One-class SVM and Supervised Machine Learning Models for uncovering associations of non-coding RNA with diseases

by

JUAN MANUEL GUTIÉRREZ CÁRDENAS

submitted in accordance with the requirements for
the degree of

DOCTOR OF PHILOSOPHY

In the subject

COMPUTING

at the

University of South Africa

Supervisor: PROF. ZENGHUI WANG

January 2022

DECLARATION

Name: __Juan Manuel Gutiérrez Cárdenas_____

Student number: __ 53515285_____

Degree: __PhD (Computer Science)_____

Exact wording of the title of the dissertation or thesis as appearing on the copies submitted for examination:

One-class SVM and Supervised Machine Learning Models for uncovering associations of non-coding RNA with diseases

I declare that the above dissertation/thesis is my own work and that all the sources that I have used or quoted have been indicated and acknowledged by means of complete references.

SIGNATURE

9 January 2022

DATE

Acknowledgment

I would like to express my sincere gratitude to my supervisor, Prof. Zenghui Wang, for all the help, guidance, and advice given during these difficult circumstances. I must also acknowledge Prof. Ian Sanders for his help, mentoring, and friendship giving to me through all these years. Also, to Dr. Victor Ayma for his academic advice and friendship as a truthful colleague. Lastly, I am filled with gratitude to my parents for giving me the necessary encouragement to continue this path of research and continuous studying.

Abstract

The study of MicroRNAs (miRNAs), long non-coding RNAs (lncRNAs) and gene interactions may be expected to provide new technologies to serve as valuable biomarkers for personalized treatments of diseases and to aid in the prognosis of certain conditions. These molecules act at the genome level by regulating or suppressing their protein expression functions.

The primary challenge in the study of these non-coding molecules involves the necessity of finding labeled data indicating positive and negative interactions when predicting interactions using machine-learning or deep-learning techniques. However, usually we end up with a scenario of unbalanced data or unstable scenarios for using these models. An additional problem involves the extraction of features derived from the binding of these non-coding RNAs and genes. This binding process usually occurs fully or partially in animal genetics, which leads to considerable complexity in studying the process. Therefore, the main objective of the present work is to demonstrate that it is possible to use features extracted for miRNAs sequences in the development of diseases such as breast cancer, breast neoplasms, or if there is any influence with immune genes related to the SARS-COV-2.

We performed experiments focusing on the erb-b2 receptor tyrosine kinase 2 (ERBB2) gene involved in breast cancer. For this purpose, we gathered miRNA-mRNA information from the binding between these two genetic molecules. In this part of our research, we applied a One-Class SVM and an Isolation Forest to discriminate between weak interactions, outliers given by the one-class model, and strong interactions that could occur between miRNA and mRNA (messenger RNA).

Additionally, this study aimed to differentiate between breast cancer cases and breast neoplasm conditions. In this section we used the information encoded in lncRNAs. The additional feature used in this part was the frequency of k-mers, i.e., small portions of nucleotides, along with the data from the energy released in miRNA folding. The models used to discriminate between these diseases were One-Class SVM, SVM, and Random Forest.

In the final part of the present work, we described a subset of probable miRNA binding with SARS-COV-2 RNA, focusing on those miRNAs with a relationship with genes involved in

the immunological system of the human body. The models used as classifiers were One-Class SVM, SVM, and Random Forest.

The results obtained in the present study are comparable to those found in the current literature and demonstrate the feasibility of using one-class models combined with features from the coupling of non-coding genes or mRNAs and their relationships with forms of breast cancer and viral infections. This work is expected to establish a basis for future avenues of research to apply one-class machine-learning models with feature extraction based on genomic sequences to the study of the relationship between non-coding RNAs and various diseases.

KEYWORDS: mRNAs, lncRNAs, k-mers, sequence features, Breast Neoplasms, Breast cancer, SARS-CoV-2, One-class models, Supervised Learning, Unsupervised learning

List of Abbreviations

BLOSUM	BLOcks of Amino Acid SUbstitution Matrix
DL	Deep learning
ERBB2	erb-b2 receptor tyrosine kinase 2
lncRNAs	Long non-coding RNAs
miRNAs	MicroRNAs
ML	Machine learning
MFE	Minimum-Free-Energy
mRNA	Messenger RNA
RF	Random forest
RNA	Ribonucleic acid
RNAi	RNA interference
SARS-CoV-2	Severe acute respiratory syndrome coronavirus 2
siRNA	Small interfering RNA
SVM	Support vector machine

Table of Contents

List of Figures

List of Tables

List of Publications

1. Gutiérrez-Cárdenas, J. and Wang, Z. (2021) 'One-class models for validation of miRNAs and ERBB2 gene interactions based on sequence features for breast cancer scenarios', ICT Express. doi: 10.1016/j.icte.2021.03.001
2. Gutiérrez-Cárdenas, J. and Wang, Z. Classification of Breast Cancer and Breast Neoplasm scenarios based on Machine Learning and sequence features from lncRNAs-miRNAs-diseases associations, Interdisciplinary Sciences: Computational Life Sciences. doi: 10.1007/s12539-021-00451-6
3. Gutiérrez-Cárdenas, J. and Wang, Z. Prediction of binding miRNAs involved with immune genes to the SARS-CoV-2 by using sequence features extraction and One-class SVM, Submitted to the Computers in Biology and Medicine journal.

CHAPTER 1
Introduction

1.1 Background

miRNAs are small non-coding RNAs that are not involved in the process of protein production. However, they can bind to specific genes and regulate or repress them (Loh et al., 2019). When this regulation occurs, the genes begin to change their protein production, leading to the appearance or control of certain diseases within the human body (López-Camarillo and Marchat, 2013; Condorelli et al., 2014). These miRNA molecules are evolutionarily conserved, and their lengths are no longer than 19 to 25 nucleotides (Loh et al., 2019; Han, 2004). When miRNAs bind to genes, they can have perfect complementarity, for example, in plants (Schwab et al., 2005), and full or partial complementarity as it occurs in humans (Condorelli et al., 2014). It is well known that miRNAs can repress or inhibit the molecular proteins produced by affected genes, which could lead to various diseases (Ardekani and Naeini, 2010; Bartel. 2004; Shen et al., 2014). The interactions of miRNAs and other forms of non-coding RNAs in the prognosis of certain diseases are undeniable, particularly in breast cancer diagnosis scenarios (Loh et al., 2019).

Throughout the course of breast cancer, miRNAs serve as oncogenes or tumor suppressors, which has been an active research topic in recent years. Studies have classified breast cancer types or predicted patients' prognoses in specific clinical cases (López-Camarillo and Marchat, 2013). These molecules can interact with specific genes by increasing or decreasing gene regulation (Negrini and Calin, 2008), thereby silencing them, inhibiting translation, or even degrading them (Bartel, 2004; Loh et al., 2019; López-Camarillo and Marchat, 2013; Negrini and Calin, 2008). Additionally, miRNAs can aid in the formation of oncogenes or tumor suppressor genes in breast cancer scenarios (Loh et al., 2019; Negrini and Calin, 2008).

One-class SVM models are commonly used to detect novel or anomalous data, returning a value of +1 if the data are enclosed within a region and -1 if they are in the region's outbound (Schölkopf et al., 2001). As an SVM, such models generate a dimensional map using kernels to separate the data from a particular origin by a maximum margin, as demonstrated by Schölkopf et al. (2001). Samples that fall within this enclosed region are assigned to a positive (+1) class, whereas outsiders are assigned to a negative (-1) class. Hence, such models exploit the accuracy of SVMs using data involving only one training class. In

contrast, supervised learning models require two classes to perform predictive classification tasks. An inconvenience that arises with two class models, for example, for classification tasks, is that, on some occasions, the samples that belong to one class are insufficient, or we have only samples from one (positive) class. Moreover, acquiring samples from the second (negative) class is challenging or cannot be achieved directly (Sedaghat et al., 2018; Irigoien et al., 2014). Another attractive differentia of the one-class SVM is that it does not require a negative class to train the model. Nevertheless, on some occasions, a subset of negative samples is chosen to test the model's performance, a step that is not mandatory (Eude and Chang, 2018). Additionally, the use of a single class obviates the need to obtain labeled data; however, while tweaking hyperparameters or validating the output of such models, extra caution should be taken. Certain samples may be chosen to represent a synthetic negative class as a quality control measure for this model.

Similarly, other types of non-coding RNAs, such as long-coding RNAs (Wapinski and Chang, 2011) have also been objects of study as being present in certain diseases. In breast cancer, lncRNAs have a role in metastasis by altering the chromosomal landscape. Other examples include the lncRNA HOTAIR, which binds to HOXD genes (Harries, 2012; Wapinski and Chang, 2011), acting as a tumor suppressor in the Gas5 molecule (Wapinski and Chang, 2011), or by regulating the expression of the lncRNA known as LSINCT5 (Harries, 2012). MiRNAs can operate as oncongenes or tumor suppressors in breast cancer and its metastatic form. For example, it has been reported that the miRNA-331 interacts with the ERBB2 o HER2 gene (Loh et al. 2019; McAnena et al.,2019) and that miR-124a and miR-26b interacts with SerpinB2 (Loh et al. 2019).

Feature extraction should be performed on data prior to processing using machine-learning modes, which is relevant to non-coding RNA studies. Considering the use of sequence feature analysis such as the *k*-mer frequency, Wen et al. (2019) applied this procedure to a convolutional neural network; the term *k*-mer refers to the length of sequence nucleotides. For example, 1-mer implies only a single nucleotide, whereas 2-mer would have two nucleotide combinations. The researchers employed this technique to identify lncRNA-mRNA correlations in mice, chickens and humans. Interestingly, they discovered that increasing the k-mer number over three had a minor effect on accuracy.

miRNAs bind not only to mRNA or genes in humans or other species, but also to external or endogenous RNA such as that of viruses. Thus, the miRNAs may also act as if they were

interacting with an endogenous host gene or mRNA by repressing or regulating their primary functions. Additionally, in these cases, the miRNAs may even disable viral reproduction to regulate the spread of the virus on a host species. This binding occurs because miRNAs cannot discriminate between viral mRNA and that of the host organism (Nersisyan et al., 2020). However, further studies have concluded that there is no evidence that mRNA viruses can also produce miRNAs (Yousefi et al., 2020). Nonetheless, evidence has indeed ben adduced that miRNAs could interfere with the functions of the SARS-CoV-2 virus. This outcome could lead to a promising field of research, given the current pandemic.

1.2 Problem statement and research questions

MicroRNAs (miRNAs) bind to different genes, up-regulating, down-regulating, or suppressing their protein expression. This type of regulation may result in the occurrence of certain diseases such as various forms of cancer (Chen H. et al., 2018; Loh et al., 2019; Penyige et al., 2019; Prosenjit et al., 2018; Rehman et al., 2019). However, the study of the interactions of miRNAs or other non-coding RNAs with specific genes, is complex. This phenomenon occurs because multiple miRNAs may interact with a given gene, implies a considerable uncertainty in predicting which miRNAs will bind to which genes (Prosenjit et al., 2018), (Yan et al., 2007). Moreover, binding of miRNAs and genes in humans does not follow a strict complementarity as it does in plants. Consequently, direct observation of miRNAs and gene matching is not possible (Witkos et al., 2007) in contrast to the matching known to occur in plants (Schwab et al., 2005). This remains as a major challenge in the field.

Various studies have been conducted on predicting or classifying interactions between diverse miRNAs and genes. In recent years, it has become possible to perform in-silico computational experiments to supplement or replace of in-vitro experiments in genetic laboratories, also known as wet-lab facilities. Statistics, machine learning, and deep-learning techniques have become widely popular in the field of genetics. Therefore, researchers have realized that using machine-learning or deep-learning techniques could unveil interesting connections between miRNAs and genes, predict their interactions, or classify miRNAs according to observable characteristics. However, some inherent difficulties remain with the use of most existing supervised machine and deep-learning methods. For instance, in classification and prediction tasks, it is generally necessary to include at least two classes to construct a model to differentiate between different outcomes. However, obtaining properly

labeled datasets is often impossible. In genetic experiments, the procurement of differentiable classes may be considered unfeasible owing to the cost and time involved in wet-lab experiments.

Nevertheless, obtaining such labeled data is often complex. For example, a sample from one class might be too scarce which is a common situation in miRNA and gene interactions (Tran et al., 2008; Sedaghat et al., 2018; Yousef et al., 2008; Yousef et al., 2010). Datasets may be imbalanced, with an insufficient number of samples from one class, which tends to lead to some classes being under-sampled or over-sampled. However, the use of synthetically created samples to address this issue can also cause bias in the results of the model.

In summary, two main problems exist in the study of miRNA and gene interactions, the first of which is related to the difficulty of finding samples from a single validated class. Second, unbalanced dataset may be largely useless for training models based on supervised learning. Therefore, we believe that one-class techniques or anomaly detection techniques are required in the field of in-silico experiments, which has been dormant for some time. These techniques obviate the abovementioned necessity of two or more balanced classes affecting supervised models, and hence are suitable in genetics where one class's presence is scarce. Consequently, these models could enable the discovery of interactions between non-coding RNAs, genes, and even messenger RNAs (mRNAs) in the presence of only one class (Sedaghat et al., 2018; Irigoien et al., 2014).

Regarding the feature extraction needed for the application of machine-learning models, in contrast to deep-learning models, one trend involves the use of features obtained based on the genomic expression of miRNA-gene binding (Pham et al., 2019). Another trend relates to the extraction of features derived from the study of the nucleotides in genomic sequences. This approach cannot be directly applied owing to the lack of a perfect match as in plant miRNAs. However, it might hypothetically be possible to extract features from the analysis of nucleotides present in non-coding RNAs features' binding that could serve as inputs for a one-class model. Furthermore, the construction of a set of relevant attributes would involve the study of pairing sites, accessibility, or evolutionary conservation data is of uttermost importance.

Another part of our research is directed to miRNA-gene binding and other types of non-coding RNAs, such as long coding RNAs (lncRNAs). After a review of the pertinent literature, this research discovered that there are currently no studies evaluating the prediction

of lncRNA-miRNA and their association to disease. Some studies, however, have examined these associations separately, e.g., associations between lncRNA and-miRNA (Wen et al., 2019), between lncRNA and diseases (Guo et al., 2019), or between miRNAs and diseases (Fu and Peng, 2017), to name a few. Therefore, it might be hypothesized that the integrated study of lncRNA and miRNA together could be used to discover relationships with the development of specific diseases based on the different investigations found in the literature regarding miRNAs and disease associations.

The SARS-COV-2 pandemic has led to diverse research efforts to unveil the mechanisms of human response exposure to this viral strand. Previous studies on miRNAs along these lines have explicated the interaction of miRNAs with viral mRNAs based on experimental data Different studies on the relationship between miRNAs and the SARS viral genome have been proposed, for example, in the work of Pierce et al. (2020) and Ahmadi & Moradi (2020), among many others. However, we could not find any works in the relevant literature that used one-class models for these viral scenarios, which seemed natural because the single-class seems evidently to obtain in this context, as a list of the available miRNAs that could bind to the SARS viral mRNA. Therefore, apart from using a one-class model, it would be helpful to use features based on the genomic sequence of both miRNAs and mRNAs from the viral strand. In this scenario, the prediction of miRNA binding to viral mRNA is considered anomalous. We believe that this study can use the information from genomic alignment, minimum free energy released in a binding process, and information on the *k*-mers of the sequences. Additionally, our research considers whether the existing literature may validate our results by analyzing whether some diseases could be related to the miRNAs found.

Having thus noted the problem definition considered herein, we note four main research questions addressed by the present thesis.

R1. How can features based on sequence binding provide consistent results if used in machine-learning models?

R2. How is it possible to discriminate between cases of benign and malignant cancer scenarios using features extracted from lncRNA and miRNAs, considering particularly those related to breast cancer and breast neoplasm diseases?

R3. How is it possible to find miRNA binding with viral genome RNA strands such as that of SARS-COV-2 using features based on sequence analysis, k-mers, and one-class SVMs?

1.3 Research Objectives

To approach the research questions stated, we consider herein the following objectives.

RO1. Extract useful features based on the nucleotide sequence binding between miRNAs and genes, focusing on the ERBB2 gene involved in the development of breast cancer.

RO2. Determine the feasibility of using the extracted features with one-class unsupervised models and compare the results with the existing literature.

RO3. Validate cases of breast cancer and breast neoplasm using the features and models defined in previous research objectives.

RO4. Determine probable miRNA binding with SARS-CoV-2 using a one-class SVM and attributes extracted from the sequence binding between miRNAs, genes involved in the immune systems and the viral mRNA genome.

In the present study, research question R1 is mapped with the objectives RO1 and RO2; research question R2 is mapped with objective RO3, while question RO3 is mapped with objective RO4.

1.4 Overview of the research methodology

Various datasets for obtaining the miRNAs, lncRNAs, diseases, and SARS viral genomes were used for the current project, which are explained in detail in the following chapters. Of note, all data was obtained from public datasets, the author of the current thesis performed no wet-lab experiments. Rather, all experiments were performed using computational means or in-silico. The data available are free to use, and references to these original data are naturally provided.

This study used supervised and unsupervised machine-learning methods in relation to the methods used in our experiments. We examined both supervised methods, including an SVM and a random forest classifier, and unsupervised methods including isolation forest and one-class SVM. As the theme of the present work was to demonstrate the benefits of using well-known machine-learning methods with feature engineering from nucleotide sequences, in contrast to deep-learning methods of high computational complexity, we wanted to prove that the findings of our models are reasonably close to those reported in the literature.

The abovementioned methods required a set of features as inputs to the learning models. We focused on the use of features extracted from the alignment between genetic molecules. We used Python and BioPython to perform sequence manipulation on these matters, e.g., alignment of genetic sequences or extraction of *k*-mers, to mention a few of the procedures we employed. Additionally, we used the Vienna package with a Windows Python plugin for our experiments to obtain energy values related to the binding of genetic sequences.

To answer the research questions thus established, we performed the following tasks.

R1. How can the use of features based on sequence binding with machine-learning models provide consistent results?

T1: We performed experiments using sequence features extracted from mirWalk and miRTargetLink to select a list of miRNAs that are related to certain diseases. This database served to measure the quality of these molecular interactions. These features were used as inputs for a one-class SVM and an isolation forest model.

R2. How to discriminate between benign and malignant cancer scenarios using features extracted from lncRNAs and miRNAs, considering as a particular case those related to breast cancer and breast neoplasm diseases?

T2: For this task, we selected samples of lncRNAs and miRNAs that were associated with breast cancer and breast neoplasm scenarios. We extracted features related to *k*-mers, sequence alignment between miRNAs and lncRNAs, and folding energy values. These features served as inputs for a one-class SVM model. The obtained results were validated using SVM and random forest models. The results show the possibility of discriminating between benign and malignant breast tissue samples using the features mentioned above and of validating them with unsupervised and supervised models.

R3: How to find miRNA bindings with viral genome RNA strands such as that of SARS-COV-2 using features based on sequence analysis, *k*-mers, and one-class SVMs?

T3: In this scenario, we extracted features from the genomic sequence of SARS-CoV-2 and miRNAs present in the human body. With the features obtained, we fed a one-class SVM model, and obtained several novel or distinct miRNAs with a preference to bind to the untranslated region of the SARS genome (5'UTR region). The literature validated the miRNAs that were obtained. Additionally, aim to determine whether there is a relationship

with miRNAs that interact with immune systems, and whether they could be prone to bind to the mRNA of the SARS-CoV-2 coronavirus.

1.5 Research contribution

As discussed in Section 1.1, many studies have used deep learning and miRNA expression profiles to predict miRNA-gene binding. Deep-learning models do not require a preceding feature engineering step because they strive to find patterns in the data that serve as input. However, training such models involves considerable computationally complexity and associated time requirements. In contrast, methods such as those of machine-learning models do require feature engineering or extraction techniques. Additionally, there are many machine-learning and deep-learning methods available in the literature. In practice, many known problems might be resolved by focusing on a limited number of models. For example, Fernandez-Delgado (2014) and Hand (2006) mentioned that there is no need to increase the complexity of such models, as this does not reflect an increase in their accuracy. Such conclusions have been receiving results in a variety of responses in the field of data science. Unfortunately, no further studies have corroborated these findings in comparisons between machine learning and more complex models, such as those used in deep learning. Clearly deep-learning models are of considerable benefit in various applications in data science. However, we believe that the use of conventional machine-learning methods involving feature engineering or extraction processes remains a fruitful avenue of research for some bioinformatics tasks, e.g., that of the prediction of binding sites of miRNA in genes or relationships between other non-coding RNAs such as lncRNAs.

We also hypothesize that the use of sequence binding features was relatively neglected in the study of non-coding RNA in favor or gene expression models. One probable drawback of using sequence features is that non-coding RNA binding in human genes does not necessarily have perfect matching or complementarity. However, we believe that using features such as sequence alignments, *k*-mers, and the energy released in a binding procedure could be of benefit for predicting non-coding RNAs to genes or finding relationships between them and diseases.

Since the advent and increase in computational power, many tasks in genetics and biology have been largely automated. However, the results of bioinformatics or computational experiments, known as in-silico experiments, always requires wet-lab validation. Such experiments are not straightforward and demand considerable time and effort; — this

overhead increases with the use of supervised machine-learning or deep-learning models. At least two classes are required for supervised learning, and there should not be so much divergence in quantities among them, i.e., the datasets should be balanced. The presence of imbalanced datasets requires techniques to create artificial data, which induces some bias into the results. One could argue that it would be advisable to generate data from two or more classes in wet-lab. However, this also involves the abovementioned problems. Considerable overhead in terms of effort and time would be required to perform such experiments. Therefore, we hypothesize that it would be fruitful to focus on the use of unsupervised learning using well-established, effective models such as SVM. Therefore, we decided to work with a one-class SVM. This model has the advantage of SVMs in terms of the quality of the results, while avoiding various difficulties noted above. Therefore, we concluded that such models would be suitable to test the prediction of non-coding RNAs interactions or relationships with diseases, and the experimental results validated this supposition.

In conclusion, the research contribution of the current work would be:

i. This research work will allow researchers to glimpse a perspective in using sequence features extracted from genetic sequences when the binding of non-coding RNAs and genes occurs instead of using genomic expression. It is essential to mention that using features from genetic sequences has been rarely used in the bioinformatics field.
ii. This study gives importance to the use of simple machine learning methods, specifically to one-class unsupervised techniques such as One-class SVM, which could perform remarkably well with unbalanced data, do not need to use extensive computational resources as it could occur with Deep learning models, and that has been dormant in the Bioinformatics field for quite a while.
iii. This research proposes an architecture that could extract information from sequence features, which could be used as input to one-class or two-classes machine learning models to classify or predict the relationship between non-coding RNAs and diseases.

Additionally, this research focuses on providing a practical research contribution as described by Ngwenyama (2014).

1.6 Thesis Structure

The present thesis is divided into the following chapters, and we provide a short description of each constituent part.

Chapter 1: Introduction

We begin by describing the background of the problem to be solved and stating the necessary research questions and objectives to be completed in the present thesis.

Chapter 2: Literature Review

In this chapter, we outline some relevant concepts regarding genetics, bioinformatics, and unsupervised one-class models.

Chapter 3: Research Methodology

In this part, we describe the methodology followed and the datasets used. A detailed description of the methods and data followed for our different research outcomes is given in chapters 4 to 6.

Chapter 4:

This section aimed to establish a relationship between these non-coding RNAs and breast cancer by using sequence features extracted from the binding of miRNAs and the ERBB2 genes. For our research purposes, we established a comparison of one-class models as Isolation Forest and One-Class SVM.

Chapter 5:

In this chapter we applied sequence features for validating non-coding RNAs, miRNAs and lncRNAs, in discriminating between breast cancer and breast neoplasm situations.

Chapter 6:

In this section, we used sequence features and one-class SVM to predict miRNAs' binding to the mRNA SARS-CoV2 mRNA. Additionally, we compared our one-class model with supervised models to show that the one-class model is more suitable for imbalanced datasets.

Chapter 7: Conclusions and future work

We end the work with a section on our Conclusions and a description of the research contribution, providing set of references reviewed and the necessary appendices.

1.7 Summary

In this chapter, we have presented a broad view of the involvement of miRNAs in certain diseases and how unsupervised models could serve as a valuable technique to predict bindings between miRNAs and genes or to validate their interaction. This study has identified several challenges involved in the research of non-coding RNA interactions with diseases. One is related to the scarcity of validated samples, which would enable the direct application of supervised models. The second involves the feature manipulation that should be performed in using the nucleotide-binding characteristics between miRNAs and genes or mRNAs in contraposition with the expression of miRNAs. The latter is derived from the natural imperfect matching in the miRNAs of animals, in contrast to those found in plants.

CHAPTER 2
Literature Review

2.1 Introduction

In this chapter, we introduce some basic concepts of genetics and bioinformatics. We further describe one-class unsupervised models, their characteristics, and differences from two-class models. Additionally, in the following chapters, we review several articles that describe the use of these one-class models to predict interactions between non-coding RNAs' and genes, viral RNA, and various diseases.

2.2 Basics of Genetics and Bioinformatics

This section describes some basic concepts on RNA, the definition of non-coding RNAs, and the description of a pair of relevant molecules in this classification, known as microRNAs (miRNAs) and long non-coding RNAs (lncRNAs). We also mention some bioinformatics algorithms designed to support in-silico computational experimentations, as an alternative to laboratory or wet-lab experiments. Of note, most of the material in Section 2.2 is based on the work of Hartl (2020), except where otherwise is indicated.

2.2.1 Central Dogma

The biological importance of the process of creation of proteins by genes in the human body is undeniable. This process, when altered, can lead to diverse diseases. However, even though the fabrication of proteins is controlled by information encoded within DNA, this control procedure is not direct and follows the Central Dogma rule. The Central Dogma refers to the fact that genetic information encoded in the DNA is not translated directly to proteins, but requires an intermediate molecule known as RNA (ribonucleic acid) to perform a transcription process. The RNA contains information encoded in nucleotides, which are conventionally denoted by four characters A, C, U, G (DNA has the same structure, except that the nucleotide U corresponds to a T base), and this simple semantic mapping suggests the basis for the field of computational genomics. A particular type of RNA known as messenger RNA (mRNA) transports DNA information to be used as a template to initiate the translation process. After the translation process takes place, a polypeptide chain of proteins is created, comprising three nucleotide base blocks known as codons. Of note, the DNA double helix contains two portions known as 3' to 5' and vice versa; RNA starts at the 3' portion when it

copies information from the DNA. This process of replicating DNA and subsequent transcription and translation into proteins by the mRNA and other RNA molecules involved seems like a straightforward procedure. However, in plants and animals, the silencing of protein production by some genes occurs due to the appearance of some unique RNAs called double-stranded RNA (dsRNA) in a process called RNA interference (RNAi). These types of RNA are called small interfering RNA (siRNA), and their function was long unknown, and they were even considered junk DNA.

2.2.2 Basic sequence algorithms in Bioinformatics

This study focuses on the extraction of features from the different peculiarities occurring in bindings between non-coding RNA and genes. Notably, this binding relates to the full or partial complementarity between these two genetic molecules' nucleotides when they match or bind together. For the purposes of this research, it is convenient to describe in more detail how a pairwise alignment between these molecules occurs, the energy released when this procedure takes place, and, finally, the concept of *k*-mers, a grouping of a certain number of nucleotide bases.

a) Sequence Alignment Algorithms

The similarities between a pair of genetic strands composed by different nucleotides or base pairs may be relevant to such investigations. One method used to determine this similarity is to compare both sequences with a scoring function. These functions consider a positive score for a match and a negative score for gaps that could be allowed to match pairs of nucleotides from both strands (Needleman, 1970). Usually, a couple of strands are put formulated into an array structure for computational processing. In such arrays, as an alternative to the use of indices, nucleotides can be recorded in each row and column index. For example, in Fig. 2.1, we show the pairing of two small DNA sequences.

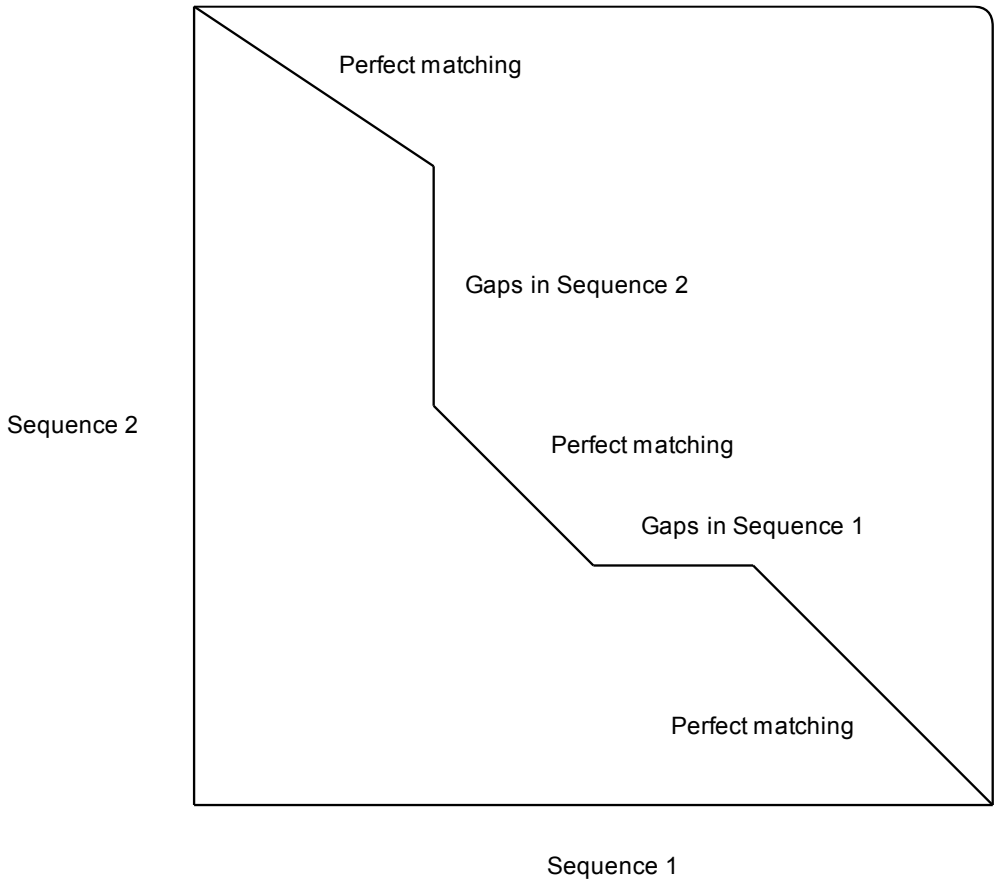


Figure 2.1 Global alignment between a pair of sequences (based on Needleman, 1970).

In Fig. 2.1, it can be noted that when a perfect match is obtained between a pair of nucleotides, such as A-A or C-C, a diagonal line is formed between them. The occurrence of gaps between a pair of sequences allows probable matchings when the alignment surpasses these gaps. For example, a pair of sequences, one being ACT and the other ACGT, we may obtain the following match.

AC-T

ACGT

This alignment is represented by a vertical line in our graph because nucleotide G matches a gap represented by a dash. A similar situation can occur when a gap is present in the opposite sequence. Notably, each gap has a penalizing score which could be used to penalize large portions of gaps. Scoring in nucleotides could have a value of +1, indicating a match, or a value of -2 when a mismatch or gap occurs. However, in aligning a pair of protein strands composed by joining amino acids containing three nucleotides or codons, special matching matrices are used, such as the BLOSSUM 62 matrix. Finally, the alignment score is

calculated by backtracking the path of matches and mismatches and using dynamic programming (Pevsner, 2015).

b) Binding and Minimum Free Energy

RNA presents different organizations related to their internal composition. The easiest way to visualize an RNA strand of nucleotides is simply as a sequentially arranged string of nucleotide, e.g.,

5'AAUUGCGGGAAA...UUCA3'

This formation is known as the primary structure. However, an RNA strand can be represented in a secondary and tertiary formation, as depicted in Fig. 2.2.

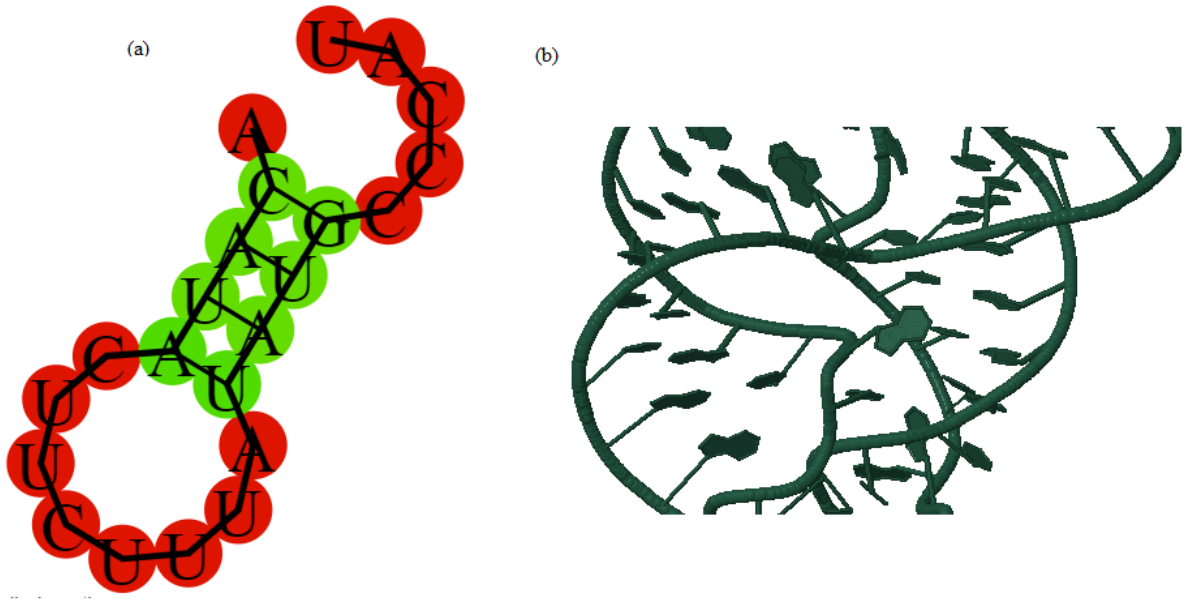


Figure 2.2 Secondary structure from the miRNA hsa-miR-1-5p (a) and the tertiary structure of the mammalian signal recognition particle (image a) generated with RNAFold Web Server and image (b) generated with RNA Composer).

In Fig. 2.2 (a), we can observe the formation of a hairpin loop in the lower part of the strand and a complementary matching between nucleotide bases in the middle. Many methods have been developed to predict secondary structures based on the principle of thermodynamics. These thermodynamic principles state that these molecules are more stable when they present lower energy, which is related to the concept of minimum free energy (MFE). For example, Fig. 2.3 shows a secondary structure with some matchings and mismatchings. The energy of

the total model was calculated based on the energy of the adjacent or neighboring nucleotides.

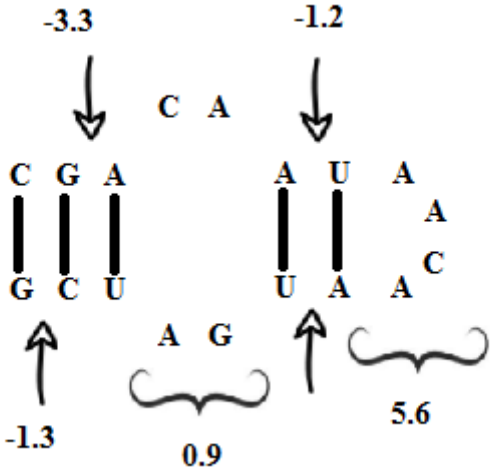


Figure 2.3 Prediction of the total free energy from an RNA sequence (based on Sloma et al. 2020).

As one can see in Fig. 2.3, those sites with matching have a negative value, while those that present a gap or have a loop present a positive score. This concept reinforces the idea that when the total summation of the numbers shown is more negative, the molecule is more stable (Sloma et al. 2020). This principle of energy-stability also applies when two molecules bind together when they are involved in the dimerization process.

c) K-mers

A *k*-mer is defined as a short substring of a predefined length; *k* simply denotes the number of nucleotides. For example, if we consider the following DNA string,

AAACCTGGACCTT

a 2-mer will be the joining of a pair of nucleotides that gives

AA, AA, AC, CC, CT

In this example, this study also considers another term called the sliding window, which refers to a process of traversing the DNA sequence by considering a single nucleotide at a time with a given quantity. The number of probable *k*-mers in a sequence is given by the formula 4^k, where *k* represents the number of mers considered. For example, a value of *k*=2 implies 16 different combinations considering our four nucleotide bases; *k*=3 implies 64

different combinations, and so on. The use of *k*-mers has various applications in bioinformatics, such as in the reconstruction of sequences given these terms. However, one exciting application involves finding shared *k*-mers between a couple of genomes or nucleotide strands; this sharing could serve to find synteny blocks defined by similar genes in the same order in different genomes. This *k*-mer analysis can also be analyzed in complementary parts of different genomes (Compeau and Pevzner, 2015). It is useful to recall here that a complementary strand occurs when A binds to T, and C binds to G.

2.2.3 Non-coding RNAs: miRNAs and lncRNAs

Of note, a secondary structure called a stem-loop or hairpin loop is associated with the appearance of dsRNA. A stem-loop occurs when an mRNA folds, but it ends with some nucleotides become or remain unpaired, resulting in observable loop structures (Scitable, 2014). This stem-loop structure usually contains mismatches that rise to miRNA molecules, which were first found in Caenorhabditis elegans nematode in the transcription process of a locus (position) lin-4 of these species (Lee, 1993).

miRNAs are produced in the cytoplasm, and they use an enzyme that cleaves dsRNA in small single-stranded pieces and is called dicer. These chunks of 25 nucleotides are then aggregated into an RNA-induced silencing complex (RISC). The two strands generated by the dicer serve as a guiding RNA targeting the RNA by complementary base-pair. A complementary base-pair occurs as a matching between nucleotides in the pattern A-T, C-G; however, in RNA, the T base is replaced by a U base. The RISC complex components differ among species, but this difference occurs more frequently in a component called argonaute. Interestingly, following the formation of the RISC complex, the two molecules that appear, namely siRNA and miRNA, act differently. siRNA usually shows perfect or almost perfect complementarity with the target RNA; however, this is unlikely to occur with the miRNA. This is because the guide and target RNA originate from different parts of the genome. This characteristic allows multiple miRNAs to target multiple genes (Prosenjit et al., 2007). The RISC complex then attaches to the RNA, and in this process, it can destabilize the mRNA and inhibit mRNA translation. This effect on translation can result in the regulation of genes across different species. They are directly related to the cell formation process, which could directly influence diseases such as tumoral formations.

Another group of molecules also influence gene expression, which are called long non-coding RNAs (lncRNAs). These molecules are approximately 200 nucleotides long and are

RNA molecules that do not translate into proteins. The particular characteristic of these RNAs is that they originate near protein-coding genes; therefore, they could include 5' and 3' non-coding regions or even exons and introns. Some of these lncRNAs are degraded, but they can also affect gene regulation because of their abundance. In conclusion, molecules derived from RNA, such as miRNAs and lncRNAs, influence the normal functioning of a gene, and this intervention could lead to the development of different diseases.

2.2.4 miRNAs and lncRNAs in disease scenarios

In this section, we focus on describing the involvement of miRNAs and lncRNAs in certain diseases. For the purposes of this research, this study focuses on breast cancer scenarios and the SARS-CoV2 coronavirus.

miRNAs can be involved in the occurrence of certain diseases, such as cancer. Chen et al. (2018) demonstrated that mir-25-3p was upregulated in cases of triple-negative breast cancer (TNBC), which is a form of breast cancer that presents in younger patients, and is associated with a less encouraging prognosis due to high metastasis rates (Gupta et al., 2019). In Loh et al. (2019), the author described a set of miRNAs named OncomiRs that suppress the expression of genes involved in tumor suppression in breast cancer scenarios, leading to breast tumorigenesis. Therefore, the importance of the study of miRNAs aids in the diagnosis of cancer scenarios, as they can be used as potential biomarkers and the prognosis of different types of cancer (Prosenjit et al., 2007).

We have described cancer scenarios; tumors also occur which may not necessarily be malignant, which are known as neoplasm. For example, according to Coleman (2020), specific differences exist between neoplasms and cancers. This author defines a neoplasm as irregular growth that can occur in any tissue. However, Kinzler and Vogelstein (2002) differentiate it from cancer, stating that cancer is like an abnormal growth but is prone to affect surrounding tissues. It is valuable to note that there could also be benign and malignant tumors or neoplasms, and the latter are classified as cancer types (Coleman,2020).

LncRNAs are a category of non-coding RNAs that also appear in breast cancer scenarios, for example, in the formation of metastasis in breast tissue by modifying the chromosome landscape. Another example will be the lncRNA HOTAIR that influences HOXD genes (Harries (2012)) in the case of tumor suppression, for instance, as it occurs in the presence of the Gas5 lncRNA (Wapinski and Chang, 2011) or in the expression that occurs in LSINCT5

(Harries, 2012). miRNAs also participate in breast cancer and metastasis scenarios, acting as tumors or oncogenes, as noted above.

It has been demonstrated that miRNAs can also bind to other forms of mRNAs, such as viruses (Lamkiewicz et al., 2018; Trobaugh and Klimstra, 2017; Nersisyan et al., 2020), in which case the mRNA is from an endogenous RNA. Their function in viruses is the same as those found in humans and in plants in that they regulate the translation of proteins or even influence viral reproduction. The possibility of compatibility of human miRNA with viral RNA is relatively simple, owing principally to the fact that the miRNA cannot differentiate between species in this process (Nersisyan et al., 2020). Even though this scenario can occur, there is still no firm evidence that the mRNA from viruses could also produce miRNAs that could affect the host organism (Yousefi et al., 2020). Nevertheless, because of the current pandemic, there is great interest in identifying human miRNAs that bind to the SARS-CoV-2 virus. The studies of these interventions help determine whether miRNAs could interfere with the viral RNA functions, for example, by stopping their replication into the human body or using them as potential biomarkers (Jafarinejad-Farsangi et al., 2020).

2.3 One-Class models

2.3.1 Isolation Forest

The isolation forest is an anomaly detection technique proposed by Liu et al. (2008) based on the ensemble method known as random forest. It operates by dividing a subspace into regions on the hypothesis that outliers could be enclosed in regions that do not require splitting a decision tree into many partitions. The leaf's distance, in which the outlier is present, to the root serves as an outlier score (Aggarwal, 2017).

2.3.2 One-class SVM

The one-class SVM model was designed to perform, novelty detection with only one training class, as proposed by Schölkopf et al. (2001). It returns a value of +1 if the data are enclosed within a region, and -1 otherwise. This model generates a mapping using kernels to separate the data from the origin by a maximum margin (Schölkopf et al., 2001). In contrast, classic SVM models require at least two classes that could be separated by a decision boundary. By design, the data is not labeled, but it can be marked as positive or negative to support the application of metrics to assess the quality of our model.

The notion of kernels, as in SVM, is also applied to a one-class SVM to transform a set of data points to another dimension by using the kernel function. The decision boundary in this model is based on

$$\bar{W}.\Phi(\bar{X}) - b = 0 \quad (1)$$

In Eq. 1 $\Phi(\bar{X})$ corresponds to the transformation of $\bar{X}$ into a higher-dimensional space, and b is a bias variable. We aim to formulate this as an optimization problem in which the value of $\bar{W}.\Phi(\bar{X}) - b$ is positive for holding as many of the examples that belong to the N training set, because we believe that most of the samples will be enclosed in the positive class. Therefore, if we have the contrary case in which $\bar{W}.\Phi(\bar{X}) - b$ is negative, we can apply a slack penalty of $max\{b - \bar{W}.\Phi(\bar{X}).0\}$. In this case, we are rewarding that the origin is farther away from the separating hyperplane. Considering the further necessity of a regularization term $\frac{1}{2}\|\bar{W}\|^2$ leads to the following objective function, given as Eq. 2.

$$Min\ J = \frac{1}{2}\|\bar{W}\|^2 + \frac{C}{N}\sum_{i=1}^{N} max\{b - \bar{W}.\Phi(\bar{X}), 0\} - b \quad (2)$$

2.4 Related studies

The importance of the study of non-coding RNAs and their relation with diseases has been an active focus of study over the years. However, the undertaking of experiments made in-vitro has been replaced for their computational counterparts (Zheng et al., 2019). These computational experiments are diverse, ranging from the prediction of mRNA hairpin structures (Tran et al., 2008) to the prediction of diverse diseases such as leukemia or other forms of cancers like the ones mentioned in various studies (Spinosa and de Carvalho, 2004 or Rehman et al., 2019, Loh et al., 2019; McAnena et al., 2019). In this field of genetics mixed with computing or Bioinformatics, the use of Machine Learning models has received particular importance. For example, Sedaghat et al. (2018) used supervised and unsupervised techniques for predicting miRNA and mRNA biding. However one problem, that occurs with most supervised techniques is that they need the presence of two or more classes; a situation that could lead to the appearance of imbalanced datasets due to the difficulty or scarcity to find samples from one of the opposite classes (Sedaghat et al., 2018; Irigoien et al., 2014). Even though this problem with the data could occur, unsupervised techniques or models that use only one class to discriminate their components into what we could call opposite classes have found their niche for dealing with this problem of imbalanced data (Yousef et al. 2008; 2010). The data extracted and used as features for these ML models are mostly related to

gene expression between miRNAs and genes; however, some studies have considered the extraction of information from the binding of genetic sequences analyzing a specific number of nucleotides bound known as k-mers. Some early studies showed the possibility of studying this k-mer complementarity between these genetic components, for example, in plants. This particular characteristic was due to their binding of these genetic sequences presenting a good complementarity (Zhang et al., 2020); or using these features with CNN like Wen et al. (2019).

Concerning the study of miRNA and their interaction with diseases, such as some forms of cancer, it is worthy of mentioning that these are not the only non-coding RNA considered. For instance, other forms like lncRNAs can also be studied. For example, Guo et al. (2019) applied kernel profile techniques with autoencoders and a random forest to predict the relationship between lncRNAs and colorectal cancer. The relationship between lncRNA and mRNA and their association with different species like humans, mice, and chickens using k-mers frequency analysis with CNN model was also considered Wen et al. (2019).

The study of miRNA and other forms of RNA interactions, like those found in viral forms, has also been an exciting part of research in later years. Some authors like Lamkiewicz, 2018; Trobaugh, 2017; Nersisyan et al., 2020 demonstrated that the human miRNA could also bind to viral RNA. This exciting feature is due that miRNA is not able to differentiate between the host mRNA or viral mRNA (Nersisyan et al., 2020). These studies could be rather interesting because some researchers like Jafarinejad-Farsangi et al. 2020 mentioned that miRNAs could be used as biomarkers for several diseases, bringing novel genetic treatments for diverse diseases.

2.5 Summary

This chapter has reviewed some basic biology germane to the following chapters. We have also described some characteristics of the one-class methods used. Additionally, we gave a brief review of the bioinformatics algorithms utilized for extracting features from our datasets to be used in our supervised two classes and one-class models.

CHAPTER 3
Research Methodology

3.1 Introduction

This chapter describes the general methodology adopted in Chapters 4 to 6. For details of the exact methodology followed for each deliverable of the current thesis, the reader is suggested to visit the chapters mentioned above. Various non-coding databases were used to extract features based on their nucleotide characteristics with a focus on miRNAs and lncRNAs. Additionally, this study also extracted information related to genes involved in immunological processes in the human body and genes present in diseases such as breast cancer, neoplasms, and viral diseases such as the SARS-CoV-2.

3.2 Methodology

Our experimental research procedure is illustrated in Figure 3.1:

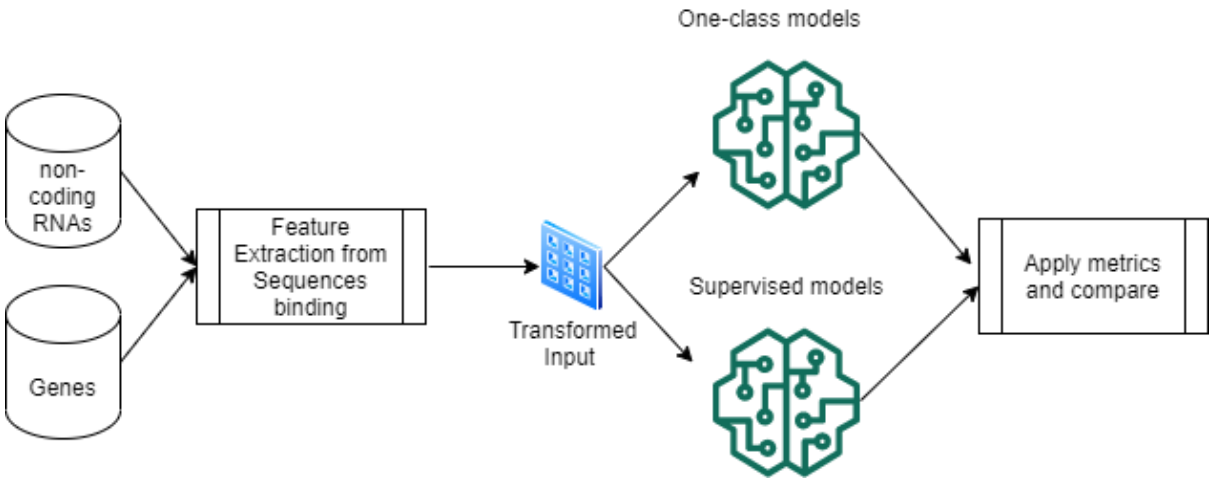


Figure 3.1 Proposed methodology

In general, we extracted information on the non-coding RNAs, lncRNAs, and miRNAs related to their sequence and ID name or alias. We also obtained information on genes associated with the entire course of breast cancer, viral forms of mRNA, and genes involved in the human body's immune processes. With these two types of information, this study predicted or validated the possibility of binding between non-coding RNAs and these genes. This information served as input for our machine-learning models. These machine-learning models need to pre-process the information that serves as their input, a process known as feature engineering. Regarding this process, we used bioinformatics algorithms to extract

information occurring when there is a binding or coupling between these non-coding RNAs and the genes considered in the experiments. These algorithms are focused, generally, on obtaining sequence alignments between miRNAs, lncRNAs, and genes, and the MFE occurring in the formation of the secondary structure of RNAs binding of these molecules.

Considering the feature extraction used for each of the models presented in Chapters 4 and 6, the procedure we made was to extract the features from each dataset containing information about the miRNAs and then match this information to the genes related to these miRNAs. The same procedure was applied to match diseases and genes or miRNAs involved. The attributes selected were numerical primarily, so there was no need to apply transformations from categorical data, such as the use of one-hot-encoding or similar techniques. However, we applied the min-max regularization technique to ensure that our numeric data was within a specific range between 0 and 1. Additionally, when we needed a two-class model, and for labeling purposes, we chose those non-coding interactions with genes or mRNA that have a strong interaction backed up by experimental or literature support. For the opposite class were those that have predicted interactions or were non-supported by the literature ones. The specific details for extracting these features are described in Chapters 4 to 6.

This study used a one-class SVM and isolation forest to detect the existence of outliers in our data, which is also known as novelty detection of interesting bindings between non-coding RNAs and genes. These binds are important because they allowed us to hypothesize their relationship in the outcome or prognosis of breast cancer, breast neoplasms, or binding to the SARS-CoV-2 RNA gene. We aimed at using machine learning supervised learning models such as SVM and random forest. Also. we applied the grid search algorithm for tuning the hyperparameters of the different models and used cross-validation or modification of this algorithm to obtain the necessary metrics to evaluate our models. Accuracy and F1-score metrics were employed, along with a schema of scoring based on weights to compensate for the possibility of data imbalance.

3.3 Software tools and Databases used

The databases used are described in Chapters four to six in detail. We used the software tools Python, BioPython, and the Vienna package to extract RNA energy values from their binding and secondary structure.

3.3.1 Brief description of the databases used

a) MirWalk

It is a database that uses the Watson-Crick complementarity between genetic sequences to find probable binding between miRNAs and a specific gene (Dweep et al., 2011; 2013)

b) miRTargetLink

It provides a star-type graph showing a list of miRNAs and their relationship with specific genes. Their results are validated by experimental means or prediction techniques (Hamberg et al., 2016).

c) lncRNASNP2

This dataset contains information about lncRNA relationships with diseases. The information found in this dataset is backed up by experiments and publications (Miao et al., 2018).

d) NCBI FASTA sequences

For gathering information about the SARS-CoV-2 Genome (Severe acute respiratory syndrome coronavirus 2 isolate Wuhan-Hu-1, complete genome, 2020), we have used the data available at NCBI with accession number obtained from GenBank.

e) miRbase

This dataset contains information about miRNAs found in different species, but we had concentrated our interest in human miRNA (Kozomara, 2014).

f) InnateDB

We needed to obtain a list of genes involved in immune processes in the human body, and this data was gathered from the InnateDB (Breuer et al., 2013).

3.3.2 Brief description of the software used

a) BioPython

Biopython is a set of libraries that can be used under Python to manipulate genomic sequences, allow sequence alignment via different algorithms, and interact with other genetic databases (Cock et al., 2009). It is valuable to mention that many of the processes we have made in the present study using this package could have been done from scratch. Additionally, we chose Python because of its easiness to manipulate string sequences,

considering that genetic sequences could be seen as a string formed of nucleotides; other tools like R might also be used.

b) Vienna package

The Vienna package (Hofacker, 2003) is a set of libraries programmed in C that helps predict RNA molecule's secondary structure. Even though this package is oriented to C, it could be easily imported to work with Python. From this package, we were interested in the use of two libraries called RNAfold and RNAcofold functions. The RNAfold function yields the MFE associated with the formation of a secondary structure by RNA. The RNAcofold also calculates this energy when dimerization occurs; this process appears when two genetic molecules bind together but with a greater degree of binding affinity.

3.4 Summary

This brief chapter provides valuable information on our general methodology and tools used in developing current research described in the following chapters. To avoid redundancy, detailed information considering the databases used and the different modifications applied to the data or model are described with sufficient detail in Chapters 4-6.

CHAPTER 4
One-class models for validation of miRNAs and the ERBB2 gene interaction by using sequence features

4.1 Introduction

In past years, researchers discovered the capability of predicting miRNAs and gene interactions by grouping miRNAs that promote or decrease gene expression (Yousef et al., 2010). These computer models require less time and resources than their in vitro counterparts (Zheng et al., 2019)., but the well-known no-free-lunch theorem also applies here. In order to train a model for classification, certain methods, such as supervised learning, require labeled data to distinguish if a sample belongs to a particular class. Several authors have pointed out this problematic situation in the analysis of miRNA and mRNA interactions (Tran et al., 2008; (Sedaghat et al., 2018; Yousef et al., 2008; 2010). Similarly, contexts involving a limited set of interaction samples or the presence of some weak or unrepresentative interactions could result in heavily unbalanced data.

The challenge of predicting miRNA hairpins from mRNA hairpin topologies was described by Tran et al. (2008). MiRNA hairpins with lengths ranging from 21 to 25 nucleotides can theoretically be made from RNA hairpins with sizes of 60 to 90 nucleotides. In this particular instance, the difficulty is that the available dataset of miRNA hairpins was somewhat small. For this reason, using a two-class classifier was not possible, whereas the use of one-class model was. In summary, there were two major issues: first, obtaining labeled data was complex due to a lack of validated or weak miRNA-mRNA interactions datasets. Second, such datasets may be imbalanced. In either case, some limiting factors are evident in the straightforward application of supervised classifier models.

One-class classification or novelty detection refers to the development of computer models designed to find evidence of the presence of a given class in a single set of data. In the study of Spinosa and de Carvalho (2004) they pointed out the utility of using this type of anomaly detection in bioinformatics. Specifically, the authors used a one-class SVM for recognizing ALL-B leukemia samples in a dataset that included specimens from different types of the disease, including ALL-T and AML. The dataset they selected contained only a small number

of records, ranging from 17 to 27 or 30 registers per leukemia class; however, it had many attributes, with roughly 7000 features in all. The accuracy for each dataset differed because the authors employed diverse hyperparameter tuning for their different models. According to their findings, the AML type attained an accuracy of roughly 85% for the ordinary class, and 60% for the class containing the majority of outliers.

Yousef et al. (2008; 2010) used different models based on one-class algorithms to predict the existence of miRNAs by using the secondary structure or sequence information as features from these molecules. The authors advocated for the adoption of a one-class model since obtaining negative data based on the positive miRNA class is typically a complex and biased operation. To validate their proposal, they predicted a set of miRNAs linked to the Epstein-Barr virus. They found sensitivity values of 72% and specificity values of 99% when employing secondary structure characteristics in human data in conjunction with a one-class SVM. However, the study does not explicitly mention hyperparameter tuning.

Rehman et al. (2019) used machine-learning classifiers to validate miRNAs associated in breast cancer. The dataset utilized in this work was obtained from the National Cancer Institute's Genomic Data Commons Data Portal (Jensen et al., 2017), which comprised samples from 1207 individuals with 1881 miRNA attributes. The samples included 1103 tumoral samples, 104 healthy samples and seven metastatic samples. This study showed an imbalance between the number of patient records and features, with the former being underrepresented. For this reason, the authors advocated using feature selection techniques such information gain, chi-squared, or least absolute shrinkage and selection operator (LASSO) to choose the most relevant miRNAs for use as features in SVMs and random forest classifiers (Rehman et al., 2019).

Two significant approaches have been pursued in the study of miRNA and mRNA interactions, one relating the study of the characteristics of the part of the sequences involved in the binding, such as accessibility, evolutionary conservation data or pairing sites. A second strategy takes into account the negative association between miRNA and mRNA expression levels (Pham et al., 2019). In this part of our research, we adopted a sequence-based technique.

Concerning the use of validation of miRNAs and mRNAs in cancer scenarios by using features obtained from the sequence interactions reactions, we found no research work focusing on this scenario using unsupervised models, other than the studies of Yousef et al.

(2008; 2010). We note that whereas Sedaghat et al. (2018) suggested a combination of supervised and unsupervised strategies for miRNA target prediction, the findings obtained favored SVM-supervised binary classifiers. We believe that unsupervised techniques could discover interesting reactions between miRNAs and cancer genes, which might be inadvertently missed using supervised techniques. Additionally, a drawback in the study of genes and miRNA interactions involves the complexity of obtaining samples from the positive (or negative) category in the right amount (Sedaghat et al., 2018; Irigoien et al., 2014), even with the possibility of resulting in an imbalanced dataset categorization situation, these scenarios complicate the deployment of supervised models with two or more classes.

Regarding our one-class model categorization, we used two well-known techniques: isolation forest and one-class SVM. Concerning the sequence features of miRNA and gene interactions found in these relationships, we used the data from mirWalk, as opposed to Tran et al. (2008), Sedaghat et al. (2018), Yousef et al. (2008), and Yousef et al. (2010) that used gene expression measurements, which may result in unbalanced data. MirWalk was discovered to be based on data derived from the sequence-based mechanism that occurs during miRNA interactions. However, the dataset must be processed to obtain a subset of the negative or opposite classes. This manipulation will allow us to use metrics such as precision or specificity to validate our models. We decided to test our approach by considering the interactions between a gene of interest and miRNAs that present weak or do not have literature support evidence for obtaining the opposite class needed. The chosen miRNAs interact with the ERBB2 gene, whose expression can contribute to breast cancer, and are validated utilizing miRNA-gene interaction instruments like mirTargetLink (Hamberg et al., 2016).

4.2 Background

4.2.1 miRNA and mRNA interaction

Multiple miRNAs have been observed for diverse mRNA targets and vice versa (Loh et al., 2019). The biogenesis process of miRNAs starts with the formation of a molecule known as pri-miRNA, which, by the action of the nuclear RNASE III Drosha, it cleaves the primiRNA to form a hairpin-shape-based pre-miRNA (Han, 2004), also known as a precursor miRNA (Loh et al., 2019). This pre-miRNA is transported to the cytoplasm and is affected by the Dicer and TRBP complex to form a mature miRNA duplex. This miRNA duplex is placed on an Argonaute (AGO) protein with an RISC, which unwinds the miRNA duplex into two

miRNA strands, one named mature miRNA and the other as a passenger strand (Loh et al., 2019). The process of binding of the mature miRNA and mRNA can affect the translation of mRNA or induce its degradation (Loh et al., 2019; Sarshad et al., 2018).

4.2.2 miRNAs in Breast Cancer

The role of miRNAs in breast cancer scenarios is undeniable, being present in the appearance of tumoral masses or even helping to predict the prognosis in some scenarios (López-Camarillo and Marchat, 2013). By interacting with specific genes, miRNAs can increase or decrease their gene regulation (Negrini and Calin, 2008) or even degrade them (Bartel, 2004; Loh et al., 2019; López-Camarillo and Marchat, 2013; Negrini and Calin, 2008). This abnormal situation could result in the formation of oncogenes or tumor suppressor genes in diseases such as breast cancer (Loh et al., 2019; Negrini and Calin, 2008). An oncogene is a mutated gene that could be involved in abnormal cell growth that could lead to cancer prognosis, while a tumor suppressor gene restrains a specific protein that acts inversely as an oncogene, which is why they are called antioncogenes (NCI, 2020). In breast cancer, miRNAs can act as oncogenic miRNAs (oncomiRNAs), which are usually upregulated in these scenarios by suppressing tumor suppressor genes. While tumor suppressor miRNAs (tsmiRs) act as inhibitors of oncogenes being downregulated, they could lead to a breast malignancy scenario (Loh et al., 2019). For example, miRNA-331 interacts with ERBB2 or HER2 gene by promoting the metastasis in breast cancer patients (Loh et al., 2019; McAnena et al., 2019), miR-124a and miR-26b interact with SerpinB2 (Loh et al., 2019) which, according to the data mining from literature online resource knowns as Diseases (Pletscher-Frankild et al., 2015), this gene is involved in breast cancer scenarios as an anti-metastasis agent.

In this part of our research, we focus on a subset of miRNAs that interact with HER/ERBB2 genes. These genes are among the molecular subtypes of breast cancer, accounting for approximately 15% to 20% of all active cases (Sareyeldin et al., 2019). The importance of this subtype is its notable aggressiveness. Additionally, the miRNAs could act as a promising biomarker in focused gene therapy, especially in HER2-positive breast cancers (Sareyeldin et al., 2019). HER2 interacts with the ERBB2 gene, as the latter is the target of trastuzumab medical treatment (Patel et al., 2020) in HER2-positive breast cancer cases.

4.2.3 Databases related to miRNA, mRNA, and disease interactions

a) MirWalk

According to (Dweep et al., 2011; 2013), the mirWalk algorithm as a predictor uses the Watson-Crick complementarity to find probable binding sites between miRNAs and a target gene. This search starts from a heptameter (seven nucleotides) from positions marked as 1 or 2, a section known as seed value in the miRNA, and extends the search until a mismatch is found. The binding sites found are marked with start and end indicators, along with the region in which it occurs. The region or promoter region could be the 5'-UTR, the 3'-UTR, or the CDS or coding sequence region. MirWalk (MirWalk, 2020) uses TarPMir (Ding et al., 2016) as a prediction tool by using a random forest model to determine the probable binding sites. Some of the features used by mirWalk are briefly described in Table 1 and can be found in the article by (Ding et al., 2016).

b) miRTargetLink

miRTargetLink (Hamberg et al., 2016) is an online tool that allows the retrieval of a list of miRNAs with a relationship with a specified gene. Their output consists of a star-type graph that provides information about strong, weak, or predicted interactions. Strong interactions are those that have validated experimental results using, for example, the luciferase assay. Those with weakly validated results used microarray analysis techniques, and the predicted interactions were retrieved by querying databases such as miRanda. For our research, we selected a subset of strong and weak interactions given by miRTargetLink (Hamberg et al., 2016), as shown in Figure 4.1. These data will become the input for the one-class SVM classifier to detect novelties or outliers in a training set (strongly validated interactions) and a test set (weakly validated interactions).

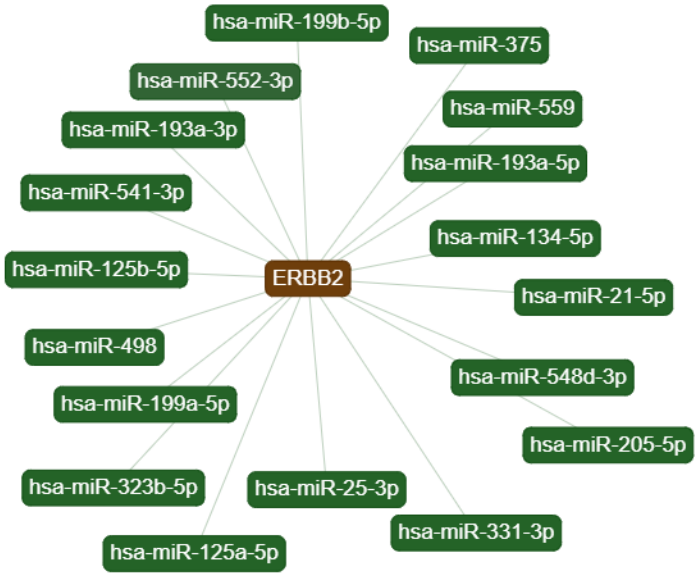


Figure 4.1 miRNA validated strong interactions with the ERBB2 gene obtained by performing a search query in miRTargetLink (Hamberg et al., 2016)

4.3 Methodology

Our methodology consists of two main parts. First, we extracted the miRNA-mRNA sequence binding characteristics of mirWalk (mirWalk, 2020). In the second step, we gathered a set of miRNAs that interacted with the ERBB2 gene, a list obtained from miRTargetLink (Hamberg et al., 2016). From these samples, we extracted some of them to form part of a minor validation subset that would work as a negative class. This negative class contains weak or no interactions, as validated in the literature. Hereafter, a more refined description of the steps is as follows:

a) Select a tuple of genes and miRNAs that, by being the gene expression upregulated or downregulated by an miRNA's interaction, could result in a probable cancerous state. For our proof of concept, we chose the ERBB2 gene.

b) The list of gene-miRNA interactions is divided into strong and weak interactions. We will obtain this information from databases such as mirWalk (2020) and validate them via a literature review.

c) We divided our dataset into two subsets, one containing strong and validated interactions between the ERBB2 gene and miRNA, and the other containing weak interactions or those not validated by the literature. Afterward, we will use the necessary metrics for validating these interactions, such as sensitivity and specificity.

d) To determine the presence of outliers in our dataset, we used an isolation forest model. These outliers would represent components with weak interactions. The procedure (c) and the application of the isolation forest model enabled us to acquire the metrics necessary to evaluate this initial model.

e) We applied a one-class SVM classifier on the two datasets acquired in step c) to verify for the occurrence of outliers. These outliers were interpreted as weak interactions between the gene and miRNAs. We will compare the results produced from both models, one-class SVM and isolation forest, utilizing a confusion matrix and obtaining their respective metrics such as accuracy, sensitivity, specificity, and F1-score.

Of note both one-class models are fitted to the training and test datasets from step c). We hypothesize that the first dataset, which contains strong interactions, should have few outliers, but the second dataset, which includes weak interactions, should contain more than half of the outliers. We corroborate our findings by examining a confusion matrix with its accuracy, recall, precision, and F1-score metrics and by analyzing some of the outliers identified throughout a review of the actual literature (Gutiérrez-Cárdenas and Wang, 2021a).

4.3.1 Extraction of data categorization of samples

For our experimental procedures, we used data downloaded from the mirWalk (2020) webpage. The mirWalk dataset contains miRNA and gene interactions predicted and validated by wet-lab experiments. The data records are in CSV format, and it gives us a set of attributes that we describe in Table 4.1, and is based on the articles of Stitch et al. (2018) and Dweep et al. (2011, 2013, 2014):

Table 4.1 Features present in mirWalk (mirWalk, 2020) for miRNA and gene interactions.

Attribute Name	**Description**
mirnaid	Contains the id number of a selected miRNA.
refseqid	Identification number that points to the NCBI reference sequence database.
genesymbol	Human gene symbol.
start	Start binding position. It considers heptamer sequences

	(seven nucleotides or longer) using the Watson-Crick complementarity.
end	End binding position.
bindingp	Indicates the binding probability. Higher values should be considered as the best ones. This p-value is obtained from the application of the random forest model by TarPmirR (Ding et al., 2016).
energy	Usually, energy should be considered the MFE related to the free energy that arises in the processes of formation of the secondary structure of RNA molecules (Einert and Netz, 2011; Kertesz et al., 2007). However, according to Dweep et al., (2013), other algorithms for measuring the free energy have been considered, but they are not mentioned in the work.
seed	Corresponds to the position of region 1 or 2 in which a heptamer is found. It has a value of 1 or 0 depending on if there was a pairing between nucleotides 2 or 7 of the miRNA (Ding et al., 2016).
accessibility	Energy measure that quantifies how much an mRNA sequence is open to pairing with a miRNA **(**Kertesz et al., 2007)**.**
au	Denotes the Adenylate/Uridylate rich elements (ARES), which are found in the untranslated region of the mRNAs responsible for coding proto-oncogenes and cytokines, among other factors (Chyi-Ying and Ann-Bin, 1995). Importantly, mammalian miRNAs pair to the 3' of the UTR and that the functional sites are embedded or flanked in AU high enriched sites Grimson et al. (2007). Contains the transcript of 30 nucleotides (NT) in the upstream and downstream within the prediction site (mirWalk, 2020).

phylopstem	This feature refers to the stem-loop or hairpin loop (Ding et al., 2016). The hairpin stem presents preserved or conserved regions that are found in paired sites (Mohammed et al., 2013).
phylopflank	This flanking conservation is the average phylop obtained in both 40 nt upstream and downstream in the binding site. It seems that is calculated by the Phast software (http://compgen.cshl.edu/phast/) and it looks for preserved or syntenic sites (Ohler, 2004; Grimson et al., 2007).
me	This feature measures the probability of pairing along different positions of miRNA. Its name derives from match (m) and else (e) (Ding et al., 2016).
number_of_pairings	Number of paired positions in the 3' end (Ding et al., 2016).
binding_region_length	Longitude where the binding of the miRNA and mRNA occurs.
longest_consecutive_pairings	By TarPMir convention, provides the longest consecutive pairs allowing only two mismatches at the end of the 5' region (Ding et al., 2016).
position	The position of the longest consecutive pairs (Ding et al., 2016). The values could be CDS (Coding sequence), 3 UTR (Untranslated region) or 5 UTR.
validated	Contains all valid interactions in mirTarBase (Huang, H; et al., 2019).
TargetScan	Provides information if the miRNA and mRNA interaction is available in the TargetScan database (Agarwal, 2015).
miRDB	Provides information if the miRNA and mRNA interaction is available in the mirDB database (Chen and Wang, 2020).

As shown in Table 4.1, mirWalk provides various useful features regarding validated and predicted miRNA and gene interactions. Most of these features were extracted from the TarPmirR software (Ding et al., 2016), but we also found additional bibliographic material for describing each feature more accurately.

For the gene to be the subject of our study, we chose ERBB2. In terms of miRNAs interacting with ERBB2, we retrieved a list of miRNAs with weak or strong evidence for interaction and predicted them from miRTargetLink Human (Hamberg et al., 2016). We focused only on interactions that present strong and weak support for our proof of concept. In certain situations, we could not locate miRNA and gene interactions due to the absence of the gene name ERBB2 in the file. In this situation, we used the GeneCards database to look up the ERBB2 gene's aliases (Stelzer et al., 2016). Table 4.2 shows the complete list of miRNAs and their evidence support type (according to Hamberg et al. (2016)), gene name or alias, and literature reference pointing to their association with the gene of interest.

Table 4.2 miRNA-ERBB2 interactions according to miRTargetLink *(Hamberg et al., 2016).*

miRNA	**Gene**	**Evidence**	**miRNA**	**Gene**	**Evidence**
hsa-miR-125a-5p	ERBB2	Strong (Vo et al., 2019), (Ninio-Many et al., 2020)	hsa-miR-124-3p	ERBB2	Weak(Wang et al., 2016)
hsa-miR-125b-5p	ERBB2	Strong (Ferracin et al., 2013)	hsa-miR-326	ERBB2	NA(Ghaemi et al., 2019)
hsa-miR-134-5p	ERBB2	Strong(Pan et al., 2017)	hsa-miR-4326	ERBB2	Weak (Martinez-Gutierrez et al., 2020)
hsa-miR-193a-5p	ERBB2	Strong(Xie et al., 2017)	hsa-miR-670-3p	ERBB2	NA
hsa-miR-199b-5p	NEU1	Strong(Fang et al., 2013)	hsa-miR-6739-3p	ERBB2	Weak

hsa-miR-205-5p	ERBB2	Strong(De Cola et al., 2015)			
hsa-miR-25-3p	ERBB2	Strong(Chen H. et al., 2018)			
hsa-miR-323b-5p	ERBB2	Strong(Sugita et al., 2019)			
hsa-miR-331-3p	NEU1	Strong(Zhao et al., 2016)			
hsa-miR-375-3p	ERBB2	Strong(Shen et al., 2014)			
hsa-miR-375-5p	ERBB2	Strong(Shen et al., 2014)			
hsa-miR-498-3p	ERBB2	Strong(Matamala et al., 2016)			
hsa-miR-498-5p	ERBB2	Strong(Matamala et al., 2016)			
hsa-miR-541-3p	ERBB2	Strong(Sareyeldin et al., 2019)			
hsa-miR-552-3p	ERBB2	Strong(Penyige et al., 2019)			

We analyzed the presence of outliers in our dataset by using a boxplot diagram, and we used an isolation forest model and a one-class SVM to corroborate the presence of these outliers. These miRNA and gene interactions with weak evidence would be deemed our subset of

fabricated data, which is similar in concept to producing elements to create a second class for validation purposes in this type of one-class model.

Considering the features shown in Table 4.1, we chose to focus on quantitative characteristics and to eliminate those that were irrelevant. The features not selected included the following: mirnaid, genesymbol refseqid, seed (because they were all set to 1), position, validated, TargetScan, and miRDB. We performed our experimental procedure by using all the remaining features as mentioned in the work of Yousef et al. (2008, 2010). For standardization purposes, we utilized a standard scaler that is comparable to z-score normalization with zero degrees of freedom.

4.3.2 One-class model application and hyperparameter tuning

First, we used the isolation forest model to check for outliers. The hyperparameter tuning metric was the weighted F1-score. The list of tested hyperparameters is given below.

Table 4.3 List of values that has been tested as the hyperparameters of the isolation forest model.

Hyperparameter	Values
Number of trees	List from 10 to 100 trees in intervals of 20 elements
Number of features	List that ranges from selecting the 10% of features until all of them.
Number of samples	Only available when bootstrap is set to True, we set it up to one third of the available samples
Bootstrap	True or False
Contamination	30% of outliers approximately

After testing the above hyperparameters, we ended up with the following best results: number of trees equal to 20, number of features selected 70%, number of samples 30, use of bootstrap set to True, and the contamination level in 30%; we knew this was close to the number of miRNA-ERBB2 interactions with weak support verified from the literature.

In the same way, we tuned the hyperparameters for our one-class SVM model by using the grid search algorithm with a cross-validation of ten folds. The values for testing the best combinations are listed in Table 4.4.

Table 4.4 List of hyperparameters used for the One Class SVM.

Hyperparameter	Probable values
Kernel	Polynomial, Radial Basis Function, Sigmoid
$\boldsymbol{\nu}$	A list with values from 0 to 1 divided in 99 parts
$\boldsymbol{\gamma}$	A list with values: 1e-1, 1e-2, 1e-3, 1e-4,1e1,1e2,1e3
degrees	A list with values from 1 to 6

Prior to employing the grid search algorithm, we made two further adjustments. To begin, we labeled our training and testing sets' outputs to fit our model. As a result, we assigned a value of +1 to samples with a confirmed strong miRNA-gene interaction and a value of -1 to samples with a verified weak interaction. This decision is somewhat debatable because we dealt with an unsupervised model, but it is a technique used to validate a one-class model's results. Without losing generality, this experiment considered only one class, which corresponds to miRNA interactions with the ERBB2 gene, some with strong and others with weak evidentiary support, but all the samples belonging to one class. The second modification we made was to choose an adequate metric for the scoring function of the grid search algorithm. We were unable to use accuracy or precision in this situation due to the unsupervised nature of the model, so we selected a model based on the F1-score related to recall and precision metrics and a weighted average of the results obtained from each cross-validation output. We utilized the F1-score (Aggarwal, 2017) as the scoring function for our grid search method because it is well-suited when there is the presence of imbalanced data for binary classification. Specifically, we calculated the weighted average of the F1-score, which will help us in the case of imbalanced data by assigning more weight to the class with more elements. After applying the Grid Search algorithm we found that the best hyperparameters were to choose an RBF kernel with a N value of 0.17163 and a γ value of 0.1. Of note, in the second round of grid search evaluation, we decided to drop the polynomial kernel with their respective degrees of hyperparameters because this model only performed with good results

in the test set, but when tested in the training set, the average accuracy metrics was approximately 60%. We hypothesize that this is because an RBF kernel outperforms a polynomial kernel in these situations. A summary of the selected hyperparameters is presented in Table 4.5.

Table 4.5 Selected Hyperparameters for the Isolation forest and One Class SVM.

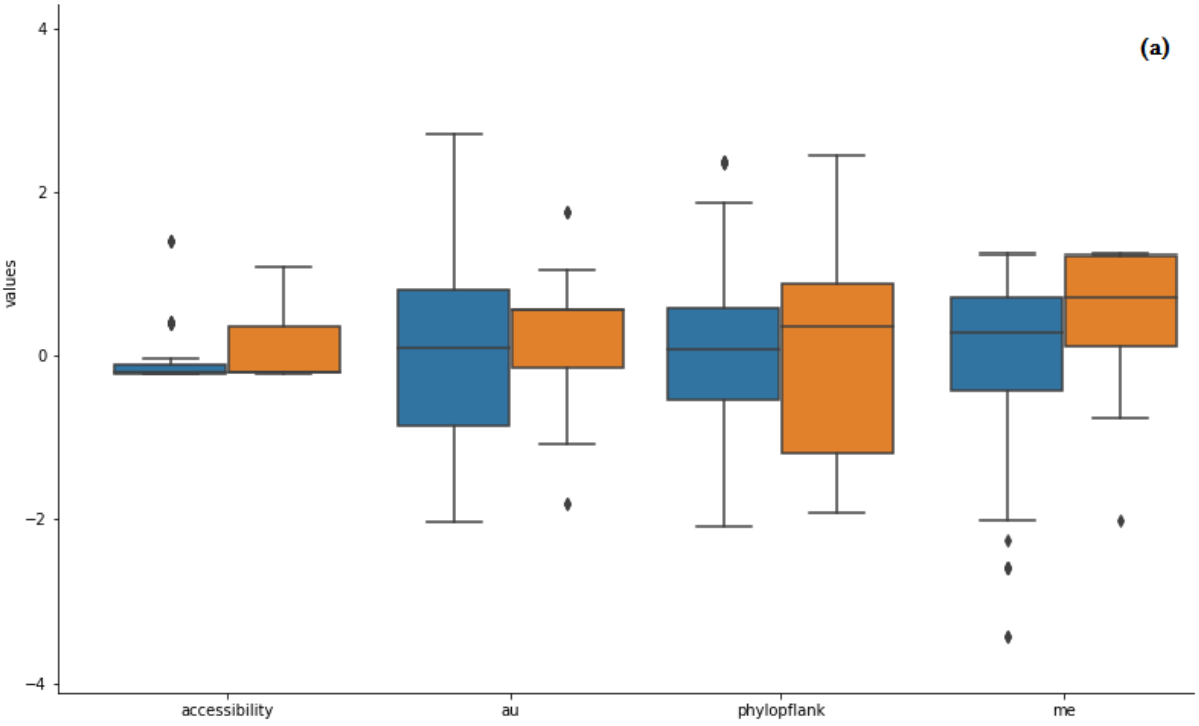
Hyperparameter	**Isolation Forest**	**One Class SVM**
Number of estimators	20	Not applicable
Number of attributes	70 %	Not applicable
Number of samples	30	Not applicable
Bootstrap	True	Not applicable
Contamination factor	True	Not applicable
Kernel	Not applicable	RBF
$\boldsymbol{\nu}$	Not applicable	0.171630
$\boldsymbol{\gamma}$	Not applicable	0.10

Finally, we separated our dataset into two parts: a training set of 123 miRNA-gene interactions and a testing set of 37 interactions, considering that we would apply the whole dataset to the one-class SVM model. However, we wanted to test our data separately to see how it performed with a subset containing only strongly support interactions (training set), and then fit this model to weakly supported interactions (test set). For this purposes, first, we applied the one-class SVM to the dataset that contained only the training set to learn to detect anomalous data therein, and then subsequently trained our model with a subset of the normal (strong interactions) dataset, approximately 70% of the samples of all the data, and then fitted this model to the test set. This procedure allowed us to obtain a confusion matrix for validation purposes and to calculate the accuracy of our model.

4.4 Results

4.4.1 Exploratory analysis

Before applying our one-class SVM model, we decided to visualize the selected features available from mirWalk to detect the presence of outliers in our data via a boxplot diagram (see Figure 4.2).



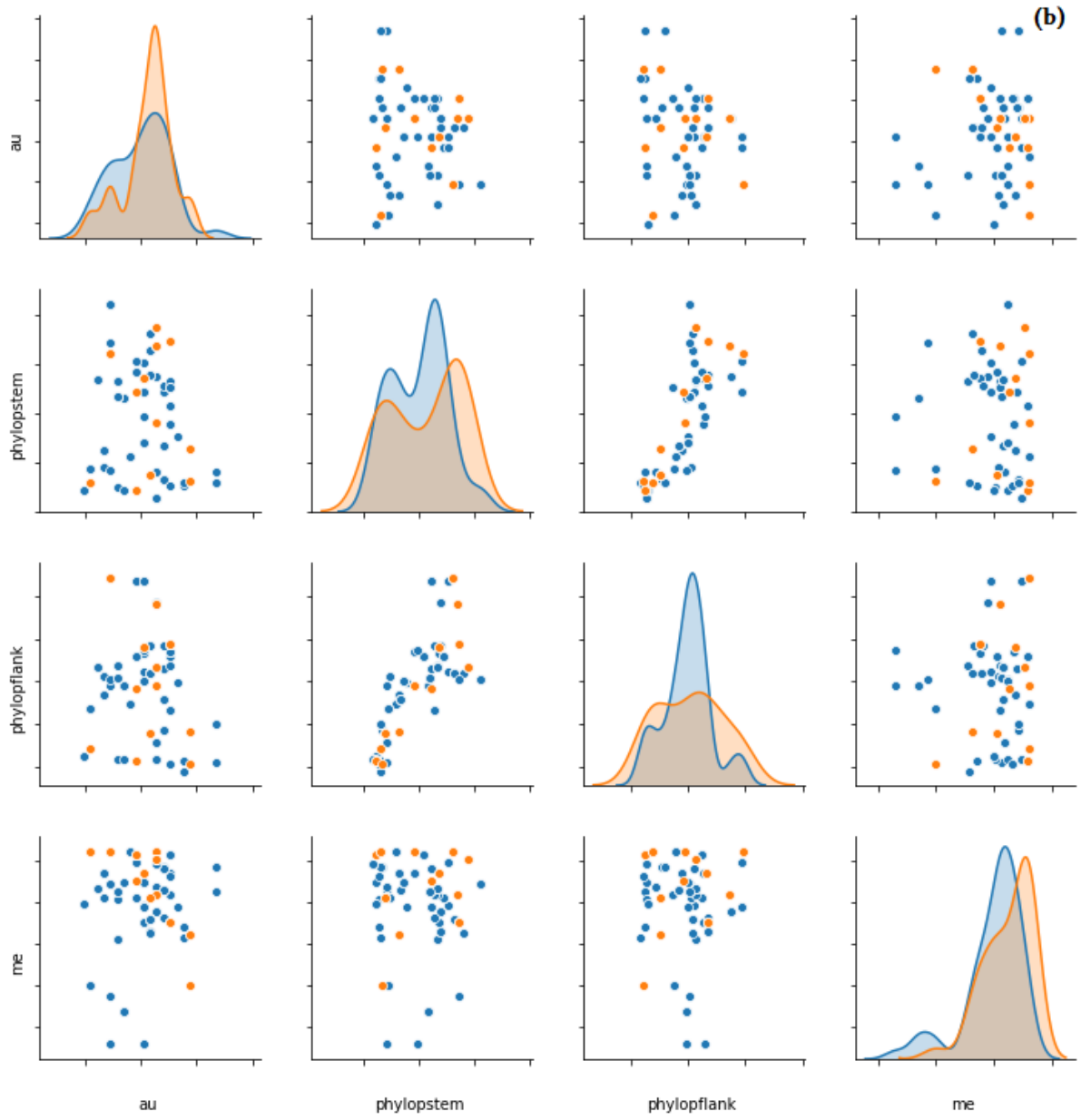


Figure 4.2 Boxplot (a) and correlation plot (b) of a subset of the features selected from the miRNA and ERBB2 gene interactions.

Figure 4.2(a) shows the presence of outliers in a subset of the features of strong and weak evidence support. These outliers can be found in the samples that present weak support, for example, accessibility, au, phylopflank, and me. We also found outliers in those samples that present strong support in miRNA and gene interaction, and we observed them in the au, phylopstem, and me, to cite a few. These results are depicted in the correlation plot in Figure 4.2 (b). The descriptions of these features are presented in Table 1.

4.4.2 Comparison of isolation forest vs one-class SVM

We applied our one-class model to the dataset. First, we tested the isolation forest with the hyperparameters selected and described in Table 4.5, and then we proceeded to test the one-class SVM and performed a comparison between both models by considering the confusion matrices obtained from both models (see Tables 4.6 and 4.7). Additionally, we performed a comparison test using metrics such as accuracy, sensitivity, specificity, and F1-score.

Table 4.6 Confusion matrix of the Isolation Forest modified version.

	True Positive	True Negative
Predicted Positive	87	18
Predicted Negative	36	19

Table 4.7 Confusion matrix of the One Class SVM modified version.

	True Positive	True Negative
Predicted Positive	99	5
Predicted Negative	24	32

For the interpretation of the confusion matrix in Tables 4.6 and 4.7, we determined that the true positives are those miRNAs that we found belong to the class of miRNAs that interact precisely with the ERBB2 gene, remembering that we are dealing with an unsupervised classification. In contrast, the True Negatives reflect those miRNAs with no strong evidence of their miRNA and ERBB2 interactions. The metrics used for assessing both models included accuracy, a sensitivity (true positive rate or recall) of 80.49%, specificity (true negative rate), and F1-score (see Table 4.8). We believe that using accuracy as the primary parameter for comparing these models is misleading, as the percentage of true positives and false negatives should be considered in medical systems for obvious reasons. Furthermore, in this research, the F1-score, which is the harmonic mean between precision and recall (NCI, 2020), is a metric that is appropriate in cases when we can have the presence of an imbalanced dataset. By comparing our two one-class models, we found that the one-class SVM ruled the Isolation Forest in terms of sensitivity and specificity, which are essential

measures for considering whether a person could have a medical condition and, in this case, a breast cancer.

Table 4.8 Isolation Forest with One Class SVM metrics.

Model	Accuracy	Sensitivity	Specificity	F1-Score
Isolation Forest	66.251%	70.730%	51.352%	76.320%
One-class SVM	81.882%	80.490%	86.491%	87.220%

4.5 Discussion

In this part of our research, we used a one-class SVM to find miRNA and ERBB2 gene interactions when there is the presence of only a sole set of data to extract these relationships, and when it is unfeasible to find another subset that could serve as a second class as input for supervised classifiers. Our method obtained a subset of genes that, by being treated as outliers, allowed us to determine which miRNA interactions have an insignificant relationship with an oncogene, for example, in breast cancer scenarios. We were unable to uncover literature evidence of applying one-class classifiers to examine miRNA and gene interactions using features of the miRNA-gene sequences for breast cancer scenarios until the time of authoring the current research. Nevertheless, there exist some proposals, as in the research work of Tran et al. (2008) and Yousef et al. (2008), where a one-class classifier was used to predict miRNA hairpins or miRNA prediction using sequence characteristics. An interesting aspect of these studies is that they used sequence features, which is an approach that we have also considered in this part of our research, in contrast to gene expression data.

The metrics used for validating our results were precision, recall, and F1-measure with a previous hyperparameter selection of the kernel to be used and the ν parameter (Tran et al., 2008). While the one-class models are designed to detect anomalies in unbalanced datasets, we discovered that the authors also transformed a portion of the training data to test data when applying this model. Finally, this technique is useful for applying metrics such as precision or the F1-score (Tran et al., 2008). A comparable approach was used in the present proposal, which was to use as testing data those samples extracted from the majority of

miRNA interactions and labeling some samples, which had weak verified breast cancer correlations considered the literature. These samples were tagged as negative ones.

Regarding the use of one-class SVMs, some authors such as Yousef et al. (2008; 2010) pointed out the importance of these methods when we are in the presence of data where a) we can have only access to the set of positive data and b) the generation of negative samples has the potential to produce skewed results, and there is currently no straightforward mechanism for obtaining this data. Yousef et al. (2008) found that the one-class models had higher sensitivity and lower specificity than their two-class models. When the authors attempted to identify miRNAs in the Epstein-Barr virus, the sensitivity metrics for the various one-class models were roughly 82 %, with no information regarding the specificity metric. However, our one-class SVM model, presented in the current research, obtained a sensitivity of 80.49% and specificity of 86.49%, giving more stable results.

We demonstrated that it is possible to achieve relatively good accuracy and F1-scores by obtaining values of 81.88% and 87.22% in our current research. These findings may pave the way for further research into miRNA-ERBB or other oncogene interactions in breast cancer. We discovered around 19.51% of the outliers after applying the one-class SVM to the training data. Nevertheless, despite the strong evidence, we decided to investigate the current literature. For example, we observed that hsa-miR-25-3p interacts with the ERBB2 gene in nine distinct ways, although only one has been identified as a false negative. At this point, it is valuable to consider that a miRNA can attach to distinct sections of the mRNA or have varied values for features like free energy, flanking conservation, or stem-loop, which may have influenced their final categorization. The establishment of a voting system, similar to that utilized in a KNN model, might aid in classifying a miRNA as an outlier. A similar issue arose with hsa-miR-125a-5p, where one of the ten interactions in our sample was similarly flagged as a false negative.

Concerning the performance of isolation forest against one-class SVMs, one could claim that the isolation forest low performance is due to how tree-based models classifies its data. As it is known, random forest classifiers tend to partition space into rectangular sectors, whereas SVM models can employ several types of kernels to create smooth separating areas. The situation in which an SVM model performs better than RF in genomic data has been mentioned in Statnikov and Aliferis (2007), The researchers compared SVM and RF with 18 diagnostic and prognostic datasets in this study. In these comparisons, SVM outperformed RF

in 13 datasets showing statistically significant differences in performance in 7 of them using a permutation test. A permutation test is a statistical model to show that the results of a classifier are not the results of mere chance and consider a null distribution in which the premise is stated as if the features and the labels in a classification system are independent (Ojala and Garriga, 2009). The authors manifest that SVM is efficient in cases where there are many variables or features, and it can be suitable for complex classification functions. It is worthy of mentioning that this assumption does not necessarily imply that other methods, such as RF, should not be used in genomic data. We hypothesize that this could explain why models based on SVM, like One-class SVM, which also uses separating hyperplanes and kernel function, could outperform other models based on decision trees such as Isolation Forest.

As a concluding remark, the significance of this discovery is to stimulate more research and the use of unsupervised learning techniques in conjunction with datasets such as mirWalk (2020) to discover novel miRNA and mRNA interactions, in contrast to the use of supervised techniques that require labeled data that is not feasible when the data belong to only one class, making the distinction a hard one.

4.6 Summary

We have demonstrated that using a single classifier model to validate miRNAs and gene interactions is feasible based on the genetic sequence features between these binding molecules. This unsupervised technique is employed when we have scarce data, and it is impossible to find labeled data. In addition, the use of a one-class model requires a special type of treatment in the tuning of hyperparameters and modified metrics to test the accuracy of the results. We used the extracted features obtained from miRNA and mRNA sequence binding to validate the interactions between miRNAs and the regulation of the ERBB2 mRNA gene present in cancer scenarios. The results obtained are reasonably comparable to those of other studies that use a subset of the sequence feature binding processes (Irigoien et al., 2014; Rehman et al., 2019); our results showed 82.49% sensitivity and 86.49% specificity. Regarding future research, it would be interesting to face efforts in using unsupervised techniques for finding novel relationships between miRNAs and genes, because in this case, one can work only with one unique class and that there is no need for class differentiation by labeling as supervised models.

CHAPTER 5
Differentiation of Breast Cancer and Breast Neoplasm scenarios based on Machine Learning and nucleotide sequence features from lncRNAs-miRNAs-diseases associations

5.1 Introduction

Non-coding RNAs, including as lncRNAs and miRNAs, have an inevitable role in various disorders, including the genesis of neoplasms and cancer. However, the scarcity of validated datasets and their imbalances make their direct study difficult. Furthermore, few studies have combined machine-learning algorithms with genomic sequence information acquired from miRNAs and long noncoding RNAs, compared to other approaches such as deep-learning techniques paired with genomic expression as features.

Some authors, such as Fu et al. (2017), have described the application of deep-learning algorithms to miRNA and illness connections. They based their research on the assumption that a group of miRNAs with similar functions would be linked to similar diseases. The researchers used disease semantic similarities and miRNA profile kernel similarities. In the end, these features were integrated as inputs for a stacked autoencoder model. Guo et al. (2019) employed an analogous method to predict lncRNAs in glioma or colorectal cancer scenarios, employing semantic similarity and kernel profile techniques with clinical data and autoencoders combined with a random forest. Other methods were suggested by Huang, Y. et al. (2019), who employed raw data without feature manipulation, as well as a network topology containing interactions between miRNA and lncRNA with a graph convolution autoencoder.

Wen et al. (2019) employed a convolutional neural network (CNN) model and as a sequence feature they chose a k-mer frequency analysis. They employed the aforementioned methodologies to identify lncRNA-mRNA interactions in a variety of taxa, including humans, mice, and chickens. The authors demonstrated that using values of k-mers greater than three had no effect on the models. We hypothesize that in these cases, a lengthier string

of k-mers is unlikely to be discovered in a genomic sequence due to their uniqueness, lowering their frequency of appearance.

Apart from studies that independently validated these connections, we found no current studies in the existing literature that examined the prediction of lncRNA-miRNA and their association with illnesses. Additionally, the analysis of lncRNAs and their association with diseases is a topic that may shed light on how these non-coding RNAs (ncRNAs) impact certain disease scenarios. This understanding may aid in the development of novel genetic treatments geared toward the personalized treatment of particular illnesses by utilizing these miRNAs and disease associations (Wen et al., 2019).

The purpose of this study was to associate ncRNA molecules, lncRNAs, and miRNAs with two closely related diseases: breast cancer and breast neoplasms. Based on the understanding that, while breast tissue can be affected by both diseases, the presence of a neoplasm does not always indicate the development of breast cancer. We combined supervised and unsupervised machine learning approaches with feature extraction from non-coding RNA sequences collected from public repositories to develop our models.

5.2 Materials and methods

5.1.1 Datasets

For our experiments in this section, we decided to work with two diseases: breast neoplasms and breast cancer. In both cases, we will consider the lncRNAs and the miRNAs that interact with each other and that have a relationship with this disease. The set of steps that we followed in our experiments are given below.

Step 1. We obtained a list of lncRNAs related to breast cancer and breast neoplasm cases available at the lncRNASNP2 site (Miao et al., 2018); the file name was lncRNA_associated_disease_experiment.txt. The data found in this link contain information on lncRNA relationships with diseases backed up by experiments and publications. These data contain the diseases, PubMed ID, and the corresponding lncRNA-related genes. An example is provided below (Gutiérrez-Cárdenas and Wang, 2021b).

Disease	Pubmed	lncRNA
Atherosclerosis	23861667	NONHSAT130416.2

Glioma	24833086	NONHSAT090275.2
Lung adenocarcinoma	24721325	NONHSAT015484.2

The lncRNA identification number, NONHSAT015484.2, for example, is based on the NonCode database's nomenclature (Zhao et al., 2016). From this list of diseases, we selected those lncRNA IDs related to Breast Cancer scenarios. The information from this dataset would serve as an initial input to be considered for the training dataset.

We gathered data on predicted lncRNAs associated with this disease to obtain a test dataset concerning breast neoplasm data. We chose this condition because, according to the reviewed literature, not all breast neoplasms progress to breast cancer. The list obtained was from the lncRNASNP2 database (Miao et al., 2018), referred to as lncRNA-associated diseases predicted by TAM (Lu et al., 2010). TAM is a software program that employs miRNA categories, including family categorization, disease association, or functionality, to annotate miRNAs and disease relationships. TAM uses a hypergeometric test to select which miRNAs are overexpressed or under-expressed (Rivas et al., 2007).

Step 2. The preceding stage gathered data on a set of records containing validated lncRNAs that have a relationship with breast cancer and those related to Breast Neoplasm predicted by experimental methods. Furthermore, we compiled a collection of miRNAs associated with these disorders. The lncRNASNP2 dataset (Miao et al., 2018) contains a list of miRNAs and their illness associations. The authors managed to predict interactions of each lncRNA with miRNAs in this dataset by confirming the data from miRBase and verifying their findings using TargetScan, miRanda, and Pita. Lastly, they employed enrichment analysis (Miao et al., 2018), a technique that Zu et al. (2013) evaluated.

To provide more details about the miRNAs gathered, we obtained a couple of miRNA datasets from the lncRNASNP2 (Miao et al., 2018) sites. The miRNAs collected were considered if their association with the lncRNAs was conserved (file name mirnas_lncrnas_conserved) or if the interaction was predicted (file name mirnas_lncrnas_validated). Notably, the differences in the quantity of conserved data, defined as interactions between miRNAs and lncRNAs that have not changed over time or are supported by experimental methods, were negligible in comparison to the latter. Diverse studies like the ones of Rehman et al. (2019) and Zhan et al. (2020), have emphasized the

absence of empirically verified miRNA-lncRNA connections or miRNA-disease relationships.

To recap, we chose as the training dataset the one that contained only breast cancer registers, the file that contained the lncRNAs alias related with breast cancer scenarios, from step 1. Then, we added up the miRNAs associated with lncRNAs, but chose those that their relationship was conserved (Step 2). We followed a similar procedure with the breast neoplasm data. In this scenario, we created a list of validated miRNAs and lncRNAs and mapped them to lncRNASNP2 (Miao et al., 2018). Finally, we created a class comprised of 454 breast cancer registers and a second class comprised of 9525 breast neoplasm cases. As previously stated, there were fewer validated breast cancer scenarios than projected breast neoplasm data.

Step 3. We retrieved the lncRNA and miRNA genomic sequences from the previous steps' data in this stage. Afterward, we extracted features from this genetic data and used them as input for our machine-learning algorithms. In FASTA format, the miRNA sequences were obtained from miRBase's mature miRNA sequences (Kozomara and Griffiths-Jones, 2014). To collect the lncRNA genomic sequences, we web-scraped the NonCode website (Zhao et al., 2016). We used the BeautifulSoup library and the BioPyhon package to manipulate the sequence for feature extraction. Additional details about the selected features are covered in the next section. Finally, in order to work with our machine-learning models, we partitioned our data into training and testing sets.

5.1.2 Methodology and experiments

Figure 5.1 illustrates the schema of the steps of the experimental procedure performed after obtaining the data for breast cancer and breast neoplasms. We considered the IDs, lncRNA sequences, and miRNA sequences in each dataset. The following features were extracted: From the lncRNA and miRNA sequences, we obtained the frequency of 2-mers present in these associations; the energy of miRNA folding secondary structure was also used, and the energy obtained from the co-folding of the miRNA with the lncRNA section from the best sequence alignment between both sequences. Additionally, we extracted a feature consisting of 5-mer fragments of the miRNA sequence with a one-nucleotide sliding window. Afterwards, we joined each 5-mer subsection and joined them to form a strand aligned with the lncRNA. We used the score from this alignment as an additional feature. It was necessary to normalize the values of all features.

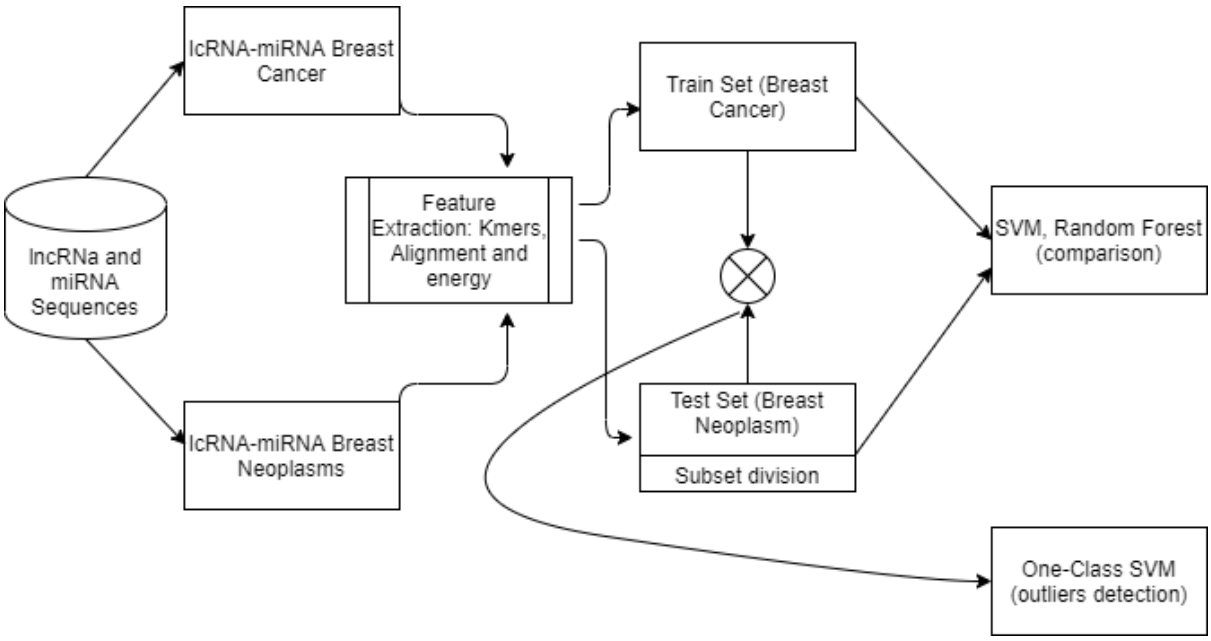


Figure 5.1 Architecture of the proposed model.

We collected 453 records for the training set, which contained data on breast cancer, and 413 records for the test set containing breast neoplasms data. Notably, the test data was obtained from the complete dataset of the breast neoplasm class using a random sample without replacement and roughly corresponded to 4.5 percent of a total of nearly 9300 records. We used the training and testing sets for our supervised models, SVM, and random forest, and validated our results via five-fold cross-validation for the SVM model. We devised a similar technique for the random forest model, but without using cross-validation. Cross-validation was not applied because the random forest model uses bootstrapping and selects a subset of features and data at random. Therefore, we decided to iterate ten times over the entire test dataset for the random forest case, and in each repetition roughly 4.5% was selected to test our model. Finally, we averaged the obtained results for this model; and compared their respective metrics after testing both models to determine which supervised model was the best.

The data amount between the training set (breast cancer) and the testing set (breast neoplasms) was highly uneven. For this purpose, we opted for a one-class SVM, a method that is prone to work with only one class. In this model, we combined data from breast cancer and breast neoplasms. Therefore, we considered the breast cancer data as the negative class since their quantity was limited compared with the breast neoplasm records. The weighted scoring function schema proposed by Pang et al. (2005, p. 292) was used for the selection of hyperparameters and the testing of this model. It is worth mentioning that we tuned the

hyperparameters of these models by using GridSearch Cross-Validation and OOB (out of the bag error) for the number of estimators in the random forest model.

5.1.1.1 Features extracted

In this study, we worked with those features that are obtained from the binding among miRNAs and lncRNAs molecules; this contrasts with techniques that use gene expression technique data and, because of that, will require in-vitro experiments. From these data, we obtained a new set of attributes based on the k-mers data that occurs in the process of binding between these sequences molecules along with the energy released during this process. Consequently, these features could be generated in-silico so that there would not be a need for gene expression data and in-vitro experiments, as mentioned before. For our machine-learning models, we extracted features based on two large subsets. One is related to the energy released when binding occurs, and the other is related to the frequency of k-mers between miRNAs and lncRNAs. For the energy features, we used the Vienna package (Hofacker, 2003), which included the RNAfold and RNAcofold functions The MFE that arises in forming the secondary structure of RNA is returned by the RNAfold function. We also evaluated the amount of energy released when dimerization happens between a pair of RNA sequences, in this case, the lncRNA and the miRNA sequences. To measure the energy that is produced in this dimerization process, we used the Vienna package's RNAcofold function. Readers interested in learning more about these functionalities are encouraged to visit the Vienna webpage at https://www.tbi.univie.ac.at/RNA/tutorial/.

We were unable to apply the function RNAcofold directly to the lncRNAs due to the differences in size between lncRNAs and miRNAs. The reason is that the length of the miRNA is approximately a third of the length of the lncRNA. As a result, we used a sliding window of 3-mers to extract lncRNA subsequences. The Needleman-Wunsch algorithm was used to align these chunks with the miRNA sequence, and the lncRNA portion with the best alignment score was chosen to compute the dimer energy fold; for this task, we used the RNAcofold tool. To manipulate the miRNA sequence to match the alignments in the lncRNASNP2 database (Miao et al., 2018), we used the BioPython package. This manipulation consisted of flipping the miRNA sequence to get the lncRNA and miRNA co-folding sequence.

Considering the features related to the frequency of *k*-mers, we calculated the average frequency of 2-mers for the miRNA and lncRNA sequences. Concerning the last feature, we

extracted the miRNA 5-mer sequences and used the Needleman-Wunsch algorithm to align them with the lncRNA. All quantitative results from the features were normalized using a standard scaler function. Furthermore, to avoid duplicate records between the breast cancer and breast neoplasm datasets, we coded a script that deleted records that were duplicated many times.

5.1.1.2 Machine Learning models

To validate the presence of outliers in our dataset, we employed a one-class SVM model. To do this, we combined breast cancer records, which will serve as our outlier subgroup, with Breast Neoplasm data, which worked as our majority class. The benefit of this one-class model is that it achieves the accuracy of a two-class SVM-based technique while using only one training class. Additionally, we do not require labeled data because we are working with a single class; nonetheless, greater attention should be exercised while tuning hyperparameters or evaluating the output of such models. Therefore, it is recommended that certain samples should be chosen to represent a negative or opposite class to assess the correctness of our model.

We collected roughly 453 records classified as breast cancer and 9170 records categorized as breast neoplasms. As previously stated, we hypothesize that when breast cancer data are combined with breast neoplasm records, the minority class, breast cancer records, might be deemed as a set of outliers. We combined these two datasets, breast cancer and breast neoplasms, into one. Afterward, we separated our data into training and test datasets using the 80/20 golden ratio. The training subset contained around 7698 records, which were categorized according to whether they were breast cancer or breast neoplasms. Due to the fact that a one-class SVM needs only one class to work with, we employed these 7698 records for this model. Additionally, we determined that 371 records (train outliers) corresponded to breast cancer samples and 7327 (normal train samples) belonged to breast neoplasm samples after using the previously indicated labeling. We used the one-class SVM model for the first time to investigate whether the model could discriminate between cancer and neoplasms.

In terms of hyperparameter adjusting, we acquired 371 records for the breast cancer samples, yielding an outlier proportion of about 5%, which will be used as the v parameter in our one-class SVM. GridSearch was used to adjust the hyperparameters for each evaluated kernel (linear, polynomial, and RBF) with cross-validation of five-folds and an average-weighted average for the scoring function. When datasets are unbalanced, this type of scoring function

is used. In a second attempt to validate the accuracy of our model, we used our one-class model once more. However, in this scenario, we employed the entire set of breast neoplasm subsets as the training set and the breast cancer subset as the testing set, concluding by verifying the model's accuracy.

Although one-class SVM models are more suitable to work with a single dataset, it is prudent to test it with a subset of the entire set. This subset of testing data could be used for accuracy, sensitivity, and specificity metrics. To accomplish this, we decided to use the majority class, which contains 9170 records from the Breast Neoplasm dataset, as the training set, and the testing class, which includes 453 records from breast cancer samples. Breast cancer data would be deemed outliers due to their rarity compared to the majority of records.

After identifying the occurrence of outliers using the one-class SVM, we tuned the supervised machine-learning models' hyperparameters. We used fivefold cross-validation and a weighted scoring function for the SVM model. Further, we used 4.5 percent of neoplasm class samples without replacement as test data, and this percentage did not significantly show an imbalance from the number of breast cancer class samples. We validated our findings through repeated tests using different subsets. Considering the random forest model, we applied a similar process, but because this model relies on bootstrapping, we skipped cross-validation. Rather than that, we executed the model within a loop that iterated ten times and chose the same sample percentage as previously stated. Therefore, for each test of the supervised models, the number of repetitions was increased. We repeated this procedure ten times in the first run and then averaged the results for the supervised model under test, repeating the process with 20, 30, or 40 iterations. We noticed that each iteration produced similar results when we compared the results obtained.

5.3 Results

5.3.1 Descriptive statistics results

Because various features obtained in our dataset, it was impossible to visualize the results accurately, so we decided to apply a dimensionality reduction technique. Thus, we applied PCA to visualize the data corresponding to breast neoplasm and breast cancer scenarios considering their respective features. In our analysis, we discovered that two major components explained 25% of the data collected, with values of 0.14828369 0.10792987]. A plot of the data is presented in Figure 5.2.

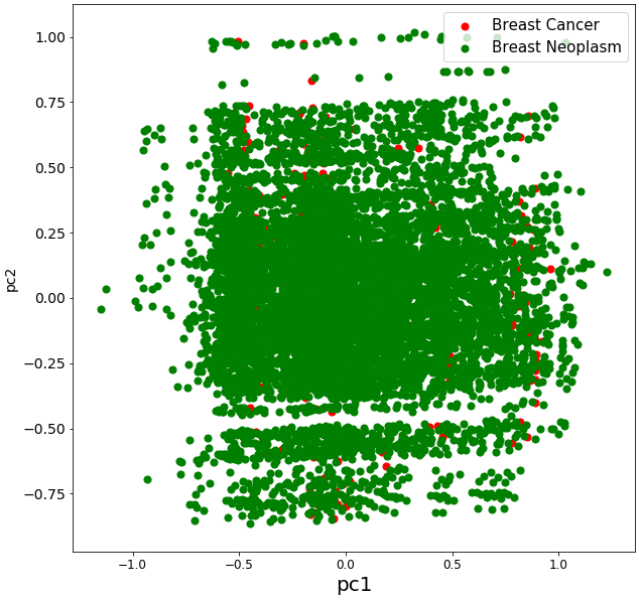


Figure 5.2 Two-component plot of the Breast Cancer and Breast Neoplasm dataset.

We generated a boxplot of the normalized features extracted from both datasets to rule out outliers. By examining the plot in Figure 5.3, we can see that certain dimers (groups of two nucleotides), such as the UCm (m from miRNA) and the AGl (l from lncRNA), presented a high degree of outliers.

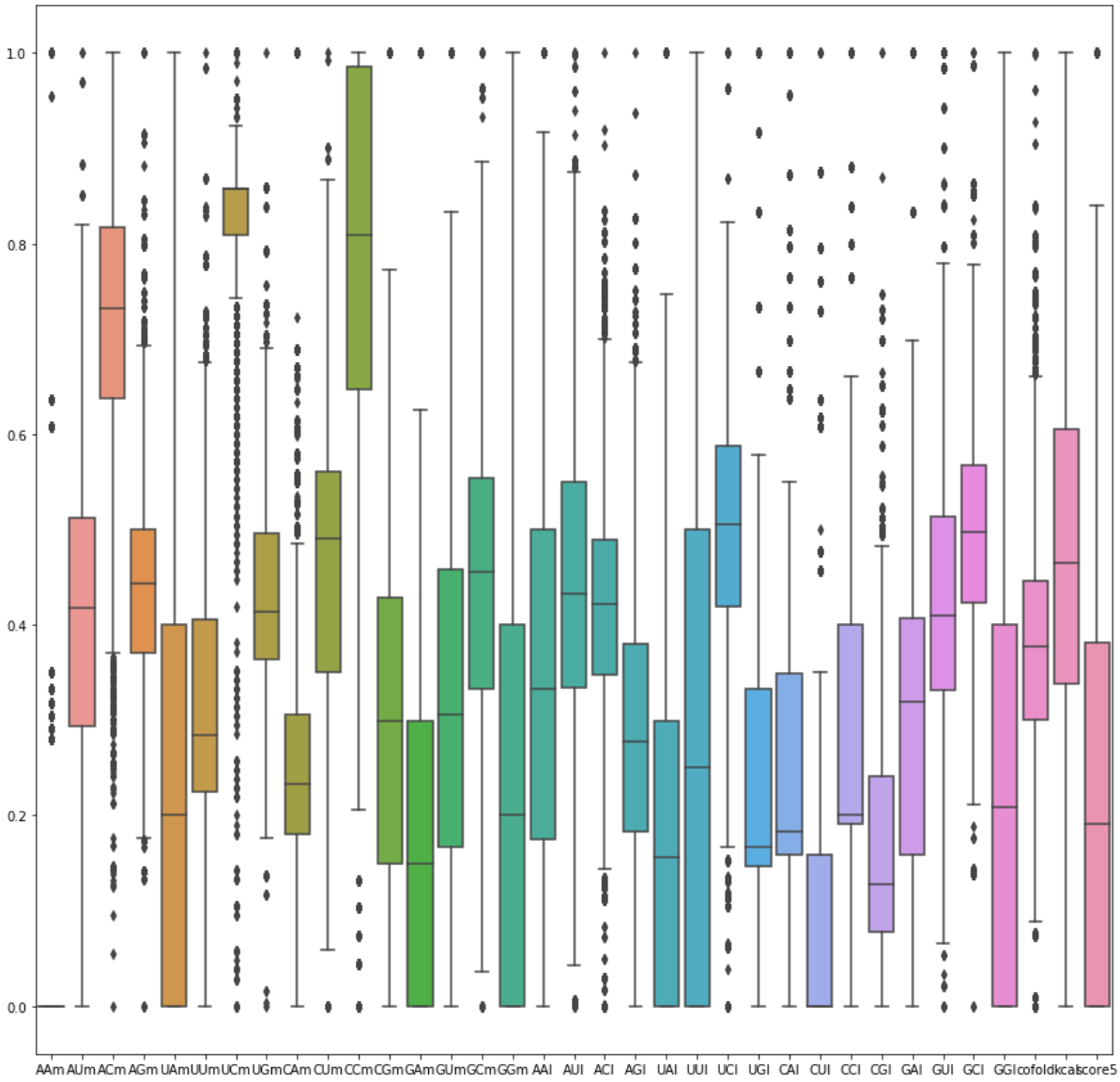


Figure 5.3 Normalized 2-mer frequencies, secondary structure energy, cofold, and 5-mer matching score boxplot.

Additionally, as shown in Figure 5.4, we plotted their respective probability density function (PDF) for each of the normalized features derived from the breast cancer and breast neoplasm datasets. We noticed certain values on this graph that overlapped, such as ACm or GUm, indicating that the densities of these pair of 2-mers were nearly equal. However, we discovered instances where the PDFs from breast cancer samples were overexpressed compared to those from breast neoplasms. We observed this trend in the 2-mers of CUm, CUl, UUm and UGm, , but only in the UCm 2-mer did the breast neoplasm PDF have a high value compared to the breast cancer samples.

Figure 5.4 PDF of the distinct features found from the Breast Cancer and Breast Neoplasm datasets.

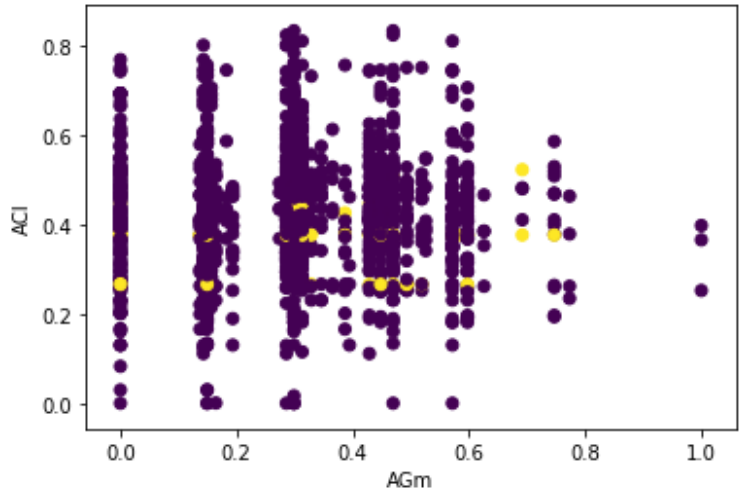
5.3.2 One-class SVM

The outcomes of the hyperparameter tuning of the one-class SVM model were: nu-value of 0.0506 with an RBF kernel and a value of gamma of 1e-05. We trained our prototype with

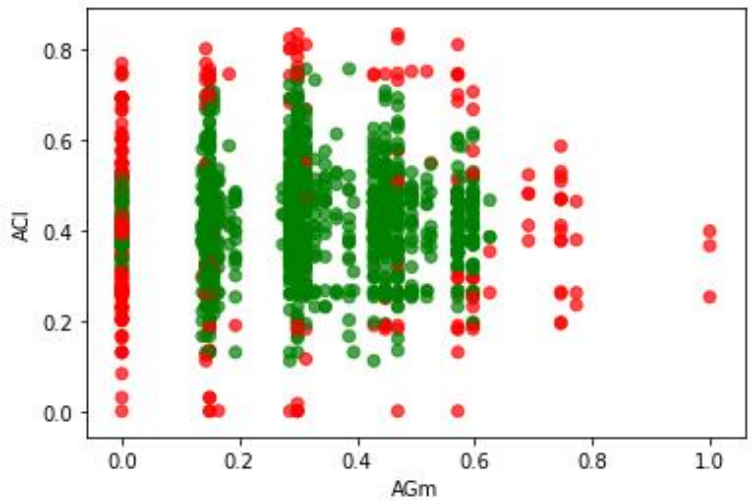
the AGm, ACl, and GUl features and plotted our findings using only the AGm and ACl features for visualization purposes. We chose these features because their pdf plots demonstrated meaningful differentiation, as illustrated in Figure 5.4. Figures 5.5(a) and 5.5(b) depict the plot results for the training and test sets, respectively. It is worth noting that we employed a one-class SVM again for validation procedures, but this time with all 9170 records from the breast neoplasm dataset as the training set and 453 records from the breast cancer dataset as the test set.

After selecting 20% of the data for testing purposes, as mentioned in Section 4.1.2.2, we achieved the following results: for the accuracy metric, we acquired a 95.44 %; for sensitivity, we obtained a 93.19 %; and for specificity, we obtained a value of 97.97 %.

These results suggest that it was feasible to distinguish between breast cancer cases and breast neoplasm scenarios using the features derived from our dataset.



(a)



(b)

Figure 5.5 (a) One-class SVM training and 5.5 (b) testing results.

5.3.3 Supervised models

We implemented two supervised models, SVM and random forest, to determine if the acquired features could be used to predict breast cancer and breast neoplasm scenarios. Following hyperparameter optimization, we obtained the ideal parameters for the SVM, which were RBF as the kernel, a penalization factor of 100, and a Gamma value of 0.01.

The accuracy score was calculated using four-fold cross-validation using a weighted function, and the results are as follows: Training Accuracy was 89.232 % with a standard deviation of 0.0105, while Testing Accuracy was 88.797 % with a standard deviation of 0.0204.

The following hyperparameters were chosen for our Random Forest model: Gini as the splitting criterion, for the maximum depth, a value of eight was selected; the square root of the number of instances for the maximum number of features, and a 25 for the number of trees.

The same criteria were used to evaluate the SVM model. In this case, we ran the dataset ten times and averaged the findings; the result was an accuracy of 99.65 percent with a standard deviation of 0.0038.

As a result, we can conclude that the random forest model with the provided characteristics is the most appropriate model for discriminating between breast cancer and breast neoplasm scenarios.

5.4 Discussion

A search of the relevant literature revealed no studies that indicated the interaction between lncRNA and miRNAs or their association with diseases. Although some publications, such as the ones cited in the Introduction part, treat these associations as separate units.

We used the k-mers information, the secondary structure RNA energy, co-folding RNA energy, and the alignment produced by comparing a 5-mer miRNA sequence to a lncRNA sequence as extracted features. Our machine-learning algorithms exploited these features to distinguish between breast cancer and other neoplasm samples. A one-class SVM was also utilized to check for the presence of novel samples or outliers that corresponded to the class of breast cancer data. Using this model with a subset of joined samples of breast neoplasms and cancer, we got a 95.44 percent accuracy rate. Our SVM model attained an accuracy rate

of 88.79 percent with a standard deviation of 0.020. However, we found that our random forest classifier had the best performance, with an accuracy rate of 99.650 percent and a standard deviation of 0.0038. To be clear, we tested our two-class models using 4.5 percent of distinct samples using a combination of breast cancer and breast neoplasm scenarios, but without replacement. At least ten repetitions of this test were carried out for validation purposes.

Yan et al. (2020) used a CNN to assess sequence data from lncRNA and miRNA interactions, as well as 4-mers with composition transition distribution and graph characteristics. The dataset was partitioned into positive and negative samples based on lncRNA-miRNA relations extracted from the lncRNASNP2 database (Miao et al., 2018). Positive samples chosen a priori revealed an interaction, whereas negative samples did not. Their deep-learning model achieved an accuracy rate of 93.81 %, a sensitivity of 91.58 %, and a specificity of 79.10 %. Our single-class SVM model acquired 95.44 % accuracy, 93.19 % sensitivity, and 97.97 % specificity. Nevertheless, it could be argued that the latter is an unsupervised model; however, we found the metrics mentioned above by utilizingo a mixed subset of the data as a test set for validation purposes.

Other researchers, such as Guo et al. (2019), proposed employing an autoencoder neural network in conjunction with a rotation forest to combine profile kernel similarities and Gaussian profile interactions between lncRNAs and diseases for prediction purposes. Colorectal cancer, glioma, and prostate cancer prediction interactions had an AUC value of 94.74 percent in this study. While further research is needed to compare our predictions with other deep learning or machine learning models, we showed that the results obtained from our models are comparable to those reported in the literature.

Additionally, we believe that, while deep-learning algorithms might extract relevant attributes from biological data, there is still an area of research devoted to the application of machine-learning techniques to biological data via feature engineering.

5.5 Summary

In this chapter, we collected sequence characteristics from lncRNA-miRNA and disease relationships from breast neoplasm and breast cancer scenarios and used a one-class SVM to predict these associations. We also compared the adequacy of the features extracted by applying SVM and Random Forest as supervised models. Furthermore, we demonstrated

that it is possible to distinguish between breast neoplasm and breast cancer classes despite the fact that their information may overlap due to their shared classification of abnormal tissue growth in breast samples. Our models produce results that are equivalent to those published in the literature. The current study demonstrates the feasibility of applying feature selection to non-coding sequences, which might be used to study new relationships of non-coding RNA to various diseases.

CHAPTER 6
Prediction of binding miRNAs involved with immune genes to the SARS-CoV-2 by using sequence features extraction and One-class SVM

6.1 Introduction

miRNAs belong to a group of non-coding RNAs that bind to RNA or specific genes. Usually, they bind to the RNA of different species, and some studies have demonstrated that they can also bind to viral RNA (Lamkiewicz, 2018; Trobaugh, 2017; Nersisyan et al., 2020). Under these conditions, miRNAs can bind to the mRNA of a viral genome and repress their transcription or even disable a virus's reproductive capacity. According to Nersisyan et al. (2020), miRNAs can bind to viral RNA because they cannot differentiate from the host mRNA. This relationship could be involved in the different viral RNA processes and could even regulate the spread of the disease within a person's organism. However, according to Yousefi et al. (2020), there is still no evidence that mRNA from viruses could produce miRNAs, but there is evidence that miRNAs could interfere with the SARS-CoV-2 virus their functions related to replication, translation, or interference with the host expression.

With the SARS-CoV-2 pandemic that we are facing, the study of miRNAs and how groups of miRNAs could interact with this viral disease has earned its place in the research world. For example, they can be used as potential biomarkers for detecting the disease or in genetic treatments (Jafarinejad-Farsangi et al. 2020). We hypothesized that specific miRNAs' affinity to viral RNA could indicate an underlying condition in a patient that could ameliorate or aggravate its prognosis. Therefore, in this part of our current research work, we will use a one-class SVM model to predict the binding of miRNAs to SARS-CoV-2 mRNA; and we will be using a one-class SVM model for these purposes.

One-class SVM was developed by Schölkopf et al. (2001) and has its origins in the theory of hyperplane separation between classes, in the same way that it is applied in a two-class SVM model. We selected this model based on the justification that we have a set of miRNAs that we believe could bind to the SARS-CoV-2 RNA genome, but we have only one class

corresponding to the whole set of miRNA sequences. In the current literature, many studies have reported mixed results regarding miRNAs that could potentially bind to the mRNA of this virus, but some of them are based on predictions, and may require in-vitro validation. Even in certain studies, the list of miRNAs that could potentially bind to SARS-CoV-2 mRNA is extensive (Pierce et al., 2020; Saçar and Adan, 2020), covering many miRNA samples present in datasets such as miRBase (Kozomara, 2014). For that reason, we wanted to analyze if there is a subset of miRNAs that could potentially bind to the SARS-CoV-2 genome but are considered (after applying a one-class model) outliers anomalous bindings.

We focused our research on the sequence and thermodynamic features of the binding between miRNA and viral RNA. However, we found that making a simple analysis of perfect complementarity between miRNAs and RNAs would not be fruitful. This is because animal miRNA binding does not necessarily present perfect Watson-Crick complementarity, as in the case (Schwab et al., 2005). Therefore, we extracted features related to the frequency of *k*-mers present in miRNAs and the MFE obtained from the binding of miRNAs and viral mRNA.

One possible issue for predicting miRNA and RNA viral binding is that it is not straightforward to find two separate or differentiated classes. Therefore, we can be in a situation where one class is extremely short in quantity compared to the other. Additionally, we could not find a study that used one-class models for these viral scenarios, which seems ideal for unbalanced class scenarios or when we have only samples from one class. We hypothesized that by using the whole set of miRNAs as the positive class and extracting some features based on their sequence properties primarily related to the frequency of k-mers, sequence alignment, and MFE from the sequence matching, we can find a subset of specific and interesting miRNA binding, based on the results from our one-class SVM model. Furthermore, it is important to determine whether these types of outliers or anomalous binding of miRNAs to the SARS-CoV-2 mRNA sequence are related to other diseases. They may be involved in some scenarios in which a co-morbidity can occur, such as obesity, lungs, or heart conditions, or appear in related diseases such as influenza or other diseases related to the respiratory system. In this part of our current research, we also consider the hypothesis that if these miRNAs are involved in immune gene regulation, they could also be prone to bind to the SARS-CoV-2 mRNA. For this reason, and using the features generated from miRNAs and the SARS-CoV-2 sequence binding, we will extract those miRNAs with a relationship with genes involved in the immune response from the human body. Furthermore,

we tested two supervised models, SVM and random forest, and compared their accuracy with a one-class model.

6.2 Materials and methods

6.2.1 Methodology

In this study, we aimed to predict the probable binding of miRNAs to the SARS-COV-2 virus and the 5' UTR region. For this purpose, we extracted a list of miRNAs that contain information related to their ID and genomic sequences. The miRNAs from this list were paired with the viral region described. Afterward, we applied a one-class SVM model to check for the presence of novel miRNAs or outliers that could occur in this binding.

When outliers were verified, we obtained a list of genes present in the immunology processes of the human body. Then, we extracted a new list containing miRNAs that are prone to bind to these miRNAs. This obtained list would be our positive class, while those not related form our negative class. We hypothesize that when there is a viral infection, miRNAs are prone to bind to these immune genes, and therefore, it would be relevant to predict whether there is an affinity for binding to the SARS-CoV-2 gene. We will use an SVM and RF as supervised models and a one-class model to verify our hypothesis. The schema of the proposed methodology is shown in Figure 6.1.

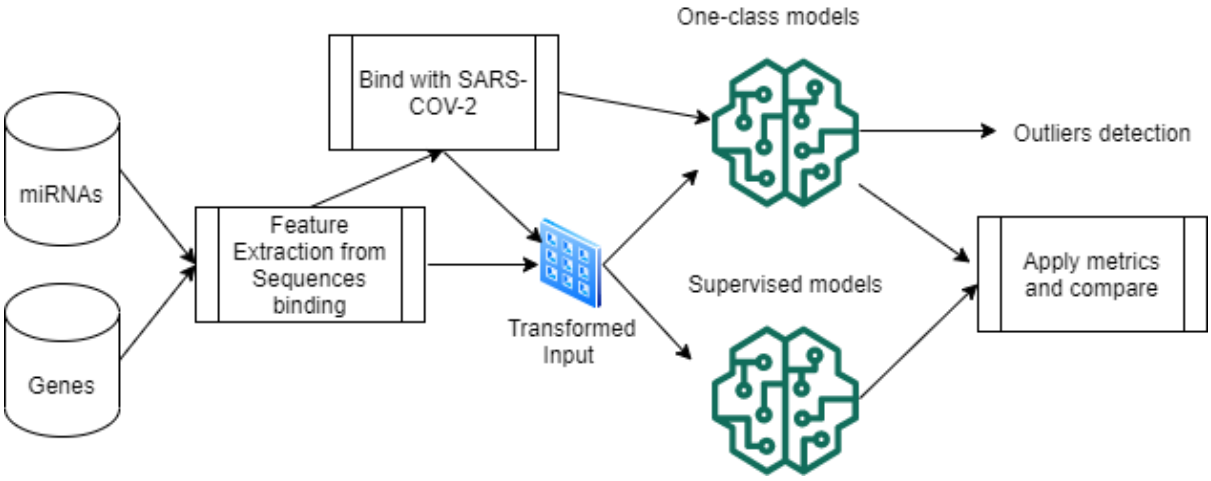


Figure 6.1 Schemata of the methodology followed.

6.2.2 Datasets

a) SARS-COV-2 Genome

Concerning the SARS-CoV-2 Genome, we have downloaded its FASTA sequence from NCBI with accession number GenBank: MN908947.3. In Fig. 2, we can observe the parts of this viral sequence, but for our current research, we will work only with the 5'UTR. According to Mukhopadhyay and Mussa (2020), this untranslated region contains a high number of conserved regions of approximately 90 nucleotides. The FASTA sequence for the 5'UTR region consists of nucleotides 1 to 265. A figure showing the different sections of the SARS-COV-2 genome is depicted in Figure 2.

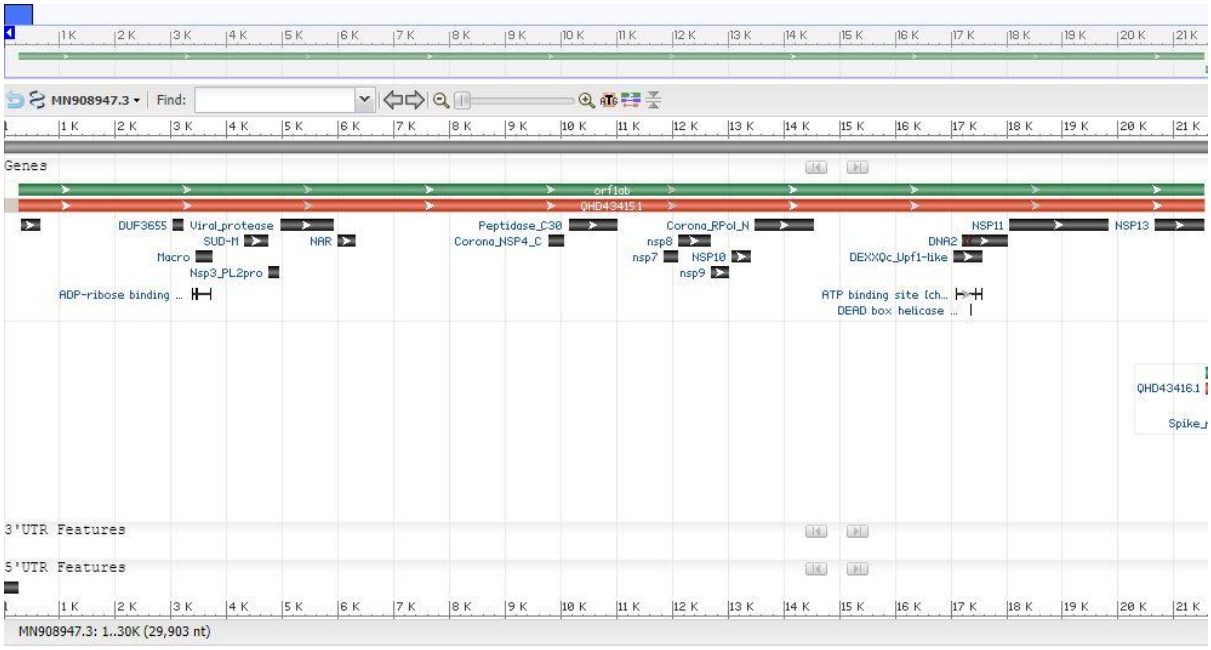


Figure 6.2 Section of the complete genome from the Coronavirus 2 isolate Wuhan-Hu-1 (Severe acute respiratory syndrome coronavirus 2 isolate Wuhan-Hu-1, complete genome, 2020)

b) miRNAs

We downloaded a list of miRNAs from miRBase (Kozomara, 2014). The data contain information about different types of miRNAs from diverse species, but we have only focused on human miRNAs. We will clean up the data by deleting those entries in which we do not find a genomic sequence from this list of downloaded miRNAs. Regarding the software used, we have worked with BioPython to read the FASTA files and extract the information related to the miRNA such as id, mirTarBase id, Species, Target Gene, type of evidence support, and the miRNA sequence. We will use only the miRNA id for reference and labeling purposes, along with the nucleotide sequence of each miRNA from all the mentioned attributes.

c) Immunology genes

We extracted a list of genes present in our organism's immune process by retrieving the InnateDB (Breuer et al., 2013). This dataset contains information on 4815 genes involved in immune processes.

6.2.3 Features extracted

Once we obtained both datasets, we extracted the features that will work as an input for our one-class SVM model. From the set of miRNA sequences, we extracted the frequency of 3-mers from their genomic sequences. We decided to choose this grouping because a 3-mer results in the formation of a protein codon. This method of extracting features from *k*-mer information was also performed by Zhang et al. (2020) with plants, and we tested our model with 2-mers and 3-mers; given the latter one a list of more refined miRNAs. This number of 3-mers was also selected as the upper threshold selected for this feature. Furthermore, other authors, such as Wen et al. (2019), demonstrated that upper values such as 4-mers or 5-mers gave results with negligible differences between both types of k-mers.

Using the genomic sequence of the 5 'UTR of the SARS-COV-2 virus, we obtained the energy generated from a matching between the miRNA sequence and this genomic region. For this purpose, we used the Vienna package (Hofacker 2003) with their RNAduplex function, which serves to calculate the hybridization of two sequences, and it is also used to obtain potential binding between mRNA and RNA. The BioPython package was used for sequence analysis and manipulation. Additionally, we performed pairwise sequence alignment between the 5'UTR and the miRNA sequence. To achieve this procedure, we first transcribed the miRNA sequence and then complemented it, because we wanted to obtain a score based on the matching between nucleotides and not a Watson-Crick base pairing. Considering the immunological genes extracted from the Immport database, we did not perform any additional manipulations.

6.2.4 One-class SVM for detection of outliers

We decided to use a one-class SVM to find anomalies or outliers in miRNA and viral RNA sequence binding with the idea that these anomalies could represent interesting interactions between these two genomic molecules. The hyperparameter tuning was validated experimentally by setting different values for ν and gamma. By tuning our hyperparameters, we chose a value of ν =0.05 and a value of gamma=1e-05. For this part of our research, we used a total of 2548 miRNAs.

6.2.5 Application of Supervised models

After determining the presence of outliers in our miRNA-viral binding dataset, we predicted whether there could be an interaction between those miRNAs with an affinity to bind to genes involved in immunological processes; from here, we extracted two classes from our whole dataset. One class would be those miRNAs that bind to these immune genes, positive class, and those that do not have a validated interaction with these genes will form our negative class.

Because SVM is a supervised model that needs two classes, we divided our 2548 records of miRNAs into a group of miRNAs that interact with immune genes. This subset forms a positive class. In contrast, those that did not present this interaction were labeled as our negative class. We applied hyperparameter tuning using GridSearch with cross-validation of ten folds and a weighted scoring schema.

We applied a random forest model with the same subsets of positive and negative classes to compare our SVM results. Before testing our model, we applied GridSearch with a weighted score to obtain the best hyperparameters of our model in the same way as the SVM model. With the data found, we tested our model's accuracy level and verified our results.

6.2.6 One-class SVM comparison with supervised models

For this model's application, we needed to select the best hyperparameters to be used with our one-class model. For this purpose, we used a five-fold grid search CV. After these parameters are found, we tested the model, but for validation purposes, we used a fraction of the negative samples, composed of those miRNAs that bind to the mRNA virus; however, they do not have a relationship with genes involved in immune processes. To validate our results, we ran our model ten times, similar to a cross-validation procedure, and chose different random samples from the negative class. Then, we applied a set of metrics to measure our results. This model was compared with the supervised models.

6.3 Results

6.3.1 One-class SVM Results for outlier detection

When we applied our one-class model for outlier detection, we found that from the 2548 miRNAs, we obtained with 88 miRNAs that could be classified as anomalies or outliers in

our dataset. As shown in Figure 6.3, the outliers, negative miRNAs, detected by our one-class model.

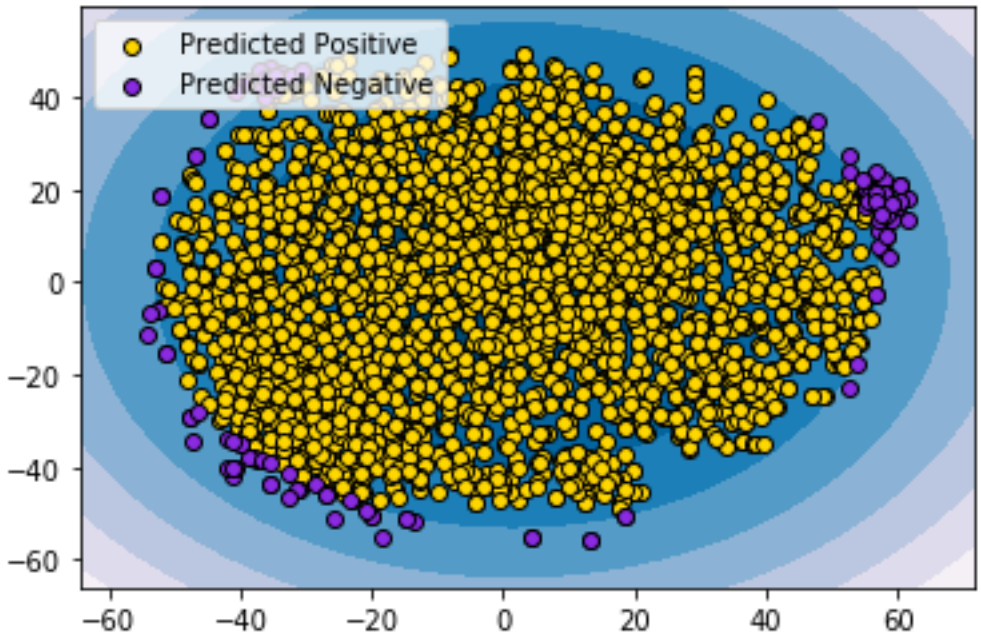


Figure 6.3 miRNAs that are outliers obtained from the One-Class SVM.

We validated the miRNA obtained from step 2.3, and the existing literature validated these results; these outcomes can be observed in Table 6.1.

Table 6.1 miRNAs predicted by the One-Class SVM and their supported literature references.

miRNA	**Reference**
hsa-miR-1182, hsa-miR-1248, hsa-miR-1253, hsa-miR-1261, hsa-miR-1278, hsa-miR-1282, hsa-miR-1323, hsa-miR-136-5p, hsa-miR-1908-5p, hsa-miR-2054, hsa-miR-298, hsa-miR-302f, hsa-miR-3182, hsa-miR-340-3p, hsa-miR-4267, hsa-miR-4291, hsa-miR-4311, hsa-miR-4435, hsa-miR-4487, hsa-miR-4493, hsa-miR-6126	(Vastrad et al., 2020)
hsa-miR-582-5p	(Vastrad et al., 2020; Ahmadi and Moradi; 2020)
hsa-miR-98-5p	[Vastrad et al., 2020; Abdullah-Al-

	Kamran, 2020; Chow and Salmena, 2020; Pradhan et al., 2020)
hsa-miR-206 (*), hsa-miR-454-3p, hsa-miR-4775	(Yousefi et al., 2020; Chow and Salmena, 2020)
hsa-let-7b-3p	(Fulzele et al., 2020; Maghsoudnia et al., 2020)
hsa-let-7e-3p	(Jafarinejad-Farsangi et al., 2020)
hsa-miR-130a-3p (*), hsa-miR-484	(Yousefi et al., 2020; Mukhopadhyay and Mussa, 2020)
hsa-miR-4793-3p, hsa-miR-6790-5p, hsa-miR-873-3p	(Sardar et al., 2020)
hsa-miR-214-5p, hsa-miR-7111-5p (*), hsa-miR-7705, hsa-miR-7848-3p	[Abdullah-Al-Kamran, 2020,27*]
hsa-miR-203b-3p, hsa-miR-362-3p (*), hsa-miR-5701	(Chow and Salmena, 2020, Mukhopadhyay and Mussa, 2020)
hsa-miR-4276	(Gasparello et al., 2020)
hsa-miR-330-3p, hsa-miR-543	(Mukhopadhyay and Mussa, 2020)

hsa-miR-570-3p	(Pradhan et al., 2020)
hsa-miR-1307-5p, hsa-miR-575	(Pierce et al., 2020)
hsa-miR-583	(Ahmadi and Moradi; 2020)
hsa-miR-6873-5p	(Van Campen et al., 2020)
hsa-miR-450a-5p	(Gasparello et al., 2020)
hsa-miR-15a-3p	(Tribolet et al., 2020)
hsa-miR-181a-3p	(Wang and Tatakis, 2020]
hsa-miR-192-3p	(Alshabi et al., 2020)
hsa-miR-216a-3p	(Fujii, 2020)
hsa-miR-3184-3p	(Santos et al., 2020)
hsa-miR-33b-3p	(Guo, H. et al., 2019)
hsa-miR-3614-3p, hsa-miR-3972, hsa-miR-412-3p, hsa-miR-4299, hsa-miR-4503, hsa-miR-4509, hsa-miR-450b-5p, hsa-miR-4645-3p, hsa-miR-4801, hsa-miR-6740-3p, hsa-miR-6879-3p, hsa-miR-8069	(Fulzele et al., 2020)
hsa-miR-877-5p	(Morales et al., 2020)
hsa-miR-6809-5p	(Saçar and Adan, 2020)

hsa-miR-1249-5p, hsa-miR-3146, hsa-miR-3616-3p, hsa-miR-3621, hsa-miR-4265, hsa-miR-4479, hsa-miR-4508, hsa-miR-4536-3p, hsa-miR-4538, hsa-miR-4768-3p, hsa-miR-4787-5p, hsa-miR-4798-5p, hsa-miR-4800-5p, hsa-miR-5092, hsa-miR-5705, hsa-miR-598-5p, hsa-miR-6125, hsa-miR-670-5p, hsa-miR-6834-5p	NA

6.3.2 Results of supervised models, SVM and RF

a) SVM

For this part, we divided our entire subset into two classes and then applied a GridSearch with a weighted score to obtain the best parameters to be used with this supervised model. By intersecting the data acquired from the Immport database, we identified a list of 818 miRNAs that interacted with genes involved in the immunological process and 1730 miRNAs that were not involved. It is worth mentioning that the proportion between the positive and negative genes is unbalanced, and for that reason, we will only choose a subset of the negative class, at random, for our experimentation purposes. The size of this subset was approximately 10% that of the negative samples. Therefore, we will end up with 818 records for the positive class and 173 samples from the negative class. This proportion is roughly 21% of the positive class. The parameters selected, with a ten-fold grid search cross-validation, were: kernel=rbf, C=0.01, and gamma=10-5. We found a training accuracy of 82.54%, standard deviation of 0.0008, test accuracy of 82.54%, and standard deviation of 0.003. Even though these results seemed promising, we found a misclassification for the true negative class and type-II error, and we validated these results by obtaining an ROC curve from our results. The results of the ROC curve are shown in Figure 6.4.

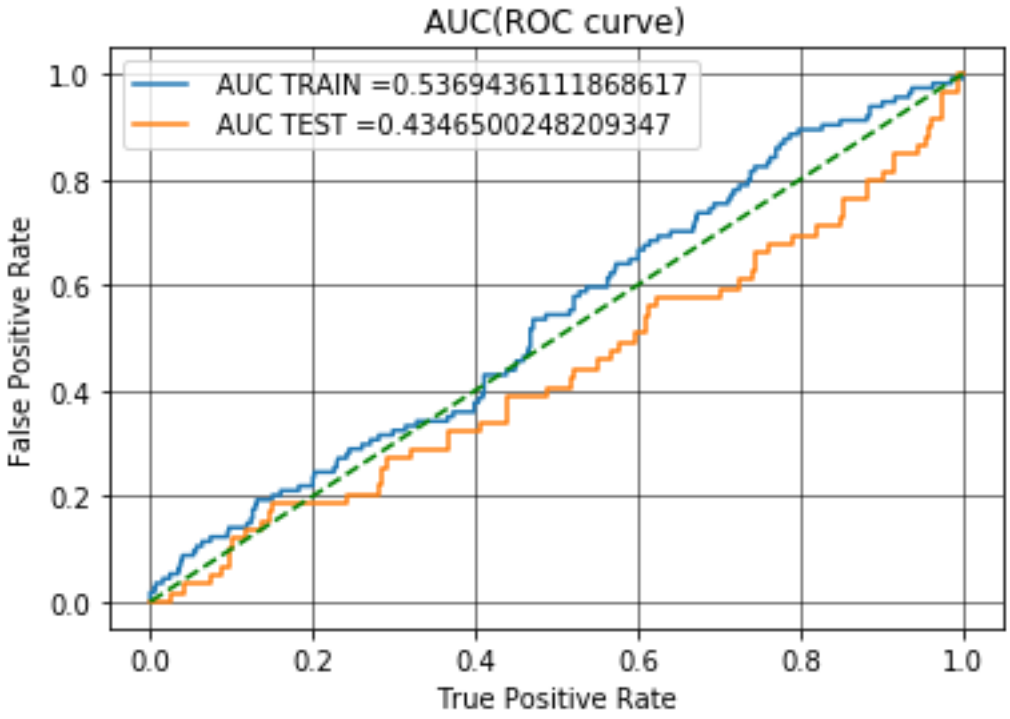


Figure 6.4 ROC curve obtained from the SVM model.

Our SVM model results showed that, although we tried to balance the positive and negative classes by extracting a subset of the latter, our classifier had trouble discriminating samples from the negative class. These samples corresponded to those miRNAs prone to bind to the viral mRNA, but they did not have validated interactions with genes that participate in immunological processes.

b) Random Forest

After dividing our data into positive and negative samples and applying the grid search algorithm for hyperparameter tuning, we selected the best hyperparameters that could be suitable for our model. We found that these hyperparameters were splitting criteria=entropy, maximum depth of the trees=12, number of maximum features was set to default, and the number of trees or estimators was equal to 10. With these hyperparameters, we obtained an accuracy of approximately 82.88% in the test set, which is the same as before; however, we found a type-II error with no values for the true negative classes again. From these results, which usually occurred in imbalanced scenarios, we decided to use a one-class model to see how this model behaves with our data.

6.3.3 One-class SVM comparison with supervised models

From the application of a grid search algorithm, we found that the best hyperparameters were kernel=rbf, ν =0.01, and gamma= 0.03449. With these data, we execute our model ten times with different subsets of samples from the negative class, and, in the end, we averaged the results of the metrics selected. The values that we found were more stable and promising than those obtained using the supervised models. We obtained an average accuracy of 90% with a standard deviation of 0.02, sensitivity values of 96.18% with a standard deviation of 0.01, and a specificity of 76.39% with a standard deviation of 0.1. As we can observe, the one-class SVM yielded more stable results than the supervised models we tested.

The purpose of this part of our research was to identify miRNAs that interact with immune genes that could bind to SARS-CoV-2. Based on the results obtained, we hypothesize that the miRNAs present in the body's immune response could bind to this viral strand, and therefore their future study to know more about how our body responds, via their immune system, to this infection is of utmost importance.

6.4 Discussion

When we used our one-class SVM model to find miRNAs classified as novelties or anomalies, we compared them with the existing literature, and we found some interesting results. For instance, we found that, according to Maghsoudnia et al., 2020, miRNA let-7b was found to target specific respiratory chain genes, and it has been used in drug targeting in apoptotic cells. We hypothesized that drug therapies against SARS disease could be made due to the similarity of respiratory disease scenarios. Gasparello et al., 2021 found that hsa-miR-450a-5p potentially binds to the IL-8 gene, which is involved in what is known as cytokine storm, and this relationship is one of the predictors of patient survival at the time of hospitalization. Another case is how miRNA hsa-miR-192-3p binds to NR1H4 and is responsible for SARS-CoV-2 progression (Ahmadi and Moradi, 2020). Among other interesting findings, miRNA hsa-miR-6809-5p, according to Ahmadi and Moradi (2020), binds to the S-region or spike gene from the SARS-CoV-2 genome; however, we found that it could also bind to the 5'UTR region.

Even though the genome from the SARS-CoV-2 virus differs from that found in influenza cases, we found that some miRNAs are also present in influenza cases, but that they are prone to bind to the 5 UTR region from the SARS-CoV-2 mRNA sequence. We found that this binding occurs with hsa-miR-6873-5p (Abdullah-Al-Kamran et al., 2020) and hsa-miR-4276 (Chow and Salmena, 2020). Other types of miRNAs also appear in some diseases that could

result in a co-morbidity in SARS-CoV-2 virus scenarios, such as hsa-miR-7111-5p, which binds the HOXC8 gene up-regulating it, and it is present in obesity scenarios (Chow and Salmena, 2020).

Other interesting results, but with no direct relation with the SARS-CoV-2 given by the current literature (values of NA in Table 1), were those miRNAs obtained as outliers from our one-class model, but they are related to the symptomatology or the organs attacked by this disease. For example, hsa-mir-3146 is present in rhinosinusitis and hsa-mir-4508 in some unspecific heart diseases (Fulzele et al., 2020). Other interesting results were about miRNAs hsa-miR-4787-5p and hsa-miR-4800-5p. According to Maghsoudnia et al. (20209, the expression of hsa-miR-4787-5p is used as a biomarker and could be involved in acute aortic dissection cases, which is a highly morbid disease; specifically, this miRNA is upregulated in these cases. Individuals with an individual predisposition to develop this disease are elderly males with a history of hypertension, being one of the factors present in patients with a severe prognosis from SARS-CoV-2. Another miRNA, hsa-miR-4800-5p, also appears in vascular diseases such as the previous one and, more specifically, to Kawasaki disease (Sardar et al., 2020). Based on this evidence, we hypothesize that these binding miRNAs, treated as outliers from our one-class SVM, deserve special attention because their presence and affinity to the SARS-COo-2 virus could mean that these miRNAs are present in persons with some pathologies.

With regard to the application of two-class supervised models, we found that the results were a little misleading. This is because, even though we had an acceptable level of accuracy of approximately 82% for both models, we concluded that there were no true negative samples classified correctly when we verified our confusion matrices. We arrived at the conclusion that we were dealing with a pseudo-imbalanced class. We coined this term of pseudo-imbalanced class because even the partition of our data in positive and negative classes with the golden rule of 70/30 for training (positive) and test (negative) classes, the models were difficult to classify the negative classes correctly. This situation did not occur when we applied our one-class SVM model, which showed more stable results discussed in the Results section of the present article.

It is worth mentioning that, at the moment of conducting the present research, we were unable to find literature regarding the use of one-class models for the study of interactions between miRNAs, immune genes, and SARS-CoV-2. Furthermore, we believe that the study

of miRNAs that bind to these viral strands, which are involved in the regulation of the immune system, could fill up a research gap in attempts to understand how our immune system reacts in the presence of this viral infection.

6.5 Summary

The interaction between host miRNAs and SARS-CoV-2 mRNA could provide a potential field of research to find new therapeutics that could alleviate the current pandemic situation we are currently involved. Using a one-class SVM model to a set of human miRNAs, we were able to find a subset of these miRNAs prone to bind to the 5'UTR region of the SARS-CoV-2 mRNA genome. The results validated from the literature also gave us some results in which the miRNAs found were related to other forms of diseases, such as obesity and lung damage. Additionally, we were able to find promising results in the study of miRNAs involved with genes that participate in the immunological response of the body. These miRNAs could bind to the SARS-CoV-2 viral mRNA, establishing some new avenues for future research in this field considering that these miRNAs are present in the immunological response of the human body and serve to counterattack this type of viral infection.

CHAPTER 7
Conclusions and Future Work

7.1 Summary of the study

In general, we have aimed with this research work to explore the possibility of using One-class models with features extracted from the genetic sequences of non-coding RNAs and genes. In addition to predicting their probable couple or binding between these molecules, we used them to validate their participation in certain diseases, such as breast cancer, breast neoplasm scenarios, and probable influence of SARS-CoV-2 RNA.

a) Chapter 4 explored the possibility of using one-class models to validate the interactions between miRNAs and the ERBB2 gene, responsible for the prognosis of breast cancer scenarios. The extracted features were derived from the genomic sequences of these molecules. The results obtained were comparable to those found in the literature reviewed, with a sensitivity of 80.49% and specificity of 86.49% for the one-class SVM model.

b) Chapter 5 sought to differentiate between breast cancer and breast neoplasm scenarios by studying the interaction between miRNAs and lncRNAs and their relationship with these diseases. The extracted features were related to sequence features, considering alignments between sequences and k-mers. The obtained results obtained accuracy results of 95.44% for the one-class model, 88.79% for the SVM, and 99.65% for the random forest model. As in the previous model, the results obtained were comparable to those found in the current literature. The novelty of this study is that no studies have examined the interactions between these molecules and breast cancer or neoplasms at the moment of the present research.

c) Chapter 6 describes our application of a one-class SVM model to predict probable interactions between miRNAs that relate to genes involved in the immune system and to verify, via machine-learning models, if there could exist a probable binding between these miRNAs and the SARS-COV-2 RNA. The results obtained were 90.90%, 96.18%, and 76.39% for accuracy, sensitivity, and specificity, respectively. The novelty of this chapter is that there are no studies, at present, that investigate the interaction of miRNAs involved with genes that take part in the immune system and that they could probably bind to the SARS-CoV-2 RNA sequence.

7.2 Conclusions

In the present study, we demonstrated the feasibility of the use of One-class models, in contraposition with two class supervised techniques, joined with features extracted from the binding of genetic sequences of non-coding RNAs and genes that could induce breast cancer scenarios or immune genes that could be prone to bind to the SARS-COV-2 viral strand.

The results obtained via the application of one-class SVM models serve for those scenarios in which the application of a two-class supervised model is not possible because of the scarcity of the data. Additionally, the use of features obtained from the genetic sequence that arise in the process of binding between non-coding RNAs and genes is a promising path in the research of the interactions of these two genetic molecules and their relationships with diseases, such as some form of cancers or virally caused infections.

7.3 Future work

We believe that using computationally expensive models such as those used in machine learning and focusing on one-class models to outperform the difficulties of differentiating classes in bioinformatics scenarios is a promising approach. Additionally, these studies should be accompanied by probable methods for extracting useful features based on the genetic sequences of these molecules. These studies could also be expanded to study other forms of non-coding RNAs. Furthermore, it is of utmost importance to study the relationship of these miRNAs in the influence of viral forms, such as SARS variants or other diseases, that could serve in the near future as biomarkers or to predict the future prognosis of patients with potential diseases. Even though we aimed to use one-class ML models in this study, which have a scarce presence in the current literature, it would be advisable to compare them with DL models such as recurrent neural networks or convolutional models. However, and according to Ockham´s razor principle, which points out the simplicity in the construction of models (Audi, 2015), we believe there is still a promising field of research in classic Machine Learning models, for example, by using XGBoost or a simple logistic regression, which performs adequately well in the presence of imbalanced data (Shahri, et al., 2021) or even outperforms some DL models (Shwartz-Ziv and Armon, 2021).

References

Abdullah-Al-Kamran, K., Abul Bashar, M., Khademul, I., n.d. SARS-CoV-2 proteins exploit host's genetic and epigenetic mediators for the annexation of key host signaling pathways that confers its immune evasion and disease pathophysiology.

Aggarwal, C.C., 2017. Outlier Analysis. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-47578-3

Agarwal, V., Bell, G.W., Nam, J.-W. and Bartel, D.P. (2015). Predicting effective microRNA target sites in mammalian mRNAs. eLife, [online] 4. Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4532895/ [Accessed 23 Dic. 2021].

Ahmadi, A., Moradi, S., 2020. In silico analysis suggests the RNAi-enhancing antibiotic enoxacin as a potential inhibitor of SARS-CoV-2 infection (preprint). In Review. https://doi.org/10.21203/rs.3.rs-96308/v1

Alshabi, A.M., Shaikh, I.A., Vastrad, B.M., Vastrad, C.M., 2020. Identification of Differentially Expressed Genes and Enriched Pathways in SARS-CoV-2/ COVID-19 using Bioinformatics Analysis (preprint). In Review. https://doi.org/10.21203/rs.3.rs-122015/v1

Ardekani, A.M., Naeini, M.M., 2010. The Role of MicroRNAs in Human Diseases 2, 19.

Audi, R. (Ed.), 2015. The Cambridge Dictionary of Philosophy (Third Edition). Cambridge: Cambridge University Press. doi:10.1017/CBO9781139057509.

Bartel, D., 2004. MicroRNAs: Genomics, Biogenesis, Mechanism, and Function. Cell 116, 281–297. https://doi.org/10.1016/S0092-8674(04)00045-5

Breuer, K., Foroushani, A.K., Laird, M.R., Chen, C., Sribnaia, A., Lo, R., Winsor, G.L., Hancock, R.E.W., Brinkman, F.S.L., Lynn, D.J., 2013. InnateDB: systems biology of innate immunity and beyond—recent updates and continuing curation. Nucleic Acids Research 41, D1228–D1233. https://doi.org/10.1093/nar/gks1147

Chen, H., Pan, H., Qian, Y., Zhou, W., Liu, X., 2018. MiR-25-3p promotes the proliferation of triple negative breast cancer by targeting BTG2. Mol Cancer 17, 4. https://doi.org/10.1186/s12943-017-0754-0

Chen, Y., Wang, X., 2020. miRDB: an online database for prediction of functional microRNA targets. Nucleic Acids Research 48, D127–D131. https://doi.org/10.1093/nar/gkz757

Chyi-Ying A., C., Ann-Bin, S., 1995. AU-rich elements: characterization and importance in mRNA degradation. Trends in Biochemical Sciences 20, 465–470. https://doi.org/doi.org/10.1016/S0968-0004(00)89102-1

Chow, J.T.-S., Salmena, L., 2020. Prediction and Analysis of SARS-CoV-2-Targeting MicroRNA in Human Lung Epithelium. Genes 11, 1002. https://doi.org/10.3390/genes11091002

Coleman, W.B., 2020. Neoplasia, in: Essential Concepts in Molecular Pathology. Elsevier, pp. 55–80. https://doi.org/10.1016/B978-0-12-813257-9.00004-8

Compeau, P. & Pevzner, P. (2015). 'Finding shared k-mers' in Compeau and Pevzner Bioinformatics Algorithms: An Active Learning Approach, Second Edition Vol. 2, Active Learning Publishers, p. 326.

Condorelli, G., Latronico, M.V.G., Cavarretta, E., 2014. microRNAs in Cardiovascular Diseases. Journal of the American College of Cardiology 63, 2177–2187. https://doi.org/10.1016/j.jacc.2014.01.050

De Cola, A., Volpe, S., Budani, M.C., Ferracin, M., Lattanzio, R., Turdo, A., D'Agostino, D., Capone, E., Stassi, G., Todaro, M., Di Ilio, C., Sala, G., Piantelli, M., Negrini, M., Veronese, A., De Laurenzi, V., 2015. miR-205-5p-mediated downregulation of ErbB/HER receptors in breast cancer stem cells results in targeted therapy resistance. Cell Death Dis 6, e1823–e1823. https://doi.org/10.1038/cddis.2015.192

Ding, J., Li, X., Hu, H., 2016. TarPmiR: a new approach for microRNA target site prediction. Bioinformatics 32, 2768–2775. https://doi.org/10.1093/bioinformatics/btw318

Einert, T.R., Netz, R.R., 2011. Theory for RNA Folding, Stretching, and Melting Including Loops and Salt. Biophysical Journal 100, 2745–2753. https://doi.org/10.1016/j.bpj.2011.04.038

Eude, T., Chang, C., 2018. One-class SVM for biometric authentication by keystroke dynamics for remote evaluation: One-class SVM for biometric authentication by keystroke dynamics for remote evaluation. Computational Intelligence 34, 145–160. https://doi.org/10.1111/coin.12122

Fernandez-Delgado, M., Cernadas, E., Barro, S., Amorim, D., 2014. Do we Need Hundreds of Classifiers to Solve Real World Classification Problems? J. Mach. Learn. Res. 15, 3133–3181.

Ferracin, M., Bassi, C., Pedriali, M., Pagotto, S., D'Abundo, L., Zagatti, B., Corrà, F., Musa, G., Callegari, E., Lupini, L., Volpato, S., Querzoli, P., Negrini, M., 2013. miR-125b targets erythropoietin and its receptor and their expression correlates with metastatic potential and ERBB2/HER2 expression. Mol Cancer 12, 130. https://doi.org/10.1186/1476-4598-12-130

Fang, C., Zhao, Y., Guo, B., 2013. MiR-199b-5p targets HER2 in breast cancer cells. J. Cell. Biochem. 114, 1457–1463. https://doi.org/10.1002/jcb.24487

Fu, L., Peng, Q., 2017. A deep ensemble model to predict miRNA-disease association. Sci Rep 7, 14482. https://doi.org/10.1038/s41598-017-15235-6

Fujii, Y., 2020. The Etiology of COVID-19 in Silico by SARS-Cov-2 Infection with the Quantum MicroRNA Language-AI. VIJ 4. https://doi.org/10.23880/vij-16000243

Fulzele, S., Sahay, B., Yusufu, I., Lee, T.J., Sharma, A., Kolhe, R., Isales, C.M., 2020. COVID-19 Virulence in Aged Patients Might Be Impacted by the Host Cellular MicroRNAs Abundance/Profile. Aging and disease 11, 509. https://doi.org/10.14336/AD.2020.0428

Gasparello, J., Finotti, A., Gambari, R., 2021. Tackling the COVID-19 "cytokine storm" with microRNA mimics directly targeting the 3'UTR of pro-inflammatory mRNAs. Medical Hypotheses 146, 110415. https://doi.org/10.1016/j.mehy.2020.110415

Ghaemi, Z., Soltani, B.M., Mowla, S.J., 2019. MicroRNA-326 Functions as a Tumor Suppressor in Breast Cancer by Targeting ErbB/PI3K Signaling Pathway. Front. Oncol. 9, 653. https://doi.org/10.3389/fonc.2019.00653

Grimson, A., Farh, K.K.-H., Johnston, W.K., Garrett-Engele, P., Lim, L.P., Bartel, D.P., 2007. MicroRNA Targeting Specificity in Mammals: Determinants beyond Seed Pairing. Molecular Cell 27, 91–105. https://doi.org/10.1016/j.molcel.2007.06.017

Guo, H., Ha, C., Dong, H., Yang, Z., Ma, Y., Ding, Y., 2019. Cancer-associated fibroblast-derived exosomal microRNA-98-5p promotes cisplatin resistance in ovarian cancer by targeting CDKN1A. Cancer Cell Int 19, 347. https://doi.org/10.1186/s12935-019-1051-3

Guo, Z.-H., You, Z.-H., Wang, Y.-B., Yi, H.-C., Chen, Z.-H., 2019. A Learning-Based Method for LncRNA-Disease Association Identification Combing Similarity Information and Rotation Forest. iScience 19, 786–795. https://doi.org/10.1016/j.isci.2019.08.030

Gupta, I., Sareyeldin, R., Al-Hashimi, I., Al-Thawadi, H.A., Al Farsi, H., Vranic, S., Al Moustafa, A.-E., 2019. Triple Negative Breast Cancer Profile, from Gene to microRNA, in Relation to Ethnicity. Cancers 11, 363. https://doi.org/10.3390/cancers11030363

Gutiérrez-Cárdenas, J., Wang, Z., 2021a. Classification of Breast Cancer and Breast Neoplasm Scenarios Based on Machine Learning and Sequence Features from lncRNAs–miRNAs-Diseases Associations. Interdiscip Sci Comput Life Sci 13, 572–581. https://doi.org/10.1007/s12539-021-00451-6

Gutiérrez-Cárdenas, J., Wang, Z., 2021b. One-class models for validation of miRNAs and ERBB2 gene interactions based on sequence features for breast cancer scenarios. ICT Express S2405959521000333. https://doi.org/10.1016/j.icte.2021.03.001

Hamberg, M., Backes, C., Fehlmann, T., Hart, M., Meder, B., Meese, E., Keller, A., 2016. miRTargetLink—miRNAs, Genes and Interaction Networks. IJMS 17, 564. https://doi.org/10.3390/ijms17040564

Han, J., 2004. The Drosha-DGCR8 complex in primary microRNA processing. Genes & Development 18, 3016–3027. https://doi.org/10.1101/gad.1262504

Hand, D.J., 2006. Classifier Technology and the Illusion of Progress. Statist. Sci. 21, 1–14. https://doi.org/10.1214/088342306000000060

Harries, L.W., 2012. Long non-coding RNAs and human disease. Biochemical Society Transactions 40, 902–906. https://doi.org/10.1042/BST20120020

Hartl, D.L., 2020. Essential genetics and genomics, 7th ed. Jones and Bartlett Publishers, Boston.

Hofacker, I.L., 2003. Vienna RNA secondary structure server. Nucleic Acids Research 31, 3429–3431. https://doi.org/10.1093/nar/gkg599

Huang, H.-Y., Lin, Y.-C.-D., Li, J., Huang, K.-Y., Shrestha, S., Hong, H.-C., Tang, Y., Chen, Y.-G., Jin, C.-N., Yu, Y., Xu, J.-T., Li, Y.-M., Cai, X.-X., Zhou, Z.-Y., Chen, X.-H., Pei, Y.-Y., Hu, L., Su, J.-J., Cui, S.-D., Wang, F., Xie, Y.-Y., Ding, S.-Y., Luo, M.-F., Chou, C.-H., Chang, N.-W., Chen, K.-W., Cheng, Y.-H., Wan, X.-H., Hsu, W.-L., Lee, T.-Y., Wei, F.-X., Huang, H.-D., 2019. miRTarBase 2020: updates to the experimentally validated microRNA–target interaction database. Nucleic Acids Research gkz896. https://doi.org/10.1093/nar/gkz896

Huang, Y.-A., Huang, Z.-A., You, Z.-H., Zhu, Z., Huang, W.-Z., Guo, J.-X., Yu, C.-Q., 2019. Predicting lncRNA-miRNA Interaction via Graph Convolution Auto-Encoder. Front. Genet. 10, 758. https://doi.org/10.3389/fgene.2019.00758

Irigoien, I., Sierra, B., Arenas, C., 2014. Towards Application of One-Class Classification Methods to Medical Data. The Scientific World Journal 2014, 1–7. https://doi.org/10.1155/2014/730712

Jafarinejad-Farsangi, S., Jazi, M.M., Rostamzadeh, F., Hadizadeh, M., 2020. High affinity of host human microRNAs to SARS-CoV-2 genome: An in silico analysis. Non-coding RNA Research 5, 222–231. https://doi.org/10.1016/j.ncrna.2020.11.005

Jensen, M.A., Ferretti, V., Grossman, R.L., Staudt, L.M., 2017. The NCI Genomic Data Commons as an engine for precision medicine. Blood 130, 453–459. https://doi.org/10.1182/blood-2017-03-735654

Kertesz, M., Iovino, N., Unnerstall, U., Gaul, U., Segal, E., 2007. The role of site accessibility in microRNA target recognition. Nat Genet 39, 1278–1284. https://doi.org/10.1038/ng2135

Kozomara, A., Griffiths-Jones, S., 2014. miRBase: annotating high confidence microRNAs using deep sequencing data. Nucl. Acids Res. 42, D68–D73. https://doi.org/10.1093/nar/gkt1181

Lamkiewicz, K., Barth, E., Barth, E., Ibrahim, B., 2018. Identification of potential microRNAs associated with Herpesvirus family based on bioinformatic analysis (preprint). Bioinformatics. https://doi.org/10.1101/417782

Lee, R. C., Feinbaum, R. L., & Ambros, V., 1993. The C. elegans heterochronic gene lin-4 encodes small RNAs with antisense complementarity to lin-14. Cell, 75(5), 843–854. https://doi.org/10.1016/0092-8674(93)90529-y

le Maire, A., Germain, P., Bourguet, W., 2020. Protein-protein interactions in the regulation of RAR–RXR heterodimers transcriptional activity, in: Methods in Enzymology. Elsevier, pp. 175–207. https://doi.org/10.1016/bs.mie.2020.02.007

Liu, F.T., Ting, K.M., Zhou, Z.-H., 2008. Isolation Forest, in: 2008 Eighth IEEE International Conference on Data Mining. Presented at the 2008 Eighth IEEE International Conference on Data Mining (ICDM), IEEE, Pisa, Italy, pp. 413–422. https://doi.org/10.1109/ICDM.2008.17

Loh, H.-Y., Norman, B.P., Lai, K.-S., Rahman, N.M.A.N.Abd., Alitheen, N.B.M., Osman, M.A., 2019. The Regulatory Role of MicroRNAs in Breast Cancer. IJMS 20, 4940. https://doi.org/10.3390/ijms20194940

López-Camarillo, C., Marchat, L.A., 2013. MicroRNAs in Cancer, 1st ed. CRC Press, Boca Raton, FL.

Lu, M., Shi, B., Wang, J., Cao, Q., Cui, Q., 2010. TAM: A method for enrichment and depletion analysis of a microRNA category in a list of microRNAs. BMC Bioinformatics 11, 419. https://doi.org/10.1186/1471-2105-11-419

Maghsoudnia, N., Baradaran Eftekhari, R., Naderi Sohi, A., Norouzi, P., Akbari, H., Ghahremani, M.H., Soleimani, M., Amini, M., Samadi, H., Dorkoosh, F.A., 2020. Mitochondrial delivery of microRNA mimic let-7b to NSCLC cells by PAMAM-based nanoparticles. Journal of Drug Targeting 28, 818–830. https://doi.org/10.1080/1061186X.2020.1774594

Martinez-Gutierrez, A.D., Cantú de León, D., Millan-Catalan, O., Coronel-Hernandez, J., Campos-Parra, A.D., Porras-Reyes, F., Exayana-Alderete, A., López-Camarillo, C., Jacobo-Herrera, N.J., Ramos-Payan, R., Pérez-Plasencia, C., 2020. Identification of miRNA Master Regulators in Breast Cancer. Cells 9, 1610. https://doi.org/10.3390/cells9071610

Matamala, N., Vargas, M.T., González-Cámpora, R., Arias, J.I., Menéndez, P., Andrés-León, E., Yanowsky, K., Llaneza-Folgueras, A., Miñambres, R., Martínez-Delgado, B., Benítez, J., 2016. MicroRNA deregulation in triple negative breast cancer reveals a role of miR-498 in regulating BRCA1 expression. Oncotarget 7, 20068–20079. https://doi.org/10.18632/oncotarget.7705

McAnena, P., Tanriverdi, K., Curran, C., Gilligan, K., Freedman, J.E., Brown, J.A.L., Kerin, M.J., 2019. Circulating microRNAs miR-331 and miR-195 differentiate local luminal a from metastatic breast cancer. BMC Cancer 19, 436. https://doi.org/10.1186/s12885-019-5636-y

Miao, Y.-R., Liu, W., Zhang, Q., Guo, A.-Y., 2018. lncRNASNP2: an updated database of functional SNPs and mutations in human and mouse lncRNAs. Nucleic Acids Research 46, D276–D280. https://doi.org/10.1093/nar/gkx1004

Mohammed, J., Flynt, A.S., Siepel, A., Lai, E.C., 2013. The impact of age, biogenesis, and genomic clustering on Drosophila microRNA evolution. RNA 19, 1295–1308. https://doi.org/10.1261/rna.039248.113

Morales, L., Oliveros, J.C., Fernandez-Delgado, R., tenOever, B.R., Enjuanes, L., Sola, I., 2017. SARS-CoV-Encoded Small RNAs Contribute to Infection-Associated Lung Pathology. Cell Host & Microbe 21, 344–355. https://doi.org/10.1016/j.chom.2017.01.015

mirWalk, 2020, mirWalk database, viewed 5 March 2020, http://mirwalk.umm.uni-heidelberg.de/resources/

Mukhopadhyay, D., Mussa, B.M., 2020. Identification of Novel Hypothalamic MicroRNAs as Promising Therapeutics for SARS-CoV-2 by Regulating ACE2 and TMPRSS2 Expression: An In Silico Analysis. Brain Sciences 10, 666. https://doi.org/10.3390/brainsci10100666

NCI, 2020, The website of the National Cancer Institute, viewed 5 March, 2020, https://www.cancer.gov

Needleman, S. B. & Wunsch, C. D. (1970), 'A general method applicable to the search for similarities in the amino acid sequence of two proteins', J. Mol. Biol. 48 , 443-453.

Negrini, M., Calin, G.A., 2008. Breast cancer metastasis: a microRNA story. Breast Cancer Res 10, 303. https://doi.org/10.1186/bcr1867

Nersisyan, S., Engibaryan, N., Gorbonos, A., Kirdey, K., Makhonin, A., Tonevitsky, A., 2020. Potential role of cellular miRNAs in coronavirus-host interplay. PeerJ 8, e9994. https://doi.org/10.7717/peerj.9994

Ninio-Many, L., Hikri, E., Burg-Golani, T., Stemmer, S.M., Shalgi, R., Ben-Aharon, I., 2020. miR-125a Induces HER2 Expression and Sensitivity to Trastuzumab in Triple-Negative Breast Cancer Lines. Front. Oncol. 10, 191. https://doi.org/10.3389/fonc.2020.00191

Ngwenyama, O. 2014. Foundations for understanding The Process of Knowledge Reproduction in Academic Research. Presentation lecture for PhD students held at Rhodes University, March 2014.

Ohler, U., 2004. Patterns of flanking sequence conservation and a characteristic upstream motif for microRNA gene identification. RNA 10, 1309–1322. https://doi.org/10.1261/rna.5206304

Ojala, M., Garriga, G.C., 2009. Permutation Tests for Studying Classifier Performance, in: 2009 Ninth IEEE International Conference on Data Mining. Presented at the 2009 Ninth IEEE International Conference on Data Mining (ICDM), IEEE, Miami Beach, FL, USA, pp. 908–913. https://doi.org/10.1109/ICDM.2009.108.

Pan, J.-Y., Zhang, F., Sun, C.-C., Li, S.-J., Li, G., Gong, F.-Y., Bo, T., He, J., Hua, R.-X., Hu, W.-D., Yuan, Z.-P., Wang, X., He, Q.-Q., Li, D.-J., 2017. miR-134: A Human Cancer Suppressor? Molecular Therapy - Nucleic Acids 6, 140–149. https://doi.org/10.1016/j.omtn.2016.11.003

Pang-Ning T., Steinbach M., and Kumar V. (2005). 'Alternative Metrics' in Pang-Ning, T. Introduction to Data Mining, (First Edition). USA: Addison-Wesley Longman Publishing Co., Inc., p. 292.

Penyige, A., Márton, É., Soltész, B., Szilágyi-Bónizs, M., Póka, R., Lukács, J., Széles, L., Nagy, B., 2019. Circulating miRNA Profiling in Plasma Samples of Ovarian Cancer Patients. IJMS 20, 4533. https://doi.org/10.3390/ijms20184533

Pevsner J. (2015). 'Global Sequence Alignment: Algorithm of Needleman and Wunsch' in Pevsner J. Bioinformatics and Functional Genomics, (Third Edition). UK: Wiley-Blackwell, pp. 96-100.

Pham, V.V., Zhang, J., Liu, L., Truong, B., Xu, T., Nguyen, T.T., Li, J., Le, T.D., 2019. Identifying miRNA-mRNA regulatory relationships in breast cancer with invariant causal prediction. BMC Bioinformatics 20, 143. https://doi.org/10.1186/s12859-019-2668-x

Pierce, J.B., Simion, V., Icli, B., Pérez-Cremades, D., Cheng, H.S., Feinberg, M.W., 2020. Computational Analysis of Targeting SARS-CoV-2, Viral Entry Proteins ACE2 and TMPRSS2, and Interferon Genes by Host MicroRNAs. Genes 11, 1354. https://doi.org/10.3390/genes11111354

Pletscher-Frankild, S., Pallejà, A., Tsafou, K., Binder, J.X., Jensen, L.J., 2015. DISEASES: Text mining and data integration of disease–gene associations. Methods 74, 83–89. https://doi.org/10.1016/j.ymeth.2014.11.020

Pradhan, U.K., Anand, P., Sharma, N.K., Kumar, P., Kumar, A., Pandey, R., Padwad, Y., Shankar, R., 2020. Various RNA-binding proteins and their conditional networks explain miRNA biogenesis and help to reveal the potential SARS-CoV-2 host miRNAome system (preprint). Bioinformatics. https://doi.org/10.1101/2020.06.18.156851

Prosenjit, P., Chakraborty, A., Sarkar, D., Langthasa, M., Rahman, M., Bari, M., Singha, RK.S., Malakar, A.K., Chakraborty, S., 2018. Interplay between miRNAs and human diseases. J Cell Physiol 233, 2007–2018. https://doi.org/10.1002/jcp.25854

Rehman, O., Zhuang, H., Muhamed Ali, A., Ibrahim, A., Li, Z., 2019. Validation of miRNAs as Breast Cancer Biomarkers with a Machine Learning Approach. Cancers 11, 431. https://doi.org/10.3390/cancers11030431

Rivals, I., Personnaz, L., Taing, L., Potier, M.-C., 2007. Enrichment or depletion of a GO category within a class of genes: which test? Bioinformatics 23, 401–407. https://doi.org/10.1093/bioinformatics/btl633

Saçar Demirci, M.D., Adan, A., 2020. Computational analysis of microRNA-mediated interactions in SARS-CoV-2 infection. PeerJ 8, e9369. https://doi.org/10.7717/peerj.9369

Santos, J.M.O., Peixoto da Silva, S., Gil da Costa, R.M., Medeiros, R., 2020. The Emerging Role of MicroRNAs and Other Non-Coding RNAs in Cancer Cachexia. Cancers 12, 1004. https://doi.org/10.3390/cancers12041004

Sardar, R., Satish, D., Gupta, D., 2020. Identification of Novel SARS-CoV-2 Drug Targets by Host MicroRNAs and Transcription Factors Co-regulatory Interaction Network Analysis. Front. Genet. 11, 571274. https://doi.org/10.3389/fgene.2020.571274

Sareyeldin, R.M., Gupta, I., Al-Hashimi, I., Al-Thawadi, H.A., Al Farsi, H.F., Vranic, S., Al Moustafa, A.-E., 2019. Gene Expression and miRNAs Profiling: Function and Regulation in Human Epidermal Growth Factor Receptor 2 (HER2)-Positive Breast Cancer. Cancers 11, 646. https://doi.org/10.3390/cancers11050646

Sarshad, A.A., Juan, A.H., Muler, A.I.C., Anastasakis, D.G., Wang, X., Genzor, P., Feng, X., Tsai, P.-F., Sun, H.-W., Haase, A.D., Sartorelli, V., Hafner, M., 2018. Argonaute-miRNA Complexes Silence Target mRNAs in the Nucleus of Mammalian Stem Cells. Molecular Cell 71, 1040-1050.e8. https://doi.org/10.1016/j.molcel.2018.07.020

Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C., 2001. Estimating the Support of a High-Dimensional Distribution. Neural Computation 13, 1443–1471. https://doi.org/10.1162/089976601750264965

Schwab, R., Palatnik, J.F., Riester, M., Schommer, C., Schmid, M., Weigel, D., 2005. Specific Effects of MicroRNAs on the Plant Transcriptome. Developmental Cell 8, 517–527. https://doi.org/10.1016/j.devcel.2005.01.018

Scitable, 2014, hairpin loop (mRNA), viewed 27 February 2020, https://www.nature.com/scitable/definition/hairpin-loop-mrna-314/

Sedaghat, N., Fathy, M., Modarressi, M.H., Shojaie, A., 2018. Combining Supervised and Unsupervised Learning for Improved miRNA Target Prediction. IEEE/ACM Trans. Comput. Biol. and Bioinf. 1–1. https://doi.org/10.1109/TCBB.2017.2727042

Severe acute respiratory syndrome coronavirus 2 isolate Wuhan-Hu-1, complete genome. (2020). NCBI Nucleotide. [online] Available at: https://www.ncbi.nlm.nih.gov/nuccore/MN908947.3?report=graph [Accessed 23 Dec. 2021].

Shahri, N.H.N.B.M., Lai, S.B.S., Mohamad, M.B., Rahman, H.A.B.A., Rambli, A.B., 2021. Comparing the Performance of AdaBoost, XGBoost, and Logistic Regression for Imbalanced Data. ms 9, 379–385. https://doi.org/10.13189/ms.2021.090320

Shen, Z.-Y., Zhang, Z.-Z., Liu, H., Zhao, E.-H., Cao, H., 2014. miR-375 inhibits the proliferation of gastric cancer cells by repressing ERBB2 expression. Experimental and Therapeutic Medicine 7, 1757–1761. https://doi.org/10.3892/etm.2014.1627

Shwartz-Ziv, R., Armon, A., 2021. Tabular Data: Deep Learning is Not All You Need. arXiv:2106.03253 [cs].

Sloma M., Zuker M., Mathews D. (2020) 'Predictive Methods Using RNA Sequences' in Baxevanis A., Bader G., Wishart D. Bioinformatics, (4th Edition). USA: John Wiley & Sons, Inc., pp. 155-159.

Spinosa, E.J., de Carvalho, Andre, 2004. SVMs for novel class detection in Bioinformatics, in: Brazilian Workshop on Bioinformatics. pp. 81–88.

Statnikov, A., Aliferis, C.F., n.d. Are Random Forests Better than Support Vector Machines for Microarray-Based Cancer Classification?, in: AMIA Annual Symposium Proceedings. Presented at the AMIA Symposium, pp. 686–690.

Stelzer, G., Rosen, N., Plaschkes, I., Zimmerman, S., Twik, M., Fishilevich, S., Stein, T.I., Nudel, R., Lieder, I., Mazor, Y., Kaplan, S., Dahary, D., Warshawsky, D., Guan-Golan, Y., Kohn, A., Rappaport, N., Safran, M., Lancet, D., 2016. The GeneCards Suite: From Gene

Data Mining to Disease Genome Sequence Analyses. Current Protocols in Bioinformatics 54. https://doi.org/10.1002/cpbi.5

Sticht, C., De La Torre, C., Parveen, A., Gretz, N., 2018. miRWalk: An online resource for prediction of microRNA binding sites. PLoS ONE 13, e0206239. https://doi.org/10.1371/journal.pone.0206239

Sugita, B.M., Pereira, S.R., de Almeida, R.C., Gill, M., Mahajan, A., Duttargi, A., Kirolikar, S., Fadda, P., de Lima, R.S., Urban, C.A., Makambi, K., Madhavan, S., Boca, S.M., Gusev, Y., Cavalli, I.J., Ribeiro, E.M.S.F., Cavalli, L.R., 2019. Integrated copy number and miRNA expression analysis in triple negative breast cancer of Latin American patients. Oncotarget 10, 6184–6203. https://doi.org/10.18632/oncotarget.27250

Tran, D.H., Pham, T.H., Satou, K., Ho, T.B., 2008. Prediction of microRNA Hairpins using One-Class Support Vector Machines, in: 2008 2nd International Conference on Bioinformatics and Biomedical Engineering. Presented at the 2008 2nd International Conference on Bioinformatics and Biomedical Engineering, IEEE, Shanghai, China, pp. 33–36. https://doi.org/10.1109/ICBBE.2008.15

Tribolet, L., Kerr, E., Cowled, C., Bean, A.G.D., Stewart, C.R., Dearnley, M., Farr, R.J., 2020. MicroRNA Biomarkers for Infectious Diseases: From Basic Research to Biosensing. Front. Microbiol. 11, 1197. https://doi.org/10.3389/fmicb.2020.01197

Trobaugh, D.W., Klimstra, W.B., 2017. MicroRNA Regulation of RNA Virus Replication and Pathogenesis. Trends in Molecular Medicine 23, 80–93. https://doi.org/10.1016/j.molmed.2016.11.003

Van Campen, H., Bishop, J.V., Abrahams, V.M., Bielefeldt-Ohmann, H., Mathiason, C.K., Bouma, G.J., Winger, Q.A., Mayo, C.E., Bowen, R.A., Hansen, T.R., 2020. Maternal Influenza A Virus Infection Restricts Fetal and Placental Growth and Adversely Affects the Fetal Thymic Transcriptome. Viruses 12, 1003. https://doi.org/10.3390/v12091003

Vastrad, B., Vastrad, C., Tengli, A., 2020. Bioinformatics analyses of significant genes, related pathways, and candidate diagnostic biomarkers and molecular targets in SARS-CoV-2/COVID-19. Gene Reports 21, 100956. https://doi.org/10.1016/j.genrep.2020.100956

Vo, D.T., Karanam, N.K., Ding, L., Saha, D., Yordy, J.S., Giri, U., Heymach, J.V., Story, M.D., 2019. miR-125a-5p Functions as Tumor Suppressor microRNA And Is a Marker of Locoregional Recurrence And Poor prognosis in Head And Neck Cancer. Neoplasia 21, 849–862. https://doi.org/10.1016/j.neo.2019.06.004

Vogelstein, B., Kinzler, K.W. (Eds.), 2002. The genetic basis of human cancer, 2nd ed. ed. McGraw-Hill, Medical Pub. Division, New York.

Wang, Y., Chen, L., Wu, Z., Wang, M., Jin, F., Wang, N., Hu, X., Liu, Z., Zhang, C.-Y., Zen, K., Chen, J., Liang, H., Zhang, Y., Chen, X., 2016. miR-124-3p functions as a tumor suppressor in breast cancer by targeting CBL. BMC Cancer 16, 826. https://doi.org/10.1186/s12885-016-2862-4

Wang, Y., Tatakis, D.N., 2020. Integrative mRNA/miRNA expression analysis in healing human gingiva. J Periodontol JPER.20-0397. https://doi.org/10.1002/JPER.20-0397

Wapinski, O., Chang, H.Y., 2011. Long noncoding RNAs and human disease. Trends in Cell Biology 21, 354–361. https://doi.org/10.1016/j.tcb.2011.04.001

Wen, J., Liu, Y., Shi, Y., Huang, H., Deng, B., Xiao, X., 2019. A classification model for lncRNA and mRNA based on k-mers and a convolutional neural network. BMC Bioinformatics 20, 469. https://doi.org/10.1186/s12859-019-3039-3

Witkos, T.M., Koscianska, E., Krzyzosiak, W.J., 2011. Practical Aspects of microRNA Target Prediction. CMM 11, 93–109. https://doi.org/10.2174/156652411794859250

Xie, F., Hosany, S., Zhong, S., Jiang, Y., Zhang, F., Lin, L., Wang, X., Gao, S., Hu, X., 2017. MicroRNA-193a inhibits breast cancer proliferation and metastasis by downregulating WT1. PLoS ONE 12, e0185565. https://doi.org/10.1371/journal.pone.0185565

Xu, J., Wong, C.-W., 2013. Enrichment Analysis of miRNA Targets, in: Ying, S.-Y. (Ed.), MicroRNA Protocols, Methods in Molecular Biology. Humana Press, Totowa, NJ, pp. 91–103. https://doi.org/10.1007/978-1-62703-083-0_8

Yan, X., Chao, T., Tu, K., Zhang, Y., Xie, L., Gong, Y., Yuan, J., Qiang, B., Peng, X., 2007. Improving the prediction of human microRNA target genes by using ensemble algorithm. FEBS Letters 581, 1587–1593. https://doi.org/10.1016/j.febslet.2007.03.022

Yang, S., Wang, Y., Lin, Y., Shao, D., He, K., Huang, L., 2020. LncMirNet: Predicting LncRNA–miRNA Interaction Based on Deep Learning of Ribonucleic Acid Sequences. Molecules 25, 4372. https://doi.org/10.3390/molecules25194372

Yousef, M., Jung, S., Showe, L.C., Showe, M.K., 2008. Learning from positive examples when the negative class is undetermined- microRNA gene identification. Algorithms Mol Biol 3, 2. https://doi.org/10.1186/1748-7188-3-2

Yousef, M., Najami, N., Khalifav, W., 2010. A comparison study between one-class and two-class machine learning for MicroRNA target detection. JBiSE 03, 247–252. https://doi.org/10.4236/jbise.2010.33033

Yousefi, H., Poursheikhani, A., Bahmanpour, Z., Vatanmakanian, M., Taheri, M., Mashouri, L., Alahari, S.K., 2020. SARS-CoV infection crosstalk with human host cell noncoding-RNA machinery: An in-silico approach. Biomedicine & Pharmacotherapy 130, 110548. https://doi.org/10.1016/j.biopha.2020.110548

Zhang, P., Meng, J., Luan, Y., Liu, C., 2020. Plant miRNA–lncRNA Interaction Prediction with the Ensemble of CNN and IndRNN. Interdiscip Sci Comput Life Sci 12, 82–89. https://doi.org/10.1007/s12539-019-00351-w

Zhao, D., Sui, Y., Zheng, X., 2016. miR-331-3p inhibits proliferation and promotes apoptosis by targeting HER2 through the PI3K/Akt and ERK1/2 pathways in colorectal cancer. Oncology Reports 35, 1075–1082. https://doi.org/10.3892/or.2015.4450

Zhao, Y., Li, H., Fang, S., Kang, Y., wu, W., Hao, Y., Li, Z., Bu, D., Sun, N., Zhang, M.Q., Chen, R., 2016. NONCODE 2016: an informative and valuable data source of long non-coding RNAs. Nucleic Acids Res 44, D203–D208. https://doi.org/10.1093/nar/gkv1252

Zheng, K., You, Z.-H., Wang, L., Zhou, Y., Li, L.-P., Li, Z.-W., 2019. MLMDA: a machine learning approach to predict and validate MicroRNA–disease associations by integrating of heterogenous information sources. J Transl Med 17, 260. https://doi.org/10.1186/s12967-019-2009-x

Appendices

Appendix A – Articles accepted

Article 1

KICS

ICT Express

Available online 30 March 2021

In Press, Corrected Proof

One-class models for validation of miRNAs and ERBB2 gene interactions based on sequence features for breast cancer scenarios

Juan Gutiérrez-Cárdenas [a, b], Zenghui Wang [a]

Show more

+ Add to Mendeley Share Cite

https://doi.org/10.1016/j.icte.2021.03.001

Abstract

One challenge in miRNA–genes–diseases interaction studies is that it is challenging to find labeled data that indicate a positive or negative relationship between miRNA and genes. The use of one-class classification methods shows a promising path for validating them. We have applied two one-class classification methods, Isolation Forest and One-class SVM, to validate miRNAs interactions with the ERBB2 gene present in breast cancer scenarios using features extracted via sequence-binding. We found that the One-class SVM outperforms the Isolation Forest model, with values of sensitivity of 80.49% and a specificity of 86.49% showing results that are comparable to previous studies. Additionally, we have demonstrated that the use of features extracted from a sequence-based approach (considering miRNA and gene sequence binding characteristics) and one-class models have proven to be a feasible method for validating these genetic molecule interactions.

Keywords

MiRNAs; Breast cancer; One-class models; Unsupervised learning

Article 2

Original research article | Published: 21 June 2021

Classification of Breast Cancer and Breast Neoplasm Scenarios Based on Machine Learning and Sequence Features from lncRNAs–miRNAs-Diseases Associations

Juan Gutiérrez-Cárdenas & Zenghui Wang

Interdisciplinary Sciences: Computational Life Sciences (2021) | Cite this article

107 Accesses | **1** Altmetric | Metrics

Abstract

The influence of non-coding RNAs, such as lncRNAs (long non-coding RNAs) and miRNAs (microRNAs), is undeniable in several diseases, for example, in the formation of neoplasms and cancer scenarios. However, there are challenges due to the scarcity of validated datasets and the imbalance in the data. We found that the research of associations between miRNAs-lncRNAs and diseases is limited or done separately. In addition, those investigations, which use Machine Learning models joined with genomic sequence features extracted from miRNAs and lncRNAs, are few compared with using some methods such as genomic expression or Deep Learning techniques. In this paper, we propose a structure of using supervised and unsupervised machine learning models with genomic sequence features, such as k-mers, sequence alignments, and energy folding values, to validate miRNAs and lncRNAs association with breast cancer and neoplasms scenarios. Using One-Class SVM for outlier detection and comparing two supervised models such as SVM and Random Forest, we manage to obtain accuracy results of 95.44% for the One-class model, with 88.79% and 99.65% for the SVM and Random Forest models, respectively. The results showed a promising path for the study of sequence features interactions joined with Machine Learning models comparable to those found in the existing literature.

Appendix B – Principal functions from source code from Chapter 4

```python
def convertMatch(mirname,dftarbase,dfMirWalk):
    #extract from dftarbase only those rows that match with mirnam
    mirExtract=dftarbase["mirna"]==mirname
    dfMirnaTar=dftarbase[mirExtract]
    print (dfMirnaTar.shape)

    #extract validated records from mirWalk
    lUniqueGeneT=[]
    lUniqueGeneT=dfMirnaTar["geneName"].unique()
    print (lUniqueGeneT)

    #extract matches from mirWalk according to the list of unique genes form dfMirnaTar

    #we have to get rid of the duplicates in dfMirnaTar first, those that have the same genename
    #it seems it would not be straightforward there are mixed values of positive and negative in tar base
    #and there is no way how to relate it with mirWalk, and mirWalk contains some needed numerical data

    df3 = dfMirWalk[dfMirWalk.genesymbol.isin(lUniqueGeneT)]

    for index, row in df3.iterrows():
        row = dfMirnaTar[row['genesymbol'] == dfMirnaTar['geneName']]['positive_negative']
        df3.loc[index,'positive_negative'] = row.iloc[0]

    print (df3)
    df3.to_csv("mirWalkFilter.csv",header=True,index=False)
    return df3
```

```python
def oneSVM(df,df1):
    #add colum output with s value as strong
    df["output"]=1      #put s as output indicating it is a strong value
    listDrop=["refseqid","seed","position","validated","TargetScan","miRDB"]
    dfD=df.drop(listDrop,axis="columns")

    #test df
    df1["output"]=-1
    y=df["output"].to_list()+df1["output"].to_list()
    dfTest=df1.drop(listDrop,axis="columns")

    listTrain=["start","end","bindingp","energy","accessibility","au","phylopstem","phylopflank","me","

    X_train=dfD.loc[:,listTrain]
    X_test=dfTest.loc[:,listTrain]
    scaler=preprocessing.StandardScaler().fit(X_train)
    X_train=scaler.transform(X_train)
    X_test=scaler.transform(X_test)
    dfTrain=pd.DataFrame(X_train)
    print("dfTrain")
    dfTrain.columns=listTrain
    print(dfTrain)
    dfTrain["output"]="s"
    print(dfTrain.head(3))
    dfTest=pd.DataFrame(X_test)
    dfTest.columns=listTrain
    dfTest["output"]="w"
    dfJoin=dfTrain.append(dfTest)
    dfJoin.to_csv("dataJoin2.csv")

    #gridSearch(dfTrain,dfTest)
    #dfJoin=dfJoin.sample(frac=1).reset_index(drop=True)


    dfJoin.to_csv("dataJoin.csv")
    print("shafe of dfJoin ",dfJoin.shape)
    print(dfJoin.head(3))
    sns.pairplot(dfJoin,vars=listTrain,hue="output")

    plt.figure(figsize=(40,15))

    cdf=pd.concat([dfTrain,dfTest])
    mdf=pd.melt(cdf,id_vars=["output"],var_name="data",value_name="values")
    sns.boxplot(x="data",y="values",data=mdf,hue="output")

    plt.show()

    #hyperparameters tuning
    kernels=["poly","rbf","sigmoid"]
    #nus=np.arange(0,1,0.01).tolist()
```

Python file

```
    nus=np.linspace(0.01,1,99)
    gammas=[1e-1, 1e-2, 1e-3, 1e-4]
    degrees=[1, 2, 3, 4, 5, 6]
    #print (nus)
    acc=0
    params=()

    #X_train and X_test contains the train and test set with the features selected
    print ("shape train",X_train.shape)
    print ("shape test",X_test.shape)

    #=======temp variables
    X_trainT=X_train
    X_testT=X_test

    allData=np.concatenate((X_train,X_test))
    print ("shape allData",allData.shape)


    #best parameters  {'degree': 1, 'gamma': 0.1, 'kernel': 'sigmoid', 'nu': 0.1716326530612245}
    print("final results")
    oneClass=OneClassSVM(kernel="rbf",nu=0.17163, gamma=0.1)
    #oneClass=OneClassSVM(kernel="poly",degree=3,nu=0.7979, gamma=0.0001)
    oneClass.fit(X_trainT)    #train with the normal data, two cases: casse 1 train with X_trainT and test in X_trainT, case 2 train
    predMirna=oneClass.predict(X_testT)
    print ("One class SVM predictions")
    print(predMirna)
    print([i+2 for i,x in enumerate(predMirna) if x == -1])
    print(predMirna.size)
    print (predMirna[predMirna==-1].size/predMirna.size)
    print("other ",len(predMirna)," outlier ",sum(map(lambda x : x<0, predMirna)))

    #isolation forest
    #np.random.shuffle(allData)
    print ("isolation forest")
    listRIsoF=[]
    isolationF=IsolationForest(n_estimators=20,max_features=0.7,max_samples=30,behaviour="new",bootstrap=True,contamination=0.3)
    for i in range(0,100):
        isolationF.fit(X_trainT)    #originally was allData, lets change it to XtrainT
        y_pred=isolationF.predict(X_testT)
        listRIsoF.append(sum(map(lambda x : x<0, y_pred)))
    print (sum(listRIsoF)/100)
```

```
def gridSearch(dfTrain,dfTest):
    print("Gridsearch\n")

    #let´s add a column 1 for strong and 0 for weak
    dfTrain["output"]=1
    dfTest["output"]=-1
    dfJoin=dfTrain.append(dfTest)
    print(dfJoin.shape)
    dfJoin.to_csv("dataGS.csv")
    y=dfJoin["output"]
    dfJoin = dfJoin.sample(frac=1).reset_index(drop=True)   #shuffle dataframe with replacement and also the indexes
    X=dfJoin.drop("output",axis="columns")
    X.to_csv("dataXGS.csv")

    kernels=["linear","rbf","sigmoid","poly"]
    #nus=np.arange(0,1,0.01).tolist()
    nus=np.linspace(0.01,1,99)
    gammas=[1e-1, 1e-2, 1e-3, 1e-4]
   # gammas=[1e-1, 1e-2, 1e-3]
    degrees = [1, 2, 3, 4 ,5]
    #gammas=["scale","auto"]

    scoring = make_scorer(metrics.f1_score,
                      average="weighted")

    iso =  IsolationForest(contamination=0.3)
    """oneClass=OneClassSVM()
    print(oneClass.get_params().keys())
    grid=GridSearchCV(estimator=oneClass,param_grid=dict(kernel=kernels,nu=nus,degree=degrees,
                                                         gamma=gammas),scoring=scoring,n_jobs=-1,verbose=5,refit='False',cv=5)"""

    parameters = {'n_estimators':[10,20,40,60,80,100], 'max_features':[0.1, 0.2, 0.3,0.4,0.5,0.6,0.7,0.8,0.9,1.0], 'max_samples':[10,20,30,40,50],'bootstrap':[True,False]}
    clf = GridSearchCV(iso, parameters,  scoring=scoring,n_jobs=-1,verbose=5,cv=10)
    clf.fit(X,y)
    print ("best parameters ",clf.best_params_)
```

Appendix C – Principal functions from source code from Chapter 5

```
from subprocess import PIPE
import subprocess
from Bio import SeqIO
from Bio.Alphabet import generic_dna
from Bio.Seq import Seq
from Bio import pairwise2
from Bio.pairwise2 import format_alignment
from Bio import SeqUtils
import re
from itertools import product
import pandas as pd
import csv
import numpy as np
from bs4 import BeautifulSoup
import requests
import time
import os

def energy(rna,option=0):
    #obtains the energy according to the Vienna package
    rna1="GCGCUUCGCCGCGCGCC&GCGCUUCGCCGCGCGCA"
    rna2="GCGCUUCGCCGCGCGCC&GCGCU"
    rna3="GCGCUU&GCGCU"
    rna4="AAGUUCGGGGGGUGGAU&TTTTAGAAATCCACCTA"
    #rna="GCGCUUCGCCGCGCGCC"
    if (option==1):
        p = subprocess.Popen('RNAcofold.exe', stdin=PIPE,stdout=PIPE,shell=True,env={'PATH': 'D:\\ViennaRNAPackage'})
        #p = subprocess.Popen('RNAduplex.exe -p', stdin=PIPE,stdout=PIPE,shell=True,env={'PATH': 'D:\\ViennaRNAPackage'})
        answer = p.communicate(rna.encode())
        #answer="GCGCUUCGCCGCGCGCCGCGCUUCGCCGCGCGCA (((...))).(((((((...))).))))))). (-17.60)"

        #extract kcal
        cofold = re.findall("[+-]?[0-9]+(?:\.[0-9]+)?", str(answer[0]))
        #print(answer[0],"\n")
        print(cofold[0])
        return cofold[0]
    else:
        if (option==2):

            p = subprocess.Popen('RNAfold.exe', stdin=PIPE,stdout=PIPE,shell=True,env={'PATH': 'D:\\ViennaRNAPackage'})
            answer=p.communicate(rna.encode())

            kcal = re.findall("[+-]?[0-9]+(?:\.[0-9]+)?", str(answer[0]))
            #print(kcal[0])
            return kcal[0]

```

```
def nwScore(mrna,lncr):
    """
    this method will try to obtain the nw from an mirna and a noncode lncrna sequence
    maybe it would be needed to have a sliding window of n-mers
    """
    mirna=Seq(mrna)
    mirnab=mirna.back_transcribe()
    mirnac=(mirna.reverse_complement()).complement()
    score=[]
    best=""
    scoreB=0
    cad=kmersTest(lncr,len(mirna))
    lncrnaseq=""
    for i in cad:
        align=pairwise2.align.globalxx(i,mirnac) #lncrna vs mirna
        for a in align:
            if (a[2]>scoreB):
                lncrnaseq=a[0]
                mirseq=a[1]
                scoreB=a[2]
                best=format_alignment(*align[0])

    lnc=Seq(lncrnaseq)
    lncrnaT=lnc.transcribe()
    lncrnatG=re.sub('-','',str(lncrnaT))
    seqEnergy=lncrnatG+"&"+mirnac

    cofold=energy(seqEnergy,1)
    kcal=energy(mirnab,2)

    kmers=[]
    ckmers=[]
    dicKmers=countKMers(str(mirna).upper(),2)
    #probable kmers
    probK=[''.join(c) for c in product('AUCG', repeat=2)]

    for i in probK:
        kmers.append(i)
        ckmers.append(0)
    #list of counted k-mers
    for k,v in dicKmers.items():
        #print (k," ",v)
        ckmers[kmers.index(k)]=v


    #lets extract the kmers from the lncrna
    lncrna=Seq(lncr)
    lncrna=lncrna.transcribe()
    dicKmersL=countKMers(str(lncrna).upper(),2)
```

```
    lncmers=[]
    for i in probK:
        kmers.append(i)
        lncmers.append(0)
    #list of counted k-mers
    for k,v in dicKmersL.items():
        #print (k," ",v)
        lncmers[kmers.index(k)]=v



    score5mers=matchKmers(mirna,lncr)

    return ckmers,lncmers,cofold,kcal,score5mers
```

```
def matchKmers(mirna,lncr='AG'):
    #this method would extract the 5-mers from the miRNA sequence according to the paper
    #with this 5-mers it will form a sequence and then will match these to the lncrna
    lnc=Seq(lncr)
    lncrnTr=lnc.transcribe()
    dicKmers=countKMers(str(mirna).upper(),5)
    kmerSeq=[]
    for k,v in dicKmers.items():
        kmerSeq.append(k)
    kmerS=''.join(kmerSeq)
    return kmerS


def countKMers(seq,k):
    counts={}
    numKmers=len(seq)-k+1
    for i in range(numKmers):
        kmer=seq[i:i+k]
        if kmer not in counts:
            counts[kmer]=1
        else:
            counts[kmer]=counts[kmer]+1
    return counts
```

```
def getlncList():
    #file lncrna-diseases_experiment Disease    Pubmed  lncRNA
    #extract lncRNA and disease, where disease equals breast cancer
    #inputFile="lncrna-diseases_experiment.txt"
    inputFile="lncrna-diseases_predicted.txt"
    data=pd.read_csv(inputFile,sep='\t')
    #print (data.head())
    #values=data[data['Disease']=="breast cancer"]
    values=data[data['Disease']=="Breast Neoplasms"]
    """
         Disease    Pubmed           lncRNA
    33   breast cancer  21506106  NONHSAT140597.2
    """
    listlnc=[]
    values=values["lncRNA"].drop_duplicates()
    #['NONHSAT140597.2', 'NONHSAT070566.2', 'NONHSAT130416.2'...
    listlnc=values.tolist()
    print(listlnc,"\n")

    #get alias
    listalias=[]
    """
    ID  gene    transcript  alias
    pubmed has been dropped and there are some genes thar are empty
    """
    inputFileA="human_lncrna_alias.txt"

    colnames=["ID","transcript","alias"]
    dataA=pd.read_csv(inputFileA,sep='\t',usecols=colnames)
    for i in listlnc:
        value=dataA[dataA["transcript"]==i].index.values
        alias=dataA.iloc[value]["alias"]
        listalias.append(''.join(alias.values))

    #still in the file is the presence of [] signs, but because we will need to curate
    #the sequences mostly by hand, there would be not so much problem
    #update: lets save lncrnaName, alias, disease
    for item in listalias:
        print(item)
    with open("transcriptsAliasP.csv","w") as f:
        i=0
        for item in listlnc:
            f.write(item+",")
            f.write(listalias[i]+",")
            f.write("Breast Neoplasm")
            f.write('\n')
            i=i+1
```

```
def getmirnasC():
    #it will extract a lits of mirnas, experimental or validated, by comparing the generated csv files
    #that are transcriptsAlias and transcriptAliasP (experimental)
    #with the files mirnas_lncrs_validated and mirnas_lncrs_conserved
    #this metthod do the conserved ones
    #fileName="transcriptsAlias.csv"
    fileName="transcriptsAliasP.csv" #lets try this
    fileCons="mirnas_lncrnas_conserved.txt"
    data=pd.read_csv(fileName,names=['lncrna'],usecols=[0])
    #print(data.head())
    print(data['lncrna'])
    datamirnaC=pd.read_csv(fileCons,sep='\t',names=["lncrna","mirna"])
    print(datamirnaC.head())
    listDF=pd.DataFrame(columns=["lncrna","mirna"])

    print(data.iloc[0]["lncrna"])

    #example to search for the ncrna in the datamirnaC, we can save the lncrna and mirna found on a csv file
    #datatest="NONHSAT153368.1"
    #print("testing")
    #datatest="NONHSAT039742.2"
    #print(datamirnaC[datamirnaC["lncrna"]==datatest])

    print("TEST")
    #listV=[]
    for item in data["lncrna"]:
        value=datamirnaC[datamirnaC["lncrna"]==item]
        #print(len(value))
        if (len(value)>0):
            listDF=listDF.append(value)

    listDF.to_csv("lncrna_mirna_consP.csv",index=False)
```

```
def testbs(data):
    #in this code the sequence should stip the last digit of the version
    #data="NONHSAT146018"
    #page = requests.get("http://www.noncode.org/show_rna.php?id=" + data)
    #print(page.content)
    page = requests.get("http://www.noncode.org/show_rna.php?id="+data)

    soup = BeautifulSoup(page.content, "html.parser")

    element = soup.findAll('table', class_="table-1")[1]
    element2 = element.findAll('tr')[1]
    element3 = element2.findNext('td')
    your_data = str(element3.renderContents(), "utf-8")
    return your_data
```

```
def addMirSeq():
    inputFile="mature.fa"
    inputLnc="lncrna_mirna_validated_Seq.csv"
    dataC=pd.read_csv(inputLnc)
    #print (dataC["mirna"])
    print (dataC.head())
    fastaSeq=SeqIO.parse(open(inputFile),'fasta')
    #test="hsa-miR-323a-5p"

    rowNumber=0

    test="hsa-miR-29b-3p"

    for item in dataC["mirna"]:
        fastaSeq=SeqIO.parse(open(inputFile),'fasta')
        for j in fastaSeq:
            if (j.name==item):
                seqM=j.seq
                print(item," ",seqM)
                dataC.loc[dataC.index[rowNumber], 'seqMirna'] = str(seqM)
            else:
                seqM=""
        rowNumber=rowNumber+1

    dataC.to_csv("test3.csv",index=False)
    for item in fastaSeq:
        print (item.name)
        print (item.seq)
```

```
def extractFeatures():
    #this method would read the lncrna_mirna_SeqM files and modify the mirna sequences
    #so that this could be matched with the lncrna by using the nwScore method
    #the features generated must be added in the lncrna_mirna_SeqM
    #file="lncrna_mirna_validatedPSeq_10kM.csv"
    #we have considered the first 3000
    #file="out1Small.csv"
    #file="positive.csv"
    file="4.csv"
    dataC=pd.read_csv(file)
    print(dataC.head())
    kmersP=[]
    kmersC=[]
    kmersL=[]  #lnc kmers
    kmersT=[]
    kmersV=[]
    #row index
    i=0
    c=0
    #probable kmers
    kmersP=[''.join(c)+'m' for c in product('AUCG', repeat=2)] #changed to 3 mers uncomment
    kmersP1=[''.join(c)+'l' for c in product('AUCG', repeat=2)]
    kmersT=kmersP+kmersP1
    print(kmersT)
    for seqL,seqM in zip(dataC["seq"],dataC["seqMirna"]):
        print ("SEQUENCE NUMBER ",i)
        kmersC,kmersL,cofold,kcal,score5mers=nwScore(seqM,seqL)
        j=0
        for item in kmersP:
            dataC.loc[dataC.index[i], item] = kmersC[j]/len(seqM)
            j=j+1
        j=0
        for item in kmersP1:
            dataC.loc[dataC.index[i], item] = kmersL[j]/len(seqL)
            j=j+1
        dataC.loc[dataC.index[i], 'cofold'] = cofold
        dataC.loc[dataC.index[i], 'kcal'] = kcal
        dataC.loc[dataC.index[i], 'score5'] = score5mers
        i=i+1

    dataC.to_csv("positive4f.csv",index=False)
```

```
def oneClass(dfPos,dfNeg):
    #dfPos is 15% approximately of the total registers
    listDrop=['lncrna','mirna','seq','seqMirna','Class']
    dfTrain=dfNeg
    dfTest=dfPos
    dfJoin=dfTrain.append(dfTest)
    dfJoin = dfJoin.sample(frac=1).reset_index(drop=True)
    print("join ",dfJoin.shape)
    yT=dfTest['Class']
    dfJoin.to_csv("dataJoin.csv",index=False)
    dfTrain=dfNeg.drop(listDrop,axis="columns")
    dfTest=dfPos.drop(listDrop,axis="columns")
    #gridSearch(dfTrain,dfTest)
    X_train=dfTrain
    X_test=dfTest
    allData=dfJoin.drop(listDrop,axis="columns")
    oneClass=OneClassSVM(kernel="poly",nu=0.17163, degree=3)
    oneClass.fit(allData)
    predMirna=oneClass.predict(allData)
    print ("One class SVM predictions")
    pd.options.display.max_seq_items = 2000
    print(predMirna)
    listOut=[]
    listOut=[i for i,x in enumerate(predMirna) if x == -1]
    lenListOut=len(listOut)

    c=0
    for item in listOut:
        elem=dfJoin.iloc[item]['Class']
        if (elem=='Breast Cancer'):
            c=c+1
    print ("value of c ",c)
    print ("percentage is ",(c*100)/(lenListOut))

    print(predMirna.size)
    print ("outilers percentage ",predMirna[predMirna==-1].size/predMirna.size)
    print("other ",len(predMirna)," outlier ",sum(map(lambda x : x<0, predMirna)))
```

```
def rf(dfNeg,dfPos):
    dfTrain=dfPos
    dfTest=dfNeg
    dfTestr=dfTest.sample(frac=0.045)
    print("train shape ",dfTrain.shape)
    print("test shape ",dfTestr.shape)
    dfTrain['Class']=1
    dfTestr['Class']=2
    df=dfTrain.append(dfTestr)
    df= df.sample(frac=1).reset_index(drop=True)
    data=df
    listDrop=['lncrna','mirna','seq','seqMirna','Class']
    y=data['Class']
    colNames=list(data.columns.values)
    print(colNames)
    colNamesF=list(set(colNames)-set(listDrop))
    print(colNamesF)
    min_max_scaler=preprocessing.MinMaxScaler()
    data1=data.drop(listDrop,axis="columns")
    x_scaled=min_max_scaler.fit_transform(data1)

    data[colNamesF]=x_scaled
    data.to_csv("randomForestScaled.csv")

    X=data.drop(listDrop,axis="columns")
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
    print("train set ",X_train.shape)
    print("test set ",X_test.shape)


    scoring = {'accuracy':make_scorer(accuracy_score),
               'f1-score':make_scorer(metrics.f1_score,average="weighted")
               }
    model=RandomForestClassifier(criterion= 'gini', max_depth= 8, max_features= 'auto', n_estimators= 50)
    model.fit(X_train,y_train)
    pred=model.predict(X_test)
    score=accuracy_score(y_test,pred)
    print(score)
    conf_mat = confusion_matrix(y_test,pred)
    print(conf_mat)


    return score
```

```
def svm(dfNeg,dfPos):
    dfTrain=dfPos
    dfTest=dfNeg
    #test Set should be reduced to 130 samples aprox or the 4%
    dfTestr=dfTest.sample(frac=0.10) #uncomment for all the data
    dfTestr=dfTest.sample(frac=0.045)
    print("register count ",dfTestr.shape)
    #join the validated dat with the experimental
    dfTrain['Class']=1
    dfTestr['Class']=2
    df=dfTrain.append(dfTestr)
    print("train dimension ",dfTrain.shape)
    print("test dimension ",dfTestr.shape)
    df= df.sample(frac=1).reset_index(drop=True)
    df.to_csv("check.csv")
    data=df
    #normalize columns cofold   kcal    score5
    listDrop=['lncrna','mirna','seq','seqMirna','Class']
    y=data['Class']
    X=data.drop(listDrop,axis="columns")
    data2=X
    colNames=list(data2.columns.values)
    min_max_scaler=preprocessing.MinMaxScaler()
    x_scaled=min_max_scaler.fit_transform(data2)

    data[colNames]=x_scaled
    data.to_csv("checkScaled.csv")

    X=data.drop(listDrop,axis="columns")

    kernels=["poly"]
    Cs=[1e-2, 1e-1, 1, 1e1, 1e2]
    degrees = [2, 3, 4, 5]
    # gammas=[1e-1, 1e-2, 1e-3]
    #degrees = [2, 3, 4]
    model=SVC()
    scoring = make_scorer(metrics.f1_score,
                          average="weighted")

    #test with cv
    #should I not keep the original train set somewhere?
    scoring = {'accuracy':make_scorer(accuracy_score),
               'f1-score':make_scorer(metrics.f1_score,average="weighted")
               }
    model=SVC(kernel='rbf',C=100,gamma=0.01,random_state=4)
    cv=cross_validate(model, X, y, scoring=scoring, cv = 5)
    print(cv,"\n")
    print ("TRAIN ACC ",cv['train_accuracy']," accuracy ",cv['train_accuracy'].mean()," std ",cv['train_accuracy'].std()*2,"\n")
    print ("TEST ACC ",cv['test_accuracy']," accuracy ",cv['test_accuracy'].mean()," std ",cv['test_accuracy'].std()*2)
```

```
def plotdata(dfPos,dfNeg):
    df=dfPos.append(dfNeg,ignore_index=True)
    #df.to_csv("checkScaledPlot.csv")
    dfTemp=df
    print(df.shape)
    listDrop=['lncrna','mirna','seq','seqMirna','Class']
    df=df.drop(listDrop,axis="columns")

    colNames=list(df.columns.values)
    colNamesF=list(set(colNames)-set(listDrop))
    min_max_scaler=preprocessing.MinMaxScaler()
    x_scaled=min_max_scaler.fit_transform(df)

    df[colNamesF]=x_scaled
    dfTemp[colNamesF]=x_scaled  #temp
    listDropT=['lncrna','mirna','seq','seqMirna'] #temp
    dfTemp=dfTemp.drop(listDropT,axis="columns") #temp

    pca=PCA(n_components=2)
    princC=pca.fit_transform(df)
    print(pca.explained_variance_ratio_)
    princDf=pd.DataFrame(data=princC,columns=['pca1','pca2'])
    print(princDf.head())
    plt.figure()
    plt.figure(figsize=(10,10))
    plt.xticks(fontsize=12)
    plt.yticks(fontsize=14)
    plt.xlabel('pc1',fontsize=20)
    plt.ylabel('pc2',fontsize=14)
    targets=['Breast Cancer','Breast Neoplasm']
    colors=['r','g']
    for target,color in zip(targets,colors):
        indices=dfTemp['Class']==target
        plt.scatter(princDf.loc[indices,'pca1'],princDf.loc[indices,'pca2'],c=color,s=50)
    plt.legend(targets,prop={'size':15})

    plt.figure(1)
    df.to_csv("checkScaledPlot.csv")
    #sns.pairplot(df)
    plt.figure(figsize=(15,15))
    sns.boxplot(data=df)
```

```
def plotVar(dfPos,dfNeg):
    df=dfPos.append(dfNeg,ignore_index=True)
    dfTemp=df
    listDrop=['lncrna','mirna','seq','seqMirna','Class']
    df=df.drop(listDrop,axis="columns")
    colNames=list(df.columns.values)
    colNamesF=list(set(colNames)-set(listDrop))
    min_max_scaler=preprocessing.MinMaxScaler()
    x_scaled=min_max_scaler.fit_transform(df)
    dfTemp[colNamesF]=x_scaled
    listDrop=['lncrna','mirna','seq','seqMirna']
    dfTemp=dfTemp.drop(listDrop,axis="columns")
    print(dfTemp.head())
    #the melt function takes the Class column that is Breast Cancer or Breast Neoplasm and it transforms it into
    #a column vector with a new variable field, somethin like
    #Class  Breast Cancer Breast Neoplasm
    #converted to variable value
    #Class Breast Cancer
    #Class Breast Neoplasm
    #from that FacetGrid obtains col variable and kdplot obtains the value part
    g = sns.FacetGrid(dfTemp.melt(id_vars='Class'),
                      col='variable',
                      hue='Class',
                      col_wrap=5)               # change this to your liking
    #kdeplot plots the probability density of a continuous variable
    g = g.map(sns.kdeplot, "value", alpha=1, label='Data')\
         .add_legend()\
         .set_titles("{col_name}")\
         .set_axis_labels('')
```

```
def oneClassP(dfPos,dfNeg):
    dfTrain=dfNeg
    dfTest=dfPos
    dfTrain['Class']=0
    dfTest['Class']=1
    df=dfTrain.append(dfTest,ignore_index=True)

    listDrop=['lncrna','mirna','seq','seqMirna','Class']
    dfT=df.drop(listDrop,axis="columns")

    colNames=list(dfT.columns.values)
    min_max_scaler=preprocessing.MinMaxScaler()
    x_scaled=min_max_scaler.fit_transform(dfT)
    df[colNames]=x_scaled

    train, test = train_test_split(df, test_size=.2,random_state=1)
    train_normal = train[train['Class']==0]

    train_outliers = train[train['Class']==1]
    outlier_prop = len(train_outliers) / len(train_normal)
    print(outlier_prop)
    svm = OneClassSVM(kernel='rbf', nu=outlier_prop, gamma=1e-05)
    svm.fit(train_normal[['AGm','ACl','GUl']])
    plt.figure(1)
    x = test['AGm']
    y = test['ACl']
    plt.scatter(x, y,c=test['Class'])
    plt.xlabel('AGm')
    plt.ylabel('ACl')

    plt.figure(2)
    x = test['AGm']
    y = test['ACl']
    y_pred = svm.predict(test[['AGm','ACl','GUl']])
    print(y_pred)
    colors = np.array(['red', 'green'])
    plt.scatter(x, y, alpha=0.7, c=colors[(y_pred + 1) // 2])
    plt.xlabel('AGm')
    plt.ylabel('ACl')

    oneClassCM(dfTrain,dfTest)
```

```
def oneClassCM(dfTrain,dfTest):
    print(dfTrain.shape) #extract 458 for testing so to 8700
    print(dfTest.shape)  #4.9%
    listDrop=['lncrna','mirna','seq','seqMirna','Class']
    dfTrainT=dfTrain.drop(listDrop,axis="columns")
    dfTestT=dfTest.drop(listDrop,axis="columns")
    colNames=list(dfTrainT.columns.values)

    min_max_scaler=preprocessing.MinMaxScaler()
    x_scaled=min_max_scaler.fit_transform(dfTrainT)
    dfTrain[colNames]=x_scaled
    x_scaled=min_max_scaler.fit_transform(dfTestT)
    dfTest[colNames]=x_scaled
    train=dfTrain.loc[0:8700,:]
    train=train.drop(listDrop,axis="columns")
    Y_1=dfTrain.loc[8700:,"Class"]
    Y_2=dfTest['Class']

    X_test_1 = dfTrain.loc[8700:, :].drop(listDrop,axis="columns")
    X_test_2 = dfTest.drop(listDrop,axis="columns")
    X_test = X_test_1.append(X_test_2)
    svm = OneClassSVM(kernel='rbf', nu=0.05, gamma=1e-05)
    Y_test= Y_1.append(Y_2)
    X_testT=X_test.reset_index(drop=True)
    Y_testT=Y_test.reset_index(drop=True) #shuffle
    dfTemp=pd.concat([X_testT, Y_testT], axis=1)
    df= dfTemp.sample(frac=1).reset_index(drop=True)
    X_test=df.drop(['Class'],axis="columns")
    Y_test=df['Class'] #the results are the same when shuffling the test set
    svm.fit(train)
    bc = svm.predict(X_test)
    unique, counts = np.unique(bc, return_counts=True)
    print (np.asarray((unique, counts)).T)

    Y_test= Y_test.to_frame()
    Y_test=Y_test.reset_index()

    bc = pd.DataFrame(bc)
    bc= bc.rename(columns={0: 'prediction'})
```

```
360
361     TP = FN = FP = TN = 0
362     for j in range(len(Y_test)):
363         if Y_test['Class'][j]== 0 and bc['prediction'][j] == 1:
364             TP = TP+1
365         elif Y_test['Class'][j]== 0 and bc['prediction'][j] == -1:
366                 FN = FN+1
367         elif Y_test['Class'][j]== 1 and bc['prediction'][j] == 1:
368                 FP = FP+1
369         else:
370                 TN = TN +1
371     print (TP,FN,FP,TN)
372
373     accuracy = (TP+TN)/(TP+FN+FP+TN)
374     print (accuracy)
375     sensitivity = TP/(TP+FN)
376     print (sensitivity)
377     specificity = TN/(TN+FP)
378     print (specificity)
```

Appendix D – Principal functions from source code from Chapter 6

```
def align(X,Y):
    mirnab=Seq(Y)
    mirnab=mirnab.back_transcribe()
    mirnab=mirnab.complement()

    align=pairwise2.align.globalms(X,mirnab,2, -1, -.5, -.1)
    #align=pairwise2.align.globalxx(X,Y)
    #print(format_alignment(*align[0]))


    score = re.findall("[+-]?[0-9]+(?:\.[0-9]+)?", format_alignment(*align[0]))
    #print ("score is ",score[0])

    if (float(score[0])>24):
        print (align)

    return score[0]
```

```
def getMirnaSeq(file):
    mirnaList=pd.read_csv(file)
    print(mirnaList.head())
    mirnaF="mature.fa"
    rowNumber=0
    for item in mirnaList["miRNA"]:
        fastaSeq=SeqIO.parse(open(mirnaF),'fasta')

        for j in fastaSeq:
            if (j.name==item):
                seqM=j.seq
                print(item," ",seqM)
                mirnaList.loc[mirnaList.index[rowNumber], 'seqMirna'] = str(seqM)
            else:
                seqM=""
        rowNumber=rowNumber+1
    mirnaList['seqMirna'].replace('', np.nan, inplace=True)
    mirnaList.dropna(subset=['seqMirna'], inplace=True)
    mirnaList.to_csv("miRNAseq.csv",index=False)

def countKMers(seq,k):
    counts={}
    numKmers=len(seq)-k+1
    for i in range(numKmers):
        kmer=seq[i:i+k]
        if kmer not in counts:
            counts[kmer]=1
        else:
            counts[kmer]=counts[kmer]+1
    return counts
```

```
def mers(mrna):
    mirna=Seq(mrna)

    kmers=[]
    ckmers=[]
    dicKmers=countKMers(str(mirna).upper(),3)
    probK=[''.join(c) for c in product('AUCG', repeat=3)]

    for i in probK:
        kmers.append(i)
        ckmers.append(0)
    for k,v in dicKmers.items():

        ckmers[kmers.index(k)]=v
    return ckmers
```

```
def addFeatures(f):
    #add alignment 5utr, 3utr and s
    file5="5utrw.fasta"
    file3="3utrw.fasta"
    fileS="sw.fasta"

    kmersT=[]
    kmersC=[]
    kmersL=[]
    kmersP=[''.join(c)+'m' for c in product('AUCG', repeat=3)]
    kmersP1=[''.join(c)+'l' for c in product('AUCG', repeat=3)]

    kmersT=kmersP+kmersP1

    dataC=pd.read_csv(f)
    fastaSeq=SeqIO.parse(open(file5),'fasta')
    for item in fastaSeq:
        seq5Utr=item.seq

    alignS=0
    rowNumber=0
    dupScore=0
    i=0
    for seqM in dataC["seqMirna"]:
        alignS=align(seq5Utr,seqM)
        dupScore=(dupRna(seq5Utr,seqM))
        dataC.loc[dataC.index[rowNumber],'alignS']=alignS
        dataC.loc[dataC.index[rowNumber],'duplex']=dupScore
        kmersC=mers(seqM)
        j=0
        for item in kmersP:
            dataC.loc[dataC.index[i], item] = kmersC[j]/len(seqM)
            j=j+1
        rowNumber=rowNumber+1
        i=i+1

    dataC.to_csv("test5.csv",index=False)
```

```
def oneClass(df):
    listDrop=['miRNA','miRTarBase ID','Species (miRNA)','Target Gene','Species (Target Gene)','Target Gene (Entrez Gene ID)','Species (Target Gene)',
    dfTrainT=df.drop(listDrop,axis="columns")
    colNames=list(dfTrainT.columns.values)
    #print(colNames)
    min_max_scaler=preprocessing.MinMaxScaler()
    x_scaled=min_max_scaler.fit_transform(dfTrainT)
    dfTrainT[colNames]=x_scaled

    x_reduced = TSNE(n_components=2, random_state=0).fit_transform(dfTrainT)

    svm = OneClassSVM(kernel='rbf', nu=0.10, gamma=1e-05)
    svm.fit(x_reduced)

    predMirna=svm.predict(x_reduced)
    x_min, x_max = x_reduced[:, 0].min() - 10, x_reduced[:, 0].max() + 10
    y_min, y_max = x_reduced[:, 1].min() - 10, x_reduced[:, 1].max() + 10

    x_ = np.linspace(x_min, x_max, 500)
    y_ = np.linspace(y_min, y_max, 500)

    xx, yy = np.meshgrid(x_, y_)

    z = svm.decision_function(np.c_[xx.ravel(), yy.ravel()])
    z = z.reshape(xx.shape)

    plt.contourf(xx, yy, z, cmap=plt.cm.PuBu)
    plt.scatter(x_reduced[predMirna == 1, 0], x_reduced[predMirna == 1, 1], c='yellow', edgecolors='k', label='Positive miRNAs')
    plt.scatter(x_reduced[predMirna == -1, 0], x_reduced[predMirna == -1, 1], c='blueviolet', edgecolors='k', label='Negative miRNAs')
    plt.legend(loc=2)
    plt.axis('tight')
    plt.show()
    print ("One class SVM predictions")
    print(predMirna)
    print(len(predMirna))
    listA=[i+1 for i,x in enumerate(predMirna) if x == -1]
    listB=[i+1 for i,x in enumerate(predMirna) if x == 1]
    print(len(listA))
    print(len(listB))
```

```
def oneClassTwo(dfmirNeg,dfmirPos):
    dfTrain=dfmirPos
    dfTest=dfmirNeg.sample(frac=0.10)
    dfTest=dfmirNeg
    dfTrain['Class']=0
    dfTest['Class']=1
    listDrop=['miRNA','miRTarBase ID','Species (miRNA)','Target Gene','Species (Target Gene)','Target Gene (Entrez Gene ID)','Species (Target Gene)'
    df=dfTrain.append(dfTest,ignore_index=True)
    dfT=df.drop(listDrop,axis="columns")
    colNames=list(dfT.columns.values)
    min_max_scaler=preprocessing.MinMaxScaler()
    x_scaled=min_max_scaler.fit_transform(dfT)
    dfT[colNames]=x_scaled

    sacc=[]
    ssens=[]
    sspec=[]
    for i in range(0,10):
        #shoulc only pass the 5% for avoiding the disbalance, this for the one class svm model
        print ("train quantity ",dfTrain.shape)
        dfTest=dfmirNeg.sample(frac=0.05)
        print ("test quantity ",dfTest.shape)
        acc,sens,spec=oneClassCM(dfTrain,dfTest)
        print(acc," ",sens," ",spec)
        sacc.append(acc)
        ssens.append(sens)
        sspec.append(spec)
    print(sacc)
    print(ssens)
    print ("accuracy average ",statistics.mean(sacc)," ",statistics.stdev(sacc))
    print ("sensitivity average ",statistics.mean(ssens)," ",statistics.stdev(ssens))
    print ("specificity average ",statistics.mean(sspec)," ",statistics.stdev(sspec))
```

```
def oneClassGS(df):
    listDrop=['Class']
    y=df['Class'].astype(int)

    df=df.drop(listDrop,axis="columns")
    kernels=["rbf","sigmoid"]
    nus=np.linspace(0.01,1,10)
    #gammas=[1e-1, 1e-2, 1e-3, 1e-4,1e1,1e2,1e3]
    gammas=np.linspace(10e-6,1,30)
    #degrees = [1, 2, 3, 4 ,5, 6]
    scoring = make_scorer(metrics.f1_score,
                          average="weighted")
    df= df.sample(frac=1).reset_index(drop=True)
    listDrop=['Class']
    X=df
    oneClass=OneClassSVM(kernel='rbf')
    oneClass=OneClassSVM()
    print(oneClass.get_params().keys())
    grid=GridSearchCV(estimator=oneClass,param_grid=dict(kernel=kernels,nu=nus,gamma=gammas),scoring=scoring,n_jobs=-1,verbose=5,cv=5)
    grid.fit(X,y)
    print("best score ",grid.best_score_)
    print ("best parameters ",grid.best_params_)
```

```
def svm(dfNeg,dfPos):
    dfTrain=dfPos
    dfTest=dfNeg
    dfTestr=dfTest.sample(frac=0.10) #uncomment for all the data
    dfTrain['Class']=0  #it was 1
    dfTestr['Class']=1  #it was 2
    df=dfTrain.append(dfTestr)
    print("train dimension ",dfTrain.shape)
    print("test dimension ",dfTestr.shape)
    df= df.sample(frac=1).reset_index(drop=True)
    df.to_csv("check.csv")
    data=df
    listDrop=['miRNA','miRTarBase ID','Species (miRNA)','Target Gene','Species (Target Gene)','Target Gene (Entrez Gene ID)','Species (Target Gene)'
    y=data['Class']
    X=data.drop(listDrop,axis="columns")
    data2=X
    colNames=list(data2.columns.values)
    min_max_scaler=preprocessing.MinMaxScaler()
    x_scaled=min_max_scaler.fit_transform(data2)
    data[colNames]=x_scaled
    X=data.drop(listDrop,axis="columns")
    kernels=["rbf","poly"]
    Cs=[1e-2, 1e-1, 1, 1e1, 1e2]
    degrees = [2, 3, 4, 5]
    # gammas=[1e-1, 1e-2, 1e-3]
    gammas=np.linspace(10e-6,1,30)
    #degrees = [2, 3, 4]
    model=SVC(kernel='rbf')
    scoring = make_scorer(metrics.f1_score,
                          average="weighted")
    #test with cv
    #should I not keep the original train set somewhere?
    scoring = {'accuracy':make_scorer(accuracy_score),
               'f1-score':make_scorer(metrics.f1_score,average="weighted")
               }
    #model=SVC(kernel='rbf',C=100,gamma=0.01,random_state=4)
    model=SVC(kernel='rbf',C=0.01,gamma=1e-05,random_state=4)
    cv=cross_validate(model, X, y, scoring=scoring, cv = 5)
    print(cv,"\n")
    print ("TRAIN ACC ",cv['train_accuracy']," accuracy ",cv['train_accuracy'].mean()," std ",cv['train_accuracy'].std()*2,"\n")
    print ("TEST ACC ",cv['test_accuracy']," accuracy ",cv['test_accuracy'].mean()," std ",cv['test_accuracy'].std()*2)

    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
    model.fit(X_train,y_train)
```

```
    y_train_pred = model.decision_function(X_train)
    y_test_pred = model.decision_function(X_test)
    #print(y_train)
    train_fpr, train_tpr, tr_thresholds = roc_curve(y_train, y_train_pred)
    test_fpr, test_tpr, te_thresholds = roc_curve(y_test, y_test_pred)

    plt.grid()

    plt.plot(train_fpr, train_tpr, label=" AUC TRAIN ="+str(auc(train_fpr, train_tpr)))
    plt.plot(test_fpr, test_tpr, label=" AUC TEST ="+str(auc(test_fpr, test_tpr)))
    plt.plot([0,1],[0,1],'g--')
    plt.legend()
    plt.xlabel("True Positive Rate")
    plt.ylabel("False Positive Rate")
    plt.title("AUC(ROC curve)")
    plt.grid(color='black', linestyle='-', linewidth=0.5)
    plt.show()

    y_pred=model.predict(X_test)
    cm=confusion_matrix(y_test,y_pred)
    print(cm)
```

```
def rf(dfNeg,dfPos):
    dfTrain=dfPos
    dfTest=dfNeg
    dfTestr=dfTest.sample(frac=0.10)
    dfTrain['Class']=1
    dfTestr['Class']=2
    df=dfTrain.append(dfTestr)
    df= df.sample(frac=1).reset_index(drop=True)
    data=df
    listDrop=['miRNA','miRTarBase ID','Species (miRNA)','Target Gene','Species (Target Gene)','Target Gene (Entrez Gene ID)','Species
    y=data['Class']
    colNames=list(data.columns.values)
    colNamesF=list(set(colNames)-set(listDrop))
    min_max_scaler=preprocessing.MinMaxScaler()
    data1=data.drop(listDrop,axis="columns")
    x_scaled=min_max_scaler.fit_transform(data1)
    data[colNamesF]=x_scaled
    data.to_csv("randomForestScaled.csv")
    X=data.drop(listDrop,axis="columns")
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)

    """model=RandomForestClassifier()
    param_grid = {
            'n_estimators': [10,50,100,150,200,300],
            'max_features': ['auto', 'sqrt', 'log2'],
            'max_depth' : [4,8,12,16],
            'criterion' :['gini', 'entropy']
    }

    scoring = make_scorer(metrics.f1_score,
                      average="weighted")

    grid=GridSearchCV(estimator=model,param_grid=param_grid,
                       scoring=scoring,n_jobs=-1,verbose=5,cv=10)

    grid.fit(X,y)
    print("best score ",grid.best_score_)
    print ("best parameters ",grid.best_params_)

    scoring = {'accuracy':make_scorer(accuracy_score),
               'f1-score':make_scorer(metrics.f1_score,average="weighted")
               }"""
```

```
    #best parameters  {'criterion': 'entropy', 'max_depth': 12, 'max_features': 'auto', 'n_estimators': 10}
    model=RandomForestClassifier(criterion= 'entropy', max_depth= 12, max_features= 'auto', n_estimators= 10)
    model.fit(X_train,y_train)
    pred=model.predict(X_test)
    score=accuracy_score(y_test,pred)
    print(score)
    conf_mat = confusion_matrix(y_test,pred)
    print(conf_mat)
```