MACHINE LEARNING FOR PROTEINS’ FUNCTIONS

למידה ממוחשבת לפיענוח תפקודי חלבונים

Dan Ofer

This dissertation is submitted for the degree of Master of Science
December 2015

THIS WORK WAS CARRIED OUT UNDER THE SUPERVISION OF: PROF. MICHAL LINIAL
DECLARATION

This dissertation is the result of my own work and includes nothing, which is the outcome of work done in collaboration except where specifically indicated in the text. It has not been previously submitted, in part or whole, to any university of institution for any degree, diploma, or other qualification.

Signed: Dan Ofer  

Date: 18.11.2015
Systematic identification of protein function is a key problem in current biology. Most traditional methods fail to identify functionally equivalent proteins if they lack similar sequences, structural data or extensive manual annotations. I focused on identifying diverse classes of proteins that share functional relatedness but little sequence or structural similarity, notably, Neuropeptide Precursors (NPPs).

Neuropeptides (NPs) are short secreted peptides produced in neuronal cells via cleavage from longer NPPs. They may act as neuromodulators and neurohormones, with many roles in cellular communication and regulation. NPs act via GPCRs, by activating signaling cascades governing broad functions such as metabolism, sensation and behavior in all Metazoa. Existing alignment based tools fail to identify them, due to their diversity and length.

I aim to identify functional protein classes solely using unannotated protein primary sequences from any organism. This task required the identification of biologically and statistically meaningful characteristics amenable to statistical learning, which we refer to as features. This thesis focuses on feature representations of whole protein sequences, novel sequence derived engineered features, their extraction, the building of frameworks for their usage by statistical learning (“machine learning”, ML) models, and the application of ML models to biological tasks, focusing on high level protein functions.

I implemented the ideas of feature engineering to develop a platform (called NeuroPID) that extracts meaningful features for successful classification of both known and overlooked NPPs using machine learning. The platform allows mass
discovery of new NPs and NPPs. It was further expanded as a freely available webserver.

I expanded our computational approach towards identifying other challenging protein groups with alignment-free statistical features. This is implemented as a novel bioinformatics toolkit called ProFET (Protein Feature Engineering Toolkit). ProFET extracts hundreds of biophysical and sequence derived attributes, allowing the application of machine learning methods to any set of proteins.

ProFET was applied on many protein benchmark datasets with state of the art performance. The success of ProFET applies to a wide range of high-level functions such as metagenomic analysis, subcellular localization, structural fold and unique functional properties (e.g. thermophiles, nucleic acid binding). The ML approach is part of a growing body of methodologies that complement classical sequence-alignment based approaches to protein family classification.

The platforms that I developed for this thesis allow the use of a holistic machine learning approach to the pressing problem of protein function prediction. These methods and frameworks represent a valuable and implemented resource for using ML and computational data science methods on proteins.

The work presented here is based on a number of my peer-reviewed research articles (Ofer D et al., 2014; Karsenty et al., 2014; Ofer and Linial, 2015), as well as unpublished work.


**Abstract:**

Understanding the functional roles that proteins play today is a central problem in biology. In the past, protein function was assigned based on structural and similarity criteria according to sequence alignment. In many cases, the classic methods are not effective. For example, in cases where there is no similarity or structural differences between proteins with a unique and shared functional feature, the problem is highlighted for identifying abstract functions—such as identifying proteins that belong to a large family compared to identifying a closely related homolog family.

For example, the precursors of neuropeptides.

Neuropeptides are short molecules released, occurring due to the cuts of longer precursors called precursors. They are present in all living organisms and are involved in a wide range of biological functions: social behavior, reproduction, metabolism, sensation, and more.

Analytical tools that are based on sequence alignment do not succeed in identifying new neuropeptides, due to the shortness of their similarity and length.

I have built several tools that are dedicated to solving this problem, together with many other proteins that have specific functions not identified by existing methods.

I have developed NeuroPID, which is a computational platform for identifying new neuropeptides and precursors. The identification is based on quantities that we directly absorb from the sequences of the proteins. We use analytical properties and model predictive models. Additionally, in the project, we built a website that accompanies this tool, for easy access to the entire scientific community.

Furthermore, I expanded the tools and the statistical methods that I used to identify neuropeptides to identify neuropeptides and their characteristics for the families, possible functions, and sub-families. All of this was collected in ProFET (Protein Feature Engineering Toolkit).

I have therefore developed new techniques for representing complete proteins as a collection of "precursors" and a system to generate and measure these precursors from new sequences and their characteristics.

These tools and methods that I developed during these studies allow the use of computational (ML) methods to solve the many and difficult problems of protein identification and their functions. ML methods strengthen our ability to learn and identify abstract properties of proteins and characteristics of the sequences without an external database of information. The first automatic system for identifying precursors of neuropeptides. Systems for processing and identifying proteins through computational methods. New methods for identifying quantitative properties in order to classify functional differences or abstract, which classical methods failed to deal with.
Ofer D et al., 2014; Karsenty et al., 2014; Ofer and Linial, 2015)
ACKNOWLEDGEMENTS

First, I would like to thank Prof. Michal Linial. Without her I would never have entered the field, or encountered the myriad opportunities it led me to. She gave me the opportunity to try and learn, despite my beginning as a Tabula-Rasa. She was available for fascinating discussions around the clock. She always knew to see to the core of any matter, and taught me to always be aware of wider scientific meaning and purpose even while probing away in research.

I thank the mentors and professors I have had the opportunity to learn from over the years, and who gave me a chance.

I thank my friends for support, advice, feedback and comradery.

I thank the many authors whose works inspired me – Asimov, Tolkien, Pratchett, Dawkins, Moore, Gaiman, Herbert, Clarke and dozens more.

I am thankful to all the members & friends of the lab, past and present. For helping when I was stuck, for assisting when I was at a loss, for giving ideas and sharing time and advice. Thanks Nadav R., Nadav B., Kerem, Shelly, Ohad, Roni, Amos & Nati.

Last but not least, I thank my wonderful family for endless support, love and encouragement. I thank my parents, sisters, and grandparents. I would not be around without them.
## CONTENTS

1 Introduction .......................................................................................................................... 10

1.1 Machine Learning & Data Science .................................................................................. 14

1.2.1 Data Collection .............................................................................................................. 14

1.2.2 Feature Extraction .......................................................................................................... 14

1.2.3 Feature Engineering ....................................................................................................... 15

1.2.4 Feature Selection ............................................................................................................ 15

1.2.5 Model Selection .............................................................................................................. 16

1.2.6 Hyperparameter Tuning ................................................................................................ 16

1.2.7 Evaluation (Cross-Validation) ...................................................................................... 16

2 METHODS - goals & background ....................................................................................... 19

2.1 Sequence-alignment methods for protein classification: ................................................ 19

2.2 A machine learning approach for protein classification: ................................................ 20

2.3 Defining levels of Protein “Function” .............................................................................. 22

2.4 Short proteins & peptides: an overlooked niche ............................................................... 23

1.2.4 Neuropeptides: Behavioral & physiological regulators ................................................. 24

1.2.5 Neuropeptide Processing .............................................................................................. 25

1.2.6 Neuropeptide Discovery ............................................................................................... 26

2.5 Goals ................................................................................................................................ 27

3 METHODS – Data and implementation ............................................................................. 28

3.1 Protein datasets ................................................................................................................ 28

3.2 Features ............................................................................................................................ 30

3.3 Evaluation ......................................................................................................................... 33

3.4 Machine Learning Methods ............................................................................................. 34

3.5 Feature selection (RF-RFECV) ......................................................................................... 34

3.6 Software and implementation .......................................................................................... 34

4 RESULTS ............................................................................................................................. 35

4.1 Platforms ......................................................................................................................... 35

4.2 NeuroPID Results ............................................................................................................. 36

1.2.1 NeuroPID: Alternate Negative Set Performance ......................................................... 37

1.2.2 NeuroPID: Taxonomical Evaluation ............................................................................. 38

1.2.3 NeuroPeptide Statistics ............................................................................................... 38

1.2.4 NeuroPID Webserver ................................................................................................... 39

1.2.5 Neuropeptide Taxonomic Validation ......................................................................... 41

1.2.6 NeuroPID Discovery protocol ..................................................................................... 42

4.3 ProFET Results ................................................................................................................ 44

1.2.7 ProFET Benchmarks .................................................................................................... 44

1.2.8 ProFET Case studies ................................................................................................... 46
1.2.9 Sequence Alignment Comparison .......................................................... 46
1.2.10 ProFET: Feature Selection ................................................................. 47

5  הדיון וסיכום – DISCUSSION .................................................................. 49
5.1 ProFET Perspectives .................................................................................. 49
5.2 Pitfalls & Tricks ......................................................................................... 50
5.3 ML and Predictive Proteomics ................................................................. 51
5.4 Overall Conclusions .................................................................................. 53

6 Bibliography ................................................................................................. 54
7 Appendices .................................................................................................... 63
Appendix 1 NeuroPID: a predictor for identifying neuropeptide precursors from metazoan proteomes ............................................................... 64
Appendix 2 NeuroPID: a classifier of neuropeptide precursors ...................... 65
Appendix 3 ProFET: Feature engineering captures high-level protein functions .... 66
Appendix 4: NeuroPred Supplementary: Statistical Significance Analysis of Features: Neuropeptides vs Background ......................................................... 67
Statistical Significance of Features Analysis: Neuropeptides vs Background ........ 67
Appendix 5 In-Silico Cnidaria Predictions ...................................................... 71

LIST OF TABLES
TABLE 1. PERFORMANCE OF THE TOP NEUROPID ML MODELS ....................... 37
TABLE 2 NEUROPID ALTERNATIVE NEGATIVE SETS PERFORMANCE .................... 37
TABLE 3 NP CLASSIFICATION CV WITH TAXONOMICAL PARTITIONS .................... 38
TABLE 4 NEUROPID PREDICTIONS FOR INDIVIDUAL SPECIES ............................. 41
TABLE 5 NEUROPID PREDICTIONS FOR HONEY BEE AND OCTOPUS (GUNTERI) PROTEOMES ...... 43

LIST OF FIGURES
FIGURE 1 GROWTH IN SEQUENCED PROTEINS. SOURCE ..................................... 13
FIGURE 2 DISTRIBUTION OF SEQUENCE LENGTHS IN THE UNIPROTKB DATABASE .......... 13
FIGURE 3 ML CLASSIFICATION PIPELINE ................................................................ 15
FIGURE 4 K-FOLD (10-FOLD) CROSS VALIDATION ILLUSTRATION ...................... 18
FIGURE 5 METRICS ......................................................................................... 33
FIGURE 6 CROSS VALIDATION PERFORMANCE OF DIFFERENT MODELS ON NEUROPID’S FULL, ORIGINAL DATASET (NPP V1) ................................................................. 36
FIGURE 7 FEATURES’ STATISTICAL SIGNIFICANCE FOR NPS VS THE BACKGROUND (UNIREF50) DISTRIBUTION .................................................................................................................... 39
FIGURE 8 THE NEUROPID WEBSITE .................................................................... 40
FIGURE 9 NEUROPID PIPELINE ............................................................................ 42
LIST OF ABBREVIATIONS AND ACRONYMS

AA, Amino Acids
BLAST, basic local alignment search tool
PSI-BLAST, Position-Specific Iterated BLAST
CAFA, critical automatic functional annotation
CASP, critical Assessment of protein Structure Prediction
ClanTox, classifier of animal toxins
CNS, central nervous system
ICI, ionic channel inhibitor
MS, mass spectrometry
nAChR, nicotinic acetylcholine receptors
NP, neuropeptide
NPP, neuropeptide precursor polypeptide
PSSM, position specific scoring matrix
SP, signal peptide
TMD, Trans-membrane domain
TOLIP, toxin-like proteins
GPCR, G-protein-coupled receptors
PC, Prohormone Convertases
SWP, Swiss-Prot
SVM, support vector machine
SVC, support vector machine classifier
RF, Random Forests
GBR, Gradient Boosting Trees
RBF, Radial Basis Function (kernel)
CV, Cross validation
HMM, Hidden Markov Model
ANN, Artificial neural network
ROC, receiver operating characteristic
AUC, Area under ROC curve
ML, Machine learning
NN, Nearest-neighbours
F1, Balanced F1 score (mean of precision and recall)
TP, True Positives
TN, True Negatives
FP, False Positives
FN, False Negatives
Recent years have seen an exponential increase in biological data, particularly of gene and protein sequences (Figure 1). Modern Next Generation Sequencing (NGS) technologies have created a reality where the extraction of biomolecular information (i.e., their genome, proteome, transcriptome, peptidome, etc’) is an increasingly routine act for individual researchers, thanks to the plummeting costs of sequencing. This revolution is also leading to increasing diversity in what is being studied, with non-model organisms being sequenced, resulting in an explosion of novel proteins with unknown functions. Understanding protein function is a foundational issue in biology. While expert manual annotation might have sufficed in the past, when most research was limited to a handful of intensely studied, often similar, organisms (e.g., Humans & lab rats), this approach fails to scale to the modern “big data” explosion of newly discovered proteins. The flood of high-throughput and rapidly sequenced proteomic data cannot be handled by performing individual experimental studies to determine the function(s) of every single protein, as was traditionally the case. Computational prediction is currently the only feasible approach for “unlimited”-throughput identification of protein function (Loewenstein et al., 2009) The only feasible way to assign functions to this deluge of sequences is with state-of-the-art, scalable computational methods for automated functional annotation and analysis.

An additional, complementary task in the context of labelling protein functions is finding proteins which share various novel functions of interest (as opposed to annotating a given sequence with a certain function), e.g., discovering new neuronal receptors, antimicrobial activity, chaperones or secreted peptides. Thus, the two tasks are finding the functions of a sequence and finding sequences which share a function.
Overcoming the growing gap between unknown and annotated sequences requires improving and expanding automated methodologies; Improving means better predictive power, identification of functions that are currently neglected (e.g., “high level” functions (Varshavsky et al., 2007), or toxin-like short proteins (Tirosh et al., 2013; Naamati et al., 2009)), while expansion necessitates wider participation by diverse experts in the scientific community, lowering barriers of entry and incorporating the knowledge of non-computational biologists.

The most used approach in protein classification relies on distance measures between sequences according to sequence-alignment methods (e.g., Smith-Waterman (Smith and Waterman, 1981), BLAST (Basic local alignment search tool Altschul et al., 1990)). The approach is one of “guilt by association”: If sequence A has function X, and sequence B is sufficiently similar to sequence A, then sequence B is annotated with A’s functions. Similarity may be based on parts of a sequence, e.g. specific domains within a protein or short motifs (e.g. conserved binding sites).

With the growth in the amounts and diversity of protein sequences, more sophisticated methods have been introduced (e.g., PSSM, HMM-HMM) (Jaakkola et al., 2000; Biegert and Söding, 2009; Söding, 2005). These methods use multiple sequence alignments (MSA), improving sensitivity in detecting remote homologues (Edgar and Sjölander, 2004; Karplus et al., 1998).

3D-structure can serve to define seeds for defining statistical models further improving the quality of protein domains and families (e.g., Pfam) (Finn et al., 2014), but this is severely limited due to the limitations of structural prediction (Berg et al., 2002). InterPro (Mulder and Apweiler, 2007) has ~27,000 such models, covering 83% of all sequences in UniProtKB (2014_10). Function assignment is gained from mapping InterPro models to Gene Ontologies (i.e. InterPro2GO). Alternative model-free approaches have been proposed (Portugaly et al., 2002). Such approaches require alternate representations of the sequences as sets of features, followed by fitting ML models. The assessment of large-scale automatic protein functional annotations (e.g. CASP, CAFA) (Radivojac et al., 2013) and the contribution of alternative approaches has been discussed extensively (Juncker et al., 2009; Yachdav et al., 2014; Obradovic et al., 2005).

Despite the specialized strengths of the sequence-alignment/model-based methods, in many cases they fail at reliably assign function (Rentzsch and Orengo, 2009). This is best demonstrated by their limitations in classifying 3D structural folds (Dubchak et al., 1995; Ding and Dubchak, 2001; Greene et al., 2007; Kister and Potapov, 2013), a task at which even the best sequence
similarity based methods underperform. Many biological niches and functional properties are especially unsuited to this form of representation. For example, routine annotation tools fail to confidently assign function to bioactive peptides and short proteins (Naamati et al., 2009).

A number of previous studies focused on features extracted from whole protein sequences, (Nanni et al., 2010; Dobson et al., 2009; Scheraga and Rackovsky, 2014; Dubchak et al., 1995; van den Berg et al., 2014; Nanni et al., 2014; Varshavsky et al., 2007) as a starting input for machine learning methodologies. Structural benchmarks (3D structural classes, folds or superfamilies) from SCOP and CATH (Greene et al., 2007; Andreeva et al., 2014) were used to assess many of these. Specialized predictors have been presented for a wide range of structural and sequence-property tasks including secondary structure, solvent accessibility, disordered regions, domains and more (C. Z. Cai et al., 2003; Ding and Dubchak, 2001; Cheng and Baldi, 2006; Deng et al., 2015; Nagarajan and Yona, 2004; Barla et al., 2008). ML approaches have proven suitable to classify many protein properties beyond ‘just’ structure, e.g. SVMProt was tested on 50 Pfam families (C. Z. Z. Cai et al., 2003). Feature based ML classification outperformed alignment-based methods for many protein families (Varshavsky et al., 2007). Alignment based distance metrics can even be incorporated as additional features (Liu et al., 2014). However, the most likely advantage of the feature and pattern-based ML approach is in learning high-level functions (e.g. ProFET (Ofer and Linial, 2015)). Examples for such include protein-protein interactions (Bock and Gough, 2001; Cheng and Baldi, 2007), discriminating outer membrane proteins (Gromiha and Suwa, 2005), membrane topology (Nugent and Jones, 2009), and subcellular localization (Petersen et al., 2011; Yachdav et al., 2014; Goldberg et al., 2014). However, most ML protein sequence analysis studies in the field focused on residue (rather the whole-sequence) level properties, such as secondary structure, disorder or PTMs such as proteolytic cleavage. The design of feature representations for residue level properties is far easier conceptually, as features are extracted from fixed length regions (“windows”) rather than variable length whole proteins, e.g. encoding the amino acids in each window position as a feature. I would argue that this helps explains the popularity of ML methods in residue level predictive tasks, as opposed to whole protein level prediction.
This background serves as a basis for the approach I took in creating a ML platform for predicting high level protein functions (including NPPs).

We illustrate the significance of machine learning tools in identifying and annotating short bioactive proteins and peptides from many genomes. Over 5,700,000 full-length proteins from metazoans are currently archived, and some ~15% are short or secreted proteins (Figure 2). Most short sequences remain uncharacterized. Furthermore, due to incomplete genomes and transcriptomes, many sequences are actually fragments.

The systematic discovery of overlooked secreted peptides, neuropeptides and toxin-like proteins allows for new strategies in research and behavioural manipulation, shining a light on peptides as biologically and clinically important entities.

**Figure 1** Growth in sequenced proteins. Source: UniProtKB Release Statistics 2015_08.

**Figure 2** Distribution of sequence lengths in the UniProtKB database (excluding fragments). Lengths range from 2 to 35,213, Average: 356. Distribution covers 46 Million sequences. Source: UniProtKB Release Statistics 2015_08.
1.1 Machine Learning & Data Science

I describe the steps I applied during my research and will share the logic and concepts fundamental to assessing success and addressing the challenges inherent to applying ML tools to proteins. To implement any supervised ML task, a number of steps must be performed, regardless of the field. These include:

1.2.1 Data Collection
Data must be defined and gathered as a dataset. This involves the curation, acquisition and sometimes creation of the data, samples and measurable parameters. In the case of proteins, this means acquisition of the protein sequence, relevant annotations (e.g. class labels, subcellular localizations, structural fold, genomic data), storage in machine-readable formats and data cleaning (“munging”), e.g. removal of unreliable sequences, errors, and inaccurate labels.

1.2.2 Feature Extraction
Raw data must be transformed into a quantitative, computable representation. It must be amenable to representation as a fixed length vector of features, as the vast majority of ML methods require a fixed length input (of any arbitrary length). This is a completely different task for every domain: in the case of computer vision, it means extracting raw pixel values (from images or “crops” of identical sizes). In the case of stocks or Electroencephalograms, it involves extracting time-series and amplitudes. In the case of natural text, it may involve counting the words or characters (“Bag of Words”). A typical machine learning classification pipeline is shown in Figure 3.

This is extremely challenging in the case of DNA and protein sequences, due to the lack of any “baseline” features, and the differing lengths of whole-protein sequences. We don’t have a fixed size (or resizable) digital image with a known number of pixels at each location, or a document with clear “words” or sentences (unlike in natural language).
1.2.3 Feature Engineering

“Raw” extracted features are rarely sufficient, i.e. statistically significant, discriminative and distinctive enough. Typically, new features must be created from the baseline feature representation to improve performance. For example, in the case of natural text, instead of counting words, count per document word-frequency (Luhn, 1957). As a more specific example, rather than counting the frequencies of each AA, we can count the frequencies of groups of AA (Reduced AA alphabets) that share a property (e.g. positive charge – Lysine and Arginine).

Creating or “engineering” good features is one of the major problems in applied ML (David Kofoed Wind, 2014; Domingos, 2012; Qi et al., 2012; Ofer and Linial, 2015). This problem is currently nigh-impossible to automate generically, and extensive domain-specific knowledge is required to invent novel features for different problems.

1.2.4 Feature Selection

Removal of irrelevant and redundant features is important, especially when we can create more features than samples. This can be optional, typically if the number of features is not overly large and the features are non-redundant, uncorrelated and definitively “relevant” (an optimistic scenario at best). At all steps, a potential “feature explosion” (when many possible features may
be extracted), (Nanni et al., 2014; Rao et al., 2011; Ofer and Linial, 2015) “the curse of dimensionality” and overall performance must be taken into account.

Feature selection also serves to help us understand the important features in our model, so we can interpret and understand it better. Many different techniques exist for filtering features, notably filter methods (e.g. removal of features according to statistical criteria such as p-values, or highly correlated features), wrapper methods and model-based methods (e.g. deriving variable importance from model weights (Guyon et al., 2002; Guyon, 2003; Peng et al., 2005)).

1.2.5 Model Selection
Various learning algorithm exist for data analysis and classification. Various methods have different benefits and drawbacks; for example: prior performance, bias and variance, robustness to noise, suitability for imbalanced classes, training and prediction speed, ease of implementation, interpretability, non-linear feature interaction handling, and many other such considerations. We tested many different models and often ensembled (combined) multiple models (Dietterich, 1990; Karsenty et al., 2014; Ofer and Linial, 2015). Support vector machines (SVM) and decision tree-based classifiers (e.g., Random forests) were nearly always our models of choice, offering excellent performance, interpretability, handling of imbalanced classes and ease of use (Fernández-Delgado et al., 2014; Breiman, 1999; Ben-Hur et al., 2008).

1.2.6 Hyperparameter Tuning
Many parameters must be set prior to any model fitting. Such hyperparameters are tuned to optimize model learning capacity (e.g. degrees of freedom) and performance. Such parameters range from the choice of model and preprocessing (e.g. feature selection) to various parameters inherent to each model, such as kernel types in SVMs (Cristianini, 2001; Society), learning rate and network structure in neural networks (Bishop, 2006), the number of trees in RF (Breiman, 1999; Fernández-Delgado et al., 2014; Touw et al., 2013), etc’.

1.2.7 Evaluation (Cross-Validation)
Classifier performance is usually evaluated using Cross-validation (CV) (Kohavi, 1995). The model is trained on part of the data, then predictions are made on a separate validation subset which was not seen during model training. Performance is best measured by a models ability to generalize to unknown data.
Data may be further divided into a training set, validation set, and an additional “holdout” test set. A holdout is important when extensive hyperparameter tuning is involved, since the parameters are being adjusted according to the CV performance purely on validation data may be biased, (e.g. feature selection, ANN network architectures).

In k-fold cross-validation (Geisser, 1993; Kohavi, 1995), the training set is split randomly into m equal disjoint subsets/“folds”. In this work, we mostly used stratified K-fold CV (Kohavi, 1995), in which each fold preserves a percentage of samples for each class, making it useful for imbalanced data, avoiding folds populated exclusively by one class. Next, one subset is left out of the training set, and the training is performed on the remaining subsets. Finally, the labels of the instances in the subset which was left out are predicted. This procedure is repeated for each subset, so that “blind” predictions are obtained for the whole training set. The predictions are then evaluated according to some metric such as accuracy.

High performance in cross-validation should indicate that a classifier generalizes well, i.e. will correctly predict new instances as well as it did on known data. However, training data is not always drawn from the same distribution as the real world distribution, nor are its samples necessarily independent and identically distributed (i.i.d.). For example, if “redundant” or near identical sequences are in the training set (e.g. a pair of 99% identical orthologs), then near-identical sequences might be found in both the training and validation set, while in the real world, we would be unlikely to need our model for such “trivial” cases (we’d simply annotate according to sequence similarity). Thus, misleadingly high performance would be obtained by the classifier even if it over-fits the data, since it will have seen a “copy” of the “unknown” sample. This may result in poor generalization and overestimation of model performance. Therefore, it is important to reduce data redundancy (not just between the training and test sets), using tools such as Uclust (Edgar, 2010), CD-HIT (Huang et al., 2010; Sikic and Carugo, 2010) or Uniref clusters (Suzek et al., 2007, 2015). In short, care should be taken to ensure the dataset is non-redundant, and reflects real-world scenarios, rather than an idealized, trivially easy versions. I note that in my experience, many preprocessed “academic” datasets have a marked tendency to be “easier”. Competitions such as CASP, CAFA, DREAM and Kaggles’ can be a good source of challenging, realistic validation.
Figure 4 K-fold (10-fold) Cross Validation illustration. Accuracy can be any metric.
2.1 Sequence-alignment methods for protein classification:

Functional annotation is normally done using a supervised approach, *i.e.* inferring functional classification for a sequence according to existing sequences whose functions are known. The problem can be stated as follows: given are an unlabeled protein sequence $S$ and a known protein family (or class) $F$. We want to determine whether or not $S$ belongs to $F$. In general, a family is a group of proteins with similar structure and function. If the unlabeled sequence $S$ belongs to $F$, then we can infer the function of $S$.

The most naïve, supervised approach is the nearest-neighbor search (Sasson *et al.*, 2006), relying on sequence alignment and inferred homology. Given our query sequence, a database of sequences is searched with the goal of identifying similar sequences with preexisting annotations. The most common algorithm for this is BLAST (Altschul *et al.*, 1997), which performs local sequence alignment. If a significantly similar sequence is found, the query sequence will be considered to possess a similar function; *i.e.* 'guilt by association'. A rule of the thumb for this sort of inference is the ‘twilight zone’ concept: a sequence at least 100 amino acids long is likely to be a homolog if at least 30% of its amino acids are identical (Rost, 1999). Below this value the sequence is in the ‘twilight zone’, where similarity cannot be separated from randomly occurring similarity (using traditional sequence alignment).

Although direct inference is useful for many sequences, it suffers from critical caveats:

(i) In order to learn about a sequence there must exist a significantly similar sequence whose function is known, essentially precluding function prediction for unknown protein families, or
those for whom too few “samples” exist to create a statistical model (i.e. a PSSM profile or HMMs (Finn et al., 2014; Remmert et al., 2011; Lavelle and Pearson, 2010)).

(ii) Many proteins with similar sequences have different functions and would therefore be mistakenly classified as having the same function, e.g. paralogs.

(iii) Many proteins share functionalities, active sites or domains but possess significantly different sequences, despite having similar functions. E.g. NPPs/Prohormones, wherein the “prohormone” regions flanking the NPs within the sequence are relatively unconserved (Toporik et al., 2014).

(iv) Proteins may share high level function, while sharing little to no structural similarity, let alone sequence based identity. E.g. Disordered proteins (Gitlin et al., 2014; Sharma et al., 2014; Orosz and Ovádi, 2011; Uversky et al., 2000).

(v) Sequence similarity based methods lack significant statistical power given extremely short sequences, i.e peptides. E.g. a pair of sequences of length 200 with 40% similarity might have a pair of shared domains and ~80 aligned AAs. The NPP Pro-opiomelanocortin is just 36 AA long, ~15 AA are rarely a constituent unit of molecular information. Additionally, such short alignments lacks statistical power, i.e BLAST e-value (Akhtar et al., 2012).

(vi) Structural similarity is what usually defines function, but proteins with the same structure can have significantly different sequences.

2.2 A machine learning approach for protein classification:

When sequence alignment is meaningless, e.g., only a small number of sequences can be aligned, or when the information content of the multiple alignment is minimal, other methods have to be adapted. We approached the problem aiming to develop methods that are not directly sequence-alignment based, but rather rely on extracting alignment-free sequence-derived quantitative features. Each sequence is transformed into a fixed-length vector of features. A discriminative (or generative) statistical model, such as a Support Vector Machine (SVM) (Leslie et al., 2002; Ben-Hur et al., 2008; Cortes and Vapnik, 1995) or decision tree (e.g. RF (Kingsford and Salzberg, 2008; Breiman, 1999, 1996)) is then fitted to the data. Such models can then be used to classify new samples/sequences into the learned classes.

Such methods avoid the requirement for sequence alignments have shown success in learning high-level functional traits (such as the high level of protein family structural folds), while often
being far more computationally efficient than NN searches. While both the alignment-based and the sequence-derived feature approaches may use the same information as input, namely the sequence itself, they can perform very differently due to the manner in which they exploit the data and the information they extract from it. There are cases in which an intelligent choice of numerical features (i.e., those that best capture the characteristics of the sequences) can significantly outperform alignment models (e.g., HMM and PSSM), and can be extended for a variety of other problems such as gene co-expression data (Kahanda et al., 2015).

**SUPERVISED LEARNING**

In the statistical learning framework of supervised learning, a group of known, annotated samples serve for the learning/training phase. This group of sequences is referred to as the training set. Once the learning stage is complete, the computationally learned hypothesis can be used to classify unidentified samples into a learned class. The advantage of machine learning methods over the sequence-similarity approach (described above) is the fact that the learning methods can take advantage of heterogenous, non-linear features, potentially identifying the minimal conserved set of characteristics in each family of proteins and focusing on searching only for these characteristics. Furthermore, these methods can potentially learn the underlying structure of the problem, discovering unsuspected correlations and important features, and improving as the size of the dataset grows (Halevy et al., 2009). In addition, they may be far faster than the computational demands required for (all to all) sequence-similarity based approaches (Zhao et al., 2012; Edgar, 2004; Hochreiter et al., 2007). These flexible methods can be much more powerful than “brittle”, pre-defined rules (e.g. regular expressions) or the naïve sequence similarity approach.

The field of machine learning is immense with a strong impact on biology (Ben-Hur et al., 2008; Tarca et al., 2007). Machine learning algorithms include SVMs and kernels (Lodhi, 2012; Leslie et al., 2004; Cristianini, 2001; Kuang et al., 2005), decision trees, Neural Networks (NNs) (e.g. Extreme learning machines (Huang et al., 2015; McDonnell et al., 2014), Convolutional NNs, Recurrent NNs (Sønderby et al., 2015; Lipton, 2015; Hochreiter et al., 2007)), decision trees, ensembles of classifiers (such as Random Forests and AdaBoost (Breiman, 1999; An Empirical Comparison of Supervised Ensemble Learning Approaches)), and unsupervised methods for clustering (Rappoport et al., 2010).
2.3 Defining levels of Protein “Function”

Protein “function” can be defined at many levels – both in terms of molecular function, biological processes and cellular localized activity (in the manner of GO (Ashburner et al., 2000)), and in terms of “high/low level” function. For example, whether a protein is a membrane bound receptor, a G-protein coupled receptor (GPCR), a GPCR that uses cyclic AMP (Alberts et al., 2009), or an olfactory GPCR sensitive to CO2 (Tauxe et al., 2013; Lu et al., 2007; DeGennaro et al., 2013; Hallem and Carlson, 2006).

There is an obvious connection between the ‘granularity’ of a function and the evolutionary diversity of the proteins that share it (Rost, 1999; Kobayashi et al., 1998; Dubchak et al., 1995). Typically, groups of proteins that share a high-level functionality (i.e., enzymes or membrane receptor) are larger and more diverse than low-level (e.g., urease enzymes) functional groups (Shachar and Linial, 2004). Community competitions in functional assignment, such as the Critical Automatic Functional Annotation initiative show that there is considerable room for improvement (Radivojac et al., 2013; Gillis and Pavlidis, 2013; Bacardit et al., 2014).

While inferring precise, “low level” functionality is often of interest (e.g. annotation of functional homologs (Loewenstein et al., 2009; Bork and Koonin, 1998)), there is an obvious interest in successfully learning high-level functionality (Sasson et al., 2006; Naamati et al., 2009). If the training set consists of proteins that share a low-level functionality, the classifier would only be able to detect proteins that belong to the narrow function that was learned. However, if the training set consists of proteins that share a high-level functionality, the classifier will be able to detect any protein that belongs to a very broad class, even if ‘close’ representatives are unknown.

As an example, consider the case of the Major Facilitator Superfamily (MFS). This superfamily includes over 300,000 proteins capable of transporting small solutes in response to ion gradients. In general terms, proteins of the family belong to the transmembrane transport system. However, if we use classifiers of low-level functionality, we would have a classifier for the different types of MSF families (including Nitrate transporter, Sialic acid transporter and many more). If a gene for a novel subtype of MSF were found, we would not be able to identify its function at the lower level as it would not belong to any known transporter family. However, if our classifier was trained to identify “MSFs”, we could identify the sequence as a (novel) MSF.
It is difficult to learn high-level functionality, particularly when we might barely understand them on the theoretical level. While we might expect nitrate transporters from different organisms to share similar sequences due to evolutionary homology, we would not expect this of a high-level group such as all transmembrane transporters.

Structure (rather than sequence) is the major element for most (but not all!) proteins’ function (Orosz and Ovádi, 2011; Berg et al., 2002). Structural knowledge and predictive capabilities is severely lacking, despite improvements in computational prediction thanks to the ongoing, targeted mapping of the structural genomic-fold space (Portugaly and Linial, 2000), and improvements in predictive methods (Magnan and Baldi, 2014; Zhou and Troyanskaya, 2014; Sønderby and Winther, 2014; Remmert et al., 2011). Experimental determination of structure using NMR or X-ray crystallography remains slow, expensive and impossible for many proteins (Carpenter et al., 2008), with just ~110,000 determined structures for tens of millions of proteins (Berman et al., 2000; Rappoport et al., 2012; Boutet et al., 2007).

2.4 Short proteins & peptides: an overlooked niche

The ability to learn about a protein by comparing it to its (inferred) homologs has been used in functional prediction, secondary structure prediction, three-dimensional fold prediction and several other applications (Qi et al., 2012). However, the power of sequence alignment/similarity-based tools is greatly diminished for short proteins. This is because when comparing short sequences it is difficult to distinguish genuine homology from mere evolutionary noise/coincidence. For example, submitting a short amino acid sequence to a sequence-similarity search server such as BLAST will usually result in matches with insignificant e-values (a statistical measure of significance for the expectation value), even for sequences with high percentages of identity. Therefore, the detection of short proteins’ homologs using sequence-alignment tends to fail (Naamati et al., 2009).

The difficulty in the identification of homologs is only one of the problems associated with short proteins (Frith et al., 2006). Consider a newly sequenced genome. The first task is identifying potential (putative) gene products. The main steps for identifying encoded proteins include:

- Sequence similarity: While this method is the most powerful computational approach, as indicated, it fails to detect short proteins.
Comparative genomics: This method requires the aligned genomes of related species. Additionally, this method is likely to fail to detect short proteins for similar reasons to the sequence similarity approach.

Ab-initio gene prediction: The default parameters require a minimal length for potential ORFs (Open Reading Frames), which may further hinder the detection of short proteins.

High coverage of the transcriptome and proteome by high throughput (HT) technologies. Some experimental methods focus on detecting mRNA expression and others on protein expression. HT methods such as RNA-Seq (Wang et al., 2009), Ribosomal profiling (Ingolia et al., 2012) and Mass Spectrometry (MS) (Aebersold and Mann, 2003) are perhaps the best source of experimental data for detecting new proteins, but are often far from comprehensive due to the fact that many genes are only expressed under certain conditions (Ponting and Grant Belgard, 2010). Furthermore, discriminating between peptides and “noise” (e.g. random fragments) is extremely challenging, with even the best high-throughput protein expression methods such as tandem MS (MS/MS (Angel et al., 2012)) requiring special tweaking and is limited to the detection of known, highly expressed proteins (Akhatar et al., 2012; Salisbury et al., 2013; Vogel and Marcotte, 2008; Kim et al., 2011; Craft et al., 2013). Often, if a short protein is not already a known candidate, it will not be found (Lubec and Afjehi-Sadat, 2007). As a consequence of these computational and experimental difficulties, short proteins represent a relatively understudied and neglected niche (Tirosh et al., 2013; Frith et al., 2006; Tirosh et al., 2012; Naamati et al., 2009; Su et al., 2013).

1.2.4 Neuropeptides: Behavioral & physiological regulators
Neuropeptides (NPs) are short proteins (peptides) that are produced and secreted from nervous system cells (e.g. Neurons), acting as inter-cellular modulators and messengers. They are known to be key modulators in behavior, sensation and homeostasis. They function in neurobiological communication for all Metazoans, with similar roles in Cnidarians and Bilaterians (Mirabeau and Joly, 2013; Semmens et al., 2015).

The NPs are very short active peptides (~5-30 amino acids) produced from longer precursor proteins (NPPs) following proteolytic cleavages. The post-translational end products are subsequently modified and secreted as the actual NPs.

NPs normally locally modulate presynaptic or postsynaptic cell activity (Root et al., 2011;
Nathoo et al., 2001; García-López et al., 2002; Brain and Cox, 2006). NPs typically work by binding cell surface GPCRs, resulting in the initiation of a signaling cascade (Tanaka et al., 2014; García-López et al., 2002; Hewes and Taghert, 2001; Root et al., 2011). From a functional perspective, known effects of NPs include stress control (Chang and Hsu, 2004; Hökfelt et al., 2000), pain perception, social behaviors and sleep-wake cycle (Nässel, 2002), food uptake (Nguyen et al., 2011; Taghert and Nitabach, 2012), selective appetite (Beshel and Zhong, 2013; Dillen et al., 2013) and more. "Social" NPs such as oxytocin (OXT) and arginine vasopressin (AVP) regulate mating (e.g. PBAN - activates pheromone biosynthesis (Rafaeli, 2009)), complex social cognition and behavior (including pair-bonding, social recognition and maternal behavior) (Gruber and Muttenthaler, 2012; Ferguson et al., 2000; Insel and Young, 2000).

NP sequences are mostly non-alignable due to their shortness, diversity, and lack of sequence similarity (Clynen et al., 2010). Sequence similarity methods fail to predict or provide a comprehensive catalogue of NP bioactive peptides or their precursors. Their immense diversity of sequence, targets, pattern of post-translational modification and specificity (García-López et al., 2002; Hewes and Taghert, 2001) reflects their equally immense range of behavioral and physiological effects in diverse organisms (Nathoo et al., 2001; García-López et al., 2002; Shaw, 1996; Brockmann et al., 2009).

1.2.5 Neuropeptide Processing

The majority of NPs are produced from a larger precursor (NPP, as a result of a series of post-translational modifications and cleavages (Veenstra, 2000). NPP can produce multiple copies of different NPs as well as multiple copies of each NP (Smit et al., 1992; Fisher et al., 1988; Nässel, 2002). Notably, a cluster of basic residues specifies these cleavage sites by the family of Prohormone Convertases (PCs), intracellular endopeptidases such as Furin (Veenstra, 2000). However, many NPs (e.g., cathepsin L (Funkelestein et al., 2010)) do not obey the dibasic residues specificity rule. Furthermore, only a fraction of sites containing the “canonical” (“Known Motif” (Southey, Rodriguez-Zas, et al., 2006)) dibasic cleavage pattern are actually cleaved (Devi, 1991; Southey, Rodriguez-Zas, et al., 2006; Amare et al., 2006; Southey, Amare, et al., 2006; Southey et al., 2008; Tegge et al., 2008; Amare and Sweedler, 2007). Many non-secreted, non-NPPs proteins also undergo this cleavage!
1.2.6 Neuropeptide Discovery

One goal of this research was to enable the systematic identification of NPPs (and NPs) at a genome-wide scale. The difficulty in identifying NPs and classifying genes as potential NPs stems from the following: (i) Current gene annotation tools mostly rely on sequence conservation traits (Loewenstein et al., 2009). However, NPs exhibiting the same function may share minimal sequence similarity, and can even lack 3-D structural similarity (Clynen et al., 2010). The NPPs too can vary immensely, sharing only varying amounts of different internal NPs. Additionally, homologous NPPs may still produce NPs that are species specific.

(ii) Structural inference tools (Lobley et al., 2009; Radivojac et al., 2013; Kuznetsov and Rackovsky, 2003; Midic et al., 2009) fail when applied to short peptides.

(iii) NPPs undergo distinctive cleavage by PCs, but they’re not unique in this regard! Thus, even the presence of identified PC cleavage sites, via tools such as NeuroPred (Southey, Amare, et al., 2006; Southey et al., 2008) or our own forthcoming ASAP/CleavePred\(^1\), is not enough to identify NPPs. Rather, these can only help identify NPs given a NPP.

Consequently, assigning functions to known NPPs and identifying previously overlooked genes calls for developing an alternative strategy. Methods based on sequence alignment don’t work, and strict, high-bias (Hastie et al., 2009), “hand-designed” rules (e.g. regular expressions or motifs (Dinkel et al., 2014)) lack the capacity to model this problem. Towards this goal, we developed our ML approach.

\(^1\) Forthcoming work. Code and methods available at: https://github.com/ddofer/asap
2.5 Goals

To summarize, my research had a number of goals which were answered through this work. These are collated in a number of peer-reviewed articles (attached in the appendix). The goals and articles include:

- A universal feature extraction & ML framework for proteins, suited for classification of high-level functions and unified properties, without external databases or alignment.
  - Implemented as ProFET (Protein Feature Engineering Toolkit) (Ofer and Linial, 2015)\(^2\).

- Developing a universal set of feature extraction methods and novel engineered features for efficient feature representations of whole protein sequences.
  - Implemented as NeuroPID (Ofer D et al., 2014)\(^3\), and the NeuroPID web-server (Karsenty et al., 2014): [neuropid.cs.huji.ac.il](http://www.protonet.cs.huji.ac.il/neuropid)

- Creation of a publicly available platform for identifying novel Neuropeptides precursors & Neuropeptides on a massive scale.
  - Performed as part of NeuroPID (Ofer D et al., 2014), on multiple organisms, as well as further work\(^4\); See Appendix – a sample of predicted Cnidarian sequences. (Unpublished).

---

\(^2\) [https://github.com/ddofer/ProFET](https://github.com/ddofer/ProFET)

\(^3\) NeuroPID source code freely available at: [http://www.protonet.cs.huji.ac.il/neuropid](http://www.protonet.cs.huji.ac.il/neuropid)

\(^4\) [www.protonet.cs.huji.ac.il/neuropid/results](http://www.protonet.cs.huji.ac.il/neuropid/results)
3.1 Protein datasets

In gathering the protein datasets in this study, we used (i) custom sets gathered from public databases such as UniProtKB/UniRef (Suzek et al., 2007; Boutet et al., 2007) and SCOPe (V2.05) (Fox et al., 2014) and (ii) benchmarks extracted from publications. For both resources, we applied CD-Hit (Huang et al., 2010) and USearch (Edgar, 2010) to remove redundant and similar sequences according to a predefined % threshold of sequence identity. As a rule, we used only classes that contained a minimal number (~40) of samples per group (after redundancy removal). Sequences with unknown amino acid (AA), errors or sequences that are shorter than 30 AA were removed. I note that different versions of the NPP dataset were used: V1 was used in the original NeuroPID paper. V2 was used in ProFET and was based on V1, with various changes, namely a different negative set.

The datasets are:

**Specialized functions**

- Ribosomal proteins: Acquired from SWP and partitioned to Archea, Bacteria and Eukarya. Redundancy filter was set to 20–40% identity (according to the set size).
- Thermophilic proteins: The ThermoPred (Lin and Chen, 2011) dataset was used, consisting of 915 thermophilic and 793 non-thermophilic (Mesophile) proteins. Further filtered for < 40% sequence identity.
- DNA-binding proteins. Benchmark dataset from DNA binder (Kumar et al., 2007).
- RNA-binding proteins. Benchmark dataset from BindN (Wang et al., 2010).

**Subcellular localization:**

- LocTree3 benchmark (Goldberg et al., 2014) for Eukarya and Bacteria were used. Filtered at 40% identity within each class.
Mammalian subcellular localization: Protein-organelle pairs acquired from SWP.

Uncultured bacterium sequences extracted from UniProtKB and mapped by keyword annotations for major cellular compartments (membrane, cytoplasm, ribosome). Filtered by UniRef50 clusters. 15,995 sequences.

Structural classifications:

SCOPe (Release 2.05, Feb 2015) (Andreeva et al., 2014; Fox et al., 2014). Classes and folds were defined by SCOP, with 25% or 10% sequence identity filter. (8,514 and 6,721 sequences respectively).

SCOPe (Release 2.05, Feb 2015) “partial classes” used - defined by the SCOP class (marked a-k), with classes c,d removed. We also apply as a benchmark classes “a,b,f,g” at 25% sequence identity filter. Classes a,b,c,f,g were tested following similarity removal at an extremely low level (10%).

Viral properties and classes:

Virus-Host pairs: Acquired from SWP. This set includes all viral proteins partitioned by the kingdom of the hosts. Redundancy filtration (at 40% identity) was performed on the viral proteins but not on the hosts.

Capsids: Compilation of two sets of all viral capsid proteins annotated by SWP: (i) Classes according to host type. (ii) Classes according to viral replication mode.

Neuropeptides

Neuropeptide precursors (NPPs): Proteins annotated as ‘neuropeptide’ were acquired from SwissProt (SWP) and UniRef90 as a positive set. We removed proteins containing the terms ‘fragment’ and ‘receptor’.

a. In the original NeuroPID dataset, multiple non-overlapping negative sets were constructed by randomly sampling proteins with the same distribution of lengths. Multiple additional sets were constructed seperately, see results, & original NeuroPID article (Ofer D et al., 2014) for details.

b. In the V2 (ProFET) NPP dataset: we used as our negative set proteins containing Signal peptides and lacking a validated TMD. We keep all such sequences with the
same (atypical) range of lengths as the NPPs. The negatives were filtered for 10% identity. The dataset held 2,309 negatives and 1,269 NPPs. Note that we expect many unidentified NPP peptides amongst the “negatives”.

3.2 Features

All ProFET features are extracted directly from the protein sequence, and do not require external input (Saeys et al., 2007). NeuroPID used a subset of the features implemented in ProFET. ProFET and NeuroPID were implemented using Python (Rossum and Drake, 2010). Users can easily add additional features. ProFET also generates a predefined set of default features for consistency in evaluation and ease of use, callable from the command-line.

Many of the features can be restricted to a segment of a protein (e.g., the first third of a sequence). The activation of global features combined with segmental consideration is motivated by the atypical composition of different segments in numerous proteins, e.g. signal peptides (Petersen et al., 2011), flexible N-terminal linker regions (George and Heringa, 2002), transmembrane domains (Käll et al., 2007), disordered regions (Linding et al., 2003; Deng et al., 2015) and more. Sequence similarity, as measured by PSI-BLAST was extracted for some experimental comparisons, but was not implemented in our framework.

The features extractable by ProFET include:

Biophysical properties:

Many of these properties were derived from the Expasy proteomics collection (Tools et al., 2010). The important of these elementary global features has been previously validated (Varshavsky et al., 2007). All of these features are included in the NeuroPID framework.

i. Molecular weight (in Da).
ii. Sequence length (in AA).
iii. pH (I), the isoelectric point (Hobohm and Sander, 1995).
iv. Net Charge at various pH (I)s.
v. Aromaticity, the relative frequency of Phe, Trp, Tyr.
vi. Instability index, an estimate for in-vitro protein stability (Wilkins and Williams, 1997).
vii. GRAVY (Grand Average of Hydropathy), the sum of hydropathy values of all AA, divided by the number of AA in the sequence (Kyte and Doolittle, 1982).
viii. Aliphatic index, the relative volume occupied by aliphatic side chains (Ala, Val, Ile and Leu) (Ikai, 1980; Wilkins and Williams, 1997).

**Letter-based features:**

i. AA compositional frequency. i.e we counted each amino acid for each sequence, then divide counts by length. This is equivalent to Term-Frequency (TF) features, or a K-mer of size K=1 (see below).

ii. Overlapping K-mers (Kuang et al., 2005; Leslie et al., 2002). K-mers are combinations of K letters, counted in overlapping windows within the sequence.

iii. “Mirror” K-mers. This accounts for symmetrically equivalent K-mers. For example, Lysine-Arginine (KR) is counted together with RK.

iv. Reduced amino acid alphabets. Grouping AAs according to shared physicochemical properties secures compact representations. We used handpicked reduced alphabets from the literature (Murphy et al., 2000; Etchebest et al., 2007; Weathers et al., 2004; Peterson et al., 2009; Susko and Roger, 2007) and novel alphabet representations of our own (e.g. Ofer_14 and Ofer_8). For the 14 AA alphabet the following are grouped together: KR, TS and LIVM. For the 8 AA representation the grouping is for FYW, ALIVM, RKH, DE and STNQ. The other AA remain uncompressed. This was combined with the various K-mer and the various segmental composition features. E.g. the frequency of 3 letter alphabet 4-mers, grouped as {K,R}, {P}, {The remaining letters}, \(3^4 = 81\) features.

**Local potential features:**

i. Potential post-translational modification (PTM) sites. We included short motifs implemented as regular expressions, including those for “known short motif“ dibasic cleavage model (X-X-Lys-[Lys or Arg], X-X-Arg-Arg, Arg-X-X-[Lys or Arg]; where X denotes any AA (Southey, et al., 2006; Veenstra, 2000). Others include potential N-glycosylation, Asp or Asn hydroxylation motifs and a Cys spacer motif that captures the tendency of Cys to appear in a minimal window (Naamati et al., 2009).

ii. Potential Disorder (FoldIndex). Local disorder is predicted using the naive FoldIndex (Prilusky et al., 2005) and TOP-IDP methods (Campen et al., 2008). FoldIndex predicts disorder as a function of hydrophobic potential and net charge.
Information based statistics:

These features aim to capture the non-random distribution of AA in the sequence, based on the concept of information entropy. For details, see section 2.3, (C) in the original paper (Ofer D et al., 2014).

The information-based features used are:

i. Amino acid entropy. This set of features captures a property how non-randomly distributed each amino acid is in the sequence, based on the concept of molecular information entropy. (Schneider, 2010; Ofer D et al., 2014)

ii. Autocorrelation. This

ii. Binary autocorrelation & entropy. Selected letter(s), for example, K, R and C are denoted as “1” and the rest as “0”. Lag and entropy are then computed for this new representation.

Amino acid propensity-scales:

Amino acid (AA) propensity scales map each AA to a quantitative value that represents physicochemical or biochemical properties, such as hydropathicity, secondary structure propensity (Jr and Fasman, 1989) or size. Thus, dissimilar AA (for a given property or statistical propensity) will have more dissimilar values. These scales can then be used to represent the protein sequence as a time series, typically using sliding windows of different sizes. ProFET includes a wide array of scales, ranging from the established propensities for hydrophobicity and flexibility/B-factors (acquired from Expasy (Tools et al., 2010)), to independent derived scales (Atchley et al., 2005; Georgiev, 2009).

Features derived from these scales include:

i. Averages for the sequence as a whole, for different window sizes or segments.

ii. Quartile averages (e.g., average of the top, bottom 25%).

iii. Maximum and minimum values for a given scale and window-size along the entire sequence.

iv. Autocorrelation.

Transformed CTD features:

We implemented multiple CTD (Composition, Transition, Distribution) features, some based on Dubchak (Dubchak et al., 1995) and PROFEAT (Li et al., 2006) as well as novel ones based on AA propensity scale ranges and reduced alphabets. These 3 letter groupings included:
hydrophobicity, normalized Van der Waals volume, polarity, polarizability, charge, secondary structure and solvent accessibility hydrophobicity, normalized Van der Waals volume, polarity, polarizability, charge, secondary structure and solvent accessibility propensity. An additional subdivision of disorder propensity was adapted from Composition Profiler (Vacic et al., 2007): 1:'ARSQEGKP', 2:'ILNCFYVW', 3:'DHMT'.

These features are:
i. Composition (C) is the number of AA in a grouping divided by the total number of AA.
ii. Transition (T) is the number of transitions from a particular property to different property, divided by (total number of AA − 1).
iii. Distribution (D) captures is the chain length within which the first 25%, 50%, 75% and 100% amino acids of a particular property are located.

3.3 Evaluation

As mentioned previously, for each dataset performance was measured using multiple rounds of stratified cross validation or stratified shuffle-split CV (a bootstrapping with replacement variant). Multiple metrics were used and recorded, with an emphasis on metrics that are robust in the case of imbalanced classes, e.g. F1, ROC-AUC (area under the Receiving Operator Characteristic curve), see Figure 5.

Hyperparameters were tuned solely according to the training data’s CV performance, without exposure to the test set. Feature selection (where done) was performed independently by each training fold. For the exact details on the chosen models and parameters, I refer readers to the original articles (Ofer and Linial, 2015; Ofer D et al., 2014).

\[
\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \\
\text{Precision} = \frac{TP}{TP+FP} \\
\text{Recall (sensitivity)} = \frac{TP}{TP+FN}
\]

Figure 5 Metrics
3.4 Machine Learning Methods

A large variety of machine learning methods were tested. The best performing methods included SVMs (with RBF kernels), Random Forests, Gradient boosting trees, Logistic Regression and ensembles of different models. All were implemented with Scikit-Learn (Pedregosa et al., 2011).

3.5 Feature selection (RF-RFECV)

Interpretability of the features that best contributed to the performance is a crucial knowledge. Several methods for feature selection can be applied to identify a minimal set of such features, or to remove “noisy” features. In addition to classic statistical filters (e.g. FDR corrected F-test), we applied a novel method of my own devising I named RF-RFECV. It consists of combining Random Forests (an ensemble of decision tree classifiers, where each component has only a subset of the features) with the Random Feature Elimination (RFE) wrapper method. In each iterations, the weakest features are removed, and the model is then retrained with the remaining features, until the preselected desired amount of features remains. Performance of the reduced feature set is measured using new splits of the training data and cross validation. This is then repeated. The existing RFE implementations typically used an SVM or linear learners and not ensembles or random subspace based learners. This method was applied for “aggressive” post-hoc feature selection, with the goal of extracting interpretable features from the hundreds of weak, possibly redundant features. This method combined the powerful modelling capacity of RF with the RFE method, giving a tiny subset of powerful, non-linear features within a short time. We applied it to the various datasets with excellent results (Ofer and Linial, 2015).

3.6 Software and implementation

The framework was implemented using Python (Rossum and Drake, 2010). Sci-Kit Learn (Pedregosa et al., 2011) was used for ML and statistical analysis. Additional libraries included BioPython, NumPy, SciPy and Pandas (Cock et al., 2009; Oliphant, 2007; McKinney, 2011). The code is freely available online: https://github.com/ddofer/ProFET, http://www.protonet.cs.huji.ac.il/neuropid/
4 Results

4.1 Platforms

We developed Neuropeptide Precursor Identifier (NeuroPID), a platform to automatically identify neuropeptide precursors in metazoan genomes, and through them, their secreted bioactive peptide products. The NeuroPID platform was successfully applied to multiple genomes, including insects, Cnidarians (Appendix) and Octopus Vulgaris (Ofer D et al., 2014), identifying hundreds of high confidence candidates. Furthermore, we expanded the tool into a freely available, easy to use online webserver, capable of easily handling tens of thousands of sequences within minutes and annotating whole genomes.

ProFET (Protein Feature Engineering Toolkit) expanded our systematic approach into a comprehensive platform for all possible proteins. It builds on NeuroPID’s underlying framework with new methods and feature extraction techniques, as well as a comprehensive set of routines for managing the data (e.g. raw sequences, annotations, extracted features and predictions). It ranks among the state of the art for rapidly classifying high-level novel functionality in proteins, without external databases or alignment.

ProFET serves as a platform for machine learning and quantitative sequence-derived feature extraction from proteins. This did not exist before, with various scientists constantly reinventing the wheel, rewriting “glue code”, and only looking at “one-off” cases, rather than viewing this as a paradigm. In addition to utility, we provided performance, with a wealth of universal, powerful features that can be extracted from any sequence, without needing prior annotation, homologs or structural knowledge. This is notable, as a large number of existing tools are unsuited for large scale predictions or work with novel proteins due to their requirements (Saraç et al., 2010). We thus created a universal methodology for machine learning with whole protein sequences.
4.2 NeuroPID Results

The performance of the NeuroPID platform and datasets is presented here in brief, in Figure 6 and Table 1. For full details, I refer readers to the original article, included in the appendix.

Figure 6 Cross validation performance of different models on NeuroPID’s full, original dataset (NPP V1). Numerous models support the high accuracy and precision in the CV tests. Result variations measured by repeated tests were negligible. RBF = SVM with Radial Basis Function Kernel. SVC = SVM
To assess the robustness of NeuroPID we repeated the CV procedure with models trained on alternative “negative” sets (the positive set remained unchanged). We performed it for “negative” sets of secretory proteome (TMD containing proteins) and charged nuclear proteins. The rationale behind the selection of the later set was to provide a “challenging” set in which the inherent appearance of basic residues prevails (e.g., histones, transcription factors). CV performance was very high, even exceeding the original results. We did not use these alternative datasets in the final platform, for reasons that will be discussed later.

| ML model | Metric \ Negative Set | Nuclear proteins | + SP +TMD |
|----------|-----------------------|------------------|-----------|
| Random Forest | Accuracy | 0.928 | 0.967 |
|          | Precision | 0.938 | 0.916 |
|          | Recall | 0.915 | 0.891 |
|          | Area under ROC curve | 0.857 | 0.937 |

Table 2 NeuroPID Alternative negative sets performance. Performance is for a Random Forest classifier and original positive training set. SP+TMD = Swissprot proteins annotated as having a transmembrane domain and a signal peptide. Both sets were filtered for redundancy.
1.2.2 NeuroPID: Taxonomical Evaluation

Once the ML was trained and tested, we examined unseen examples derived from complete proteomes. We defined the task as identifying unseen NPPs from specific Taxa (Chordata and Arthropods). Only proteins belonging to the taxon were used to construct the negative and positive data. Only representative sequences from UniRef90 clusters were used. The results are summarized in Error! Reference source not found. The high performance of taxon specific models shows that restricting predictions and data to specific taxa typically increases performance (compared to

| ML model\Metric  | Accuracy | Precision | Recall | MCC  | AUC  |
|------------------|----------|-----------|--------|------|------|
| Random Forest    | 0.928    | 0.938     | 0.915  | 0.857| 0.928|
| Gradient Boosting| 0.93     | 0.928     | 0.929  | 0.859| 0.93 |
| Linear SVM       | 0.881    | 0.87      | 0.894  | 0.763| 0.882|
| SVM-RBF          | 0.928    | 0.918     | 0.939  | 0.857| 0.929|

Table 1), i.e. is easier, despite the reduction in data samples.

|                  | Chordata             | Arthropods            |
|------------------|----------------------|-----------------------|
|                  | SVM/Gradient Boosting| SVM/Gradient Boosting |
| Mean Accuracy    | 0.746 / 0.907        | 0.900 / 0.923         |
| Mean Precision   | 0.906 / 0.923        | 0.968 / 0.941         |
| Mean Recall      | 0.621 / 0.914        | 0.889 / 0.952         |
| Mean AUC         | 0.768 / 0.906        | 0.908 / 0.901         |

Table 3 NP classification CV with taxonomical partitions. SVM = Support Vector Machine with RBF kernel. Gradient Boosting = Ensemble of Gradient Boosting Trees.
1.2.3 NeuroPeptide Statistics

As part of our analysis, we did an exploratory analysis of the biophysical features’ distributions in the set of known Neuropeptides vs the background distribution, consisting of all (UniRef50 clustered) proteins with roughly the same range of lengths (under 900 AA long) as the NP. These results confirmed the atypicality of NPs properties, and the validity of our basic features prior to further work (i.e adding more features or model fitting). Results appear in Appendix 3. We performed a similar analysis on the final set of training data (the final positive set, and the randomly sampled negative set), some significant features are listed in Figure 6. Figure 7 Features’ statistical significance for NPs vs the background (UniRef50) distribution for: (A) amino acid frequency. (B) Several biophysical properties. Most features had statistically significant different values (Two sided t-test, \( \alpha=0.01 \); KS test< 0.01). The most individually significant feature is the occurrence of lysine (K, p-value \(<1.0e-83\)).

1.2.4 NeuroPID Webserver

As part of our efforts to ensure the use of our work by the wider scientific community, we created an online webserver for NP and NPP discovery. The NeuroPID website is available at http://neuropid.cs.huji.ac.il. For full details, I refer to the article (Karsenty et al., 2014), included in the Appendices.
In brief, the webtool can handle genomic scale sequence inputs, and outputs predicted Neuropeptide sequences. Analysis of the features, likelihood of the results, and filtering (e.g. “Display only high confidence predictions, from input sequences with a predicted signal peptide”) is also provided. Links to other websites are provided for further analysis (e.g. NeuroPred (Southey, Amare, et al., 2006; Southey et al., 2008), for predicting cleaved NPs). An illustration of some features provided by the website is presented in Figure 8.

Reference source not found. :
Figure 8 Screenshots of the NeuroPID Website (1) A summary table for an input of 4454 proteins from Xenopus laevis. A pie chart displays the distribution of prediction according to the agreement of the 4 classifiers. (2) A detailed table shows the confidence for each prediction methods. The red and green fonts indicate negative and positive predictions, respectively. The table is ranked by the Internal Score (IS). Results are linked to their FASTA sequence and to knowledge based resources (ProtoNet, UniProtKB) and analysis tools (SignalP, NeuroPred). (3) Results from the feature histogram. The position of a selected protein (dash line) in view of the distribution of the positive and negative sets is shown. The basis distributions are shown in green and red colors to indicate positive and negative instances, respectively. Data is normalized by the distributions standard deviations (denoted Feature Units, X-axis). (4) Sequence graphics with positions of basic residues. The red vertical lines show the location of dibasic and tribasic residues. The presence of single basic residues is colored green.

1.2.5 Neuropeptide Taxonomic Validation

NeuroPID was tested with several organisms whose complete proteomes were acquired from UniProt and SwissProt (ranging from 10K for A. mellifera to 23K for C. elegans). Predictions using just a single model resulted in a large number of weak predictions (ranging from 1.3K to 5.2K candidate NPPs). Including a more stringent probability threshold lead to a drastic reduction in the output, leaving only the top ranked NPP predictions. For example, using a Random Forest model, a probabilistic/‘certainty’ threshold of 0.99 (i.e. 99% of the ensemble’s component trees gave a positive prediction) reduced the unfiltered predictions for Bombyx mori from over 4000 to only 16 positive predictions (and 819 negative predictions). Similarly, the application of Gradient Boosting (GBR) for the Monarch butterfly (16183 sequences, 4856 predictions) reduced the number of predictions 10 fold at a 99.5% threshold. Experimental validation will always be the bottleneck, so probabilistic filtration for high-confidence candidates, with a high level of precision, is of obvious use.

Table 4 summarizes the results of NeuroPID predictions in view of annotated NPPs from UniProtKB (partitioned by SW and UniProt). The number of NeuroPID top predictions for most organisms matched the number and identity of curated SW identifications. In all cases, the prediction list matched the list of curated NPPs as compiled in SW (Table 4).
The NeuroPID prediction protocol for individual species is summarized in Table 4. Predicted using SVM with RBF (Radial Basis Function) kernel. SP, Signal peptide predicted according to SignalP 4.0.

| organism / taxa      | # of UniProt (UniRef90) | # of NPP in SW (UniRef90) | # of NPP in UniProt (UniRef90) | Prediction NeuroPID RBF$^a$ |
|----------------------|--------------------------|---------------------------|-------------------------------|----------------------------|
| Apis Mellifera       | 10394                    | 6                         | 19                            | 7                          |
| SP in Apis Mellifera$^b$ | 2139                     | 5                         |                               | 7                          |
| Gallus gallus        | 20760                    | 5                         | 5                             | 5                          |
| SP in Gallus gallus  | 701                      | 1                         |                               | 1                          |
| Bombyx mori          | 15250                    | 5                         | 17                            | 9                          |
| SP in Bombyx mori    | 112                      | 5                         | 5                             | 9                          |
| Octopoda             | 224                      | 4                         | 4                             | 4                          |
| SP in Octopoda       | 76                       | 3                         |                               | 3                          |

Table 4 NeuroPID predictions for individual species. Predicted using SVM with RBF (Radial Basis Function) kernel. SP, Signal peptide predicted according to SignalP 4.0.

1.2.6 NeuroPID Discovery protocol

The NPP and NP prediction protocol for NeuroPID is summarized in Figure 9.
NeuroPID was used to identify novel NPPs. In each case, the organism’s annotated NPPs were excluded from the model training phase. The NeuroPID results for NPPs from worm, bee, ant and octopus are found in www.protonet.cs.huji.ac.il/neuropid/results/. We analyzed the results for the Apis mellifera (Honey bee) and Thaumeledone gunteri (Octopoda) proteomes. Table 4 shows the results and the consistancy among the applied methodologies. Consistancy among the
decision tree classifiers (RF, GBR) was especially high. Using all 4 mentioned models, we identified multiple sequences supported by at least 3 of the predictors. (Table 5).

Table 5 NeuroPID predictions for honey bee (Apis) and octopus (Gunteri) proteomes. The number of predictions is shown (nodes) along the % of overlap for the prediction pairs. The % is calculated from the smaller number of predictions in the pair. Random Forest (RF), Extra Trees classifier (Ext-Tree), SVM-SVC (linear kernel) and Gradient boosting trees (GBR).

As an illustration, one of these sequence that was consistently identified by the different ML methods was an uncharacterized sequence from *Apis mellifera* (UniProt: H9K152, 183 amino acids). We propose this sequence to be an overlooked NPP: (i) It is a secreted protein (+SP). (ii) Using MS identification, a 52 amino acid peptide belonging to the full length ORF was identified in the honeybee brain extract (Audsley and Weaver, 2006). (iii) Pfam (Finn et al., 2014) identifies the CRF (Corticotropin-releasing factor) domain, associated with active peptides from frog, fish and mammals. This domain is postulated in stress and anxiety, osmolarity, thermoregulation, growth and metabolism in mammals (Chang and Hsu, 2004).

As part of further ongoing research into Cnidarian “Neuropeptide like peptide-hormones”, we extracted a large number of high confidence predictions using the NeuroPID website. We extracted multiple Cnidarian proteomes and predicted ORFs, and ran them through a preliminary pipeline, then NeuroPID. The initial pipeline consisted of using a number of bioinformatics tools to detect “necessary but not sufficient” properties, essential (but not exclusive) to secreted proteins or NPPs: the presence of a signal peptide (predicted using SignalP (Petersen et al., 2011) and Phobius (Käll et al., 2007)) and the lack of a transmembrane or intermembrane region.
(detected using Phobius). These secretome candidates were then run though the NeuroPID discovery protocol. The top ranked candidates for each proteome were then manually analyzed and annotated for future experimental validation. The list of high quality predictions is provided in Appendix 4. The manual annotations are unpublished and available upon request.

4.3 ProFET Results

ProFET can be used as a baseline for features and model training, or it can be customized with new or specific features. ML models can be trained and saved according to extracted features and annotations. For details about the framework, test-cases and in-depth analyses, we refer to the original article (Ofer and Linial, 2015), included in the appendix.

1.2.7 ProFET Benchmarks

Each set was measured using 15 iteration randomized stratified shuffle-split CV. For each iteration, a fraction of the data (18%) is randomly set apart as an evaluation set. The framework automatically selects the best performing classifier. The term “Dummy” applies to a classifier that always picks the majority class. Altogether, we present 17 datasets. For 76% of the datasets accuracy and F1 Scores are above 80%, while for 35% the accuracy is >90% (Figure 10).
Figure 10 Classification performance by Accuracy and F1 score for 17 datasets using ProFET. Results for Accuracy (Acc, dark blue) and the F1-score (light blue, middle) are shown. Dummy predictor is a default classifier which always predicts the largest class in a dataset (rightmost, pink). Multiple rounds of stratified shuffle-split CV were performed, and the scores were averaged, SD error bars shown.

The classification performance for DNA and RNA binding proteins meets the state of the art results obtained by special purpose predictors (Wang et al., 2010), despite our using a harder (redundancy filtered) version of the data. These specialized predictors for DNA and RNA binding proteins rely on evolutionary information (e.g., PSSMs), requiring computationally intensive alignment of candidate sequences using PSI-BLAST (both for training and inference).

On the original, easier dataset, the DNAbinder method reported only 61.42% accuracy and 63.5% using PSSM profiles (Kumar et al., 2007). Our platform reached 72% and 79% accuracy for DNA and RNA binding proteins, respectively, despite not using evolutionary information. Five of the benchmarks concern structural SCOP datasets, at the class or fold level. We did not attempt to predict secondary structure directly, although this can easily be integrated (e.g. as CTD features), and indeed we do not aim to replace dedicated structural predictors.

Naturally, classification success varies by task. For example, the success for the SCOP ‘selected class’ is very high (0.82–0.9), whereas the performance for the fold classification is much lower (0.62–0.65). In the large SCOP datasets, we added reduced alphabet (Ofer_14) 3-mer frequencies to the features. Note that SCOP 25% and SCOP 10% tasks use the same dataset (SCOPe).
differing only by the degree of redundancy removal. We conclude that this excellent performance shows the applicability of the “default” ProFET workflow and features.

1.2.8 ProFET Case studies

We selected three datasets for a more detailed analysis. Thermophiles, NPP V2 and Uncultured bacteria. It bears mentioning that the NPP V2 dataset is more challenging than that using in NeuroPID, due to the more challenging negative set consisting of secreted, processed proteins. The uncultured bacteria ‘metagenomic’ dataset is also challenging given how poorly characterized its progenitors are.

For all three sets, we obtained exceptional classification. Results were derived from K-fold stratified CV (without shuffle-split), no feature selection. Results for the Thermophiles and NPP V2 are shown below in Figure 11. Performance for the uncultured bacteria localization task was also convincing (tested via 12 rounds of stratified shuffle split CV): F1 score was 0.917 (+ 0.01 SD), Accuracy 0.916 (+ 0.01 SD).

![Confusion matrix](image1)

Figure 11. Test cases performance using ProFET. (A) Confusion matrix. Results derived from 10-fold stratified CV. The number of FP and FN is shown for thermophiles (left) and NPs (right). (B) AUC (area under receiver operating characteristics curve) for both datasets.

1.2.9 Sequence Alignment Comparison

We compared the most popular sequence alignment method available (PSI-BLAST (Altschul et al., 1997)) to our feature driven approach. The poor performance of sequence alignment methods
on structural classes has already been documented (Shah et al., 2008; Ding and Dubchak, 2001), so we chose to compare our novel test cases. This comparison was artificially biased in favor of PSI-BLAST, as we limited its predictions to the training data, rather than the whole protein universe (making the task far easier).

We used PSI-Blast (3 iterations, default parameters) on the Thermophile and NPP V2 sets. The most significant E-value for each sequence was used as an approximate distance matrix. We then trained a K-nearest-neighbors classifier, tested hyperparameters and recorded the best performance. We also used a clustering approach (Spectral clustering, K-means (Yu and Shi, 2003; Moore, 2004)), and compared the clusters to the true labels.

Clustering performance was significant lower than our own (Figure 11). The best results for the Psi-Blast test were obtained from Spectral clustering model. The F1 score was 0.56 and 0.29 for the NPP and Thermophile sets respectively. To make sure the poor performance was not dependent on the choice of ML methodology, we repeated the analysis as a supervised classification task using a K-nearest neighbors classifier (k=1). The data was split 80/20 into evaluation and hold-out sets, and the best hyperparameters were determined by 4 fold CV. For the NPP and Thermophiles sets the accuracy on the ‘evaluation set’ were 62.8% (+/0.16 SD) and 48.9% (+/-0.03 SD), respectively. The F1-score was 0.61 and 0.44 respectively. Clearly, pure sequence alignment is inferior to our feature driven approach.

1.2.10 ProFET: Feature Selection

Using the RF-RFECV method, we analyzed the top 15 features for the NPP V2 and Thermophile datasets, shown in Figure . The initial set included 771 features.

**Thermophiles informative features:**

We note the importance of AA composition, particularly charged and polar amino acid groups. Of notable importance are features involving the organizational entropy of E and Q. The relevance of these AA’s organizational entropy in thermophiles had been established (Michelitsch and Weissman, 2000), but I was not aware of this prior to this post-hoc analysis, so this insight was obtained purely from the feature selection. Merely using the AA composition would not have captured many of these features.
The CV performance (F1 score) with just 15 features reached 99.53% of that obtained using all statistically significant features (F1 score=0.906; 453 features).

**NPPs informative features:**

As opposed to the features dominating the test case of thermophilic proteins, in the case of the NPPs, a smaller set of features dominates the classification, mainly relating to the normalized frequency of putative dibasic cleavage sites. Further properties of the basic residues Lys (K) and Arg (R) repeat themselves by virtue of entropy, binary autocorrelation (6/15 features) and more. Additional features include protein size (Mw and length) and to a lesser extent some pseudo-structural propensities, such as the B-factor flexibility (‘Flex_min’), and secondary structural propensities – reflecting the importance of availability of the putative cleavage sites and atypical composition of the putative peptides. Cysteine (C) importance is expected, given the importance of cysteine cross linkages in short secreted peptides and structurally similar neurotoxins, discussed in ClanTox and our previous work (Tirosh et al., 2013).

The CV performance (F1 score) with just 15 features was 95.85% of that obtained using all statistically significant features (0.945 (± 0.01); 544 features).

![Figure 12](image-url)

**Figure 12** Top 15 features for classification of: (A) Thermophilic vs Mesophilic proteins and (B) NPPs vs non-NPP secreted proteins. Y Axis shows relative feature importance in the model. AA – Amino Acid.

Trimmed average: AA Propensity scale average for a window with the bottom and top 20% extremes removed. ‘Scale Name grouping’: ’Composition 1’ – CTD (C) Feature for a given compressed grouping, and one of its 3 subsets. Ofer_w8: 8 letter reduced alphabet grouping.
5.1 ProFET Perspectives

Several conclusions can be drawn from our results:

A. Protein centric analysis: The alignment-free feature based methods should be considered a baseline approach for whole proteins, rather than protein domains. Most of our knowledge from 3D structure and evolution relies on the properties of domains within proteins. We propose the feature-centric approach as complementary to the alignment or domain-centric one.

B. “One size fits all”: Features in ProFET are highly relevant to a broad range of proteins and classification tasks. This is in contrast to task-specific methods. Therefore, ProFET eliminates duplicated efforts for feature extraction.

C. Flexibility of use: Our presented pipeline accepts a single sequence, combined files, multiple files or a directory. It automatically labels the input into classes (if desired), and transforms the features. Users can set the desired combination of features from ProFET. From the point of view of the user, several consideration were taken:

- Our pipeline handles FASTA files and stores them, annotations, features and predictions.
- We use state of art, free python data science tools (such as Pandas, scikit-learn, biopython)
- Easy to add new features using a standardized format.
- Our framework includes feature details in the pipeline so results are easily interpretable.
- Our code is freely available.

We provide a large collated resource for protein features. Thanks to the modular design of ProFET, adding and tinkering with features is trivial. Users of ProFET can decide to focus, remove or expand any subset of the generated features.

In summary, the approach presented here is suitable and powerful for application towards modern approach for ML especially in the emerging field of Deep Learning, and unsupervised
learning of feature representations. These features can be easily experimented with, allowing additional applications of biological insight to the task of feature engineering.

5.2 Pitfalls & Tricks

When constructing a dataset for classification, definition of the classes is essential, both from a methodological and practical POV. Biology poses a number of challenges to defining data:

1. Ambiguous definitions.
2. Missing, unlabeled, noisy and missing data.
3. Imbalanced classes.
4. Picking representative sets

These are inherent to virtually any biological problem, and there is no simple solution to any of them. At best, I would note a number of hard-won guidelines and tricks.

1) Care should be taken when using predefined keywords. For example, we found that NP receptors were being included in our initial data, based on the NP keyword. Structured ontologies such as GO (Ashburner et al., 2000) can help, as can gold standard datasets from the literature and the use of manually reviewed data when constructing the initial training set.

2) As in any problem, raw data must be carefully examined then cleaned, using automatic methods. E.g. non-standard amino acids, or non-standard modifications to formats.

3, 4) Even the best models will have difficulty when the disparity between groups is extreme. When it’s 50,000:1 (1,100 NPPs vs ~56 million), even more so. In our initial work we addressed this using lengthwise stratified sampling generate a balanced negative set. We noted that this gave better generalization performance than supermajority sets, even when repeated multiple times. A second trick is stringent redundancy removal on the majority class – this maximizes diversity, albeit at the cost of making the task more difficult (an acceptable compromise). A further improvement utilized in the NPP V2 dataset was to “prefilter” our negative set using biological knowledge: Rather than sampling from the (100%) whole protein universe, we sampled from among the proteins most like our positive class (+SP,-TMD and same range of lengths). Given there are only 3.4 million such proteins, we sample from just ~8% of the proteome. This gives a set which is far more similar to our “positives”, but one more representative of the input we expect from real world users. We did not even need to sample
randomly from the negative set to get a “balanced” ratio of samples. Making the negative set “harder” is a common theme in our work, with the goal of ensuring the usefulness of our models. I note that care must be taken when looking at an unknown background for “negatives” (Charles Elkan), as many unlabeled sequences could be positives. Even more so, non-representative selections are hazardous: In NeuroPID we used proteins with SP and TMD: Proteins with SP meant we were looking at just ~10% of the proteome, and proteins with a TMD represented a large and varied set, that could not be NPs. CV performance was very high, but later examination on hold-out proteins revealed that the classifier had not learned to identify between Neuropeptides and non-NPs, but rather to identify proteins with a TMD! (i.e the TMD was too easy and distinct a feature).

5.3 ML and Predictive Proteomics

As noted, the main drawbacks in existing sequence-based methods are (i) Some functions cannot be detected by sequence-based methods; (ii) Current statistical models mostly capture local patterns rather than high-level function and (iii) rare sequences or properties with few homologs cannot be successfully used for inference or construction of good statistical model via MSA. Furthermore, inference on new sequences is growing ever more computationally costly, since sequence alignment methods rely on all vs all searching. Even the best clustering methods can only do so much to reduce the search space from tens of millions of sequences, to a few million (Rappoport and Linial, 2013; Hauser et al., 2013), and this problem is only growing. Alignment free features allow rapid inference for new samples, without needing to search anew through the existing data. This is in addition to their greater flexibility, ability to learn high level features and correlations, and the ability to integrate multiple levels of features and information. However, such ML approaches are still best suited to certain use cases, such as those raised here, e.g. high level function, cases lacking structural or sequence similarity, or searching proteins for a specific novel function. However, in the “naïve” use case of annotating highly similar sequences, “guilt by association” and alignment is likely to remain the method of choice for the foreseeable future, due to its precision in the cases where it works best. Other use cases are a different matter however, and an ever increasing number of computational methods integrate ML in their approaches (Radivojac et al., 2013; Kryshtafovych et al., 2014), albeit in a huge variety
of ways, and variety of feature representations (e.g. most integrate PSSM profiles, or existing ontological annotations rather than sequence-derived alignment free features).

Deep learning

So called “Deep learning” ML methods are variants of artificial neural networks (NN) with multiple hidden layers and domain-specific architectures (such as convolutional NN or recurrent NNs (RNN)). These have attained state of the art performance in a wide range of ML domains in recent years, notably in computer vision and natural language processing (Hochreiter and Schmidhuber, 1997; Xu et al., 2015; Kim, 2014; Krizhevsky et al., 2012; Simonyan and Zisserman, 2014; LeCun et al., 2015). It is likely that the next step in improve computational protein function and family prediction will use these powerful models, although so far most have been used only for residue level prediction (Lyons et al., 2014; Zhou and Troyanskaya, 2014; Qi et al., 2012), rather than whole sequences (Hochreiter et al., 2007). This stems mainly from two problems: 1) the difficulty of representing whole sequences in a fixed length form (Nanni et al., 2014). This could be overcome using RNNs, distributed representations (in the manner of Word2Vec) (Le and Mikolov, 2014; Goldberg and Levy, 2014; Asgari and Mofrad, 2015) or transformation of aa propensity scale representations into a fixed length time series. Little of this has been done, although I am currently pursuing this as part of future research. 2) The ability of NNs to universally approximate any function is also their Achilles heel: being very high variance methods, they require a large amount of training samples and are vulnerable to overfitting (Geman et al., 1992). Other lower variance ML methods may be less flexible, but are also less prone to overfitting, while statistical methods such as HMMs or Conditional Random fields can learn more efficiently than “vanilla” NNs. In the domain of computer vision, a successful solution is transfer learning (Yosinski et al., 2014; Sharif et al., 2014): Given a specific domain, train a large model and learned feature representation (e.g. via convolutions or embeddings (Mikolov et al., 2013)) on a challenging task with universal commonalities and a large diverse dataset (e.g. Imagenet). The feature representations may then be used to extract features in other tasks. This does not currently exist in Bioinformatics, where both data and class definitions are hard to define and gather. I do propose that such a transfer learning model for some tasks reliant on structural homology could be created using the large protein family datasets gather in SCOP, CATH or PFAM, combined with our own set of alignment free
features, autoencoders (Vincent et al., 2008; Xu et al., 2011), distributed representations of proteins as sentences (Le and Mikolov, 2014), and convolutional RNNs (Sønderby et al., 2015), but this is beyond the scope of this work. That said, we experimented with deep learning (multi-layer perceptrons) during this work and did not obtain significantly better results (unpublished). An oft ignored aspect is one of cost-benefit: methods such as RF are very fast, handle class imbalance gracefully, are robust to noisy features and high features dimensionality, do not need special input scaling, allow feature selection, and are very easy to use with few hyperparameters. This made them a far more practical choice, even more so given the varied, imbalanced datasets we faced.

5.4 Overall Conclusions

A curated set of best practices is essential to preventing scientists from continually “reinventing the wheel” (Ofer and Linial, 2015; Li et al., 2006). This would lower the barrier of entry to new researchers, allowing an influx of novel, interdisciplinary ideas from domain experts (e.g. experts on molecular toxicology). A predefined set of good, “generic” features is beneficial to getting a “good enough” baseline for different tasks, as shown in ProFET. ProFET’s performance was assessed rigorously and reached state of the art results in the vast majority of tasks. Our “default” set of features performed well in all the tasks, even when compared to existing, specialized predictors. I introduced many novel features. For example, features based on reduced alphabets (e.g. K-mers, CTD), entropy, high performance AA scales and their usage, binary autocorrelation, sequence segmentation, mirror k-mers and more. Many of these features not only improved performance while allowing a compact representation, and exposing important biologically interpretable properties (Figure ).

Profet is flexible, universally applicable to any protein sequence, was rigorously tested, is easy to use, and has state of the art “out of the box” performance.

Our Neuropeptide identification platform is the first of its kind, and offers excellent performance and scalability. Its availability as a webtool makes it a convenient tool for research into new neuropeptides. We discovered many putative NPs, and hope to enable future discoveries into peptide regulators of behavior and metabolism based on the platform and discovered NPs.
6 BIBLIOGRAPHY

Aebersold,R. and Mann,M. (2003) Mass spectrometry-based proteomics. *Nature*, **422**, 198–207.
Akhtar,M.N. *et al.* (2012) Evaluation of database search programs for accurate detection of neuropeptides in tandem mass spectrometry experiments. *J. Proteome Res.*, **11**, 6044–55.
Alberts,B. *et al.* (2009) *Essential Cell Biology*.
Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.
Amare,A. *et al.* (2006) Bridging neuropeptidomics and genomics with bioinformatics: Prediction of mammalian neuropeptide prohormone processing. *J. Proteome Res.*, **5**, 1162–7.
Amare,A. and Sweedler,J. V (2007) Neuropeptide precursors in Tribolium castaneum. *Peptides*, **28**, 1282–91.
An Empirical Comparison of Supervised Ensemble Learning Approaches
Andreeva,A. *et al.* (2014) SCOP2 prototype: a new approach to protein structure mining. *Nucleic Acids Res.*, **42**, D310–4.
Angel,T.E. *et al.* (2012) Mass spectrometry-based proteomics: existing capabilities and future directions. *Chem. Soc. Rev.*, **41**, 3912–28.
Asgari,E. and Mofrad,M.R.K. (2015) Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics. *PLoS One*, **10**, e0141287.
Ashburner,M. *et al.* (2000) Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, **25**, 25–29.
Atchley,W.R. *et al.* (2005) Solving the protein sequence metric problem. *Proc. Natl. Acad. Sci. U. S. A.*, **102**, 6395–400.
Audsley,N. and Weaver,R.J. (2006) Analysis of peptides in the brain and corpora cardiaca-corpora allata of the honey bee, Apis mellifera using MALDI-TOF mass spectrometry. *Peptides*, **27**, 512–520.
Bacardit,J. *et al.* (2014) Hard Data Analytics Problems Make for Better Data Analysis Algorithms: Bioinformatics as an Example. *Big Data*, **2**, 164–176.
Barla,A. *et al.* (2008) Machine learning methods for predictive proteomics. *Brief. Bioinform.*, **9**, 119–28.
Basic local alignment search tool Altschul,S.F. *et al.* (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.
Ben-Hur,A. *et al.* (2008) Support vector machines and kernels for computational biology. *PLoS Comput. Biol.*, **4**, e1000173.
van den Berg,B.A. *et al.* (2014) SPiCE: a web-based tool for sequence-based protein classification and exploration. *BMC Bioinformatics*, **15**, 93.
Berg,J.M. *et al.* (2002) Protein Structure and Function.
Berman,H.M. *et al.* (2000) The Protein Data Bank. *Nucleic Acids Res.*, **28**, 235–242.
Beshel,J. and Zhong,Y. (2013) Graded encoding of food odor value in the Drosophila brain. *J. Neurosci.*, **33**, 15693–704.
Biegert,A. and Söding,J. (2009) Sequence context-specific profiles for homology searching. *Proc. Natl. Acad. Sci. U. S. A.*, **106**, 3770–5.
Bishop,C.M. (2006) Pattern Recognition and Machine Learning.
Bock,J.R. and Gough,D.A. (2001) Predicting protein–protein interactions from primary structure. *Bioinformatics*, **17**, 455–460.
Bork, P. and Koonin, E. V (1998) Predicting functions from protein sequences--where are the bottlenecks? *Nat. Genet.*, 18, 313–8.

Boutet, E. et al. (2007) UniProtKB/Swiss-Prot: The Manually Annotated Section of the UniProt KnowledgeBase. *Methods Mol Biol.*, 406, 89–112.

Brain, S.D. and Cox, H.M. (2006) Neuropeptides and their receptors: innovative science providing novel therapeutic targets. *Br. J. Pharmacol.*, 147 Suppl, S202–S211.

Breiman, L. (1996) Bagging predictors. *Mach. Learn.*, 24, 123–140.

Breiman, L. (University of C. (1999) Random forest.

Brockmann, A. et al. (2009) Quantitative peptidomics reveal brain peptide signatures of behavior. *Proc. Natl. Acad. Sci. U. S. A.*, 106, 2383–8.

Cai, C. Z. et al. (2003) Protein function classification via support vector machine approach. *Math. Biosci.*, 185, 111–122.

Cai, C.Z.Z. et al. (2003) SVM-Prot: web-based support vector machine software for functional classification of a protein from its primary sequence. *Nucleic Acids Res.*, 31, 3692–3697.

Campen, A. et al. (2008) TOP-IDP-scale: a new amino acid scale measuring propensity for intrinsic disorder. *Protein Pept. Lett.*, 15, 956–63.

Carpenter, E.P. et al. (2008) Overcoming the challenges of membrane protein crystallography. *Curr. Opin. Struct. Biol.*, 18, 581–6.

Chang, C.L. and Hsu, S.Y.T. (2004) Ancient evolution of stress-regulating peptides in vertebrates. *Peptides*, 25, 1681–1688.

Charles Elkan, K.N. Learning Classifiers from Only Positive and Unlabeled Data.

Cheng, J. and Baldi, P. (2006) A machine learning information retrieval approach to protein fold recognition. *Bioinformatics*, 22, 1456–63.

Cheng, J. and Baldi, P. (2007) Improved residue contact prediction using support vector machines and a large feature set. *BMC Bioinformatics*, 8, 113.

Clynen, E. et al. (2010) Bioinformatic approaches to the identification of novel neuropeptide precursors. *Methods Mol. Biol.*, 615, 357–74.

Cock, P.J.A. et al. (2009) Biopython: Freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25, 1422–1423.

Cortes, C. and Vapnik, V. (1995) Support-vector networks. *Mach. Learn.*, 20, 273–297.

Craft, G.E. et al. (2013) Recent advances in quantitative neuropeptidomics. *Methods*, 61, 186–218.

Cristianini, N. (2001) Support Vector and Kernel Machines.

David Kofoed Wind, O.W. (2014) CONCEPTS IN PREDICTIVE MACHINE LEARNING. *Thesis, Tech. Univ. Denmark*.

DeGennaro, M. et al. (2013) orco mutant mosquitoes lose strong preference for humans and are not repelled by volatile DEET. *Nature*, 498, 487–91.

Deng, X. et al. (2015) An Overview of Practical Applications of Protein Disorder Prediction and Drive for Faster, More Accurate Predictions. *Int. J. Mol. Sci.*, 16, 15384–404.

Devi, L. (1991) Consensus sequence for processing of peptide precursors at monobasic sites. *FEBS Lett.*, 280, 189–194.

Dietterich, T.G. (1990) Ensemble Methods in Machine Learning.

Dillen, S. et al. (2013) Functional characterization of the short neuropeptide F receptor in the desert locust, Schistocerca gregaria. *PLoS One*, 8, e53604.

Ding, C.H.Q. and Dubchak, I. (2001) Multi-class protein fold recognition using support vector machines and neural networks. *Bioinformatics*, 17, 349–358.

Dinkel, H. et al. (2014) The eukaryotic linear motif resource ELM: 10 years and counting. *Nucleic Acids Res.*, 42, D259–66.

Dobson, R.J.B. et al. (2009) Global sequence properties for superfamily prediction: a machine learning approach. *J. Integr. Bioinform.*, 6, 109.

Domingos, P. (2012) A few useful things to know about machine learning. *Commun. ACM*, 55, 78.
Dubchak, I. et al. (1995) Prediction of protein folding class using global description of amino acid sequence. *Proc. Natl. Acad. Sci. U. S. A.*, 92, 8700–4.

Edgar, R.C. (2004) Local homology recognition and distance measures in linear time using compressed amino acid alphabets. *Nucleic Acids Res.*, 32, 380–5.

Edgar, R.C. (2010) Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, 26, 2460–2461.

Edgar, R.C. and Sjölander, K. (2004) A comparison of scoring functions for protein sequence profile alignment. *Bioinformatics*, 20, 1301–8.

Etchebest, C. et al. (2007) A reduced amino acid alphabet for understanding and designing protein adaptation to mutation. *Eur. Biophys. J.*, 36, 1059–69.

Ferguson, J.N. et al. (2000) Social amnesia in mice lacking the oxytocin gene. *Nat. Genet.*, 25, 284–8.

Fernández-Delgado, M. et al. (2014) Do we Need Hundreds of Classifiers to Solve Real World Classification Problems? *J. Mach. Learn. Res.*, 15, 3133–3181.

Finn, R.D. et al. (2014) Pfam: The protein families database. *Nucleic Acids Res.*, 42, D222–30.

Fisher, J.M. et al. (1988) Multiple neuropeptides derived from a common precursor are differentially packaged and transported. *Cell*, 54, 813–22.

Fox, N.K. et al. (2014) SCOPe: Structural Classification of Proteins—extended, integrating SCOP and ASTRAL data and classification of new structures. *Nucleic Acids Res.*, 42, D304–9.

Frith, M.C. et al. (2006) The abundance of short proteins in the mammalian proteome. *PLoS Genet.*, 2, 515–528.

Funke, L. et al. (2010) Unique biological function of cathepsin L in secretory vesicles for biosynthesis of neuropeptides. *Neuropeptides*, 44, 457–466.

García-López, M.J. et al. (2002) Physiology of the neuropeptides. *Rev. Neurol.*, 35, 784–793.

Geisser, S. (1993) Predictive Inference.

Geman, S. et al. (1992) Neural Networks and the Bias/Variance Dilemma. *Neural Comput.*, 4, 1–58.

George, R.A. and Heringa, J. (2002) An analysis of protein domain linkers: their classification and role in protein folding. *Protein Eng. Des. Sel.*, 15, 871–879.

Georgiev, A.G. (2009) Interpretable numerical descriptors of amino acid space. *J. Comput. Biol.*, 16, 703–23.

Gillis, J. and Pavlidis, P. (2013) Characterizing the state of the art in the computational assignment of gene function: lessons from the first critical assessment of functional annotation (CAFA). *BMC Bioinformatics*, 14 Suppl 3, S15.

Gitlin, L. et al. (2014) Rapid Evolution of Virus Sequences in Intrinsically Disordered Protein Regions. *PLoS Pathog.*, 10, e1004529.

Goldberg, T. et al. (2014) LocTree3 prediction of localization. *Nucleic Acids Res.*, 42, W350–5.

Goldberg, Y. and Levy, O. (2014) word2vec Explained: deriving Mikolov et al.’s negative-sampling word-embedding method.

Greene, L.H. et al. (2007) The CATH domain structure database: new protocols and classification levels give a more comprehensive resource for exploring evolution. *Nucleic Acids Res.*, 35, D291–7.

Gromiha, M.M. and Suwa, M. (2005) A simple statistical method for discriminating outer membrane proteins with better accuracy. *Bioinformatics*, 21, 961–8.

Gruber, C.W. and Muttenthaler, M. (2012) Discovery of defense- and neuropeptides in social ants by genome-mining. *PLoS One*, 7, e32559.

Guyon, I. (2003) An Introduction to Variable and Feature Selection 1 Introduction. 3, 1157–1182.

Guyon, I. et al. (2002) Gene selection for cancer classification using support vector machines. *Mach. Learn.*, 46, 389–422.

Halevy, A. et al. (2009) The Unreasonable Effectiveness of Data. *IEEE Intell. Syst.*, 24, 8–12.

Hallem, E.A. and Carlson, J.R. (2006) Coding of odors by a receptor repertoire. *Cell*, 125, 143–60.

Hastie, T. et al. (2009) The Elements of Statistical Learning. *Elements*, 1, 337–387.

Hauser, M. et al. (2013) kClust: fast and sensitive clustering of large protein sequence databases. *BMC Bioinformatics*, 14, 248.
Hewes,R.S. and Taghert,P.H. (2001) Neuropeptides and neuropeptide receptors in the Drosophila melanogaster genome. *Genome Res.*, **11**, 1126–1142.

Hobohm,U. and Sander,C. (1995) A sequence property approach to searching protein databases. *J. Mol. Biol.*, **251**, 390–9.

Hochreiter,S. et al. (2007) Fast model-based protein homology detection without alignment. *Bioinformatics*, **23**, 1728–36.

Hochreiter,S. and Schmidhuber,J. (1997) Long short-term memory. *Neural Comput.*, **9**, 1735–80.

Höfkel,T. et al. (2000) Neuropeptides — an overview. *Neuropharmacology*, **39**, 1337–1356.

Huang,G. et al. (2015) Trends in extreme learning machines: A review. *Neural Networks*, **61**, 32–48.

Huang,Y. et al. (2010) CD-HIT Suite: A web server for clustering and comparing biological sequences. *Bioinformatics*, **26**, 680–682.

Ikai,A. (1980) Thermostability and aliphatic index of globular proteins. *J. Biochem.*, **88**, 1895–1898.

Ingolia,N.T. et al. (2012) Identification of DNA sequencing of ribosome-ed mRNA fragments. *Nat. Protoc.*, **7**, 1534–50.

Insel,T.R. and Young,L.J. (2000) Neuropeptides and the evolution of social behavior. *Curr. Opin. Neurobiol.*, **10**, 784–9.

Jaakkola,T. et al. (2000) A discriminative framework for detecting remote protein homologies. *J. Comput. Biol.*, **7**, 95–114.

Jr,P.P. and Fasman,G. (1989) Chou-Fasman prediction of the secondary structure of proteins. In, *Prediction of protein structure and the principles of protein conformation*, pp. 391–416.

Juncker,A.S. et al. (2009) Sequence-based feature prediction and annotation of proteins. *Genome Biol.*, **10**, 206.

Kahanda,I. et al. (2015) A close look at protein function prediction evaluation protocols. *Gigascience*, **4**, 41.

Käll,L. et al. (2007) Advantages of combined transmembrane topology and signal peptide prediction-the Phobius web server. *Nucleic Acids Res.*, **35**, 429–432.

Karplus,K. et al. (1998) Hidden Markov models for detecting remote protein homologies. *Bioinformatics*, **14**, 846–856.

Karsenty,S. et al. (2014) NeuroPID: a classifier of neuropeptide precursors. *Nucleic Acids Res.*, gku363–

Kim,Y. (2014) Convolutional Neural Networks for Sentence Classification. In, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), pp. 1746–1751.

Kim,Y. et al. (2011) NeuroPedia: neuropeptide database and spectral library. *Bioinformatics*, **27**, 2772–3.

Kingsford,C. and Salzberg,S.L. (2008) What are decision trees? *Nat. Biotechnol.*, **26**, 1011–3.

Kister,A.E. and Potapov,V. (2013) Amino acid distribution rules predict protein fold. *Biochem. Soc. Trans.*, **41**, 616–9.

Kobayashi,M. et al. (1998) Overexpression of the forebrain-specific homeobox gene six3 induces rostral forebrain enlargement in zebrafish. *Development*, **125**, 2973–82.

Kohavi,R. (1995) A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. In, *International Joint Conference on Artificial Intelligence*, pp. 1137–1143.

Krizhevsky,A. et al. (2012) ImageNet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.*, 1–9.

Kryshtafovych,A. et al. (2014) CASP10 results compared to those of previous CASP experiments. *Proteins*, **82 Suppl 2**, 164–74.

Kuang,R. et al. (2005) Profile-based string kernels for remote homology detection and motif extraction. *J. Bioinform. Comput. Biol.*, **3**, 527–50.

Kumar,M. et al. (2007) Identification of DNA-binding proteins using support vector machines and evolutionary profiles. *BMC Bioinformatics*, **8**, 463.

Kuznetsov,I.B. and Rackovsky,S. (2003) On the properties and sequence context of structurally ambivalent fragments in proteins. *Protein Sci.*, **12**, 2420–33.

Kyte,J. and Doolittle,R.F. (1982) A simple method for displaying the hydropathic character of a protein. *J. Mol. Biol.*, **157**, 105–132.
Lavelle, D.T. and Pearson, W.R. (2010) Globally, unrelated protein sequences appear random. *Bioinformatics*, 26, 310–8.

Le, Q. V. and Mikolov, T. (2014) Distributed Representations of Sentences and Documents.

LeCun, Y. *et al.* (2015) Deep learning. *Nature*, 521, 436–444.

Leslie, C. *et al.* (2002) The spectrum kernel: a string kernel for SVM protein classification. *Pac. Symp. Biocomput.*, 755, 564–75.

Leslie, C.S. *et al.* (2004) Mismatch string kernels for discriminative protein classification. *Bioinformatics*, 20, 467–76.

Li, Z.R. *et al.* (2006) PROFEAT: a web server for computing structural and physicochemical features of proteins and peptides from amino acid sequence. *Nucleic Acids Res.*, 34, W32–7.

Lin, H. and Chen, W. (2011) Prediction of thermophilic proteins using feature selection technique. *J. Microbiol. Methods*, 84, 67–70.

Linding, R. *et al.* (2003) GlobPlot: Exploring protein sequences for globularity and disorder. *Nucleic Acids Res.*, 31, 3701–3708.

Lipton, Z.C. (2015) A Critical Review of Recurrent Neural Networks for Sequence Learning.

Liu, B. *et al.* (2014) Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection. *Bioinformatics*, 30, 472–9.

Lobley, A. *et al.* (2009) pGenTHREADER and pDomTHREADER: New methods for improved protein fold recognition and superfamily discrimination. *Bioinformatics*, 25, 1761–1767.

Lodhi, H. (2012) Computational biology perspective: kernel methods and deep learning. *Wiley Interdiscip. Rev. Comput. Stat.*, 4, 455–465.

Loewenstein, Y. *et al.* (2009) Protein function annotation by homology-based inference. *Genome Biol.*, 10, 207.

Lu, T. *et al.* (2007) Odor coding in the maxillary palp of the malaria vector mosquito Anopheles gambiae. *Curr. Biol.*, 17, 1533–44.

Lubec, G. and Afjehi-Sadat, L. (2007) Limitations and pitfalls in protein identifications by mass spectrometry. *Chem. Rev.*, 107, 3568–3584.

Luhn, H.P. (1957) A Statistical Approach to Mechanized Encoding and Searching of Literary Information. *IBM J. Res. Dev.*, 1, 309–317.

Lyons, J. *et al.* (2014) Predicting backbone Cα angles and dihedrals from protein sequences by stacked sparse auto-encoder deep neural network. *J. Comput. Chem.*, 35, 2040–6.

Magnan, C.N. and Baldi, P. (2014) SSpro/ACCpro 5: almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning and structural similarity. *Bioinformatics*, btu352.

McDonnell, M.D. *et al.* (2014) Fast, simple and accurate handwritten digit classification using extreme learning machines with shaped input-weights. 13.

McKinney, W. (2011) pandas: a Foundational Python Library for Data Analysis and Statistics. In, *Python for High Performance and Scientific Computing.*, pp. 1–9.

Michelitsch, M.D. and Weissman, J.S. (2000) A census of glutamine/asparagine-rich regions: implications for their conserved function and the prediction of novel prions. *Proc. Natl. Acad. Sci. U. S. A.*, 97, 11910–5.

Midic, U. *et al.* (2009) Protein sequence alignment and structural disorder. In, *Proceedings of the KDD-09 Workshop on Statistical and Relational Learning in Bioinformatics - StReBio ’09*. ACM Press, New York, New York, USA, pp. 27–31.

Mikolov, T. *et al.* (2013) Distributed Representations of Words and Phrases and their Compositionality. *Nips*, 1–9.

Mirabeau, O. and Joly, J.-S. (2013) Molecular evolution of peptidergic signaling systems in bilaterians. *Proc. Natl. Acad. Sci. U. S. A.*, 110, E2028–37.

Moore, A.W. (2004) K-means and Hierarchical Clustering. 1–24.

Mulder, N. and Apweiler, R. (2007) InterPro and InterProScan: tools for protein sequence classification and comparison. *Methods Mol. Biol.*, 396, 59–70.
Murphy, L.R. et al. (2000) Simplified amino acid alphabets for protein fold recognition and implications for folding. Protein Eng., 13, 149–52.
Naamati, G. et al. (2009) ClanTox: A classifier of short animal toxins. Nucleic Acids Res., 37.
Nagarajan, N. and Yona, G. (2004) Automatic prediction of protein domains from sequence information using a hybrid learning system. Bioinformatics, 20, 1335–60.
Nanni, L. et al. (2014) An empirical study of different approaches for protein classification. ScientificWorldJournal, 2014, 236717.
Nanni, L. et al. (2010) High performance set of PseAAC and sequence based descriptors for protein classification. J. Theor. Biol., 266, 1–10.
Nässel, D.R. (2002) Neuropeptides in the nervous system of Drosophila and other insects: Multiple roles as neuromodulators and neurohormones. Prog. Neurobiol., 68, 1–84.
Nathoo, A.N. et al. (2001) Identification of neuropeptide-like protein gene families in Caenorhabditis elegans and other species. Proc. Natl. Acad. Sci. U. S. A., 98, 14000–5.
Nguyen, A.D. et al. (2011) Neuropeptide Y and peptide YY: important regulators of energy metabolism. Curr. Opin. Endocrinol. Diabetes. Obes., 18, 56–60.
Nugent, T. and Jones, D.T. (2009) Transmembrane protein topology prediction using support vector machines. BMC Bioinformatics, 10, 159.
Obradovic, Z. et al. (2005) Exploiting heterogeneous sequence properties improves prediction of protein disorder. Proteins, 61 Suppl 7, 176–82.
Ofer D et al. (2014) NeuroPID: a predictor for identifying neuropeptide precursors from metazoan proteomes. Bioinformatics, 30, 931–40.
Ofer, D. and Linial, M. (2015) ProFET: Feature engineering captures high-level protein functions. Bioinformatics.
Oliphant, T.E. (2007) SciPy: Open source scientific tools for Python. Comput. Sci. Eng., 9, 10–20.
Orosz, F. and Ovádi, J. (2011) Proteins without 3D structure: definition, detection and beyond. Bioinformatics, 27, 1449–54.
Pedregosa, F. et al. (2011) Scikit-learn: Machine Learning in Python. J. Mach. Learn. Res., 12, 2825–2830.
Peng, H. et al. (2005) Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans. Pattern Anal. Mach. Intell., 27, 1226–38.
Petersen, T.N. et al. (2011) SignalP 4.0: discriminating signal peptides from transmembrane regions. Nat. Methods, 8, 785–6.
Peterson, E.L. et al. (2009) Reduced amino acid alphabets exhibit an improved sensitivity and selectivity in fold assignment. Bioinformatics, 25, 1356–62.
Ponting, C.P. and Grant Belgard, T. (2010) Transcribed dark matter: Meaning or myth? Hum. Mol. Genet., 19.
Portugaly, E. et al. (2002) Selecting targets for structural determination by navigating in a graph of protein families. Bioinformatics, 18, 899–907.
Portugaly, E. and Linial, M. (2000) Estimating the probability for a protein to have a new fold: A statistical computational model. Proc. Natl. Acad. Sci. U. S. A., 97, 5161–6.
Prilusky, J. et al. (2005) FoldIndex: a simple tool to predict whether a given protein sequence is intrinsically unfolded. Bioinformatics, 21, 3435–8.
Qi, Y. et al. (2012) A unified multitask architecture for predicting local protein properties. PLoS One, 7, e32235.
Radivojac, P. et al. (2013) A large-scale evaluation of computational protein function prediction. Nat. Methods, 10, 221–7.
Rafaeli, A. (2009) Pheromone biosynthesis activating neuropeptide (PBAN): Regulatory role and mode of action. Gen. Comp. Endocrinol., 162, 69–78.
Rao, H.B. et al. (2011) Update of PROFEAT: a web server for computing structural and physicochemical features of proteins and peptides from amino acid sequence. Nucleic Acids Res., 39, W385–90.
Rappoport, N. et al. (2010) PANDORA: analysis of protein and peptide sets through the hierarchical integration of annotations. *Nucleic Acids Res.*, 38, W84–9.

Rappoport, N. et al. (2012) ProtoNet 6.0: organizing 10 million protein sequences in a compact hierarchical family tree. *Nucleic Acids Res.*, 40, D313–20.

Rappoport, N. and Linial, M. (2013) Functional inference by ProtoNet family tree: the uncharacterized proteome of Daphnia pulex. *BMC Bioinformatics*, 14 Suppl 3, S11.

Remmert, M. et al. (2011) HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat. Methods*, 9, 173–5.

Rentzsch, R. and Orengo, C. a. (2009) Protein function prediction - the power of multiplicity. *Trends Biotechnol.*, 27, 210–219.

Root, C. M. et al. (2011) Presynaptic facilitation by neuropeptide signaling mediates odor-driven food search. *Cell*, 145, 133–44.

Rossum, G. Van and Drake, F. L. (2010) The Python Library Reference. *October*, 1–1144.

Rost, B. (1999) Twilight zone of protein sequence alignments. *Protein Eng.*, 12, 85–94.

Saeys, Y. et al. (2007) A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23, 2507–17.

Salsbury, J. P. et al. (2013) A rapid MALDI-TOF mass spectrometry workflow for Drosophila melanogaster differential neuropeptidomics. *Mol. Brain*, 6, 60.

Saraç, O. S. et al. (2010) GOPred: GO molecular function prediction by combined classifiers. *PLoS One*, 5, e12382.

Sasson, O. et al. (2006) Functional annotation prediction: all for one and one for all. *Protein Sci.*, 15, 1557–1562.

Scheraga, H. A. and Rackovsky, S. (2014) Homolog detection using global sequence properties suggests an alternate view of structural encoding in protein sequences. *Proc. Natl. Acad. Sci. U. S. A.*, 111, 5225–9.

Schneider, T. D. (2010) A brief review of molecular information theory. *Nano Commun. Netw.*, 1, 173–180.

Semmens, D. C. et al. (2015) Discovery of sea urchin NGFFamide receptor unites a bilaterian neuropeptide family. *Open Biol.*, 5, 150030.

Shachar, O. and Linial, M. (2004) A robust method to detect structural and functional remote homologues. *Proteins*, 57, 531–8.

Shah, A. R. et al. (2008) SVM-HUSTLE—an iterative semi-supervised machine learning approach for pairwise protein remote homology detection. *Bioinformatics*, 24, 783–90.

Sharif, A. et al. (2014) CNN Features off-the-shelf: an Astounding Baseline for Recognition. In, CVPRW '14 Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops., pp. 512–519.

Sharma, A. et al. (2014) Evaluation of sequence features from intrinsically disordered regions for the estimation of protein function. *PLoS One*, 9, e89890.

Shaw, C. (1996) Neuropeptides and their evolution. *Parasitology*, 113 Suppl, S35–S45.

Sikic, K. and Carugo, O. (2010) Protein sequence redundancy reduction: comparison of various method. *Bioinformation*, 5, 234–9.

Simonyan, K. and Zisserman, A. (2014) Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv, 1–13.

Smit, A. B. et al. (1992) Characterization of a cDNA clone encoding multiple copies of the neuropeptide APGWamide in the mollusk Lymnaea stagnalis. *J. Neurosci.*, 12, 1709–15.

Smith, T. F. and Waterman, M. S. (1981) Identification of common molecular subsequences. *J. Mol. Biol.*, 147, 195–197.

Society, M. P. Support Vector Machines and Kernels for Computational Biology Gunnar R¨.  
Söding, J. (2005) Protein homology detection by HMM-HMM comparison. *Bioinformatics*, 21, 951–960.

Sønderby, S. K. et al. (2015) Convolutional LSTM Networks for Subcellular Localization of Proteins.  
Sønderby, S. K. and Winther, O. (2014) Protein Secondary Structure Prediction with Long Short Term Memory Networks.
Southey,B.R. et al. (2008) A Python Analytical Pipeline to Identify Prohormone Precursors and Predict Prohormone Cleavage Sites. *Front. Neuroinform.*, 2, 9.
Southey,B.R., Amare,A., et al. (2006) NeuroPred: a tool to predict cleavage sites in neuropeptide precursors and provide the masses of the resulting peptides. *Nucleic Acids Res.*, 34, W267–72.
Southey,B.R., Rodriguez-Zas,S.L., et al. (2006) Prediction of neuropeptide prohormone cleavages with application to RFamides. *Peptides*, 27, 1087–98.
Su,M. et al. (2013) Small proteins: untapped area of potential biological importance. *Front. Genet.*, 4, 286.
Susko,E. and Roger,A.J. (2007) On reduced amino acid alphabets for phylogenetic inference. *Mol. Biol. Evol.*, 24, 2139–50.
Suzek,B.E. et al. (2015) UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, 31, 926–32.
Suzek,B.E. et al. (2007) UniRef: Comprehensive and non-redundant UniProt reference clusters. *Bioinformatics*, 23, 1282–1288.
Taghert,P.H. and Nitabach,M.N. (2012) Peptide neuromodulation in invertebrate model systems. *Neuron*, 76, 82–97.
Tanaka,Y. et al. (2014) Transcriptomic analysis of neuropeptides and G-protein coupled receptors (GPCRs) for neuropeptides in the brown planthopper Nilaparvata lugens. *Peptides*, 53, 125–33.
Tarca,A.L. et al. (2007) Machine learning and its applications to biology. *PLoS Comput. Biol.*, 3, e116.
Tauxe,G.M. et al. (2013) Targeting a dual detector of skin and CO2 to modify mosquito host seeking. *Cell*, 155, 1365–79.
Tegge,A.N. et al. (2008) Comparative analysis of neuropeptide cleavage sites in human, mouse, rat, and cattle. *Mamm. Genome*, 19, 106–20.
Tirosh,Y. et al. (2012) Short toxin-like proteins abound in Cnidaria genomes. *Toxins (Basel)*, 4, 1367–84.
Tirosh,Y. et al. (2013) Short toxin-like proteins attack the defense line of innate immunity. *Toxins (Basel)*, 5, 1314–31.
Tools,D. et al. (2010) ExPASy Proteomics Server. *Search*, 1–2.
Toporik,A. et al. (2014) Computational identification of natural peptides based on analysis of molecular evolution. *Bioinformatics*, 30, 2137–2141.
Touw,W.G. et al. (2013) Data mining in the Life Sciences with Random Forest: a walk in the park or lost in the jungle? *Brief. Bioinform.*, 14, 315–26.
Uversky,V.N. et al. (2000) Why are ‘natively unfolded’ proteins unstructured under physiologic conditions? *Proteins*, 41, 415–427.
Vacic,V. et al. (2007) Composition Profiler: a tool for discovery and visualization of amino acid composition differences. *BMC Bioinformatics*, 8, 211.
Varshavsky,R. et al. (2007) When Less Is More: Improving Classification of Protein Families with a Minimal Set of Global Features. 12–24.
Veenstra,J.A. (2000) Mono- and dibasic proteolytic cleavage sites in insect neuroendocrine peptide precursors. *Arch. Insect Biochem. Physiol.*, 43, 49–63.
Vincent,P. et al. (2008) Deep Learning with Denoising Autoencoders. *J. Mach. Learn.*, 27, 49–50.
Vogel,C. and Marcotte,E.M. (2008) Calculating absolute and relative protein abundance from mass spectrometry-based protein expression data. *Nat. Protoc.*, 3, 1444–1451.
Wang,L. et al. (2010) BindN+ for accurate prediction of DNA and RNA-binding residues from protein sequence features. *BMC Syst. Biol.*, 4 Suppl 1, S3.
Wang,Z. et al. (2009) RNA-Seq: a revolutionary tool for transcriptomics. *Nat. Rev. Genet.*, 10, 57–63.
Weathers,E.A. et al. (2004) Reduced amino acid alphabet is sufficient to accurately recognize intrinsically disordered protein. *FEBS Lett.*, 576, 348–352.
Wilkins,M.R. and Williams,K.L. (1997) Cross-species protein identification using amino acid composition, peptide mass fingerprinting, isoelectric point and molecular mass: a theoretical evaluation. *J. Theor. Biol.*, 186, 7–15.
Xu, K. et al. (2015) Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. 
*ICML-2015.*
Xu, Z.E. et al. (2011) Rapid Feature Learning with Stacked Linear Denoisers. 10.
Yachdav, G. et al. (2014) PredictProtein—an open resource for online prediction of protein structural and functional features. *Nucleic Acids Res.*, **42**, W337–343.
Yosinski, J. et al. (2014) How transferable are features in deep neural networks? In, *Advances in Neural Information Processing Systems 27 (Proceedings of NIPS)*., pp. 1–9.
Yu, S.X. and Shi, J. (2003) Multiclass spectral clustering. In, *Proceedings Ninth IEEE International Conference on Computer Vision.*, pp. 313–319 vol.1.
Zhao, Y. et al. (2012) RAPSearch2: a fast and memory-efficient protein similarity search tool for next-generation sequencing data. *Bioinformatics*, **28**, 125–6.
Zhou, J. and Troyanskaya, O.G. (2014) Deep Supervised and Convolutional Generative Stochastic Network for Protein Secondary Structure Prediction.
7 APPENDICES

APPENDIX 1 NEUROPID: A PREDICTOR FOR IDENTIFYING NEUROPEPTIDE PRECURSORS FROM METAZOAN PROTEOMES 64
APPENDIX 2 NEUROPID: A CLASSIFIER OF NEUROPEPTIDE PRECURSORS 65
APPENDIX 3 PROFET: FEATURE ENGINEERING CAPTURES HIGH-LEVEL PROTEIN FUNCTIONS 66
APPENDIX 4: NEUROPRED SUPPLEMENTARY: STATISTICAL SIGNIFICANCE ANALYSIS OF FEATURES: NEUROPEPTIDES VS BACKGROUND 67
STATISTICAL SIGNIFICANCE OF FEATURES ANALYSIS: NEUROPEPTIDES VS BACKGROUND 67
APPENDIX 5 IN-SILICO CNIDARIA PREDICTIONS 71
APPENDIX 1 NEUROPID: A PREDICTOR FOR IDENTIFYING NEUROPEPTIDE PRECURSORS FROM METAZOAN PROTEOMES

Bioinformatics Advance Access published December 13, 2013

Sequence Analysis

NeuroPID: A Predictor for Identifying Neuropeptide Precursors from Metazoan Proteomes

Dan Ofer1 and Michal Linial1,*
1Department of Biological Chemistry, Institute of Life Sciences, The Edmond J. Safra Campus, The Hebrew University of Jerusalem, Givat Ram, 91904, Israel

Associate Editor: Dr. John Hancock

ABSTRACT

Motivation: The evolution of multicellular organisms is associated with increasing variability of molecules governing behavioral and physiological states. This is often achieved by Neuropeptides (NPs) that are produced in neurons from a longer protein, named neuropeptide precursor (NPP). The maturation of NPs occurs through a sequence of proteolytic cleavages. The difficulty in identifying NPPs is in consequence of their diversity and the lack of applicable sequence similarity among the short, functionally related NPs.

Results: Herein, we describe NeuroPID, a machine-learning scheme that predicts Metazoan NPPs. NeuroPID was trained on hundreds of identified NPPs from the UniProtKB database. Some 600 features were extracted from the primary sequences and processed using support vector machines (SVM) and ensemble decision tree classifiers. These features combined biophysical, chemical and information-based statistical properties of NPs and NPPs. Other features were guided by the defining characteristics of the dibasic cleavage sites motif. NeuroPID reached 89-94% accuracy and 90-93% precision in cross-validation blind tests against known NPPs (with an emphasis on Chordata and Arthropoda). NeuroPID also identified NPP-like proteins from extensively studied model organisms as well as from poorly annotated proteomes. We then focused on the most significant sets of features that contribute to the success of the classifiers. We propose that NPPs are attractive targets for investigating and modulating behavior, metabolism and homeostasis, and that a rich repertoire of NPs remains to be identified.

Availability: NeuroPID source code is freely available at http://www.prototol.cs.huji.ac.il/neuropid

Contact: michail@cs.huji.ac.il

1 INTRODUCTION

Peptides are known to be key modulators in behavior, sensation and homeostasis (Bhan and Cox, 2006). Biologically active peptides that are produced and secreted from neurons and act to modulate their function are collectively called peptide modulators or Neuropeptides (NPs). NPs represent a wide array of communication that is found from Cnidarians to Bilaterians, including mammals. NP precursors (NPPs) are subjected to regulated cleavages that result in the production of functionally active NPs. The processing of NPs occurs along the secretory pathway (Gelfman and Fricker, 2010). In most instances, NPs locally modulate presynaptic or postsynaptic cell activity (Soutey, et al., 2008). From a functional perspective, known effects of NPs include stress control, pain perception, social behaviors and sleep-wake cycle (Nasel, 2002). NPs also regulate food intake, maintaining appetite and body weight. The peripheral NPs that act outside the central nervous system (CNS) regulate gastrointestinal and immunological functions (Insel and Young, 2000). For example, some NPs induce chemotactic response in attracting immature dendritic cells to the site of inflammation. In this context, substance P and K, two extensively studied NPs, induce the production and release of inflammatory cytokines from blood monocytes (Gonzalez-Roy, et al., 2007). “Social” NPs such as oxytocin (OXT) and arginine vasopressin (AVP) regulate complex social cognition and behavior (including pair-bonding, social recognition and maternal behavior) (Insel and Young, 2000). The immune diversity among NPs contributes to the wide range of behavioral tasks that are carried out.

A common feature for the majority of NPs is their production from a larger precursor (NPP). The production of short bioactive peptides is a result of a series of cleavages and maturation events (Munteanu, et al., 2007). NPP can produce multiple copies of different NPs (Munteanu and Dahms, 1994). Notably, a cluster of basic residues specifies these cleavage sites. The occurrence of dibasic residues specifies the canonical sites for intracellular endopeptidases such as Furin (Vennem, 2006). However, some NPs that act in cell-cell communication (Funkelshain, et al., 2010) (e.g. carapeen L1) do not obey the dibasic residues specificity rule. The identity and regulation of the key peptides that are responsible for the processing of the NPP remains an active research field.

NPs typically bind to surface G-proteins coupled receptors (GPCRs) that initiate a signaling cascade. An evolutionary analysis of NPPs and their cognate GPCRs suggested a diversification that occurred in certain taxonomic branches (Sikorski, 2013). The NPs and their receptors are attractive targets for drug development and translational medicine based on their role in feeding, sexual behavior and cellular homeostasis (Bhan and Cox, 2006). The goal of this research is to enable systematic identification of NPPs (and NPs) at a genome-wide scale. The difficulty in identifying NPs and classifying genes as potential NPs stems from the following: (i) Current gene annotation tools mostly rely on se-
APPENDIX 2 NEUROPID: A CLASSIFIER OF NEUROPEPTIDE PRECURSORS

NeuroPID: a classifier of neuropeptide precursors

Solange Karsenty1,2, Nadav Rappoport1, Dan Ofer3, Advu Zait1 and Michal Linial1,3

1School of Computer Science and Engineering, The Hebrew University of Jerusalem, Jerusalem, Israel, 2School of Computer Science, Harodosph Academic College, Jerusalem, Israel and 3Department of Biological Chemistry, The Alexander Silberman Institute of Life Sciences, The Sudekens Center for Computational Biology, The Hebrew University of Jerusalem, Jerusalem, Israel

Received February 21, 2014; Revised April 07, 2014; Accepted April 15, 2014

ABSTRACT
Neuropeptides (NPs) are short secreted peptides produced in neurons. NPs act by activating signaling cascades governing broad functions such as metabolism, sensation and behavior throughout the animal kingdom. NPs are the products of multistep processing of longer proteins, the NP precursors (NPPs). We present NeuroPID (Neuropeptide Precursor Identifier), an online machine-learning tool that identifies metazoan NPPs. NeuroPID was trained on 1418 NPPs annotated as such by UniProtKB. A large number of sequence-based features were extracted for each sequence with the goal of capturing the biophysical and informational-statistical properties that distinguish NPPs from other proteins. Training several machine-learning models, including support vector machines and ensemble decision trees, led to high accuracy (80–94%) and precision (90–93%) in cross-validation tests. For inputs of thousands of unseen sequences, the tool provides a ranked list of high quality predictions based on the results of four machine-learning classifiers. The output reveals many uncharacterized NPPs and secreted cell modulators that are rich in potential cleavage sites. NeuroPID is a discovery and a prediction tool that can be used to identify NPPs from unannotated transcriptomes and mass spectrometry experiments. NeuroPID predicted sequences are attractive targets for investigating behavior, physiology and cell modulation. The NeuroPID web tool is available at http://neuropid.cs.huji.ac.il.

INTRODUCTION
The functions elicited by neuropeptides (NPs) cover most aspects of metazoon life including metabolism, growth and social behavior (1). For example, in mollusks and insects, mating behavior and reproduction are regulated by NPs. More generally, NPs function in energy consumption, food uptake, pain and sensation, temperature control, appetite, mating and social behavior. NPs act through binding their cognate receptor and activating a signaling cascade (2).

Most mature NPs are short peptides (5–30 aa) that are produced from a longer polypeptide, referred to as a NP precursor (NPP). Active NPs are the products of multiple cleavages of the precursor. Such cleavages mostly occur at basic residues (Arg and Lys) motifs that flank the NPs. While dibasic residues are the hallmark of the cleavage sites for endopeptidases (3), in some NPs such canonical cleavage sites are not detected.

Several bioinformatics approaches were developed to enhance the routine BLAST searching protocol by incorporating distinctive properties of NPs (e.g. (4)). However, at present, there is no systematic approach to identify NPPs in poorly annotated metazoan genomes. The organization of the NPP as a source of multiple NPs is conserved despite a weak similarity between NPPs along the evolutionary tree (3). Currently, NPPs are sporadically identified from sequencing projects of organisms and from the collections of large-scale proteomics and transcriptomics experiments. We developed a systematic approach for identifying candidate NPPs (6). Large-scale mass spectrometry (MS) proteomics identified thousands of spectra that derived from NPP genes. Additionally, a large number of bioactive modulatory peptides were identified (e.g. (7)). For peptides collected from human samples, only about 20% of the associated genes are currently annotated as NPPs. The expansion in MS proteomics and the depth of the transcriptome coverage suggest that NeuroPID is valuable as a discovery platform for bioactive modulators. NPPs and their associated NPs are attractive targets for drug development and pharmaceutical manipulations.

OUTLINE—FEATURES TO PREDICTIONS
Figure 1 shows a prototype of an NPP that belongs to the Allatostatin family as an illustration for the challenges in identifying such genes at a genomic scale. The Allatostatins are produced in many insects to control food intake (8). The sequence of the Pacific beetle cockroach (Figure 1) contains 13 identified NPs. While each peptide has a unique se
APPENDIX 3 PROFET: FEATURE ENGINEERING CAPTURES HIGH-LEVEL PROTEIN FUNCTIONS

Bioinformatics Advance Access published July 2, 2015

Bioinformatics, 2015, 1-4
doi: 10.1093/bioinformatics/btv265
Advance Access Publication Date: 30 June 2015
Original article

Sequence analysis

ProFET: Feature engineering captures high-level protein functions

Dan Ofer and Michal Linial*

Department of Biological Chemistry, Institute of Life Sciences, The Edmond J. Safra Campus, The Hebrew University of Jerusalem, 91904, Israel

*To whom correspondence should be addressed.

Received on March 4, 2015; revised on May 2, 2015; accepted on May 28, 2015

Abstract

Motivation: The amount of sequenced genomes and proteins is growing at an unprecedented pace. Unfortunately, manual curation and functional knowledge lag behind. Homologous inference often fails at labeling proteins with diverse functions and broad classes. Thus, identifying high-level protein functionality remains challenging. We hypothesize that a universal feature engineering approach can yield classification of high-level functions and unified properties when combined with machine learning approaches, without requiring external databases or alignment.

Results: In this study, we present a novel bioinformatics toolkit called ProFET (Protein Feature Engineering Toolkit). ProFET extracts hundreds of features covering the elementary biophysical and sequence derived attributes. Most features capture statistically informative patterns. In addition, different representations of sequences and the amino acids alphabet provide a compact, compressed set of features. The results from ProFET were incorporated in data analysis pipelines, implemented in python and adopted for multi-genome scale analysis. ProFET was applied on 17 established and novel protein benchmark datasets involving classification for a variety of binary and multi-class tasks. The results show state of the art performance. The extracted features’ show excellent biological interpretability. The success of ProFET applies to a wide range of high-level functions such as subcellular localization, structural classes and proteins with unique functional properties (e.g. neurotensin precursors, thermophilic and nucelic acid binding). ProFET allows easy, universal discovery of new target proteins, as well as understanding the features underlying different high-level protein functions.

Availability and implementation: ProFET source code and the datasets used are freely available at https://github.com/ddoffer/ProFET.

Contact: michali@cc.huji.ac.il

Supplementary information: Supplementary data are available at Bioinformatics online.

1 Introduction

The most used approaches in protein classification rely on distance measures between sequences based on various alignment methods (e.g. Smith-Waterman, BLAST). With the growth in the amounts and diversity of protein sequences, more sophisticated methods have been introduced (e.g. PSI-M, Profile-Profile, HMM-HMM) (Jabbaria et al., 2006; Boding, 2005). These methods are based on multiple sequence alignments for improving remote homologous detection (Edgar and Sippl, 2004; Krapivov et al., 1998). Incorporating 3D-structure as a seed for the statistical models further improved the quality of protein domains and families (e.g. Pfam) (Finn et al., 2014; Sonnhammer et al., 1997). Currently, there are ~27,000 such models (InterPro, Müller and Apweiler, 2007) that cover 81% of all sequences in UniProtKB (2014,10). Function assignment is gained from mapping InterPro models to Gene Ontologies (i.e. InterPro2GO). An alternative model-free approach...
APPENDIX 4: NEUROPRED SUPPLEMENTARY: STATISTICAL SIGNIFICANCE ANALYSIS OF FEATURES: NEUROPEPTIDES VS BACKGROUND

The following graphs are from the supplementary material of the original NeuroPred article. They reflect differences between neuropeptides (NP and NPPs) and the complete “background” distribution of proteins drawn from Uniprot/Uniref (this set includes the NPs within itself).

Statistical Significance of Features Analysis: Neuropeptides vs Background:

Elaborations of various features: http://web.expasy.org/protparam/protparam-doc.html#ref7.
| Feature     | K-S Test P Value | T Test P Value | Graph/Plots                           |
|-------------|------------------|---------------|---------------------------------------|
| Molecular Weight | P = 1.787 2e-76  | P = 4.3665e-09 | ![Histogram of "ALL"](image)          |
|             | P = 8.023 9e-21  | P = 1.0797e-16 | ![Histogram of NP](image)              |
| PI          |                  |               | ![Histogram of "ALL"](image)          |
|             |                  |               | ![Histogram of NP](image)              |
Instability.
(A value above 40 means the protein is unstable = short half-life)

|        |       |       |
|--------|-------|-------|
| Instability | 8.666 | 5e-11 |
| GRAVY  | 1.499 | 7e-16 |

P = 0.3521

P = 0.4130
### Aromaticity

|                |                  |
|----------------|------------------|
| y= frequency   | P =              |
| of Phe+Trp+    | 2.236 5e-40 2.85|
| Tyr           | 65e-28          |

### Amino Acid Frequencies (NP vs background proteome)

| AA | A    | C    | E    | D    | G    | F    | I    | H    | K    |
|----|------|------|------|------|------|------|------|------|------|
|    | 2TTest P | 2.8364e-24 | 0.7209 | 1.4261e-45 | 1.5434e-07 | 5.1228e-22 | 6.6598e-41 | 1.7323e-45 | 0.0431 | 3.9703e-81 |

| AA | M    | L    | N    | Q    | P    | S    | R    | T    | W    | V    | Y    |
|----|------|------|------|------|------|------|------|------|------|------|------|
|    | 2TTest P | 1.1661e-15 | 0.0103 | 0.2491 | 0.8762 | 4.8877e-34 | 7.8457e-22 | 3.7566e-08 | 4.5153e-16 | 8.6019e-10 | 3.7784e-26 | 0.7476 |

![Histogram of Aromaticity](image)
APPENDIX 5 IN-SILICO CNIDARIA PREDICTIONS

The full list of predicted, high confidence Neuropeptide-like candidates in a variety of Cnidarians, including Nematostella Vectensis, Hydra Vulgaris, Ctenophora-Sea Gooseberry, and Mnemiopsis, is available on request. 68 of the highest confidence predictions for Nematostella vectensis were manually annotated, and their headers are provided here. The manual annotations themselves are available upon request, as they are part of an ongoing, unpublished research-collaboration. These are Nematostella Vectensis proteins with no known domains in PFAM, a predicted Signal peptide (according to SignalP and Phobius) and no predicted Transmembrane domain, and that are predicted to be Neuropeptides with a high confidence (majority vote by the 4 classifiers) by the online NeuroPID platform.