Content

1 Dataset analysis

The dataset consists of 5 columns:

- Timestamp: date and time in format %Y-%m-%d %H:%M:%S, type string.
- IPP Agglomerointisuhde: agglomeration factor, type float.
- Spray Gun Flow (lh): type float.
- BatchID: unique number for each process, type float.
- Minutes: duration of the process for each batch.

Since the batches occur in different time periods, there are plenty of NAN values.

2 Analysis of the dataset

We start by plotting the agglomeration factor and spray factor against time (minutes) for each batch (Fig. 1 and 2). We can appreciate there are some outliers corresponding to batches where minutes are above 500.

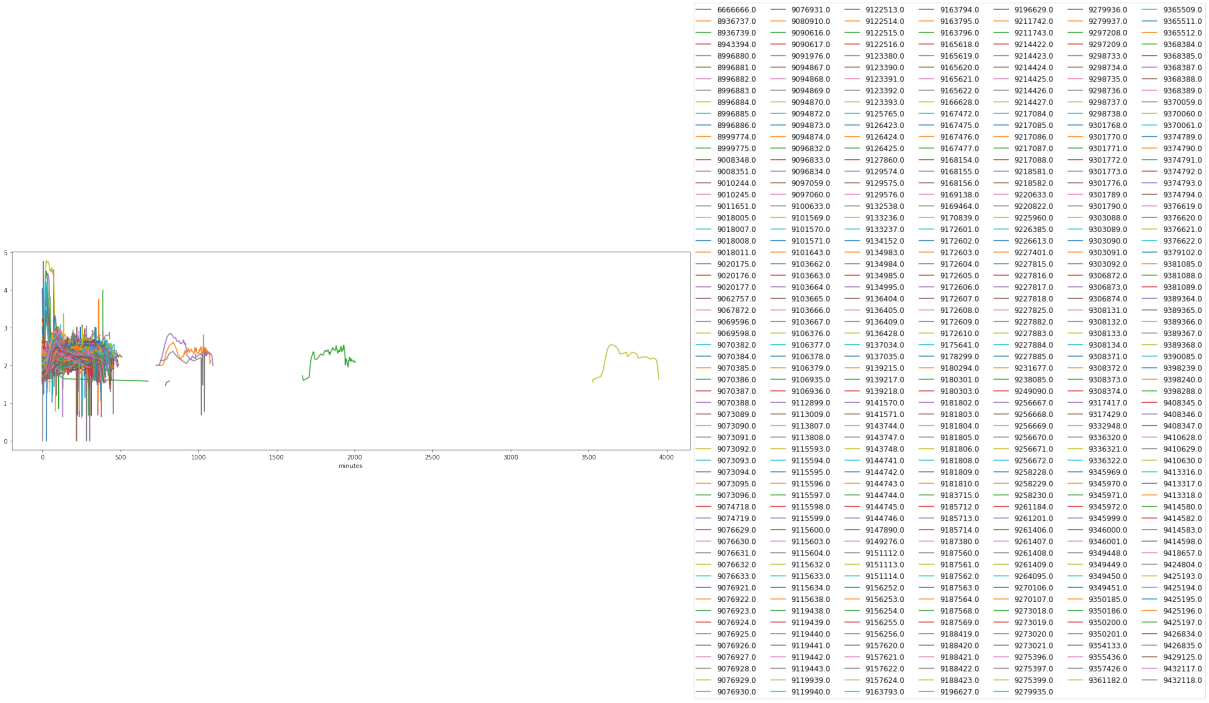

Figure 1: Agglomeration factor against time for all batches

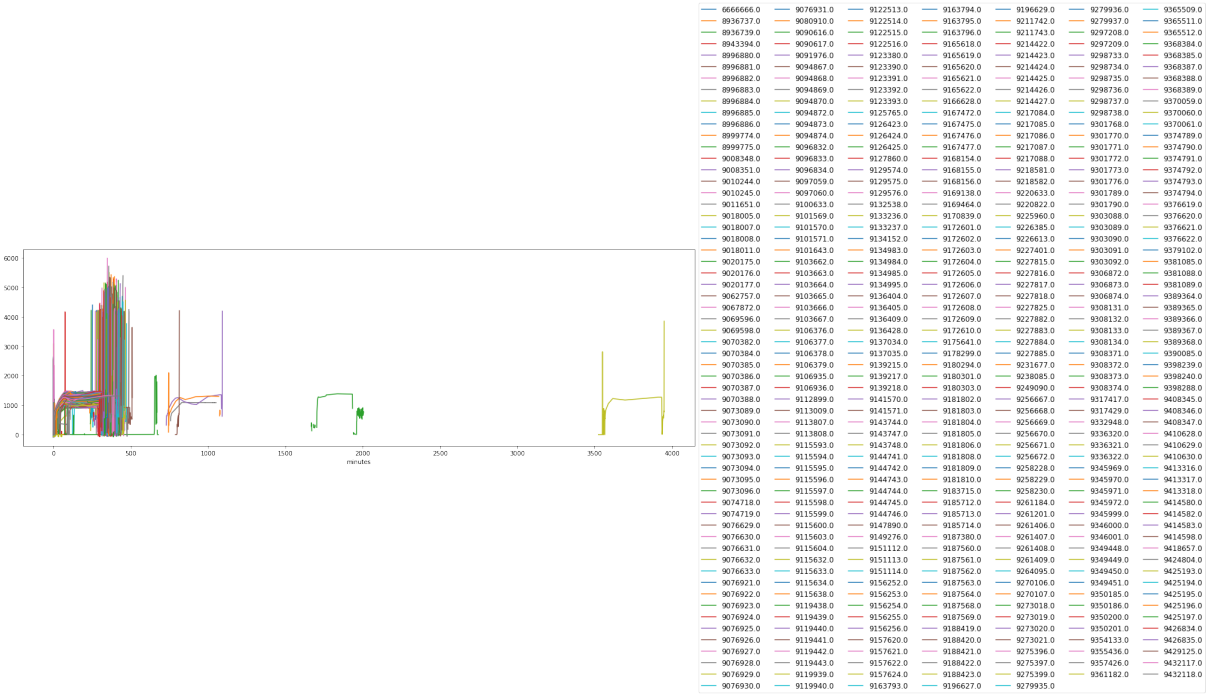

Figure 2: Spray factor against time for all batches

These outliers are filtered out and the data is visualized again. Due to the high number of batches, the information is overlapped, but we can see some patterns. In the case of the agglomeration factor (3) there is an increase of the value some minutes after starting the process and the slope tends to be 0 afterwards. In the case of the spray factor plot (4), it increases from 0 at the beginning, it becomes constant for some time with peaks at the end of the process.

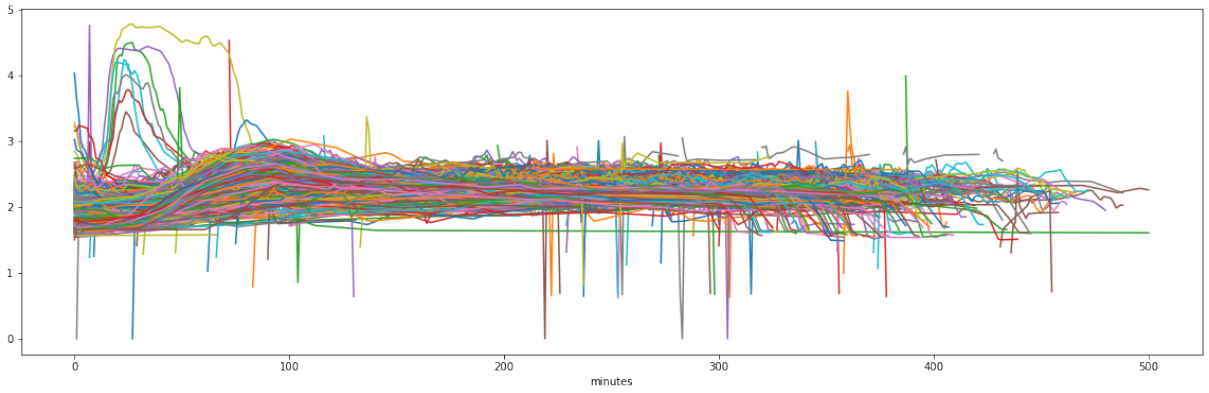

Figure 3: Agglomeration factor against time for batches with duration less than 500 minutes.

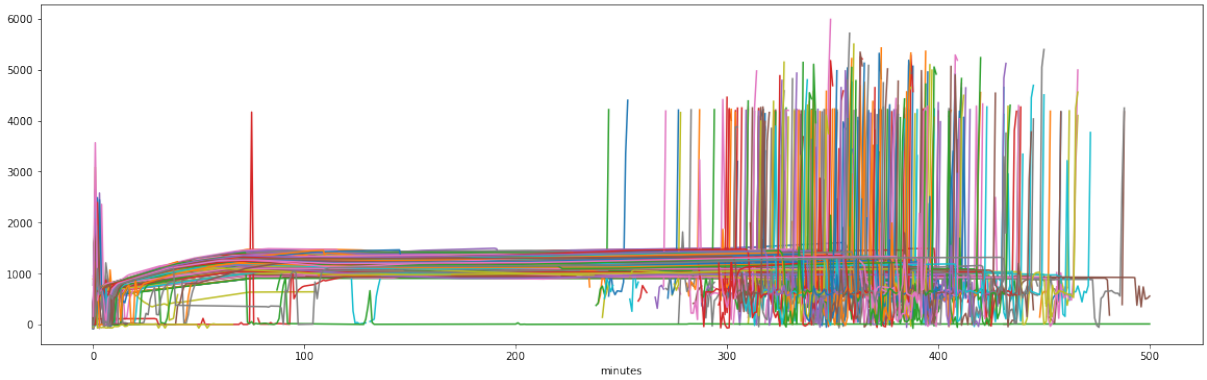

Figure 4: Spray factor against time for batches with duration less than 500 minutes.

Since the number of batches is so high, the rest of the analysis might be very difficult. Therefore, I select 3 random batches which will be used in the analysis. The selected batches are: 9073091.0, 9181809.0, 9261409.0.

Figure 5: Top plot: Agglomeration factor against time for 3 batches. Bottom plot: Spray factor against time for 3 batches

3 Individual characterization

The dataset needs to be modified the following way:

- The dataframe contains the information of only one batch.

- The timestamp values change from string to timestamp type.
- The timestamp is set as index.
- The batchID column disappears

The results showed in this section belongs to the first batch selected: 9073091.

First, we start by checking the dataset for that batch. We see the type of variable, the number of NAN values per column, the count, mean, std, min, max, 25%, 50% and 75% quantiles for each column. The function used is getDataFrameAnalysisTS.

```
(IPP Agglomerointisuhde    float64
 Spray Gun Flow (lh)       float64
 minutes                     int64
 dtype: object,
 IPP Agglomerointisuhde    0
 Spray Gun Flow (lh)       0
 minutes                   0
 dtype: int64,
        IPP Agglomerointisuhde  Spray Gun Flow (lh)     minutes
 count              299.000000           299.000000  299.000000
 mean                 2.445701          1041.287262  149.227425
 std                  0.439245           151.915574   86.831677
 min                  2.036894            72.339630    0.000000
 25%                  2.175711          1090.054077   74.500000
 50%                  2.255901          1091.798828  149.000000
 75%                  2.581746          1093.476013  223.500000
 max                  4.013786          2407.812988  302.000000)
```

Figure 6: Description and statistical summary for first selected batch.

Next, we check if the timeseries for this batch is regular by using get_TS_RegularityRatio. The value obtained is 0.9933 since the data is collected every minute and there are only 2 time points that are irregular.

3.1 Normality analysis

3.1.1 Kurtosis and skewness

The kurtosis and skewness are calculated using getUniformityStats function. The obtained values are for the agglomeration factor for one batch is:

- Kurtosis: 4.4725
- Skewness: 2.2001

Kurtosis refers to the degree of presence of outliers in the distribution. The Kurtosis for a normal distribution is 3 because when calculation the standarized fourth moment of the equation , you obtain a value of 3.

The excess of kurtosis is used in statistics and probability theory to compare the kurtosis coefficient with that normal distribution. Excess kurtosis can be positive, negative or near to zero. A positive value of Kurtosis means that it has tails and it asymptotically approaches zero more slowly than a Gaussian distribution, producing more outliers than the normal distribution. It is called leptokurtic distribution. The distribution is peaked and possesses thick tails. An extreme positive kurtosis indicates a distribution where more of the numbers are located in the tails of the distribution instead of around the mean.

Skewness is a measure of the symmetry of the data. A positively skewed distribution is a sort of distribution where, unlike symmetrically distributed data where all measures of the central tendency (mean, median and mode) are equal to each other, with positively skewed data, the measures are dispersing, which means positively skewed distribution is a type of distribution where the mean, median and mode of distribution are positive rather than negative or zero.

We have a extreme positive skewness, since the skewness value is above 1, which is not desirable for distribution, as a high level of skewness can cause misleading results. The data transformation tools are helping to make the skewed data closer to a normal distribution. For positively skewed distributions, the famous transformation is the log transformation. The log transformation proposes the calculations of the natural logarithm for each value in the dataset.The skewness of the second and third batch are around -1, a normal value. Along the report we will

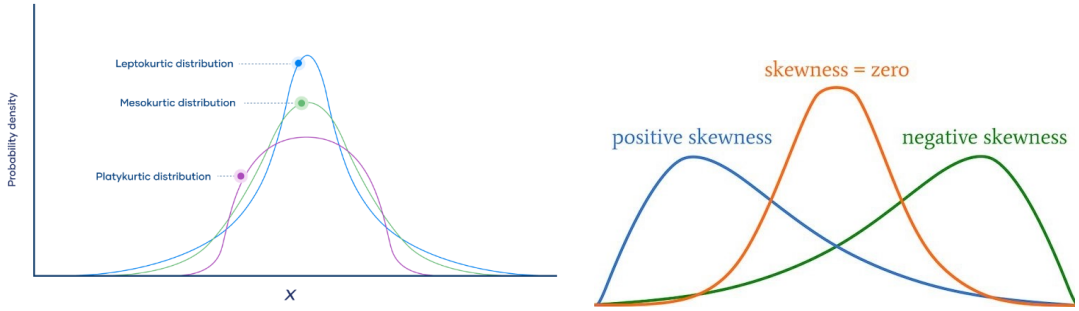

Figure 7: Example plot for different values of kurtosis and skewness.

The values of kurtosis and skewness for the spray factor in the first batch are:

- Kurtosis: 28.572
- Skewness: 0.3494

Kurtosis is more extreme but skewness is less extreme. If I had obtained high values of skewness for both parameters I would have modified the data with log transformation, but since it only occurs in only one parameter I will leave the data as it is.

3.1.2 Kolmogorov D test

Another way of checking if the dataset follows a normal distribution is by using Kolmogorov D statistics. This tests if the distribution of the dataset is different from a reference distribution (normal). The obtained p-value for the agglomeration factor and the spray factor in this batch isf 0 in both cases, lower than 0.05, so we can reject the null hypothesis of the sample coming from the reference distribution, the sample dataset doesn't follow a normal distribution for any of the parameters. This makes sense with the previous results of skewness and kurtosis.

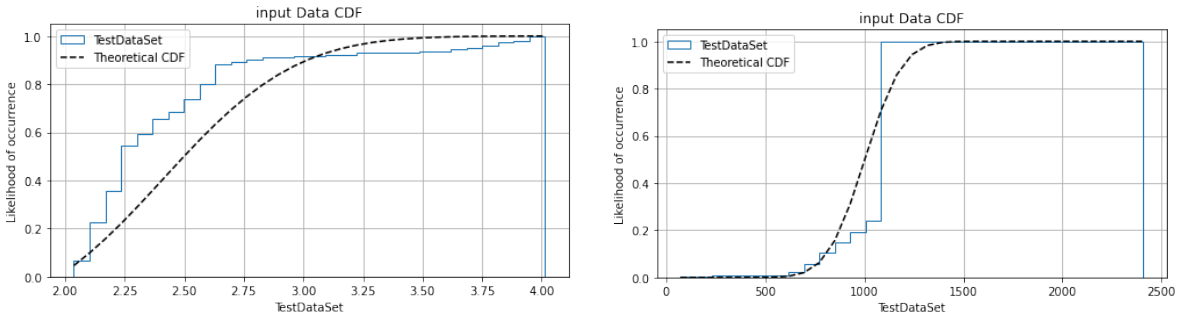

Figure 8: Kolmogorov D test for agglomeration factor in left plot and spray factor in right plot in first batch.

3.2 Detection of outliers

We detect outliers and plot them. find_plot_outliers function uses DBSCAN algorithm to find the outliers. This algorithm views clusters as areas of high density separated by areas of low density. The points are divided in core points (the circumference around this point contains a minimum number of points), border points (the number of points is less than the minimum) and noise/outliers (there are no points within the ratio).

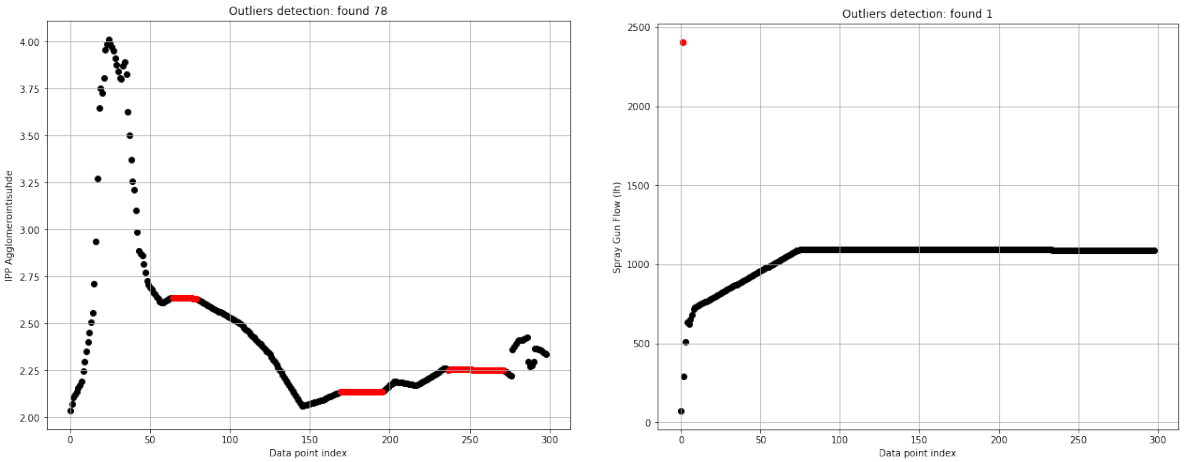

Figure 9: Outliers detection in agglomeration and spray factors for the first batch.

The agglomeration factor for this batch shows 78 outliers and the spray factor only one outlier. In the case of the spray factor, there is only one outlier which seems very evident. However, in the case of the agglomeration factor, the number of outliers is too high and they do not make sense.

We are going to optimize the parameters of the DBSCAN algorithm to obtain a better outlier detection. I calculate the optimal number of epsilon and try with different values of minPoints, but the accuracy does not improve.

I try next with the STL decomposition, where I analyze the deviation of residue and introduce some threshold for it. The plot obtained for the first batch and the agglomeration factor is:

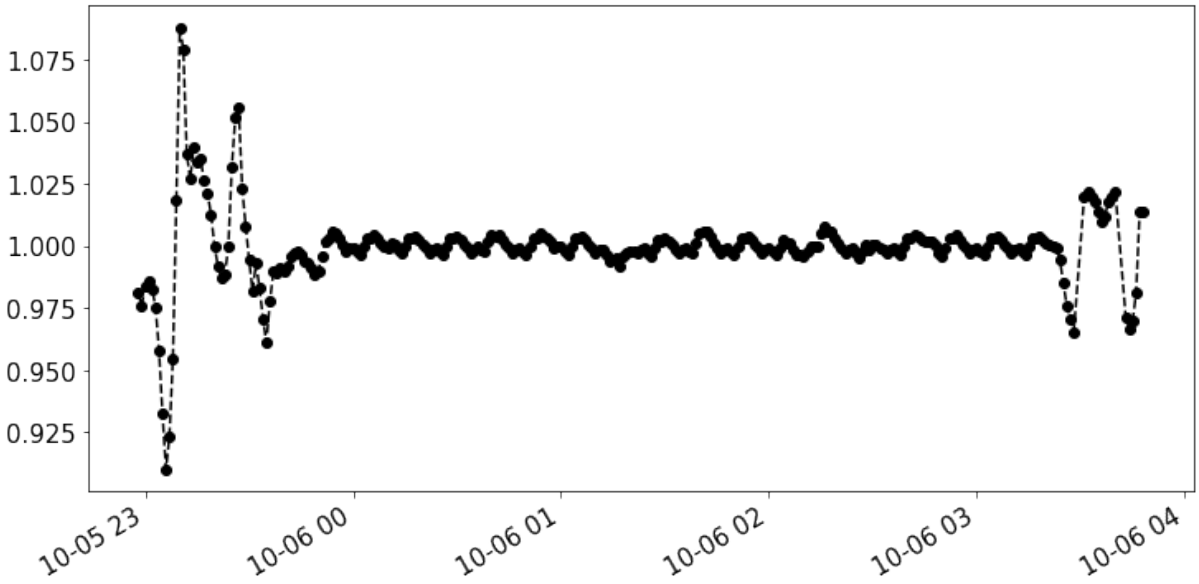

Figure 10: Plot of the deviation of residues

I can see there are some points at the beginning which could be considered outliers, but it is difficult to identify if they are outliers for sure. Therefore, I will try another method, 'Isolation Forest' method.

Isolation Forest is a unsupervised method which is based on classifications trees and I have to set the contamination fraction, the percentage of outliers that can be found in the dataset. Since the number of values in the dataset is that high, the percentage should be fairly low, so I will try with different values.

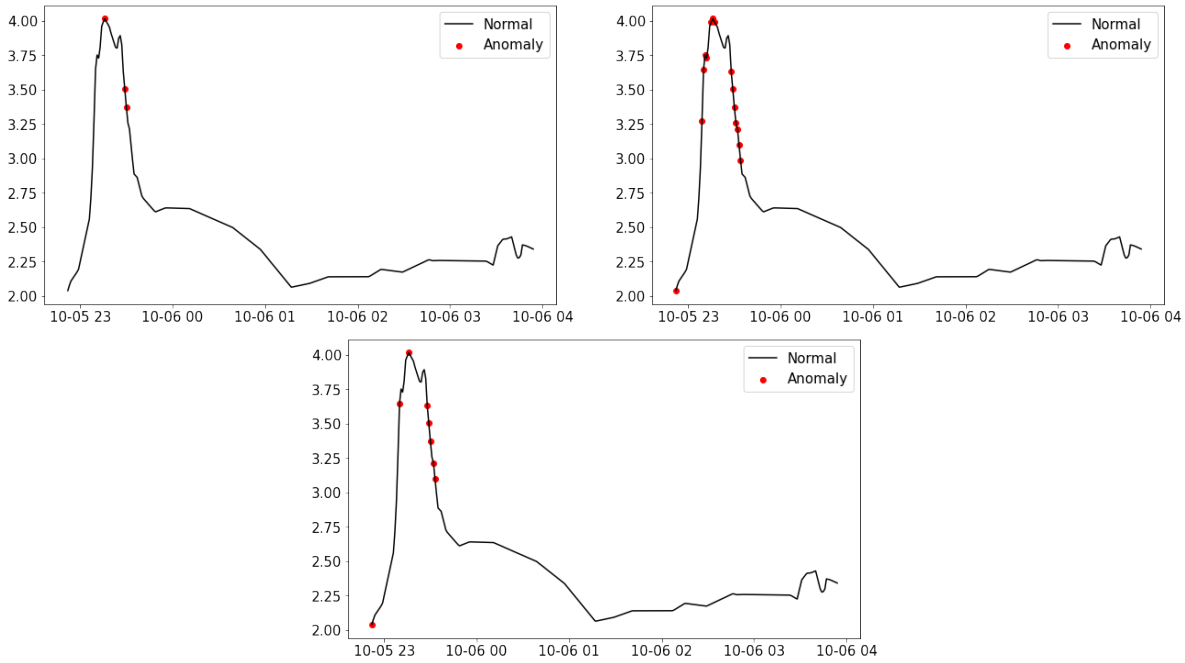

Figure 11: Outliers detected using the Isolation forest method. The left plot has a contamination percentage of 1%, the right plot has a contamination percentage of 5% and the middle plot has a percentage of 2.5%

According to the obtained plots, I would suggest that the percentage contamination should be 2-3%.

4 Patterns in a Time Series

We check the trend, seasonality and residues for the first batch. The trend is an increasing or decreasing value in the series, the seasonality is the repeating short-term cycle in the series, and the residues are the random variation in the series. The plots obtained with the multiplicative model are:

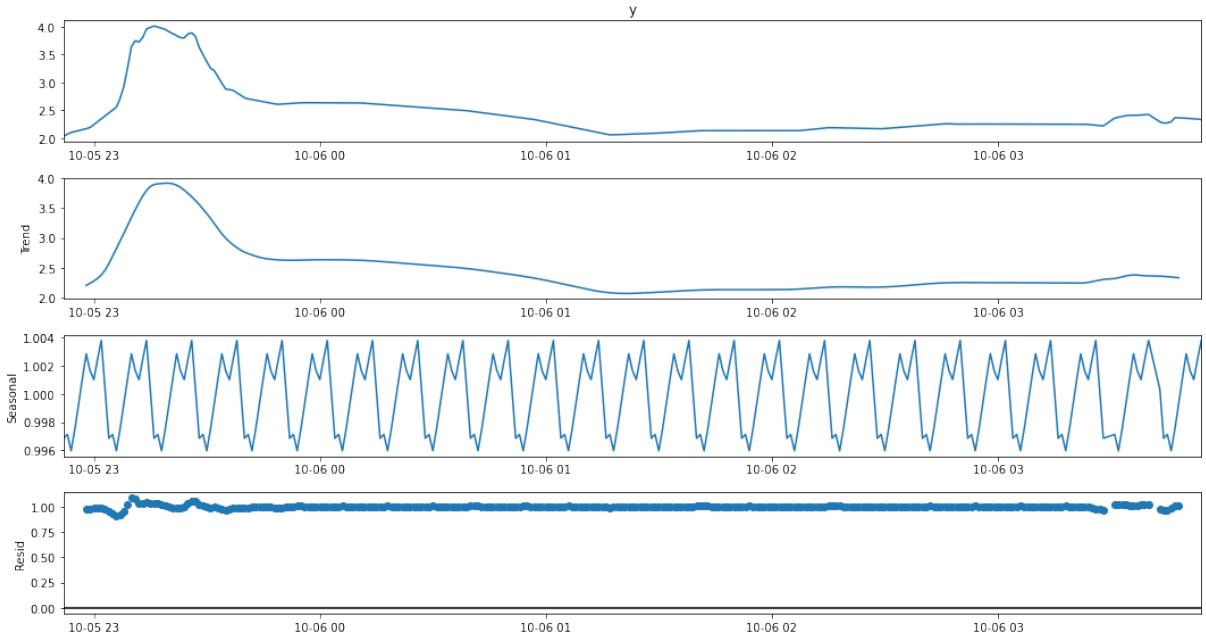

Figure 12: Decomposition of the first batch following a multiplicative model.

The plots obtained with the additive model are:

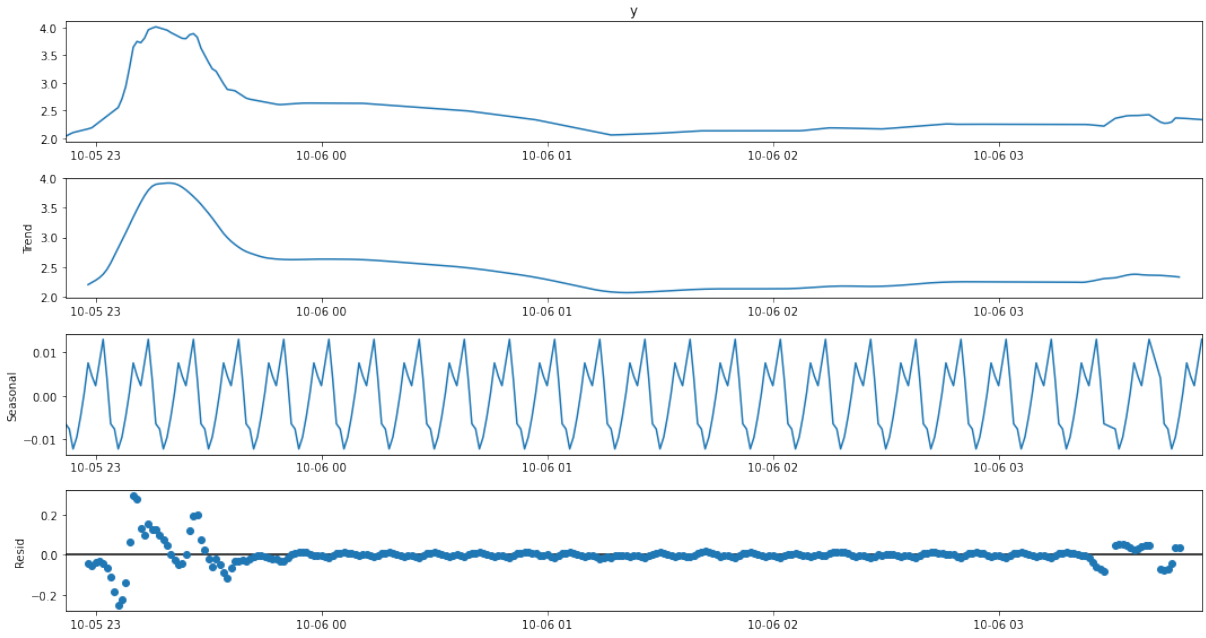

Figure 13: Decomposition of the first batch following an additive model.

There is no apparent trend in this batch, since at the beginning it increases, then it decreases but it increases again at the end. The scale for the seasonality is very small, so we can day that thee is not seasonality neither.

If we look at the residuals, we can see the same pattern. For time series, random decompositions are preferred. Since it is difficult to distinguish between the models, another approach is accessing the values for the trend, seasonality and residuals at the same point index and calculating the 'y' value at the same point.

For the multiplicative model: $y(t) = Trend * Seasonality * Noise$.

For the additive model: $y(t) = Trend + Seasonality + Noise$.

After having done the calculations, we can conclude that the time series for the first batch follow a multiplicative model. Some properties of the multiplicative model is that the magnitude of the seasonality depends on the magnitude of the data and the frequency and amplitude of the data over time changes.

5 Stationary analysis

Stationarity is a property of a time series where the values of the series is not a function of time. Therefore, statistical properties of the series are constant over time and the autocorrelation is only the correlation of series with its previous values.

The properties of the dataframes used in this part of the report are:

- It contains the value for one batch
- It contains the value for only one parameter, which is renamed to 'y'.
- Timestamp column is renamed to 'ds' and it is set as index.

The two methods used to check the stationarity of the dataset are ADF and KPSS tests (Li, 2020).

In the **ADF (Augmented Dickey Fuller) test**, the null hypothesis is that the time series is not stationary. The 3 batches are tested individually and in all cases, the p-value is bigger than 0.05, so we cannot reject the null hypothesis and none of the three datasets is stationary.

The **KPSS (Kwiatkowski-Phillips-Schmidt-Shin) test** is used to test for trend stationary. The the null hypothesis indicates the time series is stationary. The 3 batches are tested again with with this method and in all cases the p-values fall under 0.05, we reject the null hypothesis and the datasets are not stationary.

Therefore, we should convert these non-stationary time series into stationary by applying a suitable transformation. This is necessary because most of the forecasting methods use stationary data and the forecasts become more reliable. Stationarizing the series removes any persistent autocorrelation, making the predictors (lags) of the

series in the forescasting models nearly independent.

There are several methods to stationarize the dataset. I have used the **difference method** for all of the batches. After repeating ADF and KPSS tests in modified batches, the results show they are no stationary datasets.

6 Correlation analysis

6.1 Correlation between variables

We can check if there are some correlations between the parameters and make sure there is no autocorrelation detected in the residuals. For checking the correlations between parameters, we use the VAR model.

Vector AutoRegression (VAR) is a multivariate forecasting algorithm used when two or more time series influence each other. Each time series is modeled as a function of the past values, the predictors are the lags (time delayed value) of the series (Prabhakaran, July).

We calculate the lag order by selecting the one with the lowest AIC, the max lag obtained is 14. Then, we fit the model using this value and we obtain a correlation matrix between the batches. We can see there is a negative correlation between the spray factor and the agglomeration factor, this means that one parameter is increasing, the other one is decreasing. This makes sense since the spray tries to regulate the size of the beads reducing the agglomeration when this increases too much.

	Agglomeration factor	**Spray factor**
Agglomeration factor	1.0000	-0.035082
Spray factor	-0.035082	1.000

Table 1: Correlation between agglomeration factor and spray factor for one batch

6.2 Autocorrelation of lags

Then, we check the autocorrelation of the data by plotting a time series against a lag of itself. Autocorrelation -or serial correlation- is the correlation of a signal with a delayed copy of itself as a function of delay -it is the similarity between observations as a function of the time lag between them- .

The linear shape of the plot suggest that an autorregresive model is a good choice for the underlying structure of the data. It shows a strong positive autocorrelation.

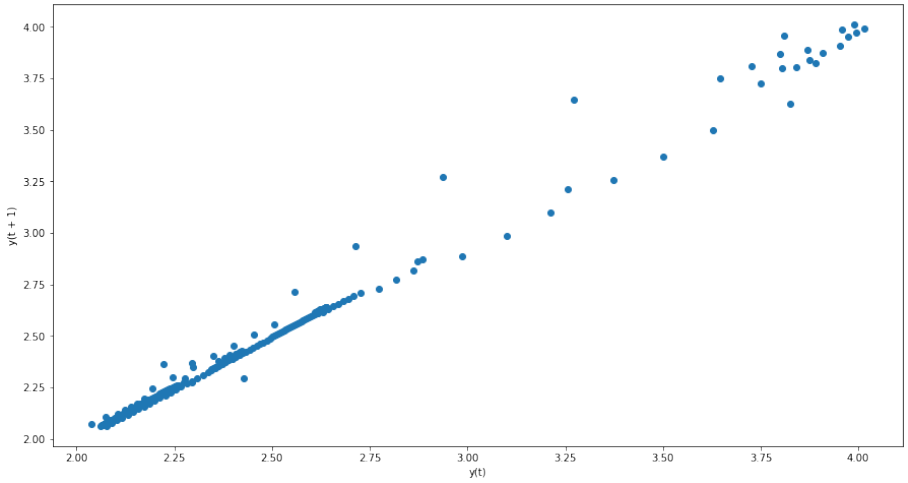

Figure 14: Lag plot of the first batch for the agglomeration factor

6.3 Autocorrelation of residuals

The residuals in a time series models are what is let over after fitting a model. For many time series models, the residuals are equal to the difference between the observations and the corresponding fitted values. Residuals are useful in checking whether a model has adequately captured the information in the data. A good forecasting method will yield residuals with the following properties:

- Residuals are uncorrelated.
- Residuals have zero mean. If the residuals have a mean other than zero, then the forecast are biased.

We check the autocorrelation of the residuals with the Durbin Watson test. Both values are around 2, this means there is no autocorrelation detected in the residuals and they are independent of each other. Therefore, we can conclude that the model has adequately captured the information in the data.

IPP Agglomerointisuhde	1.99
Spray Gun Flow (lh)	2.0

Table 2: Autocorrelation of residuals for one batch

When forecast errors are white noise, it means that all of the signal information in the time series has been by the model in order to make predictions. All that is left is the random fluctuations that cannot be modeled

7 Causality test

Granger test is applied to determine if one time series will be useful to forecast another. A prerequisite for performing the Granger causality test is that the data need to be stationary (constant mean, constant variance, and no seasonal component)

If Yt causes Xt, then Y must precede X which implies:

- Lagged vales of Y should be significantly related to X.
- Lagged values of X should not be significantly related to Y.

The null hypothesis of Granger test is that Yt does not Granger-cause Xt+1.

Grange causality only provides information about forecasting ability, it does not provide insight into the true causal relationship between 2 variables.

According to Granger causality, if a signal X1 Granger-causes a signal X2, then past values of X1 should contain information that helps predict X2 above and beyond the information contained in past values of X2 alone. This means that X1 is a predictor and X2 is a response(BANERJEE, 2020).

The Granger causality test is applied to a dataframe containing the agglomeration factor and spray factor for one batch.

	Agglomeration factor predictor	**Spray factor predictor**
Agglomeration factor response	1.0000	0.000
Spray factor response	0.3003	1.000

Table 3: Granger causation matrix between agglomeration and spray factors for the first batch.

We can find a p-value lower than 0.05 between the spray factor predictor and the agglomeration factor response, which means that spray factor can be used to predict agglomeration factor but not the other way around.

When repeating the Granger test for the second batch we obtain the following p-values:

	Agglomeration factor predictor	**Spray factor predictor**
Agglomeration factor response	1.0000	0.000
Spray factor response	0.0000	1.000

Table 4: Granger causation matrix between agglomeration and spray factors for the second batch.

In this case, the causation matrix indicates that spray factor can be a predictor for agglomeration factor, but at the same time, agglomeration factor can be a predictor of spray factor.

Related to the selection of maxlag: Granger causality has always been tested in the context of some model. The models has p past values of each of the 2 variables in the bivariate test. A conventional way to choose for this model would be to try this regression with various values of p and use track of the AIC for each lag length. And run the test using the value of p which had the lowest IC in the regressions

In general, the number of lag in the model can be different from X and Y and a Granger test will still be appropiate

8 Time lag cross-correlation

Time lag cross correlation can define the directionality between 2 signals, such as lead-and-follow relationship, in which the lead initializesa response and the follow repeats it. TLCC is measured by incrementally shifting one time series and repeatedly calculating the correlation between two signals. If the pealk correlation is at the center (offset=0), this indicates the two time series are most synchronized at that time. However, the peak correlation may be at different offset if one signal leads another.

The cross-correlation function between 2 different signals is defined as the measure of similarity or coherence between one signal and the time delayed version of another signal.