Identifying Important Entities in a Protein-Protein Interaction Network using Topological Graph Features

CS224W - Analysis of Networks

Final Report

Vincent Billaut
vbillaut@stanford.edu

Pierre-Louis Cedoz
plcedoz@stanford.edu

Matthieu de Rochemonteix
mderoche@stanford.edu

Introduction

Network biology: In the past decade, there has been an increasing interest to understand the interactions between biological entities. A major focus of network biology consists in building graphs representing the interactions between entities in diverse organisms. In particular, protein-protein interaction networks are crucial for cellular information processing and decision-making. Gaining insight on interactions between proteins is key to understanding the mechanisms that underlie the action of drugs and diseases.

Notion of importance: Drugs are designed to target very particular proteins, and diseases involve a restricted set of proteins. Such proteins are considered "important" because targeting them allows to have an action on the whole organism. A natural question that arises is whether there is a link between the *biological* importance of an entity and the topological configuration of this entity within an interaction graph. An application would be to predict, from the topological features, which proteins are involved in a disease for drug target screening. Numerous papers link the importance of proteins in some diseases to specific topological notions.

Problem Definition: In this project, we aimed at determining which features are predictive or at least significantly correlated with the biological importance of a protein. Our contribution consists in a unified framework to assess and compare the relevance of graph-based features to predict which proteins are likely to be targeted by a disease or relevant to design a drug. We first evaluated each feature independently, comparing the distribution of the feature values for all genes in the network with a set of reference genes. The reference genes correspond to our ground truth for node importance and were extracted from public databases of drug targets and cancer genes (see methods). We then implemented a machine learning classifier to predict whether a node is important or not based on a combination of topological features.

Significance: The contribution of this project was to identify essential genes and proteins in biological networks based on machine learning and network topological features. The identification of such genes is very important not only for understanding the minimal requirements for survival of an organism, but also for finding human disease genes and new drug targets. Our analysis revealed that some intrinsic topological graph features could be used to identify potential drug targets or disease causing genes. We could envision an application of these topological metrics to guide further cancer genomics research or drug screening programs.

Results: We ordered every topological features based on their correlation with a reference set of important proteins: the neighboring conductance, node2vec features, degree centrality and closeness centrality were some of the most significantly correlated. We then experimented with various classification algorithms to predict the importance of a node from all topological features. The best performing was the Gradient Boosted Trees with an accuracy of 91%, precision of 0.49 and recall of 0.56 to recover the genes involved in DrugBank. We also analyzed the importance of every feature in the random forest, which surfaced that features like node2vec and closeness centrality are very useful in identifying novel disease genes and potential drug targets.

1 Related work

A careful study of the state of the art in protein-importance prediction using topological features defined the boundaries and potentially reachable objectives of our project. Several articles could easily represent a whole additional study on their own, and we tried to take the most out of this preliminary exploration in order to be as relevant as possible in our later approach.

[1] very well fits with our will to combine relevant, non-trivial network analysis with insightful and impactful biological discoveries. The authors' goal is to use the notion of *controllability* in network analysis to produce insight into PPI networks. Building upon

the notion of control — in a directed graph, this had to do with being able to produce any *output* through the influence of a reduced set of nodes —, and in particular the minimal number of nodes needed to control a given network, they introduce the notion of *node indispensability* — which quantifies how the number of driver nodes responds to the removal of the node. They find this feature to be highly correlated with several biological importance notions, even after correcting for literature and degree biases. The key aspect that prevented us from trying to reproduce this study and including this feature into our pipeline was that our PPI network turned out to be undirected.

In [2], the authors summarize state-of-the-art computational methods for identifying essential genes and proteins in biological networks based on machine learning and network topological features. They discuss the relevance of many purely topological node features, as well as ones influenced by biological knowledge (see [3] and [4]), in learning the biological importance of proteins in the context of disease spreading or targeting. The authors also go over which models can be used for such learning, but never actually quantitatively compare their predictive power, and part of our objective has been to try and provide a framework to compare and benchmark models and sets of features against each other.

[5] describes a method to retrieve *functional groups* associated with a disease. They show that purely topological community detection methods are pretty inefficient and miss most of the biological relevance that is expected. Their solution, the "DIseAse MOdule Detection (DIAMOnD) Algorithm", takes as input a set of *seeds*, and uses link similarities to exhibit communities that do not coincide with topological clusters and are relevant to the disease. The robustness of the algorithm is also examined under several angles, to show the validity and relevance of the method. Although the output of this algorithm is very much aligned with our objectives, the biological prior is quite strong since the whole process relies on the seeds to work. As we were focused on our purely uninformed approach, we decided not to further try to reproduce DIAMOnD ourselves, but the article gave us valuable insight on how clustering methods might not be the most relevant in the end, which we partially confirmed by realizing the unimportance of the clustering coefficient feature to predict protein importance.

2 Methods/Model

The goal of this project was to identify relevant topological graph features for the prediction of biologically important genes in a network. The first step was to provide a unified and systematic study of the correlation of topological graph features with the biological importance of the genes. The second step was to use these topological features to predict the biological importance of the nodes. The ground truth label for the biological importance of the gene nodes was given by the presence or absence of this gene in a given reference set.

Convention genes and proteins: The Central Dogma of molecular biology is a framework for understanding the transfer of sequence information between DNA, RNA and proteins. DNA can be copied into mRNA (transcription), and proteins can be synthesized from mRNA (translation). Genes encode proteins but there is no 1-to-1 correspondence between them. However, in the context of network biology, a common approximation is to use interchangeably the genes and proteins names to allow for comparisons between datasets. In this project, we converted all proteins to their corresponding gene names. We did the mapping using the R package BioMart [6] but the conversion is not perfect and 9% of the proteins were lost in the process. We are not satisfied with this lost but it would require significant biological knowledge to recover all genes. In this report, we will use interchangeably "gene" and "protein".

2.1 Datasets

2.1.1 Protein-Protein Interaction Network

There are many publicly available databases that gather PPI Network data. One of the most common, that was used by several of the articles we reviewed, is the STRING[1] database. It contains a comprehensive state of knowledge on all the protein protein interactions known today, and we restricted ourselves to human genes, in order to have manageable volumes of data (the main dumps range from 17Go to 450Go, before decompression) and to be able to use relevant validation data (like disease or drug data).

For a single PPI network from the STRING database, the data consists in a directed weighted graph, the weights representing the degree of certainty of the edges[2]. Typically, the human PPI is a 450 Mb `.txt` file containing the list of edges. Throughout our analysis, we used the human PPI network, which consists in 19,576 nodes and 5,676,528 edges.

[1] Official website: `https://string-db.org/`

[2] The STRING database is a joint collaboration of dozens of researchers around the world who aim at gathering all the knowledge available on protein-protein interactions; every edge of the network stems from the compilation of numerous research works, and is therefore attached with some confidence weights; the weight we consider is an aggregate of the confidence weights available on a given edge, and is the default weight provided in the main STRING dumps.

2.1.2 Reference genes sets

What we call *validation data* is the set of datasets we use to determine whether a protein is said to have an important biological role or not. In our study, we came across or used several validation datasets:

- **Cancer Gene Census** [3] is a list of genes for which there is evidence of implication in cancer ;
- **Online Mendelian Inheritance** [4] references genes associated with genetic disorders ;
- **DrugBank** [5] is slightly different, since it is a drug target database.

  NB: For these data, the extraction process is not straighforward. Actually, the identifiers in Drugbank and other sources cannot directly be matched to the STRING data. As a result, a matching table, available through a `R` package, had to be used to do the matching. Roughly 10% of the original identifiers have not been converted, and so although we computed the topological features on the whole network, the learning and prediction tasks could only be conducted on the nodes that could be matched (about 90%).

These sources each represent different notions of importance. When we work with one of them, a node is labeled "important" if it is contained in the set, and "unimportant" otherwise.

2.2 Feature Evaluation and Selection

The first step of the pipeline was to come up with a set of topological graph features, as predictive of the biological importance of the genes as possible. To test the relevance of the topological features and their correlation with biological importance, we used a unified pipeline that allowed us to compute different metrics on a PPI Network and then to study how these are related with sets of important nodes from different sources. We adopted several approaches, and will briefly describe here the hyper-geometric test, the Mann-Whitney test and the correlation matrix.

Hyper-geometric Test The first and most intuitive approach is to rank all the gene nodes according to one topological feature, and to see if we can spot a significant difference. Given an arbitrary cutoff n the top n genes according to this feature are considered important. We compute k, the number of genes from the validation dataset that are contained in this top-n-genes cluster and compare it to K, the total number of genes in the validation dataset. An hyper-geometric test allows to test the null hypothesis that the number of genes from the validation set present in the top n genes is consistent with (meaning similar to) the rest of the sample. Under the null, we know the probability of observing k important genes in the top n, and we can therefore compute a p-value.

$$P(X = k) = \frac{\binom{n}{k}\binom{N-K}{n-k}}{\binom{N}{n}} \; with \; \begin{cases} N : Number\ of\ genes\ in\ the\ network \\ K : Number\ of\ genes\ in\ the\ validationset \\ n : Number\ of\ genes\ in\ the\ cluster \\ k : Number\ of\ genes\ in\ the\ validation\ set\ and\ the\ cluster \end{cases} \tag{1}$$

The main problem with the previous test is that the criterion is too simple. It may not be relevant to only consider the nodes for which the value of the feature is large ; moreover the threshold is set arbitrarily and makes the test even more shaky. We would like to test whether the distribution of the feature across all nodes in the network is significantly different from the distribution of the feature restrained to the biologically important nodes (present in the validation set).

Mann-Whitney Test To this end, we used a Mann-Whitney test[6], which is robust to differences in sizes between the samples to compare. It is non-parametric, and can therefore be used for empirical data like the one we are considering. The idea is to consider the two sets of observations, and mix them up. Under the null hypothesis — which is that the two samples are drawn from the same underlying population — the observations from the two samples should be evenly distributed within the joined sample. The Mann-Whitney U statistic considers the sum of the ranks of the first sample's observations when mixed with the second's. U is the smallest of U_1 and U_2, where

$$U_i = n_1 n_2 + \frac{n_i(n_i + 1)}{2} - R_i \; for\ i \in \{1, 2\} \tag{2}$$

where n_i is the number of observations in group i, and R_i is the sum of the ranks of the elements of group i when mixed up with elements of the other group.

This statistic behaves in a very known fashion under the null, which enables us to compute p-values. For the Mendelian set, instead of comparing the distribution of important nodes to the total, we compared it to the unimportant nodes (otherwise we would take advantage of the bias due to the over-representation of Mendelian-important nodes).

[3] Official website: `http://cancer.sanger.ac.uk/census`
[4] Official website: `http://omim.org/`
[5] Official website: `https://www.drugbank.ca/`
[6] `https://en.wikipedia.org/wiki/Mann%E2%80%93Whitney_U_test`

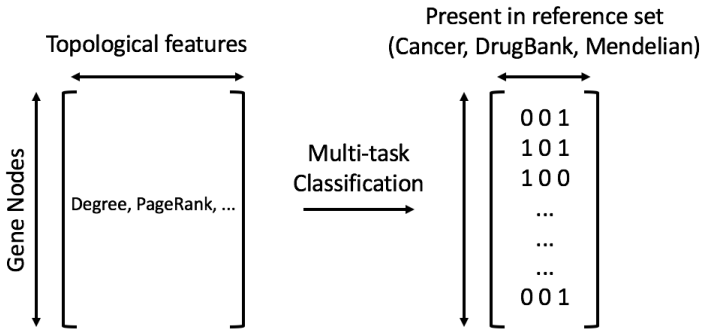


Figure 1: Node's importance prediction pipeline

Multiple Hypothesis Testing We tested an important number of correlations and computed a lot of p-values, we have to account for the problem of multiple hypothesis testing. We are computing the p-values for 8 features, 3 reference gene sets and 2 different tests which makes 48 p-values. We used the Bonferroni correction to counteract the problem of multiple comparisons. Instead of testing each individual hypothesis at a significance level of α, we used a significance level of $\frac{\alpha}{m}$, where m is the number of hypotheses. In our case, m=48 and we chose $\alpha = 0.05$ so $\frac{\alpha}{m} = 0.001$

Correlation Matrix We used a correlation matrix to (mostly visually) assess the inter-correlation between our features.

2.3 Gene Biological Importance Prediction

2.3.1 Training Procedure

The main goal of this project was to use topological graph features to predict which nodes are biologically important in a biological network. To this end, we used the features selected in the previous section to train classifiers and see if we can have a better predictive power by combining the features. The predictors matrix is composed of the graph topological features extracted for every nodes and there is a multi-task label (for each source: Mendelian, DrugBank, Cancer). Each label is a binary vector corresponding to whether a node/gene is present in the reference gene set. The general pipeline is illustrated in Figure 1.

As explained previously, a preliminary analysis of features correlation and several statistical tests allowed us to select a set of features. Our final set of features contains 74 features: 10 hand-crafted features and 64 graph embedding features from node2vec. We trained several classifiers on the full set of features but also on a reduced set comprised of the handcrafted features and the Principal Component Analysis of node2vec features:

- **Logistic Regression** with L2 penalty
- **Random Forest**
- **Gradient Boosting Regression Trees**
- **Support Vector Machine** with both `rbf` and `sigmoid` kernels
- **Neural Network** with 2 hidden layers
- **Regularized Gradient Boosting Trees** using `XGBoost`

For all those models, we used a cross-validation procedure with grid search over the hyper-parameters to fine tune the model. The majority of the classifier selection and evaluation has been done on the **DrugBank** dataset. Actually, this is the most balanced dataset we have for validation, as 13% of the samples have positive labels, whereas the coverage of Mendelian and Cancer are much more unbalanced. To counteract the class imbalance, we used metrics such as the F1 score and ROC curves to perform our model selection and we weighted the positive examples proportionally to their imbalance.

2.3.2 Validation Procedure

The usual validation procedure in machine learning is to divide the predictors matrix into a train, validation and test set. A major assumption in that case is the independence of the observations. One of the major challenge when working with topological graph features is that this assumption doesn't hold. Indeed, the topological features depend on all the nodes, and so the observations are correlated. For instance, PageRank is defined as a weighted sum of the influence of the nodes around, the degree is obviously dependent on the presence of surrounding nodes, etc.

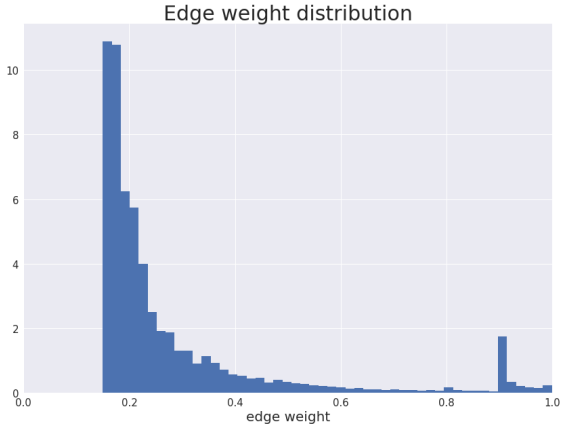


Figure 2: Edge weight distribution (scaled down to between 0 and 1)

As a consequence, information from the test set inevitably leaks into the training set. The predictions are biased, because the features of the training nodes themselves already depend on the test set. We had three possibilities to go around this problem:

- **Ignore the issue**: We compute the features once at the beginning for the whole graph, and then split the whole matrix into training and testing sets. In this case we compute the features only once and we know that all our features make topological sense. However, there is the data leakage issue, which biases the model predictions.
- **Split the graph beforehand**: We split the nodes into the training nodes and the testing nodes and compute the features independently. In this case, we avoid data leakage because the train and test sets are independent. However, the issue is that the graphs we obtain by removing a set of nodes don't respect the initial topological configuration, and most likely don't make any sense at all.
- **Validate on another PPI network** The most rigorous method would be to train a model on a given graph and evaluate on another (train on a human PPI and test on another organism). We would avoid data leakage and the features would make sense. The problem in this case is the transferability of the model across species. Also, it might be harder to find the ground truth label for node importance (drug target and disease genes) for other species, if they exist at all.

We went for the first option, which made the most sense in our case. The last option would probably be the most rigorous, but the lack of validation data made it hard to conciliate with our framework.

3 Results and Findings

3.1 Dataset Exploration Analysis

3.1.1 The STRING Protein-Protein Interaction Network

The main dataset that interests us is the human PPI network, which is publicly available on the STRING database. This network exhibits characteristics that make it challenging to run classic algorithms on it. Although it "only" consists in about 19k nodes, it is very dense, with about 5.5M edges. This makes some basic computations such as Page Rank or Clustering Coefficient very costly, which forced us, first, to work exclusively on remote servers for these computations, since we had to compute the features for every node in the network ; and second, to prune the graph and make it more computationally handleable, as detailed below. Another key aspect of it, which we did not anticipate, is that it is actually an **undirected**, weighted graph. In theory, the data is that of a directed weighted network, but we quickly realized that in practice every edge was present twice in the dataset, with the same weight every time. This makes the notions of controllability from [1], outbreak detection and influence maximization less relevant. Part of our work consisted in figuring out in what way we can still find interesting features to explain those mechanisms, even though the graph actually carries less information than we thought initially.

Finally, one interesting point to note is that the edges in the network are weighted. The weight of a link in the STRING database is a score between 0 and 1000, that corresponds to the confidence score (out of 1000) based on the number and reliability of papers that exhibit the considered link. As a result, if this score can be seen as an empirical probability of existence of a link, or a degree of certitude, it is difficult to give an interpretation in terms of real biological links or in terms of topological meaning on the graph, and we therefore did not use it extensively. The only time it proved useful was when computing a feature on the whole network was too costly (more than 150 hours were required to compute the neighboring conductance at level 4), and we had to prune the graph by applying a threshold on the weight. Figure 2 shows that a threshold around .5 is sufficient to end up with a much more manageable graph (by removing about 80 to 90% of the low-confidence edges).

3.1.2 Reference genes sets

From the several datasets that were previously mentioned, we extracted lists of proteins that play an important role in several biological problems. We then had to build matching between those lists and the PPI network nodes. One simple statistic to look at is the coverage of each of those lists on the graph. We get the following results:

Source	Cancer	Drugbank	Mendelian
Coverage	3%	12%	77%

3.2 Feature Evaluation

We have implemented and tested a list of topological features on the graph: Degree, Expected Degree[7], Clustering Coefficient, Betweenness Centrality, Closeness Centrality, PageRank, HITS scores[8], and Neighborhood conductance.

Most of those metrics have been covered in class. We introduced Neighborhood conductance. Intuitively we are interested in whether the node is part of a "central" part of the network. To evaluate this, we consider a neighborhood of the node in the network, and compute the conductance of the set. As a result, there are 2 parameters to adjust for this metric: the size of the neighborhood, *i.e.* the maximum distance of the nodes we consider to be in it, and the threshold for considering the edges. Actually, as we explained sooner, the computations of this metric can be really heavy, so we performed it on a pruned graph. The intensity of pruning is also a parameter to adjust for this metric (however, we only chose it depending on computational constraints and did not try to optimize it in terms of predictive performance).

Additionally, we worked with embedding features based on the node2vec paradigm (see [7] for more details on the algorithm). Basically the idea is to map the nodes to a finite dimensional (we used 32-dimensional) space in a way that keeps the neighbors and clusters as close to each other as possible. We decided to consider those 32-dimensional vectors as a block and not to conduct the same preliminary analysis with them as with the others. We later show the importance of these features in our prediction models, and in particular the importance of their principal components.

3.2.1 Hyper-geometric and Mann Whitney Test

To assess how predictive of biological importance each of the main features is when taken on its own, we realized the tests described in 2.2, and show the results in Tables 1 and 2. Figure 3 gives a graphical illustration of the distributions used in the Mann-Whitney test. Note that the hyper-geometric test we implemented is irrelevant for conductances, since only small values of conductance are correlated with importance (and not the largest ones).

	Cancer Genes	DrugBank Genes	Mendelian Genes
Degree	$8.753e-129$	$4.106e-35$	$4.106e-35$
Expected Degree	$6.402e-169$	$5.421e-34$	$5.421e-34$
Clustering Coefficient	0.462	0.999	0.999
Closeness	$5.970e-137$	$5.421e-34$	$5.421e-34$
Betweenness	$4.337e-138$	$8.602e-40$	$8.602e-40$
HITS (Authorities)	$5.043e-143$	$6.852e-33$	$6.852e-33$
PageRank	$2.403e-182$	$2.974e-36$	$2.974e-36$

Table 1: p-values of the Hyper-geometric test

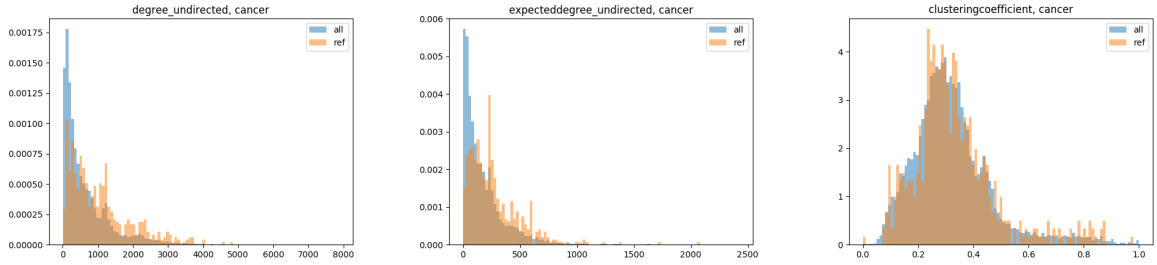


Figure 3: Distribution of features across all genes ("all") versus across reference genes ("ref")

[7] As we explained, the edges of the graph are weighted with what we can see as a confidence score of their existence. If we see this weight as the probability of the edge existing, then summing the weights of the edges surrounding a given node should give an idea of the most probable number of neighbors it has: this is the expected degree.

[8] As we're dealing with an undirected network after all, hubs and authorities are equivalent notions, and we therefore only kept one of the two

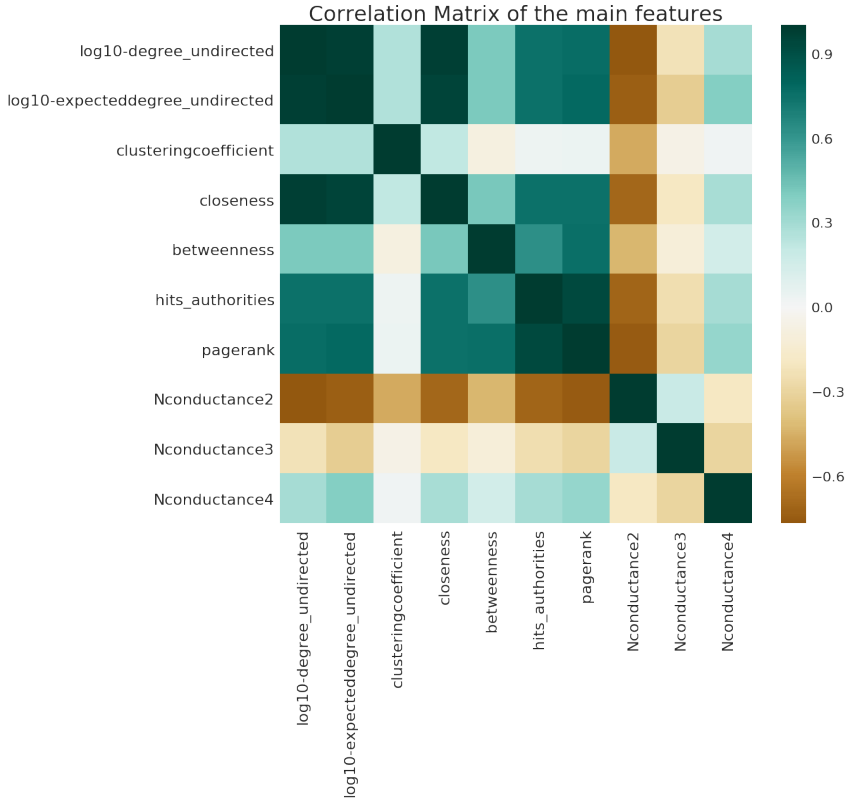


Figure 4: Correlation Matrix of the topological graph features

[H]

	Cancer Genes	DrugBank Genes	Mendelian Genes
Degree	$1.173e-70$	$8.753e-129$	$8.753e-129$
Expected Degree	$8.115e-77$	$6.402e-169$	$6.402e-169$
Clustering Coefficient	0.012	0.462	0.462
Closeness	$1.824e-77$	$5.970e-137$	$5.970e-137$
Betweenness	$2.327e-69$	$4.337e-138$	$4.337e-138$
HITS (Authorities)	$6.033e-84$	$5.043e-143$	$5.043e-143$
PageRank	$1.612e-75$	$2.403e-182$	$2.403e-182$
Nconductance 2	$5e-48$	$3.1e-121$	$3.5e-78$
Nconductance 3	$1.5e-24$	$4.2e-33$	0.0017
Nconductance 4	$1.2e-36$	$3.2e-153$	$6e-69$

Table 2: p-values of the Mann-Whitney test

3.2.2 Correlation Matrix

We used a correlation matrix — like shown in Figure 4 — to (mostly visually) assess the inter-correlation between our features. In particular, this enabled us to spot very high correlation between degree and expected degree, PageRank and HITS authorities, and PageRank and level 2 neighboring conductance. We notably dropped expected degree from our analyses, and considered that despite high correlation, all the other features were relevant to keep.

3.2.3 Interpretation: Feature selection

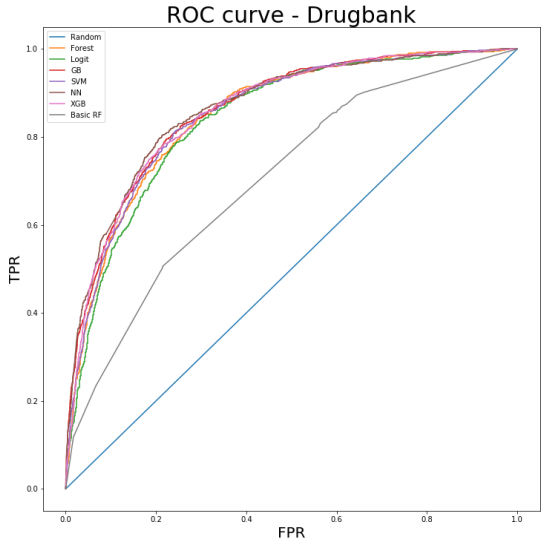
We used the p-values from the hyper-geometric test and the Mann-Whitney test to understand the correlations between the topological features and the importance of the nodes. We also used the Bonferroni correction to counteract the problem of multiple comparisons. Instead of testing each individual hypothesis at a significance level of α, we used a significance level of $\frac{\alpha}{m}$, where m is the number of hypotheses (48 in our case). We chose $\alpha = 0.05$ so $\frac{\alpha}{m} = 0.001$.

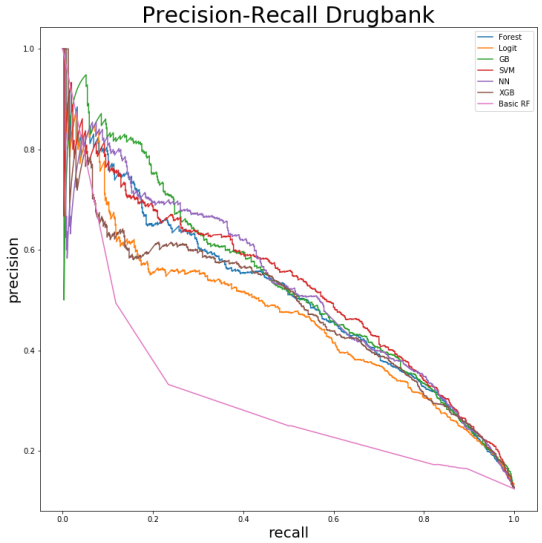
Given the p-values, we observe that all features are below the corrected significant threshold of 0.001 except the clustering coefficient. This already allows to identify that **the clustering coefficient is not likely to be an important predictor** of the biological importance of a node. However, those tests are not the only indicators of a feature's predictive power. We see that

our tests are way too conservative, in that they reject the null very easily. This means that small p-values don't necessarily mean that a feature will prove significant, and that training models is indispensable. Also, the features can be correlated between each other, and looking at the correlation matrix is a good sanity check, because this enables us to drop some features.

3.3 Gene Biological Importance Prediction

We trained various models on different sets of features (full, selected or reduced with PCA) and analyzed the results. We plotted ROC curves and Precision-Recall curves and computed the F1 score, precision and recall at the optimal threshold for the F1 score. The results with the DrugBank reference gene set are given in Figure 5.





Algorithm	F1 score	Precision	Recall
Forest	0.52	0.5	0.54
Logistic Regression	0.50	0.47	0.55
GB	0.52	0.49	0.56
SVM	0.54	0.52	0.57
NN	0.53	0.51	0.56
XGB	0.51	0.43	0.65
Basic RF	0.33	0.25	0.51

Figure 5: Predictive Algorithm Benchmark for DrugBank reference

For the sake of clarity, and since this is the most interesting dataset in terms of model selection, we only reported the results for DrugBank. It is interesting to note that the models have different behaviors for the three datasets. This is mainly due to the unbalance in those datasets. We observe that the DrugBank validation set is more appropriate to separate the models, both in the ROC curve and the PR curve, probably because it is the most balanced.

In order to further validate the choice of the model and improve on its performance, one would need to analyze the prediction errors with domain-specific knowledge. Our errors might be due to statistical reasons but also to our modeling assumptions. However, this analysis allows us to choose a predictor that has a decent performance, and based on this predictor, to evaluate the importance of the features. The Gradient Boosted Trees feature importances are reported in Figure 6.

According to those feature importance, and if we compare it qualitatively with what other predictors yield, the most important features seem to be:

- The 5 main components of the node2vec embedding
- The neighborhood conductance of degree 4
- The closeness centrality
- The PageRank score.

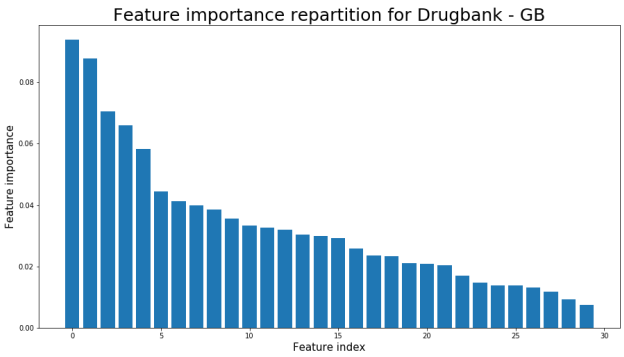


Figure 6: Predictive Algorithm Benchmark for Drugbank reference

The other handcrafted features are much less often in the top features, and often in the middle of the ranking, so we can clearly identify the previous set of features as the most important.

The clustering coefficient feature appears to have very low importance in the models, which validates our intuition from the feature evaluation section.

4 Discussion

False Positives interpretation In this study, we built a Machine Learning classifier to recognize a set of important genes using topological graph features. In this case, the ground truth for the set of important nodes was given by some external validation dataset. However, if we train a model to recognize genes involved in cancer, the true positives might not be the most useful predictions because we already knew that these genes were related to cancer beforehand. On the other hand, it might seem counter-intuitive but we would be more interested in the set of False Positives that the model classifies erroneously with high confidence. For these genes, the model predicts that they are related to cancer while the reference dataset didn't include them. These genes could therefore be good candidates for further research studies. An application of our models would be to guide cancer genomics studies or drug discovery pipelines by identifying potential new drug targets or genes related to a disease.

STRING is undirected One of the main features of the STRING network is that it is undirected. The relationships between proteins can be interpreted as symmetric interactions scores. There are no directed influences, as it was the case in some of the articles we used as starting points. This specificity of the STRING network makes some of the analyses we wanted to perform irrelevant: for instance the controllability analysis [1]. However, the approach developed in this paper is very general and therefore could be applied to a directed network such as a gene regulation network. This could yield even better results as the directed features would supplement the undirected features.

Semi-informed approach Another way of improvement would be to use a semi-informed approach, that would allow to use the module detection notion presented in [5]. Actually, in the module detection paradigm, we start from a few seed nodes and use them to retrieve the remaining community. Considering some of the labels as known and designing supervised features based on those labels and module detection algorithms may also be a way to improve our analysis, or to check if the false positives we have in our prediction are potential important nodes that are not present in our validation sets.

Biological Interpretation of features importance

- **Top predictors:** Our analysis shows that the top predictors are the closeness centrality, the longer range neighborhood conductances and the pagerank algorithm. They can all be interpreted as centrality measures of the node in the network.
- **Medium predictors:** The mid-predictors are less predictive of the node importance because they could be correlated together or they might be too local to be really representative of the importance of a node. Also, we should point out the literature bias that arises from the construction of the PPI network. The most studied nodes are more likely to be connected to a lot of other nodes and therefore have higher degrees. HITS scores are quite correlated with PageRank and they might perform better if the graph was directed. The notion of betweenness centrality may be linked to the notion of biological pathways present in the graph, hence leading to assume that the important nodes are on many shortest paths. The *mid-range* neighborhood conductances represent a more extended characterization of the centrality of the node in the network.

- **Weak predictors:** On the other hand, some features such has the clustering coefficient or the neighborhood conductances of small range have a very poor predictive power for the importance of nodes. These metrics are probably too local to really quantify the importance of a node.

Concerning the Neighborhood conductance, the pruning of the graph might have an effect on the importance of this feature. Actually, the higher the range of the neighborhood conductance considered, the more restrictive the pruning had to be. This may reduce noise in the edges and lead to a better metric.

Node2vec components are more difficult to interpret but as expected, those features allowed us to capture more sophisticated topological features of the graph, and to characterize a node based on more global metrics.

5 Conclusion

In this study, we designed a systematic approach to evaluate the relevance of topological graph features in the prediction of nodes biological importance. To achieve this goal, we built a framework that allowed us to easily test the relationship between biological importance of proteins and PPI network-based topological features, and to effortlessly switch between different sets of features and models.

This allowed us to identify some topological features that have a strong predictive power for identifying biologically important proteins. Determining such a set of relevant features is a way both to understand what the biological importance means in terms of graph theory and to build predictive models that may be used to focus some other studies.

Actually, one of the main features of the PPI network that we had to keep in mind during the project is that it is by construction both biased and incomplete, so that the analysis we perform on such datasets have to be carefully interpreted.

6 Addendum

All team members provided a significant contribution to the presented work. An approximate repartition of the tasks is the following:

- Vincent: Paper writing, statistical tests, correlation analysis, remote server calculations, overall code infrastructure/cleaning.
- Pierre-Louis: Paper writing, evaluation metrics, machine learning pipeline, hyper-geometric testvalidation set importation.
- Matthieu: Paper writing, setting up the OO pipeline, hyper-parameter tuning and model selection, node2vec and neighbouring conductance integration.

Note: Our git repository is available. We are open to sharing the data and any other material the reader might find useful.

References

[1] Arunachalam Vinayagam, Travis E Gibson, Ho-Joon Lee, Bahar Yilmazel, Charles Roesel, Yanhui Hu, Young Kwon, Amitabh Sharma, Yang-Yu Liu, Norbert Perrimon, et al. Controllability analysis of the directed human protein interaction network identifies disease genes and drug targets. *Proceedings of the National Academy of Sciences*, 113(18):4976–4981, 2016.

[2] Xue Zhang, Marcio Luis Acencio, and Ney Lemke. Predicting essential genes and proteins based on machine learning and network topological features: a comprehensive review. *Frontiers in physiology*, 7, 2016.

[3] Xiwei Tang, Jianxin Wang, and Yi Pan. Identifying essential proteins via integration of protein interaction and gene expression data. In *Bioinformatics and Biomedicine (BIBM), 2012 IEEE International Conference on*, pages 1–4. IEEE, 2012.

[4] Wei Peng, Jianxin Wang, Weiping Wang, Qing Liu, Fang-Xiang Wu, and Yi Pan. Iteration method for predicting essential proteins based on orthology and protein-protein interaction networks. *BMC systems biology*, 6(1):87, 2012.

[5] Susan Dina Ghiassian, Jörg Menche, and Albert-László Barabási. A disease module detection (diamond) algorithm derived from a systematic analysis of connectivity patterns of disease proteins in the human interactome. *PLoS computational biology*, 11(4):e1004120, 2015.

[6] Steffen Durinck, Paul T Spellman, Ewan Birney, and Wolfgang Huber. Mapping identifiers for the integration of genomic datasets with the r/bioconductor package biomart. *Nature protocols*, 4(8):1184–1191, 2009.

[7] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.