BIOINFORMATICS APPLICATIONS NOTE Vol. 25 no. 20 2009, pages 2758–2759
doi:10.1093/bioinformatics/btp409

Systems biology

FRED—a framework for T-cell epitope detection

Magdalena Feldhahn[1,*], Pierre Dönnes[1,2], Philipp Thiel[1] and Oliver Kohlbacher[1]

[1]Division for Simulation of Biological Systems, WSI/ZBIT, University of Tübingen, Sand 14,
D-72076 Tübingen, Germany and [2]Present address: Molecular Toxicology, Safety Assessment, AstraZeneca R&D,
S-15185 Södertälje, Sweden

Received on March 8, 2009; revised on June 22, 2009; accepted on June 28, 2009

Advance Access publication July 6, 2009

Associate Editor: John Quackenbush

ABSTRACT

Summary: Over the last decade, immunoinformatics has made significant progress. Computational approaches, in particular the prediction of T-cell epitopes using machine learning methods, are at the core of modern vaccine design. Large-scale analyses and the integration or comparison of different methods become increasingly important. We have developed FRED, an extendable, open source software framework for key tasks in immunoinformatics. In this, its first version, FRED offers easily accessible prediction methods for MHC binding and antigen processing as well as general infrastructure for the handling of antigen sequence data and epitopes. FRED is implemented in Python in a modular way and allows the integration of external methods.

Availability: FRED is freely available for download at
http://www-bs.informatik.uni-tuebingen.de/Software/FRED.

Contact: feldhahn@informatik.uni-tuebingen.de

1 INTRODUCTION

The detection of T-cell epitopes is a critical step in vaccine design and a key problem in immunoinformatics. Experimental studies to detect epitopes are expensive and time consuming. Computational methods are able to reduce this experimental effort and thereby facilitate the process of epitope detection (DeLuca and Blasczyk, 2007). Many computational methods (based on, e.g. position-specific scoring matrices, various machine learning methods or structural information) have been developed for this task. Many of these methods are freely available through the internet, however, few standalone implementations are available. While web-based predictions are easy to use on the small scale, it severely hampers large-scale predictions and makes a direct comparison of individual methods difficult. The development of flexible prediction and analysis pipelines that can handle large amounts of data and combine prediction methods becomes increasingly important. These pipelines include extensive and flexible pre- and post-processing in addition to the application of a prediction method. The web-based methods available today do not offer tools for flexible data processing. Since there is no uniform interface to access these methods it is difficult to include them into automated prediction pipelines. One option to provide convenient and coherent interfaces to immunoinformatics tools is through web services (Halling-Brown *et al.*, 2009); however, speed and availability tend to limit this approach particularly for large-scale studies.

Here, we present FRED, a software framework for computational immunomics, that provides a uniform interface for a variety of prediction methods and support for the implementation of custom-tailored prediction pipelines. FRED offers methods for extensive data processing as well as methods to assess and compare the performance of the prediction methods. This makes it a powerful platform for the rapid development of new algorithms and the analysis of large datasets.

2 METHODS

2.1 Implementation

FRED provides methods for sequence input, sequence preprocessing, filtering and display of the results. The general organization of FRED is depicted in Figure 1. The single prediction methods are accessed internally via a consistent interface. FRED can handle polymorphic sequences (e.g. for the study of single nucleotide polymorphisms (SNPs) in an epitope context) and offers the possibility of accessing different methods simultaneously and of combining, comparing or benchmarking the methods. FRED is easily extendable by user-defined prediction methods or methods for filtering of results. FRED is implemented in Python (release 2.6) (www.python.org). All additional software required for FRED is freely available and installation packages are included in the FRED package. The prediction methods currently available in FRED are listed in Table 1.

3 APPLICATIONS

Tutorial and documentation: with the FRED package, we provide examples that demonstrate how FRED can be used to solve typical tasks in computational immunomics with short and simple scripts. A detailed tutorial is available on the project's web site. It explains how to implement prediction pipelines, offers more detailed information on the functionality of FRED and addresses problems like choosing the right prediction method or threshold. We additionally provide a detailed documentation of the code.

Vaccine design: the selection of peptides for epitope-based vaccines is a typical application for large-scale predictions of MHC binding peptides. The following short and simple program implements a typical scenario for the selection of conserved peptide candidates for a vaccine against a virus. The scenario is based on the paper by Toussaint *et al.* (2008): a set of sequences of the hepatitis C virus core protein from four different subtypes is used. All peptides that occur

[*]To whom correspondence should be addressed.

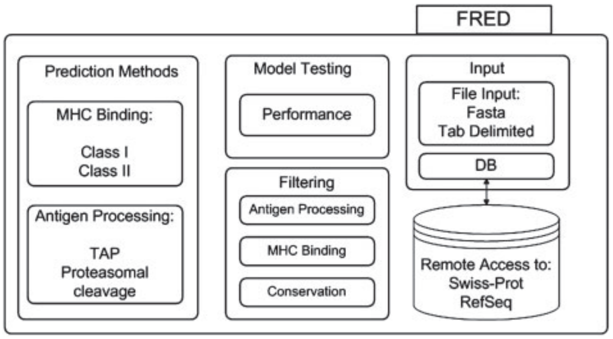

Fig. 1. FRED is organized into four major parts: sequence input, application of prediction methods, filtering of the results and model testing.

Table 1. Prediction methods currently integrated in FRED

Method	References
MHC binding:	
SYFPEITHI	Rammensee *et al.* (1999)
SVMHC	Dönnes and Kohlbacher (2006)
BIMAS	Parker *et al.* (1994)
NetMHCpan[a]	Nielsen *et al.* (2007)
NetMHC[a]	Buus *et al.* (2003)
Hammer	Sturniolo *et al.* (1999)
NetMHCIIpan[a]	Nielsen *et al.* (2008)
Proteasomal Cleavage:	
PCM method from WAPP	Dönnes and Kohlbacher (2005)
TAP Transport:	
SVMTAP	Dönnes and Kohlbacher (2005)
Additive matrix method	Doytchinova *et al.* (2004)

[a] Installation of external software is required. Due to licensing issues, we could not include the standalone versions of these methods in the FRED package.

in at least 90% of the input sequences are considered candidates for conserved epitopes. Predictions are made for 29 HLA alleles using the BIMAS method (Parker *et al.*, 1994).

```
models = [('MHC_I_BIMAS','A_0201_9'),('MHC_I_BIMAS','A_1101_9'),...
prot_set= Fred.Fred.ProteinSet()
prot_set.readFasta("hcv-core-1a1b2a3a.fasta")
pep_set = protset.getPeptides(9, 0.9, 0)[0]
pep_set.predict(models)
cands = Fred.Fred.FindCandidates_HalfmaxThresholds(pep_set,models,0.9,1)
cands.writeTabDelimitedFile('results_hcv_core.txt')
```

Integration of new methods and performance evaluation: epitope prediction is still a very active field, with new methods continuously being developed. Such methods not implemented in Python can be plugged in using command-line calls. FRED provides a number of standard measures to compare different prediction methods and to evaluate the performance w.r.t. experimental values (Matthews Correlation Coefficient, accuracy, sensitivity, specificity, area under the ROC curve, correlation and rank correlation). Different prediction methods can thus be compared with ease.

Web server development: using FRED as the basis for new applications in computational immunomics leads to a significant reduction of development time and allows the convenient combination of new methods with existing ones. An example of an application based on FRED is EpiToolKit (www.epitoolkit.org, Feldhahn *et al.*, 2008; Toussaint and Kohlbacher, 2009). Only the web-based user interface and the data management in the web server had to be newly implemented. Prediction functionality of EpiToolKit is completely provided by FRED. Through the use of Python, FRED can be integrated seamlessly in web servers/content management systems like Plone (http://www.plone.org/).

4 CONCLUSIONS

FRED is a valuable tool for performing large-scale analyses in immunoinformatics with different prediction methods and is also a software framework for the development of novel immunoinformatics methods. Ease of use, extendability and openness make it an ideal tool for addressing complex immunoinformatics problems in an uncomplicated manner.

Funding: Deutsche Forschungsgemeinschaft (SFB 685/B1).

Conflict of Interest: none declared.

REFERENCES

Buus,S. *et al*. (2003) Sensitive quantitative predictions of peptide-MHC binding by a 'Query by Committee' artificial neural network approach. *Tissue Antigens*, **62**, 378–384.

DeLuca,D.S. and Blasczyk,R. (2007) The immunoinformatics of cancer immunotherapy. *Tissue Antigens*, **70**, 265–271.

Dönnes,P. and Kohlbacher,O. (2005) Integrated modeling of the major events in the MHC class I antigen processing pathway. *Protein Sci.*, **14**, 2132–2140.

Dönnes,P. and Kohlbacher,O. (2006) SVMHC: a server for prediction of MHC-binding peptides. *Nucleic Acids Res.*, **34**, W194–W197.

Doytchinova,I. *et al*. (2004) Transporter associated with antigen processing preselection of peptides binding to the MHC: a bioinformatic evaluation. *J. Immunol.*, **173**, 6813–6819.

Feldhahn,M. *et al*. (2008) Epitoolkit–a web server for computational immunomics. *Nucleic Acids Res.*, **36**, W519–W522.

Halling-Brown,M. *et al*. (2009) Computational grid framework for immunological applications. *Philos. Trans. R. Soc. A*, **367**, 2705–2716.

Nielsen,M. *et al*. (2007) NetMHCpan, a method for quantitative predictions of peptide binding to any HLA-A and -B locus protein of known sequence. *PLoS ONE*, **2**, e796.

Nielsen,M. *et al*. (2008) Quantitative predictions of peptide binding to any HLA-DR molecule of known sequence: NetMHCIIpan. *PLoS Comput. Biol.*, **4**, e1000107.

Parker,K.C. *et al*. (1994) Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. *J. Immunol.*, **152**, 163–175.

Rammensee,H. *et al*. (1999) SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics*, **50**, 213–219.

Sturniolo,T. *et al*. (1999) Generation of tissue-specific and promiscuous HLA ligand databases using DNA microarrays and virtual HLA class II matrices. *Nat. Biotechnol.*, **17**, 555–561.

Toussaint,N.C. and Kohlbacher,O. (2009) OptiTope—a web server for the selection of an optimal set of peptides for epitope-based vaccines. *Nucleic Acids Res.*, **37**, W617–W622.

Toussaint,N.C. *et al*. (2008) A mathematical framework for the selection of an optimal set of peptides for epitope-based vaccines. *PLoS Comput. Biol.*, **4**, e1000246.