# Specificity Prediction of Plant Acyl-ACP Thioesterases using Machine Learning

Deepro Banerjee\*, Michael Jindra†, Brian F. Pfleger†, Costas D. Maranas\*

\* Department of Chemical Engineering, Pennsylvania State University, University Park, Pennsylvania, USA

† Department of Chemical and Biological Engineering, University of Wisconsin-Madison, Madison, Wisconsin, USA

## Introduction

A key value proposition of synthetic biology is providing access to chemicals which are not sustainably produced at commercial scales. Medium-chain oleochemicals, with eight to twelve-carbon free fatty acids and derivatives, are one such class of products. While these chain lengths have traditionally been sourced from the tropical crops, such as palm, palm kernel, and coconut, the 8, 10, and 12-carbon products are not major constituents of the produced oil [1]. Furthermore, the displacement of rainforest habitat due to the cultivation of the oil palm has been identified as having the single largest impact on decreasing biodiversity observed in the Southeast Asian jungle ecosystem [2]. Processes have been established to create the higher value oleochemical derivatives, such as fatty alcohols, directly from petrochemical building blocks. However, these processes yield a distribution of alcohols\*\*fatty alcohols or acids?\*\*, and thus do not provide a highly selective route to the medium-chain products [3].

As an alternative, the field of synthetic biology has achieved fatty acid and fatty alcohol distributions with over 90% of the product belonging to the C8 species [4], [5]. This has been achieved via rewiring of the fatty acid biosynthesis pathway in *E. coli*, namely by the incorporation of an engineered 8-carbon specific acyl-ACP Thioesterase (TE) from *Cuphea palustris*. Indeed, the expression of various acyl-ACP TEs, either homologs from nature or variants thereof, has enabled control over the chain-length distribution in *E. coli* production systems [4], [5], [6], [7], [8], [9] **(Figure 1)**. Of these studies, acyl-ACP TEs from select plant species have been shown to have greater native specificity toward the medium-chain substrates when compared to bacterial homologs [7], [10], [11], [12]. Thus, several efforts have been made to bioprospect genomes of plants with high fractions of the medium-chain oils to identify and implement the TE gene responsible for the narrow substrate specificity [13], [4]. While progress has been demonstrated in identifying the features which dictate specificity in acyl-ACP TEs among plants [14], the throughput for bioprospecting, characterizing, and in some cases, engineering the acyl-ACP TE is largely inhibited by the testing pipeline, which requires derivatization of the free fatty acids into fatty acid methyl esters prior to analysis with gas chromatography [15]. An *in-silico* method for inferring substrate specificity from TE primary sequence would therefore expedite this process, by homing in and expressing only the most promising homologs in a host to gain insight to its selectivity profile.

A variety of computational approaches to infer protein functional groups based on their primary sequences have been proposed, which broadly fall under two categories, generative and discriminative. The generative approach builds a model of the feature distribution for each protein category and assigns a particular class or functional group to a candidate protein sequence by evaluating how well the sequence fits the model. Methods that fall under the generative category are based on sequence similarity comparison using local alignment similarity scores [16], [17], position specific scoring tables or profiles created from a group of previously aligned sequences [18], [19], consensus sequence patterns or motifs [20] and Hidden Markov Models (HMMs) [21]–[23]. However, generative approaches are either highly dependent on the database used to search for sequence similarity (local alignment and profile based similarity search) [24] or are computationally expensive (HMMs) [25]. Moreover, Jing *et al.* showed that phylogenetic and sequence identity analysis alone were not sufficient to distinguish plant TE substrate specificity [8]. The discriminative approach on the other hand focuses on accurately learning the decision boundary between classes. Commonly used discriminative approaches rely on training Machine Learning (ML) classifiers like Support Vector Machine (SVMs) or Neural Network (NNs) to learn discriminative rules from both positive (belonging to a particular protein class) and negative (not belonging to that protein class) set of protein sequences and using the learnt rules to predict the class of any new protein sequence [26]–[28]. ML algorithms like SVMs and NNs are relatively computationally inexpensive methods that have been used before to classify proteins into different functional classes with high accuracy [25], [27]–[35]. While some of these approaches have also incorporated structural information [36], [37], most ML algorithms have utilized only information from primary sequence to distinguish between highly similar proteins (e.g., guanylyl vs. adenylyl cylases, lactate vs. malate dehydrogenases, or trypsins vs. chymotrypsins [38]).

Diagram

Description automatically generated

**Figure 1:** The acyl-ACP Thioesterase plays a key role in fatty acid biosynthesis in E. coli. By intercepting the growing acyl-ACP chains, the Thioesterase hydrolyzes the acyl chain from the ACP and redirects flux to the free fatty acid pool. These free fatty acids can be further derivatized in vivo or ex vivo.

Recent results suggest that discriminative approaches relying on ML have outperformed generative approaches both in terms of accuracy and computational efficiency of solving the protein classification problem [30]. SVM is among the most widely used discriminative learning algorithm which has been proven to be extremely effective\*\*vague, quantify!\*\* on sequence based classification [26]–[28], [30], [33], [39]–[41]. SVM has been also used to detect remote protein homologies and classify proteins in the SCOP database into major structural classes [25], [26]. Ultimately, the performance of an SVM classifier is highly influenced by the feature extraction technique employed to encode the protein sequences [42].

Feature extraction of protein sequences generates a discrete numerical representation of a protein to create feature vectors that are correlated with the desired attribute of the protein one would like to predict. In order to train an SVM, a number of feature extraction techniques for protein sequences have been suggested in literature which can be divided into two categories, kernel based methods and vector representation of protein sequences [34]. The Fisher kernel introduced by Jakkola et. al. is one of the first kernel based feature extraction technique used to classify proteins based on their sequence information [43]. It was followed by the spectrum kernel [27] and its more generalized form, the mismatch kernel [30], both introduced by Leslie et. al. which achieved similar performances in terms of accuracy when compared to the Fisher kernel but with less computational expense. The weighted degree kernel introduced by Ratsch et. al. encoded the position of the substrings within the protein sequence as opposed to the spectrum and mismatch kernel introduced by Leslie et. al. [44]. It was used to identify alternatively spliced exons in C. elegans. Apart from kernel based methods, the second class of feature representation technique extracts structural and physicochemical properties embedded in the protein sequence and converts it into a numerical vector. One of the first and simplest discrete model to represent protein sequences that falls under the second class is Amino Acid Composition (AAC) developed by Nakashima et. al. which was used to classify proteins into different folding types with high accuracy [45]. Dubchak et. al. developed the more complicated Composition-Transition-Distribution (CTD) descriptor that takes into account different physical and stereochemical properties of the amino acids in the protein sequences such as amino acid composition, predicted secondary structure, hydrophobicity, normalized van der waals volume, polarity and polarizability to construct the feature vector [26]. Chou et. al. upgraded the simple AAC encoder developed by Nakashima et. al. to a pseudo Amino Acid (pseAA) composition encoder that is able to retain some pattern specific information embedded in the protein sequence [46]. Features have also been derived from Position Specific Scoring Matrices (PSSMs) profiles, which contains evolutionary information\*\*how?\*\* [47]. While selection of the most informative feature extraction technique has resulted in an improved performance of a classifier, the use of ensemble methods which combines the output of multiple classifiers has also helped to attain greater accuracy while solving the classification problem [48].

Several studies have shown that ensemble methods performed better than any individual classification method especially in problems relevant to the protein classification domain [29], [31], [32]. Camoglu et. al used a decision tree based ensemble classifier to classify protein in the SCOP database and showed how it is possible to attain much lower error rates using the ensemble classifier than any individual method [32]. Diplaris et. al. performed an empirical study where they compared the performance of several individual algorithms to solve the motif based classification problem and demonstrated the positive effect of combining different classification algorithms on prediction accuracy [31]. Tan et. al. illustrated the advantage of using ensemble classifiers on imbalanced datasets while solving the protein fold classification problem [29]. Similarly, Caragea et. al. trained an ensemble of SVM classifiers to predict glycosylation sites in amino acid residues and found that an ensemble of SVMs outperformed an individual SVM trained on imbalanced data [49].

Herein we put forth a machine learning based discriminatory approach to predict substrate specificity from their primary sequence for novel, uncharacterized, medium-chain TEs. To test the developed computational base, we trained an SVM-based ensemble classifier with TE sequences previously characterized in *E. coli*. Information about characterized TEs were collected as a part of this study from multiple literature sources [7], [50]–[52] and they were categorized into three different classes based on their substrate specificity as (i) medium chained, (ii) long chained and (iii) mixed, yielding a multi-class classification problem. To solve the classification problem, a stacked ensemble framework was developed comprised of three base learners trained using three different feature extraction techniques \*(i.e., x, y and z)\* from protein primary sequences. A meta learner was subsequently applied which combines the output of the three base learners by applying a majority voting criterion to predict the substrate specificity class of TEs. An ensemble method was used to circumvent the imbalanced dataset and retain high prediction accuracy. The achieved classification accuracy (i.e., XX%) validates our decision of using an ensemble method with imbalanced dataset improving over previous studies [29], [49]. The base learners in the ensemble are all SVM classifiers similar to the works of Caragea et. al. [49] and Nanni et. al. [34] where they have used an ensemble of SVMs to boost prediction accuracy. SVM was selected as the primary learning algorithm because they are generalizable and can handle high dimensional datasets [53]. We have also experimented with more complex/nonlinear models such as Neural Network and Gradient Boosted Trees as the base learner, but they were both outperformed by SVM in terms of accuracy on separate held-out validation sets. Proteins were represented by their amino acid sequences. The three feature extraction techniques used were (i) the spectrum kernel introduced by Leslie et. al. [27], (ii) a variant of the spectrum kernel where amino acids were first grouped into 6 categories to create a compressed representation based on their physicochemical attributes and the spectrum kernel was used to extract features from this compressed representation, and (iii) a newly introduced feature representation technique termed autopos detection which automatically detects the positions within the protein sequence which are maximally correlated with the functional attribute of the proteins. These three different feature representation techniques were used to train the three base models in the ensemble. We have also analyzed the efficacy of using 21 other representation techniques including other kernel based methods such as mismatch kernel and methods that extract physicochemical attributes from protein sequences such as CTD [26]. They all exhibited worse performance (detailed study given in the Results section). The feature vectors generated through the extraction process were decomposed into lower dimensional and linearly uncorrelated features using Principal Component Analysis. The reduction in dimensionality of the feature vectors was performed to prevent overfitting. The lower dimensional and decomposed set of feature vectors were used to train the classifiers and predict substrate specificity of TEs. The ensemble method achieved a mean validation accuracy of 0.76 across 10,000 simulations of this study using different training and validation sets. However, the worst case accuracy across simulations was 0.45 which indicates that the method is not extremely robust to the training set. One possible reason behind the lack of robustness could be that the decision boundaries between the protein classes is not well defined. SVMs are maximal margin classifiers and the thickness of the decision boundary plays a very important role governing the robustness of the model [54]. The mean precision score of the model across the simulations for the medium chained TEs, the product of interest, was 0.84. We used this model to identify 10-carbon specific acyl-ACP TE among a set of uncharacterized TE enzymes from select plants known to have predominantly decanoyl chains in their seed oils. This study provides a exemplar of how small to medium-sized datasets can be leveraged to guide bioprospecting efforts while simultaneously supporting the endeavor of the synthetic biology community to provide access to chemicals not easily obtained through conventional methods.

## Methods

### Dataset Compilation

The dataset was compiled by integrating primary sequence and substrate specificity of TEs from multiple literature sources \cite(\*\*\*). The dataset includes primary sequence and accompanying in vivo *E. coli* product distributions for 116 acyl-ACP plant TEs previously reported in scientific and patent literature [7], [50]–[52]. *E. coli* was chosen because it remains the most common and facile method for characterization of heterologous TEs. The product distribution data was subsequently used to classify each TE into discrete categories to be used in a classification framework. Based on their product distributions, the TEs were divided into three categories, 1) the “medium-chain” category contained TE which resulted in distributions of at least 50% C8 to C12 free fatty acids, 2) the “long-chain” category contained TE which produced 50% C14 to C18 free fatty acids and less than 10% C8 to C12 free fatty acids and 3) the “mixed distribution” category contained TE which yielded distributions between 10% and 50% C8 to C12 free fatty acids. This discrete classification into three bins is a useful abstraction but it does not account for the fact that there exists TEs with a continuum of specificities across all chain lengths.

### Feature Extraction

In this work, the following three feature representation methods were employed to encode primary sequence information of the enzymes into fixed length vectors.

#### k-spectrum kernel

The k-spectrum kernel proposed by Leslie et. al. [27] is the set of all k-length contiguous subsequences present in a given input sequence. If we assume to be the set of all characters of a sequence, to be a k-mer and to be an indicator function that is 1 if occurs in a protein sequence and 0 otherwise, then the feature map is as follows:

The k-spectrum kernel, is:

#### Grouped amino acid encoded k-spectrum kernel

The grouped amino acid encoded k-spectrum kernel performs an additional pre-processing step on the primary sequences of the enzymes before encoding them into a feature vector representation similar to the k-spectrum kernel. In the preprocessing step, 20 amino acid types which can occur at a particular position in the sequence of an enzyme were categorized into five classes according to their physicochemical properties, hydrophobicity, charge and molecular size. The five classes were the aliphatic group (: GAVLMI), aromatic group (: FYW), positive charge group (: KRH), negative charged group (: DE) and uncharged group (: STCPNQ). Finally, each amino acid in the sequence was replaced by its respective class to create an encoded sequence with only five different types of values per position. For example, a primary sequence ‘MLTPWS’ is represented as . Using the encoded representation, k-spectrum kernel was used to represent the protein sequence. It should be noted that the preprocessing step significantly reduces the set of characters of the sequence and can help prevent overfitting since the kernel feature map size will be much lower than the original feature map size obtained by the k-spectrum kernel. The reduced feature map size will require much lower number of parameters to train a classifier and thus may prevent overfitting.

#### Automatic position detection method for feature extraction

The automated position selection method selects the most important amino acid residue positions based on their statistical correlation with the labels (substrate specificity category of the enzymes). In this method, multiple sequence alignment of the enzymes was performed using the mafft tool [55]. The aligned sequences were transformed into a one-hot encoded representation of each amino acid position. Theoretically, each position could be encoded by a 21 dimensional feature vector (20 types of amino acids and a gap) but in most cases a position was encoded by a vector of length 5 or even less since the types of amino acids in a specific position were usually conserved. The encoded feature space was used to select the best features which contributed the most towards the target variable’s or label’s variability. The feature selection technique was conducted using the SelectKBest module in scikit-learn [56]. Anova F-test, Chi-2 test or Mutual information can be used as the correlation metric between the features and labels based on which the positions were selected. The best ranked features were mapped back from the expanded one hot encoded feature space to the original amino acid positions. These positions were recorded as the most important determinant of enzyme substrate specificity. The method had a parameter, n, that denotes the number of positions among all the aligned amino acid sequence positions that it must extract using the feature selection algorithm before stopping. After selecting the n most important positions, amino acids in those positions were represented as a one hot encoded feature vector. Each position may also contain a gap apart from the 20 types of amino acids (result of multiple sequence alignment). Thus a 21-dimensional one hot encoded feature vector was used to represent each position. The length of the feature space obtained as a result of this encoding was 21 \* n. The correlation metric and the number of positions to be selected can be determined through hyperparameter optimization \*\*you mean linear regression??\*\*.

### Ensemble Method for Classification

Our ensemble framework involved three base learners which provided an output to a meta learner that predicted the enzyme specificity class. Although all of our base learners were trained using the same principle (either SVM or NN), the heterogeneity among them was governed by the three different feature representation techniques described in the Feature Extraction section which were used to encode the set of enzyme sequences. The outputs of the base learners were passed on to the meta learner that used a majority voting scheme to predict the enzyme specificity category; if the predictions of the three base learners were respectively, then the prediction of the meta learner was . \*\*\*Define mode \*\*\*The workflow of our ensemble model is presented in **Figure 2**.

#### The Base Learner

##### Support Vector Machine

The Support Vector Machine based learner of enzyme specificity prediction included Principal Component Analysis (PCA) for dimensionality reduction of the feature space followed by a Support Vector Classifier [57] to predict enzyme specificity class. The PCA based dimensionality reduction step was carried out to decrease the number of parameters required to train an SVM model and make the model more generalizable. The one versus one strategy was used to adapt the SVM for multi-class classification [58]. The number of PCA components, SVM model kernel, regularization parameter C and kernel coefficient gamma were selected by optimizing these hyperparameters using a 3-fold cross validation scheme described in the [Model Training](#_Model_Training) subsection.

##### Neural Network

The NN based learner of enzyme specificity prediction included Principal Component Analysis for dimensionality reduction of the feature space followed by an Artificial Neural Network based classifier to predict enzyme specificity class. The PCA based dimensionality reduction step was carried out to decrease the number of parameters required to train an NN model and make the model more generalizable. The number of PCA components, hidden layer size of NN, initial learning rate and L2 regularization parameter alpha were selected by optimizing these hyperparameters using a 3-fold cross validation scheme described in the [Model Training](#_Model_Training) subsection.

Diagram

Description automatically generated

**Figure 2:** Workflow of the ensemble model. Different feature representations create separate models, and the final model output is the majority vote of the predictions made by each individual model.

#### Model Training

The model was trained using python’s numpy and scikit-learn modules [56], [59]. At first, a random seed was specified using numpy to reproduce results. The dataset of 116 TE enzyme sequences labeled according to their corresponding substrate specificity category was divided into a training set and a validation set by a 75-25 percentage split. The training set of sequences was encoded by the three different feature representation techniques, described in the [Feature Extraction](#_Feature_Extraction) section, into three distinct feature vector representation of the sequences. The distinct feature vectors of the training set of sequences were used to train three separate base learners operating on the same principle (either PCA+SVM or PCA+NN). The hyperparameters of the base learners (number of components of PCA, kernel type, regularization parameter C, kernel coefficient in case of the SVM based learner or hidden layer size initial learning rate and regularization parameter in case of the NN based learner) were optimized using the GridSearchCV module of scikit-learn and setting the cross validation split to 3. The three base models with an optimized set of hyperparameters were used to independently predict the substrate specificity category of enzymes in both training and validation sets. The output predictions of these base learners were passed on to a meta learner that used a hard-voting based majority vote classifier to output the final prediction of the enzyme substrate specificity class. When there is a three-way tie between outputs of the base learners, the prediction of the positional feature builder based learners (which performed the best among all the base learners) was selected as the ensemble output. The training and validation accuracies of the three base learners and the ensemble model were recorded. This whole procedure was repeated 10,000 times by varying the random seed specified initially, which resulted in different training and validation set, thus affecting the model performance and yielding a distribution of training and validation set accuracies for the three base learners and the ensemble model. The objective of training our model multiple times was to check its robustness to the training set. In addition to the above mentioned hyperparameters, the k-spectrum kernel and GAA encoded k-spectrum kernel had an additional hyperparameter k that denotes the length of the motif to be considered. It was set to be 7 for both the models based on a separate validation study mentioned in detail in Supplementary Materials.

## References

[1] W. Rupilius and S. Ahmad, “Palm oil and palm kernel oil as raw materials for basic oleochemicals and biodiesel,” *Eur. J. Lipid Sci. Technol.*, vol. 109, no. 4, pp. 433–439, 2007, doi: 10.1002/ejlt.200600291.

[2] D. S. Wilcove and L. Pin, “Addressing the threats to biodiversity from oil-palm agriculture,” *Biodivers. Conserv.*, vol. 19, no. 4, pp. 999–1007, 2010, doi: 10.1007/s10531-009-9760-x.

[3] K. Noweck and H. Ridder, “Fatty Alcohols - Industrial Production,” in *Ullmann’s encyclopedia of industrial chemistry*, 5th ed., Wiley-VCH, 1988, pp. 277–295.

[4] N. J. Hernández Lozada *et al.*, “Highly Active C 8 -Acyl-ACP Thioesterase Variant Isolated by a Synthetic Selection Strategy,” *ACS Synth. Biol.*, vol. 7, no. 9, pp. 2205–2215, 2018, doi: 10.1021/acssynbio.8b00215.

[5] N. J. Hernández Lozada, T. R. Simmons, K. Xu, M. A. Jindra, and B. F. Pfleger, “Production of 1-octanol in Escherichia coli by a high flux thioesterase route,” *Metab. Eng.*, vol. 61, no. April, pp. 352–359, 2020, doi: 10.1016/j.ymben.2020.07.004.

[6] M. J. Grisewood *et al.*, “Computational Redesign of Acyl-ACP Thioesterase with Improved Selectivity toward Medium-Chain-Length Fatty Acids,” doi: 10.1021/acscatal.7b00408.

[7] L. Yuan, T. A. Voelker, and D. J. Hawkins, “Modification of the substrate specificity of an acyl-acyl carrier protein thioesterase by protein engineering,” *Proc. Natl. Acad. Sci. U. S. A.*, 1995, doi: 10.1073/pnas.92.23.10639.

[8] F. Jing *et al.*, “Phylogenetic and experimental characterization of an acyl-ACP thioesterase family reveals significant diversity in enzymatic specificity and activity,” pp. 1–16, 2011.

[9] X. Deng, L. Chen, M. Hei, T. Liu, Y. Feng, and G. Y. Yang, “Structure-guided reshaping of the acyl binding pocket of ‘TesA thioesterase enhances octanoic acid production in E. coli,” *Metab. Eng.*, vol. 61, no. January, pp. 24–32, 2020, doi: 10.1016/j.ymben.2020.04.010.

[10] T. A. Voelker and H. M. Davies, “Alteration of the Specificity and Regulation of Fatty Acid Synthesis of Escherichia coli by Expression of a Plant Medium- Chain Acyl-Acyl Carrier Protein Thioesterase,” *J. Bacteriol.*, vol. 176, no. 23, pp. 7320–7327, 1994.

[11] Y. J. Choi and S. Y. Lee, “Microbial production of short-chain alkanes,” *Nature*, vol. 502, no. 7472, pp. 571–574, 2013, doi: 10.1038/nature12536.

[12] S. Sarria, T. G. Bartholow, A. Verga, M. D. Burkart, and P. Peralta-Yahya, “Matching Protein Interfaces for Improved Medium-Chain Fatty Acid Production,” *ACS Synth. Biol.*, vol. 7, no. 5, pp. 1179–1187, 2018, doi: 10.1021/acssynbio.7b00334.

[13] P. Gordon Roessler and G. Roy, “ACYL-ACP THOESTERASE GENES AND USES THEREFOR,” 8956834 B2, 2015.

[14] F. Jing, L. Zhao, M. D. Yandeau-Nelson, and B. J. Nikolau, “Two distinct domains contribute to the substrate acyl chain length selectivity of plant acyl-ACP thioesterase,” *Nat. Commun.*, vol. 9, no. 1, p. 860, 2018, doi: 10.1038/s41467-018-03310-z.

[15] M. Politz, R. Lennen, B. Pfleger, and B. Engineering, “Quantification of Bacterial Fatty Acids by Extraction and Methylation,” *Bio Protoc.*, vol. 3, no. 21, 2016.

[16] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, “Basic local alignment search tool,” *J. Mol. Biol.*, 1990, doi: 10.1016/S0022-2836(05)80360-2.

[17] S. F. Altschul *et al.*, “Gapped BLAST and PSI-BLAST: A new generation of protein database search programs,” *Nucleic Acids Research*. 1997, doi: 10.1093/nar/25.17.3389.

[18] M. Gribskov, A. D. McLachlan, and D. Eisenberg, “Profile analysis: detection of distantly related proteins.,” *Proc. Natl. Acad. Sci. U. S. A.*, 1987, doi: 10.1073/pnas.84.13.4355.

[19] A. Bairoch, “PROSITE: A dictionary of sites and patterns in proteins,” *Nucleic Acids Research*. 1992, doi: 10.1093/nar/20.suppl.2013.

[20] T. K. Attwood, M. E. Beck, D. R. Flower, P. Scordis, and J. N. Selley, “The PRINTS protein fingerprint database in its fifth year,” *Nucleic Acids Res.*, 1998, doi: 10.1093/nar/26.1.304.

[21] P. Baldi, Y. Chauvin, T. Hunkapiller, and M. A. Mcclure, “Hidden Markov models of biological primary sequence information,” *Proc. Natl. Acad. Sci. U. S. A.*, 1994, doi: 10.1073/pnas.91.3.1059.

[22] A. Krogh, M. Brown, I. S. Mian, K. Sjölander, and D. Haussler, “Hidden Markov Models in computational biology applications to protein modeling,” *J. Mol. Biol.*, 1994, doi: 10.1006/jmbi.1994.1104.

[23] S. R. Eddy, “Multiple alignment using hidden Markov models.,” *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 1995.

[24] J. McDowall and S. Hunter, “InterPro protein classification.,” *Methods Mol. Biol.*, 2011, doi: 10.1007/978-1-60761-977-2\_3.

[25] W. Chmielnicki and K. Staçpor, “A hybrid discriminative/generative approach to protein fold recognition,” *Neurocomputing*, 2012, doi: 10.1016/j.neucom.2011.04.033.

[26] C. H. Q. Ding and I. Dubchak, “Multi-class protein fold recognition using support vector machines and neural networks,” *Bioinformatics*, 2001, doi: 10.1093/bioinformatics/17.4.349.

[27] C. Leslie, E. Eskin, and W. S. Noble, “The spectrum kernel: a string kernel for SVM protein classification.,” *Pac. Symp. Biocomput.*, 2002, doi: 10.1142/9789812799623\_0053.

[28] T. Jaakkola, M. Diekhans, and D. Haussler, “Using the Fisher kernel method to detect remote protein homologies.,” *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 1999.

[29] A. C. Tan, D. Gilbert, and Y. Deville, “Multi-class protein fold classification using a new ensemble machine learning approach.,” *Genome Inform.*, 2003, doi: 10.11234/gi1990.14.206.

[30] C. S. Leslie, E. Eskin, A. Cohen, J. Weston, and W. S. Noble, “Mismatch string kernels for discriminative protein classification,” *Bioinformatics*, 2004, doi: 10.1093/bioinformatics/btg431.

[31] S. Diplaris, G. Tsoumakas, P. A. Mitkas, and I. Vlahavas, “Protein classification with multiple algorithms,” 2005, doi: 10.1007/11573036\_42.

[32] O. Çamoǧlu, T. Can, A. K. Singh, and Y. F. Wang, “Decision tree based information integration for automated protein classification,” *J. Bioinform. Comput. Biol.*, 2005, doi: 10.1142/S0219720005001259.

[33] Z. Xing, J. Pei, and E. Keogh, “A brief survey on sequence classification,” *ACM SIGKDD Explor. Newsl.*, 2010, doi: 10.1145/1882471.1882478.

[34] L. Nanni, A. Lumini, and S. Brahnam, “An empirical study of different approaches for protein classification,” *Sci. World J.*, 2014, doi: 10.1155/2014/236717.

[35] S. Nojoomi and P. Koehl, “String kernels for protein sequence comparisons: Improved fold recognition,” *BMC Bioinformatics*, 2017, doi: 10.1186/s12859-017-1560-9.

[36] P. Khurana, R. S. Gokhale, and D. Mohanty, “Genome scale prediction of substrate specificity for acyl adenylate superfamily of enzymes based on active site residue profiles,” *BMC Bioinformatics*, 2010, doi: 10.1186/1471-2105-11-57.

[37] S. R. Amin, S. Erdin, R. M. Ward, R. C. Lua, and O. Lichtarge, “Prediction and experimental validation of enzyme substrate specificity in protein structures,” *Proc. Natl. Acad. Sci. U. S. A.*, 2013, doi: 10.1073/pnas.1305162110.

[38] G. X. Yu, B. H. Park, P. Chandramohan, R. Munavalli, A. Geist, and N. F. Samatova, “In silico discovery of enzyme-substrate specificity-determining residue clusters,” *J. Mol. Biol.*, vol. 352, no. 5, pp. 1105–1117, 2005, doi: 10.1016/j.jmb.2005.08.008.

[39] M. Deshpande and G. Karypis, “Evaluation of techniques for classifying biological sequences,” 2002, doi: 10.1007/3-540-47887-6\_41.

[40] G. Rätsch, S. Sonnenburg, and C. Schäfer, “Learning interpretable SVMs for biological sequence classification,” *BMC Bioinformatics*, 2006, doi: 10.1186/1471-2105-7-S1-S9.

[41] S. Sonnenburg, G. Rätsch, and B. Schölkopf, “Large scale genomic sequence SVM classifiers,” 2005, doi: 10.1145/1102351.1102458.

[42] H. Saigo, J. P. Vert, N. Ueda, and T. Akutsu, “Protein homology detection using string alignment kernels,” *Bioinformatics*, 2004, doi: 10.1093/bioinformatics/bth141.

[43] T. Jaakkola, M. Diekhans, and D. Haussler, “A discriminative framework for detecting remote protein homologies,” *Journal of Computational Biology*. 2000, doi: 10.1089/10665270050081405.

[44] G. Rätsch, S. Sonnenburg, and B. Schölkopf, “RASE: Recognition of alternatively spliced exons in C.elegans,” *Bioinformatics*, 2005, doi: 10.1093/bioinformatics/bti1053.

[45] H. Nakashima, K. Nishikawa, and T. Ooi, “The folding type of a protein is relevant to the amino acid composition,” *J. Biochem.*, 1986, doi: 10.1093/oxfordjournals.jbchem.a135454.

[46] K.-C. Chou, “Pseudo Amino Acid Composition and its Applications in Bioinformatics, Proteomics and System Biology,” *Curr. Proteomics*, 2009, doi: 10.2174/157016409789973707.

[47] L. Nanni, A. Lumini, and S. Brahnam, “An empirical study on the matrix-based protein representations and their combination with sequence-based approaches,” *Amino Acids*, 2013, doi: 10.1007/s00726-012-1416-6.

[48] S. Whalen and G. Pandey, “A comparative analysis of ensemble classifiers: Case studies in genomics,” 2013, doi: 10.1109/ICDM.2013.21.

[49] C. Caragea, J. Sinapov, A. Silvescu, D. Dobbs, and V. Honavar, “Glycosylation site prediction using ensembles of Support Vector Machine classifiers,” *BMC Bioinformatics*, 2007, doi: 10.1186/1471-2105-8-438.

[50] A. Jones, H. M. Davies, and T. A. Voelker, “Palmitoyl-acyl carrier protein (ACP) thioesterase and the evolutionary origin of plant acyl-ACP thioesterases,” *Plant Cell*, 1995, doi: 10.1105/tpc.7.3.359.

[51] T. A. Voelker and H. M. Davies, “Alteration of the specificity and regulation of fatty acid synthesis of Escherichia coli by expression of a plant medium-chain acyl-acyl carrier protein thioesterase,” *J. Bacteriol.*, 1994, doi: 10.1128/jb.176.23.7320-7327.1994.

[52] F. Jing, L. Zhao, M. D. Yandeau-Nelson, and B. J. Nikolau, “Two distinct domains contribute to the substrate acyl chain length selectivity of plant acyl-ACP thioesterase,” *Nat. Commun.*, 2018, doi: 10.1038/s41467-018-03310-z.

[53] A. Ben-Hur, C. S. Ong, S. Sonnenburg, B. Schölkopf, and G. Rätsch, “Support vector machines and kernels for computational biology,” *PLoS Comput. Biol.*, 2008, doi: 10.1371/journal.pcbi.1000173.

[54] Y. Yang *et al.*, “Boundary thickness and robustness in learning models,” *arXiv*. 2020.

[55] K. Katoh, K. Misawa, K. I. Kuma, and T. Miyata, “MAFFT: A novel method for rapid multiple sequence alignment based on fast Fourier transform,” *Nucleic Acids Res.*, 2002, doi: 10.1093/nar/gkf436.

[56] F. Pedregosa *et al.*, “Scikit-learn: Machine learning in Python,” *J. Mach. Learn. Res.*, 2011.

[57] B. E. Boser, I. M. Guyon, and V. N. Vapnik, “Training algorithm for optimal margin classifiers,” 1992, doi: 10.1145/130385.130401.

[58] M. Aly and <malaa@caltech Edu>, “Survey on multiclass classification methods,” *Neural Netw*, 2005.

[59] NumPy, “NumPy — NumPy,” *NumPy Website*, 2017. .