Introduction into machine learning and analysis of Breast Cancer Proteomes

Theme09 - Introduction to Machine Learning

Mats Slik

344216

BFV3

November 12, 2022

Dave Langers (LADR)

Bart Barnard (BABA)

Introduction into machine learning and analysis of Breast Cancer Proteomes

Theme09 - Introduction to Machine Learning

Mats Slik

344216

Bioinformatics

Institute for Life Science & Technology

Hanze University of Applied Sciences

Dave Langers (LADR)

Bart Barnard (BABA)

November 12, 2022

Abstract

the data were used to assess how the mutations in the DNA are affecting the protein expression landscape in breast cancer. Genes in our DNA are first transcribed into RNA molecules which then are translated into proteins. Changing the information content of DNA has impact on the behavior of the proteome, which is the main functional unit of cells, taking care of cell division, DNA repair, enzymatic reactions and signaling etc. my question is: Are there different ways to categorize breast cancer based on protein expression data, with machine learning being used to classify them without using the pam50 proteins?

Table of Contents

List of Abbreviations

EDA	Exploratory Data Analysis
TCGA	The cancer Genome Atlas Program
CPTAC	Clinical Proteomic Tumor Analysis Consortium
DNA	Deoxyribonucleic Acid
RNA	Ribonucleic Acid

List of Figures

List of Tables

1 Introduction

??

2 Methods

3 Results

my question is: Are there different ways to categorize breast cancer based on protein expression data, with machine learning being used to classify them without using the pam50 proteins?

to answer that question we must first look at the data.

The data set contains published iTRAQ proteome profiling of 77 breast cancer samples generated by the Clinical Proteomic Tumor Analysis Consortium (NCI/NIH). It contains expression values for ~12.000 proteins for each sample, with missing values present when a given protein could not be quantified in a given sample. this data was sampled from 105 originally from the TCGA (The Cancer Genome Atlas Program - NCI), which was further filterd to 77 samples containing high quality protein expression data.

When looking at the dimensions of the data set we can see there are a lot of proteins see table 1

Row.names	Tumor	NP_958782	NP_958785	NP_958786	NP_000436
blcdb9.I.CPTAC	Healthy	-0.1913	-0.1839	-0.186	-0.186
c4155b.C.CPTAC	Healthy	0.567	0.5787	0.5767	0.5767
TCGA-A2-A0CM	T2	0.6834	0.6944	0.6981	0.6871
TCGA-A2-A0D2	T2	0.1075	0.1042	0.1075	0.09751
TCGA-A2-A0EQ	T2	-0.9127	-0.928	-0.928	-0.9318
TCGA-A2-A0EV	T1	0.453	0.4726	0.4726	0.4586

```
## number of rows: 80 number of columns: 9201
```

With this distribution we can clearly see that there are a lot of attributes per instance, normally u want around ten percent of N instances as attributes. in this data set that corresponds with around 7 attributes. This limit our options on the machine learning part. /newline

After this first assessment of the data we started looking at the number of missing values as seen in the figures' fig 1 and 2 below, this is done to ascertain if there are a lot of missing values in the data set en if we neet to filter them out for better result later on in with the machine learning part

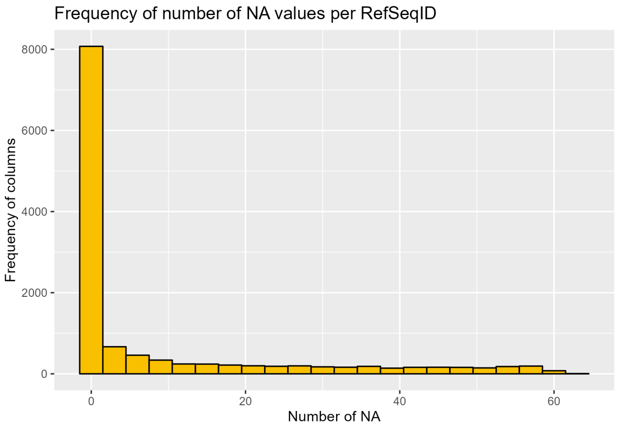

Figure 1: NA count histogram

As we can see in these figures 1 and 2 the distribution is very much to the left where a lot of proteins have one or only two missing values, further more there are still a couple of proteins that have a high number of missing values these are to be filtered out because this can create a false set of results when we are using them in our machine learning algorithm for classifying them on their cancer stage. so to

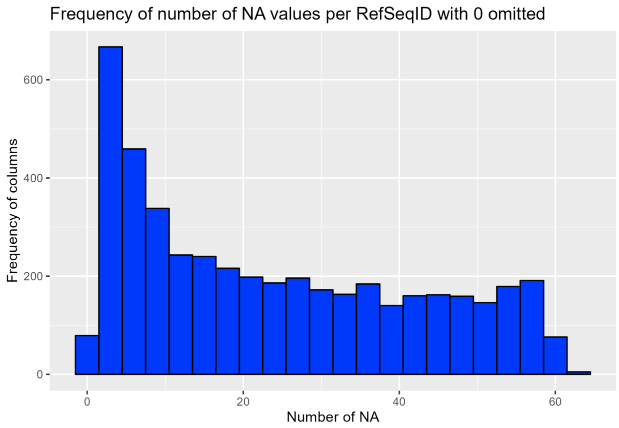

Figure 2: NA count histogram

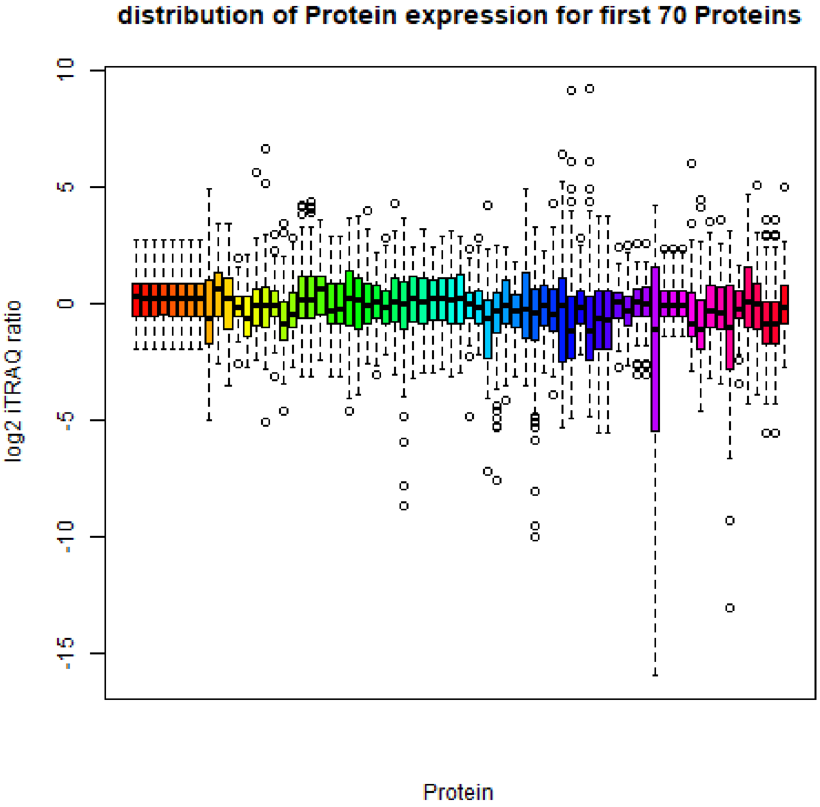

Figure 3: attribute distribution

further visualize the data we took the distribution of a couple of proteins in a multi boxplot as seen in figure 3.

In this figure 3 we can clearly see that for the first 70 protein that most have a distribution of their log2 itraq expression between 5 and -5 but there are some that have higher numbers. To further make sense of all the 12 to 9 thousand proteins in the data we calculated the standard deviation of them see figure 4

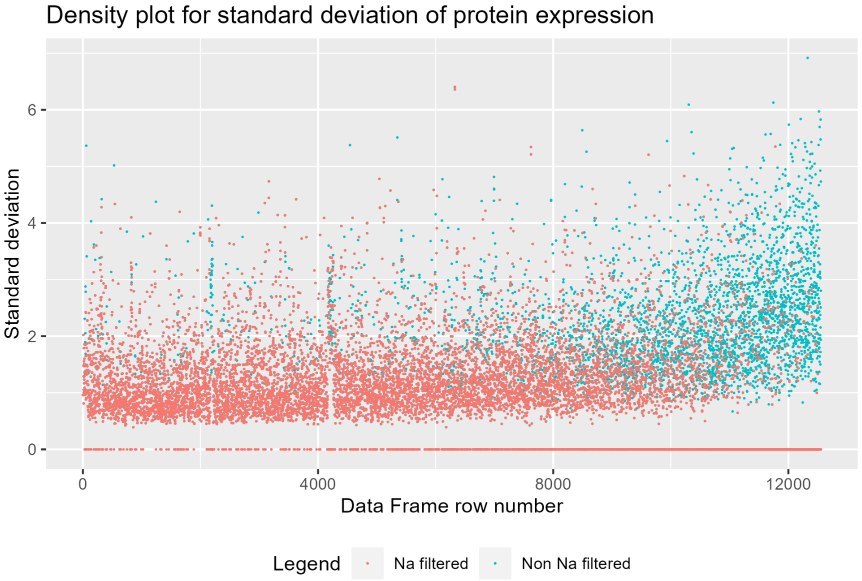

Figure 4: attribute standard deviation, comparison between NA filtered and non filterd data set

In this figure 4 we compared the normal data set and the one filtered that has had protein with more tha 10% of their values missing removed. In it, we can clearly see that a lot of proteins with high deviation are removed from the data. To make a further analyse of these samples

In this figure 5 we can see how the different samples are spread according to there cancer stages. here we can see a clear bias towards T2.

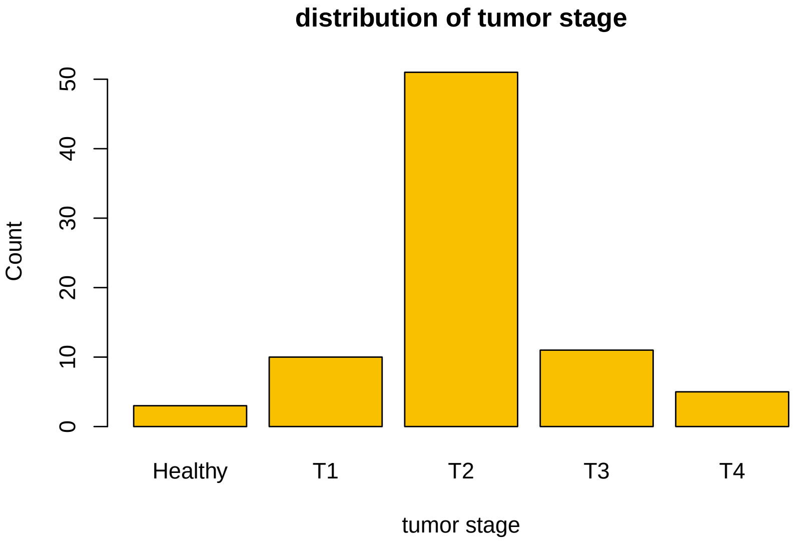

Figure 5: distribution of amount of samples per tumor stage

Table 2: Tabel with the algorithms with default settings used in initial comparison .

(1) rules.OneR '-B 6' -3459427003147861500
(2) trees.RandomTree '-K 0 -M 1.0 -V 0.001 -S 38' -9051119597407395800
(3) trees.RandomForest '-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 38' 1116839470751428740
(4) trees.J48 '-C 0.25 -M 2' -217733168393644448
(5) meta.AttributeSelectedClassifier '-E C̈fsSubsetEval -P 6 -E 6-̈S G̈reedyStepwise -T -1.7976931348623157E308 -N -1 -num-slots 1-̈W trees.J48
(6) meta.AttributeSelectedClassifier '-E C̈fsSubsetEval -P 6 -E 6-̈S B̈estFirst -D 2 -N 5-̈W trees.J48 – -C 0.25 -M 2' -1151805453487947520
(7) meta.AttributeSelectedClassifier '-E C̈fsSubsetEval -P 6 -E 6-̈S B̈estFirst -D 2 -N 5-̈W trees.RandomForest – -P 100 -I 100 -num-slots 1 -K 0
(8) meta.AttributeSelectedClassifier '-E C̈fsSubsetEval -P 6 -E 6-̈S B̈estFirst -D 2 -N 5-̈W trees.RandomTree – -K 0 -M 1.0 -V 0.001 -S 38' -115

machine learning

the results are as seen in table 3

Table 3: Tabel with a T test performed on the percentage each algoritme correctly predicted.

Dataset	(1) rules.On	(2) trees	(3) trees	(4) trees	(5) meta.	(6) meta.	(7) meta.	(8) meta.
R data frame	59.74	50.65	66.23	35.06 *	45.45	42.86 *	62.34	46.75
significance	(v/ /*)	(0/1/0)	(0/1/0)	(0/0/1)	(0/1/0)	(0/0/1)	(0/1/0)	(0/1/0)

as we can see from these results nothing really stands out from the rest, the all perform quit bad especially if u compare it with the OneR and ZeroR. this is due to having a lot of attributes in my date with relatively a low number of instances, so overfitting is a major issue. This forces use to make some selection in atributes. this in turn forces us the meta.AttributeSelectedClassifier to first make a selection of the attributes and further refine these classifiers.

So for further testing we firstly make a good baseline with the zeroR and further make comparisons with mutiple combinations of the meta.AttributeSelectedClassifier and its parameters.

The next set of test where done in the weka explorer gui.

The first of these is a ZeroR and the rest are the results of a few of the best AttributeSelectedClassifier, since most of them are extremely poor performing and not worthy of mentioning the results at all. After showing these results we shall make a conclusion about which algorithm performs the best. All of the following test runs have been made with crossvallidation with the leave one out method to maximise the limited number of instances in the data. ##### ZeroR

Table 4: Tabel with the summary of results from zeroR

Correctly Classified Instances	51	66.2338
Incorrectly Classified Instances	26	33.7662
Kappa statistic	0	x
Mean absolute error	0.2665	x
Root mean squared error	0.3613	x
Relative absolute error	100	x
Root relative squared error	100	x
Total Number of Instances	77	x

Table 5: Confusion matrix

a	b	c	d	<– classified as
0	10	0	0	a = T1
0	51	0	0	b = T2
0	11	0	0	c = T3
0	5	0	0	d = T4

As we can see from the results of ZeroR that even classifing everything as T2 scores 66% good, thus using that as a base of evaluationg the classifiers used is not very reliable and we shall take the confusing matrix and ROC curves more as a indication for a good algoritme to use for oure data.

This is the second algorithm used, and this is using attribute selection with sub set evaluation based on the best first method. as a cost matrix I assigned every wrongly classified instance as class T2 extra heavy since that class is overrepresented, and further weight that every clas that is wrongly classified a little heavier

AttributeSelectedClassifier with cost sensitive J48

Relation: R_data_frame
Instances: 77
Attributes: 9200
Test mode: 77-fold cross-validation
Evaluation cost matrix:

```
0  5  2  2
1  0  1  1
1  5  0  2
2  5  2  0
```

=== Attribute Selection on all input data ===

Search Method:
Best first.
Start set: no attributes
Search direction: forward
Stale search after 5 node expansions
Total number of subsets evaluated: 211334
Merit of the best subset found: 0.601

Attribute Subset Evaluator (supervised, Class (nominal): 9200 data.class):
CFS Subset Evaluator
Including locally predictive attributes

Selected attributes: 338,905,1188,1230,1555,2172,2277,2821,3196,3333,3719,3932,5844,6802,7234,7490,7959,8149,8538
: 19

Table 6: Tabel with the summary of results from zeroR

Correctly Classified Instances	22	28.5
Incorrectly Classified Instances	55	71.4
Kappa statistic	-0.1735	x
Total Cost	135	
Average Cost	1.7532	
Mean absolute error	0.3473	x
Root mean squared error	0.5733	x
Relative absolute error	129.1465	x
Root relative squared error	156.9896	x
Total Number of Instances	77	x

Table 7: Confusion matrix

a	b	c	d	<– classified as
1	7	2	0	a = T1
8	19	19	5	b = T2
2	8	1	0	c = T3
0	4	0	1	d = T4

in the above seen confusion matrix we are looking for a nicely made line from the top left to the bottom right, but we can clearly see that that is not the case. So this algorithm doesn't have a lot of merit for further exploration

AttributeSelectedClassifier HoeffdingTree

Relation: R_data_frame
Instances: 77
Attributes: 9200
Test mode: 77-fold cross-validation
Evaluation cost matrix:

0	5	2	2
1	0	1	1
1	5	0	2
2	5	2	0

=== Attribute Selection on all input data ===

Search Method:
Best first.
Start set: no attributes
Search direction: bi-directional
Stale search after 5 node expansions
Total number of subsets evaluated: 248373
Merit of best subset found: 0.604

Attribute Subset Evaluator (supervised, Class (nominal): 9200 data.class):
CFS Subset Evaluator
Including locally predictive attributes

Selected attributes: 338,1188,1230,1555,2172,2277,2844,3196,3333,3719,3932,5844,6802,7234,7490,7959,8149,8538
: 18

Table 8: Tabel with the summary of results from HoeffdingTree

Correctly Classified Instances	45	58.4
Incorrectly Classified Instances	32	41.5
Kappa statistic	-0.0584	x
Mean absolute error	0.2337	x
Root mean squared error	0.4332	x
Relative absolute error	86.9241	x
Root relative squared error	118.6126	x
Total Number of Instances	77	x

Table 9: labelHoeffdingTree confusionmatrixConfusion matrix

a	b	c	d	<– classified as
0	9	1	0	a = T1
5	45	1	0	b = T2
1	10	0	0	c = T3
0	5	0	0	d = T4

in the above seen confusion matrix we are looking for a nicely made line from the top left to the bottom right, but we can clearly see that that is not the case, and it classifies them mostly as t1 and t2. So this algorithm doesn't have a lot of merit for further exploration, and is not on its own not good

AttributeSelectedClassifier Ranker RandomTree

Relation: R_data_frame
Instances: 77
Attributes: 9200
Test mode: 77-fold cross-validation
Evaluation cost matrix:

0	5	2	2
1	0	1	1
1	5	0	2
2	5	2	0

Selected attributes: 4526,6015,1443,1312,8523,2172,1230,5384,2277,4206,8329,1188 : 12

Table 10: Tabel with the summary of results from Randomtree

Correctly Classified Instances	42	54.5
Incorrectly Classified Instances	35	45.5
Kappa statistic	0.1126	x
Mean absolute error	0.226	x
Root mean squared error	0.4726	x
Relative absolute error	84.0	x
Root relative squared error	129.4	x
Total Number of Instances	77	x

Table 11: labelRandomTree confusionmatrixConfusion matrix

a	b	c	d	<– classified as
3	4	3	0	a = T1
3	38	7	3	b = T2
5	6	0	0	c = T3
0	4	0	1	d = T4

confusion matrix

in the above seen confusion matrix we are looking for a nicely made line from the top left to the bottom right, but we can clearly see that that is not the case, and it classifies them mostly wrong as t2. So this algorithm doesn't have a lot of merit for further exploration

AttributeSelectedClassifier greedystepwise with OneR
Relation: R_data_frame
Instances: 77
Attributes: 9200
Test mode: 77-fold cross-validation
Evaluation cost matrix:

=== Attribute Selection on all input data ===

Search Method:
Greedy Stepwise (forwards).
Start set: no attributes
Merit of best subset found: 0.601

Attribute Subset Evaluator (supervised, Class (nominal): 9200 data.class):
CFS Subset Evaluator
Including locally predictive attributes

Selected attributes: 338,905,1188,1230,1555,2172,2277,2821,3196,3333,3719,3932,5844,6802,7234,7490,7959,8149,8539
: 19

Table 12: Tabel with the summary of results from OneR

Correctly Classified Instances	45	58.44
Incorrectly Classified Instances	32	41.55
Kappa statistic	-0.0788	x
Total Cost	133	
Average Cost	1.7273	
Mean absolute error	0.2078	x
Root mean squared error	0.4558	x
Relative absolute error	77.2714	x
Root relative squared error	124.8216	x
Total Number of Instances	77	x

Table 13: labelgreedystepwise with OneR confusionmatrixConfusion matrix

a	b	c	d	<– classified as
0	9	1	0	a = T1
2	45	4	0	b = T2
0	11	0	0	c = T3
0	5	0	0	d = T4

in the above seen confusion matrix we are looking for a nicely made line from the top left to the bottom right, but we can clearly see that that is not the case, we can see that is mostly classify them as T2. So this algorithm doesn't have a lot of merit for further exploration

these are but the best of the multiple different settings that where tried, but all results where the same as these presented in the chapters here above. one if not the first things that stands out for every one of these results is that none of these results scored an accuracy of correctly predicting the class of the data better than ZeroR with its 66.2%. One got close but that was a RandomTree model without attributeselection, so it is questionable how accurate its truly is. But accuracy of correctly prediction the class is not everything, so we must also look at the confusion matrix's that where produced. As can be seen from the confusion matrix's (tables) there are two trends visible that firstly there is a huge bias towards t2 Thus, most of the models that where teste where with the meta learning AttributeSelectedClassifier , and yielded not much as explained above. The main thing that can be said of all the methods and algorithems used is they performed all bad or equally bad as just picking a random class. One more thing is that a significant portion the models that where tried with the AttributeSelectedClassifier is that they almost all chose these attributes : NP_008832 ,NP_056289,NP_001349,NP_055719,NP_001150,NP_079093,NP_000959,NP_004893,NP_004887,NP_065901

4 Discussion and Conclusion

In the results' section we can see from the figures 1 and two that although the data set was supplied with the label as high quality there are still proteins in the data with more tha 10% of there expression values missing, this combined with the need for using the expressing data with the clinical categorical data, the sample names needed to be changed to be compared. all this wasn't something to be expecting of high quality data. also in figure 5 it is clearly visible that the categorisation of the tumor stage there are a lot of T2 stages in the samples than any other. furthermore the sheer amount of proteins recorded in this data is very useful for my purpose of trying to use another classification as the PAM50 protein list

5 References

Mertins, Philipp, D R Mani, Kelly Ruggles, Michael Gillette, Karl Clauser, Pei Wang, Xianlong Wang, et al. 2016. "Proteogenomics Connects Somatic Mutations to Signaling in Breast Cancer." *Nature* 534 (May). https://doi.org/10.1038/nature18003.

6 Appendices