Universitat
Pompeu Fabra
Barcelona

Initial Setting Time Prediction for Calcium Aluminatum Cement

Francesc Xarrié
Raúl Martin
Noel Pedrosa

Index

Introduction

- Context: Cement manufacturing involves many interdependent variables and process conditions, one of them the initial setting time.
- Goal: Understand and improve feature representation to support predictive modeling.
- Approach: Combine domain knowledge with dimensionality reduction techniques (MLP, PCA, Autoencoders).
- Plan: Build representations, compress them, then test performance in predictive tasks (via MLPs).

Methodology

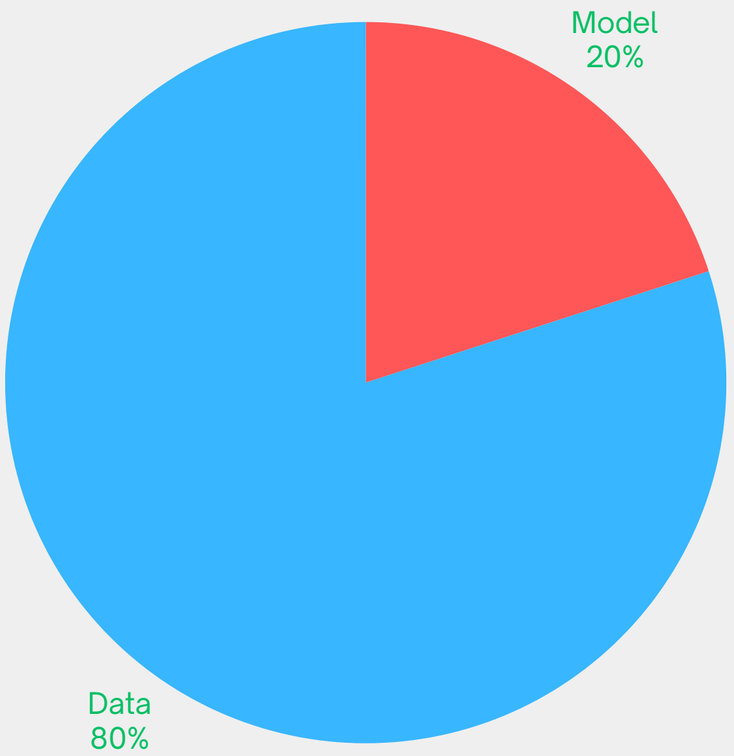
- 2 Datasets
- 1 Autoencoder
- 2 MLP
- 6 Experiments

Dataset

“A model is only as good as the dataset that feeds it.”

The 80/20 Rule of Modeling

We can’t deny the saying: most of the modeling success depends on how well we prepare the data. That’s where our focus and work was.



Raw Features Before

Composition & Chemical Features

- Chem_1 to Chem_11
- Chem_12
- Chem_Ratio_2_7
- Chem_Sum

Phase Composition

- Phase_1 to Phase_13
- Phase_Ratio_3_1
- Alum

Colorimetric Properties

- L, a, b

Fineness & Particle Size Distribution

- 90
- 902
- 753
- 454
- 305
- d10
- d50
- d63
- d90
- n
- Blaine
- BlaineS

Spectral / Profile Features

- R_profile
- G.O.F.

Moisture & Process Metrics

- H2O(Humidity)
- water
- plumb
- L.o.i.
- L.o.i. calc.

Metadata / Categorical

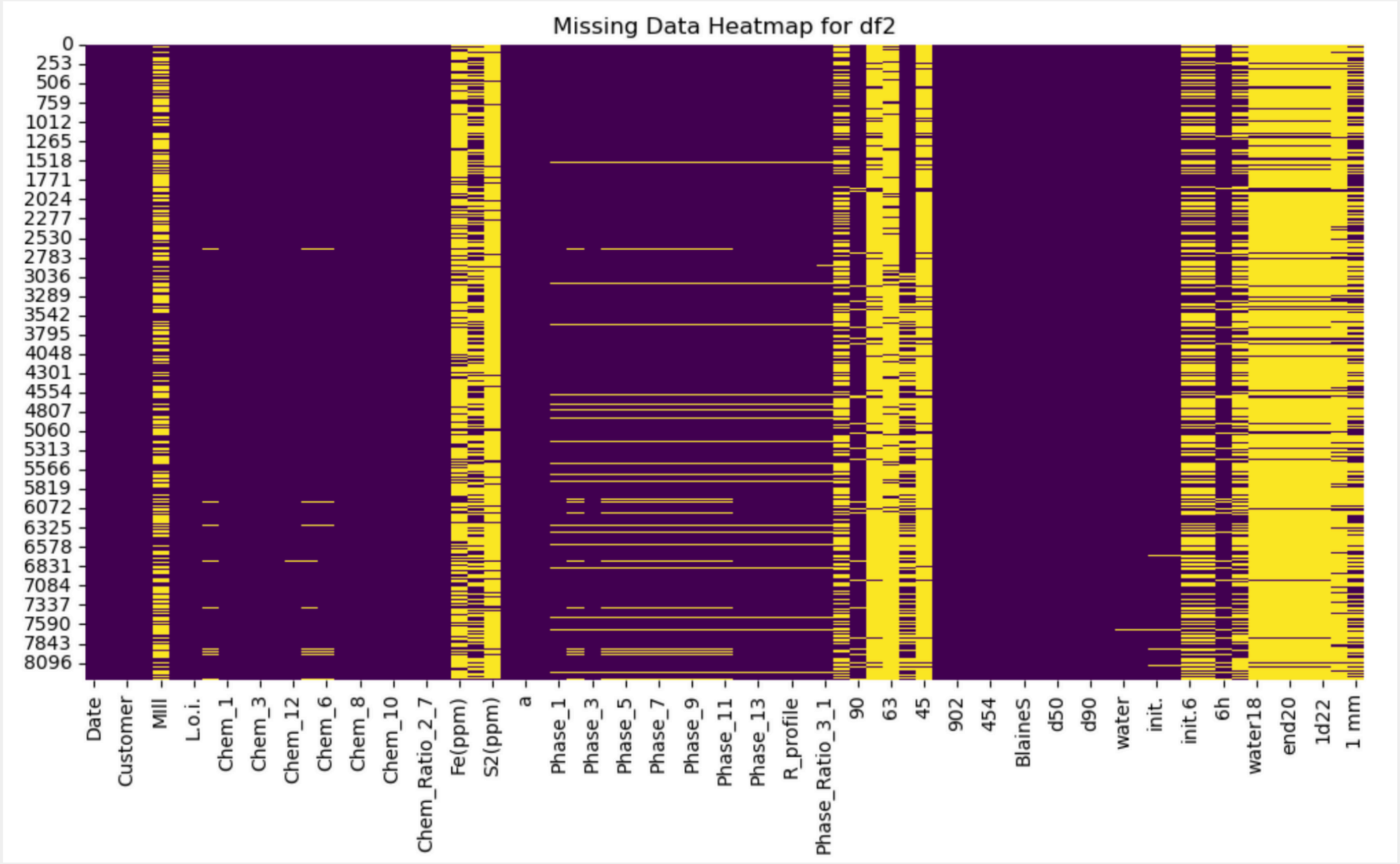
- Date
- Type
- Process

Others:

- Customer
- MIll
- Fe(ppm)
- H2S(ppm)
- S2(ppm)
- Amorph
- 75
- 63
- 50
- 45
- init.6
- end7
- 1d
- water18
- init.19
- end20
- 6h21
- 1d22
- 0,5 mm
- 1 mm

Setting Times

- init. (TARGET🎯)
- end


Missing Data Heatmap for df2
0
253
506
759
1012
1265
1518
1771
2024
2277
2530
2783
3036
3289
3542
3795
4048
4301
4554
4807
5060
5313
5566
5819
6072
6325
6578
6831
7084
7337
7590
7843
8096
Date
Customer
MIII
Lo.i.
Chem_1
Chem_3
Chem_12
Chem_6
Chem_8
Chem_10
Chem_Ratio_2_7
Fe(ppm)
S2(ppm)
a
Phase_1
Phase_3
Phase_5
Phase_7
Phase_9
Phase_11
Phase_13
R_profile
Phase_Ratio_3_1
90
63
45
902
454
BlaineS
d50
d90
water
init.
init.6
6h
water18
end20
1d22
1 mm

Raw Features After

Composition & Chemical Features

- Chem_1 to Chem_11
- ~~Chem_12~~
- ~~Chem_Ratio_2_7~~
- ~~Chem_Sum~~

Phase Composition

- Phase_1 to Phase_13
- ~~Phase_Ratio_3_1~~
- ~~Alum~~

Colorimetric Properties

- L, a, b

Fineness & Particle Size Distribution

- 90
- 902
- 753
- 454
- 305
- d10
- d50
- d63
- d90
- n
- Blaine
- BlaineS

Spectral / Profile Features

- R_profile
- G.O.F.

Moisture & Process Metrics

- H2O(Humidity)
- water
- plumb
- ~~L.o.i.~~
- ~~L.o.i. calc.~~

} Ignition Loss

- ~~6h~~

Metadata / Categorical

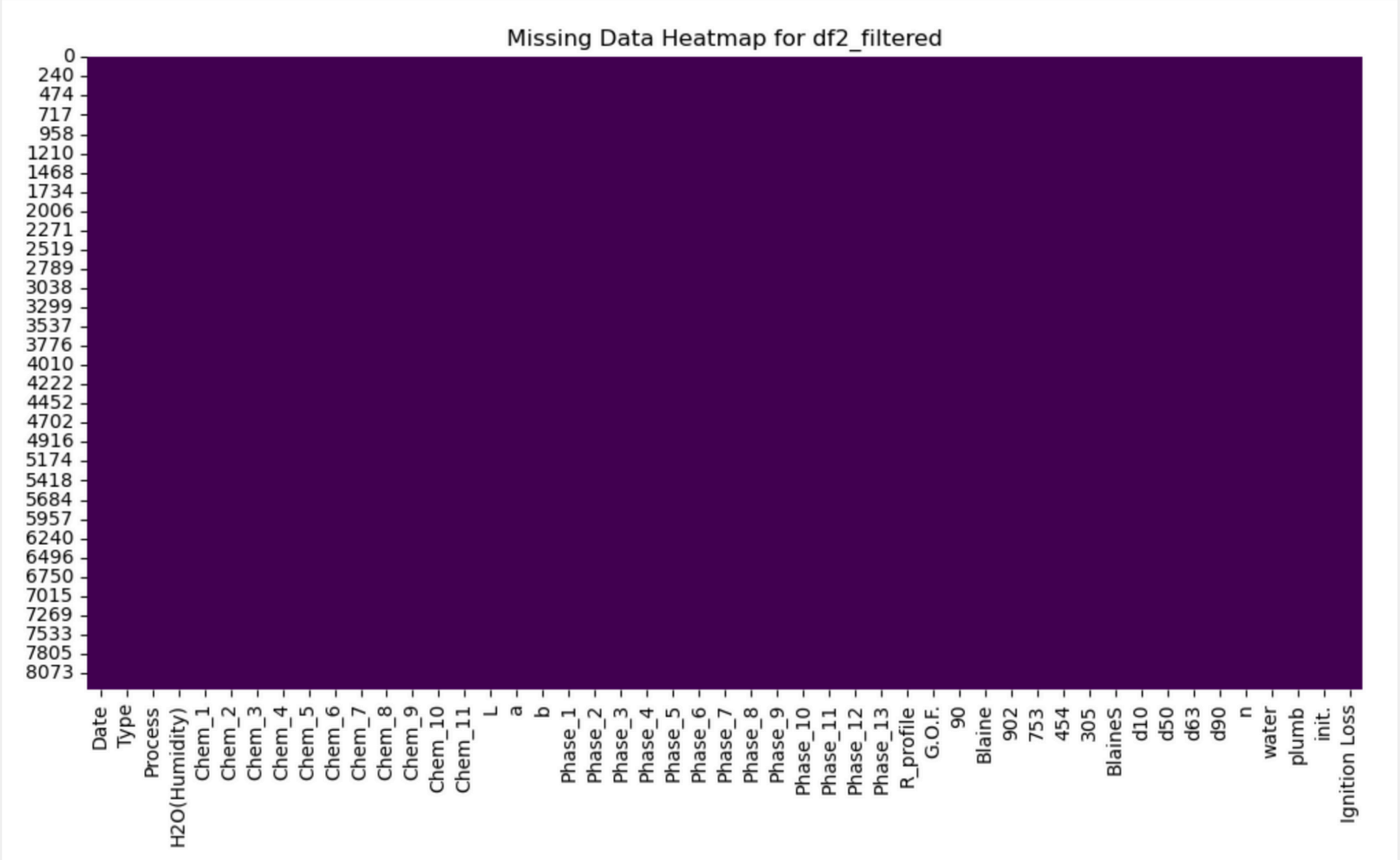
- ~~Date~~
- Type
- Process

Others:

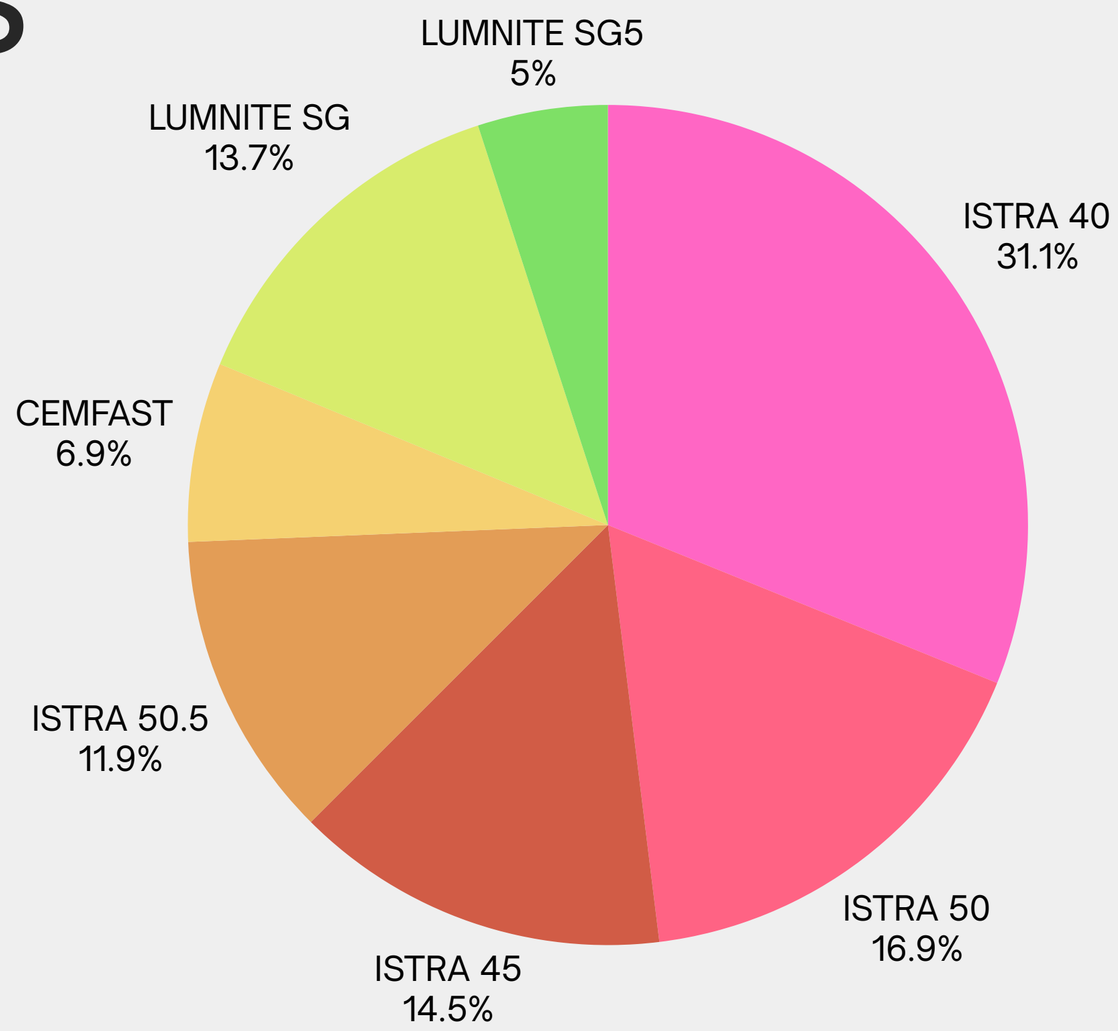
- ~~Customer~~
- ~~MIlt~~
- ~~Fe(ppm)~~
- ~~H2S(ppm)~~
- ~~S2(ppm)~~
- ~~Amorph~~
- ~~75~~
- ~~63~~
- ~~50~~
- ~~45~~
- ~~init.6~~
- ~~end7~~
- ~~1d~~
- ~~water18~~
- ~~init.19~~
- ~~end20~~
- ~~6h21~~
- ~~1d22~~
- ~~0,5 mm~~
- ~~1 mm~~

Setting Times

- init. **(TARGET🎯)**
- ~~end~~


Missing Data Heatmap for df2_filtered
0
240
474
717
958
1210
1468
1734
2006
2271
2519
2789
3038
3299
3537
3776
4010
4222
4452
4702
4916
5174
5418
5684
5957
6240
6496
6750
7015
7269
7533
7805
8073
Date
Type
Process
H2O(Humidity)
Chem_1
Chem_2
Chem_3
Chem_4
Chem_5
Chem_6
Chem_7
Chem_8
Chem_9
Chem_10
Chem_11
L
a
b
Phase_1
Phase_2
Phase_3
Phase_4
Phase_5
Phase_6
Phase_7
Phase_8
Phase_9
Phase_10
Phase_11
Phase_12
Phase_13
R_profile
G.O.F.
90
Blaine
902
753
454
305
BlaineS
d10
d50
d63
d90
n
water
plumb
init.
Ignition Loss

Cement Types



Space and Distributions of Features

Our dataset contains features that live in different mathematical spaces:

- **Simplex Space**

Chem_1 to Chem_11 and Phase_1 to Phase_13 are compositional features.

Their values sum up to 100%, meaning they are interdependent.

- **Euclidean Space**

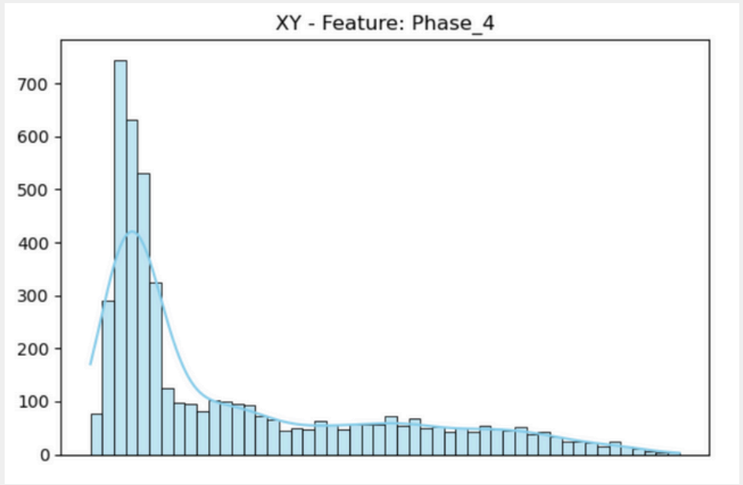
All other numerical features.

These display skewed, multimodal, or clustered distributions.

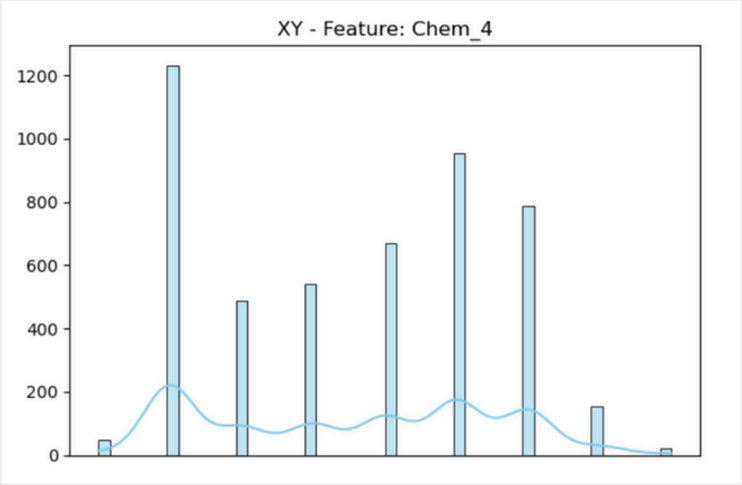
- **Categorical Features**

Type and Process are categorical and were one-hot encoded for modeling.

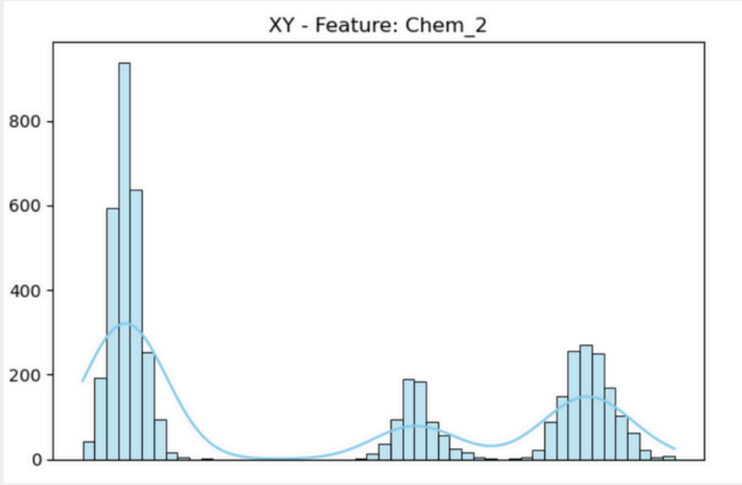
Weird Distributions –Histogram Examples



Skewed



Discrete-looking



Mixture

Features’ Transformations

- We applied the ILR transformation to features in the simplex:
  - Chem_1–11 and Phase_1–13
  - This converts them to a Euclidean space, improving model performance.

- ILR formula (for D parts):

$$\mathrm{ILR}_i = \sqrt{\frac{i}{i+1}} \cdot \log\left(\frac{g(x_1, \ldots, x_i)}{x_{i+1}}\right)$$

  where g is the geometric mean of the first i components.

- Other numeric features: we tested log, log1p, box-cox, yeo-johnson
  - Minimal gain → we kept them untransformed.

Splitting Step

- We split the full dataset into a dictionary of cement types, each containing:
  - "X_train", "X_test", "y_train", "y_test"

- Why this split?
  - By cement type → handle data imbalance (e.g., some types had very few samples)
  - Train/test split inside each → avoid data leakage and ensure valid evaluation

- Structure Example:

```
{
  "ISTRA 40": {
      "X_train": DataFrame,
      "X_test": DataFrame,
      "y_train": Series,
      "y_test": Series
   },
  "Lumnite SG": {
        ...
   },
   ...
}
```

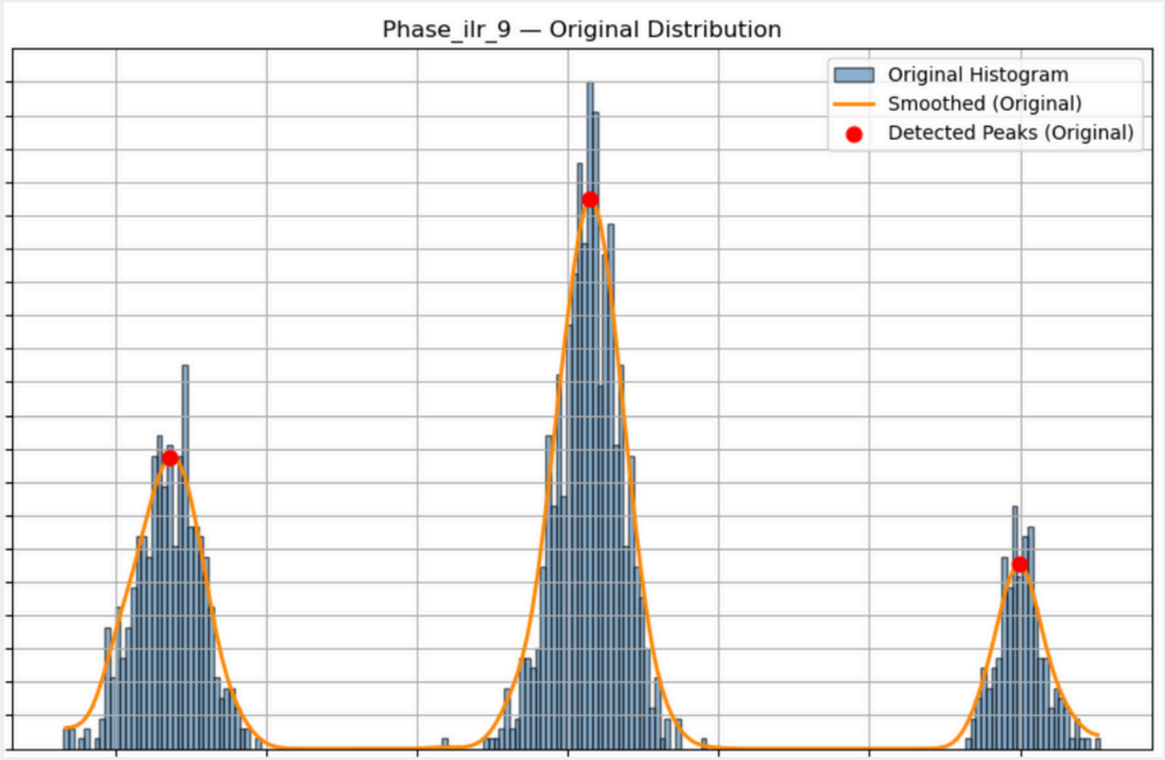
Outlier Strategy Overview

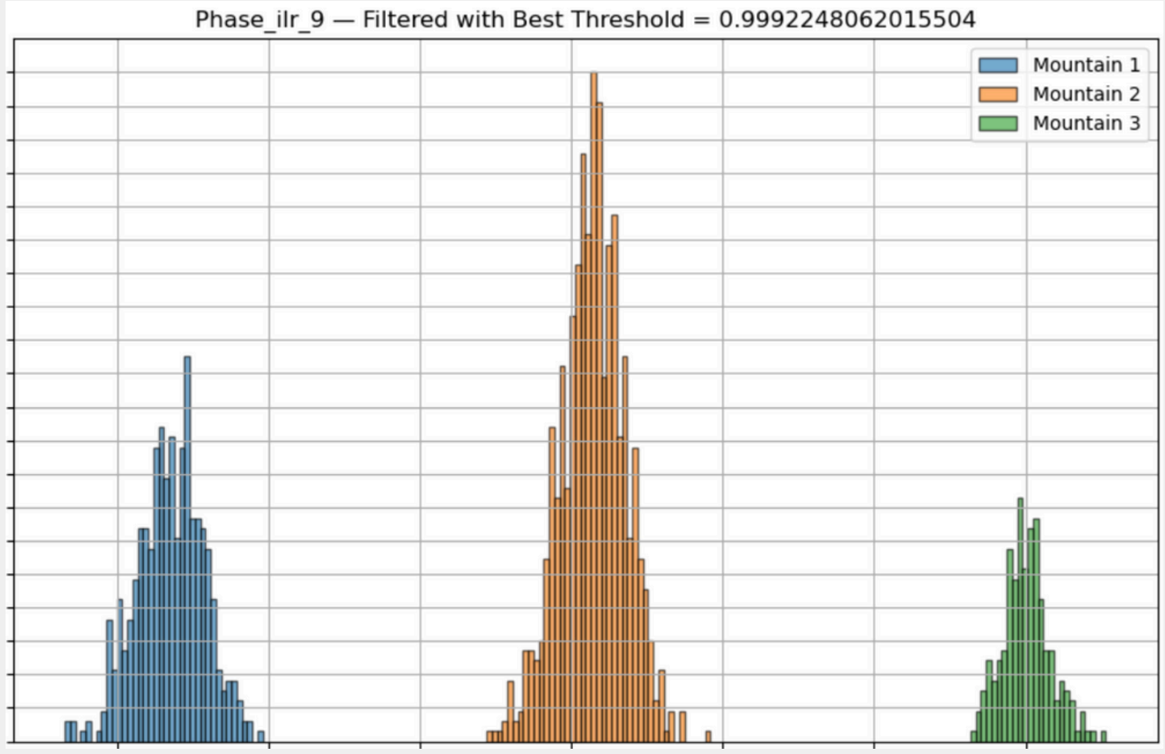
We use both:

- Univariate techniques
- Multivariate techniques

to clean only the training data, avoiding leakage into test sets.

Univariate Outlier Detection



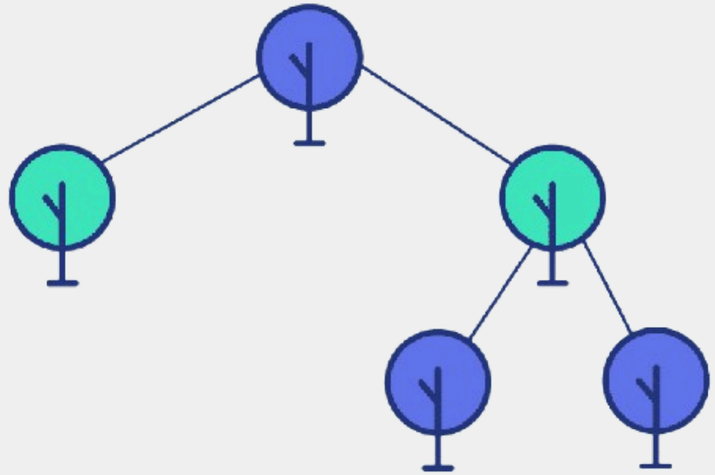


Multivariate Outlier Detection

We applied Isolation Forest, an unsupervised algorithm that detects anomalies by learning how easily samples can be isolated in a tree structure.

It performs well with high-dimensional data and does not assume any distribution.

- We used multiple contamination levels (very low values).
- For each, we measured the silhouette score to assess how well the inliers cluster together.
- We selected the best contamination level based on the highest silhouette score.



Final X_train versions

```
🧱 Cement: ISTRA 40
   ➤ OR Logic → Before: 1523, After: 1451 (Removed: 72 | 4.73%)
   ➤ AND Logic → Before: 1523, After: 1522 (Removed: 1 | 0.07%)

🧱 Cement: Lumnite SG
   ➤ OR Logic → Before: 672, After: 634 (Removed: 38 | 5.65%)
   ➤ AND Logic → Before: 672, After: 672 (Removed: 0 | 0.00%)

🧱 Cement: ISTRA 50 5.0
   ➤ OR Logic → Before: 580, After: 522 (Removed: 58 | 10.00%)
   ➤ AND Logic → Before: 580, After: 575 (Removed: 5 | 0.86%)

🧱 Cement: ISTRA 45
   ➤ OR Logic → Before: 707, After: 656 (Removed: 51 | 7.21%)
   ➤ AND Logic → Before: 707, After: 703 (Removed: 4 | 0.57%)

🧱 Cement: ISTRA 50
   ➤ OR Logic → Before: 827, After: 743 (Removed: 84 | 10.16%)
   ➤ AND Logic → Before: 827, After: 823 (Removed: 4 | 0.48%)

🧱 Cement: CEMFAST
   ➤ OR Logic → Before: 337, After: 328 (Removed: 9 | 2.67%)
   ➤ AND Logic → Before: 337, After: 337 (Removed: 0 | 0.00%)

🧱 Cement: Lumnite SG5
   ➤ OR Logic → Before: 244, After: 223 (Removed: 21 | 8.61%)
   ➤ AND Logic → Before: 244, After: 244 (Removed: 0 | 0.00%)
```

```python
raw_weights = {
    "mean_shapiro":      1.0,
    "median_shapiro":    1.5,
    "ks_p":              1.0,
    "mean_mahalanobis":  1.0,
    "median_mahalanobis": 1.5,
    "mean_skew":         0.5,
    "median_skew":       0.25,
    "mean_kurtosis":     0.5,
    "median_kurtosis":   0.25
}
```

```
📦 Saved AND for ISTRA 50
📦 Saved OR for ISTRA 45
📦 Saved OR for ISTRA 50 5.0
📦 Saved OR for ISTRA 40
📦 Saved OR for Lumnite SG
📦 Saved RAW for CEMFAST
📦 Saved OR for Lumnite SG5
```

	Cement Type	Best	Score AND	Score OR	Score RAW
0	ISTRA 50	AND	0.500000	0.466667	0.0
1	ISTRA 45	OR	0.200000	0.666667	0.0
2	ISTRA 50 5.0	OR	0.433333	0.666667	0.0
3	ISTRA 40	OR	-0.100000	0.533333	0.0
4	Lumnite SG	OR	0.333333	0.666667	0.0
5	CEMFAST	RAW	-0.066667	-0.200000	0.0
6	Lumnite SG5	OR	0.333333	0.400000	0.0

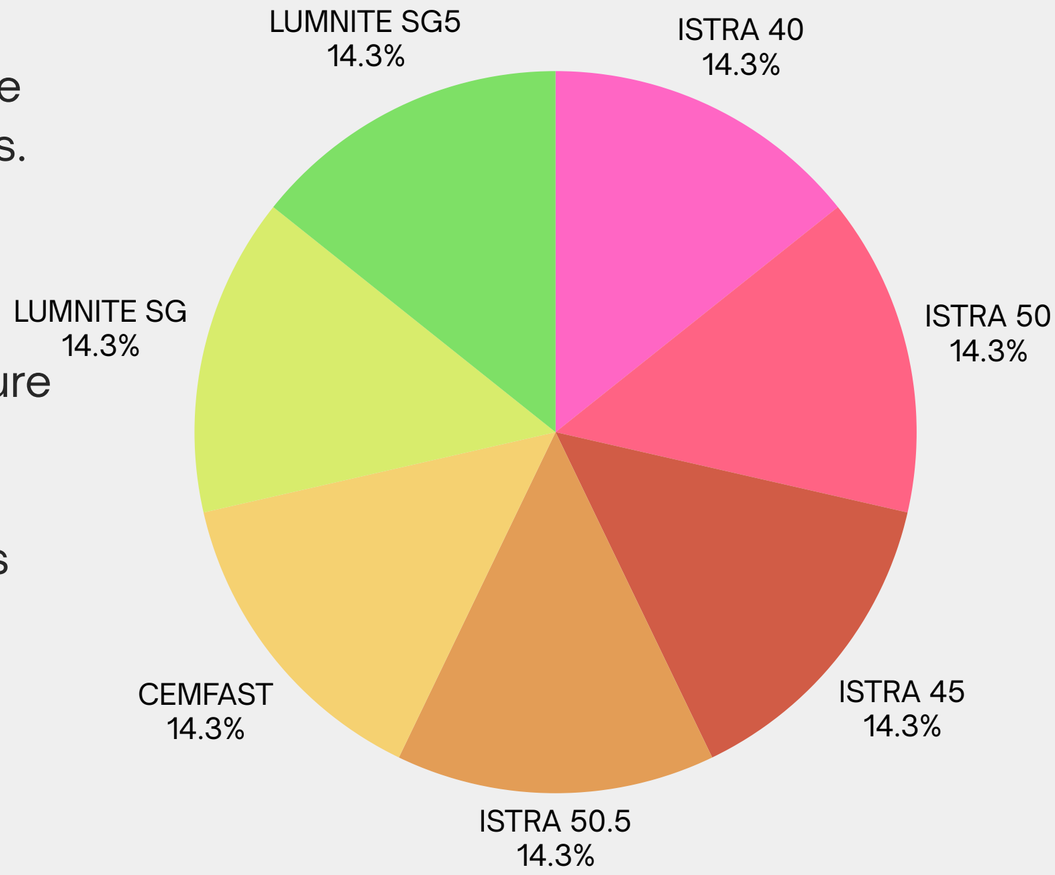
Data Augmentation

To improve model generalization and avoid bias from small sample sizes, we augmented cement types with fewer than 1000 samples.

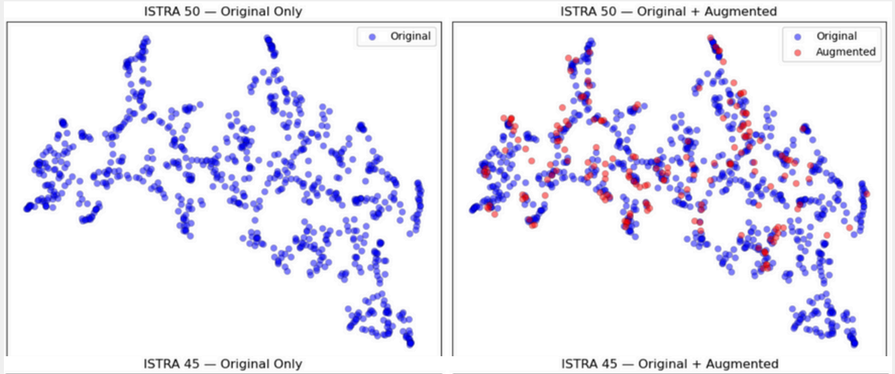
How we did it:

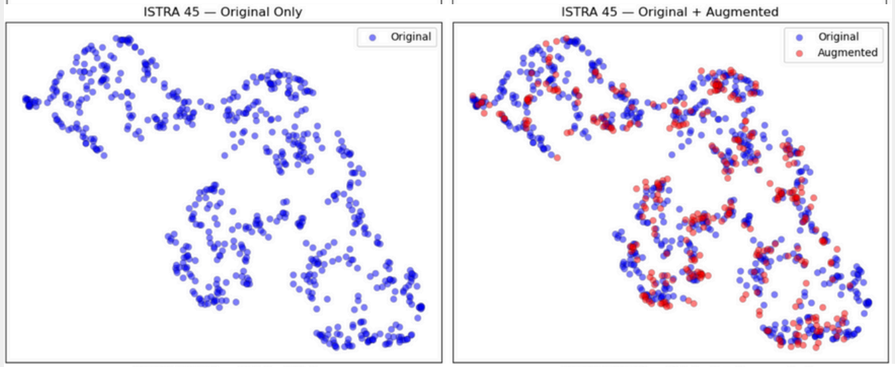
- Used K-Nearest Neighbors to interpolate new samples in feature space.
- Applied Mahalanobis distance filtering using robust estimators to remove unrealistic synthetic samples.
- Ensured the target value (init.) was predicted using the interpolation weights.

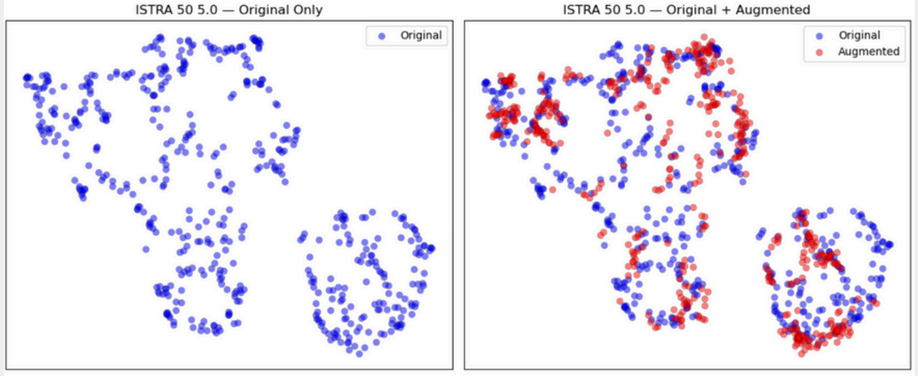
🔁 Augmentation stops once each cement type reaches 1000 samples.
✅ Result: Balanced, realistic datasets ready for modeling.

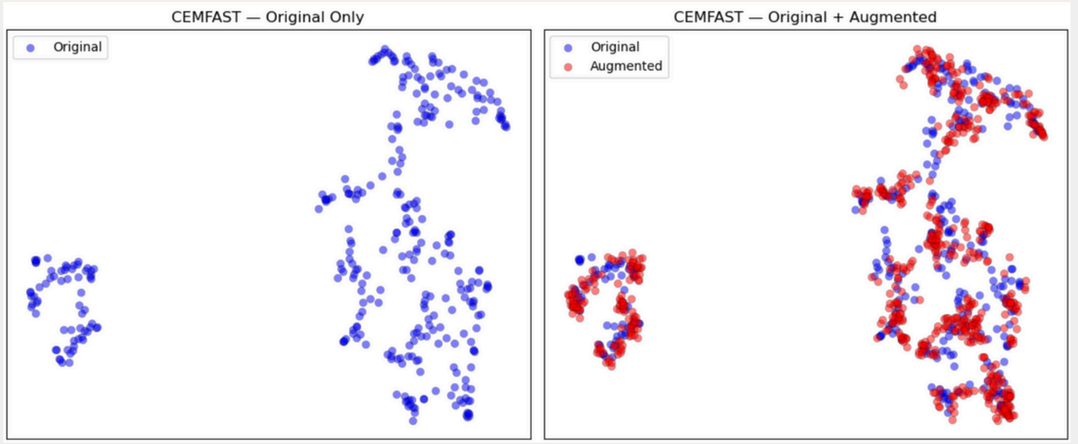


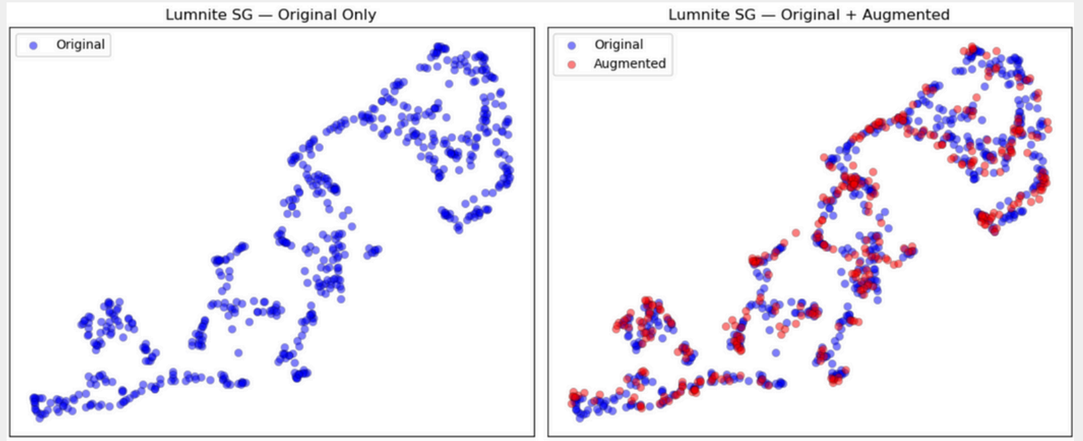
Before & After MVD

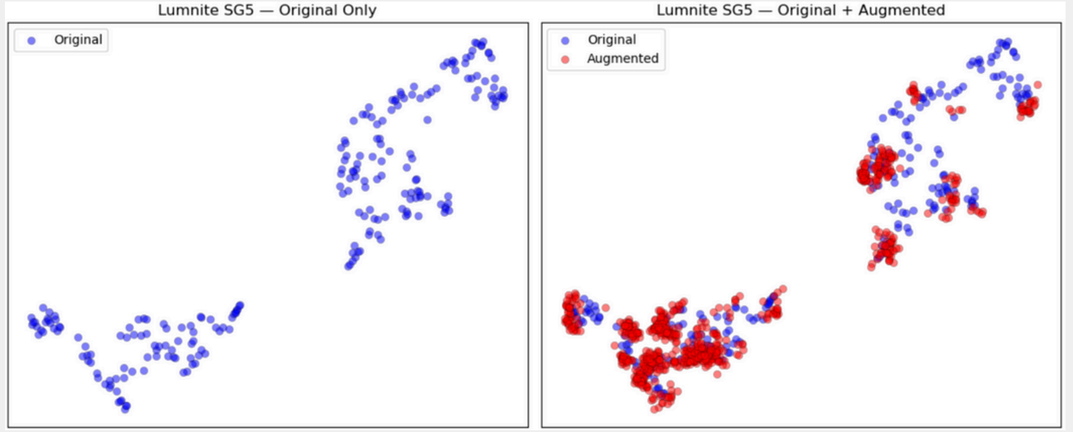




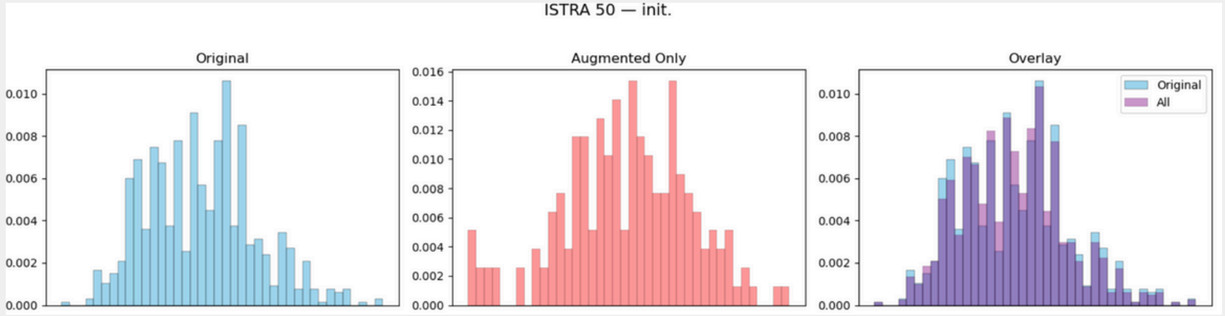


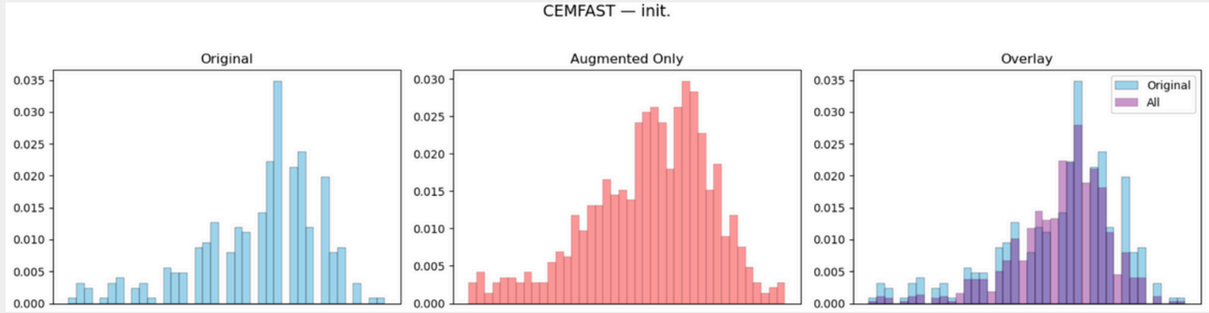


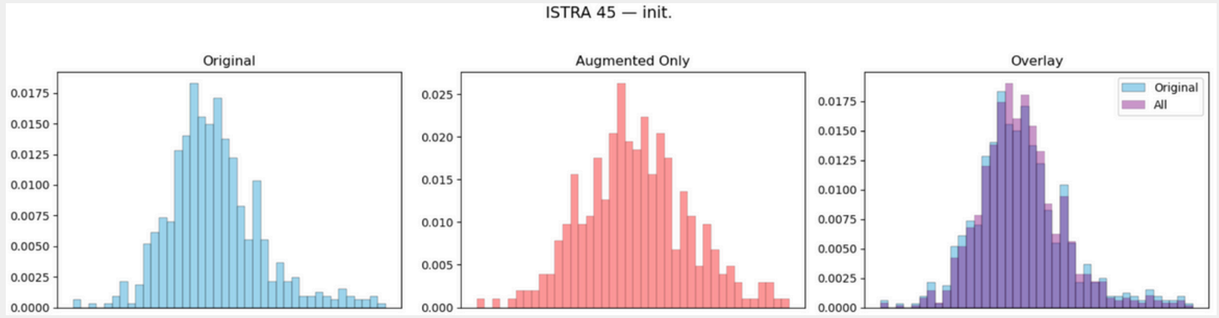


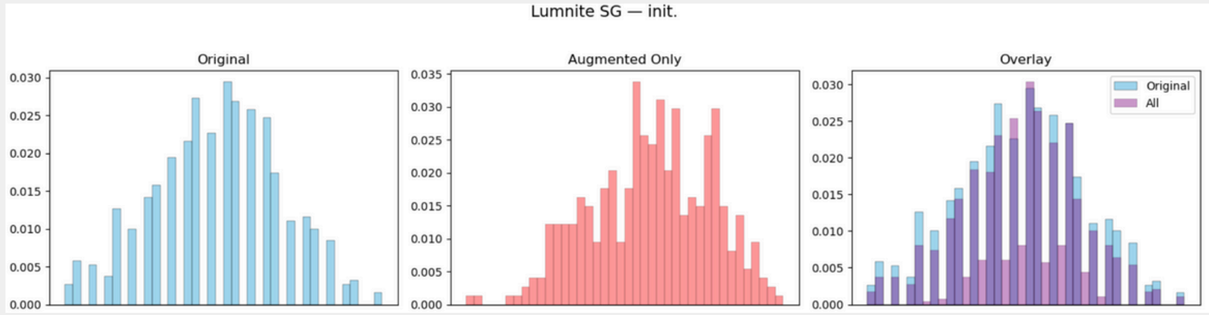


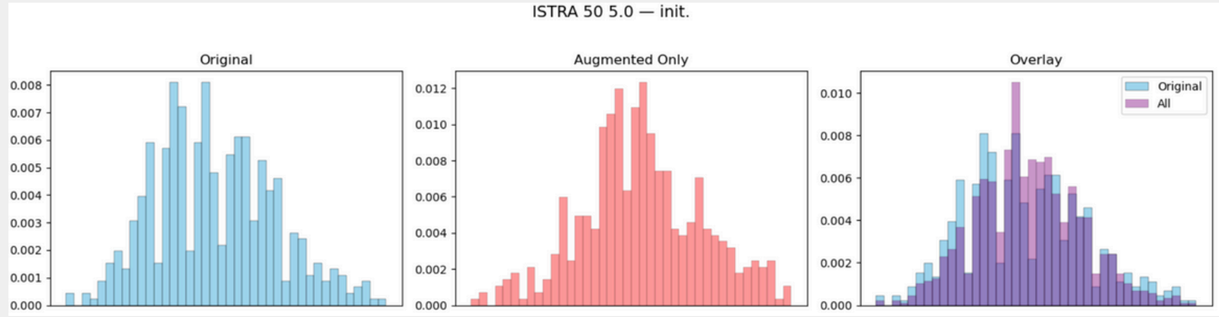
Before & After “init.”

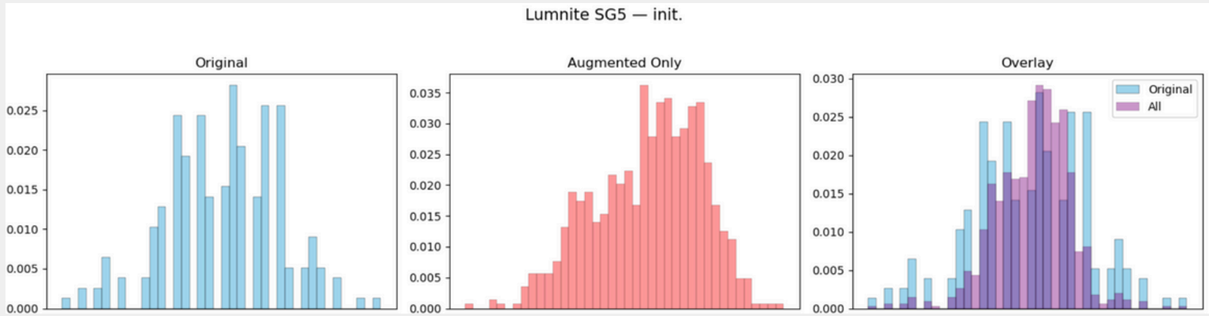




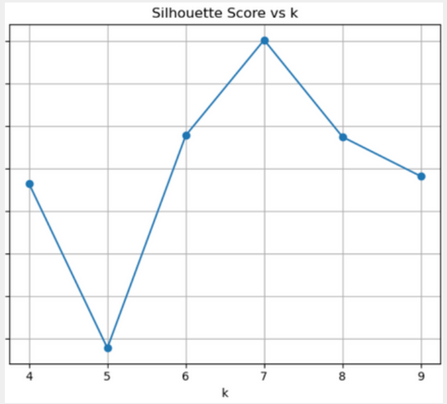


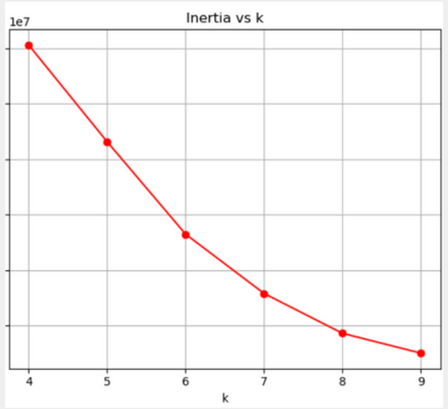


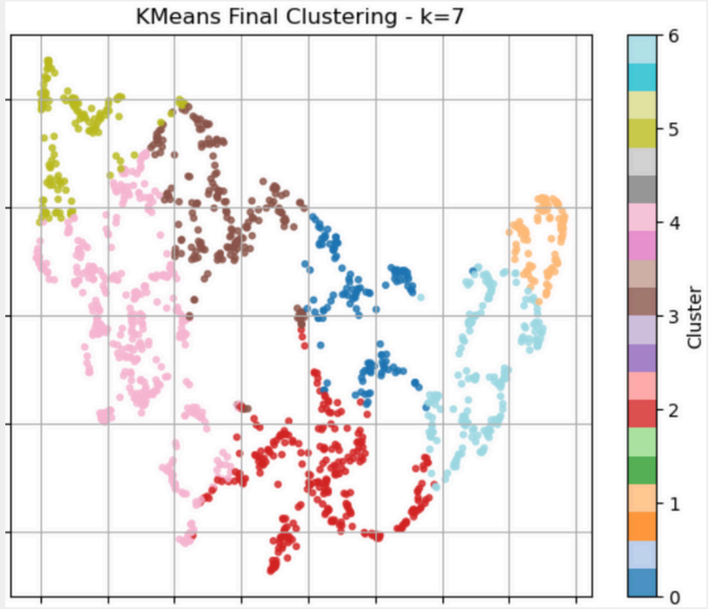


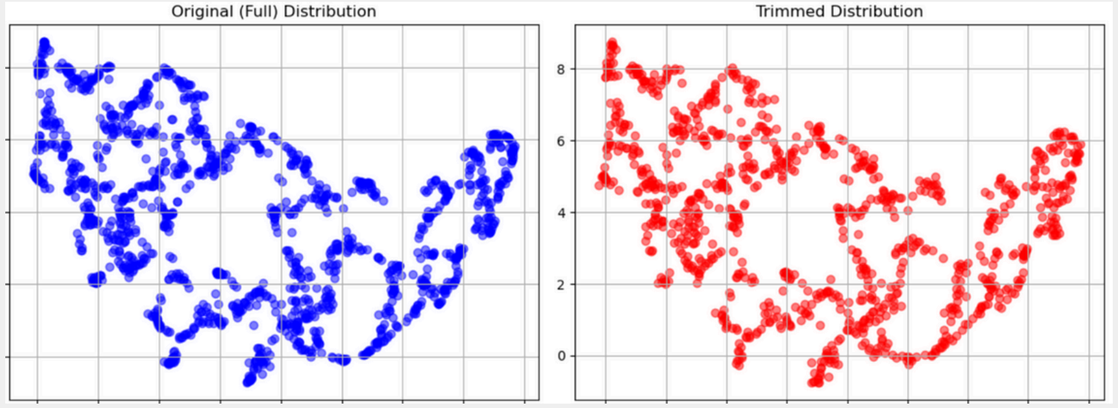


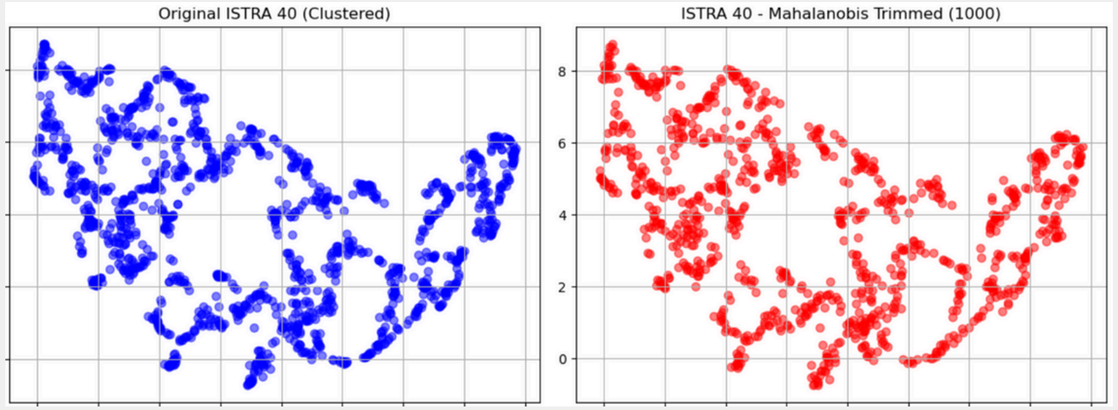
Data Trimming for ISTRA 40 (1451->1000)











- Random-Stratified
  - Mean Univariate Wasserstein Distance: 0.0618 (618189832.5% of baseline)
  - Mean Multivariate Mahalanobis Distance: 15.9651 (96.3% of baseline)
- Mahalanobis-Based
  - Mean Univariate Wasserstein Distance: 0.0881 (881427718.4% of baseline)
  - Mean Multivariate Mahalanobis Distance: 13.4818 (81.4% of baseline)

From Splits to a Unified Dataset

After augmentation and filtering, we concatenated all cement types into unified training and test sets.

To preserve the cement type and production process context, we applied **One-Hot Encoding** on:

- Cement types (Type)
- Production processes (Process)

This allows the models to learn shared patterns across types while preserving categorical identity.

Enhancing Features Power

In addition to raw features, we created new informative features:

- ILR-transformed Chem and Phase components (to map from simplex to Euclidean space).
- Ratios between chemical and phase features.
- Domain-informed aggregations (e.g., chem sums, phase ratios).

These enriched features help models learn more generalizable patterns.

Scaling for Generalization and Privacy

We used QuantileTransformer with normal output distribution to:

- Standardize feature scales,
- Obfuscate raw values for privacy,
- Make learning easier for neural models.

All features except One-Hot Encoded columns were scaled.

Scaling was applied after augmentation and enhancement, on the entire concatenated dataset.

Final Datasets Overview

Dataset Name	Augmented	Raw Chem & Phase	ILR Chem & Phase	Ratios & New Features	Scaled
cement_concat_raw	❌	✅	❌	❌	✅
cement_concat_enhanced	✅	❌	✅	✅	✅

AND WE ARE DONE WITH FEATURE ENGINEERING!

MLP1

- Hidden Layers: **1**
- Hidden Neurons: **32**
- Activation function: **ReLU**
- Loss Function: **MSELoss**
- Optimizer: SGD with Momentum (lr = 0.001, momentum = 0.9)
- Epochs: 300

MLP2

- Hidden Layers: 3
- Hidden Neurons Layer 1 : **128**
- Hidden Neurons Layer 2 : **64**
- Hidden Neurons Layer 2 : **32**
- Activation function: **GELU**
- Normalization: **BatchNorm**
- Loss Function: **MSELoss**
- Optimizer: **Adam**
- Learning Rate: **0.0005**
- **L2 regularization**
- Learning Rate Scheduler: **ReduceLROnPlateau**
- Weight Initialization: **Xavier**
- Epochs: **400**

Autoencoder

Objective:
Improve feature representation using dimensionality reduction (PCA & Autoencoders) on cement manufacturing data.

Input: 2 Datasets (Raw and Enhanced)

Steps:

1. Apply PCA block-wise to reduce redundancy based on correlation.
2. Define meaningful feature blocks based on domain (Chem_ilr, Phase_ilr, etc.).
3. Apply autoencoders block-wise to learn compressed representations.

Autoencoder Architecture

Per Block

- **Encoder:** Linear → ReLU → Dropout → Linear → ReLU → Dropout→ Bottleneck
- **Decoder:** Bottleneck → Linear → ReLU → Dropout → Linear → ReLU → Dropout → Output

Details

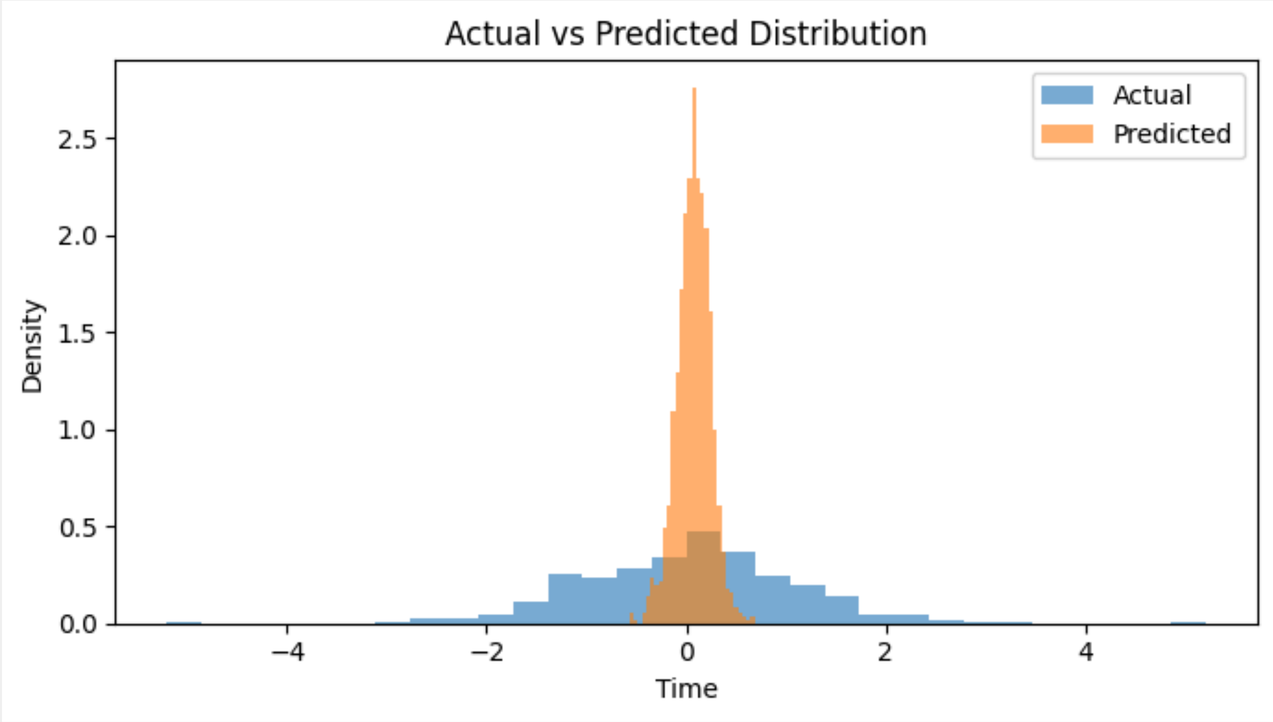
- Adaptive bottleneck size (max 16, min ½ of the input size)
- Very low dropout (0.01) for regularization
- Early stopping with patience 50
- Adaptative LR scheduler (factor 0.75)

One model per block, trained independently

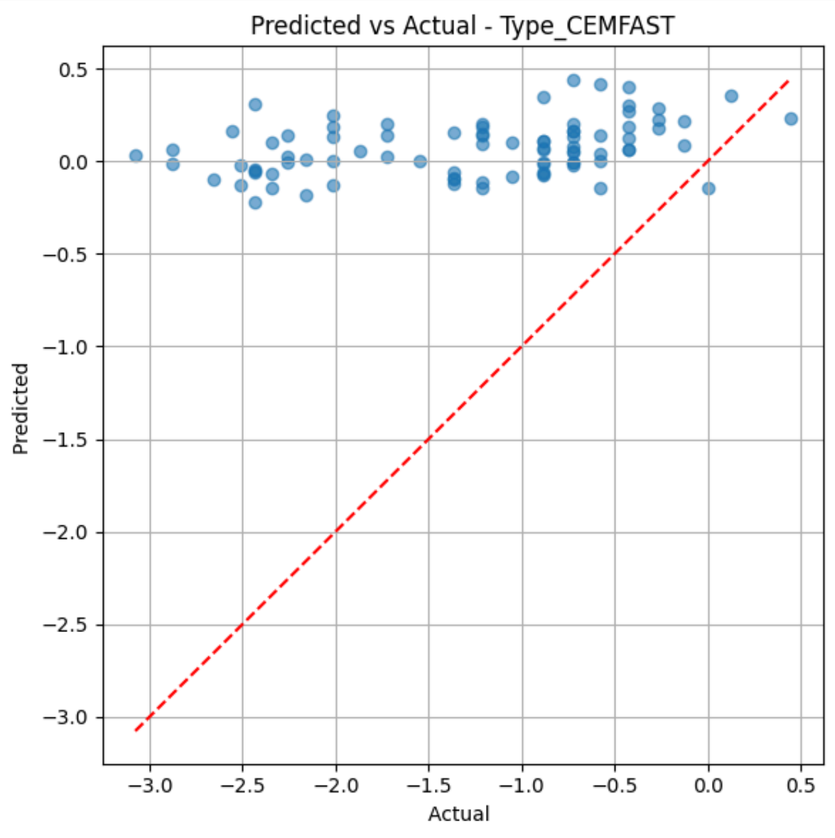
Experiments

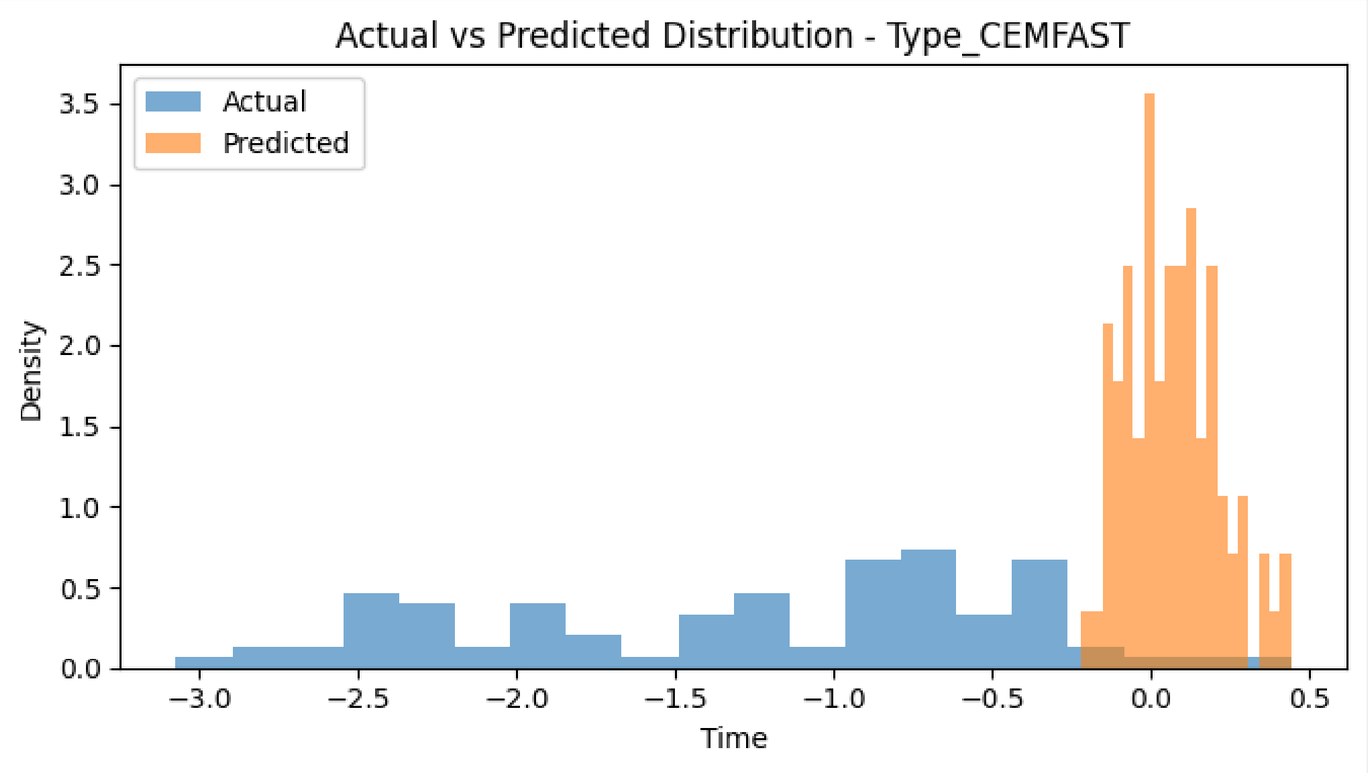
	MLP1	MLP2	Autoencoder + MLP1	Autoencoder + MLP2
Dataset1	1	2		
Dataset2	3	4	5	6

Results Experiment 1

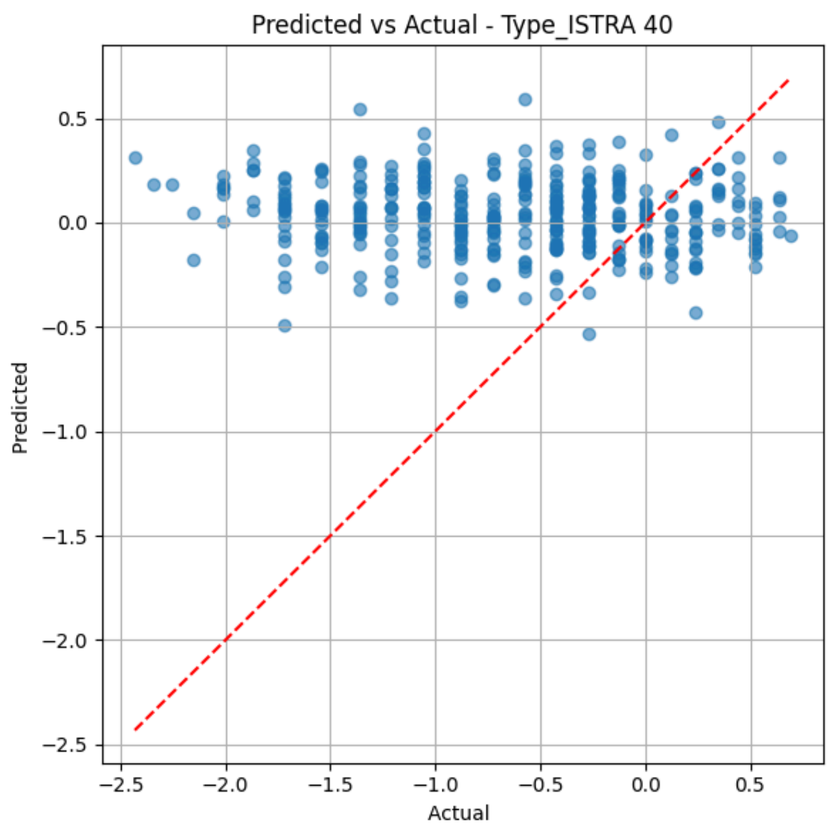


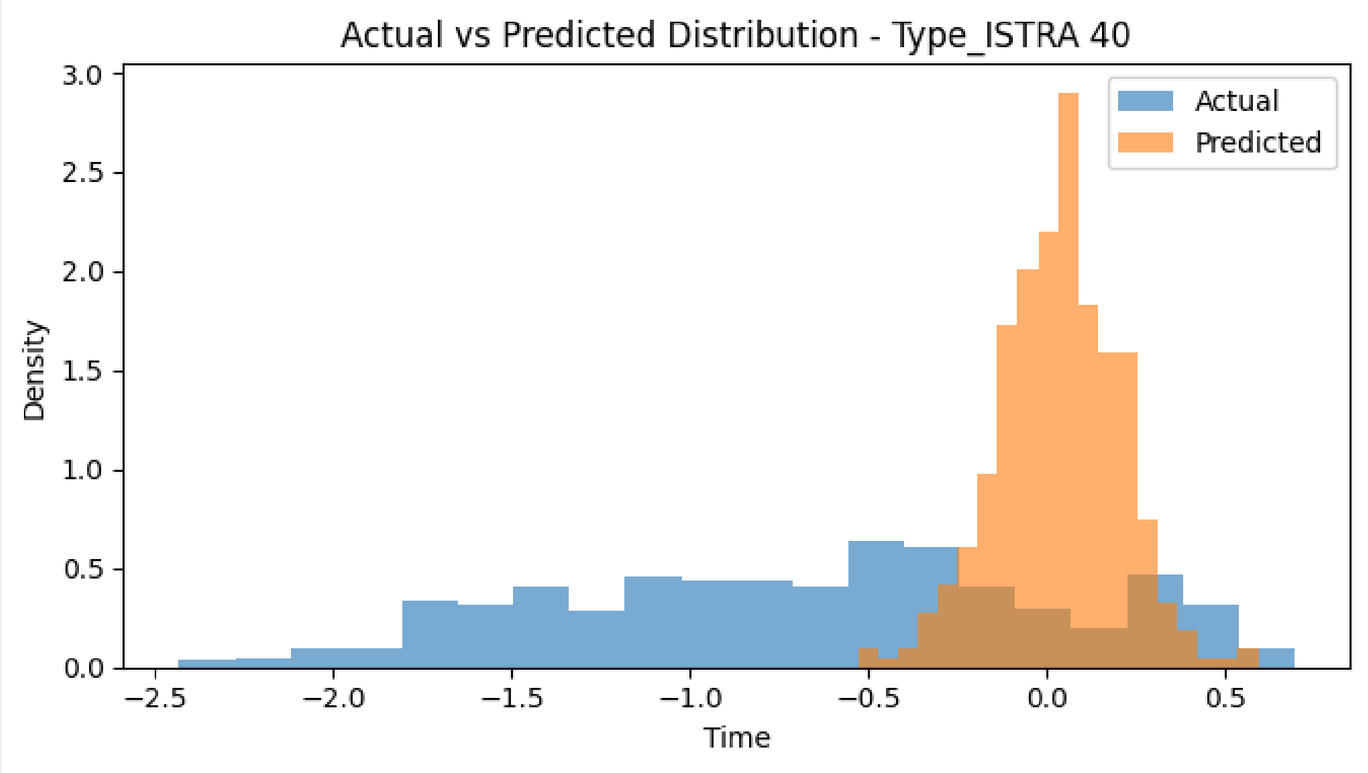
Results Experiment 1



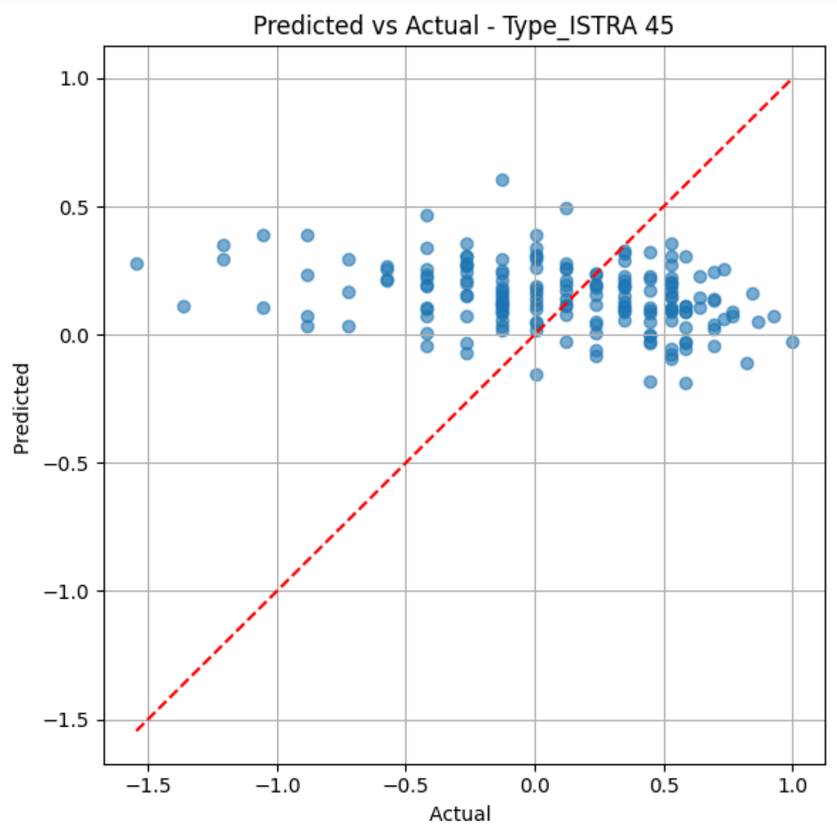


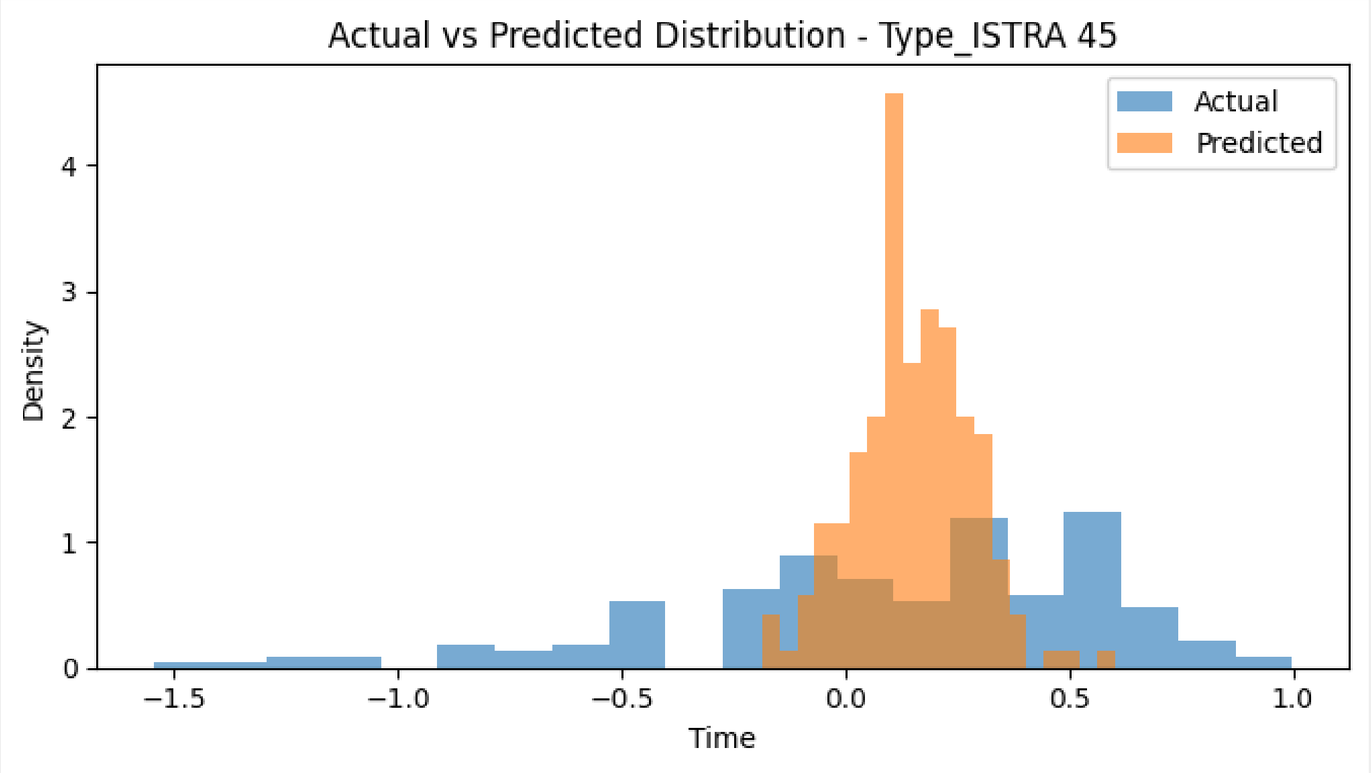
Results Experiment 1



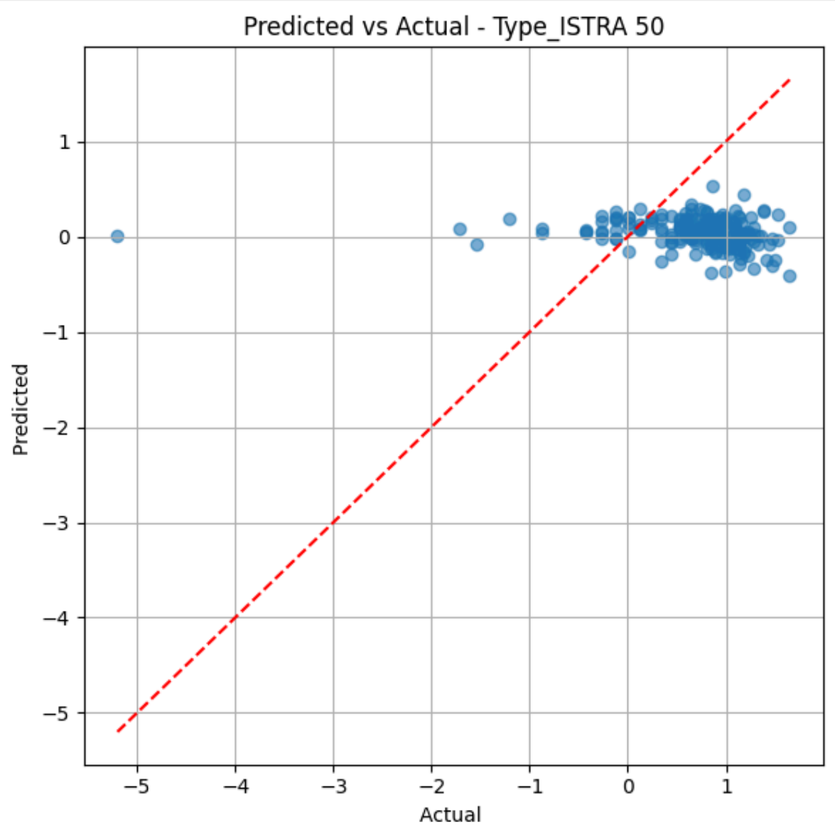


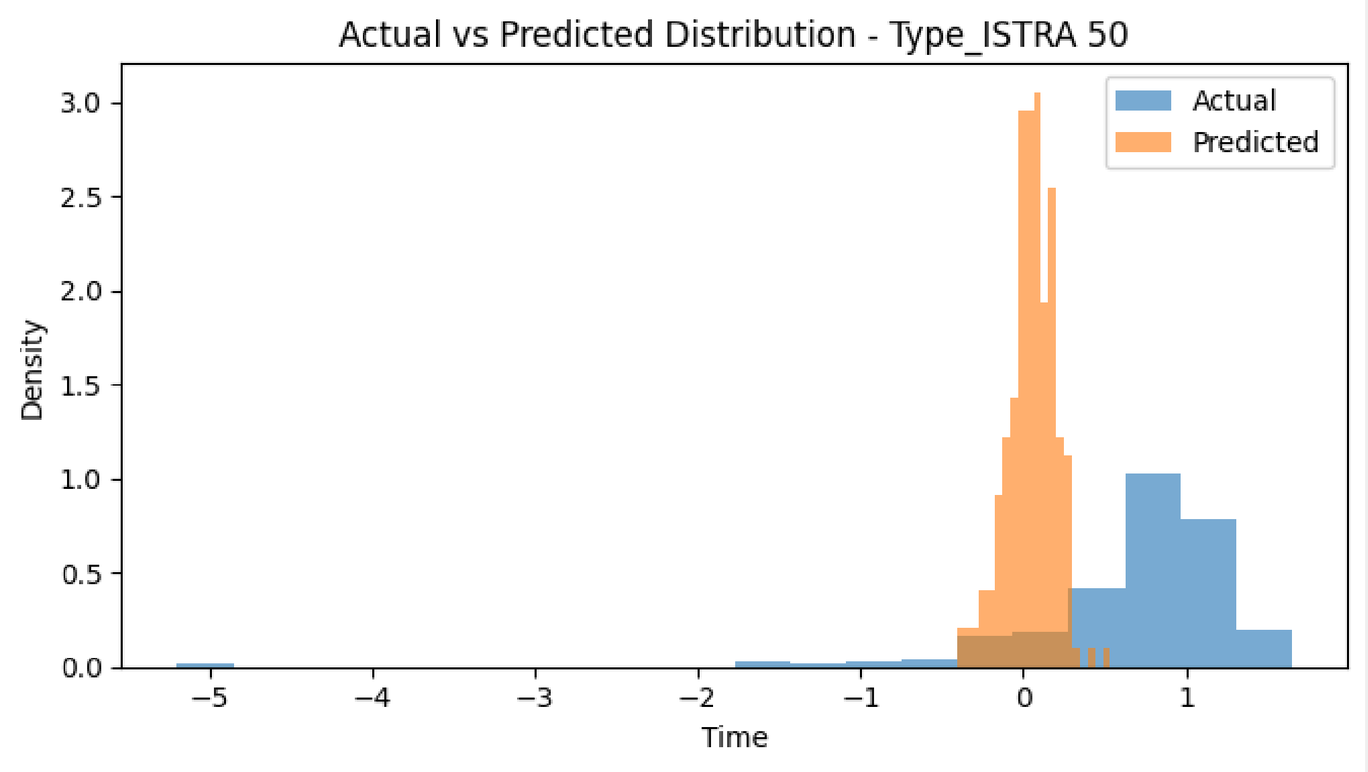
Results Experiment 1



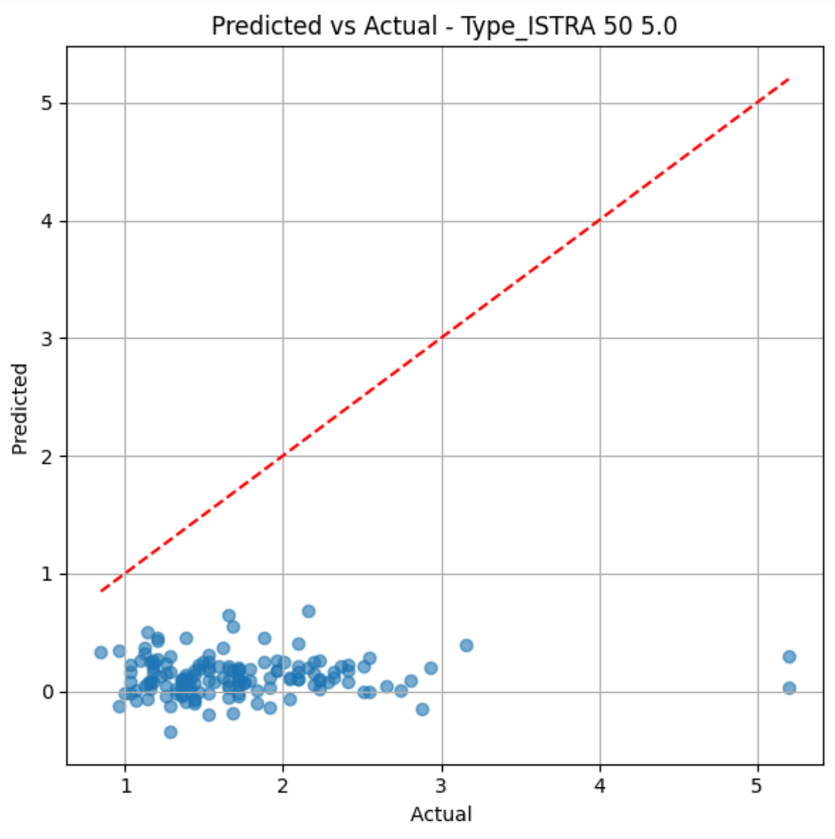


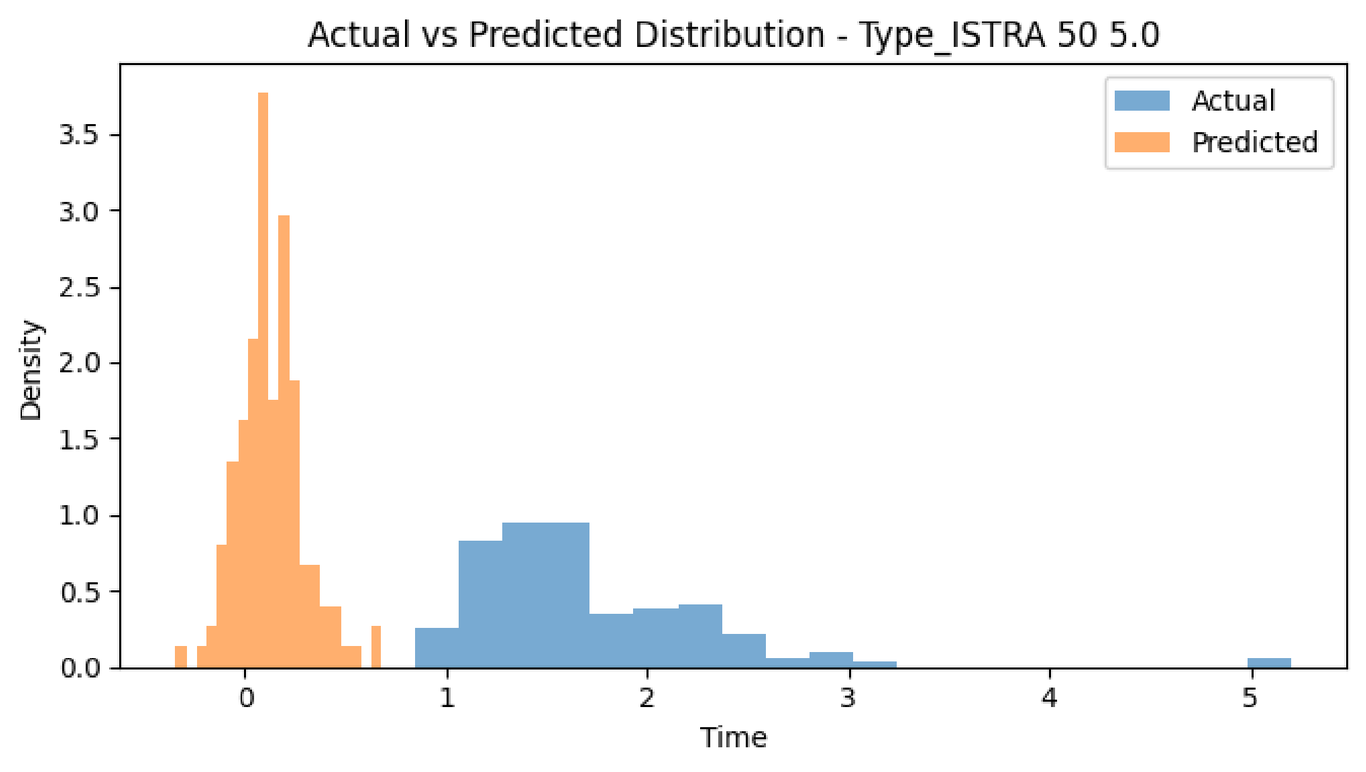
Results Experiment 1



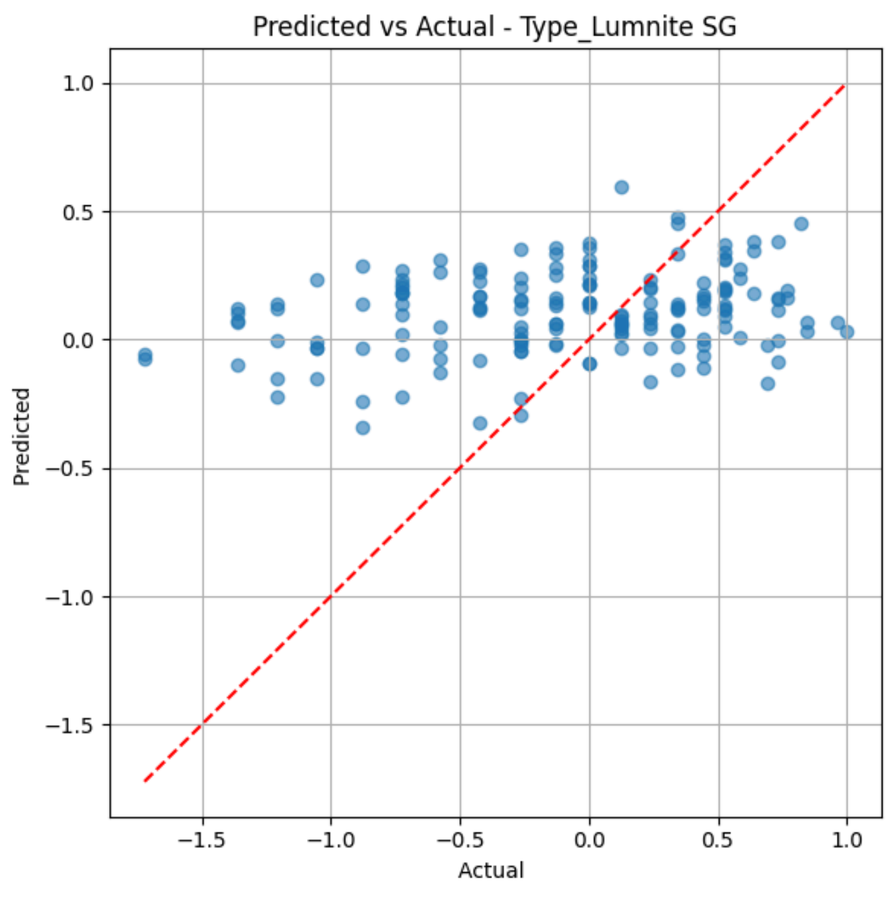


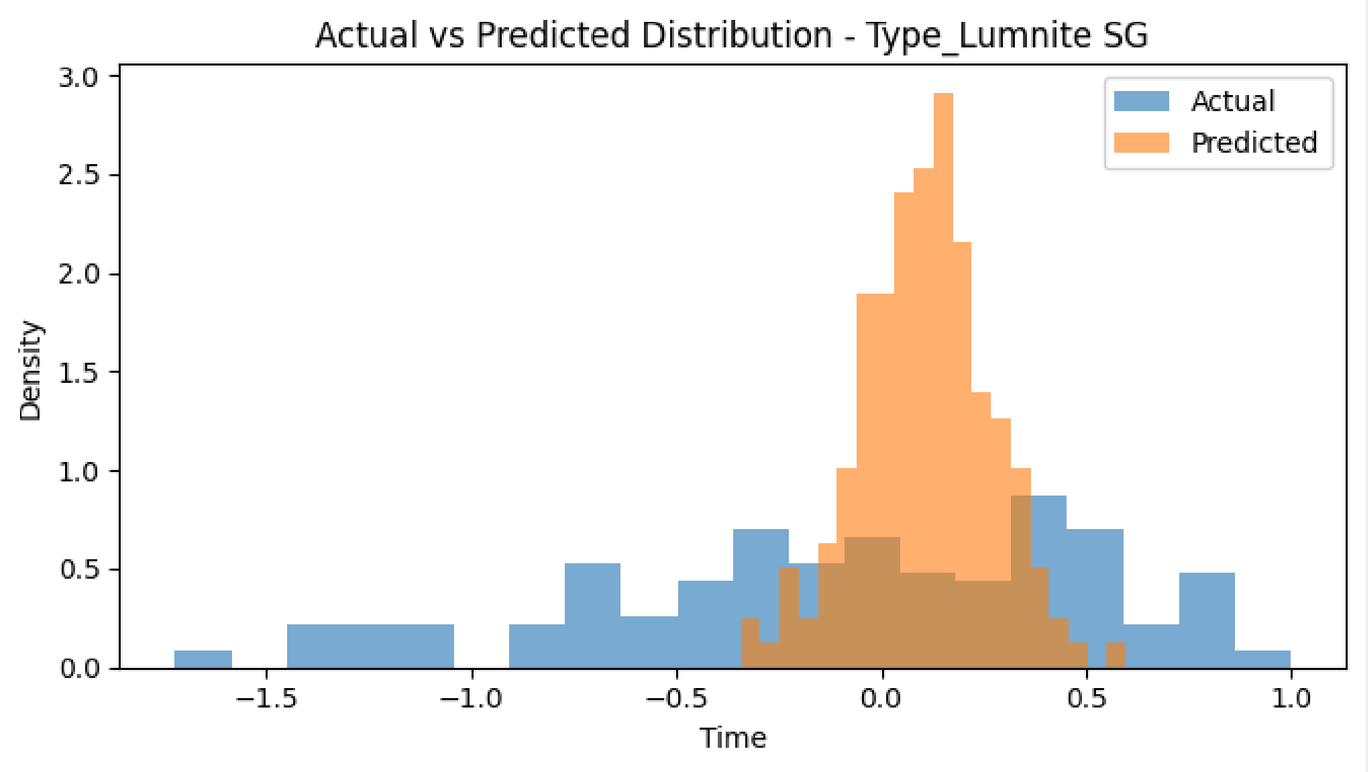
Results Experiment 1



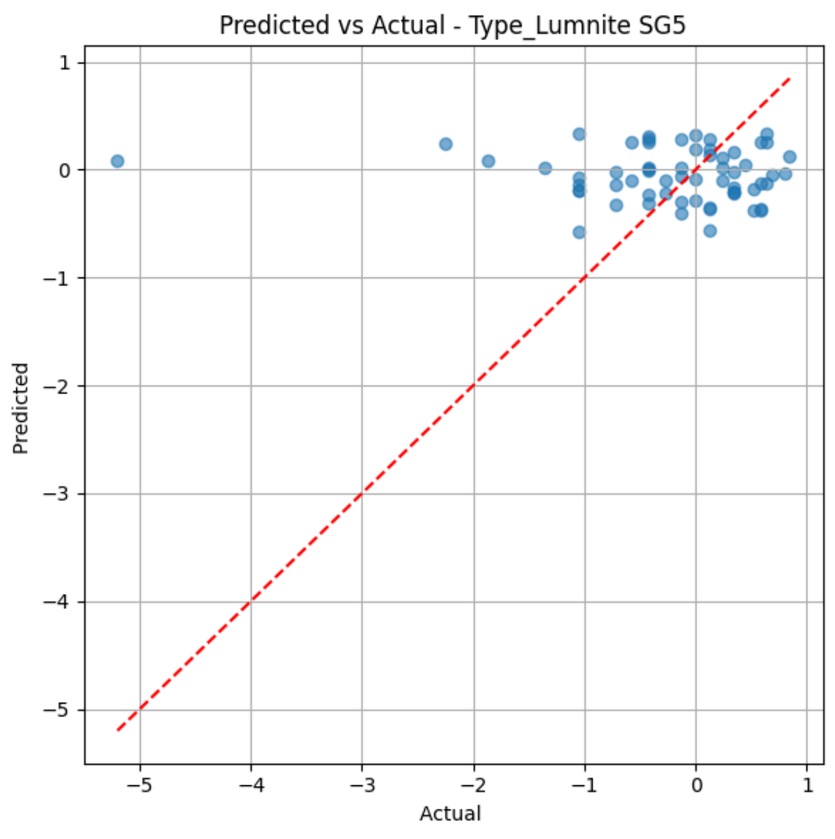


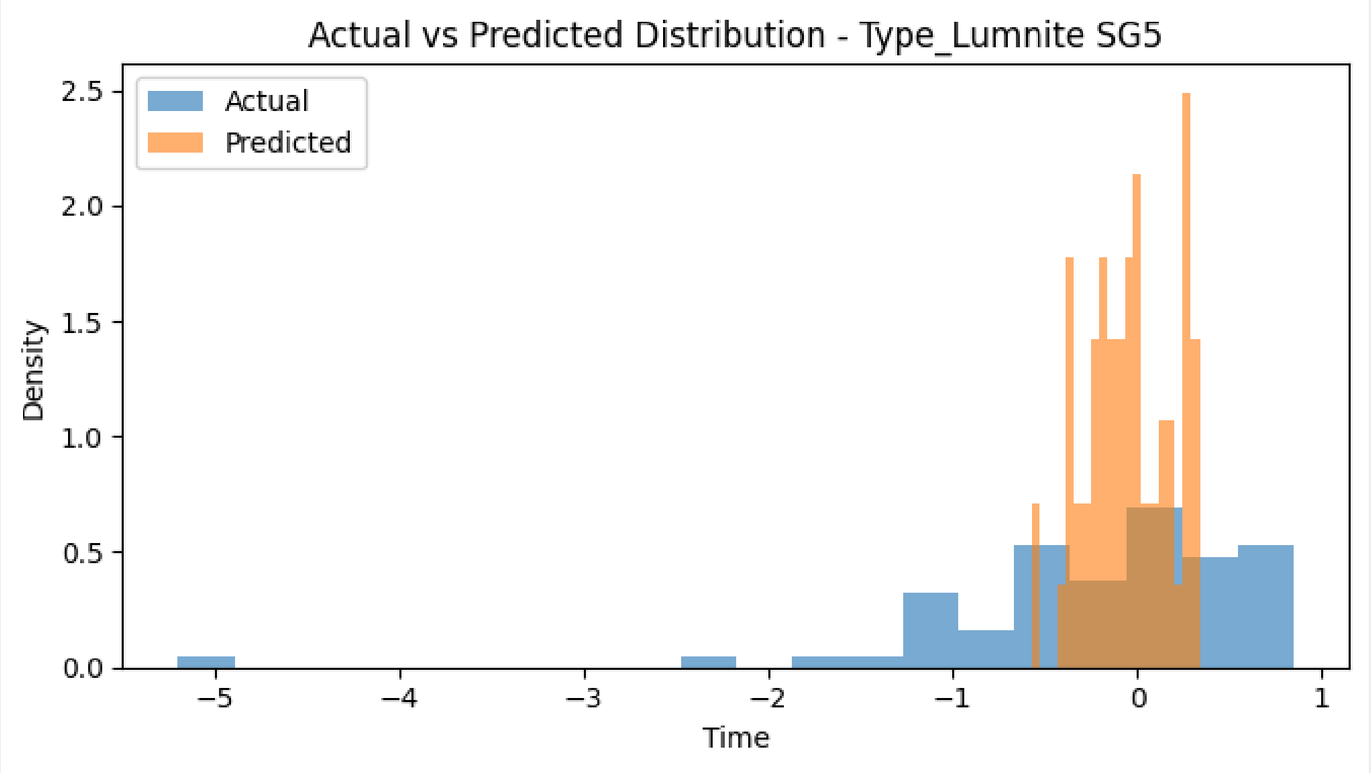
Results Experiment 1



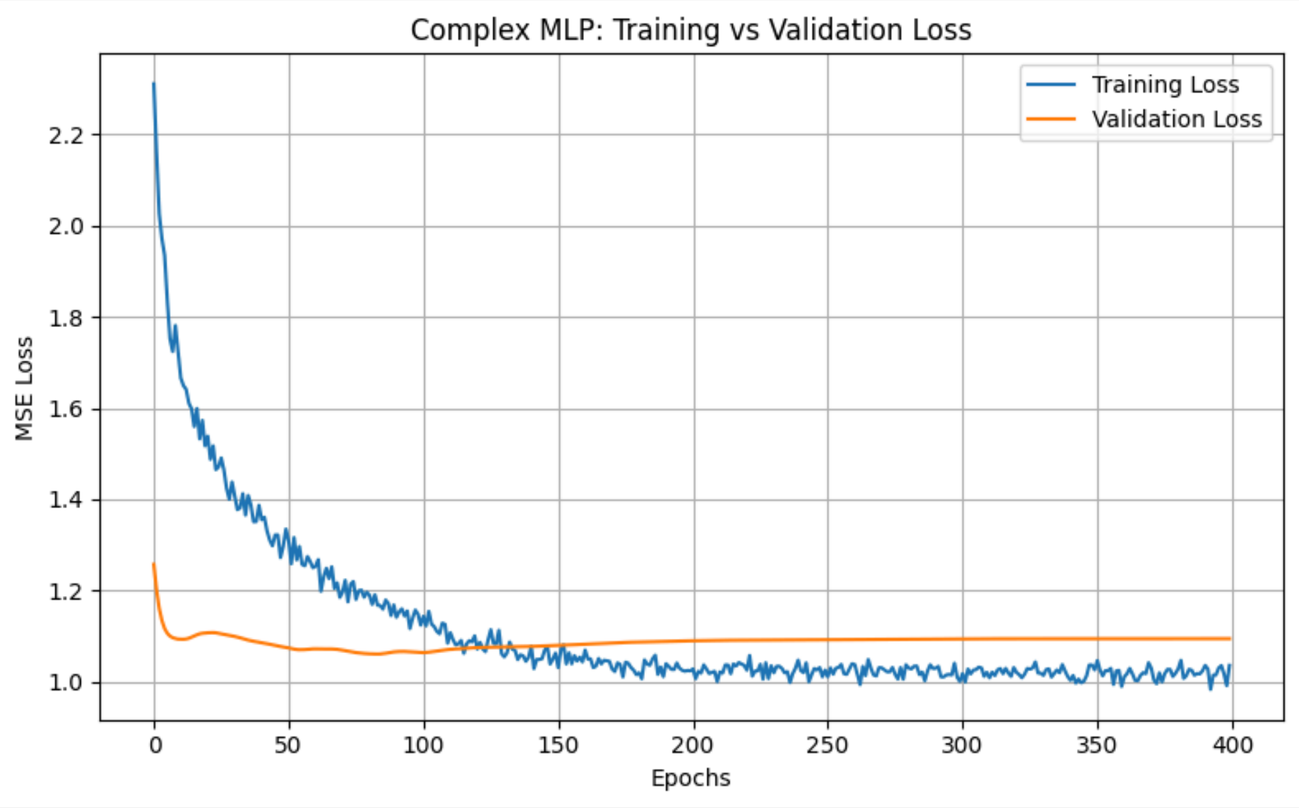


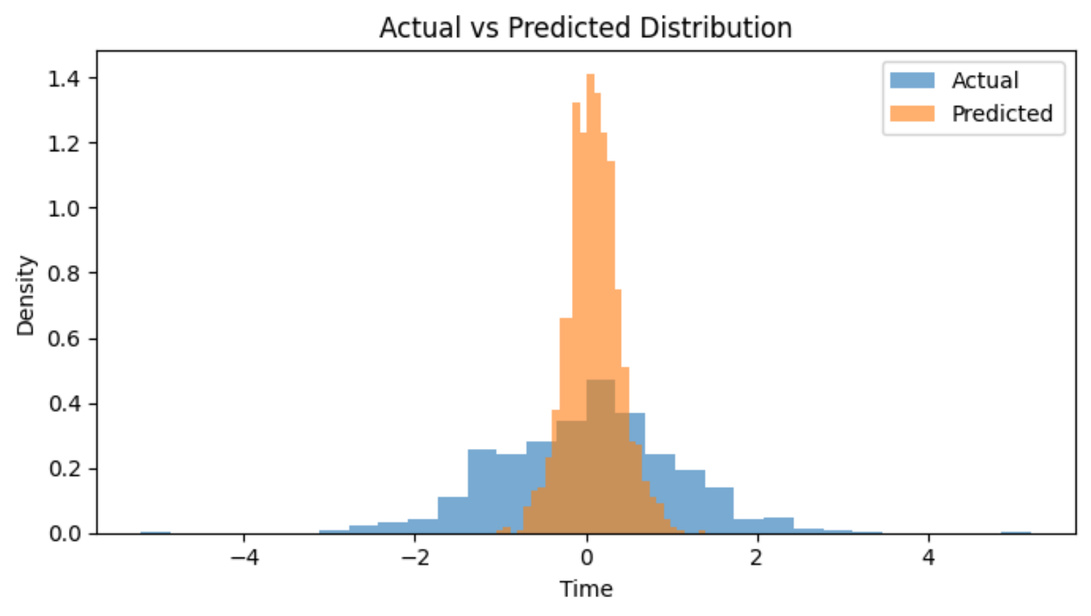
Results Experiment 1



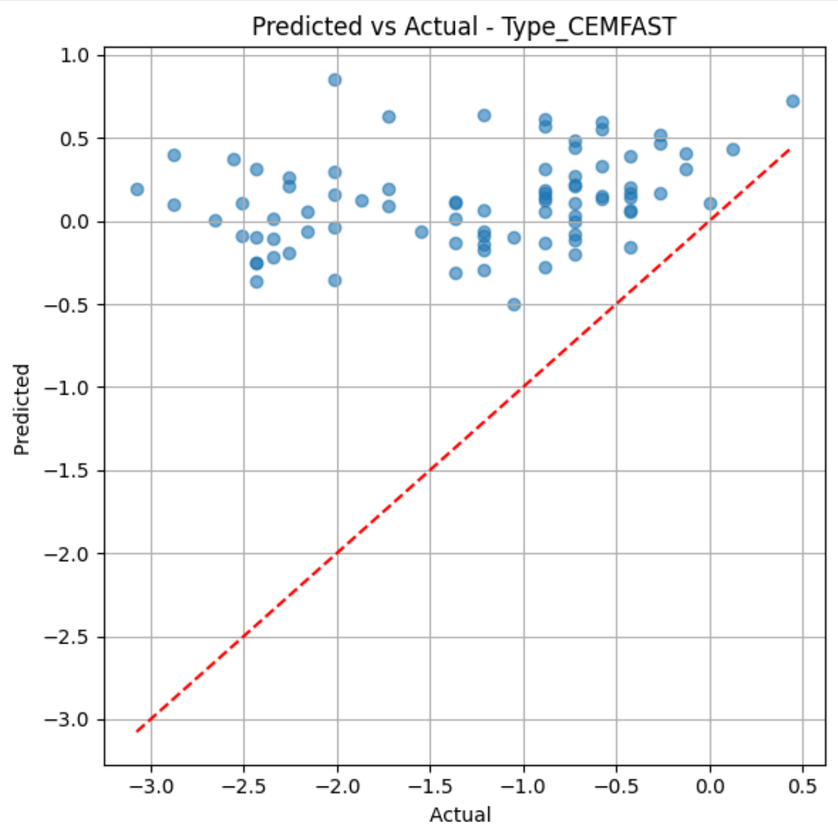


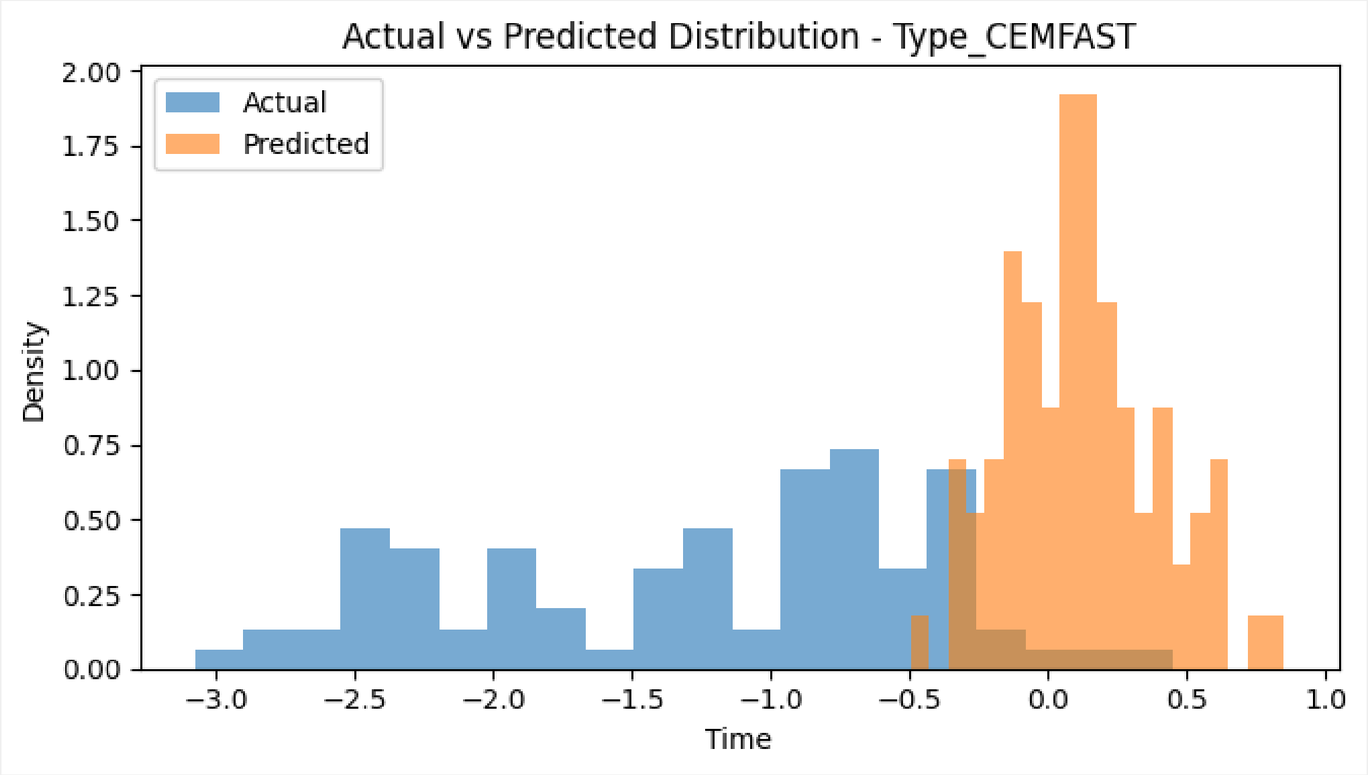
Results Experiment 2



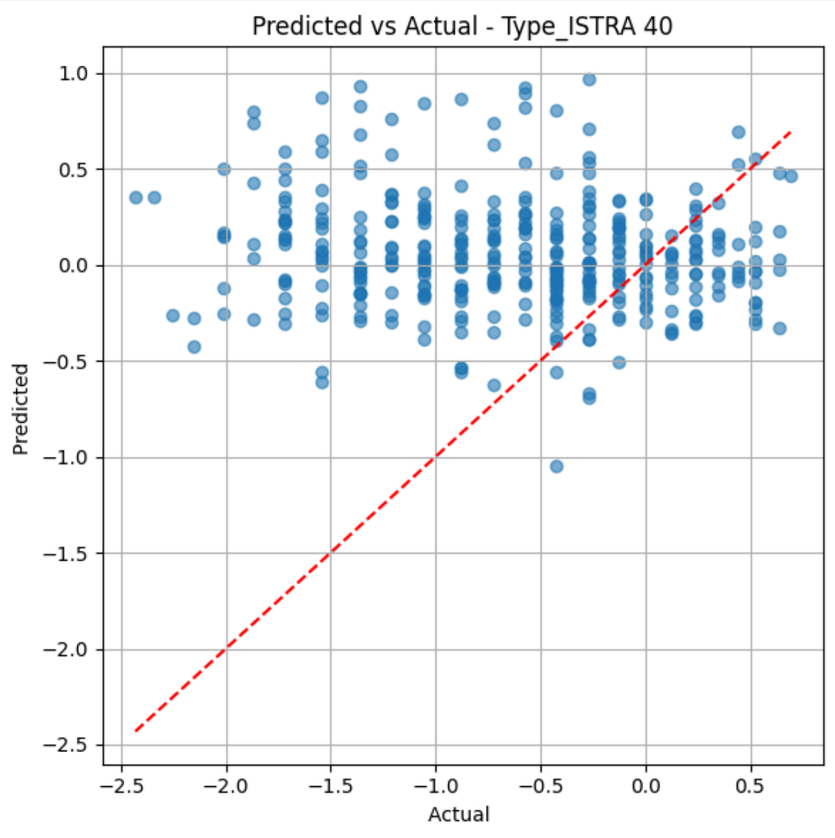


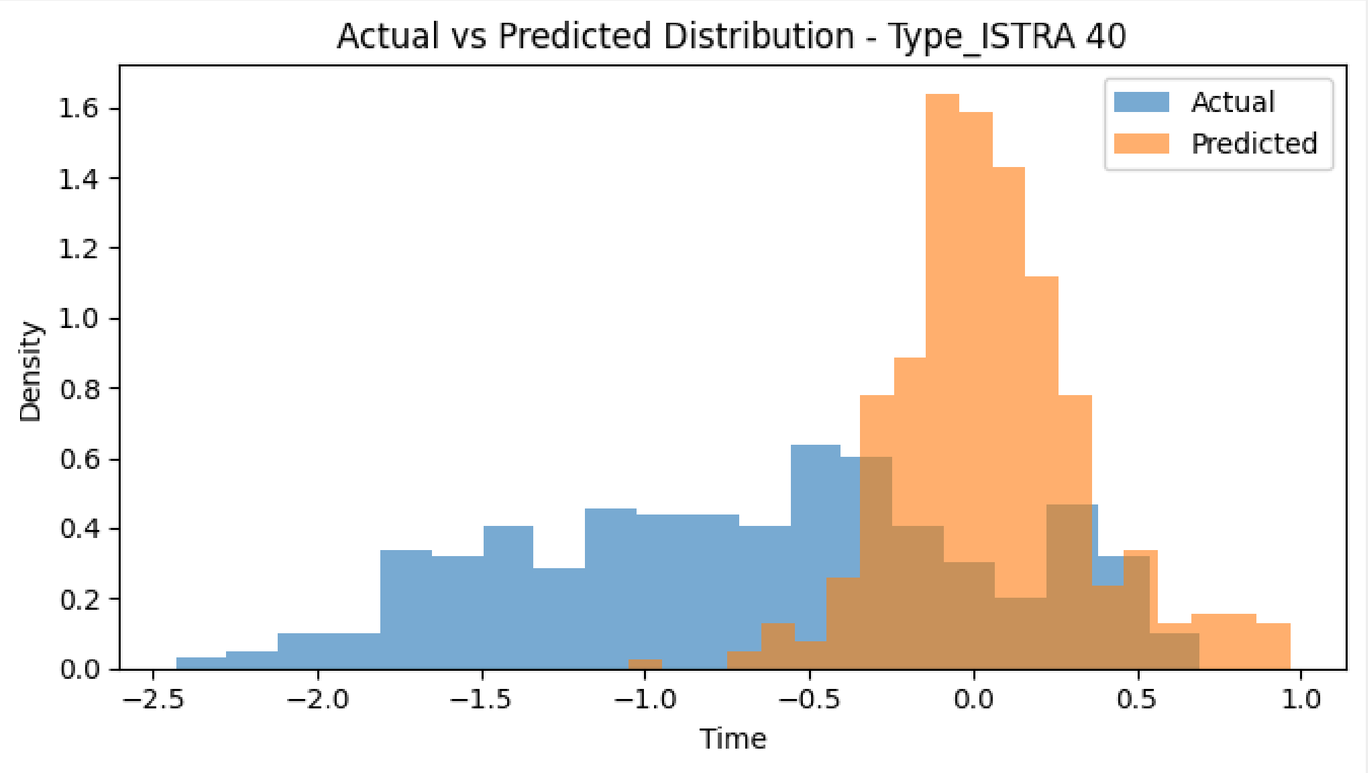
Results Experiment 2



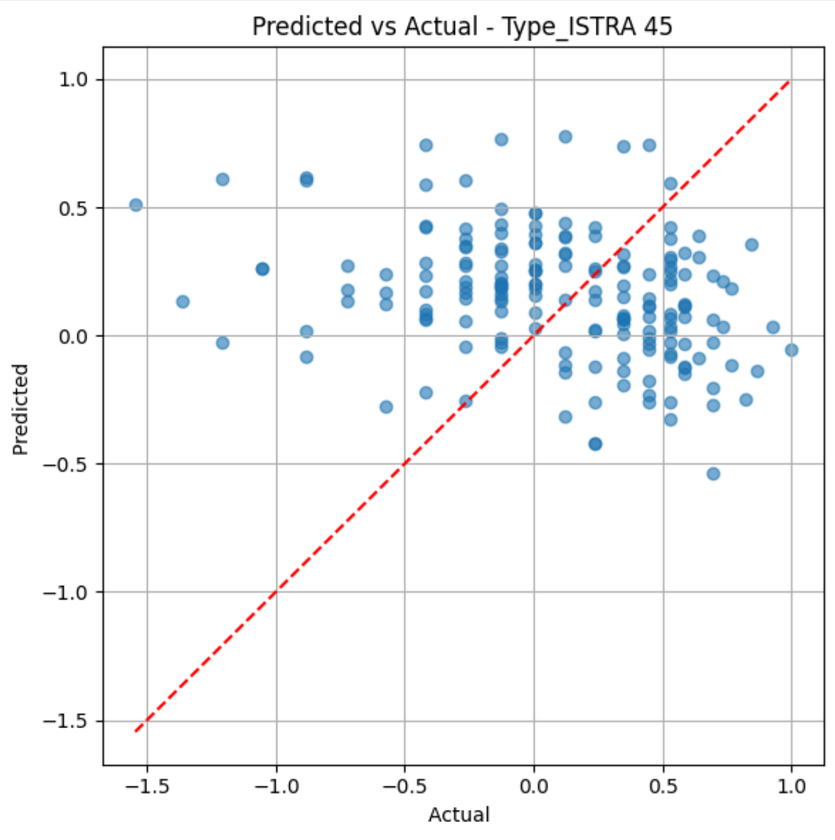


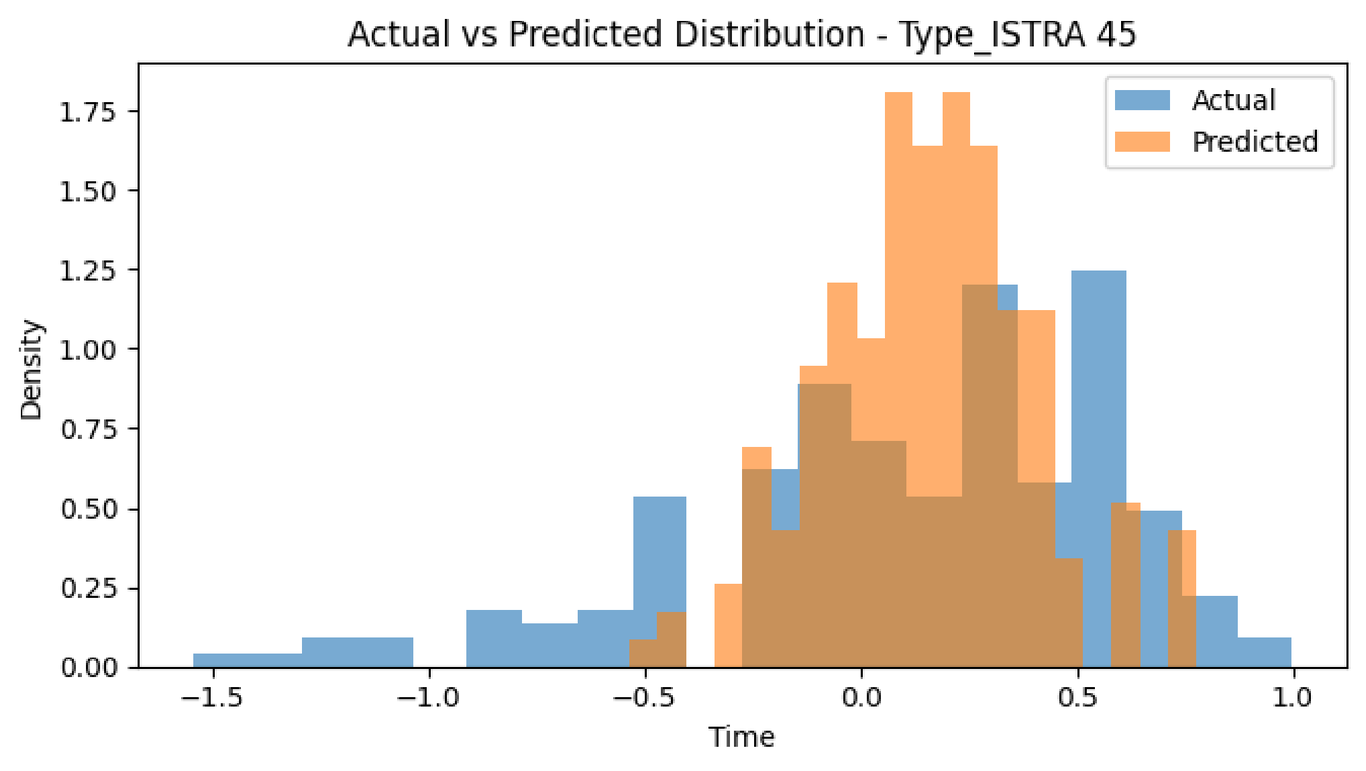
Results Experiment 2



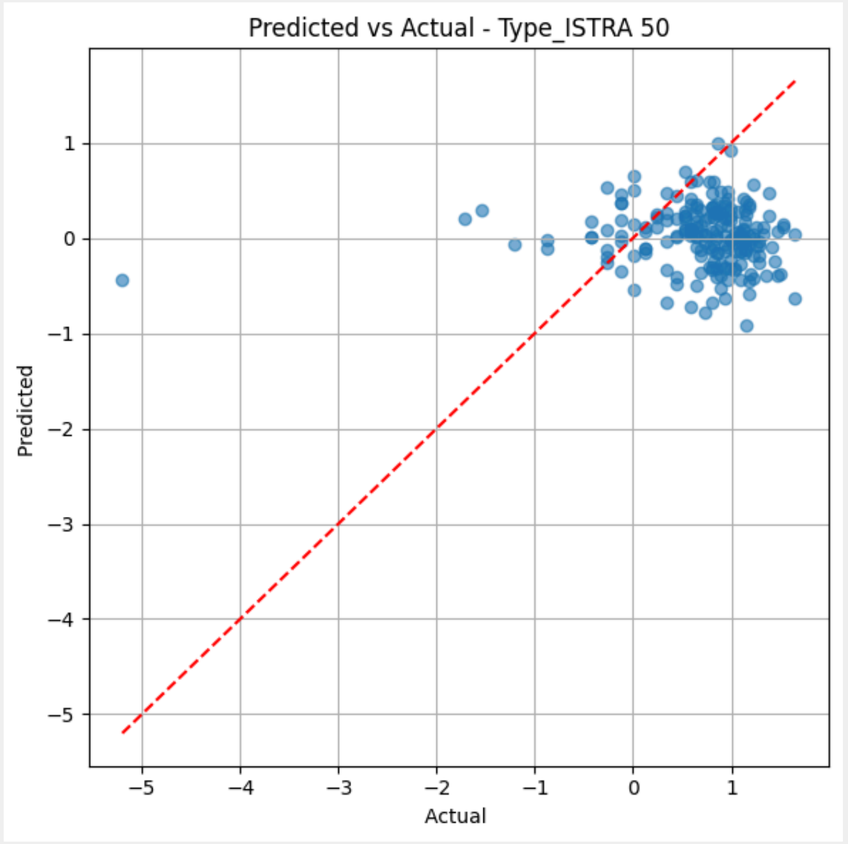


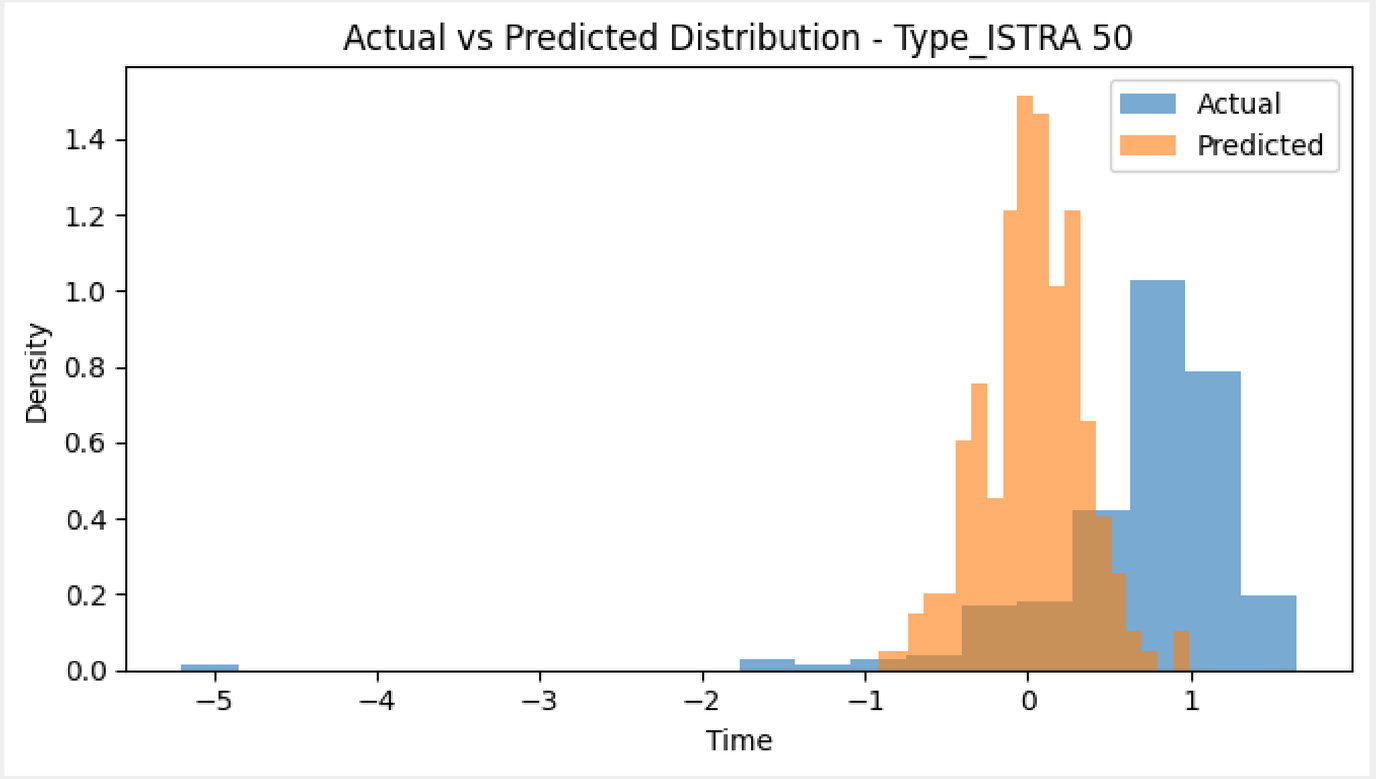
Results Experiment 2



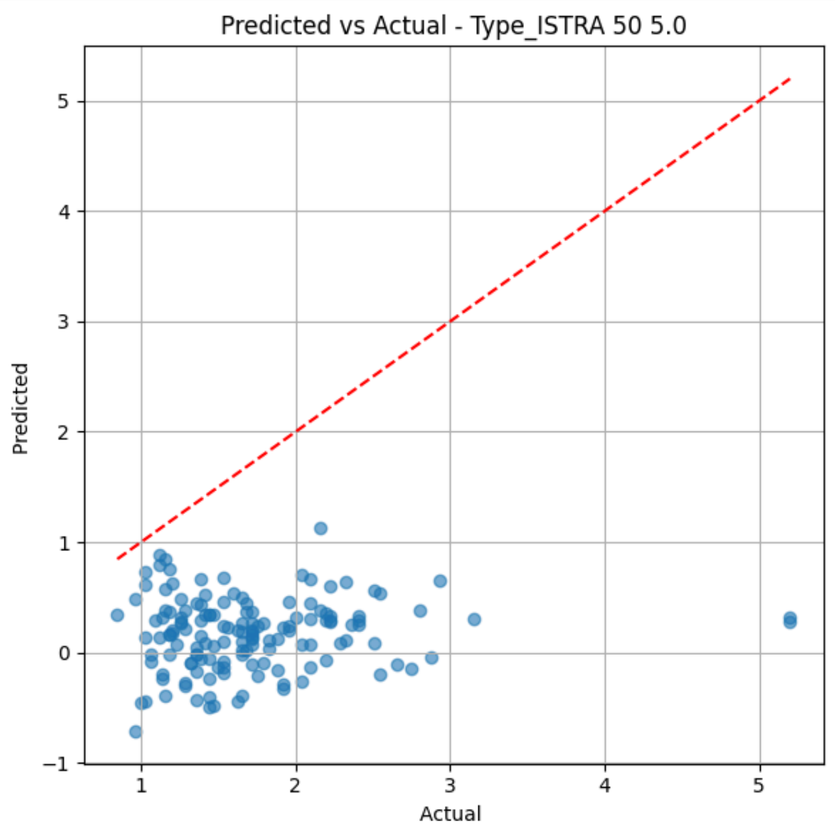


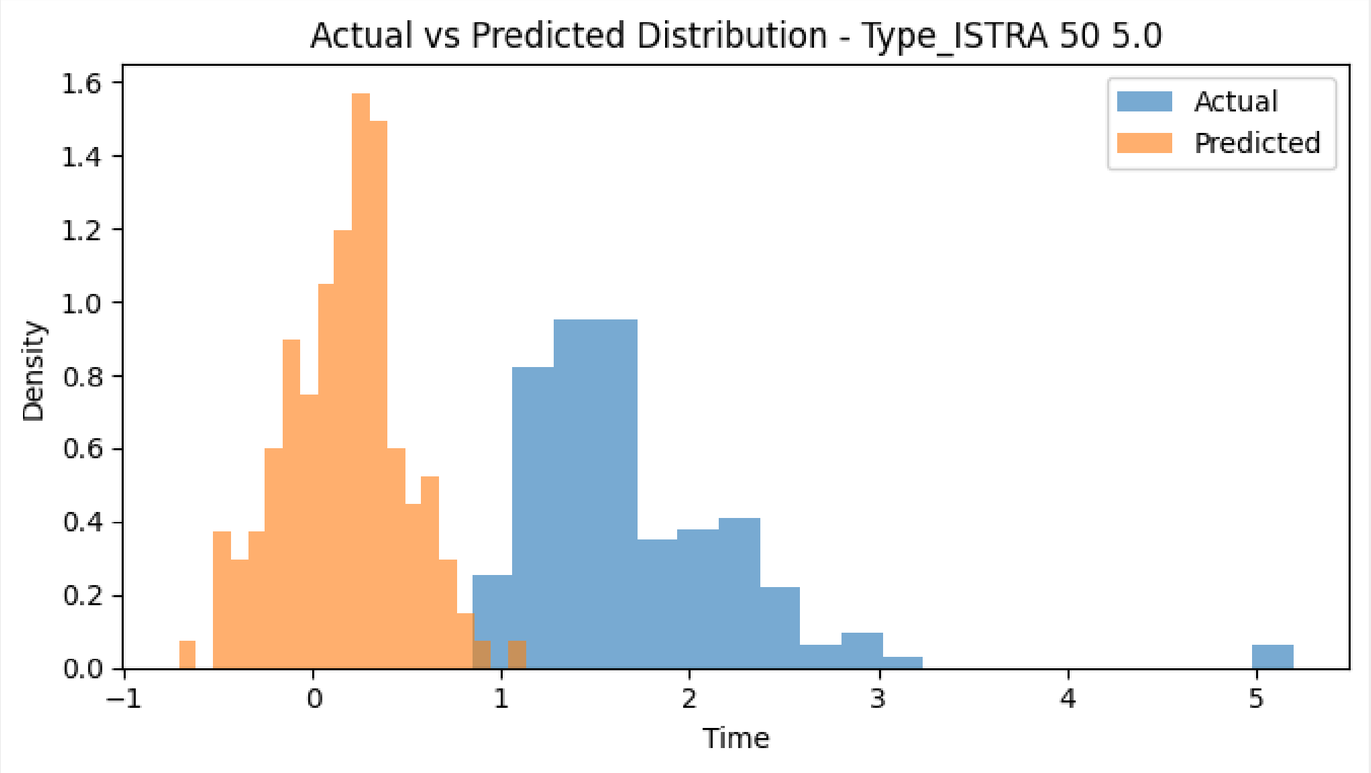
Results Experiment 2



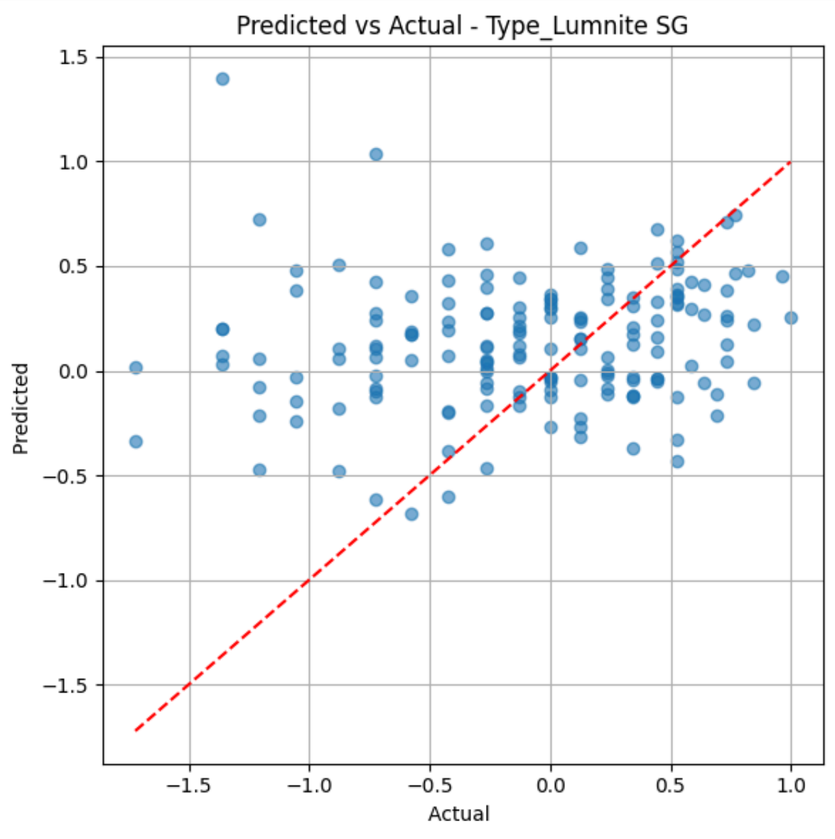


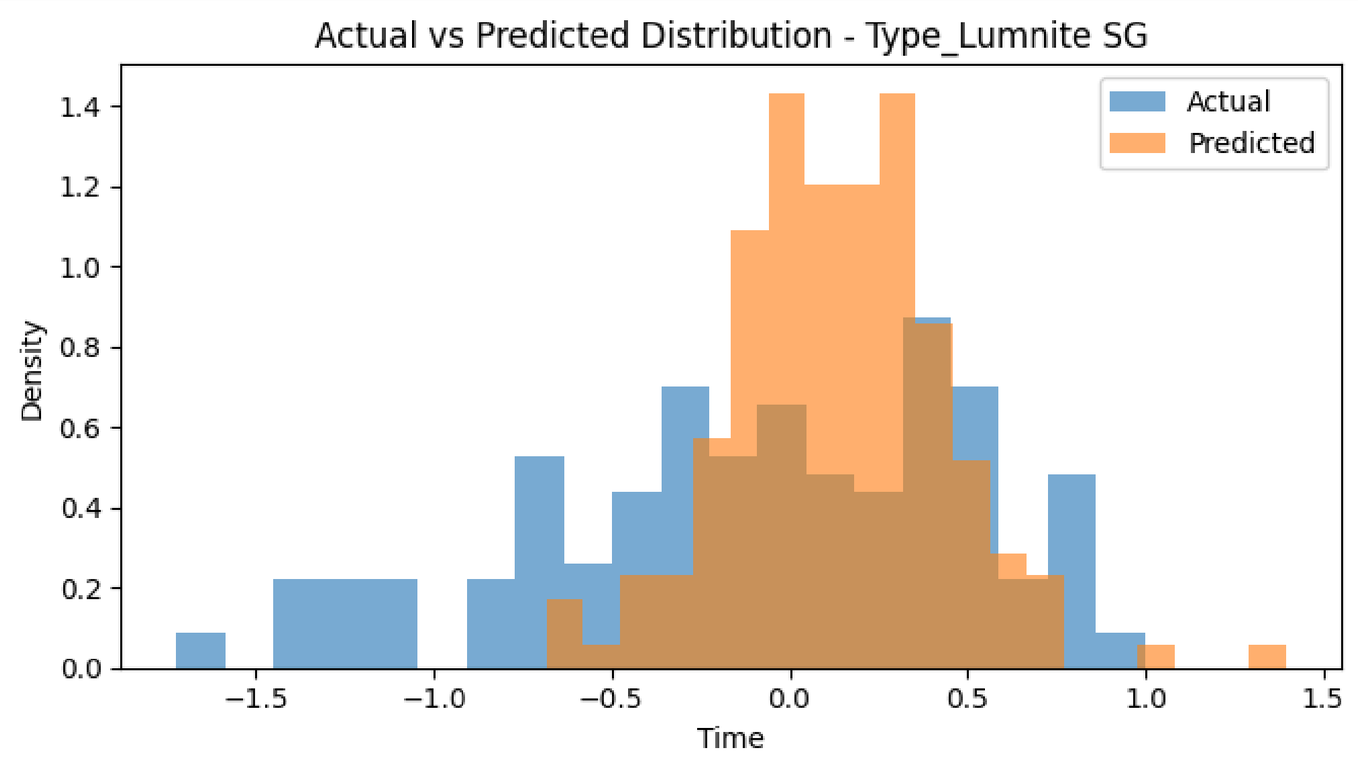
Results Experiment 2



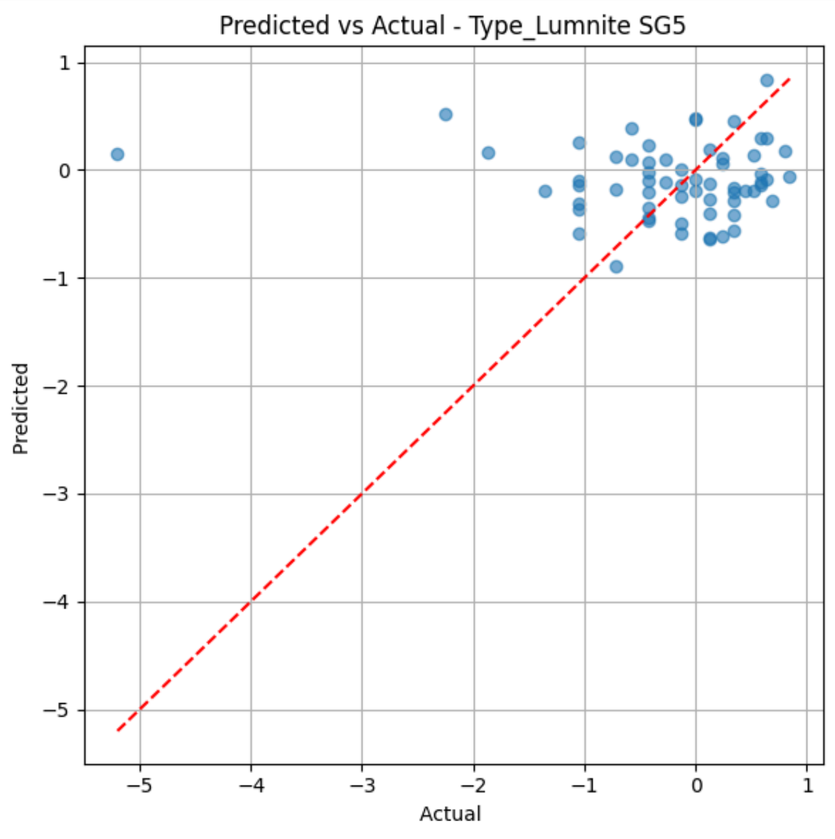


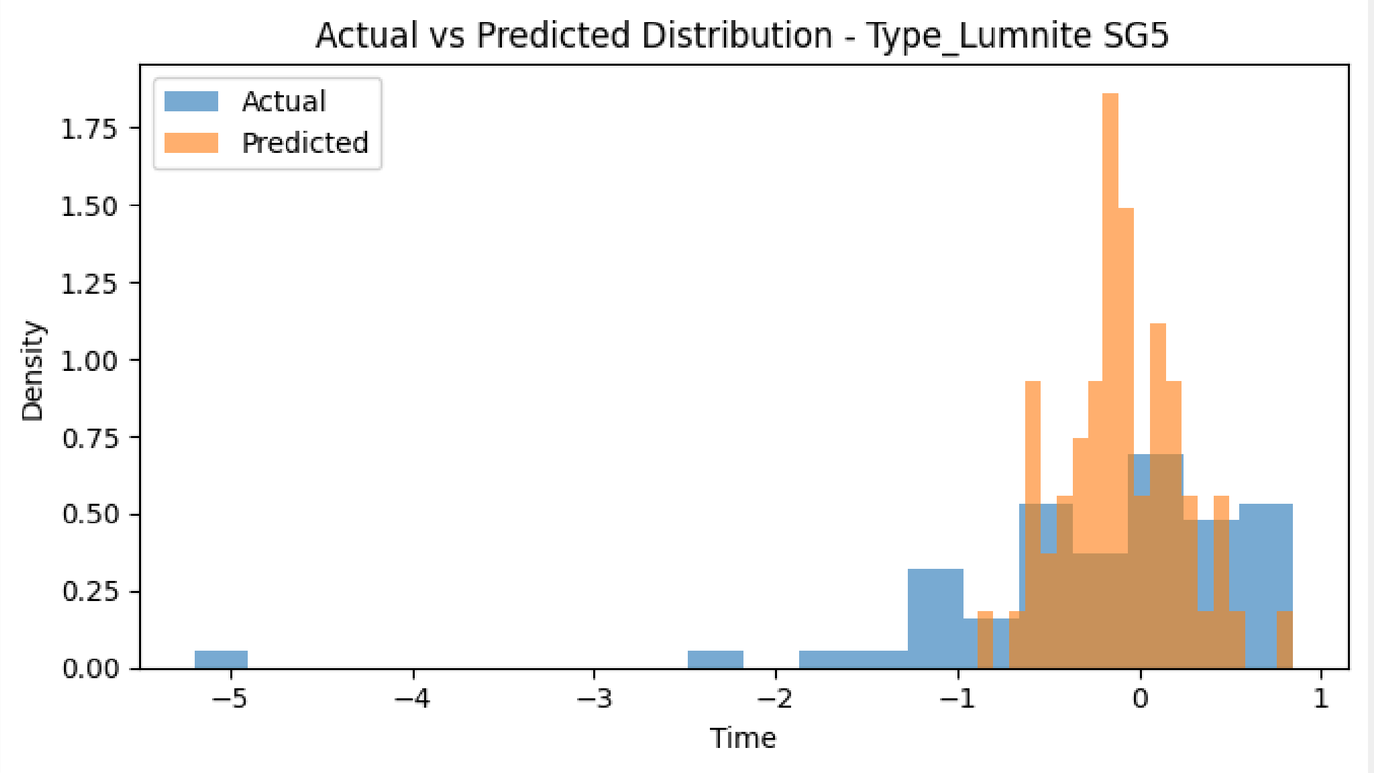
Results Experiment 2





Results Experiment 2





Results from the Autoencoder

Raw Dataset:
- Chem and Phase blocks were skipped — high RRMSE (train > 0.3, test > 0.3)

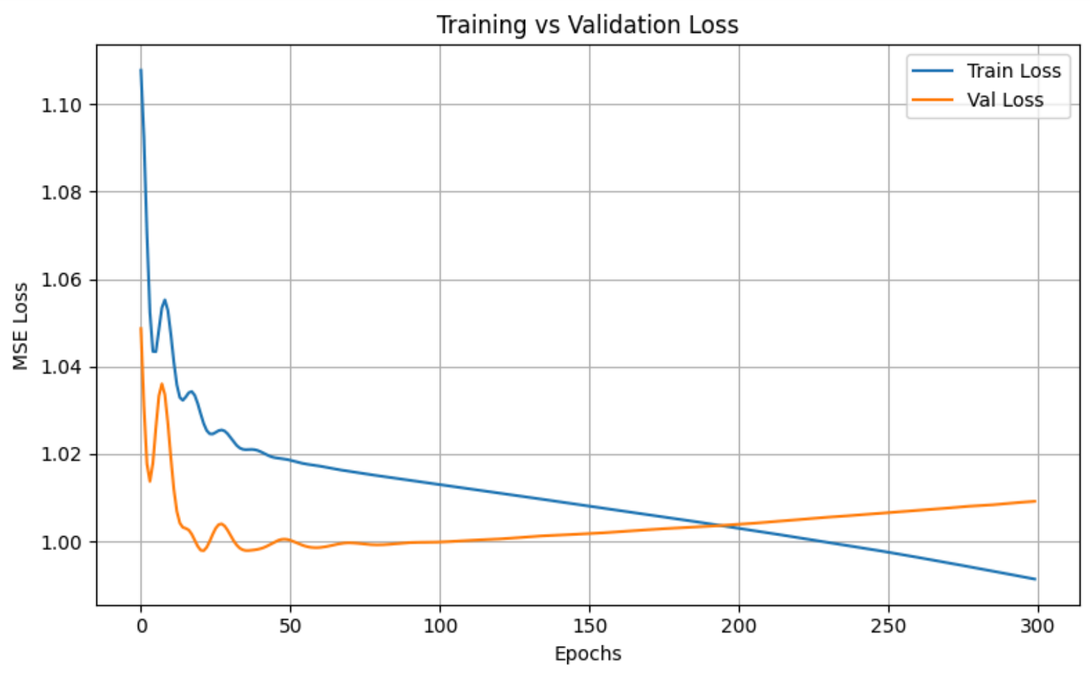
Raw + PCA:
- All blocks were skipped — poor reconstruction quality remained (RRMSE > 0.3)

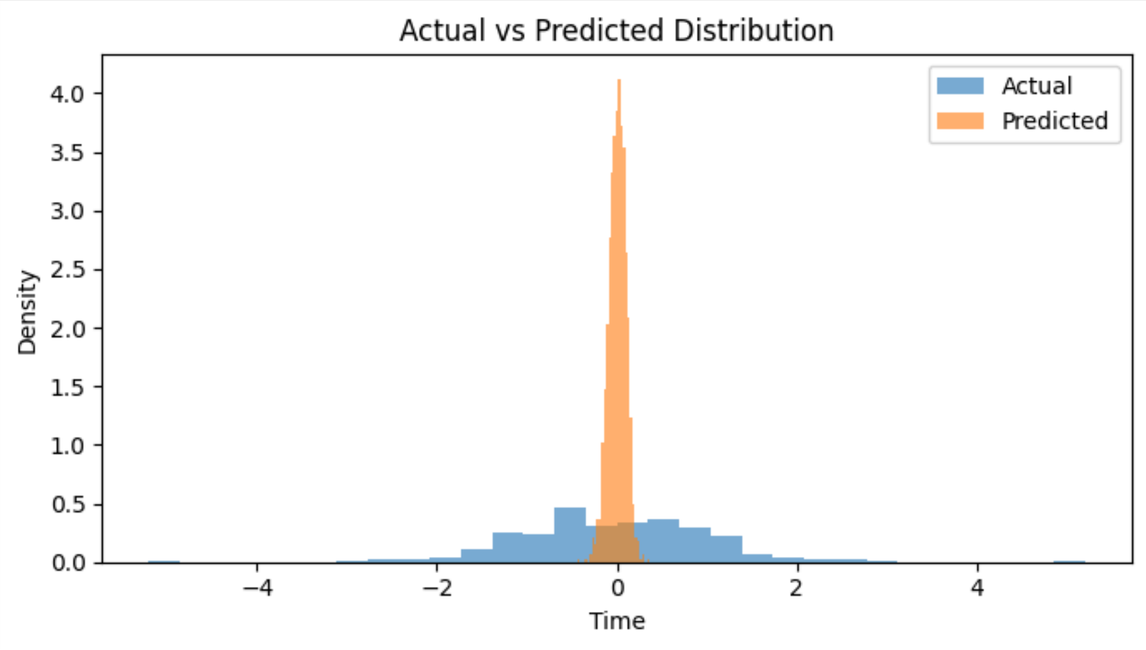
Enhanced Dataset:
- Chem_ilr and Phase_ilr blocks failed to compress well (RRMSE > 0.3)

Enhanced + PCA:
- No blocks available for training (PCA removed too many relevant features)

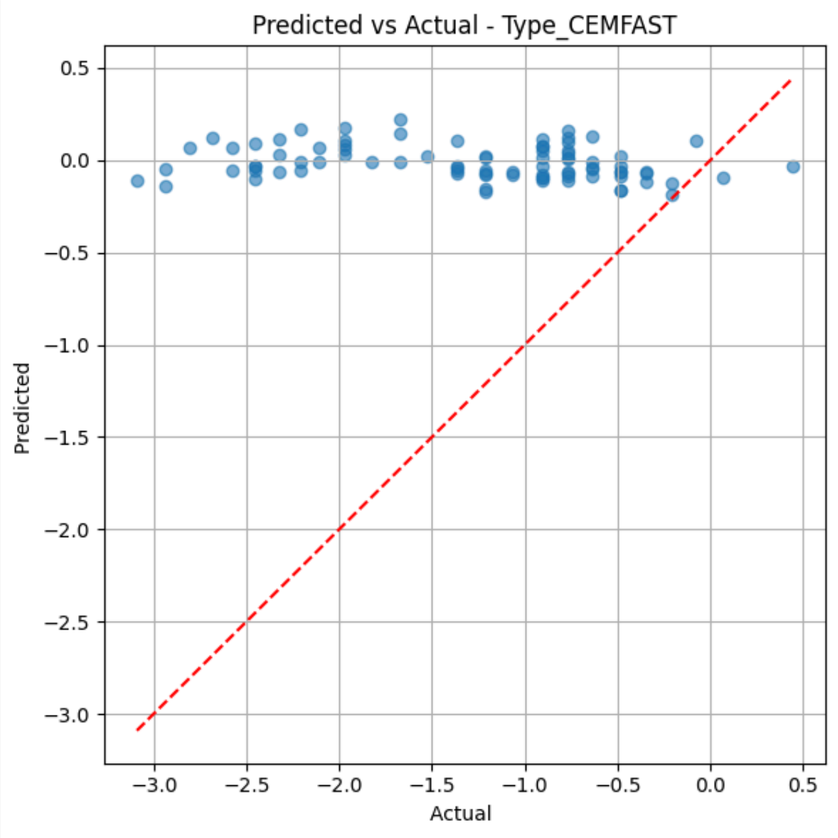
Full Enhanced block (all non-OHE features together):
- RRMSE was still above threshold
- Suggests autoencoders struggle with large, diverse input feature spaces

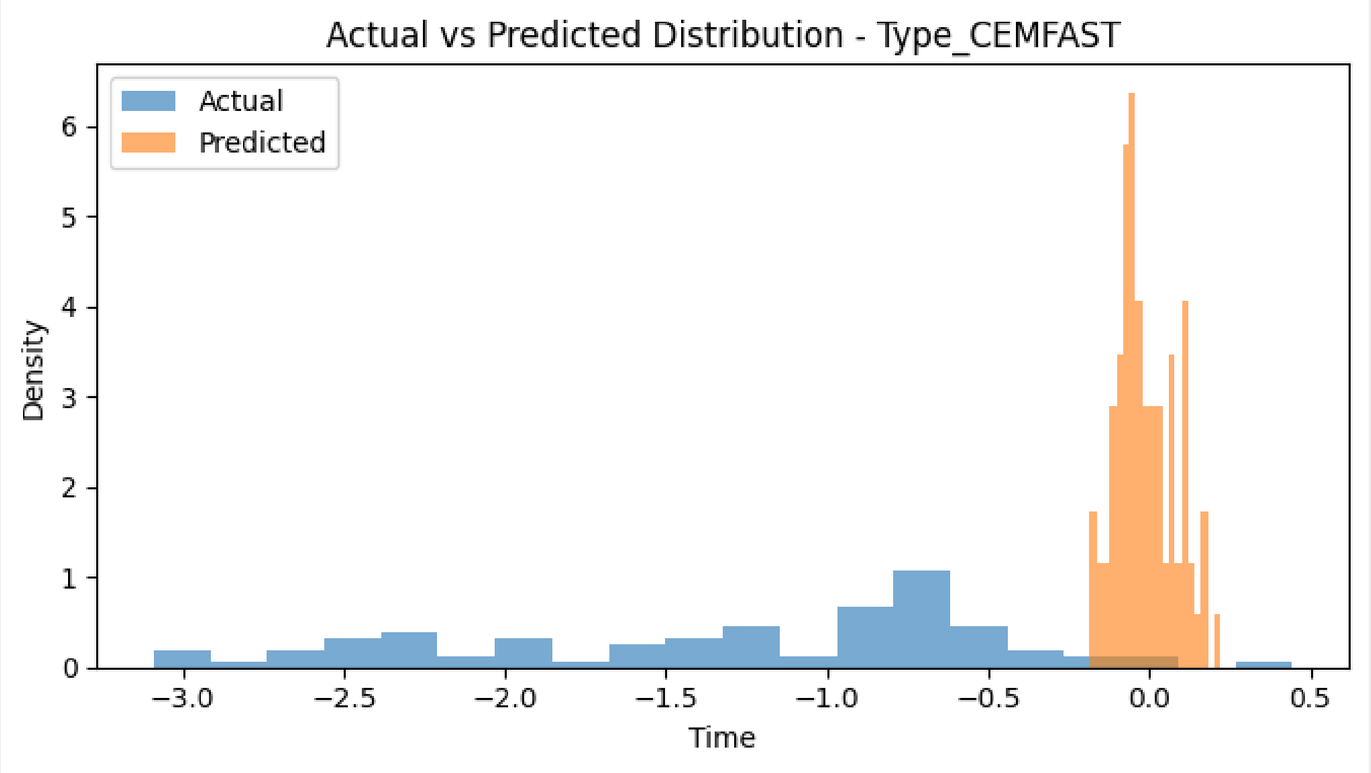
Results Experiment 3



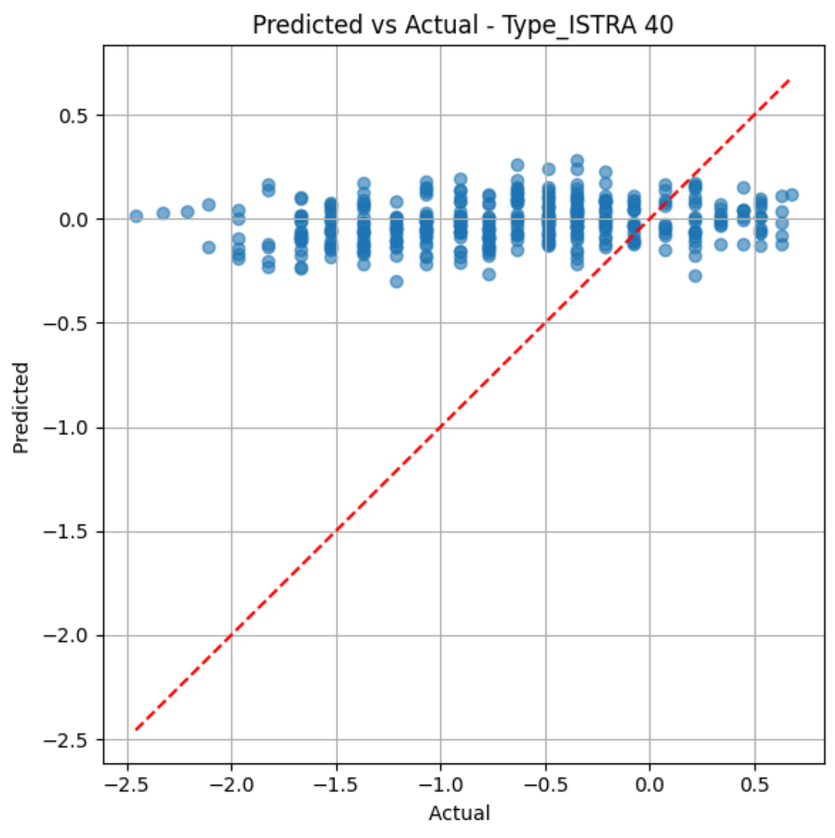


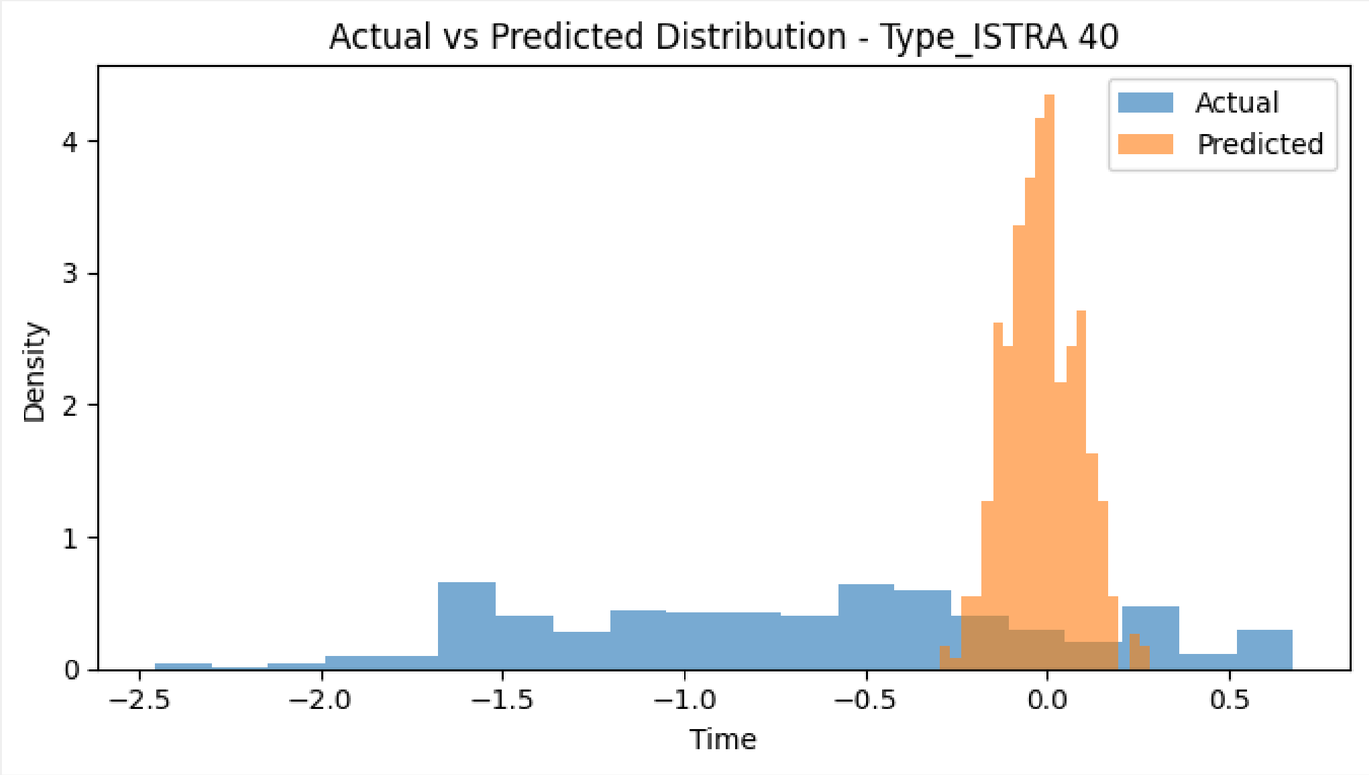
Results Experiment 3



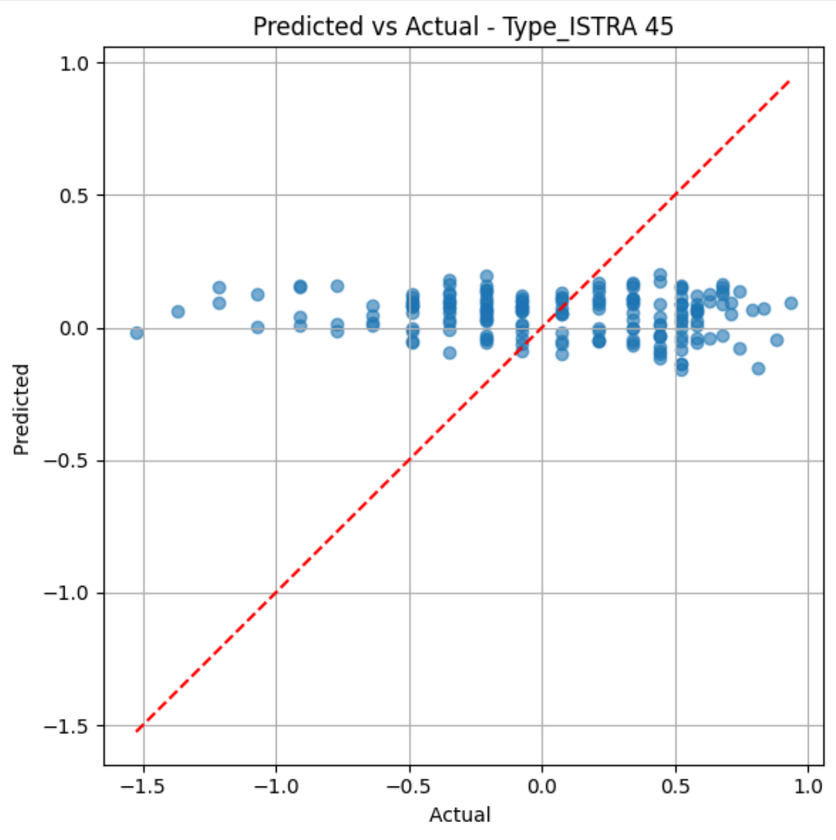


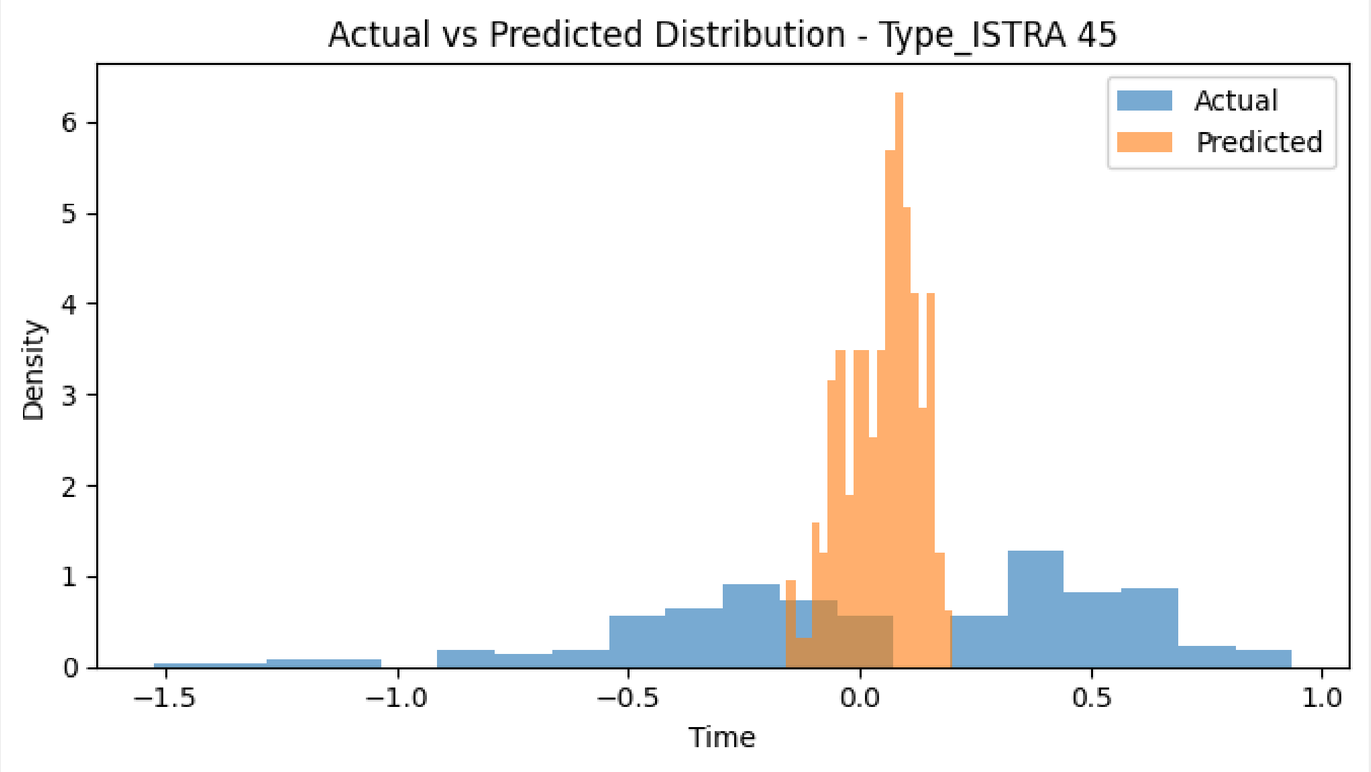
Results Experiment 3



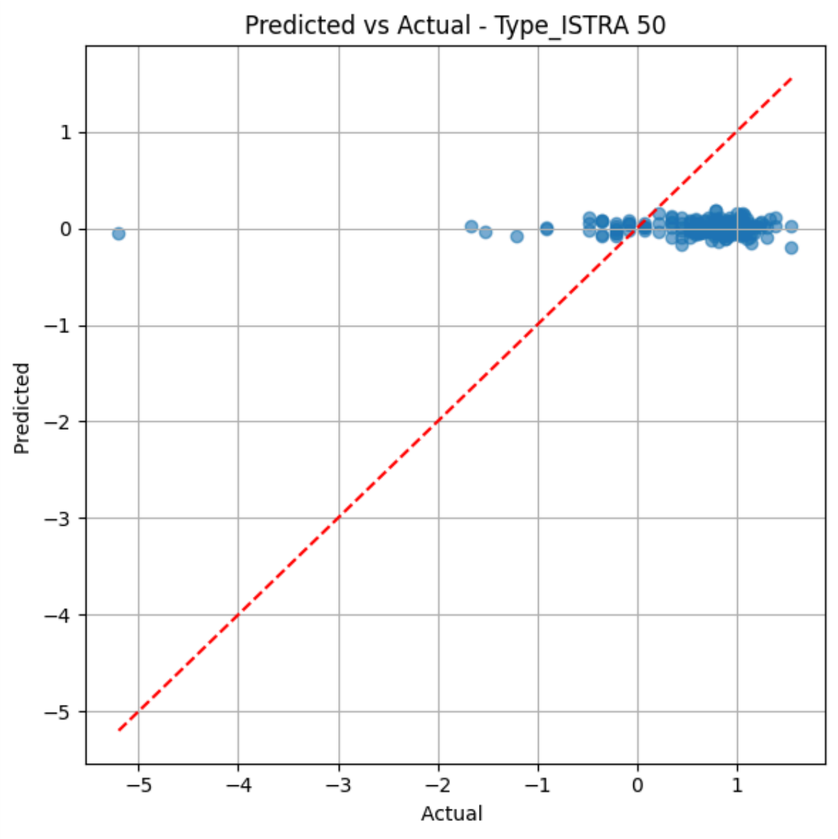


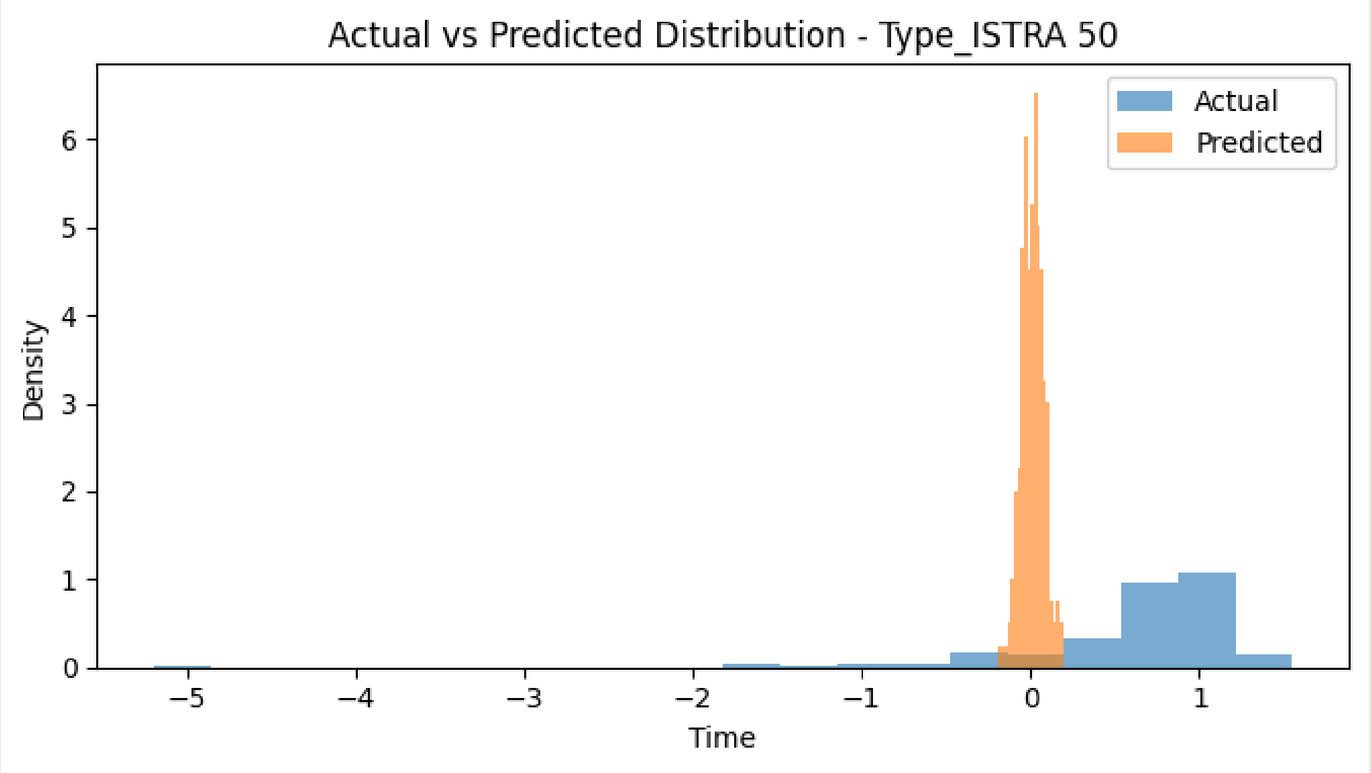
Results Experiment 3



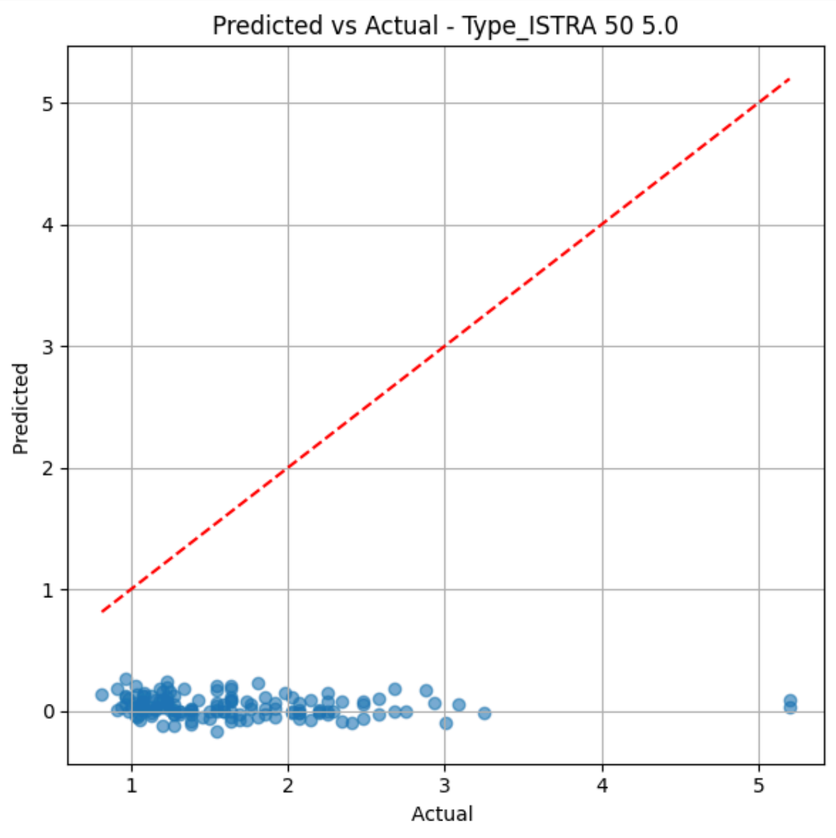


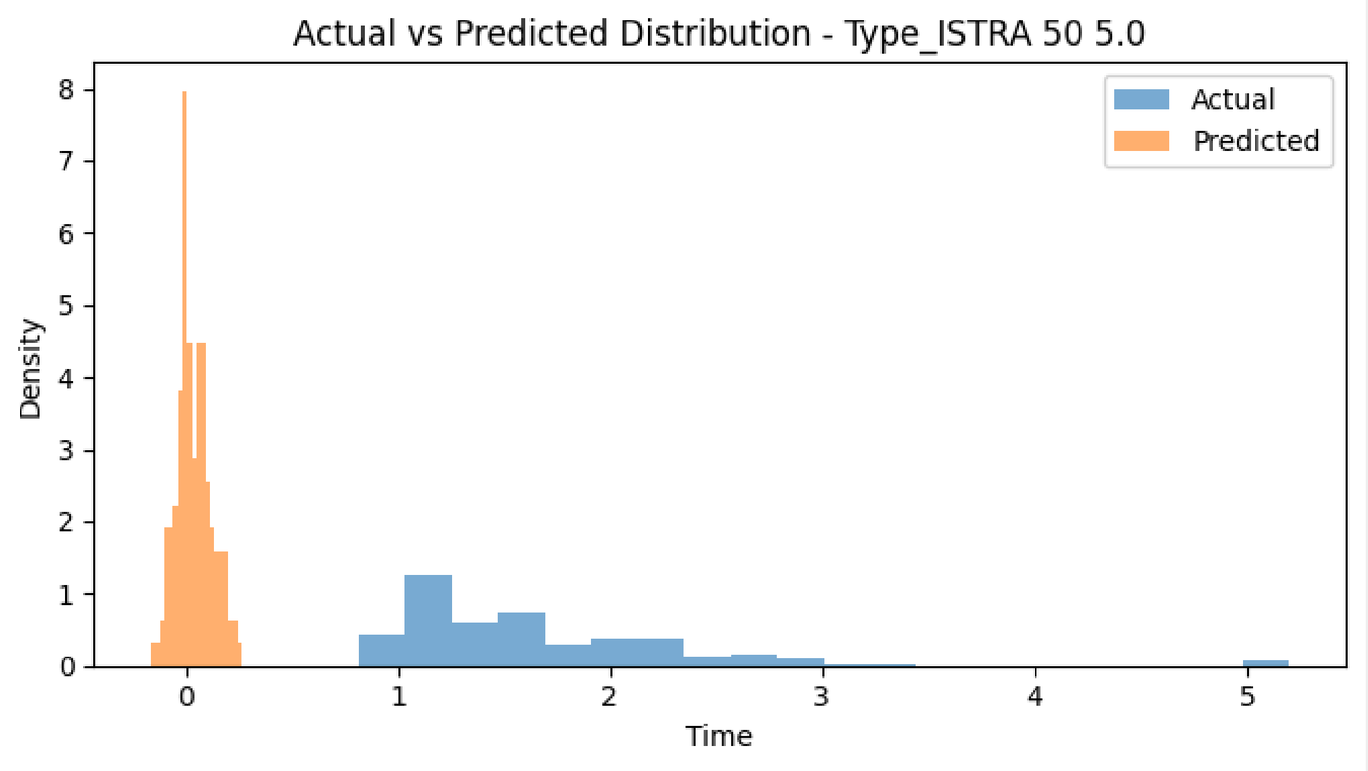
Results Experiment 3



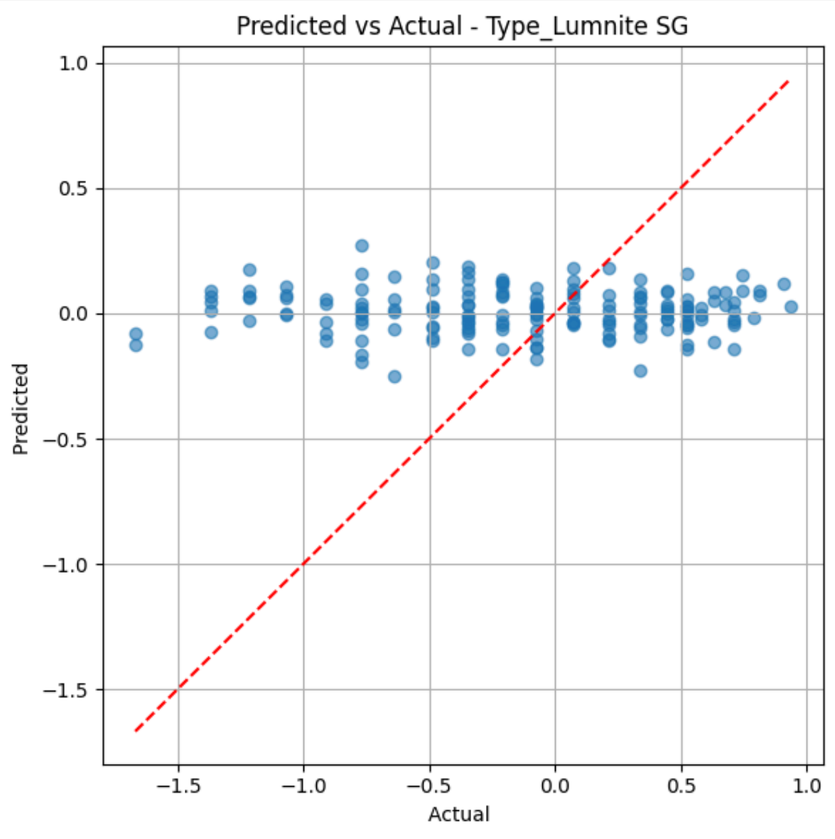


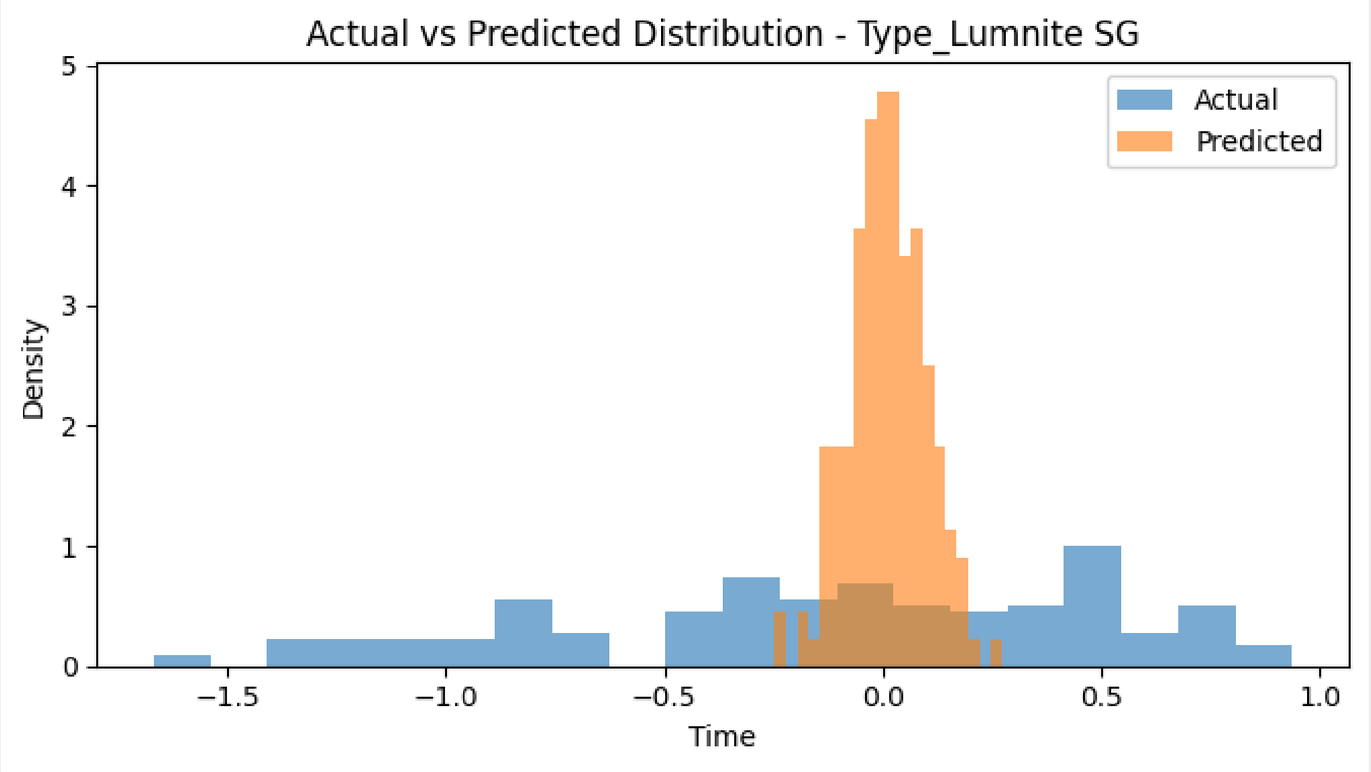
Results Experiment 3



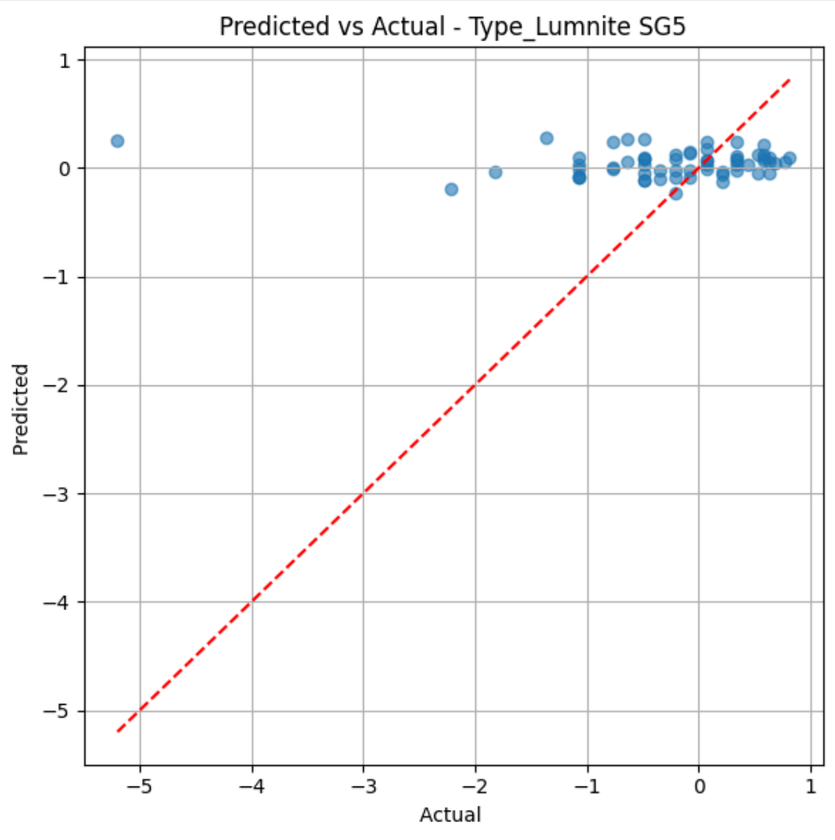


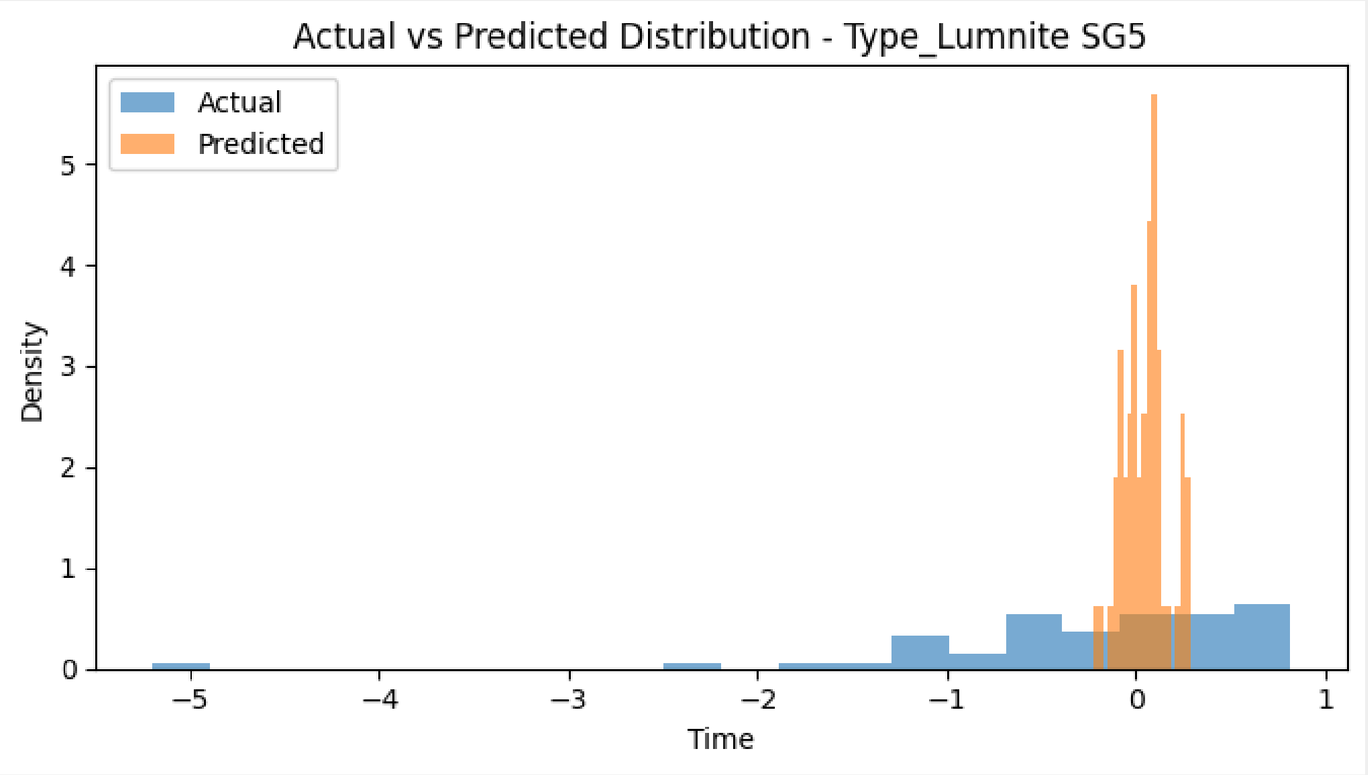
Results Experiment 3



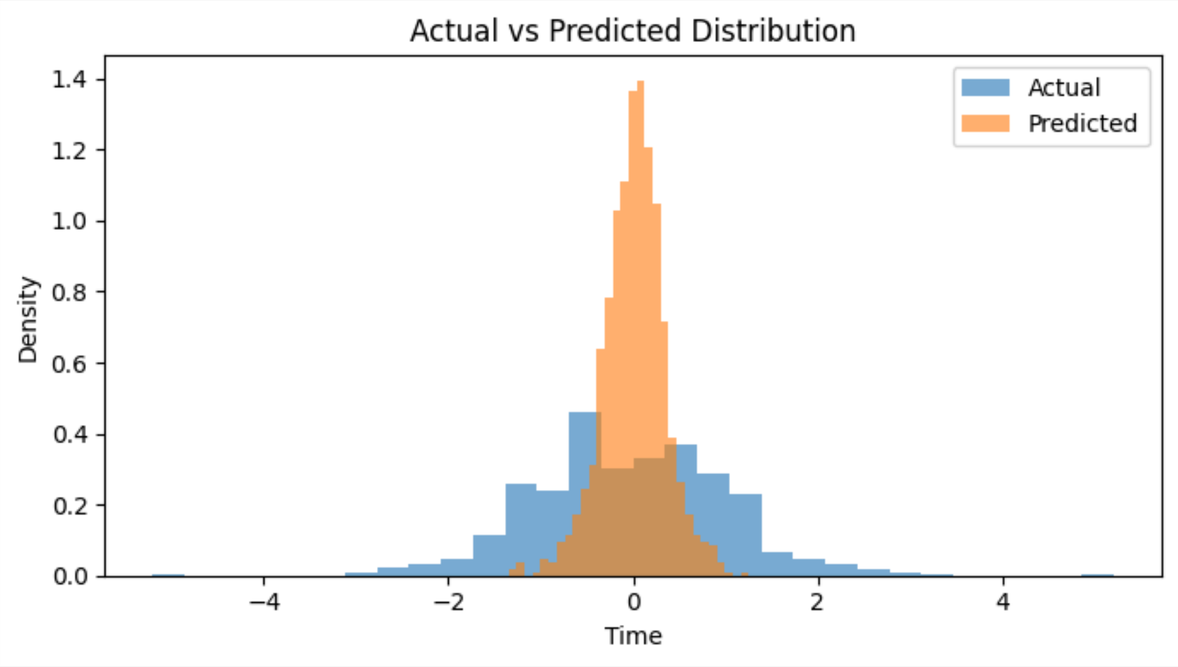


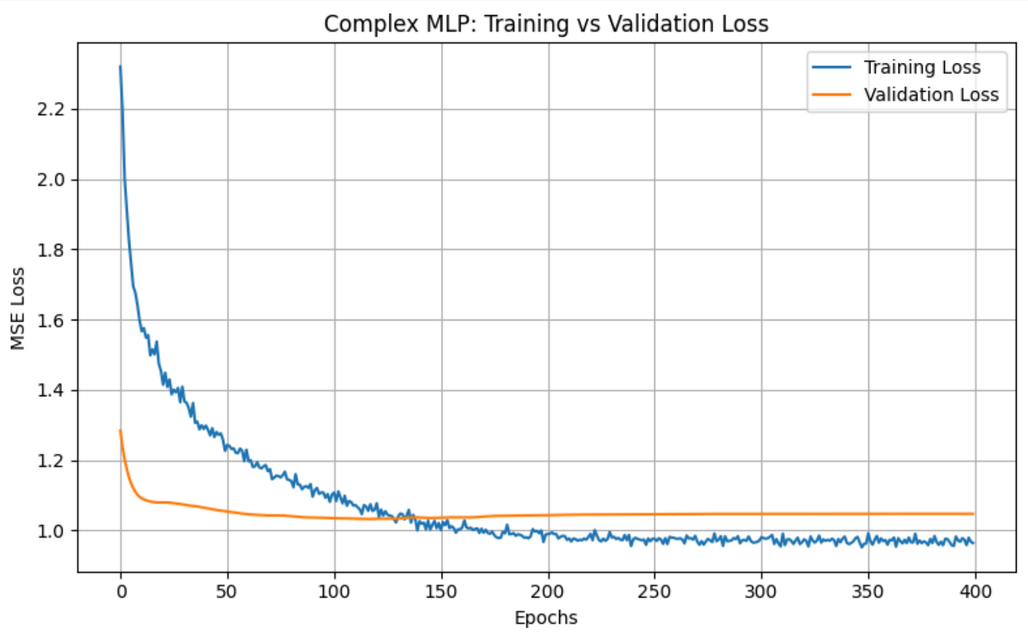
Results Experiment 3



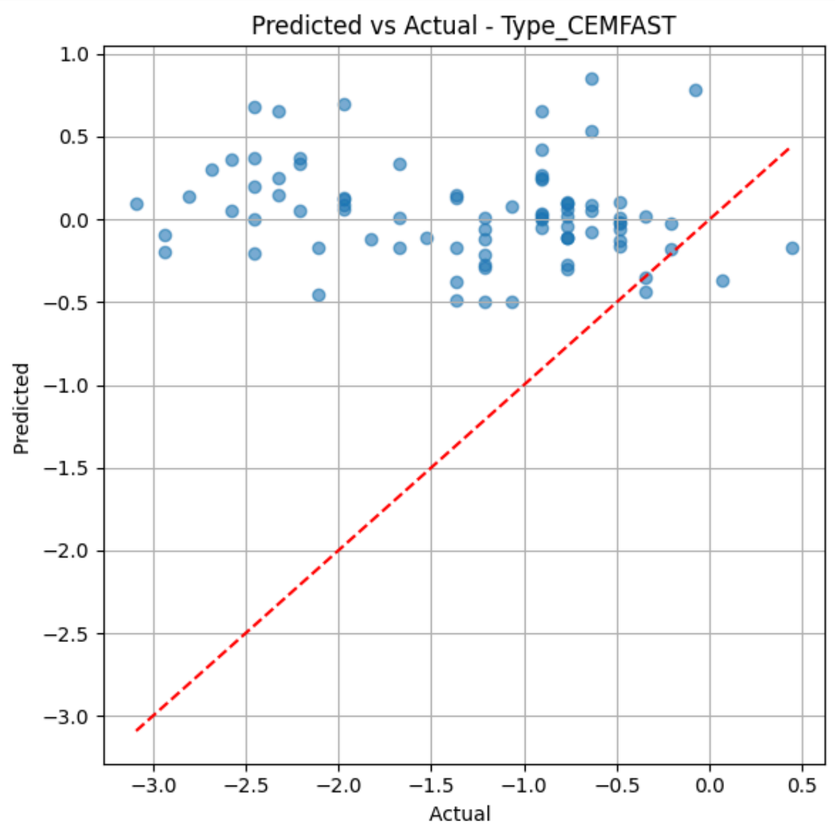


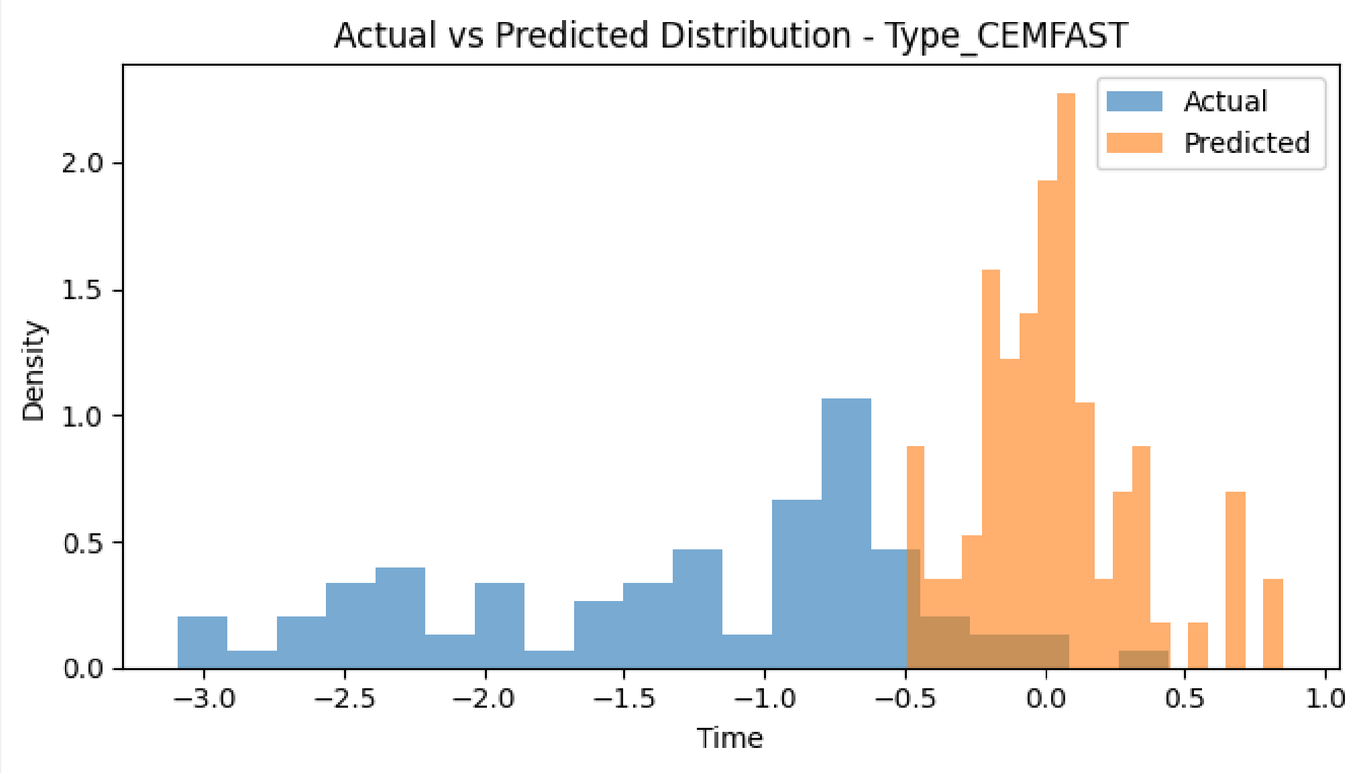
Results Experiment 4



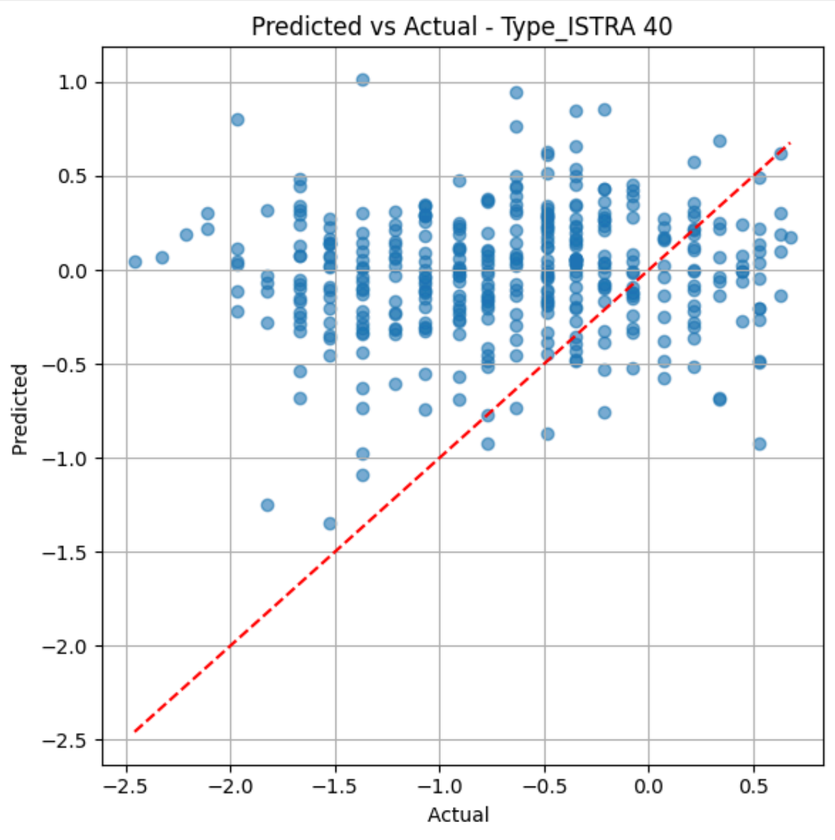


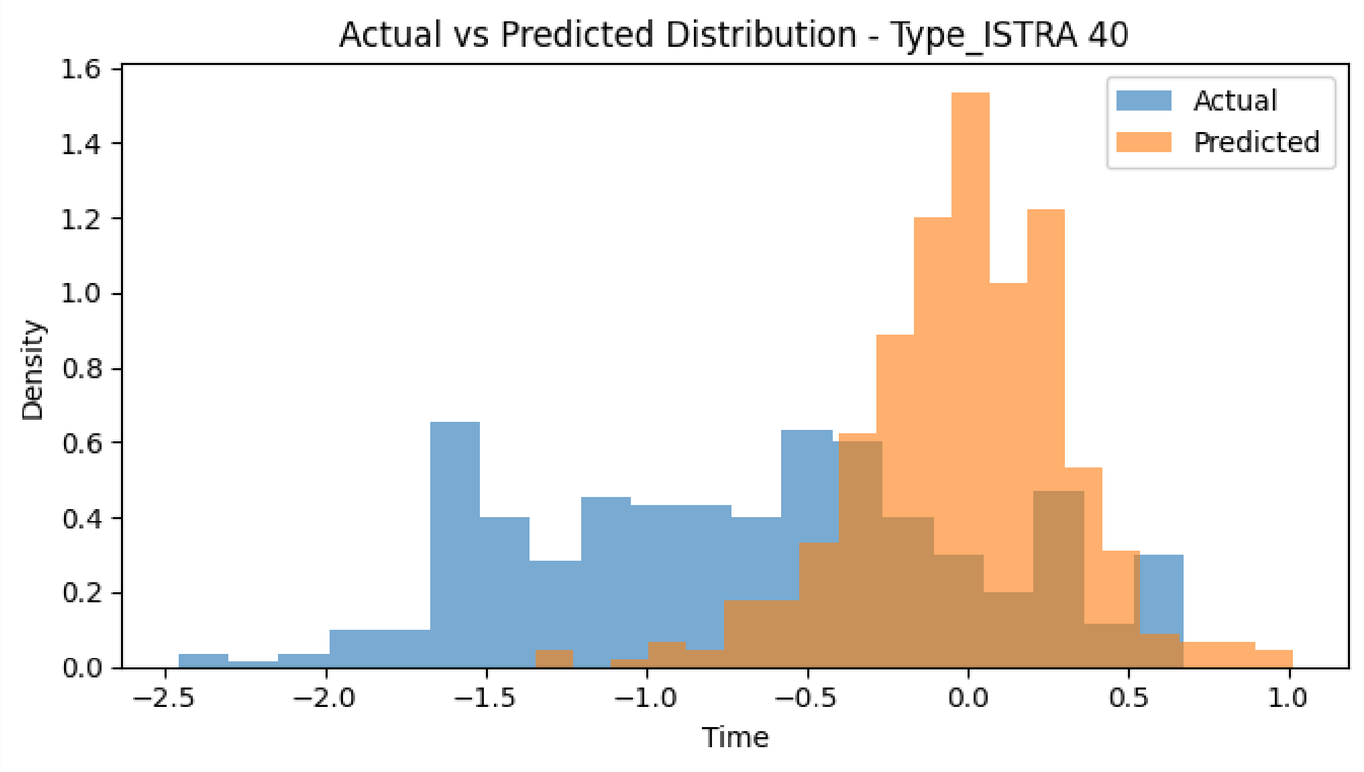
Results Experiment 4



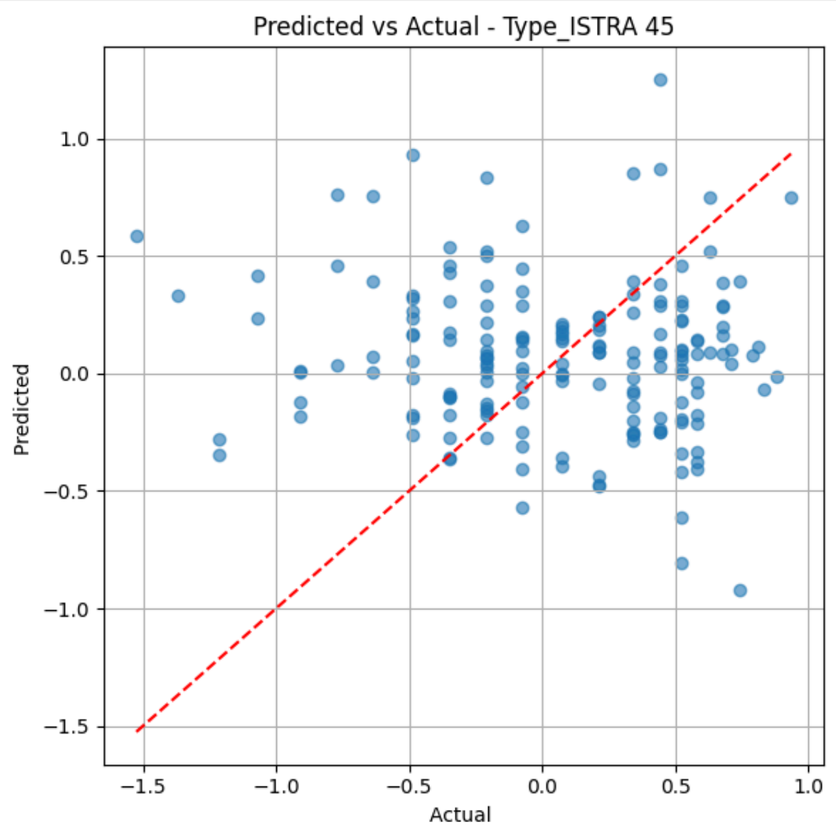


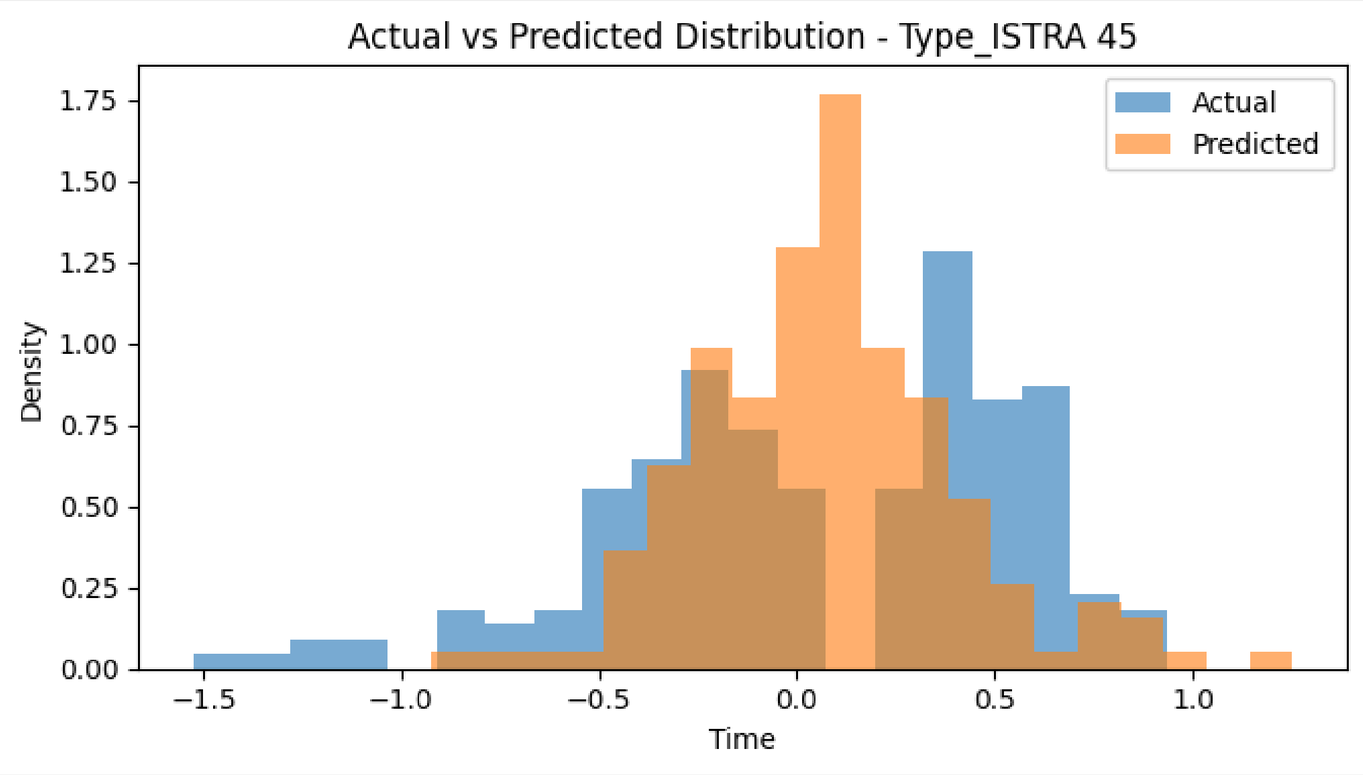
Results Experiment 4



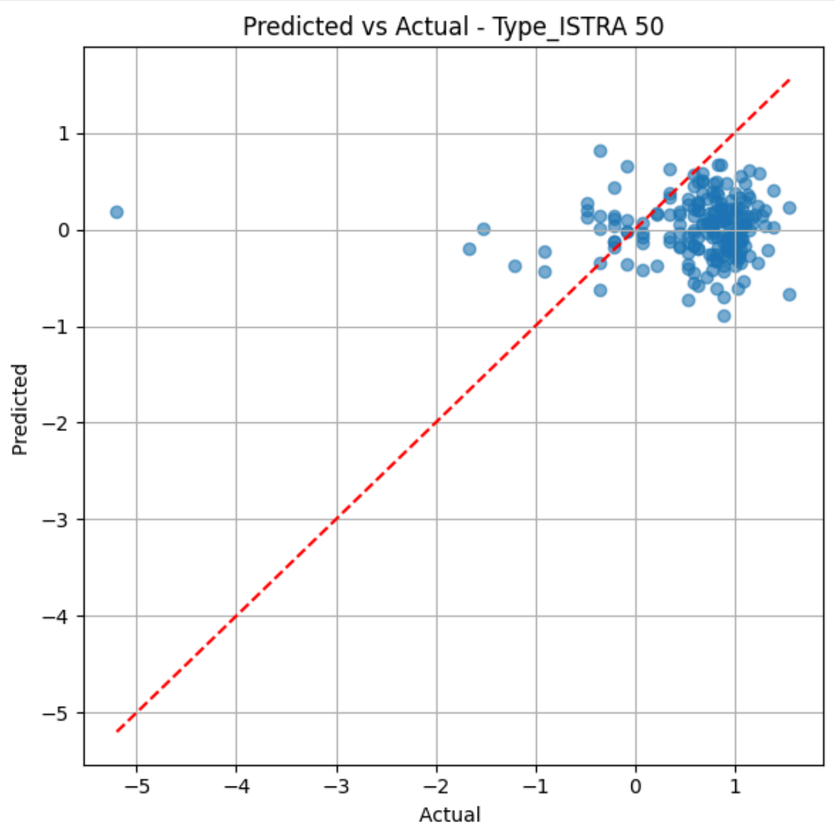


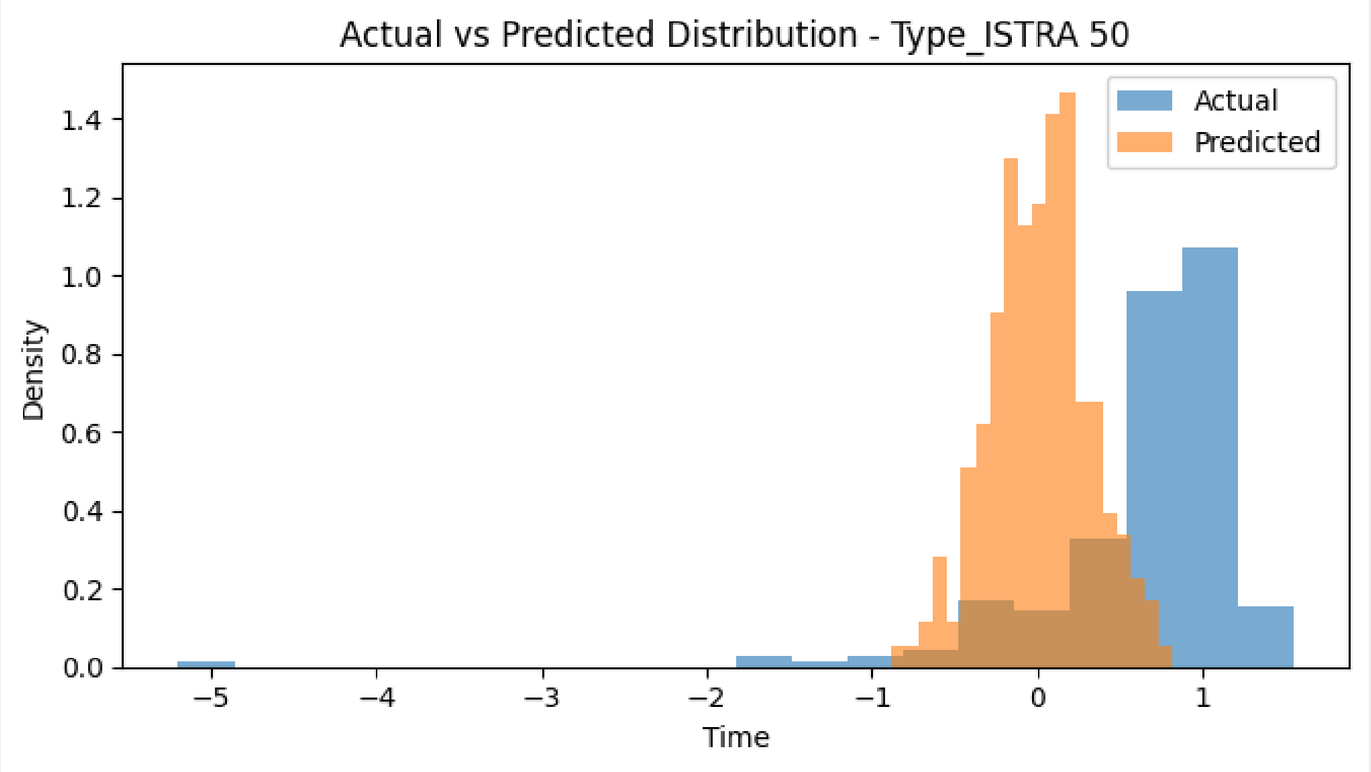
Results Experiment 4



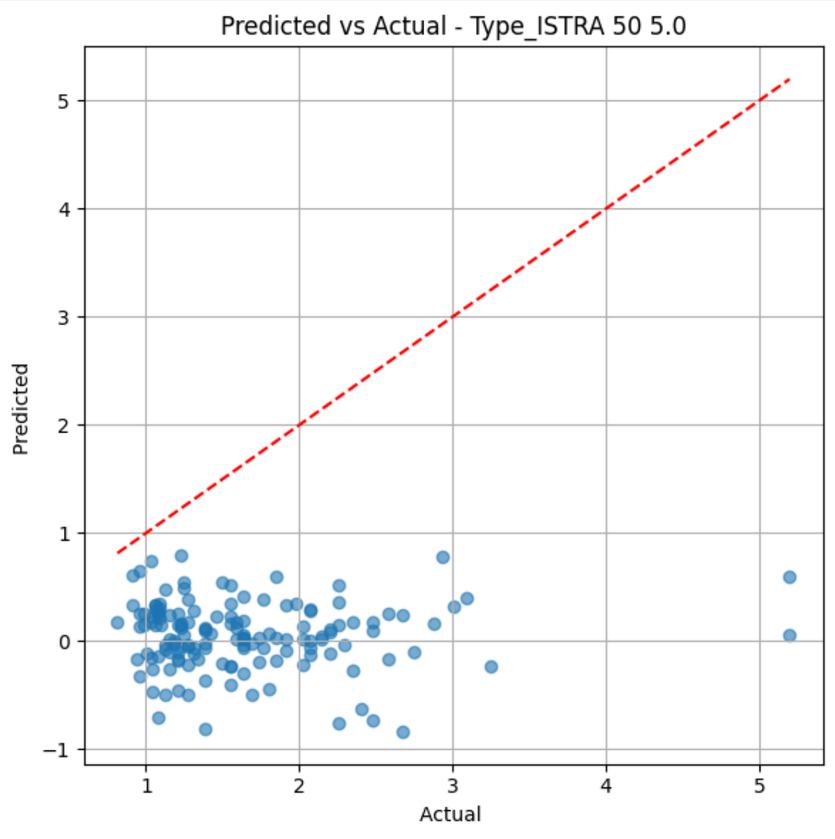


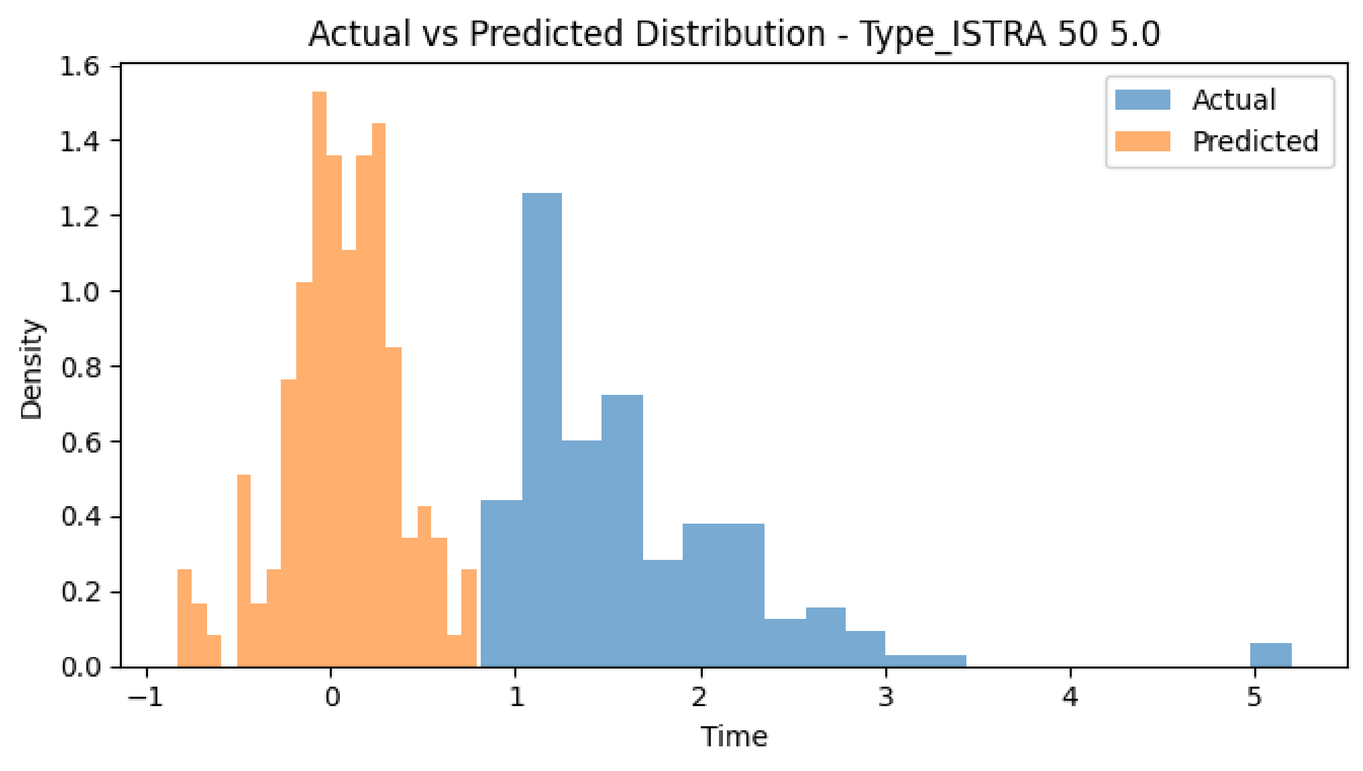
Results Experiment 4



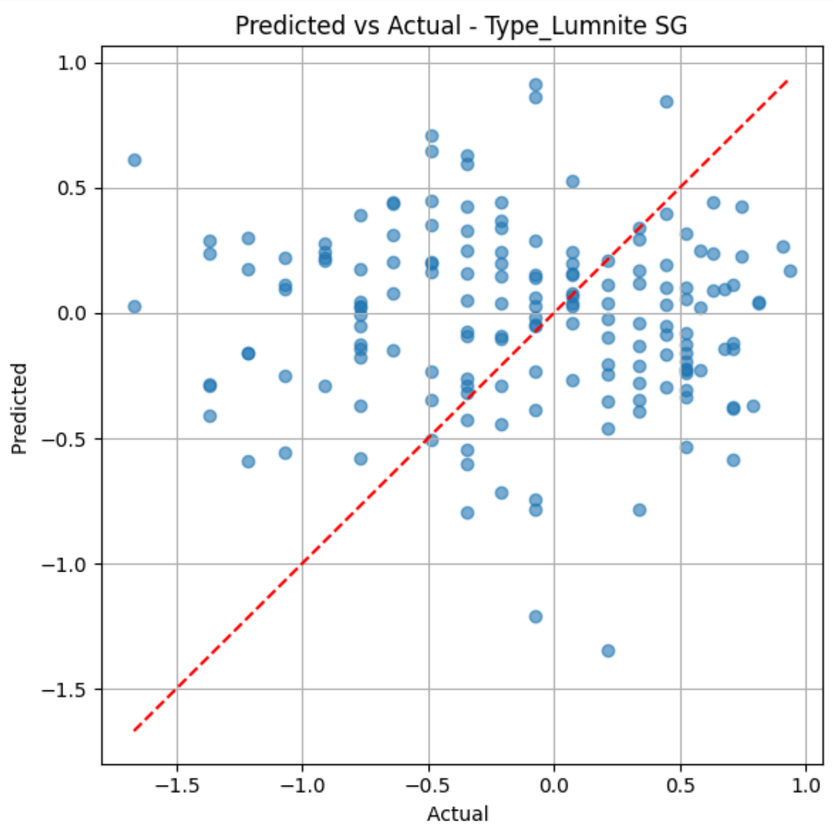


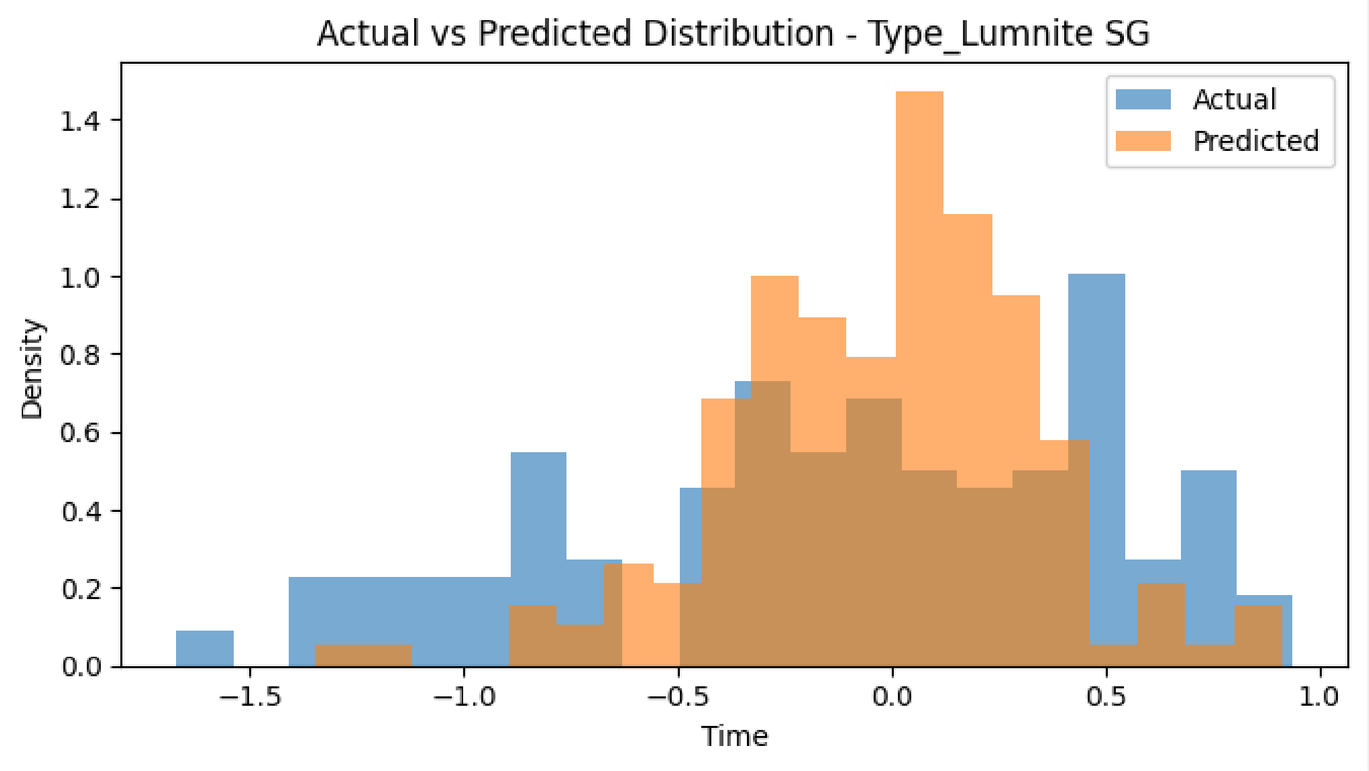
Results Experiment 4



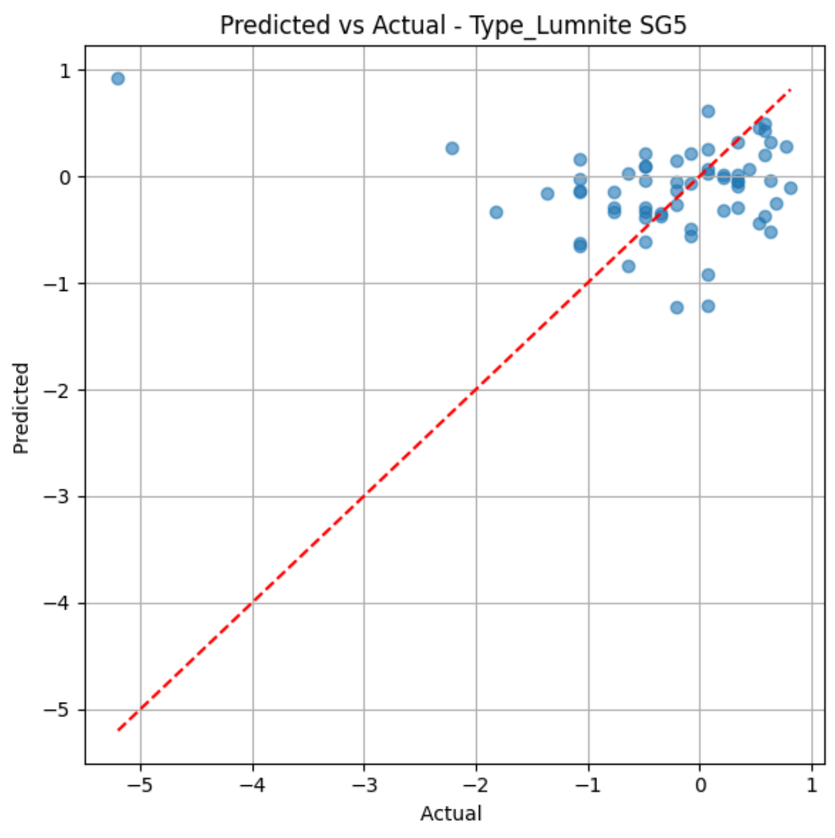


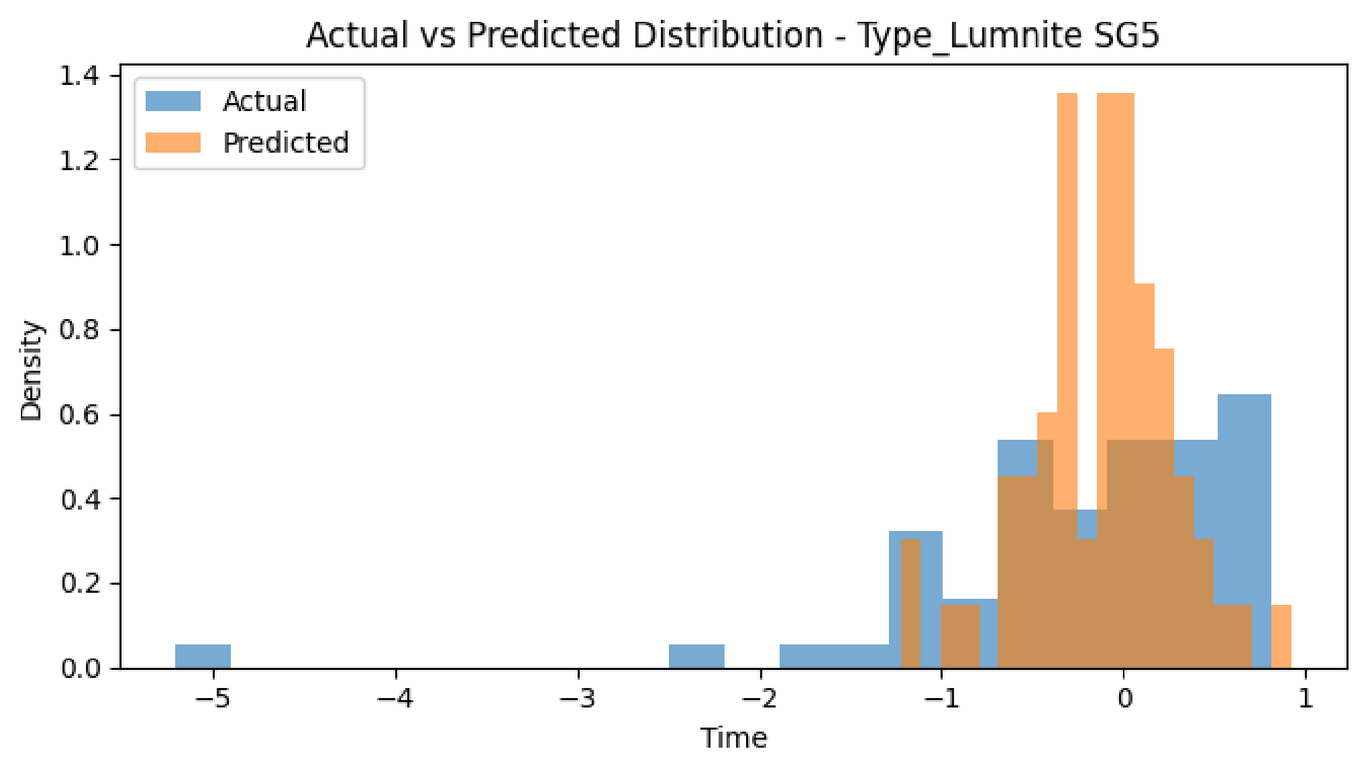
Results Experiment 4



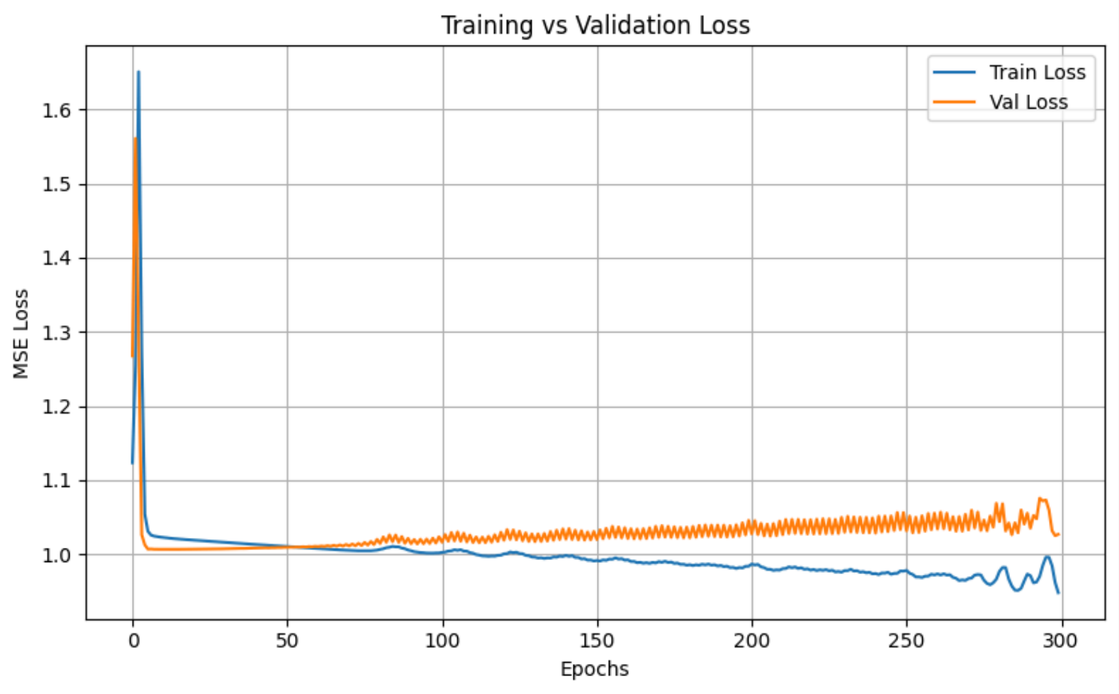


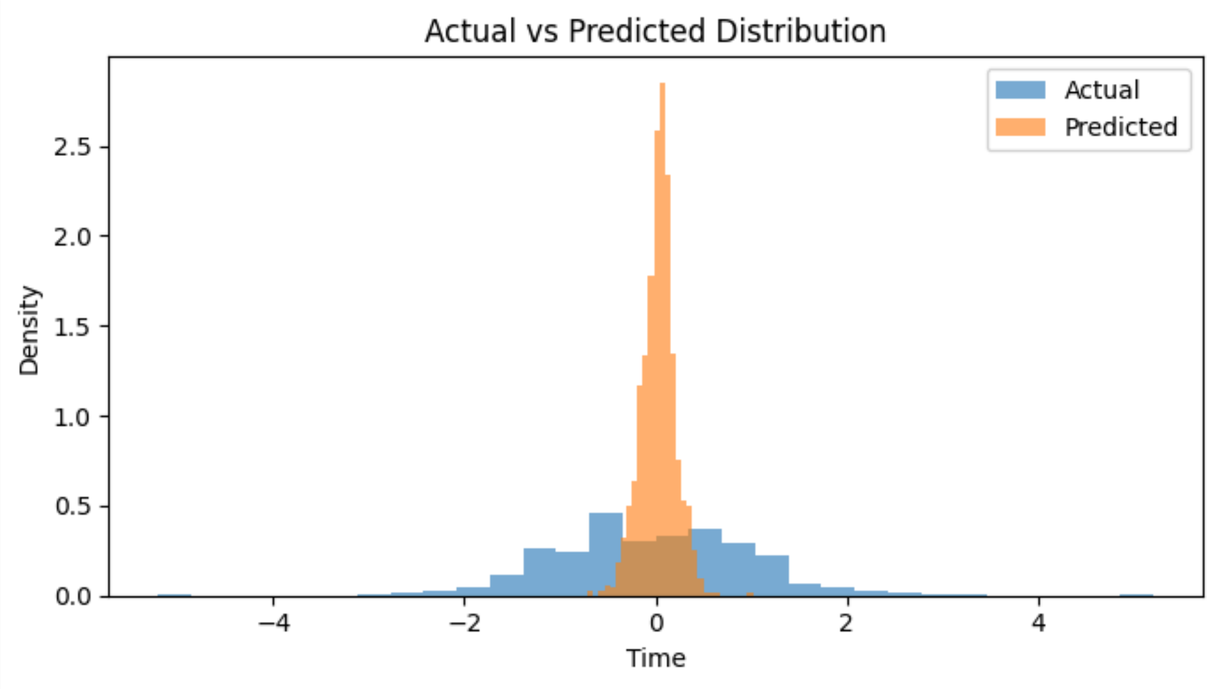
Results Experiment 4



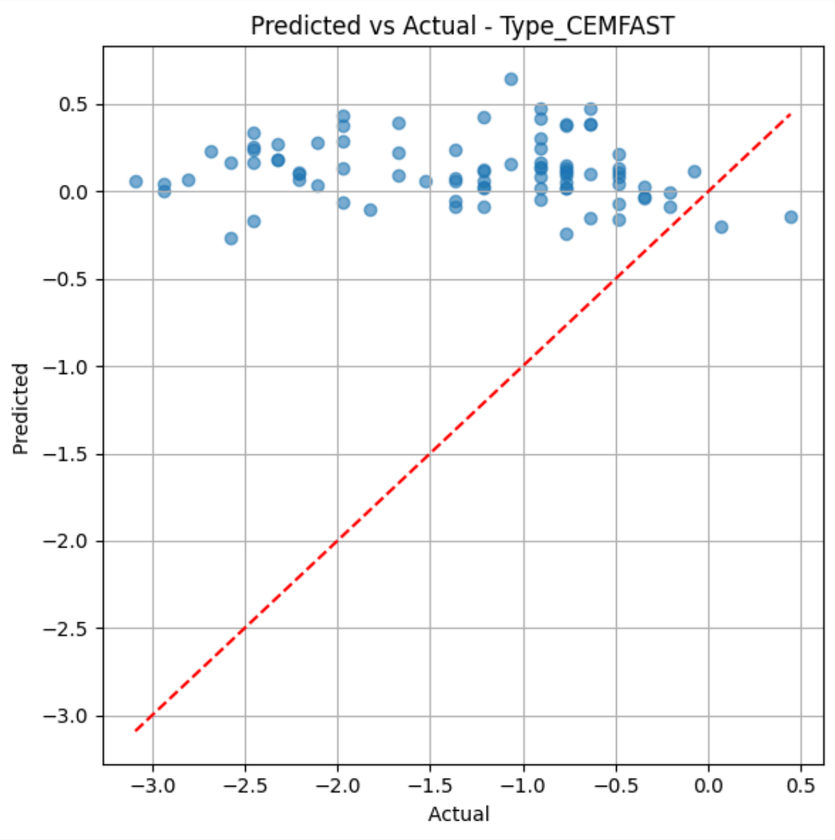


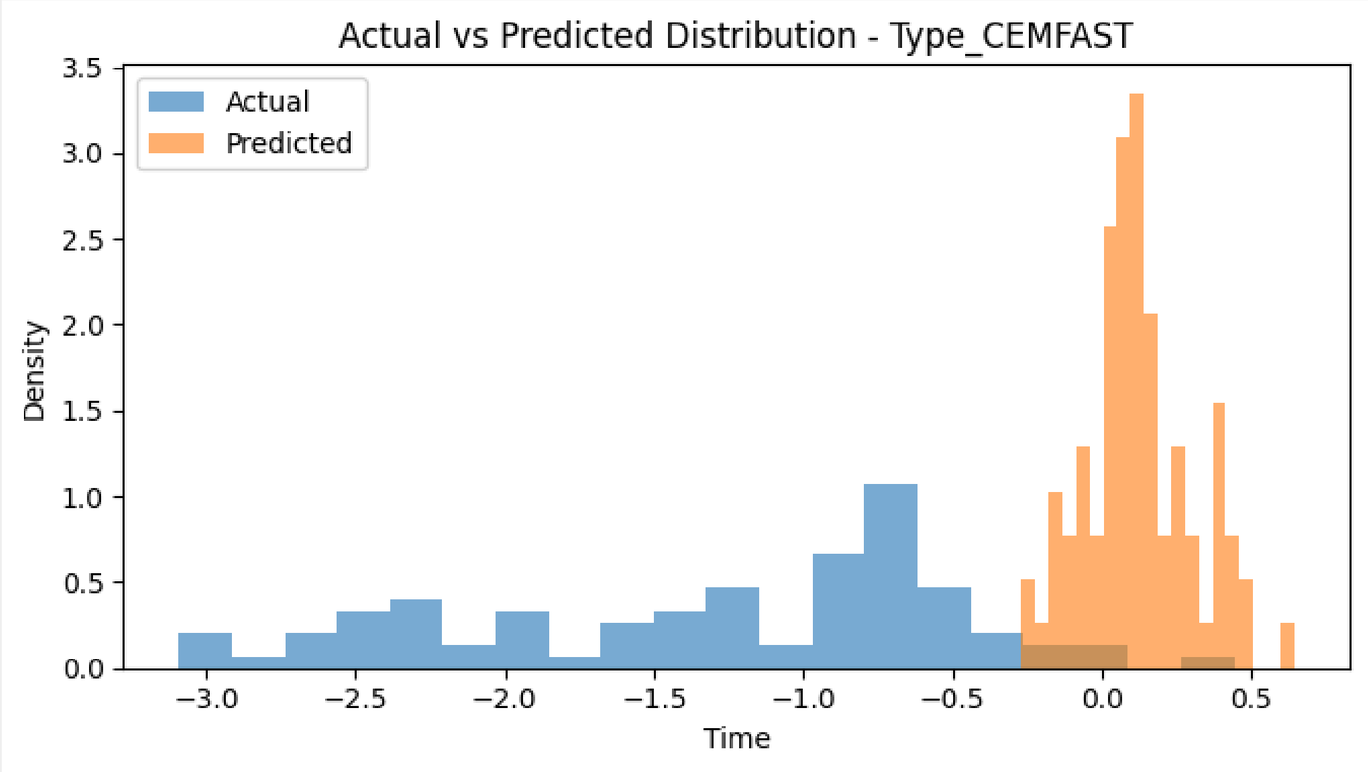
Results Experiment 5



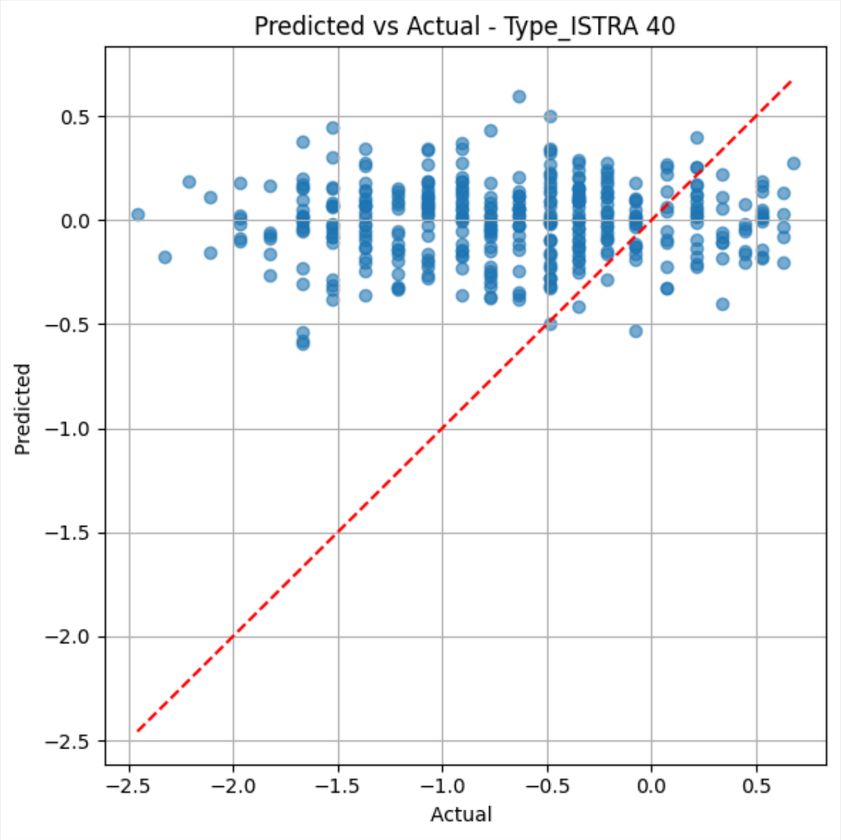


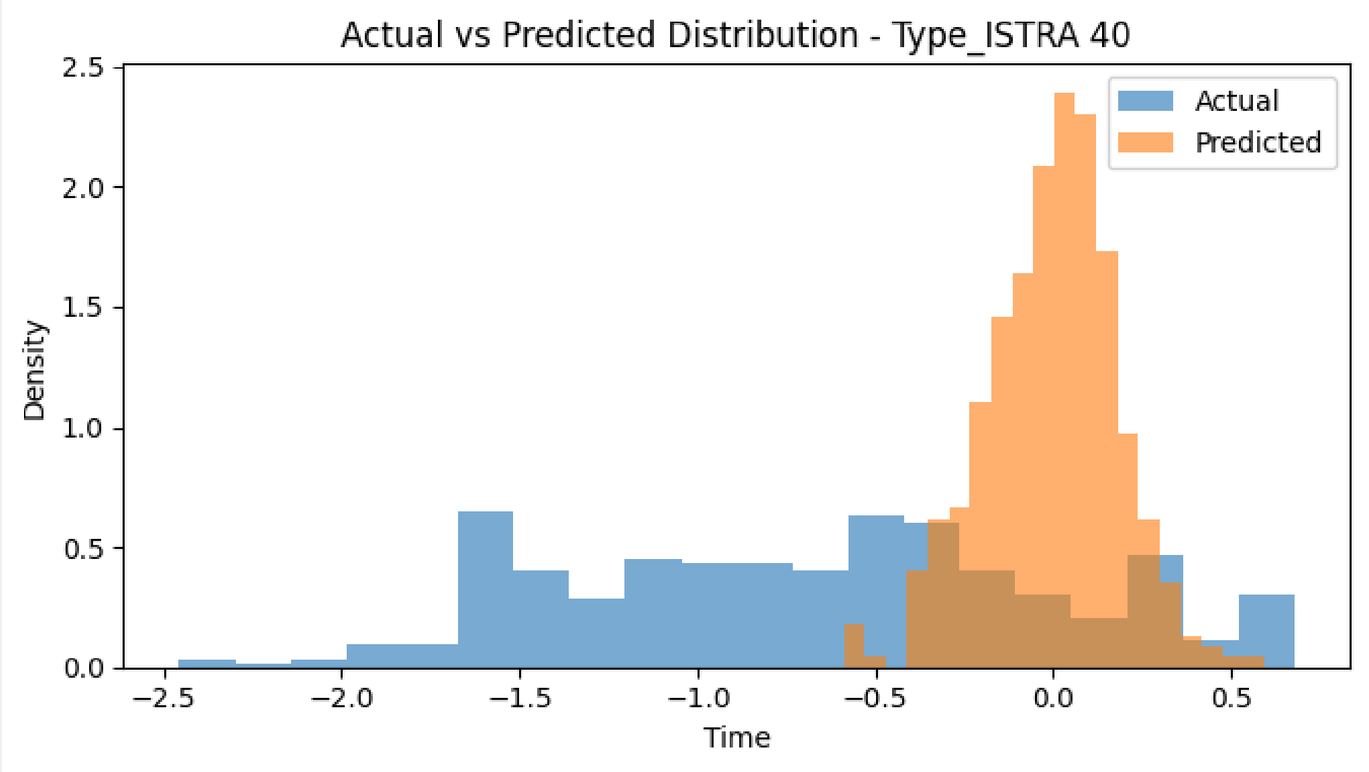
Results Experiment 5



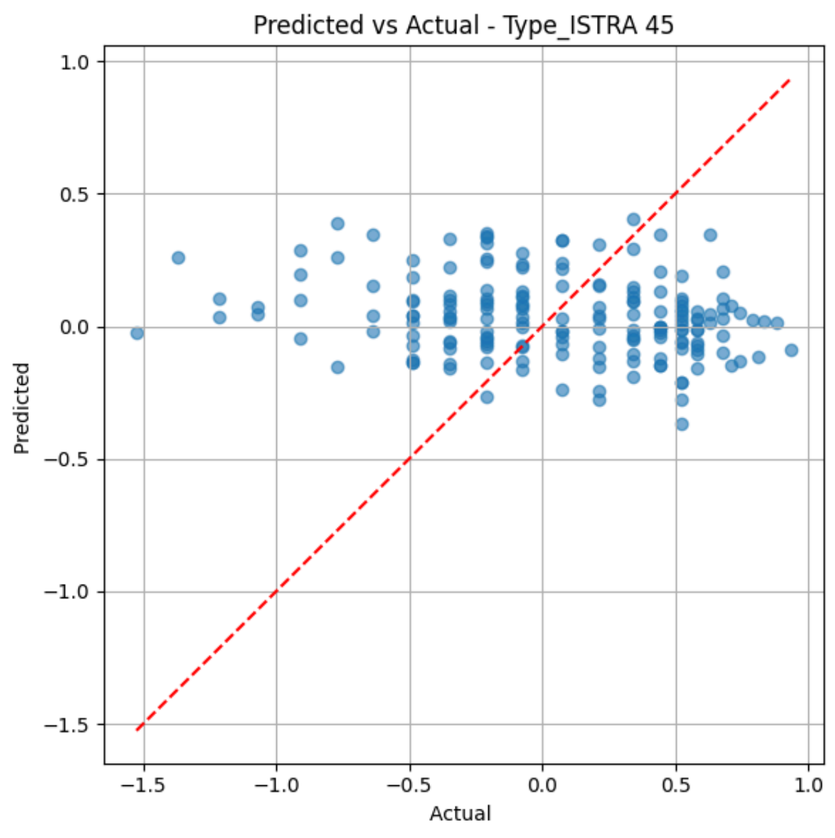


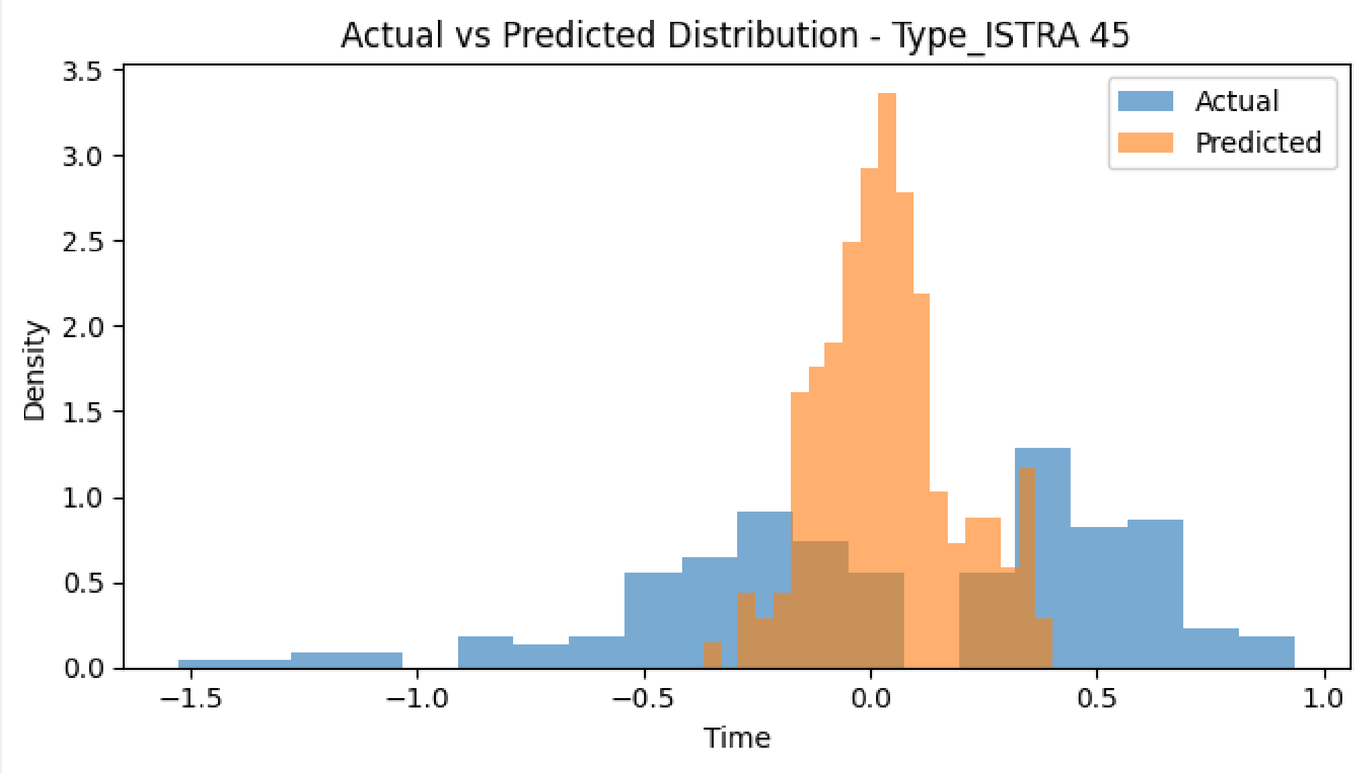
Results Experiment 5



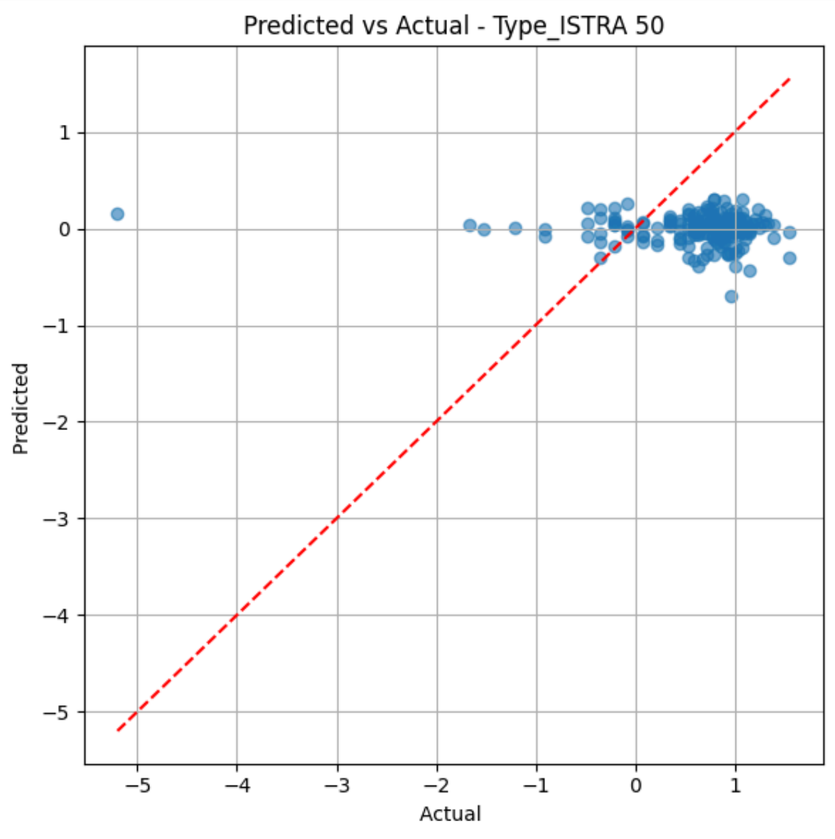


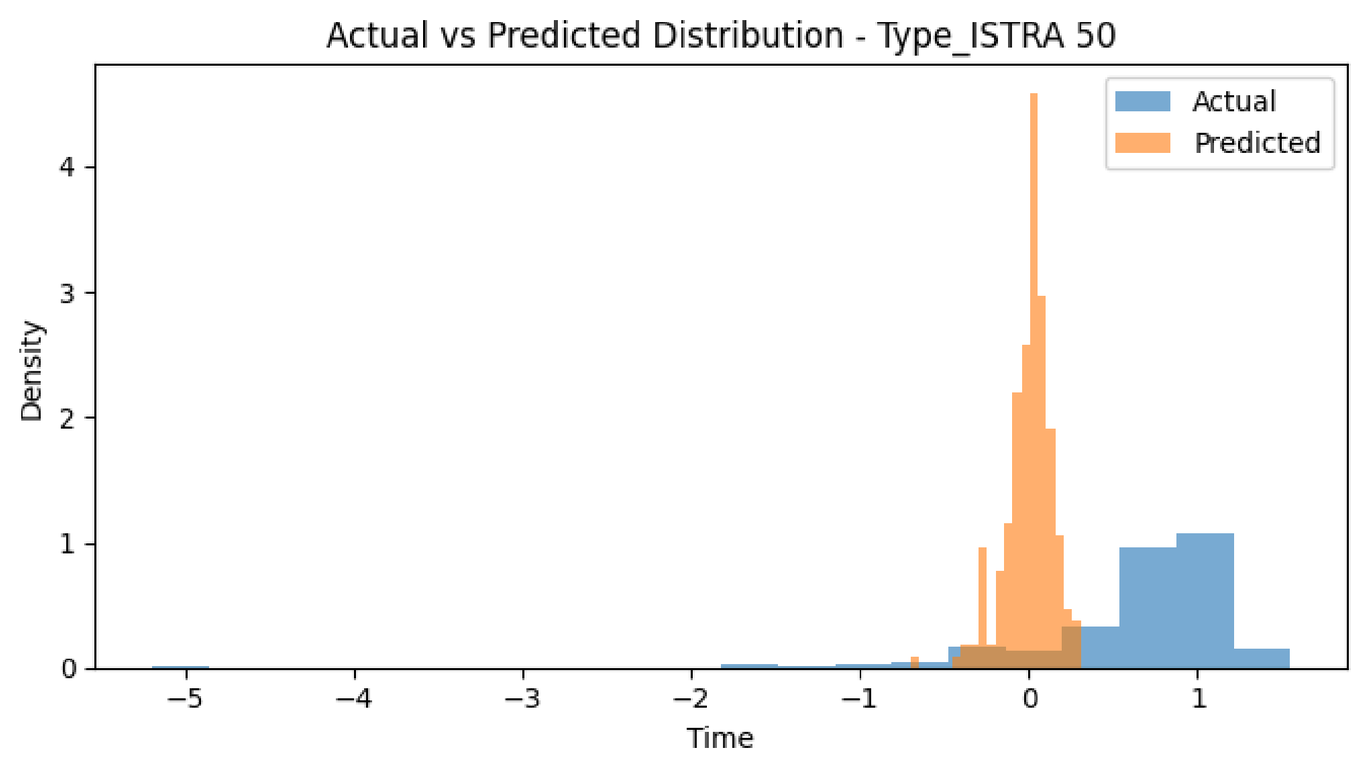
Results Experiment 5



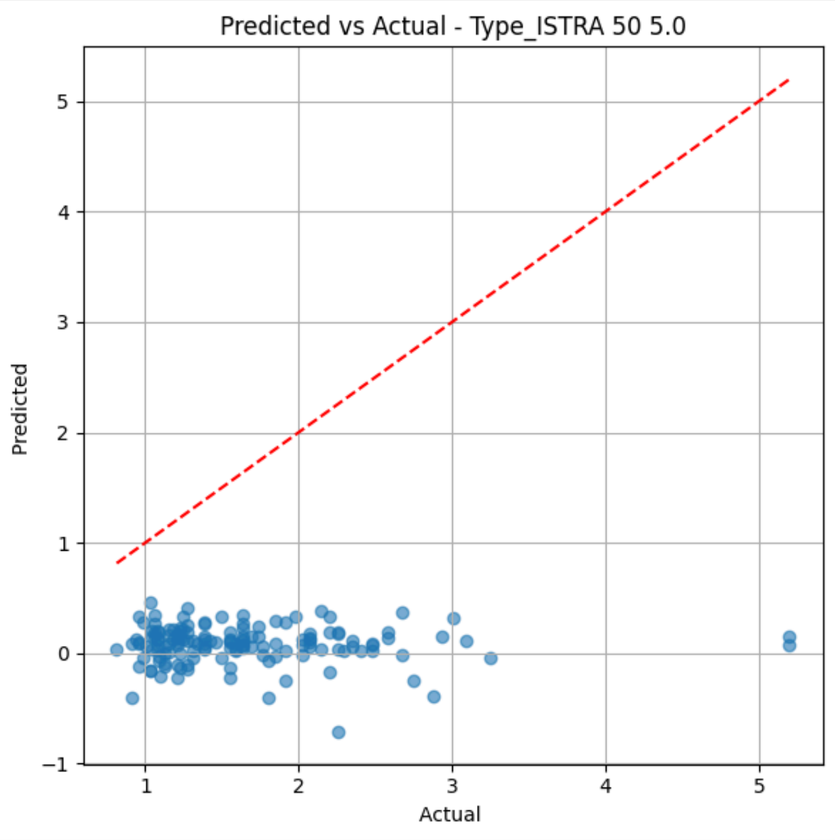


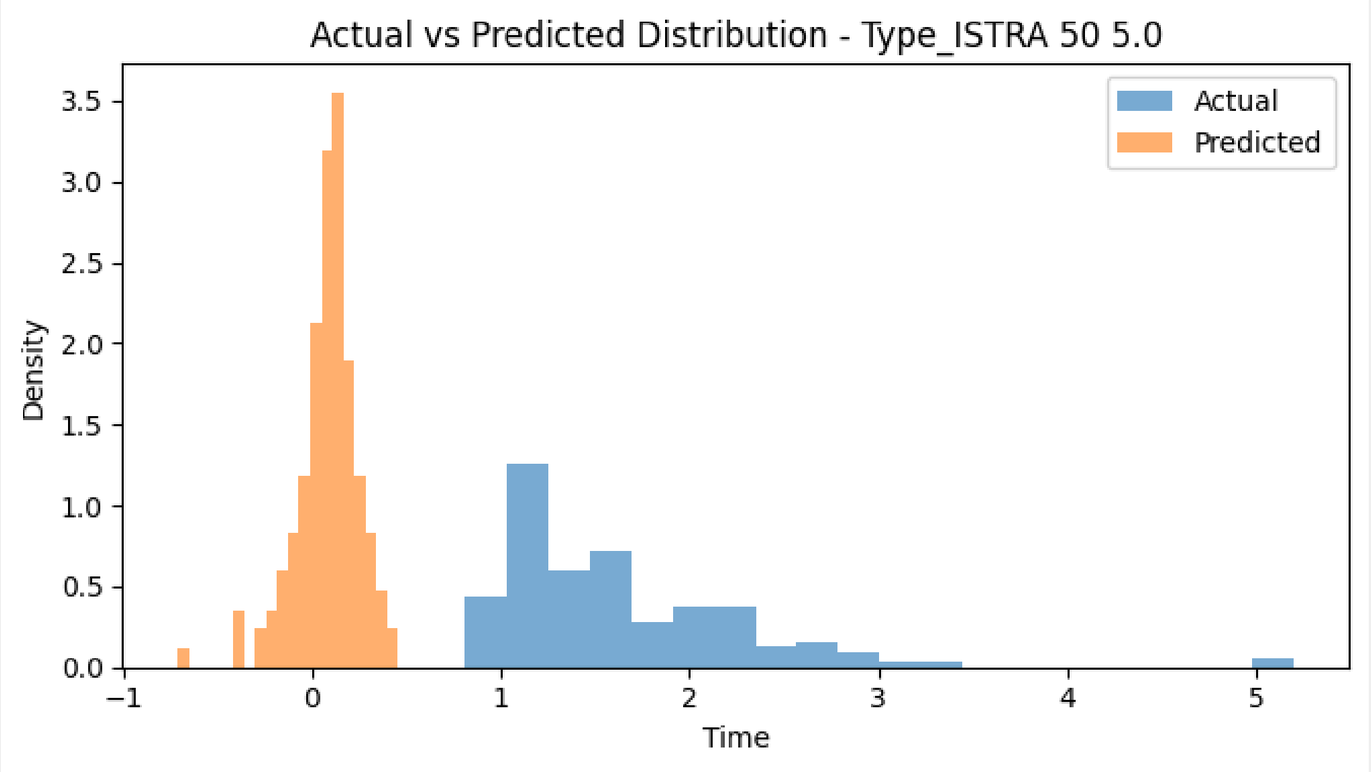
Results Experiment 5



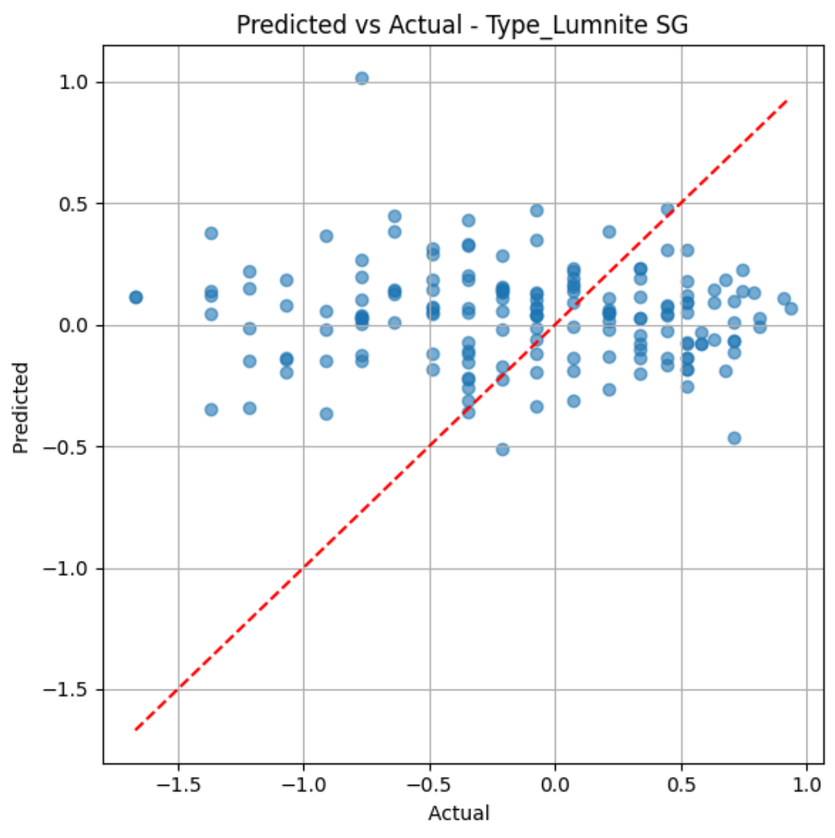


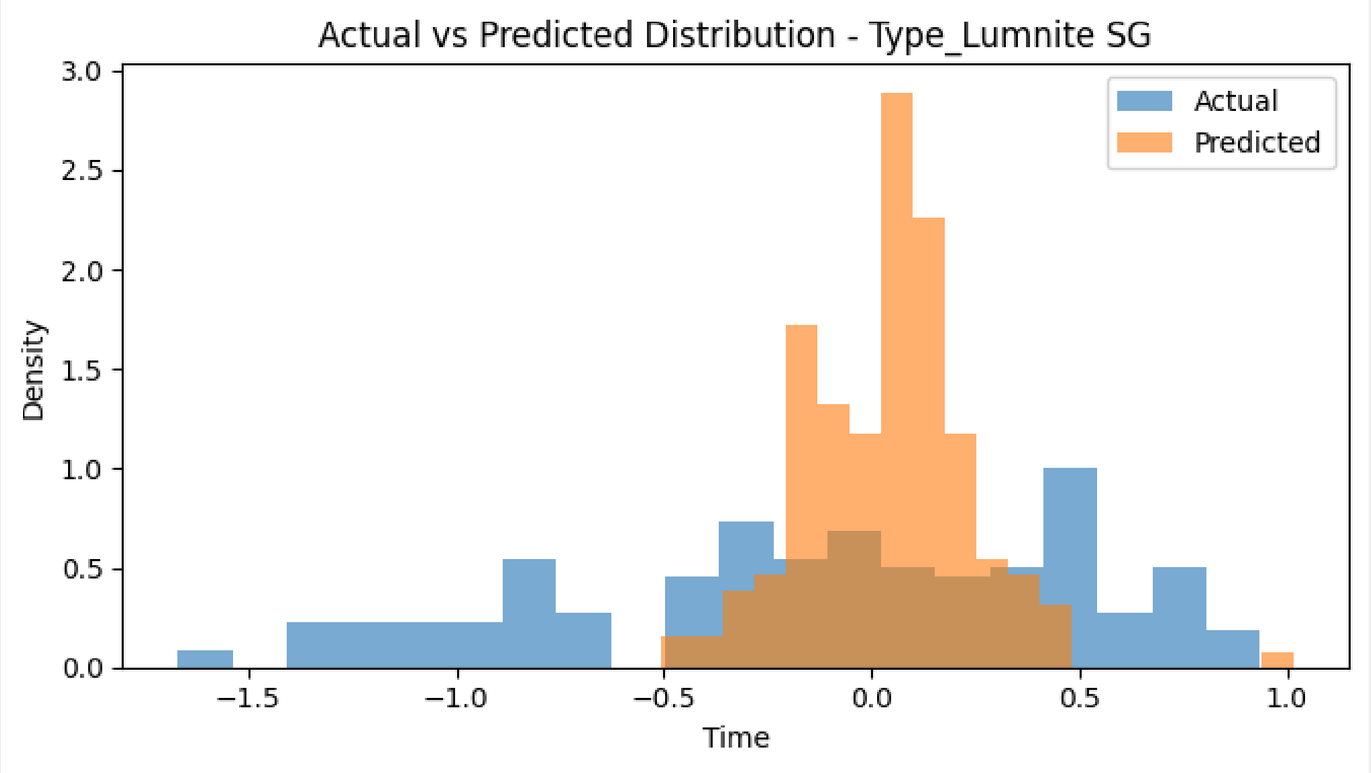
Results Experiment 5



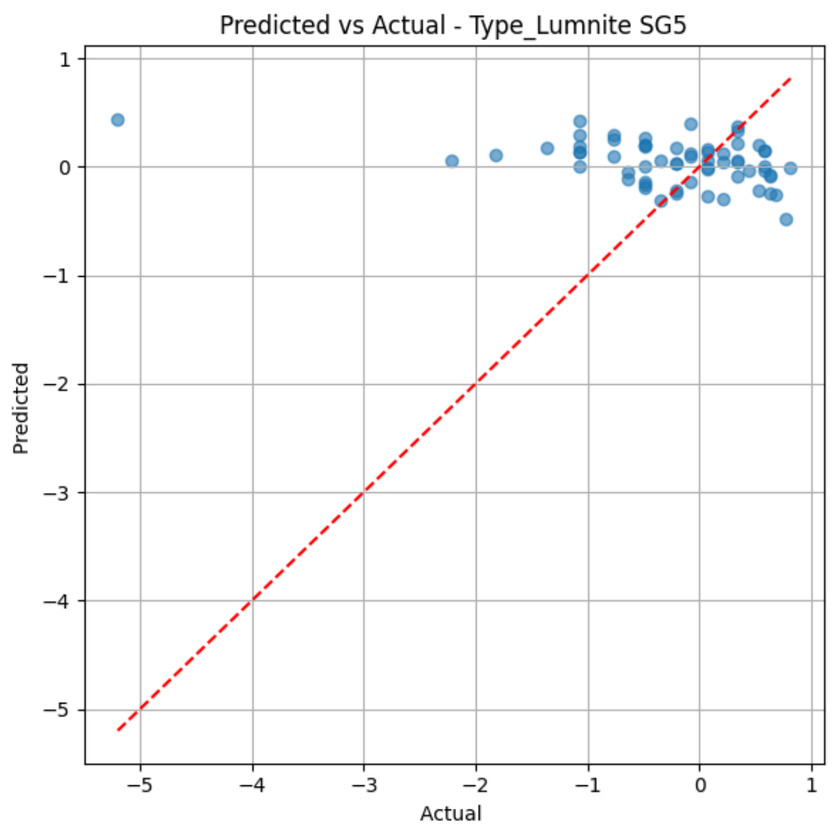


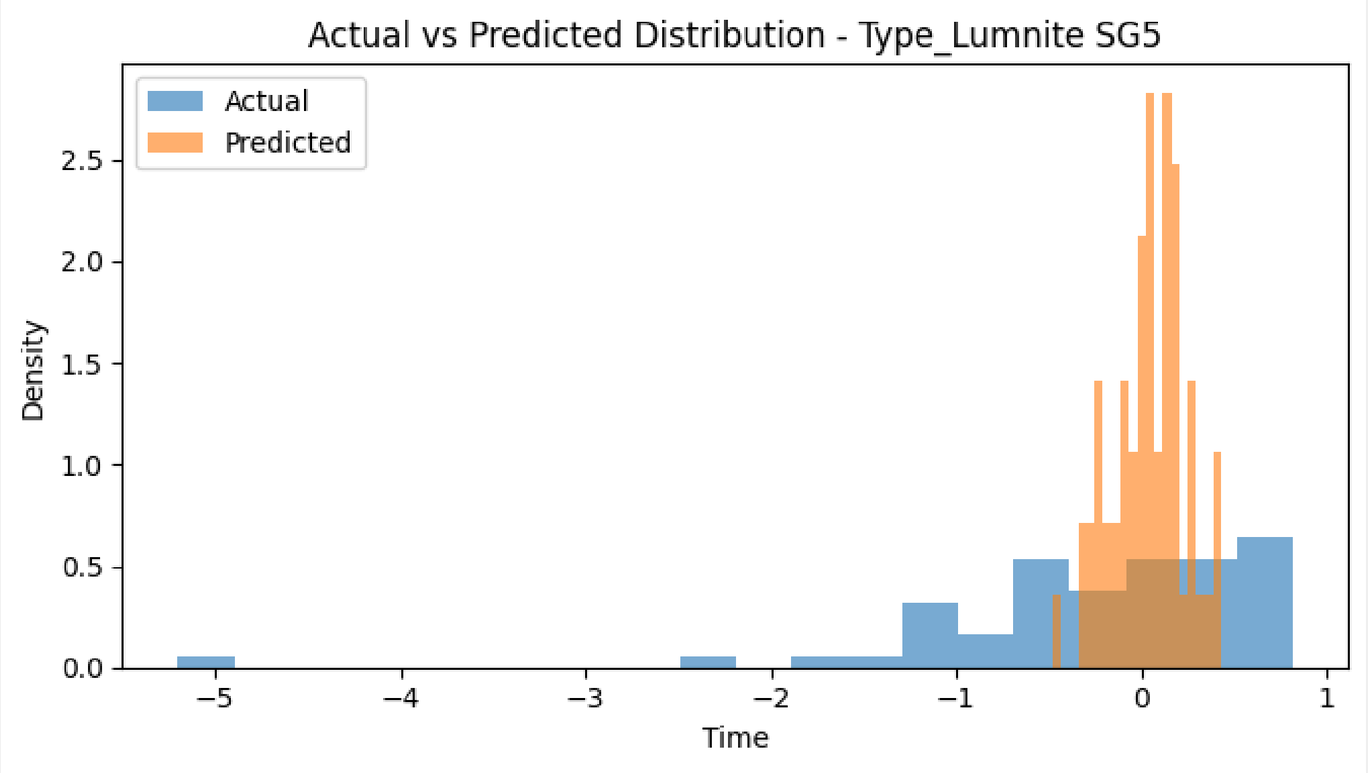
Results Experiment 5



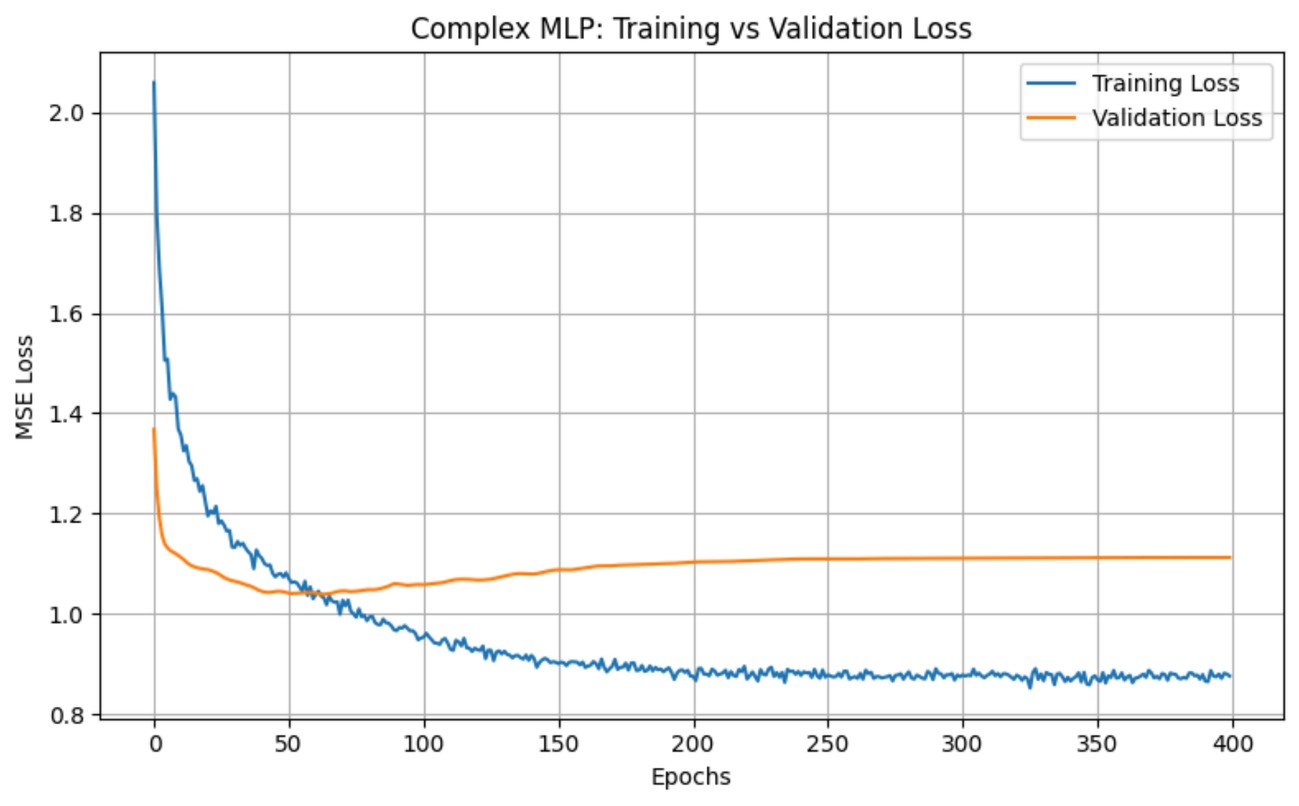


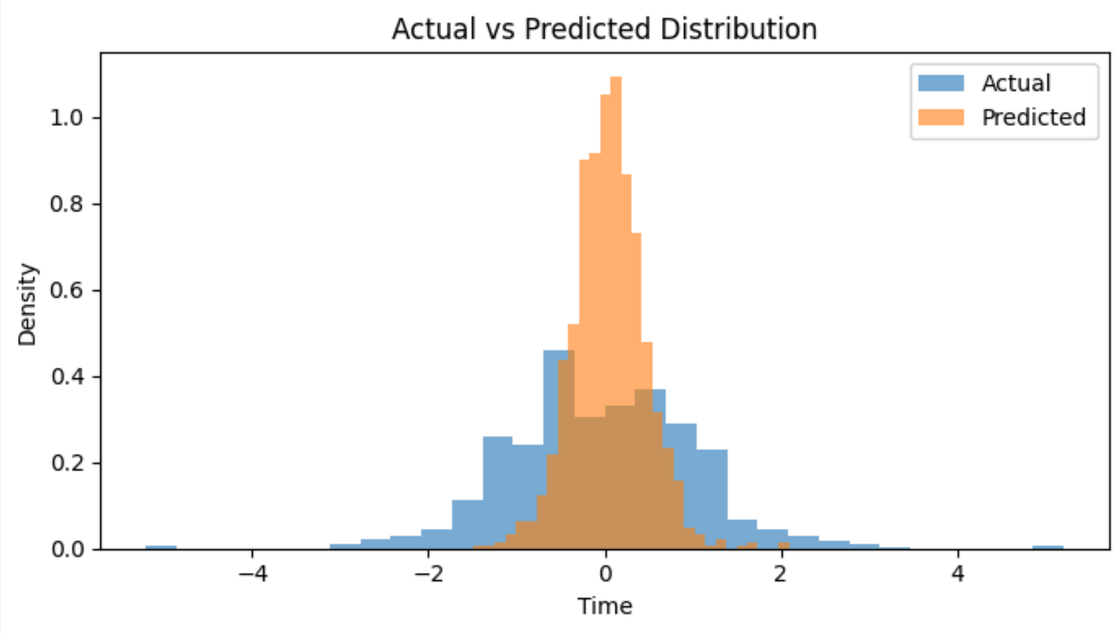
Results Experiment 5



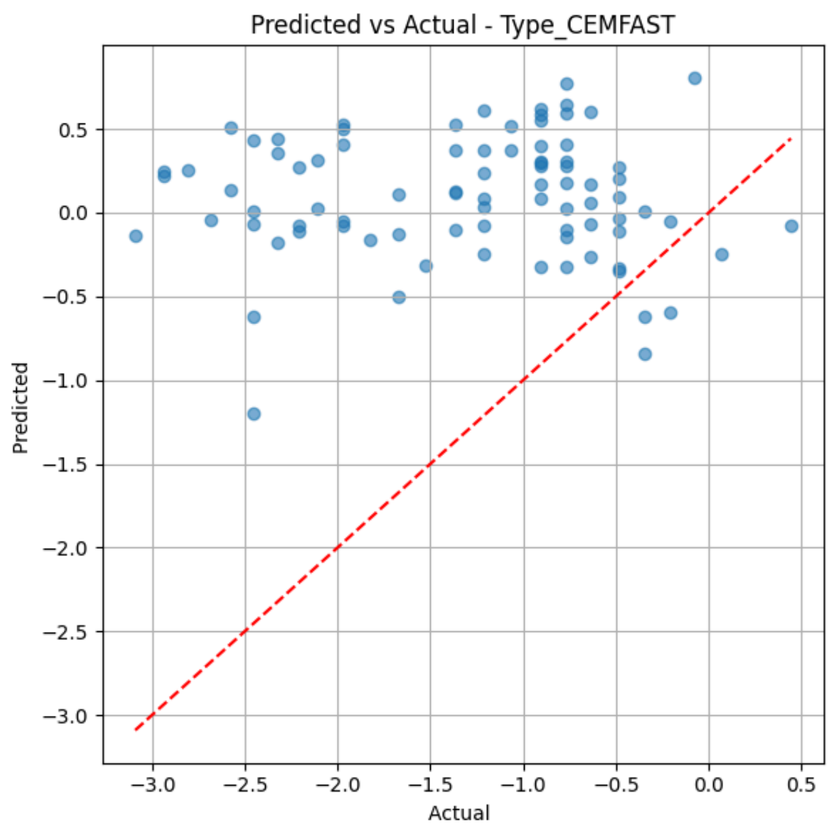


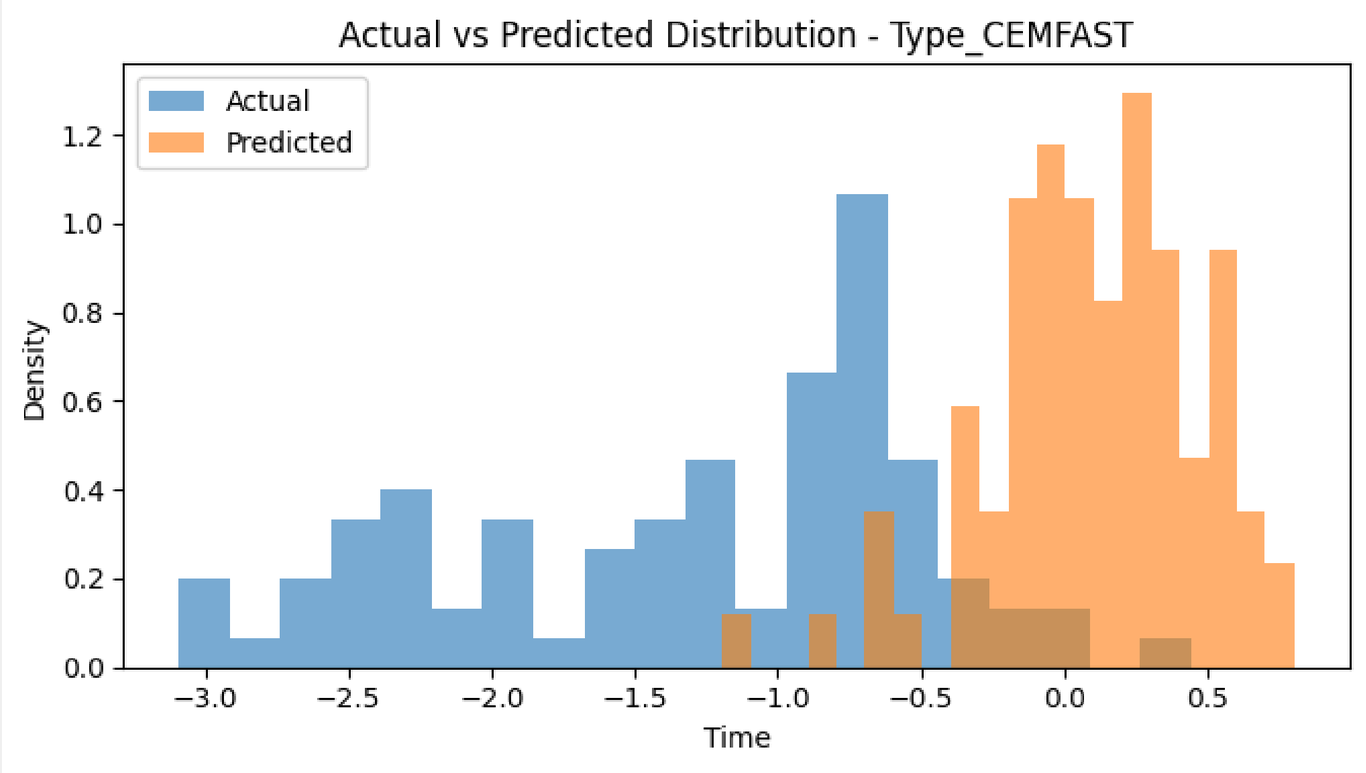
Results Experiment 6



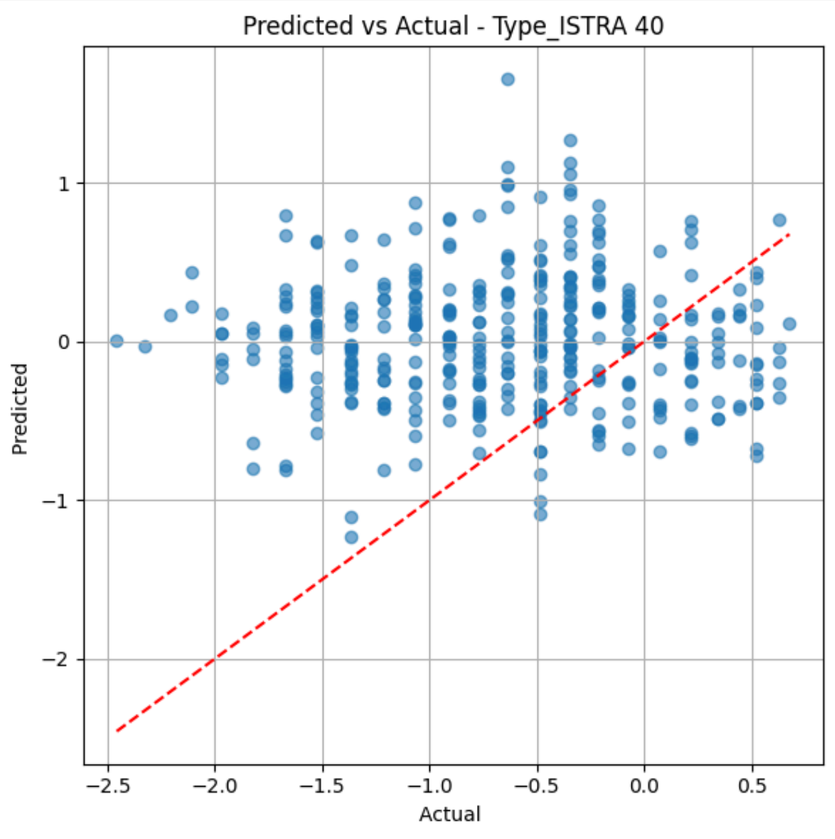


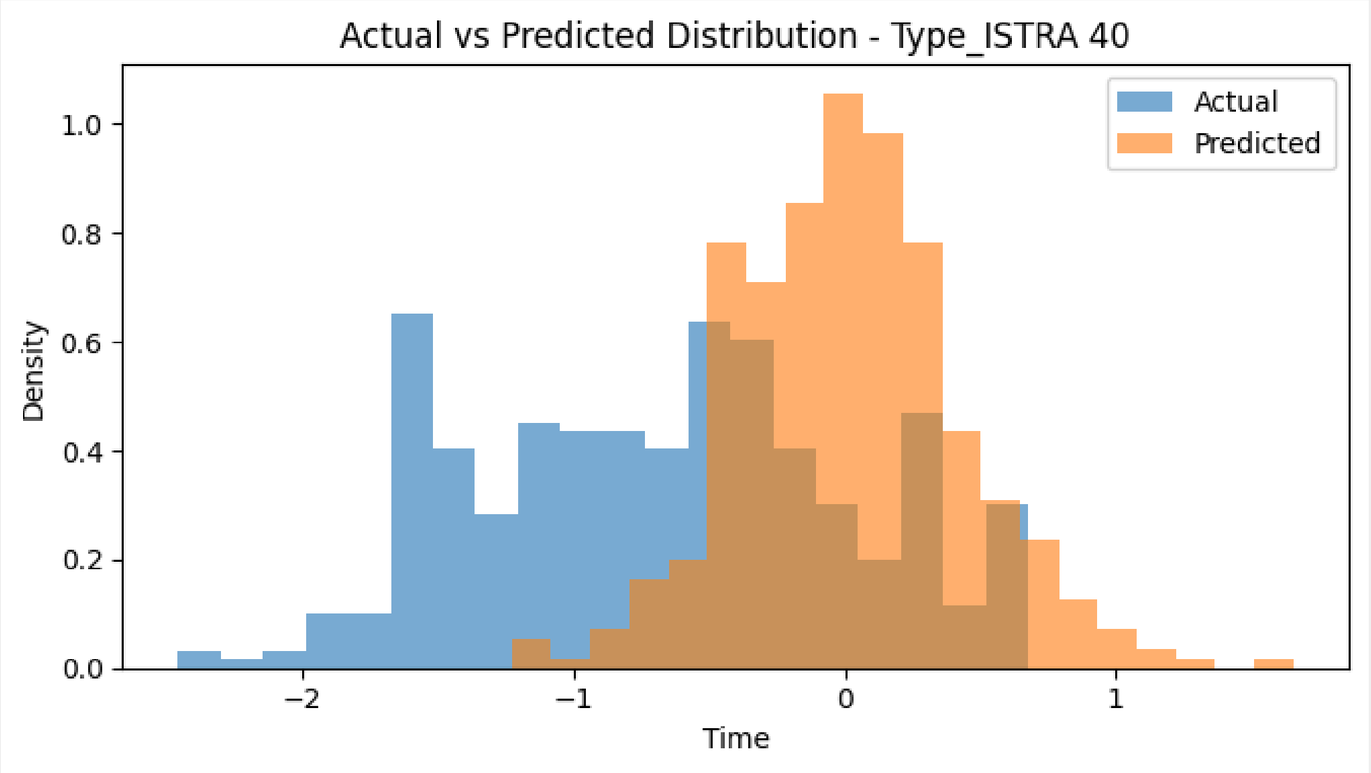
Results Experiment 6



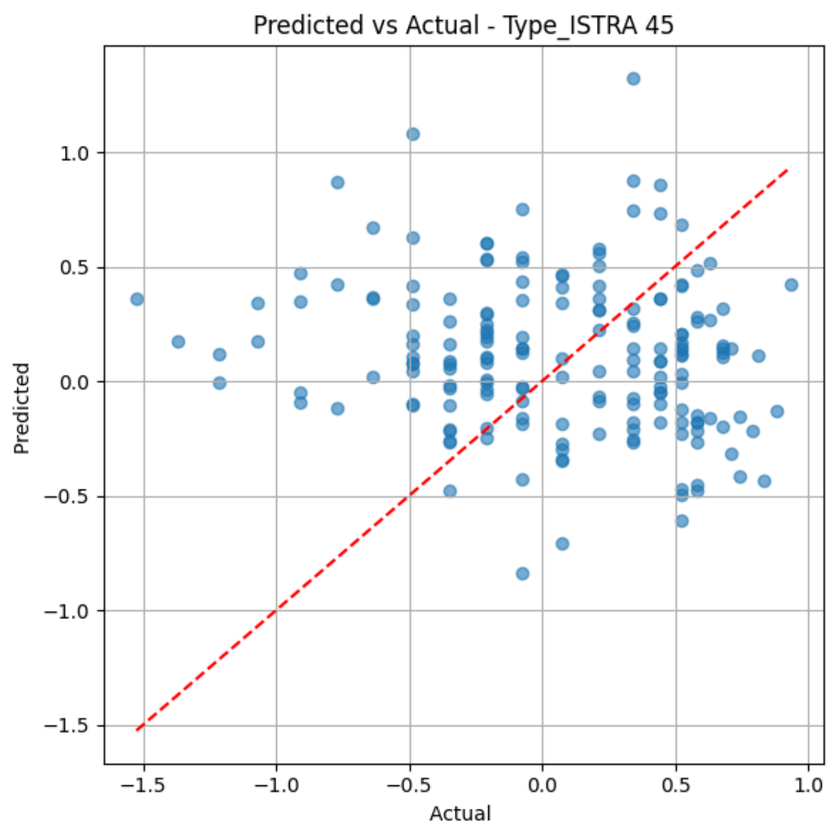


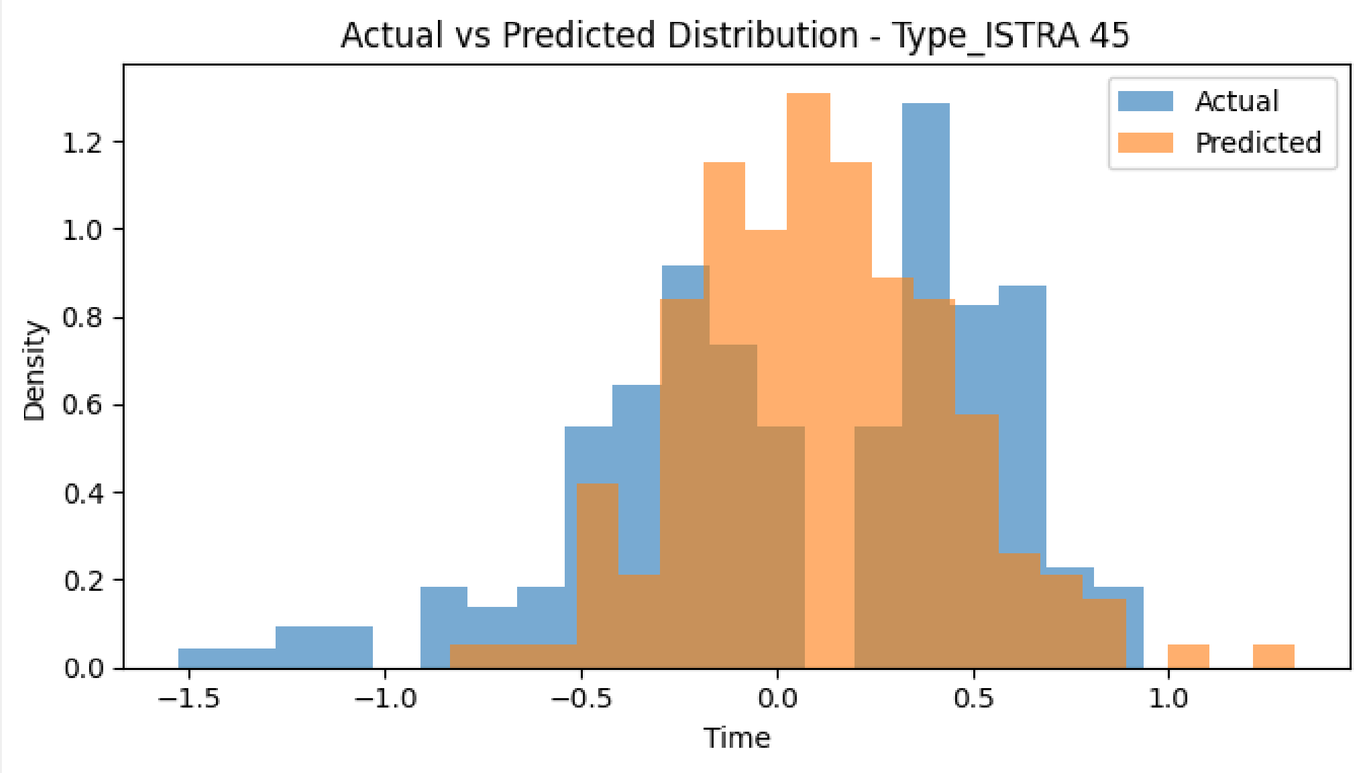
Results Experiment 6



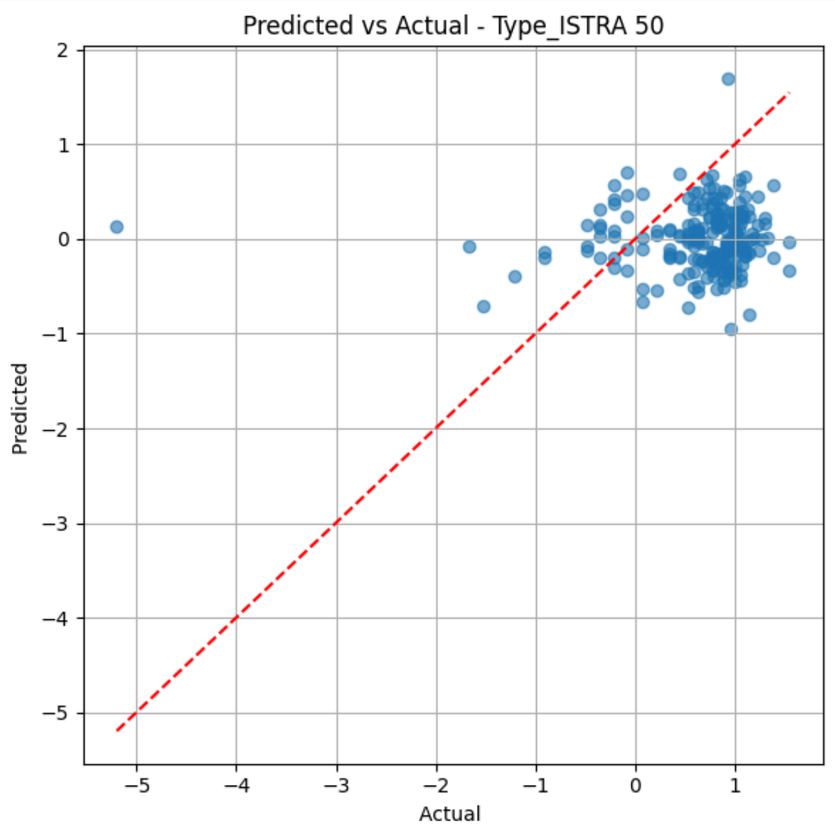


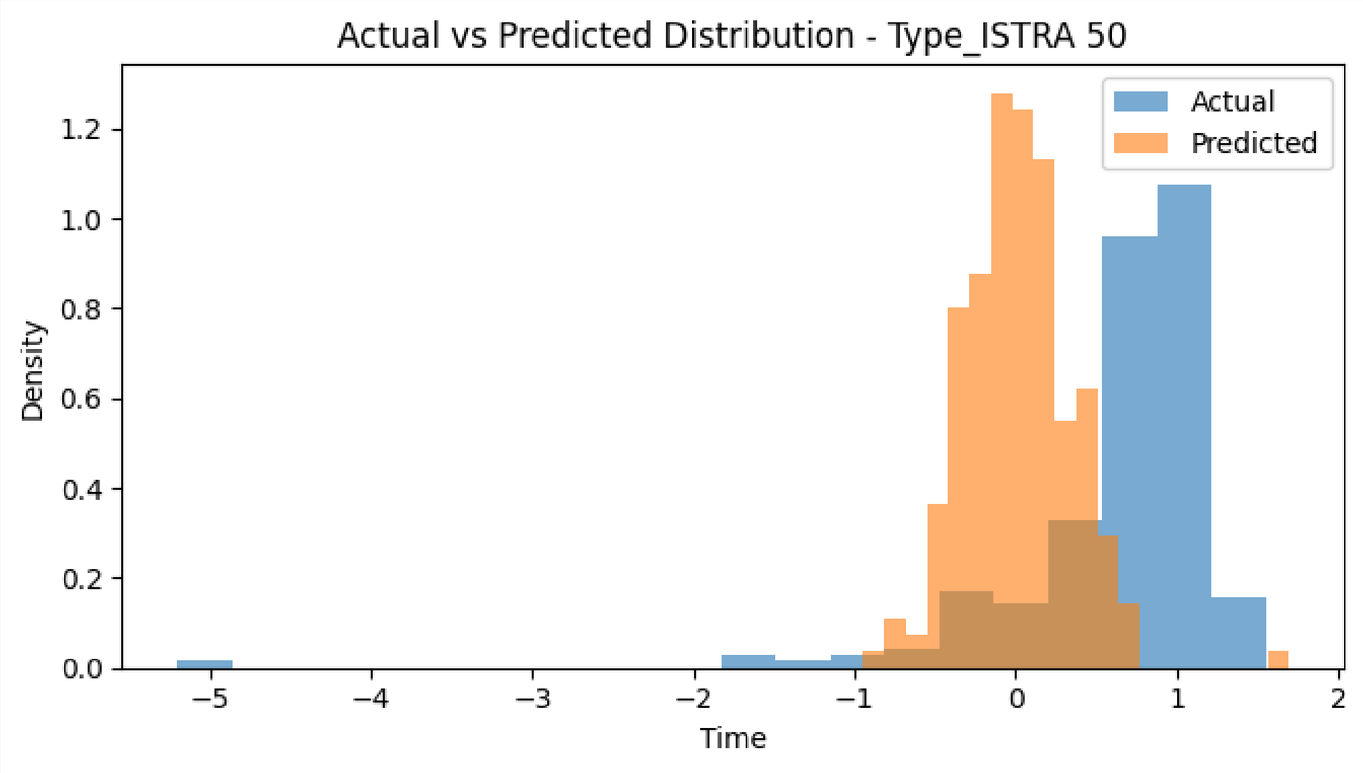
Results Experiment 6



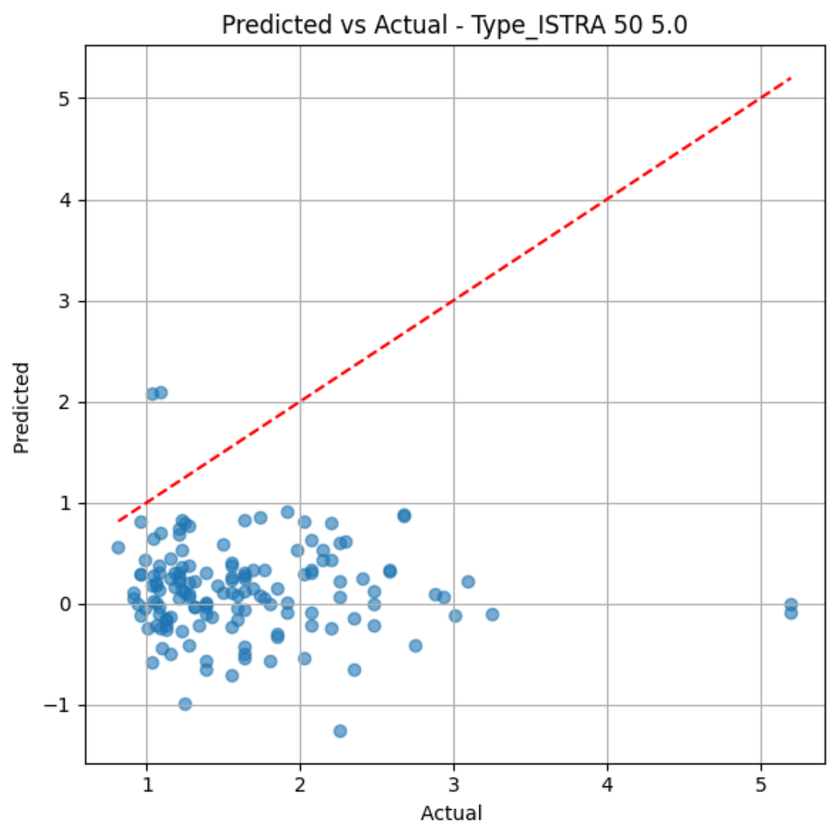


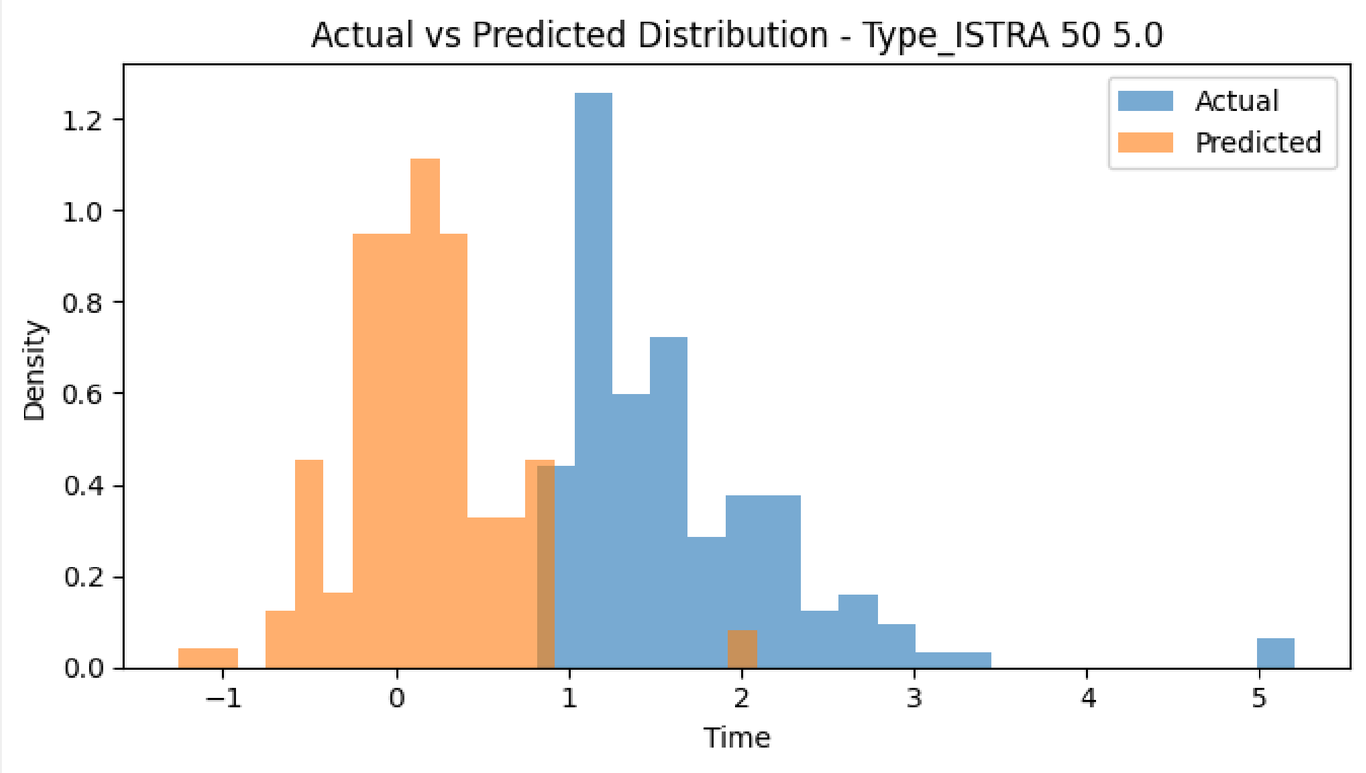
Results Experiment 6



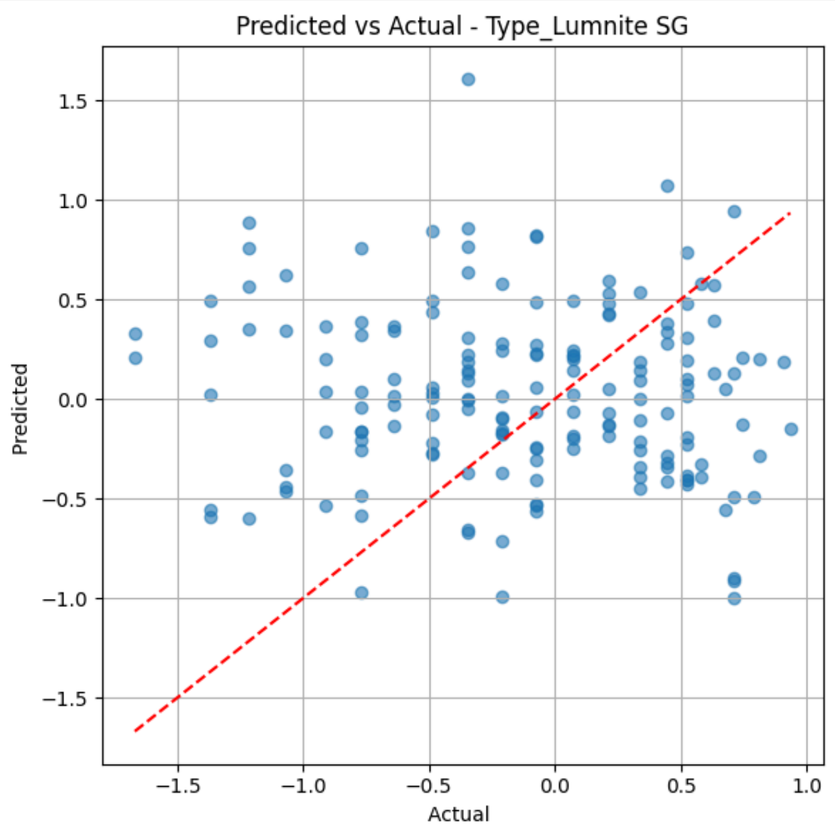


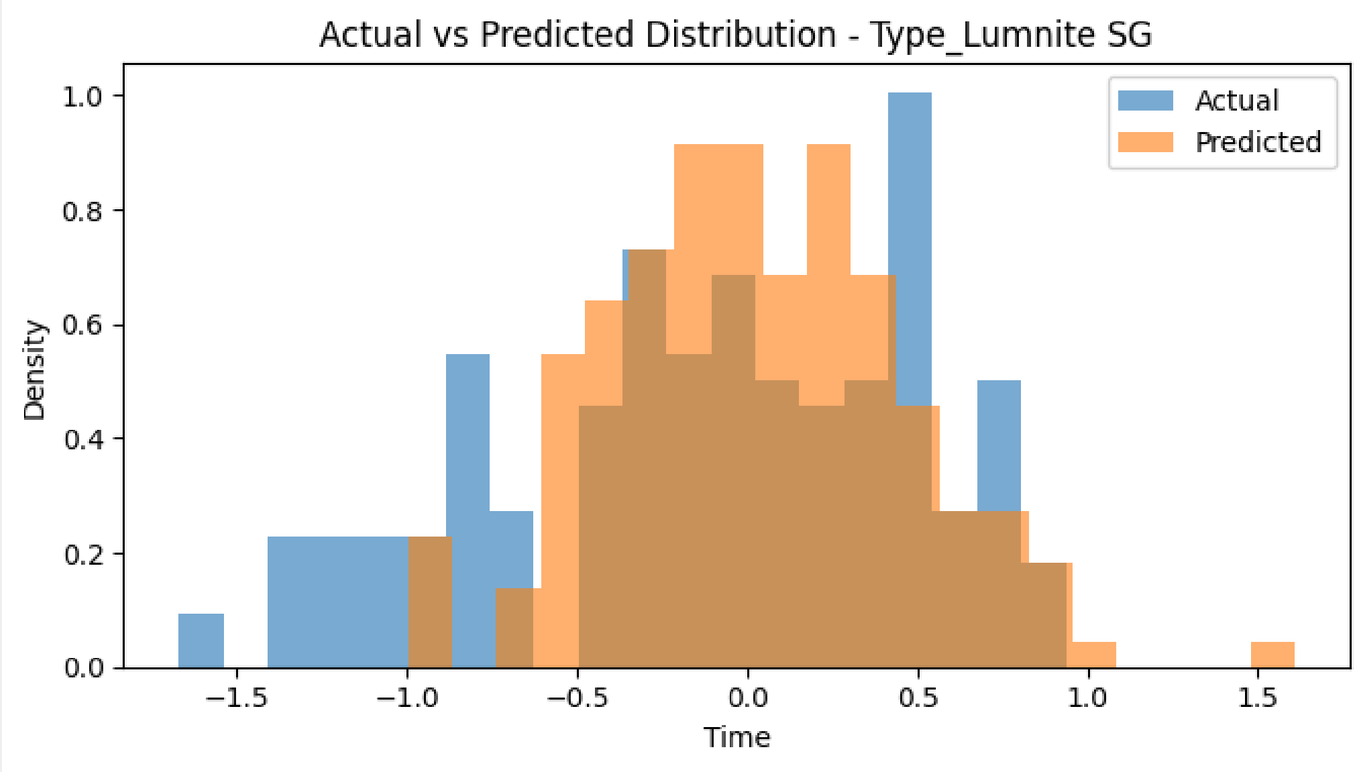
Results Experiment 6



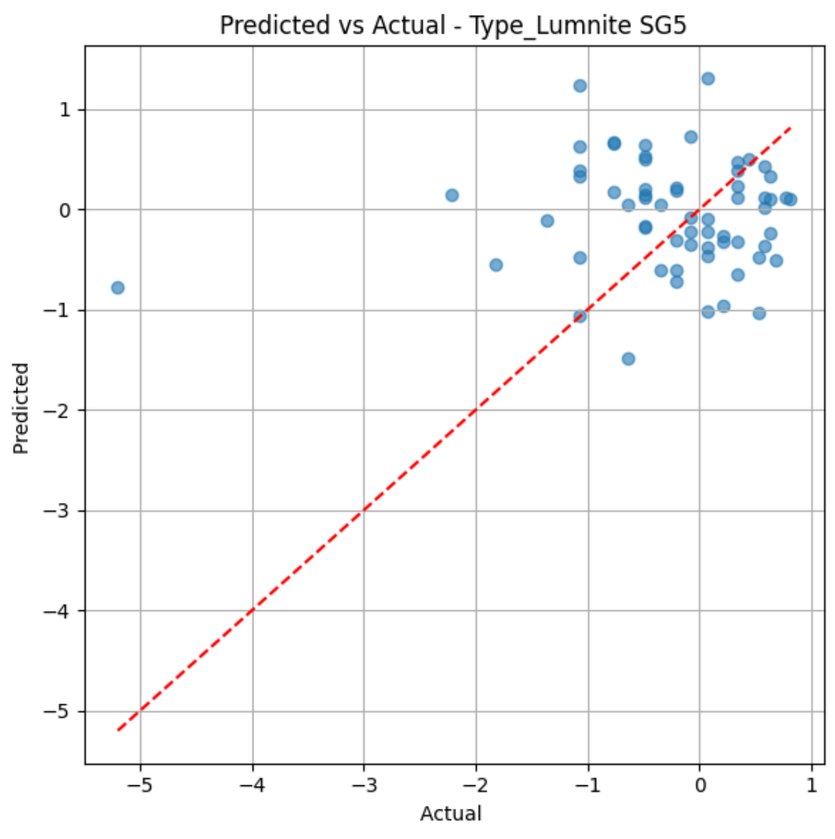


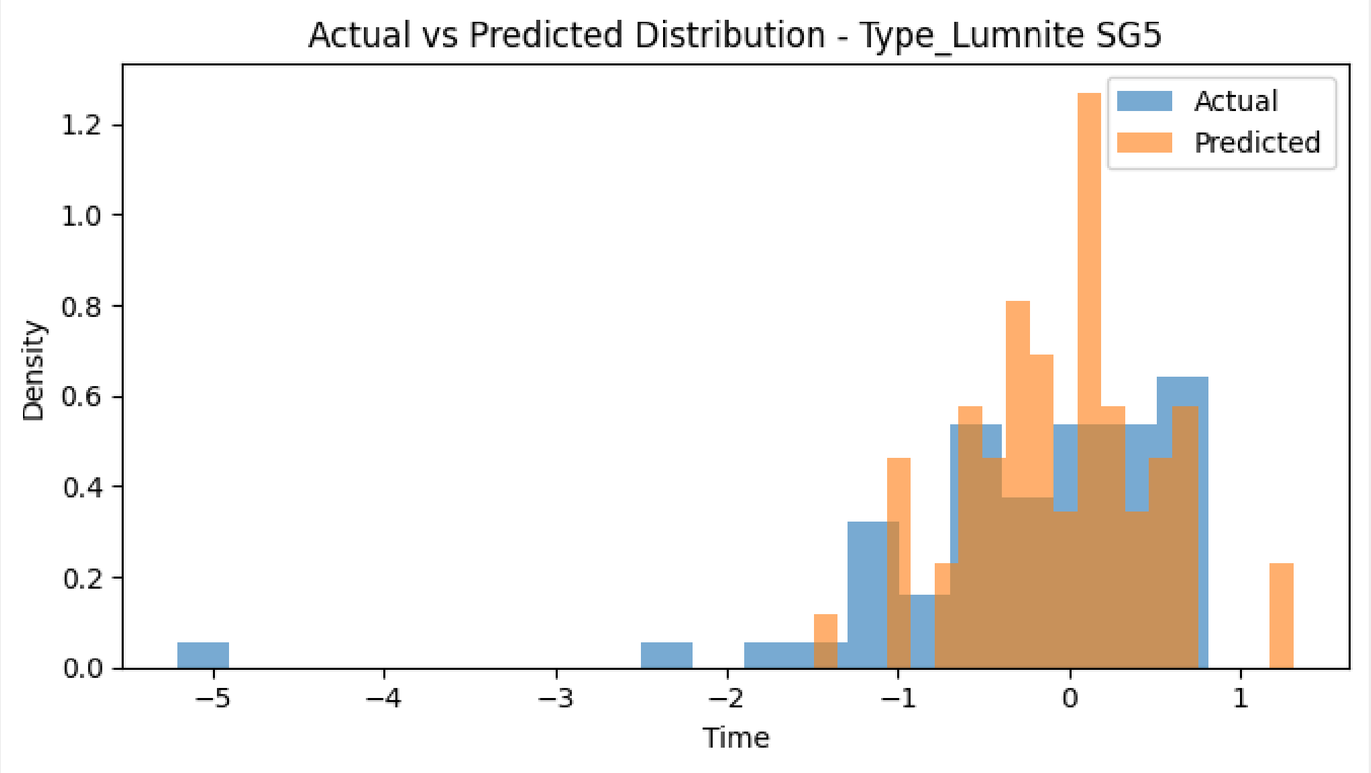
Results Experiment 6





Results Experiment 6





Conclusions

- Block-wise training allowed better control and interpretability.
- However, most blocks could not be compressed effectively (RRMSE > 0.2).
- PCA preprocessing alone did not improve compression results for autoencoders.
- Combining all non-OHE features in a single block led to high reconstruction error.

Autoencoders are sensitive to input dimensionality and redundancy — careful block design is essential for meaningful compression.

Conclusions

- Unsatisfactory results, no clear improvements amongst the different experiments.
- Even after trying to clean and transform the data, the most important information may still have been missing or hard to find.
- Models were not learning any meaningful mapping for some cement types, even after trying lots of different architectures.
- We believe this is due to:
  - High variability in real-world cement production processes.
  - A relatively small dataset for training robust models.
  - Many measurements and features are not precisely measured but taken “a ojo” (by approximation or intuition).
  - Two visually or chemically similar samples can still yield different initial setting times, making the target hard to model, because of the wrong measurements and because of the differences between cement type distributions.

THANK YOU!