LANL Earthquake Prediction Model

Group 7

CSCI 470: Machine Learning (OL Section), Canvas Group 7

Elcin Eroglu, Ryan Sundberg, Linzhi Leiker, Cristian Madrazo

THE PROBLEM:

Predict the time until a lab-induced earthquake takes place given an array of 150k seismic signals.

There is currently no accurate way to predict earthquake time according to USGS.

Motivation:

- Intriguing blend of advanced data science
- Potential to significantly impact society

!!!

The methodologies and insights gained could lay the groundwork for real-world earthquake prediction applications, thereby providing crucial lead times that could save lives and minimize economic impact.

Raw Data

The dataset consists of continuous seismic signal records from Los Alamos National Laboratory.

Each data point contains:

- **Feature:** A single seismic signal value represented
- **Target:** The time-to-failure (in seconds)

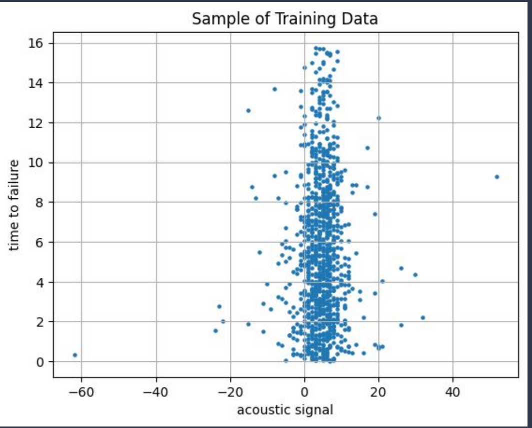


Figure 1: Sample Data Raw 1000 points

Challenges:

The main challenge in this project stems from the data's limited features, with only one seismic signal value per target. This unique setup requires:

- **Complex Feature Engineering**
- **Large Data Volume**
- **Sparse Information**

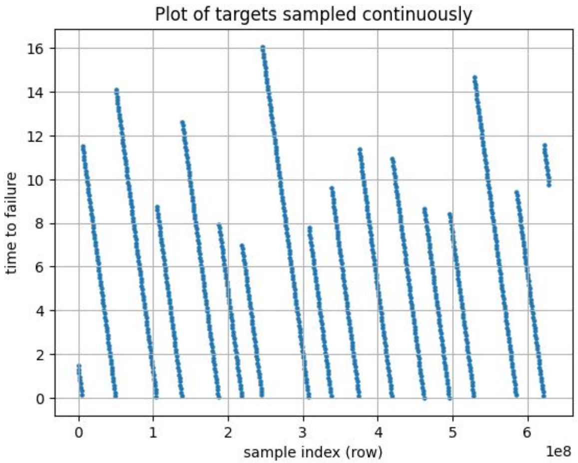


Figure 2: Raw Data was collected as a cont. sample

Initial Trials

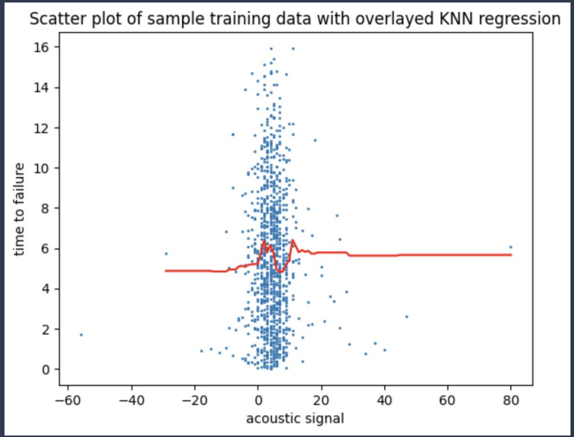


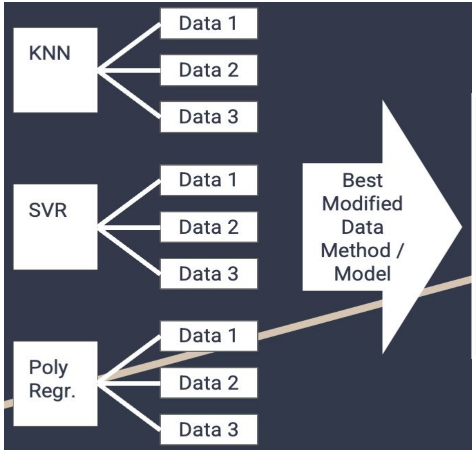
Figure 3: KNN with k = 50, norm = l2

RMSE = 6.3

Data Modification with;

- Average Method (Data 1)
- Sum Method (Data 2)
- N-dimensional method (Data 3)

After Data Modification,

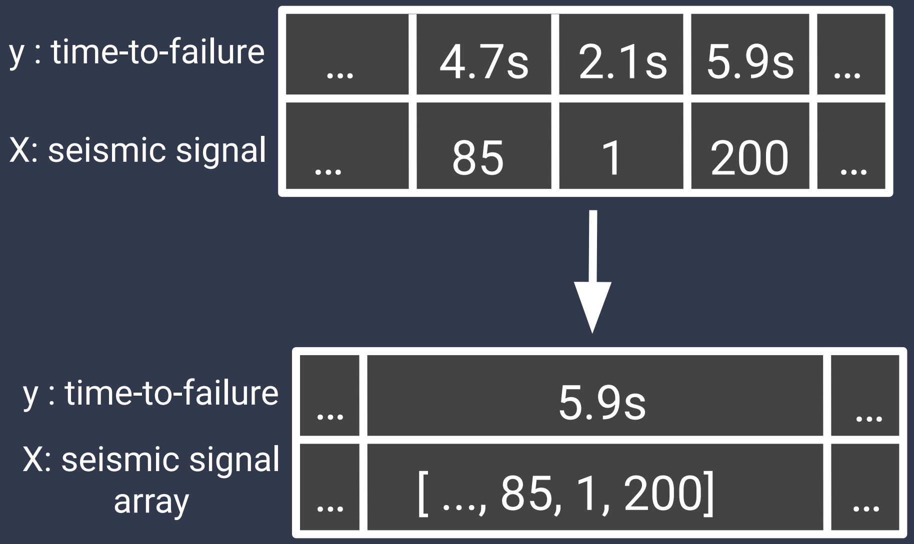


Model	Method			
Model		sum	avg	n-dim
	SVR	3.39	3.39	2.71
	Poly	3.41	3.41	3.78
	KNN	3.38	3.38	.8

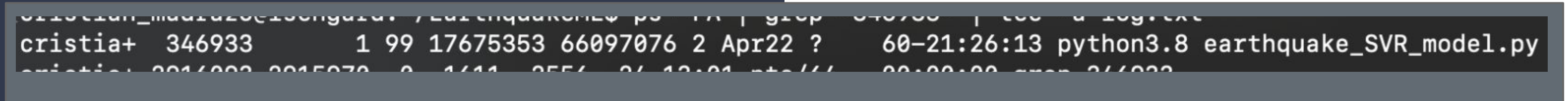
Best combination

- SVR
- Parameters:
  - Epsilon: 1
  - C Value: 100
  - Kernel: RBF
  - Degree: 3
- N-dimensional Data Method
- Scored RMSE 2.71

Data Synthesis

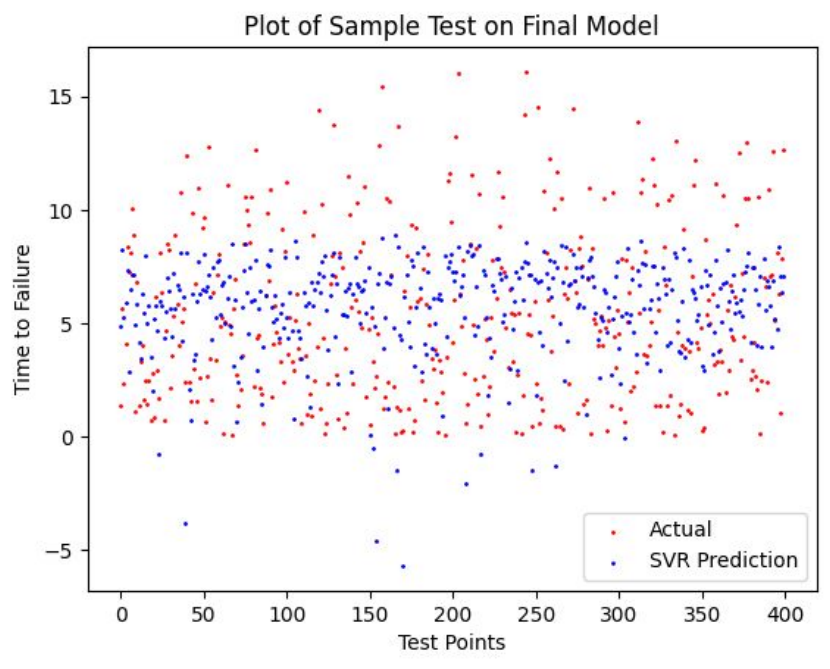


Creating features from existing data

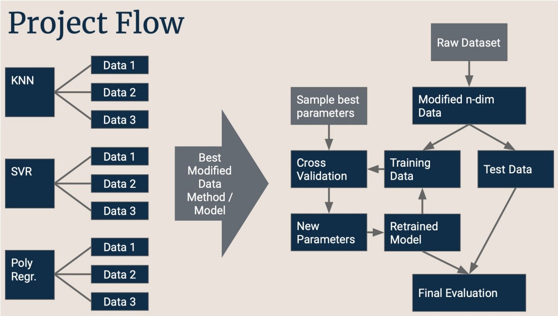


- Allows us to add context to our dataset
- Added the previous 150,000 features to every feature.
- Major disadvantages
  - Time consumption, generating a 1000-target sample took ~12 hours cpu time
  - Time inefficiency trickles down to other processes
  - Final SVR model parameter search consumed over 60 days cpu time, had to be stopped short!
  - Limited our model exploration stage because every test took hours
  - Space inefficient, raw data with >6 million targets = 10gb, while 1000-target synthesized sample = 1gb

Current Model



- After data synthesis step, we compared KNN, SVR, and Polynomial Regression on 1000-point sample
- SVR performed the best so we chose it for a larger scale parameter search
  - GridSearchCV for cross-validation
  - Synthesized data with 8000-targets
- Support Vector Regressor (SVR)
  - Kernel: RBF
  - C: 100
  - Epsilon: 0.1
  - Gamma: Auto
  - RMSE = 2.3



Demo

- Saved model using joblib library
- Created an API using pipenv and voila
  - Used widgets from ipywidget
- API intended for scientists and engineers comfortable with simple interfaces
- Requires a file as input
- Outputs a single number in seconds

Conclusion

- **Achievements:** Successfully developed predictive models for earthquake timing with significant improvement in RMSE from 6.3 to 2.3 using innovative feature engineering and Support Vector Regression.

- **Impact:** Demonstrated the potential for applying advanced machine learning techniques to complex, real-world problems in seismic prediction.

Future Work

- **Model Optimization:** Continue refining our models through extensive hyperparameter tuning and cross-validation to further reduce RMSE.
- **Data Enrichment:** Explore additional data sources and feature engineering techniques to enhance model robustness.
- **Real-World Testing:** Plan pilot studies to apply our models to real-world seismic data, assessing practical viability and reliability.
- **Alternative Models:** More experimenting with deep learning

LANL Earthquake Prediction Model

Group 7

CSCI 470: Machine Learning (OL Section), Canvas Group 7

Elcin Eroglu, Ryan Sundberg, Linzhi Leiker, Cristian Madrazo