Pattern Recognition and Machine Learning
(2022)
Assignment 8: ICA & Sequential Feature Selector
Deadline: April 3rd, 2022 23:59

Guidelines for submission

1. Perform all tasks in a single colab file.
2. Create a report regarding the steps followed while performing the given tasks. The report should not include excessive unscaled preprocessing plots.
3. Try to modularize the code for readability wherever possible
4. Submit a separate colab file [.ipynb] and report [.pdf] on the classroom
5. Submit the .py file on the floated form for the laboratory
6. Plagiarism will not be tolerated

Question 1. **[65 marks]**

Independent component analysis (ICA) is used to estimate sources given noisy measurements. ICA is a computational method for separating a multivariate signal into its underlying components. Using ICA, we can extract the desired component (i.e. conversation between you and some other person) from the amalgamation of multiple signals.

Download files from the link Music and perform the following tasks
The folder contains three signal files of piano, music and some noise respectively.

1) Read, Visualize and Listen the audio files. **[5 marks]**
2) Extract raw audio from the three wave files and merge them to create dataset X. **[5 marks]**
3) Implement ICA from scratch. For Convergence select either 1000 iterations, or when the dot product of w (demixing matrix) and its transpose is roughly equal to 1. **[35 marks]**
4) Plot mixture, real source and predicted source from the output of ICA. **[5 marks]**
5) Implement Fast ICA (import from sklearn.decomposition) selecting num_components = 3 **[5 marks]**
6) Separate, Visualize and Listen the independent component obtained from task 5. **[5 marks]**
7) Comment on the results obtained from ICA and Fast ICA and report observations in the report. **[5 marks]**

Note: For task 3 only you need to implement from scratch, for the other tasks you can use predefined libraries.

Question 2. **[35 marks]**

Sequential feature selection algorithms are a family of greedy search algorithms that are used to reduce an initial *d*-dimensional feature space to a *k*-dimensional feature subspace where $k < d$. The motivation behind feature selection algorithms is to automatically select a subset of features that is most relevant to the problem.

In a nutshell, SFAs remove or add one feature at the time based on the classifier performance until a feature subset of the desired size *k* is reached.

Install the below library for using SFS algorithms.

pip install mlxtend

Import SFS using the below

from mlxtend.feature_selection import SequentialFeatureSelector as SFS

Download dataset from the given link AirlinePassenger and perform the following tasks
There is a separate file for train and test. Download only the train.csv file.

1) Preprocess, clean and prepare the dataset based on the previous lab experience. Separate features and labels as X and Y respectively. **[5 marks]**
2) Create an object of SFS by embedding Decision Tree classifier object, providing 10 features, forward as True, floating as False and scoring = accuracy. **[5 marks]**
3) Train SFS and report accuracy for all 10 features. Also list the names of the 10 best features selected by SFS. **[5 marks]**
4) Using the forward and Floating parameter toggle between SFS(forward True, floating False), SBS (forward False, floating False), SFFS (forward True, floating True), SBFS (forward False, floating True), and choose cross validation = 4 for each configuration. Also report cv scores for each configuration. **[5 marks]**
5) Visualize the output from the feature selection in a pandas DataFrame format using the get_metric_dict for all four configurations. **[5 marks]**
6) Finally plot the results for each configuration (from mlxtend.plotting import plot_sequential_feature_selection as plot_sfs). **[5 marks]**
7) For task 2 vary the features by increasing or decreasing, observe and report the results. **[5 marks]**

Note: Bonus marks will be given if implemented any variant of SFS from scratch