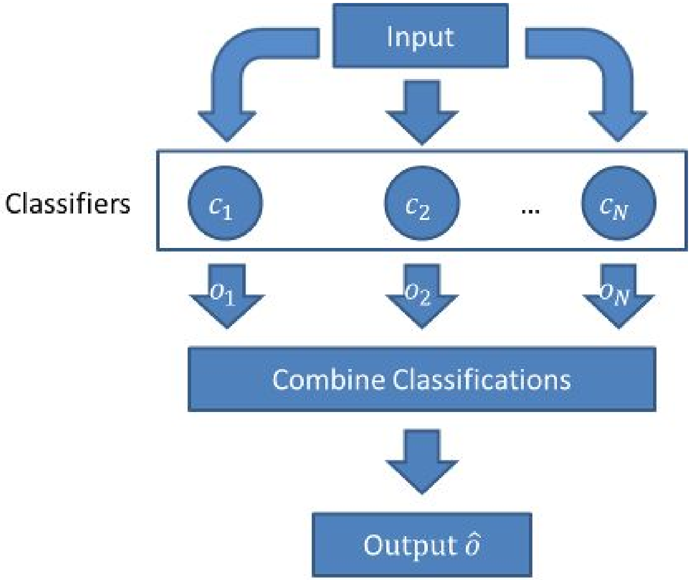
Feature Selection Inspired Classifier Ensemble Reduction

Prajwal Kailas (prajwal967@gmail.com),
14CO132, 9632080195
National Institute Of Technology Karnataka

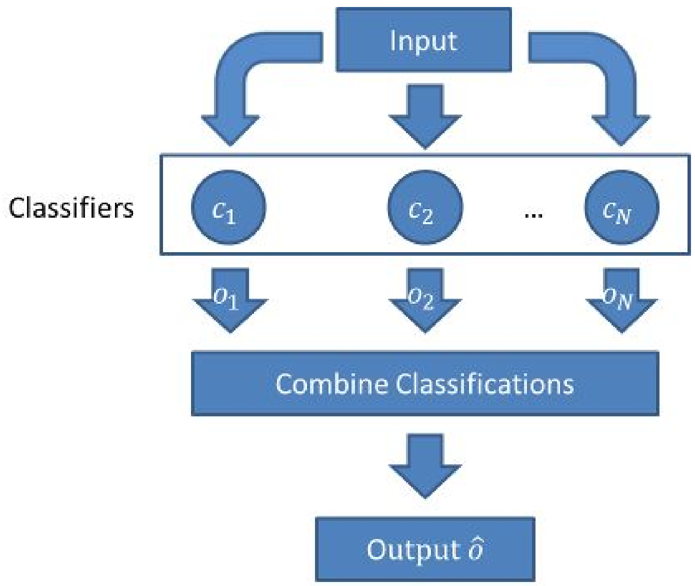
Introduction

Ensemble learning is a machine **learning** paradigm where multiple **learners** are trained to solve the same problem. In contrast to ordinary machine **learning** approaches which try to learn one hypothesis from training data, **ensemble** methods try to construct a set of hypotheses and combine them to use.



Introduction

Classifier Ensemble Reduction involves reducing the number of classifiers, C1, C2...CN to C1, C2...CK, where K<N. This is done such that the final output does not change. The **target** of classifier ensemble reduction is to reduce the amount of redundancy in a pre constructed classifier ensemble, to form a much reduced subset of classifiers that can still deliver the **same classification results**.



Objectives

:

➔ **Ensemble**

Build an ensemble of classifiers.

➔ **Feature Selection**

Locate the optimal feature subset.

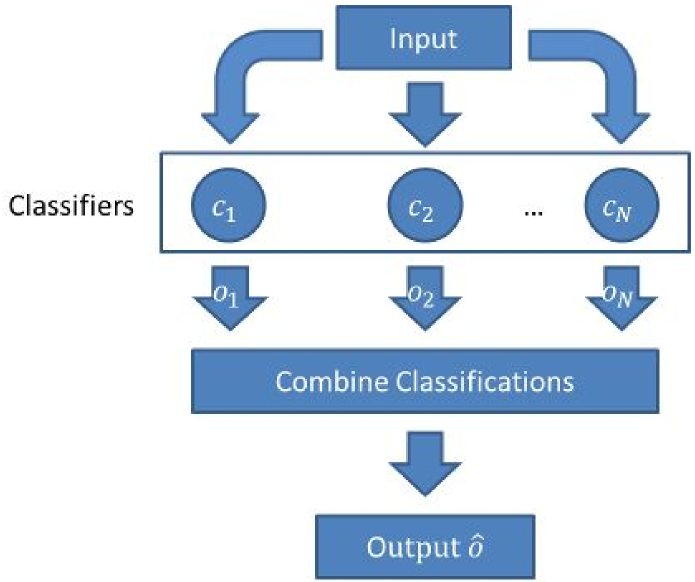
➔ **Ensemble Reduction**

Optimal subset of classifiers, the final output.

KEY METHODOLOGIES /TECHNIQUES in Feature Selection Inspired Classifier Ensemble Reduction

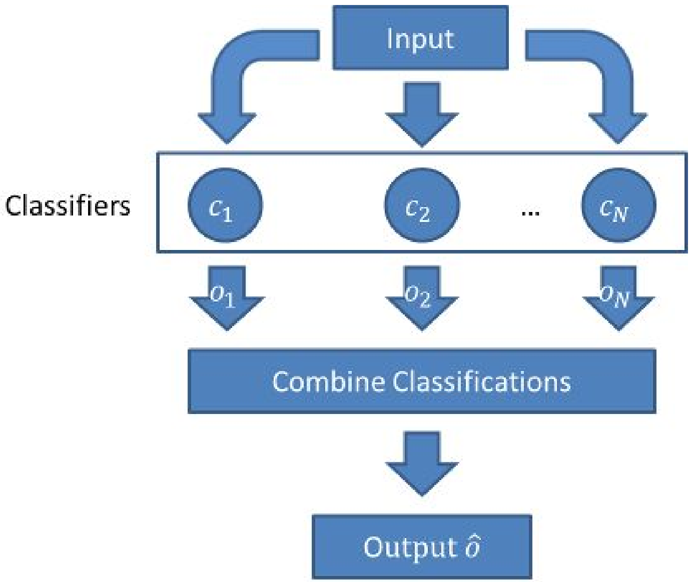
Ensemble Methodologies

The first step is to build the base classifiers, classifiers such as **Gradient Boosting (XGBoost), Multi Layer Perceptron, Random Forest Classifier, Decision Tree, Linear Regression, Logistic Regression** can be used. The desired parameter values for these models are selected and then the base models are trained.



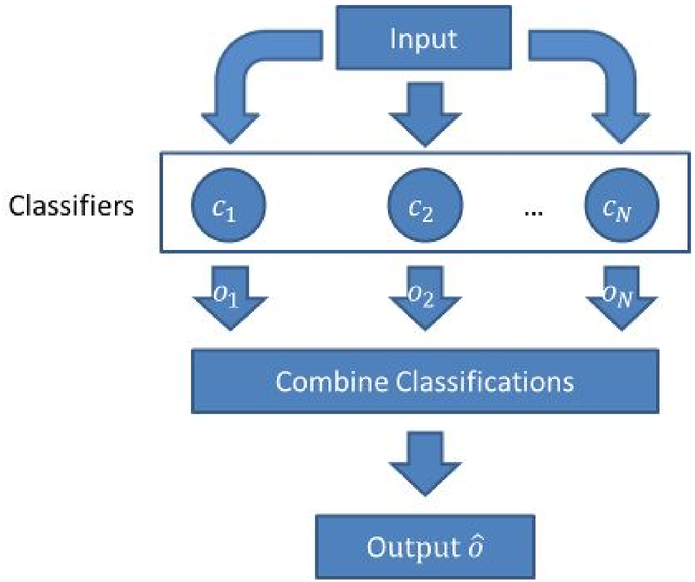
Ensemble Methodologies

Upon training the models, the trained models are used to obtain predictions on the cross validation data, the predictions obtained from the base models will be used to construct a data frame (a new dataset), which will be used to train the **ensemble classifier**. The techniques that can be used for building an ensemble classifier are **stacking, blending, weighted average** etc.



Feature Selection With Harmony Search

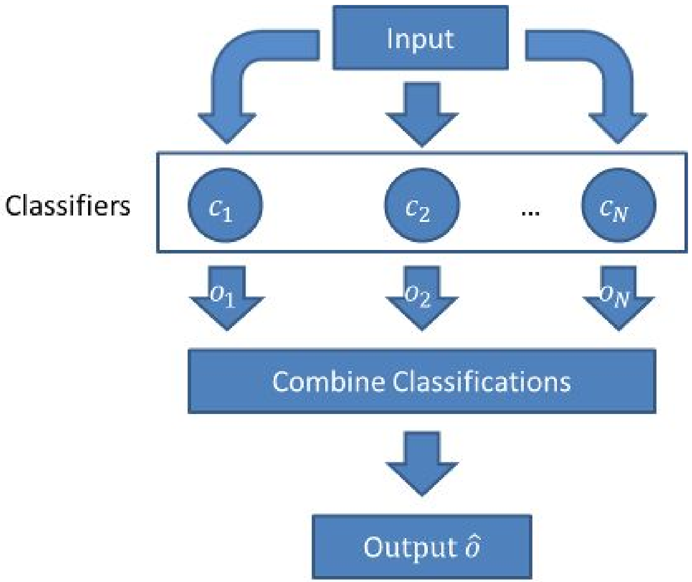
The objective of **harmony search** is to find a **solution vector** which is an **optimized** version of the **cost function**. Harmony search is based on musicians and notes. The analogy is, the a musician is a decision variable, the notes generated by the musician are values corresponding to the decision variables.



Feature Selection With Harmony Search

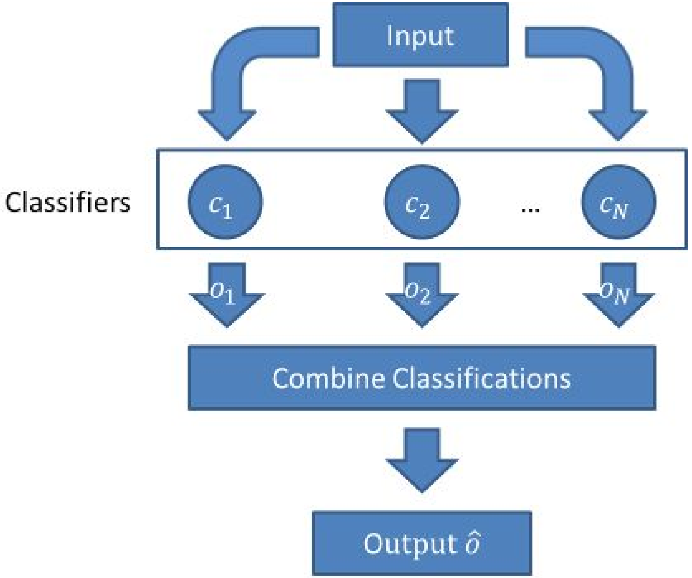
The variables and values are then permuted to find the global optimum which is the best harmony that is produced by the notes played by the musician.

In computational terms, the musician is analogous to the **feature selector**, and the **features** are analogous to the notes.



Feature Selection With Harmony Search

Each musician votes/selects a feature that needs to be included in the harmony. Finally a **voting** methodology is used to select the **optimum** notes/features which forms the optimum **harmony/cost function**.

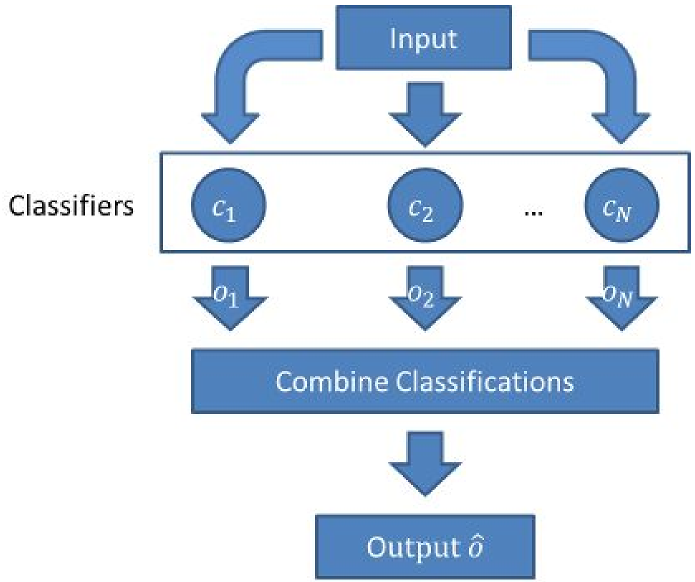


Approaches

1. Classifier Ensemble Reduction

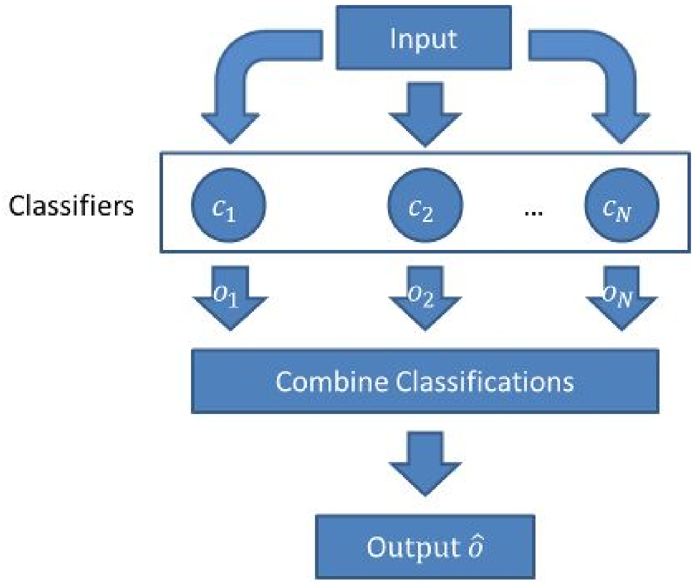
Classifier Ensemble Reduction

The first step is forming a pool of **base classifiers**, any machine learning classifier can be used in forming the pool. Once the base classifiers are constructed their predictions are gathered. Using these predictions a **new dataset** is constructed.



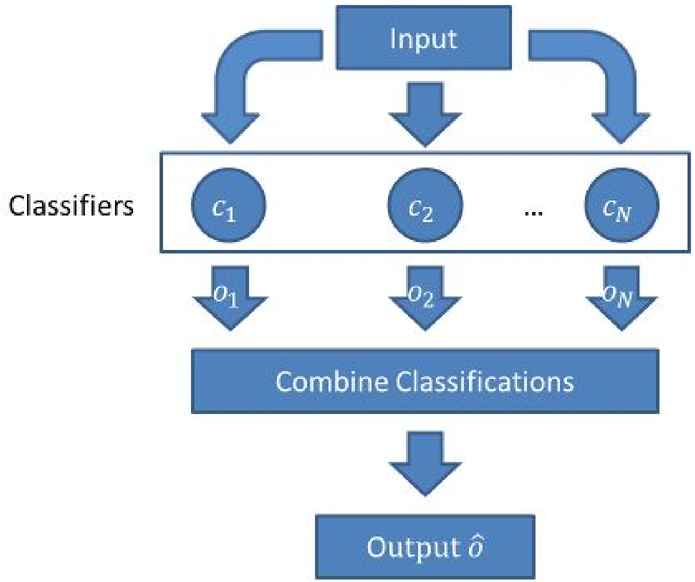
Classifier Ensemble Reduction

In the new dataset, each column represents an artificially generated feature, each row corresponds to a training instance. This can be thought of as a **ensemble decision matrix**.



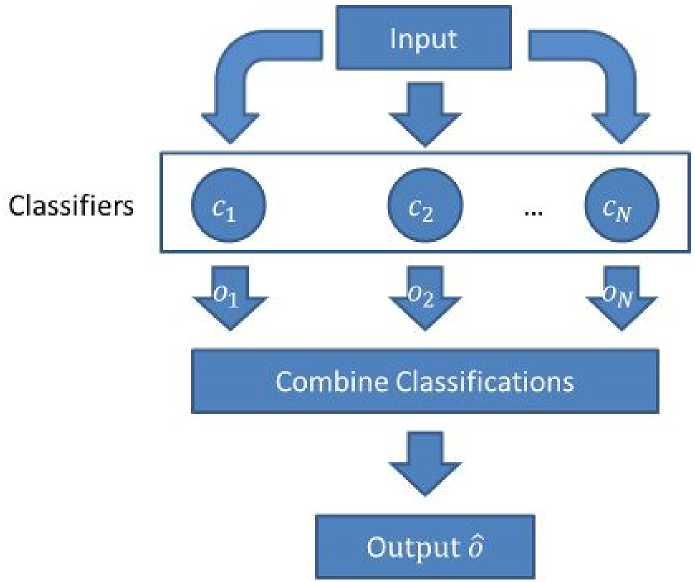
Classifier Ensemble Reduction

Harmony search is then performed on this new dataset, and an **optimal subset** of features from this dataset is selected. Finally this optimal feature subset is used to build the ensemble classifier. Once the **classifier ensemble** is constructed, new objects are classified by the ensemble members, and their results are aggregated to form the final ensemble decision output.



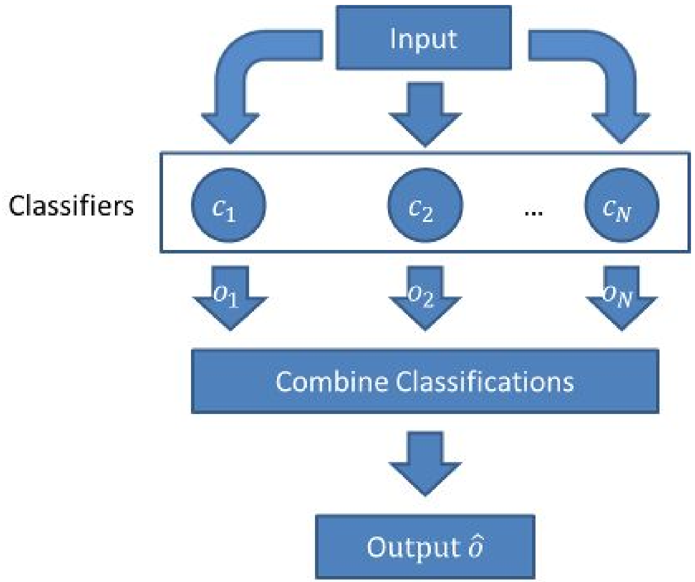
Challenges

Testing with UCI benchmarked datasets which have large size and dimension, there are significant challenges in **constructing** and **reducing** the ensembles.



Issues

Since the optimal subset calculation varies, the overall/average **execution** time varies, so it is difficult to accurately measure the execution time. Also a substantial amount of time is required in the **reduction** process.



Advantages

- Reduces memory usage.
- Reduces system runtime and computational overhead.

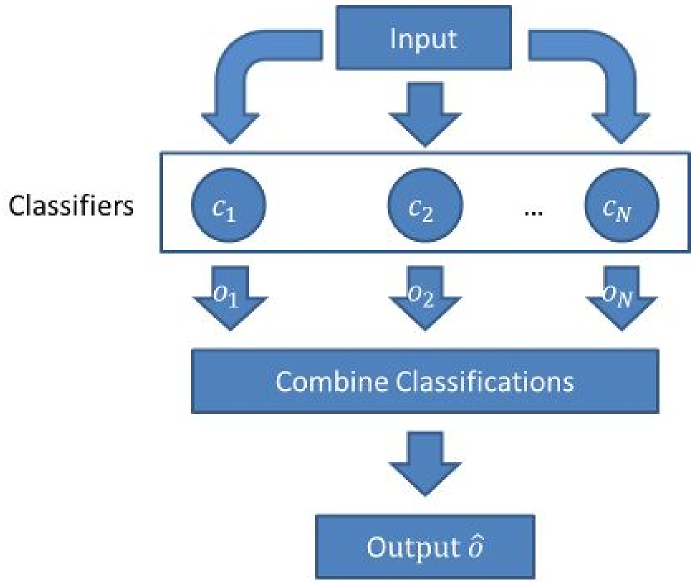
Disadvantages

- There can be a reduction in accuracy
- Harmony search is expensive

Implementation

The implementation process will contain:

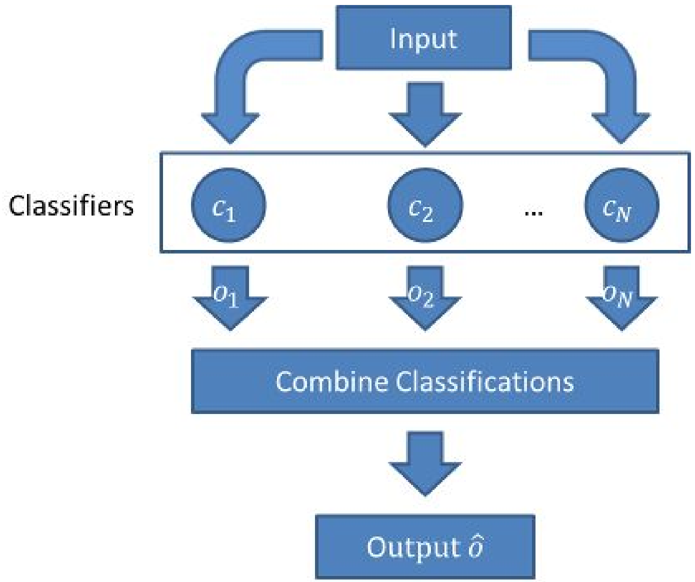
- Forming a pool of base classifiers, the algorithms included will be **Linear Regression, Gradient Boosting (XGBoost), Random Forest Classifier, Decision Tree, Logistic Regression, Multilayer Perceptron.**
- Tweak the parameters of these models and select optimum values.
- Building an ensemble classifier using **stacking, blending** and **weighted average** techniques.



Implementation

The implementation process will contain:

- Apply the harmony search algorithm to the ensemble classifier.
- Test the final ensemble classifier with a benchmarked dataset.
- Language: Python
- Libraries: functools, numpy, sklearn, keras, xgboost, joblib, hyperopt, category_encoders



Summary

The **Classifier Ensemble Reduction** is based on applying feature selection techniques to minimize redundancy (reduce the size of the ensemble) which is done by transforming an ensemble decision matrix. This is done while while maintaining and improving classification accuracy and efficiency.