Cellular Localization Prediction

Can the cellular localization Sites of proteins be predicted using probabilistic classification.

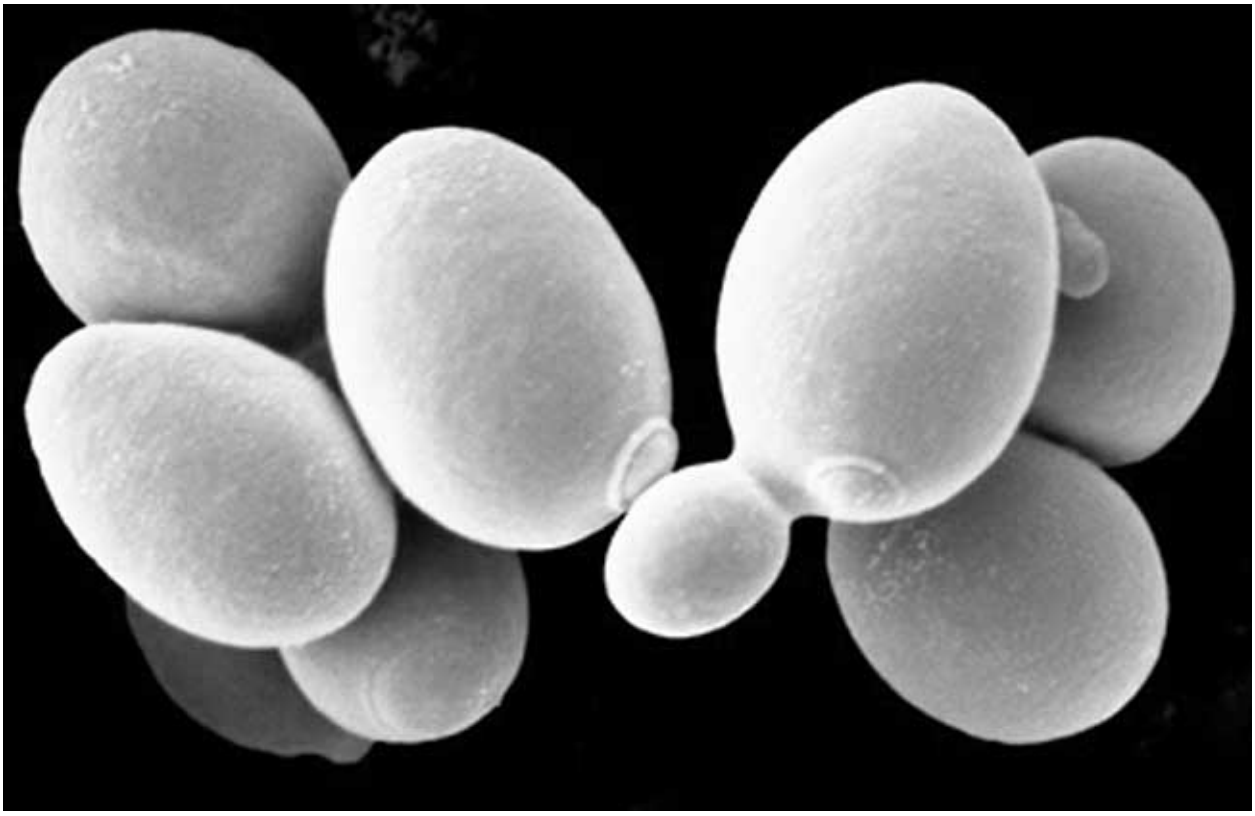

Figure 1: https://www.hj9l.com/zhidao/changshi/13854.html

John Busker
352905
Bio-Informatics
Dave Langers, and Bart Barnard
28-10-2022

Table of Contents

Introduction

Protein activities and their subcellular locations in cells are strongly connected. Finding proteins' positions within cells is frequently the initial step in understanding them, and their sites serve as a model for medication design. The amino acid sequence of a protein mostly determines where it will localize within a cell. Using this knowledge, a software program has been developed to define a simple model for the classification of yeast localization in cells. The efficient arrangement of biological data into databases and the creation of algorithms, such as string algorithms, to manipulate that data have largely defined the area of computational biology. The dataset consists of 1484 yeast sequences from SWISS-PROT using the annotations from YPD.

Research question: How accurate can the cellular localization sites of proteins be predicted using probabilistic classification?

Materials and Methods

The data has been retrieved from uci. The dataset consists of 1484 yeast sequences from SWISS-PROT using the annotations from YPD. In this section, we are going to talk about the results we found from the dataset.

Yeast proteins were classified into ten classes: **cytoplasmic:** cytoskeletal (CYT); nuc]ear (NUC); vacuolar (VAC); mitochondrial (MIT); isomal (POX); **extracellular:** including those localized against the cell wall (EXC); proteins localized to the lumen of the endoplasmic reticulum (ERL); membrane proteins with a cleaved signal (ME1); membrane proteins with an uncleared signal (ME2); and membrane proteins with no N-terminal sign (ME3), where ME1, ME2,and ME3 proteins may be localized to the plasma membrane, the endoplasmic reticulum membrane, or the membrane of a golgi body.

Eight features were used in classification. Here below are all the features

- **mcg**: McGeoch's method for signal sequence recognition.
- **gvh**: von Heijne's method for signal sequence recognition.
- **alm**: Score of the ALOM membrane spanning region prediction program.
- **mit**: Score of discriminant analysis of the amino acid content of the N-terminal region (20 residues long) of mitochondrial and non-mitochondrial proteins.
- **erl**: Presence of "HDEL" substring (thought to act as a signal for retention in the endoplasmic reticulum lumen). Binary attribute.
- **pox**: Peroxisomal targeting signal in the C-terminus.
- **vac**: Score of discriminant analysis of the amino acid content of vacuolar and extracellular proteins.
- **nuc**: Score of discriminant analysis of nuclear localization signals of nuclear and non-nuclear proteins.

It is always important to look at the data and to look at the quality of the data. No proper conclusion can be drawn from low-quality data. For recurrent errors, the data can be corrected. The data needs to be modified to make sure it is of the highest quality to perform the analysis on. This is achieved by performing exploratory data analysis on the dataset. Modifying and removing parts of the dataset to get the best possible dataset. A RandomForest classifier has been chosen for the classification of the unknown instances. The parameters have been optimized for usability and speed. See the logjournal [1].

A Java application has been developed. The software using Weka carried out the calculations required to carry out the probabilistic inference. The application can quickly classify new instances given from a file or from the command line. See the repo [2] for the code.

Results

To create a great dataset, changes need to be made to the downloaded dataset. The dataset consists of 1484 yeast sequences from SWISS-PROT using the annotations from YPD. In this section, we are going to talk about the results we found from the dataset [3].

Codebook

As it is a large dataset, it is wise to look at the codebook first. However, there is no codebook. Nevertheless, there is a file with more information on all the attributes. This is where the information is extracted to make your own codebook.

Below is a table with a description of all the attributes' abbreviations, explanations, and data types.

Name	Fullname	Datatypes
seq.name	Accession number for the SWISS-PROT database	str
mcg	McGeoch's method for signal sequence recognition	double
gvh	von Heijne's method for signal sequence recognition	double
alm	Score of the ALOM membrane spanning region prediction program	double
mit	Score of discriminant analysis of the amino acid content of the N-terminal region	double
erl	Presence of 'HDEL' substring	double
pox	Peroxisomal targeting signal in the C-terminus	logical
vac	Score of discriminant analysis of the amino acid content of vacuolar and extracellular proteins	double
nuc	Score of discriminant analysis of nuclear localization signals of nuclear and non-nuclear proteins	double
loc.site	The class is the localization site	factor

Table 1: The codebook.

The last column is the sequence's localization site. There are ten different possibilities for this. Yeast proteins were classified into ten classes: **cytoplasmic:** CYT; NUC; VAC; MIT; POX; **extracellular:** EXC; ERL; ME1; ME2; ME3.

Here are the ten in question:

Abbreviation	Fullname	Amount
CYT	cytosolic or cytoskeletal	463
NUC	nuclear	429
MIT	mitochondrial	244
ME3	membrane protein, no N-terminal signal	163
ME2	membrane protein, uncleaved signal	51
ME1	membrane protein, cleaved signal	44
EXC	extracellular	37
VAC	vacuolar	30
POX	peroxisomal	20
ERL	endoplasmic reticulum lumen	5

Table 2: Sequence localization sites.

There are alot of CYT and NUC localizations. ERL localization is the least. There are only five of these in the dataset.

Dataset

Eight features were used in classification: the presence or absence of an HDEL pattern as a signal for retention in the endoplasmic reticulum lumen (erl); The results of discriminant analysis on the amino acid content of vacuolar and extracellular proteins (vac); the result of discriminant analysis on the amino acid composition of the 20-residue N-terminal region of mitochondrial and non-mitochondrial proteins (mit); the presence or absence of nuclear localization consensus patterns combined with a term reflecting the frequency of basic residues (nuc); and some combination of the presence of a short sequence motif and the result of discriminant analysis of the amino acid composition of the protein sequence (pox) [3].

seq.name	mcg	gvh	alm	mit	erl	pox	vac	nuc	loc.site
ADT1_YEAST	0.58	0.61	0.47	0.13	0.5	0	0.48	0.22	MIT
ADT2_YEAST	0.43	0.67	0.48	0.27	0.5	0	0.53	0.22	MIT
ADT3_YEAST	0.64	0.62	0.49	0.15	0.5	0	0.53	0.22	MIT
AAR2_YEAST	0.58	0.44	0.57	0.13	0.5	0	0.54	0.22	NUC
AATM_YEAST	0.42	0.44	0.48	0.54	0.5	0	0.48	0.22	MIT
AATC_YEAST	0.51	0.4	0.56	0.17	0.5	0.5	0.49	0.22	CYT

Table 3: First six rows of the dataset.

As seen from the table, the loaded dataset. It has a char datatype for the first and last column. And every other column has a double datatype.

There are 1484 rows and 10 columns.

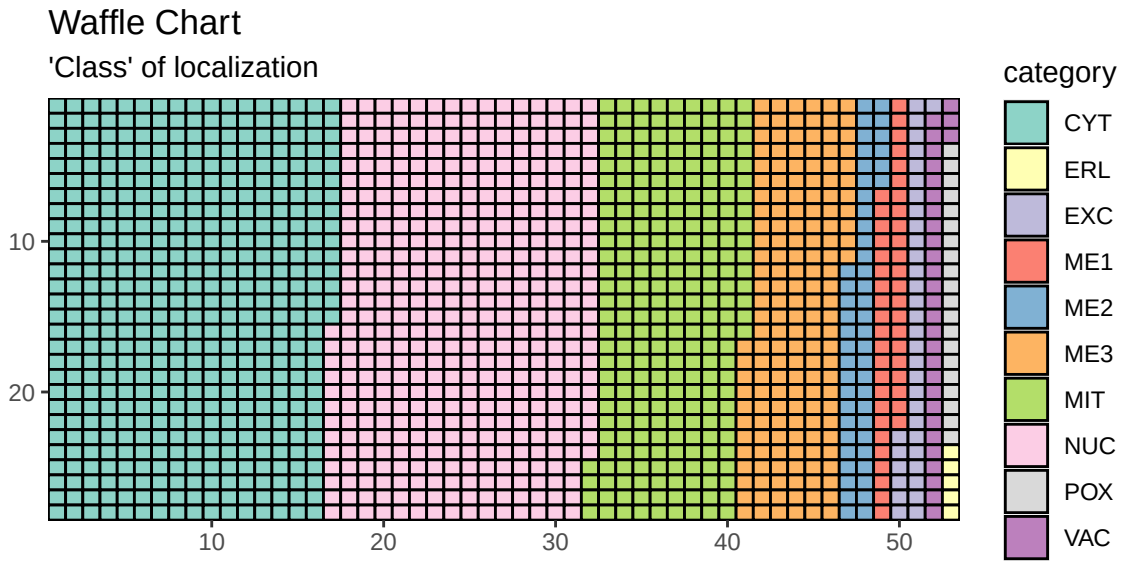

Figure 2: A waffle chart of the categorical composition

Figure 2 shows the number of each classification of the dataset. There are alot of CYT and NUC localization. And there are only 5 of ERL.

To ensure the quality of the data. The data had to be transformed before it could be worked with.

The first column has been dropped since it is not necessary. Since the sequence names contribute nothing to creating a prediction model.

mcg	gvh	alm	mit	erl	pox	vac	nuc	loc.site
0.58	0.61	0.47	0.13	0.5	0	0.48	0.22	MIT
0.43	0.67	0.48	0.27	0.5	0	0.53	0.22	MIT
0.64	0.62	0.49	0.15	0.5	0	0.53	0.22	MIT
0.58	0.44	0.57	0.13	0.5	0	0.54	0.22	NUC
0.42	0.44	0.48	0.54	0.5	0	0.48	0.22	MIT
0.51	0.4	0.56	0.17	0.5	0.5	0.49	0.22	CYT

Table 4: First six rows of the dataset with the first column dropped.

As stated on the website there are 0 missing values. This has been confirmed using R.

It can be more convenient to have a variable in a factor form instead of numeric form. The data type of the column loc.site has been successfully changed to factors.

The ERL column has changed. from a double to a logical datatype. As seen in table 5 below, only numbers 1 and 0.5 appear, but for a logical datatype you need 1 and 0. All 0.5 have been changed to 0.

0.5	1
1470	14

Table 5: A table showing distrubtion of the ERL column.

mcg	gvh	alm	mit	erl	pox	vac	nuc	loc.site
double	double	double	double	logical	double	double	double	factor

Table 6: Each column name and the typeof the datatype.

Each column has been changed so that it has the right data type.

At pox, vac and nuc, you don't see shifted peaks. At mcg and gvh you really see shifted peaks this shows a distribution of the different classes. There are small shifted peaks at alm en mit, this shows that there is difference but not so much.

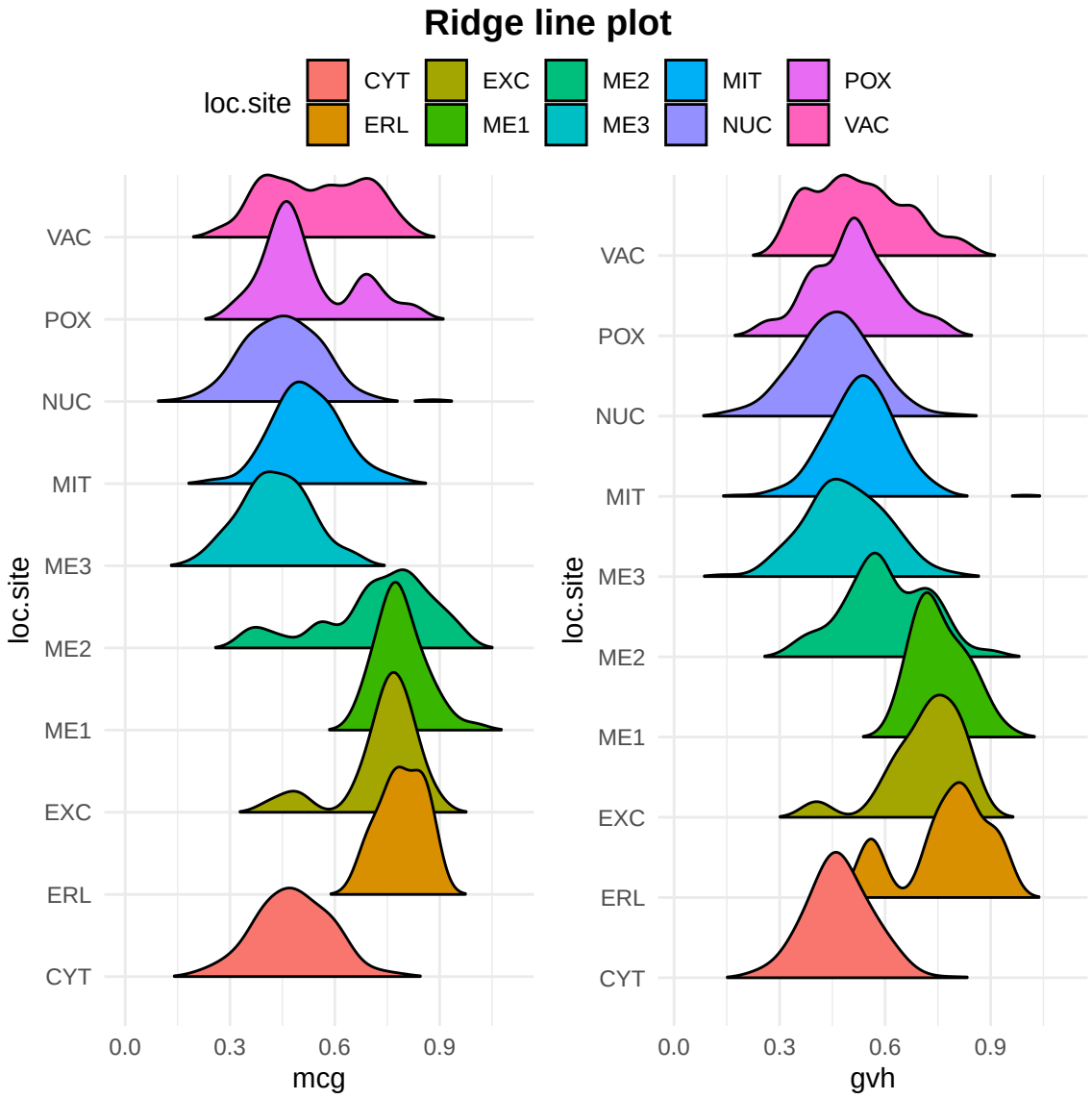

Figure 3: Ridge line plot between mcg and gvh

Now you can really see that the peaks are shifted.

We need to test the data to see if there is significant difference between it. Using a 1-way ANOVA test.

The P-value is $1.2036763 \times 10^{-146} < 0.05$. So there is a significant difference.

Machine Learning

Weka was used to test which algorithm is best to use for the classification of instances. An accurate algorithm is needed that accurately predicts the dataset. It needs a high percentage of correct classifications on the dataset. An algorithm needs to be fast for the best usage. What if there is a dataset of 1 million observations? And it should also not contain too many false negatives or false positives.

Using the experiment of Weka testing has been done on every representatives of all classifier categories. And ZeroR and OneR for a baseline performance. Decision Trees (C4.5, J48, RandomForest), Nearest Neighbor (IBk), SVM (SMO), NaiveBayes (Naïve Bayes) and Linear Logistic (SimpleLogistic). And carried

out classifications 10-fold cross validation. The result were saved in a .csv file.

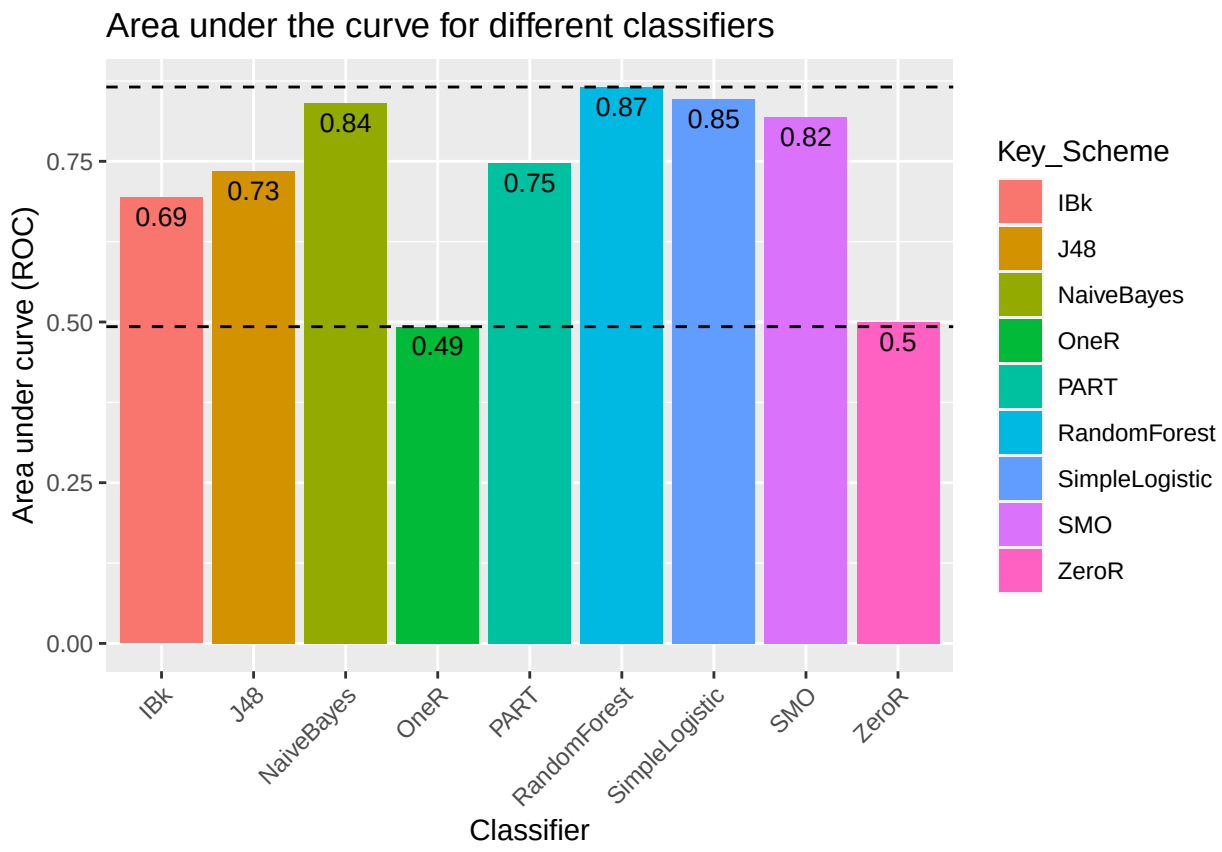

Figure 4: Area under the curve for different classifiers

The area under the curve is a way to measure an effect or phenomenon as a whole. The greater the area under the curve, the better. RandomForest has the largest area under the curve, while OneR has the lowest with 0.49 and ZeroR is a close second with 0.5.

The algorithm also needs te be precise.

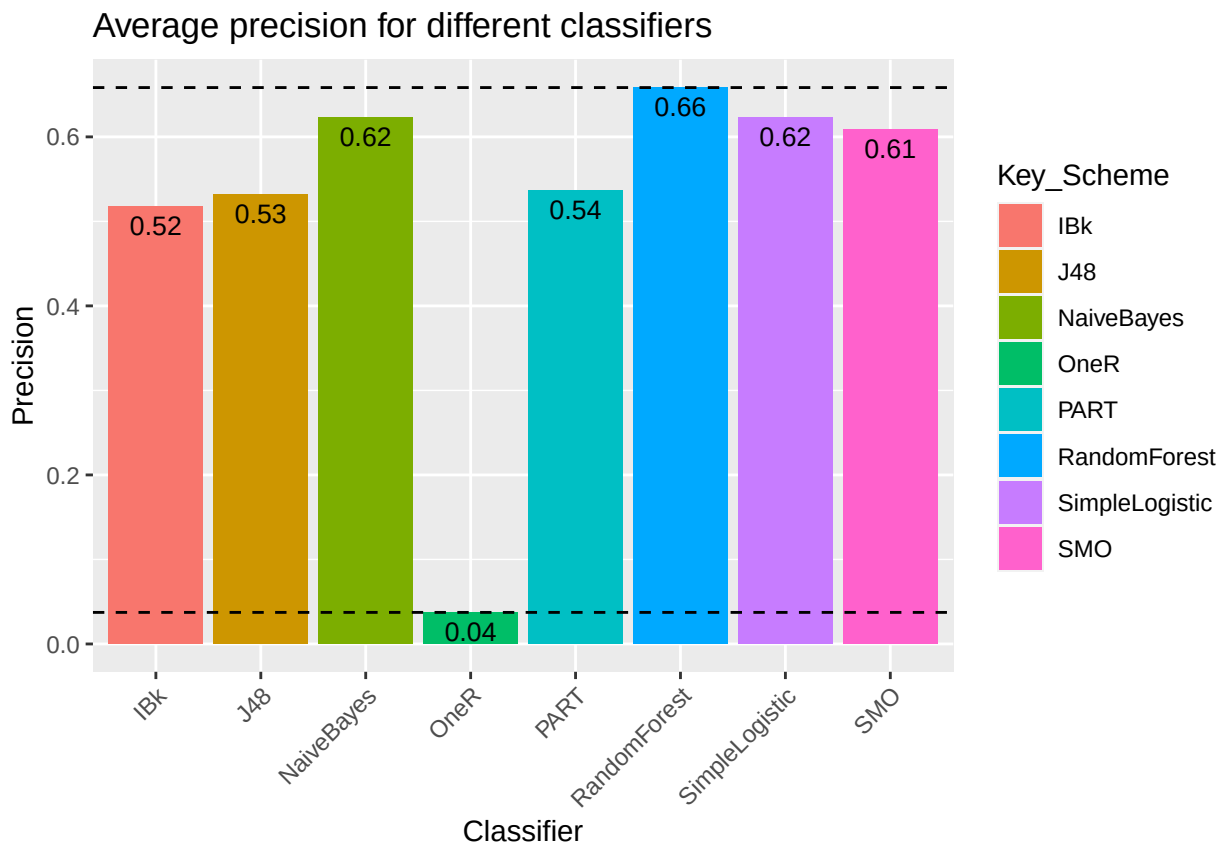

Figure 5: Average precisionfor different classifiers

RandomForest also has the highest average precision, at 0.66. OneR has the lowest, at 0.04. ZeroR is not in this plot because it only consists of NaN.

Several algorithms cannot make a prediction for the classification VAC (0.000), and for some they get an error (?). The IBk algorithm has a very low prediction for VAC. Except for ZeroR, OneR, SMO, and SimpleLogistic, all of the algorithms in the figure above were compared in the Weka experimenter.

Dataset	(1) trees.Ra	(2) rules	(3) trees	(4) lazy.	(5) bayes
data (100)	61.25	54.70 *	56.38 *	52.61 *	57.88 *
(v/ /*)		(0/0/1)	(0/0/1)	(0/0/1)	(0/0/1)

Figure 6: Ranking of chosen algorithms

RandomForest has a 'v' next to their result. This means that the difference in the accuracy for these algorithms compared to all the other algorithms is significant. And RandomForest beat all 4 algorithms.

The classification of VAC seems to be the most concerning of all. It has been removed and tested against not removed. To see if there is an significant difference.

```
Dataset                    (1) trees.Rand
-----------------------------------------
data                      (100)   61.25 |
data-weka.filters.unsuper(100)   62.66 |
-----------------------------------------
(v/ /*)                                 |
```

Figure 7: Removed class difference

There is no significant difference between removing VAC and keeping VAC.

Weka will find the optimal parameters for a classifier. With CVParameterSelection in Weka were the best parameters found.

Attribute	Description	Old.Values	New.Values
P	Size of each bag, as a percentage of the training set size.	100	50
I	Number of iterations.	100	1000
num-slots	Number of execution slots.	1	0
K	Number of attributes to randomly investigate.	0	1
M	Number of attributes to randomly investigate.	1	1
V	Set minimum numeric class variance proportion.	0.001	0.001
S	Seed for random number generator.	1	1

Table 7: Each parameter with the old and new best value if necessary.

This table shows the parameters with their default settings and their new. P, I, num-slots and K were different than the default settings. The build will take more time because it does 10 times more I(terations).

```
Dataset                    (2) trees.Ra | (1) trees
-------------------------------------------------
data                      (100)   62.51 |   63.56
-------------------------------------------------
```

Figure 8: Iterations ranked

Number 1 in the figure above is with 1000 I(terations) and number 2 is 100 I(terations). And there is not a significant difference.

Discussion and Conclusion

The research question is: can machine learning predict protein localization based on their amino acid sequences? And to answer this question properly, a clean dataset is needed to create a model. The results will show what is wrong with the data and the correlations between the different classifications. The results are not completely trustworthy because there are too few of them. Namely, there are only 1484 oberservations. With ten different classifications, this is insufficient for prediction.

Discussion

Nor was the Erl column considered. This indicates the presence of the "HDEL" substring. If this had been looked at, more correlations would have emerged. Which could later become significant. There was also something wrong with the ERL column. As stated on the site, it is a binary/logical type. Either 0 or 1. It is present or it is not present. However, there were only values of 0.5 and 1. All values of 0.5 were changed to 0.

In the case of the Pox column, something strange was noticed. Out of 1484 data points, 15 are not zero. 11 of them have the classification of Pox. This means that if the value is not zero, the classification is highly probable Pox.

Only eight classifiers had been used. From our research, we found that RandomForest is the best, however, perhaps another untested classifier is much better than RandomForest. Only the removal of VAC was considered, even though much better precision and classification would have been achieved if another classifier had been removed.

Conclusion

The aim was to create a clean dataset to be used for our machine learning project. There is a clean dataset with no missing values and the correct data types. The data provides a clear picture of protein classification and attributes. However, not much was available. With only 1484 observations, this is just too little. And there is a very big difference between the number of observations per classification. For ERL there are only 5 observations, and for CYT there are 463. There is too big a difference between these. By improving this dataset over time, the prediction can be perfected even more. These constant improvements are important to find.

1484 yeast proteins were classified into 10 classes with an accuracy of 61%.

Project Proposal

I will be taking the minor in Application Design. And we have been using the program Weka this quarter, and when you open the program, you immediately see how clustered and old-fashioned this program is. My project proposal is to design a modern Weka interface.

References

[1] = J.A. Busker, Theme 9, (2022), GitHub repository, https://github.com/AlfonsoJan/Theme9

[2] = J.A. Busker, Weka Wrapper, (2022), BitBucket repository, https://bitbucket.org/janalfonso/wekawrapper

[3] = P. Horton, K. Nakai, A Probabilistic Classification System for Predicting the Cellular Localization Sites of Proteins, (1996), AAAI, https://www.aaai.org/Papers/ISMB/1996/ISMB96-012.pdf