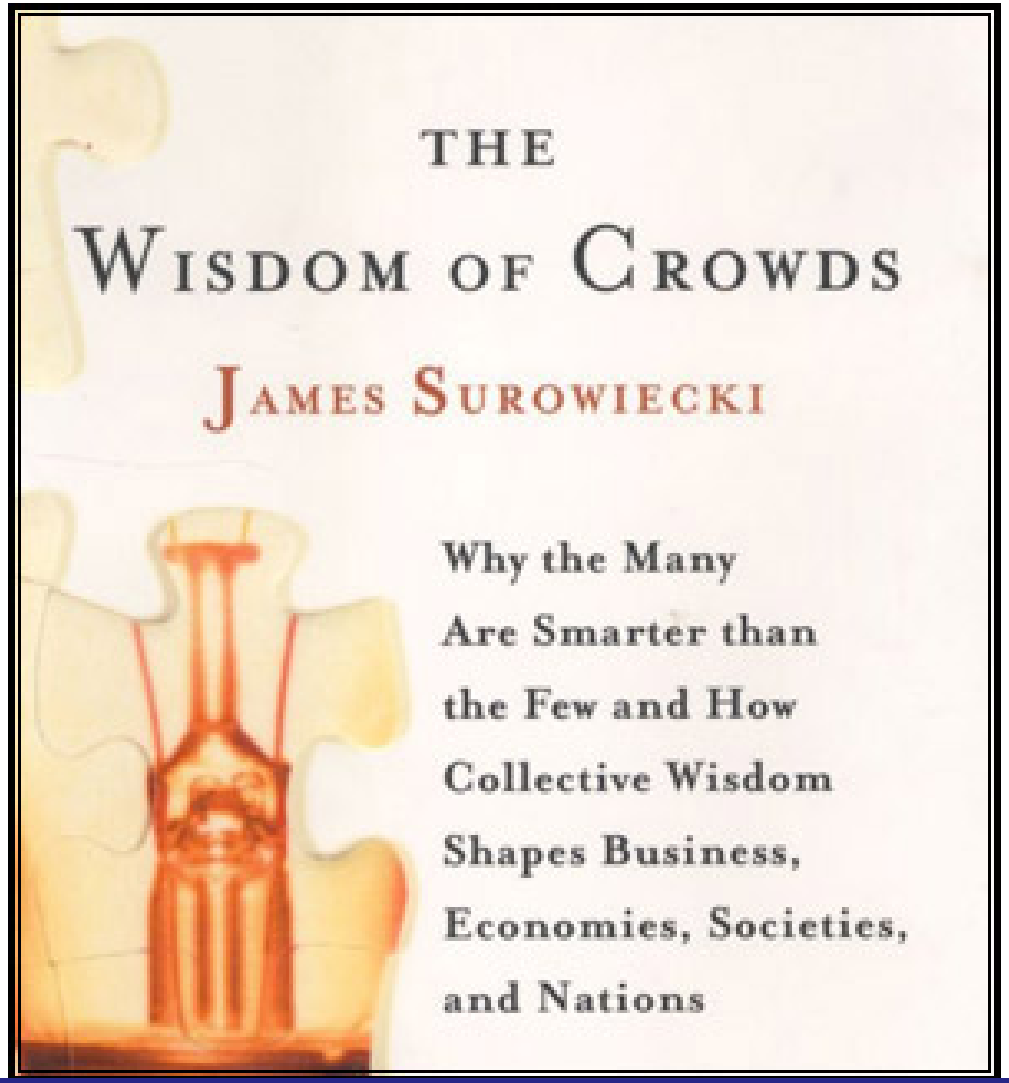
Adventures in Random Forests: Techniques for Engineering Accurate Ensembles

Professor Mohamed Medhat Gaber

School of Computing & Digital Technology
Birmingham City University

Random Forests

The Wisdom of Crowds



Random Forests

- An ensemble classification and regression technique introduced by Leo Breiman
- It generates a diversified ensemble of decision trees adopting two methods:
  - A bootstrap sample is used for the construction of each tree (bagging), resulting in approximately 63.2% unique samples, and the rest are repeated
  - At each node split, only a subset of features are drawn randomly to assess the goodness of each feature/attribute ($\sqrt{F}$ or $\log_2 F$ is used, where F is the total number of features)
- Trees are allowed to grow without pruning (in implementations, they will be pruned at a deep level).
- Typically 100 to 500 trees are used to form the ensemble.
- It is now considered among the best performing classifiers

Random Forest Tops State-of-the-art Classifiers

- 179 classifiers
- 121 datasets (the whole UCI repository at the time of the experiment)
- Random Forest was the first ranked, followed by SVM with Gaussian kernel

Reference

Fernandez-Delgado, M., Cernadas, E., Barro, S., & Amorim, D. (2014). Do we need hundreds of classifiers to solve real world classification problems?. The Journal of Machine Learning Research, 15(1), 3133-3181.

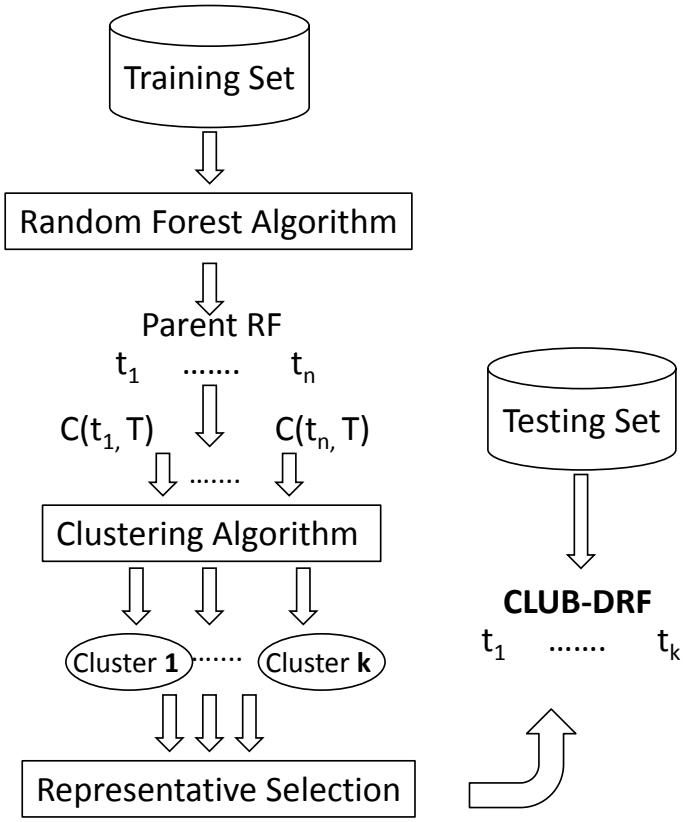
Pruning: From Forests to Small Gardens

How is Diversity Related to Clustering?

- The aim of any clustering algorithm is to produce cohesive clusters that are well separated
- A good clustering model diversifies among members of different clusters
- Inspired by this observation, we hypothesised that *if trees in the Random Forest are clustered, we can use a small subset (typically one tree) from each cluster to produce a diversified Random Forest*
- The benefits are two fold
  - An increased diversification
  - A smaller ensemble, leading to faster classification of unlabelled instances

CLUB-DRF

- We termed the method *CLUster Based Diversified Random Forests (CLUB-DRF)*
- Three stages are followed:
  - A Random Forest is induced using the traditional method
  - Trees are clustered according to their classification pattern
  - One or more representative are chosen from each cluster to form the pruned Random Forest



CLUB-DRF Settings

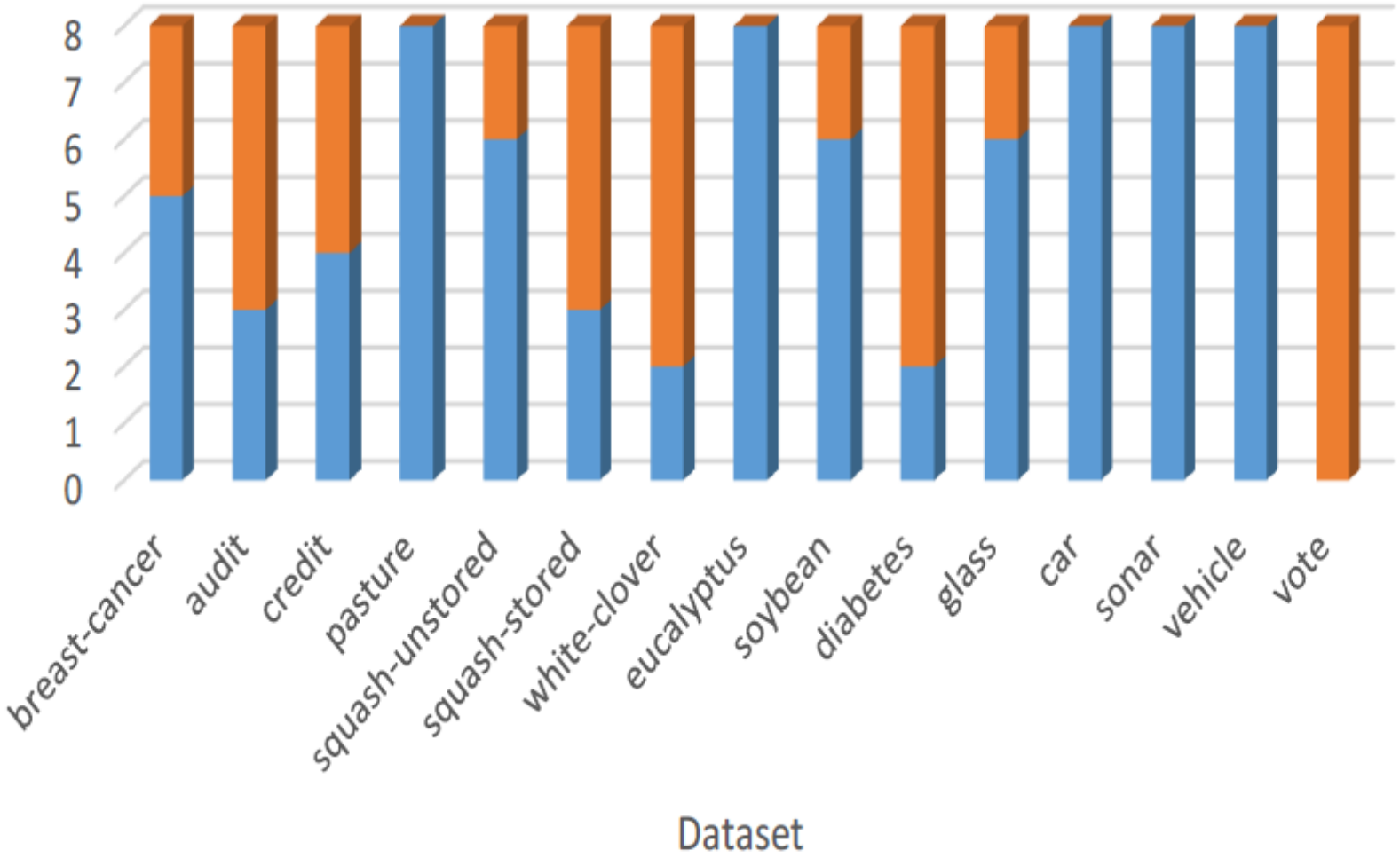
A number of settings are needed as follows:

- The clustering algorithm used
- The number of clusters of trees
- The number of trees representing each cluster
- The criteria for choosing the representatives
  - Random
  - Best performing

Experimental Setup

- We tested the technique over 15 datasets from the UCI repository
- We generated 500 trees for the main Random Forest
- We used k-modes to cluster the trees
- We used the following values for k: 5, 10, 15, 20, 25, 30, 35, and 40
- We used one representative tree per cluster based on the Out Of Bag (OOB) performance
- Repeated hold-out method used to estimate the performance

Summarised Results



Pruning Results

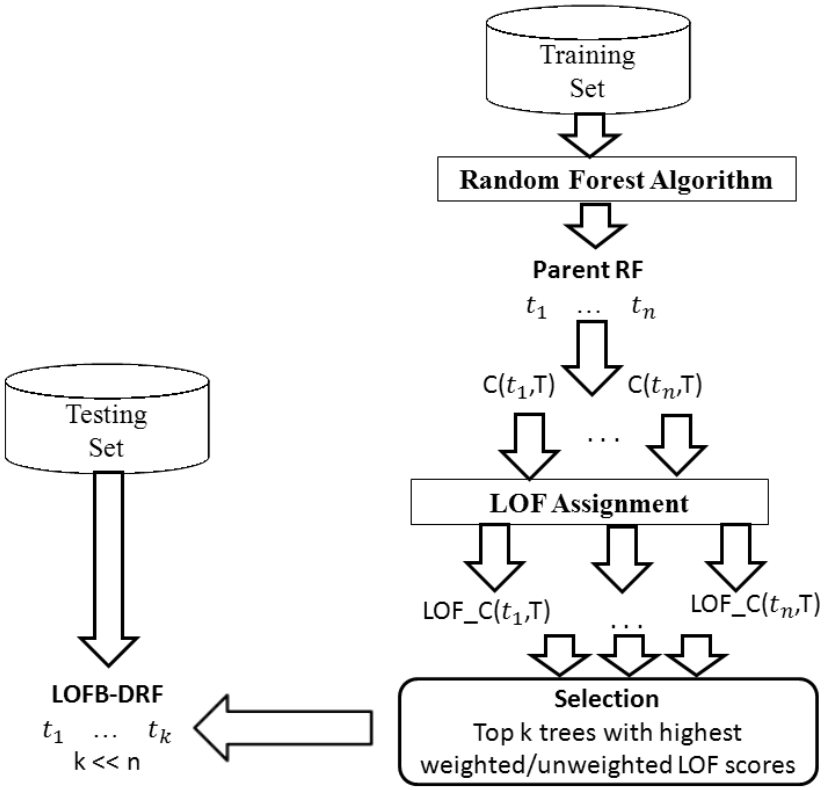
Dataset	Maximum Pruning Level	Best Performance Pruning Level
breast-cancer	99%	96%
credit	99%	99%
pasture	99%	98%
squash-unstored	98%	98%
squash-stored	99%	98%
white clover	94%	94%
eucalyptus	99%	98%
soybean	99%	97%
diabetes	96%	96%
glass	99%	99%
car	99%	99%
sonar	99%	99%
vehicle	99%	98%

How is Diversity Related to Outlier Detection?

- Outliers are out of the norm instances that are thought to be generated by a different mechanism
- By analogy, trees that are significantly different (diverse) from the set of other trees in the Random Forest can be seen as outliers
- Local Outlier Factor (LOF) assigns a real number to each instance to represent its peculiarity
- Inspired by this analogy, we hypothesised that *a diverse ensemble of trees can be formed using outlier detection method*

LOFB-DRF

- We termed the method *Local Outlier Factor Based Diversified Random Forests (LOFB-DRF)*
- It follows similar steps to *CLUB-DRF*
- Each tree is assigned *LOF* value
- Trees are then chosen according to two criteria
  - Predictive accuracy
  - LOF value



LOFB-DRF Settings

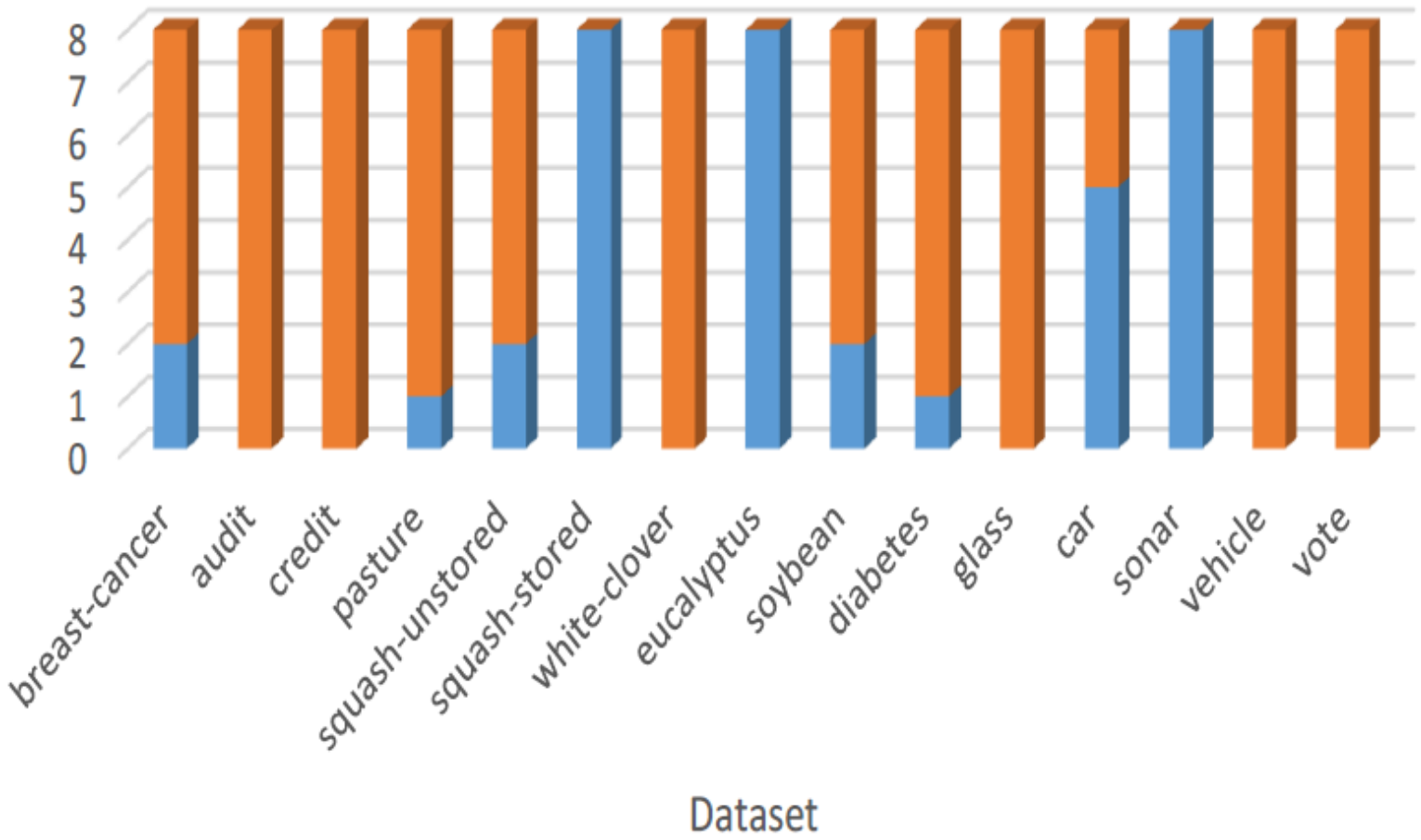
A number of settings are needed as follows:

- LOF setting of the number of nearest neighbours
- Options of combining LOF with predictive accuracy
  - Using LOF only ruling out predictive accuracy
  - Using a combination strategy

Experimental Setup

- We tested the technique over 10 datasets from the UCI repository
- We generated 500 trees for the main Random Forest
- We used LOF with 40 nearest neighbours
- We used $[rank = normal(LOF) \times accuracy]$ for each tree, where $normal(LOF), accuracy \in [0, 1]$
- Trees with the higher *rank* are chosen as representatives
- We used the following values for representative trees: 5, 10, 15, 20, 25, 30, 35, and 40
- Repeated hold-out method used to estimate the performance

Summarised Results



Pruning Results

Dataset	Maximum Pruning Level	Best Performance Pruning Level
breast-cancer	97%	95%
squash-unstored	95%	93%
squash-stored	99%	98%
eucalyptus	99%	99%
soybean	98%	97%
diabetes	96%	96%
car	99%	99%
sonar	99%	99%

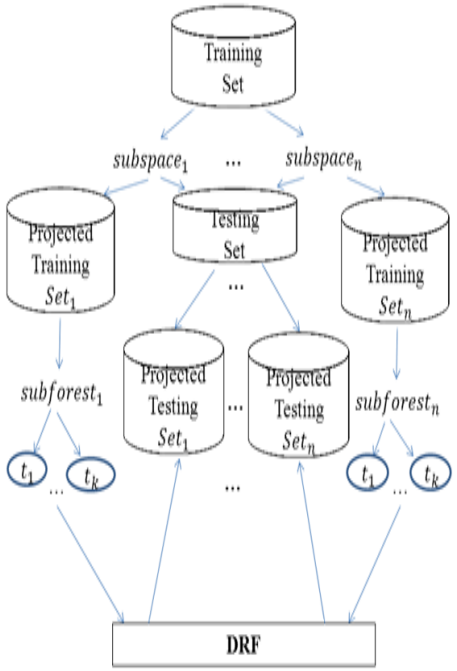
Some Food for Thought

- In CLUB-DRF:
  - Exploring other methods for choosing tree representatives from each cluster (e.g., varying the number of representatives per cluster)
  - Using other clustering techniques
- In LOFB-DRF:
  - Exploring other options for combining LOF value and predictive accuracy
- Using LOF and predictive accuracy for the choice of tree representatives in each cluster
- Applying both methods to other ensemble classification techniques

Feature Interaction: Choosing the Right Seeds

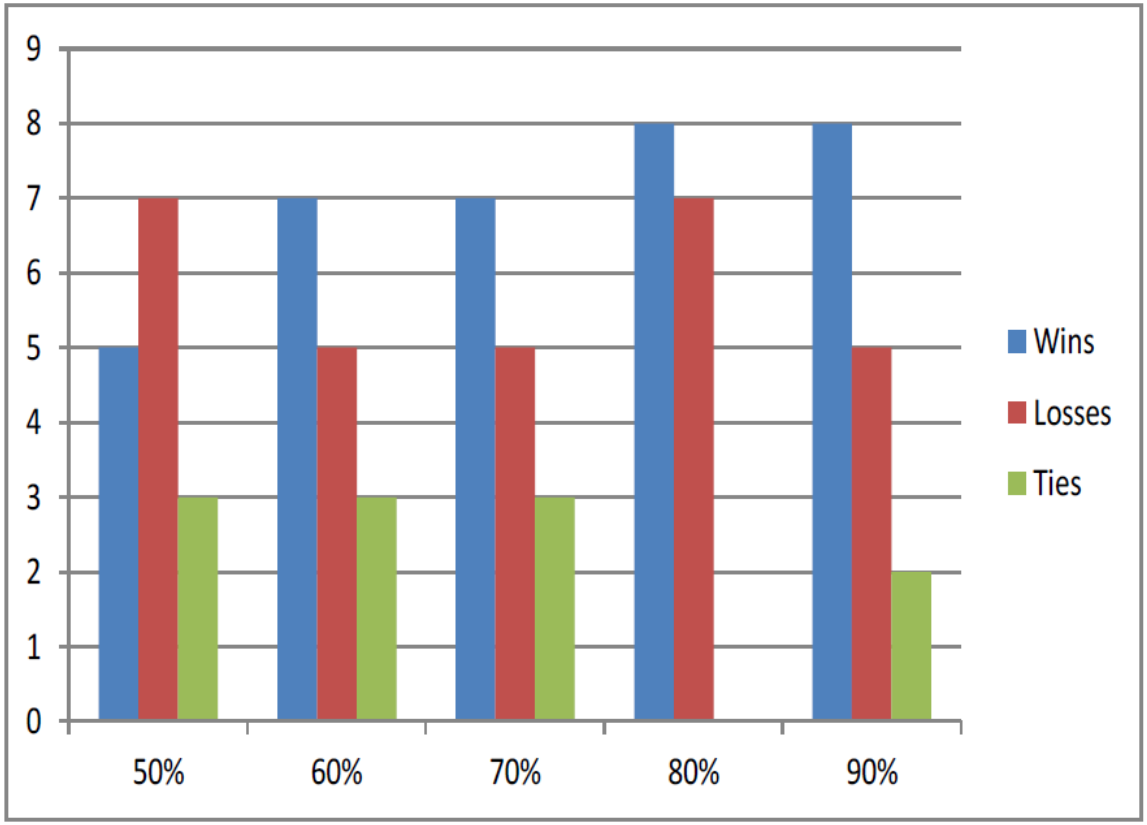
How is Diversity Related to Random Subspacing?

- Random projection of features can help further diversification of Random Forests
- Utilising random subspacing, a diversified random forests (DRF) is developed
- DRF is composed of a fixed number of subforests
- Each subforest is constructed from a subspace, all subforests have the same number of trees
- A weight is assigned to each subspace based on the discrimination power



Summarised Results

- 20 subforests, 25 trees each.



Limitation of DRF

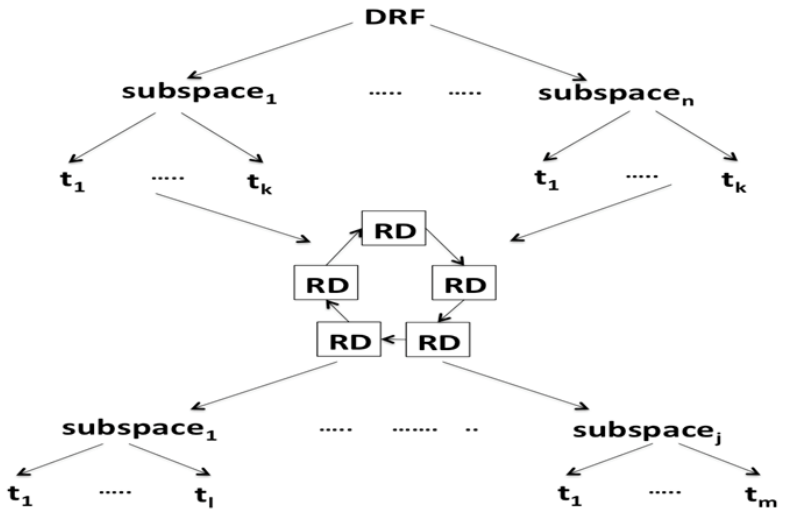
- Weighting is done through explicit measurement of discrimination power of features, ignoring feature interaction
- Two conditions may limit the success of DRF:
  - A large proportion of correlated attributes in the dataset can invalidate the weight
  - An increase in the proportion of noisy features can weaken the subforests' discrimination power.

What is Replicator Dynamics

- A simple model of evolution used extensively in evolutionary game theory and other disciplines
- It provides a convenient way to represent selection among a population of diverse types
- To illustrate how it works, assume that selection occurs between periods after dividing time into discrete periods
- The proportion of each type in the next period is given by the replicator equation as a function of the type's payoffs and its current proportion in the population
- Types that score above the average payoffs increase in proportion, while types that score below the average payoffs decrease in proportion
- The amount of increase or decrease depends on a type's proportion in the current population and on its relative payoffs

Replicator Dynamics for Feature Interaction

- Three stages are followed:
  - Create a number of subspaces through random projection of features
  - A Random Forest model is built using the traditional method for each subspace
  - Assess each subforest accuracy using OOB
  - Iterate using Replicator Dynamics, growing or shrinking subforests for a pre-set number of iterations



Replicator Dynamics Model

The Model

$$\dot{x}_i = x_i[f_i(x) - \phi(x)] \tag{1}$$

such that

$$\phi(x) = \sum_{j=1}^{n} x_j f_j(x) \tag{2}$$

where
x_i is the proportion of type i in the population,
$x = (x_1, \ldots, x_n)$ is the vector of the distribution of types in the population,
$f_i(x)$ is the fitness of type i (which is dependent on the population), and
$\phi(x)$ is the average population fitness (given by the weighted average of the fitness of the n types in the population).

Replicator Dynamics DRF Settings

A number of settings are needed as follows:

- Number of subforests
- Initial number of trees in each subforest
- Number of Replicator Dynamics iterations
- Growing and shrinking mechanism

Growing/Shrinking Mechanism

Constant

$$treesToAdd = \beta \tag{3}$$

$$treesToRemove = \gamma \tag{4}$$

Variable

$$treesToAdd = \lfloor((subforestAccuracy(i) - DRFAccuracy) \times numTrees)\rfloor \tag{5}$$

$$treesToRemove = \lfloor((DRFAccuracy - subforestAccuracy(i)) \times numTrees)\rfloor \tag{6}$$

where *subforestAccuracy(i)* refers to the accuracy of subforest(i) being processed, and *numTrees* refers to the initial number of trees that was used to construct the sub-forest.

Experimental Setup

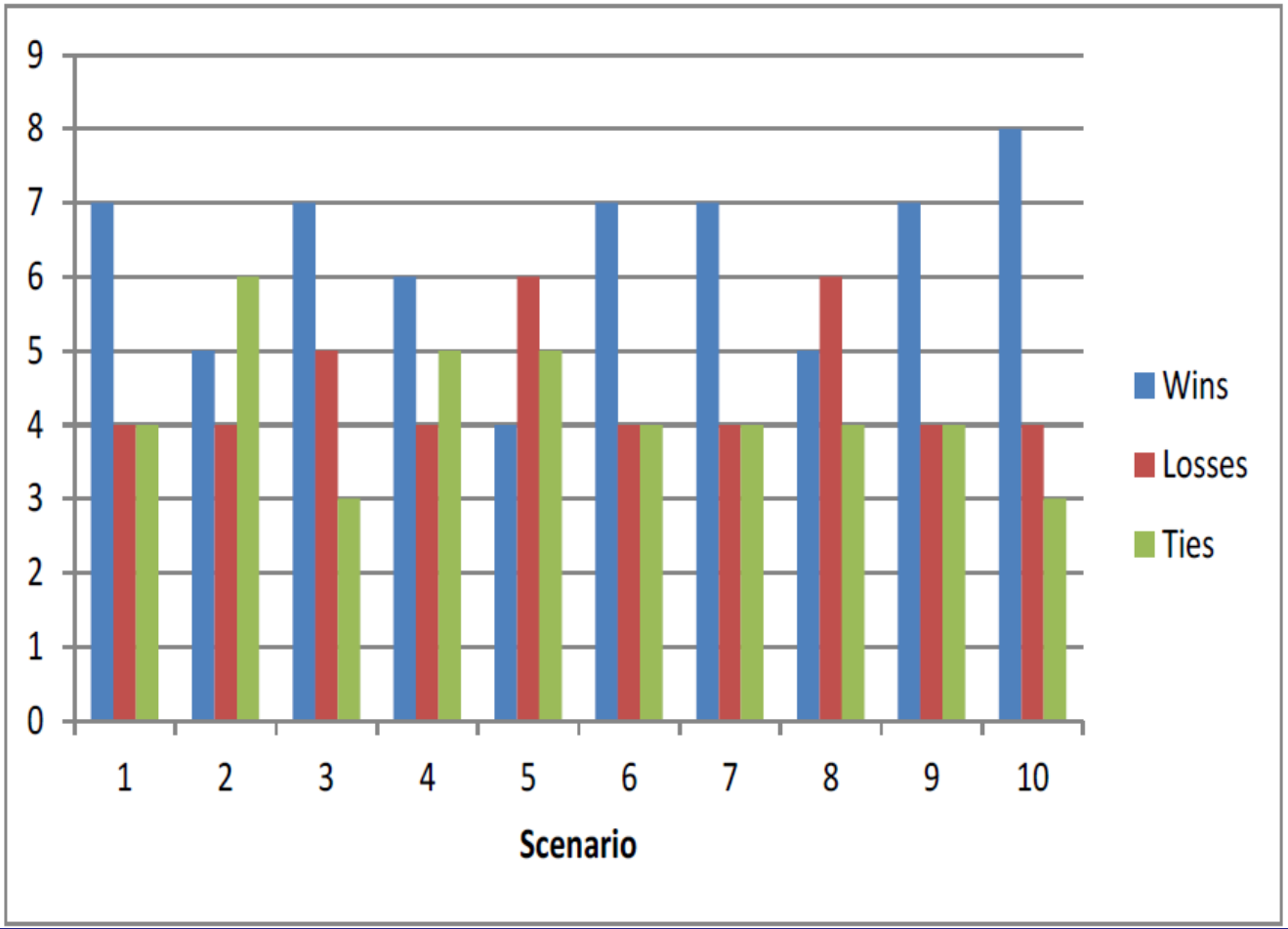
- We tested the technique over 15 datasets from the UCI repository
- We generated 500 trees for all subforests (10, or 20 subforests)
- Each subforest has initially 50 trees (when building 10 subforests), or 25 trees (when building 20 subforests)
- We used the following values for Replicator Dynamics iterations: 25, 50, 100, 150, and 1000 iterations.
- Repeated hold-out method used to estimate the performance

Settings Experimented

Scenario#	Number of Sub-forests	Number of Trees Per Sub-forest	Number of Iterations
1	10	50	25
2	10	50	50
3	10	50	100
4	10	50	150
5	10	50	1000
6	20	25	25
7	20	25	50
8	20	25	100
9	20	25	150
10	20	25	1000

Results

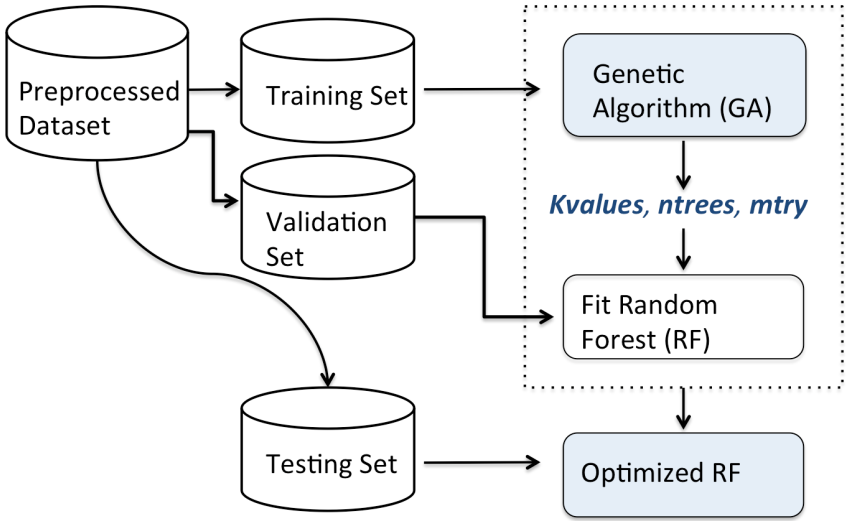
- Better results were reported with constant growth/shrinkage.



Class Decomposition: Making Your Forest More Colourful

Why to Use Class Decomposition?

- Data sets are decomposed using clustering of each class to reveal hidden categories.
- Random Forests technique is built.
- Such class decomposition has two advantages: (1) diversification of the input that enhances the ensemble classification; and (2) improving class separability, easing the follow-up classification process.



Parameter Optimisation

Parameter Tuning

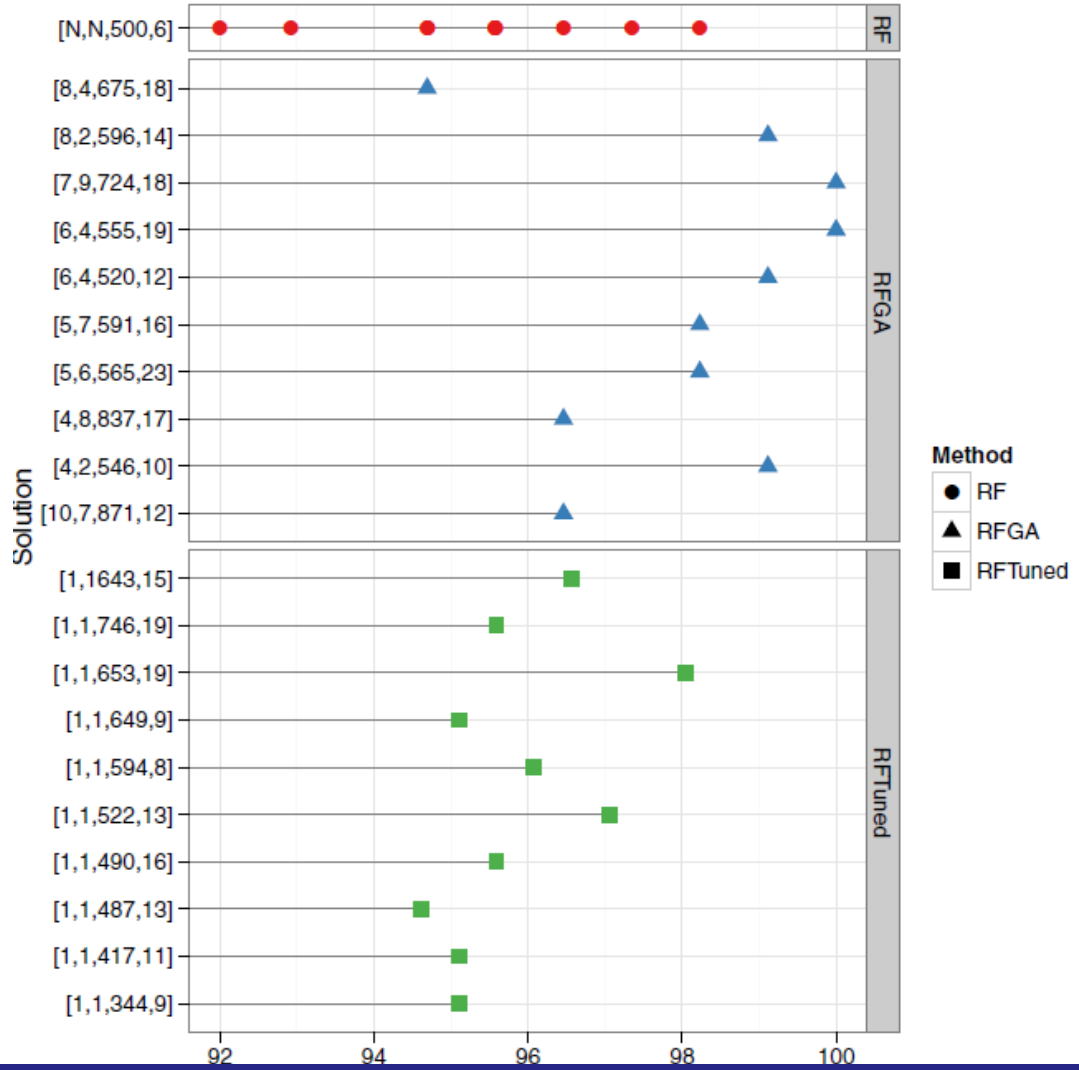
However, to be able to apply Random Forests on such class decomposed data, three main parameters need to be set:

1. number of trees forming the ensemble,
2. number of features to split on at each node, and
3. a vector representing the number of clusters in each class,

Genetic Algorithm

The large search space for tuning these parameters has motivated the use of Genetic Algorithm to optimise the solution.

Sample Results



Summary

Summary

- Random Forest has proved superiority over the last few years
- A number of methods were presented in this talk aiming at improving the performance Random Forests:
  - Ensemble pruning using clustering & outlier ranking.
  - Feature interaction using random subspaces & replicator dynamics.
  - Diversification using class decomposition optimised using genetic algorithm.
- Results showed the potential of these methods to further enhance the predictive accuracy of the method, and some other metrics.
- These methods still provide opportunities for further enhancement and combination.

Publications

1. Elyan E., and Gaber M. M., **A Genetic Algorithm Approach to Optimising Random Forests Applied to Class Engineered Data**, *Information Sciences*, Volume 384, April 2017, pp. 220-234, Elsevier. [h5-index = 81, ranked 5th in top publications of Engineering & Computer Science (general), Impact Factor = 3.364].
2. Elyan E., and Gaber M. M., **A Fine-Grained Random Forests using Class Decomposition: An Application to Medical Diagnosis**, *Neural Computing and Applications*, November 2016, Volume 27, Issue 8, pp. 2279-2288, Springer. [h5-index = 35, Impact Factor = 1.492].
3. Fawagreh K., Gaber M. M., and Elyan E., **Random Forests: From Early Developments to Recent Advancements**, *Systems Science & Control Engineering*, Volume 2, Issue 1, 2014, pp. 602–609, Taylor & Francis. [h5-index = 13]
4. Fawagreh K., Gaber M. M., and Elyan E., **An Outlier Ranking Tree Selection Approach to Extreme Pruning of Random Forests**, *Proceedings of 17th Engineering Applications of Neural Network Conference (EANN2016)*, Aberdeen, UK, 2-5 September 2016, Springer Verlag. [h5-index = 5]
5. Fawagreh K., Gaber M. M., and Elyan E., **CLUB-DRF: A Clustering Approach to Extreme Pruning of Random Forests**, *Proceedings of AI-2015, The Thirty-fifth SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*, Cambridge, UK, 15-17 December 2015, pp. 59–73, Springer Verlag. [h5-index = 8]
6. Fawagreh K., Gaber M. M., and Elyan E., **A Replicator Dynamics Approach to Collective Feature Engineering in Random Forests**, *Proceedings of AI-2015, The Thirty-fifth SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*, Cambridge, UK, 15-17 December 2015, pp. 25–41, Springer Verlag. [h5-index = 8]
7. Fawagreh K., Gaber M. M., and Elyan E., **Diversified Random Forests using Random Subspaces**, *Proceedings of the 15th International Conference on Intelligent Data Engineering and Automated Learning*, 10-12 September 2014, in Salamanca, Spain, Lecture Notes in Computer Science, Volume 8669, pp. 85–92, Springer. [h5-index = 10]