Neural Computing and Applications (2020) 32:323–334
https://doi.org/10.1007/s00521-018-3655-2

S.I. : IWANN2017: LEARNING ALGORITHMS WITH REAL WORLD APPLICATIONS

Combining feature engineering and feature selection to improve the prediction of methionine oxidation sites in proteins

Francisco J. Veredas[1] · Daniel Urda[2] · José L. Subirats[1] · Francisco R. Cantón[3] · Juan C. Aledo[3]

Received: 22 December 2017 / Accepted: 24 July 2018 / Published online: 3 August 2018

Abstract

Methionine is a proteinogenic amino acid that can be post-translationally modified. It is now well established that reactive oxygen species can oxidise methionine residues within living cells. For a long time, it has been thought that such a modification represents merely an inevitable damage derived from aerobic metabolism. However, several authors have begun to contemplate a possible role for this methionine modification in cell signalling. During the last years, a number of proteomic studies have been carried out with the purpose of detecting proteins containing oxidised methionines. Although these proteomic works allow to pinpoint those methionines being oxidised, they are also arduous, expensive and time-consuming. For these reasons, computational approaches aimed at predicting methionine oxidation sites in proteins become an appealing alternative. In the current work, we address methionine oxidation prediction by combining computational intelligence methods with feature engineering and feature selection techniques to improve the efficacy of several machine learning models, while reducing the number of input characteristics needed to get high accuracy rates. We compare random forests, support vector machines, neural networks and flexible discriminant analysis models. Random forests give the best AUC (0.8124 ± 0.0334) and accuracy rates (0.7590 ± 0.0551) by using only a reduced set of 16 characteristics. These results surpass the outcomes of previous works. In addition, we present an end-user script that has been developed to take a protein ID as an input and return a list with the oxidation state of all the methionine residues found in the analysed protein. Finally, to illustrate the applicability of this tool, we have selected the human α1-antitrypsin protein as a case study. This protein was selected because it was not present among the set of proteins used to build up the predictive models but the protein has been well characterised experimentally in terms of methionine oxidation. The prediction returned by our script fully matches the empirical evidence. Out of the nine methionine residues found in this protein, our model predicts the oxidation of only two of them, M351 and M358, which have been reported, on the base of mass spectrometry analyses, to be particularly susceptible to oxidation.

Keywords Protein prediction · Post-translational modification · Methionine oxidation · Predictive computational model

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s00521-018-3655-2) contains supplementary material, which is available to authorised users.

✉ Francisco J. Veredas
franveredas@uma.es

Daniel Urda
daniel.urda@uca.es

José L. Subirats
jlsubirats@lcc.uma.es

Francisco R. Cantón
frcanton@uma.es

Juan C. Aledo
caledo@uma.es

[1] Dpto. Lenguajes y Ciencias de la Computación, Universidad de Málaga, 29071 Málaga, Spain

[2] Dpto. de Ingeniería Informática, Escuela Superior de Ingeniería, Universidad de Cádiz, 11519 Puerto Real, Cádiz, Spain

[3] Dpto. de Biología Molecular y Bioquímica, Facultad de Ciencias, Universidad de Málaga, 29071 Málaga, Spain

1 Introduction

Post-translational modifications (PTM) are changes that some amino acid residues can experience after proteins have been synthesised by ribosomes. On the one hand, PTMs change the physico-chemical properties of the protein; hence, they can affect the function that the protein accomplishes in the cell. On the other hand, PTMs are reversible reactions, as those modified amino acids can be reverted back to their original state; in this way, protein regulation can be fine-tuned. Some PTMs are likely to occur when some specific oxidising agents react with certain amino acids and turn them into their corresponding modified form. One of these oxidising agents is hydrogen peroxide (H_2O_2), that is part of the group of reactive oxygen species (ROS). ROS are compounds generated in all the cells subject to an aerobic metabolism. Although ROS are well known as damaging agents involved in aerobic metabolism [1], they can also take part in regulatory PTM reactions. As it has been shown in recent studies [7, 32], H_2O_2 can act as an effective cellular messenger by bringing about reversible PTMs. Cysteine and methionine, the two sulphur-containing amino acids present in proteins, are likely to experience PTMs by being oxidised by H_2O_2. These two amino acids contain a functional group that can serve as a nucleophile in the modification reaction. For its part, methionine can be oxidised into methionine sulfoxide (MetO) by the addition of an oxygen atom to its sulphur atom. This oxidation can be reverted by enzyme-catalysed reduction reactions [3]: MetO is reduced back to methionine by certain enzymes named methionine sulfoxide reductases, which are known to be present in all aerobic cells [22]. However, the role of MetO residues in cellular redox-dependent regulation remains largely unexplored [21].

Methionine oxidation can affect the activity and stability of the protein [18], as it has been shown that the direct oxidation of specific methionine residues can either down-regulate [17, 29] or up-regulate [12, 13, 30] protein function. Furthermore, methionine oxidation can also affect protein function indirectly, by coupling oxidative signals to other sorts of PTMs, such as protein phosphorylation [28].

The acknowledgement of methionine oxidation as a probable mechanism to the redox-dependent modulation of protein activity and cellular mechanisms, has motivated recent proteomics studies. In this sense, proteome-wide studies of methionine oxidation have identified a large number of proteins as potential targets of oxidative signals, in both Arabidopsis [19] and human [16]. Moreover, these proteomics approaches have determined the precise sites of methionine oxidation on the target proteins. However, these experimental studies are often excessively expensive, time-consuming and arduous. It is in this scenario where the development of computational methods for predicting methionine oxidation sites becomes a highly valuable alternative.

In the field of protein phosphorylation, which can be considered the most widely studied PTM, the use of computational methods for prediction of phosphorylation sites in proteins has become a very popular approach [8, 31, 34]. Unfortunately, to the best of our knowledge, there are no such methods for methionine oxidation site prediction, and only some efforts have been recently devoted by these same authors to this purpose [2, 33]. Thus, in the current study, we have addressed this issue by combining computational intelligence models with feature engineering and feature selection strategies aimed at improving the efficacy rates obtained in our previous work. However, as the performance results given in [33] for the test dataset were not computed with resampling methods (e.g. bootstrap), they are not fairly comparable to other similar studies, such as the current one. On the contrary, in [2] we used different machine learning (ML) models—such as random forests (RF), support vector machines (SVM) and neural networks (NN)—and computed their performance rates from 100 bootstrap resamples. RFs gave maximum mean AUC (area under the ROC curve) and accuracy rates of 0.7998 and 0.7468, respectively (see Aledo et al. [2, Table 2—testing set data]). Given the implicit difficulty of predicting methionine oxidation accurately, a slight improvement of those efficacy rates could be of significant importance to drive future empirical work. For this reason, in this paper we combine feature engineering and selection techniques to slightly—but significantly—improve the performance of several ML approaches when we use them to predict methionine oxidation sites in proteins.

Finally, to provide a tool for methionine oxidation prediction may provide an end-user R script [27] that is publicly available at github.com/fveredas/MOPM was developed. By using a RF trained with methionine oxidation data (see Sect. 2), the script takes the PDB ID of a protein as an input (i.e. the 4-character unique identifier of that protein in the Protein Data Bank) and returns a table with the prediction of oxidation for all the methionine residues found in that protein. To show the applicability of this tool, a real case study is addressed and described in Sect. 4.

The rest of the paper is organised as follows. Section 2 reports the materials and methods used in this study: It describes the datasets used for the experiments; the procedure followed to extract features from the primary and tertiary structures of proteins; the feature engineering strategies used to get a more effective feature set; the ML methods and techniques used to build and tune the models for methionine oxidation prediction; and finally, it gives the details of the performance measures and the validation strategies used to assess and compare the prediction efficacy of the models. Since one of the main objectives of this study is to improve the results obtained in our previous work, these materials and methods are mostly similar to those in [2]. Next, the results obtained with four different ML models are shown in Sect. 3 and analytically compared to those in our previous work. Section 4 describes an application of the methionine oxidation predictive model developed to address a real case study. Finally, Sect. 5 presents the conclusions and limitations of this study.

2 Materials and methods

As previously stated, one of the aims of this work is to improve the performance of the models and strategies for methionine oxidation prediction designed in our previous work in [2]. For this reason, some parts of the materials (datasets) and methods (ML models, resampling strategies, class-imbalance addressing, etc.) published in that work are reused in the current study as a starting point for further development of more effective approaches. Thus, in order to avoid redundancy, we just cite and summarise in this section the most important methodological points shared by both studies. Moreover, we explain in more detail the novelties and methodological contributions of our current work with respect to that previously published, which mainly consist in the feature selection and feature engineering strategies we use herein to improve methionine oxidation prediction. Since another ML model, originated from the discriminant analysis paradigm, has also been introduced in this study to complement the ML-model set analysed comparatively, it is also described in detail in the following subsections.

2.1 Feature extraction, engineering and selection

In Fig. 1, the complete process of feature extraction, selection and engineering to get the five different datasets used in this study, i.e. *1D*, *3D*, *All*, *SBF* and *SBF-Int*, is outlined. The different stages of this process are described in detail in the following subsections.

2.1.1 Feature extraction

In [2, Section *Datasets*], we described the original experimental dataset supplied by Ghesquière et al. [16], which was obtained from complex mass spectrometry experiments aimed at identifying an extensive set of oxidised methionine residues. That original dataset was composed of more than 1600 different proteins, for which over 2000 methionine residues were experimentally observed as being oxidation sensitive. From this dataset, we selected a subset of 774 proteins that exhibited a degree of oxidation equal or greater than 20%. This threshold was arbitrary established to discriminate methionine residues that appeared as oxidised—in at least 20% of the population of protein molecules of the experimental set—from those others that did not. After removing redundancy and filtering out low quality structures, we assembled a collection of 113 proteins of known structure, containing 975 methionine residues from which 122 out were oxidation-prone. Finally, this dataset was used to extract a set of primary and tertiary characteristics that are explained in detail in [2, Section *Feature Extraction*] and summarised here briefly. This dataset can be downloaded from github.com/fveredas/PredictionOfMethionineOxidationSites.

For each methionine residue being analysed, 76 features from both the *primary* (1D) and the *tertiary* (3D) structure of the protein were extracted. These features are summarised as follows:

76 independent variables (input):

- 52 protein's *primary-structure* features:
  - 40 *distance* variables:
    - NT_X: distance (number of positions in the primary structure) from the analysed methionine to the closest X residue towards the N-terminus.
    - CT_X: distance (number of positions in the primary structure) from the analysed methionine to the closest X residue towards the C-terminus.

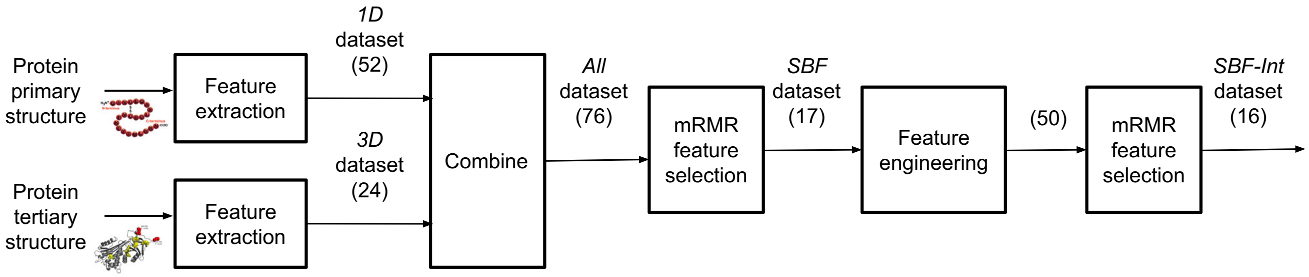


Fig. 1 Five different datasets. Diagram of the complete process or feature extraction, selection and engineering. Five datasets, each one composed of different feature sets, were used in this study: *1D*, *3D*, *All*, *SBF* and *SBF-Int*. The number of input characteristics of each dataset is shown in parentheses

  - 8 *phosphorylation-related* features, with X being either S (serine), T (threonine) or Y (tyrosine):
    - `Met2X`: distance between the analysed methionine and the closest X phospho-acceptor, computed as $\min(NT\_X, CT\_X)$.
    - `Met2X_PTM`: distance between the analysed methionine and the closest X phosphosite.
    - `closer10res`: number of phosphorylatable residues in a radius of 10 amino acids from the analysed methionine.
    - `away.ptm`: calculated according to $\min_{X \in \{S,T,Y\}}(Met2X\_PTM)$.
  - 4 *conservation* variables obtained after computing multiple sequence alignment (MSA):
    - `entropy`: Shannon base-21 entropy.
    - `mean.entropy`: mean *entropy* at all positions of the analysed protein.
    - `sd.entropy`: standard deviation of *entropy* at all positions of the analysed protein.
    - `fM`: relative frequency of methionine at the position of analysed methionine, after multiple sequence alignment (MSA).
- 24 protein's *tertiary-structure* features:
  - 4 *phosphorylation* variables:
    - `closest.ptm.pdb`: distance in Å between the methionine and the closest serine, threonine or tyrosine experimentally shown to be phosphorylated.
    - `closest.ptm.chain`: distance in Å between the methionine and the closest serine, threonine or tyrosine experimentally shown to be phosphorylated and present in the same polypeptide chain that the methionyl residue.
    - `closer10A.pdb`: number of phosphorylatable sites within a sphere of radius 10 Å centred at methionine.
    - `closer10A.chain` number of phosphorylatable sites, found on the same polypeptide chain, within a sphere of radius 10 Å centred at the methionine.
  - 16 *S-aromatic* variables, with X being either Y (tyrosine), F (phenylalanine) or W (tryptophan):
    - `Xd.pdb`: distance in Å between the sulphur atom of the analysed methionine and nearest X aromatic residue.
    - `Xd.chain`: distance in Å between the sulphur atom of the analysed methionine and nearest X aromatic residue within the same polypeptide chain.
    - `nX.pdb`: number of X residues at a distance $<$ 7Å from the Met.
    - `nX.chain`: number of X residues, within the same polypeptide molecule, at a distance $<$ 7Å from the Met.
    - `numberBonds.chain`: computed as $\sum_{X \in \{Y,F,W\}} nX.chain$.
    - `numberBonds.pdb`: computed as $\sum_{X \in \{Y,F,W\}} nX.pdb$.
    - `closestAro.chain`: computed as $\min_{X \in \{Y,F,W\}}(Xd.chain)$.
    - `closestAro.pdb`: computed as $\min_{X \in \{Y,F,W\}}(Xd.pdb)$.
  - 4 *accessibility* properties:
    - `SASA.chain`: solvent accessible surface area given the atomic coordinates of the single polypeptide chain harbouring the methionine.
    - `SASA.pdb`: solvent accessible surface area given the atomic coordinates of the whole protein.
    - `Bfactor`: the B factor of the sulphur atom from the methionine of interest extracted from the PDB file.
    - `dpx`: depth of the sulphur atom from the considered Met.

Finally, the dependent variable (output) was defined as:

- `oxidised`: binary variable indicating whether the methionine is oxidised ($>$ 20%) or not.

2.1.2 Feature engineering and selection

As it has been summarised above, for each methionine residue present in a certain protein a total of 76 characteristics were evaluated, of which 52 were derived from the primary structure, while the remaining 24 features were extracted from the tertiary structure of the protein. These three collections of features will be referred to as *All* (76 features), *1D* (52 features) and *3D* (24 features), respectively. With the purpose of comparing them with the results obtained in our previous works [2, 33], in Sect. 3 we have included the performance analysis of four ML predictive models when applied to these three different feature sets.

In addition to these three collections of characteristics, we have also used a feature subset composed of the most relevant features. For this purpose, we used the *minimum redundancy maximum relevance* (*mRMR*) method [11] to rank the importance of the 76 original features. The resulting ranking is obtained by means of a method that is based on the concept of *mutual information*. Thus, a score for each feature is computed as a trade-off between the relevance to the output (oxidised) and the redundancy between the input variables. Basically, the *mRMR* filtering algorithm defines a ranking score as follows: for each input feature to be tentatively selected, its score is computed as the mutual information between the target (oxidised) and this tentative feature, minus the average mutual information of all previously selected features and this tentative feature. In this way, a final subset composed of those 17 features with *mRMR* scores greater than zero was identified as the "optimal" feature set. We named it the *SBF* (*Selection of Best Features*) feature set.

Subsequently, we have selected the two first variables in the *mRMR* ranking, i.e. SASA.pdb and NT_M predictors, to compute all the interaction terms that result from multiplying these two variables by all the 17 variables in the *SBF* feature set. In this way, we obtained 33 new interaction variables that added up to the *SBF* feature set to give a total of 50 variables. After that, we applied again the *mRMR* filtering algorithm to these 50 input variables and ranked them again by their *mRMR* score. This ranking gave us a new collection of 16 features comprised of those variables with *mRMR* scores greater than zero. We named this new collection of 16 variables the *SBF-Int* feature set.

In Table 1, the lists of input variables for the *SBF* and *SBF-Int* feature sets are shown sorted by *mRMR* score (see above, as well as Aledo et al. [2], for a detailed explanation of these features).

Finally, the data for each resulting input variables in these five feature sets (i.e. *1D*, *3D*, *All*, *SBF* and *SBF-Int*) were analysed by visually inspecting its distribution histogram. Those variables showing right- or left-skewed histograms were *log*-transformed to get more symmetric distributions and improve their predictive power [35].

2.2 Machine learning models

Three of the four machine learning methods used in the current study to predict methionine oxidation—i.e. RF, SVM and NN—were already introduced and explained in detail in [2, Section *Machine learning methods*]. Furthermore, for those three ML models, a deep view of the strategies for hyper-parameter tuning was also given in that paper. Although those three models and model-selection approaches have been used in the current study, they are not described here again but only summarised in Table 2 in order to avoid redundancy.

Moreover, to complement those three ML models used in [2], we have also included in this study a fourth model, originated from a different ML paradigm: flexible discriminant analysis (FDA). Many classification models, such as ridge regression, the lasso or adaptive regression splines (MARS) [15], can be extended to create discriminant variables. In particular, MARS can be used to create a set of discriminant functions that are nonlinear combinations of the original predictors. This conceptual paradigm is referred to as FDA [24]. In this study, we have followed a bagging approach for FDA, which uses MARS basis functions to compute a FDA model for each bootstrap sample. The only parameter to be tuned for model selection was the *maximum number of terms* (including intercept) in the pruned model [15], usually known as nprune (see Table 2), which is used to enforce an upper bound on the model size. The optimal nprune parameter was chosen as that in the range $\{1,\ldots,25\}$ that gave the highest AUC rate. The maximum degree of interaction (Friedman's *mi*) was fixed to 1; thus, an additive model (i.e. no interaction terms) was used.

2.2.1 Model training and validation

The main methodological aspects regarding model selection, training and validation can be summarised as follows:

- For model selection, training and validation (see Tables 3 and 4), the dataset was split into three independent subsets:
    - *training* 80% (98 'positive'; 683 'control')
    - *evaluation* 6.66% (8 'positive'; 57 'contro')
    - *testing* 13.33% (16 'positive'; 113 'control')

Table 1 Feature engineering *SBF* and *SBF-Int* sets

SBF (17)	SASA.pdb (5.09e-02), NT_M (1.37e−02), Met2Y (7.92e−03), Met2S_PTM (3.76e−03), NT_D (3.39e−03), CT_Q (3.03e−03), CT_F (1.45e−03), NT_E (1.58e−03), CT_H (1.2e−03), numberBonds.chain (2.2e−03), Bfactor (1.39e−03), Met2S (1.26e−03), CT_C (1.13e−03), CT_E (8.16e−04), NT_R (3.08e−04), NT_H (2.18e−04), dpx (7.17e−04)
SBF-Int (16)	SASA.pdb × NT_M (6.72e−02), Met2Y (6.86e−03), NT_M × Met2S (3.79e−03), NT_D (3.48e−03), CT_Q (2.9e−03), Met2S_PTM (3.72e−03), NT_E (1.26e−03), numberBonds.chain (2.43e−03), Bfactor (1.47e−03), CT_C (1.04e−03), NT_M × CT_F (1.38e−03), CT_H (1.29e−03), CT_E (7.74-04), NT_H (3.3e−04), SASA.pdb × Met2Y (4.07e−03), NT_R (3.7e−05)

An interaction term between *A* and *B* variables is represented as $A \times B$

mRMR scores are given in brackets

Table 2 Model selection summary

Model	Tuned hyper-parameter	Fitted value (tenfold CV—five repetitions, AUC metric)
RF	`mtry`	$\lfloor\sqrt{number\ of\ predictors}\rfloor$
	`#trees`	1000
SVM (RBF kernel)	σ	Fixed by empirical methods [6]
	`cost`	Best of $\{2^i\}_{i=-2}^{9}$
NN (1 hidden layer)	`size`	Best of $\{1, 2, \ldots, 20\}$
	`decay`	Best of $\{0.001, 0.01, 0.1, 0.5\}$
FDA	`nprune`	Best of $\{1, 2, \ldots, 25\}$

See Aledo et al. [2] for further details

Table 3 Performance rates with four different ML models

Feature set	AUC	Accuracy	Sensitivity	Specificity	*F*-measure	MCC
RF						
1D (52)	0.7174	0.6434	0.8125	0.6195	0.3611	0.2873
3D (24)	0.8006	0.6667	0.8750	0.6372	0.3944	0.3414
All (76)	0.8429	0.7674	0.8125	0.7611	0.4643	0.4087
SBF (17)	0.8216	0.6434	0.8125	0.6195	0.3611	0.2873
SBF-Int (16)	0.8355	0.8605	0.6250	0.8938	0.5263	0.4547
SVM						
1D (52)	0.7024	0.5581	0.7500	0.5310	0.2963	0.1852
3D (24)	0.5996	0.7054	0.1875	0.7788	0.1364	-0.0270
All (76)	0.7522	0.7209	0.6250	0.7345	0.3571	0.2562
SBF (17)	0.8042	0.7209	0.7500	0.7168	0.4000	0.3246
SBF-Int (16)	0.7445	0.7054	0.6875	0.7080	0.3667	0.2750
NN						
1D (52)	0.6571	0.5116	0.6875	0.4867	0.2588	0.1151
3D (24)	0.7793	0.7209	0.7500	0.7168	0.4000	0.3246
All (76)	0.8164	0.7364	0.7500	0.7345	0.4138	0.3408
SBF (17)	0.8252	0.7132	0.8125	0.6991	0.4127	0.3504
SBF-Int (16)	0.8346	0.8372	0.5000	0.8850	0.4324	0.3437
FDA						
1D (52)	0.6626	0.4884	0.8125	0.4425	0.2826	0.1708
3D (24)	0.8595	0.7597	0.8125	0.7522	0.4561	0.3998
All (76)	0.8523	0.7597	0.6250	0.7788	0.3922	0.2993
SBF (17)	0.8274	0.7829	0.6875	0.7965	0.4400	0.3621
SBF-Int (16)	0.8296	0.7829	0.7500	0.7876	0.4615	0.3951

- Six performance measures of the out-of-bag (OOB) samples for tenfold cross-validation with five repetitions (50 resamplings) were computed: *AUC*, *accuracy*, *sensitivity*, *specificity*, *F-measure*, and *Matthews-Correlation-Coefficient (MCC)*.
- For bootstrap resampling (see Table 5), 100 random resamples were generated and tenfold cross-validation (with five repetitions) was used to train and fit each model (RF, SVM, NN and FDA).
- The *caret* R package [23, 27] (R version 3.4.1) was used for model fitting with SVM (package *kernlab* [20]), NN (package *RSNNS* [4]), RF (package *randomForest* [26]) and FDA (package *glmnet* [14]).
- Class imbalance was counteracted by determining alternative ROC curve cut-off points by using an independent *evaluation* dataset from which we extract the point on the ROC curve that is closest to the model with 100% sensitivity and 100% specificity [22] (as it is illustrated in Fig. 2).

Table 4 Model tuning. Best hyper-parameters

Feature set	cut	mtry	Number trees
RF			
1D	0.4155	7	1000
3D	0.3650	4	1000
All	0.4030	8	1000
SBF	0.3565	4	1000
SBF-Int	0.5040	4	1000
Feature set	**cut**	**sigma**	***C***
SVM			
1D	0.1168	0.0101	32
3D	0.1348	0.0345	32
All	0.1338	0.0070	32
SBF	0.1205	0.0336	4
SBF-Int	0.1109	0.0385	4
Feature set	**cut**	**size**	**decay**
NN			
1D	0.1210	13	0.001
3D	0.1744	3	0.001
All	0.2048	1	0.001
SBF	0.1652	11	0.001
SBF-Int	0.1745	18	0.001
Feature set	**cut**	**degree**	**nprune**
FDA			
1D	0.09047	1	20
3D	0.10182	1	4
All	0.13925	1	20
SBF	0.14201	1	26
SBF-Int	0.09558	1	20

3 Results

Using the sets of characteristics described in Sect. 2.1.2, we applied a number of ML predictive models, i.e. RF, SVM, NN and FDA, which were intensively tested in a comparative approach. The results obtained from these comparative studies are presented in the following paragraphs.

Table 3 shows the performance rates of four single predictive models analysed when they are applied on the testing dataset and after using the evaluation dataset for ROC-threshold adjustment (see Fig. 2 and Sect. 2.2.1). As it can be observed in the table, the best results have been obtained with the *SBF-Int* feature set that gives AUC rates above ~0.75 for all the classifiers, with RF, NN and FDA giving AUC rates greater than ~0.83. RF gave the highest accuracy rate of 0.8605, with the *SBF-Int* feature set, followed by the NN model (accuracy 0.8372), but the former shows a better balance between sensitivity and specificity than the latter. However, although FDA did not result in the best accuracy numbers (maximum 0.7829 accuracy for *SBF* and *SBF-Int* feature sets), it gave high AUC and—together with SVM (using *SBF* feature set) and NN (using *All* feature set)—one of the best balances between sensitivity and specificity rates. This last outcome makes the classical FDA model highly competitive for methionine oxidation prediction. All these results have been also supported by the MCC and F-measures rates of these models, that are also shown in Table 3.

Table 4 shows the list of parameters being tuned. For each predictive model, the best values for its tuned hyper-parameters (see Table 2) are computed as those with the highest averaged AUC for that model, via tenfold cross-validation (with five repetitions) on patterns in the training dataset. The ROC cut-offs (*cut*) obtained from the evaluation dataset after model fitting and training are also shown in the table.

In Fig. 3, the 30 most important input variables as estimated by the RF on the training set are shown along with their averaged decrease in Gini index [5]. The *All* feature set was used for this figure. As it can be seen in the figure, the most important characteristics for the RF are those 3D features regarding methionine solvent accessibility area (see Sect. 2.1.1), i.e. SASA.pdb and SASA.chain, followed by the primary variable NT_M that measures the distance from the analysed methionine to the closest methionine towards the N-terminal direction. Consistently, these SASA.pdb and NT_M input variables are the two first ones in the mRMR filter ranking (the SASA.chain variable is highly correlated with the SASA.pdb variables, so that the *mRMR* algorithm sets it aside to a further position in the ranking).

As results in Tables 3 and 4 (also illustrated by Figs. 2 and 3) correspond to single ML models applied on a same training/testing set, a more comprehensive evaluation of the predictive potential of each ML model is needed. Moreover, more robust and reliable results must be computed and shown, aimed at allowing more fair comparisons between similar studies, such as those of our previous work in [2, 33]. In this vein, Table 5 and Fig. 4 show the results from a bootstrapping strategy: for each ML model and feature set (*1D*, *3D*, *All*, *SBF* and *SBF-Int*), 100 bootstrap resamples were generated and tenfold cross-validation (with five repetitions) were used to train and fit each model (see also the p values from the paired Wilcoxon signed-rank test [9, 10, 25] in Table S1). Mean performance rates and standard deviation on the training and testing datasets (after adjustment of ROC cut-off probability on the evaluation datasets) are also shown in Table 5.

Table 5 Performance rates for four different ML approaches: mean (sd)

Feature set	AUC	Accuracy	Sensitivity	Specificity	*F*-measure	MCC
RF						
1D (52)	0.6969 (0.0428)	0.6195 (0.0608)	0.6819 (0.1292)	0.6108 (0.0822)	0.3038 (0.0432)	0.1961 (0.0566)
3D (24)	0.7625 (0.0378)	0.7014 (0.0492)	0.6945 (0.1144)	0.7020 (0.0668)	0.3620 (0.0434)	0.2753 (0.0520)
All (76)	0.7953 (0.0356)	0.7501 (0.0547)	0.6841 (0.1117)	0.7589 (0.0717)	0.4031 (0.0535)	0.3235 (0.0598)
SBF (17)	0.8024 (0.0311)	0.7357 (0.0591)	0.7008 (0.1028)	0.7404 (0.0766)	0.3966 (0.0547)	0.3174 (0.0580)
SBF-Int (16)	0.8124 (0.0334)	0.7590 (0.0551)	0.6994 (0.0994)	0.7672 (0.0714)	0.4185 (0.0558)	0.3431 (0.0584)
SVM						
1D (52)	0.6546 (0.0440)	0.6118 (0.0694)	0.5967 (0.1365)	0.6137 (0.0918)	0.2718 (0.0453)	0.1420 (0.0598)
3D (24)	0.6525 (0.0484)	0.6318 (0.0877)	0.5798 (0.1371)	0.6391 (0.1124)	0.2788 (0.0459)	0.1515 (0.0606)
All (76)	0.7649 (0.0339)	0.7066 (0.0622)	0.6569 (0.1329)	0.7135 (0.0844)	0.3542 (0.0469)	0.2623 (0.0564)
SBF (17)	0.7373 (0.0396)	0.6891 (0.0617)	0.6305 (0.1312)	0.6974 (0.0828)	0.3320 (0.0529)	0.2305 (0.0629)
SBF-Int (16)	0.7297 (0.0377)	0.6891 (0.0637)	0.6288 (0.1132)	0.6983 (0.0837)	0.3324 (0.0497)	0.2303 (0.0553)
NN						
1D (52)	0.6433 (0.0505)	0.5844 (0.0645)	0.6148 (0.1459)	0.5809 (0.0873)	0.2633 (0.0420)	0.1301 (0.0636)
3D (24)	0.7646 (0.0442)	0.7246 (0.0621)	0.6739 (0.1133)	0.7314 (0.0810)	0.3768 (0.0530)	0.2904 (0.0602)
All (76)	0.7799 (0.0395)	0.7236 (0.0551)	0.6769 (0.1125)	0.7302 (0.0728)	0.3756 (0.0496)	0.2901 (0.0565)
SBF (17)	0.8072 (0.0309)	0.7388 (0.0614)	0.6873 (0.1191)	0.7461 (0.0816)	0.3945 (0.0567)	0.3145 (0.0604)
SBF-Int (16)	0.8168 (0.0291)	0.7423 (0.0585)	0.6883 (0.1108)	0.7501 (0.0767)	0.3980 (0.0561)	0.3183 (0.0605)
FDA						
1D (52)	0.6619 (0.0425)	0.6212 (0.0667)	0.5943 (0.1322)	0.6252 (0.0886)	0.2760 (0.0453)	0.1484 (0.0607)
3D (24)	0.7495 (0.0409)	0.7030 (0.0576)	0.6592 (0.1309)	0.7091 (0.0773)	0.3514 (0.0477)	0.2586 (0.0598)
All (76)	0.7956 (0.0362)	0.7277 (0.0644)	0.6898 (0.1347)	0.7329 (0.0866)	0.3840 (0.0529)	0.3032 (0.0604)
SBF (17)	0.7876 (0.0377)	0.7417 (0.0532)	0.6742 (0.1147)	0.7515 (0.0685)	0.3905 (0.0559)	0.3077 (0.0650)
SBF-Int (16)	0.7969 (0.0353)	0.7505 (0.0566)	0.6799 (0.1012)	0.7604 (0.0734)	0.4030 (0.0534)	0.3224 (0.0579)

The best overall results on the testing sets (high AUC and accuracy rates, with balanced sensitivity and specificity) were obtained with RFs applied to the *SBF-Int* feature set, showing significant differences (with p value $<$ 0.05 in most cases) with respect to the other ML models analysed, and for all the feature sets studied. The most significant differences were found between RF and SVM, whereas in the comparisons between RF and NN/FDA, the former gave best or equal performance rates than the latter ones. As it can be observed in Tables S1 and S2 (this latter shows the p values from the paired Wilcoxon signed-rank test of the paired comparisons between the different feature sets with only RF), although very similar results were obtained with both the *SBF-Int* subset (of only 16 features) and the complete set (of 76 characteristics), significant differences were found in favour of the *SBF-Int* feature set (p value < 0.05 for AUC comparisons), which means that this reduced feature set composed of *mRMR*-selected variables and enriched with a few interaction terms is capable of extracting all the predictive power from the original primary and tertiary characteristics.

In general, as it can be seen in Table 5 and Fig. 4, NNs and FDAs show similar efficacy rates, with accuracy and AUC numbers that are significantly lower than those obtained using RFs. NNs and FDAs show also slightly worse balances between sensitivity and specificity rates than RFs. Overall, the worst results were obtained with SVMs. However, when compared to the results obtained with SVM and NN in our previous work [2, Table 2], a very significant improvement (see Table S3) has been achieved now in the predictive capabilities of both ML models and, in particular, in the balance between sensitivity and specificity rates, even though we use here a much more reduced set of features (16 features against the 54 features used in our previous study).

Based on the AUC rates shown in Table 5, the best results for three of the four ML models were obtained with the *SBF-Int* feature set (i.e. with feature engineering + selection). The only model that does not seem to benefit from these two techniques is SVM, for which using all the features gives the best AUC rates. Even so, those results obtained with SVM are significantly lower than the best performance rates of the other three models when using feature engineering + selection, and even when using feature selection only. That is the reason why we consider that feature engineering and selection (i.e. the *SBF-Int* feature set) provide, in general, a clear advantage over using the *All* feature set.

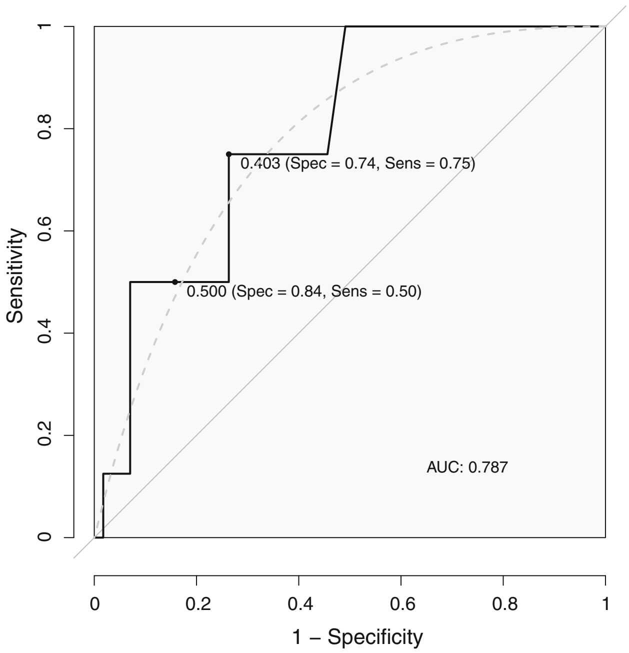


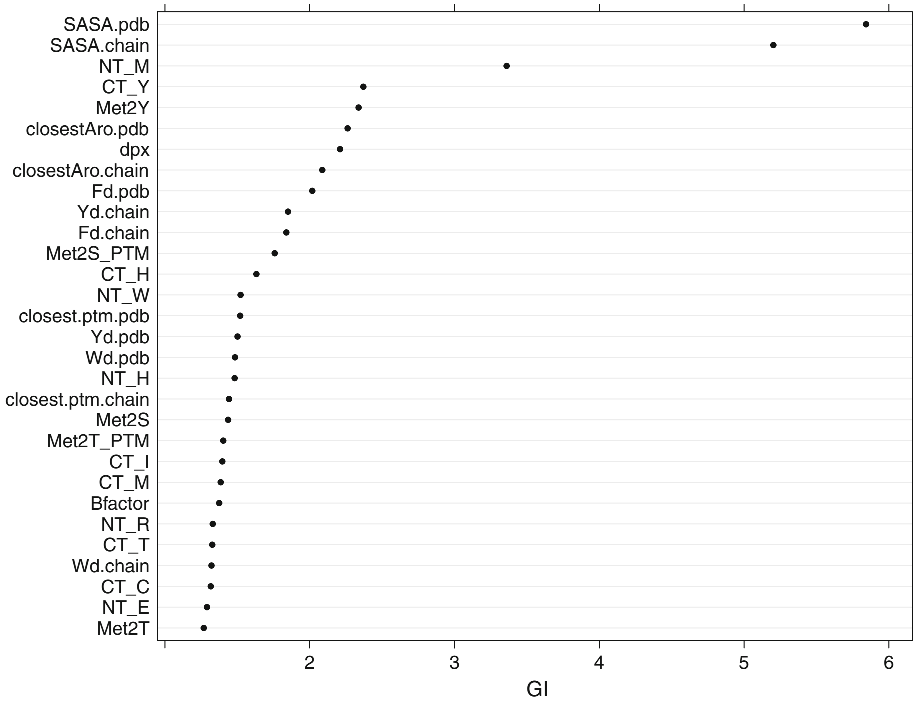
Fig. 2 ROC curve with two different thresholds. ROC curve of the RF classifier on the evaluation dataset (using the *All* feature set). Two different thresholds have been highlighted on the curve, along with their corresponding specificity and sensibility rates: 0.5 (original) and 0.403 (alternative). The theoretical maximal area of reference (i.e. AUC = 1) has been also coloured grey. Dotted grey line represents the smoothed ROC curve. Solid grey line means random guess

In Table S3, the *p* values from paired Wilcoxon signed-rank tests of the comparisons between the results obtained with the current RF, SVM and NN models (applied on the *SBF-Int* testing dataset) and those from our previous study in [2, Table 2—RF– mRMR] are shown. Since FDA models were not used in [2], they are not shown in Table S3. In order to get the *p* values in Table S3, we ran model-to-model hypothesis tests to compare the performance rates (obtained from 100 bootstrapping resamples) of each model when it is applied on the current *SBF-Int* dataset as well as on the *mRMR* dataset of Aledo et al. [2]. As it can be observed in the table, the results obtained with the current models (that take advantage of combined feature engineering and feature selection strategies) improve those of Aledo et al. [2], with differences—most of which are statistically significant—in all the analysed metrics greater than one percentage point in favour of the current approach.

Although we have thoroughly compared four different ML models on five different feature sets, it would be also interesting to compare them to a much simpler model, such as one based entirely on solvent accessibility measures. This idea is motivated by Fig. 3, in which it seams that solvent accessibility variables (i.e. `SASA.pdb` and `SASA.chain` features) could have likely predicted the oxidation of methionine residues just as well as the models based on more complex feature sets.

To evaluate this idea, we have launched 100 bootstrap repetitions of RFs that use only `SASA.pdb` and `SASA.chain` variables to predict methionine oxidation. Although relatively high efficacy rates are obtained with these two variables (see Table S4), they are in general

Fig. 3 Variable importance. The 30 most important variables ordered by GI (averaged decrease in Gini index) as estimated by the final RF model on patterns in the training *All* dataset



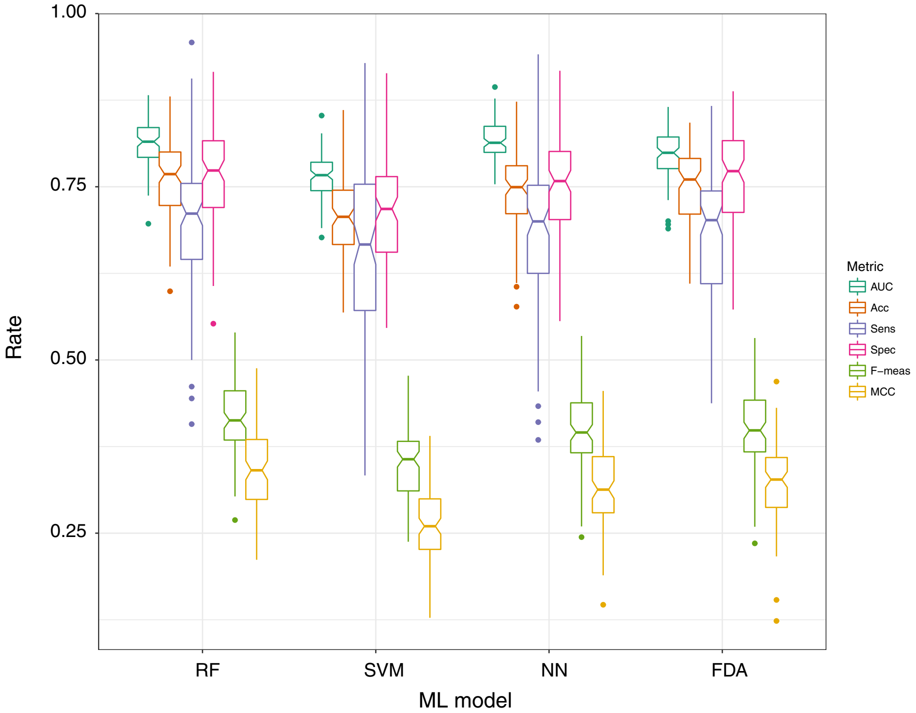


Fig. 4 Performance rate box-plots. Notched box-plots of the performance rates on the testing sets (after adjustment of ROC cut-off probability on the evaluation datasets) from bootstrapping resamples. Those feature sets for which the classifiers gave the best AUC rates have been used for the box-plots: *SBF-Int* dataset for RF, NN and FDA; *All* dataset for SVM. (Number of resamples = 100)

below the performance rates obtained with the datasets used in this study (see Table 5—*RF* row).

4 Application of methionine oxidation prediction: a case study

In order to make our predictive model easily usable, we have included it in an end-user R script that is publicly available at github.com/fveredas/MOPM. This script uses a RF predictive model trained with all the patterns in the *SBF-Int* dataset. It works by taking a PDB ID (the 4-character unique identifier of every protein in the Protein Data Bank) and downloading all the files from public repositories (UniProt) which are required to extract the protein's primary and tertiary features needed as inputs to the RF predictive model. Finally, the script does all the necessary computations to return a list with the prediction of oxidation for all the methionine residues found in the protein being analysed.

To show the utility of this tool, a real case study has been analysed. Figure S1 shows a representation of the tertiary structure of *alpha-1 antitrypsin* (A1AT). This protein is commonly found in the blood plasma and the lung parenchyma. The protein is composed of 418 amino acids, nine out of which are methionine residues. In order to highlight them, these methionyl residues have been represented as spheres in the figure. In [29], Taggart and collaborators used mass spectrometry analysis to identify two out of these nine methionine residues, namely the residues at positions 351 and 358, as oxidation-prone. Moreover, as stated in [29], the oxidation of these two methionine residues was found to be related with the pulmonary emphysema disease. These two methionine residues identified in Taggart's experiments are those represented in red in Figure S1, along with their probability of being oxidised as that given by our RF classifier. The seven remaining methionine residues reported as non-oxidised in [29] are represented in green colour in the figure. Moreover, in Table S5, the output of the R script for methionine oxidation prediction is shown. Given that the protein PDB ID (namely 3CWM for A1T1 protein), the protein chain to be analysed (A) and the probability threshold (e.g. 0.504) have been supplied to the R script, it computes and gives the probability of oxidation of each one of the nine methionine residues, together with a label "Yes" or "No" (as a function of the given probability threshold) for those residues predicted as oxidised or non-oxidised, respectively. As it can be seen, our predictive model could successfully predict the oxidability nature of each methionine residue in the protein, in full agreement with the experimental results reported in [29].

5 Conclusions

In this study, we have followed a machine learning approach to train and fit different classification models that have been validated and comparatively tested in a task of methionine oxidation prediction. Starting with an experimental dataset obtained from mass spectrometry experiments and composed of thousands methionine residues found in a vast set of human proteins, we have assembled a collection of more than one hundred proteins containing almost one thousand methionine sites, a scant ten per cent of which was identified as oxidation-prone in experiments. Using one-dimensional information from the primary linear sequences of the analysed proteins as well as three-dimensional information from the tertiary structure of those proteins with resolved spatial structure, an initial feature set composed of more that seventy characteristics was extracted. We have then applied feature engineering strategies to transform the original dataset as well as to generate and select new input terms that allowed us to significantly increase the efficacy of the analysed classifiers by using a reduced set of characteristics.

For comparison purposes, we have trained and fitted four different machine leaning models, i.e. random forests, support vector machines, neural networks and flexible discriminant analysis models, for methionine oxidation prediction. Using tenfold cross-validation and bootstrap resampling strategies, we have shown that random forests get the best efficacy rates (highest mean AUC and accuracy) among all the models (with high significant differences found in the majority of paired comparisons between the models). Given the unbalanced distribution in favour of negative samples in the dataset, an important result to be remarked here is that random forests get also a balanced compromise between sensitivity and specificity rates. These outcomes were surely contributed by the sampling strategies used during the training phase, along with the adjustment of the probability decision threshold by using the ROC curve of an evaluation dataset that was set aside during training. These results were obtained with a reduced feature set composed of sixteen “engineered” characteristics, which allows the models to give the best accuracy rates in general, when compared to the complete feature set or other subsets of primary and tertiary characteristics.

When compared with our previous work, the feature engineering dataset gave better results for all the analysed machine learning models, with statistically significant differences found in the majority of the model-to-model comparisons analysed. Moreover, better balances between sensitivity and specificity rates were also obtained in general by using the strategies and methods presented in this paper.

Finally, a remarkable fact to be pointed out is that those better efficacy rates were obtained with a reduced feature set that was less than one-third the size of that giving the best results in our previous studies. This sixteen-feature “reduced” set makes the predictive model much more robust, as it reduces dramatically the dependency of the model on some tertiary characteristics, such as phosphorylation or S-aromatic features, which are often hard to extract—or even unavailable—from proteins with unresolved or partially resolved structure.

As a practical application of the predictive models trained and fitted in this study, a R script for methionine oxidation prediction has been implemented. This script makes use of a random forest predictive model and works by simply taking a protein identifier as an input and returning a table with the prediction of oxidation for all the methionine residues found in the protein as output. The applicability of this specialised software has been shown on a practical case study for which oxidised and non-oxidised methionine residues were successfully identified in a given protein. However, the necessity of extracting tertiary-structure features limits the applicability of the predictive model to only those proteins with resolved spatial structure. Thus, if the spatial structure of a certain protein was completely (or partially) unresolved, the script could not count on all the necessary information to extract the 3D features related to all its methionine residues.

Acknowledgements This work was partially supported by the project TIN2017-88728-C2-1-R, MINECO, Plan Nacional de I+D+I.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Aledo JC (2014) Life-history constraints on the mechanisms that control the rate of ROS production. Curr Genomics 15:217–230. https://doi.org/10.2174/13892029156661405152306l5. http://www.eurekaselect.com/122198/article
2. Aledo JC, Cantón FR, Veredas FJ (2017) A machine learning approach for predicting methionine oxidation sites. BMC Bioinform 18(1):430. https://doi.org/10.1186/s12859-017-1848-9
3. Arnér ES, Holmgren A (2000) Physiological functions of thioredoxin and thioredoxin reductase. Eur J Biochem 267(20):6102–6109. https://doi.org/10.1046/j.1432-1327.2000.01701.x
4. Bergmeir C, Benítez JM (2012) Neural networks in R using the stuttgart neural network simulator: RSNNS. J Stat Softw 46(7):1–26. https://doi.org/10.18637/jss.v046.i07. http://www.jstatsoft.org/v46/i07/
5. Breiman L, Friedman J, Stone C, Olshen R (1984) Classification and regression trees. Chapman & Hall, New York. https://www.crcpress.com/Classification-and-Regression-Trees/Breiman-Friedman-Stone-Olshen/p/book/9780412048418
6. Caputo B, Sim K, Furesjo F, Smola A (2002) Appearance-based object recognition using SVMs: which kernel should I use? In: Proc of NIPS workshop on statistical methods for computational experiments in visual processing and computer vision, Whistler, vol 2002
7. Collins Y, Chouchani ET, James AM, Menger KE, Cochemé HM, Murphy MP (2012) Mitochondrial redox signalling at a

glance. J Cell Sci 125(Pt 4):801–806. https://doi.org/10.1242/jcs.098475

8. Datta S, Mukhopadhyay S (2015) A grammar inference approach for predicting kinase specific phosphorylation sites. PLoS One 10(4):e0122,294. https://doi.org/10.1371/journal.pone.0122294
9. Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30, http://dl.acm.org/citation.cfm?id=1248547.1248548
10. Dietterich TG (1998) Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput 10:1895–1923. https://doi.org/10.1162/089976698300017197. https://www.mitpressjournals.org/doi/10.1162/089976698300017197
11. Ding C, Peng H (2005) Minimum redundancy feature selection from microarray gene expression data. In: Computational systems bioinformatics CSB2003. Proceedings of the 2003 IEEE bioinformatics conference CSB2003, vol 3(2), pp 523–528. https://doi.org/10.1109/CSB.2003.1227396
12. Drazic A, Miura H, Peschek J, Le Y, Bach NC, Kriehuber T, Winter J (2013) Methionine oxidation activates a transcription factor in response to oxidative stress. Proc Natl Acad Sci USA 110(23):9493–9498. https://doi.org/10.1073/pnas.1300578110
13. Erickson JR, MlA Joiner, Guan X, Kutschke W, Yang J, Oddis CV, Bartlett RK, Lowe JS, O'Donnell SE, Aykin-Burns N, Zimmerman MC, Zimmerman K, Ham AJL, Weiss RM, Spitz DR, Shea MA, Colbran RJ, Mohler PJ, Anderson ME (2008) A dynamic pathway for calcium-independent activation of CaMKII by methionine oxidation. Cell 133(3):462–474. https://doi.org/10.1016/j.cell.2008.02.048
14. Friedman J, Hastie T, Tibshirani R (2010) Regularization paths for generalized linear models via coordinate descent. J Stat Softw 33(1):1–22. http://www.jstatsoft.org/v33/i01/
15. Friedman JH (1991) Multivariate adaptive regression splines. Ann Stat 1–67. https://projecteuclid.org/euclid.aos/1176347963
16. Ghesquière B, Jonckheere V, Colaert N, Van Durme J, Timmerman E, Goethals M, Schymkowitz J, Rousseau F, Vandekerckhove J, Gevaert K (2011) Redox proteomics of protein-bound methionine oxidation. Mol Cell Proteomics 10(5):M110.006,866. https://doi.org/10.1074/mcp.M110.006866
17. Härndahl U, Kokke BP, Gustavsson N, Linse S, Berggren K, Tjerneld F, Boelens WC, Sundby C (2001) The chaperone-like activity of a small heat shock protein is lost after sulfoxidation of conserved methionines in a surface-exposed amphipathic alpha-helix. Biochim Biophys Acta 1545(1–2):227–237. https://doi.org/10.1016/S0167-4838(00)00280-6. https://www.sciencedirect.com/science/article/pii/S0167483800002806?via%3Dihub
18. Jacques S, Ghesquière B, Van Breusegem F, Gevaert K (2013) Plant proteins under oxidative attack. Proteomics 13(6):932–940. https://doi.org/10.1002/pmic.201200237
19. Jacques S, Ghesquière B, De Bock PJ, Demol H, Wahni K, Willemns P, Messens J, Van Breusegem F, Gevaert K (2015) Protein methionine sulfoxide dynamics in arabidopsis thaliana under oxidative stress. Mol Cell Proteomics 14:1217–1229. https://doi.org/10.1074/mcp.M114.043729. http://www.mcponline.org/content/14/5/1217.long
20. Karatzoglou A, Smola A, Hornik K, Zeileis A (2004) kernlab—an S4 package for Kernel methods in R. J Stat Softw 11(9):1–20. https://doi.org/10.18637/jss.v011.i09. http://www.jstatsoft.org/v11/i09/
21. Kim G, Weiss SJ, Levine RL (2014) Methionine oxidation and reduction in proteins. BBA-Gen Subjects 1840(2):901–905. https://doi.org/10.1016/j.bbagen.2013.04.038. https://www.sciencedirect.com/science/article/pii/S0304416513001931?via%3Dihub
22. Kim HY (2013) The methionine sulfoxide reduction system: selenium utilization and methionine sulfoxide reductase enzymes and their functions. Antioxid Redox Signal 19(9):958–969. https://doi.org/10.1089/ars.2012.5081
23. Kuhn M (2008) Building predictive models in R using the caret package. J Stat Softw 28(5):1–26. https://doi.org/10.18637/jss.v028.i05. https://www.jstatsoft.org/v028/i05
24. Kuhn M, Johnson K (2013) Applied predictive modeling. Springer, New York. https://doi.org/10.1007/978-1-4614-6849-3. https://www.springer.com/fr/book/9781461468486
25. Lacoste A, Laviolette F, Marchand M (2012) Bayesian comparison of machine learning algorithms on single and multiple datasets. In: Proceedings of the fifteenth international conference on artificial intelligence and statistics, vol 22, pp 665–675. http://proceedings.mlr.press/v22/lacoste12/lacoste12.pdf
26. Liaw A, Wiener M (2002) Classification and regression by randomForest. R News 2(3):18–22. http://cran.r-project.org/doc/Rnews/
27. R Core Team (2017) R: a language and environment for statistical computing. R Foundation for Statistical Computing, Vienna. https://www.R-project.org/
28. Rao RSP, Møller IM, Thelen JJ, Miernyk JA (2014) Convergent signaling pathways–interaction between methionine oxidation and serine/threonine/tyrosine O-phosphorylation. Cell Stress Chaperon 20(1):15–21. https://doi.org/10.1007/s12192-014-0544-1. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4255251/
29. Taggart C, Cervantes-Laurean D, Kim G, McElvaney NG, Wehr N, Moss J, Levine RL (2000) Oxidation of either methionine 351 or methionine 358 in alpha 1-antitrypsin causes loss of anti-neutrophil elastase activity. J Biol Chem 275:27,258–27,265. https://doi.org/10.1074/jbc.M004850200. http://www.jbc.org/content/early/2000/06/23/jbc.M004850200.long
30. Tang XD, Daggett H, Hanner M, Garcia ML, McManus OB, Brot N, Weissbach H, Heinemann SH, Hoshi T (2001) Oxidative regulation of large conductance calcium-activated potassium channels. J Gen Physiol 117(3):253–274. https://doi.org/10.1085/jgp.117.3.253. http://jgp.rupress.org/content/117/3/253.long
31. Trost B, Kusalik A (2011) Computational prediction of eukaryotic phosphorylation sites. Bioinformatics 27(21):2927–2935. https://doi.org/10.1093/bioinformatics/btr525. https://academic.oup.com/bioinformatics/article/27/21/2927/219032
32. Veredas FJ, Aledo JC, Cantón FR (2017a) Methionine residues around phosphorylation sites are preferentially oxidized in vivo under stress conditions. Sci Rep 7(40403):1–14. https://doi.org/10.1038/srep40403. https://dx.doi.org/10.1038%2Fsrep40403
33. Veredas FJ, Cantón FR, Aledo JC (2017b) Prediction of protein oxidation sites. In: Rojas I, Joya G, Catala A (eds) Advances in computational intelligence: 14th international work-conference on artificial neural networks, IWANN 2017, June 14–16, Proceedings, Part II. Springer, Cham, Cadiz, Spain, pp 3–14. https://doi.org/10.1007/978-3-319-59147-6_1. https://www.springer.com/in/book/9783319591469
34. Xue Y, Ren J, Gao X, Jin C, Wen L, Yao X (2008) GPS 2.0, a tool to predict kinase-specific phosphorylation sites in hierarchy. Mol Cell Proteomics 7(9):1598–1608. https://doi.org/10.1074/mcp.M700574-MCP200
35. Zumel N, Mount J (2014) Practical data science with R, 1st edn. Manning Publications Co., Greenwich. https://www.manning.com/books/practical-data-science-with-r