POLYTECHNIC UNIVERSITY OF TURIN

Master degree course in Biomedical Engineering - eHealth

Master Degree Thesis

Data integration for the analysis of the oncogenic potential of gene fusions

Machine learning

Supervisors
Prof. Elisa Ficarra
Eng. Marta Lovino

Candidate
Venere Sabrina Barrese
matricola: 245877

Accademic year 2019-2020

To my father,
Straight ahead.
Always.

Abstract

The cells' life cycle is strictly related to the DNA replication and DNA transcription. Under certain conditions the DNA may break and create aberrant products known as gene fusions. A gene fusion is made up of two genes, usually coming from different chromosomes and called gene pairs. After the breaking event, a portion of both genes can be lost, and the point in which each gene breaks is known as breakpoint.

Gene fusions have been proven to be related to certain types of cancers, and in this case they are defined as driver gene fusions. Gene fusion detection tools are commonly used to identify gene fusions in a biological sample. However, these tools detect a high number of putative fusions in tumor samples and sometimes do not confidently label them as oncogenic. This suggests the need to gain more insights into the role of gene fusions in cancer.

This thesis examines three elements associated with the gene pairs and their role in the evaluation of gene fusions' oncogenic potential: transcription factors (TFs), gene ontologies (GOs) and micro-RNAs (miRNAs). Under the assumption that these elements, along with the information deduced from the gene names and the breakpoints, can characterize a gene fusion, two machine learning methods were used to discern the driver fusions from the passenger events (e.g. gene fusions not related to cancer): the support vector machines (SVMs) and the multilayer perceptron (MLP).

The classifiers were trained on 1765 thoroughly validated gene fusions and tested on 5246 samples. The training samples and the oncogenic test samples come from an ensemble of databases that were analyzed by DEEPrior [1], while the healthy test samples were extracted from Babiceanu's paper[2].

The developed method first exploits the information related to the gene names and the breakpoints to extract the following features for both the genes: the percentage of retained gene after the fusion event, the putative role assigned by the *Cancermine*[3] database and whether the two genes are transcribed in the same direction or not. The training and the cross-validation were performed on the training set using these features returning a cross-validation AUC higher than 88% for both the linear SVM and the MLP.

Then the relationship between the transcription factors and the genes [4] was examined. The complete set of 181 transcription factors was used to train and cross-validate the training set. The combination of the previously

defined features and the 181 transcription factors led to an improvement in the performance metrics of both classifiers.

An analogous process was carried out to integrate the information coming from the gene ontologies. The GOs were obtained using the Biomart tool[5] gathering a total of 5125 features.

Finally, the association between miRNAs and genes was retrieved from Targetscan [6] as a list of probabilities defining the strength of the relationship between miRNAs and genes. A total of 333 miRNAs were identified as features.

The final developed method is a MLP with 4 layers trained using the initial features, TFs, GOs and miRNAs. The cross-validation performance metrics were 90%, 86%, 99%, 0.88 respectively for accuracy, precision, recall, AUC. The same metrics computed on the test set were: 81%, 78%, 86%, 0.81.

The complete pipeline proved to be able to integrate the different sources of data and discriminate, with adequate reliability, the driver from the passenger gene fusions. The tool returned higher performances compared to the results obtained by *Oncofuse*[7], a similar tool found in the literature.

Acknowledgements

To my supervisors prof. Elisa Ficarra and eng. Marta Lovino, I'm deeply grateful for the support and help that you gave me. In several occasions you provided me with invaluable teachings allowing me to finish this incredible learning experience as a more knowledgeable and confident woman. You went on and beyond for me during the hardest time of my life and guided me until I was able to reach this goal, thank you.

To my parents, for giving me the opportunity to achieve my ambitions and graduate from the university of my dreams. And to my dad, thank you for believing in me. I know how long you waited for this moment but the circumstances of life took you away from me right before you could see me finally reaching the end of this journey. I know how proud you would have been, thank you for everything.

To Lele, my companion for life, the one that was there for me, always. I'm so happy we've grown together during this amazing adventure, with you I've spent the most wonderful years of my life. There are no words to explain how much I'm grateful for your support, your precious advice and your friendship. To say that I love you like you were my brother would be an understatement, so let's just say that you are my person. Thank you.

To Alhadji, when I met you I didn't know how to ask for what I wanted. You helped me through my insecurities and fears, you understood me and you taught me that I deserve to go after the things I want. Thank you.

To all of my friends, the ones that live close to me and the ones that live far, you were not only a part of my journey, you were the people that helped me leave my comfort zone. With you, I share the best memories of these university years, thank you for the laughs, the support and the love.

To everyone that was part of this journey, even for a little while, I'm thankful for everything we shared.

Contents

List of Tables 7

List of Figures 8

1 Introduction 13

2 Background 15
2.1 Gene fusion . . . 15
2.2 Transcription factors . . . 16
2.3 micro-RNA . . . 17
2.4 Gene ontology . . . 17
2.5 Initial features . . . 18
2.6 Feature selection . . . 19
2.7 Machine learning . . . 20
2.7.1 Support Vector Machine . . . 20
2.7.2 Neural Networks . . . 21

3 Methods 27
3.1 Pipeline . . . 27
3.1.1 Data integration . . . 29
3.1.2 The five initial features . . . 29
3.1.3 Feature selection . . . 31
3.1.4 Transcription factor features . . . 32
3.1.5 Gene ontology features . . . 32
3.1.6 miRNA features . . . 33
3.2 SVM . . . 34
3.2.1 MLP . . . 34
3.3 Oncofuse . . . 35
3.4 Additional information . . . 37

4 Results 39
4.1 SVM cross validation . . . 39
4.2 SVM testing . . . 47
4.3 MLP cross validation . . . 57
4.4 MLP testing . . . 60
4.5 Oncofuse . . . 64

5 Discussion 69
5.1 SVM . . . 69
5.1.1 The importance of the five initial features . . . 69
5.1.2 The complete feature set . . . 70
5.1.3 Tuning of the parameters . . . 71
5.2 MLP . . . 72
5.3 Comparison with Oncofuse . . . 73

6 Conclusion 75

Bibliography 77

List of Tables

3.1 Approximate number of features obtained with the random forest feature selection by using different values of threshold . 31

List of Figures

2.1 Representation of the gene fusion event 15
2.2 Matrix obtained by ENCODE 16
2.3 Matrix obtained by Targetscan 17
2.4 Table obtained by Ensembl 18
2.5 The five features deducible from the gene name and the breakpoints 19
2.6 Confusion matrix 24
3.1 Complete pipeline.
Legend: DP=DEEPrior, CM=CancerMine, g37=grch37, g38=grch38, ENC=ENCODE, Ens=Ensembl, TS=TargetScan 28
3.2 Obtaining the TF features for the training set from the gene attribute matrix 32
3.3 Obtaining the GO features from the ensembl table 33
3.4 Obtaining the miRNA features from the matrix of targetscan . 34
4.1 Comparison of the performances obtained using different kernels and different subsets of features 40
4.2 Comparison of the performances obtained using the linear kernel and different subsets of features. The number of features was reduced using the random forest feature selection 42
4.3 Comparison of the performances obtained using the gaussian kernel and different subsets of features. The number of features was reduced using the random forest feature selection . . 42
4.4 Comparison of the performances obtained using the sigmoid kernel and different subsets of features. The number of features was reduced using the random forest feature selection . . 43
4.5 Comparison of the performances obtained using the polynomial kernel and different subsets of features. The number of features was reduced using the random forest feature selection 43
4.6 Cross validation results for linear kernel with coeff = [0.001, 0.01, 0.1, 10] 44

4.7 Cross validation results for rbf kernel with coeff = [0.001, 0.01, 0.1, 10] and gamma = [0.001, 0.01, 0.1, 10] 45
4.8 Cross validation results for sigmoid kernel with coeff = [0.001, 0.01, 0.1, 10] and gamma = [0.001, 0.01, 0.1, 10] 45
4.9 Cross validation results for polynomial kernel with coeff = [0.001, 0.01, 0.1, 10], gamma = [0.001, 0.01, 0.1, 10] and degree = 2 46
4.10 Confusion matrix of the training set obtained with the SVM model characterized by sigmoid kernel, coeff = 0.001 and gamma = 0.001 46
4.11 AUC values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.01, 0.1, 10] and the range of gamma values was: [0.001, 0.01, 0.1]. The degree for the polynomial kernel was equal to 2. 48
4.12 Accuracy values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.01, 0.1, 10] and the range of gamma values was: [0.001, 0.01, 0.1]. The degree for the polynomial kernel was equal to 2. 48
4.13 AUC values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.1, 10, 100] and the range of gamma values was: [0.001, 0.01]. The degree for the polynomial kernel was equal to 2. 49
4.14 Accuracy values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.1, 10, 100] and the range of gamma values was: [0.001, 0.01]. The degree for the polynomial kernel was equal to 2. 49
4.15 Performance metrics for the model characterized by the following parameters: rbf, gamma = 0.01, coeff = 0.1 50
4.16 Confusion matrices for train and test set. The parameters of the SVM model were: rbf, gamma = 0.01, coeff = 0.1 50
4.17 AUC values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.1, 10, 100] and the range of gamma values was: [0.001, 0.01]. The degree for the polynomial kernel was equal to 2. 51

4.18 Accuracy values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.1, 10, 100] and the range of gamma values was: [0.001, 0.01]. The degree for the polynomial kernel was equal to 2. 52
4.19 Confusion matrices for train and test set. The parameters of the SVM model were: rbf, gamma = 0.01, coeff = 0.1 52
4.20 Performance metrics for the model characterized by the following parameters: rbf, gamma = 0.01, coeff = 0.1 53
4.21 AUC values for gamma = [0.001, 0.005, 0.01, 0.05, 0.1], coeff = [0.01, 0.05 ,0.1, 5, 10] 54
4.22 AUC values for gamma = [0.005, 0.01, 0.05], coeff = [0.1, 2, 5] 54
4.23 AUC values for gamma = [0.04,0.05,0.06], coeff = [0.5,1.5,2] . 55
4.24 AUC values for gamma = [0.02,0.03,0.04], coeff = [0.4,0.5,0.6] 55
4.25 AUC values for gamma = [0.035,0.04,0.045], coeff =[0.55,0.6,0.65] 56
4.26 Accuracy values for train and test set in the final tuning phase 57
4.27 Confusion matrix for the best SVM model: kernel = rbf, gamma = 0.04 and coeff = 0.6 58
4.28 Performances of the best SVM model: kernel = rbf, gamma = 0.04 and coeff = 0.6 58
4.29 Comparison of the performances obtained using different subsets of features 59
4.30 Comparison of the performances obtained using different subsets of features. The number of features was reduced using the random forest feature selection 59
4.31 Accuracy values for the train and test sets obtained with different MLP models. The range of the learning rate was = [0.0001, 0.001, 0.01] and the range for the dropout value was = [0, 0.1, 0.2, 0.3, 0.4, 0.5] 60
4.32 Accuracy values for the train and test sets obtained with different MLP models. The range of the learning rate was = [0.002, 0.004, 0.008] and the range for the dropout value was = [0.3, 0.4] 61
4.33 Accuracy values for the train and test sets obtained with different MLP models. The learning rate was equal to 0.002 and the dropout value was equal to 0.3. Legend: 0 = 512, 1 = 256, 2 = 128, 3= 64, s = sigmoid, t = tanh, r = relu. 62

4.34 Accuracy values for the train and test sets obtained with different MLP models. The learning rate was equal to 0.002 and the dropout value was equal to 0.3. Legend: 0 = 512, 1 = 256, 2 = 128, 3= 64, s = sigmoid, t = tanh, r = relu. 63
4.35 Confusion matrices for train and test set of the best MLP model. The parameters were: learning rate = 0.002, dropout = 0.3, layers = 256-512-512-256, activation functions = relu-sigmoid-relu-sigmoid 64
4.36 Performance metrics for the best MLP model. The parameters were: learning rate = 0.002, dropout = 0.3, layers = 256-512-512-256, activation functions = relu-sigmoid-relu-sigmoid . . . 64
4.37 Figure 2 of Oncofuse 65
4.38 Comparison of the assumed performances of Oncofuse with respect to the performances obtained by the best MLP model 66
4.39 Confusion matrices for the train and test set obtained after training and testing the best MLP model found earlier on the data provided by Oncofuse 66
4.40 Metrics for the train and test set obtained after training and testing the best MLP model found earlier on the data provided by Oncofuse 67
4.41 Comparison of the assumed performances of Oncofuse with respect to the performances obtained by the optimal MLP model 67
4.42 Confusion matrices for the train and test set obtained after training and testing the optimal MLP model 68
4.43 Metrics for the train and test set obtained after training and testing the optimal MLP model 68

Chapter 1

Introduction

Under particular conditions the DNA may be subjected to alterations and rearrangements. As a consequence, the DNA may break and two genes belonging to different parts of the DNA may be joined. This abnormal juxtaposition of different genes, which usually belong to two different chromosomes, is defined as gene fusion or chimera.

Many studies have focused on the possible correlation between gene fusions and cancer. In particular, according to the paper written by Mitelman F. et al.[8], gene fusions are accountable for about 20% of human cancer morbidity. The authors stated that there is evidence that events like translocations, and the corresponding gene fusions, not only occur in various malignancies but may also be responsible for oncogenesis.

Some gene fusions have been linked to particular types of cancers, for example, several studies concerning the implication of BCR-ABL gene fusion in leukemia[9] have highlighted the necessity of investigating more in-depth this phenomenon. Analogously other studies correlate the TMPRSS2-ERG gene fusion to prostate cancer[10] and the BCAM-AKT2 to ovarian cancer[11]. Nevertheless, there is a vast number of gene fusions that seem to never engage in any harmful behavior. One of the possible reasons for this may be found by considering the breaking event.

As already mentioned, a gene fusion obtained consequently to the break of the DNA involves two genes: the one closer to the promoter region of the fusion is defined as 5' gene whilst the gene closer to the end of the fusion is defined as 3' gene. If a gene at the 5' position that is characterized by a

promoter with low "transcription power" fuses with a potentially oncogenic gene the behavior of the latter would be mitigated by the new promoter region. On the other hand, if the gene at the 5' position were to carry a powerful promoter region that gene fusion would likely cause some damage to the cells. In the paper written by Stenman G. et al. [12] the authors investigated the effect of MYB-NFIB fusion on MYB gene. As a matter of fact he stated that the truncation of the MYB gene, following the breaking event of the DNA, resulted in the loss of important binding sites. These binding sites were specifically related to microRNAs that are responsible for the deregulation of the gene, thus their loss may negatively impact the repression of MYB by miRNAs, for instance MYB-NFIB transcripts may be overexpressed as a consequence of this loss.

A confident and direct correlation between a given gene fusion and cancer would be of benefit for scientists and would allow physicians to make faster and definite diagnoses. However, each gene fusion requires thorough testing and inspection to be considered a driver for a tumor.

As a matter of fact, to assess the presence of a particular gene fusion in a sample an experimental validation must be performed. The validation of gene fusions may be achieved with PCR or through a functional validation. The latter in particular requires a significant amount of resources and is more expensive than PCR. Since the validation of a gene fusion can be both time consuming and expensive over the last 20 years a lot of effort has been put into detecting and prioritizing gene fusions, for this purpose, tools like RNA-Seq[13], Pegasus[14], DEEPrior[1] and Oncofuse[7] have proven to be valuable resources.

The aim of this thesis is to contribute to this research with a helpful tool able to classify gene fusions into oncogenic or non-oncogenic. This analysis aims to be of support to physicians that may need guidance when they come across a potentially harmful chimeric gene.

Chapter 2

Background

2.1 Gene fusion

A gene fusion is the result of two different genes joining after a breaking event at the DNA level (2.1). The fusion event can be subsequent to structural DNA rearrangements, transcription read-through, or trans/cis-splicing of pre-mRNA [15].

A fusion involves a gene at the 5' position which would be closer to the promoter region and a gene at the 3' position namely the gene at the other end of the fusion, and it is characterized by two breakpoints, one for each gene. Also known as chimeric genes, these genomic aberrations can be identified using the RNA-seq, a next-generation sequencing technique that investigates reads that span the fusion breakpoints. Unfortunately, there is a low consensus among different sequencing tools that results in a source of uncertainty, which is difficult to handle in the subsequent gene fusion analysis [16].

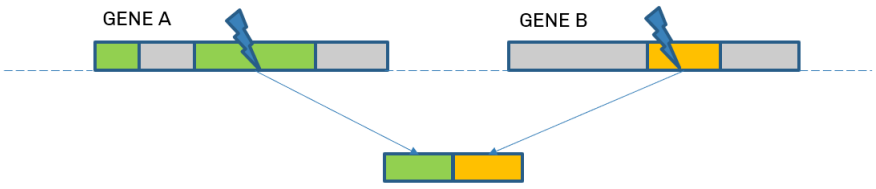


Figure 2.1. Representation of the gene fusion event

Gene fusions are often associated to tumors and are defined as driver mutations in many different types of cancer[17][18][19], however not all gene

fusions result in damaging effects on the organism[20][21], from the aforementioned considerations arises the need for a tool able to discern between this two occurrences.

This thesis focuses on analyzing some of the elements that partake in the normal activities of chimeric genes and how they can be linked to oncogenic behaviors. In particular, this study took into consideration transcription factors and microRNAs along with inherent characteristics of each gene involved in the fusion such as gene ontologies, breakpoints, strands, definition of a gene (e.g. oncogene, tumor suppressor, driver, other).

2.2 Transcription factors

The transcription of genes requires proteins named transcription factors that bind to specific sequences of DNA and convert or transcribe it into RNA [22]. Since transcription factors are responsible for the expression of a gene, their role in gene fusions has been studied extensively. Their action as enhancer or silencer has been linked by several studies [7] to the expression of chimeric genes making them more or less risky for the human organism. As stated by Rainer Renkawitz [23] "Transcription factors can be defined by their location relative to the transcriptional start site of a particular gene", for this reason this thesis investigated only correlations between transcription factors and the gene at the 5' position of the gene fusions. The table used in the study of this thesis (2.2) was provided by the *ENCODE project*. It gathers data from 1651393 gene-transcription factor associations, this association is defined "by binding of transcription factor near transcription start site of gene"[4]. The table includes 181 different transcription factors and 22819 genes.

Gene attribute matrix

	TF1	TF2	...	TF181
Gene 1	0	0	...	1
...	...	...	...	...
Gene 22819	0	1	...	0

Figure 2.2. Matrix obtained by ENCODE

2.3 micro-RNA

The production of proteins is strictly dependent on the information contained in mRNAs. In this process small non-coding RNAs, defined as microRNAs, can exert the function of regulating the amount of produced protein by binding to mRNAs.[6][24] Incorrect functioning of miRNAs has been linked to a large number of cancers by many studies [25][26][27]. Furthermore, miRNAs that regulate the post-transcriptional gene expressions of chimeras may increase the production of erroneous proteins possibly leading to critical situations for the human organism.

The aim of this thesis is to find a correlation between genes and miRNAs able to characterize the gene fusion for the classification process under the assumption that they may retain information indicative of tumor development. For this purpose, the table provided by *Targetscan* [6] has been used (2.3). Unlike the case of the transcription factors analysis, both genes have been taken into consideration when assigning the values to the features, and in case of ambiguity, only the greatest value of probability was retained.

miRNA matrix

	miRNA1	miRNA2	...	miRNA333
Gene1	0	0.6	...	0
Gene2	0.2	0.1	...	0.34
Gene3	0	0.78	...	0.1
...	...	...	...	...

Figure 2.3. Matrix obtained by Targetscan

2.4 Gene ontology

Gene ontologies provide a collection of semantic descriptions of the functions of genes and gene products, they may be useful when looking for common behaviors and patterns among different genes. In particular, the *Ensembl*[5] has gathered this information along with other specific details concerning the evaluated gene. The biomart tool has been used in this phase to query

information related to each gene belonging to the dataset and annotate the associations between a given gene and a certain number of gene ontologies (2.4). For this purpose the algorithm makes use of the gene ontology terms in the form GO:xxxx. For instance, the research uncovered that ERG is involved in DNA binding, protein binding, cell differentiation, regulation of transcription, etc according to the gene terms assigned by the biomart tool. The hypothesis is that genes with similar functions may exhibit the same oncogenic potential [28][29].

Gene info from biomart			
Gene name	Gene ensg	strand	GO term
Gene1	ENSG01	1	GO:1
Gene1	ENSG01	1	GO:3
Gene2	ENSG02	-1	GO:2
...			...

Figure 2.4. Table obtained by Ensembl

2.5 Initial features

The first five features that I obtained are deducible from the gene name and the breakpoints, they are displayed in figure 2.5:

- percentage of retained 5' gene
- percentage of retained 3' gene
- definition of 5' gene
- definition of 3' gene
- whether the two genes transcribe in the same strand

The decision to include the percentage of the retained gene after to the gene fusion event is related to the assumption that if genes remain whole they may not engage in harmful behaviors but instead keep the same function they

had in the unbroken version. This feature is deduced from the breakpoints and the start and end coordinates of the gene according to the associated genome version.

Moreover, another detail is deduced from the genome version: the strand which the genes transcribe on, namely either the positive one or the negative one. The occurrence of two genes transcribing on different strands is rare but possible, therefore it has been assumed useful to investigate this potential circumstance to gain insight on the possible oncogenic behavior.

Finally, a crucial characteristic has been deduced from the database provided by *Cancermine*[3] that is the collocation of the genes in either one of these categories: driver, tumor suppressor, oncogene, other. The idea is to examine the occurrence of potentially threatening genes in gene fusions in order to classify correctly the dangerous chimeras.

Training set

	% 5p	% 3p	Role 5p	Role 3p	Same strand
Genefusion_1	60	87	1	2	1
Genefusion_2	12	65	3	2	0
...	...	...	...	...	...
Genefusion_1765	99	10	0	1	1

Figure 2.5. The five features deducible from the gene name and the breakpoints

2.6 Feature selection

Before proceeding with the classification it is advisable to reduce the feature set in order to lower the computational cost and improve the reliability of the classifier. Feature selection aims at reducing the number of features by choosing the best subset of features for the specific classification task.[30]

Many algorithms have been designed to perform the feature selection, the one that has been used in this thesis is the random forest[31]. A random forest is an ensemble of decision trees in which each decision tree obtains a classification starting from the training sample at the root level and making

a series of decisions, namely branches. Each decision is related to the value returned by a specific feature, according to the value being greater or smaller than the default value the branch divides into sub-branches, in the end after a certain number of decisions the final label, called leaf, is reached.[32]

In particular, a random forest gathers a defined number of decision trees, it is trained on the training set and can be subsequently pruned to return the most useful features. Pruning refers to reducing the number of branches to the ones closest to the root, in other words, it takes into account a smaller number of features by reducing the decision process to the very first branches. The threshold value can be varied to obtain a larger or smaller subset of features, where the smallest set is made up of the features that have been categorized as most "important" by the algorithm.[33]

2.7 Machine learning

To achieve a reliable classification it is mandatory to choose a suitable machine learning tool. Machine learning (ML) is an implementation of artificial intelligence theories.[34] ML algorithms aim at learning from input data, improving with experience and make decisions [35]. Classifiers in general take in input a set of sample and features to perform decision-related tasks depending on the chosen algorithm and return a label in output.[36] Various kinds of algorithms have been conceived by the scientific community, in this thesis in particular the chosen tools are the support vector machine and the multilayer perceptron.

2.7.1 Support Vector Machine

The support vector machine (SVM) is a non-probabilistic binary classifier, it constructs a hyperplane to divide the set of samples according to the classification into one of the two possible labels.[37] If the classification task is not linear then one of the characteristics of the SVM can be changed, namely the kernel. In fact, some types of kernels can be used to map the samples into a higher-dimensional space.[38] An SVM is characterized by several parameters that need to be tuned according to the specific classification task, these parameters are: kernel, gamma, C (or coefficient) and degree.

The kernel is responsible for the mapping of the samples into the dimensional space, in particular, there are many types of kernels, the most famous

are: linear, rbf, sigmoid and polynomial.[39] A linear kernel works well with linearly separable problems, it divides the bidimensional feature space in two with an imaginary line. The SVM classifier can solve more difficult tasks by using the other kernels that are able to upgrade the dimensional space.[40]

2.7.2 Neural Networks

Neural networks or NN are powerful machine learning tools inspired by the way in which the brain addresses a learning task. NNs are characterized by components named neurons or nodes, which are organized in layers, there are three types of layers:

- input
- output
- hidden

The neuron is composed of a set of connecting links where each link is characterized by:

- a numeric weight
- a function which computes the weighted sum of the inputs
- an activation function which returns the predicted outcome

Typical activation functions are for example step, sign and sigmoid, their goal is to emulate the response of a biological neuron.

During the training phase, each neuron gains knowledge by updating the weights. Supervised learning requires labeled training samples, unlike unsupervised learning. In this thesis, supervised learning was chosen and the classifier is a multilayer perceptron. This kind of neural network is characterized by a feed-forward learning approach meaning that the target label specifies the desired output for a given input, the weights are randomly initialized and iteratively updated in proportion to the error between the desired output and the calculated output.[41][42]
The calculation is made using the following formula:

$$w_{j,i}(t+1) = w_{i,j}(t) + \Delta w_{i,j} \tag{2.1}$$

$$\Delta w_{i,j} = \eta * [t_j - y_j] * \phi(net_j) * x_i \tag{2.2}$$

In equation 2.2 η is the learning rate, $t_j - y_j$ is the error term, net_j is the weighted sum of the inputs, ϕ is the activation function and x_j is the i-th input component.[43]

The number of layers and nodes is central to the issue of classification, for instance, a single-layer network is able to perform linearly separable tasks, whereas more layers allow the classifier to address non-linear problems.

Optimizer

An additional important characteristic of the network is the optimization rule, in other words the way that the total error in the output is minimized.

A typical optimizer is the gradient descent (*sgd*). This optimization model mimics a downhill movement along a curve that presents local minima and global minima. A local minimum is a portion of the curve that reaches a low point and then ascends whereas the global minimum is the lowest portion of the curve.[44]

The *sgd* optimizer updates the weights of the neurons iteratively and the extent of each update depends on the learning rate. For a high value of learning rate the weights are updated of a considerable amount, this would allow the classifier to overcome a local minimum and hopefully reach the global minimum hence the best performances.

A similar optimizer often used in machine learning applications is *Adam* but in this case the value of learning rate is variable during the training.[45]

For this thesis the chosen optimizer was *Adam*.

Overfit

A possible issue that may arise when training a neural network is overfitting. Overfit happens when the classifier learns to perfectly fit the training data obtaining an outcome of 0 misclassifications, this eventuality reduces the generalization ability of the classifier meaning that when presented with unseen data the classifier may perform poorly.[46]

When building a neural network it may be beneficial to drop some of the information learned during the training phase in order to improve the generalization ability of the classifier. To do so the dropout is a useful option, the value is a probability and may be set between 0 (not losing any information)

and 1 (dropping all of the information), in other words, it pushes the model to ignore a portion of the neurons during the training phase. A suitable value of dropout helps in preventing the phenomenon of overfitting.[47]

The training phase of the neural network may include the validation, if this is the case then a set of samples is used to test the performances of the classifier at each epoch and some action may be performed according to the obtained results. The validation set may be either a subset of the training set or a separate set of samples. It is crucial to perform the following testing phase on a set of samples different from the validation set to obtain reliable results. When introducing validation in the model it may be beneficial to implement an early stopping step. Early stopping is designed to prevent overfitting of the model, this method considers either the loss or the accuracy of the validation data and stops the training when the value ceases to decrease or increase respectively.[48]

Cross Validation

Cross validation is a method commonly used to test if the training performances of a classifier are robust.

In this thesis a k-fold cross-validation was implemented. It consists of dividing the samples belonging to the training set into k subsets, k-1 subsets are used for the training while the last fold is used to perform testing. The training and testing processes are repeated for each possible subset (e.g. k times).[49]

K-fold cross-validation returns k sets of performance metrics. In the following sections of this thesis the k values of each metric are averaged and the result is returned as the final cross-validation performance.

Performance evaluation criteria

Once the classifier has completed the training phase, the model is tested on the testing set and the performances may be evaluated using a few different methods.[50]

The most important indicator is the confusion matrix, a table with as many columns and rows as the number of different labels. For a binary classification for instance, the confusion matrix would be a 2 by 2 table displaying the real classes on the top and the predicted classes on the left as shown in figure 2.6

	Real classes		
		True	False
Predicted classes	Positive	62	24
	Negative	8	159

Figure 2.6. Confusion matrix

A few metrics can be deduced from the confusion matrix to help in the evaluation of the performances, they all vary between 0 and 1 where 0 represents the worst classification possible and 1 an ideal classification.

- Accuracy = the most widely used to rapidly assess the efficacy of the classification, the higher the accuracy the higher the number of correctly classified samples

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.3}$$

- Precision = gives an idea of how many real positive samples were detected by the classifier out of all the samples that the classifier labeled as positive. With a high number of false-positive the precision will result poor.

$$Precision = \frac{TP}{TP + FP} \tag{2.4}$$

- Recall = evaluates how many real positive samples were detected by the classifier with respect to all of the samples that were actually positive. With a high number of misclassified negatives the recall, will result in a low value.

$$Recall = \frac{TP}{TP + FN} \tag{2.5}$$

- f1 score = is a weighted average of precision and recall, convenient in case of unbalanced classes

$$f1 = 2 * \frac{recall * precision}{recall + precision} = 2 * \frac{TP}{2TP + FP + FN} \tag{2.6}$$

- False-positive rate = defined as the number of false positives divided by the number of real negative cases

$$FPR = \frac{FP}{FP + TN} \tag{2.7}$$

- AUC = is defined as the area under the ROC curve. The ROC curve can be obtained by simply plotting the false positive rate vs the recall, the AUC would be the area below the ROC curve. The higher the AUC the better the classification results.

Legend:
TP = true positive, FP = false positive, FN = false negative, TN = true negative

Chapter 3

Methods

To classify gene fusions in either oncogenic or non oncogenic gene fusions two machine learning methods were explored in this thesis: SVM and MLP. The classifiers were trained on a thoroughly validated set of 1765 gene fusions that had previously been used for the training of the DEEPrior tool [1]. Of those 1765 samples, 1059 were labeled as oncogenic gene fusions and the remaining 706 as non-oncogenic. Among the positive training samples, some notorious driver gene fusions can be found such as the aforementioned TMPRSS2-ERG and BCR-ABL1 gene pairs.

Both *test set 1* and *test set 2* used by DEEPrior[1] were included in the validation and testing of the classifiers. In particular *test set 2* included 2622 positive samples and 0 negative samples, this set was integrated with Babicenau's database [2] of chimeric gene fusions found in healthy cells to create a balanced testing set. The final testing set, including the 2624 negative samples for a total of 5246 gene fusions, was used to test the performances of the MLP and the SVM. Concerning *test set 1* which contains 156 samples (122 positive and 34 negative samples) it was used in combination with 200 samples of the training set as a validation set for the MLP training phase.

The general pipeline is displayed in figure 3.1 complete of each section explained in the following paragraphs.

3.1 Pipeline

The pipeline that leads to the classification process follows the object-oriented principles.

A separate script calculates the five features integrating data coming from different databases.

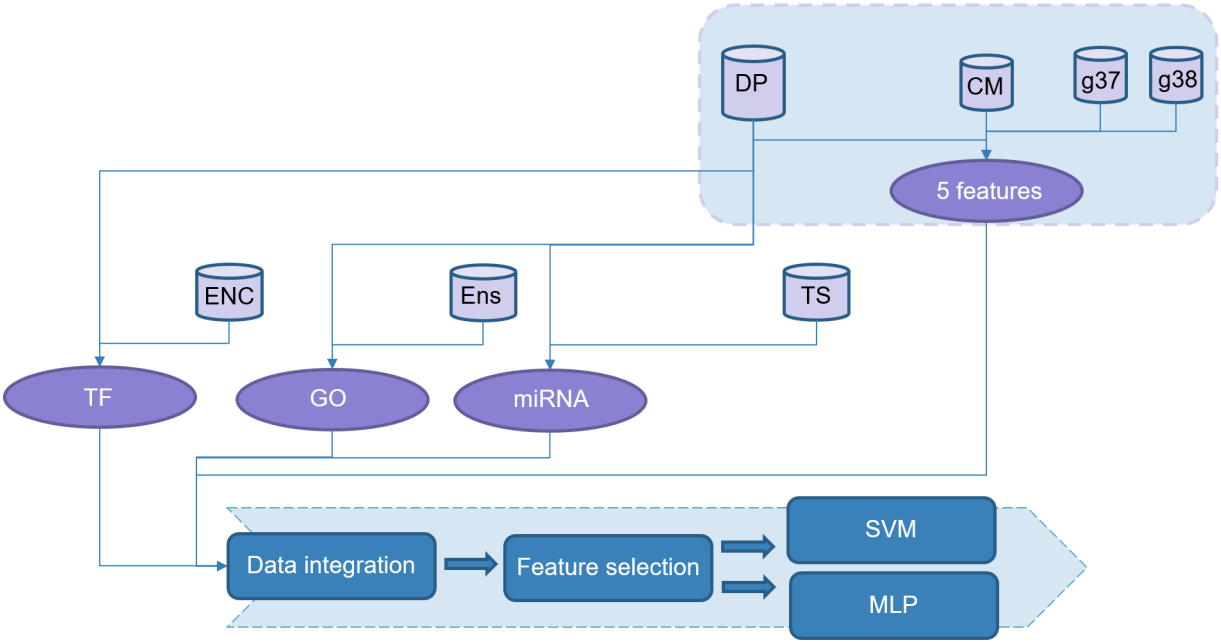


Figure 3.1. Complete pipeline.
Legend: DP=DEEPrior, CM=CancerMine, g37=grch37, g38=grch38, ENC=ENCODE, Ens=Ensembl, TS=TargetScan

To obtain the rest of the features the pipeline exploits the information contained in the other databases including the list of samples analyzed by DEEPrior.

Each of the feature sets is then assembled by the *Data integration* class that returns the complete feature set. Later the set is filtered by the feature selection step.

Finally the SVM and the MLP perform the classification using the input features returned previously by the feature selection.

In the following sections each object reported in figure 3.1 is explained in more detail.

3.1.1 Data integration

The data integration method collects the four feature dataframes that are created by different coding blocks or scripts, specifically:

- the five initial features (percentage of retained genes, belonging strands, the definition given by *Cancermine*)

- the transcription factor matrix
- the gene ontology matrix
- the micro RNA matrix

and returns a set complete of all of the defined features for each gene pair and the label vector. The training, testing and validation sets are characterized by 5644 features at this point. The total set of features includes: five initial features, 181 transcription factors, 332 miRNA families and 5125 gene ontologies.

3.1.2 The five initial features

A separate script computes the five initial features. Concerning the feature related to the belonging strand it was defined as 1 when both genes affected by the fusion event belonged to the same DNA strand and 0 otherwise. The information about the belonging strand was already disclosed by DEEPrior in the original datasets.

For the retained percentage of the two genes involved in the gene fusion, the algorithm exploits the information contained in the corresponding genome version to calculate the value.

- If the gene is at the 5' position and transcribes in the positive direction then the percentage is calculated as: the breakpoint coordinate (bp) minus the start coordinate (s) of the gene divided by the length (L) of the gene

$$\%retained = \frac{bp - s}{L} \tag{3.1}$$

- If the gene is at the 5' position and transcribes in the negative direction then the percentage is calculated as: the end coordinate (e) of the gene minus the breakpoint coordinate (bp) divided by the length (L) of the gene

$$\%retained = \frac{e - bp}{L} \tag{3.2}$$

- If the gene is at the 3' position and transcribes in the positive direction then the percentage is calculated as: the end coordinate (e) of the gene minus the breakpoint coordinate (bp) divided by the length (L) of the gene

$$\%retained = \frac{e - bp}{L} \tag{3.3}$$

- If the gene is at the 3' position and transcribes in the negative direction then the percentage is calculated as: the breakpoint coordinate (bp) minus the start coordinate (s) of the gene divided by the length (L) of the gene

$$\%retained = \frac{bp - s}{L} \tag{3.4}$$

A few cases have led to an inconsistent result (either smaller than 0 or greater than 100) due to possible mistakes in the data, possibly correlated to reporting of the wrong strand of the DNA. Those outliers were therefore automatically dropped to avoid obtaining uncertain results with the classification of these gene fusion pairs.

Finally, the definition given by *Cancermine*[3] (either driver, tumor suppressor, oncogene or other) was simply assigned to each gene belonging to the gene pair and reported in two separate features. This information was extracted using a text-mining tool, as stated by Jake Lever et al. [3] "this machine learning system is then applied to all abstracts in PubMed and all full-text papers in the Pubmed Central Open Access subset". The table is updated regularly to keep track of any new publication.

3.1.3 Feature selection

A random forest algorithm was used to reduce the number of features according to the given threshold value, whilst the number of decision trees was set to 50. The threshold value refers to the importance of the features, this measure of informativeness is returned by the random forest classifier. A threshold equal to 0 would return the features identified as the most useful to make a confident prediction, which would already be a subset of the entire feature set. A higher threshold is related to a smaller number of highly informative features, while a smaller threshold value returned a bigger feature

set. The randomity of the algorithm prevents the definition of a fixed number of obtained features for each threshold value but table 3.1 provides an approximate range of values. From the starting 5644 possible features (comprising of 5125 GOs, 181 TFs, 333 miRNAs and 5 initial features) a number of features ranging from a few thousands to a few dozens can be obtained by varying the threshold value.

The feature selection step was introduced to obtain a number of features that would ensure an acceptable computational time for the training phase and hopefully higher performances in the testing.

Threshold value	Number of selected features
0	2070 - 2090
0.0001	930 - 940
0.001	160 - 170
0.01	15 - 20

Table 3.1. Approximate number of features obtained with the random forest feature selection by using different values of threshold

Two additional options were introduced in this tool to narrow down the features to particular sets and assess the performances of the classifier when using the specified sets of features. The user can choose to use one or more specific sets, for instance the 181 transcription factor features, to perform the classification. Moreover the user can select one or more specific sets and obtain the most informative features among the selected ones using the random forest selection method. For example a combination of sets like the five initial features along with the miRNAs would be reduced to a number of features smaller than 338 according to the selected threshold value.

3.1.4 Transcription factor features

To obtain the features related to the transcription factors the tool first collects the complete set of genes contained in the three analyzed datasets (e.g. the training, testing and validation set). This list is used to obtain a reduced version of the "gene attribute matrix" provided by *ENCODE*[4] from 22819 to 5058 rows. Moreover, it preserves the rows corresponding to the genes at the 5' position and transfers them in a new feature matrix placing them in correspondence to the gene pair of belonging. The result is a new matrix

(3.2) with as many rows as the samples of the corresponding set and 181 columns.

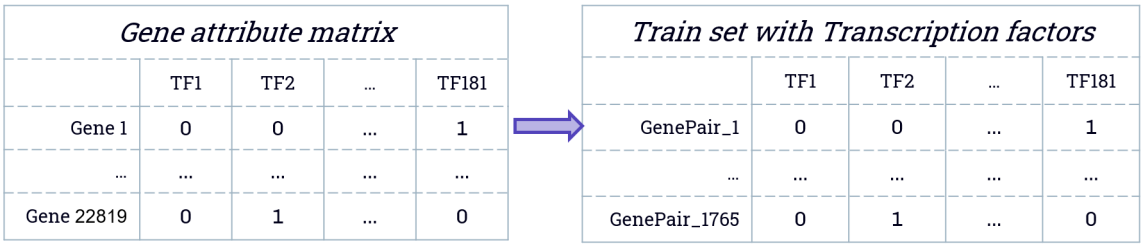


Figure 3.2. Obtaining the TF features for the training set from the gene attribute matrix

3.1.5 Gene ontology features

The first step to retrieve the gene ontology features is uploading the list of the genes included in the "gene attribute matrix", previously used to obtain the transcription factors, on the biomart tool[5]. A query was executed to obtain the gene ontology terms which were then noted in a text file. Three GO matrices were obtained from this file, one for each dataset: train, test, validation.

At this point, the GO matrices vary in length according to the gene ontologies associated with each gene belonging to the analyzed dataset. Then for each set, a new matrix is built with the gene pairs on the rows and all of the gene ontologies occurring in the training set on the columns, when there is a correlation between either one of the genes involved in the gene fusion and one of the gene ontologies then the corresponding cell is marked with a 1.

Finally, the 3 gene ontologies matrices were obtained, each one with 5125 gene ontology features and a number of rows equal to the number of gene fusions in the dataset (3.3).

3.1.6 miRNA features

With a separate algorithm a matrix, similar to the gene attribute matrix for the transcription factors, has been obtained for the miRNA – gene associations using the "predicted targets" file provided by *Targetscan*[6]. Each row of the file contained, amongst other information, a miRNA, a gene name and

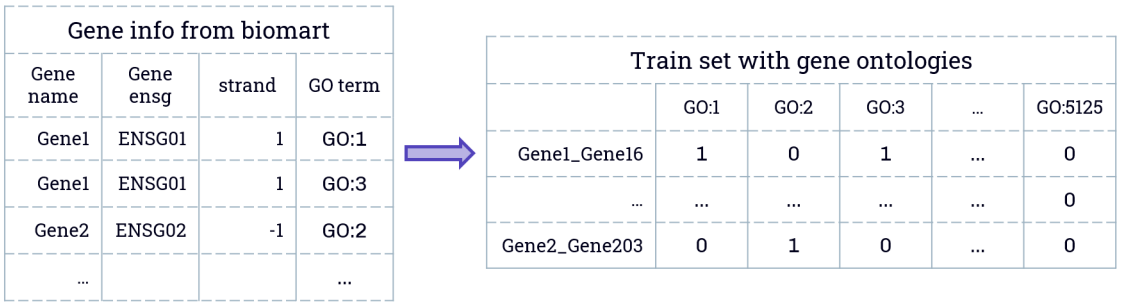


Figure 3.3. Obtaining the GO features from the ensembl table

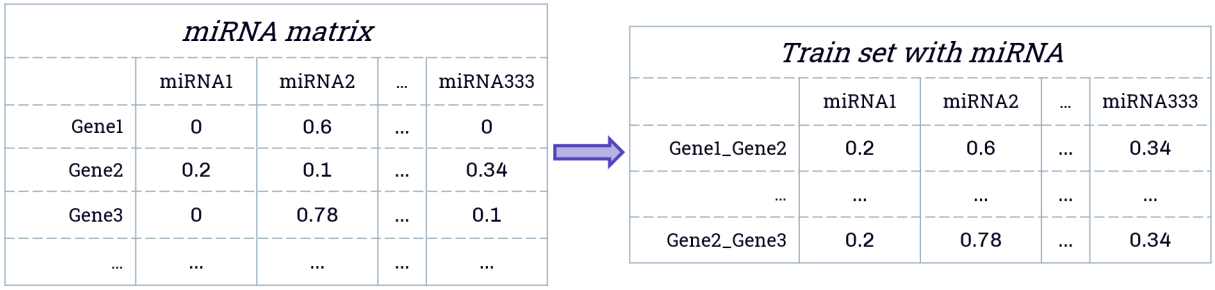
the associated probability of conserved targeting (Pct) for the considered gene - miRNA pair. In general Pct is an indicator of the probability of the involvement of a particular miRNA in the transcription process of a given gene. The Pct ranges between 0 and 1 and it is described by Friedman R. [51] as a Bayesian estimate of the probability that a site is conserved following miRNA targeting rather than any other reason unrelated to miRNA targeting.

The algorithm that I developed identified the greatest Pct for each gene in the file and saved the gene – probability associations in a separate file.

Analogously to what was done to obtain the transcription factor features the final matrices were characterized by the gene fusion samples on the rows and the miRNA families belonging to the training set on the columns (3.4). The cell corresponding to a given miRNA family that was characterized by a probability value for any of the genes belonging to the gene pair was filled with that value. In case both genes involved in the gene fusion were correlated to the same miRNA family with different probability values then only the greatest probability was retained.

3.2 SVM

The parameters of the SVM algorithm were meticulously tuned, four of the most famous kernels were implemented: linear, rbf, sigmoid and polynomial. For each kernel the coefficient was varied in the range: 0.001, 0.01, 0.1, 10 and the values were subsequently narrowed down to the best ones after accurate evaluation of the performances on the testing set. The tuning of the gamma value for the rbf, sigmoid and polynomial kernels, followed the same



miRNA matrix

	miRNA1	miRNA2	...	miRNA333
Gene1	0	0.6	...	0
Gene2	0.2	0.1	...	0.34
Gene3	0	0.78	...	0.1
...	...	...	...	...

Train set with miRNA

	miRNA1	miRNA2	...	miRNA333
Gene1_Gene2	0.2	0.6	...	0.34
...	...	...	...	...
Gene2_Gene3	0.2	0.78	...	0.34

Figure 3.4. Obtaining the miRNA features from the matrix of targetscan

pattern. In particular, gamma was varied in the range: 0.0001, 0.001, 0.01 and the range of values were narrowed down until the optimal combination of coefficient and gamma value was reached. Moreover for the polynomial kernel, one value of degree was used: 2.

3.2.1 MLP

The designed multilayer perceptron model is characterized by 4 layers, for each layer the number of nodes was respectively: 512, 256, 128, 64, and the order of these values was varied to obtain different kinds of architectures and test the performances of the network with each of them. In particular, four architectures were used, the number of nodes for each layer is reported below in the form x-x-x-x. The first number refers to the layer closest to the input layer while the last is the number of nodes that characterizes the layer closest to the output layer.

- 512-256-128-64
- 512-512-256-128
- 512-512-512-512
- 256-512-512-256

The activation functions were varied as well to check which configuration would return the highest performances. Different combinations of the following functions were used:

- Sigmoid

- Tanh
- Relu

Other parameters that have been varied were the learning rate, the dropout and the number of epochs.

- Learning rate: 0.0001, 0.001, 0.01
- Dropout: 0.1, 0.2, 0.3, 0.4
- Number of epochs: 50 or 1000

The chosen optimizer was Adam with decay value equal to $1 * 10^{-6}$.

The MLP can run on two different modalities: by using the validation set and implementing the early stopping or by executing a fixed number of epochs. As already stated, the validation set used by the MLP is a combination of test set 1 and 200 samples belonging to the training set, the goal was to validate on a set that included information coming from a different source with respect to the training set. In this case, the 200 samples used for validation were not considered in the training phase.

The early stopping class is characterized by 50 epochs of patience and considers the validation accuracy to decide at which epoch the model needs to be stopped. In other words when the validation accuracy does not improve for 50 consecutive epochs the training process stops and the model corresponding to the stopping epoch is loaded to perform the subsequent evaluations.

3.3 Oncofuse

To test the robustness of the method it was decided to apply the same model and features to the datasets used by *Oncofuse*[7]. The samples provided by *Oncofuse*, namely the supplementary material, lacked the information related to the breakpoint, for this reason, the retained percentages of the genes could not be calculated but the rest of the features were obtained with no further complication.

The MLP was trained on 524 samples and tested on 21799 samples, as stated by Mikhail Shugay et al[7] the datasets were labeled according to the database they belonged to, in particular for the training:

- 268 positive samples from TICDB

- 56 healthy samples from NORM
- 200 healthy samples from RTH

and for the testing:

- 198 positive samples from NGS1
- 419 positive samples from NGS2
- 1135 positive samples from CHIMERDB2A (higher confidence)
- 2212 positive samples from CHIMERDB2B
- 1366 positive samples from CHIMERDB2C (lower confidence)
- 391 healthy samples from CGC (unbroken genes)
- 16078 healthy samples from REFSEQ (unbroken genes)

The entire set of objects composing the algorithm that trains and tests on the Oncofuse data were analogous to the ones previously described, except for a few adaptations.

The validation set option was adapted for these datasets in the following way: the training set remained unchanged, the validation set included the samples coming from the NGS and the CGC databases and the remaining samples were used for testing.

The number of nodes was reduced to accommodate the fewer number of samples present in this training set, the new values were 128, 64, 32 and 16 respectively for the first, second, third and fourth layer.

An additional method was designed to compare the results obtained by the MLP to the results of figure 2 of the paper written by Shugay M. et al. The obtained bar diagram shows the percentage of driver gene fusions detected by the classifiers (Oncofuse's Bayes Network and the MLP proposed by this thesis). The percentage is calculated for each source of data (e.g. TICDB, CHIMERDB, NGS etc...). For each database i that was analyzed, the percentage of driver gene fusions was calculated as: the number of detected driver gene fusions divided by the total number of samples belonging to database i multiplied by 100.

$$driver\%_i = \frac{TP_i + FP_i}{samples_i} * 100 \tag{3.5}$$

3.4 Additional information

The software was implemented in Python 3.7, the neural network was developed in Keras 2.3.1 and Tensorflow 2.0.0. Computational resources were provided by HPC@POLITO (http://www.hpc.polito.it)[52].

Chapter 4

Results

In this chapter I present the results obtained with the classifiers illustrated previously. For both classifiers k-fold cross-validation was performed on 10 folds. Next, I carried out a tuning phase of the hyper-parameters to identify the best model. In the final testing phase the model that returned the highest performances was determined. The results of the comparison of the best classifier with Oncofuse are illustrated at the end of this chapter.

4.1 SVM cross validation

Firstly, to assess the contribution of each set of features for the classification task a series of training experiments were carried out. Below there is a diagram that summarises the performances of the classifier for each set and for every possible combination of feature sets.

The parameters were not tuned during this estimate of the performances. Therefore it was decided to keep a fixed value for gamma and C equal to respectively 0.001 and 0.4. Although the values were not optimal for the task, the goal of this assessment is to determine which feature set or combination of feature sets classifies better. Gene ontologies were excluded from this evaluation because of the high number of features and will be analyzed subsequently in combination with the random forest technique. Therefore in figure 4.1 the mean of the f1 score over the 10 folds is displayed, each of the 7 bars represents one of the following combinations of feature sets:

- 5 features
- 181 TFs

- 5 features and 181 TFs
- 333 miRNAs
- 5 features and 333 miRNAs
- 181 TFs and 333 miRNAs
- 5 features, 181 TFs and 333 miRNAs

Moreover the bar diagram is divided into 4 sections, in fact the cross validation was performed for each of the following kernels:

- linear
- rbf
- sigmoid
- polynomial

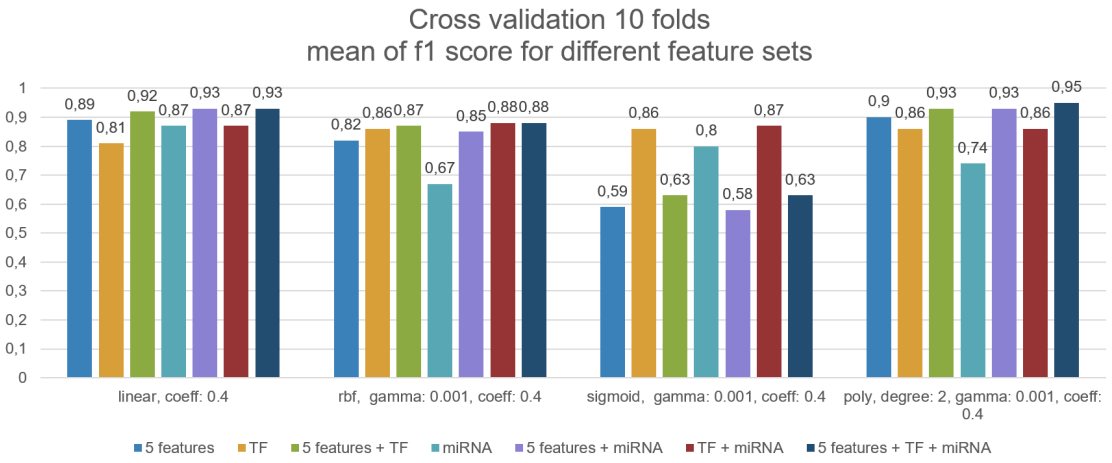


Figure 4.1. Comparison of the performances obtained using different kernels and different subsets of features

The results obtained with the cross-validations (4.1) highlighted that the sigmoid kernel is characterized by the lowest performances with different combinations of features sets. The combination of the three datasets resulted in the highest performances with a mean of f1 score of 0.93 over the 10 folds for the linear SVM and 0.95 for the polynomial kernel.

Once the utility of each feature set had been evaluated it was considered useful to examine the cross-validation results for different number of features by using the random forest selection method.

In the following evaluation, the feature selection technique was a cascade of selection by subset of feature and random forest where the threshold value was fixed and equal to 0.0005. The values for gamma and C are the same as the previous diagram. In the bar diagrams below the cross-validation results are displayed for each of the kernels, respectively:

- linear kernel figure 4.2
- gaussian kernel figure 4.3
- sigmoid kernel figure 4.4
- polynomial kernel figure 4.5

The number of features used for each combination of feature sets was determined by the random forest feature selection method. The diagrams of figure 4.2, 4.3, 4.4 and 4.5 explore the cross-validation performances obtained for each of the following combinations:

- 271 GOs
- 172 TFs
- 252 miRNAs
- 5 features + 171 TFs
- 5 features + 231 GOs
- 5 features + 236 miRNAs
- 178 GOs + 109 TFs
- 204 miRNAs + 157 TFs
- 102 miRNAs + 193 GOs
- 5 features + 175 GOs + 112 TF
- 5 features + 146 TFs + 191 miRNAs
- 5 features + 172 GOs + 109 miRNAs

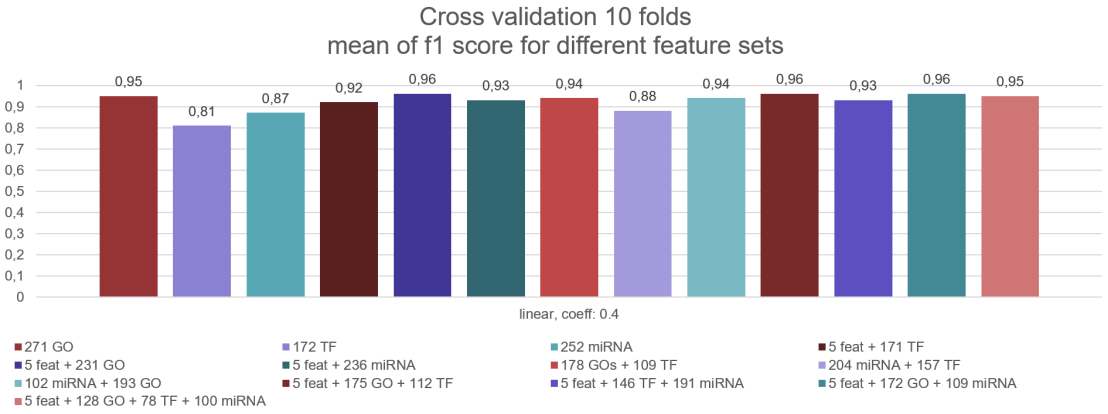


Figure 4.2. Comparison of the performances obtained using the linear kernel and different subsets of features. The number of features was reduced using the random forest feature selection

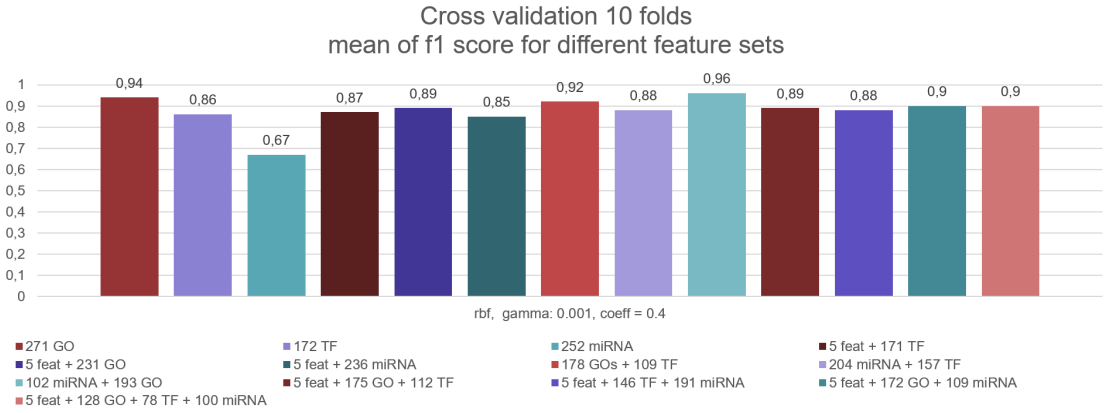


Figure 4.3. Comparison of the performances obtained using the gaussian kernel and different subsets of features. The number of features was reduced using the random forest feature selection

- 5 features + 128 GOs + 78 TFs + 100 miRNAs

Overall the highest performances were reached when using the complete set of features, with a maximum of 0.96 mean of f1 with the polynomial kernel and 0.95 with the linear coefficient. The sigmoid kernel appears to be again the one that reaches the lowest results.

The random forest feature selection greatly reduced the number of gene ontologies from 5025 to 271, this number lowers furtherly when the GOs are combined with other features. Nevertheless, the GO seems to be the set of

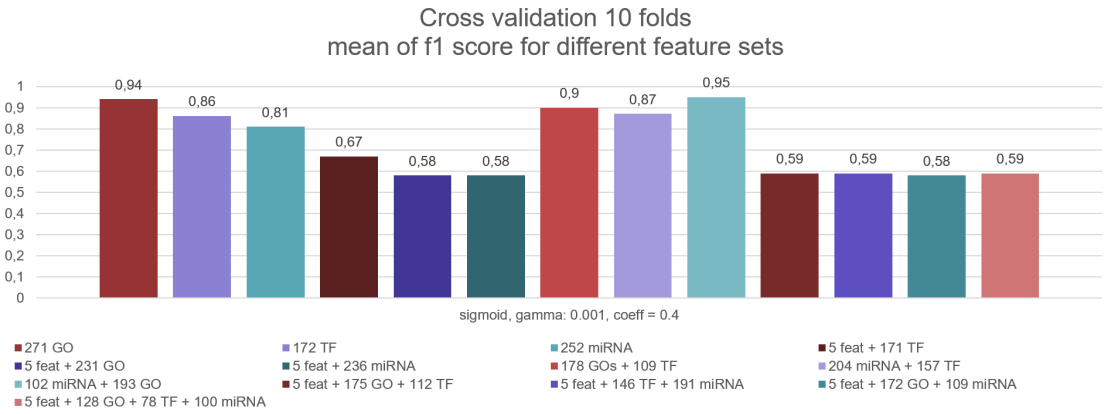


Figure 4.4. Comparison of the performances obtained using the sigmoid kernel and different subsets of features. The number of features was reduced using the random forest feature selection

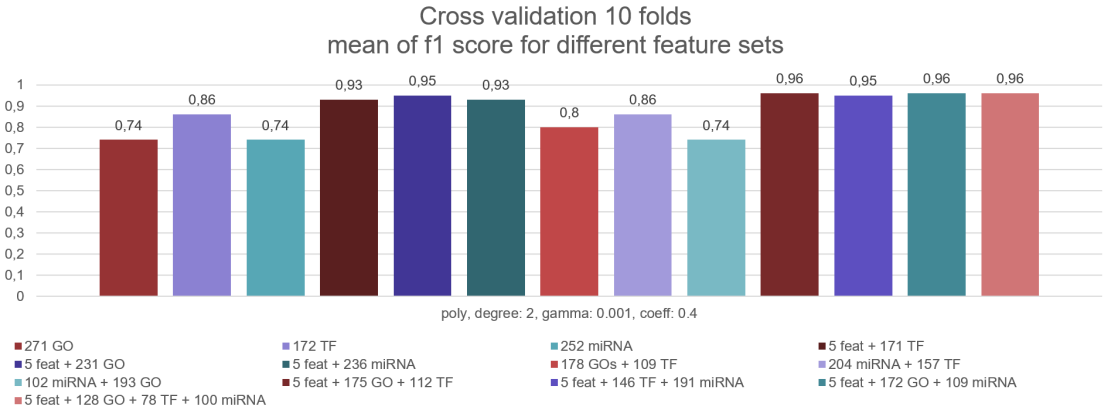


Figure 4.5. Comparison of the performances obtained using the polynomial kernel and different subsets of features. The number of features was reduced using the random forest feature selection

features that returns the highest results when evaluating its performances on the training set with 0.95 as mean of f1 with the linear kernel, 0.94 with the gaussian, 0.94 with the sigmoid and 0.74 with the polynomial.

The performances of the linear kernel are generally high, between 0.81 and 0.95 along with the results obtained with the polynomial kernel, between 0.74 and 0.96.

Once the utility of the complete set of features had been settled the parameters were finely tuned to obtain the best possible results. For the following evaluations of the performances the entire set of features was used with the random forest feature selection method. The threshold value for the random forest was equal to 0.0005, once the set of features was defined it was preserved and used for the rest of the evaluations for consistency of results.

Figure 4.6 shows the mean of accuracy for the linear kernel and the following coefficient values:

- 0.001
- 0.01
- 0.1
- 10

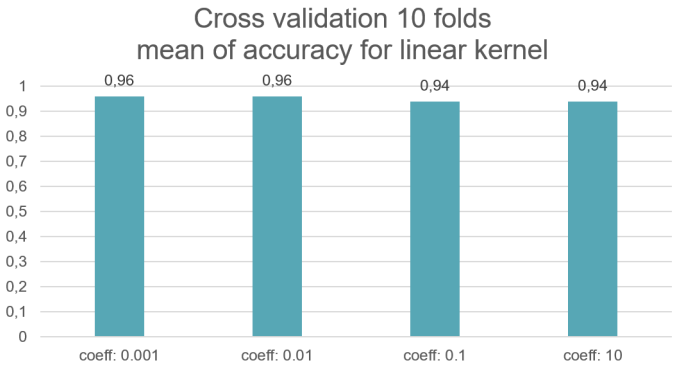


Figure 4.6. Cross validation results for linear kernel with coeff = [0.001, 0.01, 0.1, 10]

Figure 4.7 shows the mean of accuracy for the gaussian kernel, the same coefficient values used previously and the following gamma values:

- 0.001
- 0.01
- 0.1
- 10

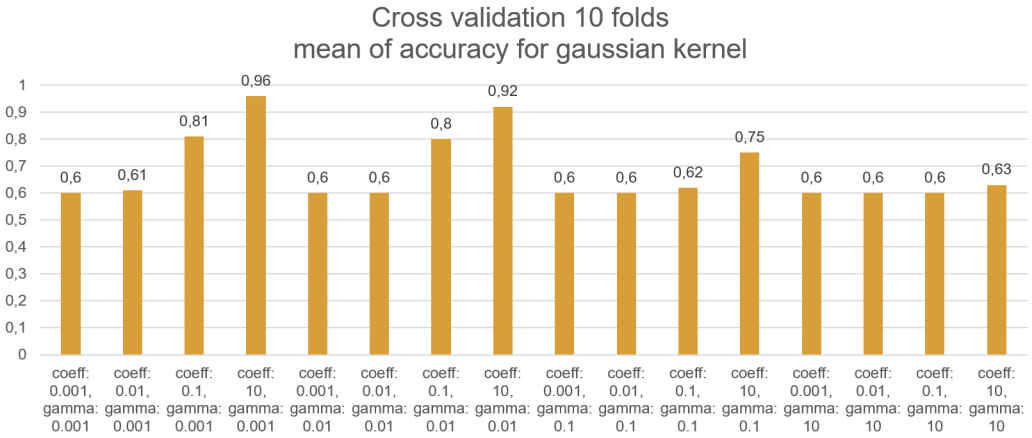


Figure 4.7. Cross validation results for rbf kernel with coeff = [0.001, 0.01, 0.1, 10] and gamma = [0.001, 0.01, 0.1, 10]

Figure 4.8 displays the mean of accuracy for the same gamma and coefficient values used for the rbf kernel.

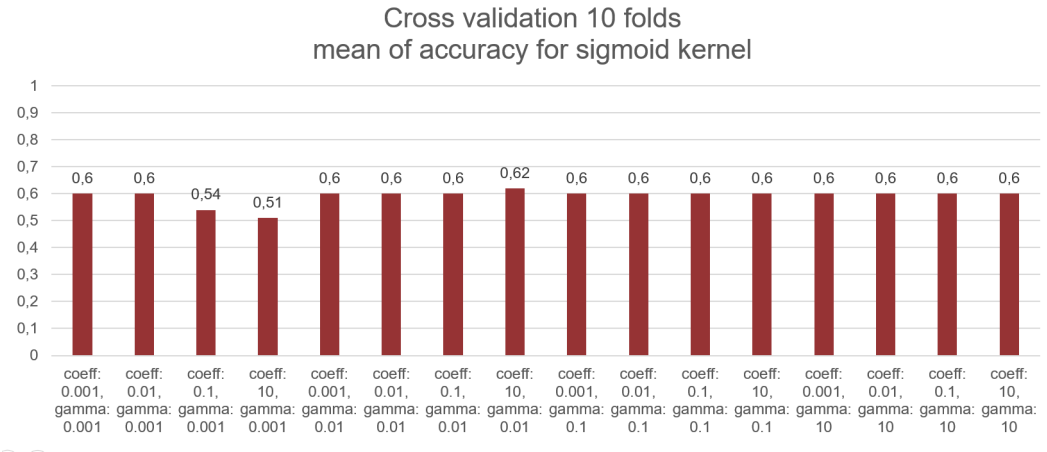


Figure 4.8. Cross validation results for sigmoid kernel with coeff = [0.001, 0.01, 0.1, 10] and gamma = [0.001, 0.01, 0.1, 10]

Figure 4.9 displays the mean of accuracy for the same gamma and coefficient values used previously and for degree equal to 2.

Since the sigmoid kernel (4.8) was the one that confirmed to have the lowest performances with a 0.6 mean of accuracy it was excluded from the

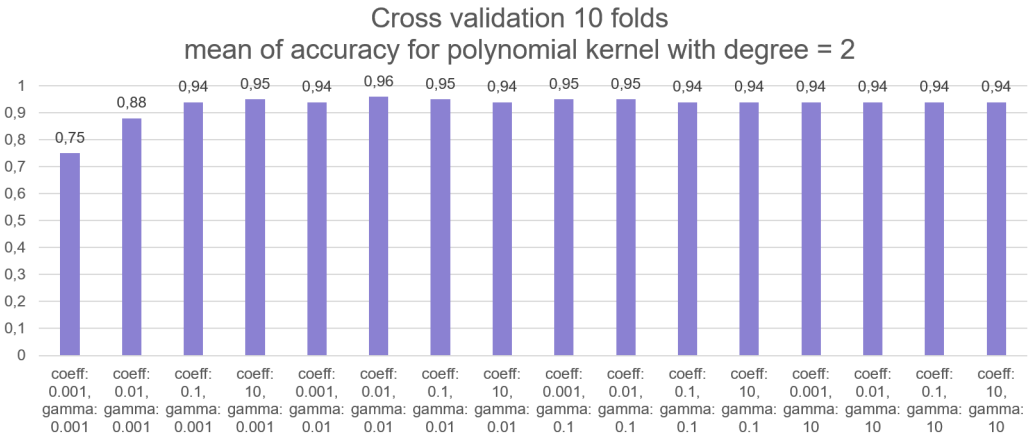


Figure 4.9. Cross validation results for polynomial kernel with coeff = [0.001, 0.01, 0.1, 10], gamma = [0.001, 0.01, 0.1, 10] and degree = 2

following step of testing. Moreover, to each of the configurations with sigmoid kernel characterized by a 0.6 value of mean of accuracy the confusion matrix of figure 4.10 was associated.

Confusion matrix train

	True	False
Positive	0	597
Negative	0	882

Figure 4.10. Confusion matrix of the training set obtained with the SVM model characterized by sigmoid kernel, coeff = 0.001 and gamma = 0.001

Furthermore, the rbf kernel configurations (4.7) characterized by a low value of coefficient and high value of gamma consistently performed worse than other combinations of values of lambda and gamma with about 0.6 mean of accuracy. Therefore, the ranges of values of lambda and coefficient were reduced respectively to: [0.01, 0.1, 10] and [0.001, 0.01, 0.1] for the following testing phase.

The polynomial kernel (4.9) seems to fit well the training data with a prevalence of high results for high values of coefficient and low values of gamma with a 0.95 mean of accuracy, consistently with the previous observation. The same can be said for the configurations obtained with the linear kernel (4.6).

4.2 SVM testing

The different possible configurations for the SVM classifier were tested on the previously defined test set and the feature selection method was the random forest with threshold value equal to 0.0005. For this test 288 features were selected among which:

- The percentage of retained gene at the 5' and at the 3' position along with the definition given by *Cancermine* for both genes
- 121 gene ontologies
- 93 miRNAs
- 70 transcription factors

The performances of the classifiers on the train set and on the test set are shown in figure 4.11 and 4.12. Figure 4.11 illustrates the obtained AUC values for linear, gaussian and polynomial kernel with the following ranges of parameters:

- range of values for gamma = [0.001, 0.01, 0.1]
- range of values for coefficient = [0.01, 0.1, 10]

On the other hand figure 4.12 displays the corresponding accuracy values for the same parameters.

The performances obtained with the model characterized by the rbf kernel, coefficient 0.1 and gamma 0.001 were the highest. The AUC for this model was 0.86 on the training set and 0.73 on the test set. The corresponding accuracy values are equal to 0.86 on the training set and 0.75 on the testing set. The polynomial kernel returns similar results with gamma 0.001 and coefficient 0.01.

The second testing phase was characterized by the same threshold value for the random forest. This time 275 features were retained of which:

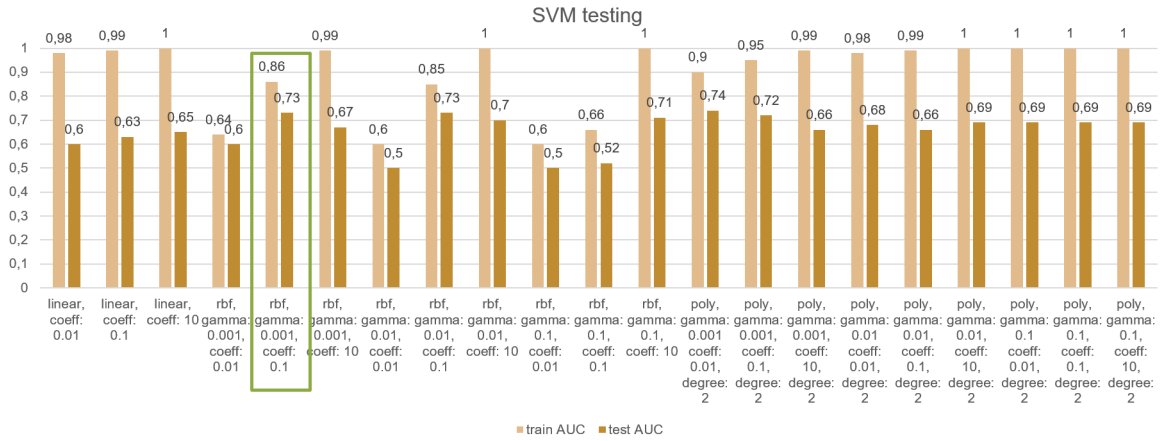


Figure 4.11. AUC values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.01, 0.1, 10] and the range of gamma values was: [0.001, 0.01, 0.1]. The degree for the polynomial kernel was equal to 2.

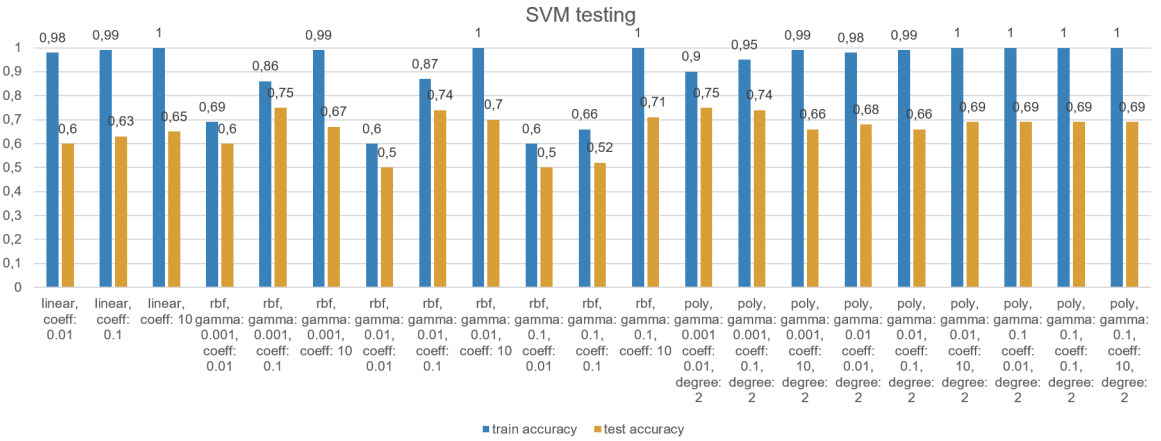


Figure 4.12. Accuracy values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.01, 0.1, 10] and the range of gamma values was: [0.001, 0.01, 0.1]. The degree for the polynomial kernel was equal to 2.

- 5 initial features
- 123 gene ontologies
- 86 miRNAs
- 61 transcription factors

Figure 4.13 illustrates the AUC values obtained for the same ranges of parameters used previously whilst figure 4.12 displays the corresponding accuracy values.

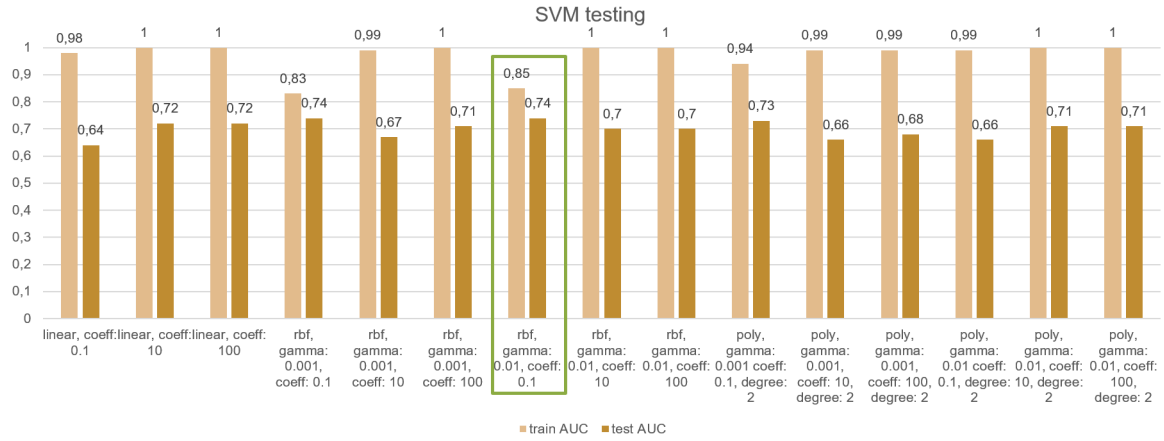


Figure 4.13. AUC values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.1, 10, 100] and the range of gamma values was: [0.001, 0.01]. The degree for the polynomial kernel was equal to 2.

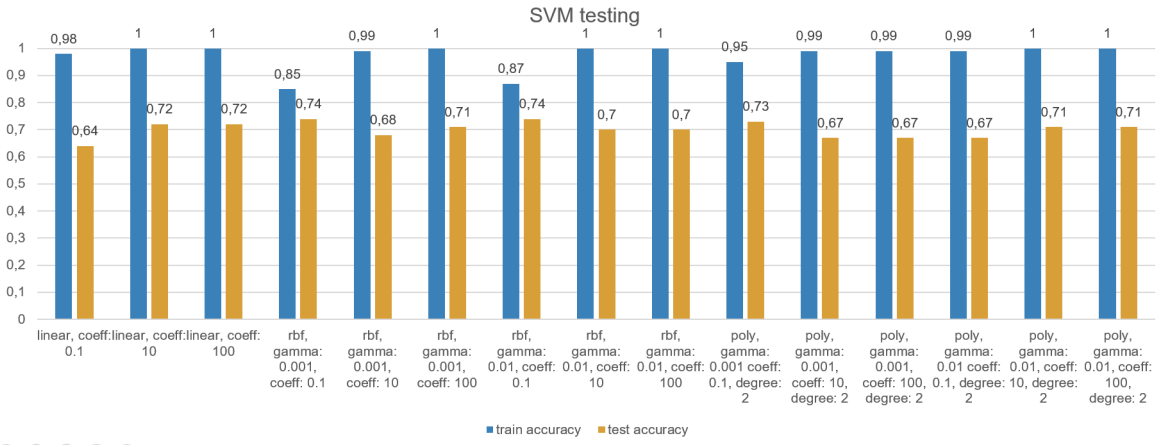


Figure 4.14. Accuracy values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.1, 10, 100] and the range of gamma values was: [0.001, 0.01]. The degree for the polynomial kernel was equal to 2.

Concerning the SVM configuration that reached the highest results on the testing set, using the new set of 275 features, it was characterized by the

following parameters:

- kernel = rbf
- gamma = 0.01
- coeff = 0.1

The performance metrics obtained for this configuration of parameters are displayed in figure 4.15.

	Train	Test
Accuracy	0,87	0,74
Precision	0,83	0,68
Recall	0,98	0,91
AUC	0,85	0,74

Figure 4.15. Performance metrics for the model characterized by the following parameters: rbf, gamma = 0.01, coeff = 0.1

Furthermore the corresponding confusion matrices are displayed in figure 4.16.

Confusion matrix train

	True	False
Positive	488	195
Negative	17	979

Confusion matrix test

	True	False
Positive	1509	1111
Negative	239	2319

Figure 4.16. Confusion matrices for train and test set. The parameters of the SVM model were: rbf, gamma = 0.01, coeff = 0.1

If the threshold value for the random forest feature selection method increases then the number of features lowers. In particular, with a threshold equal to 0.001, the number of selected features was 176 of which:

- 5 initial features
- 71 gene ontologies
- 44 miRNAs
- 56 transcription factors

And the obtained performances for training and testing using this smaller set of 176 features are plotted in figure 4.17 and 4.18. The range of values for gamma and coefficient used to obtained the AUC and accuracy values displayed respectively in figure 4.17 and 4.18 were:

- range of values for gamma = [0.001, 0.01]
- range of values for coefficient = [0.1, 10, 100]

Similarly to what was reported previously, both the train and test set performances were showed in the bar diagrams.

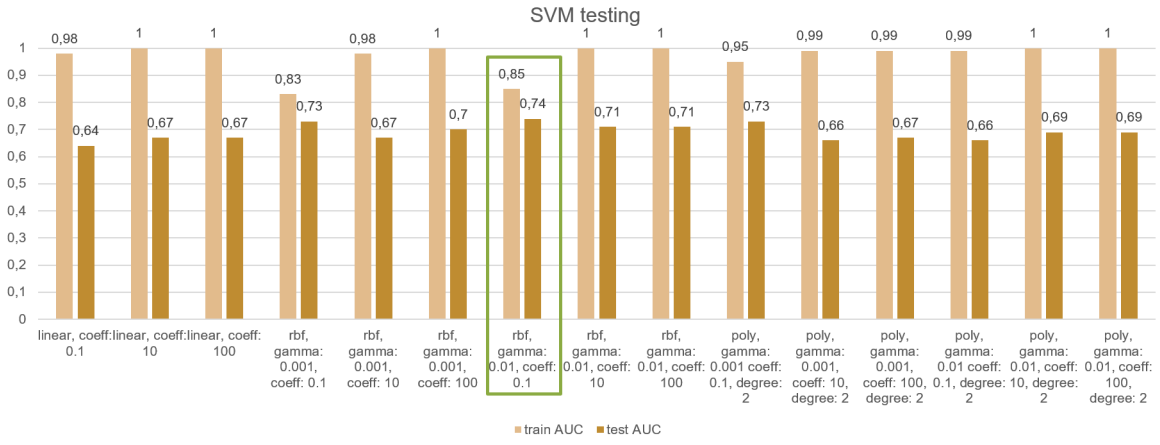


Figure 4.17. AUC values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.1, 10, 100] and the range of gamma values was: [0.001, 0.01]. The degree for the polynomial kernel was equal to 2.

It was found that when reducing the number of features the results were analogous. In fact, the best configuration was the same as the previous one,

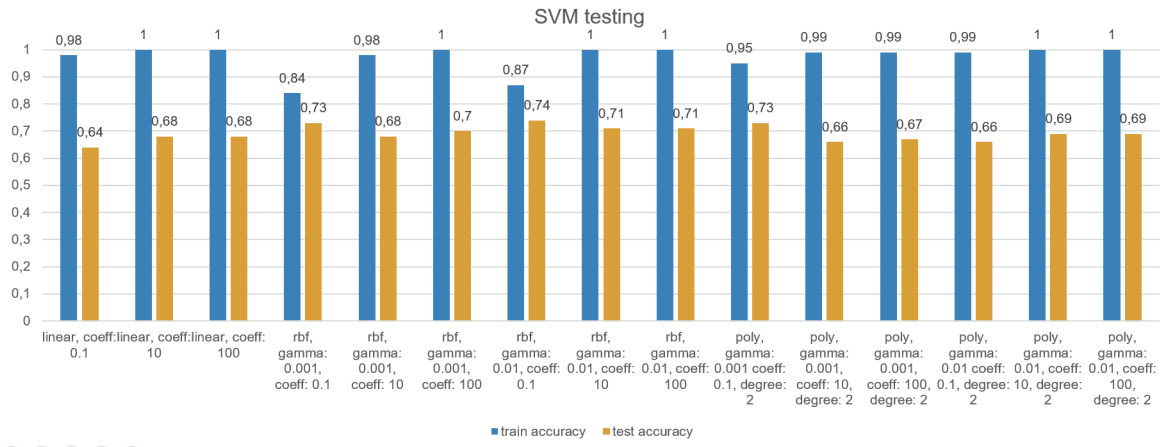


Figure 4.18. Accuracy values for the train and test sets obtained with different SVM models. The kernels were: linear, rbf and polynomial while the range of coefficient values was: [0.1, 10, 100] and the range of gamma values was: [0.001, 0.01]. The degree for the polynomial kernel was equal to 2.

with the following parameters: kernel = rbf, gamma = 0.01, coeff = 0.1. The confusion matrix of this model and the metrics are reported respectively in figure 4.19 and figure 4.20. The results for the training set were illustrated as well as the results for the testing set.

Confusion matrix train

	True	False
Positive	496	187
Negative	24	972

Confusion matrix test

	True	False
Positive	1527	1093
Negative	236	2322

Figure 4.19. Confusion matrices for train and test set. The parameters of the SVM model were: rbf, gamma = 0.01, coeff = 0.1

Subsequently a phase of fine-tuning of the parameters was performed. The results obtained for the gaussian kernel are displayed below for the AUC values returned by the cross-validation phase and the AUC values returned

	Train	Test
Accuracy	0,87	0,74
Precision	0,84	0,68
Recall	0,98	0,91
AUC	0,85	0,74

Figure 4.20. Performance metrics for the model characterized by the following parameters: rbf, gamma = 0.01, coeff = 0.1

when testing on the test set. The selected features were the same for each of the following evaluations, namely 178 of which:

- 5 initial features
- 76 gene ontologies
- 38 miRNAs
- 54 transcription factors

Firstly the two parameters were varied in a range of values close to the best solution obtained previously:

- range of values for gamma = [0.001, 0.005, 0.01, 0.05, 0.1]
- range of values for coefficient = [0.01, 0.05 ,0.1, 5, 10]

Figure 4.21 shows the AUC values obtained during the first tuning phase. The highest test AUC reached was equal to 0.77 and was obtained for gamma = 0.05 and coefficient = 5.

For the second tuning phase, the highest test AUC was equal to 0.77 and was obtained for gamma = 0.05 and coeff = 2. Figure 4.22 reports the obtained AUC values for the cross-validation phase and the testing phase. In this second tuning phase the possible values for the parameters were the following:

- range of values for gamma = [0.005, 0.01, 0.05]

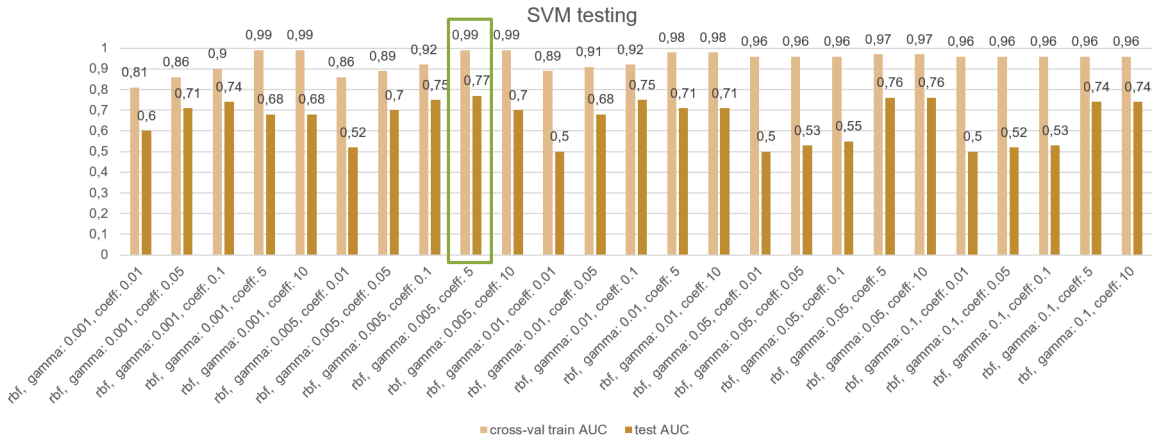


Figure 4.21. AUC values for gamma = [0.001, 0.005, 0.01, 0.05, 0.1], coeff = [0.01, 0.05 ,0.1, 5, 10]

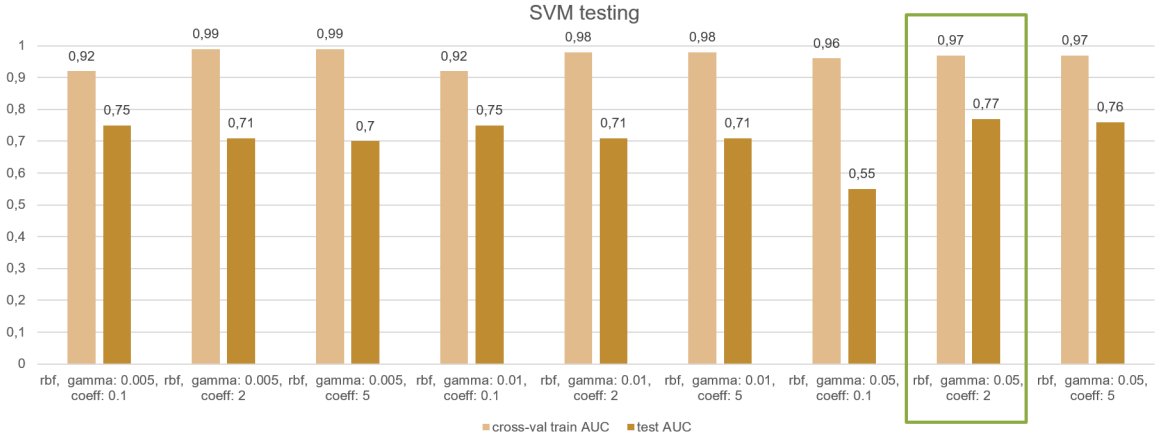


Figure 4.22. AUC values for gamma = [0.005, 0.01, 0.05], coeff = [0.1, 2, 5]

- range of values for coefficient = [0.1, 2, 5]

Figure 4.23 illustrates the AUC values obtained for the third tuning phase. In this case the possible values for the parameters were the following:

- range of values for gamma = [0.04,0.05,0.06]
- range of values for coefficient = [0.1, 2, 5]

The highest test AUC value was equal to 0.79 and it was obtained for gamma = 0.04 and coeff = 0.5.

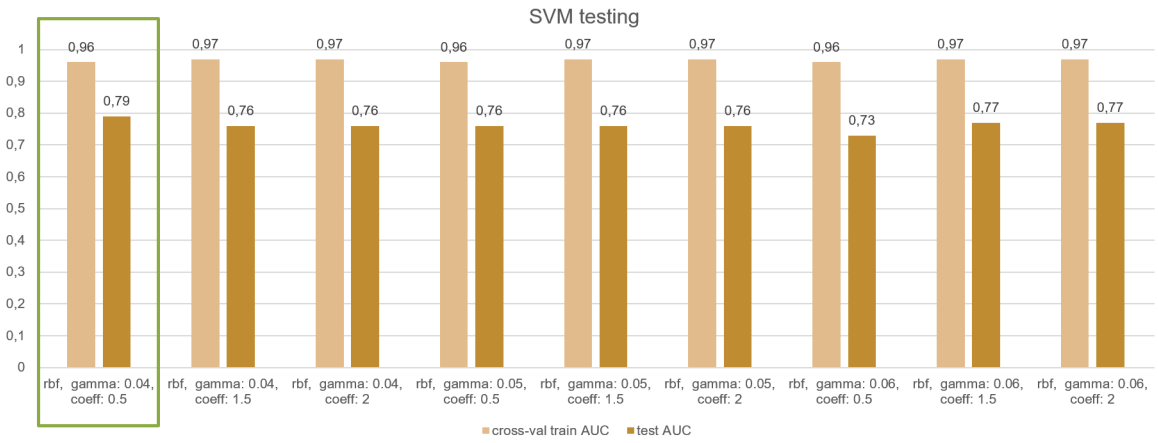


Figure 4.23. AUC values for gamma = [0.04,0.05,0.06], coeff = [0.5,1.5,2]

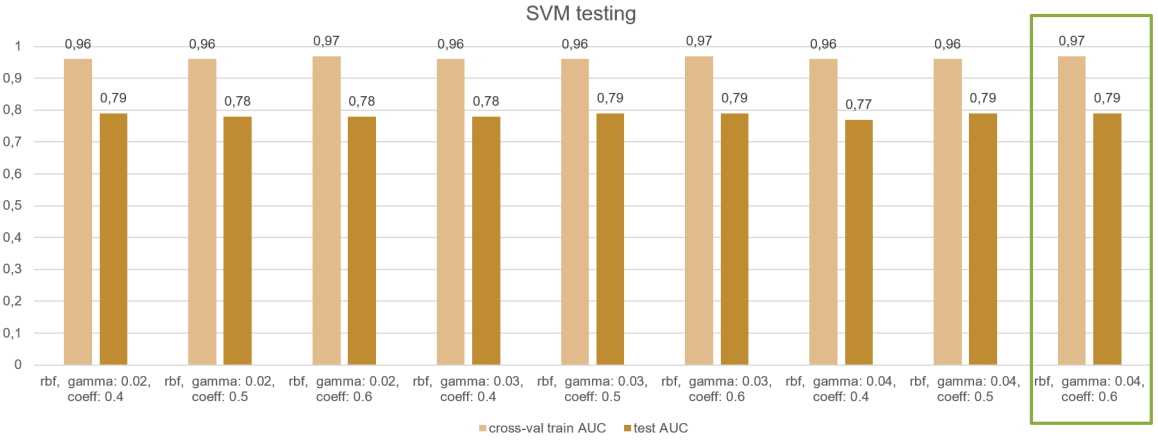


Figure 4.24. AUC values for gamma = [0.02,0.03,0.04], coeff = [0.4,0.5,0.6]

In the fourth tuning phase the highest test accuracy was reached, namely 0.794. The corresponding AUC values are displayed in figure 4.24 and the possible values for the parameters were:

- range of values for gamma = [0.02,0.03,0.04]
- range of values for coefficient = [0.4,0.5,0.6]

In this case, the highest test accuracy value was returned by two different combinations of parameters.

- gamma = 0.02, C = 0.5, precision = 0.788, recall = 0.797, AUC = 0.780
- gamma = 0.04, C = 0.6, precision = 0.756, recall = 0.860, AUC = 0.795

Since the latter configuration was characterized by the highest recall and AUC value the parameters were varied around those values for the last tuning phase.

Figure 4.25. AUC values for gamma = [0.035,0.04,0.045], coeff =[0.55,0.6,0.65]

In the fifth and final tuning phase the highest performances for the SVM were reached. Figure 4.25 displays the cross-validation train AUC and the test AUC obtained for the following ranges of parameters:

- range of values for gamma = [0.035,0.04,0.045]
- range of values for coefficient = [0.55,0.6,0.65]

In particular, the accuracy values are summarised in the diagram of figure 4.26.

Finally the highest performances were reached.

To sum up, in the beginning different SVM models were tested and the results highlighted that the SVM characterized by the gaussian kernel was the one that returned the highest performances.

When testing on the test set with different sets of features the chosen best rbf model resulted in a AUC value higher than 0.73 each time (figures 4.11, 4.13, 4.17).

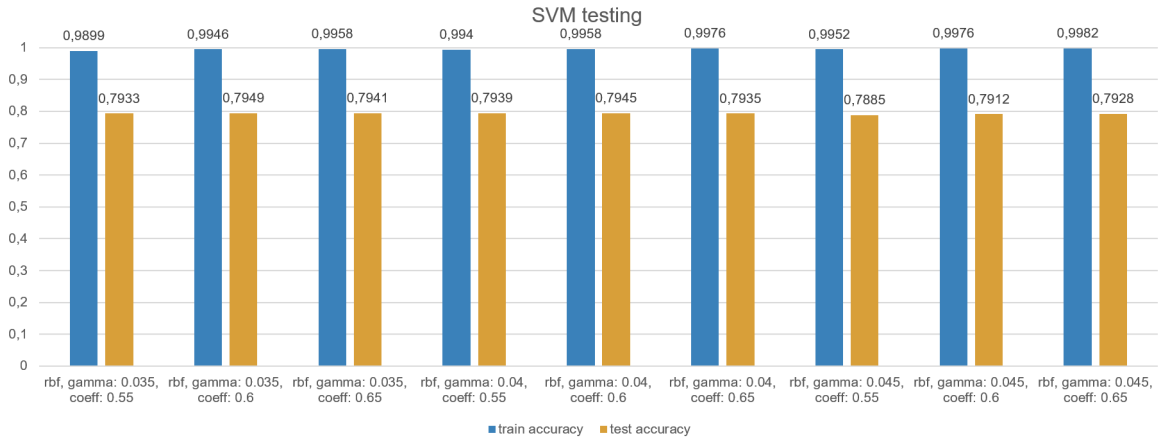


Figure 4.26. Accuracy values for train and test set in the final tuning phase

Furthermore the fine-tuning of the parameters led to the best possible SVM model, increasing the obtained test AUC at each step of the tuning phase from a starting value of 0.77 (figures 4.21, 4.22) up to 0.79 (figures 4.23, 4.24, 4.25).

After these evaluations the best SVM model was identified and the optimal configuration of parameters to obtain the highest performances is the following:

- Kernel = rbf
- Gamma = 0.04
- Coeff = 0.6

This model returned the highest accuracy, equal to 79,45% . Furthermore this configuration of parameters returned a recall equal to 0.8604 and precision equal to 0.7568. The AUC value was high as well and equal to 0.7952. The confusion matrices for the chosen SVM model are displayed in figure 4.27 and 4.28

4.3 MLP cross validation

An analogous evaluation was performed for the multilayer perceptron classifier. The parameters for the cross-validation phase were:

Confusion matrix train		
	True	False
Positive	676	7
Negative	0	996

Confusion matrix test		
	True	False
Positive	1913	707
Negative	357	2201

Figure 4.27. Confusion matrix for the best SVM model: kernel = rbf, gamma = 0.04 and coeff = 0.6

	Train	Test
Accuracy	0,99	0,79
Precision	0,99	0,76
Recall	1	0,86
AUC	0,99	0,79

Figure 4.28. Performances of the best SVM model: kernel = rbf, gamma = 0.04 and coeff = 0.6

- 4 layers with number of nodes = 512-256-128-64
- Activation functions = four sigmoids
- Learning rate = 0.001
- Number of training epoch = 30
- Dropout = 0.2 at each layer

The accuracy for the cross validation on each feature set is displayed in figure 4.29

Similarly to what has been done to assess the training performances of the SVM, the gene ontologies were excluded from the previous evaluation because of the high number of features and are analyzed below in combination with the random forest technique (figure 4.30).

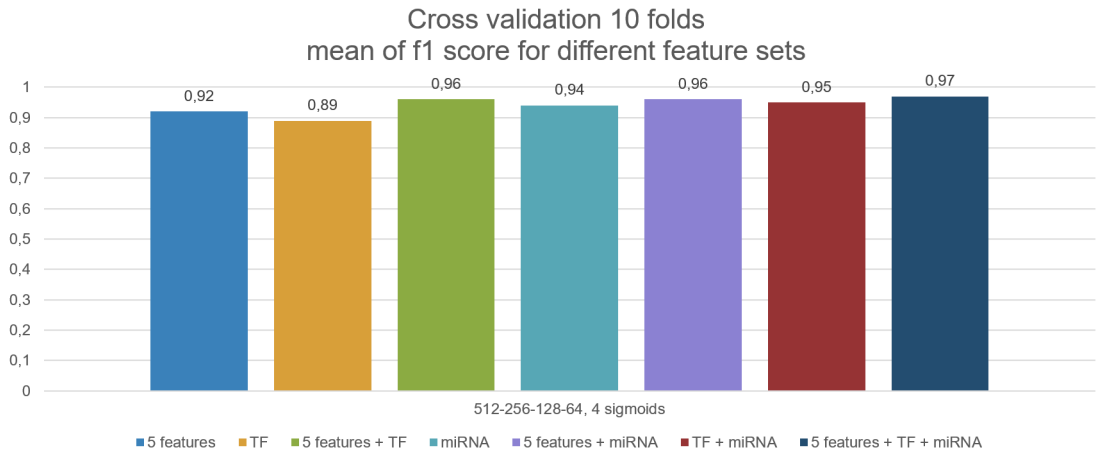


Figure 4.29. Comparison of the performances obtained using different subsets of features

With the same threshold as before for the random forest feature selection method (e.g. 0.0005) different feature sets were chosen and the training performances of the MLP classifier were calculated. Other parameters of the MLP such as learning rate and dropout were kept equal.

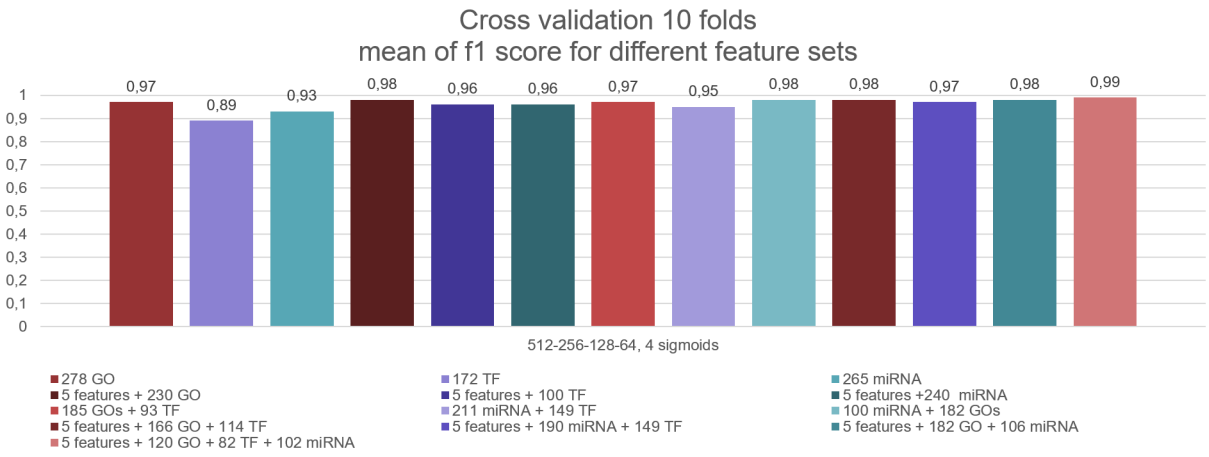


Figure 4.30. Comparison of the performances obtained using different subsets of features. The number of features was reduced using the random forest feature selection

Analogously to what has been observed for the SVM, the complete set of features reaches the highest performances in the cross-validation phase. In particular, the mean of the f1 score over the 10 folds resulted equal to 0.99

when the selected set of features involved each of the available elements.

4.4 MLP testing

The tuning of the parameters has been performed for each configuration of the model. Firstly, the number of epochs was fixed and equal to 50 for consistency of the results. The number of features was fixed as well and equal to 309 of which:

- 5 initial features
- 136 gene ontologies
- 95 miRNAs
- 73 transcription factors

The chosen configuration for the following evaluation (figure 4.31) was characterized by 4 equal layers with number of nodes equal to 512 for each layer. The activation functions for each layer were respectively: relu, sigmoid, relu and sigmoid.

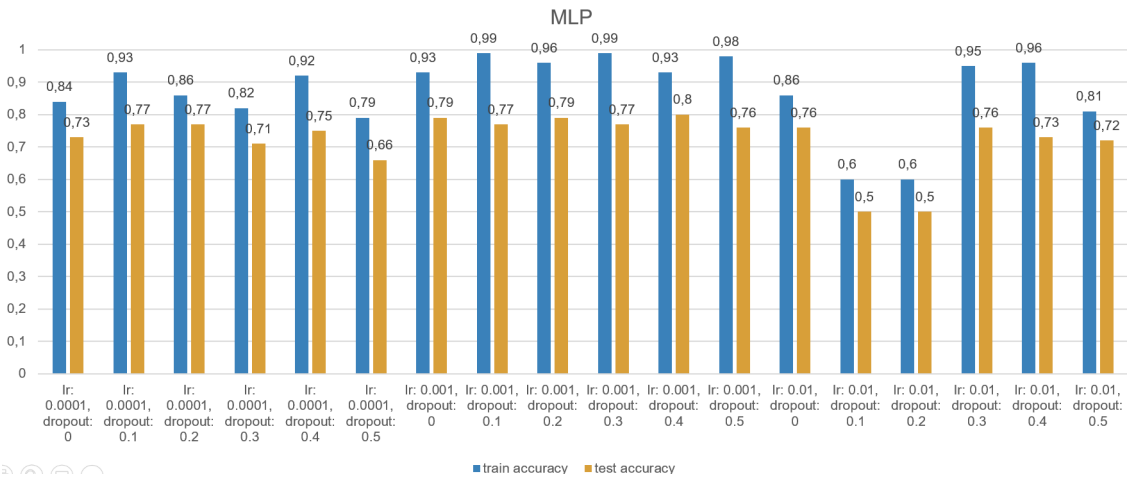


Figure 4.31. Accuracy values for the train and test sets obtained with different MLP models. The range of the learning rate was = [0.0001, 0.001, 0.01] and the range for the dropout value was = [0, 0.1, 0.2, 0.3, 0.4, 0.5]

For learning rate equal to 0.001 and dropout 0.4 the testing accuracy was equal to 0.8. The following evaluation (figure 4.32) tuned more finely by

varying the values in the range of minimum 0.002, and maximum 0.008 for the learning rate and either 0.3 or 0.4 for the dropout. The validation set was used as a combination of 200 fixed samples from the training set and 151 samples from test set 1, the early stopping was implemented with patience equal to 50.

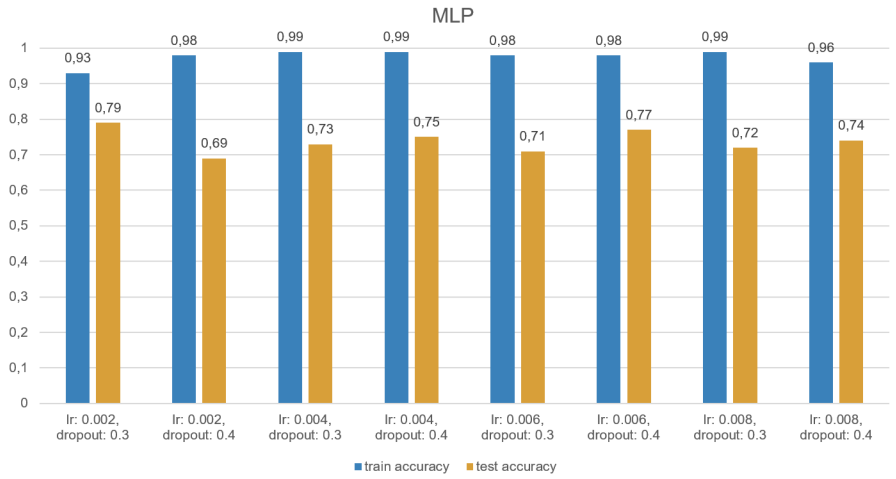


Figure 4.32. Accuracy values for the train and test sets obtained with different MLP models. The range of the learning rate was = [0.002, 0.004, 0.008] and the range for the dropout value was = [0.3, 0.4]

The configuration characterized by learning rate equal to 0.002 and dropout equal to 0.3 returned a 0.79 accuracy on the test set, therefore this configuration was chosen to perform the last evaluation. The performances of the MLP were obtained for different number of nodes and activation functions as described below.

- Number of nodes:
  - 512-512-512-512
  - 256-512-512-256
  - 516-256-128-64
  - 512-512-256-128
- Activation functions:
  - 4 sigmoids
  - 4 tanhs

– 4 relus
– tanh-sigmoid-tanh-sigmoid
– relu-sigmoid-relu-sigmoid
– sigmoid-relu-sigmoid-relu
– sigmoid-tanh-sigmoid-tanh
– relu-tanh-relu-tanh
– tanh-relu-tanh-relu

The accuracy results for the cross-validation phase and the test phase are displayed in figure 4.33 and 4.34.

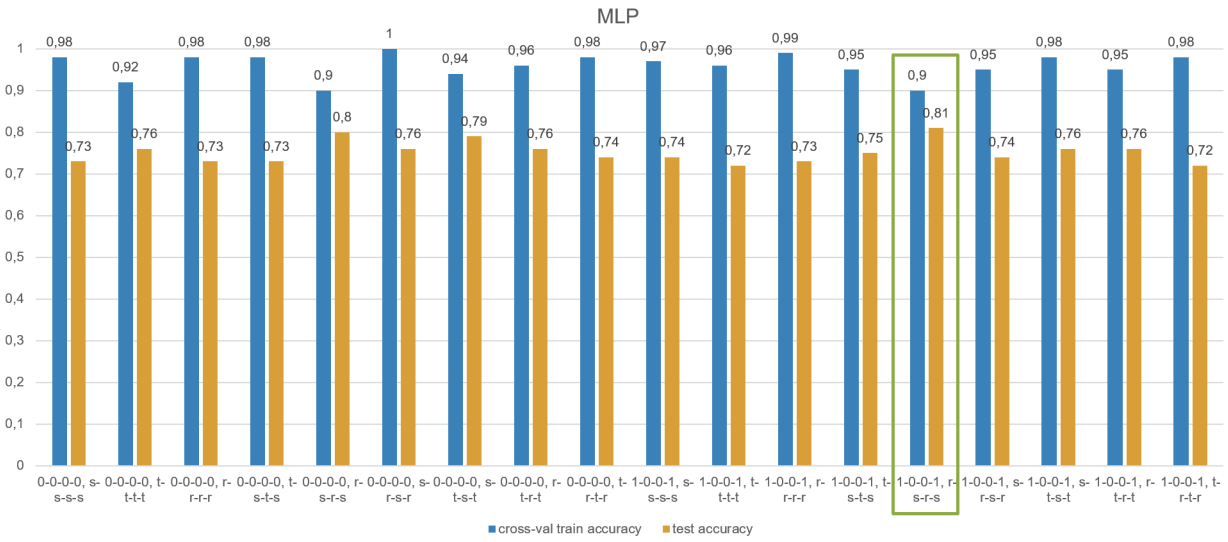


Figure 4.33. Accuracy values for the train and test sets obtained with different MLP models. The learning rate was equal to 0.002 and the dropout value was equal to 0.3. Legend: 0 = 512, 1 = 256, 2 = 128, 3= 64, s = sigmoid, t = tanh, r = relu.

To sum up, the first tuning phase was characterized by a fixed number of epochs equal to 50 and led to obtaining a value of accuracy on the test set equal to 80%.

The second tuning phase involved an early stopping step and tuned more finely the values of learning rate and dropout reaching a maximum of 79% accuracy on the test set.

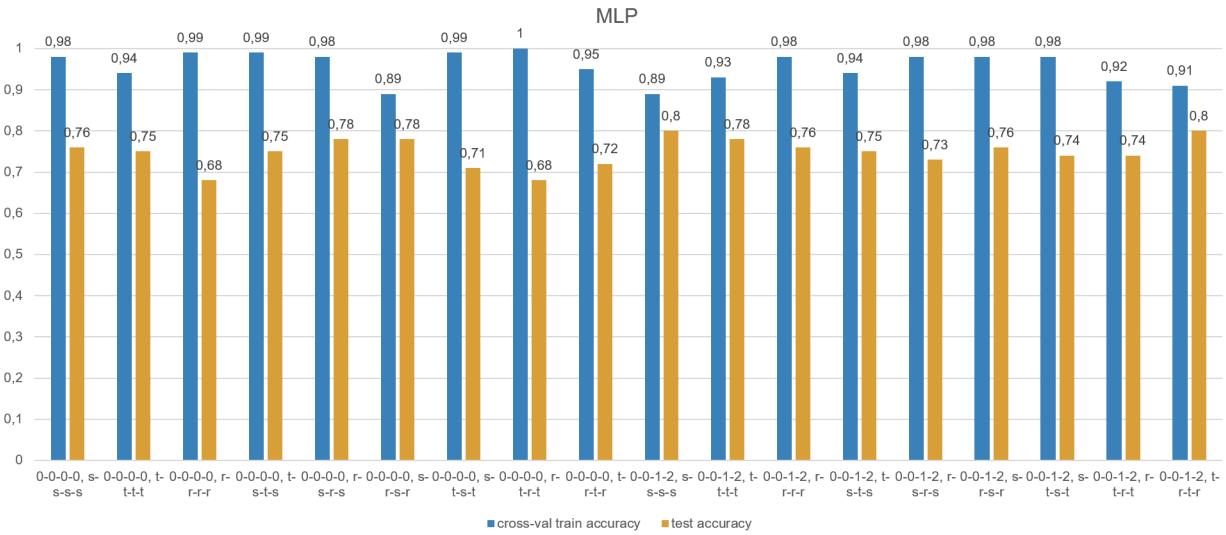


Figure 4.34. Accuracy values for the train and test sets obtained with different MLP models. The learning rate was equal to 0.002 and the dropout value was equal to 0.3. Legend: 0 = 512, 1 = 256, 2 = 128, 3= 64, s = sigmoid, t = tanh, r = relu.

Finally the number of nodes for each layer and the type of activation functions were varied while the best learning rate and dropout identified earlier were maintained equal to the previous values.

The highest accuracy was finally reached, the corresponding values were 90% accuracy in the cross-validation phase and 81% accuracy on the test set. The parameters that characterize the MLP model that reached the highest performances are the following:

- activation functions = relu-sigmoid-relu-sigmoid
- number of nodes per layer = 256-512-512-256
- Learning rate = 0.002
- Dropout = 0.3

The best MLP model defined earlier returned the confusion matrices displayed in figure 4.35 and the metrics of figure 4.36.

Confusion matrix train		
	True	False
Positive	452	145
Negative	4	878

Confusion matrix test		
	True	False
Positive	1990	630
Negative	359	2199

Figure 4.35. Confusion matrices for train and test set of the best MLP model. The parameters were: learning rate = 0.002, dropout = 0.3, layers = 256-512-512-256, activation functions = relu-sigmoid-relu-sigmoid

	Train	Test
Accuracy	0.9	0.81
Precision	0.86	0.78
Recall	0.99	0.86
AUC	0.88	0.81

Figure 4.36. Performance metrics for the best MLP model. The parameters were: learning rate = 0.002, dropout = 0.3, layers = 256-512-512-256, activation functions = relu-sigmoid-relu-sigmoid

4.5 Oncofuse

Shugay M. et al. introduced with their paper a valuable tool to predict the oncogenic potential of gene fusions. The comparison with the results obtained by the authors with this tool represents an opportunity to assess the performances of the method that I present in this thesis. As already stated the authors trained and tested their tool on a total of 9 databases. The results obtained by Oncofuse on each of the analyzed databases are displayed in Fig. 2 of the paper reported here in figure 4.37.

The sets of features used by Oncofuse included some of the elements covered in this thesis (e.g. transcription factors and gene ontologies) but the breakpoints of the gene fusions were not considered and were therefore not available. Therefore the two correlated features identified in this thesis had

to be excluded from the training and testing phases.

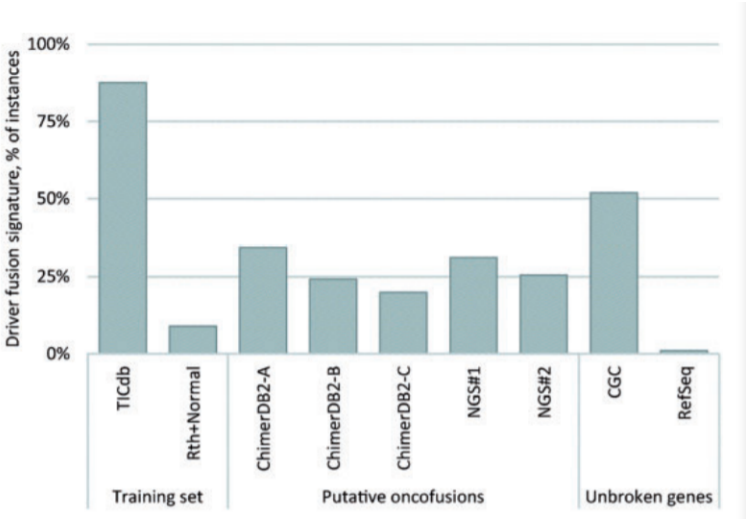


Figure 4.37. Figure 2 of Oncofuse

The results reported by Oncofuse were compared to the performances obtained using the best MLP model found in the previous phase. The selected features for this evaluation were 281 of which:

- The 3 available initial features
- 93 transcription factors
- 155 micro RNAs
- 30 Gene ontologies

The maximum number of epochs was set to 30 and the number of nodes per layer were respectively 64, 128, 128, 64. For lack of precise values for the following examinations, the performances of Oncofuse are approximated as displayed in the diagram of figure 4.38.

The confusion matrices and the performances obtained with this model are displayed below respectively in figure 4.39 and figure 4.40.

A second performance evaluation included a MLP with the following parameters:

- Learning rate = 0.03

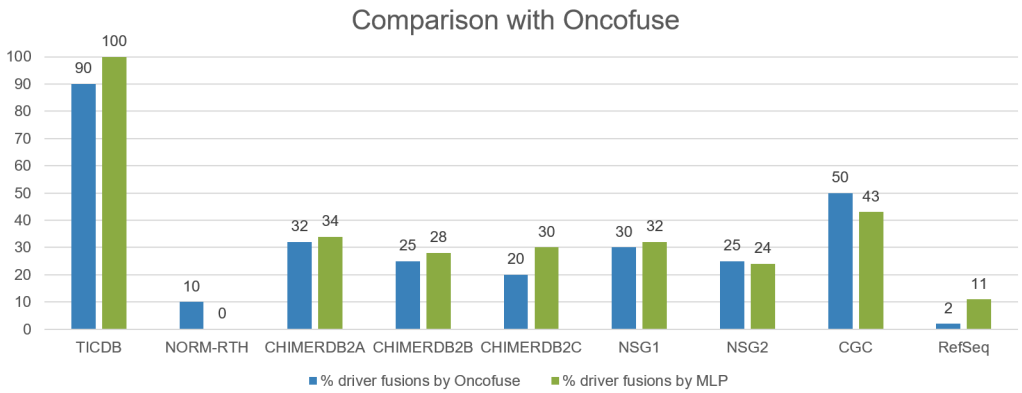


Figure 4.38. Comparison of the assumed performances of Oncofuse with respect to the performances obtained by the best MLP model

Confusion matrix train

	True	False
Positive	256	0
Negative	0	268

Confusion matrix test

	True	False
Positive	14527	1942
Negative	3764	1566

Figure 4.39. Confusion matrices for the train and test set obtained after training and testing the best MLP model found earlier on the data provided by Oncofuse

- Number of epochs = 50
- Number of nodes = 256-128-64-32
- Activation functions = relu-sigm-relu-sigm
- Dropout = 0.4

These parameters were chosen after a tuning phase that took into consideration the results obtained with the new training data.

For this test, the number of features was decreased to 44 using the random forest selection method with a threshold equal to 0.004. The selected features were:

	Train	Test
Accuracy	1	0,74
Precision	1	0,45
Recall	1	0,29

Figure 4.40. Metrics for the train and test set obtained after training and testing the best MLP model found earlier on the data provided by Oncofuse

- The 3 available initial features
- 3 transcription factors
- 31 micro RNAs
- 7 Gene ontologies

The comparison of the performances of the model proposed by Shugay M. et al. with respect to the results obtained by the model defined previously are displayed in figure 4.41.

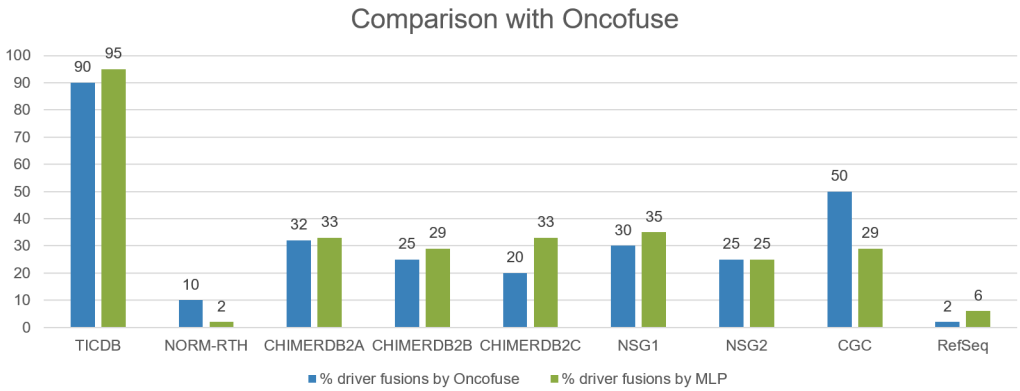


Figure 4.41. Comparison of the assumed performances of Oncofuse with respect to the performances obtained by the optimal MLP model

The confusion matrices and the performances are displayed in figure 4.42 and 4.43.

Confusion matrix train

	True	False
Positive	251	5
Negative	13	255

Confusion matrix test

	True	False
Positive	15355	1114
Negative	3689	1641

Figure 4.42. Confusion matrices for the train and test set obtained after training and testing the optimal MLP model

	Train	Test
Accuracy	0,97	0,78
Precision	0,98	0,6
Recall	0,95	0,31

Figure 4.43. Metrics for the train and test set obtained after training and testing the optimal MLP model

Chapter 5

Discussion

5.1 SVM

The first cross-validation results obtained with different configurations of SVMs and different sets of features highlighted the benefit of exploiting information coming from two of the three analyzed elements (e.g. transcription factors and micro-RNAs) as well as the information deduced from the gene names and the breakpoints.

5.1.1 The importance of the five initial features

Concerning the performances of the linear kernel during the first cross-validation experiment, it was found interesting that a feature set composed only by the five initial features seemed to perform well in the training phase. The information deduced from the breakpoints, the definition of the genes and the belonging strand resulted in a very high value of accuracy on their own. Furthermore, when these five features were added to the features set of either one of the other two elements the accuracy consistently improved with respect to the results obtained using the transcription factors or the microRNAs only. This improvement is evident in the results of the cross validation of figure 4.6 that displays the obtained mean of f1 of the SVM with the linear kernel. The fact that the random forest selection method almost always retained those five features confirms their significance in the classification of gene fusions.

As already stated, a likely oncogenic chimera may be obtained when one or both genes are identified as known oncogenes, this notion, along with

the extent of the retained oncogene, may be crucial for the classification. For instance, the training set contains the SS18-SSX4 gene pair labeled as an oncogenic gene fusion. In this case, SS18 is defined as "oncogene" and 78,84% is retained after the gene fusion event. On the other hand, SSX4 is defined as "other" and 40,28% is retained. Since a good portion of the gene closer to the promoter is retained it is reasonable to assume that the final gene fusion will express some oncogenic behavior.

Certainly, it is clear from the subsequent observation on the testing set that these five features alone are not sufficient to confidently make a correct prediction, nevertheless, the information they provide is valuable.

5.1.2 The complete feature set

The second cross-validation evaluation resulted in a maximum of 0.95 mean of f1 score for the linear kernel and 0.96 for the polynomial kernel. The results obtained with the remaining two kernels tended to be generally low but this behavior, at least for the gaussian kernel, was imputable to the use of non-optimal parameters, namely the values for gamma and for the coefficient.

The subsequent evaluation took into consideration the 3 main elements (e.g. transcription factors, micro-RNAs and gene ontologies). The number of features was reduced with the random forest feature selection method. The first observation is that the number of gene ontologies was greatly reduced, from 5025 to 271, indicating that the number of useful gene ontologies for the classification task is very small compared to the entire set of GOs. The mean of the accuracy for the cross-validation phase tended to be generally higher for the 271 gene ontologies when compared to the mean of accuracy values obtained with the other two sets (173 TF and 252 miRNA).

In particular, the use of miRNAs in the classification of gene fusions distinguishes this thesis work from other works like *Oncofuse* that did not consider them. It should be noted that the MLP cross validation results obtained using the information related to 333 microRNAs outperform the results obtained using the five features or the ones returned by the model trained on the 181 TF. This outcome highlights the importance of the role that miRNAs may have in oncogenic gene fusions.

The combination of gene ontologies and micro-RNAs returned a comparable or slightly higher mean of accuracy than the other two combinations of feature set implying the importance of including miRNA in the final feature

set.

However, the results obtained when including features from each of the three sets were the highest, this justifies the final decision to use the complete feature set in further evaluations.

The random forest feature selection method picked a comparable amount of features from each set implying that it is fundamental to extract information from each of these different sets to perform well in the classification task.

5.1.3 Tuning of the parameters

In the cross-validation phase, the accuracy values obtained with the gaussian and the sigmoid kernel still tended to be lower than the ones obtained with the linear and polynomial kernel suggesting the need for further tuning of the parameters.

When trying different values of coefficient for the linear kernel no substantial difference emerged. The mean of the accuracy was slightly higher with a lower coefficient value.

The gaussian kernel showed an interesting pattern of increased accuracy for high values of coefficient and low values of gamma. These configurations reached the values obtained with the linear kernel implying that a higher result could be obtained with further tuning of these parameters. The highest result obtained with the gaussian kernel was equal to the higher result reached by the linear kernel.

The sigmoid kernel on the other hand showed no improvement with the tuning of parameters, the cross-validation confirmed that this model is unable to perform this classification task with adequate reliability. Moreover the classifier built using the sigmoid kernel was often characterized by a 60% mean of accuracy on the training set pointing out to a classification of the entire set in either one of the two classes.

The polynomial kernel returned consistently high results with each combination of the parameters. The maximum mean of accuracy was comparable to the maximum results obtained with the gaussian kernel and the linear kernel.

During the testing phase, the highest accuracy results were returned by the rbf kernel and the polynomial kernel. The accuracy value of the test set reached 75% for these two kernels when using a fairly high number of

features (e.g. 288) that in this case excluded one of the initial features (e.g. 'same strand'). When running the experiment once again with the same threshold value the feature 'same strand' was retained and the results were again high for the rbf. On the other hand, the performances of the polynomial kernel were slightly lower than the ones achieved by the gaussian kernel. Since in the majority of the experiments each of the five initial features was retained by the random forest feature selection method the results of the second experiments were preferred to apply the subsequent decisions.

When reducing the number of features, in fact, the results were equivalent, the gaussian kernel performed consistently throughout these 3 experiments and was therefore chosen for the final tuning of the parameters.

Starting from a larger range of gamma and coefficient values the best possible parameters were identified. The high performances reached by the non-linear kernel imply that the problem itself may not be linearly separable.

The last small range of coefficient and gamma values returned consistently high results (figure 4.25) with respect to the accuracy value on the test set. In the end, the configuration that ensured the highest recall possible on the test set, or in other words minimized the false-negative samples, was preferred.

5.2 MLP

The cross-validation phase with the MLP classifier returned the highest results when using a combination of the three main sets of features, in agreement with what was found with the SVM classifier.

Even with the best MLP model a few hundred negative samples were classified as positive and vice versa. When comparing the samples that were misclassified by the MLP and the ones misclassified by the SVM it was found that:

- about 4/10 of samples that were classified as oncogenic by the SVM but were actually non-oncogenic were also classified as false positives by the MLP
- about 4/10 of samples that were classified as non-oncogenic by the SVM but were actually oncogenic were also classified as false negatives by the MLP

This comparison took into consideration two models that reached a comparable accuracy value (about 75%) and demonstrates that both models have

a certain degree of agreement when it comes to misclassified samples.

The final MLP model returned the highest performances in both the cross validation phase and the testing phase. This outcome led to the conclusion that MLP models should be preferred with respect to the SVM models when approaching the classification of oncogenic gene fusions.

5.3 Comparison with Oncofuse

The results of the MLP model when trained and tested on the samples provided by Oncofuse, although not optimal, were able to outperform the ones illustrated by the paper.

When using an MLP model with the same parameters reported as optimal in the study of this thesis (with the exception of the number of nodes for each layer that were reduced) the model seemed to overfit but still managed to detect a slightly higher percentage of positive test samples and obtain comparable results with the negative test samples.

When tuning the MLP model to suit better the new data the results on the two provided sets for the training were high, outperforming the ones obtained by Oncofuse:

- 95% of TICDB samples were correctly classified as driver gene fusions as opposed to the assumed 90% reported by Oncofuse
- 2% of the Normal samples were incorrectly classified as driver gene fusions as opposed to the assumed 10% reported by Oncofuse

The obtained accuracy on the test set was somewhat high (e.g. 74%) but it must be noted that the recall value was considerably low. Concerning the positive test samples, the paper clarifies that they are definable as 'putative' since these gene pairs were obtained by detection tools and not experimentally validated. It should be also noted that the content of the used databases could be nowadays defined as outdated when compared to the data used in the earlier phases of this thesis.

Nevertheless, even without the notion of the retained percentage of genes, the optimal MLP model was able to minimize the number of detected driver fusions of the 'unbroken genes' (negative testing samples) and obtained comparable or slightly higher results than the paper.

The MLP classifier performed similarly in each of the three sets of samples belonging to CHIMERDB correctly classifying as oncogenic about 1/3 of

the samples regardless of the high-confidence/low-confidence differentiation illustrated by Oncofuse.

Chapter 6

Conclusion

The results demonstrated that both classifier models (e.g. SVM and MLP) are able to discern between driver and passenger gene fusions.

The performances on both the train set and the test set are satisfactory, the results seem even more reliable when observing that the models were trained on thoroughly validated samples and then tested on a considerable amount of different gene fusions.

The gene fusions belonging to the test set consisted of different genes (the vast majority of them was never seen by the classifier) and that led to robust performances.

The selected features are therefore thought to be able to extract valuable knowledge to perform the classification task. The unique combination obtained by the random forest selection method using transcription factors, gene ontologies and micro-RNAs along with the five features identified at the beginning of this study led to about 80% accuracy on the test set with both classifiers.

The high accuracy values were associated to a high recall on the test set as well equal to 86% for both the best SVM model and the chosen MLP model. This result highlights the efficiency of the classifiers that not only correctly classify the majority of the samples but are also able to keep the rate of false negatives relatively low.

Eventually the MLP was the final model proposed by this study, since this classifier was able to reach sligthly higher accuracy and AUC values on the test set compared to the best SVM model.

Moreover, the comparison with the literature demonstrated that the MLP model was able to reach comparable or slightly higher results than the ones illustrated by *Oncofuse*[7].

This comparison confirmed the quality of the performances obtained in this thesis suggesting that the proposed model can be a valuable tool to classify gene fusions in oncogenic or not oncogenic.

In conclusion, in this thesis, I presented a robust classifier model that, in combination with the features extracted and selected during this study, may represent a suitable tool for the classification of gene fusions.

Bibliography

[1] Marta Lovino, Maria Serena Ciaburri, Gianvito Urgese, Santa Di Cataldo, and Elisa Ficarra. DEEPrior: a deep learning tool for the prioritization of gene fusions. *Bioinformatics*, 36(10):3248–3250, 02 2020.

[2] Babiceanu Mihaela, Qin Fujun, Xie Zhongqiu, Jia Yuemeng, Lopez Kevin, Janus Nick, Facemire Loryn, Kumar Shailesh, Pang Yuwei, Qi Yanjun, Lazar Iulia M, and Li Hui. Recurrent chimeric fusion RNAs in non-cancer tissues and cells. *Nucleic Acids Res*, 44(6):2859–72, 04 2016.

[3] Lever Jake, Zhao Eric Y., Grewal Jasleen, Jones Martin R., and Jones Steven J. M. Cancermine: a literature-mined resource for drivers, oncogenes and tumor suppressors in cancer. *Nature Methods*, 16:505–507, 2019.

[4] ENCODE Project Consortium. The encode (encyclopedia of dna elements) project. *Science*, 306(5696):636–40, 10 2004.

[5] Andrew D Yates, Premanand Achuthan, Akanni, et al. Ensembl 2020. *Nucleic Acids Research*, 48(D1):D682–D688, 11 2019.

[6] Vikram Agarwal, George W Bell, Jin-Wu Nam, and David P. Bartel. Predicting effective microrna target sites in mammalian mrnas. *eLife*, 4:e05005, aug 2015.

[7] Mikhail, Shugay and Iñigo, Ortiz de Mendíbil and José L, Vizmanos and Francisco J, Novo. "oncofuse: A computational framework for the prediction of the oncogenic potential of gene fusions". *Bioinformatics*, 29(20):2539–46, 10 2013.

[8] Mitelman Felix, Johansson Bertil, and Mertens Fredrik. The impact of translocations and gene fusions on cancer causation. *Nature Reviews Cancer*, 7:233–245, 2007.

[9] Ren Ruibao. Mechanisms of bcr–abl in the pathogenesis of chronic myelogenous leukaemia. *Nature Reviews Cancer*, 5:172–183, 2005.

[10] Scott A. Tomlins, Bharathi Laxman, Sooryanarayana Varambally, Xuhong Cao, Jindan Yu, Beth E. Helgeson, Qi Cao, John R. Prensner, Mark A. Rubin, Rajal B. Shah, Rohit Mehra, and Arul M. Chinnaiyan. Role of the tmprss2-erg gene fusion in prostate cancer. *Neoplasia*, 10(2):177 – IN9, 2008.

[11] Kalpana Kannan, Cristian Coarfa, Pei-Wen Chao, Liming Luo, Yan Wang, Amy E. Brinegar, Shannon M. Hawkins, Aleksandar Milosavljevic, Martin M. Matzuk, and Laising Yen. Recurrent bcam-akt2 fusion gene leads to a constitutively activated akt2 fusion kinase in high-grade serous ovarian carcinoma. *Proceedings of the National Academy of Sciences*, 112(11):E1272–E1277, 2015.

[12] Goran Stenman, Mattias K. Andersson, and Ywonne Andren. New tricks from an old oncogene. *Cell Cycle*, 9(15):3058–3067, 2010. PMID: 20647765.

[13] Heyer Erin E. et al. Diagnosis of fusion genes using targeted RNA sequencing. *Nature Communications*, 2019.

[14] Abate Francesco et al. Pegasus: A Comprehensive Annotation and Prediction Tool for Detection of Driver Gene Fusions in Cancer. *BMC systems biology*, 2014.

[15] Natasha S. Latysheva and M. Madan Babu. Discovering and understanding oncogenic gene fusions through data intensive computational approaches. *Nucleic Acids Research*, 44(10):4487–4503, 04 2016.

[16] Fawcett Gloria and Eterovic A. Karina. Identification of genomic somatic variants in cancer. *Advances in Clinical Chemistry*, 78:123–162, 2017.

[17] Ren Shancheng, Peng Zhiyu, et al. Rna-seq analysis of prostate cancer in the chinese population identifies recurrent gene fusions, cancer-associated long noncoding rnas and aberrant alternative splicings. *Cell Research*, 22:806–821, 2012.

[18] Kumar-Sinha Chandan, Tomlins Scott A., and Chinnaiyan Arul M. Recurrent gene fusions in prostate cancer. *Nature Reviews Cancer*, 8:497–511, 2008.

[19] Maher Christopher A., Kumar-Sinha Chandan, Cao Xuhong, et al. Transcriptome sequencing to detect gene fusions in cancer. *Nature*, 458:97–101, 2009.

[20] Hui Li, Jinglan Wang, Xianyong Ma, and Jeffrey Sklar. Gene fusions and rna trans-splicing in normal and neoplastic human cells. *Cell Cycle*, 8(2):218–222, 2009. PMID: 19158498.

[21] Hui Li, Jinglan Wang, Gil Mor, and Jeffrey Sklar. A neoplastic gene

fusion mimics trans-splicing of rnas in normal human cells. *Science*, 321(5894):1357–1361, 2008.

[22] Nature education. Transcription factor / transcription factors.

[23] Rainer Renkawitz. *Transcription Factors and Regulation of Gene Expression*, pages 1886–1890. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[24] Ha Minju and Kim V. Narry. Regulation of microrna biogenesis. *Nature Reviews Molecular Cell Biology*, 15:509–524, 2014.

[25] Pillai Ramesh S. Microrna function: Multiple mechanisms for a tiny rna? *RNA*, 11:1753–1761, 2005.

[26] George Adrian Calin and Carlo Maria Croce. Microrna-cancer connection: The beginning of a new tale. *Cancer Research*, 66(15):7390–7394, 2006.

[27] Martin D. Jansson and Anders H. Lund. Microrna and cancer. *Molecular Oncology*, 6(6):590 – 610, 2012. Cancer epigenetics.

[28] Paul D. Thomas. *The Gene Ontology and the Meaning of Biological Function*, pages 15–24. Springer New York, New York, NY, 2017.

[29] Li Li, Zhang Kangyu, Lee James, Cordes Shaun, Davis David P, and Tang Zhijun. Discovering cancer genes by integrating network and functional properties. *BMC Medical Genomics*, 2, 2009.

[30] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers Electrical Engineering*, 40(1):16 – 28, 2014. 40th-year commemorative issue.

[31] Yanjun Qi. *Random Forest for Bioinformatics*, pages 307–323. Springer US, Boston, MA, 2012.

[32] Houtao Deng and G. Runger. Feature selection via regularized trees. In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2012.

[33] H Menze Bjoern et al. A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data. *BMC Bioinformatics*, 2009.

[34] Jaime G. Carbonell, Ryszard S. Michalski, and Tom M. Mitchell. 1 - an overview of machine learning. In Ryszard S. Michalski, Jaime G. Carbonell, and Tom M. Mitchell, editors, *Machine Learning*, pages 3 – 23. Morgan Kaufmann, San Francisco (CA), 1983.

[35] M. I. Jordan and T. M. Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.

[36] Sotiris B Kotsiantis, I Zaharakis, and P Pintelas. Supervised machine

learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, 160(1):3–24, 2007.

[37] Vikramaditya Jakkula. Tutorial on support vector machine (svm). *School of EECS, Washington State University*, 37, 2006.

[38] D. J. Sebald and J. A. Bucklew. Support vector machine techniques for nonlinear equalization. *IEEE Transactions on Signal Processing*, 48(11):3217–3226, 2000.

[39] Shun-ichi Amari and Si Wu. Improving support vector machine classifiers by modifying kernel functions. *Neural Networks*, 12(6):783–789, 1999.

[40] Bernhard Schlkopf, Alexander J. Smola, and Francis Bach. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press, 2018.

[41] Daniel Svozil, Vladimir Kvasnicka, and Jiri Pospichal. Introduction to multi-layer feed-forward neural networks. *Chemometrics and intelligent laboratory systems*, 39(1):43–62, 1997.

[42] Svozil Daniel, Kvasnicka Vladimír, and Pospichal Jirí. Introduction to multi-layer feed-forward neural networks. *Chemometrics and Intelligent Laboratory Systems*, 1997.

[43] M. k. Alsmadi, K. B. Omar, S. A. Noah, and I. Almarashdah. Performance comparison of multi-layer perceptron (back propagation, delta rule and perceptron) algorithms in neural networks. In *2009 IEEE International Advance Computing Conference*, pages 296–299, 2009.

[44] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[45] Jingzhao Zhang, Sai Praneeth Karimireddy, Andreas Veit, Seungyeon Kim, Sashank J Reddi, Sanjiv Kumar, and Suvrit Sra. Why adam beats sgd for attention models, 2019.

[46] Tom Dietterich. Overfitting and undercomputing in machine learning. *ACM computing surveys (CSUR)*, 27(3):326–327, 1995.

[47] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, January 2014.

[48] Lutz Prechelt. Early stopping-but when? In *Neural Networks: Tricks of the trade*, pages 55–69. Springer, 1998.

[49] Tzu-Tsung Wong. Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation. *Pattern Recognition*, 48(9):2839–2846, 2015.

[50] Mohammad Hossin and MN Sulaiman. A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process*, 5(2):1, 2015.

[51] Robert C. Friedman et al. Most mammalian mRNAs are conserved targets of microRNAs. *Advanced*, 2008.

[52] HPC@POLITO. Hpc, a project of academic computing within the department of control and computer engineering at the politecnico di torino.