AIHack 20
Shell Challenge

Team Just Vibes

Problem Statements

1. Can we predict compressor trips?
2. How far in advance can we identify unstable compressor conditions?
3. Can we provide causality information for engineers to investigate?

Stationarity Tests

Augmented Dickey-Fuller Test was performed on the clean data.

This gave that all columns are stationary with the exception of columns:

- Flow Rate Indicating 8
- Pressure Difference 27
- Pressure Indicating 42
- Pressure Indicating 44
- Pressure Emergency Action 63
- Temperature Indicating 120
- Unknown Computing 132
- Classified Indicating 185
- Classified Indicating 187
- Pressure Difference 200

Markov Property

Does the future state only depend on current state?

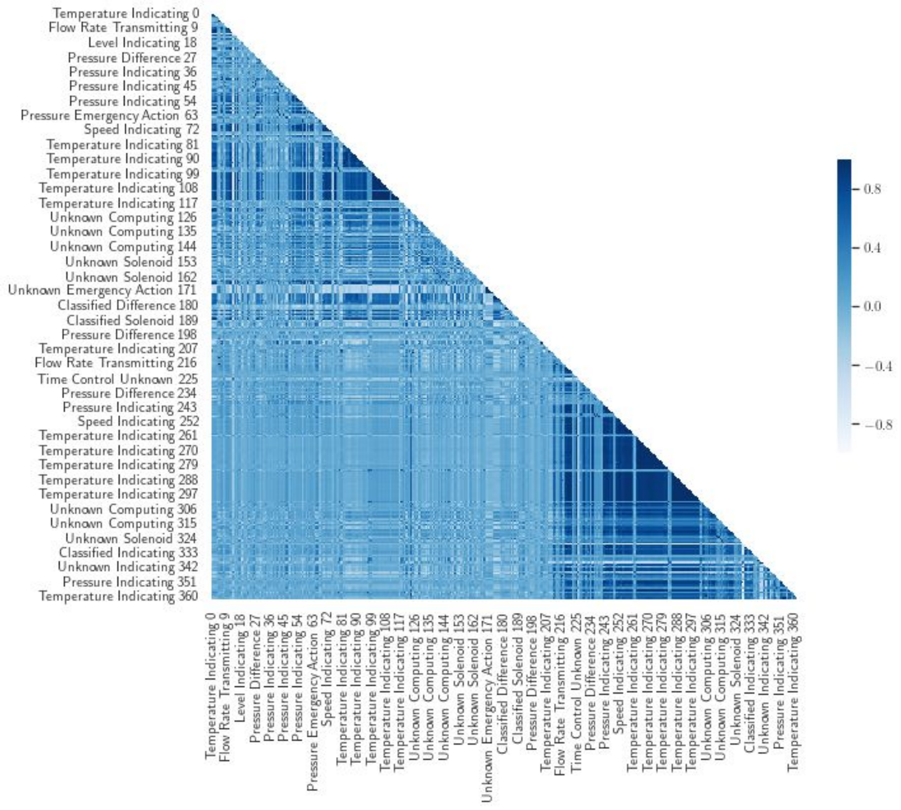
Let’s fit a VAR(2) model to the data and inspect coefficients.

- Smaller coefficients mean the process is less explainable by a linear model.
- Mean absolute coefficient for t - 1 regression: 0.073
- Mean absolute coefficient for t - 2 regression: 0.043

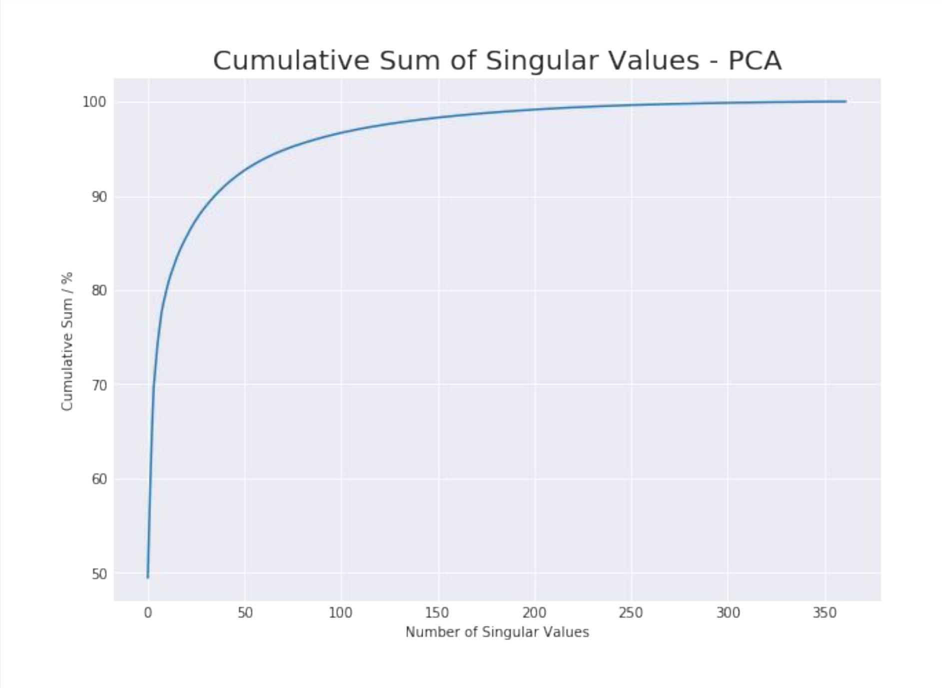
Non significant coefficients, we can assume Markov Property.

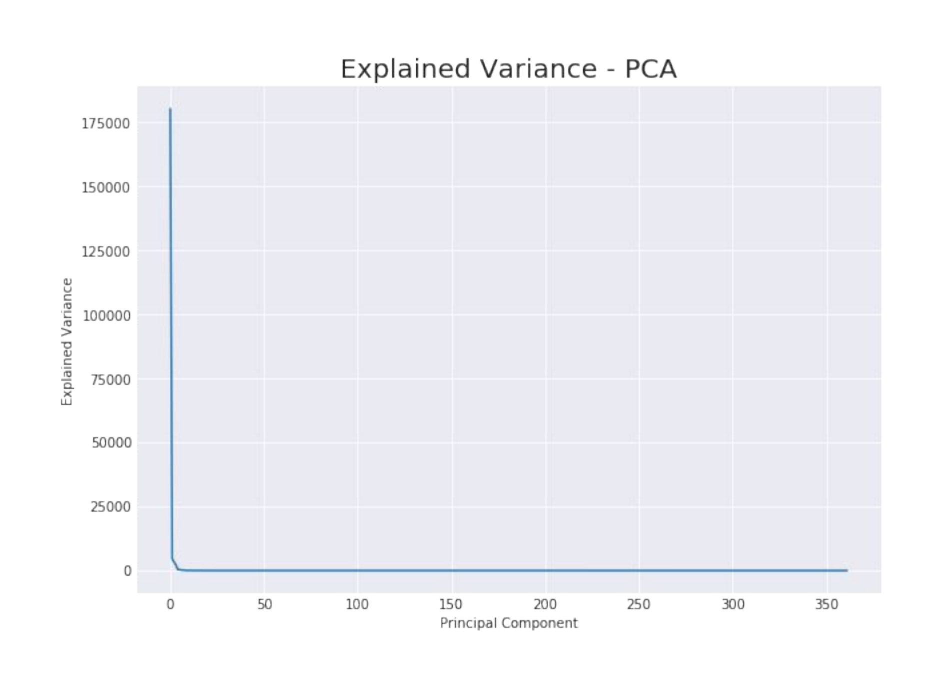
Correlation Analysis

- Features are linearly correlated.
- Mean correlation: 0.22.
- Can we find principal components?



Principal Component Analysis (PCA)

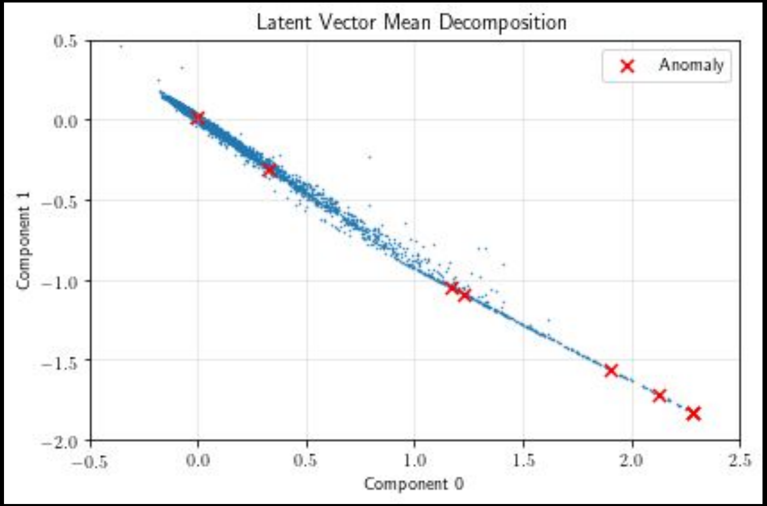




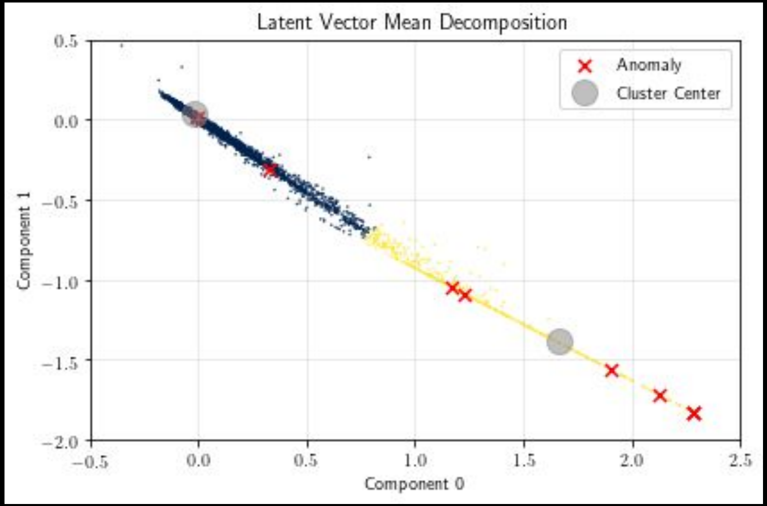
Anomaly Detection with VAE

- KL Divergence term in the objective function ensures that similar samples are as close to each other in the latent space as possible.
- Instead of looking at the reconstruction error let’s decompose the latent vector and attempt to find relationships within it.

Inspecting the latent p(z|x) vector in the VAE



K-means
Clustering



VAE is a generative model so we can sample the latent space to produce unseen samples and make a 'digital twin'. For example, we can sample the anomalous cluster and observe what unseen anomalous behavior looks like.

Brief findings and results

- 77% (7 out of 9) of given anomalies predicted correctly.
- ~2% of the test set (21000 points overall) classified as anomalous.

Investigating causality using p(x|z) distributions

Output is a Normal distribution vector per feature.

Our cause ranking approach:

1. Get p(x|z) distribution at output layer.
2. Calculate z score per feature (how far away a datapoint is from the mean).
3. Rank feature causality by z score.

This approach works, but the features are heavily correlated.

Demonstration

www.shellchallenge.ml