IDENTIFYING INFLUENTIAL MEMBERS OF PARLIAMENT USING TOPOLOGICAL FEATURES IN A CO-VOTATION NETWORK

Marcelo Granja Nunes

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientador: Geraldo Zimbrão da Silva

Rio de Janeiro
Dezembro de 2017

IDENTIFYING INFLUENTIAL MEMBERS OF PARLIAMENT USING TOPOLOGICAL FEATURES IN A CO-VOTATION NETWORK

Marcelo Granja Nunes

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

_____
Prof. Geraldo Zimbrão da Silva, D.Sc.

_____
Prof. Geraldo Bonorino Xexéo, D.Sc.

_____
Profa. Jonice de Oliveira Sampaio, D.Sc.

RIO DE JANEIRO, RJ - BRASIL
DEZEMBRO DE 2017

Nunes, Marcelo Granja

Identifying influential members of parliament using topological features in a co-votation network / Marcelo Granja Nunes. – Rio de Janeiro: UFRJ/COPPE, 2017.

XII, 80 p.: il.; 29,7 cm.

Orientador: Geraldo Zimbrão da Silva

Dissertação – UFRJ/ COPPE/ Programa de Engenharia de Sistemas e Computação, 2017.

Referências Bibliográficas: p. 77-81.

1. Ciência de Redes. 2. Aprendizagem de máquina. 3. Identificação de influência. I. Silva, Geraldo Zimbrão da. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

“Geeks like to think that they can ignore politics.
You can leave politics alone, but politics won't leave you alone”

- Richard Stallman

Agradecimentos

Gostaria de agradecer meus professsores, Geraldo Zimbrão, Geraldo Xexéo, Daniel Figueiredo, Alexandre Assis, Jayme Szwarcfiter, Claudia Werner e outros pelas atenção dada por eles durante meus estudos no PESC. Gostaria também de agradecer meus colegas Victor Burztyn e Rosangela Oliveira pelo parceirismo. Gostaria de agradecer Valeria Formina, Katia Kovalchuk, André Ikeda e outros amigos que me fizeram companhia nessa estadia no Rio de Janeiro. Finalmente, gostaria de agradecer meus país e minhas avós pela longa ajuda e minhas irmãs pela afeição delas.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

IDENTIFICANDO PARLAMENTARES INFLUENTES USANDO PROPRIEDADES TOPOLÓGICAS EM UMA REDE DE COVOTAÇÃO

Marcelo Granja Nunes

Dezembro/2017

Orientador: Geraldo Zimbrão da Silva

Programa: Engenharia de Sistemas e Computação

Este trabalho propõe investigar se é possível identificar parlamentares influentes usando apenas informações de resultados das votações. Dados oriundos das sessões de votação da plenária da Câmara dos Deputados do Brasil durante o ano de 2015 foram usados para criar uma rede de co-votação. Nessa rede, vértices representam parlamentares e arestas ponderadas representam a semelhança entre parlamentares quanto ao seu comportamento de voto. Uma lista de deputados influentes elaborada por um grupo de pesquisa em ciência política foi utilizada como referência.

Inicialmente, explorou-se ranquear os deputados conforme diferentes conceitos de centralidade de rede. Em seguida, essas diferentes propriedades topológicas foram utilizadas de maneira conjunta como atributos para alimentar algoritmos de classificação. Os resultados indicaram que, segundo a métrica da precisão média observada ao longo da curva de precisão-recuperação, esse método desempenhou quase 3 vezes melhor que o esperado caso se montasse uma lista de parlamentes influentes de maneira aleatória. Isso sugere que os resultados de votações parlamentares codificam informações relativas ao grau de influência dos congressistas.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

IDENTIFYING INFLUENTIAL MEMBERS OF PARLIAMENT USING TOPOLOGICAL FEATURES IN A CO-VOTATION NETWORK

Marcelo Granja Nunes

December/2017

Advisor: Geraldo Zimbrão da Silva

Department: Systems Engineering and Computer Science

This works proposes to investigate whether influent members of the parliament can be identified solely by using voting results. Data from voting sessions in the plenary of the Brazilian House of Representatives during the year of 2015 were used to create a co-voting network. In this network, vertices are congressmen and weighted edges represent pairwise similarity between congressmen regarding their voting behavior. Ground truth data about most influential congressmen were obtained from a report prepared by a political science think tank.

Initially, congressmen were ranked according to different centrality metrics. Afterwards, those topological proprieties were combined by using them as input features to feed classification algorithms. Results indicate that, as measure by the average precision over the precision-recall curve, this method performed almost 3 times better than what would be expected if influent congress were selected by random chance. This suggests that information regarding congressmen influence is encoded into voting results.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

TABLE OF SYMBOLS

AUROC Receiver operating characteristic curve

CART Classification and Regression Trees

CCDF Complementary cumulative distribution function

DIAP *Departamento Intersindical de Assessoria Parlamentar*

kNN k-nearest neighbors

RF Random forest

1 INTRODUCTION

This first chapter introduces the motivation for this dissertation and exposes the problem faced. Afterward, the objectives are explained and also provides a guide of the following chapters

1.1 Motivation

In most democratic societies, the main area of daily politics is the parliament. This political environment is ruled by a complex phenomenon in which its understanding is a theme of public interest due to its representative mission. Consequentially, in the interest of promoting transparency and accountability, as well as to stimulate citizen participation, considerable media and academic efforts were invested into clarify and communicate congregational activities to the overall public.

In Brazil, there are some organizations dedicated to the task abovementioned. Those range from public media companies such as *TV Senado* and *TV Câmara*, to private media companies such as *Globo* and *Record* that have dedicated sections - e.g. *Congresso em Foco* - both in printed media and on television channels. Outside main media, there are academic groups such as *Casa das Garças* (associated to PUC-Rio) dedicated to the theme. Finally, there are think tanks linked to political parties such as *Fundação Perseu Abramo* (associated to the Workers Party) and *Instituto Teotônio Vilela* (associated to the Social-Democratic Party). Arguably, the most established of those is Department for Parliamentary Advisory (DIAP) [*Departamento Intersindical de Assessoria Parlamentar*]. DIAP was created in 1983 with the initial purpose of providing parliament support for labor unions. Over time, the scope of its activities grew as it started to publish regular briefings about the Brazilian national political environment.

Currently, DIAP most known publication is the *Cabeças do Congresso report* (Heads of Congress, in direct translation). It is published every four years and provides an in-depth analysis of the profile of the most influential congressmen elected for the Brazilian National Congress. Its short list of most influential politicians in the Congress is widely considered a trustworthy source and often used among researchers, journalists and lobbyists.

Despite the richness of the information provided by this in-depth report, writing

them require long working hours from individuals with expert knowledge, consequentially, these reports are substantially expensive to produce. Thus, their scope is limited to high profile parliaments, leaving out parliaments of smaller states or subnational parliaments.

Those publications can provide a considerable boost to citizen awareness and participation, as well as institutional transparency and accountability. Therefore, I believe that the investigation of scalable alternatives is a valuable research agenda, helping to produce at least some insights from these reports.

1.2 Objective

The objective of this dissertation is to investigate whether it is possible to identify the most influent congressmen without the use of expert knowledge. To do so, it proposes a methodology based on empirical data from the voting session outcomes. To the best of my knowledge, this is the first study to try to identify influential members within parliaments using quantitative methods.

A congregational network will be built using a vector-based data regarding voting outcomes of each congressman on each voting session. In this network, vertices are congressmen and edges reflect pairwise congressmen distance concerning their voting behavior. An initial exploratory analysis of this co-votation network was performed to make better sense of the data. It was initially proposed multiple rankings based on different concepts of network centrality in isolation. Afterwards, a more complex model was proposed by combining multiple centralities as input features for classification algorithms. Those proposed model were then validated against a ground truth source.

I implemented the proposed methodology using data for the Brazilian House of Representatives elected in 2014 and considering the voting sessions during the year of 2015. Ground truth data about the most influential congressmen were obtained using well-established reports from a reputable think tank.

1.3 Dissertation structure

This dissertation is divided into six chapters. This first chapter provides an introduction the problem, which is identifying influential members in voting pools. The second one provides a literature review on the uses of complex networks and machine

learning in political sciences. It outlines some of the most seminal works and its findings. In addition, it is also identified the limitations in the current state of knowledge and discusses on how this work may fill some of this gap.

The third chapter intends to provide readers an understanding of concepts in complex networks and machine learning used in this work, such as algorithms for supervised classification and network metrics.

The fourth chapter details the methodology used while approaching the problem. It describes the datasets used, how data was preprocessed, which machine learning algorithms were used, how they were trained and how the model was validated.

The fifth chapter outlines the results obtained, interprets them and discusses what are the accomplishments and shortfalls of proposed methodology. Finally, the sixth chapter contains a brief conclusion and provides suggestions for future works.

2 LITERATURE REVIEW

2.1 Networks and political science

There is a long-lasting notion that power is an intrinsically relational phenomenon: it rises from the capacity that one actor has to affect other actors. Nevertheless, according to Lazer (2011), the use of Network Science by political scientists is fairly recent. On the other hand, sociology, communications, medicine, and anthropology have adopted this framework much earlier. Moreno's work (1934) pioneered the network approach and lead the expansion in the number of studies published in this field, commonly called sociometry, during the following decades.

Two groundbreaking works on network diffusion were published bringing much attention to the field. The first was Milgram (1967) who conducted a famous experiment that demonstrated that social networks had small average distance among its members, the so-called “small world” phenomenon. Later, Granovette (1973) published his work demonstrating the importance of weak social ties, i.e. ties connecting people who loosely connected, but belonged to different communities. This paper became was very influential and was the most cited network-related paper in social sciences according to Lazer (2011).

After the 1970s, the conceptual foundations of literature in the field were relatively mature, and authors started to tackle quantitative aspects of networks. Across different academic communities, two network-related ideas gained large traction during this period. First, the concept of social capital, an then, a generation of physicists who joined the social network communities contributed with two major findings: the discovery of scale-independence in networks by Barabási & Albert (1999), and the simple mechanisms in the model proposed by Watts & Strogatz (1998) that lead networks with the small world propriety.

For this dissertation, those works are also highly important since they deal with how diffusion mechanisms operate in fields as different as innovation diffusion or epidemiology. In particular, those works highlighted how weak links could led to the small-world propriety, they demonstrated that most connections that an individual have are within a small, highly connected cluster of peers. However, a small proportion of them connect to distant individuals, quickly accelerating diffusion across the network. Such “long-haul” links have a disproportional importance within the network. Those differences of importance among nodes and link led research on social influence can

be strongly related to the concept of network centrality.

A prime example of this was written by Padgett & Ansell (1993) for a network of marriage and trade among traditional Florencian families during the Renascence. This work makes a compelling argument that the most influential family, the Medici, ascended to power after a successful campaign building connection with other families. Those connections were not more numerous than the ones build by other families, but were carefully crafted to connect families that otherwise would be disconnected. Thus, the Medici family built themselves as the main intermediary for transactions across those groups.

Similarly, to the case of the Florencian networks, the object of this work is to verify whether the influence of congregational representatives can be observed solely from the outcome of the plenary voting sessions. To the best of my knowledge, this is the first study to try to identify influential members within parliaments. To accomplish this, a congressional similarity network will be build connecting congressmen.

There is a significant amount of quantitative analysis of parliaments that is not necessarily related to the network approach. A couple studies have focused on using dimensionality reduction techniques on roll call voting datasets to identify structures within parliaments such as clusters of like-minded politicians or to describe politicians using a low dimensionally space representing ideology. Others have attempted to predict national elections results or issue voting considering the nature of the subject. Some papers focused on longitudinal aspects of those structures such as shifting political alliances or ideological consistency over time.

Most empirical works investigated parliaments using roll call votes as their data source. Other dataset commonly used were social media, opinion polls and datasets from voting advice applications, websites on which politicians and citizens can reveal their opinions regarding political issues to create a match between both parties according to their political similarity.

A common approach used by works studying roll call votes is to build a network in which vertices are congressmen and weighted edges represent voting similarity. Poole (2007) explored the longitudinal aspects of this network within one congress cohort to verified how consistent individual congressmen were across time. Waugh et al. (2009) ranked congressmen by their network centrality and used network modularity to track polarization changes across different cohorts of the American

Congress. They also studied how modularity could be used to predict reelection for party political leadership, and found that periods of medium modularity are more likely to lead to changes in existing leadership.

Dal Maso et al. (2014) computed the importance of a deputy within a coalition by how much modularity the network would lose if that deputy diverged from its community and moved to the farthest community. Based on this metric, they characterize the heterogeneity of the government coalition and account each political party contribution to the stability of the Italian government over time.

Porter et al. (2007) built a network of committees and sub committees within U.S. house of representatives where nodes were committees and edges were weighted by common membership among committees. After doing a Single Value Decomposition on roll call votes of congressmen, they also did a hierarchical clustering of those committees and sub committees to determine its degree of political ideology.

MacOn et al. (2012) studied the voting network of the United Nations General Assembly using different network construction techniques and compared how each of those formulations handled community detection.

In additional to roll call voting, congressmen may also support a legal project by cosponsoring it, i.e. signing his or her name to a proposal that has been introduced for consideration. Zhang et al. (2008) built a co-sponsorship network to investigate a similar question done by Porter et al. (2007), regarding congressmen membership among the network committees. This same work also calculated co-sponsorship network modularity to verify how polarization changes across different cohorts of the American Congress, similarly to the work done by Waugh et al. (2009).

Twitter data was used by Peng et al. (2014) to observe interactions among American congressmen to investigate homophily effect within this network, i.e. the preference that nodes have to establish links to other nodes that are similar to them.

Data from voting advice applications was analyzed by (Hansen & Rasmussen, 2013) who applied Single Value Decomposition to identify singular vectors and verify how cohesive candidates within a political parties were. Etter et al. (2014), used a similar methodology on two datasets, one containing candidates' opinions expressed before the elections and another with their actual votes casted by elected members of Swiss parliament. Then, they used politician position in a space of singular vector to

verify how consistent elected congressmen were when compared to their position while running as candidates.

Within the Brazilian context, Baptista (2015) developed a model to identify when congressmen deviated from its original group regarding his voting behavior. Their methodology was based on deviance from ideal points according to spacial vote theory, a methodology that measures the position of politician within a political continuum.

Another group of studies attempted to predict the results of elections. For example, sentiment analysis on Twitter data were used by Tumasjan et al. (2010) and Sang & Bos (2012) to predict the results of national elections in Germany and Netherlands.

Outside the strict domain of political science, Takes & Heemskerk (2016) did an extensive study of a global network of corporate control. He did so by building a network of firms as nodes and edges representing shared directors between these firms. Afterward, using the concept of centrality he identified the importance of firms and their participation into the network using two concepts developed in his work: centrality persistence and centrality ranking dominance. The former is a measure of the persistence of a partition centrality in the full network, while the second indicates whether a partition is more dominant at the top or the bottom of the centrality ranking.

Finally, advances in natural language processing stimulated some works applying it to predicted voting outcome for a legal project based on its text. Gerrish & Blei (2012)used labeled latent Dirichlet allocation and item response theory to develop a model. Sim et al. (2013) did a related work in which they build a domain-informed Bayesian Hidden Markov Model to infer the proportions of ideologies using candidates speeches. Gu et al. (2014) built a topic-factorized ideal point estimation model by modeling the ideal points of legislators and bills for each legal project topic.

2.2 The context: The Brazilian Parliament

In Brazil, the parliament is divided into two chambers. The lower chamber is the House of Representatives and the higher chamber is the Senate. Every law project must have its text approved by both chambers before it sanctioned by the Republic President, the maximum representative of the Executive power. Parliament members are affiliated to parties and must represent their home states.

The House of Representatives is composed by 513 deputies with four-year mandates. The number of deputies belonging to a given state is roughly proportional to its population (There are an upper and a lower bound of how many deputies a state can have. Consequentially, the smallest states have eight deputies, while the largest state, *São Paulo*, has 70 deputies). On the other hand, the Senate is composed of 81 senators elected for 8 years mandates. Each Brazilian state is represented by exactly three senators.

The number of parliament member related to each political party is unconstrained. Even though there are cases of acting congressional representative that doesn't belong to any party, they usually belong to one. The election for both chambers is placed every four years. However, while all the seats in the House of Deputies are change during an election, only a fraction of the seats in the Senate is changed (one seat per state on a given election and two seats per state in the following election). In both houses, reelection is a common phenomenon and it is allowed without any restriction. During the last election in 2014, 274 of the elected deputies (53,4% of the total) were reelected. Attendance to the national elections is mandatory for every Brazilian between ages 18 and 65 (although many don't voting since fees for absents are small). Voting is optional for those aged between 16 and 18 and those over 65 years old.

Both the House of Representatives and Senate have several thematic committees with a limited number of parliament members. Those committees debate and draft proposals of legal projects that after being approved by those committees proceed to be evaluated by the plenary of each chamber, the general assemblies that unite all the members of that chamber. In this dissertation, I will focus on the activity of the House of the Representatives plenary. This decision was made since the number of members in the House of Representatives is large and with a rich diversity of profiles among its members. Additionally, the restriction of the analysis to the plenary simplifies this work, since it unites all those actors in the same context and prevents the additional complexity of having to compare deputies that are not acting on the same context, i.e. deputies belonging to different committees.

Legal projects can be clustered into different types: *Medida Provisória* (express law enacted by the executive that expires after 30 days unless approved by the congress), Ordinary Law, Complementary Law, and Constitutional Amendment. For most legal projects, a project is approved if it reaches the minimum quorum of at least

50% of the valid votes (not including no-shows). Exceptions to this rule are complementary laws that require at least 50% of the available number of votes in the house (including non-shows). Also, constitutional amendments require bills to be voted twice on each chamber and to have at least 60% of the available votes on each voting session. For the 69 legal projects voted in 2015, 19 were *medidas provisórias* (temporary presidential orders), 25 were ordinary laws, 10 were complementary laws and 14 were constitutional amendments.

The possible outcomes for each congressional representative position are: yes, no, no-show, abstention, and obstruction. While the yes, no and no-show positions have obvious meaning, the abstention and obstruction positions require further explanation. A congressional representative can avoid making a clear signal to the public regarding whether his position on a legal project by using those two options. If voting an abstention, one still helps the proposition by providing quorum. On the other hand, a non-shown can be damaging to the proposition by removing quorum. However, by voting an abstention a congressman can still contributes to legal project by helping the voting session achieve the required minimum quorum to approve it, 50% for most legal projects and 60% for constitutional amendments. Finally, obstruction can only be called by the leader of a party and results in the withdrawal of members of that party from the voting session. This appeal is usually used as a final resource by parties that are strongly against the proposed legal project in question.

3 THEORETICAL FUNDAMENTS

This work will rely upon two frameworks, the first one is Machine learning and the second one is about Network Theory. This chapter provides the readers a conceptual understanding of these themes in order to provide the foundation for the methodology section.

3.1 Machine learning

This subsection presents the theoretical tools used in machine learning. It starts by providing some basic terminology, followed by a presentation of the main metrics used to quantify network structures and finally discuss some of the relevant patterns observed across many examples of real-world networks.

3.1.1 Machine learning types

Machine learning is the field of computing that aims to develop computational methods capable to do activities without explicit human programming. They do this by relying on their accumulated experience, i.e. its available data. Due to this, machine learning algorithms are fairly generic and can be used on several problems after adapting the algorithms to the problem domain Christiano Silva & Zhao (2016).

Machine learning algorithms are usually divided into three categories that reflect the type of feedback the algorithm receives. In ascending order of feedback presence, these types are: unsupervised learning, reinforcement learning and supervised learning.

In unsupervised learning, no feedback at all is provided. In reinforcement learning, the algorithm receives feedback whether its output has made a positive or negative contribution towards reaching its goal. Finally, in supervised learning, the algorithm has explicit information regarding what is the desired output for each input and it is expected that the algorithm can identify patterns using this data.

Those divisions may not be so evident. For example, semi-supervised learning deals with problems for which some instances of the available data provide feedback, while other instances do not. This lack of feedback may be due to actual lack of information, or even due to noise. Therefore, in practice instead of being a dichotomy concept, there is a continuum ranging from unsupervised learning up to supervised

learning (Russell & Norvig, 2009).

In supervised learning, feedback is provided by labels associated to each data items. Producing these labels is an expensive, time-consuming, and prone to errors procedure that is commonly done by humans. Therefore, in most domains, labeled data is scarce while in comparison to the abundance of unlabeled one. Still, the presence of labeled data is fundamental to the development of high performance supervised learning algorithm in the first place which them leads to insights about the problem. Afterwards, those insights may lead to the application of unsupervised algorithms on the wider unlabeled data with successful results.

Examples of some of those methods can be seen on Figure 1 and Figure 2. Figure 1 exemplifies a clustering task, i.e. defining groups in which the included data items (represented by white circles) are similar to each other, while different data items belong to different groups. Again, by being an unsupervised method, this group allocation occurs without having any ground truth information. On the other hand, in Figure 2, is an example of a classification task. The model was build using a set of labeled data items (represented by circles). Later, the model was applied to infer the label of new data items (represented by white triangles).

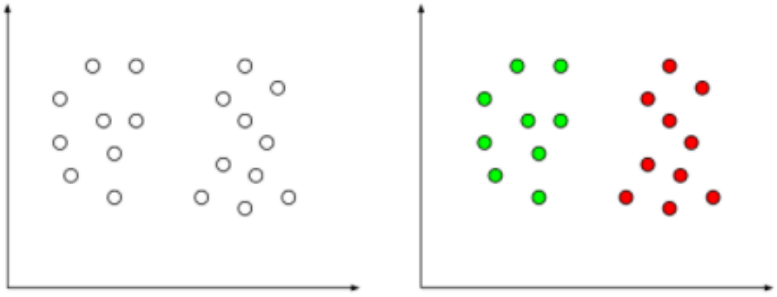

Figure 1: Example of unsupervised learning.

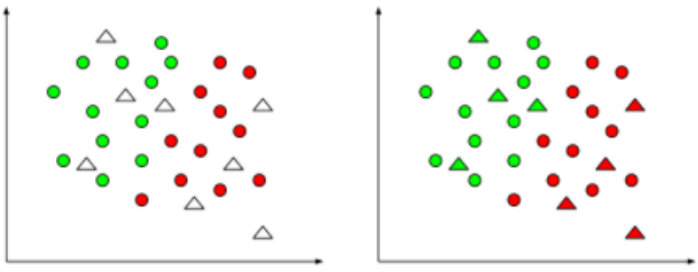

Figure 2: Example of supervised learning.

Since the methods used in this dissertation are mainly supervised, the rest of this section will focus on detailing those algorithms and some of the techniques associated to them.

3.1.2 Practical machine learning

This section outlines practical advice provided by Bishop (2006) and Domingos (2012) for approaching prediction modeling by machine learning. It involves the following steps:

1. Exploratory data analysis

Stage dedicated to understand the overall structure of dataset. It involves checking statistical summaries for variables, e.g. mean, range, and plotting variables against each other to check for clusters, correlations and outliers.

2. Preprocessing the data

Subsequently, the process of cleaning or transforming the data starts. It is often mentioned that this is the most time-consuming process. Tasks in this step include: merging separated datasets and reshaping them, handling missing data, taking transformations of variables, removing outliers.

3. Construction the model

This stage is executed interactively and is composed of three sub stages that are highly interconnected: Feature engineering, model training and model evaluation.

3a. Feature Engineering

Feature Engineering is perhaps the most critical and creative task in predictive modeling. In many problems, the datasets available has few useful features readily available, e.g. image and audio analysis, while in others there are too many features that are not relevant. Therefore, it is important to consider technics to extract useful features or that reduce the number of unimportant features.

3b. Train model

This task involves the actual training of the model. For classification tasks, the most common models are: Logistic Regression, Support Vector Machine, K-nearest

neighbors, Naïve Bayes, Decision Tree, Neural Networks and ensemble methods such as Random forest and Random Forests.

Almost every model requires choosing some sort of parameters, known as hyper-parameters. Therefore, in addition to deciding which model to train, one must also decide how to calibrate these hyper-parameters.

3c. Evaluate model

The models need to be evaluated regarding their ability to generalize its results into additional data. This requires an external evolution known as cross-validation that computes the error rate for model predictions applied to data items that were not used for training.

4. Select best model.

Since there is a significant number of design choices required while building predictive models, some sort of feedback is required to decide which design is more appropriate. In addition, the purpose of a model is to be able to generalize data inputs outside its training experience. Both problems can be solved by using an empirical procedure, known as cross-validation.

The simplest form of cross-validation is to randomly split the dataset into two parts, one part is called training dataset, while the other part is called testing dataset. The model should be trained using the training dataset, while the testing dataset must be set apart and only be used after the model is built, in order to measure its generalization capability.

A more complex form is the k-fold cross-validation. On this method, the original dataset is randomly partitioned into k parts, also known as folds, of equal size. A total of k models a built by keeping a single different partition on each model being used as testing dataset on each of them. The idea is that for each model, the training dataset and the testing dataset are disjoint data.

The parameter k is usually set to 5 or 10, but when little data is available one may consider the extreme case where k = n, a situation known as leave-one-out cross validation. While this leads to a better use of the available data, this is often infeasible on large datasets due to time needed.

3.1.3 Performance metrics

There are several definitions for performance metrics, but this section will focus only on the ones related to classification tasks.

On classifications problems in which a data item can only have a single label, the confusion matrix is a useful tool to evaluate the performance of an algorithm. For problem with m classes, then the confusion matrix would be squared matrix with size m x m. Each data item is added to the matrix by its column from the predicted class, while the data row would represent the actual class. Table 1 demonstrates an example of a confusion matrix for a problem with only two classes.

Table 1: Confusion matrix for binary classification.

	Actual Positive	Actual Negative
Predicted Positive	True Positive (TP)	False Positive (FP)
Predicted Negative	False Negative (FN)	True Negative (TN)

The confusion matrix is often summarized using single value metrics. The most popular of them is the accuracy, i.e. the fraction of data instances that were correctly classified. Formally defined as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Even though the concept of accuracy is very intuitive, it is not the most adequate in many cases. Some classification problems possess very unbalanced classes, i.e. the number of instances in one class may be much greater than the number of instances in other ones. In this case, a naive method that predicts every data instance as belonging to the dominating class can have a high accuracy score while not being useful.

Therefore, additional metrics are required. The confusion matrix provides more performance metrics that can be computed to handle problems with unbalanced classes such as the precision, recall and F-1 metrics.

Precision is the fraction of the instances classified as positive and that were indeed positive. Formally:

$$Precision = \frac{TP}{TP + FP}$$

Recall is the fraction of the positive instances that were correctly classified as so. It is also called as the True Positive Rate. Formally, it is defined as:

$$Recall = TruePositiveRate = \frac{TP}{TP + FN}$$

Both metrics range from 0 to 1 and, in most prediction systems, there is a tradeoff between precision and recall. A system may be calibrated to improve its precision score, but it does so at expensive of is recall score and vice versa. Therefore, it is important to have a criterion to decide an optimum point.

This tradeoff can be commonly seen in the Precision-Recall Curve as the one exemplified in Figure 3.

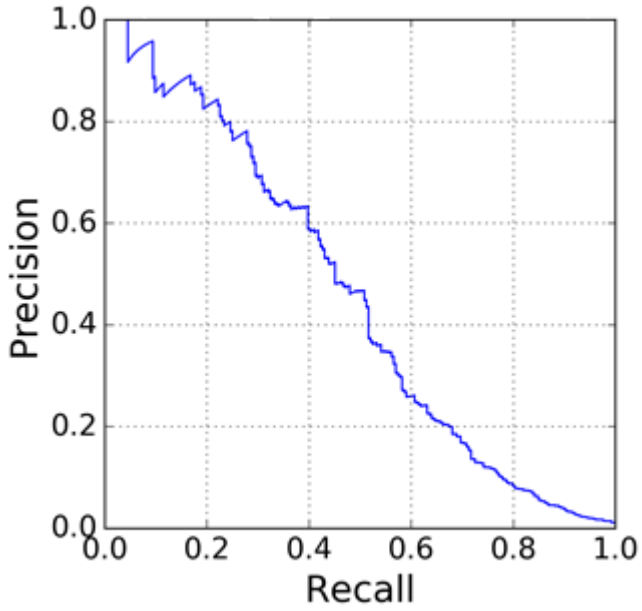

Figure 3: Example of the precision-recall curve.

Finally, F-beta is an approach that attempts to combine precision and recall into a single metric whose value ranges from 0 to 1. It is defined as:

$$F_\beta = (1 + \beta^2)\frac{precision * recall}{(\beta^2 * precision) + recall}$$

The parameter β is adjusted to decide how to priorize between precision and recall. In most cases, beta is set as 1, leaving both metrics with equal importance. Another important tool for the models' evaluation is the Receiver Operating Characteristic Curve, also known as the ROC curve. This curve is plotted based on two

other error metrics obtained from the confusion matrix: the True positive rate and the false positive rate.

The true positive rate, also known as sensitivity, is defined similarly to the recall metric, the probability of detection of a positive instance. On the other hand, the false positive rate, also known as False Discovery Rate, is the probability of negative instance is considered positive. Therefore, it is formally defined as:

$$FalsePositiveRate = \frac{FP}{FP + TN}$$

The ROC curve is plotted with the false positive rate in the x axis and the true positive rate in the y axis similar to the example on Figure 4.

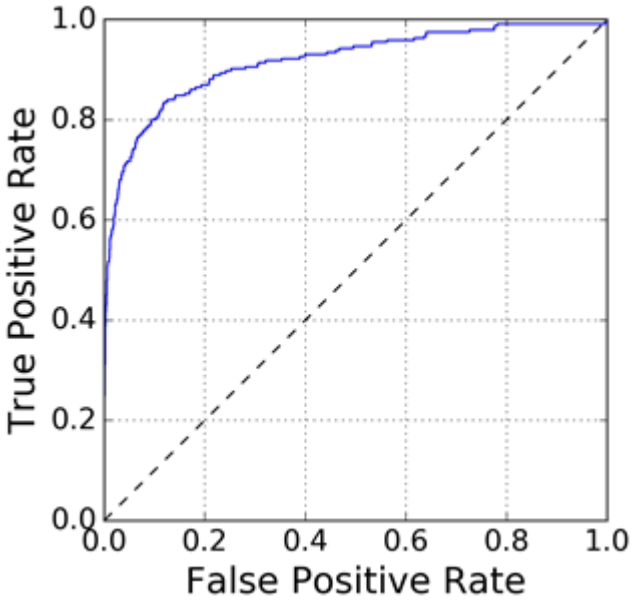

Figure 4: Example of the receiver operating characteristic curve.

Similar to the tradeoff between precision and recall, true positive rate and false positive rate are inversely related metrics. A popular metric to mediate the tradeoff between them is to compute the area under the ROC curve, also as known as AUROC, whose value varies between 0 and 1, will 1 being perfect classification. Hosmer& Lemeshow (2004) defined evaluation criteria for interpreting the quality of the results of the AUROC as shown in Table 2.

Table 2: Interpretation of AUROC values.

AUROC	Value
0.5 |– 0.7	No discrimination
0.7 |– 0.8	Acceptable discrimination
0.8 |– 0.9	Excellent discrimination
0.9 |– 1.0	Exceptional discrimination

The Precision-Recall and the ROC curve are connected in a way that the curve for a given model is superior in the ROC curve if, and only if, it is also superior in the precision-recall curve. However, as pointed out by Davis & Goadrich (2006), for highly skewed dataset, the precision-recall curve provides a more informative picture of a performance model.

3.1.4 K-Nearest neighbors

The k-nearest neighbor's method is an extension of a lookup table in which the predicted class for a data instance is based on a plurality of votes from the k nearest data instances. Therefore, the k-nearest algorithm is nonparametric model since the model cannot be summarized by a set of parameters of fixed size.

The main parameters to be adjusted in this model are: the value of k, the distance function, and the voting weights to each of the k selected instances. The first metric, k, must be in an optimal point since setting the value of k too low leads to overfitting the data, while setting it too high underfits it.

Regarding the distance function, the most popular approach is to use simple Euclidean distance voting weights. Although more complex function can be used to account for variation among features, such as Mahalanobis distance. The main recommendation on this issue is to normalize data to keep them in the same scale and prevent a few large-scale features from contributing disproportionally to the overall distance.

Finally, regarding voting weights, it is usual to have instances with the same weight while keeping k an odd number to prevent draws. Another popular approach is to weight instances proportionally to their closeness.

In low-dimensional spaces k-nearest neighbors with abundance of data tend to perform well. However, the in high-dimensional spaces the curse of dimensionality quickly decreases the density of instances in the space. Another disadvantage of this method is that is executed in linear time, making it unfit for application that requires high scalability (Russell & Norvig, 2009).

3.1.5 Decision trees

On this section introduces the concept of decision trees and discuss some of its properties. Later, it presents how its learning algorithm works and how ensemble methods can be used to build committees using the Random forest algorithm.

3.1.5.1 Decision tree structure and proprieties

Decision trees are hierarchical structures for decision making that resemble the way human make decisions. Despite its simplicity, decision trees have been successfully applied to a wide range of problems classification (Kotsiantis & Zaharakis, 2007).

An example of a decision tree can be seen on Figure 5. The node in tree can be decision nodes or leaf nodes. An instance initiates on the root node of the tree and it is subjected to if-else conditions on decision nodes until it reaches a leaf node.

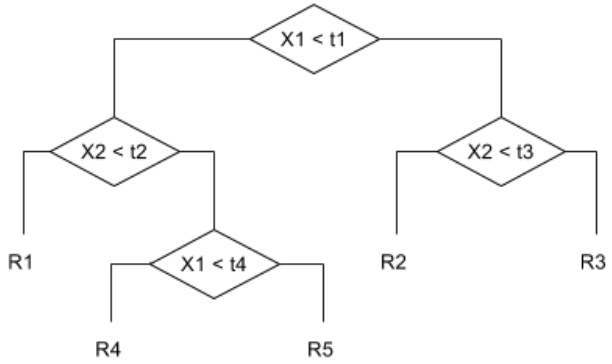

Figure 5: Example of a decision tree.

Consequentially, in the case of binomial classification, a decision tree with n

decision nodes divides itself into n+1 partitions and assigns a probability to instances in each partition to belong to a given class, as shown in Figure where data instances within regions $\{R_1, \dots R_6\}$in which each of them has a probability of being part of a class given by $\{c_1, \dots . c_6\}$.

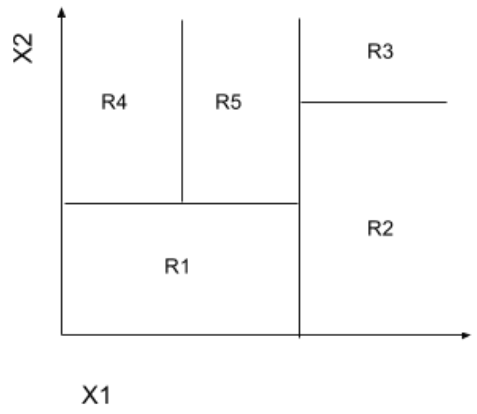

Figure 6: Example of partition of the input space for a decision tree.

One of the most remarkable propriety of decision trees is that it is a white box model. This is particularly important for many applications subjected to legal restrictions that requires algorithms to be able to be inspected by humans (Kotsiantis, 2013).

Also, most decision trees algorithms are versatile in a way that they can be used on categorical data with no need to do previous encoding into numerical data. This is a contrast, since most algorithms are usually specialized in analyzing datasets that have only one type of variable.

Additionally, decision trees can be robust towards missing data. According to Hastie et al. (2011), there are two ways to handle such cases. 1) Consider a missing input as a new category and treat if differently from situation where the information is available. 2) Rank surrogate variables that are highly correlated with the missing one. This way, if the main variable is not available, it uses the next best available variable as a proxy.

On the other hand, decision trees also have challenges. They are susceptible to data fragmentation, i.e. having too few data instances on leaf nodes. These over-complex trees usually do not generalize well to new instances. However, using ensemble methods of simple trees with a low depth or a minimum number of instances on each leaf node can avoid this problem.

Also, decision trees are unstable to small variations in the data that might lead to the creation of a completely different tree. Again, this problem can be mitigated by using ensemble methods. Finally, on problems with imbalanced classes, it is recommended to balance the dataset to avoid biased trees.

3.1.5.2 Learning a decision tree

The CART (Classification and Regression Trees), created by Breiman et al. (1984) is one of the most widely adopted algorithm and will be used here to better understand decision tree learning.

In a classification decision tree, where a node m represents a region R_m with N_m sample, the proportion of class k data instances in this node is:

$$\hat{p}_{mk}(\boldsymbol{x}) = \frac{1}{N_m} \sum_{\boldsymbol{x_i} \in R_m} I(\boldsymbol{y_i} = k)$$

The classification of data instances in this node would follow a majority rule.

A decision tree learning algorithm must find the optimal partitioning such that values of p_{mk} and R_j minimize a given error metric. Laurent (1976) discovered that finding the optimal partition of data is a NP-complete problem. Therefore algorithms to find global optimization are impractical. A common approach to work around this is to use a recursive greedy strategy.

The algorithm starts with all available data instances and finds the variable that optimizes the first partition. Therefore, four variables must be found, the split variable x_p, the split threshold or split subset (respectively, for the case a numerical or a categorical split variable), and the outcome values for the left and right branches of the tree.

The quality of the split is measured using an error metric. For classifications tasks, the most common metrics are: misclassification rate, cross-entropy and the Gini index. They are formally defined by Hastie et al. (2011) as:

Misclassification error: $\frac{1}{N_m} \sum_{\boldsymbol{x_i} \in R_m} I\big(\boldsymbol{y_i} \neq k(m)\big) = 1 - \hat{p}_{mk(m)}$

Gini index: $\sum_{k \neq k'} \hat{p}_{mk} \hat{p}_{mk'} = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$

Cross-entropy: $-\sum_{k=1}^{K} \hat{p}_{mk} log \hat{p}_{mk}$

Despite having their origins on different fields, those metrics are reasonably conceptually similar by measuring the impurity of a node. Since Gini and cross-entropy is differentiable, they are usually preferred for using a numerical optimization. An example of the curve for those measures can be seen on Figure 7. On this figure, the horizontal axis is the percentage of data instances belonging to class 1. In all metrics, the maximum impurity is when the node has an equal number of instances from both classes.

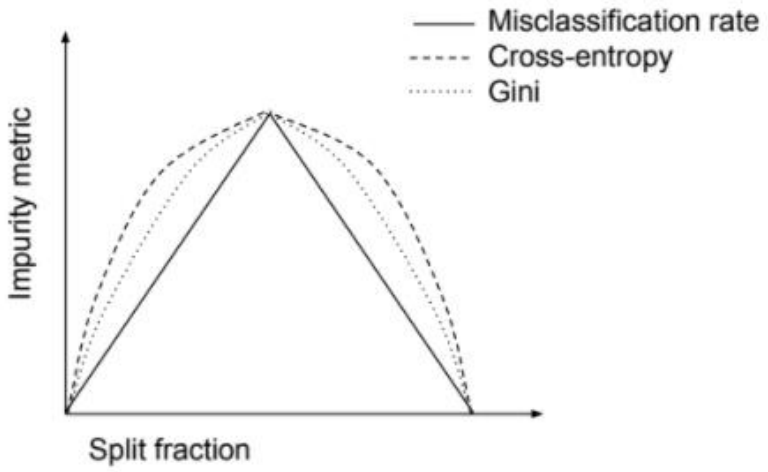

Figure 7: Example of impurity metric curves for binary classification.

The solution for a classification tree would mainly deal with finding the split variable and its threshold. Considering Gini index as the error metric, this can be formulated as:

$$\{x_p, s, c_1, c_2\} = \underset{x_p,s}{\operatorname{argmin}}[arg\min_{p_{mk}} \sum_{x_i \in R_1} \hat{p}_{mk}(1-\hat{p}_{mk}) + \underset{p_{mk}}{\operatorname{argmin}} \sum_{x_i \in R_2} \hat{p}_{mk}(1-\hat{p}_{mk})$$

For a numerical or categorical variable with q distinct values, the variable can be split in *q - 1* possible ways. However, unsortable categorical variables must be split considering subsets of it. Searching a variable of such type can be computationally expensive if the number of categories *q* is large, leading it to have $2^{q-1} - 1$ possible partitions. In addition to this, according to Loh (2011), those categorical variables may became overrepresented in the model.

Overall, it is computationally viable to do a strong approach and explore every possibility to find the optimal partition. After finding it, the algorithm must be recursively applied to each of the leaf of the newly constructed node, until some given stop criteria

is met.

Engineering the stop criteria is an important modeling decision since it greatly helps preventing overfitting the model. The most common criteria are limiting the depth of the tree or limiting how few data instances can be a terminal leaf. Another popular approach is splitting a node only if the quality of this division is above a certain threshold (Murphy, 2012).

Given the greedy nature of the decision model, an end criterion may prematurely prevent tree growth. For example, a low-quality node partition may generate two regions on which high quality partitions can be found. Therefore, it is a common approach to grow an overly complex tree with a loose criteria and afterwards prune the tree (Kotsiantis & Zaharakis, 2007).

The pruning stage is also a greedy strategy interaction, starting at leaf nodes and proceeding to the tree complexity, using some regularization. Regularization is the process of optimizing a model by not only minimizing its training error, but also introducing a model complexity penalty. It can be formulated as in the equation below.

$$E_\lambda(T) = E(T) + \lambda|T|$$

Where $E(T)$ is the error term for all data instances that are in the training dataset, $|T|$ is the complexity metric for the tree, e.g. the number of nodes, or the depth of the tree. λ is a regularization parameter that adjusts how much the optimization model focus on reducing training error vs. reducing tree complexity. Determining the optimal value for these parameters is usually done using some sort of cross-validations, e.g. k-folding.

3.1.5.3 Other decision tree algorithms

Many learning algorithms have been proposed as decision trees. The first one was the AID, proposed by Morgan & Sonquist (1963), later, the CART (Classification and Regression Trees) by Breiman et al. (1984) and the C4.5 model by Quinlan (1993) were published with large adoption on practical applications. Those later models were very similar to each other, but they differ in some aspects, since C4.5 does not supports numerical target variables. On the other hand, the C4.5 enables multiple partitioning in the decision node instead of the usual partition seen in CART algorithms.

As previously mentioned, using of unordered categorical variables with many

categories leads to over representing them. Some new algorithms are designed to avoid this bias as pointed out by a literature reviews done by Loh (2011). Those algorithms are: CRUISE, GUIDE, and QUEST. On this paper will focus on explaining the CART algorithm since an optimized version of is implemented on SciKit-Learn and will be used.

3.1.5.4 Models using tree committees

To mitigate the high variance observed in decision trees models, it is common to create committees with many decision trees, each one created with a subsample of the dataset. The individual prediction models that belong to this committer are combined following a set of rules to form the final prediction. There are two paradigms on how to build embedded models: parallel and sequential.

An example of a parallel model is Bagging, a short name for Bootstrap aggregation, it was created by Breiman (1996). A variation of this procedure is known as Random Forests, created by Breiman (2001). Random forests sample isn't only a subset of the dataset, but also has each decision tree built from only a sampled subset of the input features. This allows this algorithm to build a large collection of de-correlated trees Hastie et al. (2011).

An example of a sequential model is the popular AdaBoosting algorithm, which was first introduced in Freund& Schapire (1996) and became one of the most widely used models. Later, the Gradient Boosting was developed in Friedman (2001).Gradient Boosting is applicable to both classification and regression problems by using any differentiable loss function. Unlike Random Forests, on Gradient Boosting, each tree model is built using the error obtained by the previous tree in order to adjust the weight of its data instances.

Among those models, random forest is most likely the most popular since its implementation is easily deployable in distributed systems. Additionally, its performance is comparable to gradient boosting despite requiring very little tuning.

The algorithm for random forest as presented by Hastie et al. (2011):

1. For b = 1 to B
    a. Collect a sample from the dataset with size N

b. Recursively grow a tree similarly to the algorithm shown in the previous section. However, for each split node, randomly subsample p input features among the m input features as splitting candidates.

2. Output trees $\{T_b\}_1^B$

The optimum values of B and p is usually chosen empirically, although a good initial approximation for p is $p = \sqrt{m}$. While predicting, each tree casts a single class vote. The final decision made by the overall committee is done by using majority voting among individual votes.

Committees of trees are known to provide high performance classifiers both with low variance and bias. However, the price paid while in contrast to single decision trees is the lack of clear interpretability.

3.2 Network theory

This subsection presents the theoretical tools used to describe and analyze networks. It starts by providing some basic terminology, followed by a presentation of the main metrics used to quantify network structures and finally discuss some of relevant
patterns observed across many examples of real-world networks.

3.2.1 Formal definition of graph

A graph G, is defined as an ordered pair (V, E) where *V* is a non-empty set of vertices and *E* is a set of edges connecting the vertices such as$E \subseteq \{(u, v) \vee u, v \in V\}$.

This formal definition is mostly used across mathematical literature, however, depending on the source, the terminology may be different. In graph theory, the preferred words are graph, vertex and edge. In network science, on the other hand, the word networks, node and link are preferred NEWMAN (2010). In this work will give preference to the second terminology.

For example, Figure 8 is a network with 6 vertices and 7 edges. While vertices have simple names, edges have composite names indicating the source and the destination vertices. Therefore, edges in Figure 8 are named (1,2), (1,5), (2,3), (2,4),

(3,4), (3,5), and (3,6).

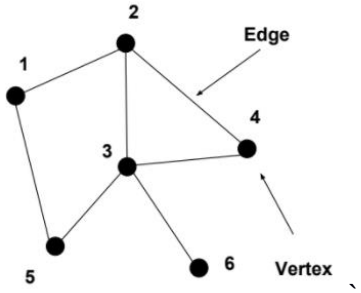

Figure 8: Network example.

Edges have many nuances across different networks. In some cases, the relationship between two entities may not be mutual, i.e. entity v_i may be lead to entity v_j, but entity v_j may not lead to entity v_i. This led to the conceptualization of directed edges. Network with directed edge are called directed networks or digraphs.

Some networks, called multigraphs may contain many edges, called multiedges, connecting the same pair of vertices. Also, some networks have self-loop, i.e. edges where the source the destination vertices are the same. An example of both can be seen on Figure 9.

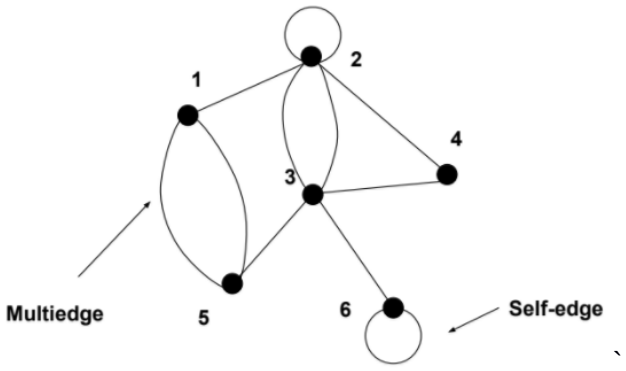

Figure 9: Network example with multiedges and self-loops.

Additionally to the complexity of its topological structure, real-world networks have a large heterogeneity in the capacity or intensity of its edges, as reminded by Barrat et al. (2004). Therefore, it is important to model more than just the presence or absence of connections within a network. This led to conceptualizing weighted networks, i.e. networks in which its edges have weights that reflect the strength of that edge.

An example of a weighted, directed network is a city road map in which each vertex is an intersection and the edges indicates streets connecting those intersections. Edge direction indicated permitted driving directions and edge weights represent distances. Additionally, since two or more different streets may connect the same intersections, multigraphs may be used to represent such cases.

3.2.2 Network representation

Adjacency matrices are a mathematically convenient representation to networks. A network with $|V|$ vertices is represented by a squared matrix, A with order $|V|$. A_{ij} is 1 if the edge connecting vertex i to vertex j is present and 0 if the edge is absent. For example, the adjacency network for the example in Figure 9 is shown below.

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Note that for undirected networks, the adjacency matrix is symmetric, i.e.$A_{ij} = A_{ji}$, while this usually is not true for directed networks. If the network is a multigraph, A_{ij} reflects the number of edges connecting vertex v_i to vertex v_j, self-loops in undirected network, however, and requires additional attention. Since each edge in an undirected self-loop connects to the same vertex twice, each self-looping edge should be counted as 2.

For a weighted network, the adjacency matrix is called a weighted matrix A, it is defined as $A_{ij} = w_{ij}$, where w_{ij} is the weight of the edge connection going from v_i to v_j. This way, edges not-contained in the network are represented with a zero, while every existing edge must have $w_{ij} > 0$. An example of a weighted network can be seen on Figure 10.

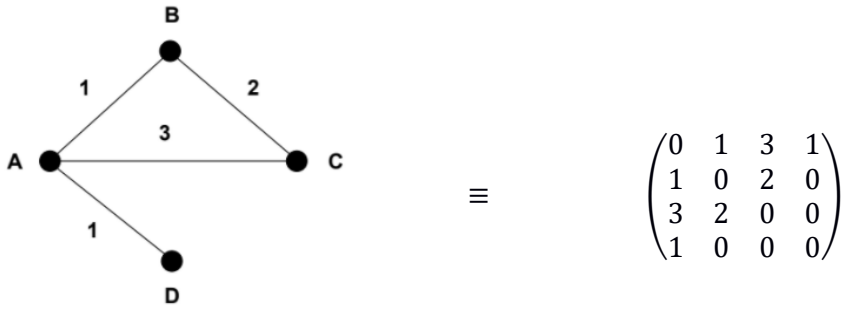

Figure 10: Example of a weighted network and its adjacency matrix.

In some cases, useful network analysis tools are available only to unweight networks. In those cases, some adaptations may allow applying standard techniques for unweighted graphs into cases with weighted networks. The most common adaptations involve mapping the weight matrix of the weighted network into an adjacency network.

The simplest way to do this operation is to use a threshold function, so that only weights larger than a threshold value are derived 1 (and 0 if below the threshold). Also, as pointed out by Newman (2004), in many cases weighted networks can be analyzed using a simple mapping from its original format to the format of an unweighted multigraph.

As a final note, adjacency matrices are not the only available representation of networks. Most real-world networks have many vertices and comparatively few edges. This leads to the presence of many zeros in its adjacency matrices. The proportion of zeros in an adjacency matrix is known as sparseness. Therefore, the adjacency matrix for most real-world networks is an inefficient form to storage network in terms of computer memory.

A better representation in this case would be to use a list of edges. Thus, the representation would be through a list of tuples, where each tuple contains the source vertex, the destination vertex and, optionally, the weight of the edge.

3.2.3 Network metrics

Networks can be better understood by observing some key metrics. This section will define some metrics that are considered the concepts of connectivity and centrality.

3.2.3.1 Connectivity

Two vertices are adjacent if there is an edge connecting them. For example, in Figure 10, A and B are adjacent, but C and D are not. Note that on digraphs it is possible that vertices A are adjacent to B, but the opposite may not be true. Furthermore, the neighborhood of a vertex v_u, $N(u)$, is the set of all notes that are at adjacent to v_u. Formally, $N(u) = \{v: (u, v) \in E\}$.

An important global metric is the degree of the connectivity in a network. Conceptually, this simply is the fraction of the possible edges that are present into the network. Formally, it is defined as:

$$d(u) = \sum_{v \in N(u)} A_{vu}$$

The value of the connectivity of network must range between 0 and 1. The formal definition of whether a network is considered dense of not relies on concepts of theoretical networks, which are not described in this short introduction to the topic. According to Newman (2004),with those theoretical networks it is possible to infer the connectivity of the networks as its number of vertices tends towards infinity. In these cases, a network is considered dense if its connectivity tends to a constant as the size of the networks tends towards the infinite. On the other hand, the network is considered sparse if the connectivity tends to zeros as the size of the networks tends towards the infinite

3.2.3.2 Path

Path is an ordered sequence of edges that visits distinct vertices (except possibly the first and last in case of the closed path, in which case the path is known as a cycle). For directed networks, each edge traveling by a path must be used in its correct direction. For undirected networks, edges can travel in both directions. Additionally, the length of a path is defined as the number of edges traveled it.

3.2.3.3 Components

A vertex v is reachable from vertex u if there is one path departing from u and heading to v. A network is connected if, for every vertex in this network, this vertex is reachable from any other vertex. In case this does not occur, the network is partitioned

into sub networks known as connected components. Each connected component is a connected network on itself and does not have any connection with another connected component. For directed networks, the definition of connected components is further detailed into segments, weakly connected components and strongly connected components. In the former case, edge direction is ignored and the network is treated as an undirected network. In the latter case, edge direction is considered. A common pattern across many real-world networks is that the size of the largest component, known as gigantic component, is a large fraction of the size of the entire network.

3.2.3.4 Shortest path

The shortest path between two vertices in a graph, also known as geodesic path, is a path of vertices that travels from the source vertex to the destination one, passing by the minimum number of edges among those two vertices.

For a weighted graph, the definition of shortest path is similar, but instead of minimizing the absolute number of paths, what is minimized is the sum of the weights along the travelled edges. In both cases, there may be more than one shortest path between two nodes if distances travelled along the proposed paths are the same.

3.2.3.5 Degree and strength

The degree of a vertex u is $k(u) = N(u)$, i.e. the number of edges it has. On digraphs, degrees can be further calculated as in-degree and out-degree, by respectively considering only the number of vertices that have edges with destination or source in the vertex u.

The degree concept was extended to weighted networks by Barrat et al. (2004). In this case, it is defined as the total sum of the weighted edge connecting the vertex. For a vertex u, the strength is:

$$s(u) = \sum_{v \in N(u)} W_{vu}$$

The use of this definition also allows expanding the concepts of in-degrees and out-degree for weighted networks, creating the concepts of in-strength and out-

strength.

Vertex degree and strength are a local metrics in the sense that it depends on the vertex itself and its neighbors. Still, the network-wide aggregation of this metric can be meaningful. Such aggregation is known as average network degree and strength. They are defined as: $\bar{k} = \sum_{v \in V} k$ for unweighted networks and $\bar{s} = \sum_{v \in V} s$ for weighted networks.

3.2.3.6 Clustering coefficient

In many real-world networks, especially social networks, the vertices tend to create highly connected clusters. Watts & Strogatz (1998) proposed the clustering coefficient for a vertex i, as the degree to which its neighbors form edges among themselves. Formally:

$$CC_i = \frac{2|E_i|}{k_i(k_i - 1)}$$

Where E_i is the number of connections among vertex i neighbors and k_i is the degree of the vertex i. In this way, CC_i must be a value ranging from 0 to 1. Similarly to the metric degree, the clustering coefficient is a quasi-local measure that can be aggregated network-wide. The network-clustering coefficient is defined as:

$$CC = \sum_{v \in V} CC_i$$

3.2.4 Network centrality

The concept of centrality measure attempts to quantify the importance of a vertex within the network. A vast number of centrality metrics were proposed in the literature and a summary of the most important ones can be seen in Newman. This work will mostly use Strength, Betweenness, Closeness, and PageRank. The following section describes one each of them.

3.2.4.1 Degree

The simplest and most intuitive concept of vertex centrality is to consider its degree, or strength. In additional to its simplicity, another advantage of this metric is

that it is calculated by using only the neighborhood of each vertex. Therefore, unlike other concepts of centrality such as the Betweenness, Closeness and PageRank, computing the degree of a single vertex does not requires knowledge of the complete network.

Since different networks have different sizes, it is important to normalize the degree when comparing different networks. This is usually done by dividing the degree of a vertex by the maximum possible number of edges in order to keep the degree range from 0 to 1.

3.2.4.2 Betweenness

The concept of betweenness was introduced by Freeman (1978). While closeness was a centrality metric based on distance, betweenness is based on flow. It is closely related to the concept of shortest path.

The intuition behind it can be better understood considering a scenario of a transportation network in which vertices are cities and edges are roads. Consider that each city needs to send messages to every other city in a uniform distribution. In this case, the betweenness metric measure would be how many messages would pass by a given city within a timeframe.

In the context of political science, a famous case of the importance of betweenness centrality was shown by Padgett& Ansell (1993), after building a network of marriages among traditional Florecian families during the Renascence. In this network, vertices are families and edges are marriages connecting different families. This work shown that the Medici family, had a distinct position connecting separated clusters of families. Consequentially, they rose to power by becoming main intermediary for transactions across those groups.

The calculation of the betweenness centrality requires the calculation of the shortest path between every possible pair of vertices in a network. Once this step is done, the betweenness centrality of a vertex v is the fraction of all the possible shortest paths in the network that travels along the vertex v (NEWMAN, 2005). Formally, the Betweenness B_v is defined as:

$$B_v = \sum_{s \neq v \in V} \sum_{t \neq v \in V} \frac{n_{st}^v}{n_{st}}$$

Where n_{st}^{v} is 1 if the shortest route between s and t passes throughout vertice v and 0 if otherwise. Since it is possible to have multiple minimum paths connecting the same pair of vertices, it is important to normalize the equation with the term n_{st} which is the number of shortest paths from vertices s to t.

As reminded by Christiano Silva & Zhao (2016), a consequence of this definition is that vertices with high betweenness have an important role in the communications within the network. In other words, the vertices with high betweenness are the ones in which the largest number of messages need to cross throughout it. This way, those nodes may either observe the message or charge a fee for its relaying service. Finally, the removal of vertices with high betweenness also tends to bring a discretionally high impact on the disruption of a communications within the network. In real-world situations, of course, not all vertices exchange communications with the same frequency, and in most cases, communications do not always take the shortest path, due to, for example, political or physical reasons. Also, the assumption that the frequency of messages exchange among the vertices is uniform often does not hold.

3.2.4.3 Closeness

Closeness is a distance related centrality metric, as said Sabidussi (1966) when defined the closeness centrality of a vertex v as the inverse of the average distance from for all other vertices in the network within its reach.1 Formally, the closeness centrality of a vertex i, known as x_i, is defined as:

$$x_i = \frac{n-1}{\sum_{j \in J} d_{uv}}$$

Where J is every vertex in the network that is reachable from v except itself, n is the number of vertices in the network, and d_{uv} is the shortest path from vertex u to vertex v.

In social network context, vertices with high closeness, can message other people faster than vertices with lower closeness. Despite being a very intuitive metric, in the context of social networks, closeness has some problems. Those networks have small world propriety, i.e. the average distance uses logarithmic scale to the number of vertices. Consequentially, the range of values for closeness is rather short making it

.

hard to distinguish high rank vertices from low rank vertices as pointed by Watts& Strogatz (1998).

For example, in Watts & Strogatz (1998), a network of actors was built using data from Internet Movie Database. Vertices were actors and edges were present between a pair of actors if they stared in the same movie. The actor with the highest centrality is Christopher Lee with 2.41, while the actress with the lowest centrality is Leia Zanganeh with 8.66. While those values are clearly distinct, there are other half million vertices in the network whose closeness centrality are compressed within this narrow range as pointed by Padgett & Ansell (1993)

3.2.4.4 Eigenvector

Eigenvector, Katz, and PageRank, are a class of centrality metrics that are based on recursive centrality Newman (2010). Recursive centrality is akin to an extended form of centrality degree in the sense that the more central are the neighbors of a vertex v, the more central is the vertex v itself. Therefore, a vertex may become more central by either connecting to lots of other vertices, or by connecting to a few vertices that have high centrality Newman, 2004.

The Eigenvector centrality has the most straightforward definition: The Eigenvector centrality of a vertex i, known as x_i is proportional to the sum of the centrality of its neighboring vertices J as in:

$$x_i = \sum_{j \in J} A_{ij} x_j$$

Where A is the adjacency matrix and x_j is the centrality of the neighbor vertex j. Similar to other recursive centralities metrics, it is a common practice to calculate the Eigenvector centrality using successive interactions that provide better estimates. The first interaction is:

$$x(t) = A^t x(0)$$

One calculates $x(0)$ as linear combinations of the eigenvector v_i of the adjacency matrix A. After some algebraic manipulation and considering the limit case when $t \to \infty$, the definition becames as first proposed by Bonacich (1987).

$$x_i = \kappa_1^{-1} \sum_{j \in J} A_{ij} x_j$$

Where κ_1 is the largest of the eigenvalues of **A**.

3.2.4.5 Katz

Eigenvector centrality is successful at achieving the desirable proprieties of allowing a vertex to be influent by either having a high number of edges with low rank vertices or by having a small number of edges with high rank vertices. Still, it leaves some problems when considering directed networks. A vertex has many edges pointing towards other vertices but no edge pointing towards it. Therefore, it would have a null rank to contribute to its neighbors.

A solution considered to this issue is to provide a minimum amount of rank to each vertex regardless of its connections. This centrality measure is known as Katz centrality and was proposed in 1953 by Katz (1953) formally defined as:

$$x_i = \alpha \sum_{j \in J} A_{ij} x_j + \beta$$

Where α and β are positive constants. Like eigenvalue centrality, α is the largest eigenvalue of the adjacency matrix while β is the minimum rank that a vertex may have. By convention, β has value equal to 1.

3.2.4.6 PageRank

Katz centrality leaves one additional opportunity for improvement. If a vertex has many edges connecting towards other vertices, its rank is allocated to each neighboring vertex regardless of how selective is their endorsement. PageRank centrality innovation has the objective to dilute the endorsement provided by a vertex proportionally to the number of vertices being targeted by it. Formally, PageRank centrality for a vertex i it is defined as:

$$x_i = \alpha \sum_{j \in J} A_{ij} \frac{x_j}{k_j^{out}} + \beta$$

Where α and β are the same positive constants seen in Katz centrality, and

k_j^{out} is the out-degree of the vertex i, i.e. the number of edges, or the sum of the weights, that the vertex v has pointing to other vertices.

While PageRank is defined for directed networks, some special cases for undirected networks occur. In those cases, the results for proportional to the degree of the vertices in the graph (Grolmusz, 2012).

3.2.4.7 Choosing centrality metrics

As a final thought, it must be highlighted that although those centrality metrics can be applied to any network, some of them are more appropriate than others since each one is based on a different idea. Still, there is little theory base on how to select the best centrality metric. It is usually recommended to evaluate the resulting rank of such metrics and compare them with a ground truth source which were often built by human experts in the domain playing this evaluation role.

3.3 Network construction techniques

This section describes some network construction techniques. Networks are often used to model local relationships between data points and then build global structures. Also, framing a problem as network can provides an opportunity to gain additional insight about the phenomena being studied by using the tools provided by network science.

Constructing a network involves two steps. First, one chooses the unity of analysis for the dataset and defines vertices according to this choice. Second, one generates edges across those vertices. Although the first step is straight forward, the second stage relies on two critical design decisions: the choice of the similarity function and the network formation technique Christiano Silva & Zhao (2016).

Similarity functions have been employed in several fields such as machine learning, information retrieval, pattern matching, and fuzzy logic Russell & Norvig (2009). Consequentially, existing literature provides many choices for the decision of these functions. While choosing the similarity function, it is important to consider the nature of the feature being used to compute similarity. Those data types are:

Categorical: This type of feature has two or more categories that have no hierarchy among them. For example, party affiliation is a category since no party is

more important than other and there is a discrete number of them.

Ordinal: Similar to categorical features but possessing a hierarchy. For example, in some surveys the education attainment is collected using the following options without high school, high school degree, college degree, postgraduate degree.

Numerical: Features that are measure in a numerical continuum such as income and years of education.

Given that the focus of this work will be using categorical data, it will focus only on similarity metrics for this data type.[2]

Two similarity functions were used: Jaccard and a modified version of Jaccard to create asymmetric directed edges.

Jaccard similarity is the most popular similarity function since its introduction by Jaccard (1901). One benefit of the Jaccard similarity is that it ignores categories that are not present in the data, e.g. congressmen who did no-show on voting sessions. It can be defined as:

$$s_{Jaccard}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Using Jaccard similarity leads to the creation of an adirectional network. Since not all relationships in networks are fully mutual, it is important to not to lose this lack of symmetry. Therefore, I modify Jaccard similarity by making the denominator only consider the cardinality of the source vertex. This leads to the creation of a directional network.

$$s_{assymetric}(A, B) = \frac{|A \cap B|}{|A|}$$

Finally, in some cases, it is convenient to use the dual of the similarity function, the distance (or dissimilarity) function, since many topological metrics, e.g. closeness and betweenness, require edge weights to reflect the distance among nodes. In these cases, a similarity metric can be converted into a distance one by taking its complement, i.e. $distance = 1 - similarity$.

2 Arguably, as will be better presented in the methods sections, the data used could be considered ordinal since, according do domain experts, there is some hierarchy among the possible voting options.

4 DATASET AND METHODOLOGY

This chapter provides to the readers a detailed implementation of the present methodology and the datasets used on this dissertation, setting ground for the results exposed in the next chapter.

4.1 Datasets

This work used two main databases of public data available: The House of Representative voting results during the year of 2015 and the *Cabeças do Congresso* report Queiroz (2015).

The work is focused on analyzing a single year in order to simplify this first attempt to answer its research question.

4.1.1 House of Representatives voting results

The House of The Representative's website keeps an API, which registers the position of each congressman for every legal project voted into the main plenary of this house Brasil (2015).

Since this API only records voting sessions in the plenary, activities in the thematic committees were not consider in this work. Nevertheless, for the year in question, 2015, the system presented detailed information on the outcomes of 249 plenary voting sessions regarding 69 legal projects.

4.1.2 DIAP report

Our research question is based on the hypothesis that topological characteristics of a network can be used to identify the most influential vertices, in this case, the congressmen in the House of Representatives. To verify this, one must have a ground truth source for information regarding the top influencing congressmen. This source was the report *Cabeças do Congresso*, published by the Inter-Union Department for Parliamentary Advisory (DIAP) [*Departamento Intersindical de Assessoria Parlamentar*], Queiroz (2015). Although this report has been published since 1994, the list itself has been published by DIAP since 1986. Thus, it is a reasonable source and is often referred by major Brazilian newspapers and academic works when conveying political analysis.

The core product of the report is an unsorted list of the top 100 most influential congressmen. Since the focus will only be on the House of the Representatives and the report contains both members of the House of Representatives and members of the Senate, data was filtered so that only deputies were considered. For the 2015 edition, 54 deputies were present in the list of top 100 congressmen. Given that the House of Representatives has 513 members, this shortlist represents 10.3% of the members of that house.

The report also contains deeper information about the top 100 congressmen. Those congressmen are further subdivided regarding their influence type. There are 24 congressmen as debaters, 14 known as articulators, 9 policy makers, 7 negotiators and opinion makers. I decided to simplify this work by abstaining from using this classification of influence type.

4.2 Methodology

The proposed methodology involves the major objectives:

1. To build a voting network and perform exploratory data analysis on it.
2. To identify influential congressmen by ranking of the centrality features.
3. To identify influential congressmen using classification algorithms.

Those objectives will be achieved by conducting the tasks shown on Figure 11, and the tasks will be detailed across the following subsections.

During this work was done using the Python programming language by Van Rossum et al. (2010), and the Pandas library by McKinney (2010) for data manipulation. Also, while doing network analysis, I used the Graph Tools library created by Peixoto (2014) and for tasks related to machine learning I opted for the SciKit-Learn library created by Pedregosa & Varoquaux (2011).

4.2.1 Data transformation and exploratory data analysis

The first task is to extract the two source datasets: voting results of the Brazilian House of Deputies and the *Cabeças do Congresso* Report. Afterwards, those two datasets were merged.

Also, since many of the voting congressmen are surrogates and participate only in a small fraction of the voting sessions, a minimum attendance threshold was established in order to filter the congressmen whose attendance is below this given parameter.

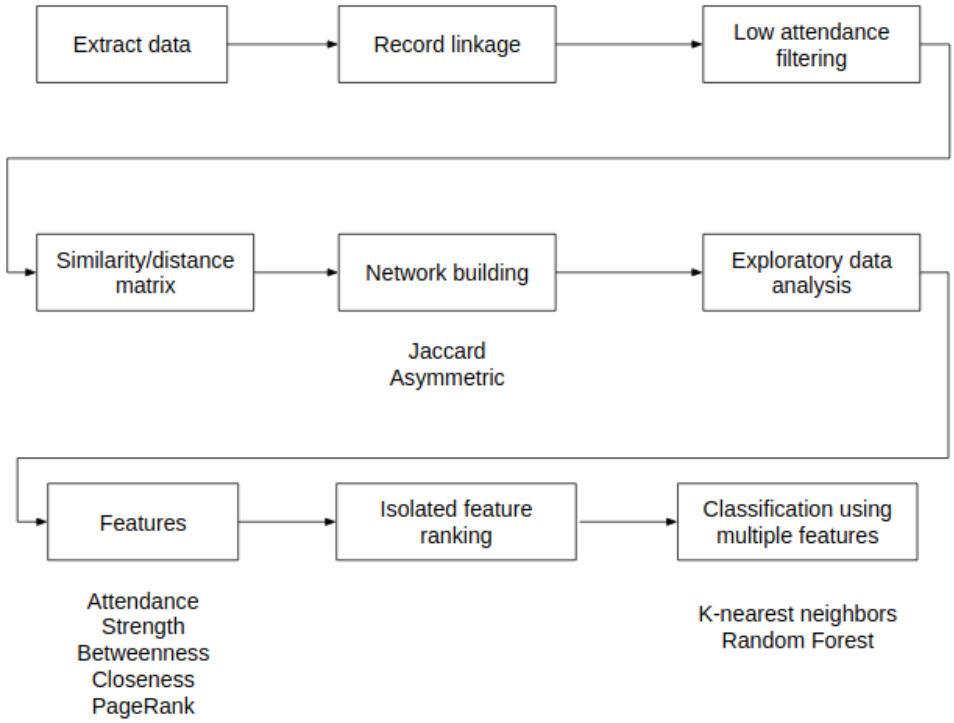

Figure 11: Tasks involved in the proposed methodology.

Afterwards, local relationships between congressmen were modeled using a co-votation network, which is a similarity network composed of vertices, which are congressmen, and edges, that reflects pairwise similarity among them regarding their voting behavior. While doing this, two similarity functions were experimented on this work: the original Jaccard function and a modified asymmetric Jaccard function. Jaccard was chosen since it is one of the commonly used similarity function in social network. On the other hand, its asymmetric modification, which was defined in the theoretical framework, was chosen since it can represent asymmetric relationships and lead to the creation of a directional network. Edge weights were also defined as being pairwise distance among congressmen, since, as opposed to using similarity since this formulation is required in some centrality definitions.

Since similarity was compute using those two metrics, this leads to the creation of two different weighted networks. The adirectional network built using the original Jaccard function will be called the Jaccard network. The directional network

built using the modified Jaccard function will be called the asymmetric network. I preferred not to call the Jaccard network as “symmetric network” in order to avoid confusion. Afterwards, I did an exploratory data analysis to describe those networks in terms of network metrics, such as: number of vertices, number of edges, maximum strength, average strength, minimum strength and average local clustering index. I was also interested in the observation of the distribution of voting session attendance and voting outcome.

In the voting dataset, each data category has a number of features, each one representing a different voting session. Each voting session has five possible labels: yes, no, abstention, no-show, and obstruction. It is important to consider no-shows differently since it conveys little information about the position of a congressman regarding a legal project. Even more important, the great number of no-show can scientifically degrade pairwise similarity among congressmen.

For example, suppose two congressmen registered as no-show in 80 voting session each across an entire year. Also those same congressmen attended 20 sessions, on which they registered completely opposite votes. If one naively considered all labels equally, such congressmen would have behaved similarly in 80% of the voting sessions and would be considered highly similar, despite voting in opposite directions when they actually casted their votes. Therefore, voting sessions where both congressmen fit within the no-show category will not be considered.

4.2.2 Identifying influential congressmen

Afterwards, I can address to the question of whether topological characteristics of a network can often be used to identify the most influential vertices. I am attempting to predict the presence or absence of a congressman in DIAP list while abstaining from using deeper information contained in the report, such as the type of influence exerted by each congressman.

I attempted to predict the most influential congressmen by using two approaches. First, I consider rankings of vertices sorted by different centrality metrics that reflect different concepts of centrality. Later, I will experiment a more elaborated approach by combining different features as input for classification algorithms.

4.2.3 Features

In both approaches, the main hypothesis is that congressmen that are influential should have high centrality features. Diverse of centrality metrics were considered: strength, betweenness, closeness, and PageRank. On directional networks, the strength feature will be replaced by two features: in-strength and out-strength. Also, on networks without directions, PageRank will not be used since it is designed to be used only on directional networks. Finally, in addition to those topological features, I also included congressmen attendance to voting sessions.

Those topological features were chosen not only because they are the most common ones, but mainly because they capture different conceptual definitions of centrality. While strength is a centrality metric based on strictly local abundance of connections, PageRank are based on recursive abundance of connections. On the other hand, closeness is based on network wide distance, and betweenness is based on network wide message flow.

Betweenness and closeness metrics were normalized in to make them easier to compare between different networks. Also, PageRank requires inputting a damping coefficient, which was set value as 0.85 since this value is the one commonly used. In addition to those features previously mentioned, another features not directly related do centrality was added: congressmen attendance to voting sessions.

Finally, since all those features are continuous in their nature, they can be inputted into models without further transformation. Still, they were standardized since some models such as k-nearest neighbors work better if their different inputs have a similar dispersion range.

4.2.4 Learning algorithms

Afterwards, I used different learning algorithms to verify which one would better fit this problem. Two algorithms were attempted: random forest (RF) and k-Nearest Neighbors (kNN). Those algorithms reflect different lines of thoughts that could provide a reasonable coverage. K-Nearest Neighbors is non-parametric algorithm that requires few hypothesis above the dataset, while random forest is a semi-parametric algorithm known to provide good performance while requiring little hyper-parameter turning.

The optimum values for the hyper-parameters on those algorithms were found

using cross-validation, i.e. several models were be built with different hyper-parameters and the best one was selected after its performance on a dataset with previously unseen data instances.

The random forests algorithm has many parameters, but I am only turning the most important ones, the maximum depth of each decision tree, and the number of base decision trees. The maximum depth parameter was experimented within the range [3, 5, 7], while the number of trees parameter was experimented within the range [3, 5, 10, 20]. Usually, the deeper are decision trees, the more complexity are their models. A high number of trees can smooth the decision function, avoiding overfitting. I chose to use the Gini index as the split quality function, and to subsample in each node the squared root of the original number of features available as it is the default option on SciKit-Learn.

On the k-Nearest Neighbor Algorithm, there are three design decisions: k, the number of neighbors considered in the model, the distance function, and the weight of each neighbor. I decided to use Euclidian distance function, and to experiment the parameter k within the range [5, 15, 40, 60]. Small values of k tend to overfit the models, while large values of k may lead to underfitting. Also, I will experiment weighting data instances in two ways: uniformly and in inverse proportion to their distance.

4.2.5 Evaluation

The data was split as follows: 75% of the data instances into the training dataset, while other 25% went into the testing dataset. After splitting them, all variables were standardized with a scale built using only data from the training dataset. Within the training set, model selection was done using k-folding with k=10.

While evaluating models, some evaluation metric had to be used to determine the best one. Due to the class imbalanced of this problem, I decided not to use accuracy. Precision or recall could also be used, but I favored using the area under the receiver operating characteristic curves (AUROC). This decision was done since precision and recall only considers the score around a specific threshold, missing information about the performance if other thresholds were selected.

Additionally, it was desirable a metric that would allow comparing this result with a benchmark. Since there is no previous work investigating on this research

question, the benchmark was a naïve model consisting of randomly considering a fraction of the deputies as being influential. Given that the House of Representatives has 513 members and 53 of them are considered influential, this naïve model randomly has average precision 0.103.

5 RESULTS

This chapter presents the results obtained, interprets and discusses them. It is divided into three subsections: (1) Data transformation and exploratory data analysis; (3) Prediction of influential congressmen using ranking of different network centralities; (4) Prediction of influential congressmen using multiple centralities as features for classification algorithms.

5.1 Data Transformation and exploratory data analysis

Once the results from the House of Deputies voting sessions and the DIAP report were obtained, they were merged. This process was largely manual and suffered from two main problems. First, there was not a unique identifier for congressmen. Also, several congressmen have different register names in the House of Representatives and in the DIAP report. Still, I was able to link them by using a mixture of information such as congressman name, political party affiliation and representing region.

Secondly, and more important, several congressmen left their positions. For example, several ministers and secretaries on regional-level are also members of the national congress. Brazilian electoral law covers this case by requiring the candidate congressman to present an ordered list of three surrogate congressman, to be progressively called according to the necessity. Therefore, voting records contain several congressmen that are surrogates. Consequentially, the Voting API registers 601 unique congressmen names across all voting sessions in 2015, more than the 513 seats available within the House of congress.

After creating a coherent dataset, the first point of interest was the distribution of congressmen attendance across all voting sections. The result can be seen on Figure 12.

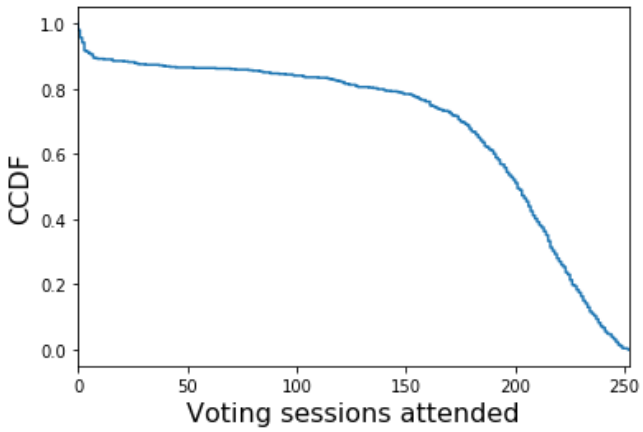

Figure 12: Complementary cumulative distribution function of congressman attendance to voting sessions.

Results show that the distribution is clearly non-normal. Around 70% of them have attended more than 66% of the voting sessions, however, some congressmen have very low voting attendance rate. The most likely explanation for the majority these extreme cases is that these congressmen are surrogates. Since this surrogates joined the Congress later and often only during a short stay, they tend to have attended fewer voting sessions.

Additionally, surrogate congressmen are often not considered while elaborating DIAP report. Therefore, to restrict the many problems caused by eventual surrogates and elected congressman who seldom participate in the congress, I opted to consider only congressmen who have attended to a minimum of 50% of the voting sections, i.e. 125 sessions. This criterion ended up excluding 115 congressmen, of which only one was considered influential.[3]

Additionally, I also evaluated how was the distribution of voting outcome during the sessions as shown in Figure 13.

3 According to the internal regulation, the President of the House of Representative is unable to vote on most sessions. Consequentially, he only participated on one voting session, the one which he got himself elected.

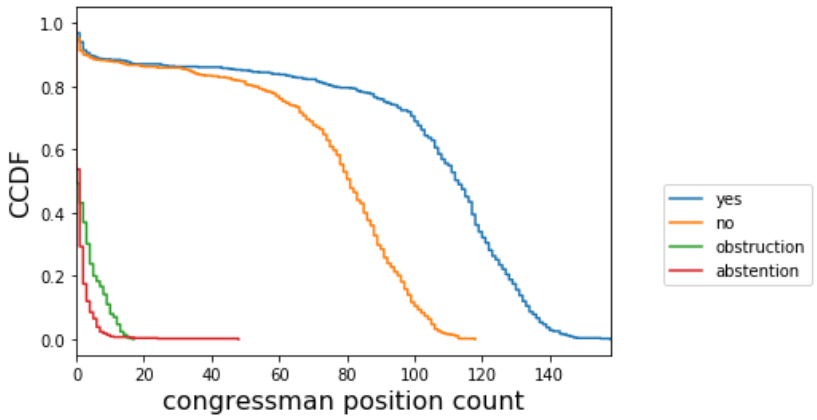

Figure 13: Complementary cumulative distribution function of congressman by voting option.

It clear that there is a trend towards approving legal projects since congressmen are more susceptible to vote yes than no. Absences are extremely rare, and a significant fraction of the congressmen has never taken this position. Finally, obstructions are also a rare outcome, but a surprisingly present position given how unknown is its existence among general public. As a final remark, there are possible interpretations of abstention and obstruction. The first means keeping a neutral position while helping the proponents of the legal project by providing the minimum quorum to conduct the voting section. The latter is limited to party leaders willing to postpone an ongoing voting section.

Afterwards, networks were built to model local relationships among congressmen. In this network, vertices are congressmen and edges weights reflect pairwise similarity or distance among congressmen regarding their voting behavior. Each congressman has a set of legal projects on which he has voted and intersections of common positions between congressmen are evaluated. Each edge actually has two weights, one of pairwise similarity between congressmen and another one of the opposite, i.e. pairwise distance between congressmen.

In this analysis, two weighted networks were built: A symmetric network built using the Jaccard similarity function, which will be called Jaccard network; and an asymmetric network built using a modified version of the Jaccard similarity function, which will be called asymmetric network.[4] This modified version was chosen since it allows representing the asymmetric relationships between congressmen, leading to the creation of a directional network. In the context of weights as similarity, the higher the

4 I preferred not call the Jaccard network as “symmetric network”, in order to avoid confusion.

share of common positions between two congressmen, the higher is their similarity. The opposite happens when weights are defined as distances. I developed both weights since some features need edges to be defined as similarity, e.g. strength and PageRank, while other require them to be defined as distance, e.g. betweenness, closeness.

The distribution of strength across those two networks can be seen in on complementary cumulative distribution functions on Figure 14. The axis x is congressmen strength and the axis y is congressmen count. On both networks the distributions are very similar, a bell shape curve.

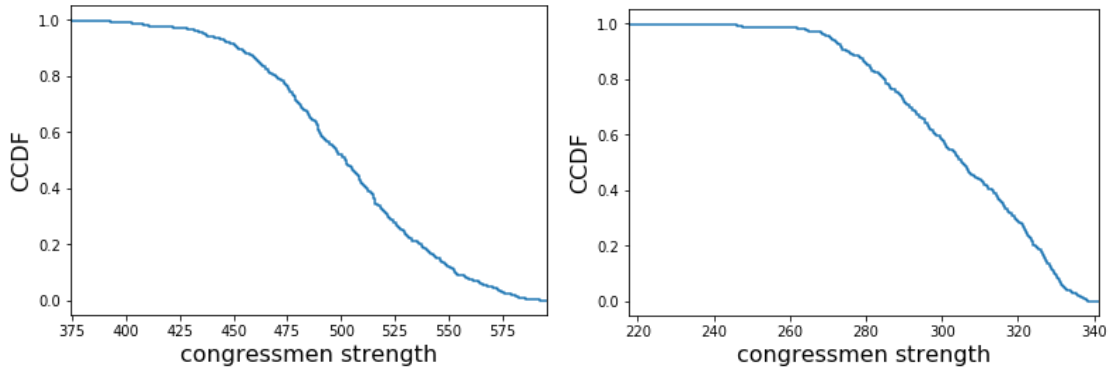

Figure 14: Complementary cumulative distribution functions for total strength on the asymmetric network (left) and strength on the Jaccard network(right).

The overall connectivity of the resulting networks can be better understood by observing summarization metrics shown in Table 3 below for both networks built. Those metrics indicate that both networks are reasonably dense.

Table 3: Networks summary metrics

	asymmetric	Jaccard
Number of vertices	487	487
Number of edges	236682	118341
Maximum (total) strength	595.61	341.18
Average (total) strength	501.88	304.04
Minimum (total) strength	374.28	217.68
Density	0.52	0.62
Average local clustering index	0.61	0.58

Since the asymmetric network is directional, it has twice the number of edges the Jaccard network has. The high connectivity in the voting networks suggests that it is very unlikely for a congressman not to find a minimum degree of consensus with his peers. This happens because congressmen voting options are intrinsically limited: they must declare a position when participating on a voting section, which is limited to only four options (“yes”, “no”, “abstention” or “obstruction”). However, voting in such limited range of options imposes a very low probability for not sharing at least one single common vote across all voting sessions. The reasonably high values observed clustering coefficient can also be explained from this observation.

Additionally, unlike the Jaccard network, the relationships in the asymmetric network are not fully mutual. It is important to know the level of symmetry of those relations. This was done by computing the Pearson correlation between every pair of edge connecting from a congressman A to a congressman B with their corresponding edge connecting from congressman B to congressman A. As expected, the resulting Pearson correlation, 0.75, is high.

Finally, I was particularly interested into seen histograms and scatter plots that may reveal how features interact with each other. Figure 15 shows a mosaic of histograms in the main diagonal and pairwise scatter plots between features in the asymmetric network. A summary of the correlation among features was calculated using Pearson correlation and can be seen on Table 4.

Table 4: Correlation matrix for features on asymmetric network.

	Attendance	In_strength	Out_strength	Betweenness	Closeness	PageRank
Attendance	1.00	-0.84	0.33	0.11	-0.33	-0.83
In_strength	-0.84	1.00	0.24	-0.11	-0.23	1.00
Out_strength	0.33	0.24	1.00	0.00	-1.00	0.26
Betweenness	0.11	-0.11	0.00	1.00	-0.01	-0.11
Closeness	-0.33	-0.23	-1.00	-0.01	1.00	-0.25
PageRank	-0.83	1.00	0.26	-0.11	-0.25	1.00

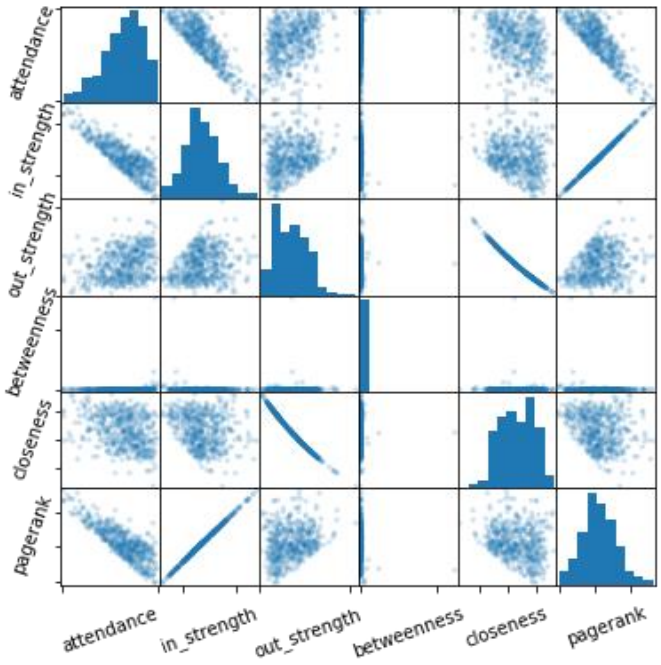

Figure 15: Mosaic of histograms and scatter plots for features on asymmetric network

Regarding the asymmetric network, the first observation to be made is that there is a strong correlation between the In-strength and PageRank. This is expected since, for undirected networks or directed networks with many mutual connections, the PageRank of vertices tend to be proportional to their degrees as was pointed out by Grolmusz(2012). More surprising, there is also a significant correlation between Out-strength and closeness. Regarding other pair of features, their correlations range from weak to modest. This is a useful since uncorrelated features may encode additional information.

Betweenness was particularly uncorrelated and with a distribution highly concentrated around zero, with few congressmen having value higher than zero. A possible reason for that is that some congressmen may have a tendency to vote with the majority of the House of Representatives, and/or register non-show on more disputed voting sessions. Since the co-votation networks is close to a complete network, two edge hops passing thru those few congressmen can be a shorter path than travelling along the edge directly connecting congressmen. Consequentially, those congressmen concentrated a disproportional share of the shortest paths in the network and, thus, concentrate much of network betweenness.

The same analysis was done in the Jaccard network. Its mosaic of histograms and pairwise scatter plots can be seen on Figure 16 and Pearson correlations between its pairs of features can be seen on Table 5.

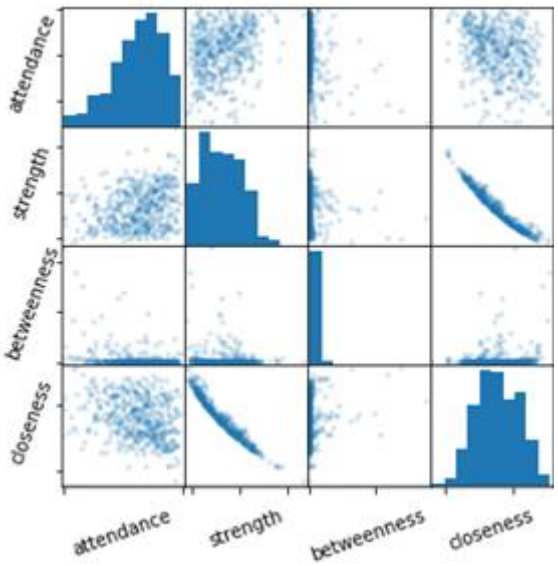

Figure 16: Mosaic of histograms and scatter plots for features on Jaccard network

Table 5: Correlation matrix for features on Jaccard network.

	Attendance	Strength	Betweenness	Closeness
Attendance	1.00	0.25	-0.29	-0.29
Strength	0.25	1.00	-0.08	-0.97
Betweenness	-0.29	-0.08	1.00	0.14
Closeness	-0.29	-0.97	0.14	1.00

Unlike in the asymmetric network, there is very high correlation among every feature pair other than the ones involving attendance and betweenness. This level of correlation among these features suggests that there is a significant redundancy among them. Although this result was surprising, further research into network science literature corroborated it. Li et al. (2015), Meghanathan (2015), Landherr et al. (2010) and Valente et al. (2008)investigated correlation among topological features and they all found that there it is possible to have significant level of correlation among centralities, despite them having very different theoretical foundations and underlying centrality concepts.

The work of Valente et al. (2008) suggests a reason why the Jaccard network suffered this problem more acutely than the asymmetric network. In addition to highlighting the how correlated were nine different centrality features across several real-world networks, that work also explored the association between centrality correlation and four different network properties - density, reciprocity, centralization and number of components.

It was found that certain conditions favor a very high level of correlations among topological metrics. Those conditions arise specially on networks with high level of reciprocity, i.e. where relationship between nodes is highly mutual. Since Jaccard similarity function resulted in non-directional network, it has complete reciprocity on every relationship. Despite asymmetric network be a directional network, it also a significant level of reciprocity in its relationships as was indicated by the Pearson correlation between edges connecting the same pair of congressmen in opposite directions.

5.2 Predicting influential congressmen

Afterwards, I addressed to the question of whether topological characteristics of a co-votation network can be used to identify most influential nodes. The *Cabeças do Congresso* by Queiroz, 2015)contains 100 influential congressmen, but only 54 of them are deputies and one of them did not vote since he was the president of the House of Deputies. Influential congressmen were defined by considering their presence or absence in the list and abstained from using deeper information contained in the report, such as what type of influence it exerted by each congressman.

I predicted influential congressmen by using two approaches. The first one was considering each topological centrality metric in isolation. Later, those same features were combined and used as input features for two classification algorithms. The performance those models were evaluated by two metrics, average precision along the precision-recall curve and area under the receiver operating characteristic curve.

Additionally, I wanted to compare this result with a benchmark. Since there is no previous work investigating on this research question, the benchmark was results excepted if influential deputies were randomly chosen. Since, the House of Deputies has 513 members and 53 of them are considered influential, this naïve model randomly selects 10.3% of deputies as influential.

5.2.1 Isolated feature ranking

I ranked congressmen in descending order for each feature built in the previous section – attendance, strength, betweenness, closeness and PageRank. Afterwards these ranking were compared with the ground truth.

Result obtained as measured by the average precision along the precision-recall curve is shown on Table 6.

Table 6: Average precision along the precision-recall curve using feature ranking.

	Asymmetric Network	Jaccard Network
Attendance	0.10	0.10
Strength	-	0.22
In-strength	0.19	-
Out-strength	0.22	-
Betweenness	0.11	0.15
Closeness	0.08	0.08
PageRank	0.19	-

Overall, both two networks performed similarly while comparing their top performing features. Since the benchmark would lead to an average precision equal to 0.103, many features performed better than what would expect by random guessing. At their best features - strength on Jaccard network or out-strength in the asymmetric network - the average precision was 0.22, more than twice the result obtained by the benchmark.

Interestingly, unlike the case of Florecian families in Padgett & Ansell (1993), betweenness centrality did not perform well. A probable reason for that is that the completeness of the co-votation network degenerated this metric by concentrated a disproportional share of the shortest paths in the network within a few congressmen, while leaving most of them with zero betweeness.

Results measured by the area under the receiver operating characteristic can respectively be seen on Table 7. Similarly to the average precision, the best performing feature were strength on Jaccard network or out-strength in the asymmetric network.

Overall, the results obtained by Jaccard network as slightly higher than those obtained by the asymmetric network.

Those results can be evaluated was according to the classification of level of discriminatory ability proposed by Hosmer & Lemeshow (2004).The top performing features, strength on Jaccard network would be considered to have an acceptable level of discriminatory ability.

Table 7: Area under the receiver operating characteristic curve using feature ranking.

	Asymmetric Network	Jaccard Network
Attendance	0.45	0.45
Strength	-	0.71
In-strength	0.68	-
Out-strength	0.69	-
Betweenness	0.49	0.55
Closeness	0.31	0.33
PageRank	0.68	-

A further understanding of the performance of this ranking can be seen by evaluating the shape of their precision-recall curves which are respectively shown in Figure 19 and Figure 20 on the asymmetric and Jaccard networks. In these figures, the expected precision obtained by the naïve benchmark is indicated by horizontal dashed lines at 0.103.

On the asymmetric network, it can be seen that the shape of in-strength and PageRank curves are similar. Again, this is expected since on directed networks with many mutual connections, the PageRank of vertices tend to be proportional to their degrees as noted in Grolmusz (2012).

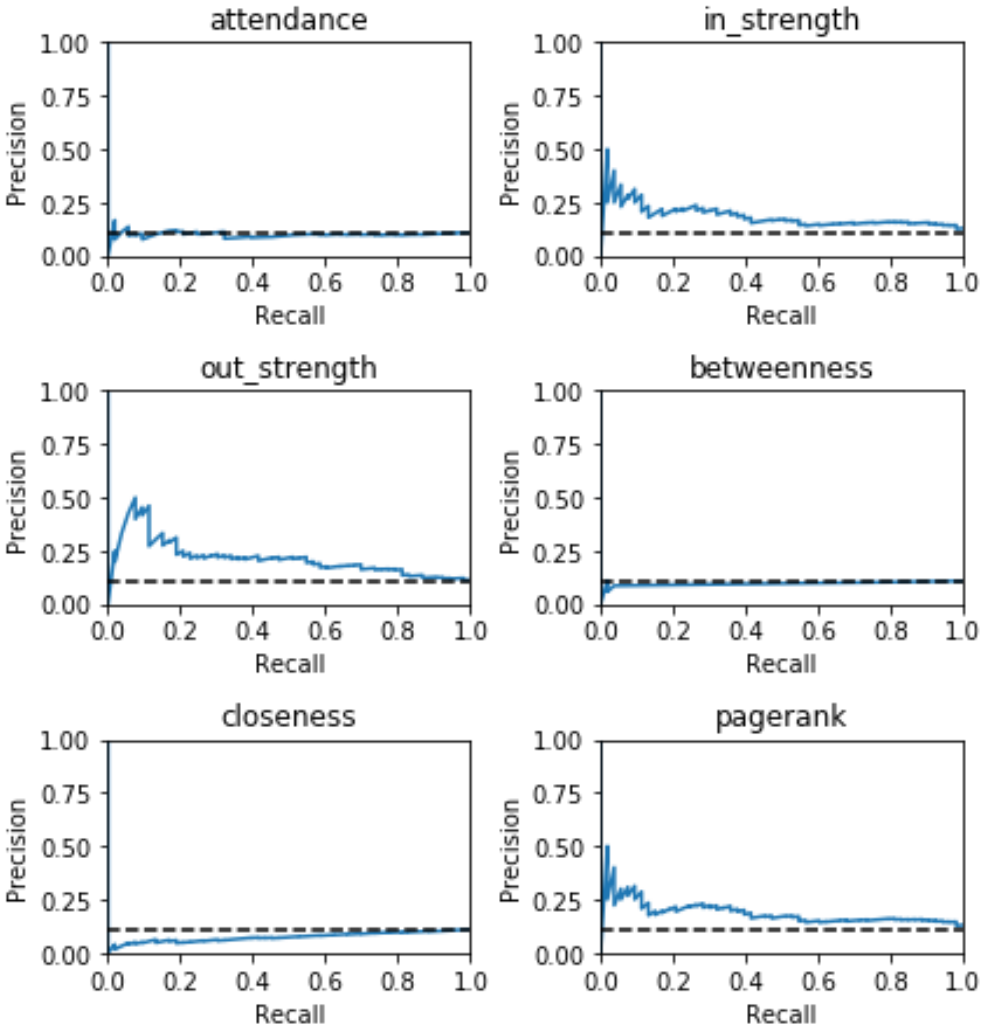

Figure 17: Precision-recall curve for prediction using feature ranking on the asymmetric network.

Those two features were able to identify congressman with a precision of roughly 20%, significantly better than the naïve benchmark, up to recalling 40% of influential congressman. After this threshold, precision still remains above the benchmark, but not in a significant way. The curve for the out-strength feature had an even better result, but which precision still limited to 25% for most of the curve until recalling 60% of congressmen. Attendance and betweenness did not differ from naïve benchmark, while closeness is seen below the benchmark.

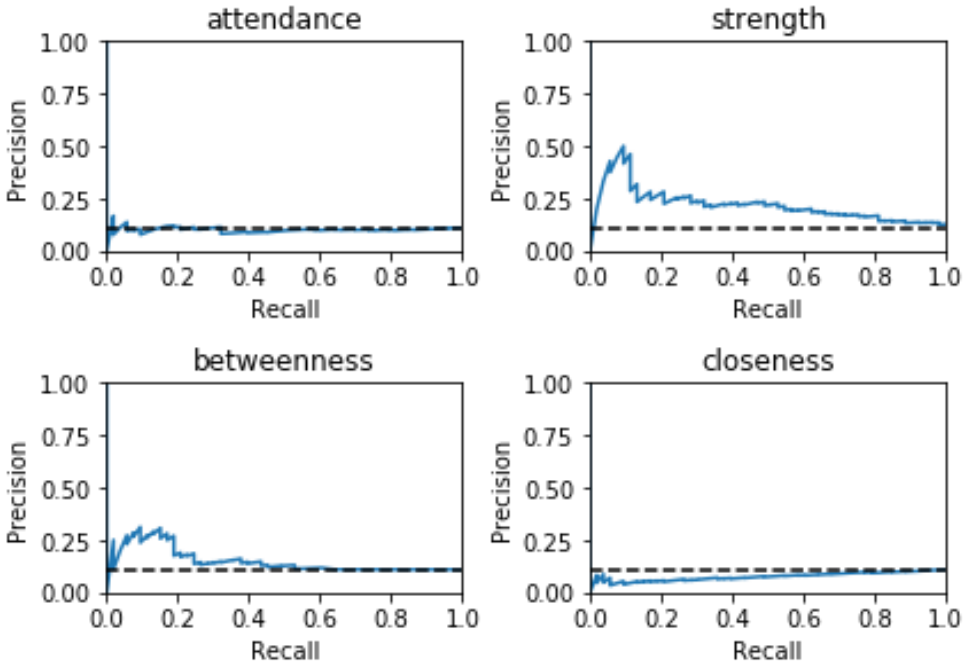

Figure 18: Precision-recall curve for prediction using feature ranking on Jaccard network.

On the Jaccard network, the strength metric was able to identify congressman slightly better by having a precision of roughly 25% upon recalling 50% of influential congressman. Again, closeness is seen below the benchmark while betweenness performance better than the naïve benchmark, but only during a short range.

Finally, the receiver operating characteristic curve for those ranking models on the asymmetric and Jaccard networks can respectively be seen on Figure 19 and Figure 20. In these figures, a classifier with no discriminatory ability would score along the diagonal dashed line.

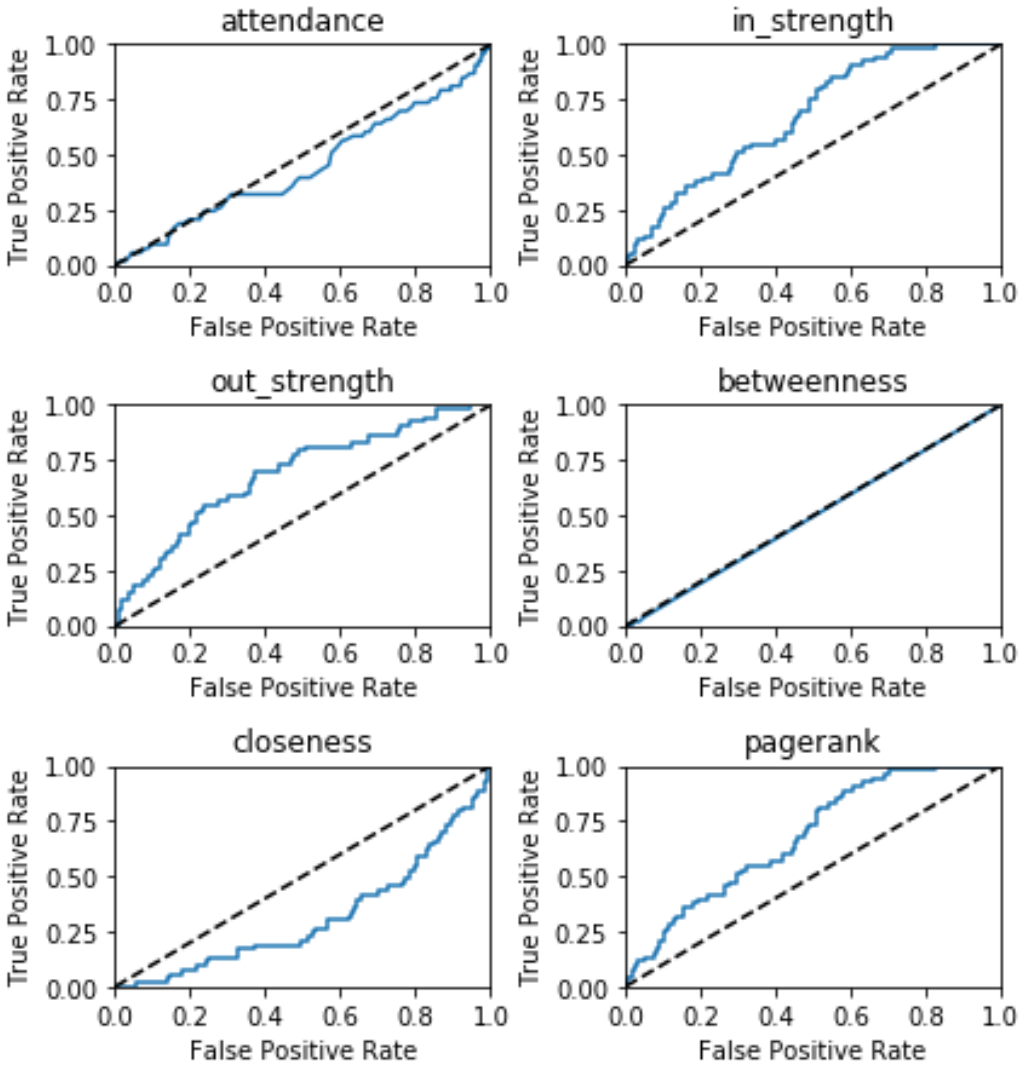

Figure 19: Receiver operating characteristic curve on the asymmetric network.

Looking at the receiver operating characteristic curve on both networks, the behavior of the closeness rank can be better understood. Since its curve is consistently below the dashed line, it seems like it is performing worse what would be expected by random chance. This suggests that, against what was expected by common sense, closeness ranking should be sorted not in descending order, but in ascending order.

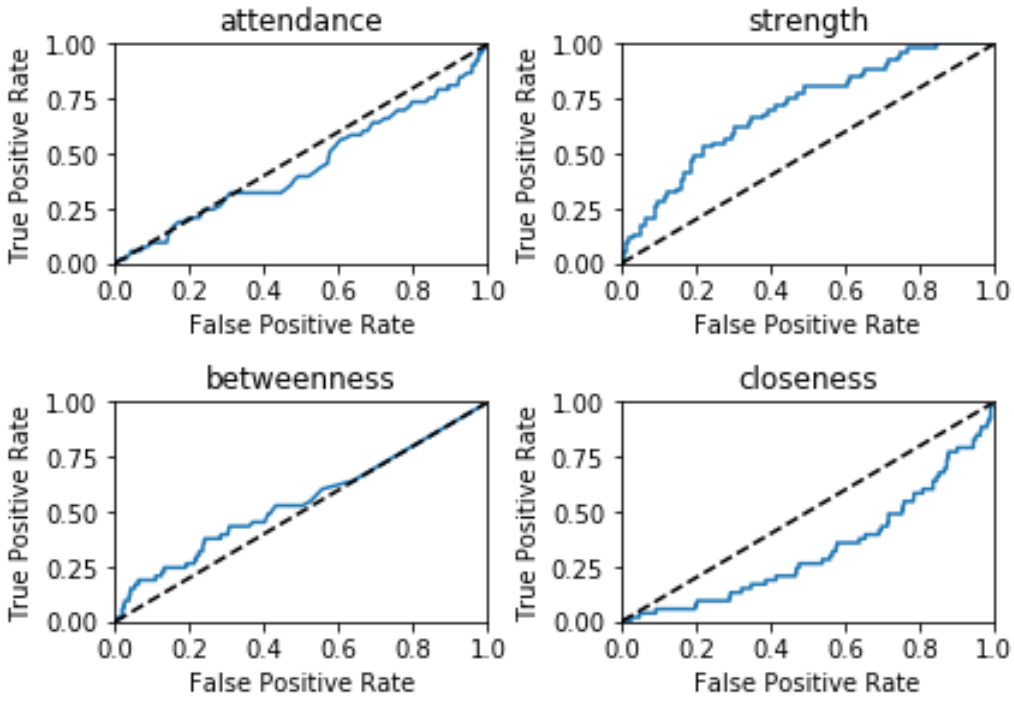

Figure 20: Receiver operating characteristic curve on Jaccard network.

5.2.2 Combining features

After evaluating the ability of each feature to identify leaders when considered in isolation, I now investigate whether it is possible to improve results by combing these features together and using them as input features feeding classification algorithms.

Since features on Jaccard network are highly correlated among themselves, they are mostly redundant and combining them would not provide much additional information. Therefore, I decided to focus this session by applying this approach only to the asymmetric network.

As mentioned in the methodology section, I used two learning algorithms to verify which one would better fit this problem. The algorithms attempted were: random forest and k-nearest neighbors. K-Nearest Neighbors was selected for being a non-parametric algorithm that requires few hypotheses above the dataset and performs well with reasonably populated datasets with a small number of dimensions. Random forest was selected since it is a robust semi-parametric algorithm known to provide good performance while requiring little hyper-parameter turning.

Since all features used are continuous in their nature, they were inputted into the model after only a standardization to keep dispersion across different variable on

the same scale. The data used to perform this standardization was obtained only from the training dataset. Hyper-parameters optimization for each model was done using k-folding and selecting the model with the best area under the receiver operating characteristic curve.

For the k-nearest neighbors, cross-validation results can be seen on Table 8.The best performing hyper-parameters was when k=60 and data instances were weighted according to their distance.

Table 8: Cross-validation results on k-nearest neighbors

K	Weigthing	AUROC (95% confidence interval)
5	uniform	0.615 (+/- 0.073)
5	distance	0.620 (+/- 0.063)
15	uniform	0.666 (+/- 0.105)
15	distance	0.689 (+/- 0.110)
40	uniform	0.724 (+/- 0.086)
40	distance	0.734 (+/- 0.091)
60	uniform	0.711 (+/- 0.088)
60	distance	0.742 (+/- 0.092)

For the random forest, cross-validation results can be seen on Table 9. The best performing result was when 20 trees were built with maximum depth of only 3 levels.

Table 9: Cross-validation results on random forest.

Number of trees	Maximum depth	AUROC (95% confidence interval)
3	3	0.691 (+/- 0.048)
5	3	0.692 (+/- 0.046)
10	3	0.703 (+/- 0.059)
20	3	0.714 (+/- 0.066)
3	5	0.579 (+/- 0.123)
5	5	0.586 (+/- 0.111)
10	5	0.632 (+/- 0.081)
20	5	0.657 (+/- 0.053)
3	7	0.518 (+/- 0.110)
5	7	0.530 (+/- 0.122)
10	7	0.553 (+/- 0.135)
20	7	0.614 (+/- 0.097)

Afterwards, the selected models were then applied to the test dataset in order to evaluate their generalization capability. Those results can be seen on Table 10.

Table 10: Performance metrics on testing dataset.

	K-nearest neighbors	Random forest
Average precision	0.28	0.30
AUROC	0.72	0.73

In both metrics evaluated, the best result was obtained by the random forest model, but closely followed by the k-nearest neighbors. The average precision obtained is 0.30, almost 3 times better than results from random guessing, 0.103. Also, this result is 36% better than the model previously built – base on congressman ranking according to its PageRank metric - which scored an average precision equal to 0.22.

Regarding the area under receiver operating characteristic curve, the performance also improved over isolated ranking features based on the asymmetric network. The result obtained, 0.73, would be considered an acceptable level of discriminatory ability in the classification proposed by Hosmer & Lemeshow (2004).

Similar to when evaluating ranking on the previous section, I was interested into view the precision-recall curve since it can lead to better understanding model performance considering all possible values classification threshold. Additionally, since those models provide individual scores for each congressman, verifying the histogram of these scores can also be insightful. While evaluating this, I will be focusing on analyzing the best performing model, the random forest.

The histogram of score assigned by the model that a given congressmen can be seen below on Figure 21.

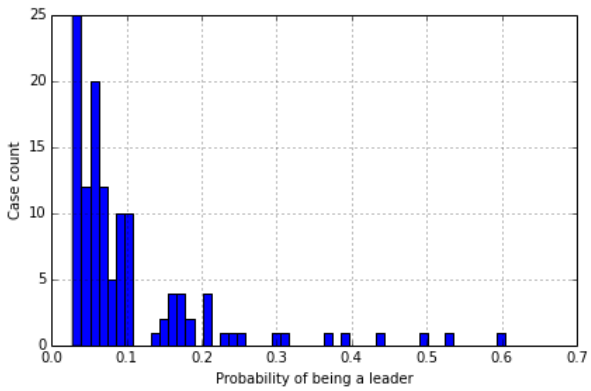

Figure 21: Histogram for model confidence that a given congressmen in the testing dataset is influential.

Most congressmen have very small scores, indicating the model have high level of confidence that they are not influential. Therefore, few congressmen would be classified as influential for as threshold in the decision function above 0.1. The way how precision and recall is affected as this threshold varies can be seen on the precision-recall curves are shown on Figure 22.

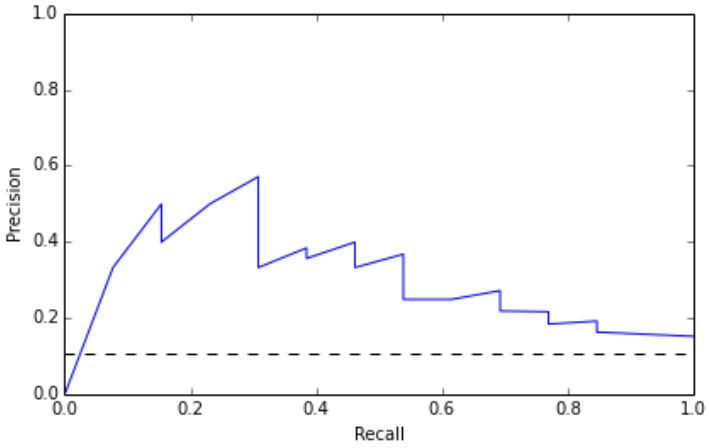

Figure 22: Precision-recall curve for results in the testing dataset.

There is clear tradeoff between precision and recall in the curve. Models with a high score threshold would flag few congressmen as leaders, keeping a relatively high precision, while having a low recall. As the threshold value decreases, the number of congressmen flagged as influential increases, increasing the recall at the expense of precision. It was relativity easy to identify approximately 50% of the leaders, while keeping precision higher than 0.30. However, identifying additional leaders remained a difficult task which could not be done without reducing the precision metric.

Finally, the receiver operating characteristic curve can be seen on Figure 23. In comparison with ROC curves obtained by isolated feature ranking, one can see that this curve not only has greater area, but has more it is also more regular.

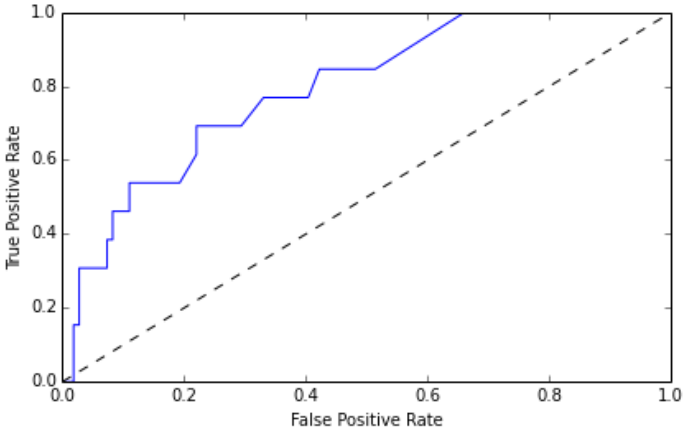

Figure 23: Receiver operating characteristic curve for results in the testing dataset.

5.2.3 Final remarks

Overall, combining centralities only marginally improved the ability to detect influencers. As was previously discussed, while different centrality features can be conceptually very different, in network with high level of reciprocity, they are prone to be highly correlated, leading to significant level of redundancy among them. While this was clearly the case in the Jaccard network, this may also have contributed to small gains observed while combining features in the asymmetric network.

Additionally, approaching the problem with classification algorithms exposed here has the major disadvantage of requiring the availability of a source ground truth data in order to train the model.

Further work investigating this topic would benefit by enriching the feature set with additional relevant features. This could be done by adding information regarding political party affiliation or participation in political coalitions. While political party affiliation is public information, participation in a political coalition can be subtle information that may require expert knowledge. Nevertheless, this information could be included not using explicit ground truth sources but implicitly using community detections techniques.

Regarding this, it would be useful to adapt some of the concepts presented by Dal Maso et al. (2014). This work divided congressmen within two groups, roughly

defined as government and opposition. Then, it computed the importance of a deputy within a group by how much modularity the network would lose if that deputy diverged from its home community and defected to other community. Based on this metric, they characterize the heterogeneity of government coalitions and accounted each political party contribution to the stability of the Italian government over time.

Another way to incorporate political parties as features would be to compute the importance of the party within the context of the house of deputies. In order to achieve this, one could apply some of the techniques developed by Takes & Heemskerk (2016), whose work partitioned a network of companies according to their nationality and developed metrics to a measure how central each partition was. This framework could be applied to the domain of legislative politics by considering political parties as partitions of the network, which allows using party influence as another feature that can help identify influential congressmen.

6 CONCLUSION

This work investigated whether it is possible to identify influential congressmen solely by using information from the results of legislative voting sessions. To the best of my knowledge, this is the first attempt within literature to address this research question. The problem was approach by two theoretical frameworks: machine learning and complex networks. First, a co-votation network encoding similarity among congress members regarding their voting outcomes was built. Afterward, topological metrics for each individual in the network were calculated using different concepts of centrality. Those metrics were used in an isolated manner to predict influential congressman based on its ranking. Afterwards, those topological features were combined as input features in for classification algorithms.

Ranking centralities individually was able to identify influential congress significantly better that what would be expected by random chance. Ranking built from Out-strength had the best results, 0.22, as measured by their average precision along precision-recall curve. This result is twice better than the results expected by random chance, 0.103. After combining those different centralities as input features in classification algorithms, the average precision along the precision-recall curve was further increased to 0.30, almost three times better than the naïve benchmark. Nevertheless, the improvement when combining features was not as significant as expected since there is a high level of redundancy among them. The work of Valente et al. (2008) suggests that this was due to the high level of reciprocity on relationships within the co-votation network. Under this condition, centrality metrics became highly correlated, despite having very different theoretical foundations and underlying centrality concepts.

Nevertheless, those results obtained are substantially. This suggests that information regarding congressmen influences is encoded into parliament voting results. Indeed, partial results of this work had already been publish in Bursztyn et al. (2016).

Finally, although this work focused on the context of the Brazilian House of Representatives during the year of 2015, it also raises the question of whether similar results could be observed in other legislative houses, such as parliaments of smaller countries or subnational parliaments.

6.1 Future work

There are some limitations present in this job. First, only deputies elected in 2014 were evaluated. It would be fundamental to corroborate findings if the same methodology were applied to deputies elected in 2010 or in previous elections. This longitudinal study could also observe how network topology varies across different legislative periods.

Additionally, the definition of influential congressional representative adopted in this work is binary, restricted to provide congressmen information with only two labels, influential or not. It would be valuable to have an ordinal ranking of influential leaders. A possible way to create this would be to merge the list obtained from *Cabeças do Congresso* report, by Queiroz (2015), with other datasets that also signal congressman influence, such as leadership position within thematic subcommittees in the House of Representatives.

Finally, results could be further improved by enriching the dataset with additional features. This could be done by adding information regarding political party affiliation or participation in political coalitions. While political party affiliation is public information, participation in a political coalition can be subtle information that may require expert knowledge. Nevertheless, this information could be included not using explicit ground truth sources but implicitly using community detections techniques.

An example such techniques can be seen in the work by Dal Maso et al. (2014) which used of community detection techniques on the Italian parliament. That work computed the importance of deputies within their community by how much modularity the network would lose if that deputy diverged from its home community and defected to another community. This and many other additional features could be created by encoding affiliation to political parties or political coalitions using other ideas from network science.

REFERENCES

BAPTISTA, V. M. P. DE S.;. Um modelo para a detecção das mudanças de posicionamento dos deputados federais. , 2015. Universidade Federal da Paraiba.

BARABÁSI, A.-L.; ALBERT, R. Emergence of Scaling in Random Networks. **Science**, v. 286, n. October, p. 509–512, 1999.

BARRAT, A.; BARTHÉLEMY, M.; PASTOR-SATORRAS, R.; VESPIGNANI, A. The architecture of complex weighted networks. **Proceedings of the National Academy of Sciences of the United States of America**, v. 101, n. 11, p. 3747–3752, 2004.

BISHOP, C. M. **Pattern Recognition and Machine Learning**. 2006.

BONACICH, P. Power and Centrality : A Family of Measures. **American Journal of Sociology**, v. 92, n. 5, p. 1170–1182, 1987.

BRASIL, C. DOS D. DO. Dados Abertos - Legislativo. Disponível em: <http://www.camara.leg.br/SitCamaraWS/Proposicoes.asmx>. .

BREIMAN, L. Bagging predictors. **Machine Learning**, v. 24, n. 2, p. 123–140, 1996. Disponível em: <http://link.springer.com/10.1007/BF00058655>. .

BREIMAN, L. Using iterated bagging to debias regressions. **Machine Learning**, v. 45, n. 3, p. 261–277, 2001.

BREIMAN, L.; FRIEDMAN, J. H.; OLSHEN, R. A.; STONE, C. J. **Classification and Regression Trees**. 1984.

BURSZTYN, V. S.; NUNES, M. G.; FIGUEIREDO, D. R. How congressmen connect: analyzing voting and donation networks in the brazilian congress. 5th Brazilian Workshop on Social Network Analysis and Mining. **Anais...** , 2016.

CHRISTIANO SILVA, T.; ZHAO, L. **Machine Learning in Complex Networks**. Cham: Springer International Publishing, 2016.

DAL MASO, C.; POMPA, G.; PULIGA, M.; RIOTTA, G.; CHESSA, A. Voting behavior, coalitions and government strength through a complex network analysis. **PLoS ONE**, v. 9, n. 12, 2014.

DAVIS, J.; GOADRICH, M. The relationship between Precision-Recall and ROC curves. Proceedings of the 23rd international conference on Machine learning - ICML '06. **Anais...** . p.233–240, 2006. Disponível em:

<http://portal.acm.org/citation.cfm?doid=1143844.1143874>. .

DOMINGOS, P. A few useful things to know about machine learning. **Communications of the ACM**, v. 55, n. 10, p. 78, 2012. Disponível em: <http://dl.acm.org/citation.cfm?doid=2347736.2347755>. .

ETTER, V.; HERZEN, J.; GROSSGLAUSER, M.; THIRAN, P. Mining Democracy. Proceedings of the second edition of the ACM conference on Online social networks - COSN '14. **Anais...** . p.1–12, 2014. Disponível em: <http://dx.doi.org/10.1145/2660460.2660476.%5Cnhttp://dl.acm.org/citation.cfm?id=2660460.2660476%5Cnhttp://infoscience.epfl.ch/record/201674>. .

FREEMAN, L. C. Centrality in social networks conceptual clarification. **Social Networks**, v. 1, n. 3, p. 215–239, 1978.

FREUND, Y.; SCHAPIRE, R. R. E. Experiments with a New Boosting Algorithm. **International Conference on Machine Learning**, p. 148–156, 1996. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.51.6252>. .

FRIEDMAN, J. H. Greedy function approximation: A gradient boosting machine. **Annals of Statistics**, v. 29, n. 5, p. 1189–1232, 2001.

GERRISH, S.; BLEI, D. M. How They Vote: Issue-Adjusted Models of Legislative Behavior. Advances in Neural Information Processing Systems. **Anais...** . p.2753–2761, 2012.

GRANOVETTE, M. S. The strength of weak ties. **The American Journal of Sociology**, v. 78, n. 6, p. 1360–1380, 1973. Disponível em: <http://www.jstor.org/stable/2776392>. .

GROLMUSZ, V. A Note on the PageRank of Undirected Graphs. , 2012.

GU, Y.; SUN, Y.; JIANG, N.; WANG, B.; CHEN, T. Topic-factorized ideal point estimation model for legislative voting network. Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '14. **Anais...** . p.183–192, 2014. Disponível em: <http://dl.acm.org/citation.cfm?doid=2623330.2623700>. .

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics) (9780387848570): Trevor Hastie, Robert Tibshirani, Jerome Friedman: Books. **The elements of statistical learning: dta mining, inference, and prediction**. p.501–520, 2011. Disponível em: <http://www.amazon.com/Elements-Statistical-Learning-

Prediction-Statistics/dp/0387848576/ref=sr_1_14?ie=UTF8&qid=1429565346&sr=8-14&keywords=machine+learning>. .

HOSMER, D. W.; LEMESHOW, S. **Applied Logistic Regression Second Edition**. 2004.

JACCARD, P. Étude comparative de la distribution florale dans une portion des Alpes et des Jura. **Bull Soc Vaudoise Sci Nat**, v. 37, p. 547–579, 1901.

KATZ, L. A new status index derived from sociometric analysis. **Psychometrika**, v. 18, n. 1, p. 39–43, 1953.

KOTSIANTIS, S. B. Decision trees: A recent overview. **Artificial Intelligence Review**, 2013.

KOTSIANTIS, S. B.; ZAHARAKIS, I. D. Supervised machine learning: A review of classification techniques. **{…} applications in computer {…}**, 2007. Disponível em: <http://books.google.com/books?hl=en&lr=&id=vLiTXDHr_sYC&oi=fnd&pg=PA3&dq=supervised+machine+learning+a+review+of+classification+techniques&ots=CWksws2Jlm&sig=UiyYZZfPRkuJiJER78xGdmGB6Po>. .

LANDHERR, A.; FRIEDL, B.; HEIDEMANN, J. A critical review of centrality measures in social networks. **Business & Information Systems Engineering**, v. 2, n. 6, p. 371–385, 2010. Springer.

LAURENT, H. Constructing optimal binary decision tress is NP-complete. **Information Processing Letters**, 1976.

LAZER, D. Networks in Political Science: Back to the Future. **PS: Political Science & Politics**, v. 44, n. 1, p. 61–68, 2011. Disponível em: <http://www.journals.cambridge.org/abstract_S1049096510001873%5Cnhttp://journals.cambridge.org.ubproxy.ub.uni-frankfurt.de/action/displayAbstract?fromPage=online&aid=7968083%5Cnhttp://journals.cambridge.org.ubproxy.ub.uni-frankfurt.de/action/displayFulltext?ty>. .

LI, C.; LI, Q.; MIEGHEM, P. VAN; STANLEY, H. E.; WANG, H. Correlation between centrality metrics and their application to the opinion model. **Eur. Phys. J. B**, v. 88, 2015. Disponível em: <https://link.springer.com/content/pdf/10.1140/epjb/e2015-50671-y.pdf>. Acesso em: 19/11/2017.

LOH, W.-Y. Classification and regression trees. **Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery**, v. 1, n. 1, p. 14–23, 2011. Disponível em: <http://doi.wiley.com/10.1002/widm.8>. .

MACON, K. T.; MUCHA, P. J.; PORTER, M. A. Community structure in the United Nations General Assembly. **Physica A: Statistical Mechanics and its Applications**, v. 391, n. 1–2, p. 343–361, 2012.

MEGHANATHAN, N. Correlation coefficient analysis of centrality metrics for complex network graphs. **Intelligent Systems in Cybernetics and Automation Theory**. p.11–20, 2015. Springer.

MILGRAM, S. An Experimental Study of the Small World Problem. **Source: Sociometry**, v. 32, n. 4, p. 425–443, 1967. Disponível em: <http://www.jstor.org/stable/2786545>. .

MORENO, J. L. **Who Shall Survive? A New Approach to the Problem of Human Interrelations**. 1934.

MORGAN, J. N.; SONQUIST, J. A. Problems in the Analysis of Survey Data, and a Proposal. **Journal of the American Statistical Association**, v. 58, n. 302, p. 415, 1963. Disponível em: <http://www.jstor.org/stable/2283276?origin=crossref>. .

MURPHY, K. P. **Machine Learning: A Probabilistic Perspective**. 2012.

NEWMAN, M. E. J. Analysis of weighted networks. **Physical Review E - Statistical, Nonlinear, and Soft Matter Physics**, v. 70, n. 5 2, 2004.

NEWMAN, M. E. J. **Networks. An introduction**. 2010.

PADGETT, J. F.; ANSELL, C. K. Robust Action and the Rise of the Medici, 1400-1434. **American Journal of Sociology**, v. 98, n. 6, p. 1259–1319, 1993. Disponível em: <http://www.jstor.org/stable/2781822>. .

PENG, T.-Q.; LIU, M.; WU, Y.; LIU, S. Follower-Followee Network, Communication Networks, and Vote Agreement of the U.S. Members of Congress. **Communication Research**, p. 0093650214559601-, 2014. Disponível em: <http://crx.sagepub.com/content/early/2014/12/02/0093650214559601?papetoc>. .

POOLE, K. T. Changing minds? Not in Congress! **Public Choice**, 2007.

PORTER, M. A.; MUCHA, P. J.; NEWMAN, M. E. J.; FRIEND, A. J. Community structure in the United States House of Representatives. **Physica A: Statistical Mechanics and its Applications**, v. 386, n. 1, p. 414–438, 2007.

QUEIROZ, A. **Os Cabeças do Congresso Nacional : uma pesquisa sobre os 100 parlamentares mais influente**. Brasilia, 2015.

QUINLAN, J. R. C4.5:Programs for Machine Learning. **Morgan Kaufmann ,San Mateo**, v. 1, 1993. Disponível em: <http://dl.acm.org/citation.cfm?id=2601107>. .

RUSSELL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach**. 3rd ed. Prentice Hall Press, 2009.

SABIDUSSI, G. The centrality of a graph. **Psychometrika**, v. 31, n. 4, p. 581–603, 1966. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/5232444>. .

SANG, E.; BOS, J. Predicting the 2011 dutch senate election results with twitter. **Proceedings of the Workshop on Semantic Analysis in …**, , n. 53, p. 53–60, 2012. Disponível em: <http://dl.acm.org/citation.cfm?id=2389969.2389976%5Cnhttp://dl.acm.org/citation.cfm?id=2389976>. .

SIM, Y.; ACREE, B.; GROSS, J.; SMITH, N. Measuring ideological proportions in political speeches. **Proceedings of EMNLP**, , n. October, p. 91–101, 2013. Disponível em: <http://hello.yanchuan.sg/assets/papers/sim2013emnlp-slides.pdf>. .

TAKES, F. W.; HEEMSKERK, E. M. Centrality in the Global Network of Corporate Control. , 2016. Disponível em: <http://arxiv.org/abs/1605.08197>. Acesso em: 4/6/2016.

TUMASJAN, A.; SPRENGER, T.; SANDNER, P.; WELPE, I. Predicting elections with Twitter: What 140 characters reveal about political sentiment. **Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media**, p. 178–185, 2010. Disponível em: <http://www.aaai.org/ocs/index.php/ICWSM/ICWSM10/paper/viewFile/1441/1852>. .

VALENTE, T. W.; CORONGES, K.; LAKON, C.; COSTENBADER, E. How Correlated Are Network Centrality Measures? **Connections (Toronto, Ont.)**, v. 28, n. 1, p. 16–26, 2008. NIH Public Access. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/20505784>. Acesso em: 19/11/2017.

WATTS, D. J.; STROGATZ, S. H. Collective dynamics of “small-world” networks. **Nature**, v. 393, n. 6684, p. 440–2, 1998. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/9623998>. .

WAUGH, A. S.; PEI, L.; FOWLER, J. H.; MUCHA, P. J.; PORTER, M. A. Party Polarization in Congress: A Network Science Approach. **arXiv**, p. 1–42, 2009. Disponível em: <http://arxiv.org/abs/0907.3509>. .

ZHANG, Y.; FRIEND, A. J.; TRAUD, A. L.; et al. Community structure in Congressional cosponsorship networks. **Physica A: Statistical Mechanics and its Applications**, v. 387, n. 7, p. 1705–1712, 2008.