Ensemble Methods

Ensemble Methods -

To combine the predictions of several base estimators built with a given learning algorithm in order to improve generalizability / robustness over a single estimator

Two families of ensemble methods are usually distinguished:

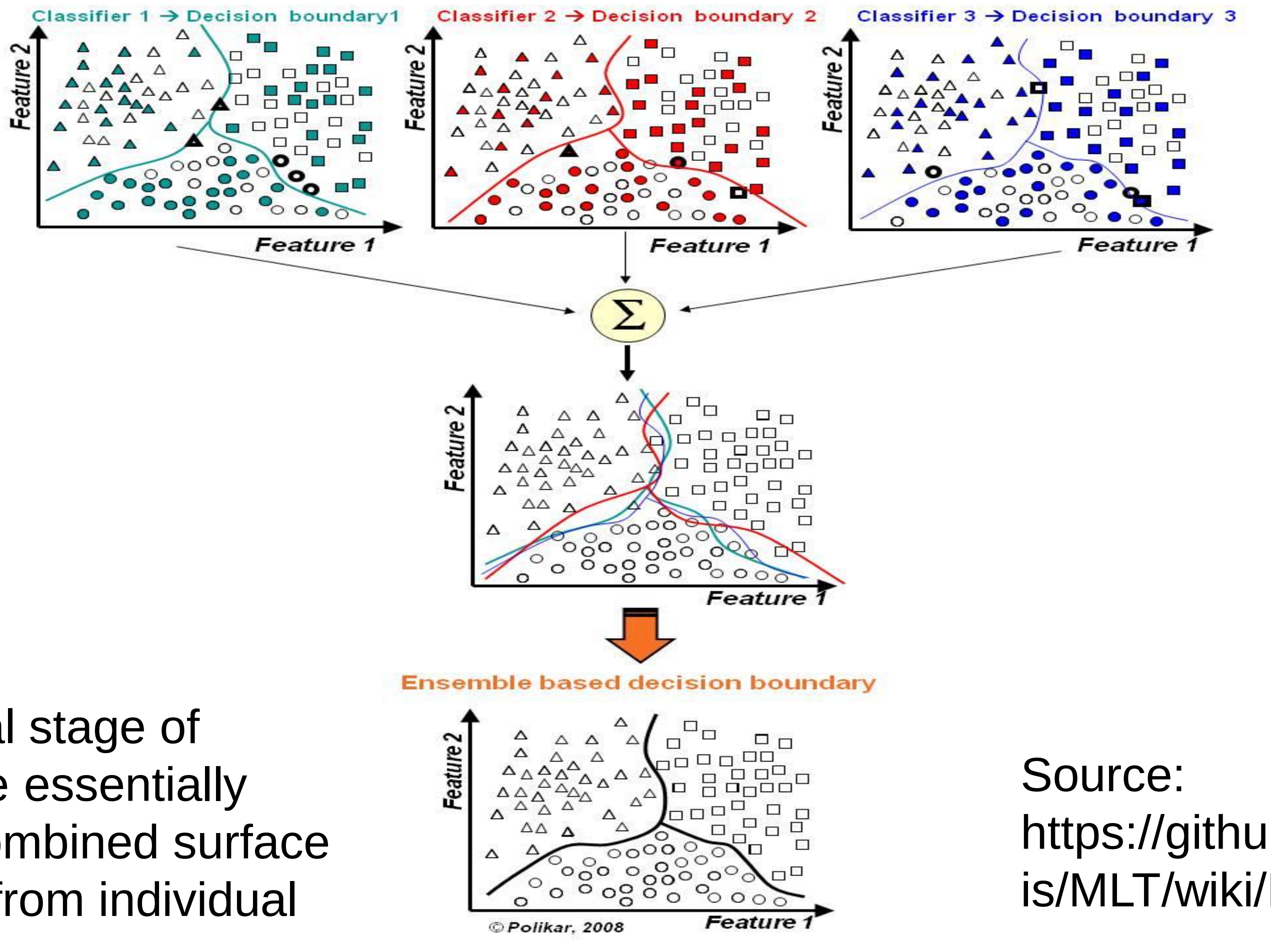
1. Averaging methods, the driving principle is to build several estimators independently and then to average / vote their predictions. On average, the combined estimator is usually better than any of the single base estimator because its variance is reduced.

   E.g. Bagging methods, Forests of randomized trees, ...

2. Boosting methods, base estimators are built sequentially and one tries to reduce the bias of the combined estimator. The motivation is to combine several weak models to produce a powerful ensemble. E.g. AdaBoost, Gradient Tree Boosting, ...

Introduction to machine learning

<u>Ensemble Methods</u> -



In the final stage of voting, we essentially have a combined surface resulting from individual surfaces

Source: https://github.com/MenuPolis/MLT/wiki/Bagging

Introduction to machine learning

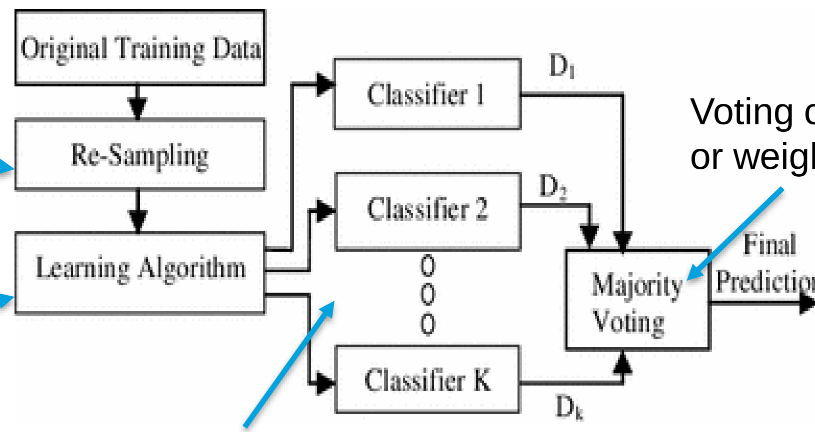
Ensemble Methods – Averaging method - **Bagging (B**ootstrap **Agg**regation) **:**

1. Designed to improve the stability and accuracy of classification and regression models
2. It reduces variance errors and helps to avoid overfitting
3. Can be used with any type of machine learning model, mostly used with Decision Tree
4. Uses sampling with replacement to generate multiple samples of a given size. Sample may contain repeat data points
5. For large sample size, sample data is expected to have roughly 63.2% (1 – 1/e) unique data points and the rest being duplicates
6. For classification bagging is used with voting to decide the class of an input while for regression average or median values are calculate

Introduction to machine learning

Ensemble Methods – Averaging method - **Bagging (B**ootstrap **Agg**regation) **:**

Re-sampling done for every classifier using a random function. For large n, 63.2% unique samples likely to be selected



Voting could be simple or weighted

Algorithm to generate classifiers. Could be Decision Tree, Naïve Bayes etc

K classifiers created in parallel and independently on respective training data

Source: https://link.springer.com/article/10.1007/s13721-013-0034-x

Ensemble Learning – **Bagging**:

Improve defaulter prediction of the decision tree using bagging ensemble technique

Description – Sample data is available at local file system as credit.csv

The dataset has 16 attributes described at https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
or in the notes page of this slide

Sol: Bagging+Credit+Decision+Tree.ipynb

Ensemble Methods – Boosting Method – **AdaBoosting** :

1. Similar to bagging, but the learners are grown sequentially; except for the first, each subsequent learner is grown from previously grown learners

2. If the learner is a Decision Tree, each of the trees can be small, with just a few terminal nodes (determined by the parameter d supplied)

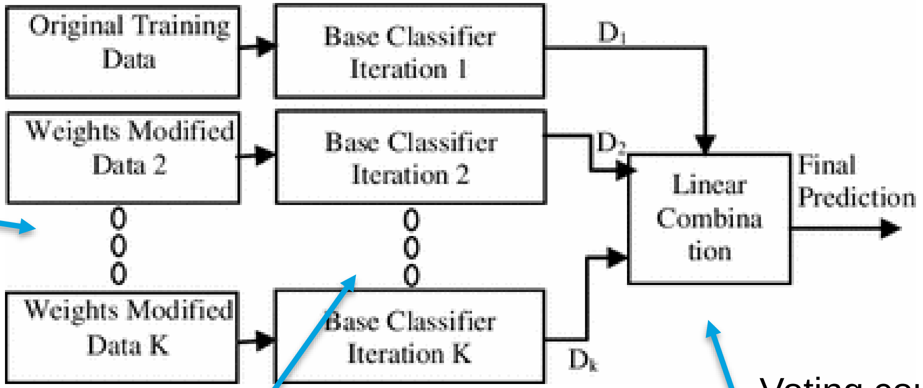
3. During voting higher weight is given to the votes of learners which perform better in respective training data unlike Bagging where all get equal weight

4. Boosting slows down learning (because it is sequential) but the model generally performs well

Introduction to machine learning

<u>Ensemble Methods</u> – Boosting method - **AdaBoosting:**

Training data from base data with focus on instances which were incorrectly classified by earlier model (if any)



K similar classifiers created in sequence with respective training data with focus on addressing the misclassified data, not the usual cost functions

Voting could be simple or weighted

It is called Adaptive Boosting as the weights are re-assigned to each instance, with higher weights to incorrectly classified instance

Source: https://link.springer.com/article/10.1007/s13721-013-0034-x

Introduction to machine learning

Ensemble Methods – Boosting Method – **AdaBoosting** :

7. Two prominent boosting algorithms are AdaBoost, short for Adaptive Boosting and Gradient Descent Boosting

8. In AdaBoost, the successive learners are created with a focus on the ill fitted data of the previous learner

9. Each successive learner focuses more and more on the harder to fit data i.e. their residuals in the previous tree

Introduction to machine learning

Ensemble Methods – Boosting Method – **AdaBoosting** :

Adapting weights with focus on erroneously classified instances

Given: $(x_1,y_1), \ldots, (x_m,y_m)$, *Where* $x_i \in X$, $y_i \in Y = \{1, 2, \ldots, K\}$

1. *Initialize the weights* $w_i^1 = 1/m$, $i=1, 2, \ldots, m$
2. *For t=1 to T*
   (a) *Fit a classifier* $h^t(x)$ *to the training data using weights* w_i^t
   (b) *Compute*

$$err^t = Pr_{i\sim w_i^t}[h^t(x_i) \neq y_i] = \sum_{i=1,\, h^t(x_i)\neq y_i}^{m} w_i^t \Big/ \sum_{i=1}^{m} w_i^t$$

If $err^t > 1/2$, *then* $t=T-1$ *and abort loop.*

   (c) *Compute*

$$\alpha^t = log\frac{1-err^t}{err^t}$$

   (d) *Set*

$$w_i^t \leftarrow \begin{cases} w_i^t.\exp(\alpha^t) & if \quad h^t(x_i) \neq y_i \\ w_i^t & otherwise \end{cases} \qquad i = 1, 2, \ldots, m$$

   (e) *Renormalize* w_i^t

3. *Output*

$$H(x) = \underset{y\in Y}{arg\,max} \sum_{t=1,\, h^t(x)=y}^{T} \alpha^t$$

Initialize weights, equal weights to all instances

Generate first classifier with equal focus on all instances

Total up weights of all error instances, express it as a ratio to total weights

If error ratio is > 50%

Calculate predictor weights (i.e. weight of the classifier)

Assign new weights to instances misclassified, else keep the weights same

Renormalize the weights across all the instances and fit next classifier

For a test instance use weighted voting to identify the class

Ensemble Learning – **AdaBoosting**:

Improve defaulter prediction of the decision tree using Adaboosting

Description – Sample data is available at local file system as credit.csv

The dataset has 16 attributes described at
https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
or in the notes page of this slide

Sol: Adaboost+Credit+Decision+Tree.ipynb

Ensemble Methods – Averaging Method – **Gradient Descent Boosting** :

1. Each learner is fit on a modified version of original data (original data is replaced with the x values and **residuals** from previous learner
2. By fitting new models to the residuals, the overall learner gradually improves in areas where residuals are initially high

Introduction to machine learning

Ensemble Methods – Averaging Method – **Gradient Descent Boosting** :

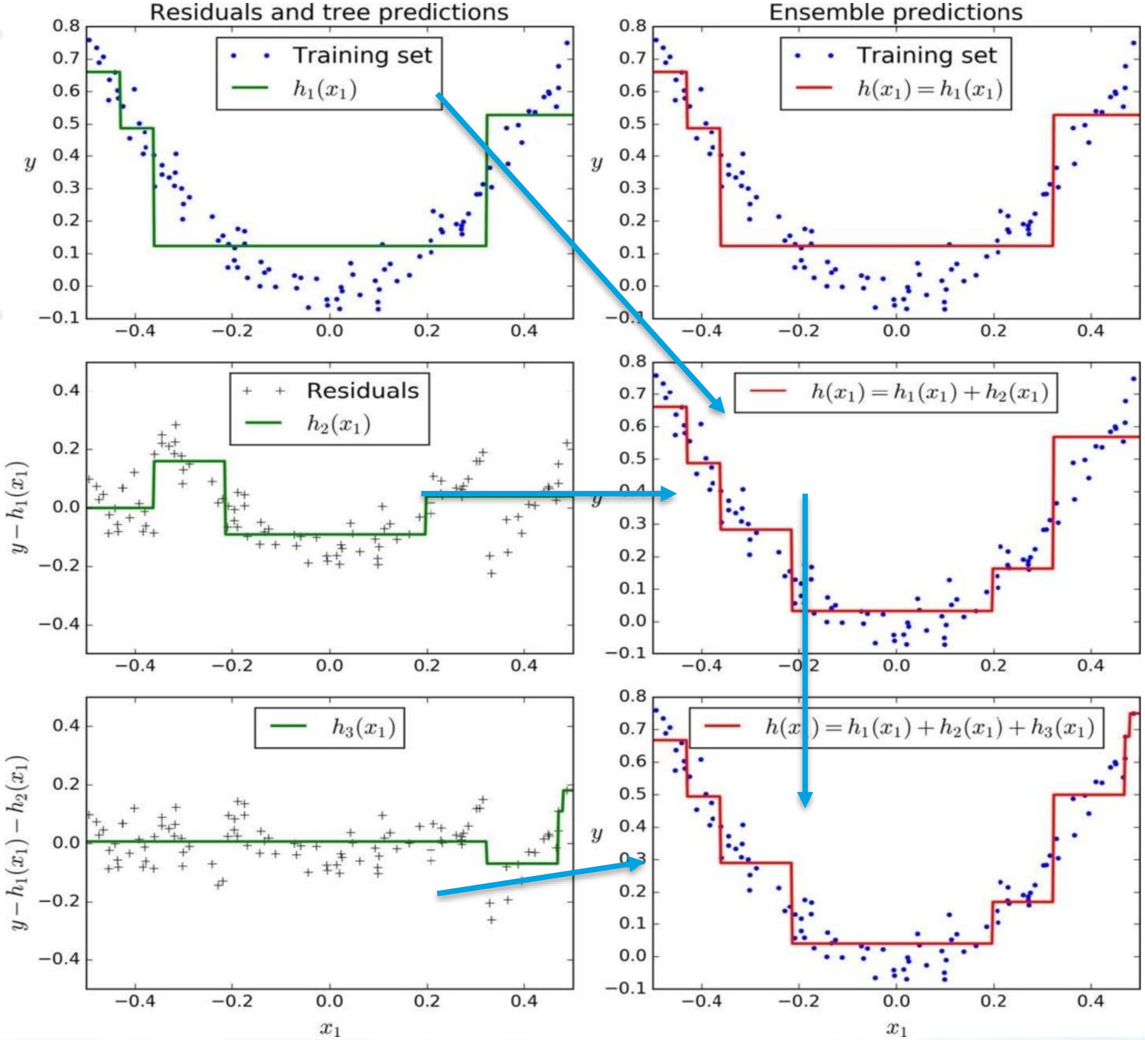
First learner results in residuals (dots that fall above and below the surface. The result (red) is same as first classifier

Next classifier focuses on the residuals of the first classifier to reclassify them as correctly as possible

The combined effect of this surface and previous classifier surface is shown in red

The third learner focusses on the residuals of the previous classifier

The combine result of the new surface with the previous surface is shown in red



Introduction to machine learning

Ensemble Learning – **Gradient Boosting**:

Improve defaulter prediction of the decision tree using Gradient boosting

Description – Sample data is available at local file system as credit.csv

The dataset has 16 attributes described at
https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
or in the notes page of this slide

Sol: GRB+Credit+Decision+Tree.ipynb

Introduction to machine learning

Ensemble Methods – **Random Forest:**

1. Each tree in the ensemble is built from a sample drawn with replacement (bootstrap) from the training set

2. In addition, when splitting a node during the construction of a tree, the split that is chosen is no longer the best split among all the features

3. Instead, the split is picked is the best split among a random subset of the features

4. As a result of this randomness, the bias of the forest usually slightly increases (with respect to the bias of a single non-random tree)

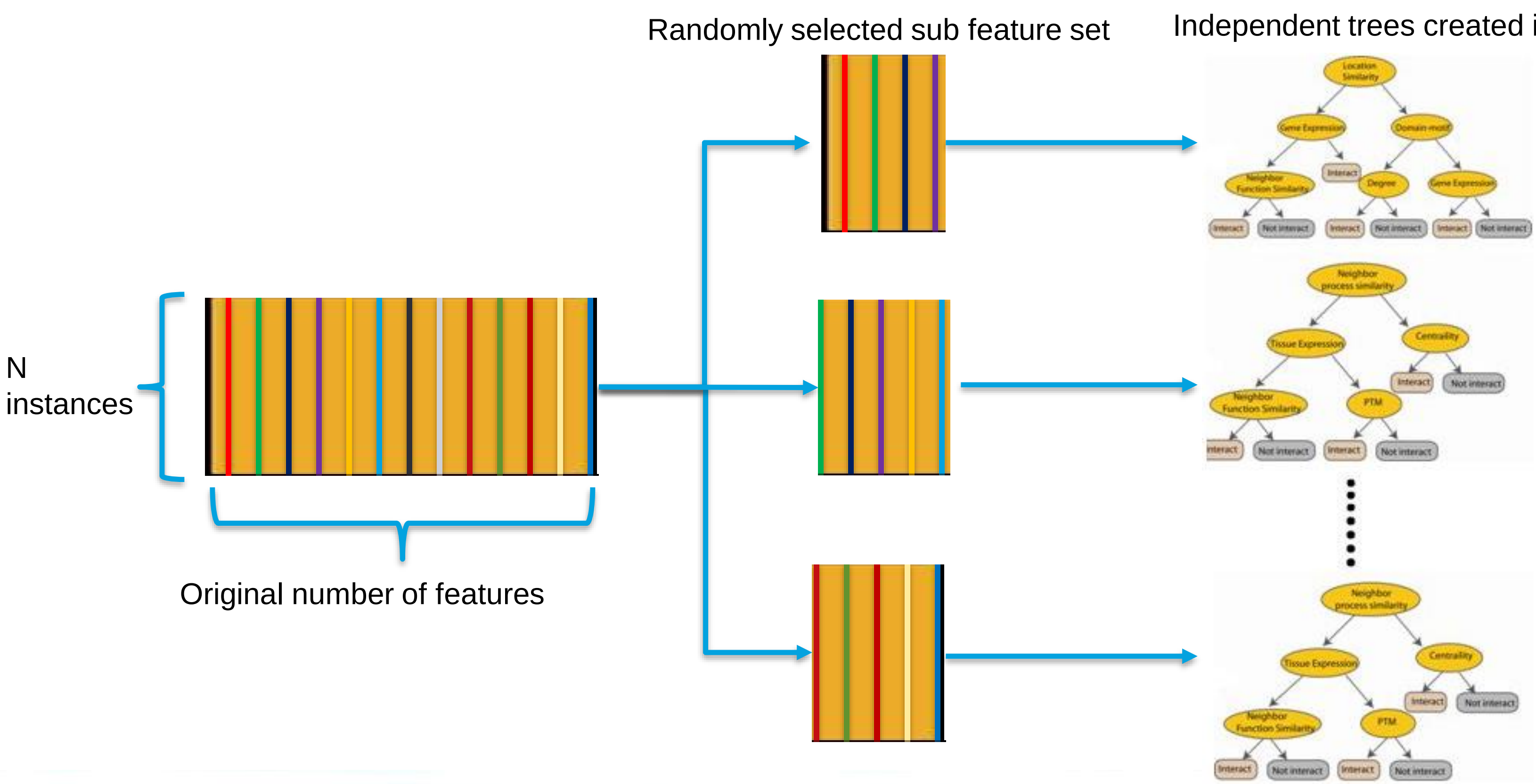
5. Due to averaging, its variance decreases, usually more than compensating the increase in bias, hence yielding overall a better result

Source: scikit-learn user guide , chapter 3 , page 231

Introduction to machine learning

Ensemble Methods - **Random Forest:**

1. Used with Decision Trees. Create different trees by providing different sub-features from the feature set to the tree creating algorithm. The optimization function is Entropy or Gini index



Ensemble Learning – **Random Forest**:

Improve defaulter prediction of the decision tree using Random Forest

Description – Sample data is available at local file system as credit.csv

The dataset has 16 attributes described at
https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
or in the notes page of this slide

Sol: RF+Credit+Decision+Tree.ipynb

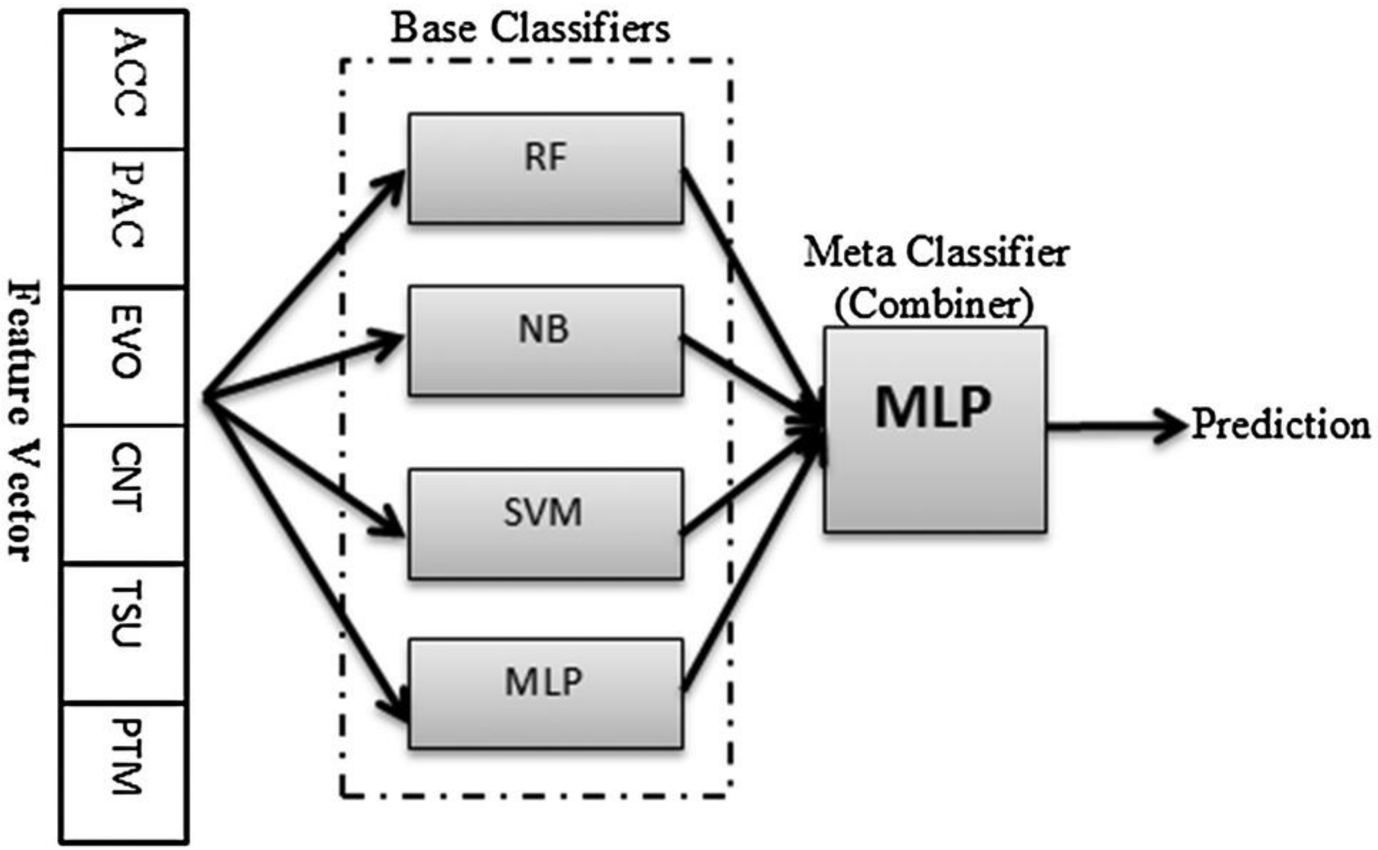
Introduction to machine learning

Ensemble Methods – **Stacking:**

1. Similar to bagging, but apply several different models to original data
2. The weights for each model is determined based on how well they perform on the given input data
3. Similar classifiers usually make similar errors (bagging), so forming an ensemble with similar classifiers may not improve the classification rate
4. Presence of a poorly performing classifier may cause performance deterioration in the overall performance
5. Similarly, even on presence of a classifier that performs much better than all of the other available base classifiers, may cause degradation in the overall performance
6. Another important factor is the amount of correlation among the incorrect classifications made by each classifier
7. If the consistent classifiers tend to misclassify the same instances, then combining their results will have no benefit
8. In contrast, a greater amount of independence among the classifiers can result in errors by individual classifiers being overlooked when the results of the ensemble are combined.

Introduction to machine learning

Ensemble Methods – **Stacking:**



Source: http://pubs.rsc.org/-/content/articlelanding/2014/mb/c4mb00410h/unauth#!divAbstract

Introduction to machine learning

Ensemble Learning – **Stacking**:

Improve defaulter prediction of the decision tree using Stacking

Description – Sample data is available at local file system as credit.csv

The dataset has 16 attributes described at
https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
or in the notes page of this slide

Sol: Stacking+Credit+Decision+Tree.ipynb

ThankYou