# Exam Assignment: Forensic Identification of Glass Fragments

Machine Learning (BSc DS) Fall 2020, IT University of Copenhagen

## 1 Introduction and formalities

This is the project description for the exam project in the Machine Learning course for the BSc program in Data Science at the IT University of Copenhagen. The exam must be submitted electronically via LearnIT no later than 14.00 on 13th December.

**Groups.** You need to work in groups of 2–3 people for this project. Please register as a group in LearnIT by 14.00 on Wednesday, 11th November. The course manager reserves the right to modify the grouping if necessary. Only one person for each group should submit the project.

### 1.1 What should be handed in?

You must hand in both a report and the source code you have developed during the project. Note that the project's evaluation will be based almost entirely on the report; the source code should be seen as a supporting document.

Make sure to use correct references to works of other people in your report, including references to the coursebook, Bishop (2006). This also applies to code; if you copy or take inspiration from code developed by other people, this should be stated clearly in your report.

**Report.** The report should be submitted as a single PDF file. There is a strict limit of 15 pages, including figures, tables, code snippets, references, and appendixes, but excluding the front page. The project must be typeset with at least 11pt font size and margins of at least 2 cms. The report must be in PDF format and have a front page that meets the ITU requirements.[1]

**Implementation and code.** Your implementation has to be in Python. The code must be handed in as a single file (either a zip or tar archive). Except where explicitly stated, there are no restrictions as to which Python libraries you may use.

Your code should be organised such that it is easy to read, i.e. you have to use descriptive names for files, functions, variables, etc. The code may be organised in regular Python source files (.py files) or Jupyter notebooks.

[1] Found at `https://itustudent.itu.dk/study-administration/exams/submitting-written-work`

## 2 Forensic identification of glass fragments

This project explores the possibility of using the elemental composition and refractive index to determine the origin of a very small glass fragment. This was studied by Evett and Spiehler (1987), who wrote in their paper:

> *Glass is a material which figures prominently in the investigation of crimes such as burglary and criminal damage in which it is common for a window to be smashed violently, either to gain access or as an act of vandalism. If a suspect is apprehended for such an offence then it is almost a routine matter to submit articles of his clothing to a forensic science laboratory so that a scientist may determine whether or not there is evidential material present.*
>
> *Evett and Spiehler (1987)*

The study by Evett and Spiehler (1987) is just one example of the many contributions to the field of forensic science that came out of the UK Forensic Science Service before it was closed in 2012. For this project, we use the data from their study, as found in (Dua and Graff, 2019), to investigate suitable classification techniques for determining the origin of a glass fragment.

### 2.1 Glass fragment data

For this project we consider the six types of glass listed in Table 1.

| Glass type | Integer code |
|---|---|
| Window from building (float processed) | 1 |
| Window from building (non-float processed) | 2 |
| Window from vehicle | 3 |
| Container | 5 |
| Tableware | 6 |
| Headlamp | 7 |

Table 1: The six glass types provided by Dua and Graff (2019). Note that there is no type 4; the labeling is a remnant from the original dataset, which covered seven types of glasses.

Measurements were taken on a total of 214 glass fragments. For each glass fragment, you have a measure of its refractive index (RI), which is a standard measurement taken for forensic purposes both because of the significant variation between types of glass and because it is possible to measure precisely even for small fragments. Further, you have available the glass fragment's chemical composition in terms of the weight percent for each of eight different elements. Overall the dataset contains 9 features as listed in Table 2

| RI | Na | Mg | Al | Si | K | Ca | Ba | Fe |
|---|---|---|---|---|---|---|---|---|
| refractive index | Sodium | Magnesium | Aluminum | Silicon | Potassium | Calcium | Barium | Iron |

Table 2: Attributes of the Glass identification dataset.

The glass fragments are divided into a training set of 149 fragments and a test set of 65 fragments.

`df_train.csv` and `df_test.csv`

## 3 Scientific requirements to the project

This project aims to investigate methods for determining the origin of a glass fragment. You should carry out and report an analysis of the glass fragment data while making sure that you cover all of the tasks set out below.

### 3.1 Visualisation and exploratory data analysis

Make use of dimensionality reduction methods such as LDA or PCA to present some visualisation of the dataset. You should include in the report some reflections on your choice of method and what the reader should learn from your visualisation.

### 3.2 Classification

You should explore at least five classification methods:

**M1.** Linear or quadratic discriminant analysis as you see fit.

**M2.** Decision Trees

**M3.** Support Vector Machines

**M4.** k-nearest neighbours using 2 features that you have chosen by dimensionality reduction.

**M5.** One or more classification methods of your own choice.

The first two methods, M1 and M2, should be implemented entirely by you. For these two methods, you may use only any standard Python libraries and the numerical libraries NumPy and SciPy. *The only machine learning library you may make use of is TensorFlow*, i.e Keras, or any other similarly high-level APIs are not allowed.

Your report should describe and discuss how you have implemented the two methods. Note that the report should be self-contained and that the assessment of your implementation is based on your description in the report. Please include a discussion of how you have asserted your implementation's correctness; for testing, you may rely on comparisons to existing implementations in Python.

For the remaining methods, M3, M4, and M5, you may use any library you wish with no restrictions.

**For each method please make sure to include**

- A brief introduction to the method.
- A thorough description of how you applied the method to the data, including details needed for an independent reproduction of your results.
- How you have gone about selecting any hyperparameters for the method.
- A discussion of how you expect the method to perform.
- Some discussion of the obtained decision boundaries. The report should include a suitable visualisation of the decision boundaries for (at least) the classification by $k$-nearest neighbors, M4.

Try to describe as much as you can without referring to code or implementation; imagine that your report should target a broad audience, such as a peer using a different framework than Python or a forensic scientist, perhaps unfamiliar with coding.

### 3.3 Discussion of the results

Your report should include a thorough discussion of the performance of each of the methods applied. In particular, you should compare the methods' performance and guide the reader in interpreting the results. Use your expert knowledge to explain the results; for instance, why do particular methods perform better than others?

### 3.4 Additional considerations

- You have been provided with a split of the database into a training set and a test set. If you had been provided with only the 214 observations as one dataset, how would you have proceeded?
- For forensic purposes, it is paramount to have some idea of the uncertainty of the classification. How would you go about addressing this?
- The data provided comes from an experimental setting. What are the implications in casework where a glass fragment is obtained as part of the evidence?

## References

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning.* Springer.

Dua, D. and Graff, C. (2019). UCI machine learning repository.

Evett, I. W. and Spiehler, E. J. (1987). Rule induction in forensic science. In *KBS in Government*, pages 107–118. Online Publications, `http://www.cs.ucl.ac.uk/staff/W.Langdon/ftp/papers/evett_1987_rifs.pdf`.