A Hybrid Method for Feature Construction and Selection to Improve Wind-Damage Prediction in the Forestry Sector

Emma Hart
Edinburgh Napier University
Scotland, UK
e.hart@napier.ac.uk

Kevin Sim
Edinburgh Napier University
Scotland, UK
k.sim@napier.ac.uk

Barry Gardiner
ISPA, INRA, Bordeaux Sciences Agro
Villenave dOrno,France
barry.gardiner@inra.fr

Kana Kamimura
Institute of Mountain Science,Shinshu University
Japan
kamimura@shinshu-u.ac.jp

ABSTRACT

Catastrophic damage to forests resulting from major storms has resulted in serious timber and financial losses within the sector across Europe in the recent past. Developing risk assessment methods is thus one of the keys to finding forest management strategies to reduce future damage. Previous approaches to predicting damage to individual trees have used mechanistic models of wind-flow or logistical regression with mixed results. We propose a novel filter-based Genetic Programming method for constructing a large set of new features which are ranked using the Hellinger distance metric which is insensitive to skew in the data. A wrapper-based feature-selection method that uses a random forest classifier is then applied predict damage to individual trees. Using data collected from two forests within South-West France, we demonstrate significantly improved classification results using the new features, and in comparison to previously published results. The feature-selection method retains a small set of relevant variables consisting only of newly constructed features whose components provide insights that can inform forest management policies.

CCS CONCEPTS

•Computing methodologies → Search methodologies; Genetic programming;

KEYWORDS

Feature-construction, Machine-Learning, Forestry

ACM Reference format:
Emma Hart, Kevin Sim, Barry Gardiner, and Kana Kamimura. 2017. A Hybrid Method for Feature Construction and Selection to Improve Wind-Damage Prediction in the Forestry Sector. In

GECCO '17, Berlin, Germany

DOI: http://dx.doi.org/10.1145/3071178.3071217

Proceedings of GECCO '17, Berlin, Germany, July 15-19, 2017, 8 pages.
DOI: http://dx.doi.org/10.1145/3071178.3071217

1 INTRODUCTION

Financial losses in the forestry sector arising from damage to trees caused by strong winds during storms can be colossal: in 2009 a storm in south-western France damaged approximately 37 million m^3 of maritime pine trees [7]. This led to losses of approximately €1.8 billion in the forestry sector, which was almost 60% of total economic losses in France that year [9]. Ten years prior to this, a storm in the same region resulted in approximately 26 million m^3 of timber loss, which was equivalent to the general harvested volume for 3.5 years in maritime pine forests in south- western France [8]. As storms are predicted by some researchers to become more intense although less frequent in the future, it is clearly likely that further catastrophic damage in these maritime pine forests is likely to occur. It is thus crucial that the forestry industry develop methods to both understand the direct causes leading to damage occurrence and to assess and predict the risk of damage in order to develop policies that lead to sustainable forest management.

One of the main forest management techniques currently used to increase income of forest owners is to undertake *thinning* – the selective removal of trees in order to improve the growth rate or health of the remaining trees [10]. However, thinning has a tendency to increase the risk of wind-damage to the remaining trees due to increased aerodynamic roughness above the canopy: this leads to higher levels of turbulence and the creation of small gaps which both increase wind penetration between trees and can act as trigger points for damage propagation during a storm. Therefore, selecting the most at risk trees for early removal is a one of the key ways to reduce wind damage risk. Mechanistic modelling has been applied using hybrid mechanistic/empirical wind-modelling software to predict damage at individual tree level with some limited success. In contrast, a generalized linear mixed model to predict damage is proposed by [2] and used with data collected from German forests, while Kamimura *et al* [13] use logistic regression to predict damage to individual trees within Aquitaine. The latter found that models could

discriminate between damaged/undamaged trees to some extent, but that there was considerable room for improvement in order to better inform forest management policy decisions.

Finding a highly accurate model is challenging due to the nature of the data: datasets are relatively small for pragmatic reasons relating to recording the data ($\approx$ 1500 trees); the number of features measured from the trees is small (typically around 8); the proportion of damaged trees in each forest can be low ($\approx$ 15%). To address these issues, we first use Genetic Programming (GP) to construct new features to augment the original dataset. The GP method introduces Hellinger distance [6] as a fitness measure to evaluate feature quality, which has recently been shown to be insensitive to imbalanced data. GP is run multiple times to construct a large set of new features. Finally, a feature-selection approach from machine-learning is then applied in conjunction with a Random Forest Classifier [5] to predict tree damage. The approach is applied to real-data obtained from two forests in South-West France — Nezer and Aquitaine. We specifically address the following research questions:

(1) Can GP produce new features that have higher fitness according to the Hellinger distance metric than the original features?
(2) Does a classifier applied to the augmented set of features constructed by GP outperform results obtained using only the original features?
(3) Does applying a variable-selection method result in a smaller number of variables with low redundancy and accurate prediction of the response variable ?
(4) Are the selected features informative for users within the Forestry sector?

We find a significant improvement in classification accuracy of approximately 16% in the Nezer forest, and 3% in the Aquitaine forest compared to learning a model from the original data. Similar improvements in the area-under-the-curve (AUC) metric are also observed. The paper is novel in terms of the application of EC techniques to inform forest management. From an EC perspective, it is novel in that it exploits GP to generate a large set of *fit* but diverse features, scored according to Hellinger distance, and combines this with a powerful feature-selection method from machine-learning to efficiently find a small number of variables with high predictive accuracy.

2 BACKGROUND

The quality of the feature set is a key factor influencing the performance of a classification algorithm [21]. Irrelevant and/or redundant features can have an adverse affect on the accuracy of a classifier, as well as increasing the time required to learn a model and the complexity of the trained classifier. Similarly, measured features collected from a real-world problem can fail to discriminate adequately between classes. To address these weaknesses, Evolutionary Computing (EC) has been used widely for both feature-selection to choose relevant features, and feature selection, to evolve new features which provide better discrimination and can be used to augment the original feature space. With respect to feature construction, methods fall into two-categories: *filter* methods are applied to data, independently of a classification technique, while *wrapper methods* use classification performance to rank new features. Here we focus on filter-based feature-construction and therefore provide a brief review of relevant literature. To avoid confusion between trees evolved by GP and forest trees, we use the term *program* to refer to a GP- tree.

The most common method for feature construction is genetic programming (GP), largely due to its ability to automatically evolve mathematical models that can be linear or non-linear and can make use of a large range of pre-defined function and terminal nodes. Approaches can be categorised along two dimensions: the method used to score new features for relevance, and the number of new features simultaneously evolved. Common methods for assessing variable relevance include *Information gain (IG)*, *Gini index (GI)* or *Chi-squared* [21]. The efficacy of these methods used in conjunction with GP to evolve a single feature is evaluated in [17], indicating little bias between metrics. Guo and Nandi [12] also evolve a single additional feature, using *scattering* as a metric, and report improved results. However, the metrics just discussed are unlikely to work well with imbalanced datasets: to counter this [19] propose a balanced-accuracy metric which they use in conjunction with a GP algorithm that evolves a program which acts as a *classifier*, therefore implicitly embedding feature-selection and construction within the model.

Multi-program methods use GP to simultaneously evolve multiple new features, e.g. [15]. However, a key difficulty of this method is the need to pre-define the number of required programs. In [20], the representation is fixed to evolve as many programs as there are original features. In contrast, Neshatian *et al* [18] evolve a single program but repeat the procedure as many times as there are classes. Cooperative co-evolution [16] has also been used to evolve multiple programs but again suffers from the need to pre-define the number required. An alternative method proposed in [1] constructs multiple features by segmenting the single best evolved program into multiple sub-trees; while this does not require pre-fixing the program number, the number of resulting trees is limited by the maximum depth.

In this paper we address the issue of pre-defining the number of new features by running a single-program GP algorithm many times. This provides a computationally cheap method of generating new features but may result in some redundancy. However, when this is combined with a powerful and efficient feature selection method from machine-learning (VSURF [11]), a minimal set of features with high predictive accuracy is obtained.

3 DATA

The first data set *Nezer* was obtained from a field survey of 29 permanent plots (400 m^2) in the Nezer Forest located in a region of south-west France. The tree sizes within this

Table 1: Summary data from the two datasets Nezer and Aquitaine

	Trees	Damaged Trees	Percentage Damage
Nezer	1029	130	0.13
Aquitaine	1691	563	0.33

data were obtained from a field survey in 1998, and damaged trees were determined after the storm in 1999. Damage occurred to 13% of trees in the dataset. The second data set *Aquitaine* was obtained from field surveys of the national forest inventory in France (Inventaire Forestier National; NFI) in the Landes de Gascogne region. The survey plots are located on a $1km \times 1km$ grid. We use data collected from 2007 to 2008 from a total of 235 plots. After storm Klaus in 2009, damaged trees in the NFI plots were identified by an additional field survey. 33% of surveyed trees were damaged. Basic statistics of both data sets are presented in table 1.

In addition to data collected from the surveys described above, spatial information was also included for each plot in the two data sets. The distance from the windward stand edge (the westerly direction for both storms) was defined as the boundary line between forests and unforested area including roads (> 3m width). Although the distance was very precise in the Nezer forest, the distance had to be estimated using the coarse plot location in the NFI data in which the exact plot positions were not publically available. The stem spacing in both data sets was the mean value calculated from the number of stems in the plot. Gap size (the distance in a westerly direction between the forest in which the plot was located and the next forest block) was also calculated. Finally a *competition index (CI)* [4] is calculated for all trees. This describes the relationship between a subject tree and neighbouring trees in order to estimate the allocation of growth resources such as water and light that are generally limited by the size and number of neighbours. This results in 8 input features used to describe each tree (see table 2), with a single output (damaged/not-damaged) indicating if the tree was damaged in the storms described above.

4 METHODOLOGY

An overview of the method is given in figure 1. First, GP is run n times, with each run evolving a single new feature. The n newly constructed features are then added to the original dataset. A feature-selection procedure VSURF [11] is then applied to the augmented dataset in conjunction with a random-forest classifier to derive a minimal set of features for prediction. These features are then used to obtain the final model using a random-forest classifier. Each step is explained below.

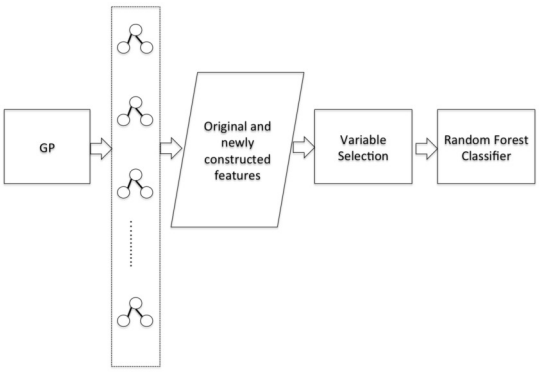

Figure 1: Overview of method: the output from multiple GP runs is used to construct a large new set of features. A variable selection approach is used extract a subset of relevant features which are used by a random forest classifier

4.1 GP

The GP algorithm is a conventional generational algorithm [14] that is initialised using ramped half and half and employs sub tree crossover and mutation to create a new population of 600 individuals each generation. An elitist strategy is employed where the best member of the population is retained. Crossover and mutation are applied with a probability of 80% and 10% respectively, otherwise a random parent is returned. The maximum initial program depth is set to 6 and the maximum bloat depth varies between 6 and 17 as detailed for each experiment.

Terminal nodes consist of the 8 original features from table 2 plus three constants (*minusOne, IntegerConstant, DoubleConstant).* The function node set contains standard arithmetic operators (<,+,-,*, *protectedDivide*,>, <) plus additional functions (*log, sin, cos, tan, abs*).

As described in section 2, the fitness functions typically used in GP-based feature construction (e.g. information-gain) perform poorly on imbalanced data. To mitigate this, we make use of an alternative fitness metric called *Hellinger distance* [6]. This has been shown analytically and empirically to demonstrate strong skew insensitivity when used as a splitting criterion in decision trees, and thus provides significant advantages over more common metrics such as information gain when dealing with imbalanced data.

Hellinger distance is defined as follows. Assume a two class classification problem in which the information available can be expressed as $\mathrm{P}(Y_y|X_x)$, where y is drawn from two classes $(+,-)$ and x is drawn from a a finite set of attribute values V, or in the case of continuous features, by discretising the variable by examining a variety of splits and selecting the most appropriate. Then, the Hellinger Distance d_H is defined as shown in equation 1.

Table 2: Summary statistics for Aquitaine and Nezer datasets

Attribute	Aquitaine		Nezer	
	Mean	Standard Deviation	Mean	Standard Deviation
Gap Size (m)	177.32	66.44	125.21	66.31
Diameter Breast Height (DBH) (cm)	29.88	14.42	19.1	11.31
Tree Height	17.74	6.89	12.19	6.57
Mean DBH (cm)	29.86	12.91	19.11	10.63
Stand Mean Height	17.73	6.64	12.52	6.63
Density (ha).	580.76	398.75	977.084	646.64
Competition Index (CI)	14.03	18.13	11.91	9.74
Average Competition Index	14.04	9.68	11.86	6.49

$$d_H(P(Y_+), P(Y_-)) = \sqrt{\sum_i (\sqrt{P(Y_+|X_i)} - \sqrt{P(Y_i|X_i)})^2} \quad (1)$$

The fitness value thus varies between a minimum of 0 and maximum of $\sqrt{2}$.

4.2 Classifier

A Random Forest classifier [5] is selected to build the predictive model. Previous work published in [13] discussed the limitations of a logistic regression model, and preliminary investigation by the authors with the original datasets comparing single decision trees, a neural network trained via back propagation and a random forest suggested that the random forest approach was most promising.

4.3 Variable Selection

Variable selection can be used to remove irrelevant variables, to select all important ones or to determine a sufficient subset for prediction [11]. We utilise an R package which performs efficient variable-selection for random forest algorithms (VSURF, [11]). A two-stage strategy is employed which is based on a preliminary ranking of the explanatory variables using the random forests permutation-based score of importance, followed by a stepwise forward strategy for variable introduction to find the best predictive set. The final predicted set contains a small number of variables with very low redundancy but sufficient for a good enough prediction of the response variable [11].

In a random forests framework, one of the most widely used scores of importance of a given variable is the increase in misclassification rate in the forest when the observed values of this variable are randomly permuted in the out-of-bag (OOB samples) [3]. Assume $errOOB_t$ represents the misclassification rate of a single tree t on an OOB_t sample. The values of a chosen feature X^j are then randomly permuted to get a perturbed sample denoted by $\widetilde{OOB}^j$, and the misclassification $\widetilde{errOOB_t}^j$ of predictor t on the perturbed sample calculated. The *variable importance VI* of variable X^j is then given by equation 2:

$$VI(X^j) = \frac{1}{ntree} \sum_t (\widetilde{errOOB_t}^j - errOOB_t) \quad (2)$$

In order to calculate the most relevant variables for prediction with a random forest classifier, the method proceeds as follows. First, 50 random forest models are learned, and the mean VI of each variables is calculated and the variables ranked in order of importance. Then:

(1) Eliminate all variables with mean VI less than a pre-defined threshold, resulting in n variables
(2) Construct n new models containing $k = 1$ to $k = n$ variables, and select the model $m*$ with lowest $errOOB$.
(3) Construct a new set of models adding the variables from $m*$ in order of importance, retaining a variable only if the resulting error decrease is larger than a threshold. Retain the final model.

The procedure is applied to the large set of newly constructed features returned from running the genetic programming algorithm, and the features selected from the final model returned.

5 EXPERIMENTS

The following procedure is used to first construct new features, and then identify the optimal subset of variables for creating a model with high accuracy and AUC.

(1) Run GP k times using the entire original dataset for a forest O, to create k new features
(2) Create 10 new datasets each containing (8+k) features (i.e. original+constructed features) by undersampling O such that each new dataset O_{U_i} contains an equal number of instances of both positive and negative classes
(3) For each O_{U_i}, use VSURF to identify the optimal subset of variables for prediction
(4) For each O_{U_i}, apply a random-forest algorithm using *cross-validation* using a feature set f containing:
    (a) $VSURF_{ind}$: the variables selected by VSURF for O_{U_i}
    (b) $VSURF_{maj}$: the variables that were selected in at least 50% of the 10 individual VSURF runs

Table 3: Analysis of Hellinger values obtained for the original 8 features and newly constructed features (in bold)

		Original Features (from table 2)								Constructed Features		
		f1	f2	f3	f3	f5	f6	f7	f8	Mean(sd)	Max	Min
Nezer	D17	0.613	0.580	0.592	0.613	0.613	0.613	0.175	0.572	**0.709 (0.038)**	**0.802**	**0.648**
	D6									0.672 (0.023)	0.635	0.733
Aquitaine	D17	0.254	0.260	0.249	0.266	0.286	0.050	0.217	0.282	**0.387 (0.048)**	**0.463**	**0.298**
	D6									0.350 (0.031)	0.299	0.443

(c) $VSURF_{All}$: the concatenated set of all variables identified by VSURF from the 10 individual VSURF runs

Note that in step (1) the entire imbalanced dataset is provided to the GP in order to construct new features using the maximum amount of data. As previously explained, the Hellinger metric is insensitive to the skew in the class distribution. In the remaining feature selection steps however, we use an under-sampling method to create balanced datasets. The Random Forest implementation provided in R (and used by VSURF) uses the Gini index to measure node impurity when selecting nodes to split during decision-tree construction and hence does not cope well with imbalanced data. As the purpose of this research is to provide a predictive model that can be used to inform forest management, it is seen as important to use standard packages which results can be easily reproduced and verified.

5.1 Experimental outline

GP was run 50 times for each set of forestry data using the parameters given in section 4.1. 50 features were evolved for bloat depth 6, and a further 50 for depth 17.

Variable selection was performed using the VSURF package [11] with default parameters. Cross-validation was performed using the R CARET package using a Random Forest classifier with 100 decision trees, and the AUC metric to optimise tuning. All other parameters default to those provided by the packages.

In all statistical comparisons, a Shapiro-Wilk test is first applied to determine whether the null hypothesis that the results are normally distributed is first applied. If the null hypothesis is rejected, a Wilcoxon rank sum test is applied to test for significance, otherwise, a Student t-test. The confidence level is considered *significant* and denoted by $\uparrow$ for p-values less than 0.05 and *extremely significant* and denoted by $\uparrow\uparrow$ if the p-value is less than 0.01.

6 RESULTS

First we examine the first research question: *can GP be used to evolve features with greater Hellinger distance than the original feature set?*. Results are show in table 3 which lists the distance metric calculated for each of the orignal features, and the (mean, maximum, minimum) of the distance metric calculated from the 50 new evolved features. Results are given for both forests from GP with depth 17 and 6. It is clear that in both datasets, the evolved features have higher fitness than all of the original features. A Wilcoxon rank sum test is used to compare the distributions at D17/D6. This shows a statistically significant difference between the D17/D6 results at the 1% confidence level.

Note that the Hellinger distance calculated for the original features in the Aquitaine data is much lower than in the Nezer forest data. In the Nezer Forest, the soil type is identical across the forest and the management approach and seedling source are likely to be consistent across the areas. In contrast, in Aquitaine there are variations in soil type, soil moisture, rooting depth, and management practices across the region. Hence, taking the data as a whole. it is more challenging for correlate any individual feature with the correct outcome.

6.1 Classification using augmented feature set

Next, we examine whether a random-forest classifier applied to the augmented dataset containing either *all* features or a subset of features selected via the VSURF procedure outperforms a random-forest classifier that *only has access to the original features*. Tables 4 to 7 provide the mean and standard deviation of the 10 runs for each variable-selection method for both program depths (as Shapiro-Wilk tests confirm that the null hypothesis that the data is normally distributed cannot be rejected). Results in bold show a significant improvement when compared to the *original* feature set. Tables 4 and 6 also provide results published in [13] — these were obtained using logistic regression using only a subset of the data in each forest, and hence are not directly comparable but provide a useful reference point.

For the Nezer forest, a significant improvement of 16% in the best case is obtained in terms of classification accuracy compared to the original features and 11% increase in the AUC to 94%. Although a statistically significant improvement is also obtained using the feature set constructed from small programs (D=6) the improvement is less marked however, with a maximum increase of 5% in accuracy and AUC.

For the Aquitaine forest — whose original features scored significantly lower according to the Hellinger metric — there is a statistically significant improvement in both classification accuracy and AUC. However, the improvement is at

Table 4: Nezer (Depth 17): Accuracy and AUC (means). Statistical comparison to results from original features. Note that result reported from [13] only used a subset of the Nezer data

	Accuracy	Std.Dev	AUC	Std.Dev
original	0.74	0.03	0.83	0.03
All	**0.87**↑↑	0.02	**0.93**↑↑	0.02
VSURF_all	**0.89**↑↑	0.02	**0.94**↑↑	0.02
VSURF_ind	**0.90**↑↑	0.02	**0.94**↑↑	0.02
VSURF_maj	**0.87** ↑↑	0.02	**0.92**↑↑	0.02
[13]	0.724		0.765	

Table 5: Nezer D6: Accuracy and AUC (means) with features constructed from programs of Depth 17. Statistical comparison to results from original features

	Accuracy	Std.Dev	AUC	Std.Dev
original	0.74	0.03	0.83	0.03
All	**0.77**↑↑	0.02	**0.86**↑↑	0.02
VSURF_all	**0.79**↑↑	0.02	**0.88**↑↑	0.02
VSURF_ind	**0.78**↑↑	0.03	**0.88**↑↑	0.03
VSURF_maj	**0.78**↑↑	0.03	**0.86**↑↑	0.03

Table 6: Aquitaine D17 : Accuracy and AUC (means) with features constructed from programs of Depth 6. Statistical comparison to results from original features. Note that result reported from [13] only used a subset of the Aquitaine data

	Accuracy	Std.Dev	AUC	Std.Dev
original	0.76	0.01	0.83	0.01
All	**0.78**↑↑	0.01	**0.86** ↑↑	0.01
VSURF_all	**0.78** ↑↑	0.01	**0.85** ↑↑	0.01
VSURF_ind	**0.79**↑↑	0.01	**0.85**↑↑	0.01
VSURF_maj	**0.79**↑↑	0.01	**0.85**↑↑	0.01
[13]	0.63		0.709	

maximum 3% in accuracy and 3% in AUC. Furthermore, reducing the depth of the programs from which the new features are constructed from 17 to 6 appears to make no difference in terms of the increase in classification accuracy. A Student t-test confirms there is no statistical difference when (D17-D6) results are compared for each variable selection method.

6.2 Effect of Variable Selection

Next, we address the third question which examines the effectiveness of variable selection. Recall from section 2 that VSURF first calculates the importance of each variable (VI) according to equation 2. Figure 2 plots the mean VI for

Table 7: Aquitaine D6: Accuracy and AUC (means) with features constructed from programs of Depth 6. Statistical comparison to results from original features

	Accuracy	Std.Dev	AUC	Std.Dev
original	0.76	0.01	0.83	0.01
All	**0.78** ↑↑	0.01	**0.85** ↑↑	0.01
VSURF_all	**0.78**↑	0.01	**0.85** ↑↑	0.01
VSURF_ind	**0.79**↑↑	0.02	**0.85** ↑↑	0.01
VSURF_maj	**0.79**↑↑	0.01	**0.85** ↑↑	0.01

Table 8: Number of variables selected by VSURF for each of the 10 undersampled datasets (GP Depth 17)

	Mean	Min	Max	Unique	Majority
Nezer	7.6	6	10	17	7
Aquitaine	2.4	1	3	5	2

a single under-sampled subset of the Nezer data. Only one of 58 variables is eliminated in this step. Note also that the multiple GP runs — which all maximise the same objective — result in a feature set that is *diverse* in terms of variable importance. Although the 50 evolved features have similar fitness according to the Hellinger metric (see table 3 which reports small standard deviation of fitness values for each dataset) they have diverse VI with respect to the random forest. Figure 3 indicates the out-of-bag error for 57 RF models, containing $k = 1$ to $k = 57$ variables. The model with lowest OOB contains 12 variables therefore this is selected for further optimisation. It is clear that adding additional variables results in overfitting. In the final stage (figure 4), models are built by adding one variable at a time, but only retaining the variable if the resulting error decrease is greater than a threshold. This results in 8 variables from the total of 58 under consideration being selected for the final predictive model.

Table 8 reports statistics relating to the number of variables selected by VSURF for each of the 10 under-sampled datasets. For Nezer, the mean number selected is 7.6. Across all 10 datasets, the total number of unique variables is 17 (corresponding to VSURF-all) with 7 variables appearing in at least 50% of datasets. In all tests, *none* of the selected variables correspond to the original 8 features, i.e. selected features are always those constructed by the GP algorithm. For Aquitaine, the number of features selected is considerably smaller: the mean is 2.4 with a maximum of 3. Of all variables selected, there a 5 unique variables, with 2 being selected in at least half of the datasets. As with Nezer, none of the original features are selected.

A statistical comparison of the three variable selection methods against each other shows that the null hypothesis (i.e. selection method A is equivalent to selection method

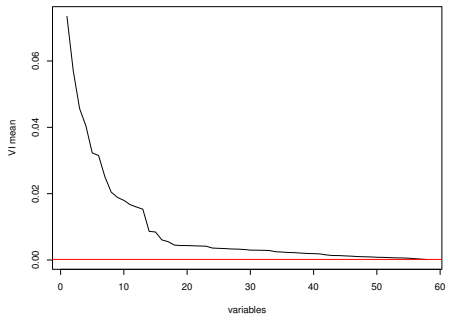

Figure 2: Mean variable importance of each of the 58 variables w.r.t 50 runs of a random forest classifier

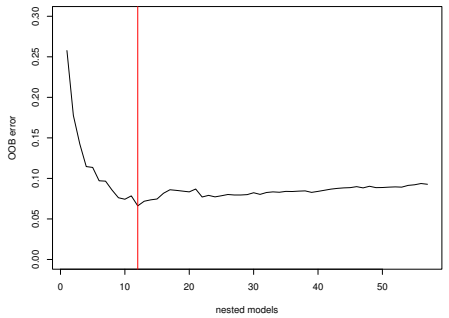

Figure 3: OOB error on classifiers contains k=1 to k=58 variables (variables added in order of VI)

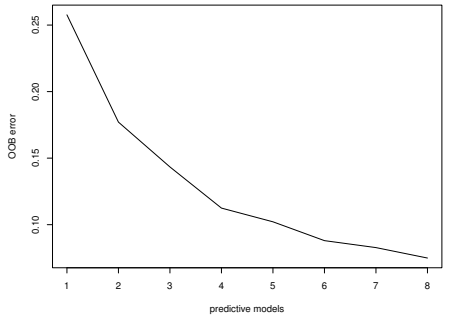

Figure 4: OOB error on classifiers built using 1-12 variables, added in order of VI and only retained if error is reduced by at least a threshold t

B) cannot be rejected except in the case of $VSURF_{ind}$ vs $VSURF_{maj}$ where the former method dominates in the case of the AUC metric. No significant differences are observed in the Aquitaine data. It is clear that an informed variable selection from the large number of features is beneficial; however, the three selection methods investigated appear to provide equivalent benefit. Reducing the number of predictive

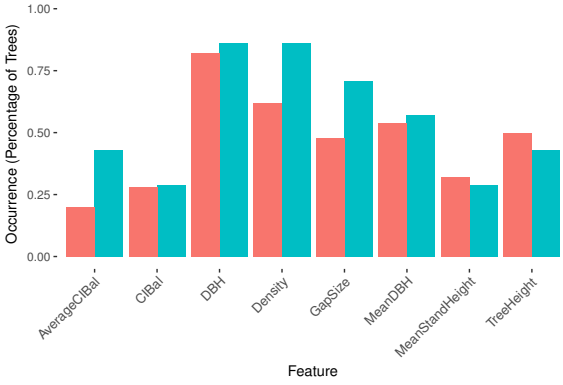

Figure 5: Nezer data: selection of original features in 50 programs evolved by GP (red) and in 7 features selected by VSURF (blue).

variables is greatly preferred by the end user as it can shed light on key features that might contribute to damage, and therefore inform management practice. This is examined in more detail in the next section.

6.3 Variable Analysis

In constructing new features, GP is essentially performing *feature selection* from the original features, i.e. each evolved program contains only a subset of the original features. Figures 5 and 6 show the proportion of the 50 *new* features that contain at least one terminal node corresponding to each of the 8 original features for Nezer and Aquitaine respectively. The figure also shows the same calculation applied only to the features selected by the VSURF procedure. For Nezer, both distributions follow similar patterns, with the variables *Diameter at breast height (DBH)* and *density* appearing most important. This resonates with current understanding of wind-damage: DBH is considered as one of the most important variables in modelling wind damage risk and stand density is important because it indicates how many trees there are in each hectare to absorb the wind loading. For Aquitaine, while *Density* remains important, the variable $AverageCIBal$ which reflects the competition index between trees also becomes important. The Nezer forest is known to be uniform in terms of tree and soil type, hence the competition index is likely to be less variable amongst individual trees. In contrast, in Aquitaine there is significant more variability between trees in stands and this variable has more relevance. A small tree within a stand will have a high $CIBal$ and the biggest trees a small $CIBal$. Hence, this measure can essentially be considered as an of inverse of the DBH measure that appears important in Nezer.

7 CONCLUSION

We have described a novel method for predicting wind damage in forests, that can be used to inform forest management techniques to minimise financial and timber loss in forests.

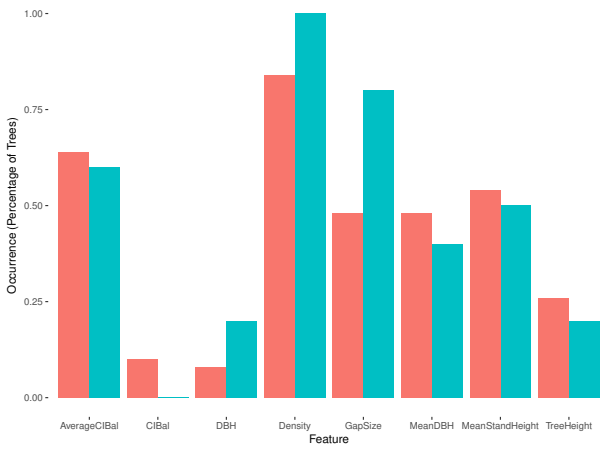

Figure 6: Aquitaine data: selection of original features in 50 programs evolved by GP (red) and in 2 features selected by VSURF (blue).

The work represents a collaboration computer scientists and forestry experts and represents a new approach within the forestry industry, which has previously relied on mechanistic modelling and models built using logistic regression to predict damage.

The approach exploited GP in order to generate a large number of useful features without having to pre-specify the number to evolve. We introduced the use of Hellinger distance to score fitness in light of the imbalanced data-set and demonstrated that this evolves features that score more highly than the original data. It is computationally inexpensive to generate large numbers of new features in this manner. We then applied an efficient method from machine-learning to perform variable-selection in conjunction with a random forest classifier. This hybridisation of EC and machine-learning techniques exploits strengths in both fields to provide a powerful means of classification.

The results show that classifiers could be built using the features that significantly outperformed classifiers that only had access to the original features. In addition, Accuracy and AUC are significantly higher than results reported in [13] that used logistic regression models on the same region, although on smaller subsets of the data used here. The variable-selection method also enabled a small number of useful features to be identified. These were then analysed in order to understand which of the original features were present in the evolved features. Importantly, the key variables identified can be readily interpreted by forestry experts, that is, the GP programs are explainable. This analysis provides additional information that can be used to guide how forests are best managed to reduce wind damage in future.

REFERENCES

[1] M. Ahmed, S.and Zhang, L. Peng, and B. Xue. 2014. Multiple feature construction for effective biomarker identification and classification using genetic programming. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. ACM, 249–256.

[2] A. Albrecht, M. Hanewinkel, J. Bauhus, and U. Kohnle. 2012. How does silviculture affect storm damage in forests of south-western Germany? Results from empirical modeling based on long-term observations. *European Journal of Forest Research* 131, 1 (2012), 229–247.

[3] K.J. Archer and R.V. Kimes. 2008. Empirical characterization of random forest variable importance measures. *Computational Statistics & Data Analysis* 52, 4 (2008), 2249–2260.

[4] G.S. Biging and M. Dobbertin. 1995. Evaluation of competition indices in individual tree growth models. *Forest Science* 41, 2 (1995), 360–377.

[5] Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5–32.

[6] D.A. Cieslak, R.T. Hoens, N. V. Chawla, and W. P. Kegelmeyer. 2012. Hellinger distance decision trees are robust and skew-insensitive. *Data Mining and Knowledge Discovery* 24, 1 (2012), 136–158.

[7] A. Colin, C. Meredieu, T. Labbe, and T. Belouard. 2010. *Etude retrospective et mise a' jour de la ressource en pin maritime du massif des Landes de Gascogne apres la tempete Klaus du 24 janvier 2009*. Technical Report IFN n2010-CER-2-077. Institut National de l'Information Geographique et Forestiere.

[8] V. Cucchi, C. Meredieu, A. Stokes, S. Berthier, D. Bert, M. Najar, A. Denis, and R. Lastennet. 2004. Root anchorage of inner and edge trees in stands of Maritime pine (Pinus pinasterAit.) growing in different podzolic soil conditions. *Trees* 18, 4 (2004), 460–466.

[9] Commission des affaires economiques. 2009. Les consequences de la tempete du 24 janvier 2009 das le Sud-Quest. http://www.assemblee-nationale.fr/13/rap-info/i1836.asp. (2009). [Online; accessed 23-01-2017].

[10] M.A. Dorning, J.W. Smith, D.A. Shoemaker, and R.K. Meentemeyer. 2015. Changing decisions in a changing landscape: How might forest owners in an urbanizing region respond to emerging bioenergy markets? *Land Use Policy* 49 (2015), 1 – 10.

[11] R. Genuer, J.-M Poggi, and C. Tuleau-Malot. 2015. VSURF: An R package for variable selection using random forests. *The R Journal* 7, 2 (2015), 19–33.

[12] H. Guo and A. Nandi. 2006. Breast cancer diagnosis using genetic programming generated feature. *Pattern Recognition* 39, 5 (2006), 980–987.

[13] K. Kamimura, B. Gardiner, S. Dupont, D. Guyon, and C. Meredieu. 2016. Mechanistic and statistical approaches to predicting wind damage to individual maritime pine (Pinus pinaster) trees in forests. *Canadian Journal of Forest Research* 46 (2016), 88–100.

[14] John R. Koza. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA.

[15] K. Krawiec. 2002. Genetic programming-based construction of features for machine learning and knowledge discovery tasks. *Genetic Programming and Evolvable Machines* 3, 4 (2002), 329–343.

[16] Y. Lin and B. Bhanu. 2005. Evolutionary feature synthesis for object recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 35, 2 (2005), 156–171.

[17] M. Muharram and G.D. Smith. 2005. Evolutionary constructive induction. *IEEE transactions on knowledge and data engineering* 17, 11 (2005), 1518–1528.

[18] K. Neshatian, M. Zhang, and P. Andreae. 2012. A filter approach to multiple feature construction for symbolic learning classifiers using genetic programming. *IEEE Transactions on Evolutionary Computation* 16, 5 (2012), 645–661.

[19] G. Patterson and M. Zhang. 2007. Fitness functions in genetic programming for classification with unbalanced data. In *Australasian Joint Conference on Artificial Intelligence*. Springer, 769–775.

[20] M. Smith and L. Bull. 2005. Genetic programming with a genetic algorithm for feature construction and selection. *Genetic Programming and Evolvable Machines* 6, 3 (2005), 265–281.

[21] B. Tran, B. Xue, and M. Zhang. 2016. Genetic programming for feature construction and selection in classification on high-dimensional data. *Memetic Computing* 8, 1 (2016), 3–15.