Statistical learning Project

[Classification of amino acids interaction type]

[Stefano Minto, Sina Rasouli, Fabiana Rapicavoli, Canabdullah Camuz]

1 Introduction

Proteins are biological entities that are composed of amino acids/residues. There are 20 amino acids, differing from each other with different properties, such as charge, size, polarity etc.

A protein must be folded to a specific structure in the 3D space, via building chemical bonds among its amino acids.
The structure tends to have the minimum energy in the folding process and that particular conformation is called native.

This process is driven by chemical and peptide interaction bonds between the amino acids, which could be:

- H-bond (HBOND)
- Van der Waals interactions (VDW)
- Disulfide bridges (SBOND)
- Salt bridges (IONIC)
- π-π stacking (PIPISTACK)
- π-cation (PICATION)

The main difference between those interactions is the strength of the bond between two chemical elements in a given protein.

The aim of this project is to try to predict the contact types, given different features of two amino acids that are in contact.

To achieve the purpose, a classification model was needed to calculate the probabilities of different contact types, which the residue pair (amino acid pair) might have.

In this Project, different models were employed to perform the task of classification, the results were evaluated with different metric scores.

2 Obtaining Data

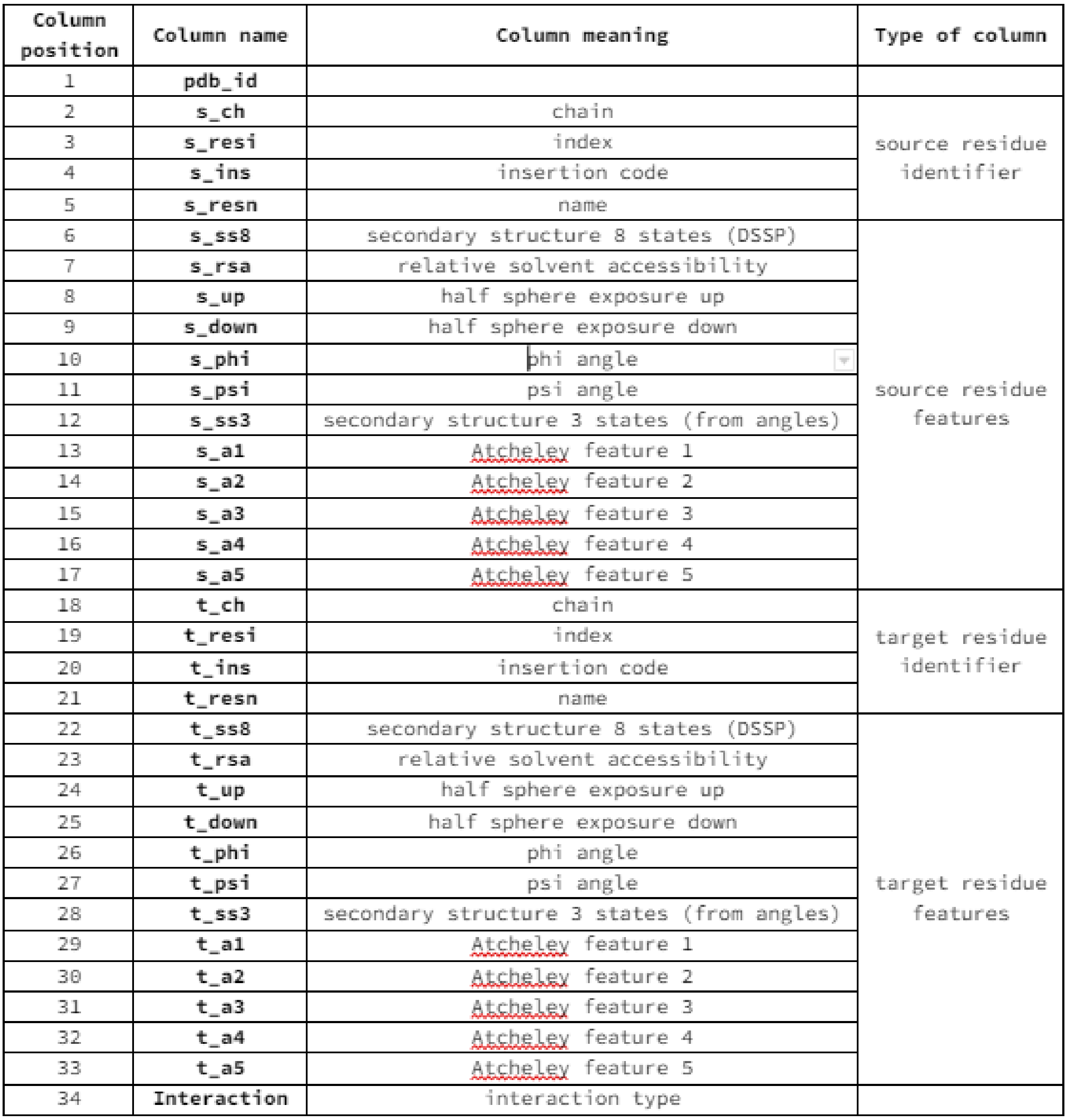
The data come from a database of 1807 tsv.file, that file are obtained by processing .pdb file, that describe the 3D structure of a protein.

PDB (Protein Data Bank) is a US based data center, containing an archive of 3D structure data for large biological molecules (proteins, DNA, and RNA), essential for research and education.

3 Explore the data

Every .tsv file contains all the list of the pairs amino acids in contact in a given protein, so every row corresponds to the interaction between two amino acids (source and target) and every column correspond to a feature.

There are 34 possible features:



Column position	Column name	Column meaning	Type of column
1	pdb_id		
2	s_ch	chain	source residue identifier
3	s_resi	index	
4	s_ins	insertion code	
5	s_resn	name	
6	s_ss8	secondary structure 8 states (DSSP)	source residue features
7	s_rsa	relative solvent accessibility	
8	s_up	half sphere exposure up	
9	s_down	half sphere exposure down	
10	s_phi	phi angle	
11	s_psi	psi angle	
12	s_ss3	secondary structure 3 states (from angles)	
13	s_a1	Atcheley feature 1	
14	s_a2	Atcheley feature 2	
15	s_a3	Atcheley feature 3	
16	s_a4	Atcheley feature 4	
17	s_a5	Atcheley feature 5	
18	t_ch	chain	target residue identifier
19	t_resi	index	
20	t_ins	insertion code	
21	t_resn	name	
22	t_ss8	secondary structure 8 states (DSSP)	target residue features
23	t_rsa	relative solvent accessibility	
24	t_up	half sphere exposure up	
25	t_down	half sphere exposure down	
26	t_phi	phi angle	
27	t_psi	psi angle	
28	t_ss3	secondary structure 3 states (from angles)	
29	t_a1	Atcheley feature 1	
30	t_a2	Atcheley feature 2	
31	t_a3	Atcheley feature 3	
32	t_a4	Atcheley feature 4	
33	t_a5	Atcheley feature 5	
34	Interaction	interaction type	

Different features

As we said, every row repeats the same piece of information twice: once for both the amino acids/residues in contact (except for the features for the id of the protein and the type of interaction, which are the same for both the amino acids in contact).

An additional division of the features is that every amino acid has two types of information: one part for its identity and another part for the actual features.

The features, related to the amino acid's identity are:

- pdb_id: the id of the protein used for classify different protein in the pdb database;
- s_ch and t_ch: the chain of the protein;
- s_resi and t_resi: index in the protein of the amino acid;
- s_ins and t_ins: insertion code, useful for comparing the same protein in different species;
- s_resn and t_resn: the name of the protein, which the respective amino acid belong.

This type of features are all categorical and we decided to eliminate them from the dataset, since they didn't give us any information, useful to understand and distinguish the bond type between the amino acids.
The informational features are:

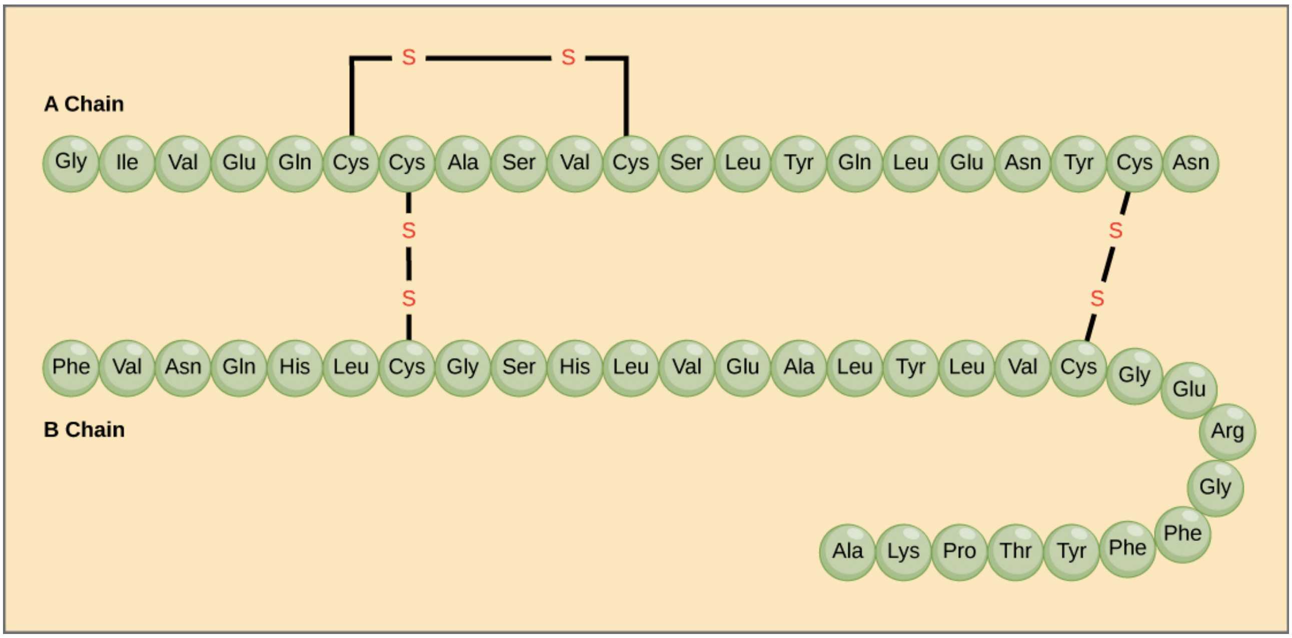
- s_ss8 and t_ss8: the only categorical features, which represent the type of the secondary structure, to which the amino acid belongs (the shape);
- s_rsa t_rsa: relativity solvent accessibility for determining their folding and stability;
- s_up, s_down, t_up and t_down: Half sphere exposure, like every solvent exposure, measures how buried amino acid residues are in a protein, both in the upper and in the downer part of the amino acids;
- s_phi, s_psi, t_phi and t_psi: The alpha carbon (C) in the center of each amino acid is held in the main chain by two rotatable bonds. The dihedral (torsion) angles of these bonds are called Phi and Psi, since these bonds aren't free to rotate because of the electronic and physical constraints;
- s_ss3 and t_ss3: the secondary structure, basically the same information of before (s_ss8 and t_ss8), but with three states, instead of eight, and in numerical form, instead of nominal.
  For this reason, we decided to delete from the dataset the two features s_ss8 and t_ss8, for redundancy;
- s_a1-5 and t_a1-5: the so-called atchley features, which describe different amino acid characteristic, in particular: polarity, secondary structure, molecular volume, codon diversity and electrostatic charge.

The 34th and last feature 'Interaction' is the response one and it indicates, obviously, the type of bond between the two residues.

Just for curiosity, after importing all the dataset merged in a single .tsv file with a python script, we studied the distribution of the chains between

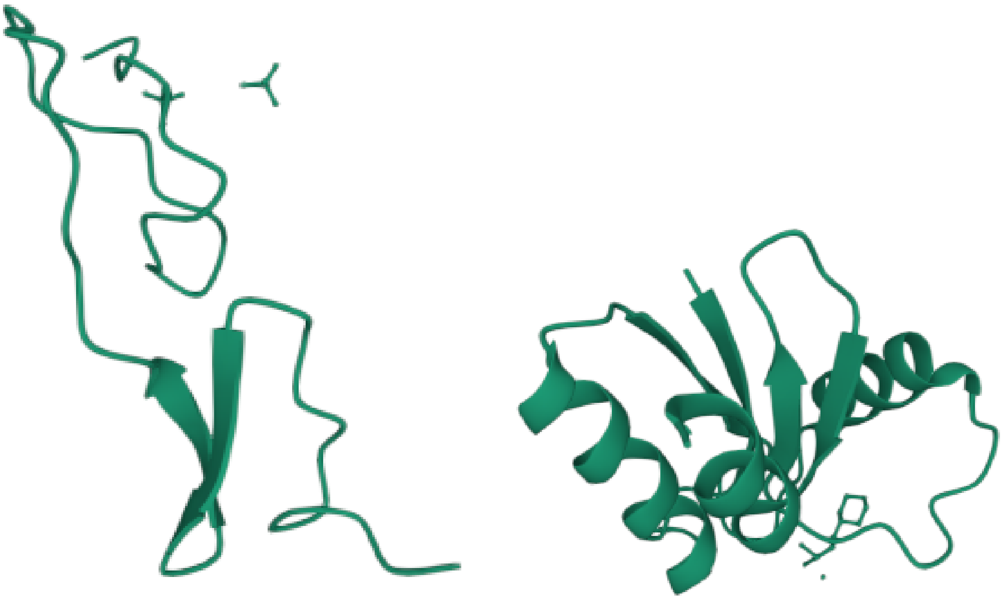
the different proteins.
In theory, a protein can be structured by more than one chain. In practise, our dataset has only protein with one chain.

Proteins are actually chains of amino acids and they represents the first structure of a protein.
Each protein chain is a linear polymer, having two distinct ends (N and C). The “sequence” of a protein chain is given as the list of amino acids in its chain, from N to C.
They all have a backbone consisting of three atoms in a row: nitrogen and two atoms of carbon, repeating as many times as needed.
Each chain has its own set of amino acids, assembled in a particular order. For instance, the sequence of the A chain starts with glycine at the N-terminus and ends with asparagine at the C-terminus.

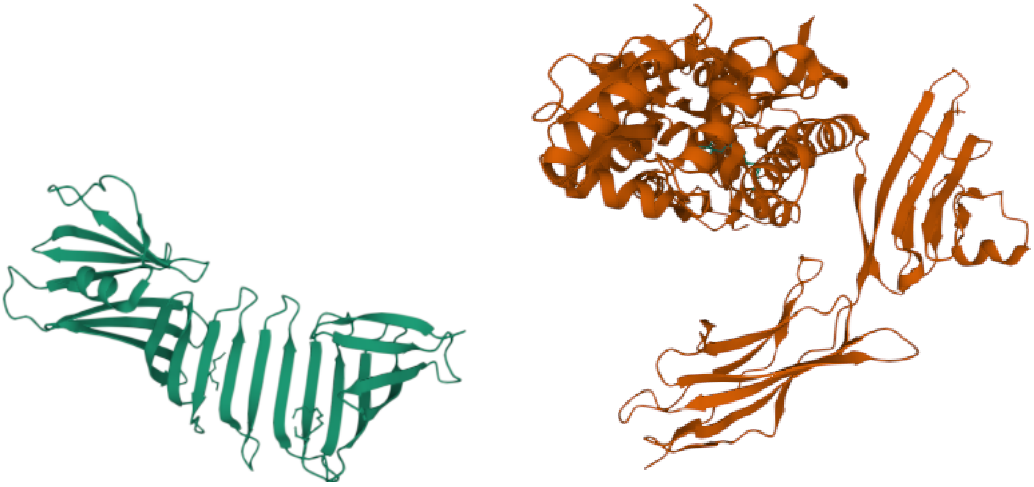


Insulin molecules of a cow, composed of chains A and B

That doesn’t mean that every chain type is equal to each other, simply the structure is very similar.
For example, we took four sample of proteins, two with chain A above and one with chain N and one with chain O below.



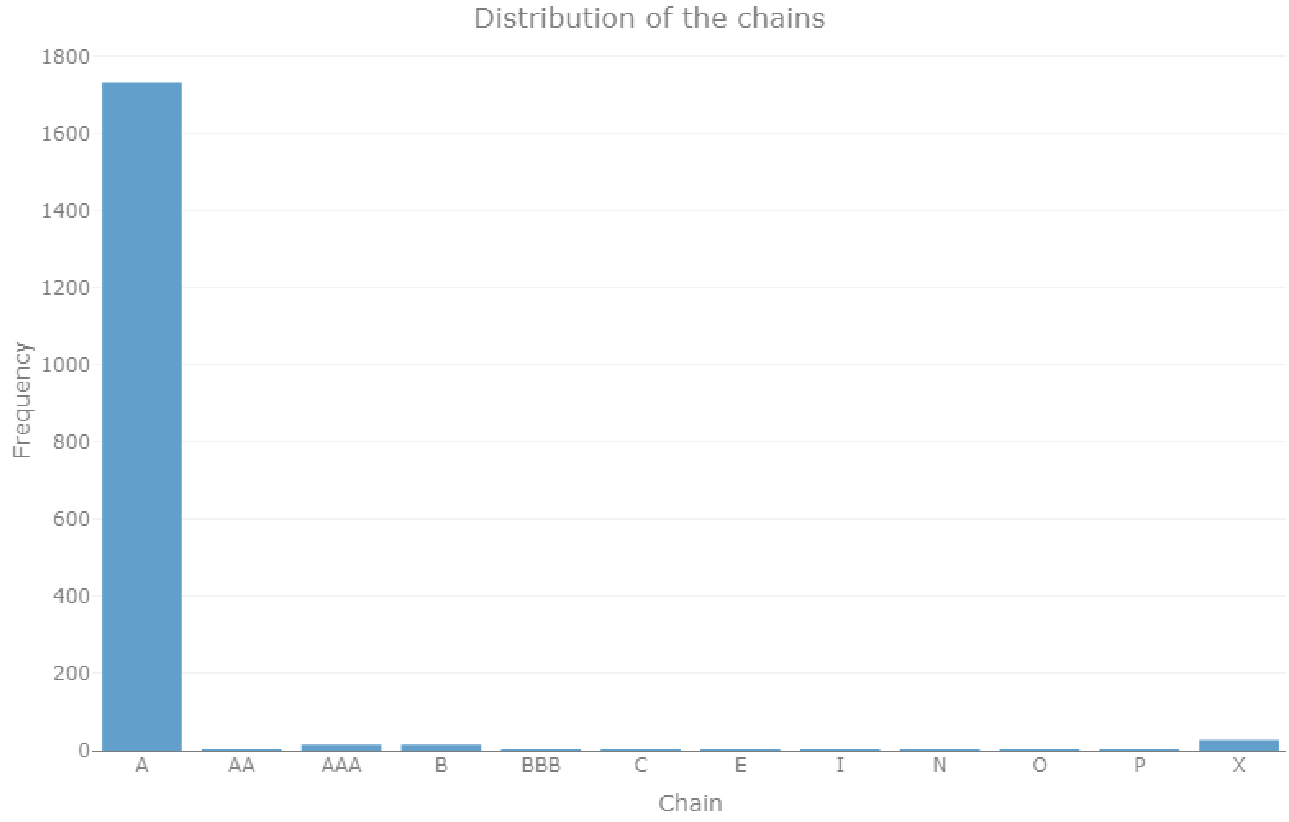
Two samples of chain A: THIOREDOXIN and HIRUSTASIN

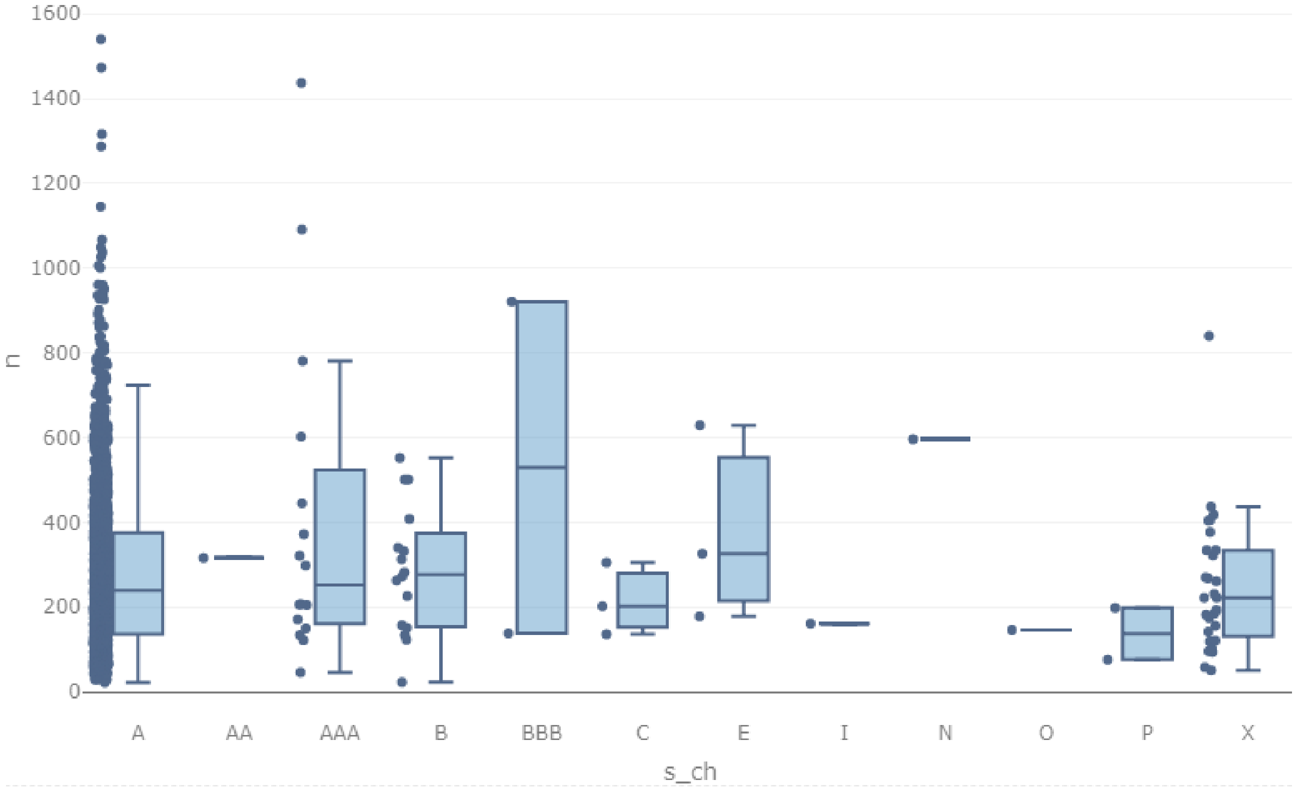


Two samples of chain O and N: a crystal Structure of Human Receptor for Advanced Glycation Endproducts and an tomic-resolution crystal structure of Borrelia burgdorferi

As we can see, the protein structures above are more similar from each other than the other ones below.

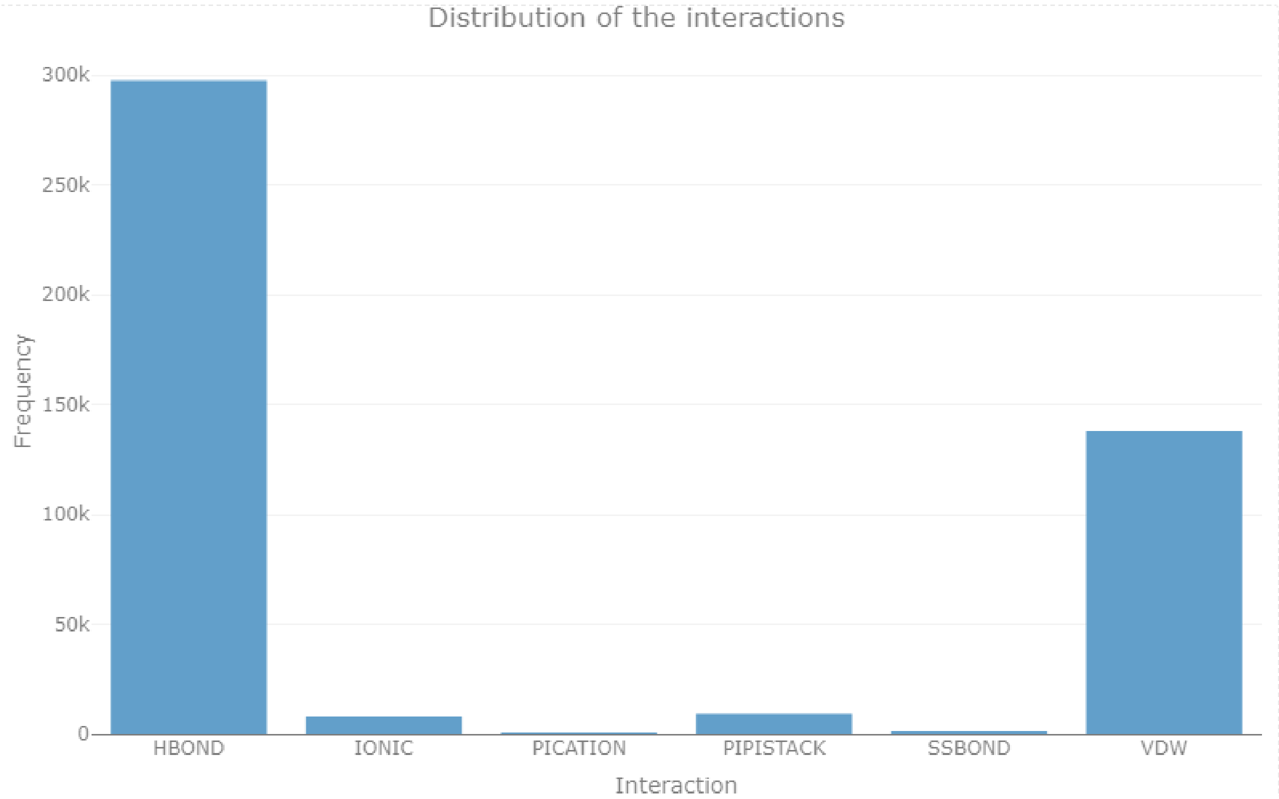
In our dataset, the distribution of the chains is extremely unequal and it shows a pick of presence of chain A, which is the most common chain in general.





Distribution of the chains

Speaking of the distribution of the interaction, our y feature, we can notice the same event:



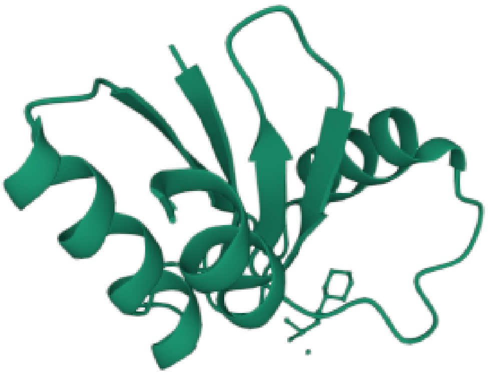
Distribution of the interactions

The most frequent classes are hydrogen bond and Van der Waals, followed by very few sample of the other classes.

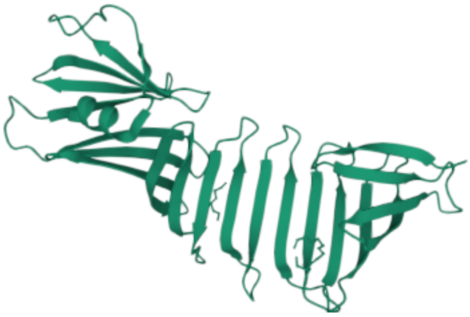
We tried to better understand the reason of this phenomena and we discovered that the most common secondary structure in a protein are beta sheet and alpha-helix, which require many of these two type of bonds.

Let’s take the same four samples of before and their respective number and type of interactions:

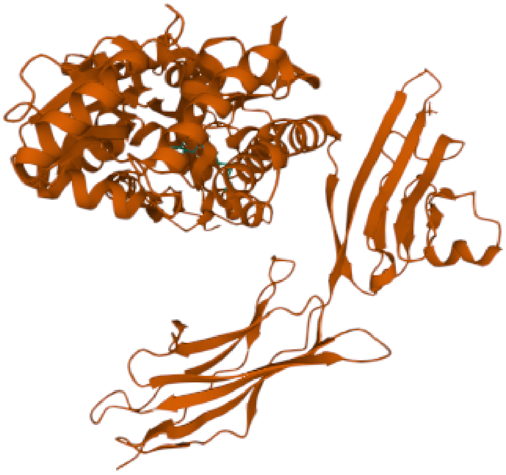
H-bond	16
π-π Stack	0
π-Cation	0
Ionic	1
Disulphide	5
van der Waals	37



H-bond	77
π-π Stack	6
π-Cation	1
Ionic	1
Disulphide	1
van der Waals	54



H-bond	162
π-π Stack	0
π-Cation	0
Ionic	4
Disulphide	0
van der Waals	128



H-bond	470
π-π Stack	18
π-Cation	1
Ionic	7
Disulphide	2
van der Waals	413

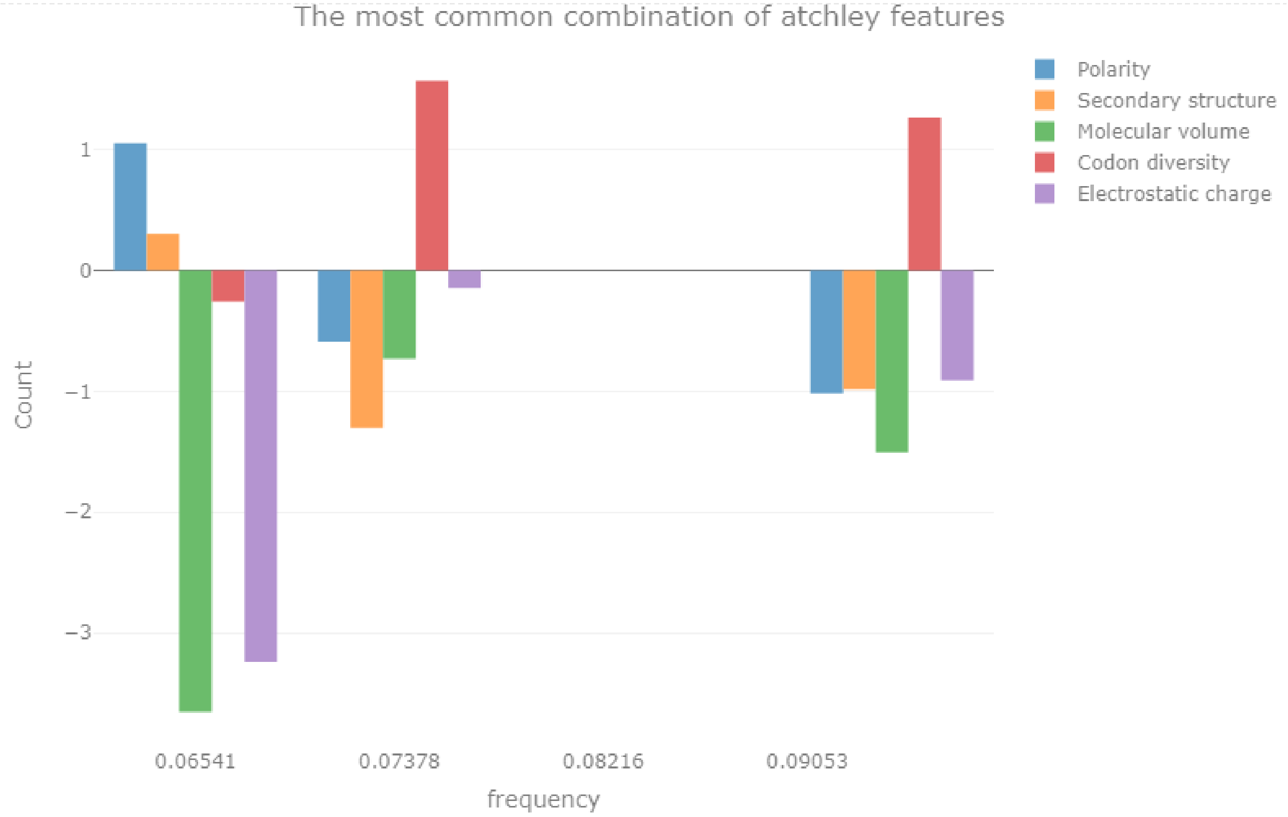
The more articulated and composed is the protein, the more HBOND and VDW bonds are present.

With these information, we tried to clean our dataset, in order to make the analysis more efficient and make the model only focus on the most important and informational features.
The data-cleaning process was needed also because, otherwise, our dataset would have been unnecessary too big, with 735.510 observations and, of course, 34 features.

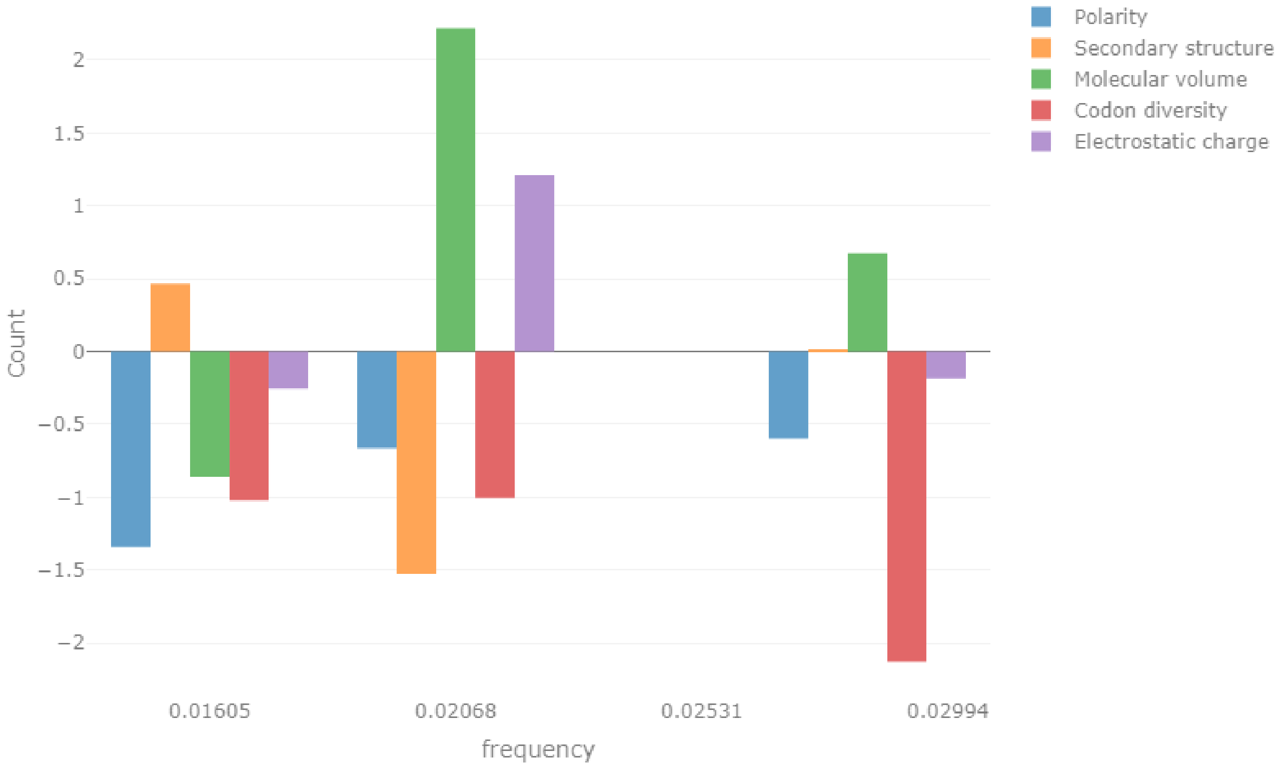
We already discussed about our decision of removing the 12 nominal features, but it wasn't enough.
In order to reduce the complexity of the problem, we eliminated the rows that contained a blank term in interaction feature and also the rows which contained at least one 'NaN' value for any features.

After that operations we obtained a data set with 454.193 rows and 21 columns.
As last curiosity, we studied the atchley features, since we thought that they were the most relevant ones.
In particular, atchley features give us the information about polarity, secondary structure, molecular volume, codon diversity and electrostatic charge.

Between our 454.193 samples, we noticed that there are just twenty combination of atchley features.
Below there are the three most and less frequent atchley features combination:



The most common distribution of atchley features



The less common distribution of atchley features

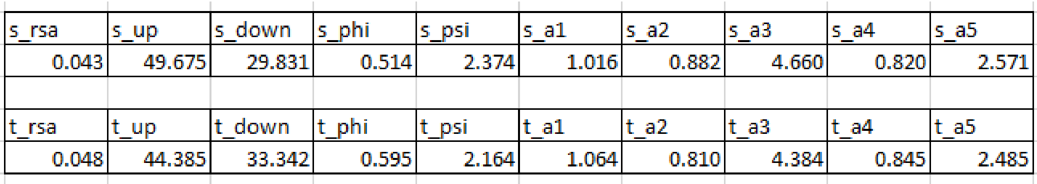
We can notice that the two most common combination of atchley features have all negative values, but the codon diversity one.
The third most common combination has, in particular, an extremely negative electrostatic charge and molecular volume values.

For the less frequent combination have negative polarity, codon diversity and electrostatic change values in common and a positive molecular value for the two less common ones.

4 Model Data

After the pre-data preparation, we started with some analysis for features selection for our model.

We checked the variance of the features in the table below 4.

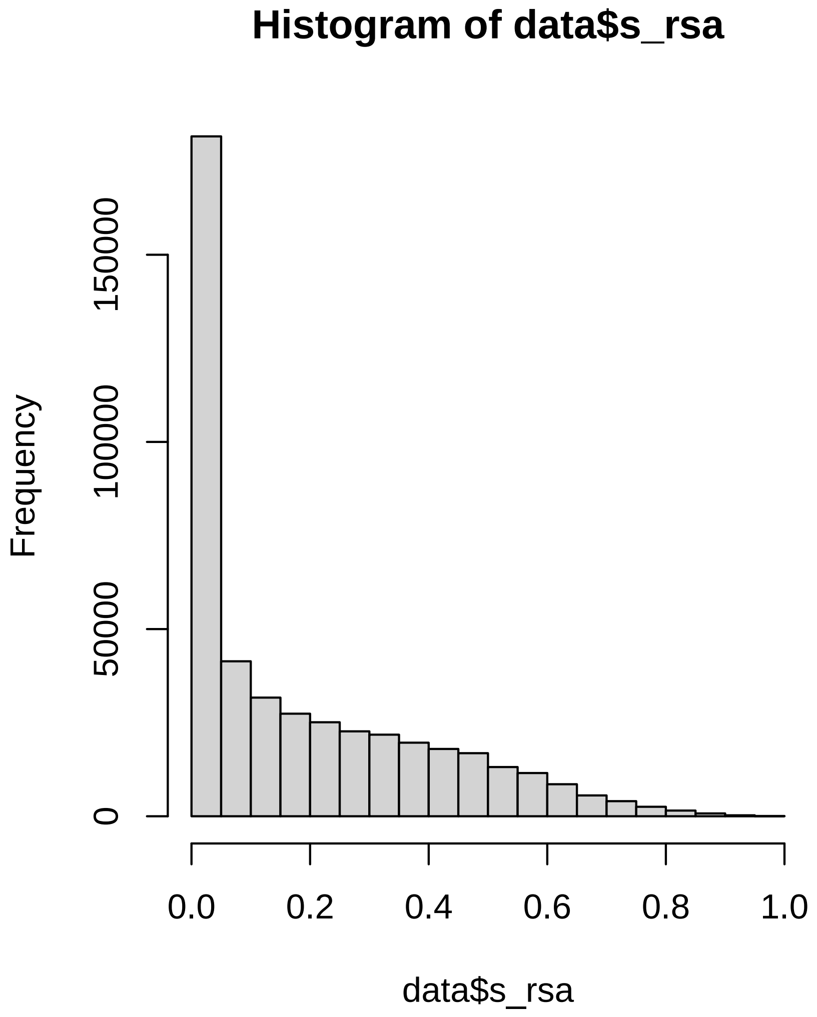


s_rsa	s_up	s_down	s_phi	s_psi	s_a1	s_a2	s_a3	s_a4	s_a5
0.043	49.675	29.831	0.514	2.374	1.016	0.882	4.660	0.820	2.571
t_rsa	t_up	t_down	t_phi	t_psi	t_a1	t_a2	t_a3	t_a4	t_a5
0.048	44.385	33.342	0.595	2.164	1.064	0.810	4.384	0.845	2.485

Variances

Self exposure up and down have the highest variance because these terms are very sensitive to the structure variability of the protein, i.e in some proteins an amino acid can be easily more buried than in others for the fact that protein can have very different conformation from each other.

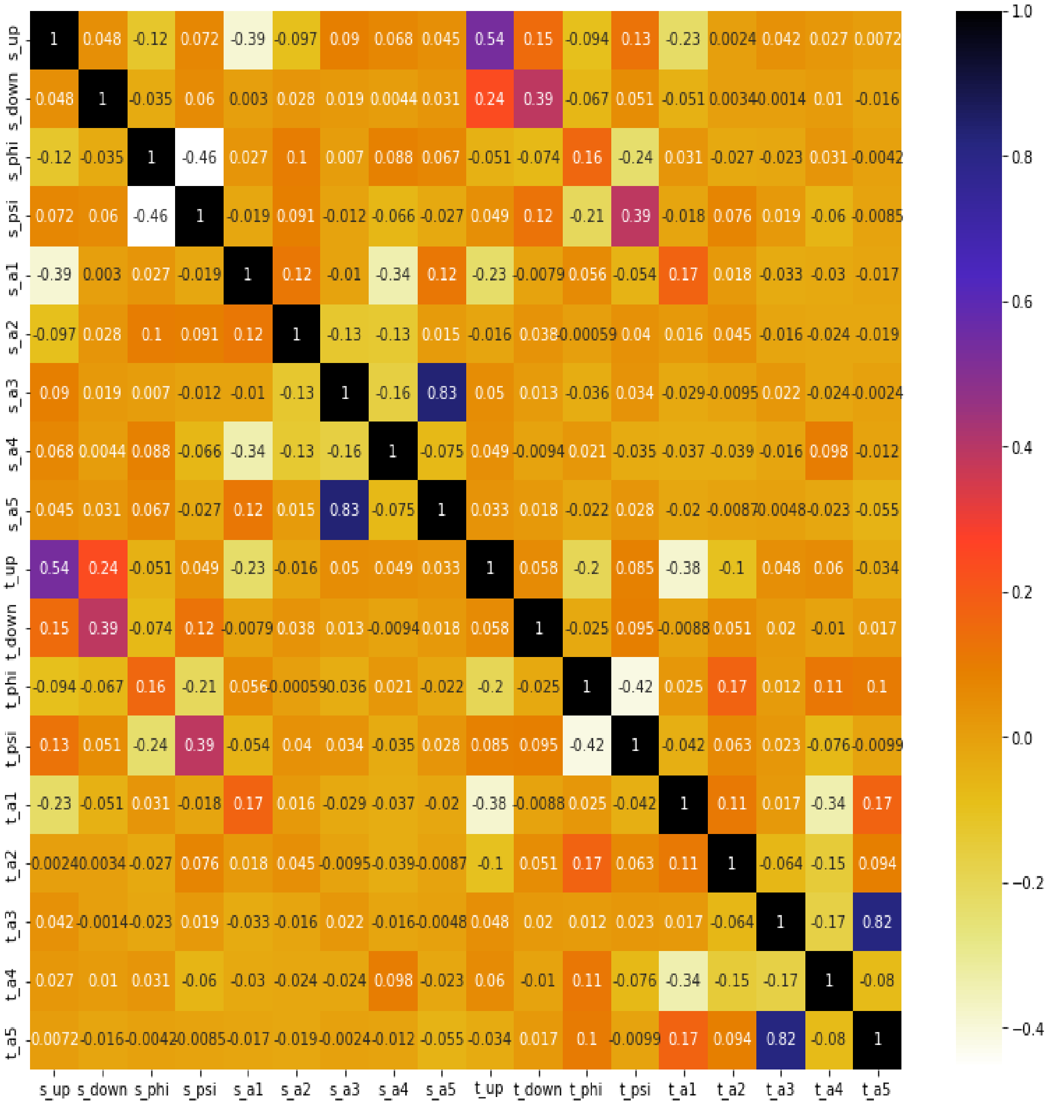
Another thing that we can notice is that Relative solvent accessibility has a very low variance. That means that feature has very low variability, as we can see from its distribution:



s-rsa distr

This result suggested us to eliminate that feature because a low variance predictor means that all values are quite similar to each others and they aren't useful as predictor.

Another thing that we can analyse is the correlation between features. We can address that, showing a correlation matrix, obtained with a python script:



	s_up	s_down	s_phi	s_psi	s_a1	s_a2	s_a3	s_a4	s_a5	t_up	t_down	t_phi	t_psi	t_a1	t_a2	t_a3	t_a4	t_a5
s_up	1	0.048	-0.12	0.072	-0.39	-0.097	0.09	0.068	0.045	0.54	0.15	-0.094	0.13	-0.23	0.0024	0.042	0.027	0.0072
s_down	0.048	1	-0.035	0.06	0.003	0.028	0.019	0.0044	0.031	0.24	0.39	-0.067	0.051	-0.051	0.0034	-0.0014	0.01	-0.016
s_phi	-0.12	-0.035	1	-0.46	0.027	0.1	0.007	0.088	0.067	-0.051	-0.074	0.16	-0.24	0.031	-0.027	-0.023	0.031	-0.0042
s_psi	0.072	0.06	-0.46	1	-0.019	0.091	-0.012	-0.066	-0.027	0.049	0.12	-0.21	0.39	-0.018	0.076	0.019	-0.06	-0.0085
s_a1	-0.39	0.003	0.027	-0.019	1	0.12	-0.01	-0.34	0.12	-0.23	-0.0079	0.056	-0.054	0.17	0.018	-0.033	-0.03	-0.017
s_a2	-0.097	0.028	0.1	0.091	0.12	1	-0.13	-0.13	0.015	-0.016	0.038	-0.00059	0.04	0.016	0.045	-0.016	-0.024	-0.019
s_a3	0.09	0.019	0.007	-0.012	-0.01	-0.13	1	-0.16	0.83	0.05	0.013	-0.036	0.034	-0.029	-0.0095	0.022	-0.024	-0.0024
s_a4	0.068	0.0044	0.088	-0.066	-0.34	-0.13	-0.16	1	-0.075	0.049	-0.0094	0.021	-0.035	-0.037	-0.039	-0.016	0.098	-0.012
s_a5	0.045	0.031	0.067	-0.027	0.12	0.015	0.83	-0.075	1	0.033	0.018	-0.022	0.028	-0.02	-0.0087	-0.0048	-0.023	-0.055
t_up	0.54	0.24	-0.051	0.049	-0.23	-0.016	0.05	0.049	0.033	1	0.058	-0.2	0.085	-0.38	-0.1	0.048	0.06	-0.034
t_down	0.15	0.39	-0.074	0.12	-0.0079	0.038	0.013	-0.0094	0.018	0.058	1	-0.025	0.095	-0.0088	0.051	0.02	-0.01	0.017
t_phi	-0.094	-0.067	0.16	-0.21	0.056	-0.00059	-0.036	0.021	-0.022	-0.2	-0.025	1	-0.42	0.025	0.17	0.012	0.11	0.1
t_psi	0.13	0.051	-0.24	0.39	-0.054	0.04	0.034	-0.035	0.028	0.085	0.095	-0.42	1	-0.042	0.063	0.023	-0.076	-0.0099
t_a1	-0.23	-0.051	0.031	-0.018	0.17	0.016	-0.029	-0.037	-0.02	-0.38	-0.0088	0.025	-0.042	1	0.11	0.017	-0.34	0.17
t_a2	0.0024	0.0034	-0.027	0.076	0.018	0.045	-0.0095	-0.039	-0.0087	-0.1	0.051	0.17	0.063	0.11	1	-0.064	-0.15	0.094
t_a3	0.042	-0.0014	-0.023	0.019	-0.033	-0.016	0.022	-0.016	-0.0048	0.048	0.02	0.012	0.023	0.017	-0.064	1	-0.17	0.82
t_a4	0.027	0.01	0.031	-0.06	-0.03	-0.024	-0.024	0.098	-0.023	0.06	-0.01	0.11	-0.076	-0.34	-0.15	-0.17	1	-0.08
t_a5	0.0072	-0.016	-0.0042	-0.0085	-0.017	-0.019	-0.0024	-0.012	-0.055	-0.034	0.017	0.1	-0.0099	0.17	0.094	0.82	-0.08	1

Correlation matrix

Some features are highly correlated and, in order to reduce the dimensionality of the features, a threshold of 0.8 was set and variables with a correlation value greater than this value were dropped from the training set (sa_5, ta_5).

This decision was made because, when independent features are highly correlated, changes in one variable would result also in the other and the model output would fluctuate significantly, given small changes.

We also checked the correlation between different features, to understand how some features are influenced from others.
For example, as we expected, x feature is correlated with y features etc.

The features that remains after that pre-features selection are: **up, down, phi, psi, a1, a2, a3, a4** for both residues in contact, so in total 16 predictors.

At this point, we started to build our models predictions, we have chosen three models and we applied different techniques to increment their accuracy or we applied different data modifications to get important information about the data set, in particular the models are:

- **Naive Bayes classifier**
- **Multinomial logistic regression**
- **Linear discriminant analysis**

An important thing to remark is that we reduced the dimension of our data by 50% because, otherwise, our data set was too big (454.193 observations for 17 features) and, if we had not done that, the operation (of features selection, in particular) would have been very slow and it would have been cause crash/freeze our machines.

4.1 Naive Bayes classifier

The first model we analysed is a multi-classification model, called Naive Bayes classifier.
First of all, we calculated the accuracy of that model with all pre-selected features and the 10 k fold accuracy is **0.5906**. We also calculated its confusion matrix, obtained from comparison of training set with 20% of the observations and prediction of it by the model:

	HBOND	IONIC	PICATION	PIPISTACK	SSBOND	VDW
HBOND	22704	77	26	150	0	7466
IONIC	3176	744	0	0	0	1081
PICATION	0	0	0	0	0	0
PIPISTACK	331	0	0	550	0	647
SSBOND	649	0	0	0	82	494
VDW	5923	63	34	323	0	5564

The accuracy is not very high. In particular, the precision for each interaction is:

- HBOND interaction has a precision of 0.6925
- Ionic interaction has a precision of 0.8416
- Pication interaction has a precision of 0.0
- Pipistack interaction has a precision of 0.5376
- SSBOND interaction has a precision of 1
- VDW interaction has a precision of 0.3648

In order to manage the different distribution of the interactions, we applied a re-sampling method. In particular, we set 5.000 observations for each interaction.
For interactions that did not have enough samples, we applied over-sampling and for those that exceed the 5.000 in the number of observations, we applied under-sampling.
Then we studied how the model performed with a more balanced data set.

We obtain a 10 k fold accuracy of **0.7982** on the re-sampled data, but the confusion matrix hides a very interesting new particular:

	HBOND	IONIC	PICATION	PIPISTACK	SSBOND	VDW
HBOND	507	20	0	6	0	269
IONIC	120	1020	0	0	0	116
PICATION	82	0	1109	37	0	148
PIPISTACK	31	0	0	1050	0	96
SSBOND	28	0	0	0	974	42
VDW	215	65	3	12	0	396

The precision of HBOND and VDW interactions is very low, respect to the others ones because these two interactions are very biased.

4.2 Multinomial logistic regression

We trained a model with all 16 features, to see how it performs.
The accuracy obtained is : **0.6748** and its confusion matrix is:

	HBOND	IONIC	PICATION	PIPISTACK	SSBOND	VDW
HBOND	26942	726	31	182	3	10014
IONIC	55	46	0	0	0	31
PICATION	0	0	0	0	0	0
PIPISTACK	87	0	0	118	0	165
SSBOND	1	0	0	0	4	9
VDW	2678	28	23	621	71	3582

There are a lot of missed predicted interaction. In particular, the model tends to classify many observations as HBOND.
This is probably due to the high frequency of this interaction.

4.2.1 Features selection

We tried a subset selection approach. In particular, we couldn't apply a lattice structure approach, because we had a data set too big, with many predictors and doing it would caused a very heavy computational effort. In fact, if we had wanted to choose the best subset selection, we should have trained 2^p models.

In order to avoid that, we tried to use a step wise features selection. Probably, it chosen the best subset of features, but, anyway, it's a good compromise.

With backward step wise selection we obtained that the subset of features selected are: **s-up, s-down, s-psi, s-a1, s-a2, s-a3, s-a4, t-up,**

t-psi, t-a1, t-a2, t-a3, t-a4, so 13 predictors out of 16.

The accuracy obtained with backward features selection is: **0.6760** and this is the confusion the matrix obtained:

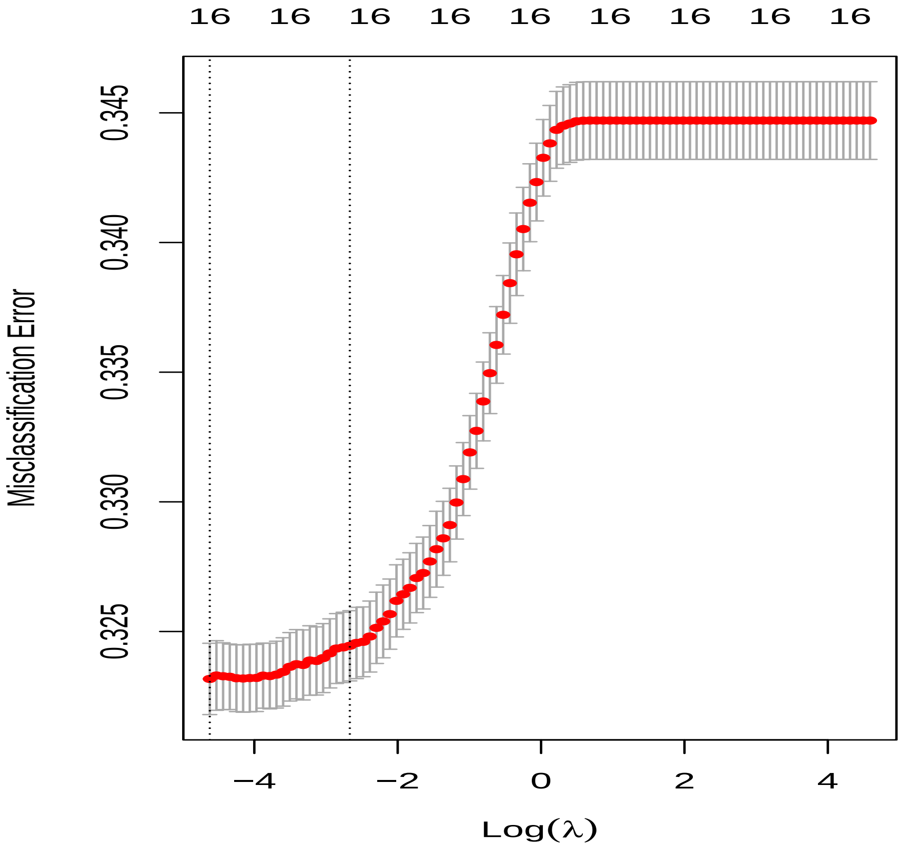
	HBOND	IONIC	PICATION	PIPISTACK	SSBOND	VDW
HBOND	26986	712	27	211	0	9955
IONIC	70	62	0	0	0	43
PICATION	0	0	0	0	0	0
PIPISTACK	88	0	0	105	0	171
SSBOND	13	0	0	0	47	49
VDW	2606	26	27	605	31	3583

Also with this features selection, the accuracy has very low slightly increase, respect to the model with all predictors.

4.2.2 Shrinkage method

At this point, we tried to implement the two shrinked method, studied during the course: Ridge and Lasso regression, which doesn't select features, but hold all of them and try to find the best lambda term to shrink the predictors.

Firstly, we tried with Lasso regression and we can notice how misclassification error changes in function of lambda:



behaviour of misclasification error in function of lambda

The minimum value of lambda, that minimizes the ridge regression is : **0.0098**.
The accuracy obtained with ridge regression approach is: **0.6788**, but, looking at confusion matrix, we can notice only 3 classes out of 6 are predicted.
This is probably due to the fact that there is a big gap between number of observations of different interaction types, so the shrinkage technique tends to minimizes those predictor, that doesn't give a big improvement on overall accuracy.

	HBOND	IONIC	PICATION	PIPISTACK	SSBOND	VDW
HBOND	27162	751	24	242	10	10118
PIPISTACK	10	0	0	22	0	36
VDW	2591	49	30	657	68	3647

After that, we tried to build a model using lasso regression.
The minimum value of lambda, that minimizes the lasso regression is: **0.0050** and the accuracy obtained with that technique is: **0.6783**.
The confusion matrix obtained is below and also here only 3 out of 6 classes are predicted.

	HBOND	IONIC	PICATION	PIPISTACK	SSBOND	VDW
HBOND	27339	749	25	253	12	10318
PIPISTACK	10	0	0	23	0	37
VDW	2414	51	29	645	66	3446

Although the accuracy is slightly increased, respect to the Multinomial model, those models seems not to be very useful, because they predict only half of classes.

We also tried to train and test that model in the re-sampled data, as we did for the Naive Bayes Classifier, but the 10 k fold accuracy is the same. We can notice from its confusion matrix the HBOND and VDW interactions are badly predicted, contrary to the other interaction.

	HBOND	IONIC	PICATION	PIPISTACK	SSBOND	VDW
HBOND	507	20	0	6	0	269
IONIC	120	1020	0	0	0	116
PICATION	82	0	1109	37	0	148
PIPISTACK	31	0	0	1050	0	96
SSBOND	28	0	0	0	974	42
VDW	215	65	3	12	0	396

4.3 Linear Discriminant Analysis

Finally, we tried to build a model using LDA, firstly without features selection or shrinkage method.
The full model gave a 10 k fold cross validated accuracy of **0.67357**, with its confusion matrix below of:

	HBOND	IONIC	PICATION	PIPISTACK	SSBOND	VDW
HBOND	26707	705	23	188	0	9686
IONIC	57	58	0	0	0	46
PICATION	0	0	0	0	0	0
PIPISTACK	242	0	0	244	18	432
SSBOND	13	0	0	0	51	58
VDW	2744	37	31	489	9	3579

4.3.1 Subset features Selection

We applied a stepwise features selection in both direction for the LDA model and the features selected was of ll features out of 16.

The model, obtained with the stepwise selection, have the same number of predictors of the full model and it's exactly the same model, so it has the same accuracy.
The overall accuracy for that model is very similar to the others.

5 Intreprenting the data

As we can see from the previous section, there is not a big improvement in accuracy with the techniques of features selection or shrinkage method.

This is principally due to the fact that there is a very big difference in the number of observation between the different features. In fact, HBOND and Van Der Wals interactions (in particular) has very big number of samples.

However, that features, as we noticed for the Naive Bayes Classifier, are not well predicted. In fact, for re-sampled dataset, the precision is very low, respect to the other interactions, so they are biased predictions.

The confusion matrix, obtained with the model trained with the full dataset, is a false friend, because it seems that the best interaction predicted are HBOND and Van Der Wals, but it's only due to the fact that they have way more observations, respect to the others interactions types.

To get more deep into that point, we built a model, without HBOND and Van Der Vals interaction.
This was the most biased model predictions.

Here, to get more observations of the less frequent classes, we used all the dataset and not only the 50% of it, like we did for the previous models.

We trained a multinomial model, with all predictors on that and checked how it performs.
we get a k-fold accuracy that it's near to perfection (0.9999).
This result confirms our supposition.

	IONIC	PICATION	PIPISTACK	SSBOND
IONIC	1768	0	0	0
PICATION	0	121	0	0
PIPISTACK	0	0	2047	0
SSBOND	1	0	0	166

The most biased interaction are also the most frequent interactions. That bias is probably due to the fact that HBOND and Van Der Vals are low energy interactions, so they can have a wide range of possible values in the features.
On the contrary, other interactions are more stronger, so they probably have a stricter range of features values to describe them and they are simpler to discriminate from each others.