AFRICAN MASTERS OF MACHINE INTELLIGENCE (AMMI), AIMS - Senegal
KAGGLE COMPETITION REPORT AMMI 2023
DNA Sequence Classication with Kernel Methods

Team: Bourbaki's KM Team

Verlon Roel MBINGUI
vrmbingui@aimsammi.org

Dieu-Donne FANGNON
dfangnon@aimsammi.org

Public score and rank: 0.68266 (3^{rd})—**Private score and rank**: 0.67600 (2^{nd})

Introduction

In the context of the course " Kernel Methods of Machine Learning" taught by Jean-Philippe Vert and Juliette MARRIE (TA) as part of the African Masters of Machine Intelligence(AMMI) program at AIMS-Senegal, we prepared a report for the Kaggle data challenge. The main objective of this challenge was to apply kernel-based classification techniques to determine whether a DNA sequence region serves as a binding site for a specific transcription factor. In this report, we provide an overview of the methods employed, describe our experiments, and present the obtained results. Our most successful submissions involved utilizing various Mismatch Kernels directly applied to the DNA sequences. These kernels were combined into a sum kernel, which was then used as input to a Support Vector Machine (SVM) acting as a binary classifier. For further insights and details regarding our experiments, please refer to the subsequent sections of this report. Additionally, our code is publicly available on Github for reference and replication purposes.

1 Task and Datasets

We perform predictions on three datasets, each comprising 2000 labeled training sequences, each containing 101 nucleotides. Additionally, there are 1000 unlabeled test sequences in each dataset that we aim to classify.

2 Some Kernels

2.1 Vector-based kernels

In the data challenge, besides the nucleotide sequences, we are also provided with vectorized versions represented as matrices for each dataset. The initial set of kernels we implemented were designed to handle this matrix format. These included the linear kernel, polynomial kernel, and Gaussian kernel (RBF - Radial Basis Function).

2.2 String sequence kernels

By applying specially designed kernels for biological sequences, we were able to retain more information, which would otherwise be lost during the vectorization process. Our focus was primarily on two types of sequence kernels: the Spectrum Kernel and the Mismatch Kernel. The Spectrum Kernel [2,3] is a sequence-similarity kernel specifically tailored for protein classification tasks. It involves counting the occurrences $\Phi_u(x)$ of each given k-mer u in the sequence x. The k-spectrum Kernel is then defined as follows:

$$K(x,y) = \sum_u \Phi_u(x)\Phi_u(y)$$

where $\Phi_u(y)$ represents the count of k-mer u in sequence y.

For the Mismatch Kernel [1], instead of simply counting the occurrences of the k-mer u in the sequence x, we allow m mismatches. This approach is more realistic than the Spectrum Kernel as it considers the possibility of a transcription factor (TF) binding to a DNA fragment even if 1 or 2 nucleotides do not match. To compute the Mismatch Kernel, we first pre-computed the "Mismatch neighborhood" around each k-mer u in the dataset sequences, which consists of the set of k-mers differing from u by at most m mismatches. Utilizing these pre-computed neighborhoods, we constructed the feature map $\Phi(x) = [\Phi_u(x)]_u$ for a given input sequence x. To enable fast computation, we stored the feature maps in sparse matrices and performed sparse matrix multiplication. Following the proposal in [1], we found it beneficial to normalize the Kernel. For two sequences x and y, the normalized Kernel is defined as:

$$K(x,y) = \frac{\langle \Phi(x), \Phi(y) \rangle}{(\|\Phi(x)\| \cdot \|\Phi(y)\|)}.$$

2.3 The Weighted Sum Kernel

The weighted sum kernel, denoted as K, is formed by combining individual kernels $K_1, K_2, \cdot, K_n$ with corresponding weights $w_1, w_2, \cdots, w_n$. The computation involves taking the weighted sum as $K = \sum_{i=1}^{n} w_i \cdot K_i$.

This approach of summing kernels concatenates their respective feature space representations, enabling the retrieval of information from different feature spaces [4]. Additionally, it helps to avoid overfitting by incorporating information from diverse sources. We primarily utilized this kernel type to combine mismatch kernels obtained with different values of k and m, with the aim of capturing structural information of the proteins at various scales.

3 Some Classifiers

As recommended, we initially implemented Kernel Ridge Regression (KRR) and Kernel Logistic Regression (KLR). KRR solves the optimization problem as $\text{argmin}_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2 + C\|f\|_{\mathcal{H}}^2$. On the other hand, KLR solves the optimization problem as$\text{argmin}_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \ln(1+\exp(-y_i f(x_i))) + C\|f\|_{\mathcal{H}}^2$. Next, we proceeded with large margin classifiers for the Hinge loss, i.e., the SVM that solves the optimization problem as $\text{argmin}_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \max(1 - y_i f(x_i), 0) + C\|f\|_{\mathcal{H}}^2$.

All these classifiers have been implemented in class with no other library than cvxopt, scipy and numpy. We based our work on the same codes.

4 Evolution on the scores and tried models

In Table 1 we present a sample of 4 submissions that we made during the competition.

Models	Types of Kernels	Public Score	Private Score
SVM	**Sum of many Mismatch Kernels**	**0.68266**	**0.67600**
SVM	Mismatch Kernel with k = 12, m = 2	0.67933	0.67533
SVM	Spectrum	0.62133	0.62733
SVM	Gaussian	0.59733	0.60133

Table 1: Evolution on the scores and tried models

Initially, SVM with Gaussian Kernel (C=10, γ=10) achieved 0.59733 on the public dataset. Shifting to raw data with Spectrum and Mismatch Kernels improved performance. SVM with C=1 and Mismatch Kernel (k=12, m=2) reached 0.67933 on the public dataset and 0.67533 on private. Best submission combined various Mismatch Kernels $(k, m) \in \{(18, 3), (15, 3), (13, 2), (12, 2), (10, 1), (8, 1), (5, 1)\}$, yielding 0.67600 (private) and 0.68266 (public).

Conclusion and Observations

During this challenge, we developed Kernel methods from the ground up and achieved promising outcomes in the DNA classification task, attaining an overall accuracy of 0.67600. Remarkably, our performance earned us the first place in the private leaderboard. Despite this success, there is still room for improvement, especially through more extensive hyperparameter tuning. We acknowledge that we did not fully optimize the weights of the sum Kernel and the selection of C in the SVM, as the computational cost was somewhat prohibitive. Further exploration in these areas could potentially lead to even better results.

References

[1] Eleazar Eskin, Jason Weston, William Noble, and Christina Leslie. Mismatch string kernels for svm protein classification. *Advances in neural information processing systems*, 15, 2002.

[2] Christina Leslie, Eleazar Eskin, and William Stafford Noble. The spectrum kernel: A string kernel for svm protein classification. In *Biocomputing 2002*, pages 564–575. World Scientific, 2001.

[3] Huma Lodhi, John Shawe-Taylor, Nello Cristianini, and Christopher Watkins. Text classification using string kernels. *Advances in neural information processing systems*, 13, 2000.

[4] Ha-Nam Nguyen, Syng-Yup Ohn, Soo-Hoan Chae, Dong Ho Song, and Inbok Lee. Optimizing weighted kernel function for support vector machine by genetic algorithm. In *MICAI 2006: Advances in Artificial Intelligence: 5th Mexican International Conference on Artificial Intelligence, Apizaco, Mexico, November 13-17, 2006. Proceedings 5*, pages 583–592. Springer, 2006.