# Data processing

The data was originally distributed over 10 different batch files. All these files were combined into a single dataset and an additional column ‘batch’, denoting the batch to which the observation belongs was added. Next, appropriate feature names were given to the columns. For this, the format ‘S\_F\_’ was used. The underscores refer to particular sensor and feature number. The data does not contain any missing values. These pre-process steps result in the following dataframe:



# Exploratory data analysis

This section will perform initial investigations to discover possible patterns, to spot anomalies and in general to get a better understanding of the data.

The data has 13 910 measurements and contains a total of 130 features, out of which one is the dependent variable (concentration). The independent feature set contains 128 numerical features which are 8 features from 16 chemical sensors and one categorical feature, indicating the gas type of the observation. The different gas types are: Ammonia, Acetaldehyde, Acetone, Ethylene, Ethanol, and Toluene. There is a different amount of observations for each gas, the amount of observation for each gas types is listed below.

|  |  |
| --- | --- |
| Gas | Number of observations |
| Ethanol | 2565 |
| Ethylene | 2926 |
| Ammonia | 1641 |
| Acetaldehyde | 1936 |
| Acetone | 3009 |
| Toluene | 1833 |

The target variable has 59 different concentration values situated in the interval between 1.0 and 1000 ppmv. The distribution of the target value for the complete dataset and for each particular gas is presented below. This distribution is clearly not uniform, some concentration values occur a lot and some almost never. It is also noticable, the a lot of different concentration levels occur between 0 and 250. However, the amount of different concentration levels higher than 250 is rather sparse. The distribution of the target variable is also very different for the different gases. For example, it are only the gases ammonia and Acetone with concentration levels higher than 600.

|  |  |  |
| --- | --- | --- |
|  |  |  |

There are a total of 128 numeric columns containing values extracted from the different sensors. A general overview of the feature means is plotted below. Generally, three feature types can be distinct, features with very high means (>30k), features with means between 15k and 30k and features with values around the zero. A close-up of the means of the third category shows that a majority are scattered around zero. It has to be noted, that some features have a negative mean. Plots of the standarddeviations shows a behaviour in line with the means, which is expected.

|  |  |
| --- | --- |
|  |  |
|  |  |

The following plots show the feature means for each particular gas. It is notable that the gases acetone and ethanol show a strong recidivist behaviour compared to the other gases.

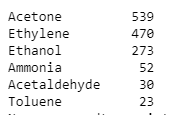
|  |  |
| --- | --- |
|  |  |

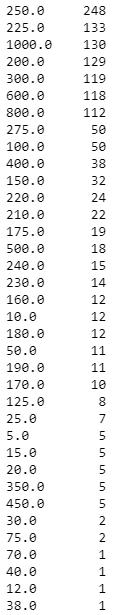
## Outlier detection

The observations of the data set where obtained in a fully computerized environment using a highly accurate measurement system. Therefore the validity of the observations is not called into question. Consequently, this section is solely performed to obtain more insights int the data and thus observations identified as outliers will not be removed or adapated.

An IsolationForest contain trees whereby the split criteria, feature and value, is chosen randomly. This random partitioning produces noticeable shorter shorter tree paths compared to normal data points. As a consequence, outliers will be typically “isolated” at the top of the tree. In addition, the trees have a much shorter max depth which results in lower memory requirement. The algorithm is quite sensitive to the **contamination** parameter**.** Thisparameterrefers to the expected proportion of outliers in the data set and is used to define which outliers are considered as outliers. The value is determined as suggested as in the original paper, which results in a value of 0.1% for our dataset.

The algorithm identifies 1387 observations as potential outliers which comes down to 0.1% of the data. The gas distribution and concentration distribution of these outliers is as follows. The fact that acetone and ethanol are that much considered as outliers is not suprising. It was already visible that these gases have remarkable higher mean values for a particular subset of features. However, the 470 observations of ethylene is rather unexpected because observations of this gas were not striking in the exploratory plots. A potential explanation for this is the fact that ethylene is the gas with second most observation in the dataset.





# Summary of the used algorithms

The goal of any linear regression algorithm is to accurately predict an output value from a given set of input features. The scikit-learn library contains an impressive amount of various models. Generally, it is often best practice to implement and compare a number of different models. 6 different regression models were used to predict gas concentrations. Each of these models have their advantages and disadvantages and outperform in a particular cases. They were chosen as such to have a variety of models. A summary of each used model:

**MLP Regressor:** A non-linear model which functions as a neural network with multiple neurons at each layer and a nonlinear activation functions between each layer. A number of parameters can be tuned to achieve optimal results. Generally, it fits very well to data of various structures and is typically more interpretable than other nonlinear models.

**KneighborsRegressor**: A non-linear model that predicts the mean of the target values of the k nearest neighbors of the observation. If fits well to a various structures of data. However, it performance is highly influenced by outliers and has difficulties when dealing with a high number of features.

**HuberRegressor**: A fast linear model designed to deal with outliers by giving them a lower weight.

**BayesianRidge**: A linear regularization regression model which uses a non-constant penalty term trained during the fit procedure. The output is less interpretable than ridge but time can be saved as no different regularization weights have to be compared.

**ElasticNet:** Elastic-Net’s loss function contains Ridge and Lasso’s penalty terms. The advantage of combining them is to smoothen Lasso’s severe feature selection and still obtain Ridge regularization characteristics. It is particulary useful when there are multiple correlated features.

**AdaBoost**: A tree based ensemble learner that trains on the residuals of previous created trees. The learning process goes much slower compared to RandomForest but it generally performs better.

PERFORMANCE METRICS