Soft-Sensor Development for Ethanol Distillation Column

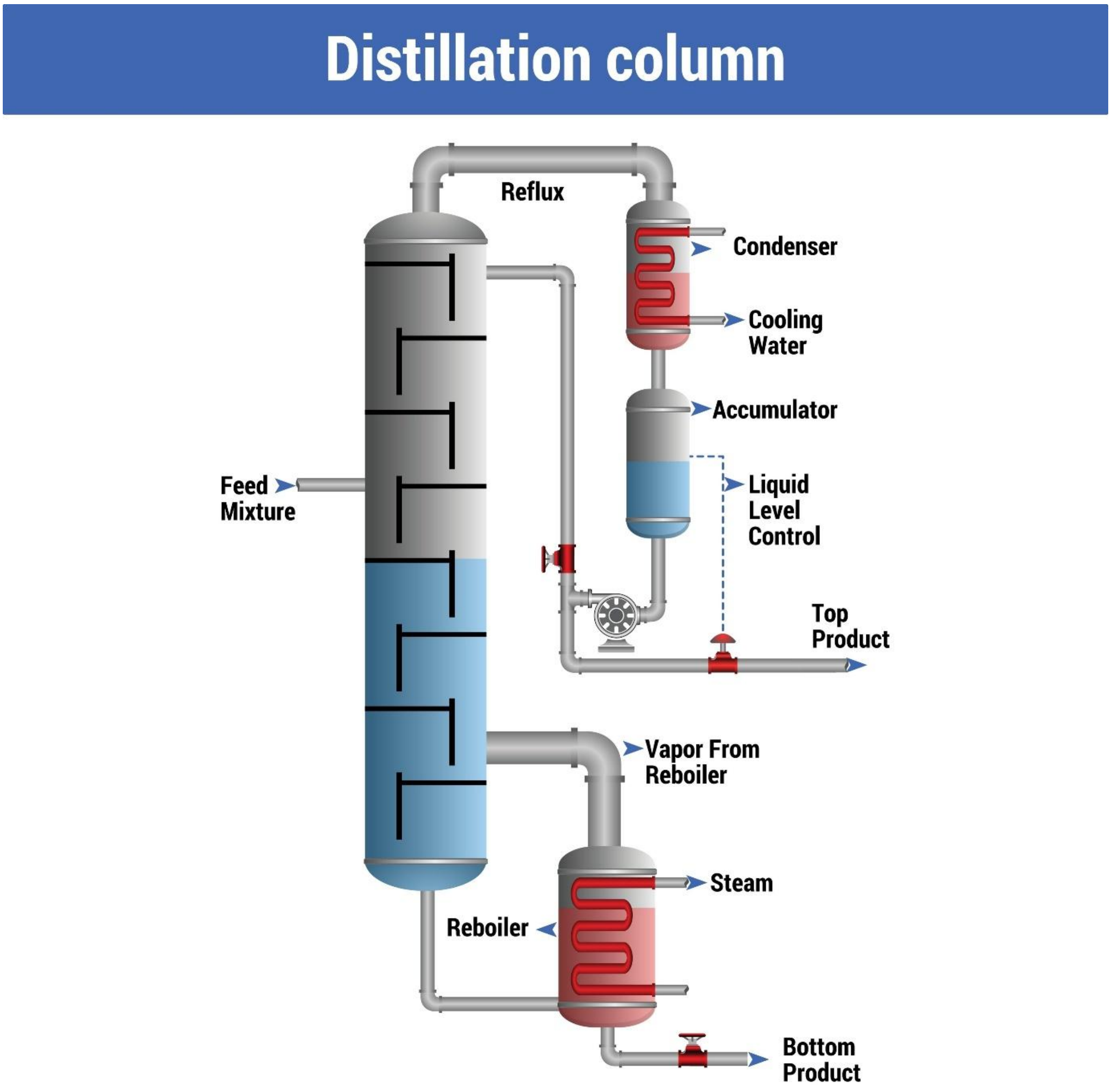
Dharmesh S

CH23B008

1. Introduction and Objective

The goal of this project was to develop a "soft sensor", a virtual sensor modelled using machine learning to predict the **ethanol composition** (purity) in a distillation column. In industrial settings, physical analysers are often expensive and have slow response times. By utilising readily available process data (temperatures, pressures, and flow rates), we can predict product quality in real-time.

My analysis utilised a dataset representing a distillation column with 14 tray temperatures (T1...T14), pressures, and flow rates (L, V, D, B, F).



2. Dataset Understanding & Preprocessing

Before feeding data into any model, the dataset had to be cleaned and prepared to ensure it respected physical reality. **The ethanol concentration** is set as the target.

2.1 Time Reconstruction

There is no explicit timestamp. However, for a dynamic process like distillation, the order of data points is critical.

- **Solution:** Reconstructed a time index assuming a constant sampling rate. Created a time_h feature where every step equals 0.1 hours.
- **Reasoning:** This allows for time-series plotting and valid train/test splitting. Without this, we cannot model the dynamic changes (lag) of the system.

2.2 Handling Missing Data & Constants

Real-world sensor data often contains gaps or "flatline" sensors.

- **Imputation:** Forced all columns to numeric types and used **linear interpolation** for missing values. Interpolated over mean-filling because process variables (like temperature) usually drift gradually; they don't jump randomly.
- **Removing Constants:** Removed columns with only one unique value (zero variance). These provide no information to the model and can actually cause mathematical instability in algorithms like Lasso regression.

3. Physics-Informed Feature Engineering

A key part of my code was creating features that represent the actual chemical engineering principles occurring in the column.

3.1 Reflux Ratio (R)

Calculated the Reflux Ratio using the formula:

$$R = \frac{L}{V + 10^{-9}}$$

where L is liquid flow, and V is vapour flow.

- **Significance:** In distillation theory, the reflux ratio is the primary control variable. Increasing reflux generally improves the separation efficiency (purity). Explicitly giving this ratio to the model made it easier for the algorithm to learn the separation dynamics.
- 10^{-9} is added to V to prevent Division by Zero error.

3.2 Temperature Profiling

Instead of just feeding raw temperatures (T1, T2...), engineered features are used to describe the *shape* of the temperature profile:

- **Differentials (Delta T):** Calculated as the difference between adjacent trays ($T_{i+1} - T_i$). This indicates where the mass transfer is happening most aggressively.

- **Temperature Range:** $T_{max} – T_{min}$. This proxies the overall energy balance of the column.

3.3 Lag Features (Process Dynamics)

Distillation columns are not instant. If the steam flow is changed at the bottom, it takes time for the vapour to travel up and affect the top product purity.

- **Implementation:** Created "Lag features" for the ethanol target.
- **Reasoning:** This allows the model to look at the *past* behaviour to predict the *current* state, effectively capturing process delays and control system dynamics.

4. Modelling Strategy

4.1 Chronological Splitting

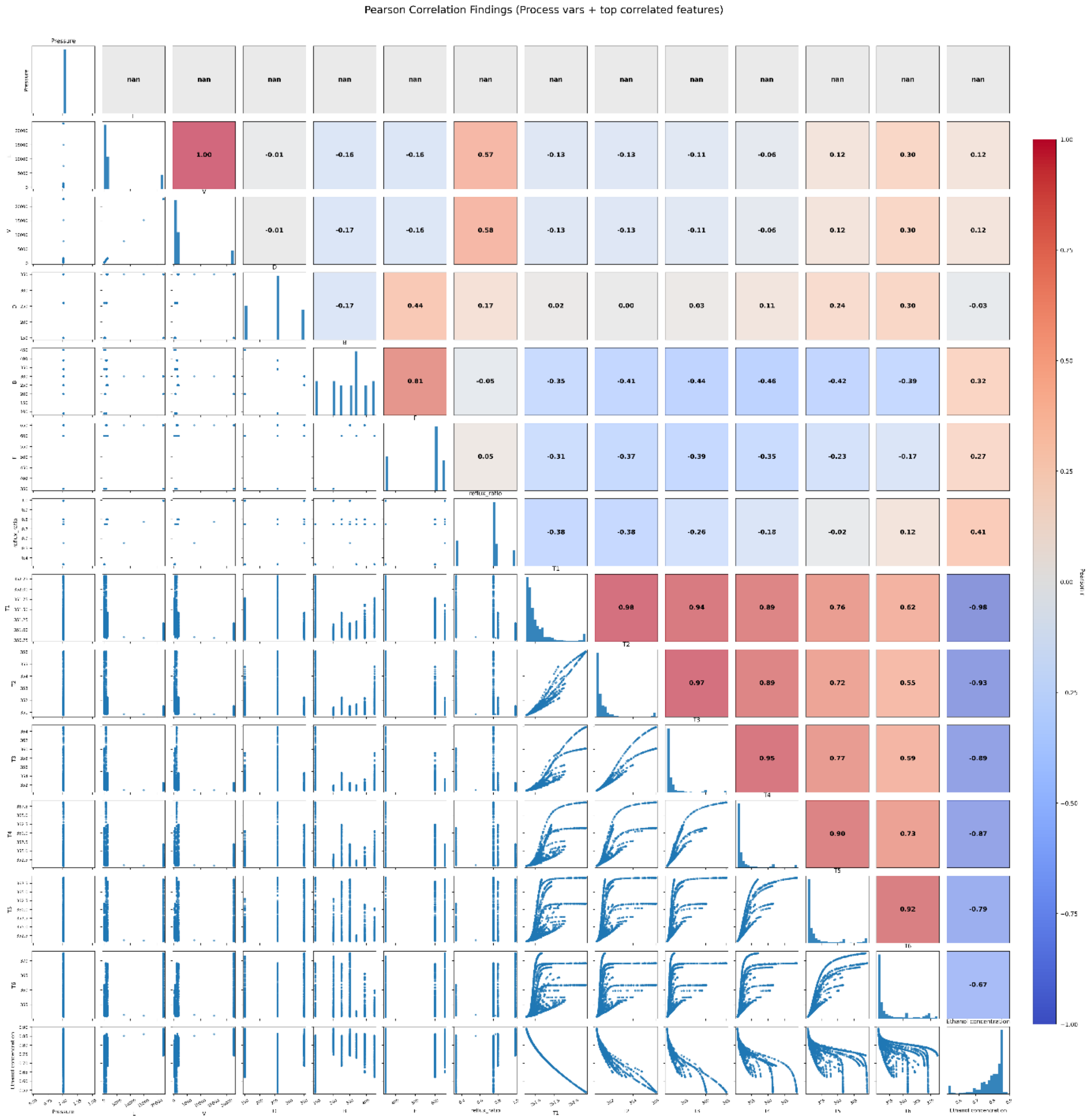
Specifically avoided random shuffling of the data. Instead, used the **first 80% of the rows for training and the last 20% for testing**.

- If shuffled data is used for training the model, it will use "future" data points to predict the past. This is known as **data leakage** and would result in an unrealistically high accuracy that would fail in a real plant.

4.2 Exploratory Analysis (PairGrid)

This helps us understand the correlation between variables. The diagonal histograms showed the distribution of the data, while the scatter plots helped identify multicollinearity, instances where variables (such as adjacent tray temperatures) move in perfect synchrony.

The PairGrid is only generated for process variables and the variables with high correlation.



5. Pseudo Code

Step	Action	Description
1	**LOAD dataset**	Initial data ingestion.
2	**CLEANING**	TRY reading with separators (,, ;, \t). FORCE all columns to numeric.
3	**IMPUTATION**	INTERPOLATE missing values (preserves time-series continuity). FILL any remaining NaNs with column median.

Step	Action	Description
4	**TIME INDEX**	IF no time column: CREATE time_h = step * 0.1 (uniform 0.1 hr sampling).
5	**FEATURE ENGINEERING**	DETECT ethanol as target. IF L and V exist: CREATE reflux_ratio = L / (V + 1e-9).
6	**TEMPERATURE FEATURES**	IDENTIFY T1...Tn. CREATE: dT features, T_range, T_mean (capture profile shape).
7	**DYNAMIC FEATURES**	CREATE lag features for ethanol (1–5 steps). DROP rows with NaNs from lagging.
8	**FINAL PREP**	REMOVE constant columns. SPLIT dataset **chronologically** (80% / 20%). STANDARDIZE features.
9	**REGRESSION**	TRAIN: Linear, Ridge, Lasso, MLP. EVALUATE: MAE, RMSE, R^2 (Soft-Sensor development).
10	**CLASSIFICATION**	qcut target into 3 classes. Train Logistic Regression. Generate classification report (Alarm System).
11	**CLUSTERING**	Standardize temperatures. SELECT best k (2–10) using silhouette score. Fit KMeans. PLOT mean cluster profiles (Operating Modes).

6. Results & Discussion

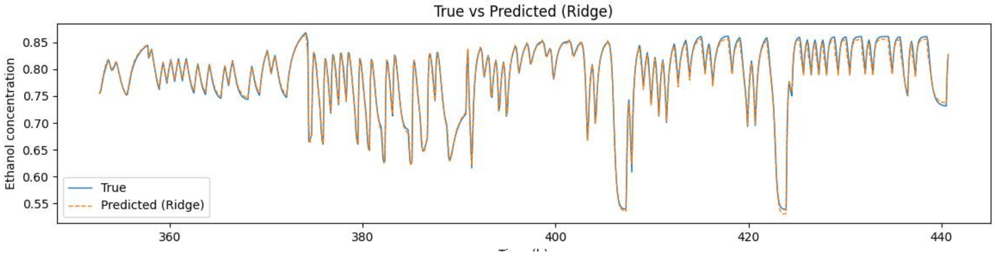
6.1 Regression Analysis (Predicting Purity)

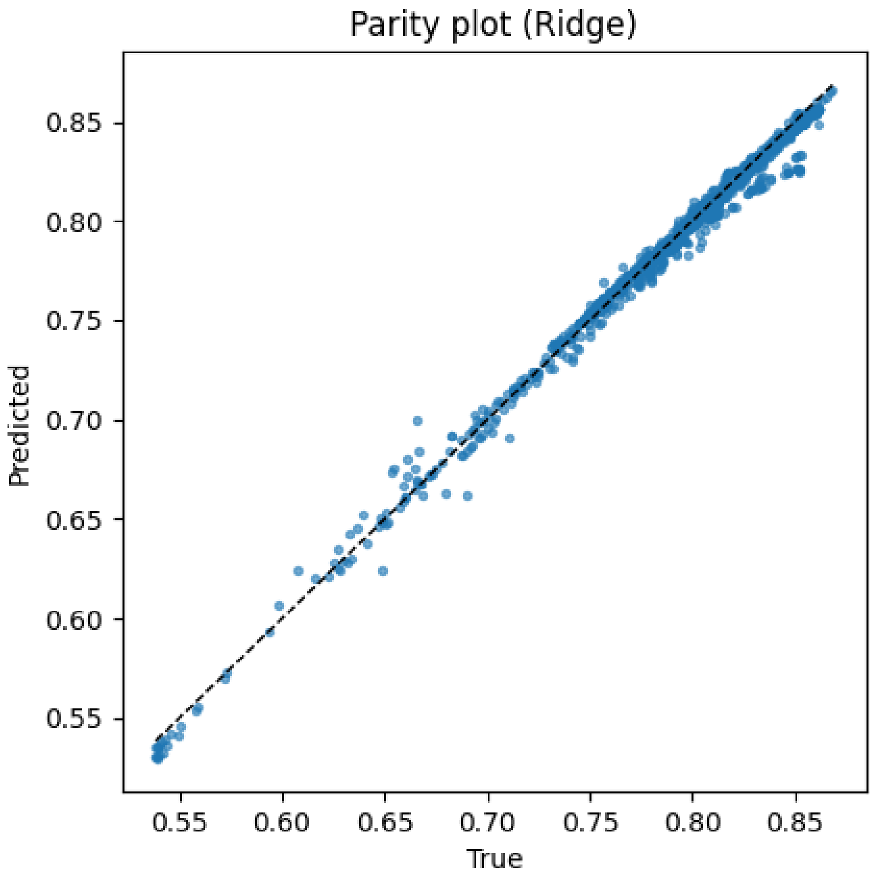
Trained using four regressors.

Model	MAE	RMSE	R2 Score
Ridge	**0.0043**	**0.0062**	**0.991**

Model	**MAE**	**RMSE**	**R2 Score**
Lasso	0.0049	0.0067	0.989
MLP (Neural Net)	0.1037	0.1274	-2.765
Linear Regression	0.2944	0.3539	-28.06

- **The Failure of Linear Regression:** The standard Linear Regression model failed spectacularly, with a negative R^2. This is due to **multicollinearity**. Tray temperatures (T1, T2, T3...) are highly correlated. This confuses the standard Ordinary Least Squares (OLS) algorithm, causing the coefficients to become unstable and potentially explode.
- **The Success of Ridge/Lasso:** The Ridge regression performed best ($R^2 = 0.99$). Ridge adds **L2 Regularization**, which penalizes large coefficients. This effectively handles the correlated temperature inputs, resulting in a robust, near-perfect prediction.



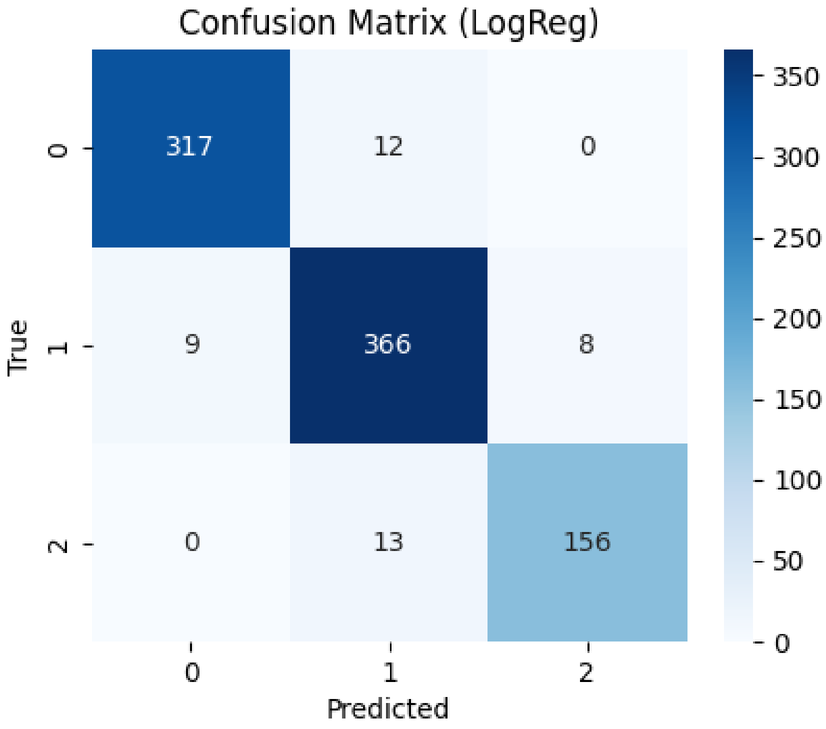


6.2 Classification (Quality Control)

Converted the continuous ethanol purity into three discrete classes (Low, Medium, High) using quantile binning (qcut).

- **Result:** The Logistic Regression classifier achieved **95.23% accuracy**.
- **Application:** This model acts as an "alarm system." Even if we don't know the exact percentage, the operator can be alerted if the system enters a "Low Purity" state.

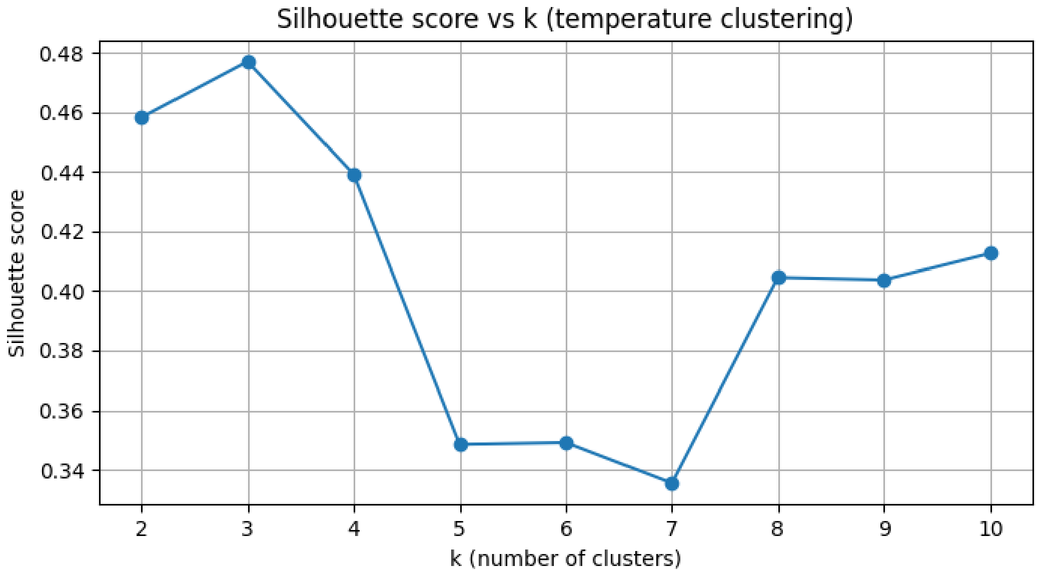
Class	Precision	Recall
0 (Low)	0.97	0.96
1 (Medium)	0.94	0.96
2 (High)	0.95	0.92



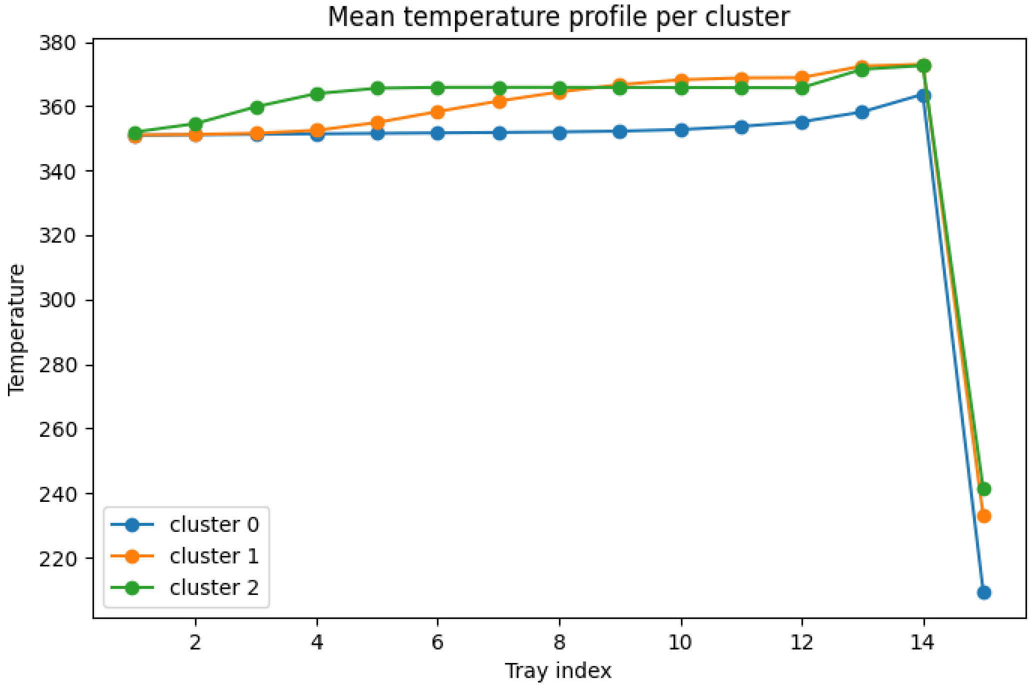
6.3 Unsupervised Clustering (Operating Modes)

Used KMeans clustering on the temperature data to see if different "operating modes" could be detected without being told what they were.

- **Finding K:** Used the Silhouette Score to determine the optimal number of clusters. The score peaked at **k = 3,** suggesting that the column has three distinct operating regimes.



- **Interpretation:** By plotting the mean temperature profile for each cluster, we can likely identify these states as "Stable Operation," "Disturbance/Upset," and "Startup/Transition".



7. Conclusion

This project demonstrated that a **Ridge Regression model** is the optimal choice for this distillation column, achieving an R^2 of 0.99.

Key Learnings:

1. **Physics Matters:** Including the Reflux Ratio and Lag features was essential for capturing the process dynamics.

2. **Regularization is Mandatory:** Simple linear regression cannot handle the highly correlated nature of tray temperatures; regularization (Ridge/Lasso) is required to prevent model failure.

3. **Data Continuity:** Respecting the time-series nature of the data (via chronological splitting and lag features) ensures the model is realistic and does not "cheat" by looking into the future.

4. **The Neural Network (MLP) underperforms, confirming the dataset is primarily linear:** The MLPRegressor showed a disastrous negative R^2, indicating that its complexity was unnecessary and the underlying relationship between inputs and output is fundamentally better captured by simpler, regularized linear models (Ridge/Lasso).

5. **The Classification model is robust with (>95%) accuracy:** This model provides a reliable digital alarm system, allowing operators to quickly detect and correct off-specification product purity events.

6. **Clustering reveals real operating regimes, supporting advanced monitoring or fault detection:** The identification of distinct temperature profile clusters is invaluable for advanced process monitoring, indicating when the column transitions between stable and unstable operational modes.