Predicting Semiconductor Factory Failures

Jared Nielsen

December 12, 2018

1 Abstract

Automation is the present and future of manufacturing, yet fully automated systems are still out of grasp. Predicting and preventing equipment failure is a major step in this process. We apply predictive models to production data from a semiconductor factory in Phoenix. Which measurements (such as motor speed, operating temperature, or vibration frequency) are most helpful in determining equipment health?

2 Introduction

A single equipment failure in semiconductor manufacturing can cost $100,000. Replacing machinery and recouping product loss is expensive and painful. Can we predict when factory equipment will fail? With how much advance warning?

Semiconductor manufacturing brings a unique set of challenges. To lay high-precision electronic circuitry on a chip, temperatures must be 9° Kelvin (-443° Fahrenheit). Any higher, and defects arise in the doping process. To achieve this, turbomolecular pumps (turbos) create a near-vacuum, and cryogenic pumps (cryos) cool the system. These pumps are incredibly precise and powerful, using a balanced array of electromagnets to levitate a steel cylinder spinning at thousands of RPMs.

Now suppose that one of these electromagnets becomes corroded, outputting less power. The vertical spinning cylinder quickly develops a wobble and spins out of control. With too much wobble, the cylinder will contact its container, and the entire system will literally crash and burn, destroying the pump itself and the wafer being processed.

Currently, failure prevention is done via thresholding and visual inspection. Each pump has a status monitor next to it, and passing the threshold triggers a status from green to yellow to red. Auto-shutoff capabilities are limited to prevent disruption, so we need smarter predictive models that prevent equipment failure but have fewer false positives than thresholding.

3 Objective

The long-term objective is to predict when pumps will fail, with as much lead time as possible. The goal of this paper is to identify predictive variables, following the curriculum learning framework. For example, are vibration frequencies more indicative of equipment health than operating temperature, or vice versa? Knowing predictive variables has a twofold benefit. First, it is a much more robust way to build a predictive model, preventing overfitting. Second, it enables smarter thresholding by building on technicians' intuition with verified mathematical modeling.

The major challenge is that there is no data from failed pumps. The class imbalance is 100/0, which turns the problem from a supervised learning domain into an unsupervised learning domain. Specifically, it is an unsupervised learning domain where all the data is of one class. This will require domain knowledge priors to generalize well.

Because this is a significant issue, we are developing processes to gather more data. However, it may be several weeks or even months before additional data is available. At $100,000 per failed pump datapoint, TCS isn't inclined to generate more training data. We therefore seek to identify predictive variables as a proxy for and precursor to a fully-functioning predictive model.

The rest of this paper focuses on solving this subproblem of identifying predictive variables. We hypothesize that predictive variables will be the ones with highest variance between pumps. These unique variables form a 'fingerprint' of the pump. Because they can distinguish between two properly-functioning pumps, they should also indicate a potentially failing pump.

4 Data Sources

The data for this research is obtained from production equipment via Texas Capitol Semiconductor (TCS), in coordination with BYU's Perception, Control and Cognition Lab. Because TCS is primarily a manufacturing and maintenance facility, their data science pipeline is still developing. The data consists of an unstructured single-table database, received as the concatenation of several CSV files.

The original data came in 14 million rows and 38 columns, intermixing two different pump types (turbo and cryo), various unstandardized bitflags, and always-null columns. This was merged with a separate index file linking pump ids to pump type, and then separated according to pump type. Since the Python package `pandas` is unable to handle 2GB files, this data munging was done batch-wise. Data was also upsampled from seconds to minutes, removing duplicates. This significantly reduced the input parameters without loss of signal.

Conversations with managers and technicians indicated that turbo pumps would have higher business value than predicting cryo pumps. The remainder of this paper focuses on turbo pumps.

5 Feature Engineering

To tackle this single-cluster unsupervised learning problem, we identify predictive variables by solving a subproblem. Which features contain enough variance to identify well-functioning pumps from each other? Though all pumps are functioning healthily, there is operating variation due to length of service and job type. The assumption is that features which distinguish one functional pump from another will also be instructive in distinguishing a working pump from a malfunctioning one.

The following columns were included:

- **Pump ID** – unique identifier for each pump
- **Timestamp** – datetime for the observation
- **Vibration (top of cylinder)** – floating point number
- **Vibration (bottom of cylinder)** – float
- **Rotor Speed** – float

- **Temperature** – float
- **Voltage** – float
- **Current** – float

These features were chosen after consulting with the technicians, who are domain experts in manual anomaly detection. One technician hypothesized that ramp-up and cool-down time could also indicate pump health. That is, a pump that takes 10 minutes to start up is more worrisome than one which can ramp up in 5 minutes.

To test this hypothesis, we engineered the following features:

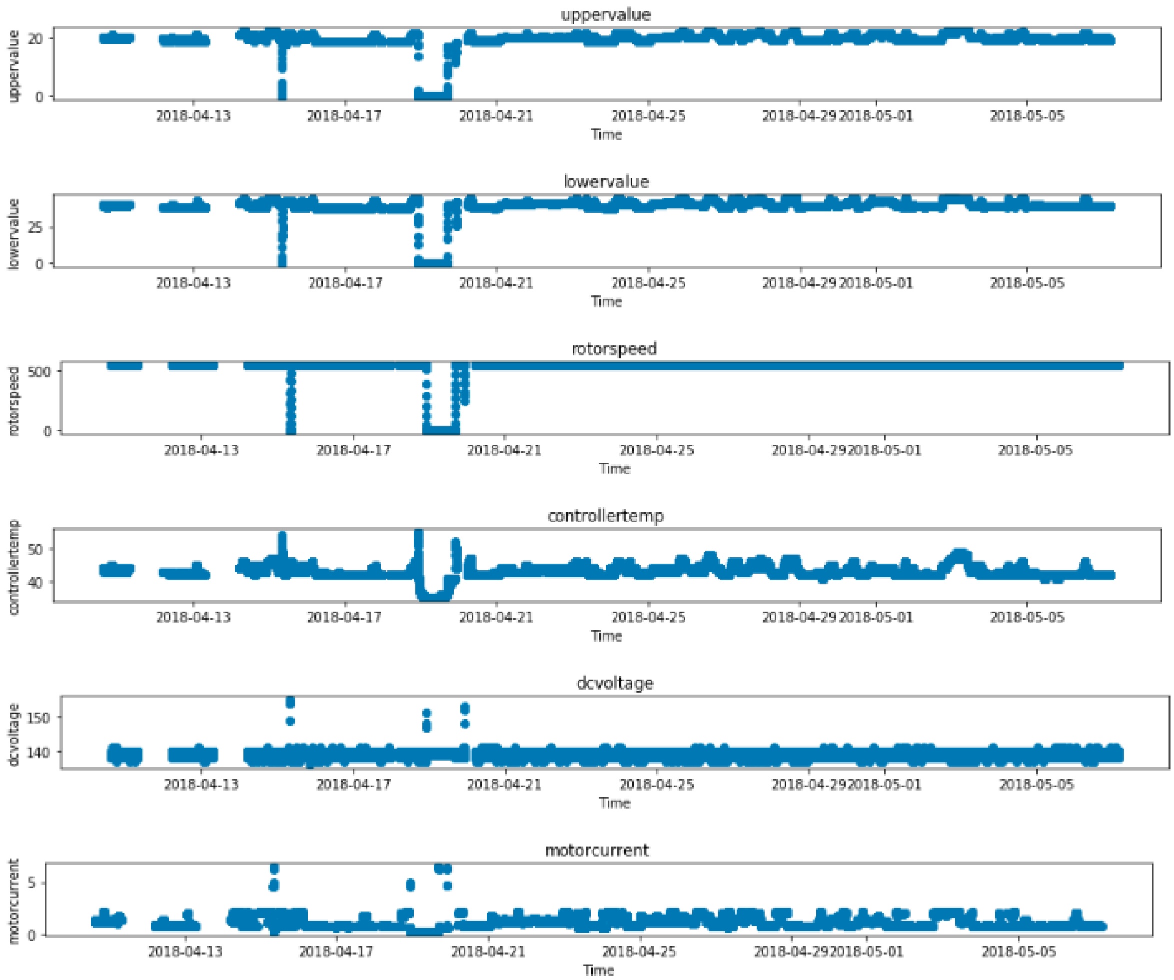
- **Separate Continuous Data into Short Time Intervals** – 64-minute intervals, both because it nearly corresponds with an hour and fits neatly into a 1-dimensional convolutional neural network (CNN) for machine learning classification.
- **Feature Similarity Metric** – The normalized dot product is used as a similarity measure between features in the same interval. This is demonstrated in the Gram matrix visualization.
- **Piecewise Linear Regression** – Linear regression, with the additional possibility of using multiple line segments to fit the data. To strike a balance between bias and variance, there is a penalty for using many piecewise functions. The piecewise linear fit can then be used to numerically calculate the ramp-up and cool-down slope. This is one of the paper's significant innovations.

6 Data Visualization

To identify predictive features, we now step through several visualizations of pump function.

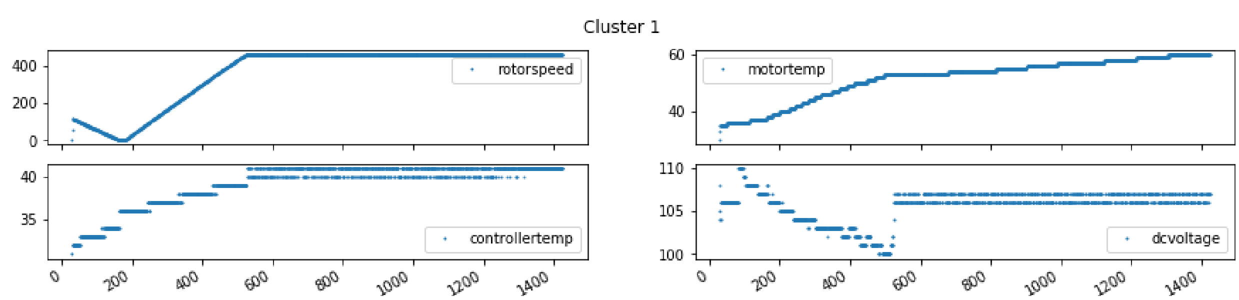
6.1 Pump Data over Several Months

This is a single pump operating from April 2018 – May 2018. Note that with the exception of a single blip around April 19, the pump functions nearly continuously.



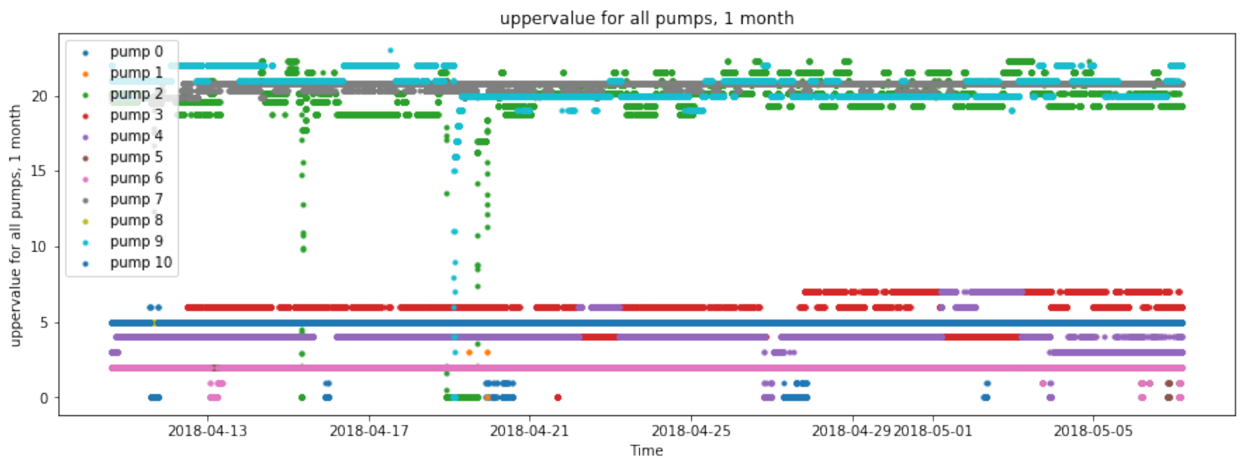
6.2 Pump Data during One Day

This is the same pump zoomed in on the first cluster of points before April 13. Note the ramp-up speed in the first plot is clearly visible and measurable with a piecewise linear regression. It takes around 400 timesteps.



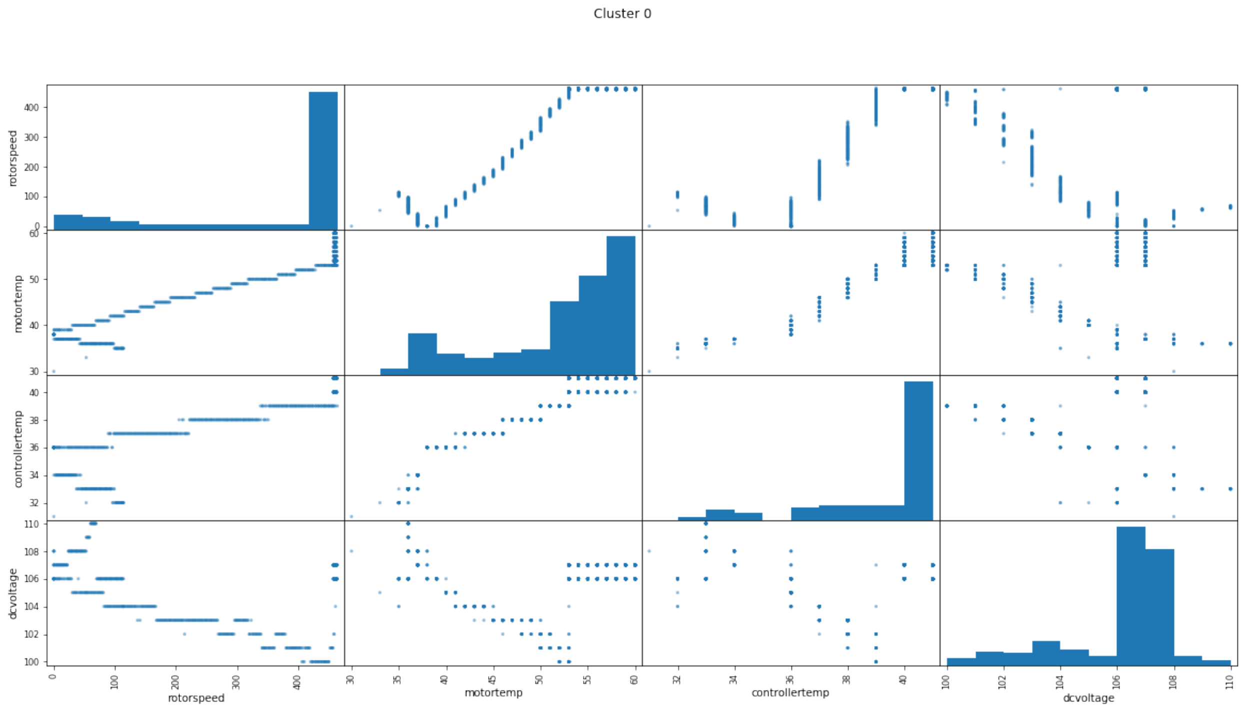
6.3 Comparing Vibration Values for All Pumps for One Month

Now we begin to examine the variance of certain features between different pumps. Features with high variation between pumps are likely to be predictive features. Below are 10 different pumps and their vibration values over the course of a month. Note that each pump has a clear median operating vibration, with clear separation between two larger classes. This is likely a technical detail of the pump model or the type of job being run.



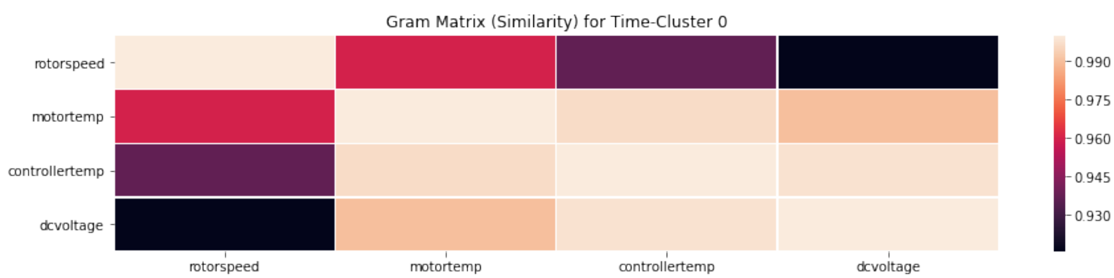
6.4 Comparing Features for a Single Interval – Scatter Matrix

This scatter matrix describes a time interval for a single pump. Each row refers to a feature (speed, temperature, voltage) and each column also refers to those same features. Where a feature intersects itself, a histogram is displayed; otherwise a 2-dimensional scatterplot is displayed. This is a qualitative tool for noticing correlation. For example, motor speed and motor temperature are highly correlated, as one would expect. However, motor temperature is inversely correlated with voltage. This unintuitive observation was surprising even to the technicians who repair these pumps daily. Understanding why is a promising avenue to explore.



6.5 Comparing Features for a Single Interval - Gram Matrix

Feature similarity is computed via the normalized dot product, then plotted for each pair of features. The diagonal always takes the max value of 1. Despite the apparent disparity in colors, examining the colorbar legend shows that all possible combinations have a similarity measure > 0.90. This shows that the normalized dot product is likely not a useful feature.

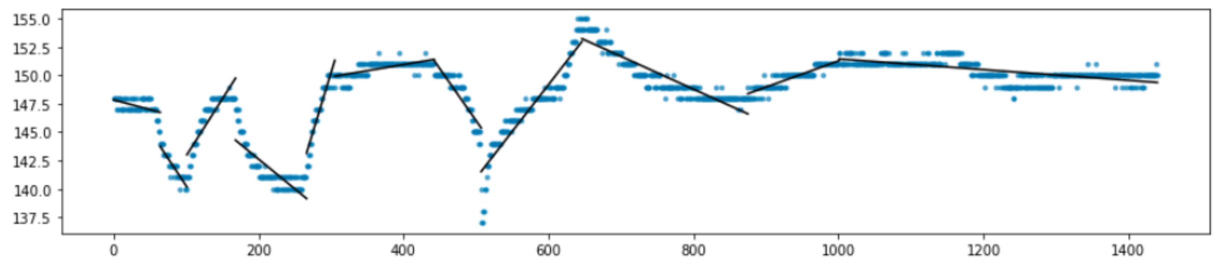


6.6 Piecewise Linear Regression

Piecewise linear regression shows the most promise for automated anomaly detection. Note that the piecewise linear regression model automatically chooses the optimal number of line segments. Below we visually demonstrate a properly fit model, an overfit model, and a model used to calculate ramp-up slope.

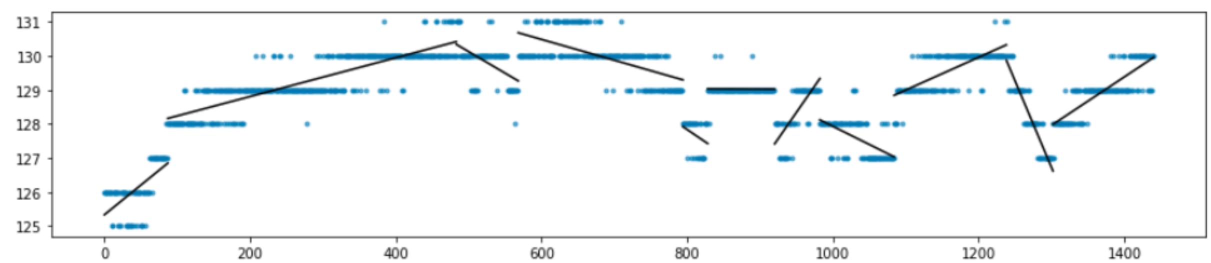
Here we see a piecewise linear model properly fitting to the data.

Num segments: 11



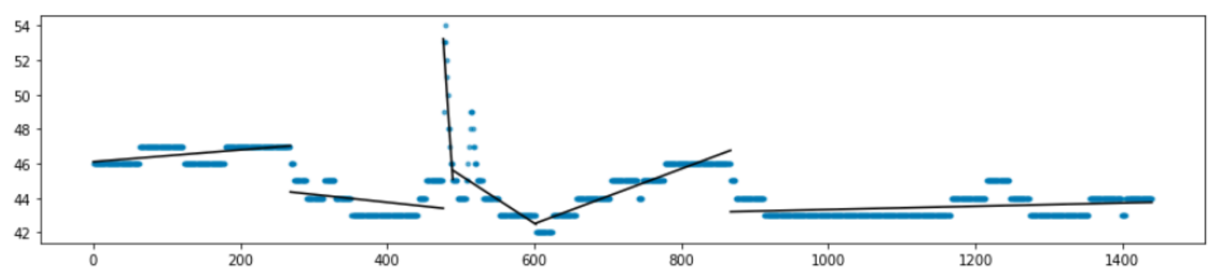
Here we see too many segments. An overfit model can easily produce lines of ridiculously high slope, so care is needed.

Num segments: 11



Here we see a brief spike in the data, along with the corresponding piecewise linear segment of high slope. This may be normal operating procedure, but is cause for a second look. Note that the line slope is calculated numerically, allowing more advanced thresholding procedures.

Num segments: 6

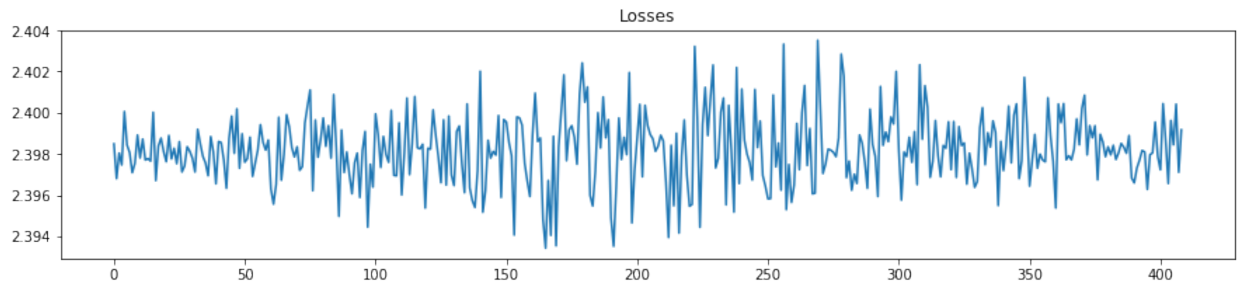


7 Ablation Testing

This section contains a brief overview of interesting methods which did not work.

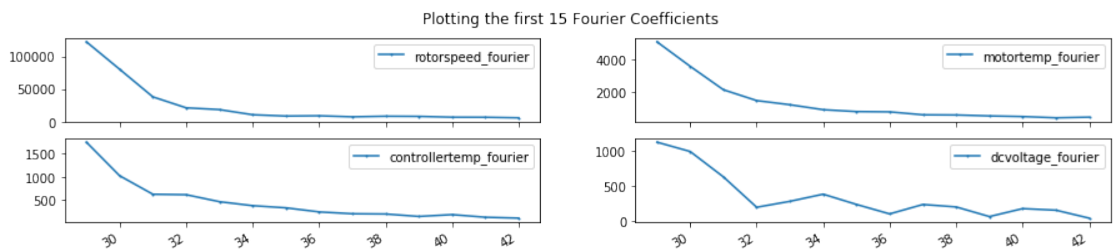
7.1 Convolutional Neural Networks

To test nonlinear separability, we trained a 1-dimensional CNN on a multi-class prediction problem. Given pump features over an hour interval, the CNN attempted to predict the pump ID. This did not work. The loss, hovering around 2.4, is equal to the loss from random guessing. While further architecture experimentation and training methodologies can be explored, this attempt suggests that better feature extraction may be needed to properly distinguish pump ids.



7.2 Fourier Coefficients

We also hypothesized that the Fourier transform of values could provide useful features. Below is a representative sample. Voltage appears to have oscillatory values, likely because electrical equipment functions at certain set frequencies rather than a normally distributed range. High-frequency voltage components may be a useful feature, but more fine-tuned data collection is needed to know for sure.



8 Results and Conclusion

While a functional predictive model will require further work, this analysis has produced several important insights.

First, piecewise linear regression can be applied automatically to measure ramp-up and cool-down time. Technician experience suggests these are a significant indicator of pump health. After gathering more data and setting appropriate bounds, this is an effective metric to track pump health over time.

Second, motor temperature and voltage are inversely correlated. This was a surprise to the technicians who monitor the software. This is an important line of investigation.

Finally, the Fourier transform of the voltage has an oscillatory nature. Is this by design or an interesting side effect? High-frequency vibrations are likely to give great predictive power. Measuring these high-frequency components requires a mechanical engineering solution closely integrated with data science.

9 Code

The code for this project is hosted on GitHub at https://github.com/jarednielsen/cryopumps.

- Data Cleaning
- Feature Engineering
- Generating Visualizations