

Forensic Glass Identification

(or, Jeez Glass is Really Weird)

Zakee Khattak

0 Abstract

Machine learning can be used to identify the source of broken glass - building windows, vehicle windows, containers, tableware, or headlamps - given a dataset with samples. A nearest-neighbor algorithm with three neighbors is the best choice to apply to this problem, while a logistic regression algorithm does poorly.

1 Background

Glass is not merely glass - its chemical composition and refractive ability can identify it as belonging to building windows, vehicle windows, containers, and tableware, as well as others. This is especially significant in forensics, as broken glass can be quite common in crime scenes, and the glass - if identified correctly - can be used as evidence in court.

2 Objective

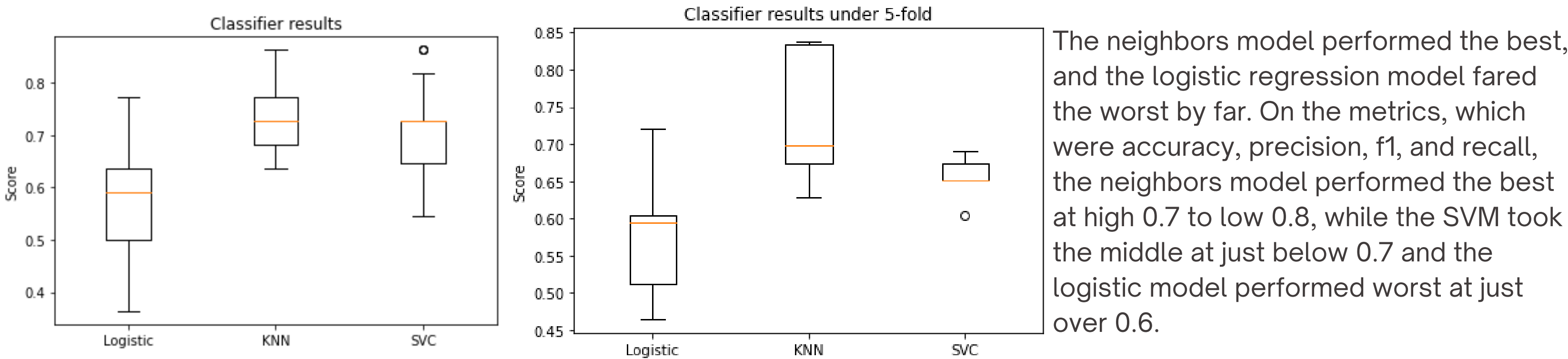
Train a model that can accurately identify where a piece of glass comes from, given its chemical composition and refractive index.

3 Approach

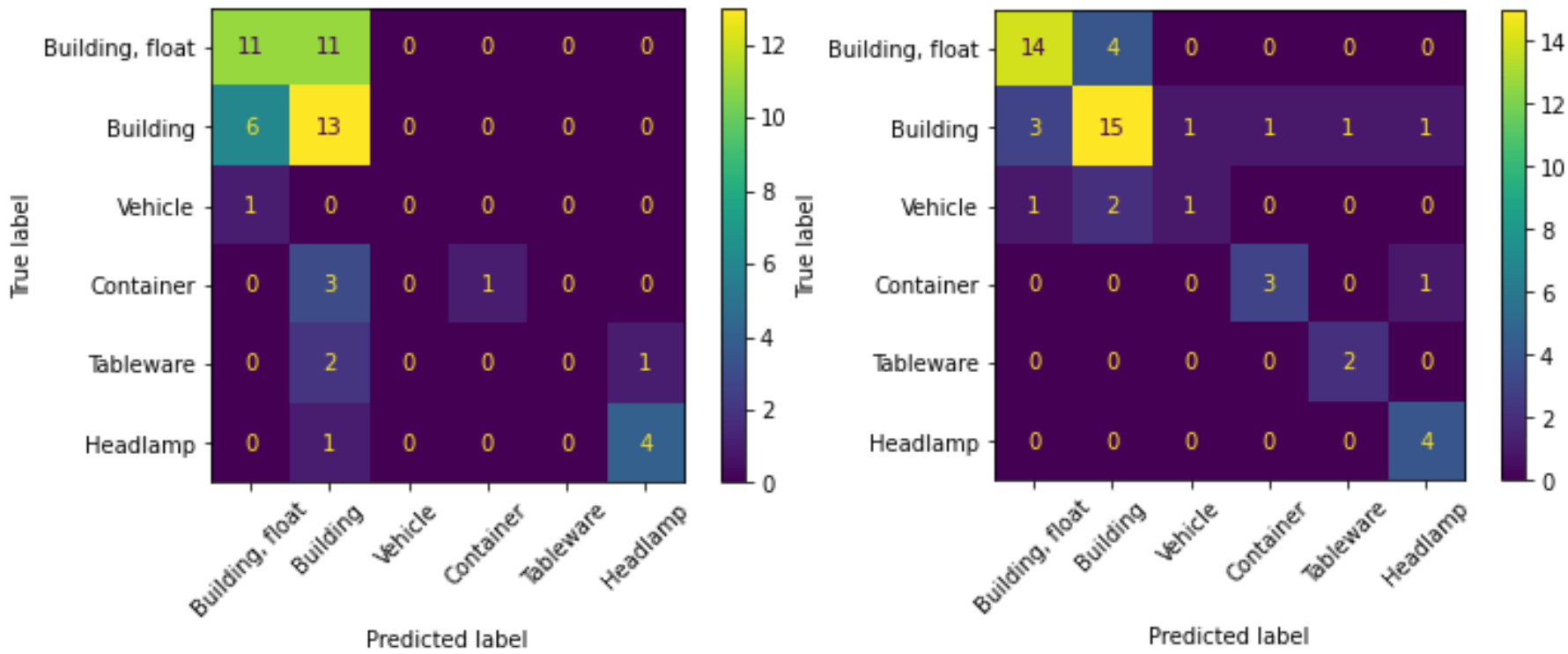
Train several multi-class classifiers, using logistic regression, K-Nearest-Neighbor, and Support Vector Machine algorithms.

The data set used has 214 instances with 9 features and six classes. Two features were dropped due to low information gain. Models were tested with both 10% train-test split and 5-fold cross validation.

4 Results



5 Analysis and Conclusion



Logistic regression (confusion matrix far left) seemed to have a hard time differentiating between the different types of building glass, which was not shared between the models. All models had difficulty with the types of glass that were less represented in the data set, save for the neighbors model (confusion matrix near left).

In conclusion, while the logistic regression and SVM models failed to live up to expectations, the neighbors model performed to an acceptable degree.

6 References

German, B.. (1987). Glass Identification. UCI Machine Learning Repository.
Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.
API design for machine learning software: experiences from the scikit-learn project, Buitinck et al., 2013.