Forest Cover Classification

David Nicolay
Department of Computer Science
Stellenbosch University
Stellenbosch, South Africa
26296918@sun.ac.za

***Abstract*—This document is a model and instructions for LaTeX. This and the IEEEtran.cls file define the components of your paper [title, text, heads, etc.]. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.**

***Index Terms*—component, formatting, style, styling, insert.**

I. Introduction

introooo TODO remove "Fig. 1", even at the beginning of a sentence.

TABLE I
TABLE TYPE STYLES

Table Head	Table Column Head		
	Table column subhead	*Subhead*	*Subhead*
copy	More table copy[a]		

[a]Sample of a Table footnote.

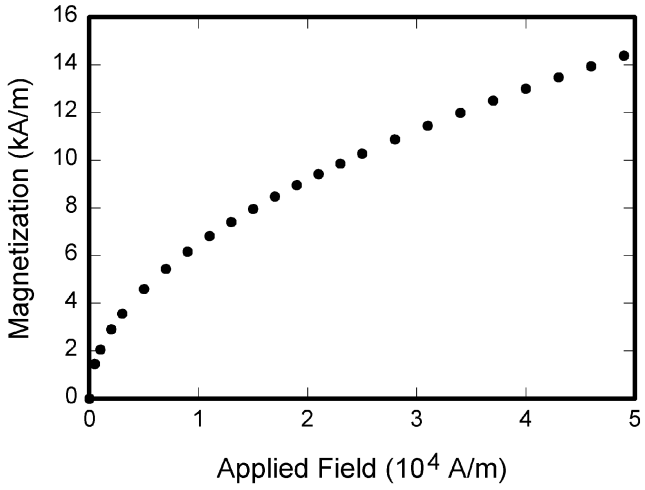

Fig. 1. Example of a figure caption.

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity "Magnetization", or "Magnetization, M", not just "M". If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write "Magnetization (A/m)" or "Magnetization {A[m(1)]}", not just "A/m". Do not label axes with a ratio of quantities and units. For example, write "Temperature (K)", not "Temperature/K".

II. Introduction

III. Background

A. k-nearest Neighbours (kNN)

The nearest neighbour algorithm is an example of instance-based learning where the prediction for an instance is made by comparing the instance to similar training instances. This is a lazy learning approach since we delay processing training data until prediction is needed. Since kNN is also similarity-based, it is non-parametric, which means that it does not learn a mapping from the input space to the output space.

The k-nearest neighbour algorithm is presented in Algorithm 1 which explains that we calculate the distance from each data point in our training dataset to the distance. From there, we choose the closed k observations and perform inference. In a classification context we may perform majority voting to determine the class, in a regression context we use the mean or median of the k nearest observations.

In order to use the algorithm we need to define a value of k which decides the number of nearest neighbour instances to consider when predicting. Choosing a value a low value for k results in a high variance, but low bias leading to an unstable model which tends to overfit. Choosing a large value for k results in a smaller variance, but high bias, leading to a more stable model which tending to underfit. Measures such as k-fold cross validation assist in the selection of a value of k.

Algorithm 1 k-Nearest Neighbors Algorithm

```
function kNN(D, x, k)
    for all x' ∈ D do
        d = distance(x, x')
    end for
    sort(d)
    S = set of k patterns in D closest to x
    return class as majority class in S
end function
```

Notation: D denotes the training dataset, x is the query instance, k is the number of nearest neighbours, and distance$(\cdot)$ is a distance metric (e.g., Euclidean distance).

B. Classification Trees

Classification trees represent a fundamental approach in supervised machine learning for predictive modelling, where the learned model forms a hierarchical tree structure with non-terminal nodes representing decisions on descriptive features and terminal leaf nodes representing target feature predictions [1]. A classification tree is a type of decision tree where leaf nodes represent different discrete classes.

Classification trees employ recursive partitioning where a training dataset is systematically divided into increasingly homogeneous subsets based on feature values. It uses a greedy strategy to select the feature to partition upon at each split that maximises information gain [2]. The information gain criterion, derived from Shannon's entropy measure, quantifies the reduction in uncertainty achieved by partitioning the data according to a specific feature test. Formally, given a dataset D with M different classes, the entropy is calculated as:

$$H(D) = -\sum_{m=1}^{M} p(y_m) \log_M p(y_m)$$

where $p(y_m)$ represents the probability of class y_m occurring in D. When the dataset is partitioned on feature α into O outcomes, the information gain is computed as:

$$\text{gain}(\alpha) = H(D) - H_\alpha(D)$$

where

$$H_\alpha(D) = \sum_{o=1}^{O} p_o \, H(D_o)$$

represents the weighted entropy after the split.

Since we recursively partition until either a stopping condition is satisfied or all subsets are homogeneous, classification trees inherently lead to overfitting, as the algorithm attempts to perfectly classify all training instances, often creating overly complex trees that capture noise rather than underlying patterns [1]. To remedy this, post-pruning techniques are used to remove branches that do not improve generalisation performance.

Classification trees can naturally handle both categorical and numerical features. Classification tree induction algorithms can cope with missing values. Unfortunately, classification trees exhibit an axis-aligned bias which restricts decision boundaries to be parallel to feature axes.

C. Expectations With Respect to Data Quality Issues

Since kNN relies heavily on distance metrics, it will be severely impacted by missing values since it cannot compute meaningful distances between points. This can however be handled by ignoring the missing values in the distance calculation and scaling up the weight of the non-missing descriptive features. The kNN algorithm can actually be used as an imputer itself as well. If nothing is done the algorithm will either fail or give biased distances. Classification trees on the other hand handle missing values more gracefully. Classification trees can adjust the gain ratio calculation

$$\text{gainRatio}(x) = (1 - F) \times \big(H(D) - H_x(D)\big)$$

where F is the fraction of missing patterns.

With regards to the Facet feature being correlated with the Aspect feature, kNN will be moderately impacted since the distance calculation used will overweight certain areas of similarity. There is no built in mechanism to handle feature correlation as a data issue. Classification trees will be minimally impacted since trees naturally handle correlated features through feature selection. • The Facet feature is correlated with the Aspect feature. • The Inclination feature contains only noisy values. • There remain features with outliers. • There remain features with numeric ranges that differ significantly from one another. • There are numerical and categorical features. • Feature Water Level has cardinality of one. • Feature Observation ID has a unique value for each observation. • The class distribution remains skew

IV. Methodology

V. Empirical Procedure

VI. Research Results

VII. Conclusion

References

Please number citations consecutively within brackets [**?**]. The sentence punctuation follows the bracket [**?**]. Refer simply to the reference number, as in [**?**]—do not use "Ref. [**?**]" or "reference [**?**]" except at the beginning of a sentence: "Reference [**?**] was the first ..."

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors' names; do not use "et al.". Papers that have not been published, even if they have been submitted for publication, should be cited as "unpublished" [**?**]. Papers that have been accepted for publication should be cited as "in press" [**?**]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [**?**].

References

[1] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[2] J. Kelleher, B. Namee, and A. D'Arcy, *Fundamentals of Machine Learning for Predictive Data Analytics, second edition: Algorithms, Worked Examples, and Case Studies*. MIT Press, 2020. [Online]. Available: https://books.google.co.za/books?id=wtGMEAAAQBAJ