**Experimentation**

**Overview**

To test the data, a wide range of models were used. In each, their usage is compared to uses previously identified in the literature. The concept behind the model is explained and the method used is detailed. Results are then thoroughly discussed.

Initial tests will be classification problems, followed by regression.

Two *classes* of model are used: linear and ensemble.

Different types of encoding will also be tested, as well as the influence of weather on prediction.

**Pre-processing**

Some pre-processing is necessary before using the data output from the LOAD stage. This mostly takes the form of encoding categorical variables and dropping unnecessary columns.

One-hot encoding TIPLOCs is impractical, as there are approximately 80,000. Nor is there any structure to their designations. Instead, their corresponding STANOX area is used. There are 89 STANOX areas[[1]](#footnote-1), corresponding to geographical areas. The first two digits of each STANOX is the area to which it belongs.

Dates are split into their respective parts: year, month, day, day of week. Hour and minutes are encoded into a single minutes variable. Different variables are used for both origin and destination.

It is possible that a cyclical encoding would yield better results; this will be experimentally tested. Redundant columns (mostly IDs) are also discarded.

The actual time of departure must also be discarded. If the chosen model is to be applied to real-life data, this field could not be known ahead of time. It is likely it has considerable predictive power, as trains have a limited amount of delay recovery possible.

Three columns had too many NaN values: sleepers, reservations, and branding.

**Regression**

Initial results (using only two days’ worth of data) were very promising: better than 90% accuracy on out-of-the-box sklearn regression models. This is misleading, however. Approximately 7% of trains are delayed. The models have simply learned to predict ‘not delayed’, an example of the accuracy paradox.

That said, decision trees often perform well on unbalanced datasets, and did so here.

Better to use precision and recall.

Recall is the *true positive rate*.

1. https://wiki.openraildata.com//index.php?title=STANOX\_Areas [↑](#footnote-ref-1)