PREDICTING BACTERIA SPECIES BASED ON REPEATED LOSSY MEASUREMENTS OF DNA SNIPPETS

Solution to the Kaggle's Tabular Playground Series (Feb 2022) competition

Presented by Giulio Vaccari

03/2022

INTRO: K-MER SEQUENCES

- In bioinformatics, k-mers are substrings of length k contained within a biological sequence
- Primarily used within the context of computational genomics and sequence analysis, in which k-mers are composed of nucleotides (i.e. A, T, G, and C)
- An example of 10-mer is the following DNA segment: **ATATGGCCTT**
- In this task we will work with data from a genomic analysis technique that has some data compression and data loss
- In this technique, 10-mer snippets of DNA from bacteria are sampled and analyzed to give the histogram of base count
- In other words, the DNA segment **ATATGGCCTT** becomes $\mathbf{A_2T_4G_2C_2}$

THE TASK

- The task is to classify 10 different bacteria species using data extracted from their DNA
- Each bacterium sample will be described by the normalized spectrum of 286 histograms generated from its DNA
- In order to generate this spectrum, several 10-mers were sampled from the bacterium DNA and it has been counted how many times each of the 286 histogram possibilities (e.g., $\mathbf{A_0T_0G_0C_{10}}$ to $\mathbf{A_{10}T_0G_0C_0}$) occurred
- Every species has its own characteristic spectrum, and we must predict the bacterium's name from the spectrum of the sample

THE DATASET

- Large dataset with 200000 rows
- Each row of data contains the spectrum of histograms generated by repeated measurements of a bacterium sample, each row containing the output of all 286 histogram possibilities, which then has a bias spectrum (of totally random ATGC) subtracted from the results
- The data (both train and test) also contains simulated measurement errors (of varying rates) for many of the samples, which makes the problem more challenging
- Possible problem: Curse of dimensionality

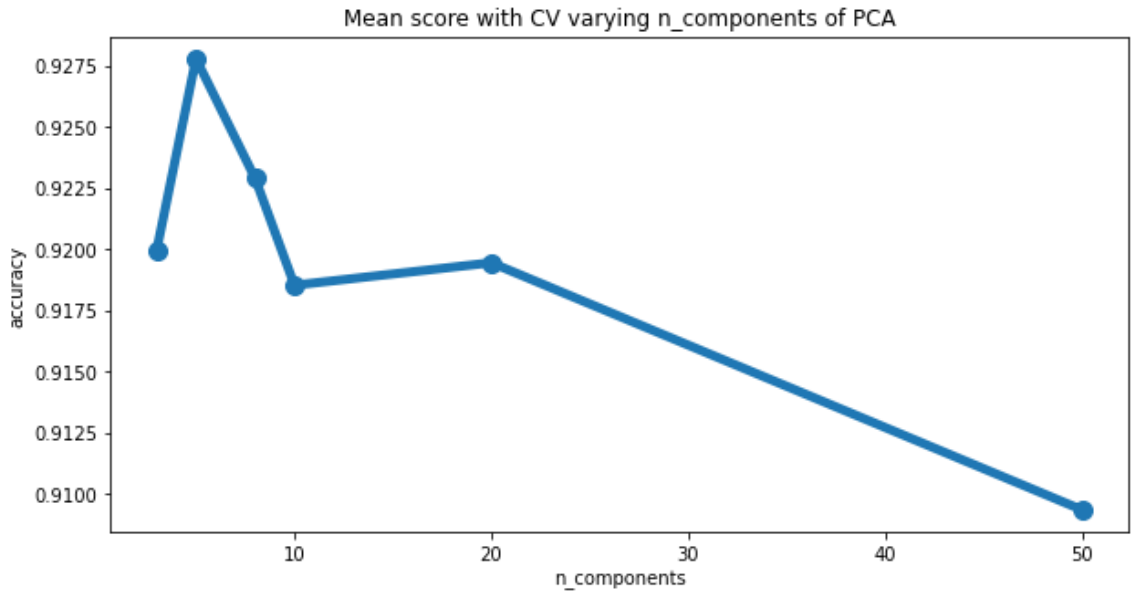
PROPOSED SOLUTION

DIMENSIONALITY REDUCTION

- Principal Component Analysis (**PCA**) to project each data point onto only the first few principal components to obtain lower-dimensional data while preserving as much of the data's variation as possible
- Feature standardization before PCA in order to remove the mean and scaling to unit variance

GRID SEARCH ON PCA

- Grid-search with cross validation to find the best number of output dimensions from the PCA transformation
- Decision Tree Classifier used to produce reference scores
- The best transformation found reduced the number of features from 286 to less than 10

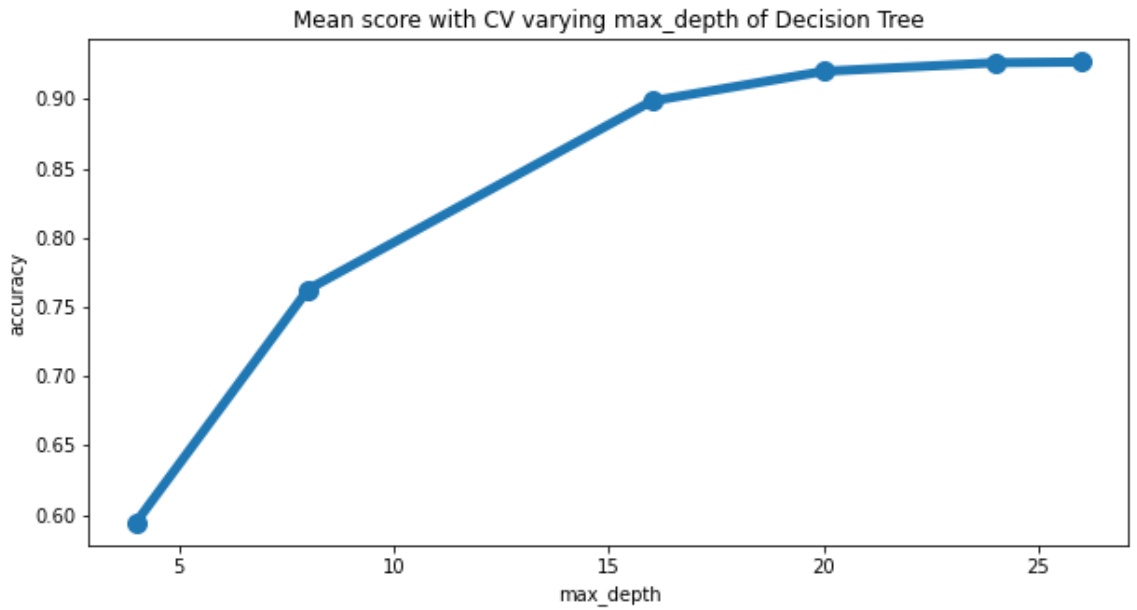


MODEL SELECTION

- Four different machine learning models have been considered:
  - Decision Tree
  - Random Forest
  - Extremely Randomized Trees
  - Deep Neural Network
- The first three non-deep models were tested using grid search with cross validation to perform hyper-parameters tuning
- The deep neural network were trained and validated using a more classical train-validation dataset split

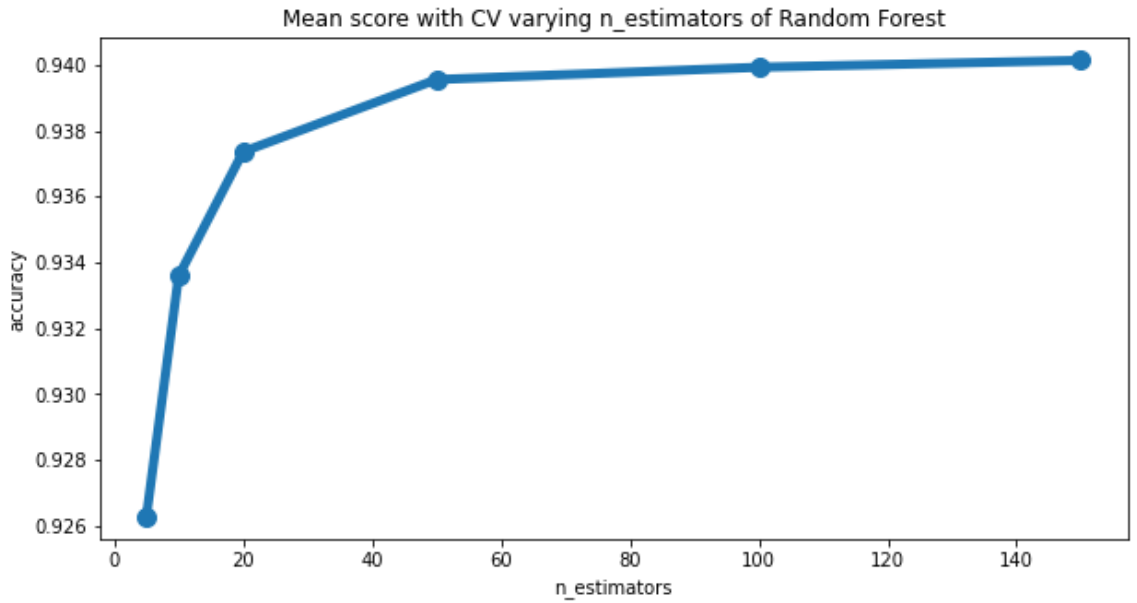
DECISION TREE

- The decision tree was tested varying its max depth in order to obtain a good variance-bias trade-off



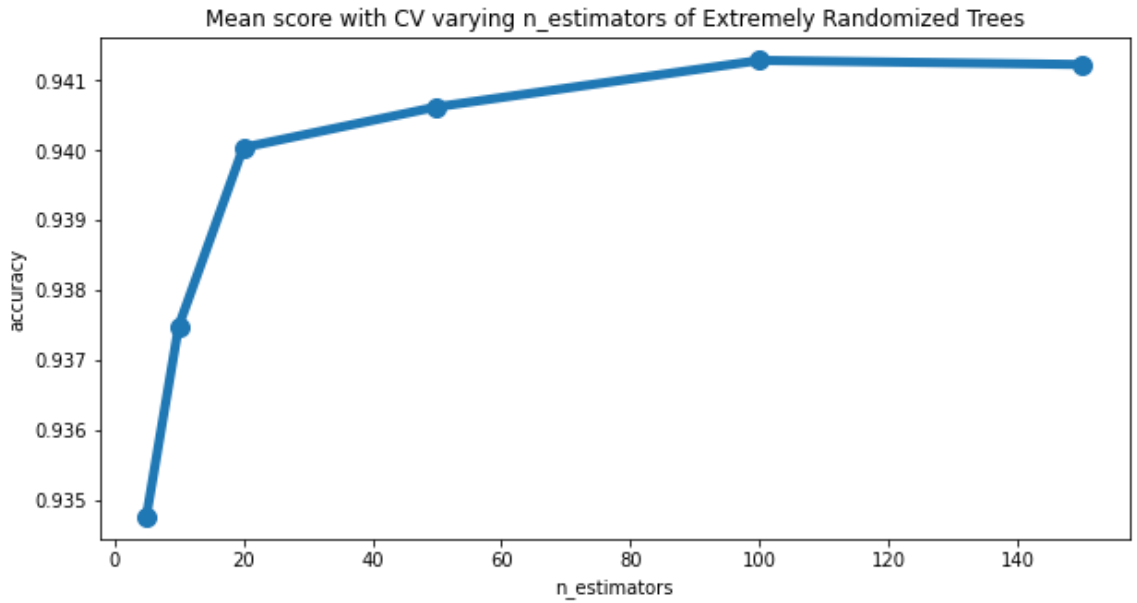
RANDOM FOREST

- In random forests each tree in the ensemble is built from a sample drawn with replacement from the training set
- Furthermore, when splitting each node during the construction of a tree, the best split is found from a random subset of the input features
- It has been tested varying the number of trees to use in the ensemble



EXTREMELY RANDOMIZED TREES

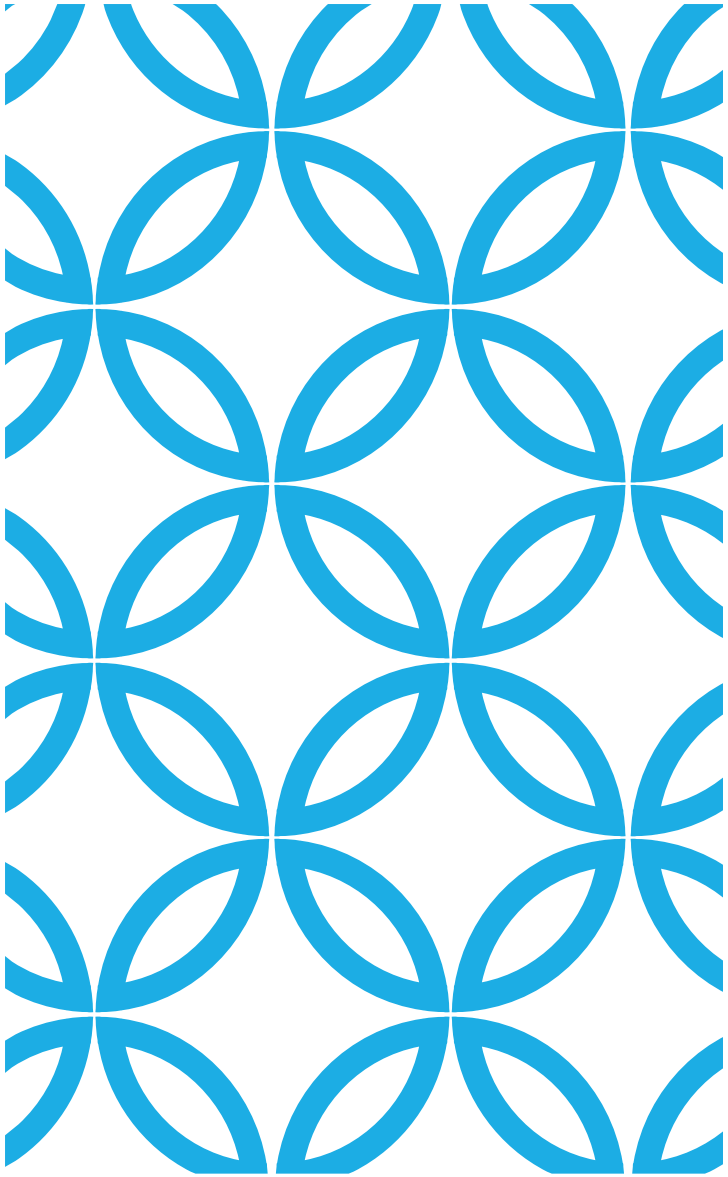
- In extremely randomized trees, randomness goes one step further compared to random forest in the way splits are computed
- As in random forests, a random subset of candidate features is used, but instead of looking for the most discriminative thresholds, thresholds are drawn at random for each candidate feature and the best of these randomly-generated thresholds is picked as the splitting rule



DEEP NEURAL NETWORK

- Deep network with 6 layers and a final softmax activation function
- It produces a probability distribution over the target classes given the input features
- After 10 epochs of training with minibatches of 64 samples, it achieved an accuracy of 0.83 on the validation set
- Worse performances compared to tree-based models

FINAL PIPELINE

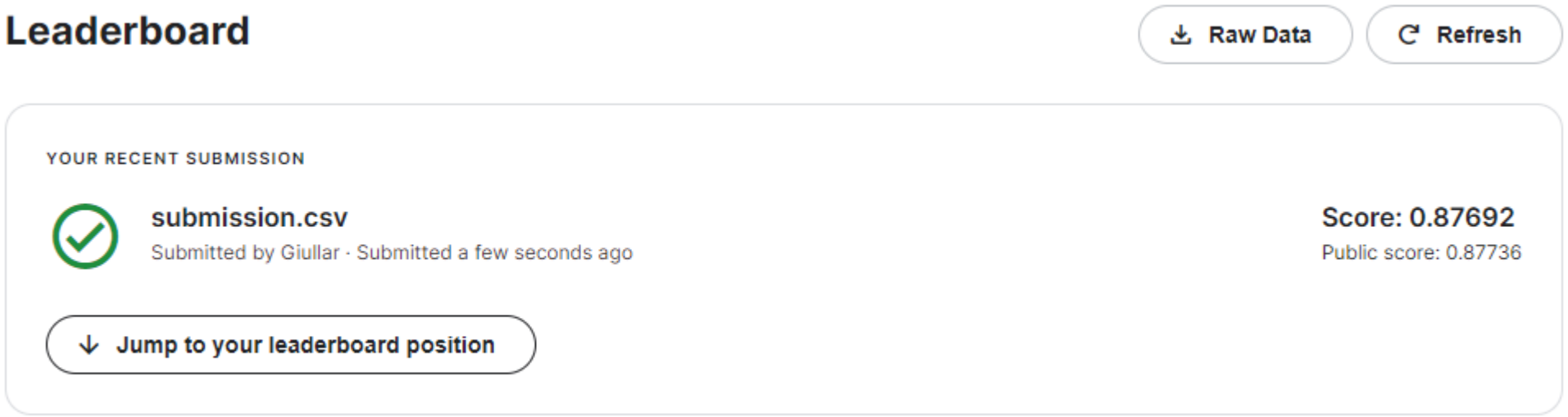


FINAL MODEL EVALUATION

RESULTS ON TEST SET

	precision	recall	f1-score	support
Bacteroides_fragilis	0.98	0.99	0.99	6646
Campylobacter_jejuni	0.98	0.99	0.99	6621
Enterococcus_hirae	0.97	0.97	0.97	6583
Escherichia_coli	0.97	0.96	0.96	6586
Escherichia_fergusonii	0.96	0.97	0.97	6579
Klebsiella_pneumoniae	0.99	0.99	0.99	6549
Salmonella_enterica	0.98	0.98	0.98	6610
Staphylococcus_aureus	0.98	0.98	0.98	6577
Streptococcus_pneumoniae	0.98	0.97	0.97	6624
Streptococcus_pyogenes	0.97	0.97	0.97	6625
accuracy			0.98	66000
macro avg	0.98	0.98	0.98	66000
weighted avg	0.98	0.98	0.98	66000

KAGGLE SUBMISSION RESULTS



REFERENCES

- Kaggle competition:
  - https://www.kaggle.com/c/tabular-playground-series-feb-2022/overview
- What is a K-mer:
  - https://en.wikipedia.org/wiki/K-mer
- The idea for this competition came from the following paper:
  - Wood et al., 2020, "Analysis of Identification Method for Bacterial Species and Antibiotic Resistance Genes Using Optical Data From DNA Oligomers", Frontiers in Microbiology, https://www.frontiersin.org/article/10.3389/fmicb.2020.00257