The model of proteolysis

Piotr Dittwald (pd219416@students.mimuw.edu.pl),
Anna Gambin (aniag@mimuw.edu.pl)

April 29, 2010

Abstract

This document presents the original approach for estimating parameters of proteolysis process. Data used to fit the model are taken from mass spectrometric experiments. For parameters estimation the Levenberg-Marquadt algorithm is used. The motivation for model is a hypothesis that discrimination between cancer patients and healthy donors can be based on activity of peptide cleaving enzymes (i.e. peptidases).

1 Introduction

1.1 The process of proteolysis

The biological processes described in this paper belong to *proteomics*, which refers to research on the protein's area. Proteins belong to the set of basic chemical combinations which condition the life on the Earth. Natural proteins consist of 20 main amino acids, which create in cells *polypeptide chains*. These chains are cut by the enzymes named *peptidases* which, according to the kind of cleavages (cutting) that they make, are divided into:

- *exopeptidases* - which cleave near the ends of the polypeptide chain,
- *endopeptidases* - which cleave the polypeptide chain in the middle.

Cleavage of the polypeptide chain causes the decomposition of proteins into peptides and amino acids which is named *process of proteolysis*. There is a is a hypothesis that this process may be useful for discrimination between cancer patients and healthy donors (see [Vil06]).

1.2 Mass spectrometry

We study blood serum data which were analyzed using *mass spectrometry*. This is an analytical method which serves for setting the mass-to-charge ratio for different chemical compounds, in our case - polypeptide chains. On the output of spectrometer we get the mass-to-charge ratio (m/z) as well as the retention time for each reading. Retention time is specific for each experiment and describes the period needed for going through the column separating examined species in the process of *liquid chromatography*.

In nature there occur many stable isotopes. The spectrometer returns many peaks (readings) corresponding to the same chemical compound, which creates so-called *isotopic envelopes*.

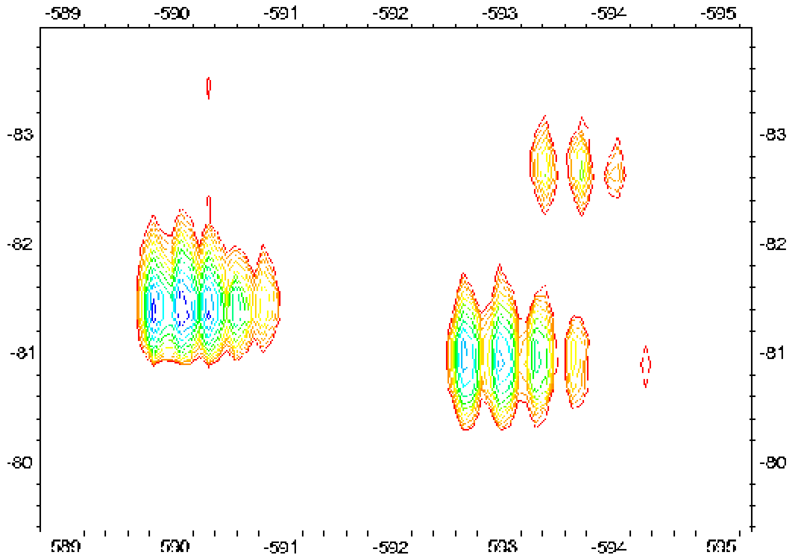


Figure 1: Clusters of peptide signals, i.e. isotopic envelopes, visualized by the Sparky tool [God].

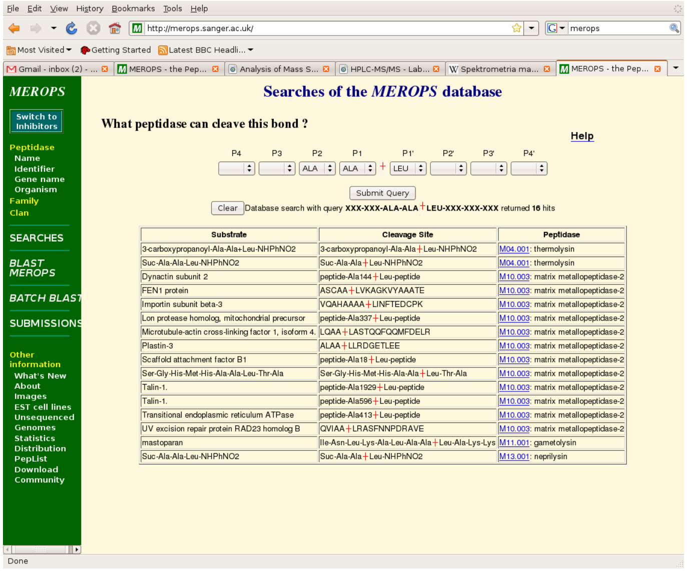
The spectrometer returns the results exceeding the minimal and not exceeding maximal mass limit. Analyzed data are obtained from the LC-MS (*liquid chromatography - mass spectrometry*) machine in the Institute of Biochemistry and Biophysics of the Polish Academy of Sciences. The data sets are acquired from the blood serum from 10 colorectal cancer patients and 10 healthy donors. There was additionally added peptidase named trypsine in order to cut long polipeptide chains into smaller parts that are under maximal mass limit of the spectrometer.

1.3 MEROPS

In order to gather the knowledge about real cleavages we used the MEROPS database [Raw08]. It allows for searching needed information through web-page (c.f. Fig. 2) as well as by downloading the data and running queries on the local machine. Cleavages are described in the database by eight amino acids - four on each side on the cutting locus (if the cleavage is near the end of amino acid chain, then appropriate loci are empty).

1.4 Previous approach - modeling exopeptidases activity

The paper [Klu08] presents the model, where cleavages were made only by exopeptidases (cleaving only one amino acid on each end of the polypeptide chain). For the use of the model an acyclic graph was built where the set $\mathcal{V}$ of vertices refers to the set of peptides and the set E of edges was characterized by the set of pairs (amino acid, end of polypeptide chain). Additionaly two vertices were added: source (associated with the process of creation of peptidases) and sink (associated with the process of degradation of single amino acids).



Searches of the *MEROPS* database

What peptidase can cleave this bond ?

P4	P3	P2	P1		P1'	P2'	P3'	P4'
		ALA	ALA	+	LEU			

Database search with query **XXX-XXX-ALA-ALA+LEU-XXX-XXX-XXX** returned **16** hits

Substrate	Cleavage Site	Peptidase
3-carboxypropanoyl-Ala-Ala+Leu-NHPhNO2	3-carboxypropanoyl-Ala-Ala+Leu-NHPhNO2	M04.001: thermolysin
Suc-Ala-Ala-Leu-NHPhNO2	Suc-Ala-Ala+Leu-NHPhNO2	M04.001: thermolysin
Dynactin subunit 2	peptide-Ala144+Leu-peptide	M10.003: matrix metallopeptidase-2
FEN1 protein	ASCAA+LVKAGKVYAAATE	M10.003: matrix metallopeptidase-2
Importin subunit beta-3	VQAHAAAA+LINFTEDCPK	M10.003: matrix metallopeptidase-2
Lon protease homolog, mitochondrial precursor	peptide-Ala337+Leu-peptide	M10.003: matrix metallopeptidase-2
Microtubule-actin cross-linking factor 1, isoform 4.	LQAA+LASTQQFQQMFDELR	M10.003: matrix metallopeptidase-2
Plastin-3	ALAA+LLRDGETLEE	M10.003: matrix metallopeptidase-2
Scaffold attachment factor B1	peptide-Ala18+Leu-peptide	M10.003: matrix metallopeptidase-2
Ser-Gly-His-Met-His-Ala-Ala-Leu-Thr-Ala	Ser-Gly-His-Met-His-Ala-Ala+Leu-Thr-Ala	M10.003: matrix metallopeptidase-2
Talin-1.	peptide-Ala1929+Leu-peptide	M10.003: matrix metallopeptidase-2
Talin-1.	peptide-Ala596+Leu-peptide	M10.003: matrix metallopeptidase-2
Transitional endoplasmic reticulum ATPase	peptide-Ala413+Leu-peptide	M10.003: matrix metallopeptidase-2
UV excision repair protein RAD23 homolog B	QVIAA+LRASFNNPDRAVE	M10.003: matrix metallopeptidase-2
mastoparan	Ile-Asn-Leu-Lys-Ala-Leu-Ala-Ala+Leu-Ala-Lys-Lys	M11.001: gametolysin
Suc-Ala-Ala-Leu-NHPhNO2	Suc-Ala-Ala+Leu-NHPhNO2	M13.001: neprilysin

Figure 2: The screenshot presenting the interface that allows the user to search cleavages made by peptidases (user may specify amino acids $P4, \ldots, P1, P1', \ldots, P4'$ on both sides from cleavage point) - *http://merops.sanger.ac.uk/cgi-bin/specsearch.pl.*

Sumarizing the model has:

- input intensities $(a_{\star i})_{i \in \mathcal{V}}$,
- output intensities $(a_{i\perp})_{i \in \mathcal{V}}$,
- cutting intensities $(a_{r(i,j)})_{i,j \in \mathcal{V}}$,

where the edge $i \rightarrow j$ is denoted by $r(i,j)$.

The parameters were estimated by Metropolis-Hastings algorithm.

Let $X_i(t)$ be a random variable denoting the number of sequences at node $i \in V$ at time t. Therefore a Markov process $(X(t), t \geq 0)$ is considered in the space of configurations $x = (x_i)_{i \in V}, x_i \in \{0, 1, \ldots\}$.

1.4.1 Theorem (Equilibrium distribution)

The process $(X(t))$ has the equilibrium (stationary) distribution π:

$$\pi(x) = \prod_{i \in \mathcal{V}} \mathrm{e}^{\lambda_i} \frac{\lambda_i^{x_i}}{x_i!}, \tag{1}$$

where the configuration of intensities $(\lambda)_{i \in \mathcal{V}}$ is the unique solution to the following system of “balance“ equations:

$$\sum_{k \in \mathcal{V}} \lambda_k a_{r(k,i)} + a_{\star i} = \sum_{j \in \mathcal{V}} \lambda_i \left(a_{r(i,j)} + a_{i\perp} \right) \tag{2}$$

for every $i \in \mathcal{V}$.

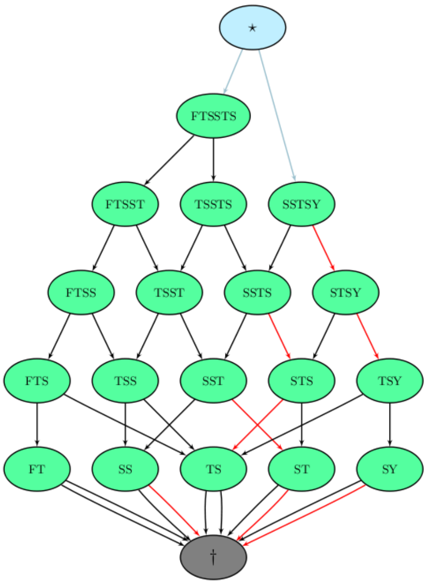


Figure 3: The scheme of the graph used in the previous approach.

1.5 Current approach

In the current model we apply more general approach for the cleavage process. Namely all possible cleavage loci (made by exo- or endopepeptidases) are considered. The data about loci was based on the information from the MEROPS database 1.3. Now the constructed cleavage graph consists of two kinds of vertices:

- those which refer to amino acid sequences (polypeptide chains),
- those refering to cleaving events characterized by the tuple *(peptidase, substrate, product no. 1, product no. 2)*.

The simple scheme of the cleavage graph is presented on Fig. 4 with added extra vertices for creation and degradation processes.

We process data from LC-MS/MS experiments (c.f. Subsection 2.1) and we build the cleavage graph filled with observed amounts of peptide sequences (c.f. Subsection 2.2). Later on, assuming stationary state, we estimate parameters for cleavage events (c.f. Subsection 2.2). We solve non-linear least squares problem by the use of Levenberg-Marquadt algorithm [Mad04] and receive estimated peptidase intensities (c.f. Subsection 2.3). In Subsection 3 we evaluate our model.

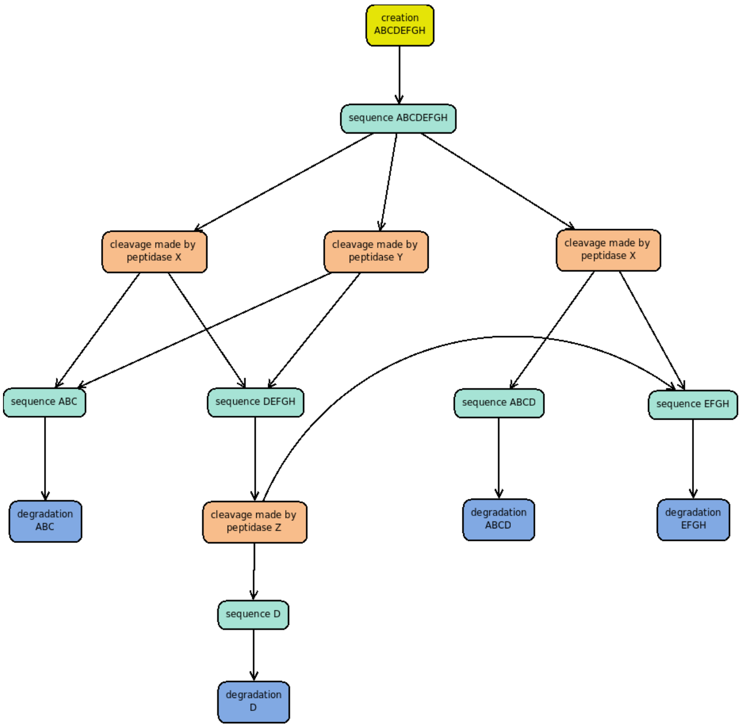


Figure 4: The scheme of the cleavage graph from the current approach.

2 Materials and methods

2.1 Data and preprocessing

Data characteristic

Mass spectrometer returns 20 data set (10 for patients and 10 for healthy donors) with e.g. mass-to-charge ratio, retention time, charge and amount for species detected in blood serum.

Local version of MEROPS database

In order to retrieve information about the cleavages we used downloaded database MEROPS - release 8.5 (see Introduction). The data about the cleavage process are stored in the table Substrate_search which consists of, e.g., the following columns:

- P4, P3, P2, P1 - amino acids on the four successive loci from the left side of the cleavage point;
- P1', P2', P3', P4' - amino acids on the four successive loci from the right side of the cleavage point;
- organism - organism of substrates (the model is built for *Homo sapiens*);
- Protease - peptidase which makes the cleavage.

Specificity matrix

We constructed the specificity matrix for each peptidase by the use of the table Substrate_search. Its construction was based on the approach presented in [Sch90].

Let us define

$$I = \{alanine, cysteine, ...\} \cup \{-\}$$

$$J = \{P4, \ldots, P1, P1', \ldots, P4'\}$$

The set I consists of natural amino acids and the empty position $(-)$ which refers to the lack of amino acid (this situation is characteristic on ends of polypeptide chains). The set J contains loci (4 from both sides) around the cleavage point.

Let $\mathcal{Q}$ be the set of peptidases and $M_c = (m_c)_{ij}$, where $(i, j) \in I \times J, c \in \mathcal{Q}$. Therefore M_c is the matrix with columns corresponding to loci around the cleavage point and rows corresponding to amino acids. The value $m_c[i][j]$ is the amount of amino acid i on position j summed for all cleavages made by peptidase c.

Let us then consider matrix F_c such that:

$$f_c[i][j] = \frac{m_c[i][j]}{\sum_{k \in I} m_c[k][j]} \tag{3}$$

Hence $f_c[i][j]$ is the frequency of amino acid i on position j in cleavages made by peptidase c.

Let us finally construct matrix $S_c = (s_c)_{ij}$, where $(i, j) \in I \times J, c \in \mathcal{Q}$ such that

$$s_c[i][j] = f_c[i][j](\log_2(20) + \sum_{k \in I} f_c[k][j] \log_2 f_c[k][j])$$

We call the matrix S_c the **specificity matrix** for peptidase c.

Pattern matrix

Let us construct the table $\mathcal{T}$ of descending positive real values indexed from 0 to $l - 1$. The values of $\mathcal{T}$ are used to separate different classes of amino acid specificity at considered locus.

Let us also construct the **pattern matrix** $P_c = (p_c)_j$, where $j \in J, c \in \mathcal{Q}, p_c[j] \subset I$. For given peptidase $c \in C$, position $j \in J$ and $i \in I$ we assume that $i \in p_c[j]$ iff the following condition is satisfied:

$$\exists_{0 \leq k < l}\, s_c[i][j] \geq \mathcal{T}[k] \wedge \forall_{0 \leq k' < k} \forall_{i' \in I}\ s_c[i'][j] < \mathcal{T}[k']$$

The following values of table $\mathcal{T}$ were used: $[0.7, 0.1, 0.01]$. Additionally we considered only peptidases with no less than 10 cleavages for substrates from human organism (we used constraints suggested in [Raw09]).

For the given peptidase $c \in \mathcal{Q}$ and the following subsequence of polypeptide chain

$$a_{P4} \ldots a_{P1} a_{P1'} \ldots a_{P4'}$$

such that $\forall_{j \in J}\, a_j \in I$ we assume that there is a cleavage between amino acids a_{P1} and $a_{P1'}$ iff

$$\forall_{j \in J}\, a_j \in p_c[j] \tag{4}$$

MS data analysis

The data sets origin from 10 patients and 10 healthy donors. The blood serum was cleaved by trypsin enzyme before LC-MS processing. For each sequence that occur in cleaving process we want to find corresponding signal in data set. Therefore we use program `mz2m` to obtain a list of mono-isotopic peak coordinates (m/z, retention time and charge) together with their intensities.

The search is processed for each sequence charged by each of eight charges (values form 1 to 8) used by MS machine. The mass-to-charge ratio of searched sequence is computed by the use of the following formula:

$$r(s) = \frac{m(s) + cm_p}{c} = \frac{m(s)}{c} + m_p \tag{5}$$

where

$$m(s) = \sum_{a \in \mathcal{Q}} \#(a, s) m_a \tag{6}$$

Variables are described below:

- $r(s)$ - mass-to-charge ratio for sequence s
- $\#(a, s)$ - how many times the amino acid a occurs in the sequence s,
- m_a - mass of the amino acid a,
- c - charge from set $\{1, \dots, 8\}$,
- m_p - mono-isotopic proton mass that equals 1.0078250321 (1 Dalton), LC-MS machine charge sequences by adding proton.

The retention time is accesible only for some sequences. We used a linear regression model [Has01] to predict retention time from amino acid composition assuming the following:

- the retention time depends on amino acids occurring in the polypeptide chain,
- only the amounts of amino acids are important, not their order in the sequence.

Assuming that we already know the retention time, mass-to-charge ratio and charge for a given sequence, we used nearest-neighbor classifier [Has01] to find appropriate locations on LC-MS spectrum. We used Euclidean metric with retention time scaled by factor 10^{-3}. Signals further than 0.05 are discarded. Intensity is returned as the observed value for appropriate sequences.

The script which makes the described search of relevant masses for given sequences is based on script written by Bogusław Kluge.

We assume that the nearest neighbor found in the described procedure corresponds to sequence and is used for setting the amount of this sequence into appropriate vertex in the cleavage graph.

2.2 Cleavage graph

Graph construction

Basing on the data about cleavages the cleavage graph is constructed. This graph consists of two kinds of vertices:

- *SeqNode* - vertices referring to amino acid sequences, which contain the field with an amount of each sequence (sometimes it is equal to 0, which means that we do not have data about this sequence),
- *ClevNode* - vertices referring to cleavages, characterized by the set *(peptidase, substrate, product no. 1, product no. 2).*

We use also already defined set $\mathcal{Q}$ referring to the set of all peptidases that make considered cleavages.

The cleavage graph construction has the following steps:

1. By the use of the file containing information about cleavages in the sets (peptidase, substrate, product no. 1, product no. 2) we have built objects *clevN* (of class *ClevNode*), objects *lSon*, *rSon* (of class *SeqNode*) and *pep* (of class *Peptidase*). We have combined the mentioned object using the scheme presented on the Figure 5.
2. We have topologically sorted objects of class *SeqNode* and assigned them on attribute *topOrder*.

Graph pruning

1. We have cut (recursively) roots, that were not identified in data set (and we also delete unused object af classes *ClevNode* i *Peptidase*).
2. We have cut (recursively) leaves, that were not identified in data set (with other unused objects)
3. We have deleted roots that are also leaves (connected components that consist of one vertex).

Model parameters

Each peptidase is characterized by the cleaving intensity. We assume that:

- given peptidase has the same intensity in all cleavages it makes,
- cutting intensity in the cleavage process depends on peptidase intensity multiplied by the portion of cleavages this peptidase makes in our graph (♣),
- cleavages process concurrently.

We also assume that the cleavage process takes place in the stationary state, which may be expressed by the kinetic equation (8) (c.f. [vKa92]).

Let us take $i \in SeqNode$ and define $In(i)$ as (maybe empty) set of pairs (a, b) such that

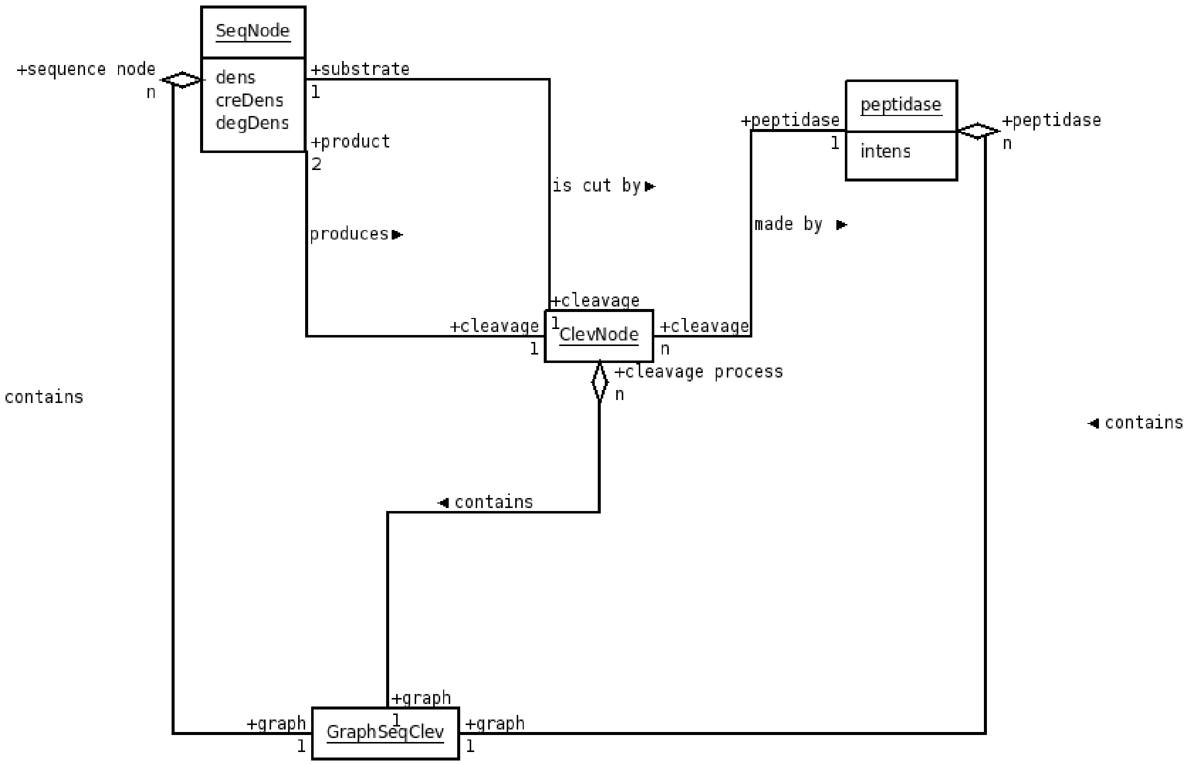


Figure 5: The basic scheme of connection between classes.

- $a \in SeqNode$,
- $b \in ClevNode$,
- b represents the cleavage where a is a substrate and i is a product.

Let us also define the following:

- $Out(i) \subseteq ClevNode$ s.t. $b \in Out(i)$ iff b represents the cleavage where i is substrate,
- s_i is an amount of sequence represented by the vertex i,
- λ_b is an intensity of cleaving by the peptidase which refers to the cleavage represented by the vertex $b \in ClevNode$,
- p_b states for peptidase associated with $b \in ClevNode$,
- $\mathcal{R} \subseteq SeqNode$ s.t. $a \in \mathcal{R}$ iff $In(a) = \emptyset$,
- $\mathcal{L} \subseteq SeqNode$ s.t. $a \in \mathcal{L}$ iff $Out(a) = \emptyset$,
- $\varphi^\star_a$ is an intensity of creation the sequence represented by $a \in \mathcal{R}$ (for $a \in SeqNode \backslash \mathcal{R}$ we set $\varphi^\star_a = 0$),

- $\varphi_a^\perp$ is an intensity of degradation the sequence represented by $a \in \mathcal{L}$ (for $a \in SeqNode \backslash \mathcal{L}$ we set $\varphi_a^\perp = 0$).

We also define

$$\varphi_b = \frac{\sum_{b' \in ClevNode}[p'_b = p_b]}{|ClevNode|}\lambda_b \tag{7}$$

where $b \in ClevNode$ and $[v]$ is the Iverson bracket for logical condition v. Equation (7) reqires the assumption (♣).

Then (at the stationary state) for vertex i the following equation holds:

$$\varphi_i^\star + \sum_{(a,b)\in In(i)} s_a \varphi_b = s_i(\varphi_i^\perp + \sum_{b \in Out(i)} \varphi_b) \tag{8}$$

Left side of the equation above refers to the sum of substances that enter to the vertex i, while right side - the sum of substances that exit this vertex.

The following equation for s_i is a simple consequence of presented formula:

$$s_i = \frac{\varphi_i^\star + \sum_{(a,b)\in In(i)} s_a \varphi_b}{\varphi_i^\perp + \sum_{b \in Out(i)} \varphi_b} \tag{9}$$

2.3 Non-linear least squares problem

In this subsection we use notation already defined in subsection 2.2.

Let us define:

- y_i - amount of peptide sequences identified in LC-MS experiment,
- $\mathcal{O} \subseteq SeqNode$ s.t. $i \in \mathcal{O}$ iff $y_i > 0$ ($\mathcal{O}$ is a set of vertices of kind $SeqNode$, for which the appropriate sequence in identified LC-MS data);
- $\mathcal{C} = (\varphi_a^\star)_{a\in\mathcal{R}}$;
- $\mathcal{D} = (\varphi_a^\perp)_{a\in\mathcal{L}}$;
- $\mathcal{P} = (\lambda_p)_{p\in\mathcal{Q}}$.

Let us define $\Phi = (\phi_i)$ recursively for each $i \in SeqNode$ where the set $SeqNode$ is sorted in topological order:

1. if $i \in \mathcal{R}$ then $\phi_i(\mathcal{C}, \mathcal{D}, \mathcal{P}) = \frac{\varphi_i^\star}{\sum_{b\in Out(i)} \varphi_b}$,
2. if $i \notin \mathcal{R}$ then $\phi_i(\mathcal{C}, \mathcal{D}, \mathcal{P}) = \frac{\sum_{(a,b)\in In(i)} \phi_a(\mathcal{C},\mathcal{D},\mathcal{P})\varphi_b}{\varphi_i^\perp + \sum_{b\in Out(i)} \varphi_b}$.

The topological order assures that the function Φ is well-defined.

We will use function $\Phi(x) = (\phi_1(x), \ldots, \phi_m(x))$, where $m = |\mathcal{O}|$ as so-called *objective function* in non-linear least squares problem described furthermore.

Let us take the function $\psi : \mathbb{R}^n \to \mathbb{R}^m$, where $m \geq n$ and function ψ is non-linear. In *non-linear least squares problem* we want to find:

$$\arg\min_x \{\Psi(x)\}$$

where $\Psi(x) = \sum_{i=1}^m (\psi_i(x))^2$.

In our case we use objective function Φ and define for each $i \in \mathcal{O}$

$$\psi_i(x) = \phi_i(x) - y_i$$

where $x \in (\mathcal{C}, \mathcal{D}, \mathcal{P}) \subseteq \mathbb{R}^m$. The assumption $m \geq n$ is fulfilled if $|\mathcal{O}| \geq |\mathcal{C}| + |\mathcal{D}| + |\mathcal{P}|$.

We applied Levenberg-Marquadt algorithm (LMA) to solve non-linear least squares problem. We used LMA implementation made by Manolis Lourakis (c.f. [Lou04]).

To make single step we have used double precision function `dlevmar_bc_dif` with box constraints and approximating Jacobian by finite difference. By the use of box constraints we have defined lower bound for estimated parameters (we have used small positive real value to grant positive values of parameters).

The main loop of estimation (implemented in `lmmoje.c`) is presented below (real names of implemented data structures and functions may be different):

```
readGraphStructure()
initiateParams()

for each dataset do
 while (j < how_many_estimations) do
  set_startingPoints()
   while (k < maxEstLimit)
    makeSomeSingleSteps()
    getInfoFromSomeSingleSteps()
  done
  getInfoFromThisEstimation()
 done
 printFinalInfoToFiles()
done
```

3 Results and discussion

3.1 Model accuracy

We apply LMA for estimating enzymatic activity for 20 data sets i.e. for each data set we estimated parameters $(\mathcal{C}, \mathcal{D}, \mathcal{P})$ (c.f. subsection 2.3).

The Table 1 presents values of appropriate parameters for each data set. The values *min*, *avg*, *sum* state for minimal value, average and sum of amounts from the set $\mathcal{O}$ (observed peptidase signals) after normalization (each parameter was divided by the maximal value of estimated parameters).

We can see that for each data set the condition $m \geq n$ from subsection 2.3 is fulfilled.

The convergence of LMA

We run LMA for each data set 11 times - each time from different starting point. We used maximal number of iterations as a stop criterion. It was set up to 200

Table 1: General information about the cleavage graphs for all 20 data sets (min, sum, avg state for minimal value, sum of amounts and their average from the set of observed peptidase signals after normalization).

Data set	$\|SeqNode\|$	$\|\mathcal{C}\|$	$\|\mathcal{D}\|$	$\|\mathcal{P}\|$	$\|\mathcal{C}\| + \|\mathcal{D}\| + \|\mathcal{P}\|$	$\|\mathcal{O}\|$	min	sum	avg
1	371	117	163	18	298	335	0.0001595	15.2973172	0.0456636
2	344	95	144	16	255	291	0.0001384	8.6406103	0.0296928
3	381	107	169	16	292	325	0.0001899	13.6410514	0.0419725
4	384	118	182	17	317	340	0.0002693	15.4997184	0.0455874
5	423	125	198	16	339	377	0.0002784	13.236821	0.0351109
6	431	124	204	18	346	389	0.0002449	12.5200073	0.0321851
7	395	118	196	20	334	361	0.0003231	12.3728661	0.0342739
8	412	121	196	20	337	365	0.0002562	11.4179697	0.0312821
9	399	116	183	18	317	349	0.000799	19.3415282	0.0554199
10	410	118	183	17	318	351	0.0004453	18.3664275	0.052326
11	414	115	191	18	324	367	0.0003759	18.6251843	0.0507498
12	433	130	203	22	355	381	0.0002195	12.3124654	0.0323162
13	427	127	193	20	340	377	0.0001663	13.7624148	0.0365051
14	360	112	162	17	291	313	0.0003307	16.2107558	0.0517916
15	368	113	170	17	300	329	0.0002257	16.3597881	0.0497258
16	368	116	171	19	306	327	0.0003153	19.2793821	0.0589584
17	308	87	139	17	243	265	0.0004551	13.7013328	0.0517031
18	384	113	175	19	307	333	0.000608	21.6694534	0.0650734
19	412	114	179	18	311	360	0.0001288	10.4731302	0.029092
20	336	100	158	18	276	299	0.0002337	13.162083	0.0440203

which turned out to be enough to stabilize LMA errors, i.e.:

$$\sum_{i \in \mathcal{O}} (\phi_i(x) - y_i)^2 \tag{10}$$

where ϕ_i is expected amount of peptide sequences in vertex i (defined in subsection 2.3) and y_i is observed amount of peptide in this vertex. The Figure 7 compares minimal and maximal LMA error with the median on each control point (every 5 iteration) for each data set separately.

The Figure 7 compares median values of LMA error for all data sets in the course of experiment. The Table 2 shows the actual value of final LMA error (after 200 iteration) of all data sets. We can see that data set no. 12 has the minimal LMA error. The Figure 8 compares minimal, maximal and median value of LMA error for data set no. 12 during the experiment.

The LMA procedure converges during first 200 iteration. Figure 6 shows that also LMA estimates of peptidase intensities become stable.

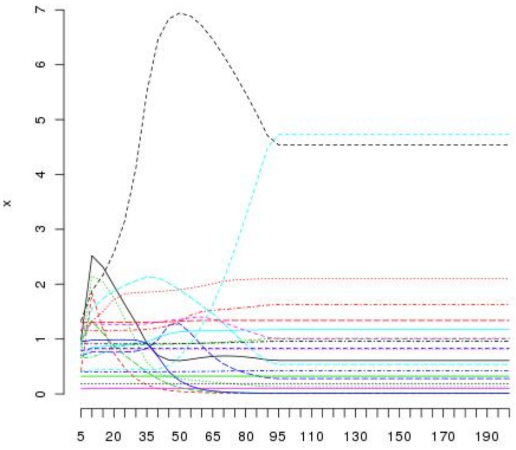


Figure 6: Estimated peptidase intensities during consecutive LMA iterations (one starting point, peptidase intensities are before normalization).

The quality of model estimation

In order to evaluate the quality of model parameters' estimation we made two computational experiments. In the first experiment we compare the results of LMA for real data with the LMA performance for randomly permuted data set (Algorithm 1). Such data set clearly does not reflects the modeled process and as we expected the LMA error is much higher (c.f. Figure 9).

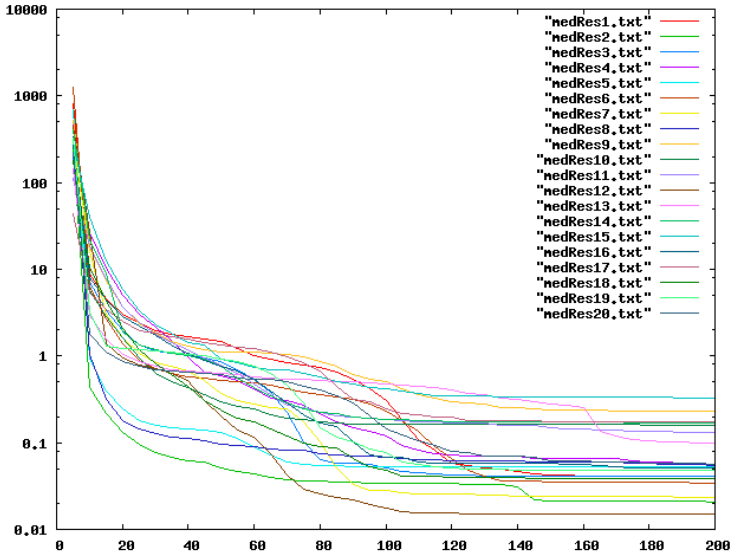


Figure 7: Comparison of median value of error for all 20 data sets.

Algorithm 1

```
repeat 1000 times:
    1. choose pair $(i, j) \in SeqNode \times SeqNode$ at random;
    2. swap amounts of sequences assigned to vertices $i$ and $j$.
```

Statistical significance of estimation quality

Next experiment shows how the final LMA error depends on input data. For data set no. 12 generated 1000 randomly permuted data sets (applying Algorithm 1) and for each data set and we calculated LMA error after 200 iterations. The histogram (Fig. 10) shows the distribution of LMA error (denoted x) as well as the final LMA error denoted by ε for real data. We can also calculate *p-value* for LMA error as $\frac{\sum_{x \leq \varepsilon} x}{1000}$. We used the family of randomly generated data sets to calculate approximated *p-values* for all 20 data sets. The calculated values are presented in Table 2. These *p-values* are oversetimated because we apply Algorithm 1 to the cleavage graph for data set no. 12.

The next experiment justifies the adequacy of the proposed model. We have run the estimation procedure to obtain the model parameters $(\mathcal{C}, \mathcal{D}, \mathcal{P})$. Then we have filled the cleavage graph with data according to our model as follows:

- for each $i \in SeqNode$ we computed $\hat{y}_i$ by the use formula

$$\hat{y}_i = \frac{\varphi_i^\star + \sum_{(a,b)\in In(i)} y_a \varphi_b}{\varphi_i^\perp + \sum_{b \in Out(i)} \varphi_b} \tag{11}$$

  (see also subsection 2.2);

Table 2: Final LMA errors for all 20 data sets. Last column presents approximated $p - value$ (calculated for LMA error distribution from histogram in Fig. 10).

Data set	**Final error**	**Approx. p-value**
12	0.0150286	0.001
2	0.020846	0.007
7	0.023652	0.01
6	0.0342462	0.035
18	0.0383774	0.05
1	0.040668	0.067
3	0.0414344	0.069
19	0.0483224	0.109
16	0.0505154	0.121
5	0.0529894	0.146
8	0.05535	0.159
4	0.0564868	0.168
20	0.0568898	0.17
13	0.0995542	0.499
11	0.1305594	0.646
10	0.16063	0.726
14	0.1711947	0.749
17	0.1735021	0.752
9	0.2304038	0.8
15	0.3304273	0.826

- for each $i \in SeqNode$ we have generated independently $y_i^\star$ from normal distribution with mean $\hat{y}_i$ and standard deviation σ (if $y_i^\star \leq 0$ then there is assigned $y_i^\star := \hat{y}_i$);
- we have chosen σ s.t. $|\sum_{i \in \mathcal{O}}(y_i^\star - y_i)^2 - \sum_{i \in \mathcal{O}}(\hat{y}_i - y_i)^2| < 0.001$

We run LMA with data generated from the described procedure, e.g. $y_i^\star$ - see Figure 11.

In order to present the complete analysis of final error we made also analogous experiment for data set permuted randomly by Algorithm 1. Figure 12 shows the results of this experiment.

The outcomes of peptidase intensities estimation for real data, permuted data and data generated from the model are presented on boxplots which present the distribution for multiple outputs of given parameter (names of peptidases associated with numbers in boxplots are presented in Table 3). Additionally in Figure 15 we have added the line showing median values from Figure 13.

We can see that the data generated from the model have smaller dispersion than real data and similar median value which confirms model accuracy. Also permuted data differ significantly from real data.

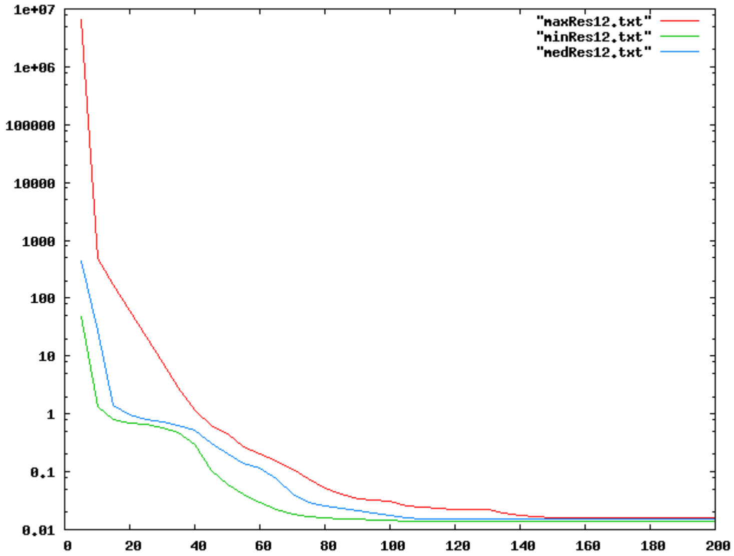


Figure 8: Minimal, maximal and median value of LMA error for data set no. 12.

3.2 Peptidases activity analysis

For each data set we constructed numerical vector of size 13 (peptidases common for all data sets) filled with estimated values of peptidases intensities. Heat map 16 shows the Kendall rank correlation [Ken48] between these data sets. Heat map 17 presents the comparison between ranks of each peptidases between different data sets.

3.3 Summary

We used original model for prediction of peptidase activity. In comparison with the previous approach described in [Klu08] our model is more general (in the sense of possible cleavage loci) and uses information about real cleavage processes. Our model produces adequate results which are very promising for accurate cleavage analysis. The estimation process based on LMA is also more efficent than previously used Metropolis-Hastings algorithm.

References

[Gam07] Gambin, A., Dutkowski, J. et al., *Automated reduction and interpretation of multidimensional mass spectra for analysis of complex peptide mixtures* (2007) International Journal of Mass Spectrometry Vol. 260, Issue 1, Pp. 20-30.

[God] Goddart, T.D., Kneller, D.G., *SPARKY 3*, TR University of California, San Francisco.

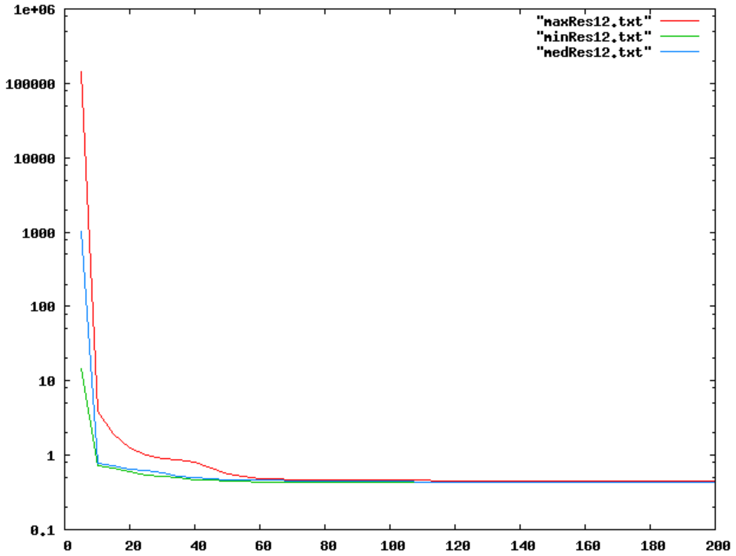


Figure 9: Minimal, maximal and median value of LMA error for data set no. 12 (permuted data).

[Has01] Hastie, T., Tibshirani, R., Friedman, J., *The Elements of Statistical Learning. Data Miming, Inference and Prediction* (2001), Springer.

[Ken48] Kendall, M., *Rank Correlation Methods* (1948), Charles Griffin & Company Limited.

[Klu08] Kluge, B., Gambin, A. & Niemiro, W., *Modeling Exopeptidase Activity from LC-MS Data* (2009), Journal of Computational Biology, Vol.16, No. 2, Pp.395-406.

[Lou04] Lourakis, M.I.A., *levmar: Levenberg-Marquardt nonlinear least squares algorithms in C/C++* (2004), *http://www.ics.forth.gr/ lourakis/levmar/.*

[Mad04] Madsen, K., Nielsen, H.B., Tingleff, O., *Methods for Non-Linear Least Squares Problems* (2004), Technical University of Denmark.

[Raw08] Rawlings, N.D., Morton, F.R., Kok, C.Y., Kong, J. & Barrett, A.J. *MEROPS: the peptidase database* (2008), Nucleic Acids Res 36, D320-D325.

[Raw09] Rawlings, N.D., *A large and accurate collection of peptidase cleavages in the MEROPS database* (2008), Nucleic Acids Research, Vol. 36.

[Sch90] Schneider, T.D., Stephens, R.M., *Sequence Logos: A New Way to Display Consensus Sequences* (1990), Nucleic Acids Research, 18: 6097-6100.

[vKa92] van Kampen, N.G., *Stochastic Processes in Physics and Chemistry* (1992) North-Holland, Amsterdam.

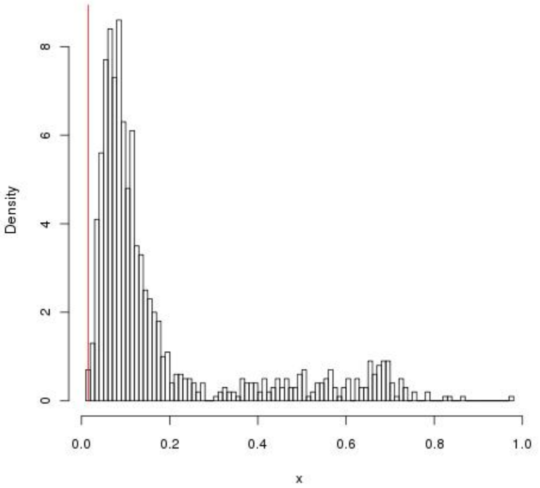


Figure 10: Histogram with LMA error for permuted data; final LMA error for real data is marked in red.

[Vil06] Villanueva, J. et. al., *Differential Exoprotease Activities Confer Tumor-Specific Serum Peptidome Patterns* (2006), Journal of Clinical Investigation, 116:271-284.

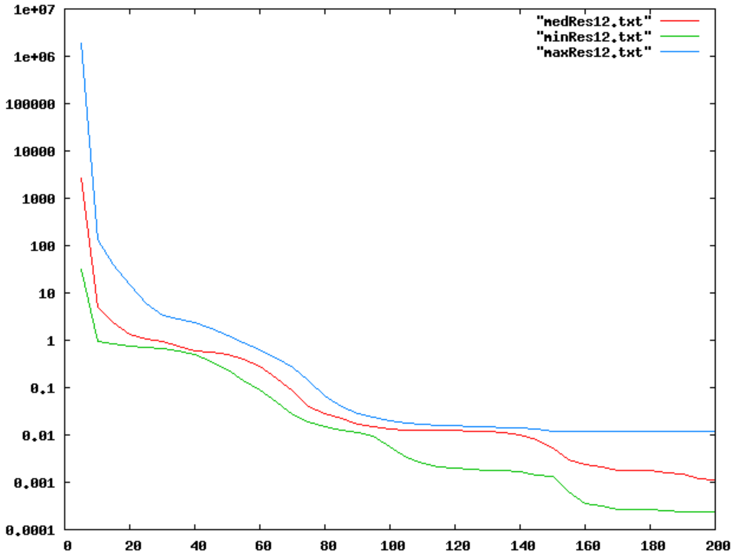


Figure 11: Minimal, maximal and median value of LMA error for data set no. 12 (e.g. data set with minimal final error) with data generated from the model. We used 11 randomly chosen starting points in each of 200 iterations.

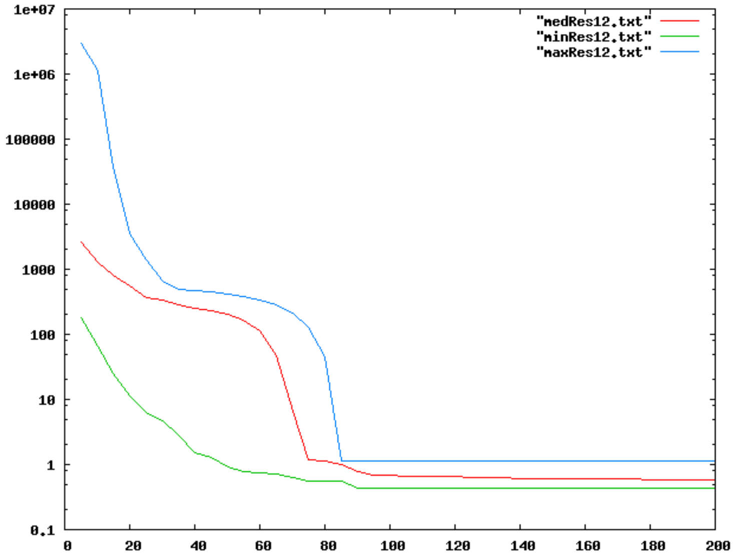


Figure 12: Minimal, maximal and median value of LMA error for data set no. 12 with data generated from the model and randomly permuted data

Table 3: Peptide names for numbers in boxplots.

No.	**Peptidase name**
1	chymotrypsin A (cattle-type)
2	elastase-2
3	cathepsin L1 (Fasciola sp.)
4	calpain-2
5	falcipain-2
6	cathepsin L
7	glutamyl peptidase I
8	plasmin
9	trypsin 1
10	signal peptidase complex (animal)
11	pseudolysin
12	cathepsin D
13	matrix metallopeptidase-7
14	angiotensin-converting enzyme compound peptid
15	membrane-type matrix metallopeptidase-1
16	HIV-1 retropepsin
17	tryptase alpha
18	peptidase 1 (mite)
19	cathepsin S
20	PgPepO oligopeptidase
21	myeloblastin
22	cathepsin B

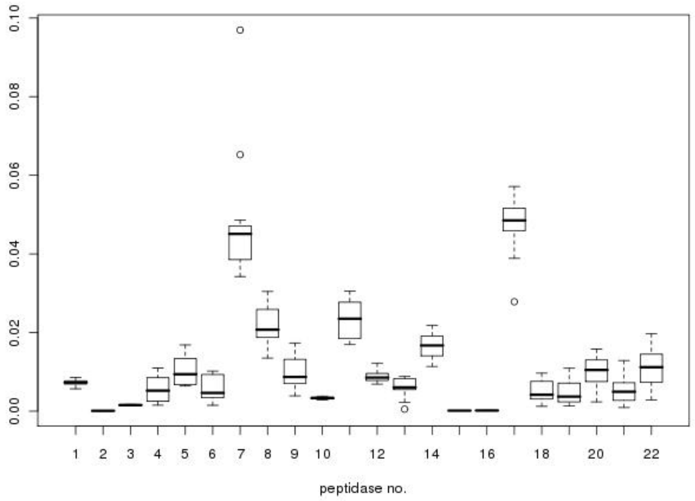


Figure 13: Boxplots for peptidases from data set 4 for real data.

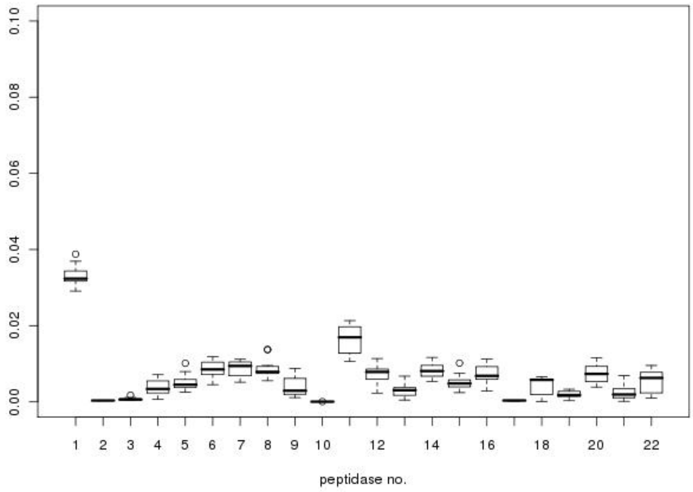


Figure 14: Boxplots for peptidases from data set 4 for permuted data.

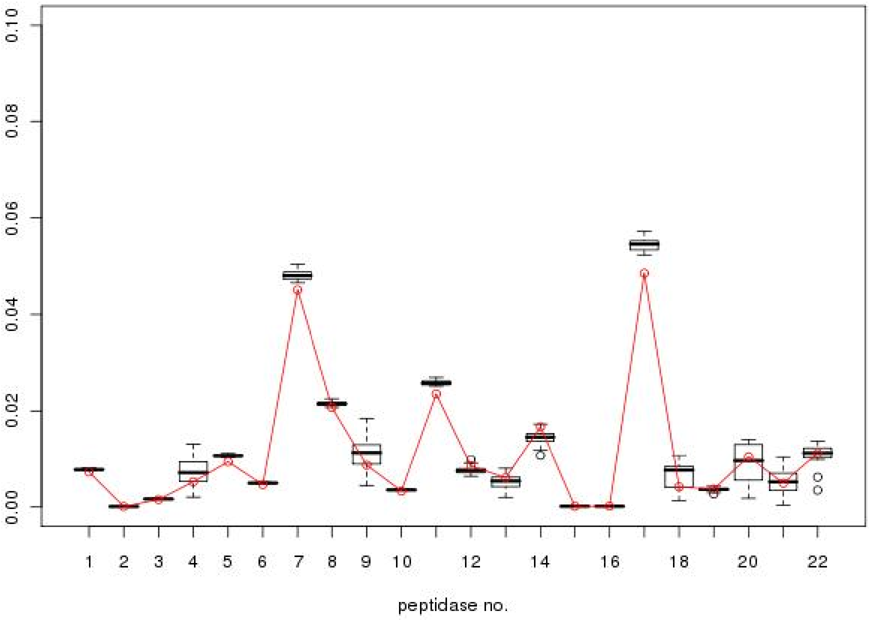


Figure 15: Boxplots for peptidases from data set 1 for data generated from the model with added line linking medians from diagram 13.

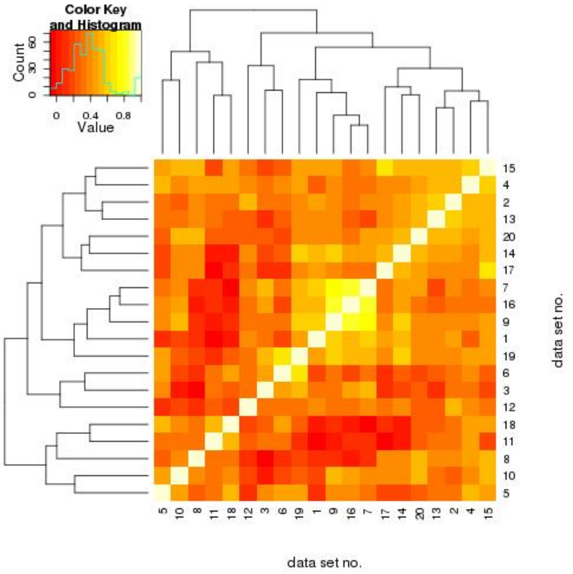


Figure 16: Kendall rank correlation between data sets.

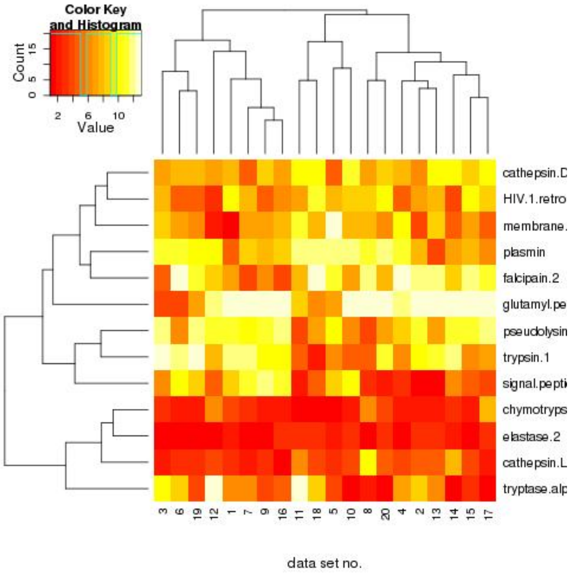


Figure 17: Ranks of peptidases for data sets.