feature Elimination stages are put in place. This process is done with ten-fold cross-validation, to ensure that the features which are not significant for the model are dropped from the features used to train the classifiers. The estimators used by each of the Recursive Feature Elimination stages are Logistic Regression, Perceptron, and Decision Tree. A model is trained for each of the estimators, to find which estimator does better at eliminating the irrelevant features.

Then, the models with the features eliminated are trained with the whole dataset and varying the weight of the target class from 1.0 to 2.0. This is done to deal with the class imbalance associated with Academic Rising Star prediction. So, three additional models are trained, one with a weight of 1.0, the other with 1.5, and the last one with a weight of 2.0. Once the model has been trained several metrics are calculated. These metrics are the following:

- Accuracy
- Precision
- Recall
- Specificity
- AUC
- Average Precision Score

However, the ones that we are going to take more into account are AUC and Average Precision Score. These metrics are stored in a new models dataset. Additionally, the confusion matrix of each model is added to the dataset. And finally, the features used by the model and the total number of these features are added to the dataset. This data will prove useful to analyze our results in Chapter 4. These metrics will then help us to know if models capable of predicting Academic Rising Stars were produced, and if so how do these models look.

As previously mentioned, the classifiers are trained with the whole dataset and tested with the whole dataset. Although, it is not a good practice to test a model with the data it was trained with, in this case it was necessary to do this. The reason behind this decision is the small amount of data available for these classification tasks. However, given the results, it can be argued in favor of the validity of this approach. This will be further discussed in Chapter 5

Chapter 4

Case Study: Clustering

4.1 The Scopus Query

In this case study, the chosen field of knowledge is *Clustering* from the Computer Science domain. This field of knowledge was chosen for two main reasons, its size and its similarity with previous datasets used to identify Academic Rising Stars. One of the limitations faced during the retrieval of data is that it is complicated to retrieve more than 50000 documents, and it was found that the field of Clustering at the time was very close to that figure. On the other side, other previous Academic Rising Star identification efforts[26, 3] have used Computer Science datasets, and choosing a dataset from the same field could provide beneficial when comparing results with these previous efforts. The Scopus Query to retrieve the data was the following:

KEY ('CLUSTERING') AND SUBJAREA (COMP) AND PUBYEAR >2009 AND PUBYEAR <2020 AND DOCTYPE (AR) OR DOCTYPE (CP)

The elements of the Scopus Query limit the amount of documents' metadata to be retrieved by Scopus. It is also important to explain the query element by element, to fully understand which documents are being used in this case study.

- KEY: The Scopus Search is limited to documents that contain 'Clustering' in the keywords section.
- COMP: The Scopus Search is limited to documents that are classified in the Computer Science area.
- PUBYEAR: The Scopus Search excludes any document before 2010 and after 2019. These elements are exclusive, so even though the query uses the years 2009 and 2020, these will not be included in the retrieved documents.
- DOCTYPE: The Scopus Search excludes documents that are not classified as articles (ar) or conference papers (cp)

It is important to mention that these are not the only keywords that can be used to retrieve documents' metadata from Scopus. However, it was determined after some trial and error that the most relevant results were retrieved using this query. On the date this query was conducted (09/09/2020), it returned 50269 documents.

4.2 Data Exploration and Preprocessing

After retrieving the documents' metadata via the Scopus Query, to classify Academic Rising Starts it was still needed to process and filter the unique authors found in the query. In the following subsections, it will be discussed the criteria used to determine which authors were considered relevant in this case study. Although it will not be discussed in this section the retrieval of the individual metrics of those authors, as it was already explained in chapter 3, in the following subsections the labeling process and the normalization of the data will be discussed.

Furthermore, the behavior of the variables will be discussed in the following subsections too. Since the classification process will only be done using authors whose first publication came in 2010, the analysis of the different author metrics will only comprehend the previously mentioned authors.

4.2.1 The Authors

Once every document was acquired from Scopus, a new dataset of unique authors was built using the documents' metadata. Not so surprisingly, 95446 unique authors were found. This is due that rarely scientific articles are published with single authorship. That is why it is not so surprising to find that there are almost double the authors than there are documents. However, it is also important to mention that in this case the authors are being retrieved by their unique Scopus Author ID. Due to this ID identification process, it was suspected that there might be IDs that refer to an author that is already referred in the dataset with another ID. However, these cases were deemed to be marginal and Scopus does a very good job unifying the authors by ID. Therefore, most of the cases of repeated IDs, are negligible, in this dataset. This conclusion was reached by sorting the author's dataset by most documents in the original document query and looking at the ones with the most documents in Scopus' web interface. Although these authors usually went by more than one Scopus ID, the number of documents in the interface and the dataset were the same. Thus, it was concluded that most likely this problem does not occur in the dataset and if it does, is most likely for authors with very few articles and not a lot of impact.

Once the authors and the number of documents that they have in the query have been determined, it is important to filter those who do not have published a significant amount of articles. As previously mentioned, this is done to remove the authors who do not publish often in the Clustering area, therefore avoiding retrieving the metrics of those authors, saving valuable time. In Figure 4.1 it can be appreciated that most of the authors in the query have published less than 20 documents in the area of Clustering.

Not all of these authors are relevant for this case study since they have published a very small number of articles in the area we are interested in. After some trial and error, it was seen fit to trim from the dataset those authors who had two or fewer articles in the author's dataset. This helped tremendously since the number of authors went down from 95446 to 11333, which is only around 12% of the original authors. Nonetheless, $11,333$ is still a good amount of authors to have.

Once those $11,333$ authors were determined as the relevant ones, their author metrics were retrieved from Scopus and SciVal. This process, as previously stated takes a significant

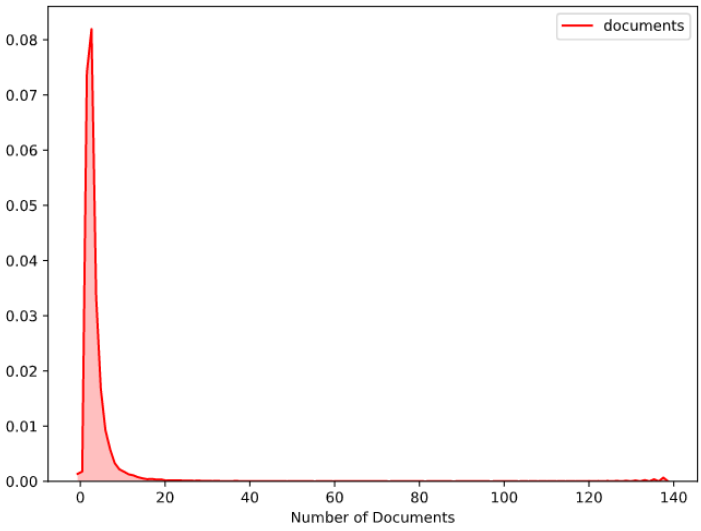


Figure 4.1: Kernel Density Estimate Plot for the number of published documents by the authors in the Clustering dataset

amount of time, and this reduction in authors took the retrieval of metrics from approximately 70 hours to merely 8 hours. As explained before, during the development of this thesis project the determination of when an author has begun to publish in Scopus indexed publications is of paramount importance. Since other Academic Rising Star identification efforts rely on an increment in the metrics of an author to determine the point where the author becomes an Academic Rising Star[3]. However, this criteria can be prone to identify as Academic Rising Star authors with long careers who suddenly make a breakthrough and their metrics increase. These authors despite becoming influential authors, may not meet the youth criteria, thus not making them attractive enough for research institutions to take into account in their hiring processes. Since it was defined in our criteria of Academic Rising Star that the author is to be in the first years of their career, it was important to visualize the publication dynamics of the authors to be used for the classification. Figure 4.2 presents the boxplot of yearly publications for authors who made their first publication in 2010. Although the median of publications stays around two publications per year, a steady increase in the number of published documents can be appreciated, especially for those authors publishing above Q3. From this visualization, we can conclude that once an author starts to publish in the Clustering area, it is most likely that the author keeps on publishing, and their number of publications per year increases. As such, for the classification task, only those authors whose first publication was done in 2010 are of interest. So, after filtering those authors, the dataset has left 451 authors, which are the ones that are going to be used for the rest of this case study.

4.2.2 Data Labeling

As previously explained, it is necessary to label the data before the classification process can be done. The authors were classified as top authors using four criteria:

- Publications in 2019

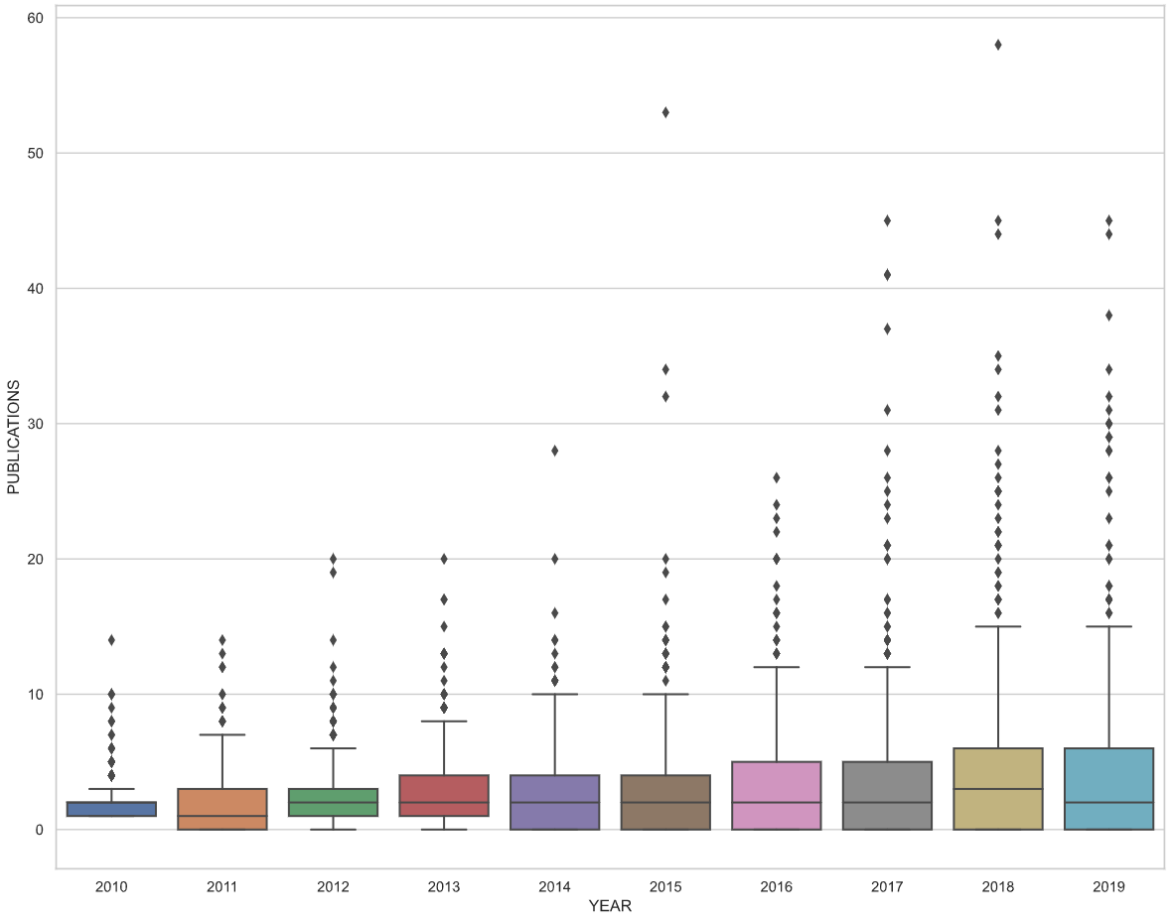


Figure 4.2: Boxplots of the number of documents published per year for authors whose first publication was made in 2010.

Metric	Group	Author Count	Proportion	Min. Value	Max. Value
Publications	Outliers	33	7.3%	15	45
Publications	Custom	61	13.5%	9.75	20.25
Citation Count	Outliers	35	7.8%	51	792
Citation Count	Custom	29	6.4%	33.15	68.85
FWCI	Outliers	24	5.3%	4.08	36.83
FWCI	Custom	33	7.1%	2.652	5.51
H-5 Index	Outliers	28	6.2%	12	29
H-5 Index	Custom	56	12.4%	7.8	16.2

Table 4.1: Labeling characteristic for each of the datasets that are used classification.

- Citation Count in 2019
- Field-Weighted Citation Impact in 2019
- H-5 Index in 2019

As previously explained, for each metric, two datasets with different labels are produced. The first dataset labels authors whose metric is greater than the outlier threshold as the positive cases. The second dataset labels authors whose metric is 35% greater or smaller than the outlier threshold. The first labeling group is referred to as *Outliers*, and the second one as the *Custom*. The reasoning as to why this is done can be found in Chapter 3. In table 4.1 the results of the labeling process are shown. In each of the datasets, the positive cases comprehend from 5.3% to 12.4%, of the total cases, which is in an acceptable range for the classification task at hand.

4.2.3 Check for Colinearity

Logistic Regression requires that no strong colinearity is present among the independent variables. To check for this assumption, a heatmap of the correlation matrix of the independent variables was made. Figure 4.3, shows the linearity among independent variables. In this case, there is some colinearity among some of the independent variables, especially among those who represent different citation metrics. However, this colinearity will be dealt with along the classification process with feature selection. It is also important to mention that Figure 4.3 only shows the correlation among the average of the metrics from 2010 to 2014. This is done to keep the heatmap at a reasonable size. Nonetheless, the complete heatmap (with the 70 independent variables) shows very similar behavior. It is also important to mention, that most of the metrics do not show high colinearity, which is good for the performance of the logistic regression.

4.3 Data normalization

Data normalization improves the performance of classifiers. For each of the labeled datasets, three additional datasets are generated. In those three additional datasets data is normalized

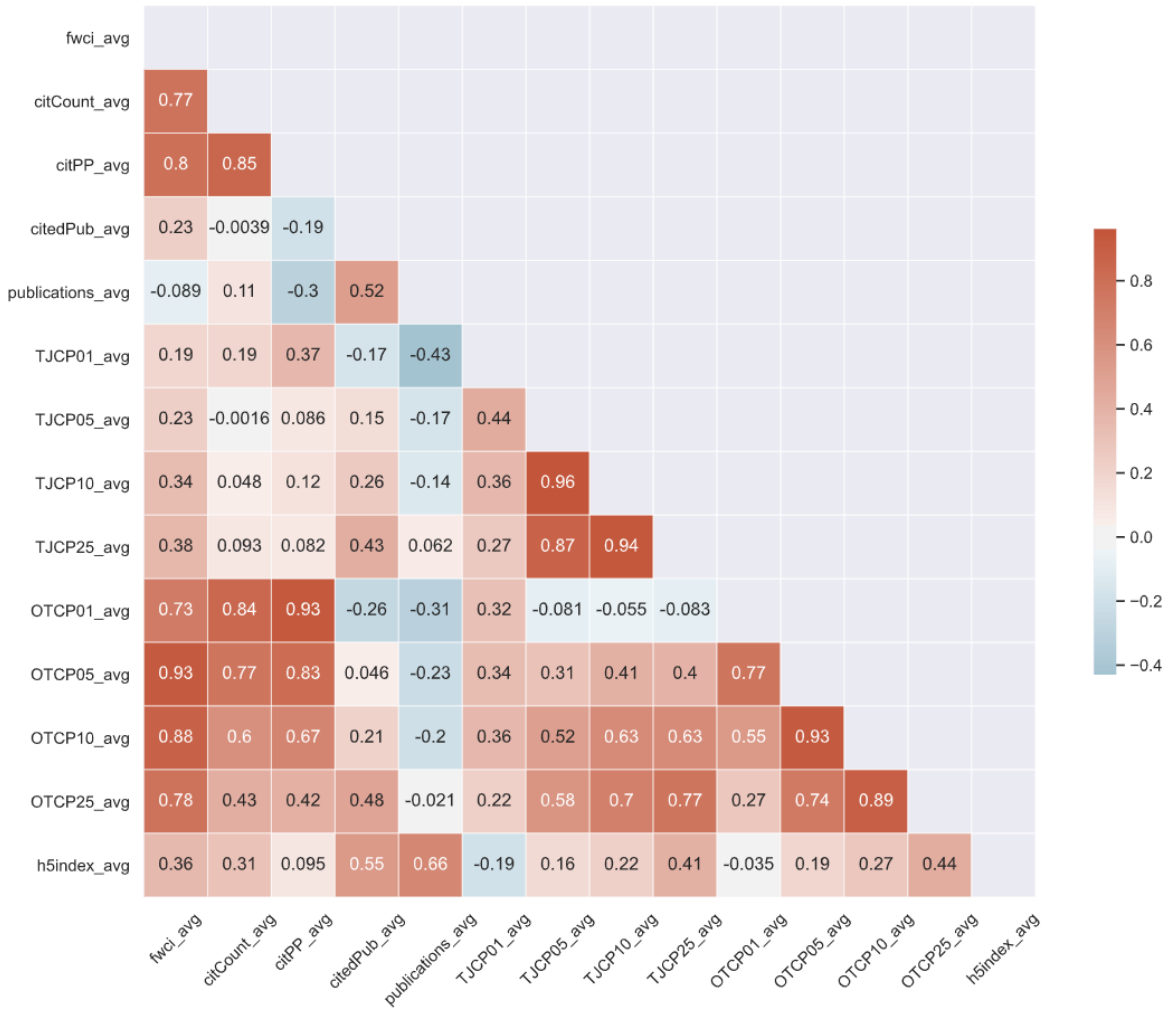


Figure 4.3: Heatmap of the correlation between the average of each independent variable from 2010 to 2014. Values closer to 1 mean a high correlation.

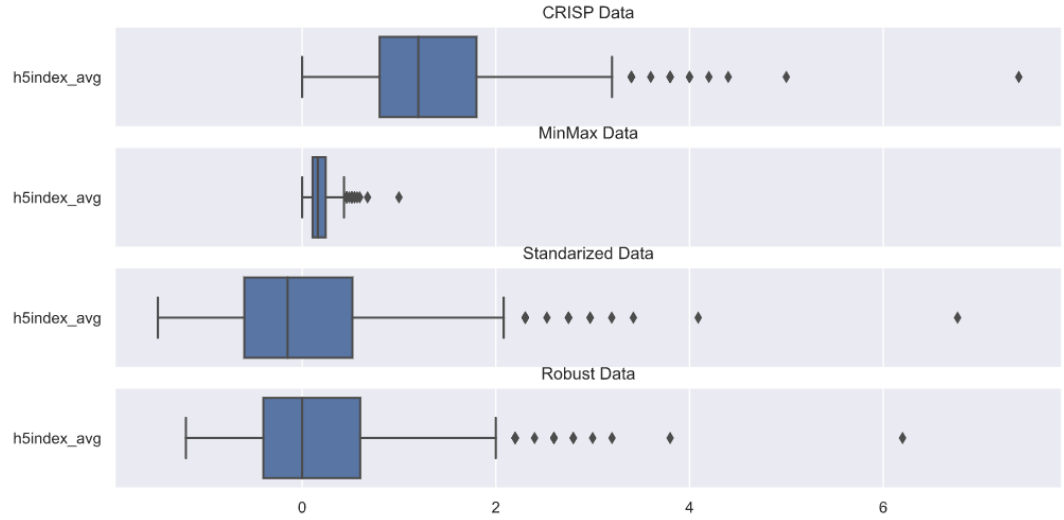


Figure 4.4: This set of boxplots show the difference in data scaling of the h5index_avg attribute when the attribute is normalized and when it is not.

using three different techniques:

- MinMax Scaling
- Standard Scaling
- Robust Scaling

In Figure 4.4 an example of the distribution of the normalized data is appreciated. In this case, Figure 4.4 is showing how the data differs between the different normalization strategies. Every attribute in the dataset is normalized in these ways.

4.4 The Dataset

For the purposes of the following classification, it is important to clarify that the classifiers were trained using different variations of this dataset. The first difference comes in the labeling of the positive cases as previously explained. The results of these different labeling strategies are shown in Table 4.1.

The next variation comes in the form of the normalization of the data. As previously explained, non-normalized metrics were used along with data that was scaled using three different methods. These were MinMax, Standard, and Robust.

And the final variation comes in the form of the set of features being used. The first variation is ALL, where every metric from 2010 to 2014 is present in the classifier. Table 4.2 shows the metrics used in this type of dataset. Each type of feature is represented from 2010 to 2014, and with the purpose of keeping the table at a reasonable size, the final digit of the feature is represented with an **X**, which represents the values from 0 to 4. This feature set has 70 individual features. It is also important to clarify that in these datasets there is a distinction between the zero and the NaN (Not a Number) values. Where applicable that a metric is zero,

Metric Abbreviation	Full Metric Name
fwci_201X	Individual Field Weighted Citation Impact from 2010 to 2014
citCount_201X	Individual Citation Count from 2010 to 2014
citPP_201X	Individual Citations per Paper from 2010 to 2014
citedPub_201X	Individual Cited Publications from 2010 to 2014
publications_201X	Individual Publications from 2010 to 2014
OTCP01_201X	Individual Outputs in the Top 1% Citation Percentiles from 2010 to 2014
OTCP05_201X	Individual Outputs in the Top 5% Citation Percentiles from 2010 to 2014
OTCP10_201X	Individual Outputs in the Top 10% Citation Percentiles from 2010 to 2014
OTCP25_201X	Individual Outputs in the Top 25% Citation Percentiles from 2010 to 2014
TJCP01_201X	Individual Publications in the Top 1% Journal Percentiles from 2010 to 2014
TJCP05_201X	Individual Publications in the Top 5% Journal Percentiles from 2010 to 2014
TJCP10_201X	Individual Publications in the Top 10% Journal Percentiles from 2010 to 2014
TJCP25_201X	Individual Publications in the Top 25% Journal Percentiles from 2010 to 2014
h5index_201X	Individual h5-index from 2010 to 2014

Table 4.2: Metrics in the datasets with the ALL feature set.

means that while there were publications by the author in that year, the value of the metric is zero. On the other hand, that the value of the metric is NaN means that there is not a value available for that metric because there were not any publications by the author in that year. For the purposes of easily dealing with this distinction, the NaN values were converted to zero.

The next variation is Average (AVG), where every metric from 2010 to 2014 is condensed in an average representation of the values of the feature from 2010 to 2014. Table 4.3 shows the metrics used in this type of dataset. This feature set has 14 individual features.

The final variation is Median (MED), where every metric from 2010 to 2014 is condensed in the median representation of the values of the feature from 2010 to 2014. Table 4.4 shows the metrics used in this type of dataset. This feature set has 14 individual features.

Nonetheless, each of these datasets has 451 observations (or authors). These correspond to the authors whose first document was published in 2010.

4.5 Classification

Once the different datasets are properly labeled and normalized, the three different classifiers were tested, Logistic Regression, and Support Vector Machine. In the following subsections, the parameters that were used for the different configurations of classifiers will be discussed. Furthermore, the results of these classification exercises through their relevant metrics will be presented.

Additionally, the classifiers will be trained with one of the three different sets of features.

- ALL: It comprehends every feature that refers to an individual year from 2010 to 2014.
- AVG: This set contains only the average of the different features from 2010 to 2014.

Metric Abbreviation	Full Metric Name
fwci_avg	Average Field Weighted Citation Impact from 2010 to 2014
citCount_avg	Average Citation Count from 2010 to 2014
citPP_avg	Average Citations per Paper from 2010 to 2014
citedPub_avg	Average Cited Publications from 2010 to 2014
publications_avg	Average Publications from 2010 to 2014
OTCP01_avg	Average Outputs in the Top 1% Citation Percentiles from 2010 to 2014
OTCP05_avg	Average Outputs in the Top 5% Citation Percentiles from 2010 to 2014
OTCP10_avg	Average Outputs in the Top 10% Citation Percentiles from 2010 to 2014
OTCP25_avg	Average Outputs in the Top 25% Citation Percentiles from 2010 to 2014
TJCP01_avg	Average Publications in the Top 1% Journal Percentiles from 2010 to 2014
TJCP05_avg	Average Publications in the Top 5% Journal Percentiles from 2010 to 2014
TJCP10_avg	Average Publications in the Top 10% Journal Percentiles from 2010 to 2014
TJCP25_avg	Average Publications in the Top 25% Journal Percentiles from 2010 to 2014
h5index_avg	Average h5-index from 2010 to 2014

Table 4.3: Metrics in the datasets with the AVG feature set.

Metric Abbreviation	Full Metric Name
fwci_med	Median Field Weighted Citation Impact from 2010 to 2014
citCount_med	Median Citation Count from 2010 to 2014
citPP_med	Median Citations per Paper from 2010 to 2014
citedPub_med	Median Cited Publications from 2010 to 2014
publications_med	Median Publications from 2010 to 2014
OTCP01_med	Median Outputs in the Top 1% Citation Percentiles from 2010 to 2014
OTCP05_med	Median Outputs in the Top 5% Citation Percentiles from 2010 to 2014
OTCP10_med	Median Outputs in the Top 10% Citation Percentiles from 2010 to 2014
OTCP25_med	Median Outputs in the Top 25% Citation Percentiles from 2010 to 2014
TJCP01_med	Median Publications in the Top 1% Journal Percentiles from 2010 to 2014
TJCP05_med	Median Publications in the Top 5% Journal Percentiles from 2010 to 2014
TJCP10_med	Median Publications in the Top 10% Journal Percentiles from 2010 to 2014
TJCP25_med	Median Publications in the Top 25% Journal Percentiles from 2010 to 2014
h5index_med	Median h5-index from 2010 to 2014

Table 4.4: Metrics in the datasets with the MED feature set.

Positive Label Weight	MR	MED	MAD	CI	γ	Magnitude
2	1.875	0.521	0.031	[0.509, 0.539]	0.000	negligible
1.5	1.995	0.528	0.041	[0.517, 0.542]	-0.194	negligible
1	2.130	0.520	0.029	[0.500, 0.527]	0.039	negligible

Table 4.5: Summary Table for the Label Weights Post Hoc Test in the Logistic Regression Classifiers.

- MED: This sets contains only the median of the different features from 2010 to 2014.

4.5.1 Logistic Regression

In the case of Logistic regression, as previously discussed, it is affected by colinearity among the independent variables. Therefore, Recursive Feature Elimination is used to reduce the number of features to be used in the classifier. If the set of features to be used is **ALL**, the parameter Minimum Features will range from 15 to 20 (from a total of 70). In any other case, this parameter will range from 5 to 10 (from a total of 14). Three Recursive Feature Elimination estimators are used with default sklearn's default parameters and 10-fold cross-validation. The selected sets of features for each Recursive Feature Elimination estimator are used to train three Logistic Regression classifiers where the weight of the positive class is 1, 1.5, and 2. All of these models are trained with 5-fold cross-validation. In total, 4320 Logistic Regression classifiers were trained with five-fold cross-validation.

In the following subsections, the results of the training of these classifiers will be presented. Boxplots showing the Area Under the ROC Curve (AUC) and the Average Precision Score for each of the classifiers' parameters will be shown too. Additionally, to determine if the difference among parameters is significant, the results of the corresponding post hoc test are presented too.

Label Weights

As mentioned earlier, while training the classifiers one of the modified parameters was the Label Weight. The positive class was assigned a weight of 1.0, 1.5, and 2.0. The boxplots of the AUC and the Average Precision Score for each of these weights can be seen in Figure 4.5. To truly determine if there is any significant difference between the use of these weights, a post hoc test is carried out. The results of this post hoc test are presented in Table 4.5, the meaning of the columns of this table are explained at the end of Chapter 2. The non-parametric Friedman test, in this case, determined that the null hypothesis (there is no significant difference in the central tendency of the groups) is rejected with $p < 0.05$, and therefore there is no significant difference between the median values of the groups. Additionally based on running a post hoc Nemenyi test, it can be assumed that all the differences between the populations are significant. This test determined that differences greater than 0.087 in the mean rank, mean that there is a significant difference between the groups. This can be visualized in Figure 4.6.

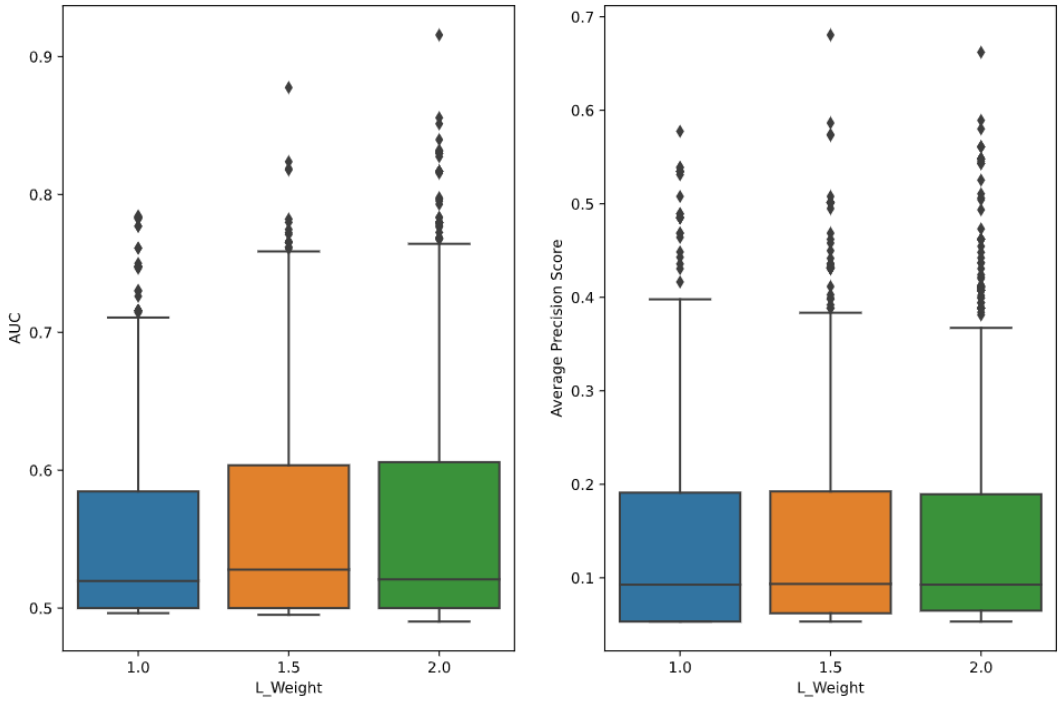


Figure 4.5: These boxplots show the AUC and Average Precision Scores for each of the positive class weights used to train the Logistic Regression Classifiers

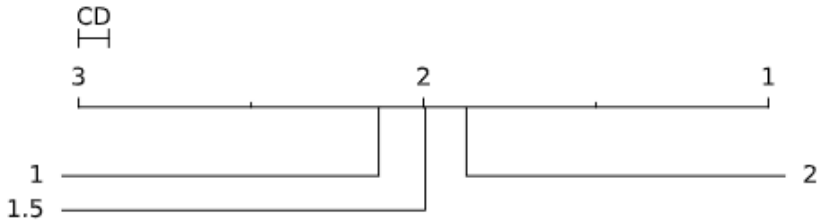


Figure 4.6: Critical Distance (CD) graph, that shows that every Positive Label Weight for the Logistic Regression classifiers is at a distance greater than the Critical Distance determined by the Nemenyi Post Hoc Test.

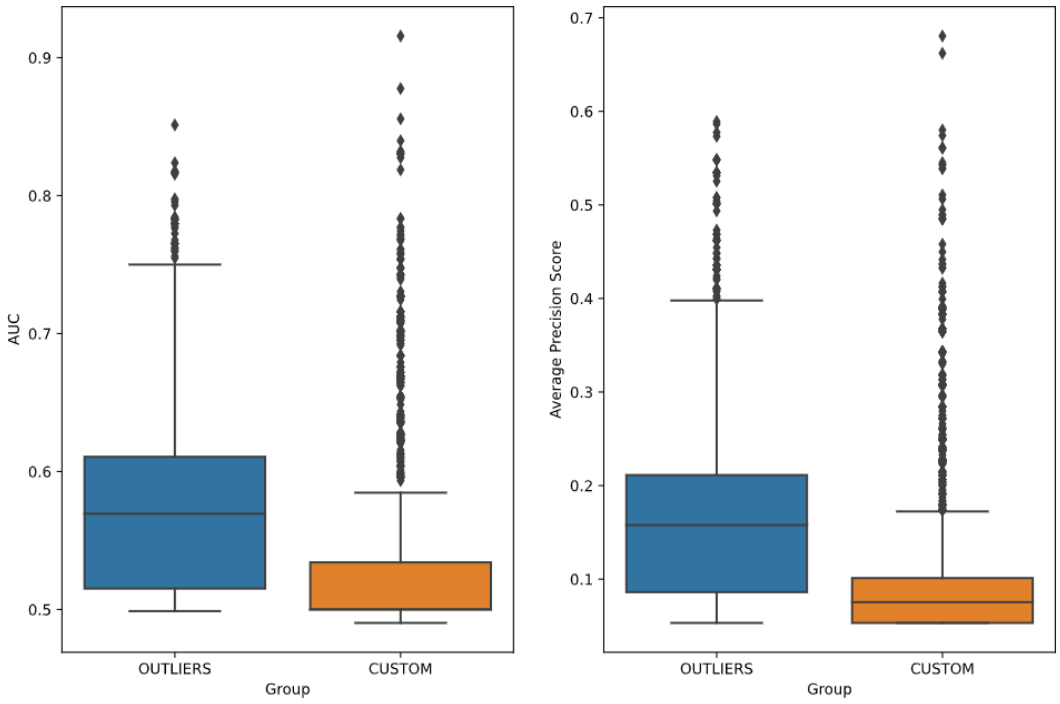


Figure 4.7: These boxplots show the AUC and Average Precision Scores for each of the positive class weights used to train the Logistic Regression Classifiers

Labeling Group

The Logistic Regression classifiers were trained with two groups of labeled datasets. One group labeled the authors in the outlier group as the positive cases, and the second group labeled the authors whose metrics were within a 70% range of the outlier threshold. These two groups are referred to as the Outliers group and the Custom group. The AUC and the Average Precision Score metrics for the classifiers trained with the data of these two groups are presented in the boxplots in Figure 4.7. It may be evident just looking at the boxplots that these two groups are significantly different. However, a post hoc test was still conducted to confirm this observation. The results of this test are shown in Table 4.6 In this case since we are dealing with just two populations, instead of the non-parametric Friedman Test, a Wilcoxon's signed-rank test was conducted to determine if there is a significant difference in the central tendency of each group. In this case, Wilcoxon's test determines with a $p < 0.05$ that we can reject the null hypothesis. Therefore, the median of one group is significantly larger than the median value of the other group. Since the comparison between only two populations only requires a Wilcoxon's test, no Nemenyi post hoc test was conducted, therefore, there is no critical distance plot to show.

Labeling Attribute

As mentioned earlier in this chapter, different datasets had different authors labeled in the positive class depending on the attribute used to classify them. In Figure 4.8 the AUC and the Average Precision Score for these labeling attributes are shown. Even though it may be

Labeling Group	MR	MED	MAD	CI	γ	Magnitude
OUTLIERS	1.218	0.569	0.072	[0.560, 0.582]	0.000	negligible
CUSTOM	1.782	0.500	0.000	[0.500, 0.500]	1.363	large

Table 4.6: Summary Table for the Labeling Groups Post Hoc Test in the Logistic Regression Classifiers.

Labeling Attribute	MR	MED	MAD	CI	γ	Magnitude
h5index_2019	1.531	0.613	0.113	[0.604, 0.621]	0.000	negligible
citCount_2019	2.343	0.540	0.060	[0.517, 0.568]	0.797	medium
publications_2019	2.961	0.500	0.002	[0.500, 0.506]	1.407	large
fwci_2019	3.166	0.500	0.000	[0.500, 0.500]	1.407	large

Table 4.7: Summary Table for the Labeling Attributes Post Hoc Test in the Logistic Regression Classifiers.

apparent that every labeling attribute is significantly different from one another, it was still convenient to run a post hoc test. The results of this test can be appreciated in Table 4.7 and in the critical distance graph 4.11. In this case, the non-parametric Friedman test is used to determine is the central tendencies of the groups are equal. In this case, that hypothesis is rejected with a $p < 0.05$. According to the Nemenyi test, the critical distance is 0.143, and therefore we can assume that all differences between populations are significant.

Normalization Method

To understand the effects of the data normalization in the dataset it is important to compare the results of the different normalization methods with the results of the non-normalized data. As it was specified previously, the data was normalized utilizing three different scalers, MinMax, Standard, and Robust. The group of boxplots in Figure 4.10 display the AUC and the Average Precision Score of the classifiers according to the data type they used. At first sight, the performance of the different normalization methods seems fairly different. A summary of the post hoc test is shown in Table 4.8. The null hypothesis of the non-parametric Friedman test is rejected with $p < 0.05$, and thus it can be concluded that there are significant differences in the central tendency of the classifier groups. The Nemenyi test determined the critical distance to be at 0.143 and therefore, all the differences between the populations are significant.

Normalization	MR	MED	MAD	CI	γ	Magnitude
STD	1.891	0.572	0.076	[0.559, 0.586]	0.000	negligible
CRISP	2.428	0.543	0.063	[0.527, 0.558]	0.421	small
Robust	2.608	0.500	0.004	[0.500, 0.521]	1.340	large
MM	3.073	0.500	0.000	[0.500, 0.500]	1.341	large

Table 4.8: Summary Table for the Normalization Methods Post Hoc Test in the Logistic Regression Classifiers.

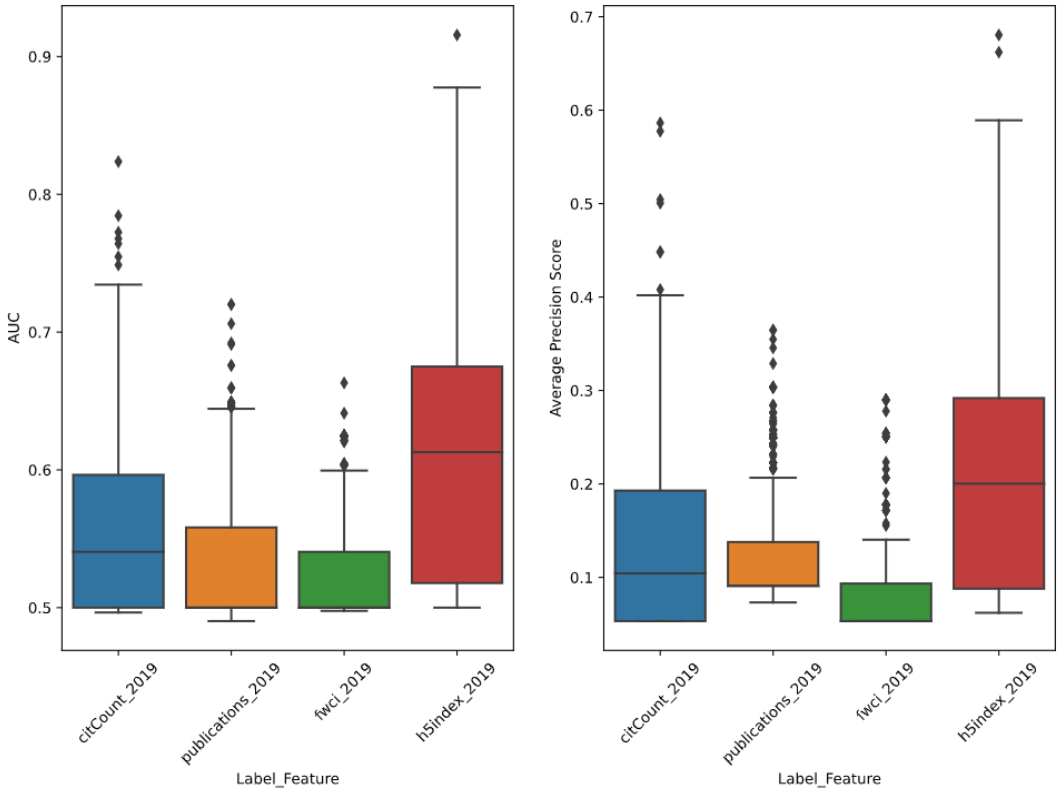


Figure 4.8: These boxplots show the AUC and Average Precision Scores for each of the positive labeling attributes used to train the Logistic Regression Classifiers. These results comprehend both the Outlier group and the Custom group.

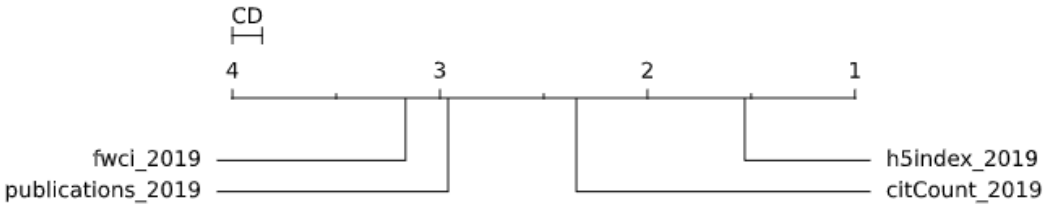


Figure 4.9: Critical Distance (CD) graph, that shows that every Labeling Attribute group for the Logistic Regression classifiers is at a distance greater than the Critical Distance determined by the Nemenyi Post Hoc Test.

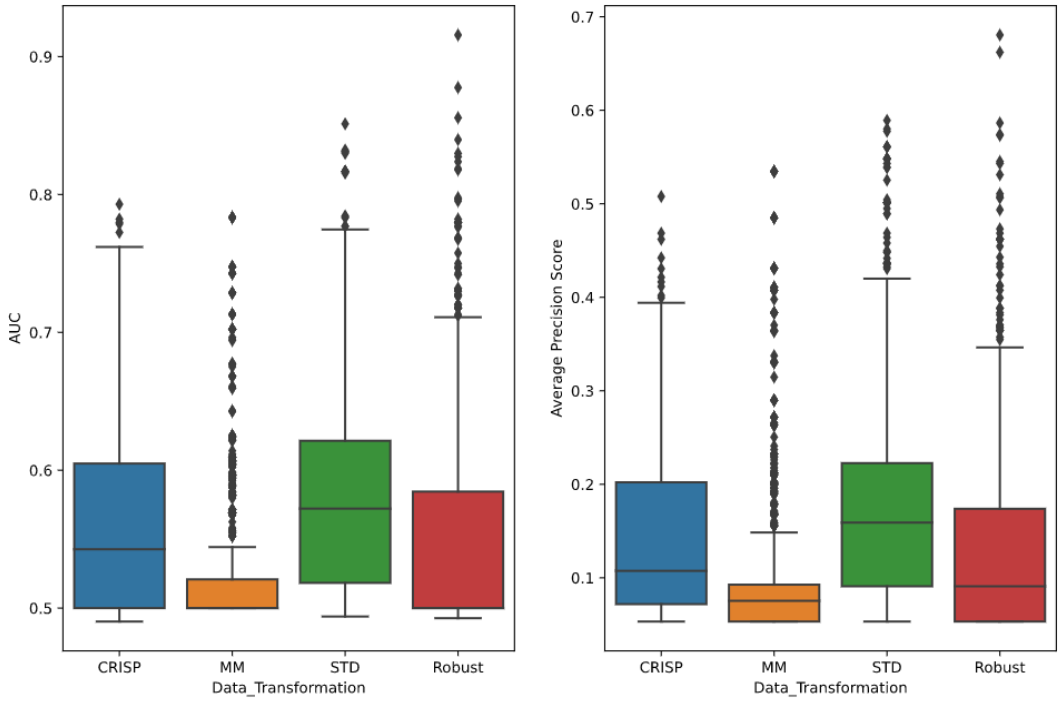


Figure 4.10: These boxplots show the AUC and Average Precision Scores for each of the normalization groups used to train the Logistic Regression Classifiers.

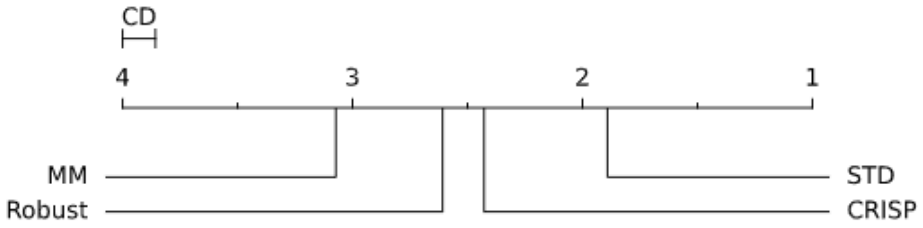


Figure 4.11: Critical Distance (CD) graph, that shows that every Labeling Attribute group for the Logistic Regression classifiers is at a distance greater than the Critical Distance determined by the Nemenyi Post Hoc Test.

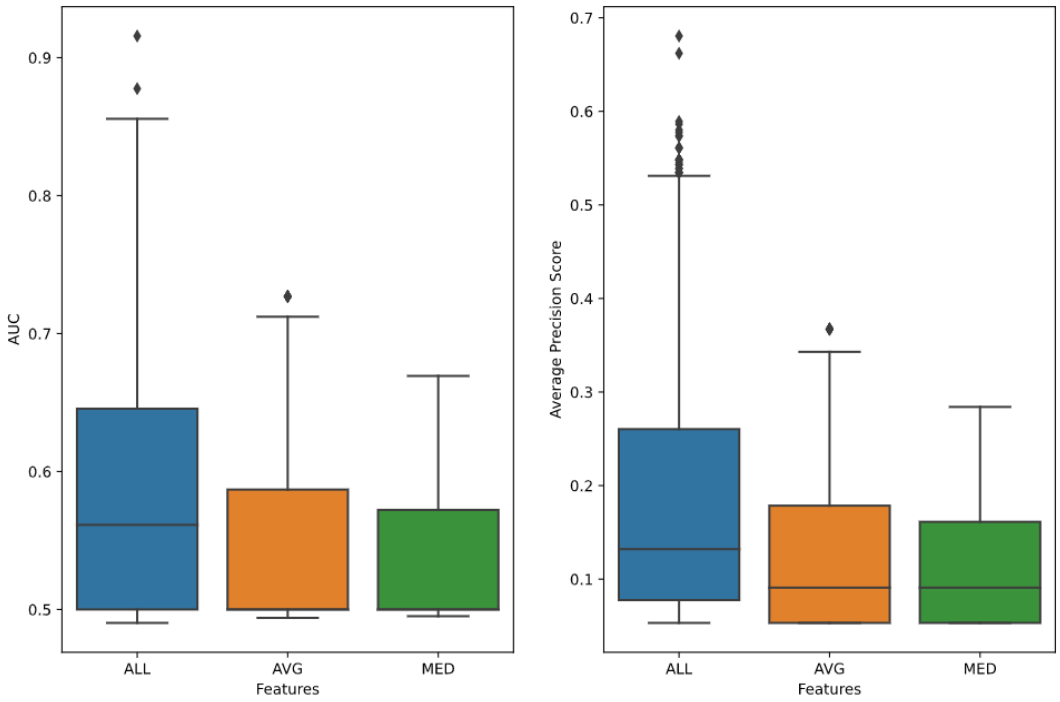


Figure 4.12: These boxplots show the AUC and Average Precision Scores for each of the feature groups used to train the Logistic Regression Classifiers.

Feature Group	MR	MED	MAD	CI	γ	Magnitude
ALL	1.491	0.561	0.091	[0.552, 0.577]	0.000	negligible
AVG	2.210	0.500	0.002	[0.500, 0.507]	0.954	large
MED	2.299	0.500	0.002	[0.500, 0.515]	0.954	large

Table 4.9: Summary Table for the Feature Groups Post Hoc Test in the Logistic Regression Classifiers.

Features Group

The classifiers were trained with three different sets of features. Every feature (except for the averages and medians of each group feature), the average of the features, and the median of the features. These features groups are referred as **ALL**, **AVG**, and **MED**. In Figure 4.12, the AUC and Average Precision Score for each group of features are shown. They seem to be fairly different, and this is confirmed by the summary of the subsequent post hoc test. The summary table of this post hoc test is shown in Table 4.9. The non-parametric Friedman test discarded the null hypothesis with $p < 0.05$, which means that the central tendencies of the different feature groups are not equal. The Nemenyi test determined that the critical distance is 0.087, then all the differences between the feature groups are significant. This can be seen graphically represented in Figure 4.13.

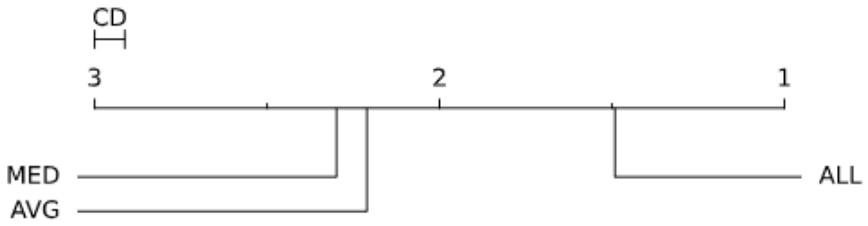


Figure 4.13: Critical Distance (CD) graph, that shows that every Feature Group for the Logistic Regression classifiers is at a distance greater than the Critical Distance determined by the Nemenyi Post Hoc Test.

RFE Estimator	MR	MED	MAD	CI	γ	Magnitude
perceptron	1.918	0.530	0.045	[0.515, 0.542]	0.000	negligible
logistic regression	2.040	0.520	0.030	[0.507, 0.536]	0.268	small
decision tree	2.041	0.521	0.031	[0.507, 0.535]	0.249	small

Table 4.10: Summary Table for the Recursive Feature Eliminator Estimator Post Hoc Test in the Logistic Regression Classifiers.

Recursive Feature Elimination Estimators

To overcome some of the drawbacks of colinearity among the independent variables, Recursive Feature Elimination was used. The three different estimators used were Logistic Regression, Perceptron, and Decision Tree. In Figure 4.14 the boxplots of the AUC and the Average Precision Score for each of the classifiers that used each type of estimator is displayed. Looking at the boxplots, it seems that the Recursive Feature Elimination groups may not be so different between them. To confirm this observation, a post hoc analysis was done, and a summary of this analysis is shown in Table 4.10. Using a non-parametric Friedman test, the null hypothesis is rejected, therefore at least one of the groups has a significantly different central tendency from the other population. The Nemenyi test determines that the critical distance, in this case, is 0.087. The differences in Mean Rank for Decision Tree and Logistic Regression are not greater than the critical distance, so we assume there is no significant difference between these two estimators. On the other hand, the perceptron estimator is indeed at a distance greater than the critical distance, and thus it is significantly different from the other two estimators. These observations can be graphically observed in Figure 4.15.

Number of Features Used

As Recursive Feature Elimination eliminates some of the features before the Logistic Regression Classifier is trained, it is also important to visualize how the elimination of features impacts the performance of the classifiers. In Figure 4.16 the AUC and the Average Precision Score of the groups of classifiers per range of the number of features is shown. To determine the range of features, the number of features was divided into five bins which are divided by quantiles. The aim of using quantiles to divide the bins was to conduct the post hoc test to

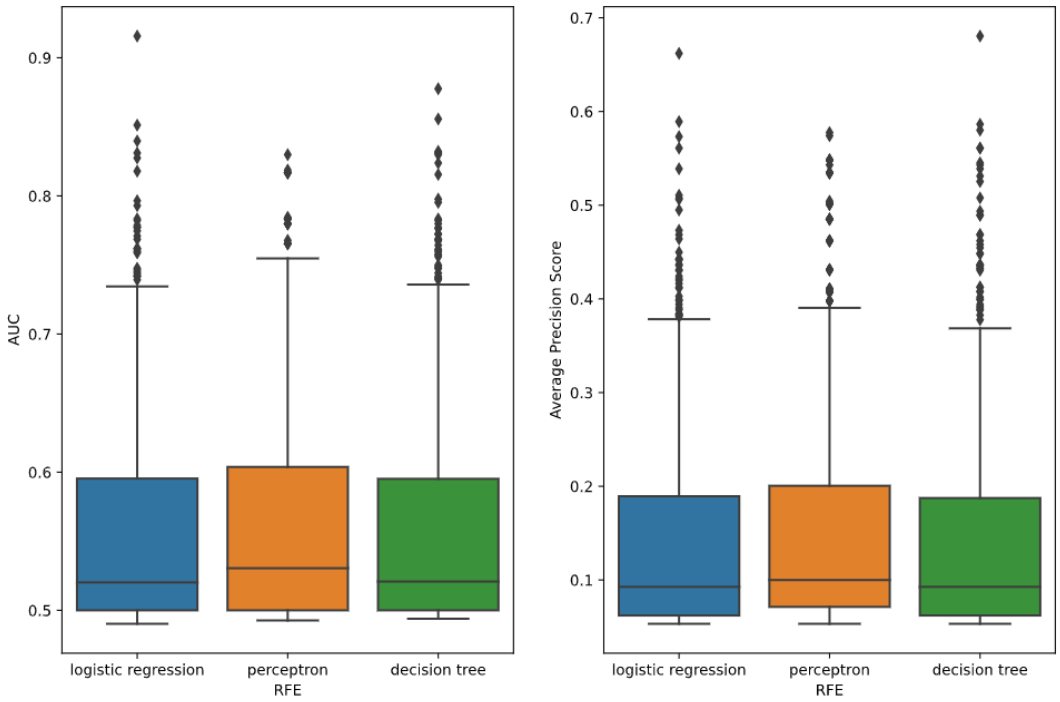


Figure 4.14: These boxplots show the AUC and Average Precision Scores for each of the estimators used in the Recursive Feature Elimination stage to train the Logistic Regression Classifiers.

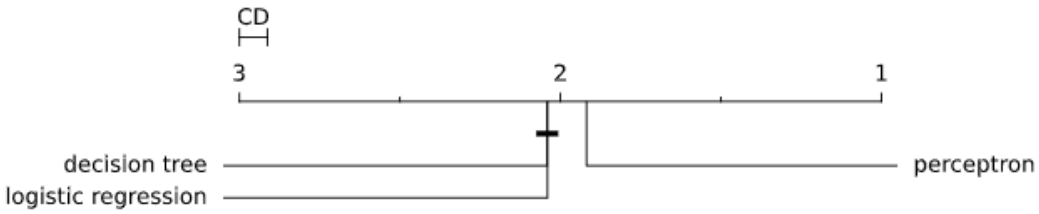


Figure 4.15: Critical Distance (CD) graph, that shows that every Recursive Feature Elimination estimator for the Logistic Regression classifiers is at a distance greater than the Critical Distance determined by the Nemenyi Post Hoc Test.

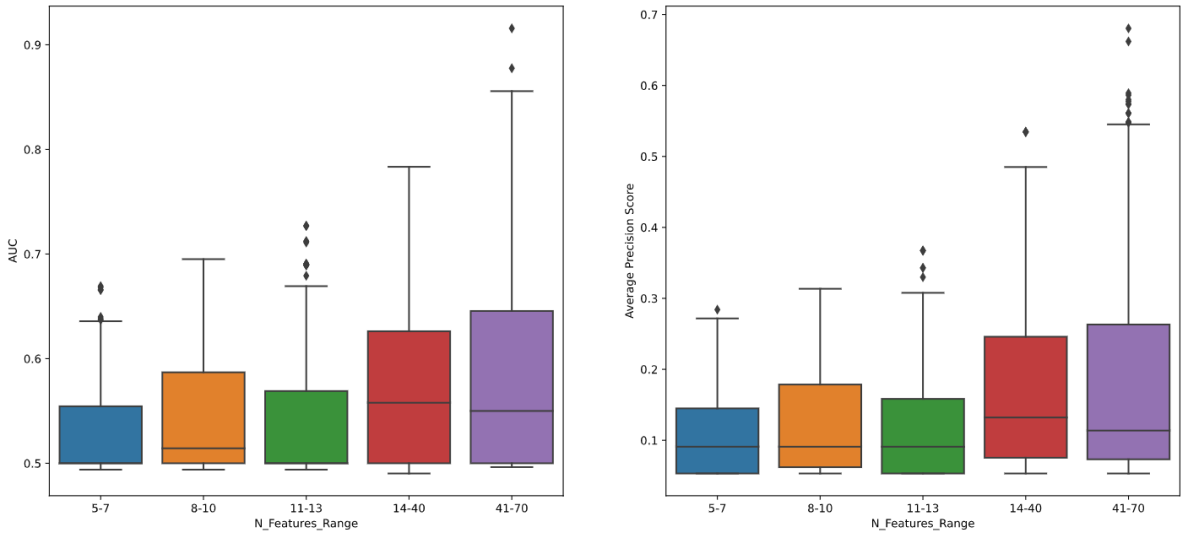


Figure 4.16: These boxplots show the AUC and Average Precision Scores for each of the ranges of number of features that resulted from using Recursive Feature Elimination before training the Logistic Regression Classifiers.

compare them as paired groups. However, since the number of resulting features depends solely on the Recursive Feature Elimination results and these have a random component, it was not possible to achieve perfectly paired groups with any number of bins. Therefore, it is not possible to conduct the post hoc test, as the groups are no longer paired. Nonetheless, the lone visualization of the metrics for each of the groups in Figure 4.16 makes it evident that the central tendency of all of the groups is different. These observations should prove useful when the impact of the number of features used in the classifiers is discussed in Chapter 5.

Exploring the best classifiers based on AUC

To understand and show what parameters worked best in training the Logistic Regression Classifiers, the classifiers whose AUC metric was in the top 10% were separated into a new dataset. These classifiers have an AUC greater than 0.64793, the this resulted in 429 classifiers, which represent 9.93% of the classifiers. To understand which features did work best, Figure 4.17 shows how prevalent were each one of the parameters in the classifiers which performed best. From looking at the Sub-figure 4.17c, it becomes evident that classifiers that used h5-index as the labeling feature performed overwhelmingly better than those labeled using any of the features. Additionally, Sub-figures 4.17e and 4.17f also show that those models which used a high amount of features, and those that used the **ALL** group of features, are the most frequent in the best performing classifiers. With regards to the data normalization methods in Sub-figure 4.17d, the only data type with less prevalence than the other normalization methods is the MinMax normalization. On the other hand, it seems that the presence of both classifiers which utilize the outliers group and the custom group shown in the Sub-figure 4.17e is more or less similar. And finally, from Sub-figure 4.17a it seems that assigning a greater weight to the label of the positive class has a positive impact on the performance of the top classifiers since an increase in the frequency of top classifiers is observed as the label weight

Feature Group	Labeling Group	Average AUC	Num. of Classifiers	Avg Num of Features
ALL	OUTLIERS	0.7257	140	42.61
ALL	CUSTOM	0.7277	123	45.93
AVG	OUTLIERS	0.6916	15	12.53
AVG	CUSTOM	0.6838	58	12.66
MED	OUTLIERS	N/A	0	N/A
MED	CUSTOM	0.6662	10	10.1

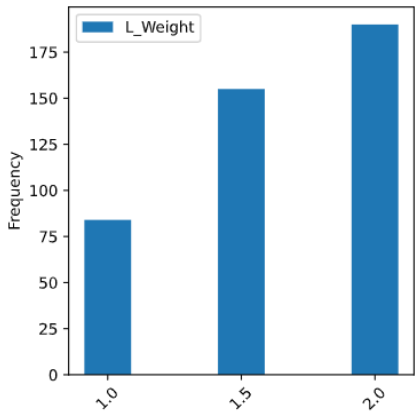
Table 4.11: Summary Table of the AUC of each Logistic Regression classifier group in the top 10% models whose labeling feature is h5-index.

grows.

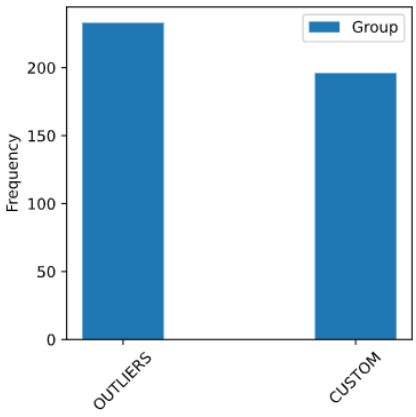
Now that it is known that the best classifiers use h5-index as the labeling target class, the analysis of this case is going to be focused on the top classifiers which use this target class. After filtering those classifiers that use other features as the labeling target class, the top classifiers dataset is left with, 346 classifiers which represent roughly 8% of all the trained classifiers. It is of interest to know in more detail which features were used in these top classifiers, that is why these classifiers are subsequently divided according to the feature set they were trained with (ALL, AVG, and MED). Additionally, these were separated again according to the feature group they were trained with (Outliers, Custom). In Table 4.11 a summary of these classifier groups are shown. In this case, none of the classifiers that used the median feature group and the outliers labeling group made it to these top classifiers. Additionally, in Table 4.11, it is shown how many classifiers are there per group. Given these results, for the following analysis, these models with the median set of features are going to be disregarded.

Best Features

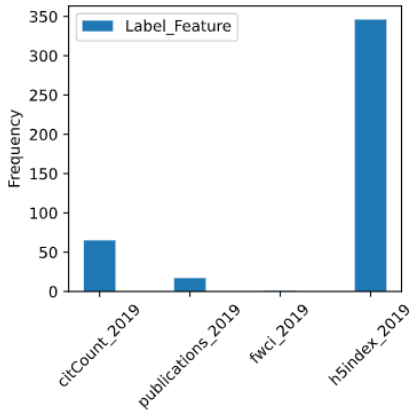
Once the best models are properly separated from the rest of the models, it is possible to visualize which features were the most predominant in these models. Tables 4.12 and 4.13 show the percentage of models in which these features appeared for the outlier and custom groups respectively when the feature set ALL was used. For the outlier group, the three most prevalent features are the fwci_2014, citCount_2014, and OTCP10_2012, which were used in every model. On the other hand, the most prevalent features for the custom group were citCount_2013 and publications_2014, which also appeared in every model. However, most of these models utilized a great number of features, which can make the explanation of the model harder to explain. On the other hand, in Tables 4.14 and 4.15 the percentages of frequency of the attributes in the models trained with the AVG feature set are shown. The first table shows the outliers group, while the second table shows the custom group. In this case, it is possible to show the frequency of every attribute in the models, since these models can only have a maximum of 14 features, while the models with the ALL feature set can have a maximum of 70 variables, and it would not be practical to show all of them. For these models trained with the AVG feature set, it is interesting that they tend to use most of the variables. According to Table 4.11, these models use on average 12.53 and 12.66 features, which is almost the 14 available features. On the other hand, the models that use the ALL feature set, use on average 42.61 and 45.93 features, which is roughly 66% of the available features.



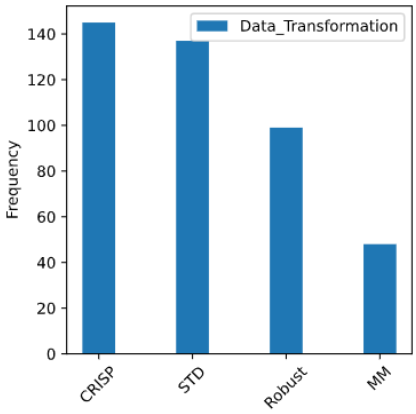
(a) Frequency of the Label Weights in the top 10% models.



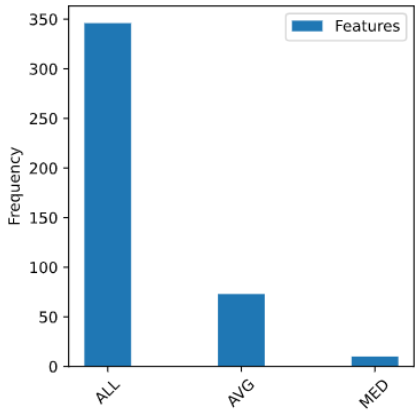
(b) Frequency of the Labeling Group in the top 10% models.



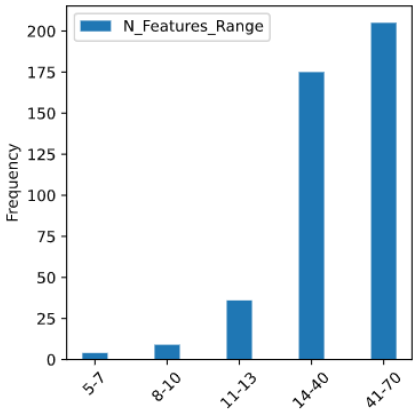
(c) Frequency of the Labeling Feature in the top 10% models.



(d) Frequency of the Data Normalization Methods in the top 10% models.



(e) Frequency of the Feature Group in the top 10% models.



(f) Frequency of the Number of Features in the top 10% models.

Figure 4.17: Frequency of the different parameters in the top 10% Logistic Regression Classifiers.

Attribute	Frequency
fwci_2014	100%
citCount_2014	100%
OTCP10_2012	100%
fwci_2012	96.43%
citCount_2012	96.43%
fwci_2013	92.86%
TJCP01_2012	92.86%
OTCP05_2013	92.86%
OTCP25_2014	92.86%
citPP_2010	92.14%
citCount_2013	91.43%
TJCP01_2013	89.29%
OTCP05_2012	88.57%
fwci_2010	85.71%
h5index_2011	85.71%

Table 4.12: Most frequent attributes in the Logistic Regression top models labeled with the h5-index attribute, for the Outliers group when the ALL feature set is used.

Attribute	Frequency
citCount_2013	100.0%
publications_2014	100.0%
citPP_2010	94.31%
fwci_2010	93.5%
citPP_2014	89.43%
citPP_2013	87.8%
citedPub_2012	86.99%
OTCP25_2014	86.99%
citCount_2012	86.18%
TJCP25_2012	86.18%
fwci_2014	81.3%
TJCP05_2012	81.3%
OTCP10_2013	81.3%
fwci_2012	79.67%
TJCP01_2013	79.67%

Table 4.13: Most frequent attributes in the Logistic Regression top models labeled with the h5-index attribute, for the Custom group when the ALL feature set is used.

Attribute	Frequency
fwci_avg	100.0%
citCount_avg	100.0%
citPP_avg	100.0%
citedPub_avg	100.0%
publications_avg	100.0%
TJCP01_avg	100.0%
OTCP01_avg	100.0%
OTCP05_avg	100.0%
OTCP25_avg	100.0%
h5index_avg	100.0%
TJCP05_avg	93.33%
TJCP10_avg	93.33%
TJCP25_avg	33.33%
OTCP10_avg	33.33%

Table 4.14: Most frequent attributes in the Logistic Regression top models labeled with the h5-index attribute, for the Outliers group when the AVG feature set is used.

Attribute	Frequency
citPP_avg	100.0%
OTCP01_avg	100.0%
OTCP25_avg	100.0%
h5index_avg	100.0%
OTCP05_avg	98.28%
citCount_avg	96.55%
TJCP10_avg	94.83%
TJCP05_avg	93.1%
fwci_avg	87.93%
publications_avg	86.21%
TJCP25_avg	86.21%
OTCP10_avg	86.21%
citedPub_avg	82.76%
TJCP01_avg	53.45%

Table 4.15: Most frequent attributes in the Logistic Regression top models labeled with the h5-index attribute, for the Custom group when the AVG feature set is used.

Group	D.T.	L. W.	Accuracy	Precision	Recall	AUC	APS	Features
CUSTOM	Robust	2	0.969	0.763	0.853	0.916	0.662	69
CUSTOM	Robust	1.500	0.973	0.867	0.765	0.878	0.680	69
CUSTOM	Robust	2	0.958	0.714	0.735	0.856	0.545	65
OUTLIERS	STD	2	0.971	0.800	0.714	0.851	0.589	53
CUSTOM	Robust	2	0.953	0.686	0.706	0.840	0.506	55
CUSTOM	STD	2	0.965	0.821	0.676	0.832	0.580	68
CUSTOM	STD	2	0.962	0.793	0.676	0.831	0.561	67
CUSTOM	STD	2	0.962	0.793	0.676	0.831	0.561	70
CUSTOM	STD	2	0.962	0.793	0.676	0.831	0.561	70
CUSTOM	Robust	2	0.960	0.767	0.676	0.830	0.543	62

Table 4.16: The top 10 Logistic Regression Classifiers, trained with the ALL feature set and h5-index as labeling feature.

The Best Models

To finish with the Logistic Regression results, all that is left to do is to present the best classifiers, both with the ALL feature set and the AVG feature set. Again, these models are the ones where the used labeling feature is h5-index and are ordered based on their AUC metric. In Table 4.16 the best 10 models for the ALL feature set are presented. In both Tables 4.16 and 4.17, D.T stands for Data Transformation, L.W. stands for Label Weight, and APS stands for Average Precision Score. It is interesting to note that 9 of these 10 classifiers were trained with the Custom group. The best two models can be considered fairly good since both boast a very high accuracy. However, the number 1 model has greater precision than the number 2 model, but the number 2 model has a greater recall than the number 1 model. It all comes down to a trade-off between false positives and false negatives. Table 4.17 shows that the achieved AUC scores with the AVG feature set are not as high as the ones achieved with the ALL feature set. Nonetheless, the precision and recall are acceptable although not as good as with the previous models. One last important thing to mention is some models are repeated in Tables 4.16 and 4.17. This is due to different Recursive Feature Elimination estimators sometimes achieve the same results, eliminating the same features. This is evidence that the Recursive Feature Elimination methods used were able to identify consistently which features were the least relevant for the Logistic Regression Classifiers.

4.5.2 SVM

Additionally to the Logistic Regression models, Support Vector Machine models were trained. In the same fashion as with the Logistic Regression Models, Recursive Feature Elimination is used to reduce the number of features to be used by the classifier. The parameters used in the Recursive Feature Elimination are the same as with the Logistic Regression. Additionally, these classifiers were trained with five-fold cross-validation as well. The kernel used is the Radial basis function kernel.

In the following subsections, the results of these classifiers are going to be presented in a

Group	D.T.	L.W.	Accuracy	Precision	Recall	AUC	APS	Features
CUSTOM	STD	2	0.945	0.696	0.471	0.727	0.367	13
CUSTOM	STD	2	0.945	0.696	0.471	0.727	0.367	13
CUSTOM	STD	2	0.945	0.696	0.471	0.727	0.367	14
CUSTOM	STD	2	0.945	0.696	0.471	0.727	0.367	14
CUSTOM	STD	2	0.945	0.696	0.471	0.727	0.367	14
CUSTOM	STD	2	0.945	0.696	0.471	0.727	0.367	14
CUSTOM	STD	2	0.945	0.696	0.471	0.727	0.367	13
CUSTOM	CRISP	2	0.942	0.682	0.441	0.712	0.343	14
CUSTOM	CRISP	2	0.942	0.682	0.441	0.712	0.343	14
CUSTOM	CRISP	2	0.942	0.682	0.441	0.712	0.343	14

Table 4.17: The top 10 Logistic Regression Classifiers, trained with the AVG feature set and h5-index as labeling feature.

Positive Label Weight	MR	MED	MAD	CI	γ	Magnitude
2	1.278	0.604	0.079	[0.600, 0.614]	0.000	negligible
1.5	1.914	0.576	0.071	[0.573, 0.583]	0.379	small
1	2.808	0.542	0.062	[0.542, 0.545]	0.884	large

Table 4.18: Summary Table for the Label Weights Post Hoc Test in the SVM Classifiers.

series of boxplots, where the performance of each type of parameter is going to be shown, just as with the Logistic Regression Classifiers. As with the Logistic Regression classifiers, the corresponding post hoc test will be conducted to determine if the different parameters have a significant difference in the performance of the SVM classifiers.

Label Weights

In Figure 4.18 the boxplots of the AUC and Average Precision Score for each weight assigned to the positive class are shown. It may be evident that a higher weight results in an increment in the metrics of the classifier, being those trained with a positive class label of 2.0, the ones with the highest median. Table 4.20 shows a summary of the values of the post hoc test. Which in this case it was determined using a non-parametric Friedman test that the difference in the median is significant for the different label weights with $p < 0.05$. The subsequent Nemenyi test established that the critical distance is 0.087, and since the differences in the mean ranks of all the label weights are greater than this critical distance, the differences between these label weights are significant. This can be appreciated in Figure 4.19.

Labeling Group

The boxplots of the AUC and Average Precision Score for each of the classification groups are shown in Figure 4.20. The medians of both populations as displayed in the boxplots are different. While the median of the Outliers group is higher than the median of the Custom

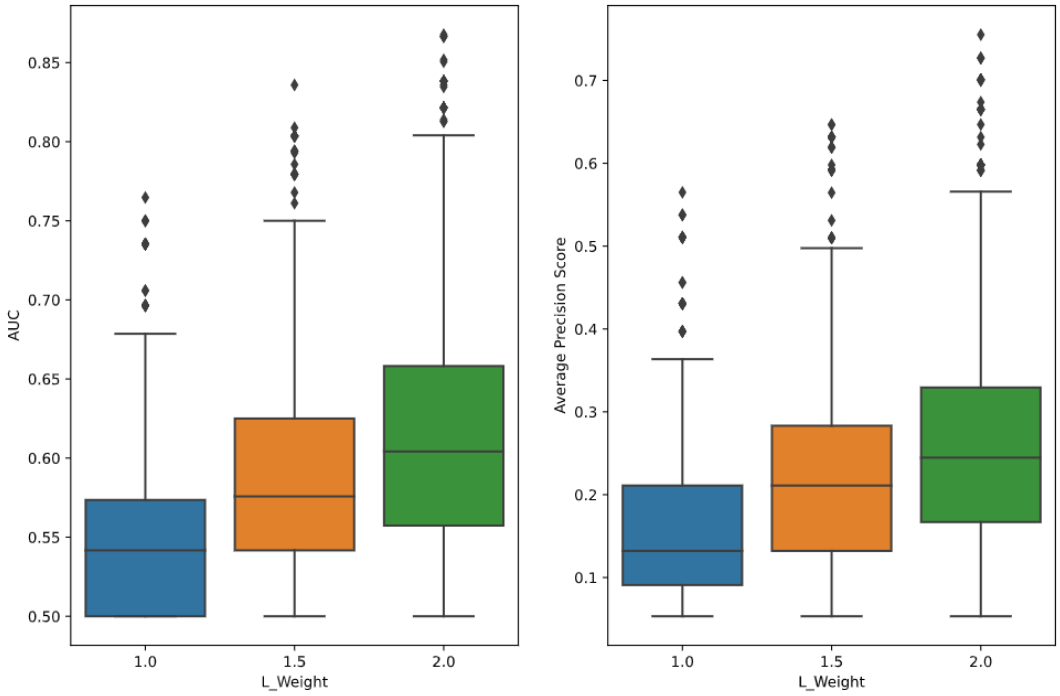


Figure 4.18: These boxplots show the AUC and Average Precision Scores for each of the positive class weights used to train the SVM Classifiers

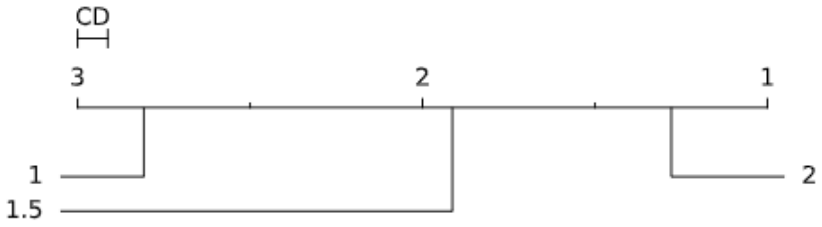


Figure 4.19: Critical Distance (CD) graph, that shows that every Positive Label Weight for the SVM classifiers is at a distance greater than the Critical Distance determined by the Nemenyi Post Hoc Test.

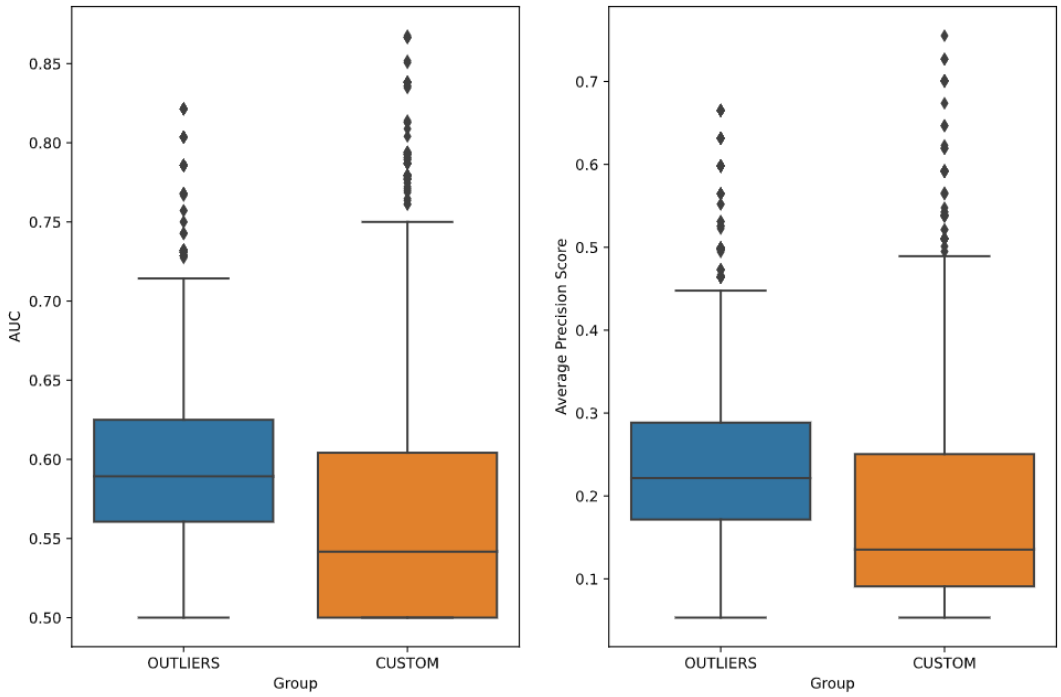


Figure 4.20: These boxplots show the AUC and Average Precision Scores for each of the positive class weights used to train the SVM Classifiers

Labeling Group	MR	MED	MAD	CI	γ	Magnitude
OUTLIERS	1.184	0.589	0.049	[0.585, 0.599]	0.000	negligible
CUSTOM	1.816	0.542	0.062	[0.537, 0.548]	0.851	large

Table 4.19: Summary Table for the Label Weights Post Hoc Test in the SVM Classifiers.

group, the Custom group seems to be able to reach better results than the Outliers group. To confirm if the two labeling groups are significantly different, a post hoc test was conducted. The summary of this test is found in Table 4.19. Since the comparison, in this case, is between two non-normal groups, then a Wilcoxon's signed-rank test is used, and the null hypothesis is rejected with $p < 0.05$, so the median of the Outliers group is significantly larger than the median value of the Custom group. For two population comparisons, there is no need for a critical distance graph.

Labeling Attribute

The boxplots of the metrics of the labeling attributes used to label the positive cases before training the classifiers are displayed in Figure 4.21. Again, the labeling attribute which performs best is the h5-index. Conducting a post hoc test, it can be confirmed that every labeling feature group is significantly different from the others. The non-parametric Friedman test determined with $p < 0.05$ that there are significant differences in the central tendencies of the different labeling attributes. The summary of the post hoc test is shown in Table 4.20, and the critical distance graph of this analysis is shown in Figure 4.22. The critical distance according

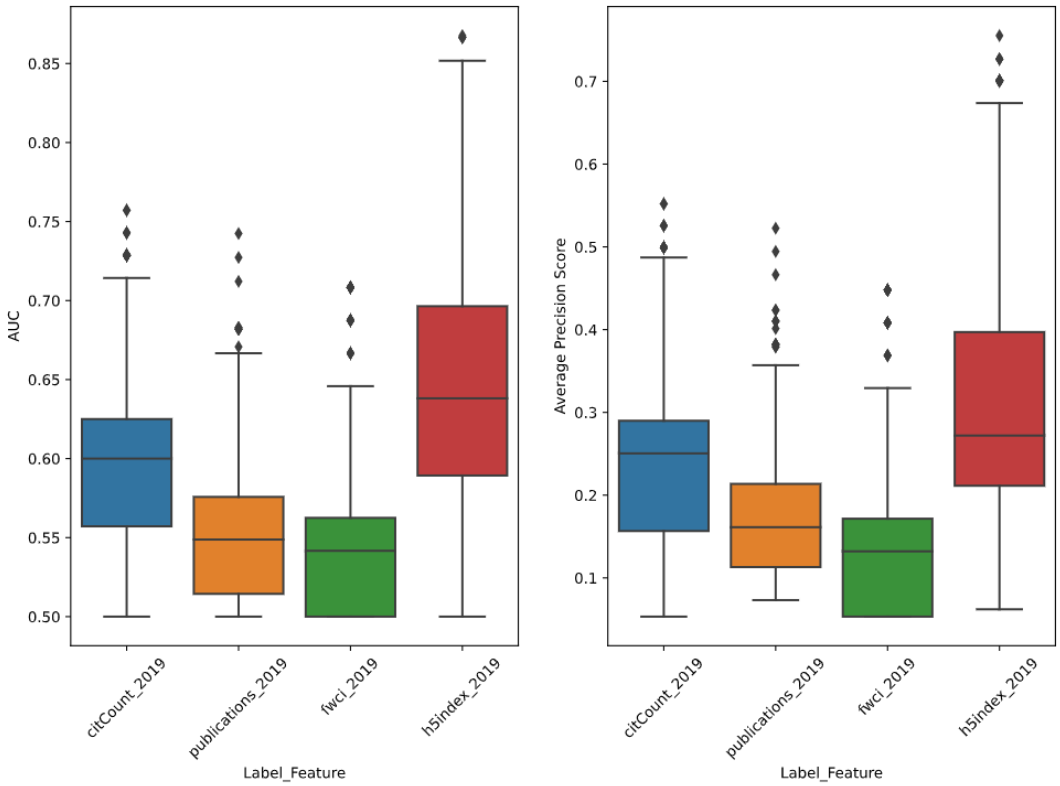


Figure 4.21: These boxplots show the AUC and Average Precision Scores for each of the positive labeling attributes used to train the SVM Classifiers. These results comprehend both the Outlier group and the Custom group.

to the Nemenyi test is 0.143 and none of the attributes is a distance smaller than the critical distance.

Normalization Method

In Figure 4.23 the effects of data normalization are compared. In these boxplots, the results of the classifiers that were trained with non-normalized (CRISP) data are found too. It seems like both CRISP data and MinMax data are not so different from each other, so we conduct a non-parametric Friedman test to confirm if there is a significant difference among the normalization methods. The test rejects the null hypothesis with $p < 0.05$ which means their difference between the central tendencies of the normalization methods is significant.

Labeling Attribute	MR	MED	MAD	CI	γ	Magnitude
h5index_2019	1.228	0.638	0.074	[0.625, 0.656]	0.000	negligible
citCount_2019	2.066	0.600	0.056	[0.598, 0.602]	0.582	medium
publications_2019	3.160	0.549	0.040	[0.545, 0.561]	1.500	large
fwci_2019	3.546	0.542	0.031	[0.542, 0.542]	1.698	large

Table 4.20: Summary Table for the Label Attributes Post Hoc Test in the SVM Classifiers.

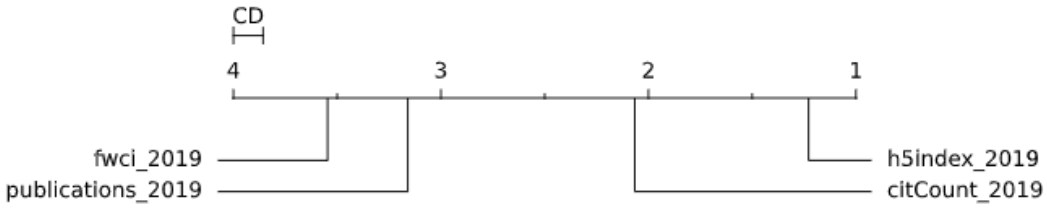


Figure 4.22: Critical Distance (CD) graph, that shows that every Positive Label Attribute for the SVM classifiers is at a distance greater than the Critical Distance determined by the Nemenyi Post Hoc Test.

Normalization Method	MR	MED	MAD	CI	γ	Magnitude
STD	1.576	0.604	0.078	[0.588, 0.606]	0.000	negligible
Robust	2.236	0.576	0.055	[0.571, 0.588]	0.419	small
MM	3.054	0.554	0.071	[0.542, 0.561]	0.675	medium
CRISP	3.134	0.561	0.059	[0.544, 0.562]	0.628	medium

Table 4.21: Summary Table for the Normalization Method Post Hoc Test in the SVM Classifiers.

However, the Nemenyi test's critical distance is calculated at 0.143, which means that any normalization methods with a difference in mean rank less than this critical distance, are not significantly different. The summary of this post hoc test is shown in Table 4.21 and the critical distance graph is shown in Figure 4.24. In the table and the graph it can be appreciated how the distance between CRISP and MM is not greater than the critical distance, therefore, the difference between these normalizations is not significant.

Features Group

Figure 4.25 presents the AUC and Average Precision Score metrics for the feature sets used by the classifiers. In terms of AUC, the three sets of features seem to be capable of achieving high scores, while AVG shows a median slightly lower than the median of the other two groups. In terms of the Average Precision Score, AVG seems to achieve lower scores, compared to the other two groups. Nonetheless, the Average Precision Scores medians for the tree feature sets are fairly low. To confirm the difference between these feature groups a post hoc test was conducted. The results of the post hoc test can be seen in Table 4.23 and in the critical distance graph in Figure 4.26. The Nemenyi test determined the critical distance at 0.087, and the non-parametric Friedman test showed that there are significant differences in the central tendencies of the feature groups.

Recursive Feature Elimination Estimators

Recursive Feature Elimination eliminates those features which do not contribute significantly to the outcome of the models. In Table 4.27, the AUC and Average Precision Score achieved by the classifiers that used each estimator in the recursive feature elimination stage is shown.

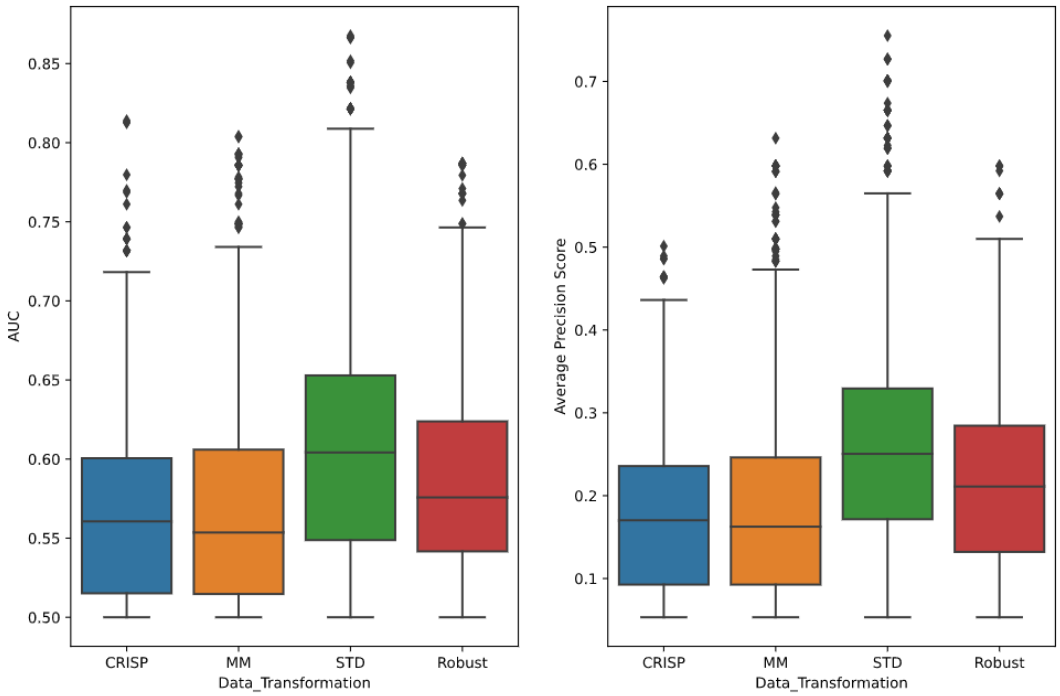


Figure 4.23: These boxplots show the AUC and Average Precision Scores for each of the normalization groups used to train the SVM Classifiers.

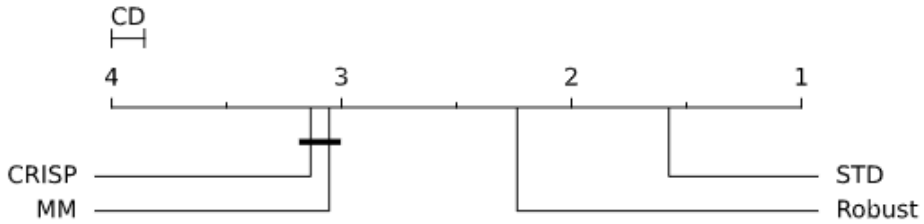


Figure 4.24: Critical Distance (CD) graph, that shows that CRISP and MM normalizations for the SVM classifiers are at a distance lesser than the Critical Distance determined by the Nemenyi Post Hoc Test.

Features Groups	MR	MED	MAD	CI	γ	Magnitude
ALL	1.441	0.588	0.081	[0.583, 0.604]	0.000	negligible
MED	1.961	0.575	0.051	[0.562, 0.583]	0.202	small
AVG	2.598	0.544	0.065	[0.542, 0.554]	0.599	medium

Table 4.22: Summary Table for the Feature Groups Post Hoc Test in the SVM Classifiers.

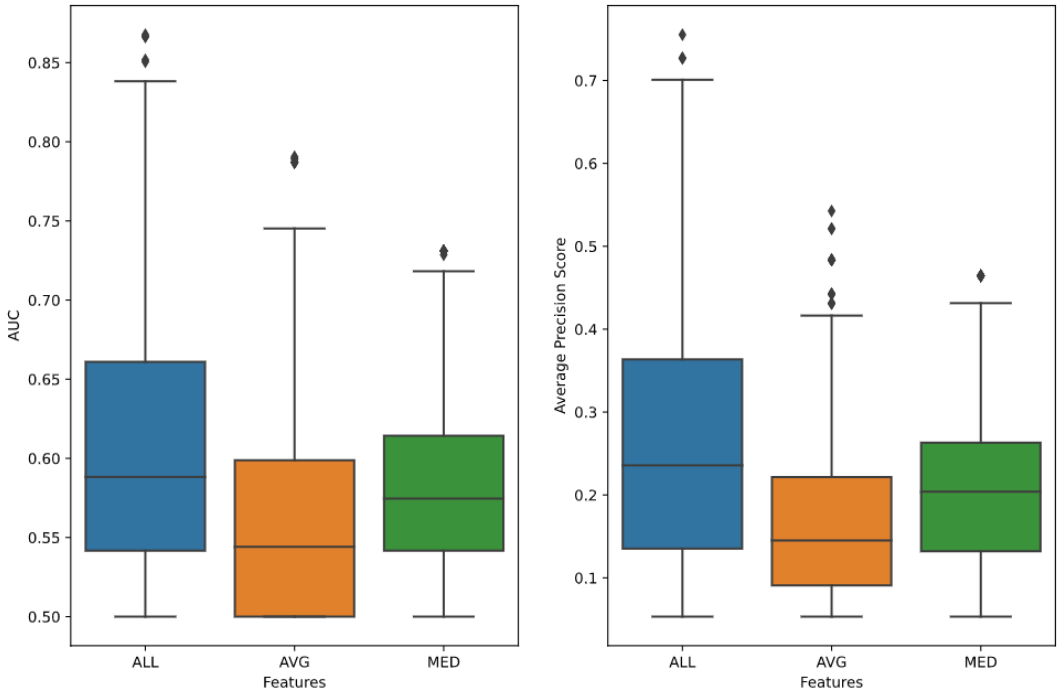


Figure 4.25: These boxplots show the AUC and Average Precision Scores for each of the feature groups used to train the SVM Classifiers.

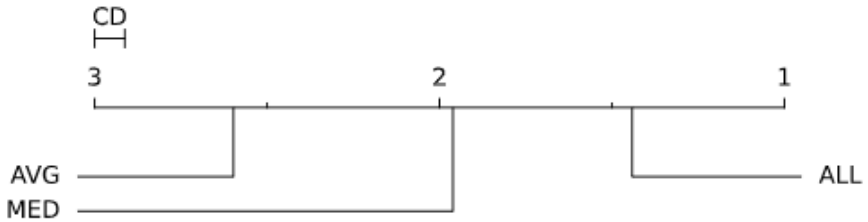


Figure 4.26: Critical Distance (CD) graph, that shows that every Feature Group for the SVM classifiers is at a distance greater than the Critical Distance determined by the Nemenyi Post Hoc Test.

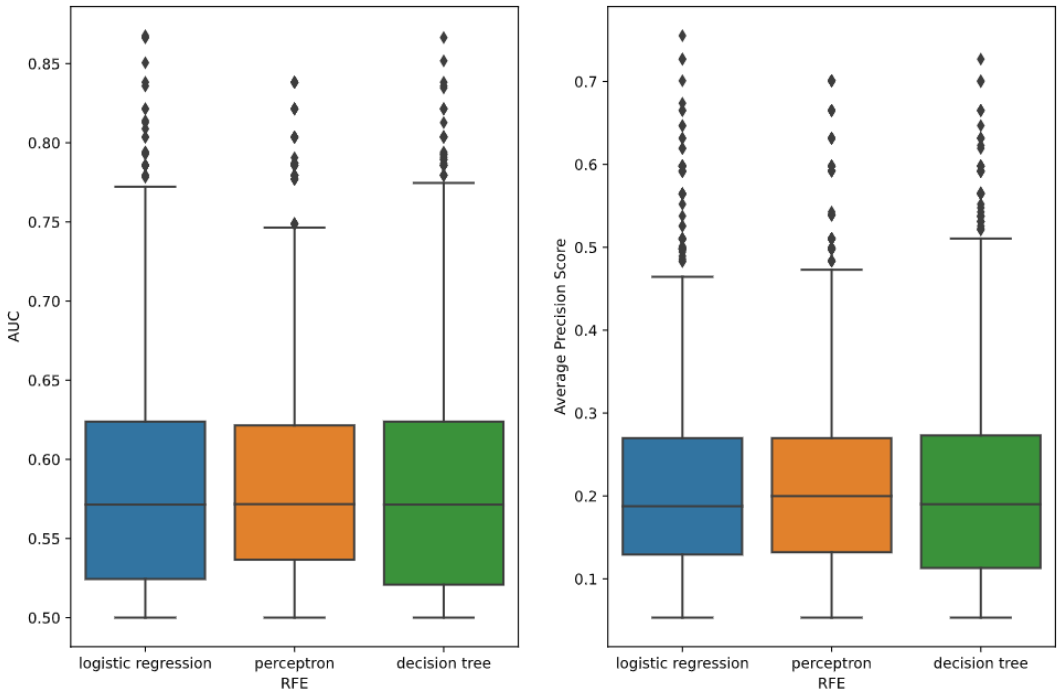


Figure 4.27: These boxplots show the AUC and Average Precision Scores for each of the estimators used in the Recursive Feature Elimination stage to train the SVM Classifiers.

RFE Estimator	MR	MED	MAD	CI	γ	Magnitude
perceptron	1.949	0.572	0.066	[0.562, 0.576]	0.000	negligible
logistic regression	2.025	0.571	0.072	[0.562, 0.576]	0.004	negligible
decision tree	2.026	0.571	0.075	[0.562, 0.576]	0.004	negligible

Table 4.23: Summary Table for the Recursive Feature Elimination Estimators Post Hoc Test in the SVM Classifiers.

It seems that the three estimators, logistic regression, perceptron, and decision tree perform similarly in terms of the AUC measurements which are not outliers. The subsequent post hoc test determined that the non-parametric Friedman rejects the null hypothesis that states that the central tendencies show no significant difference, with $p < 0.05$. Nonetheless, the Nemenyi test stated differences between estimators are significant if the difference of the mean rank is greater than the critical distance of 0.087. In this case, the difference in the mean ranks of the three estimators is not greater than the critical distance. This can be appreciated in Table **??** and in the critical distance graph in Figure 4.28.

Number of Features Used

Since the Recursive Feature Elimination stage eliminates a different number of features, it is convenient to group the number of remaining features in each classifier to visualize its AUC and Average Precision Score metrics. In this case, a quantile grouping was used to achieve groups with a similar number of classifiers. The boxplots of the metrics for these groups are

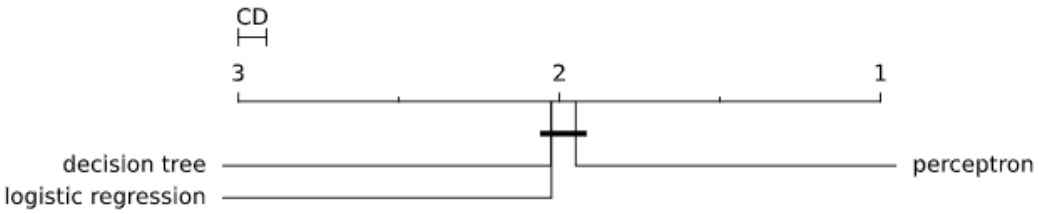


Figure 4.28: Critical Distance (CD) graph, that shows that every RFE Estimator for the SVM classifiers is at a distance lesser than the Critical Distance determined by the Nemenyi Post Hoc Test.

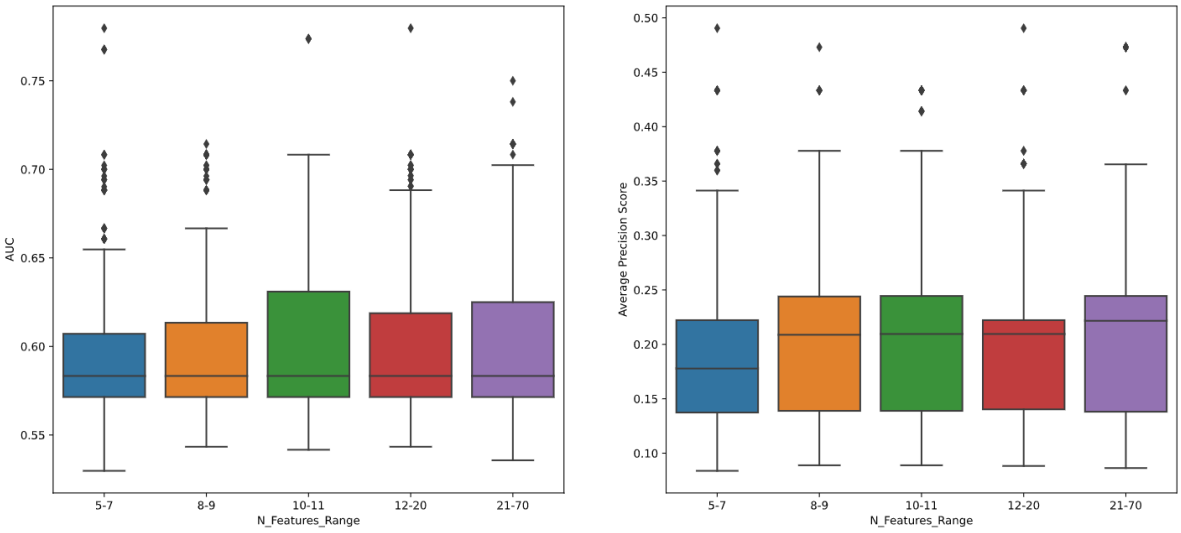


Figure 4.29: These boxplots show the AUC and Average Precision Scores for each of the ranges of number of features that resulted from using Recursive Feature Elimination before training the SVM Classifiers.

shown in Figure 4.29. In the case of the AUC, the medians of every group seem to be fairly similar, while in the case of the Average Precision Score the groups which use 8-9, 10-11 and 21-70 features have higher medians. However, conclusions about the impact of the number of features used will be better understood once the best models are presented.

Exploring the best classifiers based on AUC

Once the boxplots of the different parameters used to train the SVM Classifiers have been presented, it is important to take a look at those classifiers in the top 10% based on their AUC scores. This analysis will present a better visualization of the real effect of the parameters in the performance of the SVM Classifiers. The top 10% classifiers were those whose AUC metric is in the top 10% percentile, having an AUC score higher than 0.6786. This resulted in the selection of 421 classifiers which represent 9.75% of the total classifiers. Figure 4.30 shows the prevalence of the parameters in the top-performing classifiers. Looking at Subfigure 4.30a it can be appreciated that most of the top classifiers weighted 2.0 as the weight

Feature Group	Labeling Group	Average AUC	Num. of Classifiers	Avg Num of Features
ALL	OUTLIERS	0.7525	91	51.6
ALL	CUSTOM	0.7633	116	44.01
AVG	OUTLIERS	0.7042	7	11.71
AVG	CUSTOM	0.7212	48	11.85
MED	OUTLIERS	0.7116	40	11.83
MED	CUSTOM	0.7041	25	10.36

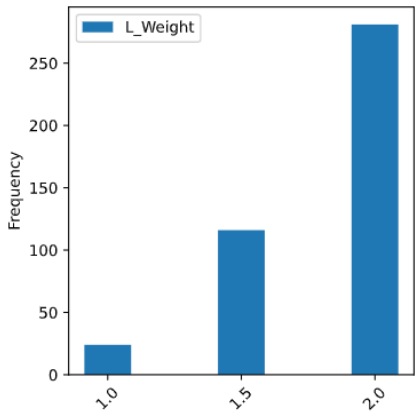
Table 4.24: Summary Table of the AUC of each SVM classifier group in the top 10% models whose labeling feature is h5-index.

of the positive label, while models with a weight of 1.0 were very scarce. Additionally, it is interesting that sub-Figure 4.30b shows that a little bit more classifiers using the outliers group are present in the top classifiers than the ones using the custom group. However, there is a significant number of classifiers trained with the custom group. As for the labeling feature, Sub-figure 4.30c, shows that classifiers that were trained with data labeled using the h5-index feature are more predominant than any other feature in the top classifiers. Surprisingly, when it comes to the data transformation used in the training of the classifiers, according to Sub-figure 4.30d, the most frequent type of data used by the top classifiers is data with Standard normalization. Sub-Figure 4.30e shows that while the most frequent set of features used by the top classifiers is ALL; AVG and MED are still used to a good extent, in around 50 classifiers each. Finally, Sub-figure 4.30f shows that the number of features used lean towards larger feature ranges. Although most of the classifiers seem to be in the 14-70 features range, these are most likely the classifiers that used the ALL set of features. A more detailed breakdown of the average of used features can be found in Table 4.24.

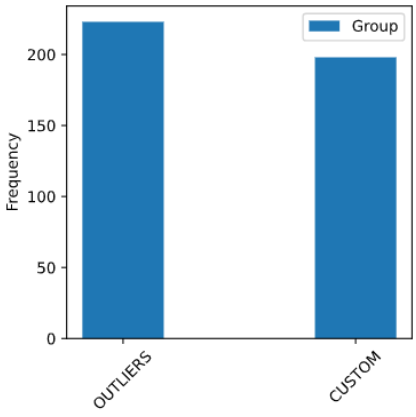
As the most frequent labeling attribute was again, h5-index, the classifier analysis is going to be focused on the classifiers that used the h5-index as the labeling attribute. Removing from the analysis those classifiers which used another labeling attribute, we are left with 327 classifiers which make up roughly 7.57% of the total classifiers. This is done again for consistency reasons, as the h5-index is the attribute that worked best for the Logistic Regression classifiers. Again, the subsequent analysis going to be carried out diving the classifiers per the feature set they used (ALL, AVG, and MED). A general summary of these classifiers is shown in Table 4.24.

Best Features

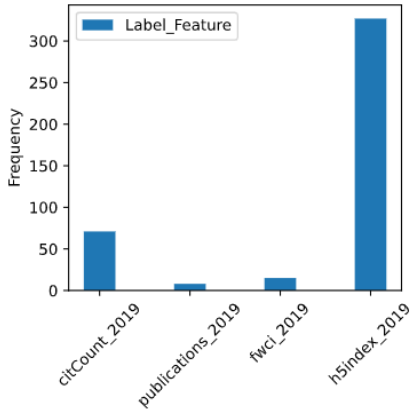
Using these top classifiers, Tables 4.25 and 4.26 show the percentage of models in which these features appeared in the classifiers trained with the feature set ALL. The first table shows the Outlier group and the second one, the Custom group. For the outlier group, the three most predominant features are related to citation, journal percentile, and FWCI. For the Custom group, the case is very similar. However, these models used a high amount of features which can make the models' explanation complicated. Additionally, these models can use up to 70 features, which would make it even harder to visualize. In these tables, only the top 14 features are shown. On the other hand, the models which use the AVG feature set can make it easier to explain which features contribute best to the classification of the authors. Tables



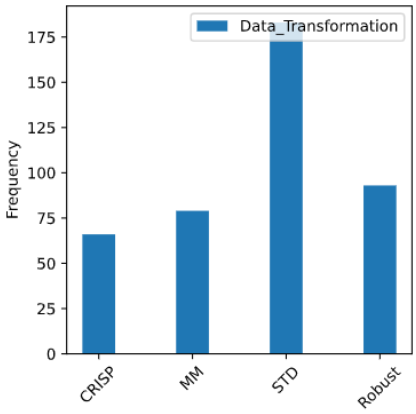
(a) Frequency of the Label Weights in the top 10% models.



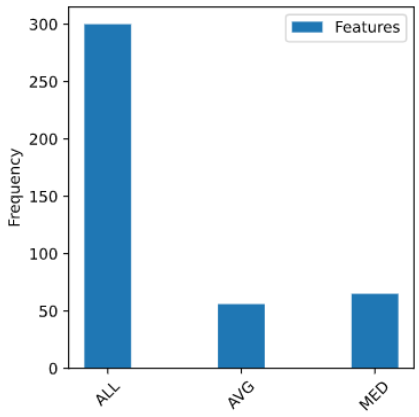
(b) Frequency of the Labeling Group in the top 10% models.



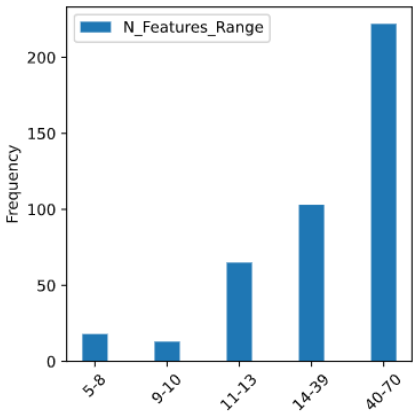
(c) Frequency of the Labeling Feature in the top 10% models.



(d) Frequency of the Data Normalization Methods in the top 10% models.



(e) Frequency of the Feature Group in the top 10% models.



(f) Frequency of the Number of Features in the top 10% models.

Figure 4.30: Frequency of the different parameters in the top 10% SVM Classifiers.

Attribute	Frequency
fwci_2012	100.0%
citCount_2012	100.0%
citCount_2014	100.0%
TJCP01_2012	100.0%
OTCP05_2013	100.0%
OTCP10_2012	100.0%
OTCP25_2014	98.9%
citCount_2013	97.8%
citPP_2010	97.8%
OTCP05_2012	97.8%
OTCP05_2011	96.7%
OTCP01_2012	95.6%
citPP_2011	94.51%
OTCP25_2011	94.51%
OTCP10_2011	92.31%

Table 4.25: Most frequent attributes in the top SVM models labeled with the h5-index attribute, for the Outliers group when the ALL feature set is used.

Attribute	Frequency
citCount_2013	100.0%
TJCP01_2013	100.0%
fwci_2010	99.14%
fwci_2014	99.14%
publications_2014	97.41%
publications_2010	92.24%
TJCP25_2013	92.24%
OTCP10_2013	92.24%
OTCP25_2012	92.24%
citCount_2011	91.38%
TJCP05_2012	91.38%
publications_2013	90.52%
citedPub_2012	87.07%
citCount_2014	83.62%
TJCP01_2012	82.76%

Table 4.26: Most frequent attributes in the SVM top models labeled with the h5-index attribute, for the Custom group when the ALL feature set is used.

4.27 and 4.28 show the frequency of features for the Outlier and Custom Groups. For the outlier group, most features are predominant, however, the least predominant are the ones related to the publications in top citation journals (TJCP). For the Custom group, these were citPP_avg, TJCP25_avg, and OTCP01_avg. Finally, Tables 4.29 and 4.30 show the frequencies of features in the classifiers trained with the MED set of features. In this case, the top features were fwci_median, citCount_median, citPP_median, and publications_median for the Outlier group, and the Custom group uses additionally 55index_median, too.

The Best Models

To show the results and metrics of the best models per feature set used, these are condensed into tables that show the parameters used and their metrics. Again, only the models whose labeling attribute is h5-index are taken into account in this analysis. Tables 4.31, 4.32 and 4.33 show these models. In the case of the ALL feature set, every top 10 model uses the Custom group with non-normalized data. However, their recall metrics are below 0.5 in most cases, while they show very high precisions. But, these models are not identifying even half of the positive cases. Again, in the models which use the AVG feature set, the top 10 models only use the custom group, however, these models are using the robust normalized data. While the AUC metric for these models is consistently higher, the recall also stays above 0.5 in the majority of models. Finally, the models which use the MED feature set do use the Outliers

Attribute	Frequency
fwci_avg	100.0%
citCount_avg	100.0%
citedPub_avg	100.0%
publications_avg	100.0%
TJCP25_avg	100.0%
OTCP01_avg	100.0%
OTCP05_avg	100.0%
OTCP10_avg	100.0%
h5index_avg	85.71%
citPP_avg	71.43%
OTCP25_avg	71.43%
TJCP10_avg	57.14%
TJCP05_avg	42.86%
TJCP01_avg	42.86%

Table 4.27: Most frequent attributes in the top SVM models labeled with the h5-index attribute, for the Outliers group when the AVG feature set is used.

Attribute	Frequency
citPP_avg	100.0%
TJCP25_avg	100.0%
OTCP01_avg	100.0%
citCount_avg	97.92%
fwci_avg	95.83%
OTCP25_avg	95.83%
h5index_avg	93.75%
OTCP10_avg	91.67%
publications_avg	89.58%
TJCP10_avg	75.0%
OTCP05_avg	70.83%
TJCP05_avg	68.75%
citedPub_avg	60.42%
TJCP01_avg	45.83%

Table 4.28: Most frequent attributes in the SVM top models labeled with the h5-index attribute, for the Custom group when the AVG feature set is used.

Attribute	Frequency
fwci_median	100.0%
citCount_median	100.0%
citPP_median	100.0%
publications_median	100.0%
OTCP10_median	95.0%
h5index_median	95.0%
OTCP01_median	90.0%
OTCP05_median	90.0%
OTCP25_median	90.0%
TJCP25_median	85.0%
TJCP10_median	80.0%
TJCP05_median	55.0%
citedPub_median	52.5%
TJCP01_median	50.0%

Table 4.29: Most frequent attributes in the top SVM models labeled with the h5-index attribute, for the Outliers group when the MED feature set is used.

Attribute	Frequency
fwci_median	100.0%
citCount_median	100.0%
citPP_median	100.0%
publications_median	100.0%
h5index_median	100.0%
citedPub_median	80.0%
TJCP25_median	80.0%
OTCP25_median	76.0%
OTCP10_median	72.0%
OTCP01_median	60.0%
OTCP05_median	60.0%
TJCP10_median	48.0%
TJCP05_median	36.0%
TJCP01_median	24.0%

Table 4.30: Most frequent attributes in the SVM top models labeled with the h5-index attribute, for the Custom group when the MED feature set is used.

Group	D.T.	L. W.	Accuracy	Precision	Recall	AUC	APS	Features
CUSTOM	STD	2	0.980	1	0.735	0.868	0.755	34
CUSTOM	STD	2	0.978	0.962	0.735	0.866	0.727	42
CUSTOM	STD	2	0.978	0.962	0.735	0.866	0.727	47
CUSTOM	STD	2	0.978	0.962	0.735	0.866	0.727	38
CUSTOM	STD	2	0.976	0.960	0.706	0.852	0.700	29
CUSTOM	STD	2	0.973	0.923	0.706	0.851	0.674	16
CUSTOM	STD	2	0.976	1	0.676	0.838	0.701	70
CUSTOM	STD	2	0.976	1	0.676	0.838	0.701	69
CUSTOM	STD	2	0.976	1	0.676	0.838	0.701	69
CUSTOM	STD	2	0.976	1	0.676	0.838	0.701	69

Table 4.31: The top 10 SVM, trained with the ALL feature set and h5-index as labeling feature.

Group	D.T.	L.W.	Accuracy	Precision	Recall	AUC	APS	Features
CUSTOM	STD	2	0.962	0.870	0.588	0.791	0.543	13
CUSTOM	STD	2	0.960	0.833	0.588	0.789	0.521	14
CUSTOM	STD	2	0.960	0.833	0.588	0.789	0.521	14
CUSTOM	Robust	2	0.956	0.769	0.588	0.787	0.484	14
CUSTOM	Robust	2	0.956	0.769	0.588	0.787	0.484	14
CUSTOM	Robust	2	0.956	0.769	0.588	0.787	0.484	14
CUSTOM	Robust	2	0.956	0.769	0.588	0.787	0.484	14
CUSTOM	Robust	2	0.956	0.769	0.588	0.787	0.484	14
CUSTOM	Robust	2	0.956	0.769	0.588	0.787	0.484	14
CUSTOM	STD	2	0.953	0.810	0.500	0.745	0.442	10

Table 4.32: The top 10 SVM Classifiers, trained with the AVG feature set and h5-index as labeling feature.

group. However, these show low recalls.

4.5.3 Classifier Comparison

While it has already been confirmed that the top SVM classifiers achieved a higher AUC and Average Precision Score metrics than the Logistic Regression classifiers. The comparison boxplots can be seen in Figure 4.31. While the AUC could achieve higher values for Logistic Regression, SVM could achieve a higher Average Precision Score. Then, to compare the best models for each classifier, the boxplots for the top 35 models trained with the AVG feature set per classifier are shown in Figure 4.32. The models trained with the AVG feature set are chosen, as these were the ones who showed consistently the best results both for Logistic Regression as SVM. The ALL feature set was not chosen as this feature set was deemed prone to overfitting.

Group	D.T.	L.W.	Accuracy	Precision	Recall	AUC	APS	Features
OUTLIERS	Robust	2	0.965	0.929	0.464	0.731	0.464	13
OUTLIERS	STD	2	0.965	0.929	0.464	0.731	0.464	13
OUTLIERS	Robust	2	0.965	0.929	0.464	0.731	0.464	13
OUTLIERS	MM	2	0.965	0.929	0.464	0.731	0.464	12
OUTLIERS	STD	2	0.965	0.929	0.464	0.731	0.464	14
OUTLIERS	Robust	2	0.965	0.929	0.464	0.731	0.464	13
OUTLIERS	STD	2	0.965	0.929	0.464	0.731	0.464	14
OUTLIERS	STD	2	0.965	0.929	0.464	0.731	0.464	14
OUTLIERS	STD	2	0.965	0.929	0.464	0.731	0.464	13
OUTLIERS	MM	2	0.965	0.929	0.464	0.731	0.464	12

Table 4.33: The top 10 SVM Classifiers, trained with the MED feature set and h5-index as labeling feature.

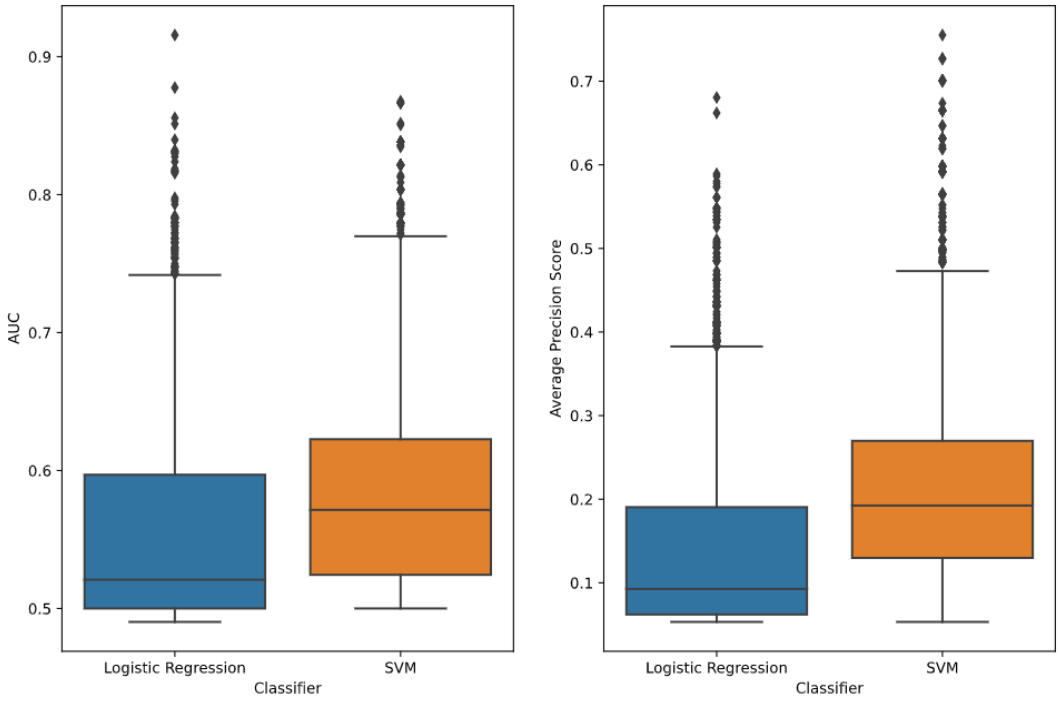


Figure 4.31: These boxplots show the AUC and Average Precision Scores for the Logistic Regression and SVM classifiers.

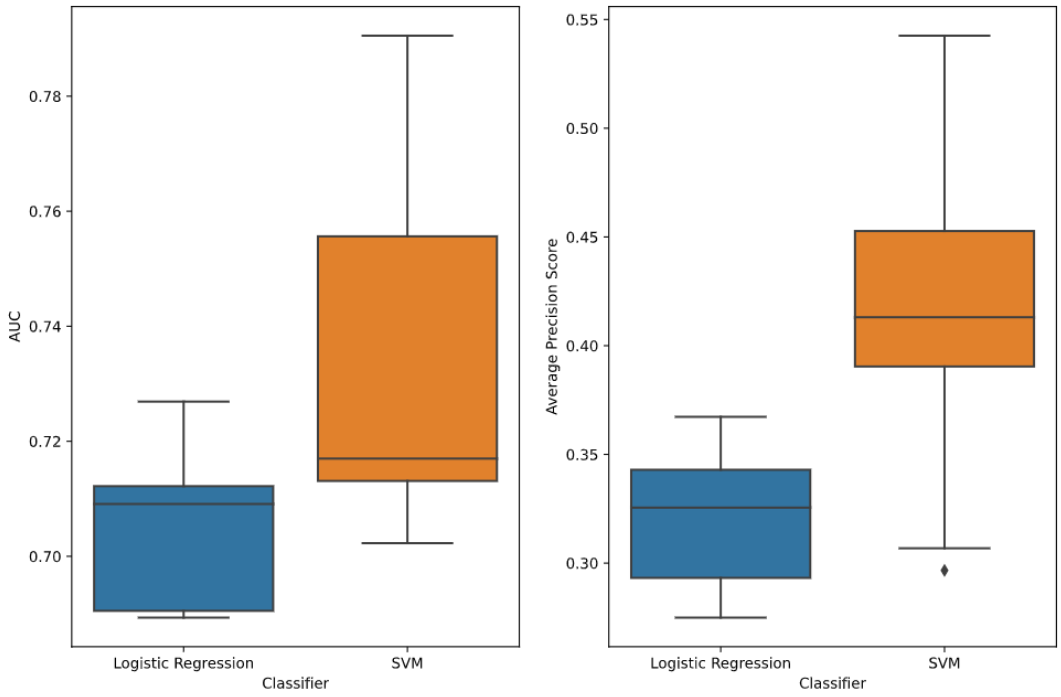


Figure 4.32: These boxplots show the AUC and Average Precision Scores for the top 35 AVG models for the Logistic Regression and SVM classifiers.

4.5.4 The Best Model

The best model with the AVG feature set was trained with the SVM classifier. This model was trained with the Custom labeling group, the data were transformed with the Standard Scaling and a Labeling Weight of 2.0. The metrics achieved by the model are shown in Table 4.34. The features used by this model are the following:

- fwci_avg
- citPP_avg
- citedPub_avg
- publications_avg
- TJCP01_avg
- TJCP05_avg
- TJCP10_avg
- TJCP25_avg
- OTCP01_avg
- OTCP05_avg

Metric	Value
Accuracy	0.9623
Precision	0.8696
Recall	0.5882
F1 Score	0.7018
AUC	0.7905
APS	0.5426
True Positive	20
False Positive	3
False Negative	14
True Negative	414

Table 4.34: Metrics for the best AVG model.

- OTCP10_avg
- OTCP25_avg
- h5index_avg

Chapter 5

Discussion

Recapitulating, the main objective of this thesis project is to prove that it is possible to predict if an author will become a top researcher in the next five years, using the metrics of the first five years of its scientific career. All of this by using exclusive data which is available through the Scopus and SciVal APIs. To prove that this is possible, a dataset was built using entirely Scopus and SciVal data, this data was used to train different classifier models, and finally, those models were evaluated, to conclude if these models are able to predict Academic Rising Stars. After the analysis of our results in Chapter 4, they indicate that it is possible to predict Academic Rising Stars, to a very useful extent.

5.1 Dataset Building

One of the first hurdles that this thesis project had to phase is the challenge of creating datasets that are comprehensive enough to undertake the Academic Rising Star identification problem. With the querying methodologies developed for this thesis project, it was possible to build a dataset of publications of the *Clustering* area in a reasonable time frame of fewer than 24 hours. In this case, the area of knowledge was well defined in the query and only took into account Journal Articles and Conference Papers. It is also important to point out that in this time frame of 24 hours, the metrics of the relevant authors were retrieved. This indicates that it is possible to build the relevant datasets in a sensible time frame.

While exploring the built dataset it was unexpected to find such a significant number of authors whose article count in the query was less than two. Trimming these authors meant being just left with 12% of authors who were initialize retrieved in the query. Although it was initially thought that this was due to different author IDs that referred to a single author, and this caused the great number of authors with less than two articles in the query. To rule out that this was the case, the same query was conducted in the web interface of Scopus, and since no repeated authors were found, it was concluded that the number of authors with less than two articles was not caused by duplicate IDs. However, due to the limitations of the data retrieval limitations, it was not possible to conduct a more detailed search as to why so many authors in the query had less than two articles. These authors were considered irrelevant for the Academic Rising Star identification task. Nonetheless, in the study case, 11,333 authors were still left, so it was still considered a good amount of data to work with.

5.2 Data Preparation

Furthermore, one of the most interesting results of this analysis is that the feature that showed the greatest potential in being predicted by past metrics was the h5-index. This does not come as a surprise as previous efforts to predict scientific success using the h-index have yielded positive results[1, 2]. Nonetheless, these efforts to predict scientific success with the h-index have shown decreasing accuracy as the time frame of the prediction grows, and that this approach does not work well for researchers with little experience[2]. However, in this thesis project, h5-index was used as a labeling feature as well as a predictive feature along with another set of features. The difference is that this approach only uses the previous 5-year h-index (h5-index) and it combines it with the features offered by SciVal. This suggests that the h5-index is a very good indicator of present scientific success. Additionally, this approach had the additional challenge of predicting the h5-index label 5 years in the future. This contrasts with the approach taken by Bin-Obaidellah[3], where the Academic Rising Star status is only predicting one year from the date of the analysis. It is convenient to explain why this one-year prediction analysis was not carried out in this thesis project. The approach taken by Bin-Obaidellah evaluates the Rising Star Status based on the increase of the metrics of an author from one year to the next one, and while this approach works well with the features in the research, our available features do not account for yearly changes. Thus, using a one-year prediction would only reduce our window to predict the Academic Rising Star status, making it more difficult to predict Academic Rising Stars as our prediction time frame is constrained. Another important difference between these two approaches is that in [3] 30% of the best ranking authors are labeled as top authors, while in our approach approximately only 6% to 12% of the authors are labeled as top authors, making a positive class even scarcer. On the other hand, the other three labeling features, FWCI, Publications, and Citation Count did not result in good classification models. This makes sense, as the h5-index takes into account both scientific production, and scientific impact. As stated by Hirsch[17], the h-index provides an estimate of the importance, significance, and broad impact of a scientist's cumulative research contributions. And using only the last five years of scientific production in the form of the h5-index results in a reliable metric to assess the current scientific impact of an author in a relevant and specific time frame.

One of the greatest concerns before training the classifiers was making sure that the data was suitable for the specific classifiers. Even though our assessment of the independent variables found that some variables had colinearity, this could be dealt with using Recursive Feature Elimination, at least that could be the case with the models based on the AVG and MED features sets, since the number of features used to train the classifiers was still reasonable. Even though we were aware that training classifiers using the ALL features sets would inevitably result in the models being trained with colinearity among the independent variables, it was still decided to train the classifiers with this feature set. The reason for this was to observe how good could the metrics of these classifiers get, despite them not being suitable for Academic Rising Star prediction. However, before going deeper into the reasoning for this, the results of the different parameters in the training of the classifiers will be discussed.

5.3 The Classification Parameters

Labeling Weights

It is also important to mention that since we are labeling less than 13% of the total researchers as top researchers (the positive label) we are bound to have a class imbalance issue that had to be dealt with. To deal with this issue, in both classifiers Logistic Regression and SVM, a higher weight was assigned to this positive class. In simple terms, when a label has a higher weight it means that it has a greater influence on the cost function of the classifiers, hence the classifier is penalized more for miss-classifying the positive label than the negative label. Our results showed that for both classifiers, Logistic Regression and SVM, the models trained with a weight greater than 1.0 were able to achieve higher AUC and Average Precision Scores. In the case of SVM, it is more notable that models trained with a positive label weight of 2.0 are more predominant than those models trained with smaller weights, especially while looking at the top 10% models. The same is true for the Logistic Regression models, however, the number of top-performing models with weights of 1.5 and 2.0 are similar. These results indicate that, assigning a greater weight to the positive labels while training the classifiers resulted in an effective way of handling the class imbalance issue. Although it was not reported in the results, in the early stages of this project, weights greater than 2.0 were tested, but these resulted in marginal gains in the true positive rate while increasing considerably the false positive rate. Therefore it was decided to leave these weights outside of the final experimentation.

Labeling Groups

As for the two labeling groups, the Outliers group, and the Custom group, these were similarly represented in the top 10% models. However, for both classifiers, it was shown that while the Custom group achieved a lower median in both AUC and Average Precision Score, these same group was capable of achieving higher metrics than the Outlier group. The reasoning behind the creation of these two labeling groups is explained in Chapter 3, where the Methodology of the project is discussed. But, in simple terms, the custom group was created to avoid classifying authors with outrageous metrics as top authors, as we suspect that it is likely that these authors did not truly start their scientific careers in 2010 as the other authors in the dataset, or maybe they are engaging in some questionable article authorship practices. But, since there is no way of determining this with the available data, it was decided to keep both of the labeling groups in the project. This also implies that some authors who may have been left out of the positive class due to the magnitude of the effect of the outliers, also have a chance of being identified by the classifiers. All in all, this scenario where both feature groups are present in the top 10% models provides evidence of the robustness of the h5-index as the labeling feature. Additionally, these results also show evidence that it is possible to adjust the positive labeling range while still obtaining good classifiers.

Data Normalization

When it comes to the normalization method, the results showed that the normalization method that achieved the best median metrics for AUC and Average Precision Score is the Standard Scaler. However, in both cases, the one that achieved the highest metrics is the Robust Scaler.

On the other hand, the normalization which performed the worst for both classifiers is MinMax Scaling. This result could suggest that the bounding range introduced by the MinMax scaler hurts the classification by constraining the outlier data to the bounding range. In this case, since there are outlier measurements that are too large compared to the rest of the measurements, this causes these other measurements to be constrained very closely together in the bounding range. In the case of the Standard and Robust Scalers, this does not happens, since these two scalers do not have bounding ranges, and effectively transform every feature to a comparable range. Moreover, another interesting observation from comparing the normalization results in both classifiers is that in SVM the Standard Scaling achieves the best results, while in Logistic Regression, Robust Scaling achieves the best results. These could be explained by the sensibility of the Logistic Regression classifier to outlier data. One of the main features of the Robust Scaler is that it tunes down the values of the outlier measurements, to decrease their influence in the model. In the case of the Logistic Regression is benefited by this Robust Scaler's feature. With these results, we can suggest that the two best normalization methods for the Academic Rising Star identification task, are Standard and Robust Scaler. Therefore, when identifying these Rising Stars it does not hurt to use both, to achieve the best possible results depending on the classifier being used.

Feature Sets

As part of the parameters involved in the training of these classifiers to identify Academic Rising Stars, three different feature groups were used. These three groups are ALL, AVG and MED. Looking at the results, they show that the classifiers in both cases the ALL feature set is the one that achieves the highest metrics. However, it is most likely due to overfitting. Since in this experiment the reduced amount of data prevents us from having a training and a testing set, the ALL feature set is more prone to overfitting than the other feature sets. Nonetheless, it was trained with the purpose of observing how high could the metrics get using the 70 metrics we had available. However, it must be noted that this feature set is not suitable for Academic Rising Star Prediction despite being the one that achieved the highest metrics. The feature sets which are more suited to predicting Academic Rising Stars are the AVG and the MED feature sets. These feature sets comprehend each metric's five-year time span into a single metric. By reducing the number of features in this way, we are providing a way to deal with overfitting and thus reducing the negative effects of training and testing with the same data. For both classifiers, the AVG feature set was the one able to achieve the highest metrics. On the other hand, the MED feature set does not achieve results as good as the AVG feature set. This result makes sense as if an author were to increase one of their metrics in their fourth and fifth year in a significant way, the median of the feature would fail to represent this increment, while the average of the feature would to an extent represent this increment.

Recursive Feature Elimination

Regarding the estimator used for the Recursive Feature Elimination, in both cases, the estimator which achieved the best results was logistic regression. However, it is also important to point out that the corresponding post hoc test to determine if the estimators were significantly different showed that for the Logistic Regression classifiers, the only different estimator was

the perceptron; while for the SVM classifiers, none of the estimators were significantly different. Hence, these results show that the chosen estimator in the training of these classifiers does not yield a significant effect. And this was later confirmed when several of the classifiers that were identified in the top 10% of AUC, shared the same parameters except for the Recursive Feature Elimination estimator. Nonetheless, we should stress the importance of having a Recursive Feature Elimination stage before training the classifiers as it is essential to overcome linearity among the independent variables, which can have a negative effect on the trained models.

Number of Features

When it comes to the number of features used by the classifiers, it makes more sense to discuss how many features used the classifiers in the top 10% of each type of classifier. In the case of the classifiers that used the ALL feature set, these used on average between 42 and 51.6 features which are more than half of the available features in the ALL feature set. But, as previously discussed, the models that use this feature set are not suitable for the Academic Rising Star prediction task. On the other hand, the models which used the AVG and MED feature sets made use of at least 10 of these features on average. However, as previously discussed, the fact that that these models utilize on average almost the 14 features available, is not likely a sign of overfitting. Mainly because these features condense the 70 features of the ALL feature set in a convenient way.

5.4 The Best Models

Looking at the best models which used the AVG feature set, the ones who achieved the best metrics are the ones trained with SVM, instead of Logistic Regression. The best Logistic Regression model trained with the AVG feature set and Custom group labeled has an AUC of 0.727, an accuracy of 0.945, a precision of 0.696, and a recall of 0.471, which means that from the 34 available top researchers to be identified by the model, it correctly classified 16 positive cases (Academic Rising Stars), failed to classify 18, and identified 7 authors as positive cases, which were not positive cases. On the other hand, the best model for the SVM classifier has an AUC of 0.791, an accuracy of 0.962, a precision of 0.870, and a recall of 0.588, which means that from the 34 available top researchers to be identified by the model, it correctly classified 20 positive cases, failed to classify 14, and identified 3 authors as positive cases which were not positive cases. What these results tell us is that it is possible to predict Academic Rising Stars using the average of their metric from their first five years since the publication of its first document. Although the metrics may not seem too high it is important to stress the difficulty of this classification task, since we are trying to predict if an author will become a top author ten years from the publication of its first article, only using the first five years of data available. For this reason, we do not expect the AVG feature set to be able to achieve metrics above 0.9. In the Academic Rising Star identification conducted by Bin-Obaidellah[3], an AUC of 0.96 using SVM was reached. However, they only identified Academic Rising Stars in the next year of the analysis, where we are identifying Academic Rising Stars over a period of five years. For this reason, we consider our metrics to be very

good and useful in the identification of Academic Rising Stars. Additionally, compared to the results obtained by Nie et. al.[26] where their best classifier achieved an F1 score of 0.819, our best classifier achieved a calculated F1 score of 0.702, which is not as bad, taking into account that the number of positive cases and data our models were trained with is far lesser than the ones used by them. Additionally, these loss in score can be interpreted as an acceptable trade off in favor of a simpler method to label the top authors.

Another interesting result is that the Custom group was more accurately identified than the Outliers group. This could be caused by the huge range of metrics that the authors in the outliers group have, while the metrics of the custom group can be expected to be in similar ranges. As it was stated when arguing the reasoning behind the Custom group, we consider this group to be more appropriate for the identification of Academic Rising Stars. At the same time, it can provide the institutions or individuals interested in identifying Academic Rising Stars with a flexible range to label the authors they are interested in predicting.

The best model achieved by this thesis project is an SVM model trained with the AVG feature set, and the Custom labeling group. For the purpose of identifying Academic Rising Stars five years from the future, identifying correctly 20 true positives, and only 3 false positives, is a remarkably good result. While identifying only 14 false negatives and 414 true negatives. Looking at this in the context of identifying Academic Rising Stars to hire them in research institutions, this would significantly decrease the number of scientists these institutions would have to check out. As for the features used by this model, it uses every available AVG feature except the citedPub features. While it could be argued that this is due to overfitting, it should be stressed that these features are a condensed representation of the original features.

5.5 Recommendations and Limitations

On the other hand, this thesis project still has some limitations. Due to time constraints and the difficulties associated with downloading and preparing the datasets, it was only possible to conduct this Academic Rising Star identification exercise for only one domain, which is Clustering. Strictly speaking, what this project proved is that it is possible to predict Academic Rising Stars in the area of Clustering. While, we would have preferred to test this methodology in more than one area of knowledge such as Point-of-Care or Additive Manufacturing, the study case conducted in this thesis project still provided evidence of the possibility of predicting Academic Rising Stars using the Scopus API and the SciVal API. This was ultimately the goal of this project.

Another important observation that should be made regarding the methodology is that it somehow hurts authors that publish in the early stages of their scientific career (undergraduates or early graduates) because they are evaluated the same as authors that have been constantly publishing since their first article. While the graph in Figure 4.2 shows that authors tend to keep on publishing more than one article yearly after the publication of its first document, it would also be important to compensate in some way for the authors that do not keep on publishing constantly from the year of their first document. While our proposed methodology does not compensate for these cases, the implementation of features that represent co-author networks or citation networks. As these could in some way ”vouch” for the potential of a

young author based on the quality of the authors it is collaborating with. Nonetheless, a way to create these features has not yet been developed within the constraints that the retrieval of data from the Scopus database has. More research needs to be conducted in these regards, however, it could be possible to calculate these features for only the authors that are going to be used for the final stage of classification, as they would hopefully cut the time and API key quota used to calculate these features.

Furthermore, due to the relatively small dataset that was utilized to train and test the predictive models, it would still be convenient to validate with a larger dataset. This was not available for this project, due to Elsevier's restrictions, as previously explained. If one more year had been available in the metrics that could have been retrieved, that would have helped to validate the models. For example, if the metrics for 2009 would have been available it would have been possible to test the models for the top authors in 2014, using metrics from 2009 to 2013. While discussing the possible ways of training these models, one idea that surfaced was to use the metrics of the first three years of metrics for the authors and then label if they were top authors in the fifth year from the publication of its first article. However, we considered that constraining the training metrics to just three years would not be as descriptive as using the metrics of five years. While the decision taken had the negative impact of not having other time frames available for testing, this has the advantage of giving the authors more time to show their potential of becoming top authors. It is also important to mention that any research institution or university that would like to implement this methodology to identify Academic Rising Stars, most likely has the means to acquire more data from Elsevier than the one this project was able to acquire. This means that these institutions would have more room to validate and tweak their models to improve their Academic Rising Star identification efforts.

Moreover, further research is needed to establish if the addition of more types of features has a positive impact on the identification of Academic Rising Stars. While this project limited itself to the metrics offered by Scopus and SciVal, it is still possible to engineer more features and assess their impact on the overall performance of the classifier models. Scientometric indicators[3] have been used to predict Academic Rising Stars, and it would be of benefit to research if these types of scientometric indicators can be calculated from the metrics available in Scopus and SciVal. Nonetheless, we would still recommend that if these features were to be calculated, the researcher would have to make sure that there is not a strong colinearity between these features and the ones we are currently using. Furthermore, research conducted by Zhang et. al[39] to predict Academic Rising Stars is focused on the academic networks. This means that the prediction of the Academic Rising Star status is predicted from the citation and publication dynamics of the collaborators of the author being evaluated. This is a sensible approach as it relies on the validation of other authors. While our approach does not take into account any feature related to the most frequent collaborators of the author being evaluated, adding features that describe in some way the robustness of quality of the closest and most frequent collaborators would probably add to the quality of the Academic Rising Stars predictions. As more metrics are offered by Scopus, it would be possible to implement them and see if they have a positive impact on the prediction of Academic Rising Stars. For example, at the moment of writing this document, now it is possible to obtain metrics regarding the amount of international collaboration of an individual author, or the type of research funds used to conduct its research. Additionally, it is possible to determine if the author is the first author in the documents taken into account for the prediction, it is possible that this

feature could also have a positive impact on the prediction performance. However, given the limitations in the data retrieval process, it would still be difficult to implement features based on citation or co-author networks.

Furthermore, another aspect that this thesis project did not tackle is explaining which metrics have the most influence in the classification of Academic Rising Stars. As such, it could be beneficial in the future to complement these classification efforts with the implementation of a pattern-miner, such as PBC4cip[24]. The implementation of pattern-mining in the dataset used in this thesis project could result in a convenient and precise way to explain which metrics and in which thresholds they have a positive impact in determining which young authors are classified as Academic Rising Stars.

Despite the positive results obtained, it could be beneficial to test this methodology with additional types of classifiers. These classifiers could be k-Nearest Neighbors or Naive Bayes, for example. As a future iteration of this work, more classifiers could be implemented, given that more computational power is available for the training of the added number of models.

All in all, we consider that these results show a positive outlook on the prediction of Academic Rising Stars using the data available from Elsevier's databases. While there is still work to be done, these results provide a solid framework from which further research can be conducted. So we can confirm based on the discussed results that it is possible to predict Academic Rising Stars using the Scopus API and the SciVal API.

5.6 Deployment

Once the model for a certain field of knowledge (in this case, Clustering) it would take little to no effort to evaluate new candidates once the model is built. However, it would still be recommended for any research institution or university that may consider implementing this methodology as part of their hiring process, to improve their model as data from upcoming years become available. For example, in our case it would be possible to improve the model by downloading the data from 2011 to 2020, now training the model using the data from 2010 to 2014, and 2011 to 2015. As more data becomes available, the performance of the prediction would possibly improve. Nonetheless, it must be stressed out that this Academic Rising Star identification methodology should only constitute an aid in the identification of potential researchers to be hired, and in no way should substitute the whole exiting hiring processes that the universities and research institutions already have in place.

Furthermore, this methodology opens up the possibility to not only identify Academic Rising Stars with the potential to become one of the top 10% researchers in their respective fields. Given that this project proposes a custom labeling group, the criteria for this custom labeling group could be adjusted at will by any interested university or research institution. It is possible that the institution would only want to identify Academic Rising Stars with the potential to become one of the top 5% researchers in their respective fields. On the other hand, it is possible that another institution could not afford researchers with such a high potential, so they could adjust the custom labeling group to identify Academic Rising Stars with the potential to become of the top 25% researchers in their respective field. The implementation of this methodology coupled with a thoughtful data exploration can enable institutions to identify researchers with the potential range they desire.

Chapter 6

Conclusion

This thesis project had the aim of confirming if it is possible to predict Academic Rising Stars using the Scopus and SciVal APIs, which was confirmed to an acceptable and satisfying extent. Retrieving the relevant documents and author metrics for our study case in the area of Clustering, then preparing the data, and finally training several classifier models showed that to a reasonable extent it is possible to predict Academic Rising Stars. Taking into account the limitations on the available data and the challenging ten-year time frame chosen to conduct these predictions, our results indicate that it is possible to predict the Academic Rising Star status in the next ten years of an author using the average metrics of the first five years of their publications.

While the approach taken to this Academic Rising Star identification task, is different from the other approaches found in the literature, it had the advantage of using only metrics and data that is already available in the Scopus and SciVal APIs. And, as discussed in Chapter 5, while the performance of the classifiers' metrics are not as high as the best in the literature, the amount of positively labeled authors and the longer time frame that this research project uses, justifies that our metrics are not as high. Therefore, the methodology developed and tested in this thesis project provides universities and research institutions a convenient way to identify young authors with high potential and then hire them to develop high-impact research. It is also very important to stress that this methodology should be considered an aid in the identification of Academic Rising Stars, and the decision of which scientist should be hired by a research institution should not be exclusively made based on the results of using this methodology. As previously discussed, the classifier is not perfect and some false positives can still be identified and the research institutions should take a closer look at the authors to confirm if they really show great potential as the classifier suggests. Nonetheless, this methodology provides a great aid to these research institutions, as instead of looking at hundreds of authors who are publishing in the fields they are interested in, they can now focus on analyzing only around 10% of these authors which are the ones the classifiers can identify.

At the same time, this methodology has a lot of room to grow, as discussed in Chapter 5. If research institutions have access to more data, the trained models will be able to achieve more precise results. At the same time, more data availability can result in experimentation with more features or indicators, which can result in even better results. However, it is also important to stress that data availability also depends on the broadness or narrowness of the selected field. If the chosen field is very broad, for example, *Computer Science*, while there

will be a lot of data to train and test the classifiers, the amount of time that retrieving the data would take may not be reasonable. However, institutions that have more availability to Elsevier's data, may not experience this drawback. On the other hand, if the chose field is too narrow or very niche, it is possible that there are not a lot of results, and the number of identified researchers may make it more convenient to look at them manually, instead of trying to train classifiers using very little data. However, a balance between these two scenarios should be balanced by the interested institutions while using this methodology to identify Academic Rising Stars. Furthermore, it is important to make clear that while the obtained models are not expected to be directly implementable for other fields of knowledge, this methodology certainly is. The citation and publication dynamics in other fields are different, and as such, the Academic Rising Star prediction models are to be trained with the data of the specific field one is interested in.

Finally, we consider that this thesis project has succeeded in providing evidence that it is possible to identify Academic Rising Stars using Elsevier's data. To the best of our knowledge, the identification of Academic Rising Stars is a topic that has already been addressed in the literature, however, no one had tried to identify these authors using the Scopus and SciVal APIs. Other Academic Rising Star identification attempts had used Web of Science[3] or ArnetMiner[39] data. We suspect this is mainly due to the challenges associated with retrieving the data from these APIs, in terms of time and download quotas. Nonetheless, Elsevier possesses one of the most comprehensive repositories of scientific publications data, thus it was important to prove that this data can be used to identify Academic Rising Stars. For this reason, we consider that this thesis project is making a significant and important contribution to the efforts to find Academic Rising Stars.

Bibliography

[1] ACUNA, D. E., ALLESINA, S., AND KONRAD, P. Scientific Success. *Nature 6* (2012), 8–9.

[2] AYAZ, S., MASOOD, N., AND ISLAM, M. A. Predicting scientific impact based on h-index. *Scientometrics 114*, 3 (2018), 993–1010.

[3] BIN-OBAIDELLAH, O., AND AL-FAGIH, A. E. Scientometric Indicators and Machine Learning-Based Models for Predicting Rising Stars in Academia. *2019 7th International Conference on Smart Computing & Communications (ICSCC)* (2019), 1–7.

[4] BRIN, S., AND PAGE, L. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems 30*, 1-7 (1998), 107–117.

[5] BROOMHEAD, D. S., AND LOWE, D. Radial basis functions, multi-variable functional interpolation and adaptive networks. Tech. rep., Royal Signals and Radar Establishment Malvern (United Kingdom), 1988.

[6] CORTES, C., AND VAPNIK, V. Support-vector networks. *Machine learning 20*, 3 (1995), 273–297.

[7] DAUD, A., ABBASI, R., AND MUHAMMAD, F. Finding rising stars in social networks. In *International conference on database systems for advanced applications* (2013), Springer, pp. 13–24.

[8] DOROGOVTSEV, S. N., AND MENDES, J. F. Ranking scientists. *Nature Physics 11*, 11 (2015), 882–883.

[9] EGGHE, L. An improvement of the h-index: The g-index. *ISSI newsletter 2*, 1 (2006), 8–9.

[10] ELSEVIER. How Scopus Works. https://www.elsevier.com/solutions/scopus/how-scopus-works/content accessed (2019, November 11).

[11] ELSEVIER. Research Metrics Guidebook, 2019. https://www.elsevier.com/__data/assets/pdf_file/0020/53327/ELSV-13013-Elsevier-Research-Metrics-Book-r12-WEB.pdf, accessed (2019, November 11).

[12] FAWCETT, T. An introduction to roc analysis. *Pattern recognition letters 27*, 8 (2006), 861–874.

[13] FRIEDMAN, M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the american statistical association 32*, 200 (1937), 675–701.

[14] GOLLAPUDI, S., AND LAXMIKANTH, V. *Practical Machine Learning*. Community Experience Distilled. Packt Publishing, 2016.

[15] GUIDE2RESEARCH. Top scientists by h-index (7 edition), May 2021.

[16] HERBOLD, S. Autorank: A python package for automated ranking of classifiers. *Journal of Open Source Software 5*, 48 (2020), 2173.

[17] HIRSCH, J. An index to quantify an individual's scientific research output. *Proceedings of the National Academy of Sciences of the United States of America 102*, 46 (2005), 16569–16572.

[18] HIRSCH, J. Does the h index have predictive power? *Proceedings of the National Academy of Sciences 104*, 49 (2007), 19193–19198.

[19] HOSMER JR, D. W., LEMESHOW, S., AND STURDIVANT, R. X. *Applied logistic regression*, vol. 398. John Wiley & Sons, 2013.

[20] HOSSIN, M., AND SULAIMAN, M. A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process 5*, 2 (2015), 1.

[21] JAMES, C., COLLEDGE, L., MEESTER, W., AZOULAY, N., AND PLUME, A. Citescore metrics: Creating journal metrics from the scopus citation index. *arXiv preprint arXiv:1812.06871* (2018).

[22] KUHN, M., AND JOHNSON, K. *Feature engineering and selection: A practical approach for predictive models*. CRC Press, 2019.

[23] LI, X.-L., FOO, C. S., TEW, K. L., AND NG, S.-K. Searching for rising stars in bibliography networks. In *International conference on database systems for advanced applications* (2009), Springer, pp. 288–292.

[24] LOYOLA-GONZÁLEZ, O., MEDINA-PÉREZ, M. A., MARTÍNEZ-TRINIDAD, J. F., CARRASCO-OCHOA, J. A., MONROY, R., AND GARCÍA-BORROTO, M. Pbc4cip: A new contrast pattern-based classifier for class imbalance problems. *Knowledge-Based Systems 115* (2017), 100–109.

[25] NEMENYI, P. Distribution-free multiple comparisons. In *Biometrics* (1962), vol. 18, International Biometric Soc 1441 I ST, NW, SUITE 700, WASHINGTON, DC 20005-2210, p. 263.

[26] NIE, Y., ZHU, Y., LIN, Q., ZHANG, S., SHI, P., AND NIU, Z. Academic rising star prediction via scholar's evaluation model and machine learning techniques. *Scientometrics 120*, 2 (2019), 461–476.

[27] NING, Z., LIU, Y., AND KONG, X. Social gene - A new method to find rising stars. *2017 International Symposium on Networks, Computers and Communications, ISNCC 2017* (2017).

[28] PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R., DUBOURG, V., VANDERPLAS, J., PASSOS, A., COURNAPEAU, D., BRUCHER, M., PERROT, M., AND DUCHESNAY, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[29] SASAKI, Y., ET AL. The truth of the f-measure. 2007, 2007.

[30] SCHREIBER, M. A modification of the h-index: The hm-index accounts for multi-authored manuscripts. *Journal of Informetrics 2*, 3 (2008), 211–216.

[31] SEKERCIOGLU, C. H. Quantifying coauthor contributions. *Science 322*, 5900 (2008), 371.

[32] SINAN, O., AND DIVYA, S. *Feature Engineering Made Easy : Identify Unique Features From Your Dataset in Order to Build Powerful Machine Learning Systems.* Packt Publishing, 2018.

[33] STOLTZFUS, J. C. Logistic regression: a brief primer. *Academic Emergency Medicine 18*, 10 (2011), 1099–1104.

[34] TABACHNICK, B. G., FIDELL, L. S., AND ULLMAN, J. B. *Using multivariate statistics*, vol. 5. Pearson Boston, MA, 2007.

[35] WEI-MENG, L. *Python Machine Learning.* Wiley, 2019.

[36] WOOLSON, R. Wilcoxon signed-rank test. *Wiley encyclopedia of clinical trials* (2007), 1–3.

[37] ZHANG, J., NING, Z., BAI, X., WANG, W., YU, S., AND XIA, F. Who are the rising stars in academia? *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries 2016-September*, Cc (2016), 211–212.

[38] ZHANG, J., XIA, F., WANG, W., BAI, X., YU, S., AND BEKELE, T. M. CocaRank : A Collaboration Caliber-based Method for Finding Academic Rising Stars. *Www* (2016), 395–400.

[39] ZHANG, J., XU, B., LIU, J., TOLBA, A., AL-MAKHADMEH, Z., AND XIA, F. PePSI: Personalized prediction of scholars' impact in heterogeneous temporal academic networks. *IEEE Access 6* (2018), 55661–55672.

[40] ZHU, L., ZHU, D., WANG, X., CUNNINGHAM, S. W., AND WANG, Z. *An integrated solution for detecting rising technology stars in co-inventor networks*, vol. 121. Springer International Publishing, 2019.

[41] ZOU, H., AND HASTIE, T. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology) 67*, 2 (2005), 301–320.

Curriculum Vitae

Jorge Antonio Ayala Urbina was born in San Luis Potosí, México, on May 22, 1994. Jorge is a candidate for the Master's Degree in Computer Science at Tecnologico de Monterrey, where he received a Bachelor of Science degree in the field of Mechatronics Engineering in 2018. While pursuing his bachelor's degree, he specialized in control and mathematical optimization. He has a particular research interest in data science and data analytics in the fields of scientometrics, sports, and public policy, and under the supervision of Dr. Ceballos Cancino he has conducted research on the application of data mining techniques to large academic and research databases.

This document was typed in using LaTeX 2_ε[1] by Jorge Antonio Ayala Urbina.

[1]The template `MCCi-DCC-Thesis.cls` used to set up this document was prepared by the Research Group with Strategic Focus in Intelligent Systems of Tecnológico de Monterrey, Monterrey Campus.