BIOINFORMATICS

Vol. 00 no. 00 2014
Pages 1–3

Chemically Aware Model Builder (camb): An R package for property and bioactivity modeling of small molecules

Daniel S. Murrell [1*], Isidro Cortes-Ciriano [2,*], Gerard J.P. van Westen [3], Ian P. Stott[4], Andreas Bender[1,†], Thérèse E. Malliavin [2,†], Robert C. Glen[1,†]

[1]Unilever Centre for Molecular Science Informatics, Department of Chemistry, University of Cambridge, Lensfield Road, Cambridge CB2 1EW, United Kingdom.
[2]Unite de Bioinformatique Structurale, Institut Pasteur and CNRS UMR 3825, Structural Biology and Chemistry Department, 2528, rue Dr. Roux, 75 724 Paris, France.
[3]European Molecular Biology Laboratory European Bioinformatics Institute Wellcome Trust Genome Campus, Hinxton, United Kingdom.
[4]Unilever Research, Bebington, UK

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

Summary: *camb* is an R package that provides an environment for the rapid generation of quantitative structure-property models for small molecules (QSAR, QSPR, QSAM, PCM). It is aimed at both advanced and beginner R users. Its capabilities include standardisation of the representation of chemical structures, computation of 905 two-dimensional and 14 fingerprint type descriptors for small molecules, 8 types of amino acid descriptors, 13 whole protein sequence descriptors, filter methods for feature selection, generation of predictive models (R package *caret*), as well as techniques to ensemble these models (R package *caretEnsemble*). Results can be visualised through high-quality, customisable plots (R package *ggplot2*).

Availability: *camb* is written in R, C++, Python and Java and is available open source at https://github.com/cambDI/camb. Two tutorials demonstrating property (QSPR) and bioactivity modelling (PCM) are available in the supplementary information.

Contact: Therese Malliavin: terez@pasteur.fr, Andreas Bender: ab454@cam.ac.uk, Robert Glen: rcg28@cam.ac.uk

1 INTRODUCTION

The advent of high-throughtput technologies over the last two decades has led to a vast increase of compound and bioactivity databases (Bender, 2010; Gaulton *et al.*, 2011; Wang *et al.*, 2012). This increase in the amount of chemical and biological information has been exploited by developing fields in drug discovery such as quantitative structure activity relationships (QSAR), quantitative structure property relationships (QSPR), quantitative sequence-activity modelling (QSAM), or proteochemometric modelling (PCM) (van Westen *et al.*, 2011).

The R programming language provides a flexible platform for statistical analyses (R Core Team, 2013), and its applicability in medicinal chemistry has been reviewed elsewhere (Mente and Kuhn, 2012). Although R is extensively used in diverse biological domains, *e.g.* genomics (Gentleman *et al.*, 2004), the availability of R packages for cheminformatics and medicinal chemistry is limited. Nonetheless, R still constitutes the most frequent choice in the medicinal chemistry literature for compound bioactivity and property modelling (Mente and Kuhn, 2012). In general, these studies share a common algorithmic structure, which can be summarised in 4 model generation steps: (i) compound standardisation, (ii) descriptor calculation, (iii) preprocessing, feature selection, model training and validation, and (iv) bioactivity/property prediction for new molecules.

Currently available R packages each provide the capability for only subsets of the above mentioned steps. For instance, the R packages *chemmineR* (Cao *et al.*, 2008) and *rcdk* (Guha, 2007) enable the manipulation of SDF and SMILES files, the calculation of physicochemical descriptors, the clustering of molecules, and the retrieval of compounds from PubChem (Wang *et al.*, 2012). On the machine learning side, the *caret* package provides a unified platform for the training of machine learning models (Kuhn, 2008).

However, currently no unified R environment exists which caters to all the steps required to go from molecular structures to a trained model ready to use on new molecules. Here, we present the R package *camb*: **C**hemically **A**ware **M**odel **B**uilder, which aims to address the current lack of an R framework encompassing all four steps mentioned above. The package has been conceived in a way that users with minimal programming skill are able to generate competitive predictive models and high-quality plots under default operation. However, each function can be used with non-default parameters to fulfil the more versatile needs of more experienced users.

Overall, *camb* enables the generation of predictive models (QSAR, QSPR, QSAM and PCM) starting with: chemical structure files, protein sequences (if required), and the associated properties

*Equal contributors

†to whom correspondence should be addressed

or bioactivities. Moreover, *camb* is the first R package that enables the manipulation of chemical structures utilising Indigo's C API (Indigo, 2013), and the calculation of: (i) molecular fingerprints and 1- and 2-dimensional topological descriptors calculated using the PaDEL-Descriptor Java library (Yap, 2011), (ii) hashed and unhashed Morgan fingerprints (Rogers and Hahn, 2010), and (iii) 8 types of amino acid descriptors (implemented in this package for the first time). Two case studies illustrating the application of *camb* for QSPR modelling (solubility prediction) and PCM are available in the online supplementary information. In the following section we detail the main functionalities provided by *camb*.

2 DESCRIPTION

This section describes the tools provided by *camb* for (i) compound standardisation, (ii) molecular and protein descriptor calculation, (iii) preprocessing and feature selection, model training, visualisation and validation, and (iv) bioactivity/property prediction for new molecules.

2.1 Compound stardardization

In order to represent all molecules in a given dataset in the same way (compound standardisation), *camb* provides the function *StandardiseMolecules* which utilises Indigo's C API (Indigo, 2013). SDF and SMILES formats are provided as molecule input options. Any molecules that Indigo fails to parse are removed during the standardisation step. The maximum number of each halogen that a compound can comprise of in order to pass the standardisation process can be defined by the user. Additional arguments of this function include the removal of inorganic molecules or those compounds with a molecular mass above or below a given thresholds. *camb* makes use of Indigo's InChI (InChI, 2013) plugin to represent all tautomers in the same SMILES representation by converting to InChI, discarding tautomeric information, and converting back to SMILES.

2.2 Descriptor calculation

Currently, *camb* supports the calculation of compound descriptors and fingeprints via PaDEL-Descriptor (Yap, 2011), and extended connectivity circular fingerprints (Rogers and Hahn, 2010) as implemented in the RDkit (Landrum, 2006). The function *GeneratePadelDescriptors* permits the calculation of 905 1- and 2-dimensional descriptors and 10 PaDEL fingerprints, namely: CDK fingerprints, CDK extended fingerprints, Kier-Hall E-state fragments Hall and Kier (1995), CDK graph only fingerprints, MACCS fingerprints, Pubchem fingerprints, Substructure fingerprints and counts (O'Boyle *et al.*, 2011), Klekota-Roth fingerprints and counts (Klekota and Roth, 2008).

Morgan fingerprints can be computed with the function *MorganFPs* through the python library RDkit (Landrum, 2006). Hashed fingerprints can be generated as *binary*, recording the presence or absence of each substructure, or *count based*, recording the number of occurrences of each substructure. Additionally, this function computes unhashed (keyed) fingerprints, where each substructure in the dataset is assigned a unique position in a binary fingerprint of length equal to the number of substructures existing in the dataset.

Since the positions of substructures in the unhashed fingerprint depend on the dataset, the function *MorganFPs* allows calculation of unhashed fingerprints for new compounds using a basis defined by the substructures present in the training dataset. This ensures that substructures in new compounds map to the same locations on the fingerprint and allows enhanced model interpretation by noting which exact substructures are deemed important by the learning algorithm.

camb also enables the calculation of 13 types of whole protein sequence descriptors from UniProt identifiers (Xiao and Xu, 2014), as well as the calculation of 8 types of amino acid descriptors (van Westen *et al.*, 2013).

2.3 Model training and validation

Prior to model training, descriptors often need to be statistically preprocessed (Andersson *et al.*, 2011). To this end, several functions (see package documentation and tutorials) are provided, *e.g.* the removal of non-informative predictors or their conversion to z-scores.

camb invokes the R package *caret* to set up cross validation frameworks and train individual machine learning models. *caret* provides a common interface to the most popular machine learning package that exist in R. These include learning methods in Bagging, Bayesian Methods, Boosting, Boosted Trees, Elastic Net, MARS, Gaussian Processes, K Nearest Neighbour, Principal Component Regression, Radial Basis Function Networks, Random Forests, Relevance Vector Machines, Support Vector Machines among others. Additionally, two ensemble modelling approaches, namely greedy and stacking optimisation, have been integrated from the R package *caretEnsemble* (Mayer, 2013), which allow a combination of models to be used allowing for more accurate models to be built. Statistical metrics for model validation have also been included (Golbraikh and Tropsha, 2002).

Model performance visualization functionality is provided. All plots are generated using the R package *ggplot2* (Wickham, 2009). Default options for plotting functions allow the generation of high-quality plots, and in addition, the layer-based structure of ggplot objects allows for further optimisation by the addition of customisation layers. Visual depiction of compounds is also possible with the function *PlotMolecules*, utilising Indigo's C API. Visualization functions are exemplified in the tutorials provided in the supplementary information.

2.4 Predictions for new molecules

One of the major benefits of having all the tools available in one framework is that it makes it easy to run new molecules through exactly the same routines that the training set of molecules was subjected to before the model training process. *camb* provides the option to provide an external test set of molecules along with a trained model, and outputs predictions on an external test set. An example of this is shown in the QSPR tutorial.

3 CONCLUSIONS

In silico predictive models have proved valuable for the optimisation of compound potency, selectivity and safety profiles. In this context, *camb* provides a complete and open framework to (i)

compound standardisation, (ii) molecular and protein descriptor calculation, (iii) preprocessing and feature selection, model training, visualisation and validation, and (iv) bioactivity/property prediction for new molecules.

4 ACKNOWLEDGEMENTS

ICC thanks the Paris-Pasteur International PhD Programme for funding. ICC and TM thank CNRS, Institut Pasteur and ANR bipbip for funding. DSM and RCG thanks Unilever for funding. GvW thanks EMBL (EIPOD) and Marie Curie (COFUND) for funding. AB thanks Unilever and the European Research Commission (Starting Grant ERC-2013-StG 336159 MIXTURE) for funding.

REFERENCES

Andersson, C. R., Gustafsson, M. G., and Strmbergsson, H. (2011). Quantitative chemogenomics: machine-learning models of protein-ligand interaction. *Current topics in medicinal chemistry*, **11**(15), 1978–1993.

Bender, A. (2010). Databases: Compound bioactivities go public. *Nature Chemical Biology*, **6**(5), 309–309.

Cao, Y., Charisi, A., Cheng, L. C., Jiang, T., and Girke, T. (2008). Chemminer: a compound mining framework for r. *Bioinformatics*, **24**(15), 1733–1734.

Gaulton, A., Bellis, L. J., Bento, A. P., Chambers, J., Davies, M., Hersey, A., Light, Y., McGlinchey, S., Michalovich, D., Al-Lazikani, B., and Overington, J. P. (2011). ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic Acids Research*, **40**(D1), D1100–D1107.

Gentleman, R. C., Carey, V. J., Bates, D. M., and others (2004). Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, **5**, R80.

Golbraikh, A. and Tropsha, A. (2002). Beware of q2! *Journal of molecular graphics & modelling*, **20**(4), 269–276.

Guha, R. (2007). Chemical informatics functionality in r. *Journal of Statistical Software*, **18**(6).

Hall, L. H. and Kier, L. B. (1995). Electrotopological state indices for atom types: A novel combination of electronic, topological, and valence state information. *Journal of Chemical Information and Computer Sciences*, **35**(6), 1039–1045.

InChI (2013). IUPAC - international union of pure and applied chemistry: The IUPAC international chemical identifier (InChI).

Indigo (2013). Indigo cheminformatics library.

Klekota, J. and Roth, F. P. (2008). Chemical substructures that enrich for biological activity. *Bioinformatics*, **24**(21), 2518–2525. PMID: 18784118.

Kuhn, M. (2008). Building predictive models in r using the caret package. *Journal of Statistical Software*, **28**(5), 1–26.

Landrum, G. (2006). Rdkit: Open-source cheminformatics.

Mayer, Z. (2013). caretensemble: Framework for combining caret models into ensembles. [r package version 1.0].

Mente, S. and Kuhn, M. (2012). The use of the r language for medicinal chemistry applications. *Current topics in medicinal chemistry*, **12**(18), 1957–1964.

O'Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., and Hutchison, G. R. (2011). Open babel: An open chemical toolbox. *Journal of Cheminformatics*, **3**(1), 33. PMID: 21982300.

R Core Team (2013). R: A language and environment for statistical computing.

Rogers, D. and Hahn, M. (2010). Extended-connectivity fingerprints. *Journal of chemical information and modeling*, **50**(5), 742–754.

van Westen, G. J., Swier, R. F., Cortes-Ciriano, I., Wegner, J. K., Overington, J. P., Ijzerman, A. P., van Vlijmen, H. W., and Bender, A. (2013). Benchmarking of protein descriptor sets in proteochemometric modeling (part 2): Modeling performance of 13 amino acid descriptor sets. *J. Cheminf*, **5**(1), 42.

van Westen, G. J. P., Wegner, J. K., IJzerman, A. P., van Vlijmen, H. W. T., and Bender, A. (2011). Proteochemometric modeling as a tool to design selective compounds and for extrapolating to novel targets. *Med. Chem. Commun.*, **2**, 16–30.

Wang, Y., Xiao, J., Suzek, T. O., Zhang, J., Wang, J., Zhou, Z., Han, L., Karapetyan, K., Dracheva, S., Shoemaker, B. A., Bolton, E., Gindulyte, A., and Bryant, S. H. (2012). PubChem's BioAssay database. *Nucleic acids research*, **40**(Database issue), D400–412.

Wickham, H. (2009). ggplot2: elegant graphics for data analysis.

Xiao, N. and Xu, Q. (2014). protr: Protein sequence descriptor calculation and similarity computation with r. R package version 0.2-1.

Yap, C. W. (2011). PaDEL-Descriptor: an open source software to calculate molecular descriptors and fingerprints (v2.16). *Journal of Computational Chemistry*, **32**(7), 1466–1474.