

## **Supplementary methods**

- **Construction of the database**
- **Tutorial 1: How to set-up the MSIPI database in Mascot (Matrix Science) & how to identify the appended peptides**
- **Tutorial2: Installation instructions for scripts**

### **Construction of the database**

All fasta sequences in ipi.HUMAN.fasta were analyzed according to the following procedure:

#### **Analysis and appending of novel N-terminal peptides:**

1. For each entry the N-terminal sequence was submitted to TargetP (including SignalP) using a cut off for mitochondria set to 0.65 to give a specificity of minimum 0.90. At least 60 residues were submitted (and up to 300 if available). Sequences shorter than 60 AA were not considered for this analysis.
2. All positive predictions of signal or transit peptides with a reliability  $\leq 2$  (see TargetP documentation) and with an identified cleavage site were used to create a new N-terminal tryptic peptide starting at the first position after the cleavage site and extended to include one additional tryptic peptide to allow for one missed cleavage.
3. The novel peptide was appended to original sequence using J as a separator. Secondly the length of the original sequence was appended to the header along with information about the appended N-terminal peptide, in the following format: SP[445,L,30,A] or M[445,L,30,A]. See below for further description of header format.
4. Proteins less than 60 residues or with negative prediction was not modified, except for the addition of the length of the sequence to the header description.
5. To all proteins for which peptides were appended the tag # was included in the header.
6. To avoid any confusion the header start was changed from >IPI: to >MSIPI: for all entries.

#### **Analysis and appending of peptides resolved from cSNP**

1. cSNP annotation was downloaded from the Ensembl biomart (<http://www.biomart.org/>)
2. cSNPs were remapped to IPI entries using Swiss-Prot Entry names and Primary accession numbers as listed in ipi.HUMAN.dat.
3. Since IPI entries are based on clusters of proteins, the individual entry may not be 100% identical to the Swiss-Prot entries. Therefore, only entries with the correct residues as annotated in the cSNP annotation were included.
4. For each cSNP – the corresponding residue was substituted and the tryptic peptide in which the cSNP reside created including the two flanking tryptic peptides.
5. For each cSNP for a given IPI entry, the novel peptides were appended to the sequence using J as a separator. Secondly the following tag was appended to the header, in the following format: SNP[371,A,81,G]. See below for further description of header format.

### **Analysis and appending of peptides resolved from Swiss-Prot conflict annotation**

1. Single residue conflict annotation was extracted from Swiss-Prot (uniprot\_sprot.dat downloaded from <ftp://ftp.expasy.org/databases/uniprot/>)
2. Conflicts were remapped to IPI entries using Swiss-Prot Entry names and Primary accession numbers as listed in ipi.HUMAN.dat.
3. Since IPI entries are based on clusters of proteins, the individual entry may not be 100% identical to the Swiss-Prot entries. Therefore, only entries with the correct residues as annotated in the conflict annotation were included.
4. For each conflict – the corresponding residue was substituted and the tryptic peptide in which the conflict reside created including the two flanking tryptic peptides.
5. For each conflict for a given IPI entry, the novel peptides were appended to the sequence using J as a separator. Secondly the following tag was appended to the header, in the following format: CON[444,P,81,G]. See below for further description of header format.

### **Adding protein sequences corresponding to the proteolytic enzyme & keratins**

1. All proteins containing one of the following words (keratin, trypsin, serum albumin, Lys-c) were extracted from Swiss-Prot (uniprot\_sprot.dat).
2. The Entry Names were changed to follow IPI convention: Example:  
>MSIPI:P31940

### **Adding decoy entries to msipi\_decoy.fasta**

1. Every sequence, after appending any possible peptides, was reversed and added as a separate entry.
2. The Entry Names were changed from e.g. >IPI:IPI00000006.1 to  
>MSIPI:REV00000006.1

## Example of modified IPI entry

```
>MSIPI:IPI00000013.1|SWISS-  
PROT:O60911|TREMBL:Q2TB86|ENSEMBL:ENSP00000259470;ENSP00000345344|REFSEQ:NP_001324|H-  
INV:HIT000252685|VEGA:OTTHUMP00000021738;OTTHUMP00000063761 Tax_Id=9606 Cathepsin L2  
precursor lng=334 # SP[336,V,18,V]SNP[371,A,81,G]SNP[411,R,81,G]CON[444,P,81,G] #  
MNLSSLVLAFLCLGIASAVPKFDQNLDTKWKYQWKATHRRLYGANEEGWRRRAVWEKNMKMI  
ELHNGEYSQKKGFTMAMNAFGDMTNEEFRQMMGCFRNQKFRKGVFREPLFLDLPKSV  
DWRKKGYVTPVKNQKQCGSCWAFSATGALEGQMFRTGKLVSLSEQNLVDCSRPOGNQG  
CNGGFMARAFQYVKENGGLDSEESYPYVAVDEICKYRPENSVANDTGFTVVAPGKEKAL  
MKAVATVGPISVAMDAGHSSFQFYKSGIYFEPDCSSKNLDHGVLVVGYGFEFEGANSNNSK  
YWLVKNSWGPPEWGSNGYVVKIAKDKNNHCGIATAASYPNVJVPKFDQNLDTKJMIELHNG  
EYSQKKGFTMAMNAFADMTNEEFRQMMGCFRMIELHNGEYSQKKGFTMAMNAFRDM  
TNEEFRMIELHNGEYSQKKGFTMAMNAFPDMTNEEFRQMMGCFR
```

To this protein four peptides were added: (added information in **bold**)

- One alternative N-terminal peptide corresponding to the removal of a signal peptide 17 residues long. Position of the appended peptide starts at 336 (residue=V). The original start position was at residue 18.
- Two alternative SNP peptides. Both for which the original position of the actual SNP is 81 (residue=G), and now at position 371 (residue A) and 411 (residue R), respectively.
- One alternative conflict peptide – also from original position 81 and now at position 444.

Note, that while signal and transit peptides positions specify the start of the peptide, the SNP and conflict positions specify the actual position of the SNP/conflict.

The occurrence of a “#” indicates that the sequence has been modified.

The occurrence of the second “#” may be used to detect if the header information has been truncated by e.g. the search engine.

## MSIPI\_slim

A third version of MSIPI contains shortened header information in the following format:

```
>MSIPI:IPI00000013.1|Cathepsin L2 precursor lng=334 # SP[336,V,18,V]SNP[371,A,81,G]SNP[411,R,81,G]CON[444,P,81,G] #
```

The changes include:

- Removal of all other identifiers than the IPI entry name
- Removal of the taxonomy tag
- Truncation of protein descriptions longer than 100 characters

## Tutorial 1: How to setup the MSIPI database in Mascot (Matrix Science) & how to identify the appended peptides

### How to set-up the MSIPI database in Mascot (Matrix Science)

1. The resulting database: msipi.fasta is setup in MASCOT like any other IPI database, but with a different parsing rule due to the new header start: >MSIPI:

Rule to parse accession string from Fasta file:

Rule 40: ">MSIPI:\{...\[^|]\*\}"

Rule to parse description string from Fasta file:

Rule 30: ">\*|\{.\*\}"

2. Change the Proteolytic enzymes and specific chemical cleavage agents which are defined in *config/enzymes* by adding:

\*

Title: TrypsinMSIPI

Cleavage[0]:KR

Restrict[0]:P

Cterm[0]

Cleavage[1]:J

Cterm[1]

Cleavage[2]:J

Nterm[2]

\*

3. Set the mass value for J in the *config/masses*, by adding:

J: 114.04293, 114.1026

4. Remember to specify this new reagent "TrypsinMSIPI" used for protein digestion.

## How to identify the appended peptides

The appended peptides may be identified a number of different ways, but here are two examples.

1. Use the length of the original proteins (which has been added to the header description) to look for peptide occurrences outside this range.
2. Use PeptideIndex.txt which contains information about all added peptides in the following format – separated by space:

- IPI entry name
- Changed Peptide
- Changed Peptide with flanking peptides
- Original from position (of changed Peptide with flanking peptides)
- Original to position (of changed Peptide with flanking peptides)
- Type of modification
- new position
- new AA
- Original position
- Original AA

Here is an example:

IPI00000270.1 KPK KPkgMTSSQWFK 29 40 Signal 158 K 30 M

In the case of N-terminal peptides, the position is the starting position.

## **Tutorial2: Installation instructions for scripts**

### **How to download, Install and run the MSipi Setup.**

#### **To Download & Install msfriend**

1. unpack1 – gzip –d ./msipi.tar.gz
2. unpack2 – tar –xvf ./msipi.tar

#### **Install third party predictors: TargetP and SignalP**

Software and licenses for TargetP and SignalP must be obtained through:

Center for Biological Sequence Analysis BioCentrum-DTU Technical University of Denmark  
Kemitorvet, Building 208 DK-2800 Lyngby Denmark **Phone:** +45 45 25 24 77 **Fax:** +45 45  
93 15 85 **Email:** [cbs@cbs.dtu.dk](mailto:cbs@cbs.dtu.dk)

1. Change the full path in the top of the signalp script
  - a. Do the change in this script: ./MSIPI/bin/signalp-3.0/signalp
  - b. See details in ../MSIPI/bin/signalp-3.0.readme
2. Change the paths in the top of the targetp script
  - a. Do the change in this script: ./MSIPI/bin/targetp-1.1/targetp
  - b. See details in ../MSIPI/bin/targetp-1.1.readme
  - c. It is important to set the tmp directory for TargetP to the tmp directory for MSIPI  
./MSIPI/tmp

#### **To Run**

Run it from ./MSIPI/DATABASES/ dir

You can e.g. run it this way: nohup ../bin/msipi.sh &

**Here follows a graphical overview of the process**

