UNIVERSITÀ DEGLI STUDI DI CAMERINO

School of Advanced Studies

DOCTORAL COURSE IN

Life and Health Science – Data Science

XXXIII cycle

MACHINE LEARNING IN CLINICAL BIOLOGY AND MEDICINE: FROM PREDICTION OF MULTIDRUG RESISTANT INFECTIONS IN HUMANS TO PRE-MRNA SPLICING CONTROL IN CILIATES

PhD Student
Dr. Alessio Mancini

Supervisor
Prof. Sandra Pucciarelli

SUMMARY

List of figures

List of tables

List of abbreviations

AS - Alternative Splicing

AUC - Area Under the ROC Curve

ACC - Accuracy

CSIs - Constitutively Spliced Introns

DSaaS – Data Science as a Service

FP - False Positives

FPR - False Positive Rate

FN - False Negatives

GA – Genetic Algorithms

GO - Gene Ontology

IR - Intron Retention

MAC - Macronucleus

MCC - Matthew Correlation Coefficient

MDR – Multi Drug Resistant

MIC - The Micronucleus

ML - Machine Learning

NGS - Next Generation Sequencing

NMD - Nonsense-Mediated mRNA Decay

NNs - Neural Networks

REST – REpresentational State Transfer

Ris - Retained Introns

ROC - Receiver Operating Characteristic Curve

SVM - Support Vector Machine

TN - True Negatives

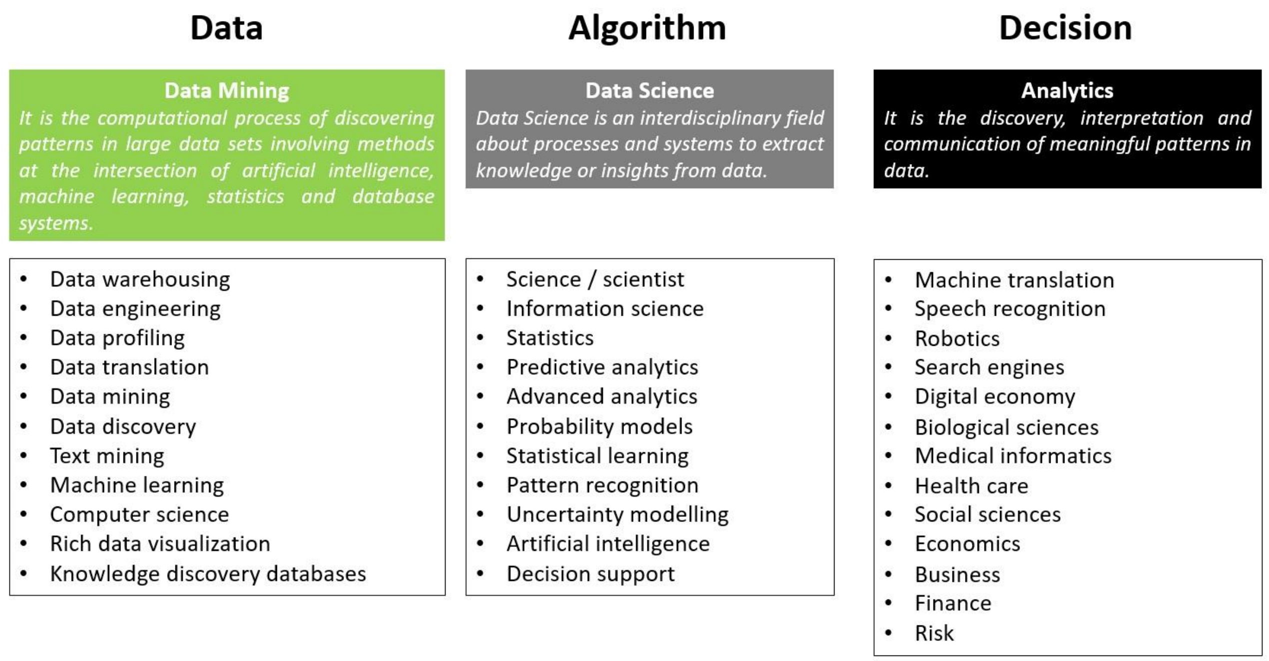
TP - True Positives

TPR - True Positive Rate

CHAPTER 1: INTRODUCTION

1.1 Data Science

Data Science is an interdisciplinary field that applies statistics and data science tools to analyze and interpret the data generated by modern technologies. Data Science allows us to better understand biological systems, and to leverage genomic technologies to benefit science, medicine, society and the economy.



Data

Data Mining

It is the computational process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics and database systems.

- Data warehousing
- Data engineering
- Data profiling
- Data translation
- Data mining
- Data discovery
- Text mining
- Machine learning
- Computer science
- Rich data visualization
- Knowledge discovery databases

Algorithm

Data Science

Data Science is an interdisciplinary field about processes and systems to extract knowledge or insights from data.

- Science / scientist
- Information science
- Statistics
- Predictive analytics
- Advanced analytics
- Probability models
- Statistical learning
- Pattern recognition
- Uncertainty modelling
- Artificial intelligence
- Decision support

Decision

Analytics

It is the discovery, interpretation and communication of meaningful patterns in data.

- Machine translation
- Speech recognition
- Robotics
- Search engines
- Digital economy
- Biological sciences
- Medical informatics
- Health care
- Social sciences
- Economics
- Business
- Finance
- Risk

Fig.1: *Data science as interdisciplinary field*[1]

This field include data mining, statistics, machine learning, analytics, and programming (Fig.1).

- Data mining applies algorithms to reveal patterns in complex datasets then used to extract new knowledge from the set. Knowledge discovery in databases is a field encompassing theories, methods and techniques, trying to make sense of data and extract useful knowledge from them. It is considered to be a multi-step process (selection, preprocess, transformation, interpretation and evaluation)[2].

- Statistical measures use this extracted data to gauge events that are likely to happen in the future based on what the data shows happened in the past.

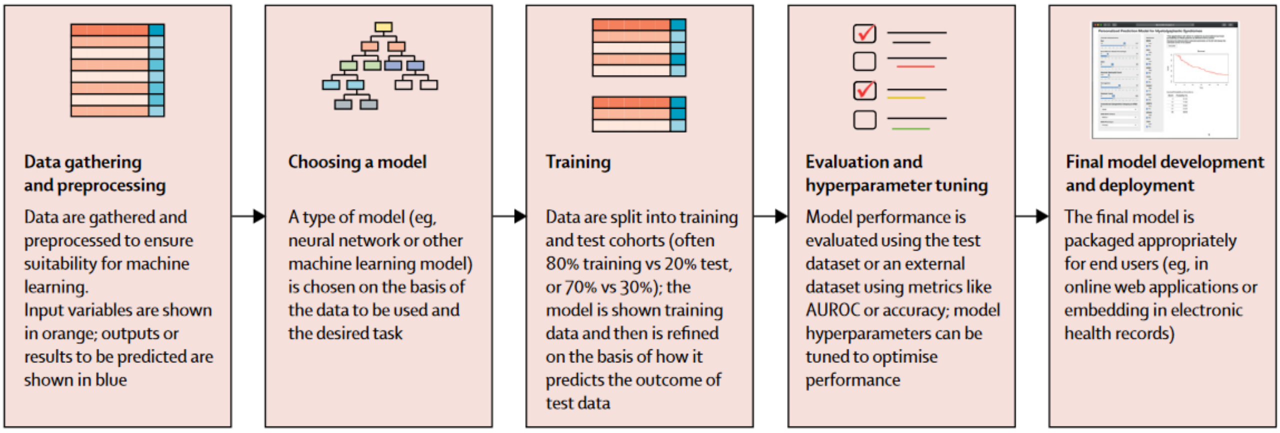
- Machine learning is an artificial intelligence tool that processes mass quantities of data that a human would be unable to process in a lifetime. Machine learning perfects the decision model presented under predictive analytics by matching the likelihood of an event happening to what actually happened at a predicted time.

- Using analytics, the data analyst collects and processes the structured data from the machine learning stage using algorithms. The analyst interprets, converts, and summarizes the data into a cohesive language that the decision-making team can understand.

Data science can be applied to practically any contexts and, as the data scientist's role evolves, the field will expand to encompass data architecture, data engineering, and data administration.

1.2 Machine learning

Machine learning (ML) is the scientific field dealing with the ways in which machines learn from experience. ML algorithms uses variables called “features” to learn, predict and build a model. A feature is an individual measurable property or characteristic of a phenomenon[3]. Choosing informative, discriminating and independent features is a crucial element of effective algorithms in pattern recognition, classification and regression. Features are usually numeric, but structural features such as strings and graphs are used in syntactic pattern recognition. For many scientists, the term “machine learning” is identical to the term “artificial intelligence”, given that the possibility of learning is the main characteristic of an entity called intelligent in the broadest sense of the word. The purpose of machine learning is creating an algorithm that can adapt and learn from their experience[4]. ML is already applied to great effect in diverse biological fields, such as the protein secondary structure prediction[5], the mechanisms of splicing[6] and the onset prediction of diseases[7].



Data gathering and preprocessing

Data are gathered and preprocessed to ensure suitability for machine learning.
Input variables are shown in orange; outputs or results to be predicted are shown in blue

Choosing a model

A type of model (eg, neural network or other machine learning model) is chosen on the basis of the data to be used and the desired task

Training

Data are split into training and test cohorts (often 80% training vs 20% test, or 70% vs 30%); the model is shown training data and then is refined on the basis of how it predicts the outcome of test data

Evaluation and hyperparameter tuning

Model performance is evaluated using the test dataset or an external dataset using metrics like AUROC or accuracy; model hyperparameters can be tuned to optimise performance

Final model development and deployment

The final model is packaged appropriately for end users (eg, in online web applications or embedding in electronic health records)

Fig.2: *Example of machine learning model development (supervised learning approach* [8]

ML tasks are typically classified into three broad categories[7]:

a) Supervised learning, in which the system infers a function from labeled training data.

b) Unsupervised learning, in which the learning system tries to infer the structure of unlabeled data.

c) Reinforcement learning, in which the system interacts with a dynamic environment.

A typical supervised ML pipeline is shown in figure 2.

1.3 Aim of the Thesis

Machine Learning methods have broadly begun to infiltrate the clinical literature in such a way that the correct use of algorithms and tools can facilitate both diagnosis and therapies. The availability of large quantities of high-quality data could lead to an improved understanding of risk factors in community and healthcare-acquired infections. In the first part of my PhD program, I refined my skills in Machine Learning by developing and evaluate with a real antibiotic stewardship dataset, a model useful to predict multi-drugs resistant urinary tract infections after patient hospitalization[9]. For this purpose, I created an online platform called *DSaaS* specifically designed for healthcare operators to train ML models (supervised learning algorithms). These results are reported in Chapter 2.

In the second part of the *PhD* thesis (Chapter 3) I used my new skills to study the genomic variants, in particular the phenomenon of intron splicing. One of the important modes of pre-mRNA post-transcriptional modification is alternative intron splicing, that includes intron retention (unsplicing), allowing the creation of many distinct mature mRNA transcripts from a single gene. An accurate interpretation of genomic variants is the backbone of genomic medicine. Determining for example the causative variant in patients with Mendelian disorders facilitates both management and potential downstream treatment of the patient's condition, as well as providing peace of mind and allowing more effective counselling for the wider family.

Recent years have seen a surge in bioinformatics tools designed to predict variant impact on splicing, and these offer an opportunity to circumvent many limitations of RNA-seq based approaches. An increasing number of these tools rely on machine learning computational approaches that can identify patterns in data and use this knowledge to speculate on new data.

I optimized a pipeline to extract and classify introns from genomes and transcriptomes and I classified them into retained (Ris) and constitutively spliced (CSIs) introns. I used data from ciliates for the peculiar organization of their genomes (enriched of coding sequences) and because they are unicellular organisms without cells differentiated into tissues. That made easier the identification and the manipulation of introns. In collaboration with the PhD colleague dr. Leonardo Vito, I analyzed these intronic sequences in order to identify "features" to predict and to classify them by Machine Learning algorithms. We also developed a platform useful to manipulate FASTA, gtf, BED, etc. files produced by the pipeline tools. I named the platform: Biounicam (intron extraction tools) available at http://46.23.201.244:1880/ui.

The major objective of this study was to develop an accurate machine-learning model that can predict whether an intron will be retained or not, to understand the key-features involved in the intron retention mechanism, and provide insight on the factors that drive IR. Once the model has been developed, the final step of my PhD work will be to expand the platform with different machine learning algorithms to better predict the retention and to test new features that drive this phenomenon. These features hopefully will contribute to find new mechanisms that controls intron splicing.

The other additional papers and patents I published during my PhD program are in Appendix B and C. These works have enriched me with many useful techniques for future works and ranged from microbiology to classical statistics.

CHAPTER 2: Machine learning models predicting multidrug resistant urinary tract infections using *"DsaaS"*

DOI: 10.1186/s12859-020-03566-7

2.1 Machine Learning in Healthcare systems

Increasingly, healthcare operators must process and interpret large amounts of complex data. Data science applications regard the extraction of knowledge from information, more than simply mining massive data sets. Machine learning (ML) methods have broadly begun to infiltrate the clinical literature and the right use of algorithms and tools can facilitate both diagnosis and therapies. The availability of large quantities of high-quality data could lead to an improved understanding of risk factors in community and healthcare-acquired infections. For instance, in the antibiotic stewardship field researchers utilized Massachusetts statewide antibiogram data to predict three future years of antibiotic susceptibilities using ML regression-based strategies [10]. International guidelines recommend to use institutional antibiograms in the development of empiric antibiotic therapies [11].

ML methods could help physicians in the empirical treatment of the urinary tract infections (UTIs). These are usually known as the most common bacterial infections with a significant financial burden on society [12]. In hospitals at least 40% of all infections are UTIs and bacteriuria develops in up to 25% of patients who require a urinary catheter for one week or more [13]. The selection of adequate treatment for the management of UTIs is increasingly challenging due to their etiology, bacterial resistance profile and evolving of adaptive strategies. Moreover, the bacteria resistance to antibiotics has risen dramatically with few therapeutic options and one of the causes is the recurrent infection that leads to development of multidrug resistance (MDR). Several risk factors are associated with UTIs, including gender and age [14]. Male patients have a lower risk of contracting uncomplicated UTIs but more prominent to contract complicated or MDR infections than women. Older adults are more prone than younger individuals in developing urinary tract infections because of incomplete bladder emptying (often related to prostatic enlargement in men), higher rate of catheter use and increased susceptibility to infection associated with frailty [15]. Moreover, infections caused by MDR organisms are more common in older adults, especially those with catheters or residing in long-term care. The resistance rates to antimicrobials in UTIs can differ from region to region, patient to patient and even from ward to ward where the patient is hospitalized. Hence, in a nosocomial infection it is therefore important to know the microorganism population in the hospitalization place [16].

Unfortunately, antibiotics are not always prescribed responsibly contributing to the development of new resistances [17]. To effectively treat patients and prevent the increases in resistance, every institution must have an up-to-date susceptibility knowledge and predictions can be used to guide prescription practices and prepare for future resistance threats [10]. To reach this goal, the literature offers a huge number of ML tools requiring minimum training in computer sciences and basic programming knowledge. These skills, obvious for researchers, are often missing in routine healthcare operators. Moreover, most of these tools require the installation, in institutional PCs, of dedicated applications that make the process even slower and less attractive for the end-user.

The first objective of this work is to design, develop and evaluate, with a real antibiotic stewardship dataset, a user-friendly, online and completely dynamic tool to train predictive ML models (supervised learning algorithms) to be applied in this field. Future works will focus on enriching *DSaaS* with additional algorithms-analysis packages to make the platform able to operate large amounts of data both from a computational and storage point of view and creating a platform useful to users to easily carry out the complete data science work pipeline.

2.2 Methods

2.2.1 *DSaaS* Platform Architecture

DSaaS (Data Science as a Service) is built on a multi-tier architecture. The front-End provides a "notebook" interface where users can interact with the platform creating interactive data science experiments. The notebook front end includes many menus, graphical tools and an assistant system that may help users driving their experiments, reading and sharing data science experiments and exchanging data set with other systems.

At the same level, *DSaaS* provides specific API, basically REST-services (Representational State Transfer) useful to integrate third party applications.

From a Business-Logic level, *DSaaS* is an engine performing data analysis as well as dataflow execution.

This level also leans on other external systems such as:

- *h2o.ai platforms* [18]: to improve machine learning models' performance on big data
- *Fluentd* [19]: for capturing and collecting information from application log file
- *Apache Flink* [20]: as a processing engine for stateful computations over bounded and unbounded data streams
- *Apache Giraph* [21]: for improving graph processing algorithms (e.g. topological data analysis)
- *Apache NiFi* [22]: for data routing, transformation, and system mediation logic
- *Active MQ* [23]: as a broker for time consuming process and asynchrony communication
- *Apache Spark* [24]: running of existing Hadoop Distributed File System HDFS [25] infrastructure provides several features like Spark SQL for query distributed data set. Spark will be able to replace the Hadoop layer and provide fast and real-time processing of massive data

Back-End level represents data storage using distributed systems such as Hadoop and Hive in a transparent way. We have currently implemented a plain *DSaaS* prototype to evaluate its effectiveness using a real database about antibiotic stewardship.

To date, *DSaaS* allows to validate simple data science models based on regression algorithms (e.g., Linear Regression, Polynomial Regression and Support Vector Regression) as well as supervised classification techniques (e.g., Support Vector Machines, Catboost, Neural Network). Moreover, *DSaaS* may be used to create and execute data science processes using an easy dataflow editor and allows to publish the results obtained as a REST services. These last features (i.e., dataflow editor and unsupervised ML algorithms) will be released in the next iteration of the platform.

We are also planning to provide *DSaaS* with a "Stewardship UI" that will help users to maintaining data quality within *DSaaS* platform [26].

From a technical point of view, *DSaaS* prototype was developed integrating R with h2o.ai and using the shiny package for the realization of the GUI and was used as working tool for the problem described above.

2.2.2 Experimental setup

The aim of this work was to build a ML model useful to predict the patient-related risk, after the hospitalization, to acquire an MDR UTI.

The dataset was built out based on the bacterial isolates reports of a hospital located in Central Italy with 288 beds and a mean of 31,000 inpatient days per semester. All patients admitted from March 2011 to march 2018 (14 six-months periods) were included in the study. Only isolates collected from infections that occurred 48 hours after admission were used and identified as nosocomial infections, as defined by the Centers for disease Control and Prevention[27].

We considered as MDR UTI a patient with a microorganism resistant to one or more antibiotic classes as defined from the CDC [28] and we assigned the value R (e.g. 1) to all MDR UTI and the value S (e.g. 0) to the rest.

We collected results from 11 wards, defined as a spatial unit provided with rooms where a unique staff of health-care and co-workers are active. In our model we considered the variable "ward" as a space subjected to few interactions with the others. Therefore, the microbial population within a ward with their related hospital infections and antibiotic resistance profiles were preserved for each ward and time-dependent.

To test the *DSaaS* platform we decided to restrict the database and to use only the urine samples, corresponding to the most commonly requested clinical test among wards. The selection of four predictors (time-period, sex, age class and ward) was primarily based on urinary infection related literature [29]. Table 1 shows the detailed operational definition of variables used in our study.

Variables		**Measurements**	**Definition**
Dependent	MDR Resistance	Discrete	Does the patient acquire a MDR infection during hospitalization? Yes or No
Independent	Gender	Discrete	Gender of the patients, Male or Female.
	Age	Continous	Age (in years) during hospitalization
	Age Class	Discrete	10 years class to witch the patient belong, from 1 to 10
	Ward	Discrete	Ward where the patient was hospitalized, from 1 to 11
	Time Period	Discrete	Time period in witch the patient was hospitalized in a ward, from 1 to 14

Table.1: *Operational definition of variables*

A total of 1486 clinical samples were considered for this study. Specimens were processed according to good laboratory practice and standard methods for identification. Duplicate data were discarded using the Bio-Mérieux VIGIguard™

software if all the following conditions were true: isolate collected from the same patient, same specimen, same ward, same species and similar antibiotic pattern (S/R=1; I/R–S/I=2) within 20 days.

DSaaS adopted the Caret v6.0–82 [30] and the GA (genetic algorithms optimization) v3.2 package [31] to automatically tune the optimal combinations of model parameters for the three ML algorithms aiming to achieve a better prediction performance. Evidence demonstrated that the class imbalance (i.e., unequal size of the dependent variable), which is just the situation in our sample, can substantially impact the performance of the method used. Therefore, we adopted synthetic minority over-sampling technique by under-sampling the adequate class and over-sampling the inadequate class to improve the model performance[32]. *DSaaS* did also automatically a 10-fold cross validation method with three repeats, which has been viewed as the de facto standard for estimating model performance [33].

Furthermore, *DSaaS* allows the user to divide the database in a training set (70% of all data points), and a test set (30%) to evaluate the predictive models. The data point splitting was made by assigning random values to the test set. The training set were used to build the classification algorithms using gradient boosting Catboost [34], Neural Networks[35] and SVM[36].

2.2.3 Performance measures

DSaaS allowed us to measure the model's performance with accuracy, AUC, sensitivity and specificity. To describe such performance measures for classification problem, it is essential to define a specific matrix, called confusion matrix, containing the number of false positives (FP), false negatives (FN), true positives (TP), and true negatives (TN). Specifically, a two-class (positive-negative) confusion matrix is a table where each row represents a predicted value and each column defines an actual value (or vice-versa): all correct prediction (TP and TN) are located in the matrix diagonal while the errors are given by all the elements outside the diagonal.

Accuracy (ACC) [37] is a value that can be directly calculated from the confusion matrix and defines how often the classifier is correct and is calculated as the ration between the number of correct predictions and the total number of predictions.

To define AUC [37] it is necessary to introduce the ROC curve (Receiver Operating Characteristic curve), namely a graph showing the performance of the classifier over all possible thresholds with respect to two parameters: the sensitivity also known as recall or true positive rate (TPR) and the false positive rate (FPR).

Sensitivity [37] is calculated as the ratio between the number of positive inputs correctly classified as positive (true positives) and the total number of positive data and measures how well the classifier made positive predictions based on all classes (i.e., it can be seen as the classifier ability to correctly detect positive inputs). FPR is calculated as the ratio between the number of negative inputs wrongly classified as positive (false positive) and the total number of negative data and measures the proportion of all the negative inputs who will be identified as positive.

AUC (Area Under the ROC Curve) measures the area underneath the ROC curve: it has a range of values from 0 to 1. The area measures discrimination, that is, the ability to correctly classify random positive and negative data.

Specificity [37] also known as true negative rate (TNR) is defined as the ratio between the number of negative inputs correctly classified as negative (true negatives) and the total number of negative data and measures how well the classifier made negative predictions based on all classes (i.e., it can be seen as the classifier ability to correctly detect negative inputs).

Finally, overall model performance was calculated by averaging model performances each time[37].

2.3 Results and Discussion

We created a cloud platform called *DSaaS* (figure 3) that allows both testing of data science models and the creation of rough but useful ML processes easily usable by non-expert users. A demo demonstrator of *DSaaS* can be found at: https://dsaas-demo.shinyapps.io/Server/ and its actual and future architecture is shown in figure 4.

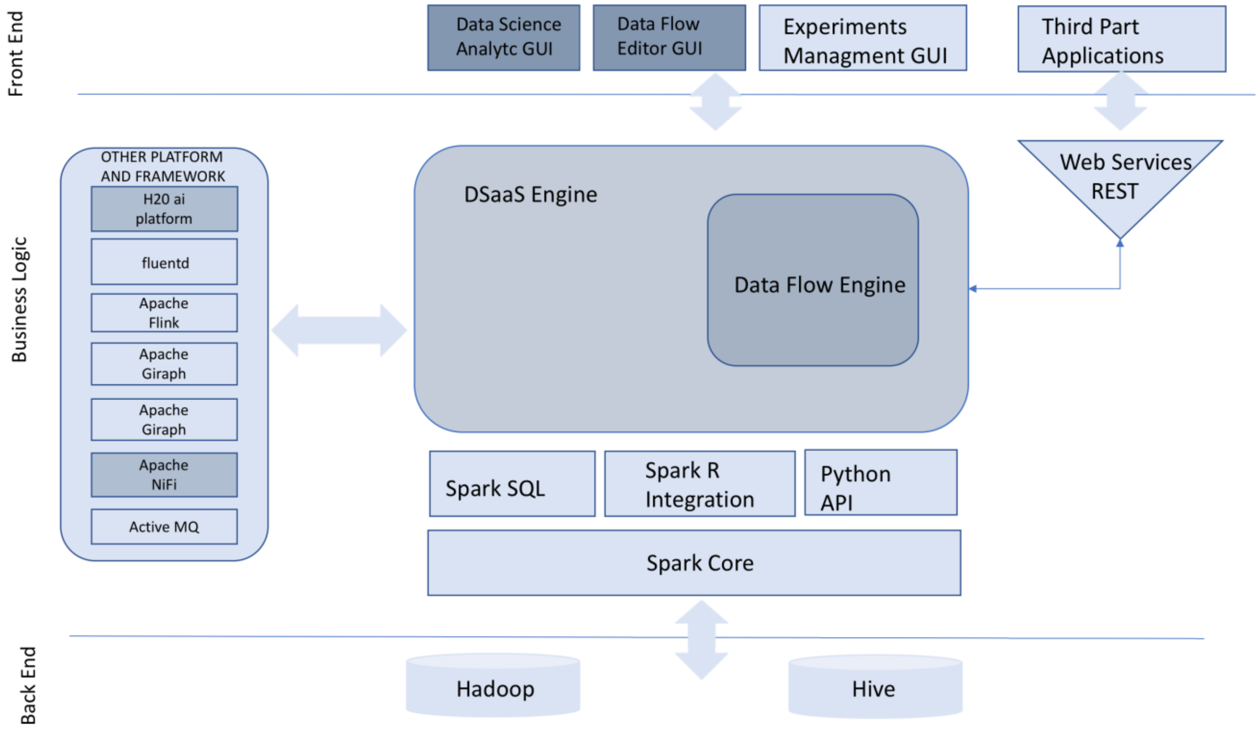
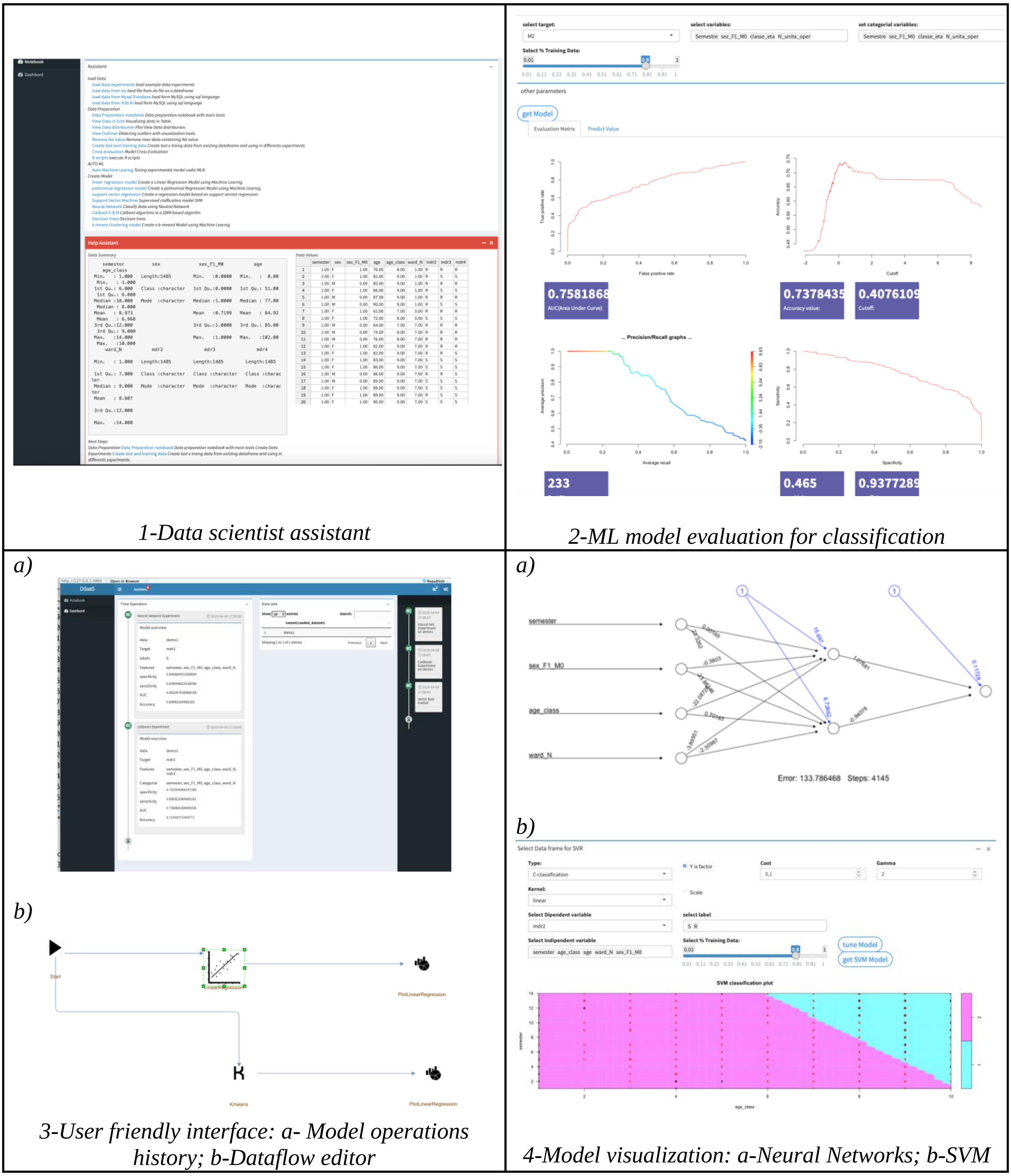


Fig.3: *The DSaaS platform*

To test the platform, we used a dataset based on antibiotic resistance information obtained from a tertiary hospital in central Italy. Several supervised learning algorithms, readily available in the platform, have been used to make antibiotic resistance predictions about MDR UTIs and results were subsequently compared to obtain the best model possible to predict further resistance outcome.



Assistent

load Data
- load data experiments *load example data experiments*
- load data from xls *load file from xls file as a dataframe*
- load data from Mysql Database *load form MySQL using sql language*
- load data from H20.AI *load form MySQL using sql language*

Data Preparation
- Data Preparetion notebook *Data preparetion notebook with main tools*
- View Data in Grid *Visualizing data in Table.*
- View Data distriburion *Plot View Data distriburion.*
- View Outliner *Detecting outliers with visualization tools.*
- Remove Na Value *Remove rows data containing NA value*
- Create test and training data *Create test e traing data from existing datafeame and using in differents experiments*
- Cross evaluation *Model Cross Evaluation*
- R scripts *execute R scripts*

AUTO ML
- Auto Machine Learing *Tuning experimental model usdin MLR.*

Create Model
- linear regression model *Create a Linear Regression Model using Machine Learnig.*
- polinomial regression model *Create a polinomial Regression Model using Machine Learnig.*
- support vector regression *Create a regression model based on support verctor regression.*
- Support Vector Machine *Supervised claffication model SVM*
- Neural Network *Classify data using Neutral Network*
- Catboot G B M *Catboot algoritms is a GBM based algoritm*
- Decision trees *Decision trees*
- k means clustering model *Create a k-meand Model using Machine Learnig.*

Help Assistant

Data Summary

```
   semester         sex               sex_F1_M0         age
   age_class
 Min.   : 1.000   Length:1485       Min.   :0.0000   Min.   :  0.00
  Min.   : 1.000
 1st Qu.: 6.000   Class :character  1st Qu.:0.0000   1st Qu.: 51.00
  1st Qu.: 6.000
 Median :10.000   Mode  :character  Median :1.0000   Median : 77.00
  Median : 8.000
 Mean   : 8.973                     Mean   :0.7199   Mean   : 64.92
  Mean   : 6.968
 3rd Qu.:12.000                     3rd Qu.:1.0000   3rd Qu.: 85.00
  3rd Qu.: 9.000
 Max.   :14.000                     Max.   :1.0000   Max.   :102.00
  Max.   :10.000
    ward_N          mdr2              mdr3              mdr4

 Min.   : 1.000   Length:1485       Length:1485       Length:1485

 1st Qu.: 7.000   Class :character  Class :character  Class :charac
ter
 Median : 9.000   Mode  :character  Mode  :character  Mode  :charac
ter
 Mean   : 8.607

 3rd Qu.:12.000

 Max.   :14.000
```

Data Values

	semester	sex	sex_F1_M0	age	age_class	ward_N	mdr2	mdr3	mdr4
1	1.00	F	1.00	76.00	8.00	1.00	R	R	R
2	1.00	F	1.00	82.00	9.00	1.00	R	S	S
3	1.00	M	0.00	85.00	9.00	1.00	R	R	R
4	1.00	F	1.00	86.00	9.00	1.00	R	S	S
5	1.00	M	0.00	87.00	9.00	1.00	R	R	R
6	1.00	M	0.00	90.00	9.00	1.00	R	S	S
7	1.00	F	1.00	62.00	7.00	3.00	R	R	R
8	1.00	F	1.00	72.00	8.00	3.00	S	S	S
9	1.00	M	0.00	64.00	7.00	7.00	R	R	R
10	1.00	M	0.00	74.00	8.00	7.00	R	R	R
11	1.00	M	0.00	76.00	8.00	7.00	R	R	R
12	1.00	F	1.00	82.00	9.00	7.00	R	R	R
13	1.00	F	1.00	82.00	9.00	7.00	R	R	S
14	1.00	F	1.00	83.00	9.00	7.00	S	S	S
15	1.00	F	1.00	86.00	9.00	7.00	S	S	S
16	1.00	M	0.00	86.00	9.00	7.00	R	R	S
17	1.00	M	0.00	89.00	9.00	7.00	S	S	S
18	1.00	F	1.00	89.00	9.00	7.00	S	S	S
19	1.00	F	1.00	89.00	9.00	7.00	R	S	S
20	1.00	F	1.00	90.00	9.00	7.00	S	S	S

Next Steps
Data Preparation Data Preparetion notebook *Data preparetion notebook with main tools Create Data Experiments* Create test and training data *Create test e traing data from existing dataframe and using in differents experiments*

1-Data scientist assistant

select target: M2

select variables: Semestre sez_F1_M0 classe_eta N_unita_oper

set categorial variables: Semestre sez_F1_M0 classe_eta N_unita_oper

Select % Training Data: 0.01 0.8 1

other parameters

0.7581868 AUC(Area Under Curve)

0.7378435 Accuracy value:

0.4076109 Cutoff:

233

0.465

0.9377289

2-ML model evaluation for classification

a)

Neural Network Experiment

Model overview

data	demo1
Target	mdr2
labels	5
Features	semester, sex_F1_M0, age_class, ward_N
specificity	0.850984931306849
sensitivity	0.649006622516556
AUC	0.662047516968158
Accuracy	0.649831649831165

catboots Experiment

Model overview

data	demo1
Target	mdr2
Features	semester, sex_F1_M0, age_class, ward_N, mdr2
Categorial	semester, sex_F1_M0, age_class, ward_N
specificity	0.732394366197183
sensitivity	0.693522580645161
AUC	0.738083189459336
Accuracy	0.71043771043771

Show 10 entries Search:

	names(Loaded_dataset)
1	demo1

Showing 1 to 1 of 1 entries Previous 1 Next

b)

3-User friendly interface: a- Model operations history; b-Dataflow editor

a)

Error: 133.786468 Steps: 4145

b)

Select Data frame for SVR

Type: C-classification

☑ Y is factor

Cost: 0,1

Gamma: 2

Kernel: linear

□ Scale

Select Dipendent variable: mdr2

select label: S R

Select Indipendent variable: semester age_class age ward_N sex_F1_M0

Select % Training Data: 0.01 0.8 1

4-Model visualization: a-Neural Networks; b-SVM

***Fig.4**: The DSaaS architecture*

Table 2 shows predictors and descriptive statistics for patients with and without an MDR urinary infection. Respectively 767 and 718 in-patients with and without hospital-acquired infections are present.

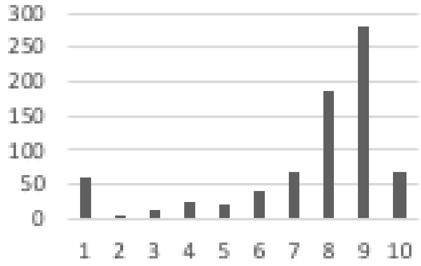
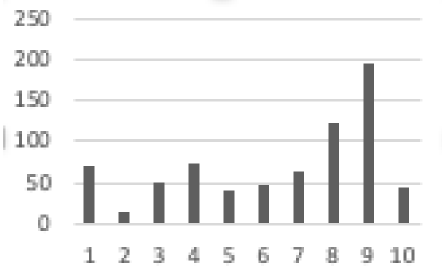
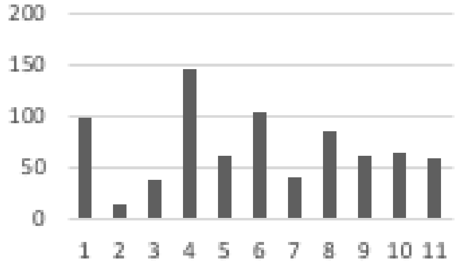
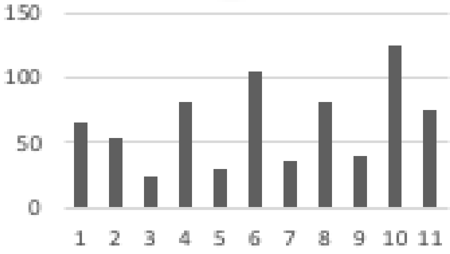
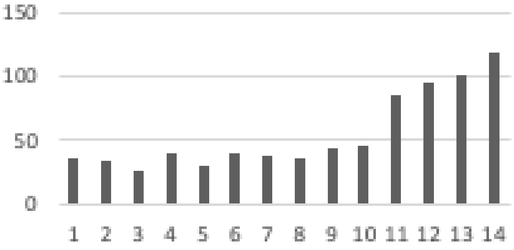
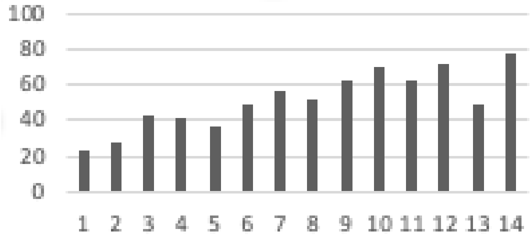
Variable	**Patients with MDR urinary infection**	**Patients without MDR urinary infection**
	Summary statistics	
Gender	Male: 267, Female: 500	Male: 149, Female: 569
Age	M: 70,0 (SD 25,5)	M: 59.5 (SD 28.7)
Age Class		
Ward		
Time Period (six-months)		

Table.2: *Descriptive statistics for patients with/without an MDR urinary infection*

Table 3 demonstrates the results of the three ML algorithms we tested with *DSaaS*. Accuracy, AUC, sensitivity and specificity were used to assess the performance of those methods. Since we adopted ten-fold cross validation for estimating model performance, the means and standard deviations of the above four metrics can be calculated for the training sample. Among the three methods employed, Catboost has the highest accuracy rate (0.711), followed by Neural Networks (0.646) and Support Vector Machine (SVM) (0.643) almost with the same performances. In terms of AUC, Catboost (0.735) has a value higher than 0.7, indicating a more than discreet classifier performance. The AUC values of Neural Networks and SVM are lower than 0.7, demonstrating poor performance. In sum, Catboost had a quite good performance while the remaining classifiers performed poorly. Finally, the sensitivity (recall) value told us that all the three classifiers had a high performance in discriminating true positives (MDR infection) with values of 0.895, 0.807 and 0.752 respectively for Catboost, SVM and Neural Networks.

Method	**Accuracy(SD)**	**AUC(SD)**	**Sensitivity(SD)**	**Specificity(SD)**
Catboost	0.711 (0.033)	0.735(0.028)	0.895(0.067)	0.489(0.122)
SVM	0.643 (0.008)	0.626(0.028)	0.807(0.028)	0.445(0.017)
NeuralNetworks	0.646 (0.020)	0.682(0.033)	0.752(0.073)	0.521(0.085)

Table 3: *Performance evaluation of models using the test set*

The field of bioinformatics is evolving from being a tool to a subject in its own right that needs new paradigms and methods to carry out biological experiments and data analysis. Planning a Data Science project is a difficult task as the purpose of the project may be unknown ex ante. Over the last few years, several development environments and platforms allowing the implementation of data science and machine learning techniques are emerging. Specifically, Bio7[38], R studio, Zeppelin, Jupiter, R analytic Flow, Window Azure, together with others, allow the creation of data science and machine learning processes in quite a simple way but still do not enable the complete definition of a data science pipeline system in a user-friendly way.

A data science team needs to work efficiently, compliant, agile and reproducible and above all faster. The current data science and ML platforms, while very performing, do not allow a quick approach to the solution of the problem. For this purpose, we designed and developed a new data science platform called *DSaaS* (Data Science as a Service) useful to easily perform ML experiments.

In our case-study, since we were dealing with data defined by binary targets describing whether an individual turned out to be affected by an MDR UTI or not, we decided to use a variety of well-known ML classification approaches previously implemented in *DSaaS*. In this way, we both studied the lending of the platform and were able to get a comparison of the classification performance using several classification models on the same dataset. Specifically, as a first step we decided to use SVM, Neural Networks and a quite new boosting method, known as Catboost, that is particularly suitable for dataset with an important presence of categorical features. Categorial data, differently form numerical quantitative data, can only assume a limited, and usually fixed number of possible values corresponding to different types or categories.

As target value we decided to assign 1 to individuals with the characteristic to have an MDR UTI (i.e., R) to two or more antibiotic classes. Therefore, in our dataset the negatives coincided with non-MDR UTIs and are described by all the points with target value equal to 0 (i.e., S). From Table 2 it can be noted that, among the three algorithms the one having highest results was Catboost: it had the best value in terms of sensitivity, AUC value, accuracy rate and finally we have a very low value for specificity. Note that, specificity measure has low results in all three methods, while we have obtained generally fair results (i.e., above 0,75) for sensitivity value. By definition of sensitivity we can conclude that our predictors have better results when a resistant data point, i.e. MDR, with target equal to 1, is considered. Hence, the used predictors have good skills in telling us if a new hospitalized patient is at risk of taking a multi-drug resistant (MDR) infection. Furthermore, as regards SVM and Neural Networks, they have similar accuracy and AUC results around the value 0,6. Finally, since we are dealing with an imbalanced dataset containing a very large number of positive samples, it is important to underline that AUC measure is to be preferred over accuracy value.

Despite numerous studies have investigated risk factors in UTIs [29], literature revealed that little of those studies adopted ML techniques for prediction. At the best of our knowledge, our study is the first adopting a ML approach in predicting the patient-related risk after the hospitalization to acquire an MDR UTIs. Further, by utilizing five differing classifiers easy to obtain from a new hospitalized patient, physicians may quickly adopt early prevention and intervention procedures and decision plans may be formulated in combination with related clinical experiences.

Furthermore, following the enhancement of the predictive model, the integration into the hospital computerized physician order entry could be done, where physicians may acquire a timely alert regarding the possibility of the onset of MDR UTIs in an early hospitalized patient.

2.4 Contributions

Alessio Mancini: Conceptualization, Methodology, Validation, Investigation, Data Curation, Writing - Original Draft & Editing

Leonardo Vito: Software, Methodology, Formal analysis, Validation, Investigation, Writing - Original Draft

Elisa Marcelli: Methodology, Writing - Original Draft

Marco Piangerelli: Writing – Review, Supervision, Methodology

Renato De Leone, Sandra Pucciarelli, Emanuela Merelli: Writing – Review, Project administration, Supervision

CHAPTER 3: Feature discovery in ciliates Retained introns using Machine Learning

3.1 The Mechanisms of intron splicing

The coding and non-coding part of genes is referred to as exons and introns respectively. When mRNA is synthesized, in eukaryotes, the mRNA precursors still contain introns transcribed from DNA template. The introns are then removed and adjacent exons are ligated [39]. The origin of introns is still not completely known. At first, processing these useless components appears very wasteful in terms of energy consumption, but today we know the existence of introns largely facilitates the diversity of gene products. The intron removal may involve different pathways, thereby producing functional distinct mRNA isoforms from a single gene. This mechanism is known as alternative splicing (AS). Exons often encode independent functional domains [39]. Therefore, AS provides a complex design to assemble different functional modules. This is an economic way to achieve the proteome diversity. Furthermore,ntrons can also play the cis-regulatory roles in splicing. For example, the intronic enhancers and silencers can promote or inhibit the splice site recognition. Furthermore,ntrons can also play the cis-regulatory roles in splicing. For example, the intronic enhancers and silencers can promote or inhibit the splice site recognition.

The length of introns affects the efficiency of transcription as well, and then the gene expression can be regulated [39]. AS is prevalent in eukaryotic genes. For example, a human transcriptome study by high throughput sequencing indicates that more than 95% human genes undergo AS[40]. It can be classified into five categories (Fig.5) and observations indicate that only a small minority of AS events are involved in the production of functional protein variants [40]. This led some authors to conclude that the vast majority of AS events correspond to splicing errors [41].

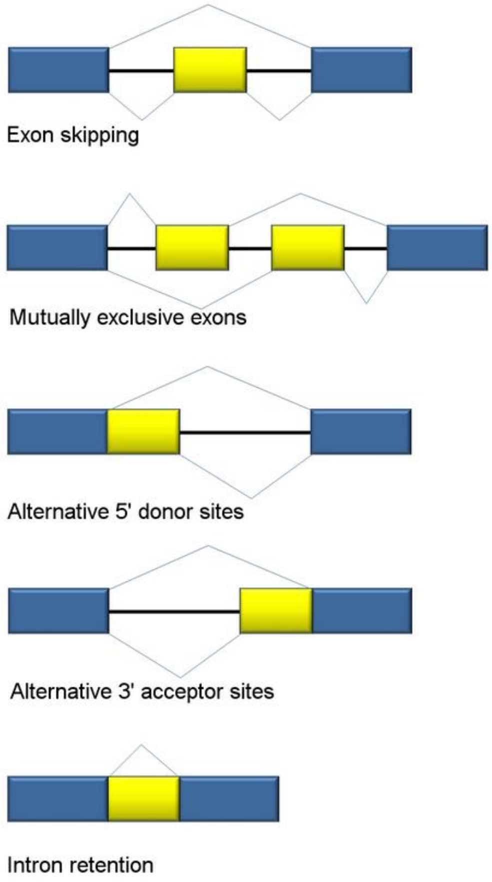


***Fig.5**: Traditional classification of basic types of alternative RNA splicing events. Exons are represented as blue and yellow blocks, introns as lines in between*[40]

1. Exon skipping: An exon may be retained or removed.
2. Mutually exclusive exons: Only one of two exons is retained
3. Alternative donor site: An alternative upstream exon boundary used.
4. Alternative acceptor site: An alternative downstream exon boundary used.
5. Intron retention: An intron may be retained or removed.

3.2 Intron recognition

To date, next generation sequencing (NGS)[42] has become one of the most promising Bioinformatic technologies to study the genomes and transcriptome structure. Since Illumina sequencers can generate highest throughput of NGS reads, they have become the most dominant platform in this field. One of the main problems of Illumina reads is the read length. DNA or RNA in the library preparation step are chopped into smaller fragments. Each fragment can be sequenced from one end up to 150bp only (single-end). The major problem of single end reads is the ambiguity when reads are mapped to multiple loci. A simple improvement to the single-end library preparation is to sequence both ends of fragments (scanning both the forward and reverse template strand). The paired-end sequencing incorporates the fragment length information, which can significantly improve the mapping and assembly accuracy. The typical fragment length of paired-end sequencing is 200-500bp. If the reference genome is available, the way to deal with NGS transcriptome reads is to map the reads back to the reference genome. Sequence alignment is an old bioinformatics problem. The classical method is to align reads back to genome using fast alignment algorithms such as BLAST indexes k-mers. These k-mer seeds are then extended using traditional alignment methods [42] effective with small libraries but not so useful for millions of very short reads. Therefore, the new NGS aligners are rapidly introduced and have become one of the prosperous fields in bioinformatics. The main drawback of these programs is the requirement of memory.

Many previously described RNA-seq aligners were developed as extensions of contiguous short read mappers, which were used to either align short reads to a database of splice junctions or align split-read portions contiguously to a reference genome, or a combination thereof. In contrast to these approaches RNA STAR[43] is designed to align the non-contiguous sequences directly to the reference genome using less amount of memory. STAR algorithm consists of two major steps: seed searching step and clustering/stitching/scoring step.

3.3 Intron Retention

Albeit part of splicing mechanism has been deciphered and several tools have been developed to find the splice junctions given a piece of DNA sequence, other tools are used to discover the AS isoforms.

Historically being considered as transcriptional noise or ‘junk’, intron retention (IR) has recently been shown to carry out important biological functions such as regulating gene expression that is coupled with nonsense mediated decay[44], producing novel isoforms[45], and targeting specific cell compartments [45]. Previous studies have shown that IR functions in the homeostatic control of the expression of some RNA processing and export factors[46,47]. More recently, it has emerged that IR also controls the expression of developmentally regulated genes in plants and animals [47,40]. For example, a set of retained introns in a murine neuroblastoma cell line was shown to negatively regulate genes with neural-associated functions. Several of these introns were linked to nuclear retention and exosome-mediated RNA turnover of the host transcripts[48]. In contrast, another set of IR events was found to control the levels of transcripts important for granulocyte maturation[49], largely through the process of nonsense-mediated mRNA decay (NMD). These recent studies suggest that different IR events control gene expression through distinct mechanisms. However, the extent to which IR operates across different primary cells and tissues to regulate gene expression via these and possibly additional mechanisms is unknown.

To study IR, first the transcriptome structure must to be known. Next generation sequencing has resulted in a vast amount of RNA-seq data, which provides a rich resource for the detection of IR in combination with bioinformatics tools. Cufflinks[50] for example assembles transcripts, estimates their abundances, and tests for differential expression and regulation in RNA-Seq samples. It accepts aligned RNA-Seq reads and assembles the alignments into a parsimonious set of transcripts. Cufflinks then estimates the relative abundances of these transcripts based on how many reads support each one, taking into account biases in library preparation protocols. Cufflinks is a more natural descendant of the expressed sequence tag-based algorithms. A minimal set of isoforms explaining all the reads is constructed. In case there is more than one possible set of the same size explaining all the reads, a cost function is used to choose the ideal set. Then, an algorithm is used to quantify the isoforms in this set by modeling the probability “ρt” that a read came from transcript “t”. An advantage of this method is that it is flexible enough to accommodate pair end reads in a natural way. Also, both isoform discovery and quantification are addressed; however, the tasks are broken apart and performed separately. With this approach, different RNA-seq replicates or different cellular states (i.e. starvation, conjugation) can be analysed at the same time and then merged to discover more isoforms and increase the transcript coverage[51].

Then, with specific tools, retained introns can be discovered. ASTALAVISTA (Alternative Splicing Transcriptional Landscape Visualization Tool)[52] employs an intuitive and complete notation system to univocally identify IR events. The method extracts AS events dynamically from custom gene annotations, classifies them into groups of common types and visualizes a comprehensive picture of the resulting AS landscape. Thus, ASTALAVISTA can characterize IR for whole transcriptome data from reference annotations as well as for genes selected by the user according to common functional/structural attributes of interest.

iREAD (intron REtention Analysis and Detector)[52], is another tool to detect IR events genome-wide from high-throughput RNA-seq data. The command line interface for iREAD is implemented in Python. iREAD takes as input a BAM file[53], representing the transcriptome, and a text file containing the intron coordinates of a genome. It then 1) counts all reads that overlap intron regions, 2) detects IR events by analyzing the features of reads such as depth and distribution patterns, and 3) outputs a list of retained introns into a tab-delimited text file.

3.4 Machine learning in genomics

Several algorithms can be used in genomics for the researcher's purposes. One of them is the supervised machine learning algorithm "Support Vector Machine (SVM)"[54]. SVM's purpose is to predict the classification of a query sample by relying on labeled input data which are separated into two group classes by using a margin. Specifically, the data is transformed into a higher dimension, and a support vector classifier is used as a threshold (or hyperplane) to separate the two classes with minimum error (Fig.6). Some studies suggest that SVM, in some cases, outperform neural networks and decision trees[55] for classification of various problems in the domain of bioinformatics.

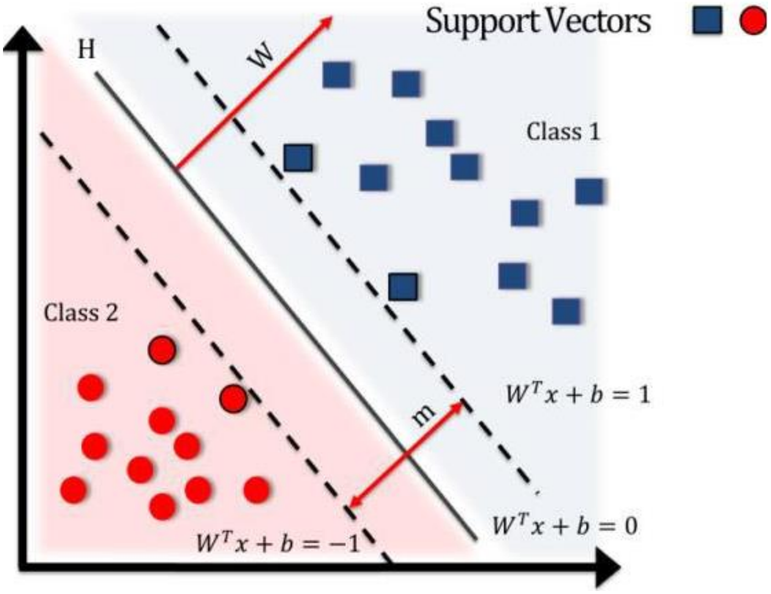


***Fig.6**: SVM classification scheme, H is the classification hyperplane; W is the normal vector to the hyperplane; m is the minimum distance between positive and negative hyperplanes[124]*

Neural Networks (NNs)[56] are adaptive nonlinear information processing systems which combine numerous processing units with a series of characteristics such as self-adapting, self-organizing and real-time learning (Fig.7). NNs have already been adapted for genomics problems such as motif discovery [57], predicting the deleteriousness of genetic variants [58], and gene expression inference [59]. There has been a growing interest to predict function directly from sequence, instead of from curated datasets such as gene models and multiple species alignment.

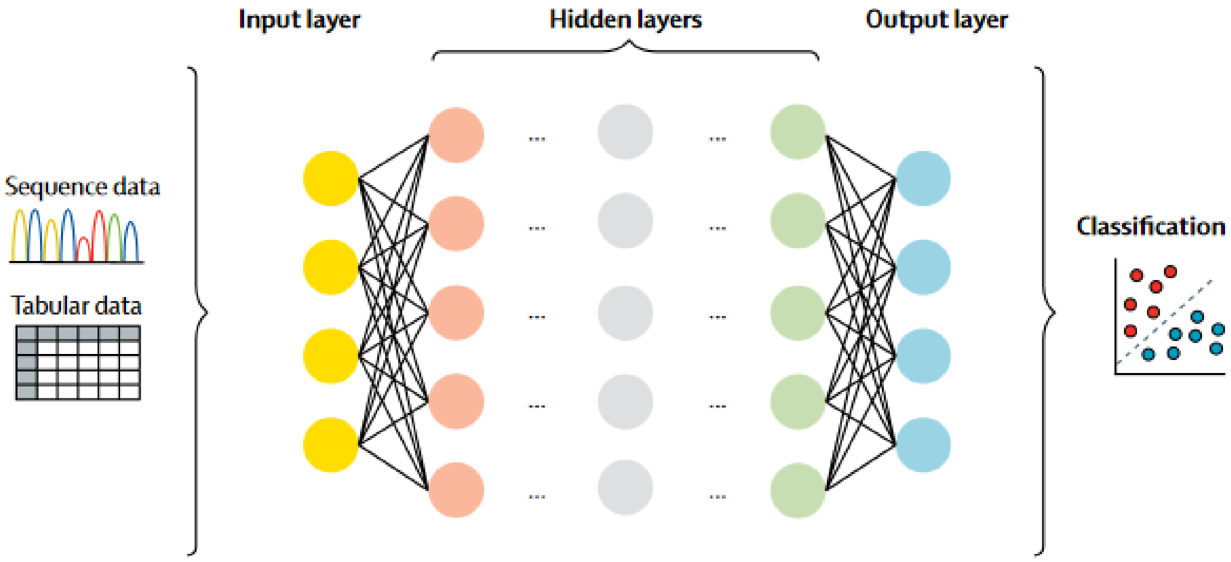


***Fig.7**: Schematic of NNs. Data are processed by a series of layered nodes, or neurons. The output can be used for classification*[56]

Other popular algorithms are decision trees[60]. The first concept of decision tree was proposed by Hunt.E.B et al in 1966. Based on it, a lot of improved algorithms have emerged. Among these, the most famous algorithm is ID3 with a choosing policy according to information gain, which was proposed by Quinlan in 1986.

C5.0[61] is another new decision tree algorithm developed based on C4.5 by Quinlan. It includes all functionalities of C4.5 and apply a bunch of new technologies, among them the most important application is "boosting" technology for improving the accuracy rate of identification on samples.

Gradient boosting[34,62] is a powerful machine-learning technique that achieves state-of-the-art results in a variety of practical tasks. For many years, it has remained the primary method for learning problems with heterogeneous features, noisy data, and complex dependencies: web search, recommendation systems, weather forecasting, and many others. This is essentially a process of constructing an ensemble predictor by performing gradient descent in a functional space. It is backed by solid theoretical results that explain how strong predictors can be built by iteratively combining weaker models (base predictors) in a greedy manner.

AdaBoost[63], acronym of "Adaptive Boosting", proposed by Freund and Schapire in 1996, was the first very successful boosting algorithm developed for binary classification. It represents a popular boosting technique that helps to combine several "weak classifiers" into one "strong classifier" (Fig.8). A weak classifier is simply a classifier that works poorly, but works better than a random guess. By merging so many models of this type, AdaBoost is able to generate a model that overall is better than the single weak classifiers taken individually. Adaboost uses many decision trees at a depth level, called decision stumps, as many as the characteristics of the model. To every iteration, a new weak classifier is introduced in sequence and aims to compensate the "deficiencies" of the previous models to create a strong classifier. The general objective of this exercise is to consecutively adapt new models to provide more accurate estimates of our variable response. Actually, AdaBoost does not only accept decision trees as weak learners: any automatic learning algorithm can be used as a basic classifier if it accepts weights on the training set.

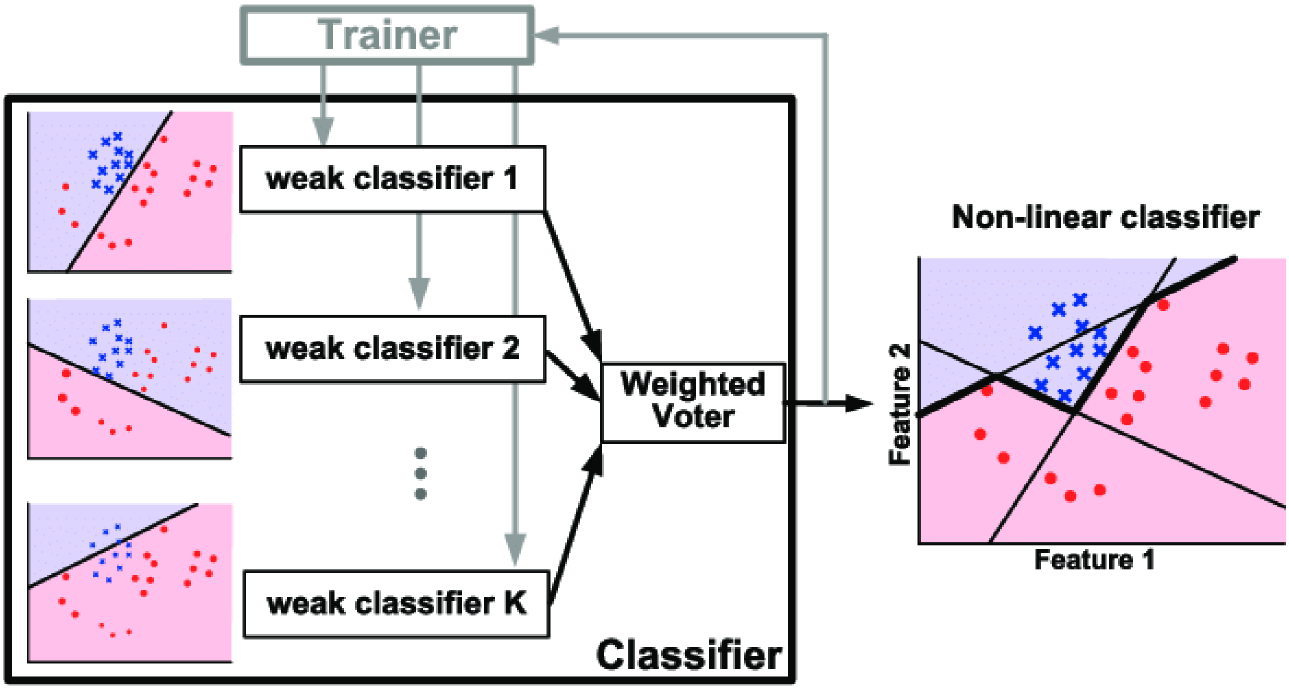


Fig.8: Illustration of AdaBoost algorithm for creating a strong classifier based on multiple weak linear classifiers[98].

3.5 Ciliates as model organisms

Studies on ciliates have contributed to several scientific milestones. Ciliates are advantageous as a model eukaryotic system because they grow rapidly to high density in a variety of media and conditions, their life cycle allows the use of conventional tools of genetic analysis, and molecular genetic tools for sequence-enabled experimental analysis of gene function have been developed[64]. In addition, although they are unicellular, they possess many core processes conserved across a wide diversity of eukaryotes (including humans) that are not found in other single-celled model systems (e.g., the yeasts *Saccharomyces cerevisiae* and *Schizosaccharomyces pombe*). These unicellular eukaryotes are characterized by the presence of hair-like organelles called cilia, which are identical in structure to flagella but typically shorter and present in larger number. Protists use motile cilia for locomotion as well as for sensory perception to detect light, odors, soluble chemicals and mechanical forces. These single-celled microorganisms are spread almost everywhere such as in lakes, ponds, oceans, rivers and soils. Moreover, there are plenty of data in the public repositories, genomes, transcriptomes and gene annotations useful to build a generalized predictive model for all the ciliates.

In hash and stress conditions, ciliates undergo a sexual phenomenon known as conjugation, that occurs by the mating between two cells of different and compatible "mating types"[65]. The aim of conjugation is mainly to increase genetic variability in the population that is obtained by genetic recombination and extensive nuclear reorganization including a complex DNA rearrangement.

Even though ciliates are unicellular organisms, they maintain the germ and somatic line within a single cell represented by two different kinds of nuclei (nuclear dimorphism). The micronucleus (MIC) is diploid and transcriptionally silent. The MIC undergoes meiosis during conjugation. The macronucleus (MAC) is highly polyploid, i.e. it contains several copies of each homologous gene; it is transcriptionally active and controls the cell phenotype. The MAC derives from the MIC after several genomic rearrangement that occur during conjugation that include DNA sequence elimination, scrambling and amplification. During the vegetative growth, the MIC divides mitotically, while the MAC by a process named amitosis where the homologous polyploid chromosomes segregate randomly into the new daughter cells[65].

Tetrahymena species have contributed to fundamental biological discoveries such as telomerase, telomeric repeats, catalytic RNA and the function of histone acetylation[66]. *T. thermophila* (Fig.9), *T. borealis, T. elliotti and T. malaccensis* genomes and transcriptomes under different physiological conditions have been sequenced, providing a large amount of data that can be used to study alternative intron splicing. According the literature *Tetrahymena* species have the highest number and percentage of genes showing AS reported in a unicellular eukaryote[67].

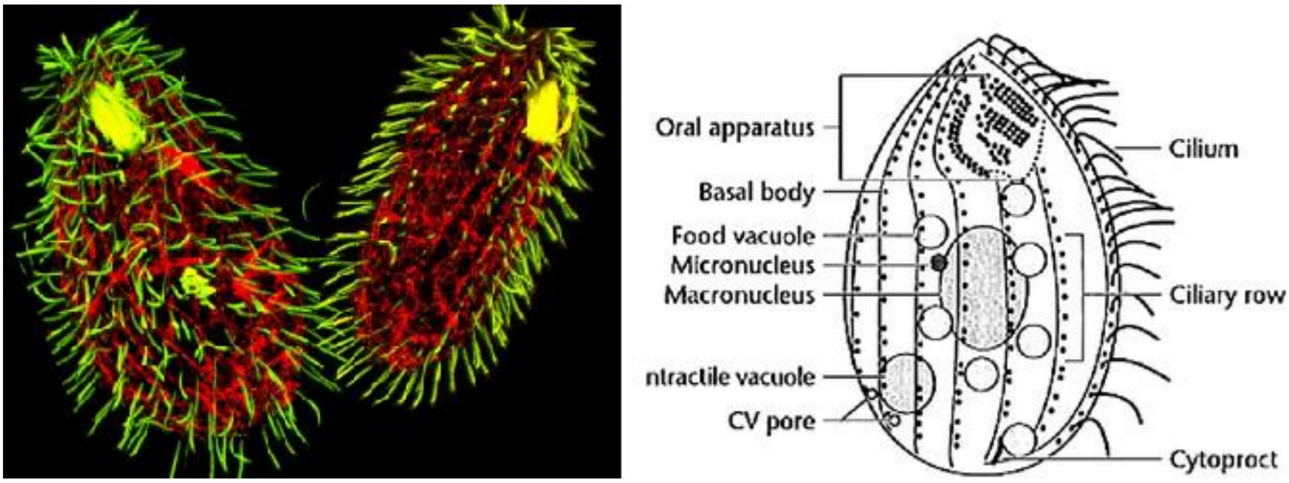


Fig.9*: Tetrahymena thermophila confocal immunofluorescence microscopic images*[125]

Paramecium tetraurelia (Fig.10) is another ciliated model organism[68] extensively used to study the mechanism of mating[69] and genome rearrangement[70], as well as the process of NMD[71]. The intron density in *P. tetraurelia* (2.3 introns per gene on average) is similar to that observed in many other unicellular eukaryotes, and some animals, such as *Drosophila*[72]. *Paramecium* introns are very short (25.1 bp on average, with 99.9% of them in the range of 20–35 bp), i.e. much shorter than RNA-seq sequence reads, which greatly simplifies the detection and classification of AS events. In particular, cases of IR can be identified directly by detecting sequence reads spanning the entire intron and its flanking exon boundaries. Moreover, given its high number of genes (~40,000), this genome allows the analysis of a large dataset of introns (>90,000 introns). Finally, this organism already proved to be a good model to reveal important general features of splicing control in eukaryotes[71].

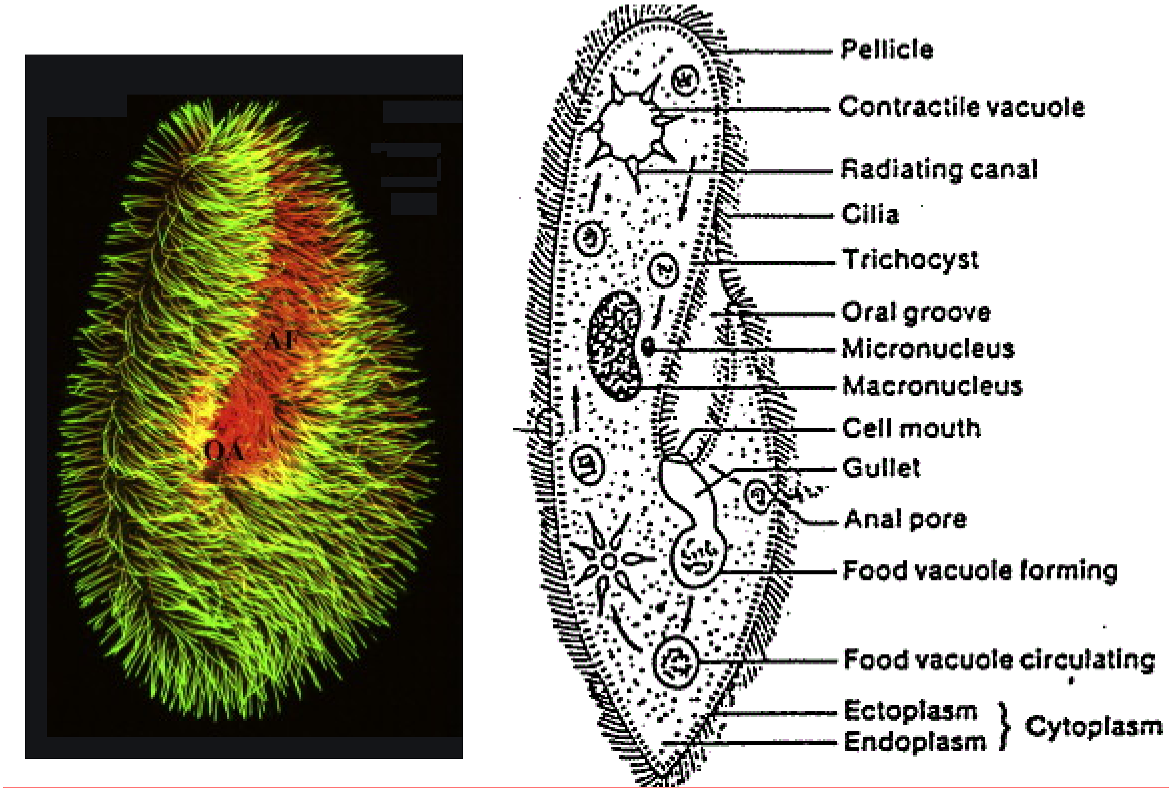


***Fig.10**: Paramecium tetraurelia confocal immunofluorescence microscopic images*[126]

Euplotes is a genus of free-living marine ciliates that play important roles as both predators of microalgae and preys of multicellular eukaryotes like flatworms[73]. *Euplotes focardii (*fig.11) is a marine ciliate of the Antarctic coastal seawaters, which lives between -1.9 °C and +1.9°C in natural conditions. It has been isolated from sediment and seawater samples collected in Terra Nova Bay by Valbonesi and Luporini in 1993[74]. It has optimal survival and multiplication rates at 4-5 °C under laboratory conditions but its viability declines sharply at temperatures greater than 8-10°C; after three hours of exposure at more than 20°C it is irreversibly damaged and dies[74]. *Euplotes focardii* has a doubling time of three days under normal feeding conditions but it shows cannibalistic phenomena under starvation conditions. It exhibits a peculiar behavior during conjugation because the formation of the conjugating pairs takes about 18-24 hours after which the process proceeds for other 10 days[74], instead of 12-16 hours as in other *Euplotes* species. One of the two partners reduce considerably its cell body and then is absorbed by the other one at the end of the reproductive process, in contrast with other ciliate species in which partners separate and complete independently macro and micronuclear development. The *E. focardii* genome and transcriptome have been studied by my research group (Mozzicafreddo et al., in press). Genome sequencing was performed form the Department of Medicine at the Harvard Medical School in Boston, USA. The annotation process for the transcriptome and genome[75] has been completed. *E. focardii* may represent an ideal model species for genome-level analysis to understand the evolutionary mechanisms of cold adaptation in psychrophilic organisms.

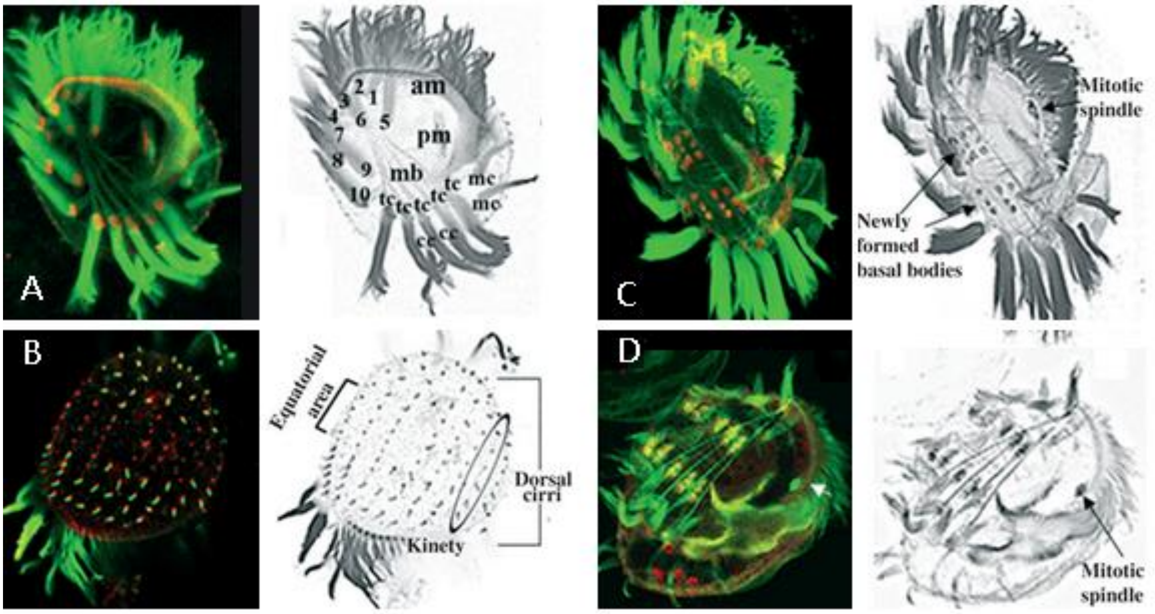


***Fig.11**: Euplotes focardii confocal immunofluorescence microscopic images*[127]*. A) Ventral view; B) Dorsal view; C) Ventral view-1 in mitosis; D) Ventral view-2 in mitosis*

As a model organism in studies of cell and environmental biology, the free-living and cosmopolitan ciliated protist *Euplotes vannus* has more than ten mating types[73] and shows strong resistance to environmental stresses. It shows intriguing features like most of ciliates, dual genome architecture (i.e., separate germline and somatic nuclei in each cell/organism), "gene-sized" chromosomes, stop codon reassignment, programmed ribosomal frameshifting (PRF) and strong resistance to environmental stressors[76]. However, the molecular mechanisms that account for these remarkable traits remain largely unknown. Both of these organisms have never been studied in terms of their AS arrangements

3.6 Methods

3.6.1 The pipeline

The figure below shows the pipeline applied for the study (Fig.12). All genome and transcriptome data were downloaded from the European Bioinformatics Institute (EMBL-EBI)[60] a part of EMBL.

The second step of the pipeline (reads cleaning, mapping and assembly) was done using Galaxy Europe[77] an open, web-based platform for accessible and reproducible computational biological research. Here users can easily run tools without writing code or using the command line interface all via a user-friendly web interface.

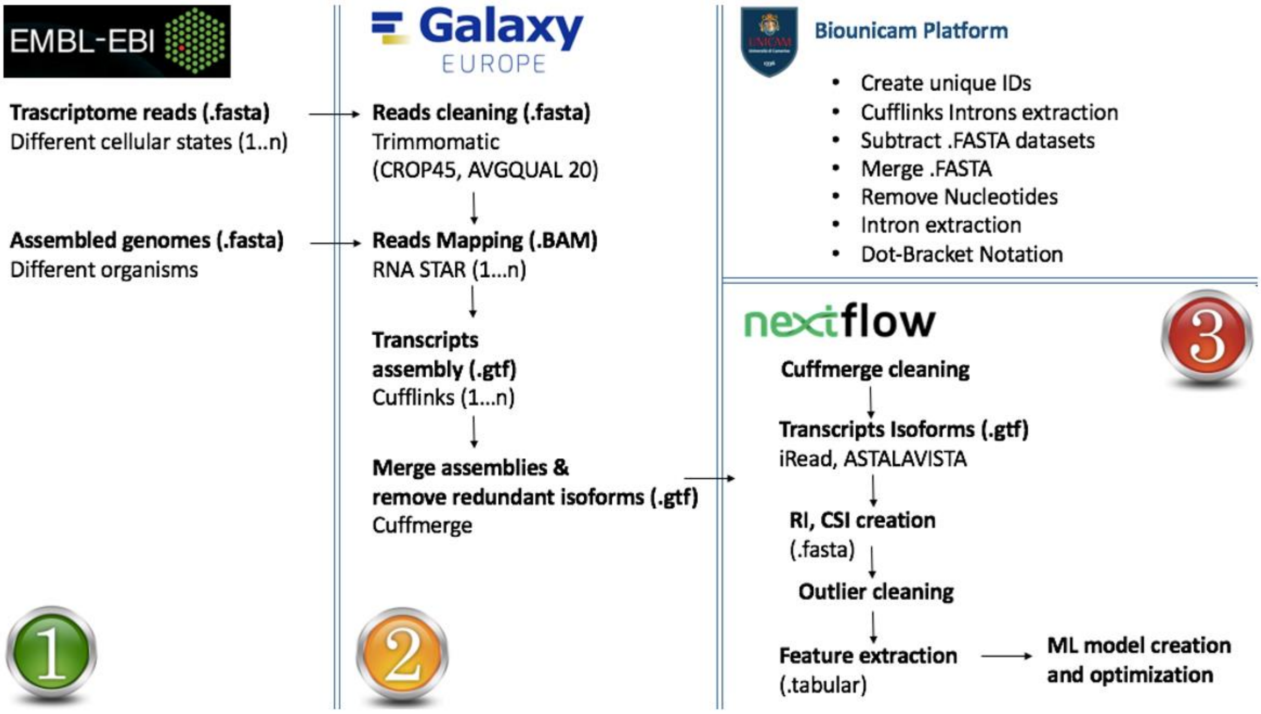


Fig.12*: The computational pipeline for the identification and classification of RIs*

For the third step, several tools were required to extract exons and introns from the galaxy outputs. Some of these were able to use input files from standard Galaxy-generated outputs, but some required custom and modified files. My colleague, Dr. Leonardo Vito, decided to bundle all the scripts I needed to edit and extract results in an online and user-friendly service called Biounicam (Fig.13). This service was developed using Node Red[78], useful to create the graphical interface and to use the individual tools as a dataflow. The Node Red components are available on Github (Fig 14) at the following address: https://github.com/leonardovito/node-red-biounicam-tool All the platform features are described in Chapter 3.6.2 Biounicam platform.

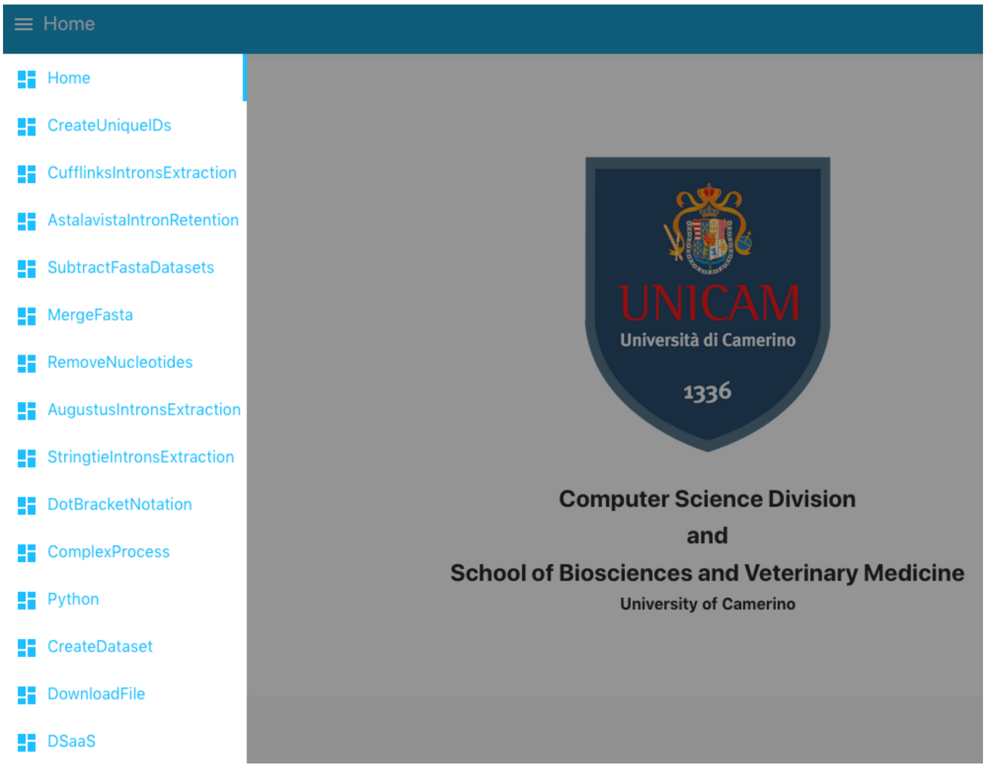
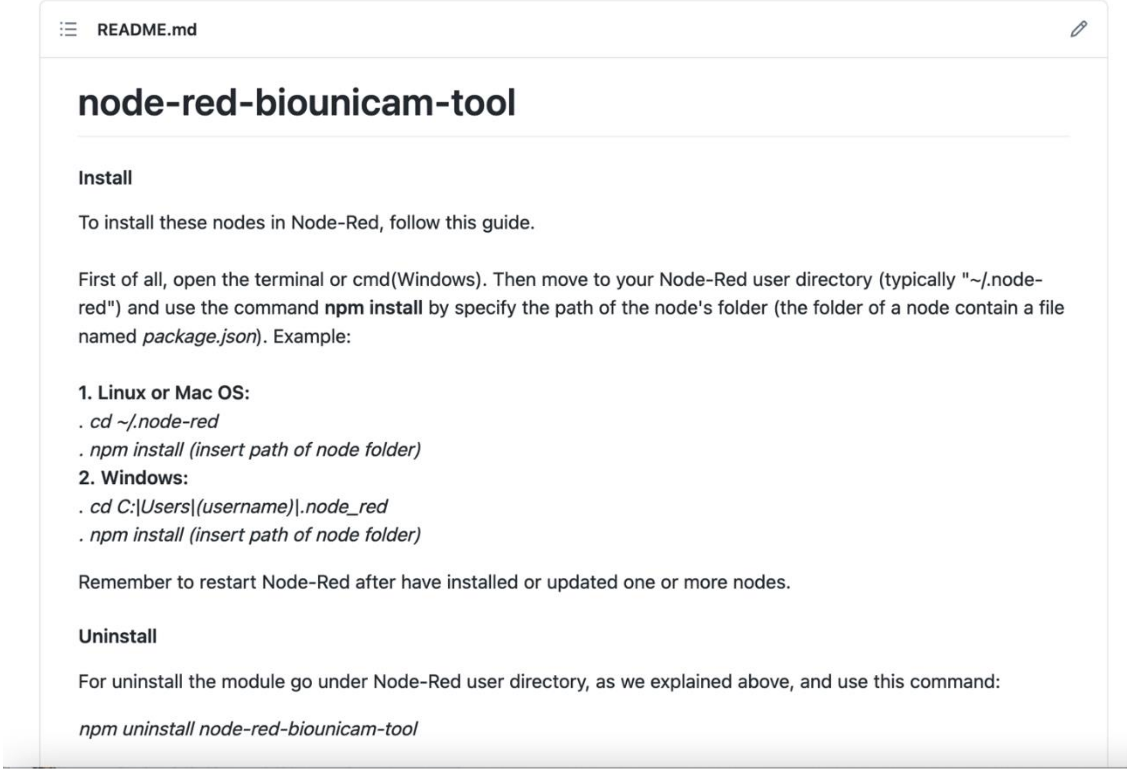


Fig.13*: Biounicam Homepage. This is the welcome page of the platform. The user can navigate it using the sections in the left window*



node-red-biounicam-tool

Install

To install these nodes in Node-Red, follow this guide.

First of all, open the terminal or cmd(Windows). Then move to your Node-Red user directory (typically "~/.node-red") and use the command **npm install** by specify the path of the node's folder (the folder of a node contain a file named *package.json*). Example:

1. Linux or Mac OS:
. *cd ~/.node-red*
. *npm install (insert path of node folder)*
2. Windows:
. *cd C:|Users|(username)|.node_red*
. *npm install (insert path of node folder)*

Remember to restart Node-Red after have installed or updated one or more nodes.

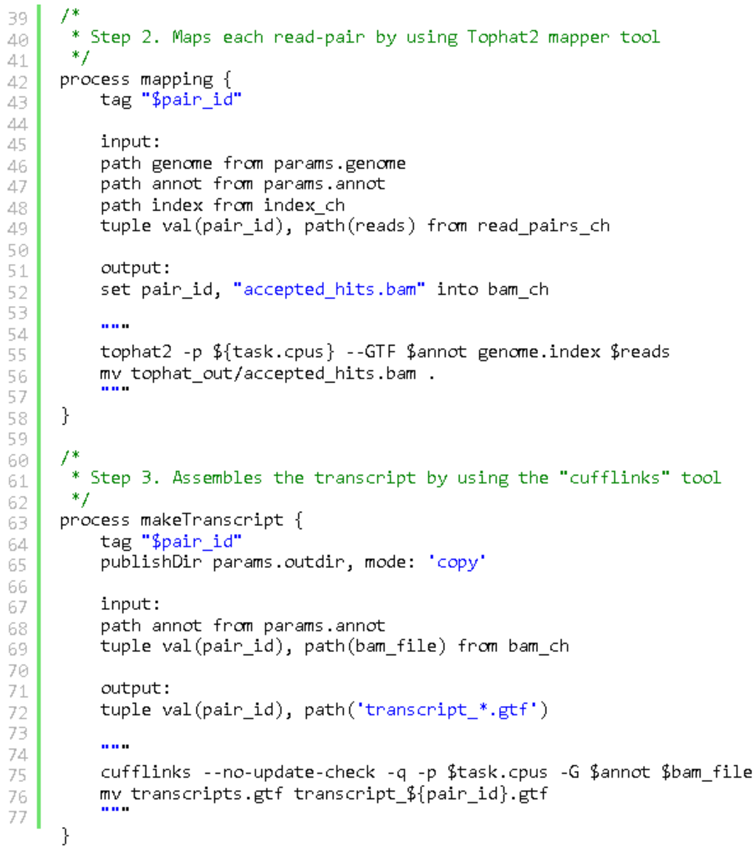
Uninstall

For uninstall the module go under Node-Red user directory, as we explained above, and use this command:

npm uninstall node-red-biounicam-tool

Fig.14*: Biounicam Github page. The user can download the source code*

Eventually to complete the last steps of the pipeline (retained introns classification, feature extraction and Machine Learning model creation) I used NextFlow[53]. It is a low-level framework useful to automate and concatenate different scripts in order to allow the following script to use as input the output of the previous script. Nextflow is a reactive workflow framework and a programming domain specific language that eases the writing of data-intensive computational pipelines. It is designed around the idea that the Linux platform is the lingua franca of data science. Linux provides many simple but powerful command-line and scripting tools that, when chained together, facilitate complex data manipulations. Nextflow extends this approach, adding the ability to define complex program interactions and a high-level parallel computational environment based on the dataflow programming model. This framework enabled us to speed up the single processes, and to execute the whole pipeline for every organism in an easy and reproducible way.



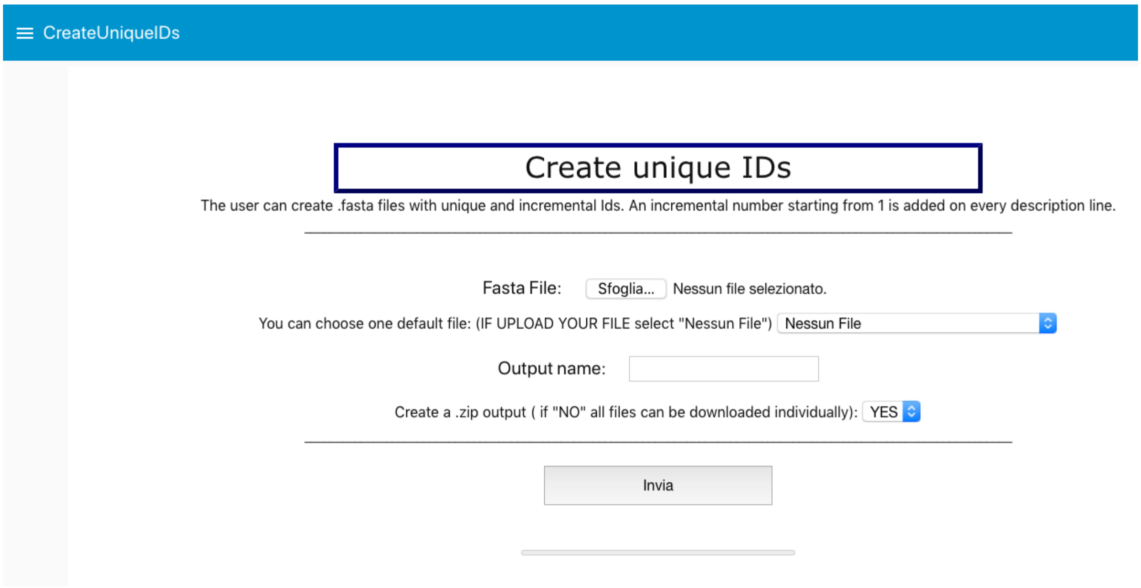
```
/*
 * Step 2. Maps each read-pair by using Tophat2 mapper tool
 */
process mapping {
    tag "$pair_id"

    input:
    path genome from params.genome
    path annot from params.annot
    path index from index_ch
    tuple val(pair_id), path(reads) from read_pairs_ch

    output:
    set pair_id, "accepted_hits.bam" into bam_ch

    """
    tophat2 -p ${task.cpus} --GTF $annot genome.index $reads
    mv tophat_out/accepted_hits.bam .
    """
}

/*
 * Step 3. Assembles the transcript by using the "cufflinks" tool
 */
process makeTranscript {
    tag "$pair_id"
    publishDir params.outdir, mode: 'copy'

    input:
    path annot from params.annot
    tuple val(pair_id), path(bam_file) from bam_ch

    output:
    tuple val(pair_id), path('transcript_*.gtf')

    """
    cufflinks --no-update-check -q -p $task.cpus -G $annot $bam_file
    mv transcripts.gtf transcript_${pair_id}.gtf
    """
}
```

Fig.15: *Nextflow code. In particular two concatenated scripts for mapping and transcriptome assembly*

Some Nextflow tasks were already present as stand-alone tools in the Biounicam platform (Cufflinks Introns extraction, ASTALAVISTA Intron retention, subtract FASTA datasets, Dot-Bracket Notation), some other were added de novo. All the Nextflow tasks are described in chapter 3.6.5 Step 3: The Nextflow Framework.

3.6.2 Biounicam platform

This is an overview of the Biounicam platform. Not all the sections were used for our purposes, like section 6, 7 and 8. The platform still yet in development and in the near future we will add new tools.



The user can create .fasta files with unique and incremental Ids. An incremental number starting from 1 is added on every description line.

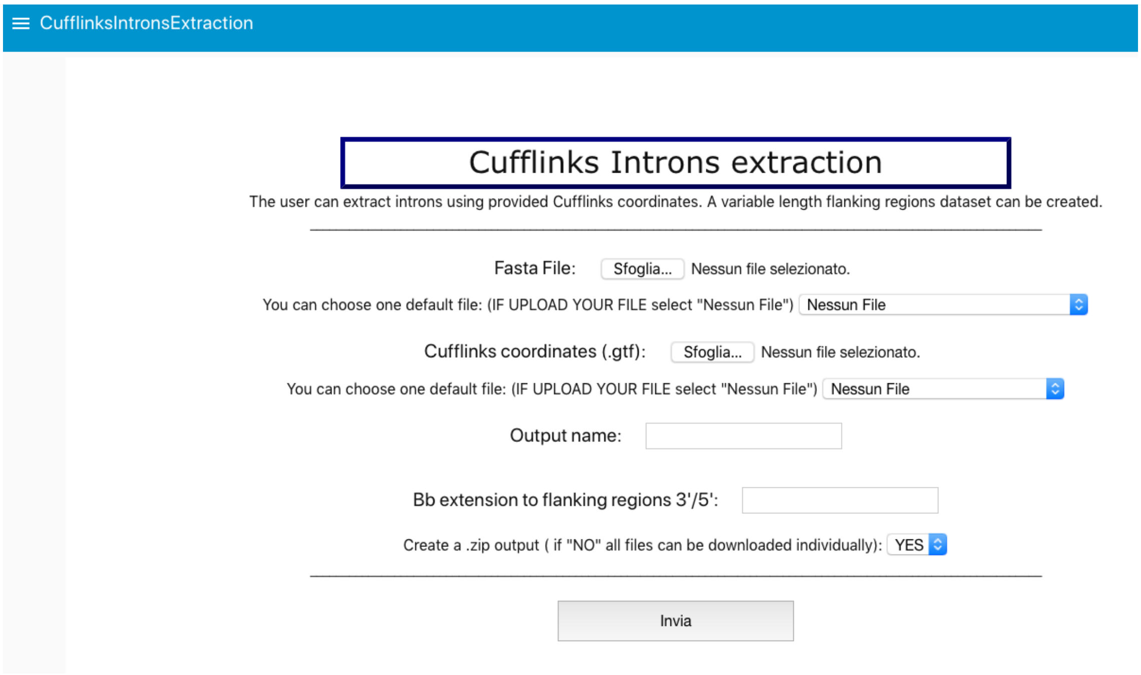
Fasta File: Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Output name:

Create a .zip output (if "NO" all files can be downloaded individually): YES

Fig.16*: Section 1. The user can create .FASTA files with unique and incremental IDs. An incremental number starting from 1 to n is added on every description line*


☰ CufflinksIntronsExtraction

Cufflinks Introns extraction

The user can extract introns using provided Cufflinks coordinates. A variable length flanking regions dataset can be created.

Fasta File: Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Cufflinks coordinates (.gtf): Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

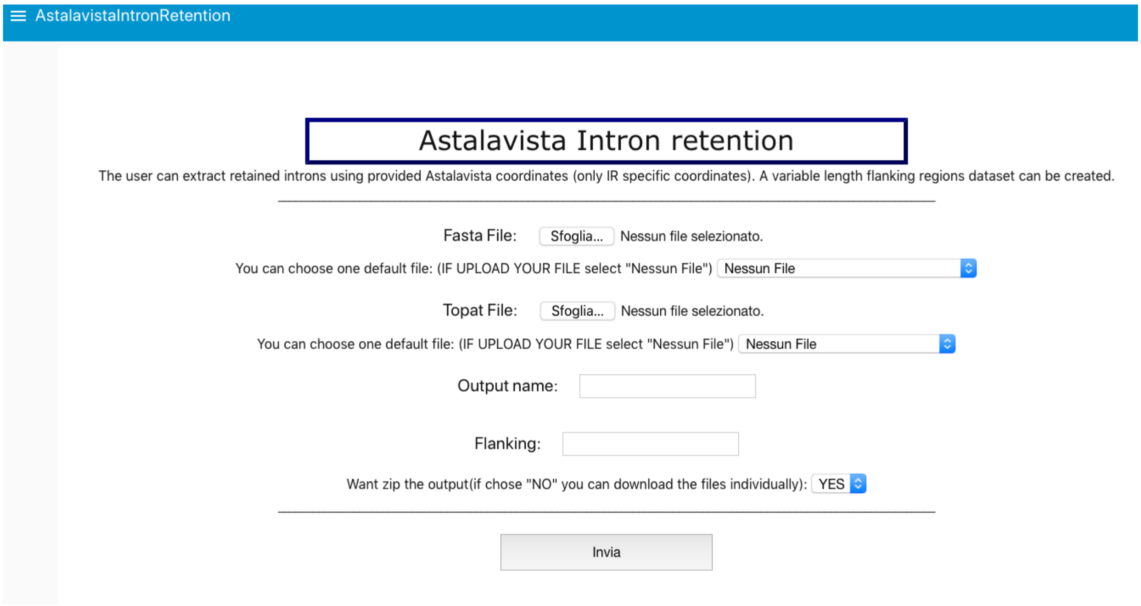
Output name:

Bb extension to flanking regions 3'/5':

Create a .zip output (if "NO" all files can be downloaded individually): YES

Invia

Fig.17*: Section 2. Cufflinks Introns extraction: The user can extract introns using provided Cufflinks (.gtf) coordinates. The user can choose to extent the intron flanking regions (5'-; 3'-) of a desired length*


☰ AstalavistaIntronRetention

Astalavista Intron retention

The user can extract retained introns using provided Astalavista coordinates (only IR specific coordinates). A variable length flanking regions dataset can be created.

Fasta File: Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Topat File: Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

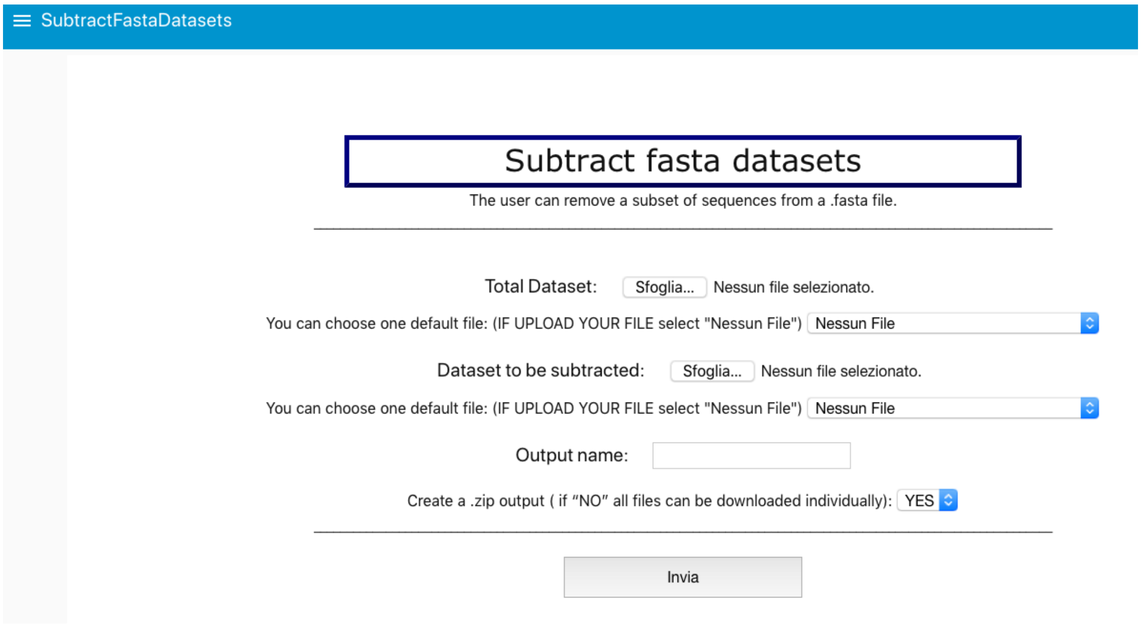
Output name:

Flanking:

Want zip the output(if chose "NO" you can download the files individually): YES

Invia

Fig.18: *Section 3. Astalavista Intron retention: The user can extract retained introns using provided Astalavista (.gtf) coordinates (only IR specific coordinates). The user can choose to extent the intron flanking regions (5'-; 3'-) of a desired length*



SubtractFastaDatasets

Subtract fasta datasets

The user can remove a subset of sequences from a .fasta file.

Total Dataset: Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Dataset to be subtracted: Sfoglia... Nessun file selezionato.

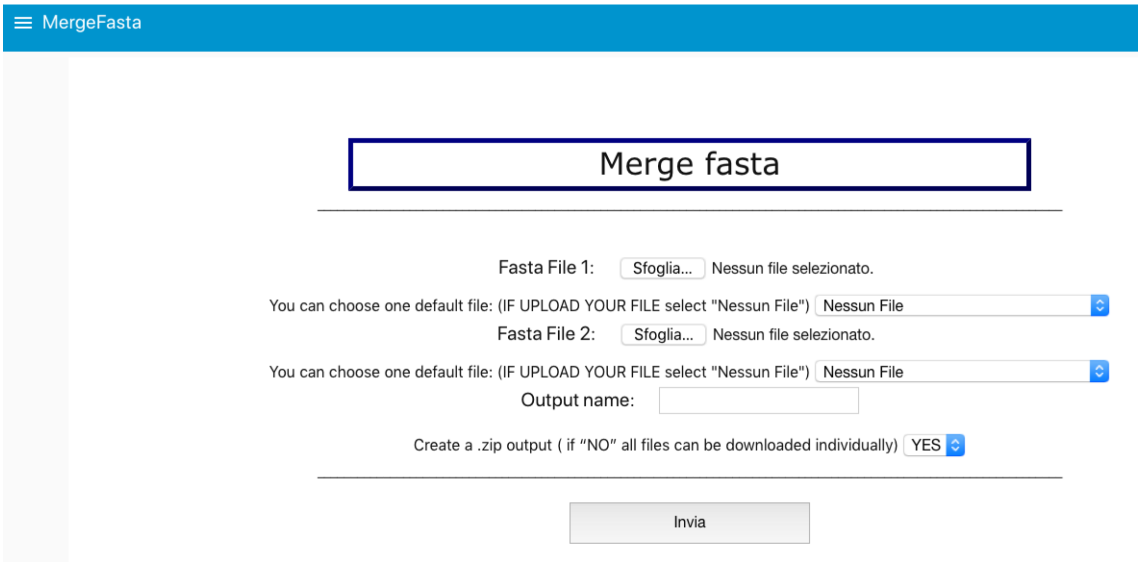
You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Output name:

Create a .zip output (if "NO" all files can be downloaded individually): YES

Invia

Fig.19: *Section 4. Subtract FASTA datasets: the user can remove a subset of sequences from a .FASTA file*



MergeFasta

Merge fasta

Fasta File 1: Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Fasta File 2: Sfoglia... Nessun file selezionato.

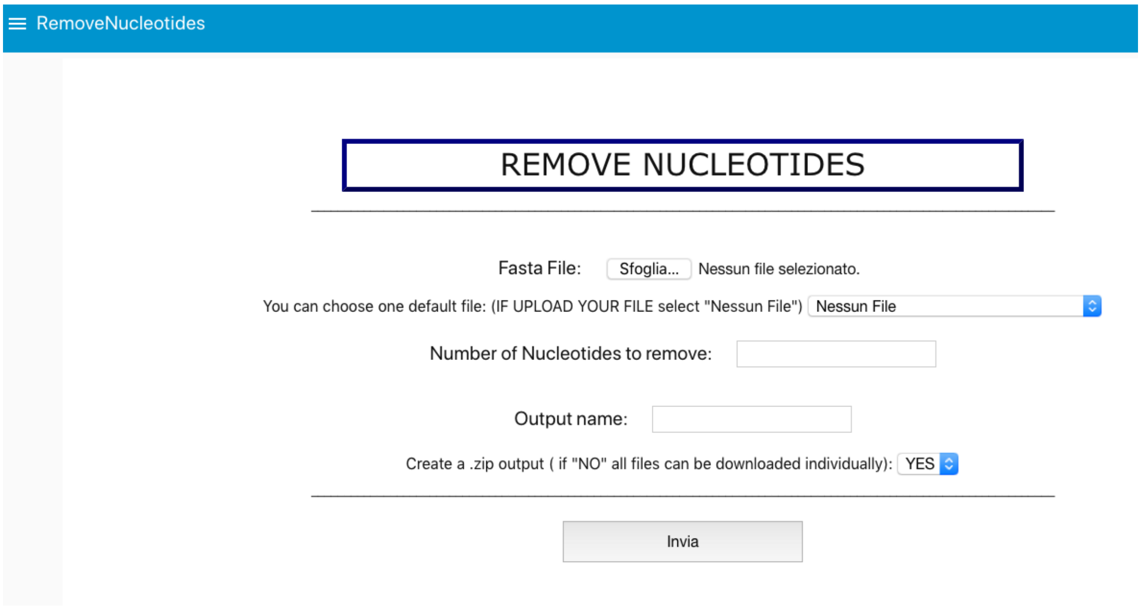
You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Output name:

Create a .zip output (if "NO" all files can be downloaded individually) YES

Invia

Fig.20: *Section 5. Merge FASTA: the user can merge two FASTA files*



REMOVE NUCLEOTIDES

Fasta File: Sfoglia... Nessun file selezionato.

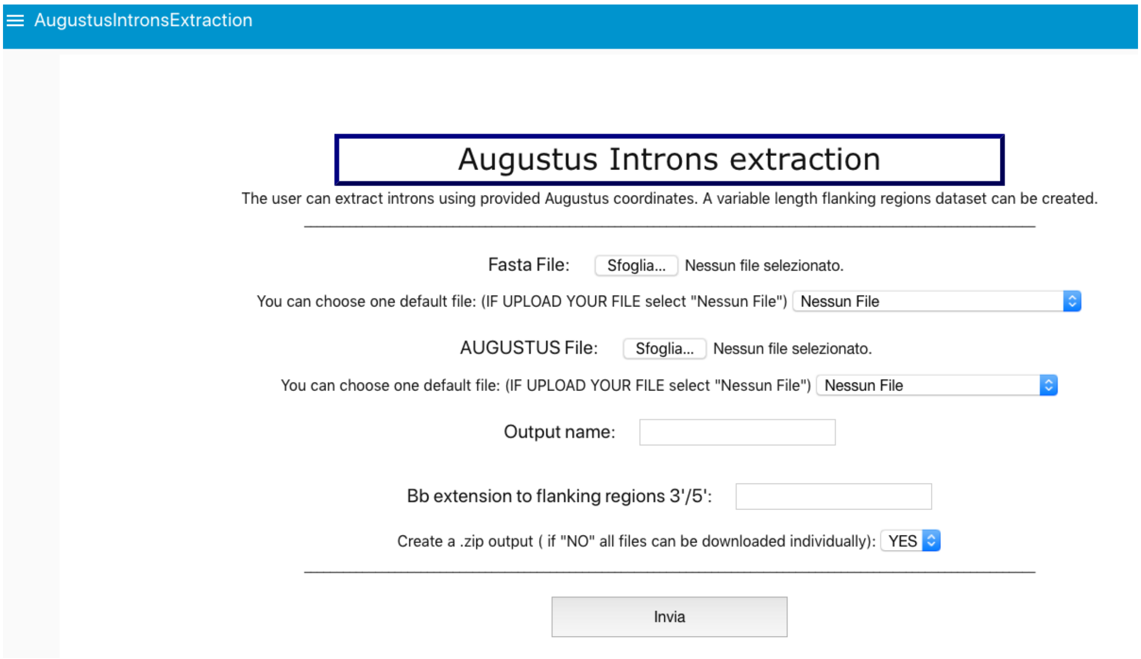
You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Number of Nucleotides to remove:

Output name:

Create a .zip output (if "NO" all files can be downloaded individually): YES

Fig.21*: Section 6. Remove Nucleotides: the user can extend or shorten the intron flanking regions (5'-; 3'-)*



Augustus Introns extraction

The user can extract introns using provided Augustus coordinates. A variable length flanking regions dataset can be created.

Fasta File: Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

AUGUSTUS File: Sfoglia... Nessun file selezionato.

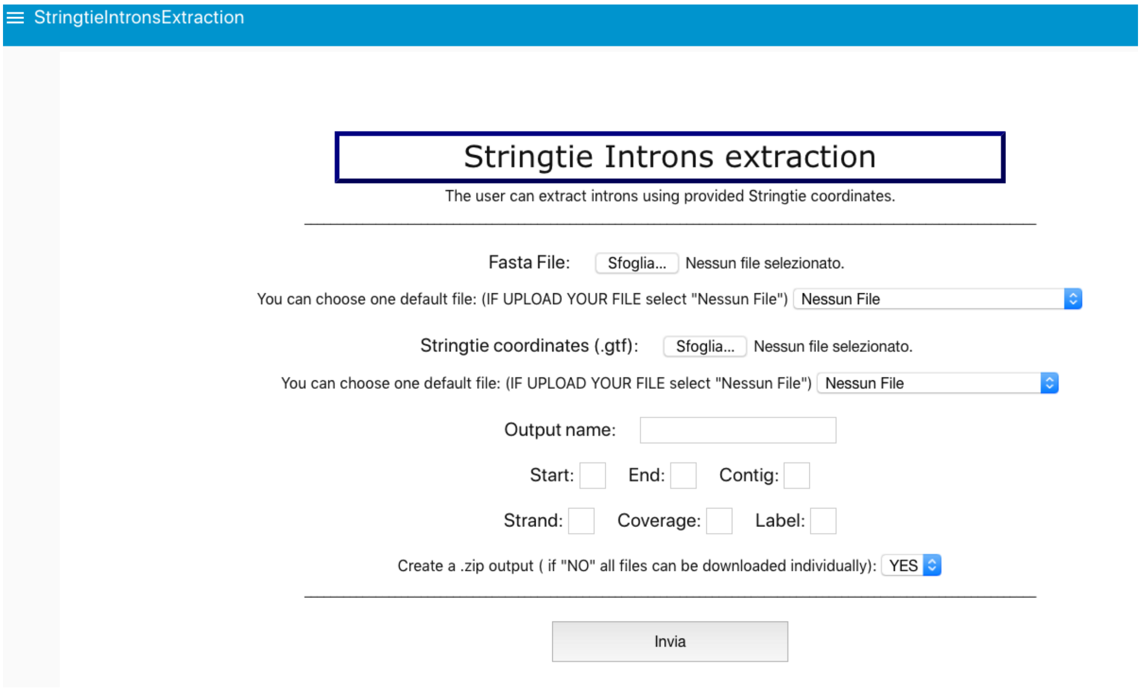
You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Output name:

Bb extension to flanking regions 3'/5':

Create a .zip output (if "NO" all files can be downloaded individually): YES

Fig.22*: Section 7. Augustus Introns Extraction: The user can extract introns using provided Augustus (.gtf) coordinates and choose to extent the intron flanking regions (5'-; 3'-) of a desired length*



StringtieIntronsExtraction

Stringtie Introns extraction

The user can extract introns using provided Stringtie coordinates.

Fasta File: Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Stringtie coordinates (.gtf): Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

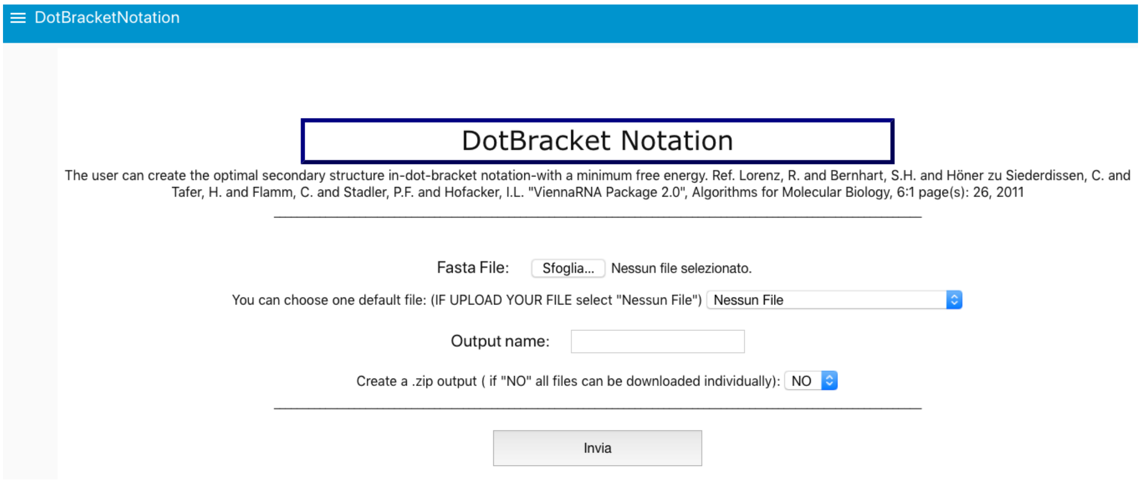
Output name:

Start: End: Contig:

Strand: Coverage: Label:

Create a .zip output (if "NO" all files can be downloaded individually): YES

Fig.23*: Section 8. Stringtie Introns Extraction: the user can extract introns using provided Stringtie (.gtf) coordinates*



DotBracketNotation

DotBracket Notation

The user can create the optimal secondary structure in-dot-bracket notation-with a minimum free energy. Ref. Lorenz, R. and Bernhart, S.H. and Höner zu Siederdissen, C. and Tafer, H. and Flamm, C. and Stadler, P.F. and Hofacker, I.L. "ViennaRNA Package 2.0", Algorithms for Molecular Biology, 6:1 page(s): 26, 2011

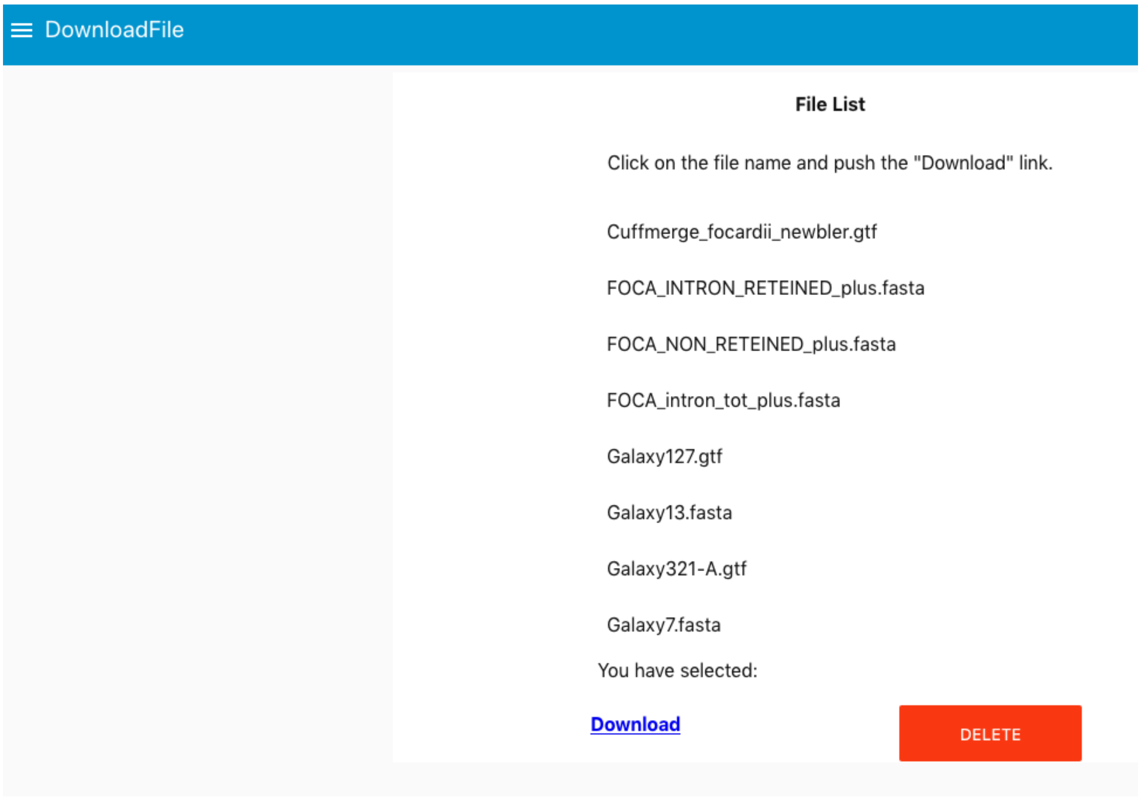
Fasta File: Sfoglia... Nessun file selezionato.

You can choose one default file: (IF UPLOAD YOUR FILE select "Nessun File") Nessun File

Output name:

Create a .zip output (if "NO" all files can be downloaded individually): NO

Fig.24*: Section 9. DotBracket Notation: the user can convert the RNA primary structure (RNA sequence) in the secondary structure with the Dot-Bracket notation*[87]



File List

Click on the file name and push the "Download" link.

Cuffmerge_focardii_newbler.gtf

FOCA_INTRON_RETEINED_plus.fasta

FOCA_NON_RETEINED_plus.fasta

FOCA_intron_tot_plus.fasta

Galaxy127.gtf

Galaxy13.fasta

Galaxy321-A.gtf

Galaxy7.fasta

You have selected:

Fig.25: *Section 10. Download File: the user can download all the data produced from the previous platform instances*

3.6.3 Step 1: Data selection

Genome data of *Tetrahymena* species and *Euplotes vannus* was obtained from the *Tetrahymena* Genome Database Wiki, a user-updatable database of information about the genes, proteins, and genomes of ciliate organisms, as determined by The Institute for Genomic Research (TIGR) and Ocean University of China[77].

Paramecium tetraurelia genome was generated by Saudemont et. al[79]. All datasets are available at http://doi.org/10.5281/zenodo.321639 [52]

Euplotes focardii genome assembly is available at NCBI with accession number CAAL01000000[80]. My research group produced the transcript reads.

All transcriptome data can be downloaded from the EMBL-EBI site using the IDs in chapter 3.7.1 Genomes and Transcriptomes (table 2).

3.6.4 Step 2: The Galaxy Europe Platform

Genome and transcriptome files in different formats (.FASTA, .FASTQ) were imported into the platform and different tools were used for each organism and its various transcriptome replicates.

3.6.4.1 Data Cleaning and Mapping

To remove low-quality and adaptor sequences within the transcriptome reads, Trimmomatic v0.32[80] tool was employed for cleaning raw reads. Trimmomatic is a fast, multithreaded command line tool that can be used to trim and crop Illumina (FASTQ) data. There are two major modes of the program: Paired end mode and Single end mode. Paired end mode maintains the correspondence of read pairs and also use the additional information contained in paired reads to better find adapter or PCR primer fragments introduced by the library preparation process. This tool works with FASTQ files (using phred + 33 or phred + 64 quality scores, depending on the Illumina pipeline used) and performs a variety of useful trimming tasks. I used the settings below:

(i) Trimming adapter fragments off raw sequence reads.

(ii) Drop the read if the average quality is below the specified level phred (>30%)[81].

(iii) Cut the read to a specified length by removing bases from the end (45bp).

These trimming steps should ensure all clean reads without low-quality bases left for downstream analyses.

The clean sequencing data were mapped to the organism reference genome using RNA STAR (v.2.4.0d-2)[43] with default parameters. After the mapping step I obtained for every transcriptome a .BAM file (data reads aligned to an assembly) and a .BED file (index).

3.6.4.2 Transcriptome assembly

For each BAM file after reads mapping, I independently assembled transcriptome states using Cufflinks v2.2.1.3[51]. It is known that ciliates and protozoa introns are averagely shorter than their animal counterparts. I found introns within the range from 40 to 150 bp, so I adjusted the parameter -I (-max-intron-length) from default 30,000 to 4,000 for Cufflinks. Meanwhile, the parameter -u (-multi-read-correct) was utilized to weigh reads mapping to multiple locations in the genome, and only highest-ranking alignments were reported.

Cufflinks assembles individual transcripts from RNA-seq reads that have been aligned to the genome. Because a sample may contain reads from multiple splice variants for a given gene, Cufflinks must be able to infer the splicing structure of each gene. However, genes sometimes have multiple AS events, and there may be many possible reconstructions of the gene model that explain the sequencing data. In fact, it is often not obvious how many splice variants of the gene may be present. Thus, Cufflinks reports a parsimonious transcriptome assembly of the data. The algorithm reports as few full-length transcript fragments or ‘transfrags’ as are needed to ‘explain’ all the splicing event outcomes in the input data.

After the assembly phase, Cufflinks quantifies the expression level of each transfrag in the sample. This calculation is made using a rigorous statistical model of RNA-seq and is used to filter out background or artifactual transfrags. For example, with current library preparation protocols, most genes generate a small fraction of reads from immature primary transcripts that are generally not interesting. As these transfrags are typically far less abundant in the library than the mature, spliced transcripts, Cufflinks can use its abundance estimates to automatically exclude them. Given a sample, Cufflinks can also quantify transcript abundances by using a reference annotation rather than assembling the reads.

3.6.4.3 Transcriptome Isoforms

Cuffmerge[51] was employed to remove the redundant isoforms in different samples. Cuffmerge is essentially a meta-assembler. It treats the assembled transfrags the way Cufflinks treats reads, merging them together parsimoniously. Furthermore, when a reference genome annotation is available, Cuffmerge can integrate reference transcripts into the merged assembly. It performs a reference annotation-based transcript assembly to merge reference transcripts with sample transfrags and produces a single annotation file for use in downstream differential analysis. Figure 24 shows an example of the benefits of merging sample assemblies with Cuffmerge.

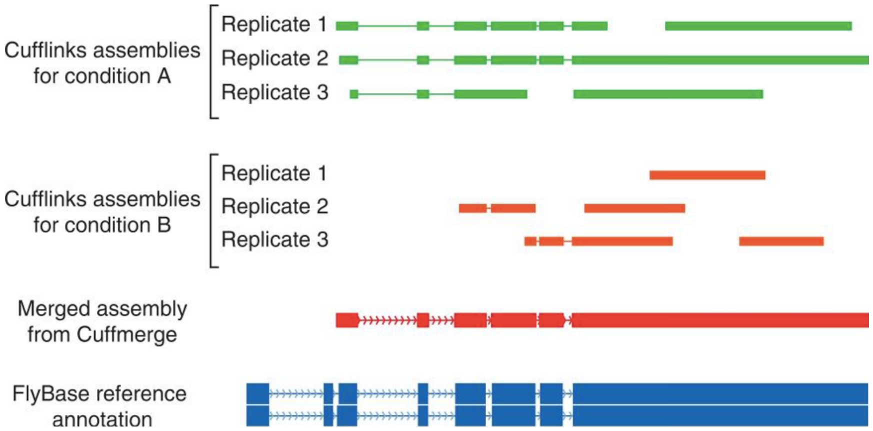

Cufflinks assemblies for condition A
Replicate 1
Replicate 2
Replicate 3
Cufflinks assemblies for condition B
Replicate 1
Replicate 2
Replicate 3
Merged assembly from Cuffmerge
FlyBase reference annotation

Fig.26: *Cuffmerge operating mode*[51]

Genes with low expression may receive insufficient sequencing depth to permit full reconstruction in each replicate. However, merging the replicate assemblies with Cuffmerge often recovers the complete gene. Newly discovered isoforms are also integrated with known ones at this stage into more complete gene models.

Once each sample has been assembled and all samples have been merged, the final assembly can be screened for genes and transcripts that are differentially expressed or regulated between samples.

3.6.5 Step 3: The Nextflow Framework

3.6.5.1 Total Introns Extraction

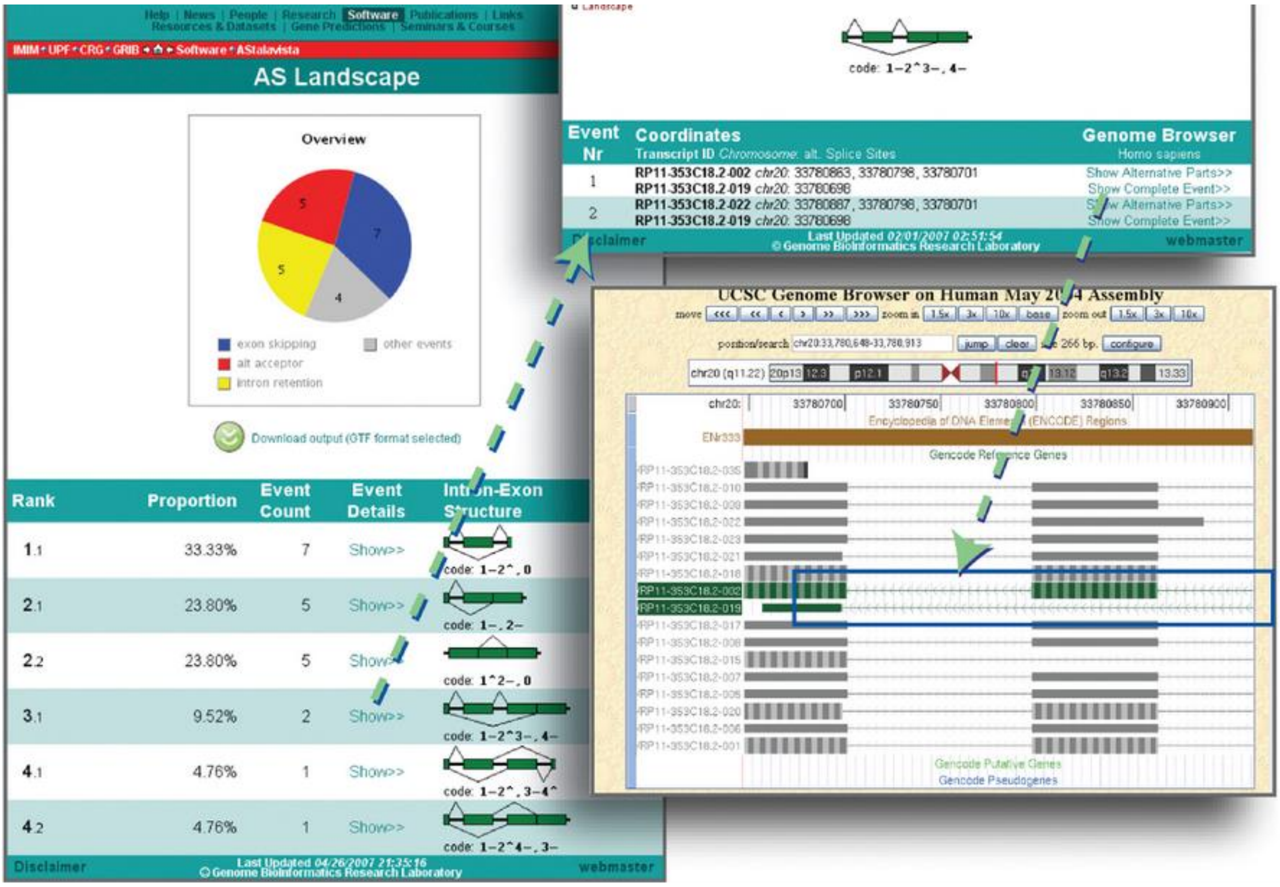
This task start from the Cuffmerge Galaxy output, remove duplicate exons and create a .FASTA file representing the sequence of all introns (5'-3').

3.6.5.2 Differentiate Retained from Constitutively Spliced Introns

This task creates two .FASTA file, respectively retained and constitutively spliced introns after cleaning them from outliers.

The retained intron sequences were generated using the ASTALAVISTA and iREAD .gtf output. During this process all introns were flipped to 5'-3' and only canonical introns were filtered.

The first tool used was ASTALAVISTA (Fig 27) an algorithm by Foissac and Sammeth[52]. It can identify all the AS events from a Cuffmerge .gtf input. For our purposes, only IR events were examined. The RIs can be directly identified by the record code of AS event. The AS code of IRevents is 1^2-,0.



Rank	Proportion	Event Count	Event Details	Intron-Exon Structure
1.1	33.33%	7	Show>>	code: 1–2^, 0
2.1	23.80%	5	Show>>	code: 1–, 2–
2.2	23.80%	5	Show>>	code: 1^2–, 0
3.1	9.52%	2	Show>>	code: 1–2^3–, 4–
4.1	4.76%	1	Show>>	code: 1–2^, 3–4^
4.2	4.76%	1	Show>>	code: 1–2^4–, 3–

Event Nr	Coordinates Transcript ID *Chromosome*: alt. Splice Sites	Genome Browser Homo sapiens
1	RP11-353C18.2-002 *chr20*: 33780863, 33780798, 33780701 RP11-353C18.2-019 *chr20*: 33780698	Show Alternative Parts>> Show Complete Event>>
2	RP11-353C18.2-022 *chr20*: 33780887, 33780798, 33780701 RP11-353C18.2-019 *chr20*: 33780698	Show Alternative Parts>> Show Complete Event>>

Fig.27: *Astalavista operating mode*[52]

The second tool was iREAD[82] (Fig 28). It takes as input these files:

1) the organism genome (.FASTA)

2)BAM and a BED file that is generated by aligning reads to a reference genome using STAR in the Galaxy Europe Platform. The BAM file needs to be sorted by coordinates (the default of STAR) and to be indexed (can be done using the 'samtools index' command).

3) a text file containing the coordinates of independent introns that do not overlap with any exons of any other isoforms or genes generated after the the Cuffmerge (.GTF) being processed by section 2 Biounicam script (Fig.17) in Nextflow.

To obtain the constitutively spliced introns I implemented the section 4 Biounicam script (Fig.19) in Nextflow able to subtract the total introns from the cuffmerge files the retained introns obtained with the two tools.

The data was checked for outliers via box and whisker plots. The observations that were beyond the range of ±3 standard deviation were deemed as outliers and were accordingly removed.

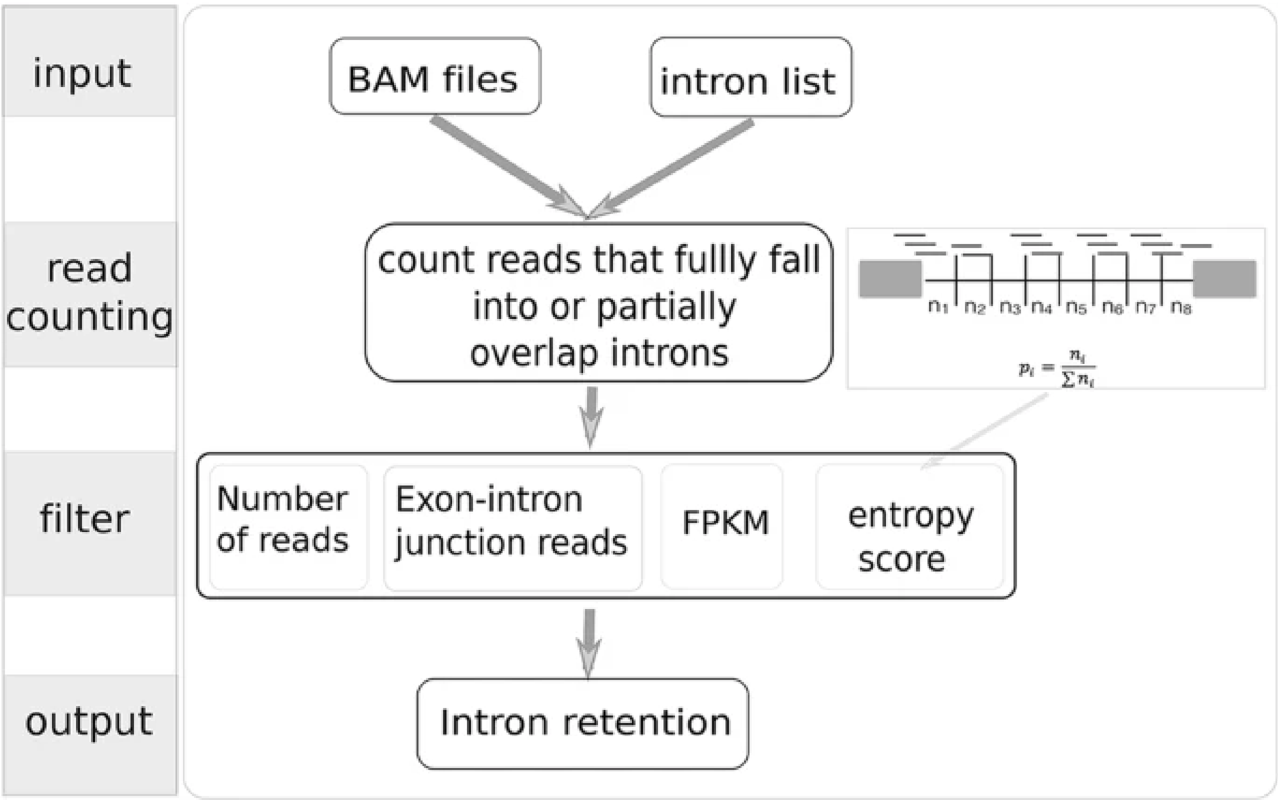

input
read counting
filter
output
BAM files
intron list
count reads that fullly fall into or partially overlap introns
n_1 n_2 n_3 n_4 n_5 n_6 n_7 n_8
$p_i = \frac{n_i}{\sum n_i}$
Number of reads
Exon-intron junction reads
FPKM
entropy score
Intron retention

Fig.28: *iRead operating mode*[82]

3.6.5.3 Feature Extraction

Retained introns are generally smaller in length, have weaker splice sites and are more G/C rich. Braunchweigh et al.[83] presented an 'IR code' in which they combined features for the task of predicting percentage intron retention. More recently, Mao et al. did a similar study, differentiating retained and non-retained introns in *Arabidopsis thaliana*[84]. The feature set used by our models is mainly based on these two works, but I added some new features never used in literature in order to test if they can higher the performances of the models like the estimation of Entropy or a Metric of Complexity. In the next future, I will add some new features to better understand the secondary structure involvement into the splicing mechanism. A tabular data-frame, with all features, was created for every retained and non-retained couple of FASTA file for every organism.

I used the following features:

- Intron length
- Percentage of normalized AG, AC, AT, GC, GT, CT pairs considering the shortest intron sequence length[85]. I determined local features of segmental nucleotides composition, which provide crucial complementary to the global features and are defined as segmental probabilities of four nucleotides correlation factors (ΦAG, ΦAT, ΦGC, ΦGT, ΦCT)
- To estimate the entropy of nucleotide sequences, I used the Dirichlet-multinomial pseudo-count entropy estimator[86], a Bayesian plug-in estimator.
- The script in Biounicam section 9 (fig.24) was used in Nextflow to determine the intron secondary structure with RNAlib-2.4.16[87] in Dot Bracket Notation. From the structure was calculated the sequent features:
    - Entropy value of intron's global and local secondary structure[86]
    - Number specific secondary structure element (Hairpin, Internal Loop, Stem, Multi-branch loop) with specific length[87]

  The RNA structures were folded using RNAfold[88] from the Vienna package with default parameters. The standard representation of a secondary structure in this library is the Dot-Bracket Notation (a.k.a. Dot-Parenthesis Notation), where matching brackets symbolize base pairs and unpaired bases are shown as dots. Based on that notation, more elaborate representations have been developed such loops, hairpins, stem-loops.

- The metric of complexity proposed by Lempel and Ziv (LZ)[89]. LZ has been used for complexity characterization of DNA sequences[89], recognition of structural regularities to characterize the responses of neurons[89], to develop new methods for discovering patterns in DNA sequences[90] and to estimate the entropy of neural discharges (spike trains)[91]. This complexity measure is related to the number of distinct substrings (i.e., patterns) and the rate of their occurrence along a given sequence[92]. LZ is calculated in two steps. First, the value of a given signal of length is binarized. The standard way of doing this is calculating its mean value and turning each data point above it to |1|s and each point below it to |0|s; as a second step, the resulting binary sequence is scanned sequentially looking for distinct structures or patterns, building up a dictionary that summarizes the sequences seen so far. Finally, the LZ index is determined by the length of this dictionary; i.e., is the number of distinct patterns found. Note that regular signals can be characterized by a small number of patterns and hence have low LZ complexity, while irregular signals require long dictionaries and hence have a high LZ complexity.

- The effective distance[93] is defined as the linear distance in nucleotides (nt) after removing the secondary structure. More specifically, removing all the bases that are part of a structured region and keeping the 2 bases corresponding to the beginning and the end of the structured region. The simplest way of calculating the effective distance between two signals in the RNA is to predict the minimum free energy structure and calculate the distance between them after discarding the positions included within the secondary structure.

- Accessibility of splicing signals[93]: when secondary structures are placed overlapping cis elements in the sequence, they can hinder the recognition of these elements by other proteins or RNAs. Therefore, when measuring the ability to recognize a signal in an RNA molecule such as a splice site, I will have to measure its accessibility, i.e. whether the signal will be available to other proteins or will be hidden by an RNA structure.

- Branch point position from 5'- within the intron.

3.6.5.4 Data preprocessing

Data preprocessing pipeline is shown in Figure 29. I randomly divided the database in a training set (70%), and a test set (30%) to evaluate the predictive models. The training set were used to build the classification algorithms using gradient boosting C5.0[61], NNs[94] and SVM[61] and ADAboost[95]. The test set was used to evaluate the models.

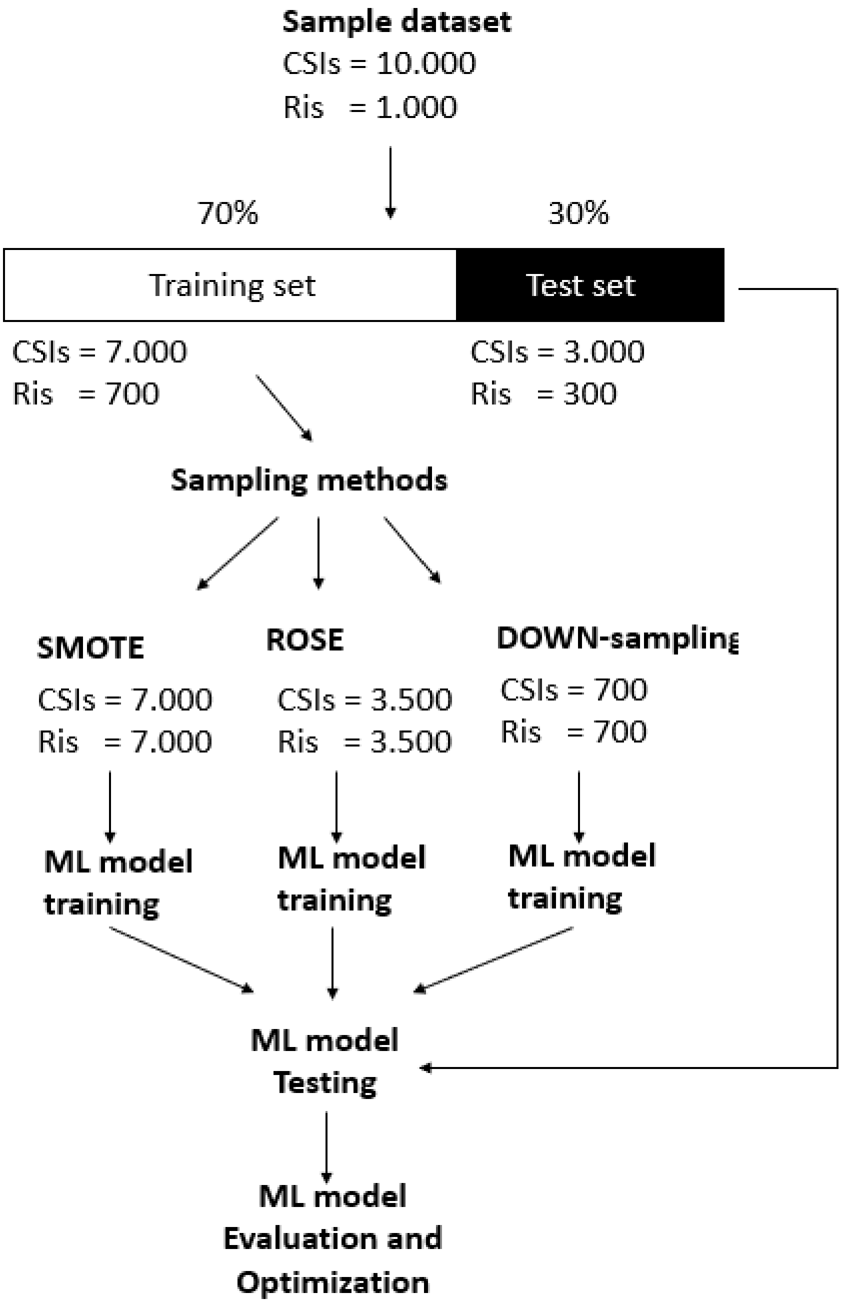


Fig.29: *Data preprocessing pipeline. The numbers shown refer to an example dataset*

Often real-world data sets are predominately composed of "normal" examples (spliced introns) with only a small percentage of "abnormal" (retained introns) examples. Evidence demonstrated that the class imbalance, which is just the situation in our sample, could substantially affect the performance of the method used.

A dataset is imbalanced if the classification categories are not approximately equally represented. It is also the case that the cost of misclassifying an abnormal example as a normal example is often much higher than the cost of the reverse error.

With our tabular data-frames the numerical difference of the two classes was approximately 1 to 10. I applied different approaches in Nextflow to solve this problem. All these approaches can be classified into three strategies: data level, algorithm level and ensemble-based strategy.

I used three classes of sampling only in the training set: under-sampling, ROSE and SMOTE. Under-sampling of the majority (normal) class has been proposed as a good means of increasing the sensitivity of a classifier to the minority class. SMOTE[96] shows that a combination of over-sampling the minority (abnormal) class and under-sampling the majority (normal) class can achieve better classifier performance (in ROC space) than only under-sampling the majority class. ROSE[97] (Random Over-Sampling Examples) generates synthetic balanced samples and thus allows to strengthen the subsequent estimation of any binary classifier. ROSE is a bootstrap-based technique which aids the task of binary classification in the presence of rare classes. It handles both continuous and categorical data by generating synthetic examples from a conditional density estimate of the two classes.

I used the Caret v6.0–82[96] and the GA (Genetic Algorithm optimization) v3.2 package[96] to automatically tune the optimal combinations of model parameters for the four Machine Learning algorithms I choose, aiming to achieve a better prediction performance.

The caret package, short for classification and regression training, contains numerous tools for developing predictive models using the rich set of models available in R. The package focuses on simplifying model training and tuning across a wide variety of modeling techniques. It also includes methods for pre-processing training data, calculating variable importance, and model visualizations (boxplot). Variable importance evaluation functions can be separated into two groups: those that use the model information and those that do not. I used a model-based approach more closely tied to the model performance and may be able to incorporate the correlation structure between the predictors into the importance calculation. Regardless of how the importance is calculated, for most classification models, each predictor will have a separate variable importance for each class. All measures of importance can be scaled to have a maximum value of 100.

3.6.5.5 Machine Learning Algorithms

C5.0 is an algorithm used to generate a decision tree developed by Ross Quinlan[61]. C5.0 is an extension of Quinlan's earlier ID3 algorithm. The decision trees generated by C5.0 can be used for classification, and for this reason, C5.0 is often referred to as a statistical classifier.

C5.0 builds decision trees from a set of training data in the same way as ID3 algorithm, using the concept of Entropy (Information Theory).

At each node of the tree, C5.0 chooses the attribute of the data that most effectively splits its set of samples into subsets enriched in one class or the other. The splitting criterion is the normalized information gain (difference in entropy). The attribute with the highest normalized information gain is chosen to make the decision. The C5.0 algorithm then recurses on the partitioned sublists.

This algorithm has a few base cases.

- All the samples in the list belong to the same class. When this happens, it simply creates a leaf node for the decision tree saying to choose that class.
- None of the features provide any information gain. In this case, C5.0 creates a decision node higher up the tree using the expected value of the class.
- Instance of previously-unseen class encountered. Again, C5.0 creates a decision node higher up the tree using the expected value.

AdaBoost combines a set of weak learners in order to form a strong classifier in a "greedy fashion"[98] , it always chooses the weak classifier with the lowest error, ignoring all others. A weak learner is any classifier such that at time t, ∊t < 0.5. It uses a decision stump because it is fast and gives a one-to-one relationship between a feature and a weak learner. The threshold is chosen such that the minimum error rate using feature t is achieved for weak learner h

$$h_t(x) = \begin{cases} +1 & if x \geq thereshold \\ -1 & otherwise \end{cases}$$

AdaBoost explicitly seeks to minimize the error according to a distribution of weights, Dt, a teach iteration. However, if we follow the logic and view as a vector of coordinates, $\alpha^{\rightarrow}$, then it can be rewrite f(x) as:

$$f(x) = \frac{\vec{\alpha} \cdot \vec{h}(x)}{\|\vec{\alpha}\|_1}$$

Here it can be view $\alpha^{\rightarrow}$ as a hyperplane and as the margin. AdaBoost explicitly minimizes the error, and implicitly maximizes the margin according to the l1 −norm at each iteration, causing it to generalize well. Because AdaBoost greedily selects features, it can take a complicated problem, one composed of many features, and create a sparse classification rule, one composed of only a few features. However, this is also a drawback. Due to the greedy nature of AdaBoost it can only minimize the error, and maximize the margin with respect to features that have already been selected. AdaBoost is also limited by the fact that it can only combine weak learners by adding them together. AdaBoost approximates the Bayesian posterior distribution by incrementally adding new weak learners (hi(x)) at each iteration. This is equivalent to formulating the overall classifier at time t as H(x) = sign[P(y = ±1|h1(x) ···ht(x) > 0.5)] [50]. If we let h1(x) ···ht(x) = ht, we can formulate the posterior distribution as:

$$P(y = \pm 1 | h_t) = \frac{P(h_t | y = \pm 1) P(y = \pm 1)}{P(h_t)}$$

The denominator is again a constant and P(y = ±1) is a shape model which must be integrated later. In this formulation, AdaBoost also approximates the ideal Bayesian distribution after a long enough t, drawing features from a very large feature pool.

We could stop here and just apply an ideal Bayesian classifier to the features selected by AdaBoost. For problems with a large number of i.i.d. examples that lie in a low-dimensional space, this would be ideal. However, our problem lies in a high-dimensional space, meaning that it would require a large number of i.i.d. examples for the Bayesian classifier to generalize well. Although we do have many examples, they are all correlated (non-i.i.d) and therefore the ideal Bayesian classifier would most likely be memorizing the posterior probability P(x1 ··· xt|y = ±1), resulting in poor generalization.

Support Vector Machine (SVM) [61] is one of the most widely used classification method, specifically designed for binary classification problems. Among other things, SVM owns its popularity to its ability to study complex nonlinear classification problem by solving a convex quadratic optimization problem. Let be a generic dataset of *N* points where $x^i \in R^n$ and y_i respectively define an input vector (or feature vector) and its associated label, which may only assume the two values +1 and -1, indicating the class to which the input belongs. SVM central idea is the construction of a hyperplane with maximum margin of separation between the two classes. Such optimal hyperplane is represented by the following equation

$$w^T x + b = 0$$

In Equation (1) w is a weight vector of the same dimension of $x^i, \forall i = 1, \dots, N$ and $b \in R$ is a threshold value. The goal of SVM would be to perfectly divide the points belonging to the two different classes. Anyway, based on the dataset in use, this is not always possible in the input space. To overcome this difficulty, SVM makes use of the so-called kernel trick: training data is nonlinearly mapped into a higher dimensional space (i.e., feature space) through a function called feature mapping. Since the algorithm can be written entirely in terms of the inner products of the features, it is not required to know the feature mapping but only the inner products (in the feature space) $\langle \Phi(x^i), \Phi(x^j) \rangle \forall i, j = 1, \dots, N$. Specifically, given a function Φ, the corresponding kernel is defined as

$$k(x^i, x^j) = \langle \Phi(x^i), \Phi(x^j) \rangle$$

Note that, in many cases the quantities $k(x^i, x^j)$ may be computationally inexpensive to calculate, even if Φ is a very complex high dimensional function.

Among the most popular kernel functions we find the polynomial kernel $k(x^i, x^j) = \langle x^i, x^j \rangle^d$, the radial basis function (RBF) kernels $k(x^i, x^j) = e^{-\vee x^i - x^j \vee\ ^2/2\sigma^2}$ and the sigmoid kernel $k(x^i, x^j) = \tanh(\gamma \langle x^i, x^j \rangle + r)$ with σ, γ and r kernel parameters.

Artificial Neural Network is a Machine Learning technique aiming to reproduce the behavior of human brain where neural cells (i.e., neurons) receive, process and transmit external data with each other. An artificial neuron is very similar to the physical one and is formed by three parts: the summing function, the activation function and the output function. In the summing function the inputs are associated with scalar weights and summed, such sum is later on compared to a threshold value. If the computed value is greater than the threshold value, the neuron is activated and an output, defined by the activation function is sent to the other connected neurons, otherwise the neuron is not activated. Multilayer perceptron (MLP) is one of the most popular NN structure, where nodes are organized in three or more layers, i.e., a collection of nodes operating together at a specific depth: an input layer, one or more hidden layers and an output layer. At the beginning of the algorithm, all weights are randomly chosen. As the algorithm proceeds, the weights are modified in order to achieve the best agreement between computed and expected output. NN is a widely used algorithm mainly in image classification problems because of its performance. However, unlike other classical models as SVM or linear regression where the user can actually look inside the algorithm to understand how and how well it is working, using NNs makes it near impossible to know how the structure is actually working[8]. The user is aware that the model is some non-linear combination of some neurons, but it is hard to determine what each neuron is doing. For this reason, NNs are also known as "black boxes".

3.6.5.6 Model validation

I did a 10-fold cross validation method with three repeats, which has been viewed as the de facto standard for estimating model performance[33]. In K Fold cross validation, the data is divided into k subsets. Now the holdout method is repeated k times, such that each time, one of the k subsets is used as the test set/ validation set and the other k-1 subsets are put together to form a training set. The error estimation is averaged over all k trials to get total effectiveness of our model. As can be seen, every data point gets to be in a validation set exactly once, and gets to be in a training set k-1 time. This significantly reduces bias as we are using most of the data for fitting, and significantly reduces variance as most of the data is also being used in validation set. Interchanging the training and test sets also adds to the effectiveness of this method.

3.6.5.7 Model evaluation

As performance measures I used accuracy, area under the ROC curve (AUC), sensitivity and specificity. To describe such performance for classification problem, it is essential to define a specific matrix, called confusion matrix, containing the number of false positives (FP), false negatives (FN), true positives (TP), and true negatives (TN). Specifically, a two-class (positive-negative) confusion matrix is a table where each row represents a predicted value and each column defines an actual value (or vice-versa): all correct predictions (TP and TN) are located in the matrix diagonal while the errors are given by all the elements outside the diagonal.

Accuracy (ACC) is a value that can be directly calculated from the confusion matrix and defines how often the classifier is correct

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{total}$$

To define AUC it is necessary to introduce the ROC curve (Receiver Operating Characteristic curve), namely a graph showing the performance of the classifier over all possible thresholds with respect to two parameters: the sensitivity also known as recall or true positive rate (TPR) and the false positive rate (FPR). The two quantities are defined as follows

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP+FN}}, \mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP+TN}}$$

Sensitivity is calculated as the ratio between the number of positive inputs correctly classified as positive (true positives) and the total number of positive data and measures how well the classifier made positive predictions based on all classes (i.e., it can be seen as the classifier ability to correctly detect positive inputs). FPR is calculated as the ratio between the number of negative inputs wrongly classified as positive (false positive) and the total number of negative data and measures the proportion of all the negative inputs who will be identified as positive.

AUC measures the area underneath the ROC curve: it has a range of values from 0 to 1. The area measures discrimination, that is, the ability to correctly classify random positive and negative data.

Specificity also known as true negative rate (TNR) is defined as

$$\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}},$$

is calculated as the ratio between the number of negative inputs correctly classified as negative (true negatives) and the total number of negative data and measures how well the classifier made negative predictions based on all classes (i.e., it can be seen as the classifier ability to correctly detect negative inputs).

The Matthew Correlation Coefficient (MCC) is a binary classification model evaluator which can directly be computed using confusion matrix elements as follows:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Once introduced the Precision value as the ratio of correctly predicted positive observations with respect to the total positive ones, F_1 score is given by

$$F_1 = \frac{2(Sensitivity \times Precision)}{Sensitivity + Precision}$$

representing the harmonic mean between sensitivity and recall values.

Still using Precision and Sensitivity (Recall) values, the precision-recall curve simply plots precision (x-axis) and recall (y-axis) for several possible values of threshold.

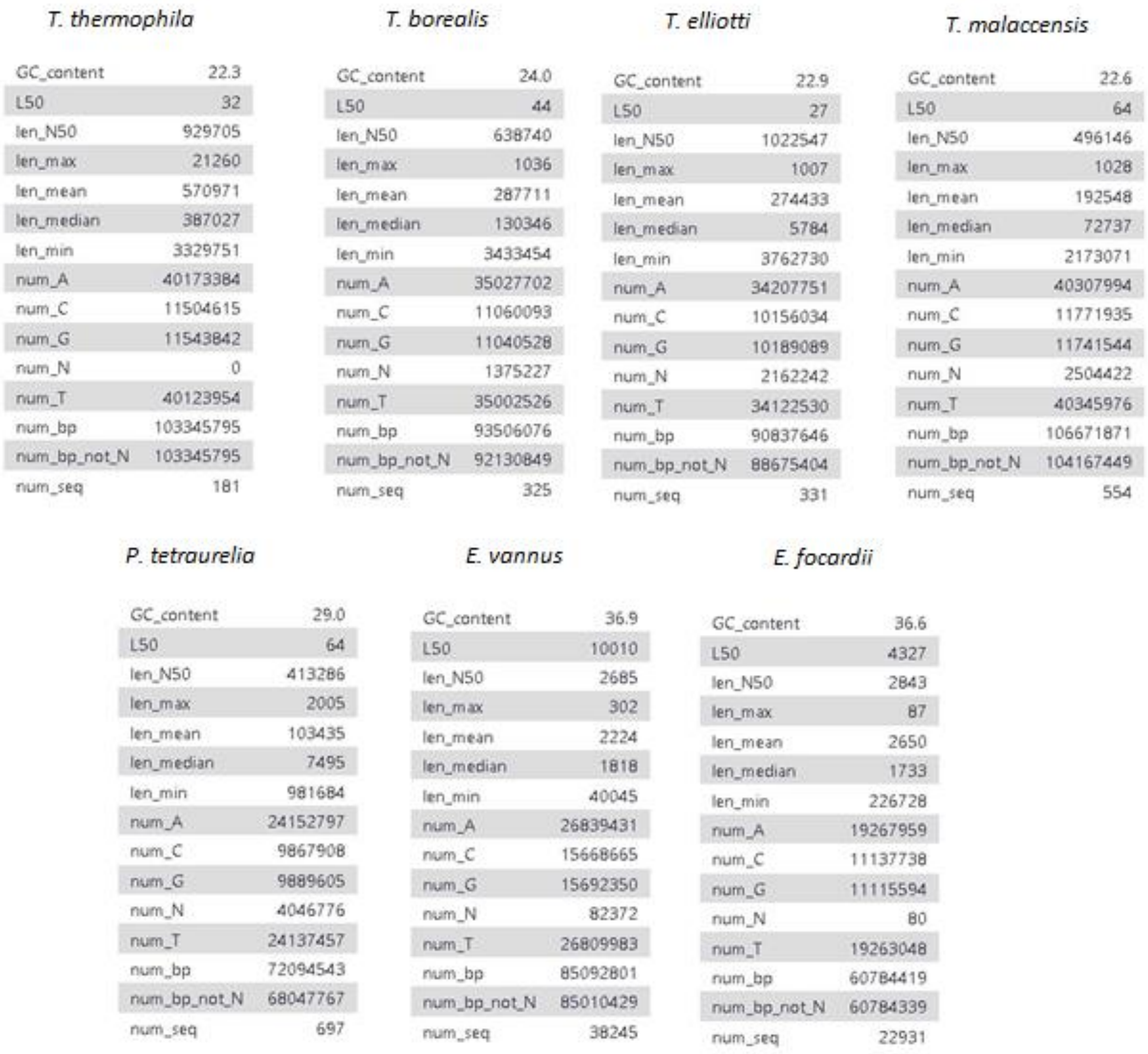
3.6.6 Gene Ontology

Gene Ontology (GO) analysis was performed by an enrichment evaluation using the Fisher’s Exact Test available of the Blast2GO software[83]. The distributions of *Tetrahymena* species genomic and retained introns GO terms were compared considering all the GO categories (molecular function, biological process and cellular component) using a p-value threshold of 0.05 and a multiple testing correction of false discovery rate (FDR) as p-value filter mode.

3.7 Results and Discussion

3.7.1 Genomes and Transcriptomes

To build the IR prediction models for I used seven ciliate genomes: four from *Tetrahymena* species, two from *Euplotes* species, and one *Paramecium* species. The lowest GC content was in *Tetrahymena* species (between 22.6 and 24.0%). *Paramecium tetraurelia* had a GC content of 29%, while the highest GC content was in the *Euplotes* species (about 36%). *Tetrahymena* species had the longest mean assembly length (>190k bp) while *Euplotes* species had the shortest (<1.8k bp). *P. tetraurelia* had a slightly greater assembly average length than the *Euplotes* species (7.4k bp). Genomes' details are represented in Table 4.



	T. thermophila	*T. borealis*	*T. elliotti*	*T. malaccensis*
GC_content	22.3	24.0	22.9	22.6
L50	32	44	27	64
len_N50	929705	638740	1022547	496146
len_max	21260	1036	1007	1028
len_mean	570971	287711	274433	192548
len_median	387027	130346	5784	72737
len_min	3329751	3433454	3762730	2173071
num_A	40173384	35027702	34207751	40307994
num_C	11504615	11060093	10156034	11771935
num_G	11543842	11040528	10189089	11741544
num_N	0	1375227	2162242	2504422
num_T	40123954	35002526	34122530	40345976
num_bp	103345795	93506076	90837646	106671871
num_bp_not_N	103345795	92130849	88675404	104167449
num_seq	181	325	331	554

	P. tetraurelia	*E. vannus*	*E. focardii*
GC_content	29.0	36.9	36.6
L50	64	10010	4327
len_N50	413286	2685	2843
len_max	2005	302	87
len_mean	103435	2224	2650
len_median	7495	1818	1733
len_min	981684	40045	226728
num_A	24152797	26839431	19267959
num_C	9867908	15668665	11137738
num_G	9889605	15692350	11115594
num_N	4046776	82372	80
num_T	24137457	26809983	19263048
num_bp	72094543	85092801	60784419
num_bp_not_N	68047767	85010429	60784339
num_seq	697	38245	22931

Table.4*: Genomes overview. Statistics were made with the FASTAstatistics tool*[77]

To map each genome, I decided to use at least four transcriptomes. The transcriptomes may derive from different cell physiological states (starvation, conjugation and growth) or all from the same state. For *Euplotes focardii*, the organism studied by my research group, only two transcriptomes were available. In table 5 there are all the transcriptomes' details including the accession numbers and the Percentage of aligned mapped reads (%), which was acceptable for all transcriptomes after the RNA STAR mapping (>90% of aligned read) except for *E. focardii* in starvation (79.45%). This poor result may be caused by bacterial contamination during transcriptome extraction or a low yield of the extraction protocol.

Organism	ID	Condition	Sequencing instrument	number of reads (bp)	Sequencing types	The Percentage of aligned mapping (%)
Tetrahymena thermophila	SRR636695	Growth-m	Illumina Analyzer	30812705	Paired	96.75
	SRR636696	Starvation-3h-VI	Illumina Analyzer	15978436	Paired	97.37
	SRR636697	Starvation-3h-V	Illumina Analyzer	8939127	Paired	90.39
	SRR636698	Starvation-15h-VI	Illumina Analyzer	8254521	Paired	85.35
	SRR636699	Conjugation-2h	Illumina Analyzer	16329020	Paired	97.05
	SRR636700	Conjugation-8h	Illumina Analyzer	13692298	Paired	92.19
Tetrahymena borealis	SRR536859	Growth	Illumina HiSeq 2000	14915466	Paired	90.20
	SRR536858	Starvation	Illumina HiSeq 2000	16292687	Paired	90.35
	SRR505873	Growth	Illumina HiSeq 2000	14923292	Paired	91.02
	SRR505872	Starvation	Illumina HiSeq 2000	15857549	Paired	90.13
Tetrahymena elliotti	SRR536843	Starvation	Illumina HiSeq 2000	16317662	Paired	90.89
	SRR536842	Growth	Illumina HiSeq 2000	14946584	Paired	90.14
	SRR505875	Starvation	Illumina HiSeq 2000	15798072	Paired	91.25
	SRR505874	Growth	Illumina HiSeq 2000	14486996	Paired	90.26
Tetrahymena malaccensis	SRR505878	Growth	Illumina HiSeq 2000	14873192	Paired	90.85
	SRR505879	Starvation	Illumina HiSeq 2000	16801891	Paired	89.69
	SRR536826	Growth	Illumina HiSeq 2000	15072192	Paired	90.22
	SRR536827	Starvation	Illumina HiSeq 2000	16718449	Paired	90.54
Euplotes vannus	SRR7670786	Growth	HiSeq X Ten	20070546	Paired	90.63
	SRR7670788	Starvation	HiSeq X Ten	26097176	Paired	89.96
	SRR7670790	Growth	HiSeq X Ten	29030314	Paired	90.75
	SRR7670785	Starvation	HiSeq X Ten	23739691	Paired	90.67
Euplotes focardii	SRR1296783	Growth-m	Illumina HiSeq 2000	27839201	Paired	91.67
	SRR1296928	Starvation-6h	Illumina HiSeq 2000	14032283	Paired	79.45
Paramecium tetraurelia	ERR1661484	Growth-m	Illumina IIx	9383971	Unpaired	90.33
	ERR1661485	Growth-m	Illumina IIx	8875969	Unpaired	90.33
	ERR1676709	Growth-m	Illumina IIx	32147503	Paired	94.22
	ERR1676710	Growth-m	Illumina IIx	32103515	Paired	93.75

***Table.5**: List of transcriptome reads used for genome mapping*

3.7.2 Exons, CSIs and RIs

The number of exons, as well as the number of RIs and CSIs below, does not represent the actual number of exons that are in a unique organism transcriptome, but the sum in the various transcriptome states after merging them with Cuffmerge. I merged the transcriptomes to increase the number of RIs and CSIs found, in order to train more effectively our Machine Learning models. Therefore, the real numbers of exons, RIs and CSIs are slightly lower than the following values.

After assembly, merging and redundant isoforms removal using Cuffmerge of every transcriptome, I got the following exons results: the highest number of exons found was in *E. vannus* (about 200k). In *P. tetraurelia* and *T. thermophila* I found about 100k exons. For the other organisms, I obtained a number of exons less than 100k. Exons from all organisms had approximately the same average length, with a median between 230 and 270 bp. Only *E. vannus* had a shorter median length of about 150 bp. The average GC content of the exons was lowest in *Tetrahymena* species (about 27%), while the highest was in *E. vannus* (37.7%) which also differed from the other *Euplotes* species *E. focardii* (32.4%).

	Exon#	GC content (%)	Median
T.ter	118972	27.2	242
T.mal	99843	26.8	248
T.bor	79241	27.0	257
T.ell	94735	26.3	269
P.tet	114027	30.4	247
E.van	214072	37.7	152
E.foc	62995	32.4	229

***Table.6**: Exons overview after the transcriptome mapped reads assembly and merging.*

After label assignment (CSI or RI) by the two tools iREAD and ASTALAVISTA I obtained the following results.

Based on the intron length distribution generated by quantile in terms of the given probabilities (0.02, 0.2, 0.4, 0.6, 0.8, 0.98), 95% RIs and CSIs were found within the range from 20 to 160bp. This suggested that extremely large introns (>1000bp) and extremely small introns (less than 20 bp) became outliers, which would cause a negative effect on classification. Another filter we applied was to select introns based on the presence of branch-point[99]. Consequently, I filtered the introns dataset obtaining the numbers in Figure 30 and 31.

iREAD found more CSIs in all organisms than ASTALAVISTA, even significantly so as in *E. vannus* in which more than twice as many were labeled.

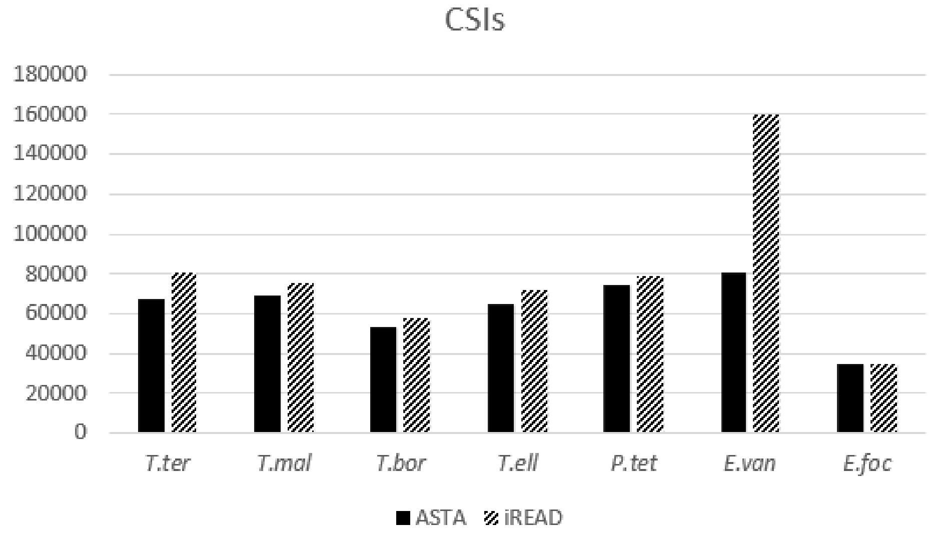


Fig.30*: Absolute number of CSIs found by ASTALAVISTA and iREAD*

The RIs pattern was different. iREAD found a higher number of retained introns in *T. malaccensis*, *T. elliotti* and *E. focardii*, while ASTALAVISTA in *T. thermophila*, *T. borealis*, *P. tetraurelia* and *E. vannus*. In this latter organism the differences were extremely marked as in the search for CSIs, in fact ASTALAVISTA labeled more than twice as many RIs.

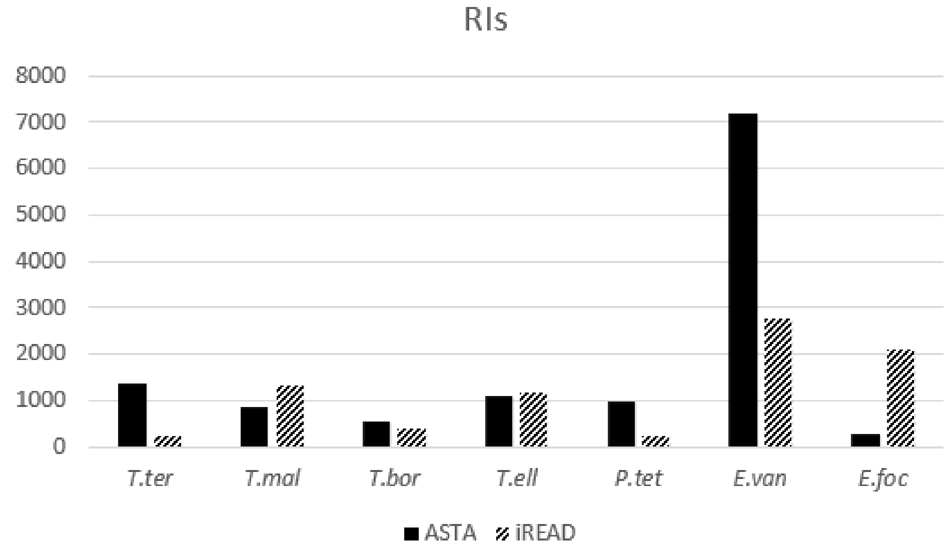


***Fig.31**: Absolute number of RIs found by ASTALAVISTA and iREAD*

If we examine the ratio between the RI and CSI, fairly consistent relationships between the two labelling tools were achieved in almost all ciliates. By contrast, in *Euplotes* species, I obtained discordant and unbalanced ratios. For example, ASTALAVISTA found more RI and less CSI than iRead in *E.vannus* with an unbalanced ratio towards IR (8.87%). Oddly, this unbalanced ratio disappears in *E.vannus* using iRead, but appears in *E.focardii* with a ratio of 6% RIs vs INR.

I found differences not only in the number of CSIs and RIs labeled by the two tools, but also in introns features. ASTALAVISTA found RIs to be generally shorter than CSIs and with lower GC content. In contrast, iREAD found completely opposite results, with longer RIs and higher GC content (Table 7). Previous reports discovered similar features as iREAD, including, lower AT content and higher GC content in Ris [100, 101].

ASTALAVISTA

	CSI#	GC_content	Median	RI#	GC_content	Median	RI vs CSI
T.ter	67051	19,3	69	1349	14,8	59	2,01
T.mal	69346	17,9	67	874	14,9	58	1,26
T.bor	53512	19,5	67	534	16,2	61	1,00
T.ell	64428	18,8	63	1074	15,7	58	1,67
P.tet	74279	28,3	25	983	23,7	25	1,32
E.van	81028	31,7	28	7190	32,3	26	8,87
E.foc	34999	23,2	29	263	23,7	27	0,75

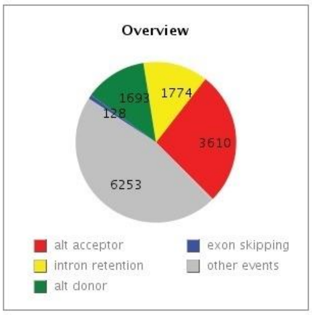
iREAD

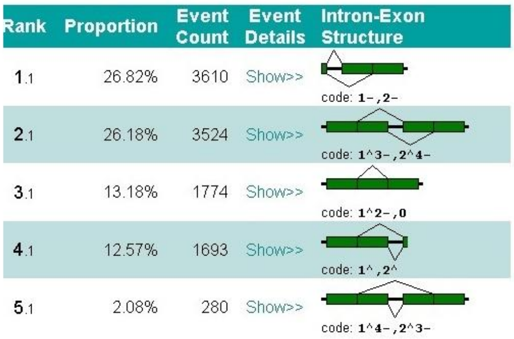
	CSI#	GC_content	Median	RI#	GC_content	Median	RI vs CSI
T.ter	80570	19,0	68	222	19,4	64	0,28
T.mal	75332	16,7	66	1330	17,7	67	1,77
T.bor	57899	18,8	67	372	19,5	75	0,64
T.ell	71853	17,5	63	1186	18,0	67	1,65
P.tet	79292	22,5	25	254	25,9	25	0,32
E.van	159488	31,6	27	2749	32,7	36	1,72
E.foc	35058	23,0	28	2119	24,5	31	6,04

Table.7*: Descriptive statistics of CSIs and RIs found by ASTALAVISTA and iREAD*

Finally, the ASTALAVISTA tool also provided us an overview of the AS that occurs in the studied organisms. The most frequent AS was intron retention followed by alt acceptor and alt donor. Only in *T. thermophila* the most frequent event was the alt acceptor (Fig.32).

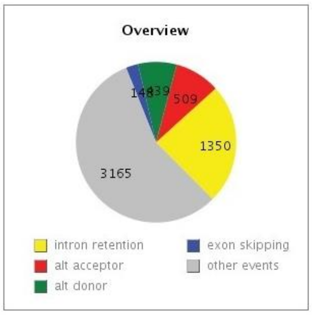
T.thermophila

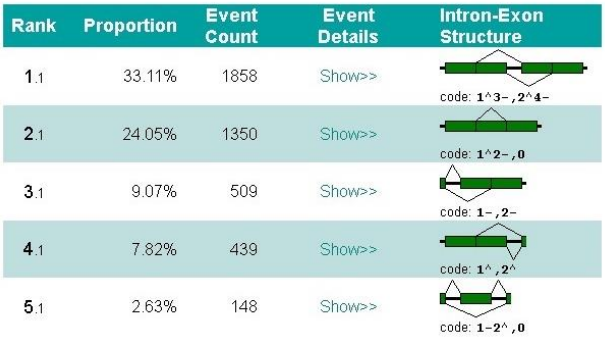




Rank	Proportion	Event Count	Event Details	Intron-Exon Structure
1.1	26.82%	3610	Show>>	code: 1-,2-
2.1	26.18%	3524	Show>>	code: 1^3-,2^4-
3.1	13.18%	1774	Show>>	code: 1^2-,0
4.1	12.57%	1693	Show>>	code: 1^,2^
5.1	2.08%	280	Show>>	code: 1^4-,2^3-

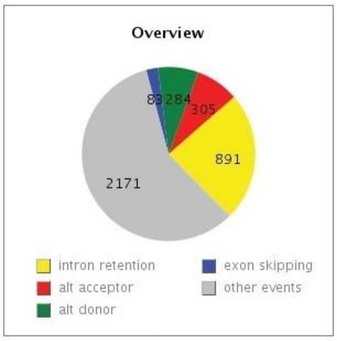
T. malaccensis

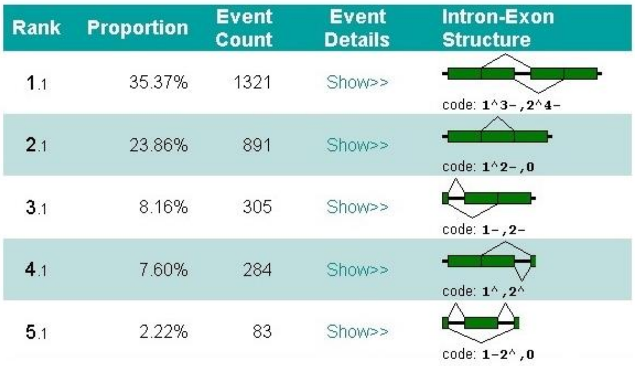




Rank	Proportion	Event Count	Event Details	Intron-Exon Structure
1.1	33.11%	1858	Show>>	code: 1^3-,2^4-
2.1	24.05%	1350	Show>>	code: 1^2-,0
3.1	9.07%	509	Show>>	code: 1-,2-
4.1	7.82%	439	Show>>	code: 1^,2^
5.1	2.63%	148	Show>>	code: 1-2^,0

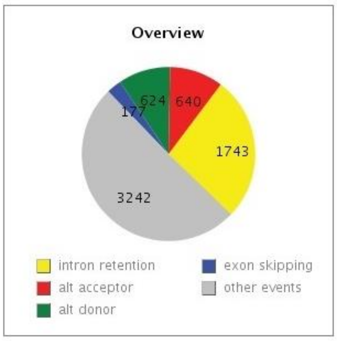
T. borealis

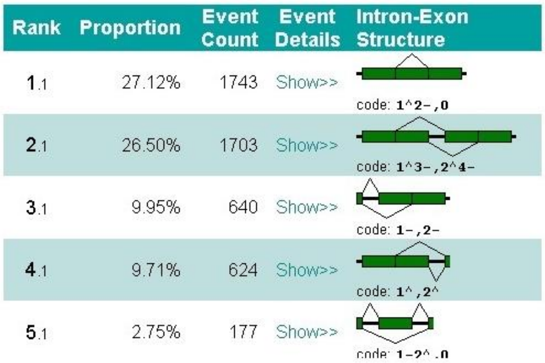




Rank	Proportion	Event Count	Event Details	Intron-Exon Structure
1.1	35.37%	1321	Show>>	code: 1^3-,2^4-
2.1	23.86%	891	Show>>	code: 1^2-,0
3.1	8.16%	305	Show>>	code: 1-,2-
4.1	7.60%	284	Show>>	code: 1^,2^
5.1	2.22%	83	Show>>	code: 1-2^,0

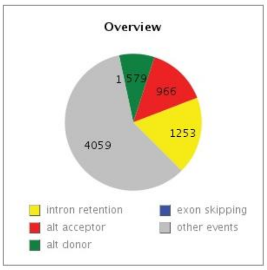
T. elliotti



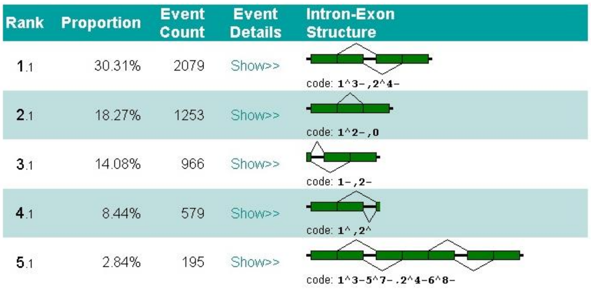


Rank	Proportion	Event Count	Event Details	Intron-Exon Structure
1.1	27.12%	1743	Show>>	code: 1^2-,0
2.1	26.50%	1703	Show>>	code: 1^3-,2^4-
3.1	9.95%	640	Show>>	code: 1-,2-
4.1	9.71%	624	Show>>	code: 1^,2^
5.1	2.75%	177	Show>>	code: 1-2^,0

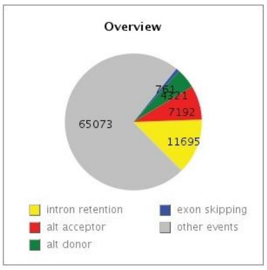
***Fig.32**: Alternative splicing landscape found by ASTALAVISTA in Tetrahymena species. The pie charts show the most represented alternative splicing isoforms. Intron retention code is 1^2-,0.*



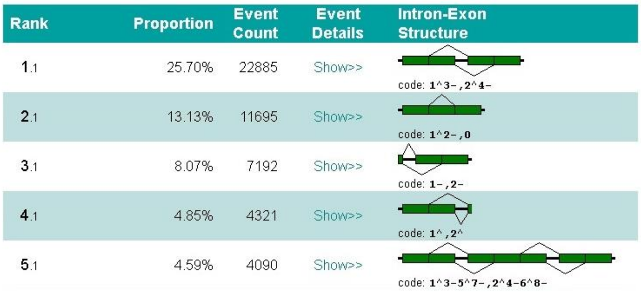
P. tetraurelia



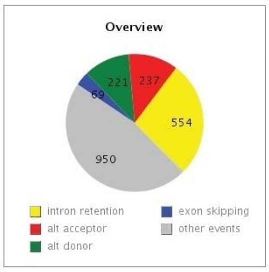
Rank	Proportion	Event Count	Event Details	Intron-Exon Structure
1.1	30.31%	2079	Show>>	code: 1^3-,2^4-
2.1	18.27%	1253	Show>>	code: 1^2-,0
3.1	14.08%	966	Show>>	code: 1-,2-
4.1	8.44%	579	Show>>	code: 1^,2^
5.1	2.84%	195	Show>>	code: 1^3-5^7-.2^4-6^8-



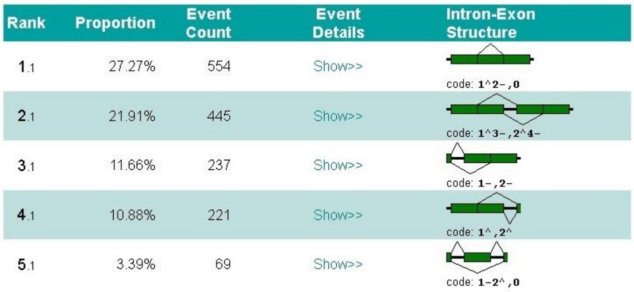
E. vannus



Rank	Proportion	Event Count	Event Details	Intron-Exon Structure
1.1	25.70%	22885	Show>>	code: 1^3-,2^4-
2.1	13.13%	11695	Show>>	code: 1^2-,0
3.1	8.07%	7192	Show>>	code: 1-,2-
4.1	4.85%	4321	Show>>	code: 1^,2^
5.1	4.59%	4090	Show>>	code: 1^3-5^7-,2^4-6^8-



E. focardii



Rank	Proportion	Event Count	Event Details	Intron-Exon Structure
1.1	27.27%	554	Show>>	code: 1^2-,0
2.1	21.91%	445	Show>>	code: 1^3-,2^4-
3.1	11.66%	237	Show>>	code: 1-,2-
4.1	10.88%	221	Show>>	code: 1^,2^
5.1	3.39%	69	Show>>	code: 1-2^,0

Fig.33*: Alternative splicing landscape found by ASTALAVISTA in P. tetraurelia and Euplotes species. The pie charts show the most represented alternative splicing isoforms. Intron retention code is 1^2-,0.*

3.7.3 Machine learning performances

The tables 8 and 9 below represent the results of the model evaluators, applied to each organism. Four machine learning algorithms, Adaboost M1, NeuralNetworks, C5.0 and SVM, were evaluated.

In this study, RIs were regarded as negative samples whereas CSIs as positive samples. However, the proportion of negative to positive samples was approximately 3:1000 in the worst scenario (iREAD, *T.ter*), which was unbalanced and the performance of classification tended to be biased towards the positive class. To address this issue, SMOTE proves to be an efficient method for classifying unbalanced dataset[102]. SMOTE is an algorithm that performs data augmentation by creating synthetic data points based on the original data. SMOTE can be seen as an advanced version of oversampling, or as a specific algorithm for data augmentation. The advantage of SMOTE is that are not generating duplicates, but rather creating synthetic data points that are slightly different from the original data points.

Only the values obtained with the SMOTE preprocessing for iREAD and Down-sampling method with ASTALAVISTA are represented in the tables since they performed better in the Matthews correlation coefficient (MCC) than the other sampling methods. For an extensive view of the results all the comprehensive tables of the other evaluators are in the appendix supplementary materials (Appendix C).

As an alternative measure unaffected by the unbalanced datasets issue, the Matthews correlation coefficient is a contingency matrix method of calculating the Pearson product moment correlation coefficient. Accuracy and F1-score, although popular, can generate misleading results on imbalanced datasets, because they fail to consider the ratio between positive and negative elements. To get a high-quality score, the classifier has to make correct predictions both on the majority of the negative cases, and on the majority of the positive cases, independently of their ratios in the overall dataset. F1 and accuracy, instead, generate reliable results only when applied to balanced datasets, and produce misleading results when applied to imbalanced cases. For these reasons, I choose MCC as a primary evaluator in our binary classification predictions, instead of using F1 score, accuracy or other evaluators.

Surprisingly, machine learning algorithms performances were poor in all the analyzed organisms after ASTALAVISTA intron labeling. After the Down-sampling processing, the highest values obtained were with *T. thermophila* using NeuralNetworks (MCC 0.104), with *E. vannus* using Adaboost M1 (MCC 0.114) and with *E. focardii* using NeuralNetworks (MCC 0.116). These values, however, were not acceptable for the model's discriminatory power because they were very close to zero (random guess). In some organisms such as *T. malaccensis*, *T. borealis* and *E. vannus* the SVM algorithm was not able to resolve the labeling due to the poor quality of the available data. The lowest performance was obtained in *P. tetraurelia*. The models were not able to confidently predict whether an intron was retained or not, a suggestion that the features used, the initial raw data or the labelling method were wrong.

The precision values (the ratio of correctly predicted positive observations to the total predicted positive observations) were much higher in all organisms (>0.85) and even in *E. vannus*, the organism with the lowest precision values, they were acceptable (about 0.5). Combining the precision values with those of Sensitivities (the ratio of correctly predicted positive observations to the all observations in actual class) yielded discrete F1-score values (the weighted average of Precision and Sensitivity) almost all between 0.5 and 0.7, but these could not be taken into account as these evaluators become less useful and biased by the majority class in highly unbalanced classes as in our experiment. Eventually, using ASTALAVISTA as labelling tool, the evaluators told us that the models using the provided features were able to discriminate the CSIs (majority class) but were not effective to recognize the RIs, causing many labeling errors.

ASTALAVISTA
Model evaluation

Tetrahymena thermophila

Algorithm	**AUC-PR**	**AUC-ROC**	**Accuracy**	**Precision**	**F1 score**	**MCC**	**Sensitivity**	**Specificity**
Adaboost M1	0,59	0,59	0,594	0,896	0,921	0,078	0,469	0,699
NeuralNetworks	0,59	0,62	0,493	0,918	0,618	0,104	0,466	0,694
C5,0	0,58	0,599	0,458	0,882	0,586	0,047	0,047	0,047
SVM	0,57	0,59	0,477	0,909	0,603	0,078	0,452	0,667

Tetrahymena malaccensis

Algorithm	**AUC-PR**	**AUC-ROC**	**Accuracy**	**Precision**	**F1 score**	**MCC**	**Sensitivity**	**Specificity**
Adaboost M1	0,57	0,57	0,524	0,931	0,667	0,045	0,52	0,564
NeuralNetworks	0,54	0,58	0,385	0,947	0,512	0,072	0,351	0,775
C5,0	0,56	0,538	0,351	0,947	0,468	0,067	0,311	0,802
SVM	-	-	-	-	-	-	-	-

Tetrahymena borealis

Algorithm	**AUC-PR**	**AUC-ROC**	**Accuracy**	**Precision**	**F1 score**	**MCC**	**Sensitivity**	**Specificity**
Adaboost M1	0,6	0,62	0,648	0,962	0,779	0,083	0,655	0,526
NeuralNetworks	0,58	0,61	0,555	0,964	0,701	0,074	0,551	0,616
C5,0	0,57	0,534	0,552	0,955	0,032	0,08	0,554	0,518
SVM	-	-	-	-	-	-	-	-

Tetrahymena elliotti

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	0,59	0,6	0,6	0,914	0,77	0,051	0,665	0,665
NeuralNetworks	0,62	0,64	0,504	0,911	0,632	0,088	0,484	0,652
C5,0	0,6	0,52	0,652	0,901	0,779	0,039	0,681	0,38
SVM	0,54	0,57	0,547	0,922	0,684	0,07	0,544	0,574

Paramecium tetraurelia

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	0,58	0,59	0,595	0,921	0,732	0,047	0,607	0,4735
NeuralNetworks	0,59	0,59	0,582	0,927	0,718	0,07	0,586	0,534
C5,0	0,57	0,549	0,629	0,924	0,76	0,065	0,645	0,465
SVM	0,56	0,56	0,525	0,924	0,667	0,048	0,522	0,563

Euplotes vannus

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	0,58	0,58	0,54	0,434	0,521	0,114	0,434	0,679
NeuralNetworks	0,56	0,58	0,532	0,628	0,542	0,084	0,476	0,609
C5,0	0,57	0,566	0,531	0,628	0,542	0,083	0,4768	0,607
SVM	-	-	-	-	-	-	-	-

Euplotes focardii

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	0,54	0,53	0,604	0,975	0,75	0,003	0,609	0,4
NeuralNetworks	0,6	0,58	0,5	0,624	0,922	0,116	0,472	0,706
C5,0	0,58	0,585	0,543	0,978	0,698	0,025	0,543	0,538
SVM	0,45	0,44	0,562	0,976	0,715	0,013	0,564	0,476

Table.8: Machine learning models performance evaluation after Down-sampling using ASTALAVISTA intron recognition.

The performance obtained with iREAD after model evaluation was overall better. The best results obtained were with the C5.0 algorithm in *Paramecium tetraurelia* (MCC 0.983) and in *Tetrahymena thermophila* with Adaboost M1 (MCC 0.979). C5.0 was also the best algorithm among those tested in all organisms, having always very good performances (MCC>0.79). In general *P. tetraurelia* and *T. thermophila* was the best performing organism obtaining an average MCC value respectively of 0.899 and 0.884. Models trained with iREAD were more capable of discriminating CSIs from RIs.

The lowest performing organisms were Euplotes species. As seen above, the two tools used (ASTALAVISTA and iRead) in these organisms encountered several difficulties in consistently labeling introns, yielding mixed results. Causes could be found in their peculiar genome features that might have biased the tools used, which were probably optimized for canonical genome structures.

Euplotes macronucleus genome are extensively fragmented to gene-sized nano-chromosomes, which facilitates the evolution of genetic code. Previous studies indicate that ciliates evolved diversified and flexible nuclear genetic code from their ancestors with ambiguous genetic codes[103]. For most species, UGA remains as stop while UAA and UAG are reassigned to code glutamine, tyrosine or glutamic acid[103]. It is opposite in Euplotes, whose UGA codon is reassigned to code cysteine while UAA and UAG are stops[104]. However, euplotids evolves another important mechanism of programmed ribosomal frameshifting at the stop codons UAA and UAG, which can solve the same problem of canonical stop codons residing in the coding regions. It is proposed that translation (either through reassignment or frame- shifting), rather than termination, is the default recognition mode for “stop” codons while termination is due to the context-specific override provided by transcript ends[103].

For these reasons, if in the future we want to study these organisms more effectively, customized versions of the tools must be used.

iREAD Model Evaluation

Tetrahymena thermophila

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	0,99	0,99	0,989	0,982	0,989	0,979	0,997	0,982
NeuralNetworks	0,97	0,97	0,841	0,87	0,832	0,684	0,798	0,883
C5,0	0,99	0,98	0,961	0,938	0,987	0,959	0,922	0,942
SVM	0,99	0,99	0,953	0,975	0,957	0,917	0,939	0,974

Tetrahymena malaccensis

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	-	-	-	-	-	-	-	-
NeuralNetworks	0,81	0,81	0,709	0,721	0,691	0,419	0,66	0,752
C5,0	0,98	0,98	0,948	0,921	0,948	0,898	0,978	0,919
SVM	0,84	0,86	0,781	0,786	0,772	0,562	0,759	0,802

Tetrahymena borealis

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	-	-	-	-	-	-	-	-
NeuralNetworks	0,91	0,92	0,783	0,785	0,777	0,567	0,77	0,797
C5,0	0,98	0,98	0,984	0,972	0,984	0,97	0,997	0,973
SVM	0,94	0,95	0,919	0,952	0,914	0,84	0,879	0,957

Tetrahymena elliotti

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	0,85	0,87	0,745	0,734	0,727	0,489	0,719	0,769
NeuralNetworks	0,84	0,86	0,751	0,73	0,738	0,501	0,747	0,754
C5,0	0,99	0,99	0,961	0,938	0,959	0,922	0,981	0,942
SVM	0,98	0,99	0,96	0,939	0,959	0,922	0,968	0,943

Paramecium tetraurelia

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	0,98	0,98	0,913	0,893	0,914	0,828	0,937	0,89
NeuralNetworks	0,98	0,98	0,923	0,966	0,919	0,851	0,877	0,969
C5,0	0,99	0,97	0,991	0,985	0,991	0,983	0,997	0,986
SVM	0,98	0,98	0,969	0,968	0,968	0,937	0,968	0,969

Euplotes vannus

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	0,94	0,92	8,831	0,81	0,825	0,633	0,841	0,823
NeuralNetworks	0,76	0,75	0,657	0,582	0,656	0,311	0,582	0,726
C5,0	0,97	0,96	0,874	0,858	0,869	0,748	0,88	0,869
SVM	0,76	0,75	0,697	0,692	0,669	0,391	0,648	0,74

Euplotes focardii

Algorithm	AUC-PR	AUC-ROC	Accuracy	Precision	F1 score	MCC	Sensitivity	Specificity
Adaboost M1	0,82	0,79	0,708	0,693	0,653	0,405	0,616	0,782
NeuralNetworks	0,74	0,78	0,641	0,622	0,578	0,271	0,539	0,727
C5,0	0,97	0,98	0,896	0,877	0,884	0,79	0,89	0,9
SVM	0,78	0,75	0,689	0,684	0,615	0,364	0,558	0,793

***Table.9**: Machine learning models performance evaluation after SMOTE, using iREAD intron recognition.*

The figure 34, shows the gain in performance brought by iREAD in all tested organisms that ranged from a gain of 4.6x to even 14.6x in the case of *P.tetraurelia*. Probably this gain is the result of a higher labeling power by iREAD compared to ASTALAVISTA, allowing the ML algorithms together with the features I used to have a good performance in almost all organisms.

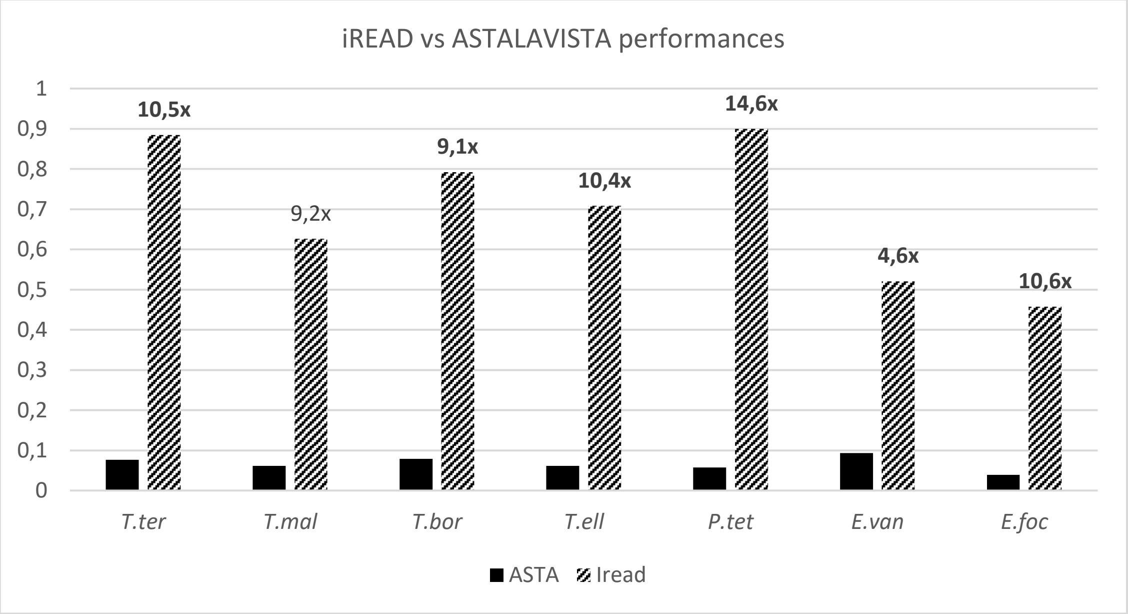


***Fig.34**: iRead vs ASTALAVISTA gain in performances. The values on the y-axis represent the averages of the models MCC.*

Analyzing iRead-labeled data performances (Figure 35), the best overall machine-learning algorithm was C5.0, with decent scores in all seven organisms. In second place was SVM and then NNs followed in last place by Adaboost M1, that despite getting the highest score in *T. thermophila,* failed the classification in two out of seven organisms.

Our results clearly demonstrate that C5.0 offers more advantageous classification performance than SVM (Fig.35). Performances of these two kinds of classifier are influenced by their respective hyperparameters. Our experience showed that the parameter optimization was easier to implement for C5.0, resulting to a stable classifier performance. In contrast, slight changes in these parameters would cause large variation in the classifier performance in SVM[105]. Although I employed Genetic Algorithm Optimization to search the optimal parameters and have obtained better classification performance in comparison with the result using traditional grid search method, the classification performance of SVM may be further improved using different type of optimization like Particle Swarm[106]. Unlike SVM, individual decision trees in C5.0 automatically utilize informative features more frequently in training process and achieve independent predictions, which were combined to gain accurate predictions[107]. Therefore, C5.0 presents significant superiority in failure tolerances and robustness, which plausibly explain its consistent advantageous performance.

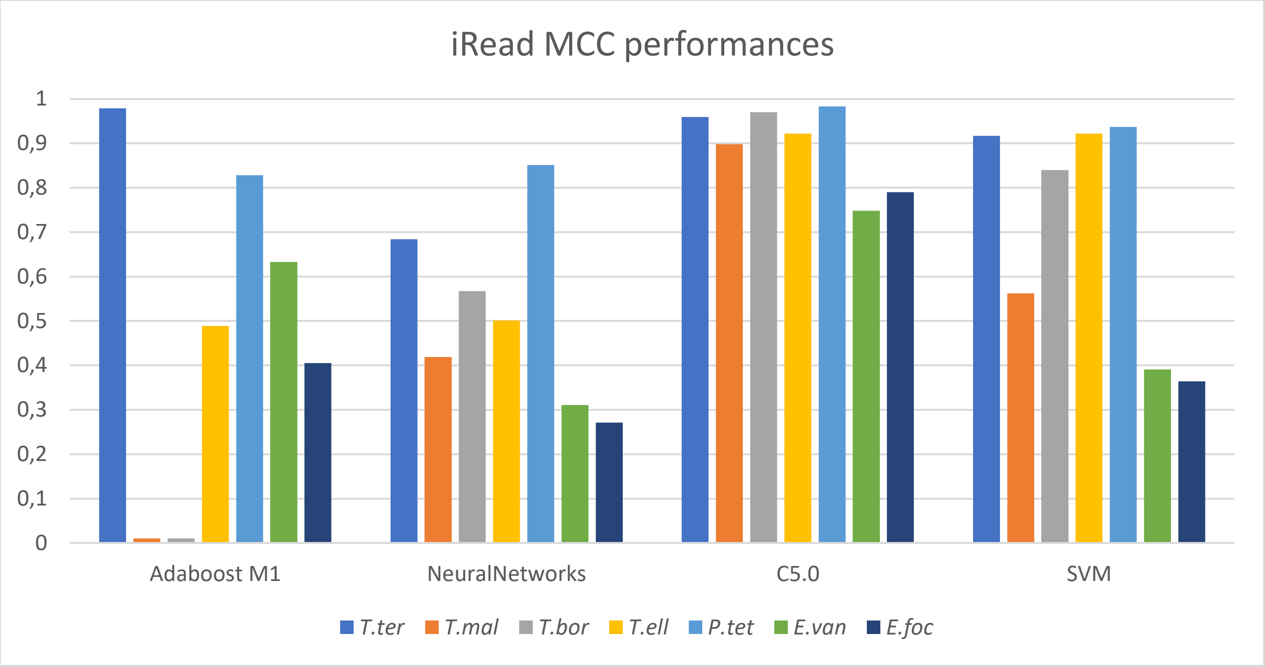


Fig.35: *Machine learning algorithms performances after iRead labelling. The values on the y-axis represent the averages of the models MCC.*

3.7.4 Feature importance and visualization

Using the CARET package on the best performing dataset (iRead-labelled), I was able to understand the features importance (Fig.36) during the ML classification process (RIs and CSIs). Moreover, this package has been useful to quickly visualize the differences between features in one group of introns or in the other (Fig.37).

In *Tetrahymena* species, I was able to recognize four main common features in this genus: Lempel-Ziv Complexity Measure in primary (Lev) and in local secondary structure (nSsLv) and two segmental sequences (ΦGC, ΦGT). This feature overlap could be due to the common origin of the datasets or to their belonging to the same genus. Figure 36 shows that the most important features in *Tetrahymena* species was The Lempel-Ziv complexity. Lempel-Ziv Complexity Measure, based on the Lempel-

Ziv-Welsh compression algorithm, is a feature that represents the complexity of the nucleotide sequence. DNA sequence can be treated as finite-length symbol strings over a four-letter alphabet (A, C, T, G)[108]. As a universal and computable complexity measure, Lev complexity is valid to describe the complexity of DNA sequences, one of the most basic properties of a symbolic sequence.

This feature, in our case, represented the repetitiveness of nucleotides on the intron sequence. The higher the Lev value was, the less repetitive the nucleotides in the sequence were. In the experiments with *Tetrahymena* this feature was significant, in fact descriptive analysis done in retrospect, showed that RIs have a higher Lev than CSIs (Fig.37), from which I could highlight that the structure of the sequence is more chaotic and less compressible. The same Lempel-Ziv measure was also calculated for the secondary structure, both in global (SsLv) and local (nSsLv) form. I can infer that *Tetrahymena* species have a less complex secondary structure in RIs than CSIs, typically with smaller structures (loops, hairpin etc.)

A simpler and more straightforward interpretation of Lev is by to focus on the quantity which is known to be an efficient estimator of the entropy rate[109]. The entropy rate is a quantity from Information Theory, which measures how many bits of innovation are introduced by each new data sample [110]. Moreover, the entropy rate is a good measure of how hard it is to predict the next value of a sequence. In effect, one half of the entropy rate approximates the probability of making an error with the best informed guess about the next sample[111].

Other two important features were ΦGC and ΦGT. The first feature (ΦGC) was higher in in RIs than in CSIs and the second was lower (Fig.37). This indicates that difference between G and C contents for segmental intron sequences in RIs was greater than that in CSIs, whereas the difference between G and C contents for segmental intron sequences was higher in CSIs than that in RIs.

Sequential ΦGC content could be higher in RI because the second largest class, GC-AG introns, appeared more frequently in RIs than CSIs[84] and also for the higher content of GC in retained introns. Instead lower values of ΦGT in RIs was consistent with previous reports[100,101].

However, the condition was different for the two *Euplotes* species and for *P. tetraurelia*. Not only the best-performing iRead-labelling features were different from the *Tetrahymena* species, but also from each other. This could be due to the origin of the datasets from three different research groups[112], but also to the different structure of the introns. Indeed, *P. tetraurelia* had the shortest introns of all organisms (median 25bp), and *Euplotes* species also had much shorter introns (median 27-28bp) than *Tetrahymena* species (63 to 68bp). In addition, these three species also had the highest GC content in their introns. This led to a change in the structure properties[113] of the introns and consequently also to a difference in the importance of the features.

Among the main features, common to *P. tetraurelia* and *E. focardii* was another entropy-based feature, the Bayesian (Laplace) estimator. The entropy of a string, in our case an intron nucleotide sequence, represents the amount of information contained in a message ("the intron"). In information theory, entropy represents the inverse measure of the compressibility of a message and depends on the probability of appearance of symbols in the relevant alphabet. In particular, higher is the entropy, more random is the message and more information it contained. In our study, the results show higher entropy in retained introns, which means more disorder in the sequence directly proportional to the information it contains.

The same Lempel-Ziv measure on the secondary structure, both in global (SsLv) and local (nSsLv) was also important in *Euplotes* species and similar to that estimated in *Tetraymena species*, with lower values in retained introns.

Lower values of ΦGT in *E. vannus* retained introns were consistent with those in the Tetrahymena species.

RNA secondary structures have been demonstrated to affect AS[114,115]. In our models only the calculation of the entropy on the secondary structures (SsLv, nSsLv) was relevant for a correct classification. The simple quantitative calculation of RNAlib-derived secondary structures and their position in the intron did not have the expected effectiveness, always being in the very bottom sections of the importance diagram. Therefore, a great challenge is how to accurately and effectively incorporate RNA secondary structures as features to enhance the performance and accuracy of our classifier. Without a doubt, a comprehensive feature extraction including both linear sequence features and RNA secondary structure features will definitely facilitate our understanding of how RIs are regulated in eukaryotes.

The different outcomes obtained from the two *Euplotes* species may be explained with the distinct niches the two ciliates come from. *E. focardii* is strictly cold-adapted (it cannot survive above 10 °C), a characteristic that favored the evolution of a AT rich genome[104] with respects the other non-cold-adapted *Euplotes* species and most probably a “looser” (more flexible) RNA secondary structure. Intron splicing/retention mechanism may be completely different in this Antarctic ciliate, mainly due to inefficient spliceosome activity at low temperatures than other controllers.

Another explanation could be related, as mentioned before, to the structural characteristics of introns and exons.

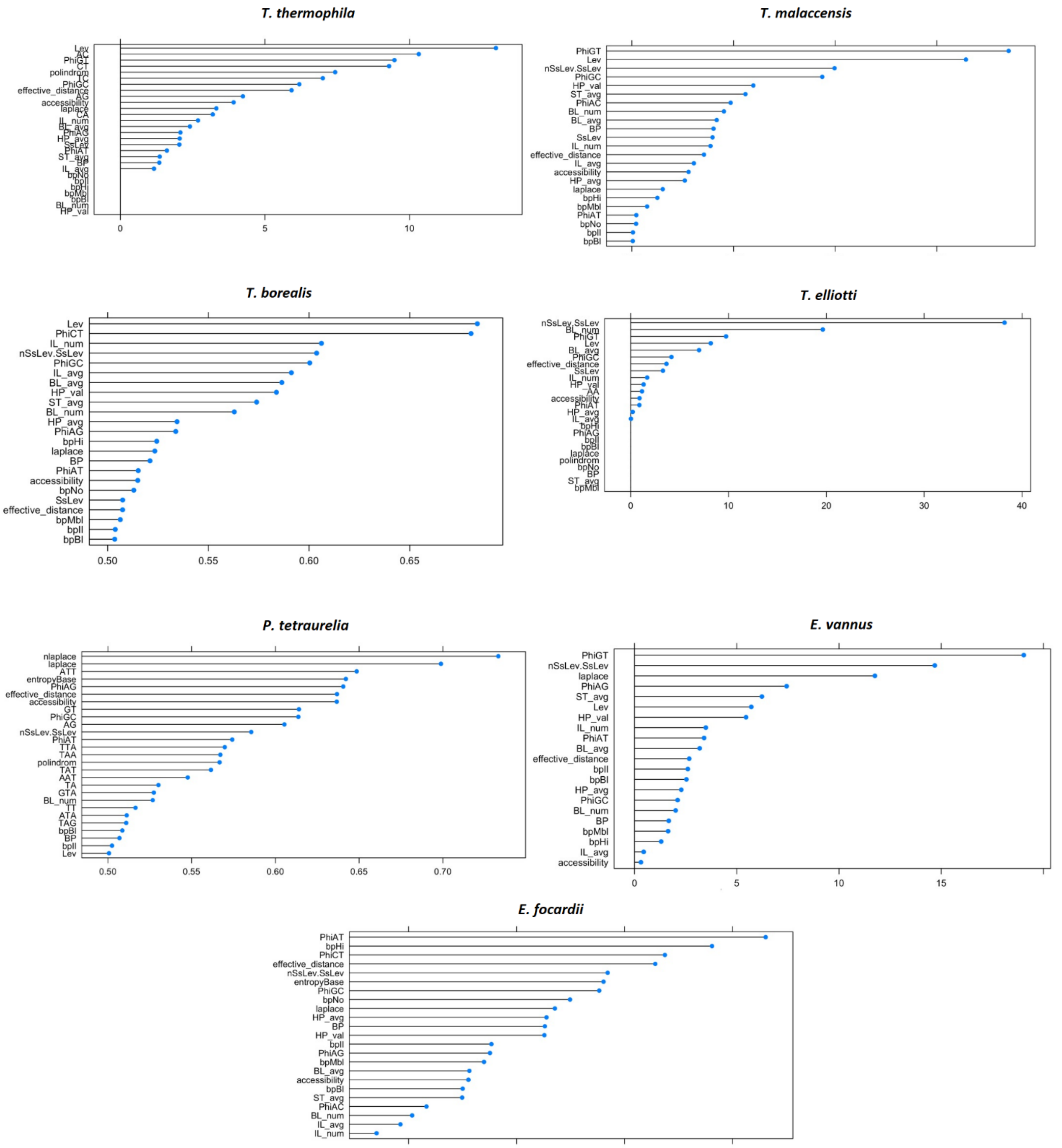


Fig.36: *Machine learning feature importance using CARET tool. A normalization step automatically scales the importance scores to be between 0 and 100.*

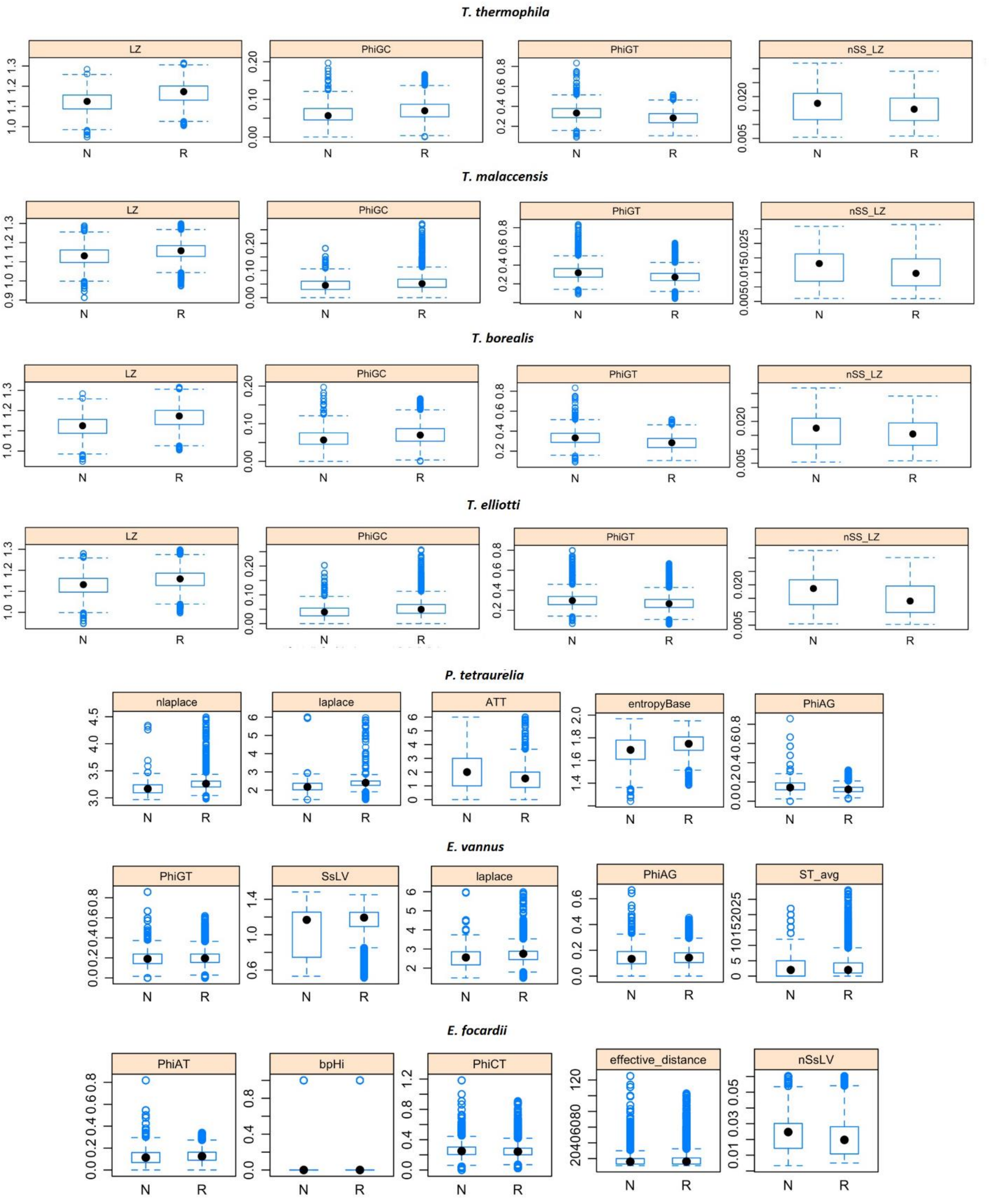


Fig.37: *BoxPlot features visualization using CARET tool. It can be visualized the differences of values in the most representative features in every organism between retained (R) and CSIs (N)*

3.7.5 Tetrahymena species Gene Ontology

In order to assess the functional differences of the sequences that mostly undergo intron retention, I performed a Gene Ontology (GO) and an enrichment analysis using the Fisher's Exact Test [27] This analysis allowed to compare term changes of retained introns normalized with the corresponding genomic sequences .

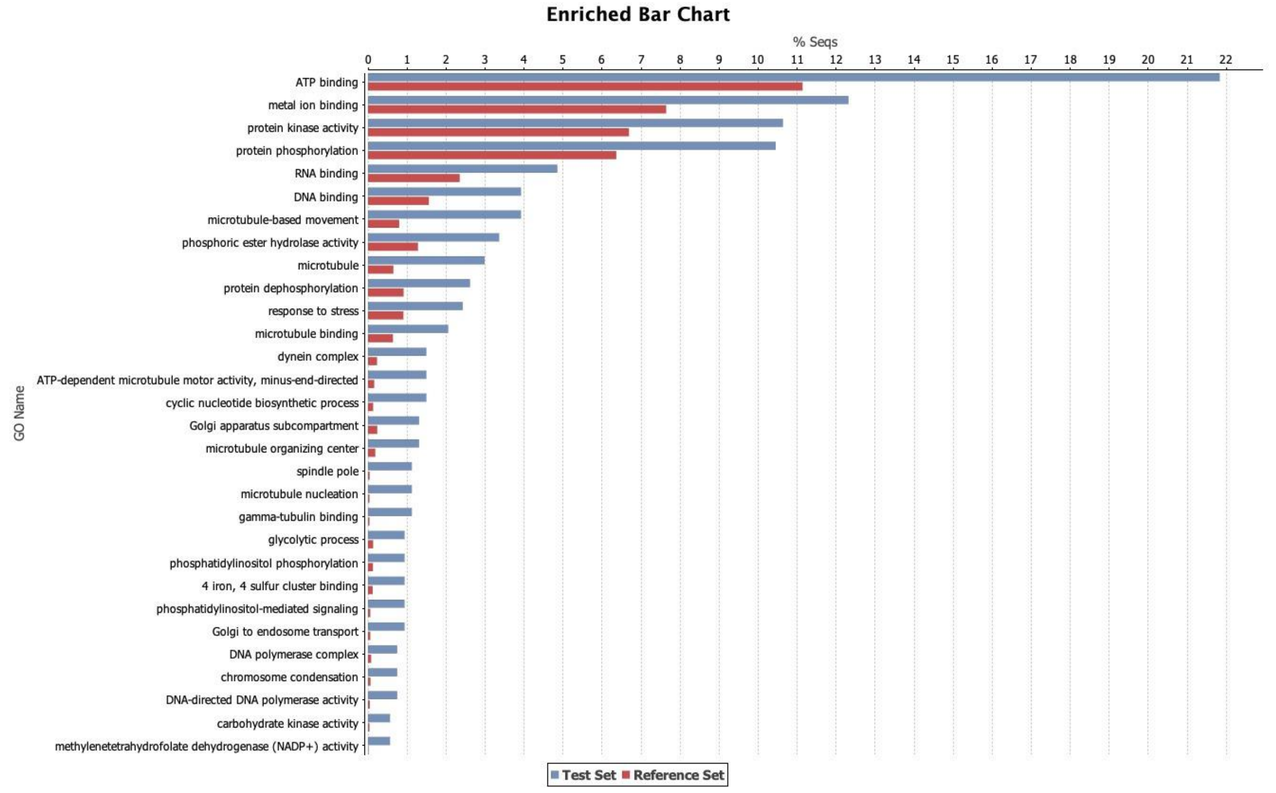


***Fig.39**: Gene Ontology (GO) and an enrichment analysis using the Fisher's Exact Test of Tetrahymena retained introns. Test set (blue): retrained intron; reference set (red): genomic sequences*

In *Tetrahymena* species, the first term was ATP binding (Fig.39). Under this category are usually included membrane proteins, categories of molecules know to be "tailored" by AS events [116]. It follows GO terms involved in protein phosphorylation in ciliates' species.

3.8 Contributions

Alessio Mancini: Conceptualization, Methodology, Validation, Investigation, Data Curation, Writing - Original Draft & Editing

Leonardo Vito: Software, Methodology, Formal analysis, Validation, Investigation, Writing - Original Draft

Marco Piangerelli: Writing – Review, Supervision, Methodology

Sandra Pucciarelli, Emanuela Merelli: Writing – Review, Project administration, Supervision

CHAPTER 4: Conclusions and Future Work

During these years of pursuing my doctorate, I encountered many challenges. Primarily was the multidisciplinary nature of the subject. I found myself interacting with different professionals, mathematicians, physicists, informaticians and bioinformaticians. Each of these figures had their own particular technical language. I was forced to learn a common, interdisciplinary language in order to achieve my goals.

Another challenge was finding myself in a field of research relatively new and definitely in its infancy at the University of Camerino. However, this gave me more freedom in my research and stimulated me to seek knowledge even from figures outside my university.

In the past, a large fraction of clinical data were underestimated. This limitation was due to both the size and complexity of the data and the absence of techniques for collecting and storing such data. These data was frequently underused and undervalued; however, new and improved methods for data collection and storage (eg, electronic health records) provide opportunities to tackle the issue of analysis. In particular, machine learning (ML) has begun to infiltrate the clinical literature broadly. Our tool,*DSaaS,* can help physicians to make easy and fast predictions models that could be helpful to treat hospitalized patients. Additionally, epidemiologists can use predictions to guide policies, research, and drug development for upcoming years. In the first version of *DSaaS,* we provided a useful prediction model for hospitalized patients on the onset of an MDR UTI with discrete performance. Moreover, our objective is to expand the *DSaaS* platform to allow not only physicians but also researchers from different fields to use our tool with a variety of databases.

Several limitations should be noted in this ML application. First, potential risk factors which were unavailable from the review of medical records will be considered for use in the next model. In the next future we will point to enhance the model with other well-known UTIs risk factors like diabetes, the presence of a catheter, sexual-related factors, antibiotic use and renal transplantation [117]. Moreover, using ML techniques, risk factors commonly neglected by traditional statistical models can be discovered. Secondly, the analyzed cases were extracted from a small-scale hospital and therefore the generalizability of our findings may be limited. In the future we wish to gather more cases from a wider variety of hospitals.

The future steps to enlarge the platform will be to develop a dataflow editor and to add unsupervised ML methods. At the end, *DSaaS* platform will allow users to carry out data science pipeline works by obtaining, cleaning, exploring and visualizing data in order to individuate patterns, apply models and understand data trends.

Once we mastered the basic Machine Learning techniques, we decided to investigate the field of alternative splicing, a topic far more complex than the previous one. Alternative splicing contributes to the majority of protein diversity in higher eukaryotes by allowing one gene to generate multiple distinct protein isoforms. It adds another regulation layer of gene expression. Moreover, around 15% of human hereditary diseases and cancers are associated with alternative splicing[118]. The developed pipeline

(Chapter 3.6.1) can be used effectively for the discovery of new features that differentiate Retained Introns from Constitutively Spliced Introns. The plasticity of the pipeline will allow in the future implementing new features and new ML algorithms that may lead to a better understanding of the phenomenon of AS also in other organisms. All the newly discovered features can be found below:

- This pipeline can be utilized to effectively discover and test for new RIs features.
- For this case study, iRead appears to be the best labeling method to use.
- The genus *Tetrahymena* is the one that responded best in Machine Learning feature discovery, achieving consistent results across species.
- *Euplotes* species have been difficult to label by the two tools ASTALAVISTA and iRead.
- For this case study C5.0 was the best Machine Learning algorithm to be used.
- On average RIs are longer, higher GC content and lower AT content than CSIs in all organisms.
- In *Tetrahymena* species the RIs have a higher primary Lempel-Ziv Complexity Measure than CSIs. This means that the sequence primary structure of RIs is more chaotic and less compressible.
- In *Tetrahymena* and *Euplotes* species the RIs have a lower Lempel-Ziv Complexity Measure than CSIs in the secondary structure. This means their secondary structures are smaller.

- In *Tetrahymena* species RIs show different features of segmental nucleotides composition, such as higher *Φ*GC and lower *Φ*GT locally.
- In *P. tetraurelia* and *E. focardii* the RIs have a higher Laplace estimator than CSIs. This means that the sequence primary structure of RIs is more chaotic directly proportional to the information it contains.
- Due to the complexity of the study, it was not possible to find a common feature between the different genus analyzed, but only within the same genera

During the case-study design, I faced several limitations. Most importantly, I encountered difficulties in putting together a harmonious and homogeneous database. All the data used came from different research groups, including my own. The main limitation, however, was the accumulation of error in each step of the pipeline that eventually led us to a result, although good, to be verified experimentally.

Below I list the main sources of error encountered:

- Different data from different research groups

  A crucial prerequisite for a successful RNA-seq study is that the data generated have the potential to answer the biological questions of interest. This is achieved by first defining a good experimental design, that is, by choosing the library type, sequencing depth and number of replicates appropriate for the biological system under study, and second by planning an adequate execution of the sequencing experiment itself, ensuring that data acquisition does not become contaminated with unnecessary biases[119]. All of these issues are amplified when using datasets from different research groups as in our work. This inevitably leads to an increase in the global error. All raw data should be produced by the group that then has the goal of identifying whether or not an intron is retained, to have the complete control. If this is not possible the data should only come from a single research group in order to limit error in this step. These two options decrease the error but inevitably lead to a drastic reduction in the amount of data available. A balance should be struck between acceptable error and the amount of data that can be used for subsequent machine learning modeling. For example, the work designed in this thesis, a multi-organism study, would have been impossible for a single research group unless a reasonable investment of time and money was made.

- Error in data selection and manipulation

There are different sources of error when manipulating transcriptome and genome data. The good practices suggest several control steps. Quality control for the raw reads involves the analysis of sequence quality, GC content, the presence of adaptors, overrepresented k-mers and duplicated reads in order to detect sequencing errors, PCR artifacts or contaminations. Percentage of mapped reads, which is a global indicator of the overall sequencing accuracy and of the presence of contaminating DNA. Other important parameters are the uniformity of read coverage on exons and the mapped strand. Despite all these measures, the researcher generates an amount although minimal of error.

- RIs labelling

There are a considerable number of tools (>10) in the literature that promise to correctly label whether an intron is retained or not[120]. RNA sequencing-based IR detection/quantification software has not been systematically benchmarked to date despite there are many challenges in IR identification. All these approaches are generally hampered by the intrinsic limitations of short-read sequencing for accurate identification at the isoform level, transcriptional 'noise' introduced by DNA contamination or unprocessed pre-mRNA transcripts[121] as for *Euplotes* species.

All of these computational methods while not perfect, are rapid methods by which the researcher can label thousands of RIs in a single day, but most importantly they do not affect the research team's budget as only a laptop and basic coding skills are needed to accomplish the tasks.

In order to have a precise validation it is necessary to rely on experimental techniques, which are slower and more expensive. Northern blot analysis is one of the original methods for RNA visualization and quantification and remains a standard technique to validate IR-transcripts discovered through mRNA-seq experiments[122]. However, Northern blot techniques are relatively involved and time consuming and are less sensitive for accurate quantification in the case of very low abundance IR RNA[123]. RT-PCR has emerged as a simple and efficient way to quantify RNA, including low abundance RNA[124]. The key principle for IR validation and quantification in qRT-PCR is again to use sequence-specific primers that align in the intron or across the intron–exon junctions. The relative abundance of IR can be assessed by using primer pairs in regions that are common to both IR and completely spliced mRNAs.

In this case the researcher is forced to make a choice, between a big amount of data produced by computational methods or the quality of classical techniques, because the quantity and quality of the data set will impact the Machine learning model performances.

- Machine Learning challenges

For any given data set the researcher wants to develop a model that is able to predict with the highest degree of accuracy possible. In Machine Learning, there are many levers that impact the performance of the model. The algorithm choice, the parameters used in the algorithm, the quantity and quality of the data set and the features used to train the model. Despite the methods the researcher applies to minimize errors and increase performance, a minimal amount of error will be inevitable.

Since it is impossible to eradicate these limitations, a future approach will be to limit their impact as much as possible. This will help us achieve the ultimate goal of discovering and applying new features to all the data set without distinction of species and/or genera, providing a unique set of features for a global AS understanding.

∑ Δ	Limitation	Solution
1	Different data from different research groups	Data should only come from a single research group
2	Errors in data selection and manipulation	All the quality control steps should be done
3	Only use IR detection/quantification software	In order to have a precise validation it is necessary to rely on experimental techniques
4	Use a limited number of machine learning algorithms, features and optimization methods	Use the correct number of features, the best hyperparameter-optimization methods and avoid model overfitting

***Table.7**: Limitations and solutions in the intron retention feature discovery pipeline*

In conclusion, the field of machine learning, especially for biologists, has been very fascinating and given the amount of publications and new techniques that are applied every day, is free from the stagnation of ideas that other fields of research are frequently suffering.

CHAPTER 5: BIBLIOGRAPHY

1. Data Science - Grace Systems. Accessed May 31, 2021. http://gracesystems.nl/data-science/
2. Aggarwal CC. *Data Classification: Algorithms and Applications*.; 2014. doi:10.1201/b17320
3. Bishop C. Pattern Recognition and Machine Learning | Christopher Bishop | Springer. Published 2007. Accessed May 22, 2021. http://www.springer.com/us/book/9780387310732%5Cnhttps://www.springer.com/us/book/9780387310732
4. The MIT encyclopedia of the cognitive sciences. Choice Reviews Online. doi:10.5860/choice.37-1902
5. Singh J, Hanson J, Paliwal K, Zhou Y. RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning. *Nat Commun*. 2019;10(1). doi:10.1038/s41467-019-13395-9
6. Rowlands CF, Baralle D, Ellingford JM. Machine Learning Approaches for the Prioritization of Genomic Variants Impacting Pre-mRNA Splicing. *Cells*. 2019;8(12):1513. doi:10.3390/cells8121513
7. Libbrecht MW, Noble WS. Machine learning applications in genetics and genomics. *Nat Rev Genet*. 2015;16(6):321-332. doi:10.1038/nrg3920
8. Radakovich N, Nagy M, Nazha A. Machine learning in haematological malignancies. *Lancet Haematol*. 2020;7(7):e541-e550. doi:10.1016/S2352-3026(20)30121-6
9. Mancini A, Vito L, Marcelli E, et al. Machine learning models predicting multidrug resistant urinary tract infections using “dsaaS.” *BMC Bioinformatics*. 2020;21(Suppl 10):1-12. doi:10.1186/s12859-020-03566-7
10. Tlachac ML, Rundensteiner E, Barton K, Troppy S, Beaulac K, Doron S. Predicting Future Antibiotic Susceptibility using Regression-based Methods on Longitudinal Massachusetts Antibiogram Data. 2018;5(Biostec):978-989. doi:10.5220/0006567401030114
11. Barlam TF, Cosgrove SE, Abbo LM, et al. Implementing an antibiotic stewardship program: Guidelines by the Infectious Diseases Society of America and the Society for Healthcare Epidemiology of America. *Clin Infect Dis*. 2016;62(10):e51-77. doi:10.1093/cid/ciw118
12. Naber KG, Bergman B, Bishop MC, et al. EAU guidelines for the management of urinary and male genital tract infections. Urinary Tract Infection (UTI) Working Group of the Health Care Office (HCO) of the European Association of Urology (EAU). *Eur Urol*. 2015;40(5):576-588. doi:https://uroweb.org/wp-content/uploads/19-Urological-infections_LR2.pdf
13. Maki DG, Tambyah PA. Engineering out the risk for infection with urinary catheters. In: *Emerging Infectious Diseases*. ; 2001. doi:10.3201/eid0702.010240

14. Foxman B. The epidemiology of urinary tract infection. *Nat Rev Urol*. Published online 2010. doi:10.1038/nrurol.2010.190

15. Woodford HJ, George J. Diagnosis and management of urinary infections in older people. *Clin Med J R Coll Physicians London*. Published online 2011. doi:10.7861/clinmedicine.11-1-80

16. Lateef F. Hospital design for better infection control. *J Emerg Trauma Shock*. Published online 2009. doi:10.4103/0974-2700.55329

17. Ventola CL. The antibiotic resistance crisis: part 1: causes and threats. *P T A peer-reviewed J Formul Manag*. 2015;40(4):277-283. doi:PubMed Central PMCID: PMC4378521

18. Cook D. *Practical Machine Learning with H2O*. 1st Editio. O'Reilly Media; 2016.

19. Fluentd Project. Fluentd. Published 2018. https://www.fluentd.org

20. The Apache Software Foundation. Apache Flink. Published 2017. https://flink.apache.org

21. The Apache Software Foundation. Apache Giraph. Published 2018. http://giraph.apache.org

22. The Apache Software Foundation. Apache NiFi. Published 2018. https://nifi.apache.org

23. Snyder B, Bosanac D, Davies R. ActiveMQ in Action. Online.

24. Zaharia M, Franklin MJ, Ghodsi A, et al. Apache Spark: a unified engine for big data processing. *Commun ACM*. Published online 2016. doi:10.1145/2934664

25. Shvachko K, Kuang H, Radia S, Chansler R. The Hadoop distributed file system. In: *2010 IEEE 26th Symposium on Mass Storage Systems and Technologies, MSST2010*. ; 2010. doi:10.1109/MSST.2010.5496972

26. Peng G, Ritchey NA, Casey KS, et al. Scientific stewardship in the open data and big data era - roles and responsibilities of stewards and other major product stakeholders. *D-Lib Mag*. Published online 2016. doi:10.1045/may2016-peng

27. CDC, NHSN. CDC / NHSN Surveillance Definitions for Specific Types of Infections. *Surveill Defin*. Published online 2014. doi:10.1016/j.ajic.2008.03.002

28. Siegel JD, Rhinehart E, Jackson M, Chiarello L. Management of multidrug-resistant organisms in health care settings, 2006. *Am J Infect Control*. Published online 2007. doi:10.1016/j.ajic.2007.10.006

29. Flores-Mireles AL, Walker JN, Caparon M, Hultgren SJ. Urinary tract infections: Epidemiology, mechanisms of infection and treatment options. *Nat Rev Microbiol*. Published online 2015. doi:10.1038/nrmicro3432

30. Kuhn M. Building predictive models in R using the caret package. *J Stat Softw*. Published online 2008.

31. Scrucca L. GA : A Package for Genetic Algorithms in R . *J Stat Softw*. Published online 2015. doi:10.18637/jss.v053.i04

32. Nitesh V. Chawla, Chawla N V., Bowyer KW, Hall LO, Kegelmeyer WP.

SMOTE: Synthetic Minority Over-sampling Technique Nitesh. *J Artif Intell Res*. 2002;16(1):732-735. doi:10.1613/jair.953

33. Little MA, Varoquaux G, Saeb S, et al. Using and understanding cross-validation strategies. Perspectives on Saeb et al. *Gigascience*. 2017;6(5):1-6. doi:10.1093/gigascience/gix020
34. Dorogush AV, Ershov V, Gulin A. CatBoost: gradient boosting with categorical features support. 2018;(Section 4):1-11. doi:arXiv:1810.11363v1
35. Haykin S. *Neural Networks: A Comprehensive Foundation*.; 1994. doi:10.1017/S0269888998214044
36. Vapnik VN. An overview of statistical learning theory. *IEEE Trans Neural Networks*. Published online 1999. doi:10.1109/72.788640
37. Kuhn M, Johnson K. *Applied Predictive Modeling*.; 2013. doi:10.1007/978-1-4614-6849-3
38. Austenfeld M. A Graphical User Interface for R in a Rich Client Platform for Ecological Modeling. *J Stat Softw*. Published online 2012. doi:http://dx.doi.org/10.18637/jss.v049.i04
39. Chou H. Local assembly and pre-mRNA splicing analyses by high-throughput sequencing data. Published online 2012.
40. Wang ET, Sandberg R, Luo S, et al. Alternative isoform regulation in human tissue transcriptomes. *Nature*. 2008;456(7221):470-476. doi:10.1038/nature07509
41. Pickrell JK, Pai AA, Gilad Y, Pritchard JK. Noisy splicing drives mRNA isoform diversity in human cells. *PLoS Genet*. 2010;6(12):1-11. doi:10.1371/journal.pgen.1001236
42. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J Mol Biol*. 1990;215(3):403-410. doi:10.1016/S0022-2836(05)80360-2
43. Dobin A, Davis CA, Schlesinger F, et al. STAR: Ultrafast universal RNA-seq aligner. *Bioinformatics*. 2013;29(1):15-21. doi:10.1093/bioinformatics/bts635
44. U B, NL B-M, Q P, et al. Widespread intron retention in mammals functionally tunes transcriptomes. *Genome Res*. 2014;24:1774-1786. doi:10.1101/gr.177790.114.1774
45. Li H-D, Funk CC, Price ND. A Tool for Intron Retention Detection from RNA-seq data. 2017;(41271370):2015-2018.
46. Kalyna M, Lopato S, Voronin V, Barta A. Evolutionary conservation and regulation of particular alternative splicing events in plant SR proteins. *Nucleic Acids Res*. 2006;34(16):4395-4405. doi:10.1093/nar/gkl570
47. Lareau LF, Inada M, Green RE, Wengrod JC, Brenner SE. Unproductive splicing of SR genes associated with highly conserved and ultraconserved DNA elements. *Nature*. 2007;446(7138):926-929. doi:10.1038/nature05676
48. Yap K, Makeyev E V. Regulation of gene expression in mammalian nervous system through alternative pre-mRNA splicing coupled with RNA quality control mechanisms. *Mol Cell Neurosci*. 2013;56:420-428.

doi:10.1016/j.mcn.2013.01.003

49. Wong JJL, Ritchie W, Ebner OA, et al. Orchestrated intron retention regulates normal granulocyte differentiation. *Cell*. 2013;154(3):583-595. doi:10.1016/j.cell.2013.06.052

50. Roberts A, Pimentel H, Trapnell C, Pachter L. Identification of novel transcripts in annotated genomes using RNA-seq. *Bioinformatics*. 2011;27(17):2325-2329. doi:10.1093/bioinformatics/btr355

51. Trapnell C, Roberts A, Goff L, et al. Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat Protoc*. 2012;7(3):562-578. doi:10.1038/nprot.2012.016

52. Foissac S, Sammeth M. ASTALAVISTA: Dynamic and flexible analysis of alternative splicing events in custom gene datasets. *Nucleic Acids Res*. 2007;35(SUPPL.2):297-299. doi:10.1093/nar/gkm311

53. Li H, Handsaker B, Wysoker A, et al. The Sequence Alignment/Map format and SAMtools. *Bioinformatics*. 2009;25(16):2078. doi:10.1093/BIOINFORMATICS/BTP352

54. Smola AJ, Schölkopf B. A tutorial on support vector regression. *Stat Comput*. 2004;14(3):199-222. doi:10.1023/B:STCO.0000035301.49549.88

55. O’Fallon BD, Wooderchak-Donahue W, Crockett DK. A support vector machine for identification of single-nucleotide polymorphisms from next-generation sequencing data. *Bioinformatics*. 2013;29(11):1361-1366. doi:10.1093/bioinformatics/btt172

56. Minsky M. Evolutionary artificial neural networks. *Adapt Learn Optim*. 2014;15(March 2013):187-230. doi:10.1007/978-3-642-37846-1_7

57. Alipanahi B, Delong A, Weirauch MT, Frey BJ. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nat Biotechnol*. 2015;33(8):831-838. doi:10.1038/nbt.3300

58. Quang D, Chen Y, Xie X. DANN: A deep learning approach for annotating the pathogenicity of genetic variants. *Bioinformatics*. 2015;31(5):761-763. doi:10.1093/bioinformatics/btu703

59. Chen Y, Li Y, Narayan R, Subramanian A, Xie X. Gene expression inference with deep learning. *Bioinformatics*. 2016;32(12):1832-1839. doi:10.1093/bioinformatics/btw074

60. PANG S, GONG J. C5.0 Classification Algorithm and Application on Individual Credit Evaluation of Banks. *Syst Eng - Theory Pract*. 2009;29(12):94-104. doi:10.1016/s1874-8651(10)60092-0

61. Salzberg SL. C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993. *Mach Learn*. 1994;16(3):235-240. doi:10.1007/bf00993309

62. Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV, Gulin A. CatBoost: unbiased boosting with categorical features. 2017;(Section 4):1-23. http://arxiv.org/abs/1706.09516

63. Dietterich TG. Ensemble Methods in Machine Learning. *Lect Notes Comput*

Sci. 2000;1857:1-15. doi:10.1007/3-540-45014-9_1

64. Eisen JA, Coyne RS, Wu M, et al. Macronuclear genome sequence of the ciliate Tetrahymena thermophila, a model eukaryote. *PLoS Biol*. 2006;4(9):1620-1642. doi:10.1371/journal.pbio.0040286
65. Zufall RA. Mating systems and reproductive strategies in tetrahymena. In: *Biocommunication of Ciliates*. Springer International Publishing; 2016:221-233. doi:10.1007/978-3-319-32211-7_13
66. Ruehle MD, Orias E, Pearson CG. Tetrahymena as a unicellular model eukaryote: Genetic and genomic tools. *Genetics*. 2016;203(2):649-665. doi:10.1534/genetics.114.169748
67. Xiong J, Lu X, Zhou Z, et al. Transcriptome analysis of the model protozoan, tetrahymena thermophila, using deep RNA sequencing. *PLoS One*. 2012;7(2):1-13. doi:10.1371/journal.pone.0030630
68. Beisson J, Bétermier M, Bré MH, et al. Paramecium tetraurelia: The renaissance of an early unicellular model. *Cold Spring Harb Protoc*. 2010;5(1). doi:10.1101/pdb.emo140
69. Orias E, Singh DP, Meyer E. Genetics and Epigenetics of Mating Type Determination in Paramecium and Tetrahymena. *Annu Rev Microbiol*. 2017;71:133-156. doi:10.1146/annurev-micro-090816-093342
70. Kapusta A, Matsuda A, Marmignon A, et al. Highly precise and developmentally programmed genome assembly in paramecium requires ligase IV-dependent end joining. *PLoS Genet*. 2011;7(4). doi:10.1371/journal.pgen.1002049
71. Jaillon O, Bouhouche K, Gout JF, et al. Translational control of intron splicing in eukaryotes. *Nature*. 2008;451(7176):359-362. doi:10.1038/nature06495
72. Irimia M, Roy SW. Origin of spliceosomal introns and alternative splicing. *Cold Spring Harb Perspect Biol*. 2014;6(6):a016071. doi:10.1101/cshperspect.a016071
73. Genome analysis of the unicellular eukaryote Euplotes vannus reveals molecular basis for sex determination and tolerance to environmental stresses. Published online 2018:1-39.
74. Valbonesi A, Luporini P. Biology of Euplotes focardii, an Antarctic ciliate. *Polar Biol*. 1993;13(7):489-493. doi:10.1007/BF00233140
75. Devaraj RR, Pucciarelli S, Mignone F, et al. Analysis of the transcriptome from the Antarctic psychrophilic and stenothermal ciliate. Published online 2011.
76. Chen X, Jiang Y, Gao F, et al. Genome analyses of the new model protist Euplotes vannus focusing on genome rearrangement and resistance to environmental stressors. *Mol Ecol Resour*. 2019;(April). doi:10.1111/1755-0998.13023
77. Galaxy. Accessed May 25, 2021. https://toolshed.g2.bx.psu.edu/repository/display_tool?repository_id=ef55e08d4640e0bc&tool_config=database%2Fcommunity_files%2F000%2Frepo_442%2Ffasta-stats.xml&changeset_revision=20ca2574216a

78. Node-RED. Accessed May 25, 2021. https://nodered.org/
79. Saudemont B, Popa A, Parmley JL, et al. The fitness cost of mis-splicing is the main determinant of alternative splicing patterns. *Genome Biol*. 2017;18(1):1-15. doi:10.1186/s13059-017-1344-6
80. Keeling PJ, Burki F, Wilcox HM, et al. The Marine Microbial Eukaryote Transcriptome Sequencing Project (MMETSP): Illuminating the Functional Diversity of Eukaryotic Life in the Oceans through Transcriptome Sequencing. *PLoS Biol*. 2014;12(6):e1001889. doi:10.1371/journal.pbio.1001889
81. Ewing B, Hillier LD, Wendl MC, Green P. Base-calling of automated sequencer traces using phred. I. Accuracy assessment. *Genome Res*. 1998;8(3):175-185. doi:10.1101/GR.8.3.175
82. Li HD, Funk CC, Price ND. iREAD: A tool for intron retention detection from RNA-seq data. *bioRxiv*. Published online 2017:1-11. doi:10.1101/135624
83. Götz S, García-Gómez JM, Terol J, et al. High-throughput functional annotation and data mining with the Blast2GO suite. *Nucleic Acids Res*. 2008;36(10):3420-3435. doi:10.1093/nar/gkn176
84. Mao R, Liang C, Zhang Y, Hao X, Li J. 50/50 expressional odds of retention signifies the distinction between retained introns and constitutively spliced introns in Arabidopsis thaliana. *Front Plant Sci*. 2017;8(October):1-16. doi:10.3389/fpls.2017.01728
85. Mao R, Raj Kumar PK, Guo C, Zhang Y, Liang C. Comparative analyses between retained introns and constitutively spliced introns in Arabidopsis thaliana using random forest and support vector machine. *PLoS One*. 2014;9(8). doi:10.1371/journal.pone.0104049
86. Wolpert DH, Wolf DR. Estimating functions of probability distributions from a finite set of samples. *Phys Rev E*. 1995;52(6):6841-6854. doi:10.1103/PhysRevE.52.6841
87. Lorenz R, Bernhart SH, Höner zu Siederdissen C, et al. ViennaRNA Package 2.0. *Algorithms Mol Biol*. 2011;6(1):26. doi:10.1186/1748-7188-6-26
88. Hofacker IL. RNA secondary structure analysis using the Vienna RNA package. *Curr Protoc Bioinforma*. 2009;26(SUPPL. 26):12.2.1-12.2.16. doi:10.1002/0471250953.bi1202s26
89. Aboy M, Hornero R, Abásolo D, Álvarez D. Interpretation of the Lempel-Ziv complexity measure in the context of biomedical signal analysis. *IEEE Trans Biomed Eng*. 2006;53(11):2282-2288. doi:10.1109/TBME.2006.883696
90. Orlov YL, Potapov VN. Complexity: An internet resource for analysis of DNA sequence complexity. *Nucleic Acids Res*. 2004;32(WEB SERVER ISS.):W628. doi:10.1093/nar/gkh466
91. Abásolo D, Simons S, Morgado da Silva R, Tononi G, Vyazovskiy V V. Lempel-Ziv complexity of cortical activity during sleep and waking in rats. *J Neurophysiol*. 2015;113(7):2742-2752. doi:10.1152/jn.00575.2014
92. Otu HH, Sayood K. A new sequence distance measure for phylogenetic tree construction. *Bioinformatics*. 2003;19(16):2122-2130.

doi:10.1093/bioinformatics/btg295

93. Plass M, Eyras E. Approaches to link RNA secondary structures with splicing regulation. *Methods Mol Biol*. 2014;1126:341-356. doi:10.1007/978-1-62703-980-2_25

94. MATLAB Deep Learning - Google Books. Accessed May 26, 2021. https://www.google.it/books/edition/MATLAB_Deep_Learning/F1coDwAAQBAJ?hl=it&gbpv=1&dq=neural+networks+Machine+Learning&printsec=frontcover

95. Support Vector Machines and Perceptrons - Google Books. Accessed May 26, 2021. https://www.google.it/books/edition/Support_Vector_Machines_and_Perceptrons/kv_cDAAAQBAJ?hl=it&gbpv=1&dq=support+vector+Machine+Learning&printsec=frontcover

96. Blagus R, Lusa L. SMOTE for high-dimensional class-imbalanced data. *BMC Bioinformatics*. 2013;14. doi:10.1186/1471-2105-14-106

97. Rose P, Lunardon AN, Menardi G, Torelli N, Lunardon MN. Package ‘ROSE.’ Published online 2021. https://cran.r-project.org/web/packages/ROSE/ROSE.pdf

98. Wang Z, Zhang J, Verma N. Realizing Low-Energy Classification Systems by Implementing Matrix Multiplication Directly Within an ADC. In: *IEEE Transactions on Biomedical Circuits and Systems*. Vol 9. Institute of Electrical and Electronics Engineers Inc.; 2015:825-837. doi:10.1109/TBCAS.2015.2500101

99. Gooding C, Clark F, Wollerton MC, Grellscheid SN, Groom H, Smith CWJ. A class of human exons with predicted distant branch points revealed by analysis of AG dinucleotide exclusion zones. *Genome Biol*. 2006;7(1):1-19. doi:10.1186/gb-2006-7-1-r1

100. Ner-Gaon H, Halachmi R, Savaldi-Goldstein S, Rubin E, Ophir R, Fluhr R. Intron retention is a major phenomenon in alternative splicing in Arabidopsis. *Plant J*. 2004;39(6):877-885. doi:10.1111/j.1365-313X.2004.02172.x

101. Sakabe NJ, de Souza SJ. Sequence features responsible for intron retention in human. *BMC Genomics*. 2007;8:1-14. doi:10.1186/1471-2164-8-59

102. Yen SJ, Lee YS. Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Syst Appl*. 2009;36(3 PART 1):5718-5727. doi:10.1016/j.eswa.2008.06.108

103. Abstract G. Genetic Codes with No Dedicated Stop Codon: Context-Dependent Translation Termination. Published online 2016. doi:10.1016/j.cell.2016.06.020

104. Mozzicafreddo M, Pucciarelli S, Swart EC, et al. The macronuclear genome of the Antarctic psychrophilic marine ciliate Euplotes focardii reveals new insights on molecular cold adaptation. *Sci Reports 2021 111*. 2021;11(1):1-20. doi:10.1038/s41598-021-98168-5

105. Wang J, Chen Q, Chen Y. RBF kernel based support vector machine with universal approximation and its application. *Lect Notes Comput Sci (including*

Subser Lect Notes Artif Intell Lect Notes Bioinformatics). 2004;3173:512-517. doi:10.1007/978-3-540-28647-9_85

106. Nakano S, Ishigame A, Yasuda K. Consideration of Particle Swarm Optimization combined with tabu search. *Electr Eng Japan (English Transl Denki Gakkai Ronbunshi)*. 2010;172(4):31-37. doi:10.1002/eej.20966
107. Pashaei E, Ozen M, Aydin N. Improving medical diagnosis reliability using Boosted C5.0 decision tree empowered by Particle Swarm Optimization. *Proc Annu Int Conf IEEE Eng Med Biol Soc EMBS*. 2015;2015-Novem:7230-7233. doi:10.1109/EMBC.2015.7320060
108. Gusev VD, Nemytikova LA, Chuzhanova NA. On the complexity measures of genetic sequences. *Bioinformatics*. 1999;15(12):994-999. doi:10.1093/bioinformatics/15.12.994
109. Ziv J, Lempel A. Compression of Individual Sequences via Variable-Rate Coding. *IEEE Trans Inf Theory*. 1978;24(5):530-536. doi:10.1109/TIT.1978.1055934
110. Cover TM, Thomas JA. Elements of Information Theory 2nd Edition. 2006. Published 2006. Accessed September 3, 2021. http://www.citeulike.org/group/1710/article/1877660%5Cnhttp://www.amazon.com/Elements-Information-Edition-Telecommunications-Processing/dp/0471241954
111. Feder M, Merhav N. Relations Between Entropy and Error Probability. *IEEE Trans Inf Theory*. 1994;40(1):259-266. doi:10.1109/18.272494
112. Chen RC, Dewi C, Huang SW, Caraka RE. Selecting critical features for data classification based on machine learning methods. *J Big Data*. 2020;7(1):1-26. doi:10.1186/s40537-020-00327-4
113. Courel M, Clément Y, Bossevain C, et al. Gc content shapes mRNA storage and decay in human cells. *Elife*. 2019;8. doi:10.7554/eLife.49708
114. Solnick D. Alternative splicing caused by RNA secondary structure. *Cell*. 1985;43(3 PART 2):667-676. doi:10.1016/0092-8674(85)90239-9
115. Jin Y, Yang Y, Zhang P. New insights into RNA secondary structure in the alternative splicing of pre-mRNAs. *RNA Biol*. 2011;8(3):450-457. doi:10.4161/rna.8.3.15388
116. Mittendorf KF, Deatherage CL, Ohi MD, Sanders CR. Tailoring of membrane proteins by alternative splicing of Pre-mRNA. *Biochemistry*. 2012;51(28):5541-5556. doi:10.1021/bi3007065
117. Zou H, Li G. Diagnosis, prevention, and treatment of catheter-associated urinary tract infection in adults: 2009 international clinical practice guidelines from the Infectious Diseases Society of America. *Chinese J Infect Chemother*. Published online 2010.
118. Jiang W, Chen L. Alternative splicing: Human disease and quantitative analysis from high-throughput sequencing. *Comput Struct Biotechnol J*. 2021;19:183-195. doi:10.1016/J.CSBJ.2020.12.009
119. Conesa A, Madrigal P, Tarazona S, et al. A survey of best practices for RNA-

seq data analysis. *Genome Biol*. 2016;17(1):1-19. doi:10.1186/s13059-016-0881-8

120. Monteuuis G, Wong JJL, Bailey CG, Schmitz U, Rasko JEJ. The changing paradigm of intron retention: regulation, ramifications and recipes. *Nucleic Acids Res*. 2019;47(22):11497-11513. doi:10.1093/nar/gkz1068
121. Vanichkina DP, Schmitz U, Wong JJL, Rasko JEJ. Challenges in defining the role of intron retention in normal biology and disease. *Semin Cell Dev Biol*. 2018;75:40-49. doi:10.1016/j.semcdb.2017.07.030
122. Fields C, Sheng P, Miller B, Wei T, Xie M. Northern Blot with IR Fluorescent Probes: Strategies for Probe Preparation. *Bio-Protocol*. 2019;9(8). doi:10.21769/bioprotoc.3219
123. Streit S, Michalski CW, Erkan M, Kleeff J, Friess H. Northern blot analysis for detection and quantification of RNA in pancreatic cancer cells and tissues. *Nat Protoc*. 2009;4(1):37-43. doi:10.1038/nprot.2008.216
124. S.A. Deepak, K.R. Kottapalli, R. Rakwal, et al. Real-Time PCR: Revolutionizing Detection and Expression Analysis of Genes. *Curr Genomics*. 2007;8(4):234-251. doi:10.2174/138920207781386960
125. Tetrahymena thermophila (protozoa) cells | 2005 Photomicrography Competition | Nikon’s Small World. Accessed May 31, 2021. https://www.nikonsmallworld.com/galleries/2005-photomicrography-competition/tetrahymena-thermophila-protozoa-cells
126. Aubusson-Fleury A, Cohen J, Lemullois M. Ciliary heterogeneity within a single cell: The Paramecium model. *Methods Cell Biol*. 2015;127:457-485. doi:10.1016/bs.mcb.2014.12.007
127. Marziale F, Pucciarelli S, Ballarini P, et al. Different roles of two γ-tubulin isotypes in the cytoskeleton of the Antarctic ciliate Euplotes focardii: Remodelling of interaction surfaces may enhance microtubule nucleation at low temperature. *FEBS J*. 2008;275(21):5367-5382. doi:10.1111/j.1742-4658.2008.06666.x

APPENDIX

Appendix A: Publications Overview

All the articles published during my PhD are listed below. In addition to the article abstracts I reported the main techniques used in each one and my contributions. These techniques have been useful to broaden my knowledge in statistics, data science and microbiology as well as to lay the foundations for writing the final manuscript.

A.1 Data Science

A.1.1 Abstract I

A Real-World Setting Study: Which Glucose Meter Could Be the Best for POCT Use? An Easy and Applicable Protocol During the Hospital Routine

DOI: 10.1177/1932296818774077

Abstract

The aim of this retrospective study is to evaluate the reliability and robustness of six glucose meters for point-of-care testing in our wards using a brand-new protocol. During a 30-days study period a total of 50 diabetes patients were subjected to venous blood sampling and glucose meter blood analysis. The results of six glucose meters were compared with our laboratory reference assay. GlucoMen Plus (Menarini) with the 82% of acceptable results was the most robust glucose meter. Even if the Passing-Bablok analysis demonstrates the presence of constant systematic errors and the Bland-Altman test highlighted a possible overestimation, the surveillance error grid analysis showed that this glucose meter can be used safely. We proved that portable glucose meters are not always reliable in routinely clinical settings.

Technique’s overview:

- The passing–Bablok regression is a statistical method for non-parametric regression analysis suitable for method comparison studies
- *The Bland–Altman plot* (difference plot) in analytical chemistry or biomedicine is a method of data plotting used in analysing the agreement between two different assays.
- The coefficient of variation (CV) is a measure of relative dispersion used to comparing distributions that have not outliers
- The Surveillance Error Grid is a modern metric for clinical risk assessments of blood glucose monitor errors that assigns a unique risk score to each monitor data point when compared to a reference value.

Contributions:

Conceptualization, Methodology, Validation, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Review & Editing

A.1.2 Abstract II

CiliateGEM: an open-project and a tool for predictions of ciliate metabolic variations and experimental condition design

DOI: 10.1186/s12859-018-2422-9

Abstract

Background: The study of cell metabolism is becoming central in several fields such as biotechnology, evolution/adaptation and human disease investigations. Here we present CiliateGEM, the first metabolic network reconstruction draft of the freshwater ciliate Tetrahymena thermophila. We also provide the tools and resources to simulate different growth conditions and to predict metabolic variations. CiliateGEM can be extended to other ciliates in order to set up a meta-model, i.e. a metabolic network reconstruction valid for all ciliates. Ciliates are complex unicellular eukaryotes of presumably monophyletic origin, with a phylogenetic position that is equal from plants and animals. These cells represent a new concept of unicellular system with a high degree of species, population biodiversity and cell complexity. Ciliates perform in a single cell all the functions of a pluricellular organism, including locomotion, feeding, digestion, and sexual processes.

Results: After generating the model, we performed an in-silico simulation with the presence and absence of glucose. The lack of this nutrient caused a 32.1% reduction rate in biomass synthesis. Despite the glucose starvation, the growth did not stop due to the use of alternative carbon sources such as amino acids.

Conclusions: The future models obtained from CiliateGEM may represent a new approach to describe the metabolism of ciliates. This tool will be a useful resource for the ciliate research community in order to extend these species as model organisms in different research fields. An improved understanding of ciliate metabolism could be relevant to elucidate the basis of biological phenomena like genotype-phenotype relationships, population genetics, and cilia related disease mechanisms.

Techniques overview:

- The constraint-based Modelling is used for a quantitative prediction of cellular and multicellular biochemical networks

Contributions:

Conceptualization, Methodology, Validation, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Review & Editing

A.1.3 Abstract III

Differences between Community- and Hospital-acquired urinary tract infections in a tertiary care hospital

http://www.newmicrobiologica.org/PUB/allegati_pdf/2020/1/17.pdf

Abstract

The aim of this retrospective study was to highlight the differences in antibiotic resistance between Hospital-acquired and Community-acquired urinary tract infections (UTIs). Antimicrobial UTIs resistance data were collected from March 2011 to March 2018. Uropathogens were identified from 41,715 patients using routine laboratory methods. Differences in antibiotic resistance between Hospital and Community (non-hospitalized) patients were statistically validated. Odds ratio (OR) and p-values was used to determine whether a particular exposure (hospitalization) was a risk factor for a particular outcome (higher antibiotic resistance). We reported a general increase of unnecessary urine cultures in both community and hospital patients. The most representative microorganism isolated from Community (58.2%) and Hospital (47.6%) was E. coli. UTIs causative bacteria in hospitalized patients was more than twice as resistant to Trimetoprim/sulphamethoxazole (OR 2.26) and Imipenem (OR 2.56), for Gram-positive and Gram-negative, respectively, than in Community patients. Nitrofurantoin was the only agent without differences in resistance rate between community and hospital UTIs. Therefore, physicians could use it as a definitive therapy for uncomplicated cystitis and as a prophylactic agent for recurrent uncomplicated cystitis. With this work, we provided a general protocol

applicable by physicians to select the most suitable, if necessary, UTIs empiric treatment.

Technique's overview:

- The odds ratio (OR) is a statistic that quantifies the strength of the association between two events, A and B.
- The p-value. A very small p-value means that such an extreme observed outcome would be very unlikely under the null hypothesis.

Contributions:

Conceptualization, Methodology, Validation, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Review & Editing

A.2 Microbiology

A.2.1 Abstract IV

Synthesis of Bioactive Silver Nanoparticles by a Pseudomonas Strain Associated with the Antarctic Psychrophilic Protozoon *Euplotes focardii*

DOI: 10.3390/md18010038

Abstract

The synthesis of silver nanoparticles (AgNPs) by microorganisms recently gained a greater interest due to its potential to produce them in various sizes and morphologies. In this study, for AgNP biosynthesis, we used a new Pseudomonas strain isolated from a consortium associated with the Antarctic marine ciliate Euplotes focardii. After incubation of Pseudomonas cultures with 1 mM of AgNO3 at 22 °C, we obtained AgNPs within 24 h. Scanning electron (SEM) and transmission electron microscopy (TEM) revealed spherical polydispersed AgNPs in the size range of 20-70 nm. The average size was approximately 50 nm. Energy dispersive X-ray spectroscopy (EDS) showed the presence of a high intensity absorption peak at 3 keV, a distinctive property of nanocrystalline silver products. Fourier transform infrared (FTIR) spectroscopy found the presence of a high amount of AgNP-stabilizing proteins and other secondary metabolites. X-ray diffraction (XRD) revealed a face-centred cubic (fcc) diffraction spectrum with a crystalline nature. A comparative study between the chemically synthesized and Pseudomonas AgNPs revealed a higher antibacterial activity of the latter against common nosocomial pathogen microorganisms, including Escherichia coli, Staphylococcus aureus and Candida albicans. This study reports an efficient,

rapid synthesis of stable AgNPs by a new Pseudomonas strain with high antimicrobial activity.

Technique’s overview:

- Pathogen isolation from human specimens
- Pathogen identification with *VITEK 2* microbial identification system
- The disk diffusion test, or agar diffusion test, or Kirby–Bauer test (disc-diffusion antibiotic susceptibility test, disc-diffusion antibiotic sensitivity test, KB test), is an antibiotic susceptibility test. It uses antibiotic discs to test the extent to which bacteria are affected by those antibiotics.

Contributions:

Methodology, Validation and Investigation of the microbiological part. Writing - Review & Editing

A.2.2 Abstract V

Antibiotic activity of the antioxidant drink effective Microorganism-X (EM-X) extracts against common nosocomial pathogens: an in vitro study

DOI: 10.1080/14786419.2018.1517344

Abstract

EM-X is a mixed consortium of beneficial microorganisms of natural occurrence (lactic bacteria, yeast and photosynthetic bacteria). The aim of this study is to evaluate the antimicrobial activity in-vitro of EM-X to the principal pathogens isolated in clinical settings and to understand if it could be a suitable adjuvant to synthetic antibiotics. According the World Health Organization we performed antimicrobial assays to the main pathogens usually found in hospital wards. After antimicrobial testing, EM-X has been shown to be most effective at a concentration of 40 mg/ml four time concentrated than the commercial original solution (10 mg/ml). The EM-X antimicrobial action, although effective against bacteria, has proved to be ineffective against the candida genus. This active range of concentration (mg/ml) may prove a very weak action of EM, but further investigations will be done to separate the active substances form the inactive ones in this complex mixture.

Technique’s overview:

- Pathogen isolation from human specimens
- Pathogen identification with *VITEK 2* microbial identification system

- The the broth microdilution method appears to be an easy and reliable method for determination of the MICs of antibiotics
- freeze-drying and filtering of substances containing active ingredients

Contributions:

Conceptualization, Methodology, Validation, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Review & Editing

A.2.3 Abstract VI

Horizontal gene transfer and silver nanoparticles production in a new Marinomonas strain isolated from the Antarctic psychrophilic ciliate Euplotes focardii

DOI: 10.1038/s41598-020-66878-x

Abstract

We isolated a novel bacterial strain from a prokaryotic consortium associated to the psychrophilic marine ciliate Euplotes focardii, endemic of the Antarctic coastal seawater. The 16S rDNA sequencing and the phylogenetic analysis revealed the close evolutionary relationship to the Antarctic marine bacterium Marinomonas sp. BSw10506 and the sub antarctic Marinomonas polaris. We named this new strain Marinomonas sp. ef1. The optimal growth temperature in LB medium was 22 °C. Whole genome sequencing and analysis showed a reduced gene loss limited to regions encoding for transposases. Additionally, five genomic islands, e.g. DNA fragments that

facilitate horizontal gene transfer phenomena, were identified. Two open reading frames predicted from the genomic islands coded for enzymes belonging to the Nitro-FMN-reductase superfamily. One of these, the putative NAD(P)H nitroreductase YfkO, has been reported to be involved in the bioreduction of silver (Ag) ions and the production of silver nanoparticles (AgNPs). After the Marinomonas sp. ef1 biomass incubation with 1 mM of AgNO3 at 22 °C, we obtained AgNPs within 24 h. The AgNPs were relatively small in size (50 nm) and had a strong antimicrobial activity against twelve common nosocomial pathogenic microorganisms including Staphylococcus aureus and two Candida strains. To our knowledge, this is the first report of AgNPs biosynthesis by a Marinomonas strain. This biosynthesis may play a dual role in detoxification from silver nitrate and protection from pathogens for the bacterium and potentially for the associated ciliate. Biosynthetic AgNPs also represent a promising alternative to conventional antibiotics against common pathogens.

Techniques overview:

- Pathogen isolation from human specimens
- Pathogen identification with *VITEK 2* microbial identification system
- The disk diffusion test, or agar diffusion test, or Kirby–Bauer test (disc-diffusion antibiotic susceptibility test, disc-diffusion antibiotic sensitivity test, KB test), is an antibiotic susceptibility test. It uses antibiotic discs to test the extent to which bacteria are affected by those antibiotics.

Contributions:

Methodology, Validation and Investigation of the microbiological part. Writing - Review & Editing

A.2.4 Abstract VII

Biogenic Synthesis of Copper Nanoparticles Using Bacterial Strains Isolated from an Antarctic Consortium Associated to a Psychrophilic Marine Ciliate: Characterization and Potential Application as Antimicrobial Agents

DOI: 10.3390/md19050263

Abstract

In the last decade, metal nanoparticles (NPs) have gained significant interest in the field of biotechnology due to their unique physiochemical properties and potential uses in a wide range of applications. Metal NP synthesis using microorganisms has emerged as an eco-friendly, clean, and viable strategy alternative to chemical and physical approaches. Herein, an original and efficient route for the microbial synthesis of copper NPs using bacterial strains newly isolated from an Antarctic consortium is described. UV-visible spectra of the NPs showed a maximum absorbance in the range of 380–385 nm. Transmission electron microscopy analysis showed that these NPs are all monodispersed, spherical in nature, and well segregated without any agglomeration and with an average size of 30 nm. X-ray powder diffraction showed a polycrystalline nature and face centered cubic lattice and revealed characteristic diffraction peaks indicating the formation of CuONPs. Fourier-transform infrared

spectra confirmed the presence of capping proteins on the NP surface that act as stabilizers. All CuONPs manifested antimicrobial activity against various types of Gram-negative; Gram-positive bacteria; and fungi pathogen microorganisms including Escherichia coli, Staphylococcus aureus, and Candida albicans. The cost-effective and eco-friendly biosynthesis of these CuONPs make them particularly attractive in several application from nanotechnology to biomedical science.

Techniques overview:

- Pathogen isolation from human specimens
- Pathogen identification with *VITEK 2* microbial identification system
- *The disk diffusion test, agar diffusion test, or Kirby–Bauer test (disc-diffusion antibiotic susceptibility test, disc-diffusion antibiotic sensitivity test, KB test), is an antibiotic susceptibility test. It uses antibiotic discs to test the extent to which bacteria are affected by those antibiotics.*

Contributions:

Methodology, Validation and Investigation of the microbiological part. Writing - Review & Editing

Appendix B: Patents

These four patents are related to the field of microbiology and concerns the characterization of four novel bacteria strains, isolated in our laboratories and deposited, belonging to *Marinomonas, Rhodococcus, Bacillus* and *Brevundimonas* genera, and the use of these bacteria for the in-vitro production of useful metabolites. These bacteria are cold tolerant and can growth in a wide range of temperatures, from 4 °C to 28 °C.

Patent-102019000014121 **(granted)**: in the presence of glucose and peptone in the culture medium, these bacteria produce fluorescent molecules that can be exploited as natural dyes. Moreover, in the presence of silver nitrate, these strains are able of silver nanoparticles biosynthesis. These nanoparticles show high antimicrobial activity against pathogenic gram-positive and gram-negative bacteria and against pathogenic yeast such as *Candida*. The production of the silver nanoparticles is rapid and low-cost. Furthermore, the biosynthesis is environmentally friendly since does not imply the use of toxic substances for nanoparticles purification. In particular, we aim in the production of material such as tissues or sponges that contain the bio-produced silver nanoparticles to be used to sterilize surfaces from pathogenic agents.

Patent-102019000024493: the patent reports a method for removing fuels from contaminated water by adding the bacterial strain Rhodococcus ef1 to diesel contaminated water long enough to grow and produce a natural phenazin-like dye and surfactants that trap and remove diesel. If *Marinomonas* ef1 and/or *Rhodococcus* ef1 are added to water contaminated with CrIII and CrVI in the presence of CrSO4 and K2CrO4, these toxic salts are transformed into nanoparticles containing Cr reduced, harmless to humans and the environment. Finally, the addition of the bacterial strain *Rhodococcus* ef1 to cocaine-contaminated water allows the bacterial strain *Rhodococcus* ef1 to degrade cocaine within 24 hours.

Both are protected internationally under the Patent Cooperation Treaty (PTC).

Patent-102019000014121 (submitted): the patent reports two bacteria able to synthesize biocellulose under room temperature, without energy expenditure. Usually this procedure take place at 30°C. No specific salt solutions are used and the culture medium is not a limiting factor. Different culture media can be used without affecting the success of the biocellulose production process. The culture media is usually mixed with glucose and mannitol, the former in concentrations of at least 3%. Our bacteria need only 1.5% of glucose.

Patent-102021000017333 (submitted): the patent reports two bacteria able to synthesize a new resin and biocellulose under room temperature, without energy expenditure. The culture media is usually mixed with sugar of fat, deriving from food waste.

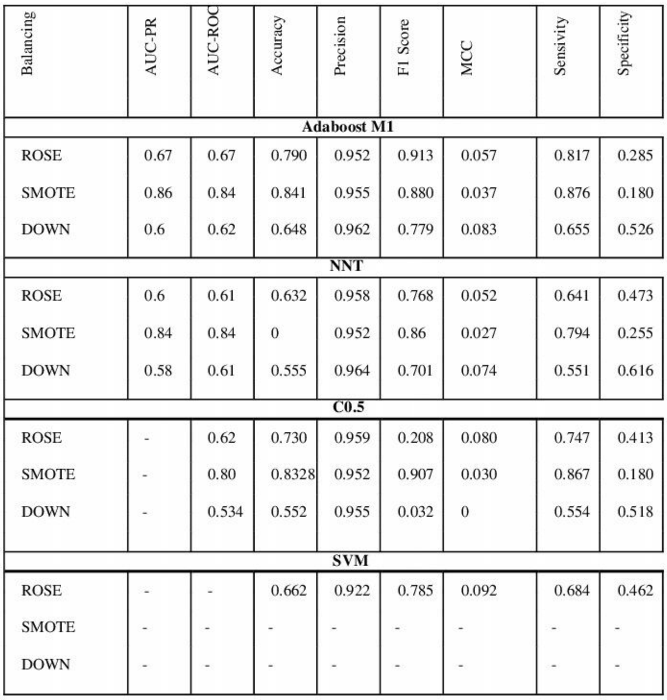
Appendix C: Supplementary Materials

Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
Adaboost M1								
ROSE	0.74	0.76	0.261	0.974	0.284	0.121	0.166	0.967
SMOTE	0.87	0.86	0.780	0.884	0.873	0.025	0.836	0.163
DOWN	0.59	0.59	0.594	0.896	0.921	0.078	0	0
NNT								
ROSE	0.77	0.79	0.208	0.960	0.190	0.080	0.106	0.967
SMOTE	0.84	0.84	0.731	0.884	0.840	0.020	0.800	0.225
DOWN	0.59	0.62	0.493	0.918	0.618	0.104	0.466	0.694
C0.5								
ROSE	-	0.792	0.224	0.957	0.222	0.085	0.125	0.958
SMOTE	-	0.830	0.794	0.882	0.883	0.006	0.885	0.121
DOWN	-	0.599	0.458	0.882	0.586	0.047	0.434	0.638
SVM								
ROSE	-	-	-	-	-	-	-	-
SMOTE	0.82	0.83	0.788	0.883	0.879	0.006	0.874	0.148
DOWN	0.57	0.59	0.477	0.909	0.603	0.078	0.452	0.667

Table A.1 ***T.thermophila*** model evaluation using ASTALAVISTA intron extraction labelling

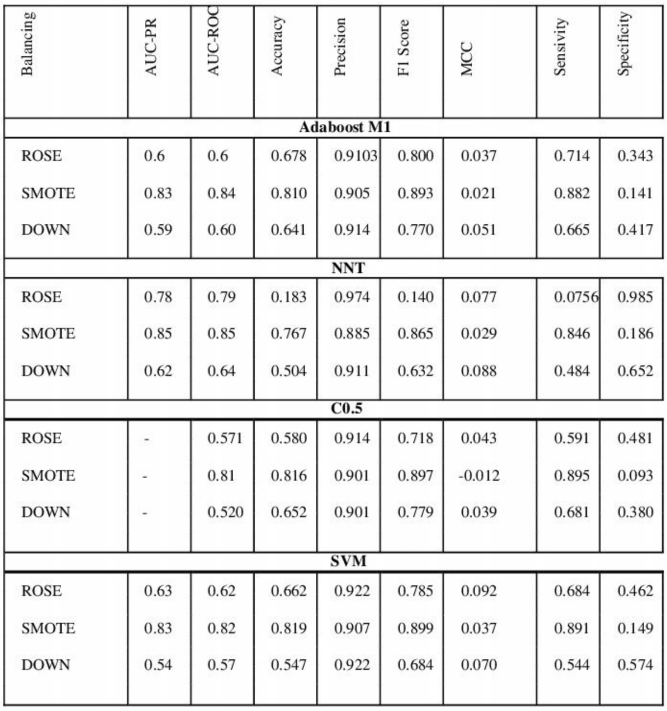
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
Adaboost M1								
ROSE	0.8	0.82	0.129	0.985	0.103	0.055	0.054	0.990
SMOTE	0.85	0.84	0.862	0.925	0.925	0.068	0.925	0.142
DOWN	0.57	0.57	0.524	0.931	0.667	0.045	0.520	0.564
NNT								
ROSE	0.82	0.84	0.124	0.983	0.092	0.051	0.048	0.990
SMOTE	0.79	0.80	0,74	0.918	0.852	-0.008	0.79	0.19
DOWN	0.54	0.58	0.385	0.947	0.512	0.072	0.351	0.775
C0.5								
ROSE	-	0.78	0.182	0	0.208	0.045	0.117	0.93
SMOTE	-	0.831	0.868	0.920	0.929	0.003	0.939	0.064
DOWN	-	0.538	0.351	0.947	0.468	0.067	0.311	0.802

Table A.2 ***T.malaccensis*** model evaluation using ASTALAVISTA intron extraction labelling



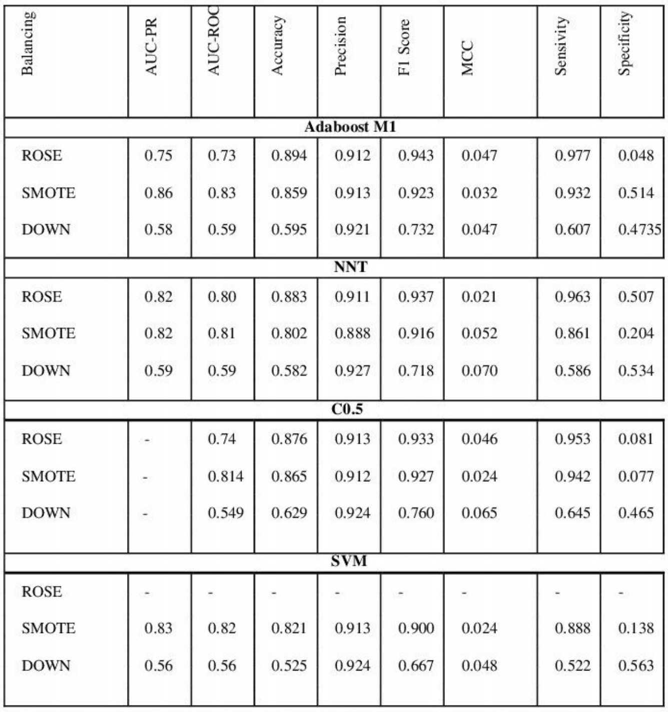
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
Adaboost M1								
ROSE	0.67	0.67	0.790	0.952	0.913	0.057	0.817	0.285
SMOTE	0.86	0.84	0.841	0.955	0.880	0.037	0.876	0.180
DOWN	0.6	0.62	0.648	0.962	0.779	0.083	0.655	0.526
NNT								
ROSE	0.6	0.61	0.632	0.958	0.768	0.052	0.641	0.473
SMOTE	0.84	0.84	0	0.952	0.86	0.027	0.794	0.255
DOWN	0.58	0.61	0.555	0.964	0.701	0.074	0.551	0.616
C0.5								
ROSE	-	0.62	0.730	0.959	0.208	0.080	0.747	0.413
SMOTE	-	0.80	0.8328	0.952	0.907	0.030	0.867	0.180
DOWN	-	0.534	0.552	0.955	0.032	0	0.554	0.518
SVM								
ROSE	-	-	0.662	0.922	0.785	0.092	0.684	0.462
SMOTE	-	-	-	-	-	-	-	-
DOWN	-	-	-	-	-	-	-	-

Table A.3 ***T.borealis*** model evaluation using ASTALAVISTA intron extraction labelling



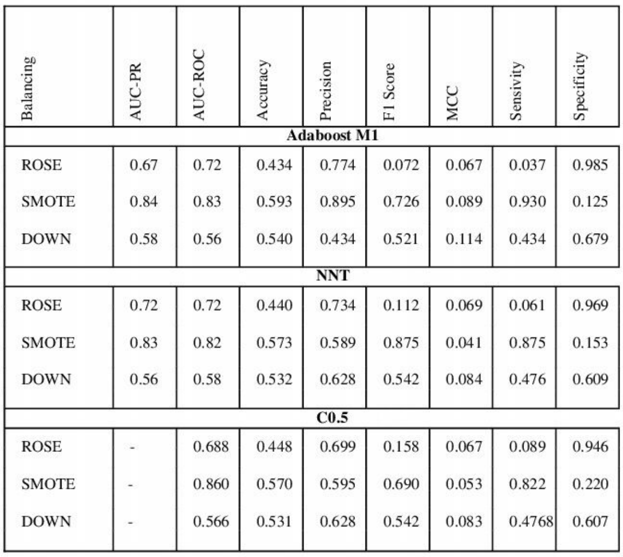
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
Adaboost M1								
ROSE	0.6	0.6	0.678	0.9103	0.800	0.037	0.714	0.343
SMOTE	0.83	0.84	0.810	0.905	0.893	0.021	0.882	0.141
DOWN	0.59	0.60	0.641	0.914	0.770	0.051	0.665	0.417
NNT								
ROSE	0.78	0.79	0.183	0.974	0.140	0.077	0.0756	0.985
SMOTE	0.85	0.85	0.767	0.885	0.865	0.029	0.846	0.186
DOWN	0.62	0.64	0.504	0.911	0.632	0.088	0.484	0.652
C0.5								
ROSE	-	0.571	0.580	0.914	0.718	0.043	0.591	0.481
SMOTE	-	0.81	0.816	0.901	0.897	-0.012	0.895	0.093
DOWN	-	0.520	0.652	0.901	0.779	0.039	0.681	0.380
SVM								
ROSE	0.63	0.62	0.662	0.922	0.785	0.092	0.684	0.462
SMOTE	0.83	0.82	0.819	0.907	0.899	0.037	0.891	0.149
DOWN	0.54	0.57	0.547	0.922	0.684	0.070	0.544	0.574

Table A.4 ***T.elliotti*** model evaluation using ASTALAVISTA intron extraction labelling



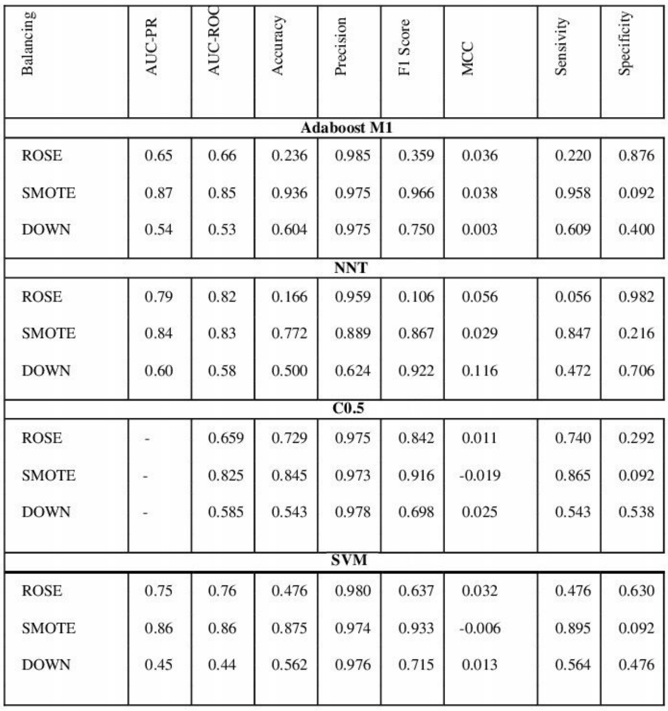
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
Adaboost M1								
ROSE	0.75	0.73	0.894	0.912	0.943	0.047	0.977	0.048
SMOTE	0.86	0.83	0.859	0.913	0.923	0.032	0.932	0.514
DOWN	0.58	0.59	0.595	0.921	0.732	0.047	0.607	0.4735
NNT								
ROSE	0.82	0.80	0.883	0.911	0.937	0.021	0.963	0.507
SMOTE	0.82	0.81	0.802	0.888	0.916	0.052	0.861	0.204
DOWN	0.59	0.59	0.582	0.927	0.718	0.070	0.586	0.534
C0.5								
ROSE	-	0.74	0.876	0.913	0.933	0.046	0.953	0.081
SMOTE	-	0.814	0.865	0.912	0.927	0.024	0.942	0.077
DOWN	-	0.549	0.629	0.924	0.760	0.065	0.645	0.465
SVM								
ROSE	-	-	-	-	-	-	-	-
SMOTE	0.83	0.82	0.821	0.913	0.900	0.024	0.888	0.138
DOWN	0.56	0.56	0.525	0.924	0.667	0.048	0.522	0.563

Table A.5 ***P.tetraurelia*** model evaluation using ASTALAVISTA introns extraction labelling



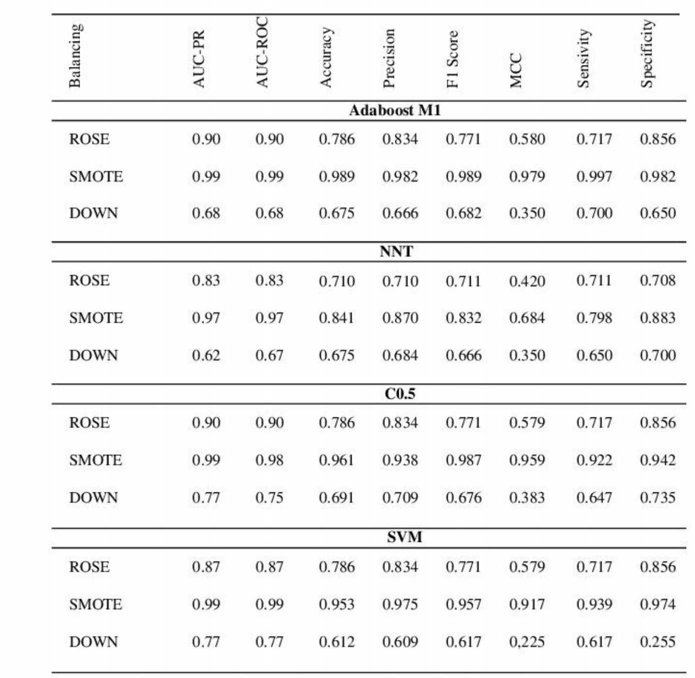
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
Adaboost M1								
ROSE	0.67	0.72	0.434	0.774	0.072	0.067	0.037	0.985
SMOTE	0.84	0.83	0.593	0.895	0.726	0.089	0.930	0.125
DOWN	0.58	0.56	0.540	0.434	0.521	0.114	0.434	0.679
NNT								
ROSE	0.72	0.72	0.440	0.734	0.112	0.069	0.061	0.969
SMOTE	0.83	0.82	0.573	0.589	0.875	0.041	0.875	0.153
DOWN	0.56	0.58	0.532	0.628	0.542	0.084	0.476	0.609
C0.5								
ROSE	-	0.688	0.448	0.699	0.158	0.067	0.089	0.946
SMOTE	-	0.860	0.570	0.595	0.690	0.053	0.822	0.220
DOWN	-	0.566	0.531	0.628	0.542	0.083	0.4768	0.607

Table A.6 ***E.vannus*** model evaluation using ASTALAVISTA introns extraction labelling



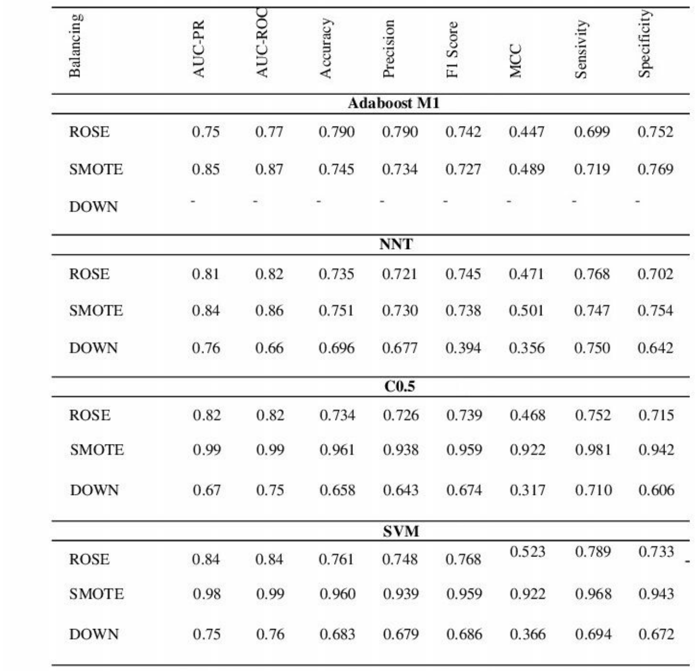
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
Adaboost M1								
ROSE	0.65	0.66	0.236	0.985	0.359	0.036	0.220	0.876
SMOTE	0.87	0.85	0.936	0.975	0.966	0.038	0.958	0.092
DOWN	0.54	0.53	0.604	0.975	0.750	0.003	0.609	0.400
NNT								
ROSE	0.79	0.82	0.166	0.959	0.106	0.056	0.056	0.982
SMOTE	0.84	0.83	0.772	0.889	0.867	0.029	0.847	0.216
DOWN	0.60	0.58	0.500	0.624	0.922	0.116	0.472	0.706
C0.5								
ROSE	-	0.659	0.729	0.975	0.842	0.011	0.740	0.292
SMOTE	-	0.825	0.845	0.973	0.916	-0.019	0.865	0.092
DOWN	-	0.585	0.543	0.978	0.698	0.025	0.543	0.538
SVM								
ROSE	0.75	0.76	0.476	0.980	0.637	0.032	0.476	0.630
SMOTE	0.86	0.86	0.875	0.974	0.933	-0.006	0.895	0.092
DOWN	0.45	0.44	0.562	0.976	0.715	0.013	0.564	0.476

Table A.7 ***E.focardii***: model evaluation using ASTALAVISTA introns extraction labelling



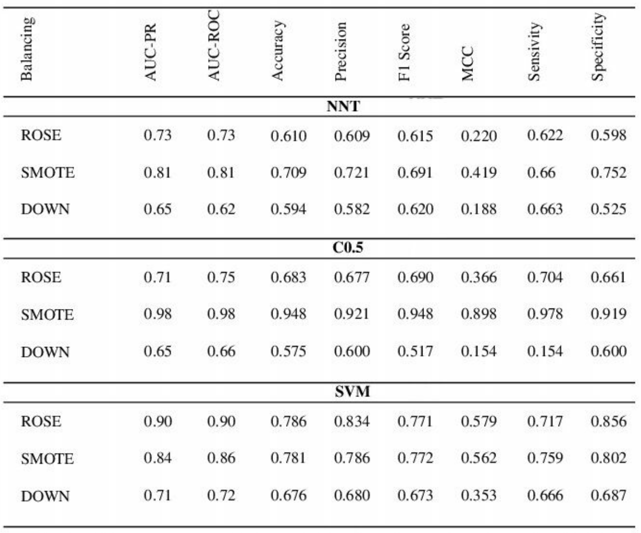
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
Adaboost M1								
ROSE	0.90	0.90	0.786	0.834	0.771	0.580	0.717	0.856
SMOTE	0.99	0.99	0.989	0.982	0.989	0.979	0.997	0.982
DOWN	0.68	0.68	0.675	0.666	0.682	0.350	0.700	0.650
NNT								
ROSE	0.83	0.83	0.710	0.710	0.711	0.420	0.711	0.708
SMOTE	0.97	0.97	0.841	0.870	0.832	0.684	0.798	0.883
DOWN	0.62	0.67	0.675	0.684	0.666	0.350	0.650	0.700
C0.5								
ROSE	0.90	0.90	0.786	0.834	0.771	0.579	0.717	0.856
SMOTE	0.99	0.98	0.961	0.938	0.987	0.959	0.922	0.942
DOWN	0.77	0.75	0.691	0.709	0.676	0.383	0.647	0.735
SVM								
ROSE	0.87	0.87	0.786	0.834	0.771	0.579	0.717	0.856
SMOTE	0.99	0.99	0.953	0.975	0.957	0.917	0.939	0.974
DOWN	0.77	0.77	0.612	0.609	0.617	0,225	0.617	0.255

Table B.1 ***T.thermophila***: model evaluation using iRead introns extraction labelling



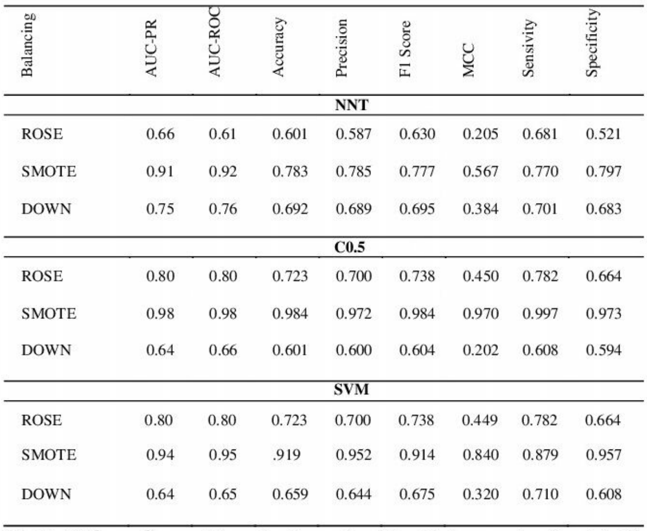
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
Adaboost M1								
ROSE	0.75	0.77	0.790	0.790	0.742	0.447	0.699	0.752
SMOTE	0.85	0.87	0.745	0.734	0.727	0.489	0.719	0.769
DOWN	-	-	-	-	-	-	-	-
NNT								
ROSE	0.81	0.82	0.735	0.721	0.745	0.471	0.768	0.702
SMOTE	0.84	0.86	0.751	0.730	0.738	0.501	0.747	0.754
DOWN	0.76	0.66	0.696	0.677	0.394	0.356	0.750	0.642
C0.5								
ROSE	0.82	0.82	0.734	0.726	0.739	0.468	0.752	0.715
SMOTE	0.99	0.99	0.961	0.938	0.959	0.922	0.981	0.942
DOWN	0.67	0.75	0.658	0.643	0.674	0.317	0.710	0.606
SVM								
ROSE	0.84	0.84	0.761	0.748	0.768	0.523	0.789	0.733
SMOTE	0.98	0.99	0.960	0.939	0.959	0.922	0.968	0.943
DOWN	0.75	0.76	0.683	0.679	0.686	0.366	0.694	0.672

Table B.2 ***T.elliotti***: model evaluation using iRead introns extraction labelling



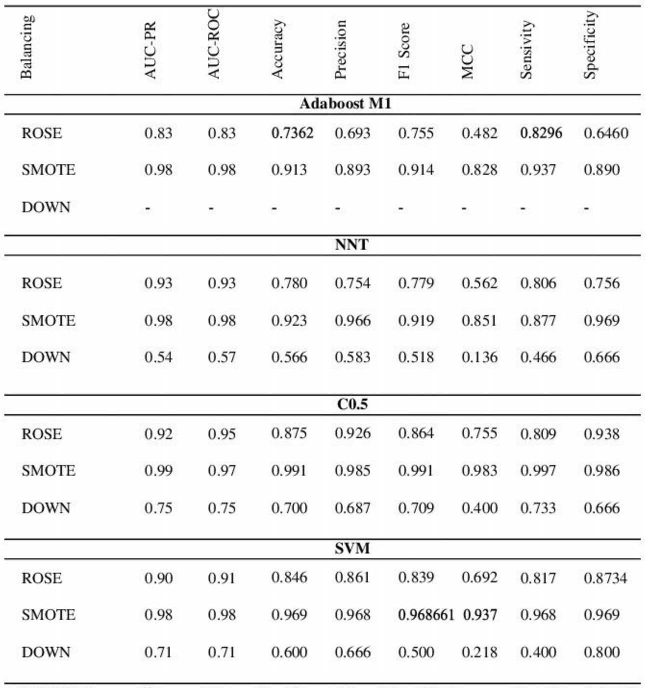
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
				NNT				
ROSE	0.73	0.73	0.610	0.609	0.615	0.220	0.622	0.598
SMOTE	0.81	0.81	0.709	0.721	0.691	0.419	0.66	0.752
DOWN	0.65	0.62	0.594	0.582	0.620	0.188	0.663	0.525
				C0.5				
ROSE	0.71	0.75	0.683	0.677	0.690	0.366	0.704	0.661
SMOTE	0.98	0.98	0.948	0.921	0.948	0.898	0.978	0.919
DOWN	0.65	0.66	0.575	0.600	0.517	0.154	0.154	0.600
				SVM				
ROSE	0.90	0.90	0.786	0.834	0.771	0.579	0.717	0.856
SMOTE	0.84	0.86	0.781	0.786	0.772	0.562	0.759	0.802
DOWN	0.71	0.72	0.676	0.680	0.673	0.353	0.666	0.687

Table B.3 ***T.malaccensis***: evaluation models using iRead introns extraction labelling



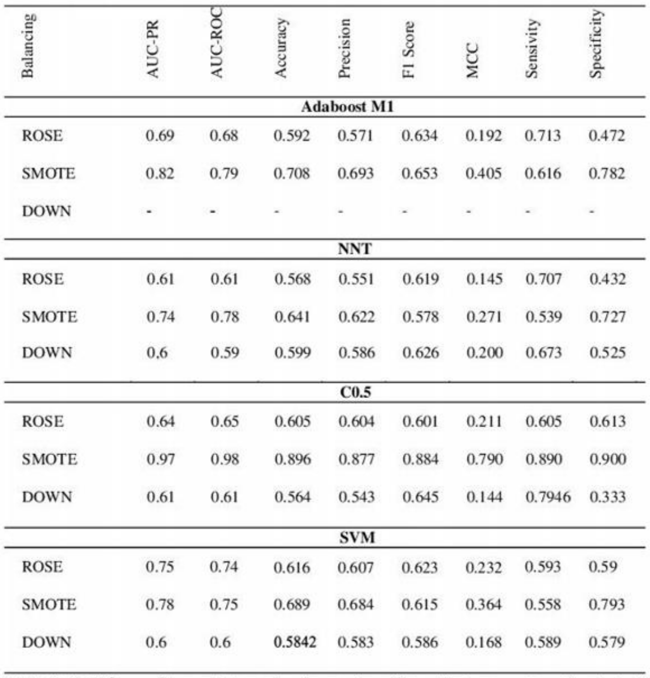
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
				NNT				
ROSE	0.66	0.61	0.601	0.587	0.630	0.205	0.681	0.521
SMOTE	0.91	0.92	0.783	0.785	0.777	0.567	0.770	0.797
DOWN	0.75	0.76	0.692	0.689	0.695	0.384	0.701	0.683
				C0.5				
ROSE	0.80	0.80	0.723	0.700	0.738	0.450	0.782	0.664
SMOTE	0.98	0.98	0.984	0.972	0.984	0.970	0.997	0.973
DOWN	0.64	0.66	0.601	0.600	0.604	0.202	0.608	0.594
				SVM				
ROSE	0.80	0.80	0.723	0.700	0.738	0.449	0.782	0.664
SMOTE	0.94	0.95	.919	0.952	0.914	0.840	0.879	0.957
DOWN	0.64	0.65	0.659	0.644	0.675	0.320	0.710	0.608

Table B.4 ***T.borealis***: model evaluation using iRead introns extraction labelling



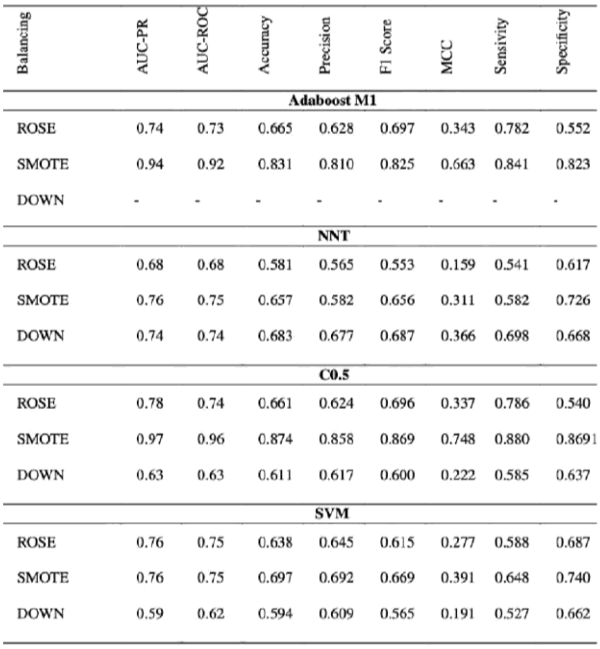
Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
				Adaboost M1				
ROSE	0.83	0.83	0.7362	0.693	0.755	0.482	0.8296	0.6460
SMOTE	0.98	0.98	0.913	0.893	0.914	0.828	0.937	0.890
DOWN	-	-	-	-	-	-	-	-
				NNT				
ROSE	0.93	0.93	0.780	0.754	0.779	0.562	0.806	0.756
SMOTE	0.98	0.98	0.923	0.966	0.919	0.851	0.877	0.969
DOWN	0.54	0.57	0.566	0.583	0.518	0.136	0.466	0.666
				C0.5				
ROSE	0.92	0.95	0.875	0.926	0.864	0.755	0.809	0.938
SMOTE	0.99	0.97	0.991	0.985	0.991	0.983	0.997	0.986
DOWN	0.75	0.75	0.700	0.687	0.709	0.400	0.733	0.666
				SVM				
ROSE	0.90	0.91	0.846	0.861	0.839	0.692	0.817	0.8734
SMOTE	0.98	0.98	0.969	0.968	0.968661	0.937	0.968	0.969
DOWN	0.71	0.71	0.600	0.666	0.500	0.218	0.400	0.800

Table B.5 ***P.tetraurelia***: model evaluation using iRead introns extraction labelling



Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
				Adaboost M1				
ROSE	0.69	0.68	0.592	0.571	0.634	0.192	0.713	0.472
SMOTE	0.82	0.79	0.708	0.693	0.653	0.405	0.616	0.782
DOWN	-	-	-	-	-	-	-	-
				NNT				
ROSE	0.61	0.61	0.568	0.551	0.619	0.145	0.707	0.432
SMOTE	0.74	0.78	0.641	0.622	0.578	0.271	0.539	0.727
DOWN	0.6	0.59	0.599	0.586	0.626	0.200	0.673	0.525
				C0.5				
ROSE	0.64	0.65	0.605	0.604	0.601	0.211	0.605	0.613
SMOTE	0.97	0.98	0.896	0.877	0.884	0.790	0.890	0.900
DOWN	0.61	0.61	0.564	0.543	0.645	0.144	0.7946	0.333
				SVM				
ROSE	0.75	0.74	0.616	0.607	0.623	0.232	0.593	0.59
SMOTE	0.78	0.75	0.689	0.684	0.615	0.364	0.558	0.793
DOWN	0.6	0.6	0.5842	0.583	0.586	0.168	0.589	0.579

Table B.6 ***E.focardii***: model evaluation using iRead introns extraction labelling



Balancing	AUC-PR	AUC-ROC	Accuracy	Precision	F1 Score	MCC	Sensivity	Specificity
				Adaboost M1				
ROSE	0.74	0.73	0.665	0.628	0.697	0.343	0.782	0.552
SMOTE	0.94	0.92	0.831	0.810	0.825	0.663	0.841	0.823
DOWN	-	-	-	-	-	-	-	-
				NNT				
ROSE	0.68	0.68	0.581	0.565	0.553	0.159	0.541	0.617
SMOTE	0.76	0.75	0.657	0.582	0.656	0.311	0.582	0.726
DOWN	0.74	0.74	0.683	0.677	0.687	0.366	0.698	0.668
				C0.5				
ROSE	0.78	0.74	0.661	0.624	0.696	0.337	0.786	0.540
SMOTE	0.97	0.96	0.874	0.858	0.869	0.748	0.880	0.8691
DOWN	0.63	0.63	0.611	0.617	0.600	0.222	0.585	0.637
				SVM				
ROSE	0.76	0.75	0.638	0.645	0.615	0.277	0.588	0.687
SMOTE	0.76	0.75	0.697	0.692	0.669	0.391	0.648	0.740
DOWN	0.59	0.62	0.594	0.609	0.565	0.191	0.527	0.662

Table B.7 ***E,vannus***: model evaluation using iRead introns extraction labelling