VSB Power Line Fault Detection

Using Machine Learning Classification Models to Detect Partial Discharge Faults in Covered Conductors in the Real Environment

Jeffrey Egan / February 2019

First, what is Kaggle?

- Kaggle is an online community of data scientists and machine learners.
- Kaggle allows users to:
  - Find and publish data sets.
  - Explore and build models in a web-based data-science environment.
  - Work with other data scientists and machine learning engineers.
  - Enter competitions to solve data science challenges.
- Kaggle got its start by offering machine learning competitions and now also offers a public data platform, a cloud-based workbench for data science, and short form Artificial Intelligence education.
- Google acquired Kaggle on 8 March 2017.

kaggle

dragonaur

Challenge: Partial Discharge Fault Detection

- Overhead power lines run for hundreds of miles to supply power to cities. These great distances make it expensive to manually inspect the lines for damage that doesn't immediately lead to a power outage (like tree branches falling on lines).
- These modes of damage lead to a phenomenon known as partial discharge — an electrical discharge which does not bridge the electrodes between an insulation system completely.
- Partial discharges slowly damage the power line, so left unrepaired they will eventually lead to a power outage or start a fire.
- The challenge is to detect partial discharge patterns in signals acquired from these power lines with a new meter designed at the ENET Centre at VŠB. Effective classifiers using this data will make it possible to continuously monitor power lines for faults – helping to reduce maintenance costs and prevent power outages.

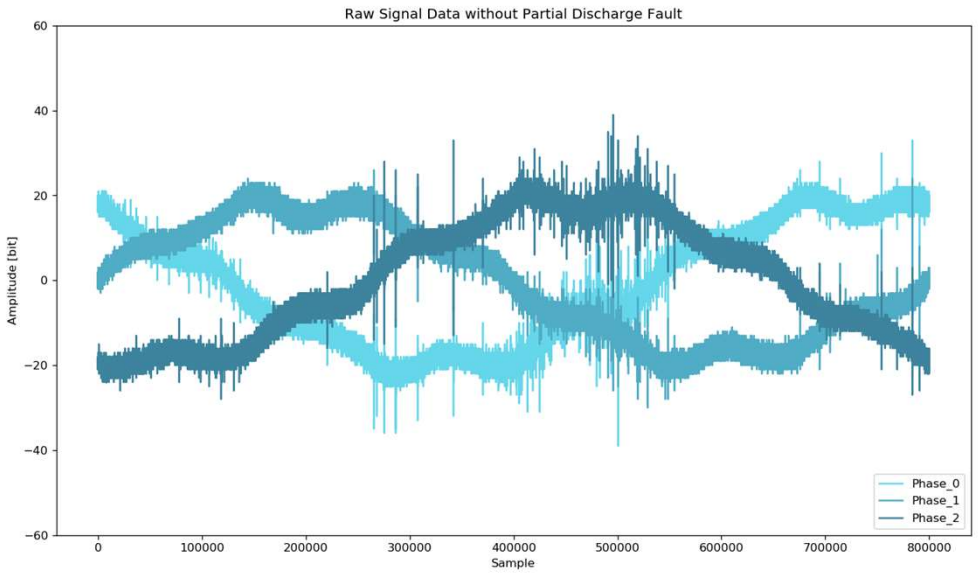
dragonaur

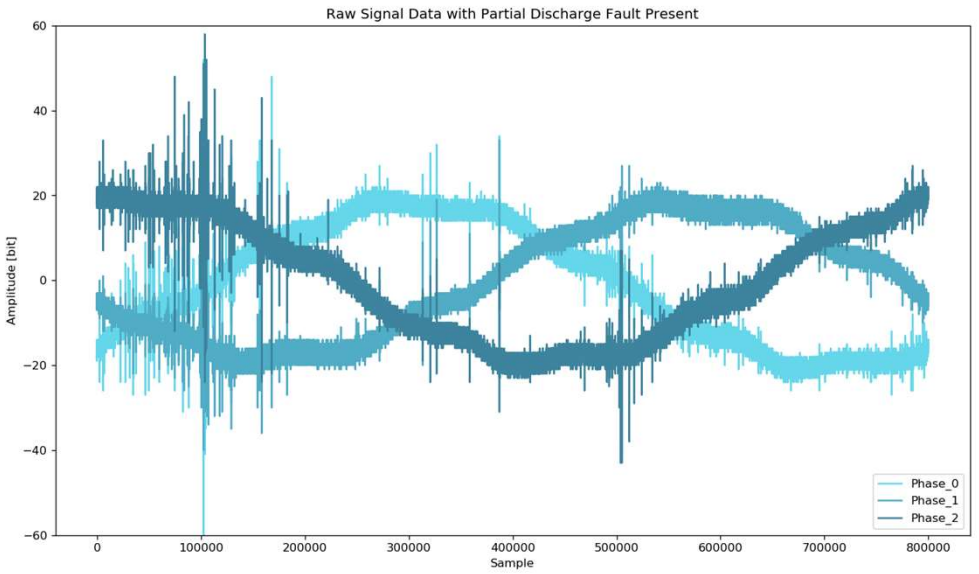
Centrum ENET

Examination Reveals Highly Unbalanced Data

- All Provided Data: Labeled & Unlabeled
  - Labeled Training Set: 2,904 Measurements = 8,712 Signals (only 30% of data is labeled)
  - Unlabeled Test Set: Perform Binary Classification: 6,779 Measurements = 20,337 Signals
- Occurrence of Faults and Non-Faults in Labeled Data
  - 8187 Signals without Fault, 213 Signals with Fault (only 2.5% of labeled signals have faults!)
- Grouping of Signal Faults per Measurement
  - Measurements with no faults present: 2710 (93.3% of measurements exhibit no fault)
  - Measurements with fault present in only 1 phase: 19 (9.7% of cases where faults present)
  - Measurements with faults present in 2 of 3 phases: 19 (9.7% of cases where faults present)
  - Measurements with faults present in all 3 phases: 156 (80.4% of cases where faults present)

Sample Measurements with 3 Phase Signals





Each signal (phase) contains 800,000 measurements of a power line's voltage, taken over 20 milliseconds.

For signal processing, that means each signal data set has a sample rate of 40 million samples per second.

The underlying electric grid operates at 50Hz, this means each signal covers a single complete grid cycle.

dragonaur

Centrum ENET

Detecting Partial Discharges is Difficult

PD Pattern Observed alongside other Noise Sources

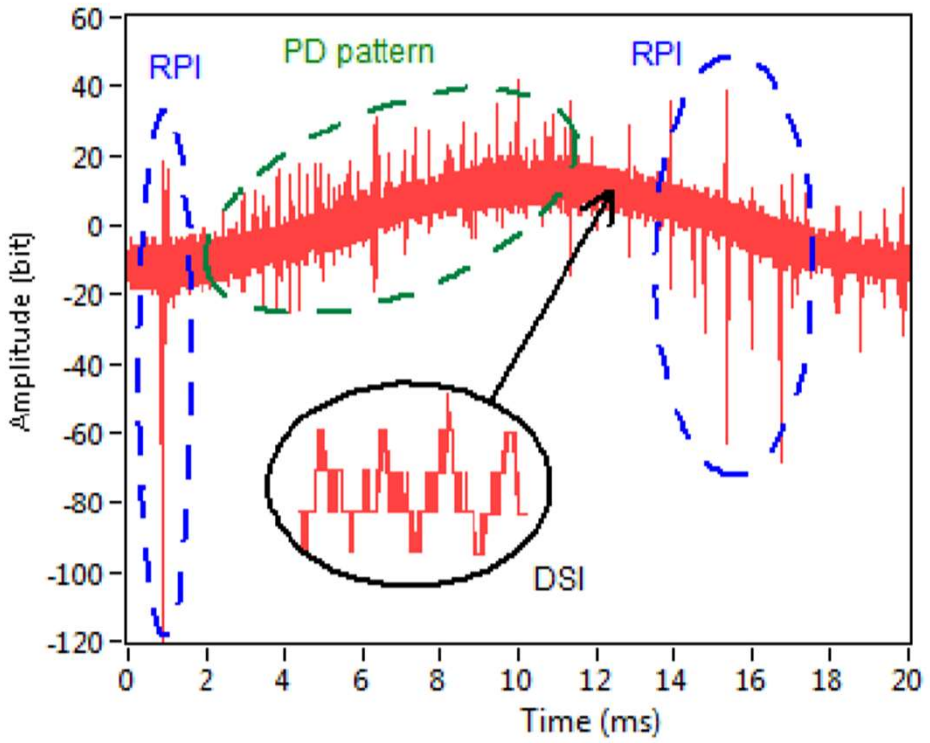


Figure from Reference [1]

Many other sources of noise could be falsely attributed to partial discharge.

- Ambient and Amplifier Noise
- PD: Partial Discharge Fault Pattern
- DSI: Discrete Spectral Interference
  - E.g. Radio Emissions, Power Electronics
- RPI: Random Pulses Interference
  - Lightning, Switching Operations, Carona

dragonaur

Centrum ENET

Identify and Cancel Carona Discharge Peaks

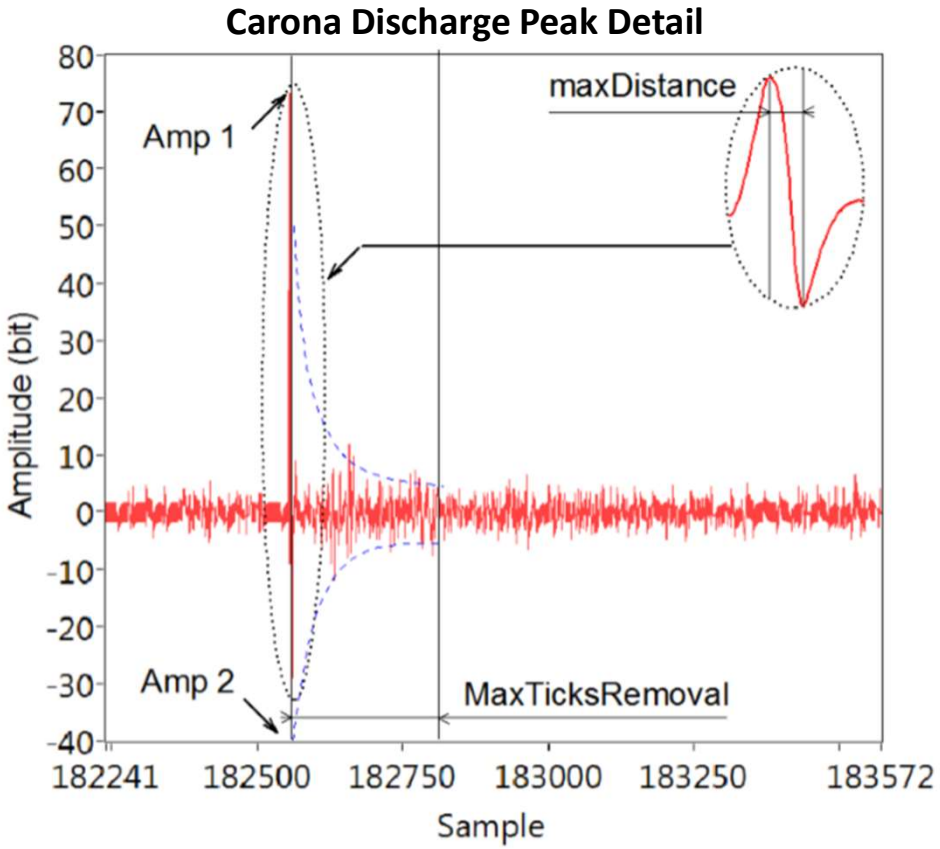


Figure from Reference [1]

- A corona discharge is an electrical discharge brought on by the ionization of a fluid such as air surrounding a conductor that is electrically charged.
- Spontaneous corona discharges occur naturally in high-voltage systems unless care is taken to limit the electric field strength.
- A corona will occur when the strength of the electric field (potential gradient) around a conductor is high enough to form a conductive region, but not high enough to cause electrical breakdown or arcing to nearby objects.

dragonaur

Centrum ENET

Focus on Portions of the Sinusoid with Rising Amplitude for Improved PD Detection

Sections of Sinusoidal Shape (1,2) with the Statistically Highest Occurrence of PD pulses

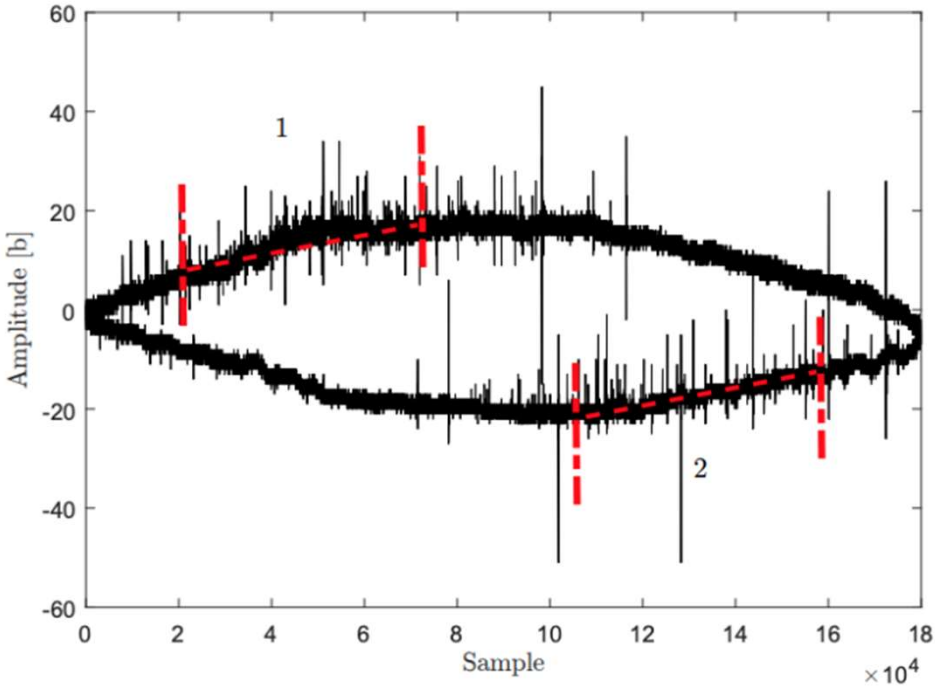


Figure from Reference [2]

- One of the most common fundamental features of the PD pattern is that it occurs on the rising amplitude edges of the sinusoidal curve.
- For future iterations of feature extraction, this allows us to omit the rest of the signal.
- This change should improve the efficiency of processing signal data as well improve the model's ability to detect true faults and avoid false positives.

dragonaur

Centrum ENET

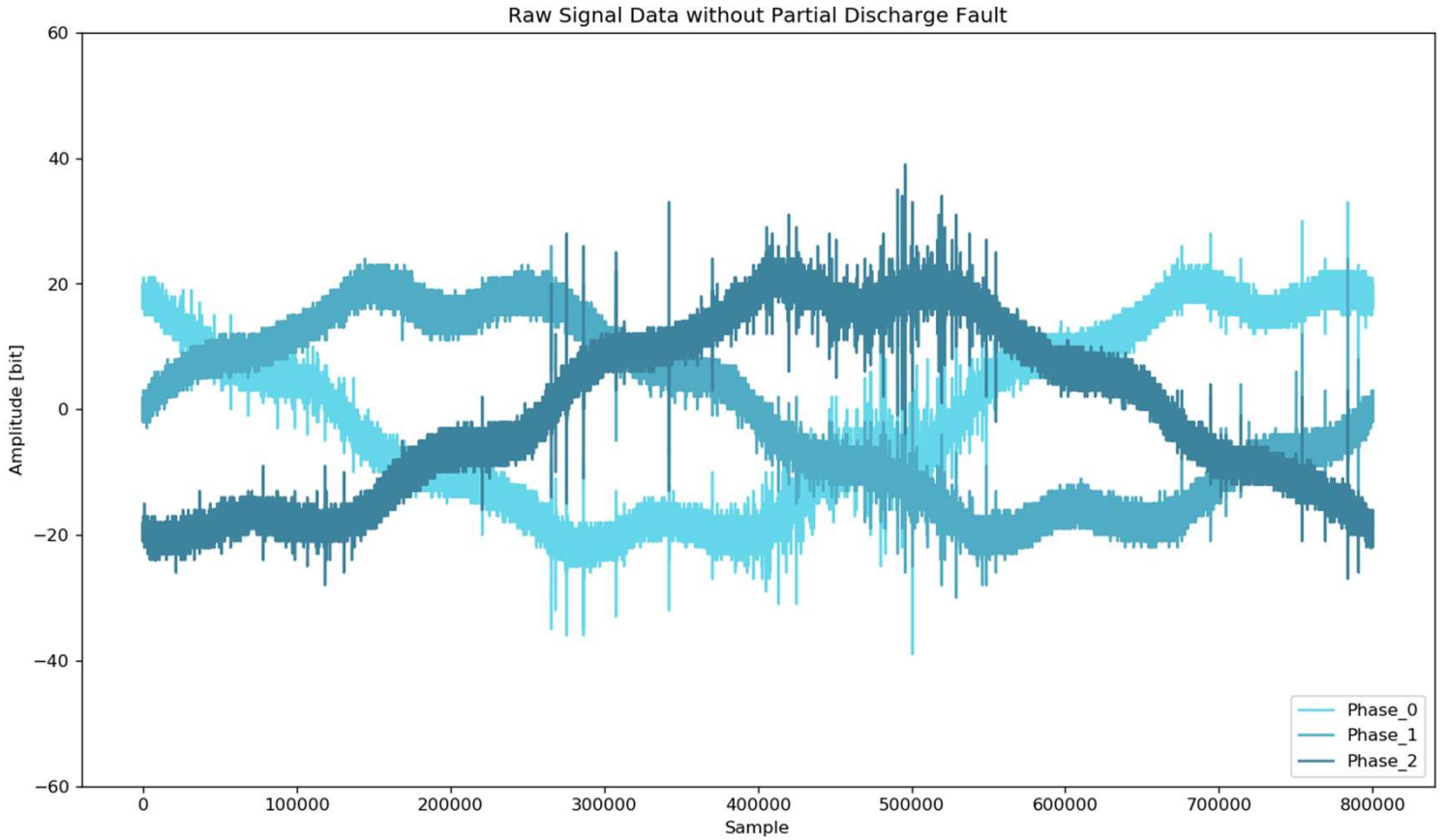
Signal Processing Approach

- Discrete Wavelet Transform to Remove Spectral Interference and Ambient Noise
  - Daubechies 4 Wavelet (db4)
- Fit a Sinusoidal Function to the DWT Signal and Detect the PD-Probable Region
- Use an Nth Discrete Difference to De-Trend the Function
- Perform Peak Detection and Subsequently Cancel of False Peaks
  - Peak Detection using Fixed Amplitude Threshold
  - Cancel High Amplitude Peaks, and Peaks Attributed to Carona Discharge

dragonaur

Centrum ENET

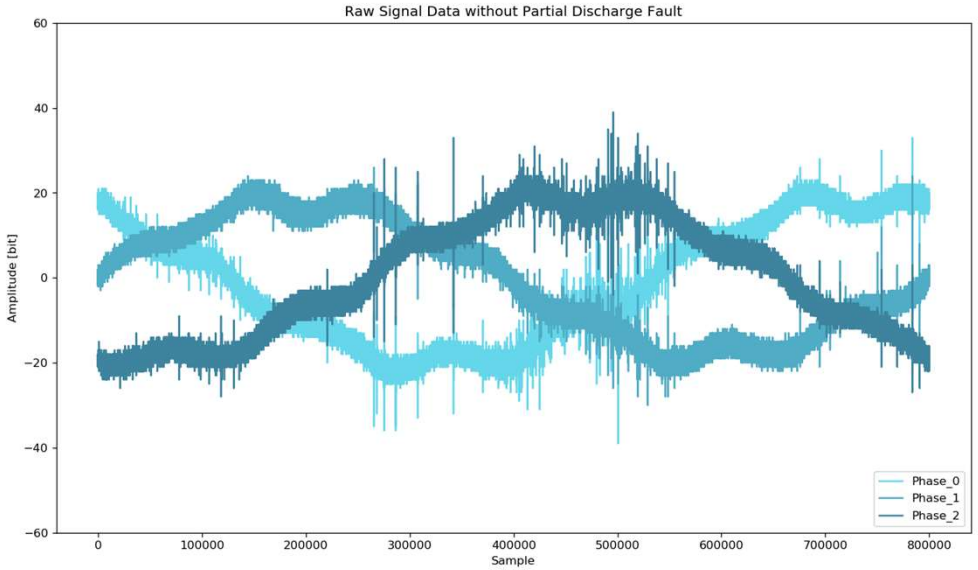
Examining the Raw Signal Data

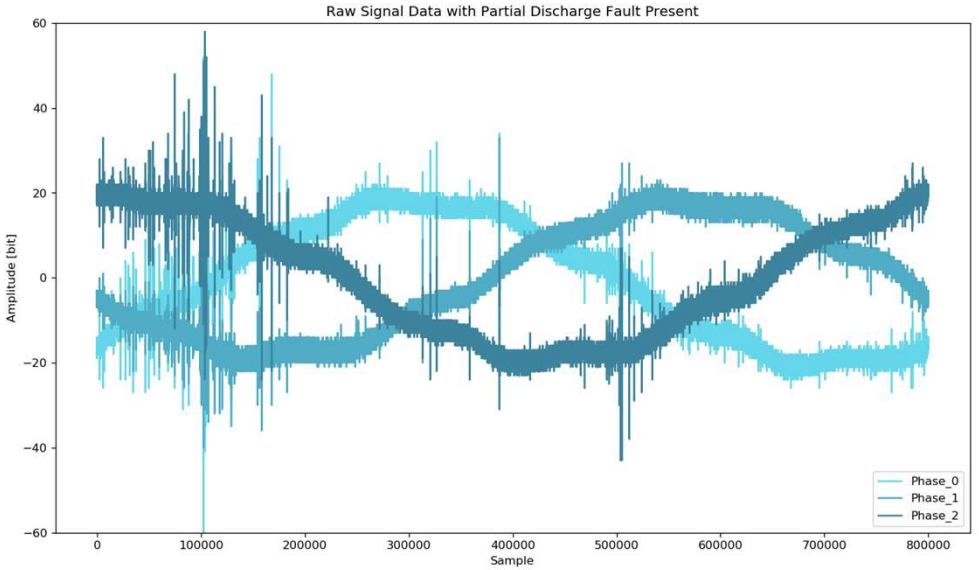


dragonaur

Centrum ENET

Examining the Raw Signal Data

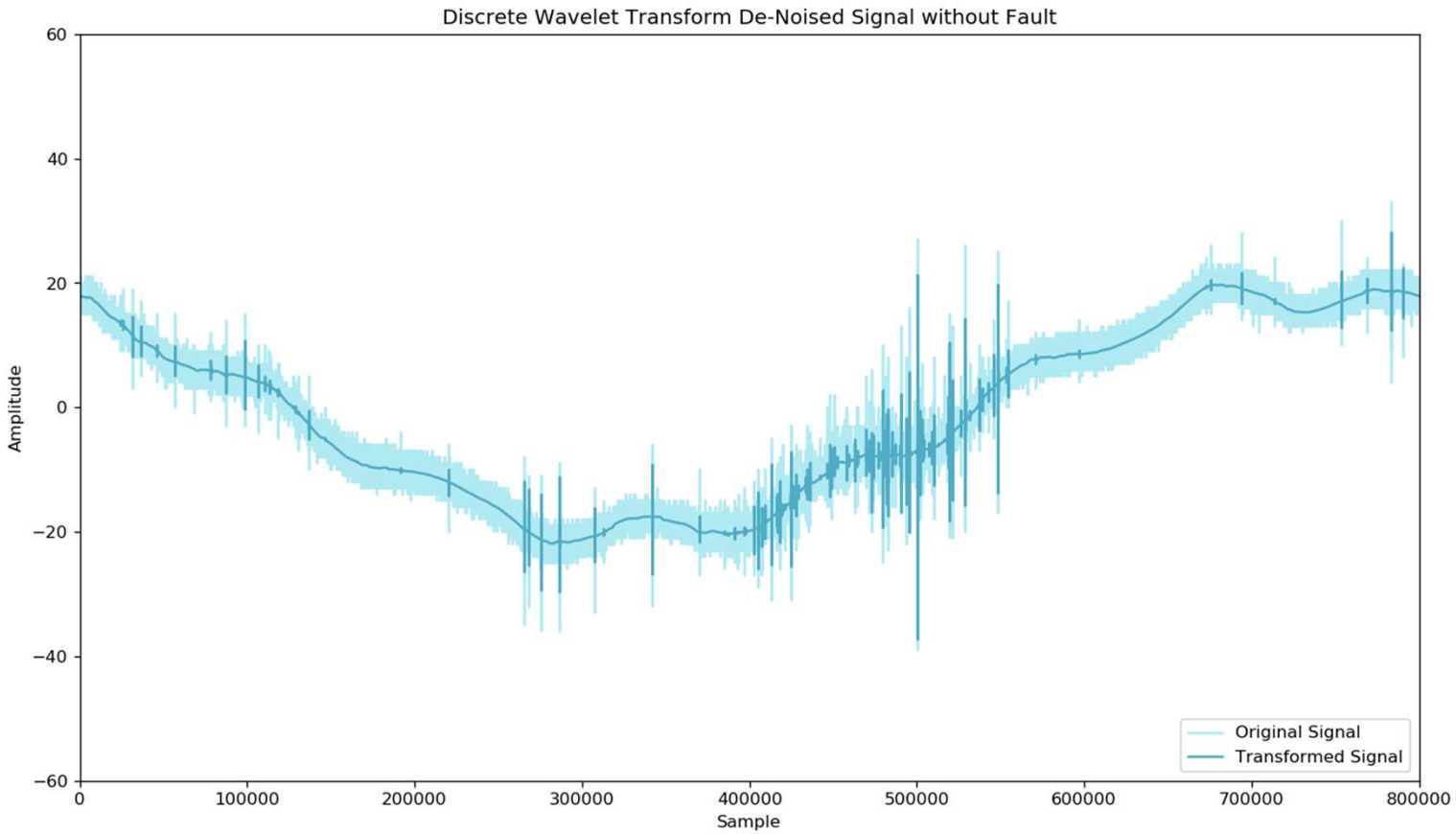




dragonaur

Centrum ENET

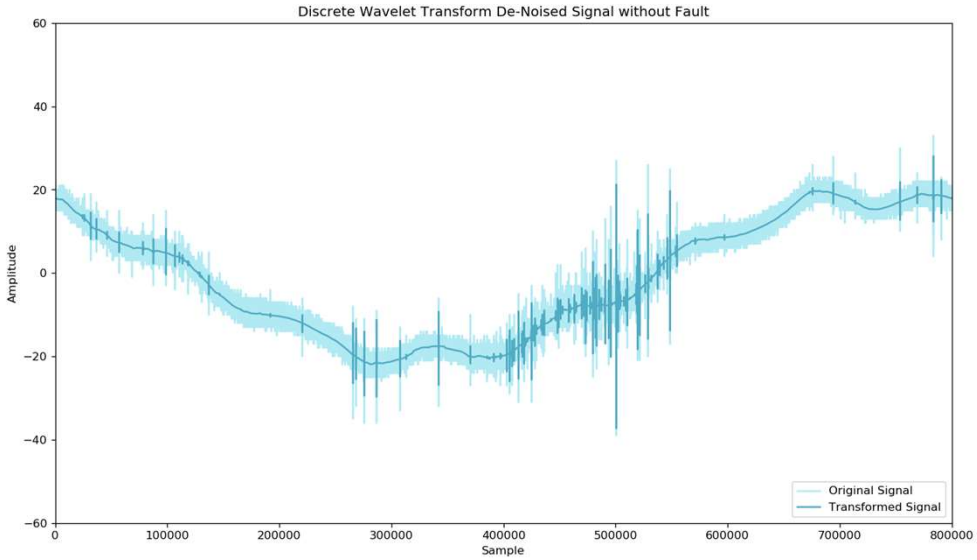
Remove DSI & Ambient Noise with DWT

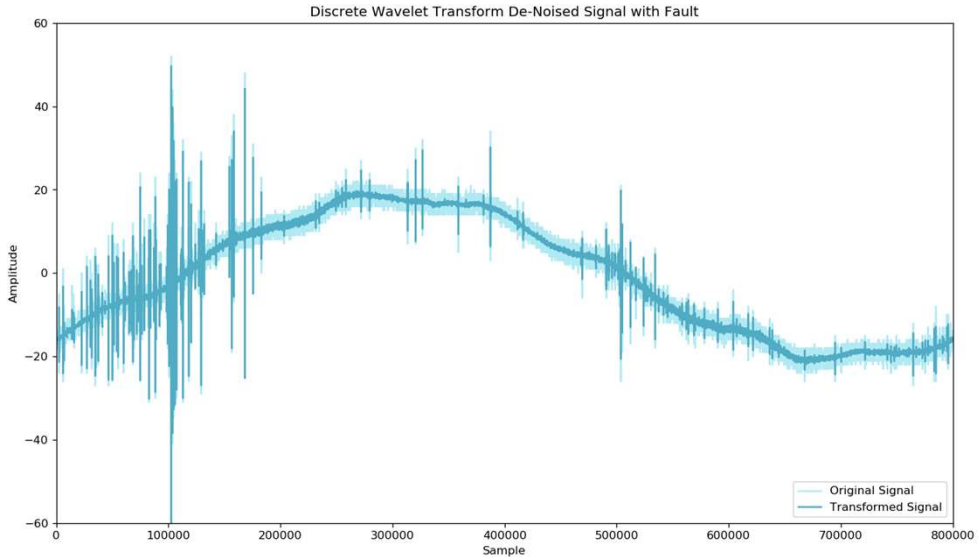


dragonaur

Centrum ENET

Remove DSI & Ambient Noise with DWT

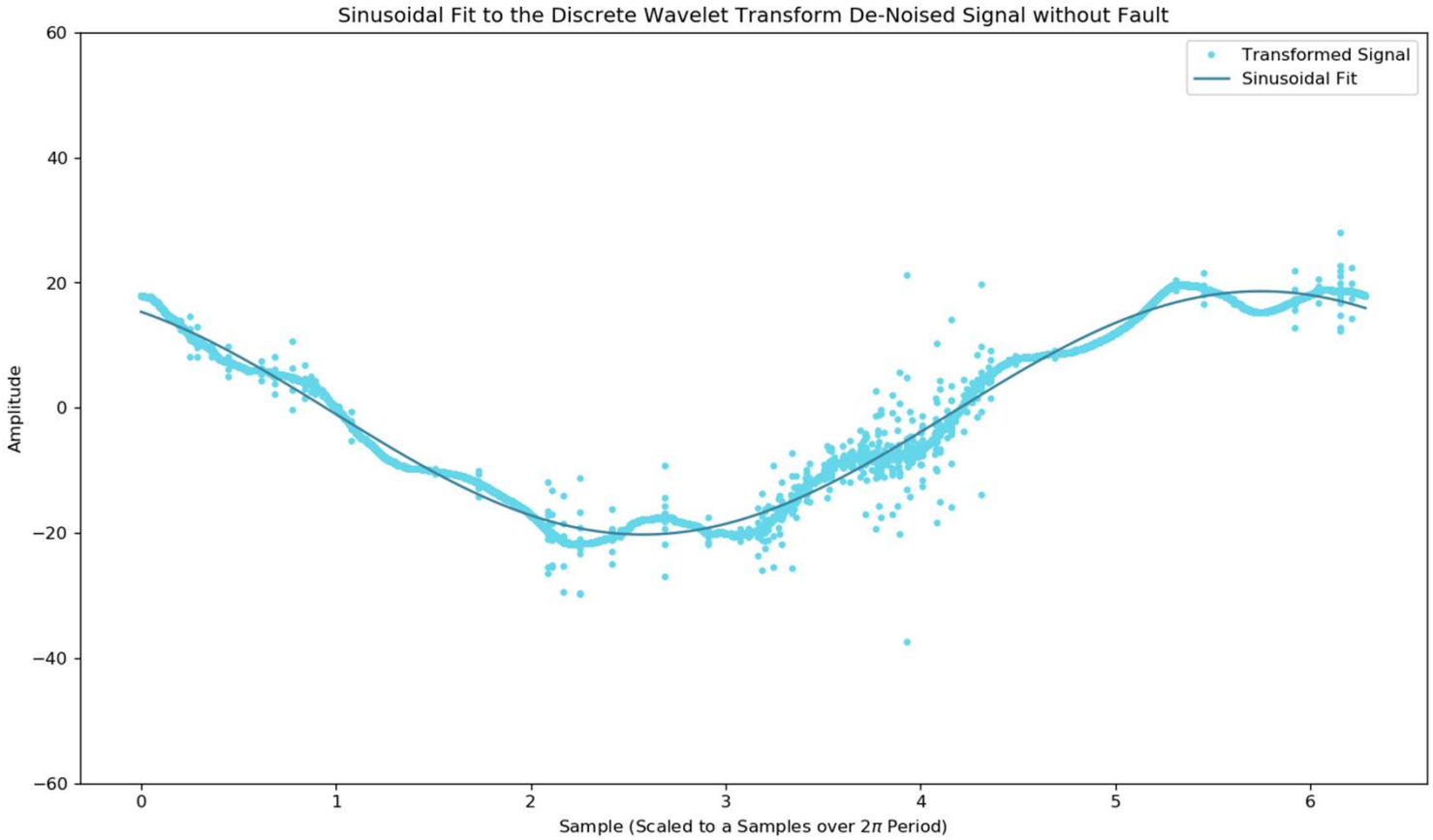




dragonaur

Centrum ENET

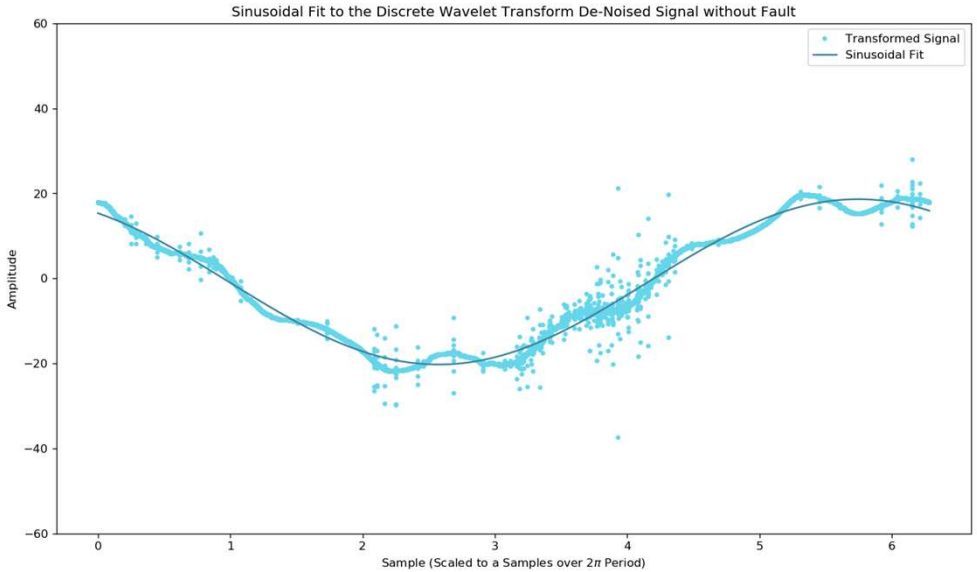
Fit a Sinusoidal Function to the DWT Signal

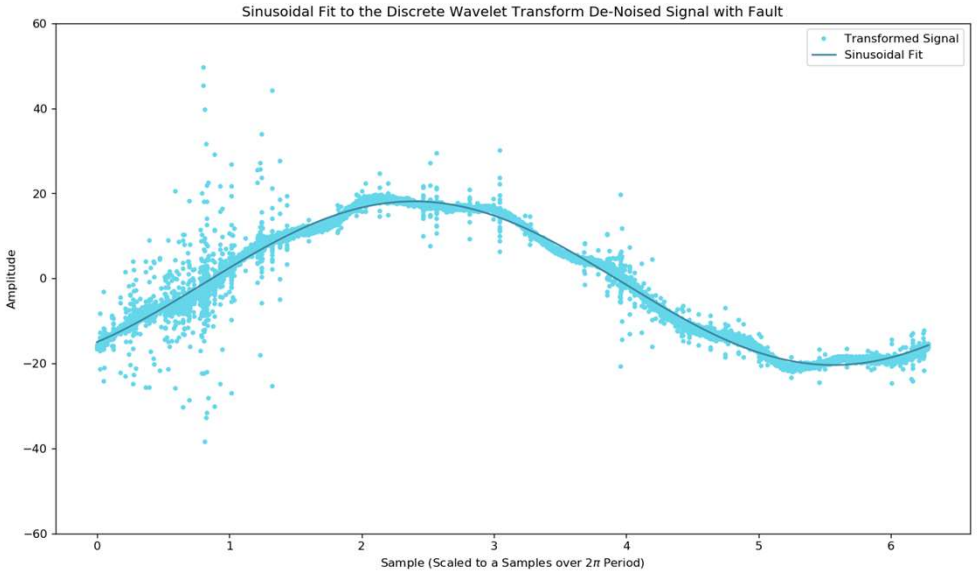


dragonaur

Centrum ENET

Fit a Sinusoidal Function to the DWT Signal

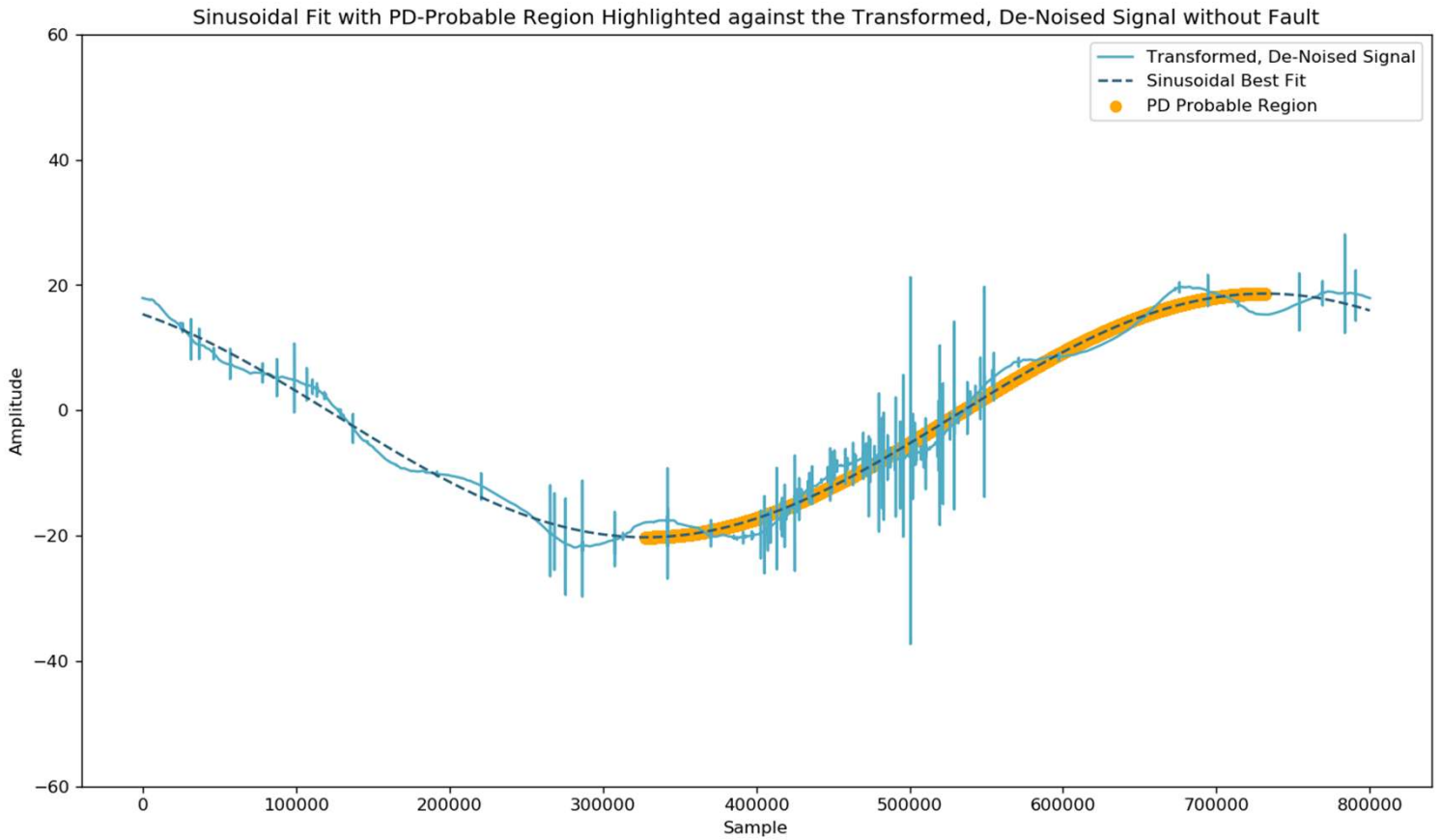




dragonaur

Centrum ENET

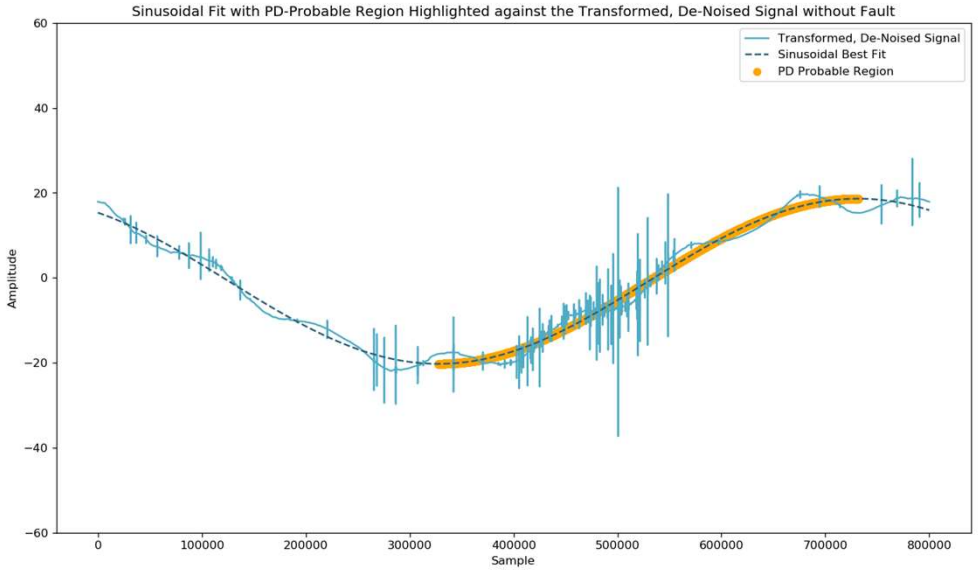
Use Sinusoid to Identify PD-Probable Region

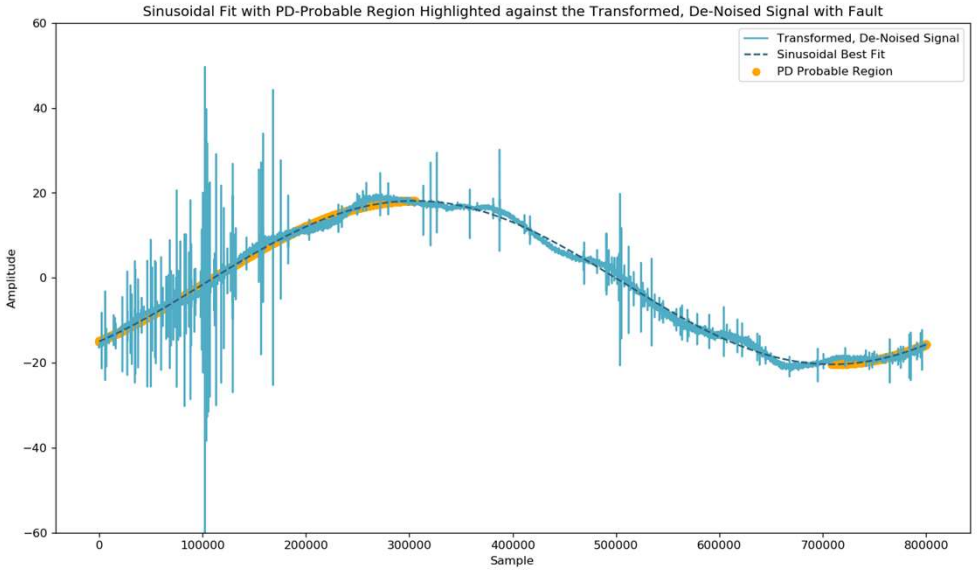


dragonaur

Centrum ENET

Use Sinusoid to Identify PD-Probable Region

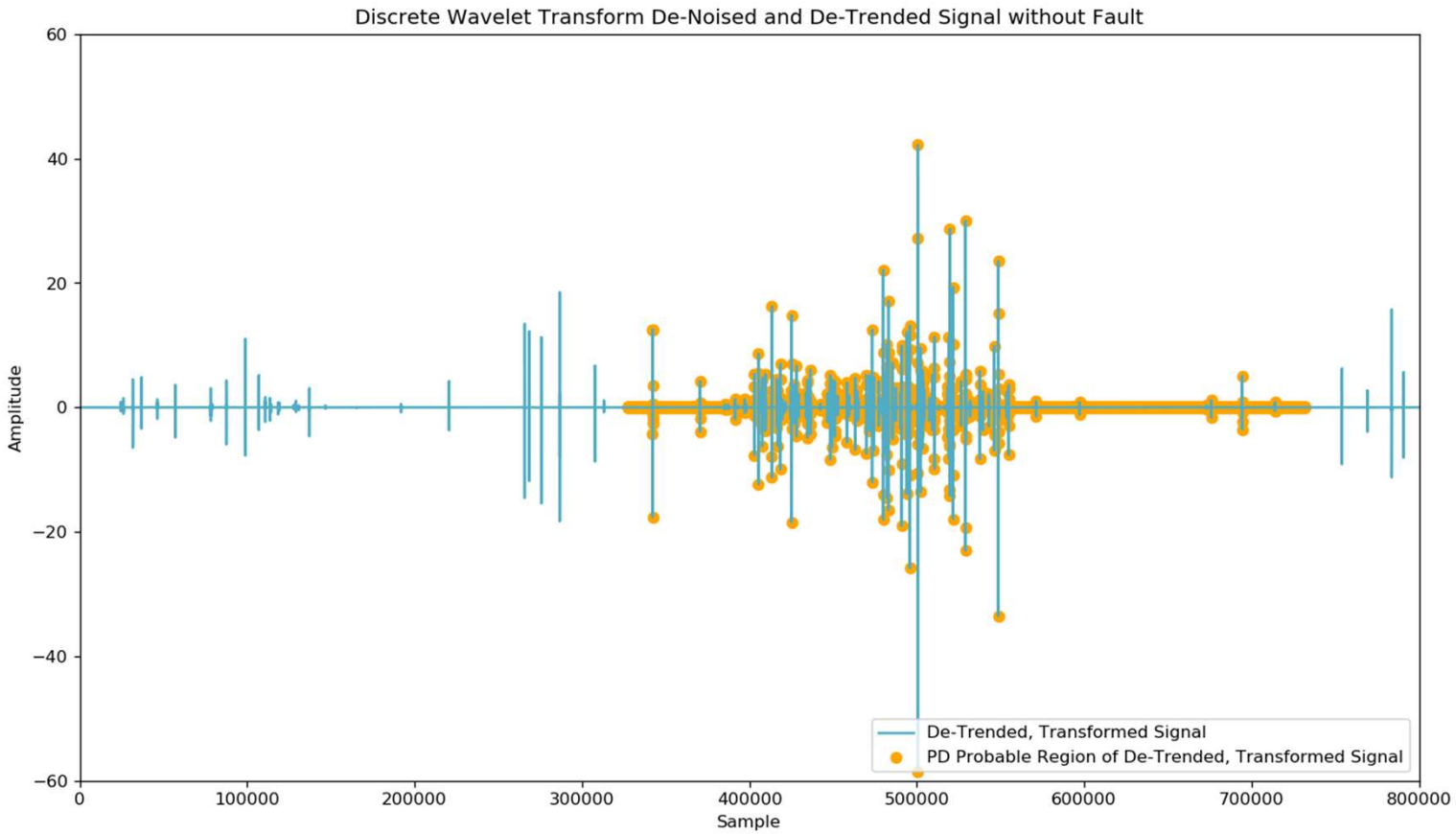




dragonaur

Centrum ENET

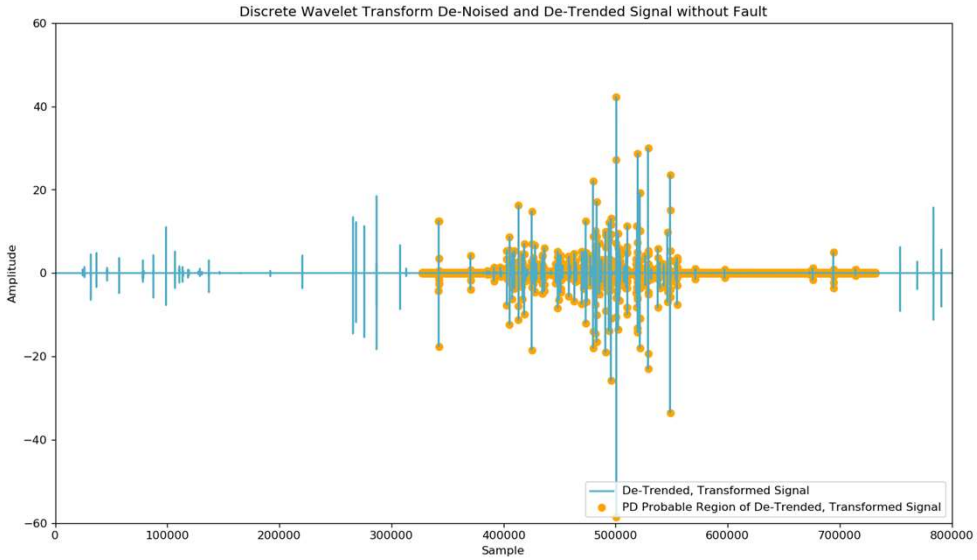
Detrend Signal to Remove Sinusoidal Element

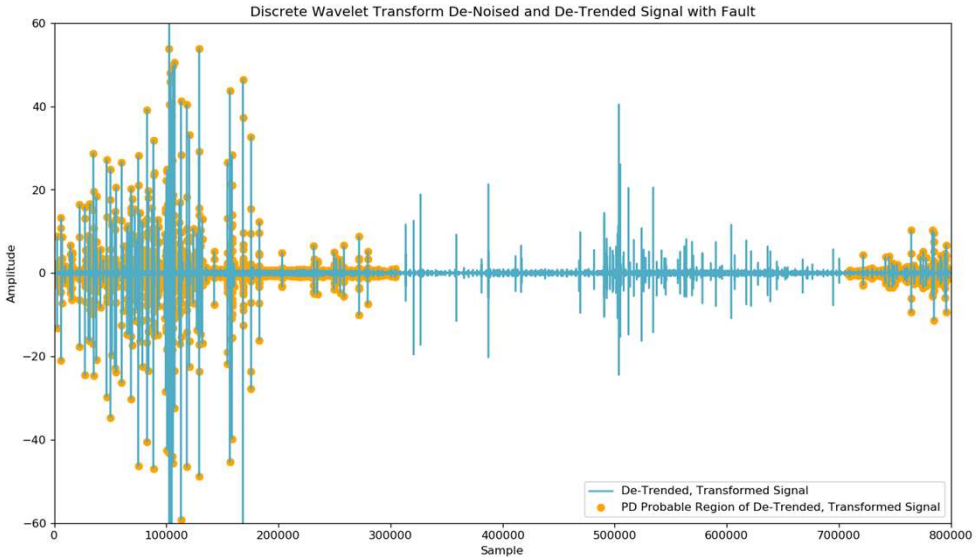


dragonaur

Centrum ENET

Detrend Signal to Remove Sinusoidal Element

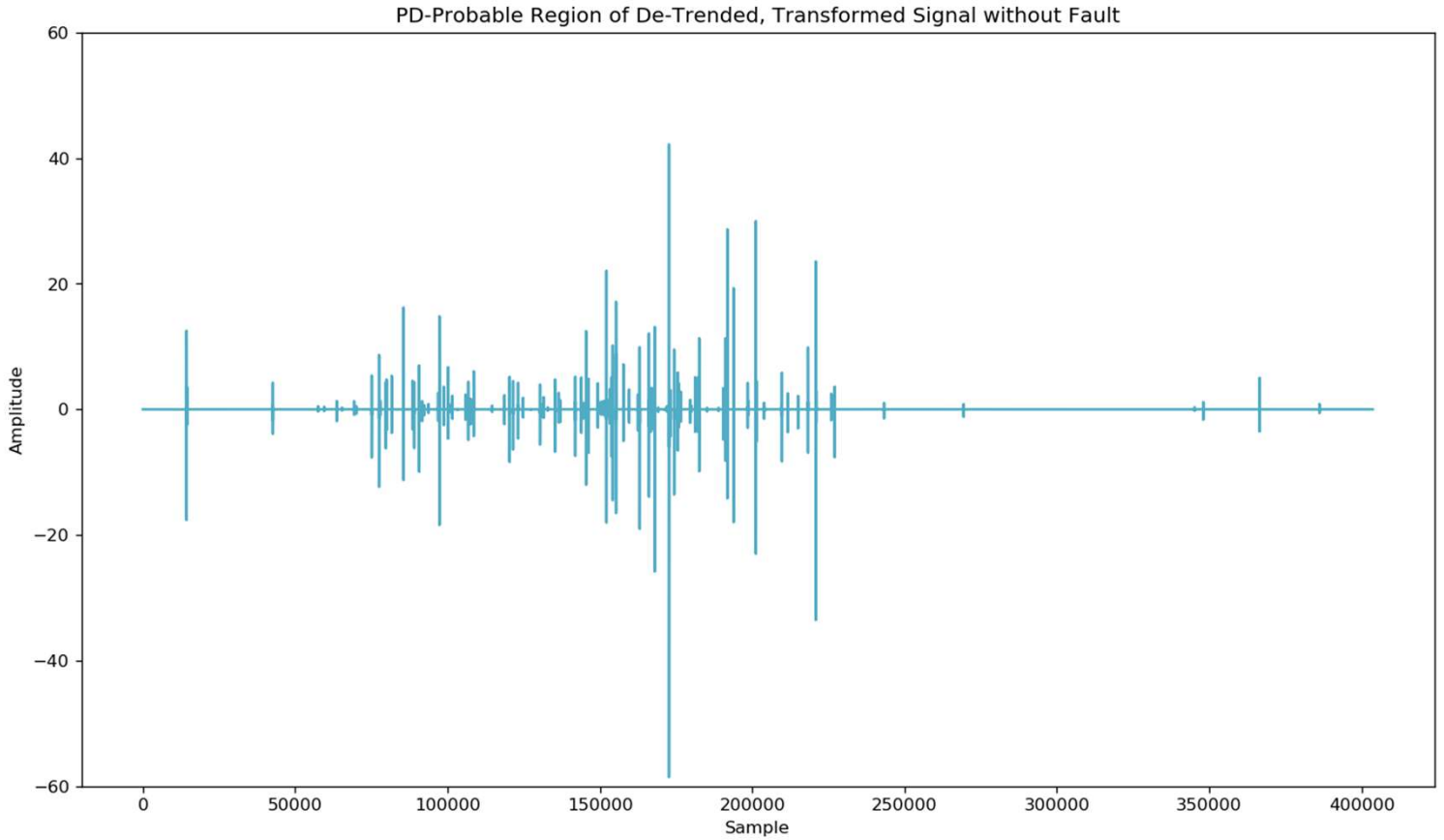




dragonaur

Centrum ENET

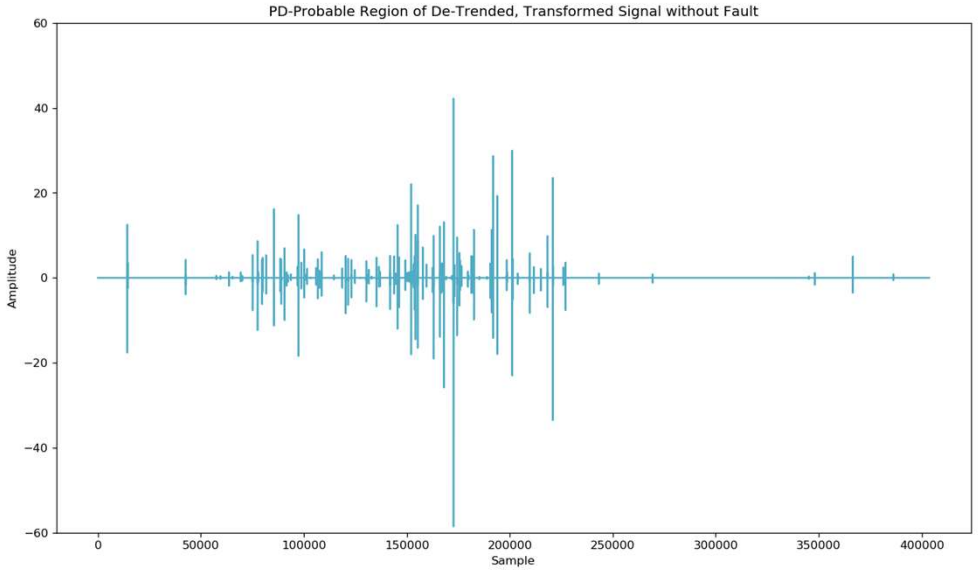
Trim to PD-Probable Region of Detrended Data

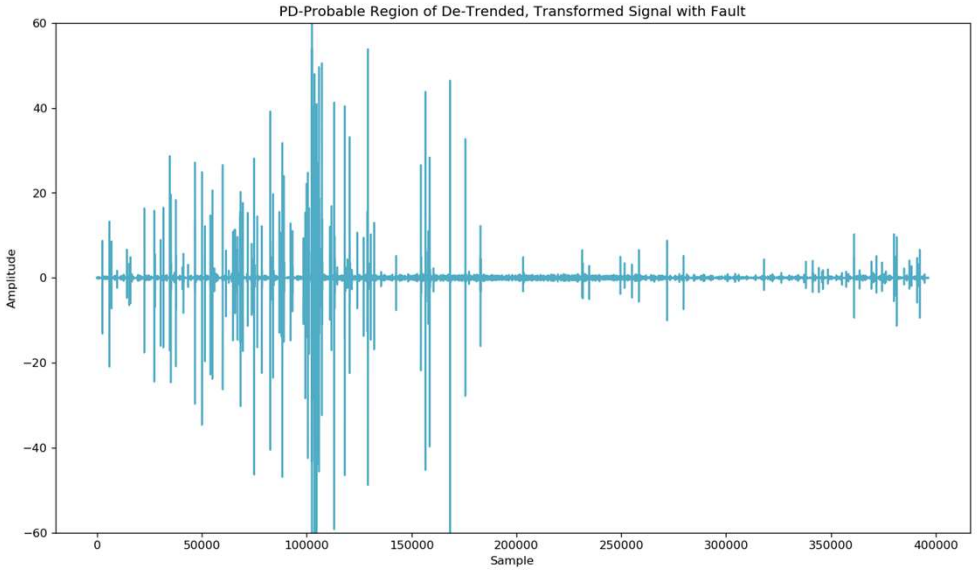


dragonaur

Centrum ENET

Trim to PD-Probable Region of Detrended Data

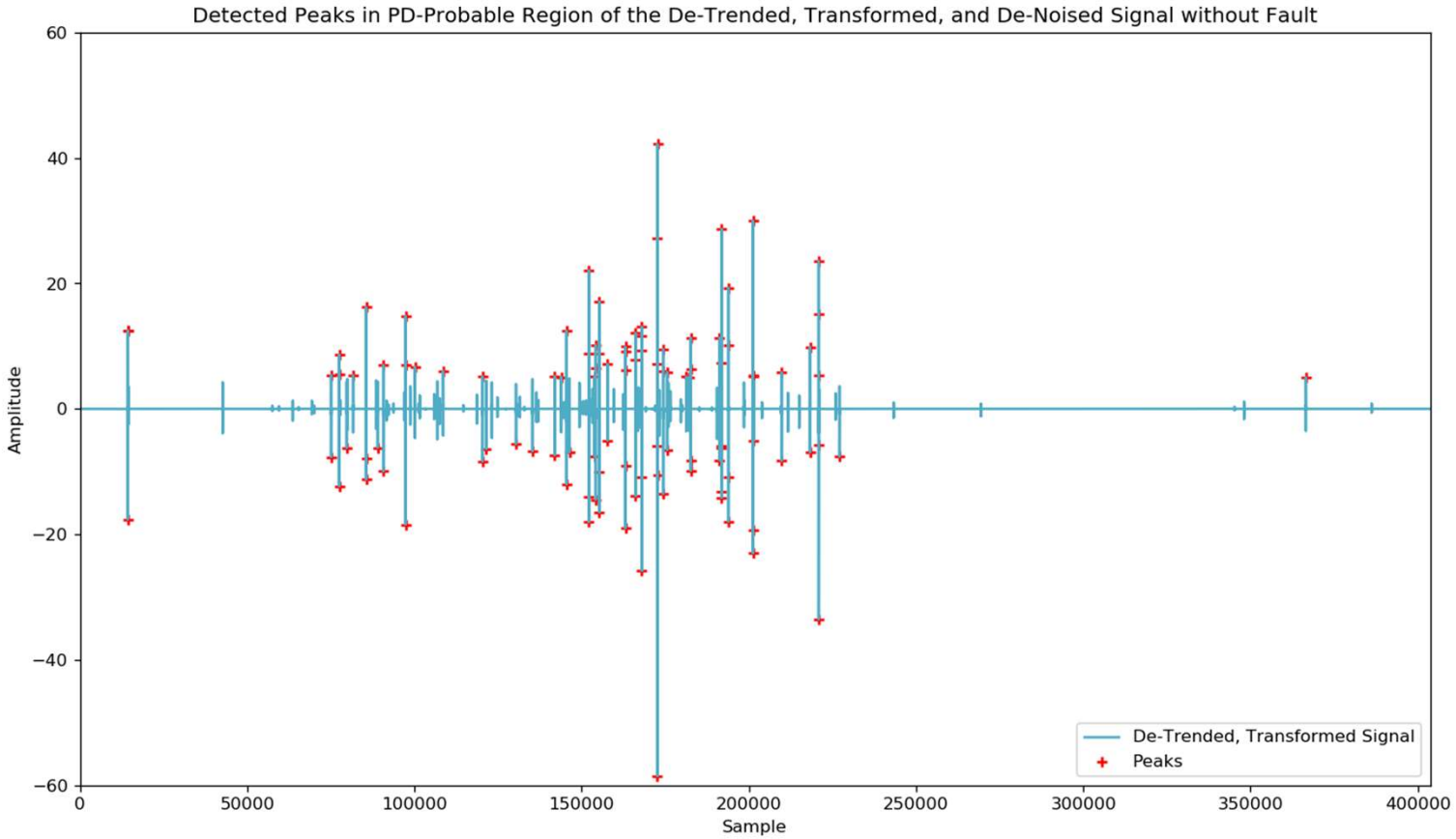




dragonaur

Centrum ENET

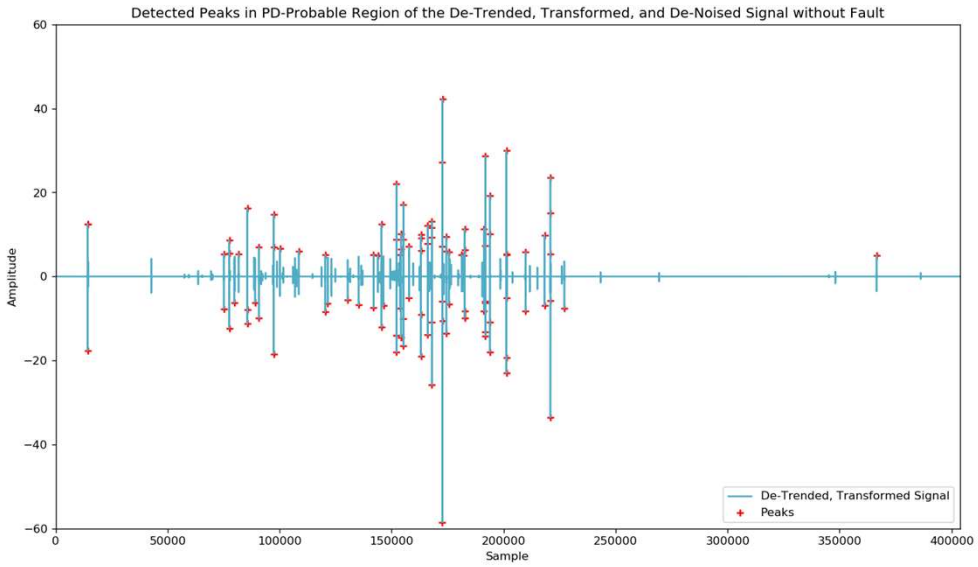
Run Peak Detection on PD-Probable Region

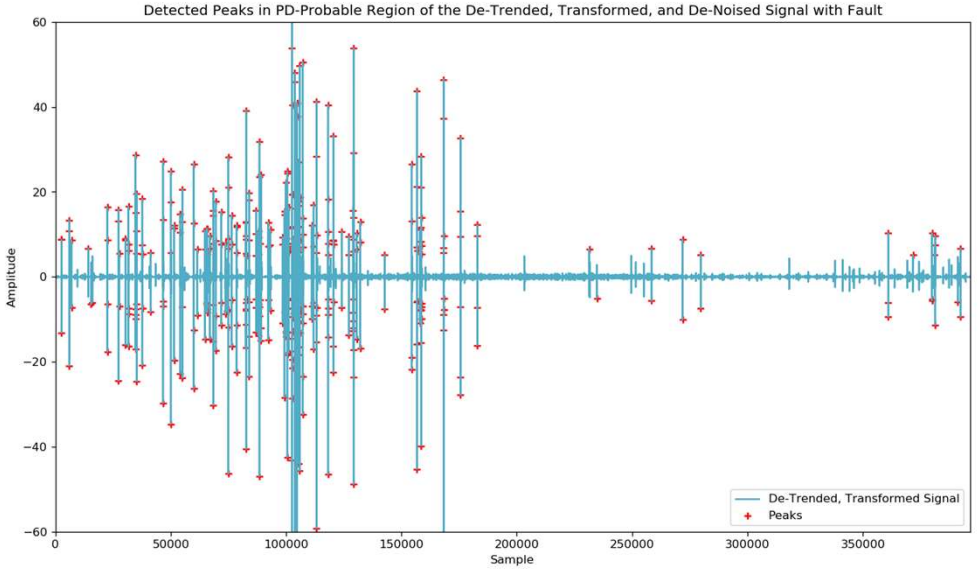


dragonaur

Centrum ENET

Run Peak Detection on PD-Probable Region

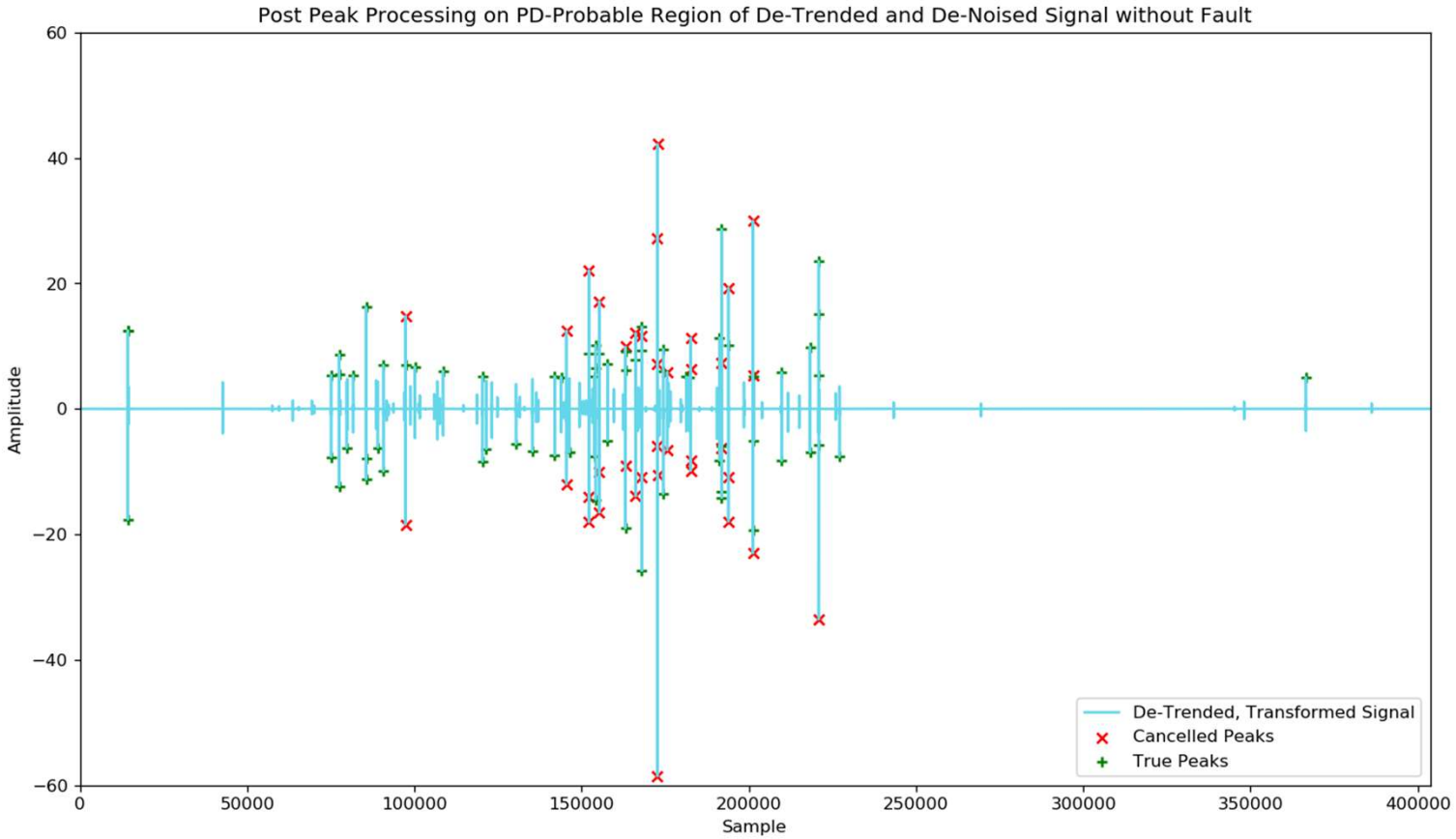




dragonaur

Centrum ENET

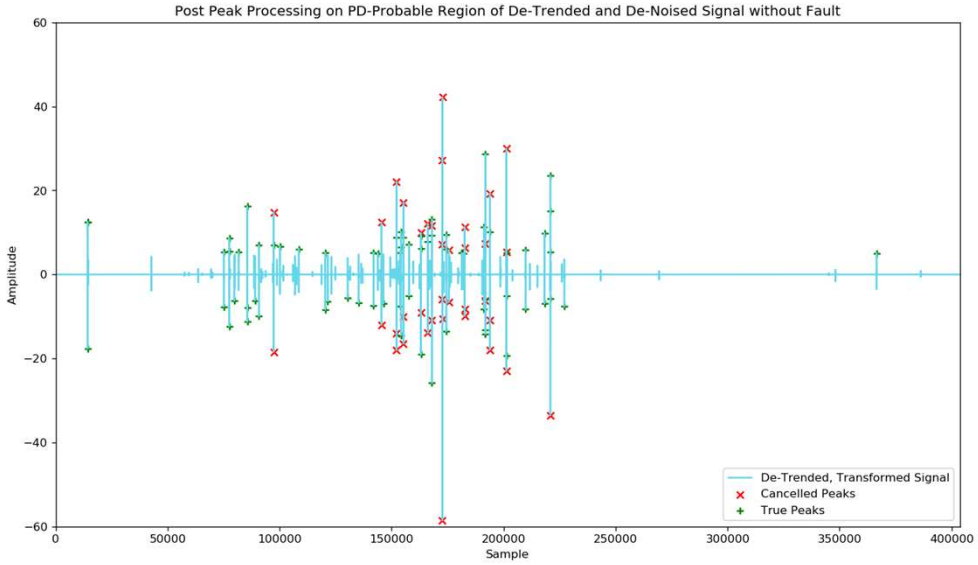
Cancel False Peaks Before Processing Signals

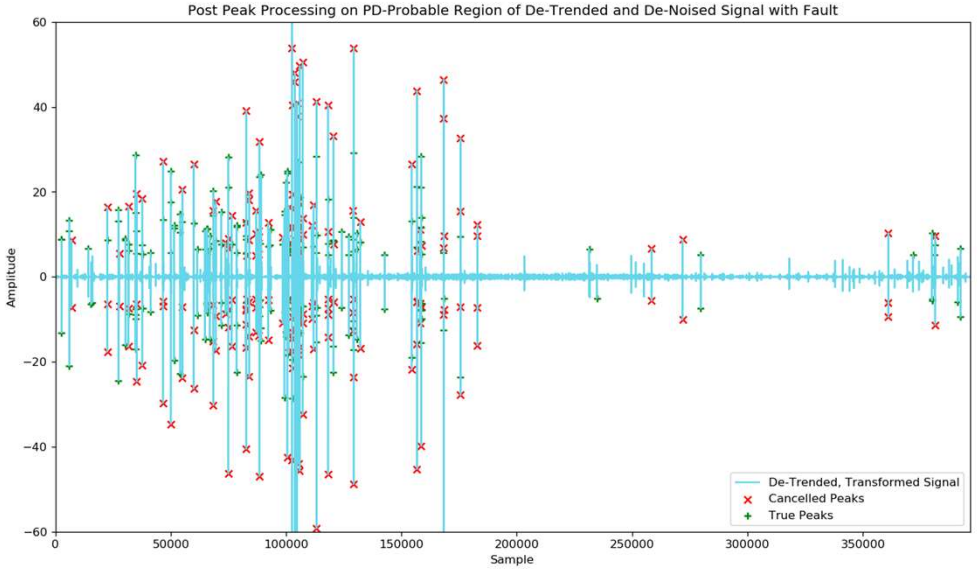


dragonaur

Centrum ENET

Cancel False Peaks Before Processing Signals





dragonaur

Centrum ENET

Perform Feature Extraction on Signal

Process the PD-Probable Region of the De-Trended, De-Noised Signal and Build a Feature Table to be used as Input to Machine Learning Models.

- 5^{th} Percentile
- 25^{th} Percentile
- 75^{th} Percentile
- 95^{th} Percentile
- Median
- Mean
- Standard Deviation
- Variance
- Root Mean Square
- Entropy
- Number of Zero Crossings
- Number of Mean Crossings
- Minimum Peak Height
- Maximum Peak Height
- Minimum Peak Width
- Maximum Peak Width
- Number of Detected Peaks
- Number of True Peaks

dragonaur

Centrum ENET

Feature Data Exploration

To Do

Add Once Extraction Complete

dragonaur

Centrum ENET

Prototype Binary Classification Models

At the root of the task is building a Machine Learning model capable of binary classification and this also equally adept at handling unbalanced classes.

- k-Nearest Neighbors (k-NN)
  - Simple implementation and quick execution
- Support Vector Machines (SVM)
  - Simple implementation and quick execution
- Random Forest Decision Trees (RF)
  - Fairly simple implementation, regarded for handling unbalanced data
- Light Gradient Boosting Framework (LightGBM)
  - Complex tuning options, distributed, and highly regarded for handling unbalanced data
  - In the binary classification case, returns the probability that a sample belongs to the positive class.

Evaluate Model Performance with MCC

- Submissions are evaluated using Matthews Correlation Coefficient (MCC) between the predicted and observed response. The MCC is given by:

$$MCC = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

where TP is the number of true positives, TN the number of true negatives, FP the number of false positives, and FN the number of false negatives.

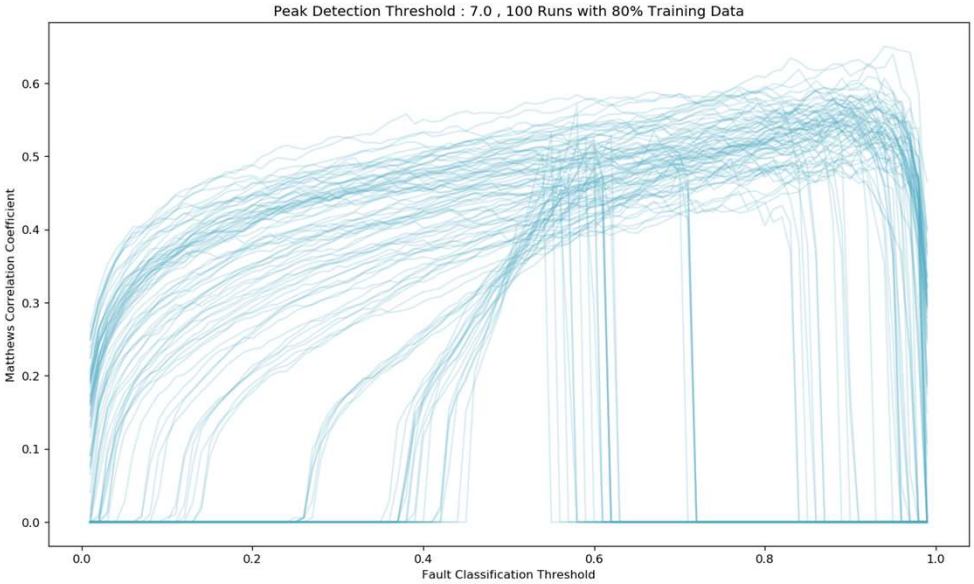
- Matthews Correlation Coefficient is an apt metric to assess a binary classification model, especially one with a highly unbalanced class where metrics like accuracy, precision, and recall don't adequately capture the effectiveness of a model.

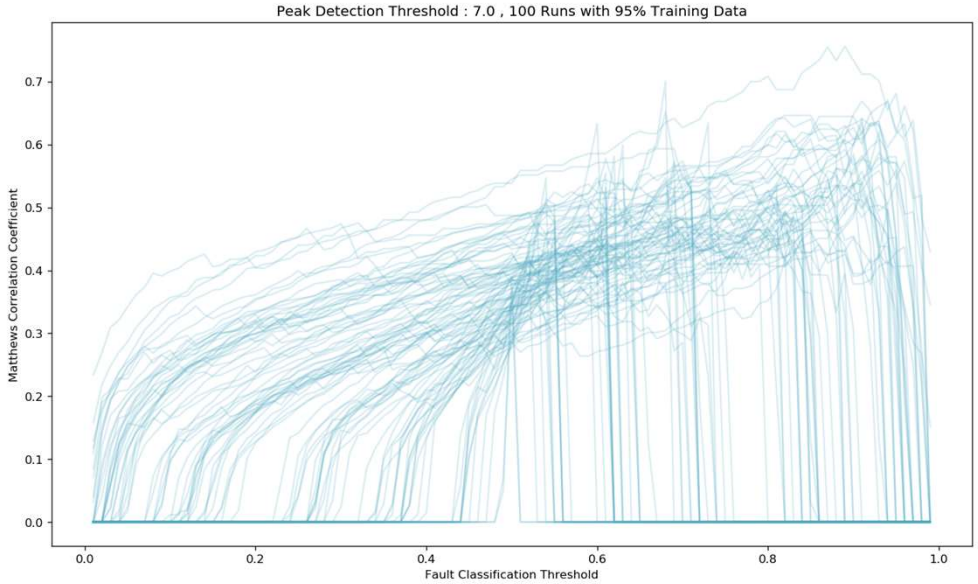
To Do

Add Once Extraction Complete
Train and Run Models in Notebook

Use Monte Carlo Analysis to Set Thresholds

Perform Monte Carlo trials that vary the `random_state` and `test_size` parameters in sklearn's `train_test_split` function and assess the impact of different thresholds on performance.





The goal is to find and set thresholds for Peak Detection and Fault Classification that routinely yield higher Matthews Correlation Coefficient scores and are robust to varied divisions of the training data.

dragonaur

Centrum ENET

Tuning LightGBM for Unbalanced Data

To Do

LGBM Params

dragonaur

Centrum ENET

Project Results

- Kaggle Leaderboard

To Do

Kaggle Leaderboard Results

dragonaur

Centrum ENET

Conclusions

- Increased experience with performing signal processing in Python
- Increased experience with binary classification machine learning tasks where the labeled data is heavily unbalanced
- Introduction and familiarity with a new (to me) machine learning framework: LightGBM

- At the conclusion of competition, my Kaggle kernel and GitHub repos went public:
  - Kaggle Kernel: https://www.kaggle.com/jeffreyegan/
  - GitHub Repo: https://github.com/jeffreyegan/VSB_Power_Line_Fault_Detection
  - Forks welcome!

Improvements

- While not uniformly true, when a fault is present in one phase, the likelihood of fault present in one or both or the other phases is also quite high.
  - Implement some logic that weights the probability of detection in the aggregate of the measurement?
  - How to use this data without introducing bias?
- At the conclusion of competition, my Kaggle kernel and GitHub repos went public:
  - Kaggle Kernel: https://www.kaggle.com/jeffreyegan/
  - GitHub Repo: https://github.com/jeffreyegan/VSB_Power_Line_Fault_Detection
  - Forks welcome!

References

1. Misak, S., et al. “A Complex Classification Approach of Partial Discharges from Covered Conductors in Real Environment.” *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 24, no. 2, 2017, pp. 1097–1104., doi:10.1109/tdei.2017.006135.
2. Vantuch, Tomas. *Analysis of Time Series Data*, 2018, dspace.vsb.cz/bitstream/handle/10084/133114/VAN431_FEI_P1807_1801V001_2018.pdf.

What's Next?

- Predicting Earthquakes for Los Alamos National Labs – Geophysics Group!

- Forecasting earthquakes is one of the most important problems in Earth science because of their devastating consequences. Current scientific studies related to earthquake forecasting focus on three key points: when the event will occur, where it will occur, and how large it will be.
- In this competition, you will address when the earthquake will take place. Specifically, you'll predict the time remaining before laboratory earthquakes occur from real-time seismic data.

- https://www.kaggle.com/c/LANL-Earthquake-Prediction

Los Alamos
NATIONAL LABORATORY
EST. 1943

dragonaur