Project 1

Door movement prediction with Time- series segmentation

Marc Vernet Sancho

The goal of the project is to solve the problem of door movement prediction applying a timeseries segmentation algorithm for each variable and a random forest method for aggregating the predictions into one unique result.

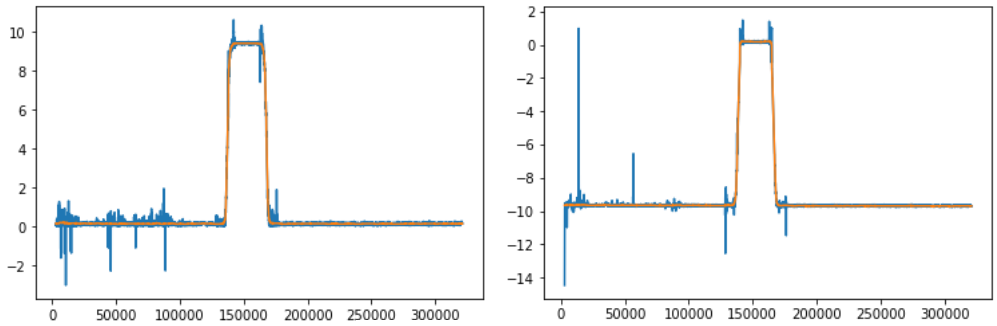
Step Detection Aggregation for Timeseries

The data of the door sensors can be considered as a separate timeseries for each variable. Considering only one variable, the solution to door opening detection can be considered a problem of segmenting the timeseries in three areas: door is opened, door is closed and door is in movement (movement can also be divided in upward or downward movement).
The problem can be considered a **Time-series segmentation problem**, for which there exists plenty of research.

Studying the plots of the variables, for some of them (x-axis and z-axis, for example) it's very easy to see that the opening and closing of the door are represented by clear steps in the timeseries value. The problem can be analyzed from a **step detection** perspective (try to detect steps in the timeseries to know when the door is being opened or closed), for which also exists extensive research.

Solution proposed

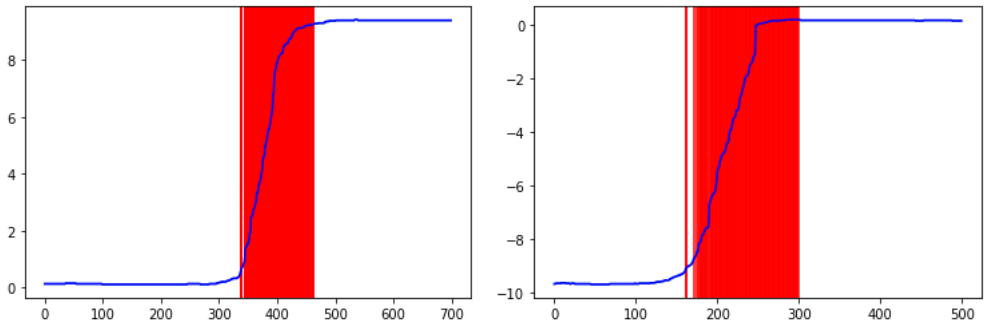
Applying a median filter does a very good job on removing noise. After the filtering, the trend on the time series is obvious (for most of the variables there's a clear step when the door is opened or closed).



x-axis and z-axis with median in orange

Next step is to find a way to detect these steps algorithmically. The proposed solution is to use the CUSUM algorithm. CUSUM algorithm is a way of detecting abrupt changes in the slope of a timeseries, and it can detect both upward and downward trends (can differentiate if the door is opening or closing).

This algorithm applied seems to give very good results on some variables. The problem in this approach is that this algorithm depends on some parameters that need to be fine tuned for each variable, so it would be required to have a different model or profile trained for each door.



x-axis and z-axis CUSUM intervals

Assuming a correct working of the CUSUM algorithm, it provides a way to detect abrupt changes on the variables that should correspond with the movement of the door. With these change points it can be known if a door is open or closed on a certain moment. A problem with this approach is that it would be impossible to know the door state on start before any movement (but because we are interested in counting opening-closing cycles it wouldn't be a problem).

The idea is that the algorithm will be able to know how much open it's the door (in percentages) based on how long the opening or closing segment has been (assuming constant opening and closing speed). Another problem is that the algorithm would calculate the percentage of opening based on the maximum opening seen until that moment (if the door has never been fully open, the algorithm is incapable of knowing how much it can be opened).

A solution to the labeling problems could be a labeling that autocorrects historical labels based on new data (percentage of opening could be wrong on the first prediction but would eventually get corrected assuming that the door is fully opened at least once).

Finally, the algorithm would return the prediction for each variable and a way to aggregate them into a single result is needed. A good solution would be to train a random forest method, but the problem is the lack of labeled data.

Labeling the data

The website TRAINSET offers a solution to label time series by hand. It's possible to convert the datasets to this format in order to label the data. But with the giant size of the datasets seems difficult to be able to label enough data for any meaningful machine learning task.

References

- Time series segmentation: https://en.wikipedia.org/wiki/Time-series_segmentation
- Step detection: https://en.wikipedia.org/wiki/Step_detection
- CUSUM: https://en.wikipedia.org/wiki/CUSUM
- TRAINSET: https://trainset.geocene.com/