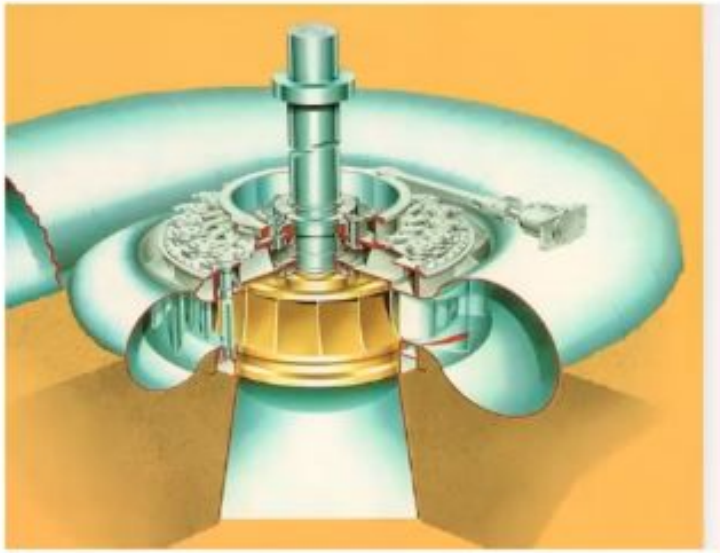
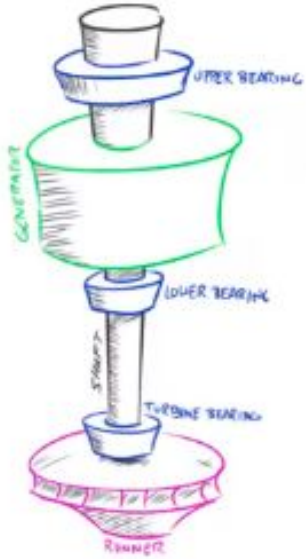
KraftHack 2022

Team Data Liberators: Hayden, Simon, Andris

Introduction

- Predicting the strain rate on 6 different bolts.
- Input data contains features such as:
  - Pressures
  - Guide Vane Openings
  - Seasonality
  - Power
- Another critical aim is to have an estimate for end of lifetime for each bolt





Data exploration

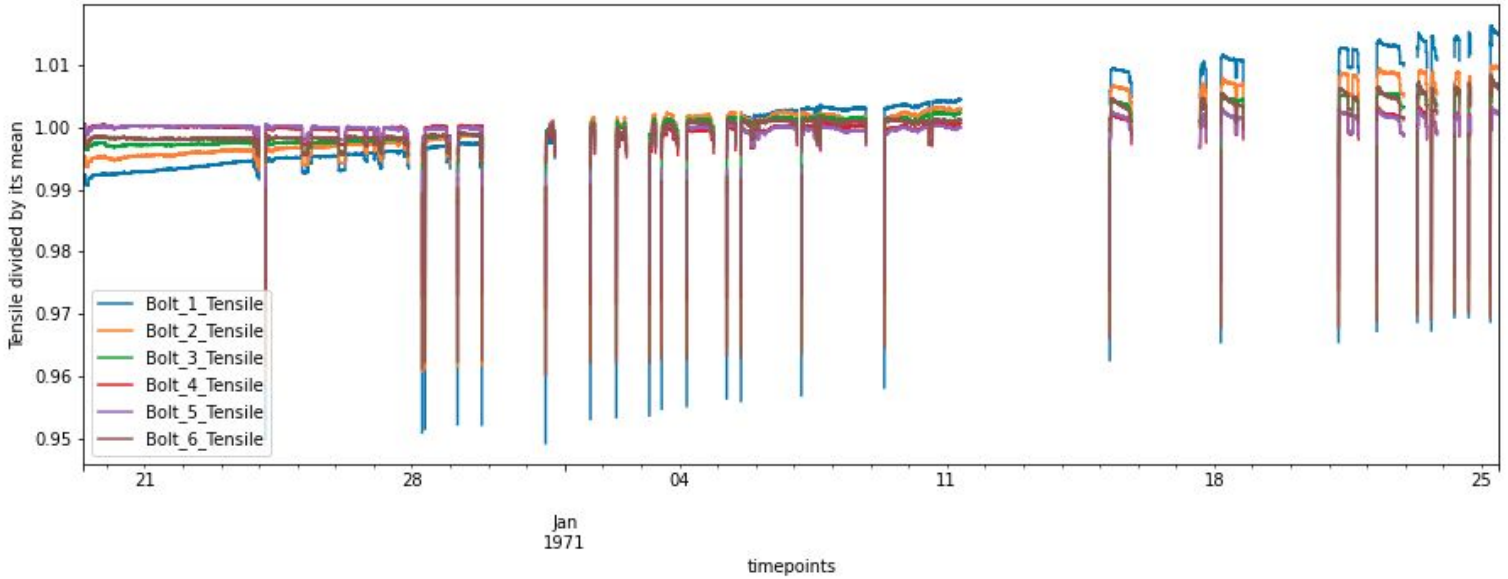
- Explored many different features, some useful, some useless (no significant improvement to the model or decreased performance)
  - Useless: Power triangle (reactive/active/real power), pressure difference
  - Useful: Seasonal features (day, month, “season” [defined according to our expert “vestlander” Simon]), “Stress deficit” (3000 - permissible stress), number of seconds in given operating condition
  - Seasonality features improved results by an order of 5-20x (depending on the bolt)

```
for response in linear_res.values():
    print(f"train mape: {response.train_mape}, test mape: {response.test_mape}")
```

```
train mape: 0.0003919099350238153, test mape: 0.0003925697137085556
train mape: 0.0004123787448502117, test mape: 0.0004129937346455044
train mape: 0.00027554781169530923, test mape: 0.0002754183199162816
train mape: 0.000300321533925371, test mape: 0.0003008249899423279 7
train mape: 0.00031577024809109893, test mape: 0.0003145418465709871
train mape: 0.00042260915041108995, test mape: 0.0004230343965192224
```

Data exploration - 2

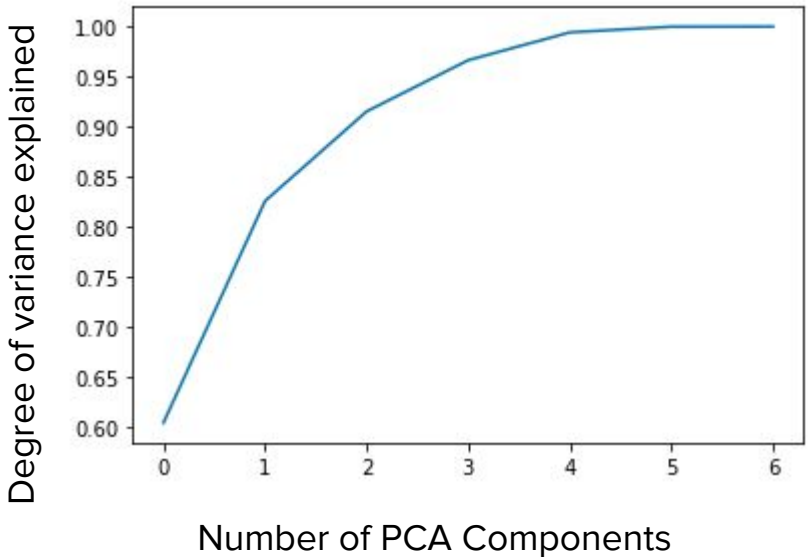
- Plotting of trends showed that the strain was increasing considerably over time for several bolts.
- Indicates that time is an important parameter for bolt strain!



Model considerations

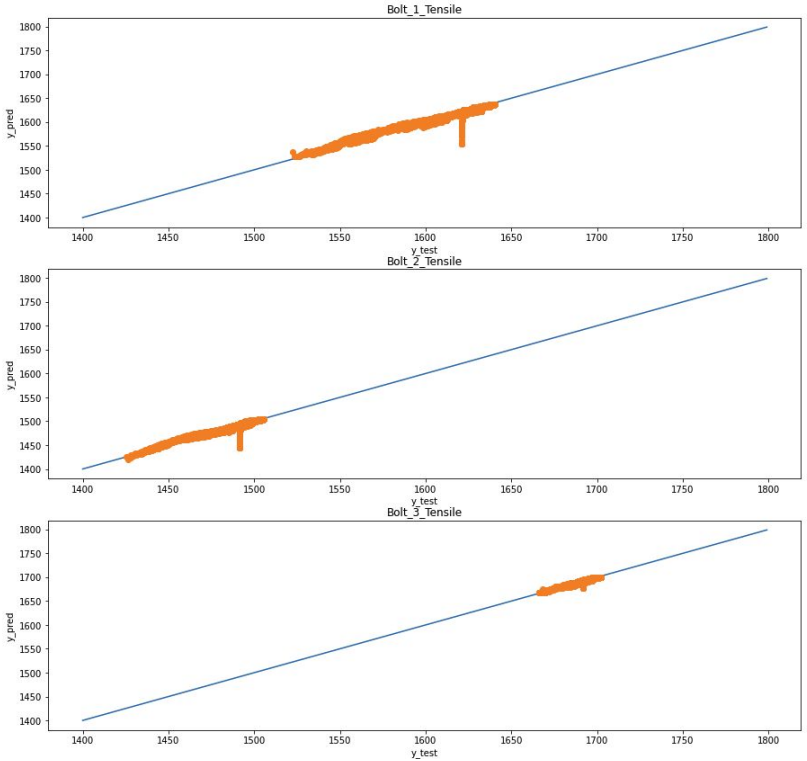
- A **multi-target regression** problem was considered initially but resulted in poor performance.
- Having a s**ingle predictive model for each bolt** achieved good accuracy and allowed us to capture the subtleties of each bolt.
- **Several feature reduction techniques** were considered at an early stage:
  - **PCA** (where number of components was selected according to those that explained the most variance)
  - Hand-crafted feature by looking at variables that showed the highest degree of correlation with target variables (i.e. strain values of each bolt)
- These feature engineering techniques were then paired with a predictive model to increase model performance.

Here 3 parameters were found to explain over 90% of dataset variance



Test metrics and gauging performance

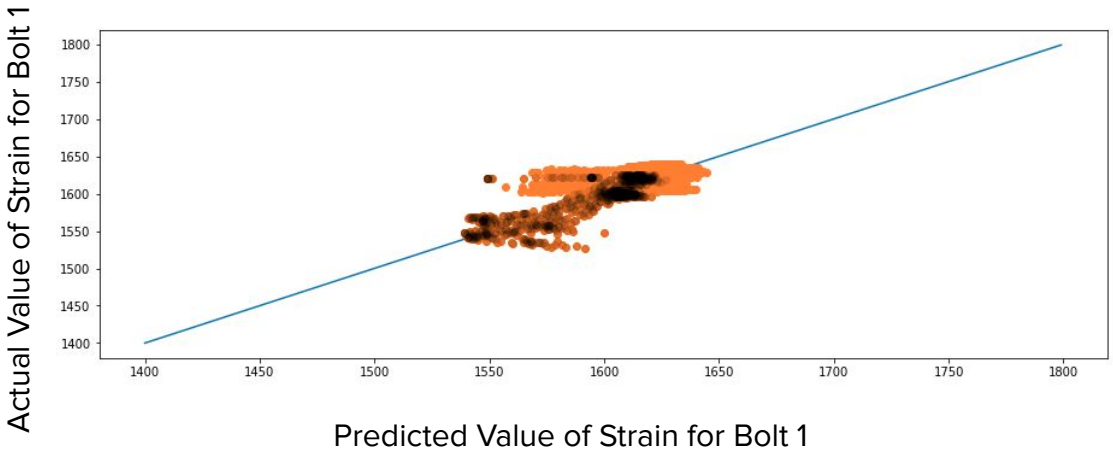
- Test size: 40%
- **“Mean Absolute percentage error”** used to gauge performance numerically since this is used as the evaluation metric in the hackathon. Is not heavily influenced by outliers.
- Plot true bolt tension vs. predicted bolt tension as a means of visual evaluation of results



Example plot of true bolt tension vs. predicted bolt tension

Examining model performance

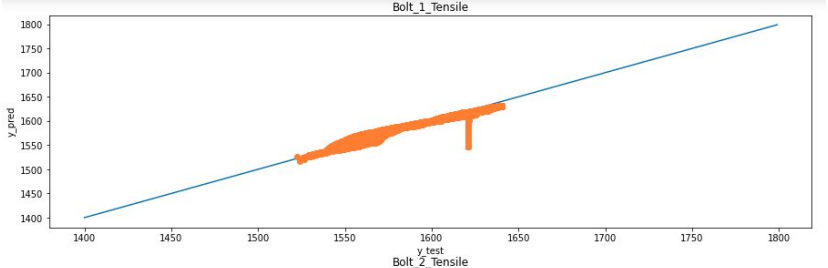
- Start-up periods had overall significantly poorer predictive performance compared to normal operation.
- It was considered to have separate models for different modes of operation but this was found to not significantly increase performance.
- It was presumed that this might be the case because:
  - There is another factor at play here which is not included in the data set;
  - Model readings for these regions are not reliable

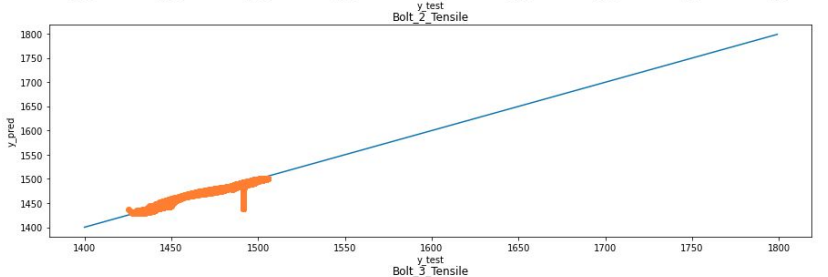


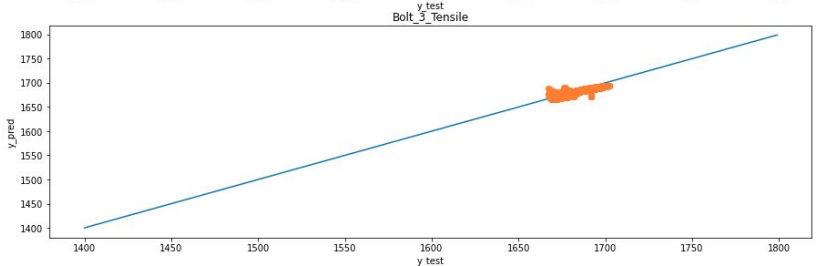
Legend
Orange: Normal operation
Black: Start-up regions

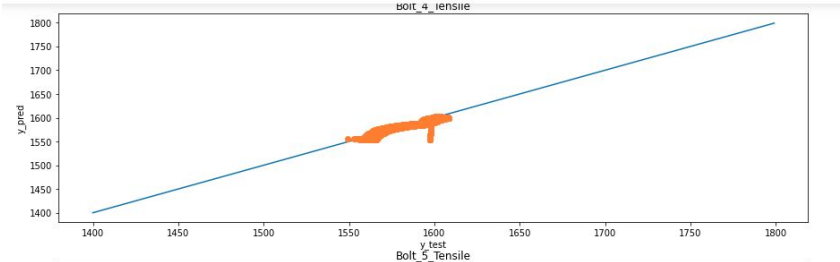
Overall Model Performance

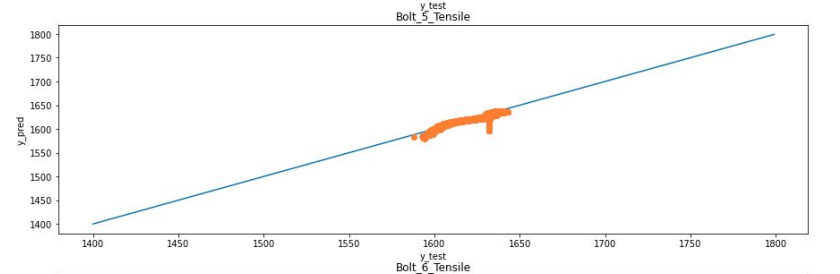
- Good model performance found with Linear Regression + additional features

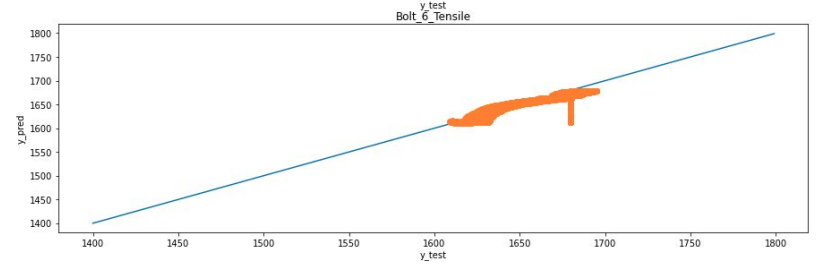






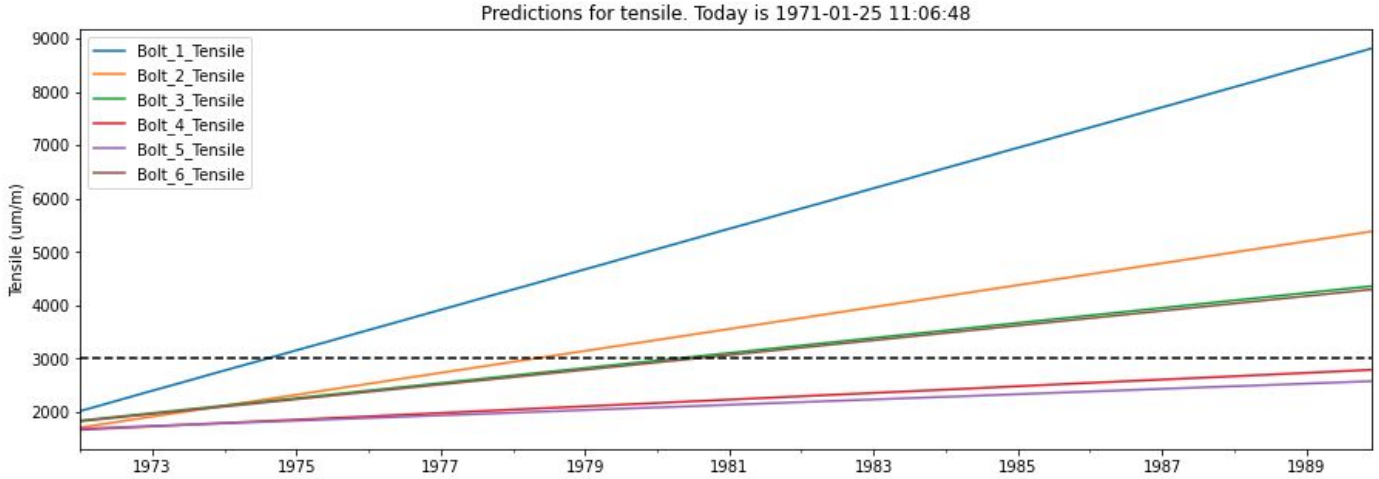






End of Life Prediction

- Using our model, we can predict the end-of-life for each bolt!
- The following bolts will fail in the near future (to the nearest month):
  - Bolt_1_Tensile fails at 1974-08-31 00:00:00
  - Bolt_2_Tensile fails at 1978-05-31 00:00:00
  - Bolt_3_Tensile fails at 1980-04-30 00:00:00
  - Bolt_6_Tensile fails at 1980-07-31 00:00:00



Putting it into production

Things to consider

- Retraining
  - Online retraining when model performance degrades. The onset of automatic retraining could be determined from changepoint detection, bayesian modelling for understanding when physics or dynamics change, or by monitoring model performance vs. actual target values over time.
  - Offline: Local training and upload of pickled model based on predetermined intervals for retraining
  - Online retraining requires a lot more work upfront, while oflline retraining requires manual work on a scheduled basis.
- Fallback when database is not reliable or data not trustworthy
  - When the database is not able to provide data to for predictions this needs to be highlighted in the dashboard to alert the user to not trust results in this region

From notebook to model:

- The code in the notebook should be rewritten to python modules in a github repository and tests should be written to verify the code works as expected.
- Data from a time series database is streamed to the model that makes predictions and write back them back to the time series database
- There are multiple alternatives for hosting the model, such as a docker container + kubernetes or Azure functions

Scalability & Transferability

Before scaling it is very important that the model and its predictions are validated offline and that Subject Matter Experts approve the results. Otherwise a lot of time will be wasted in scaling and putting the model into production when it is not valuable.

Considering that the other generators has partly different equipment, both in age and type, it is recommended to have a safety factor on the predicted values for the other generators without tensile measurements for bolts.