Homework 3 – Report

University: Shiraz University
Course: Artificial intelligence – Spring 2025
Instructor: Prof. Zohreh Azimifar
Student: Salar Rahnama – Amirreza Baghban
Student ID: 40131850 - 40131840
Assignment Title: Programming Task
Due Date: God and TAs know

Part I: Data Selection and Cleaning

City Selection

We focused on data from Sydney using the Location column in weatherAUS.csv.

Missing Value Analysis

Feature	Missing Values
MinTemp	3
MaxTemp	2
Humidity3pm	13
WindSpeed3pm	25
Pressure3pm	19
Rainfall	6

🔹 Missing Value Handling Strategy

Feature	Strategy	Rationale
Rainfall	Drop rows	Target variable — no prediction possible if missing
MinTemp/MaxTemp	Mean imputation	Normally distributed, few missing
Humidity3pm	Median imputation	More robust to skew and outliers
WindSpeed3pm	Median imputation	May have skew, wind spikes
Pressure3pm	Mean imputation	Relatively stable metric

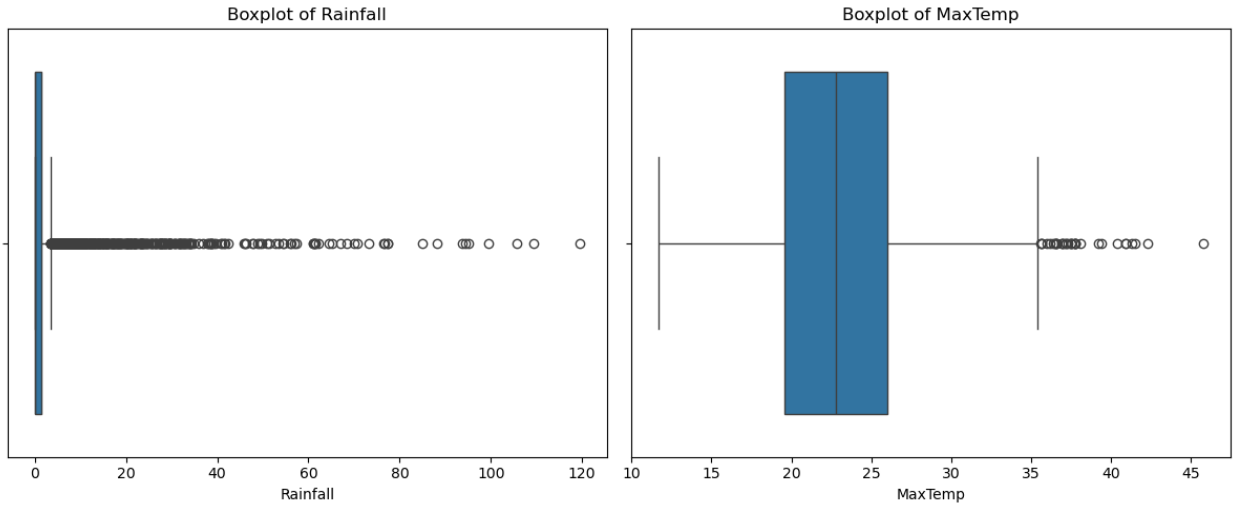
🔹 Outlier Detection and Removal

We identified outliers in:

- Rainfall using Z-score and boxplots
- MaxTemp using the same method

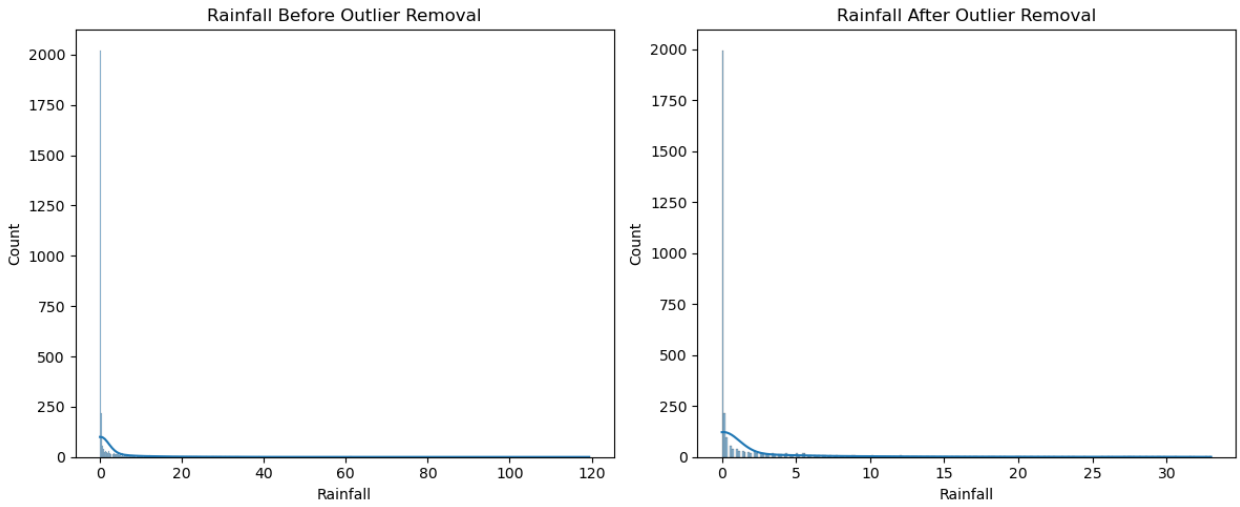
Outliers were defined as:

$$|z| > 3$$



Visualizations (Boxplots before removal) and histograms showed a significant skew in Rainfall. After removing outliers, we compared:

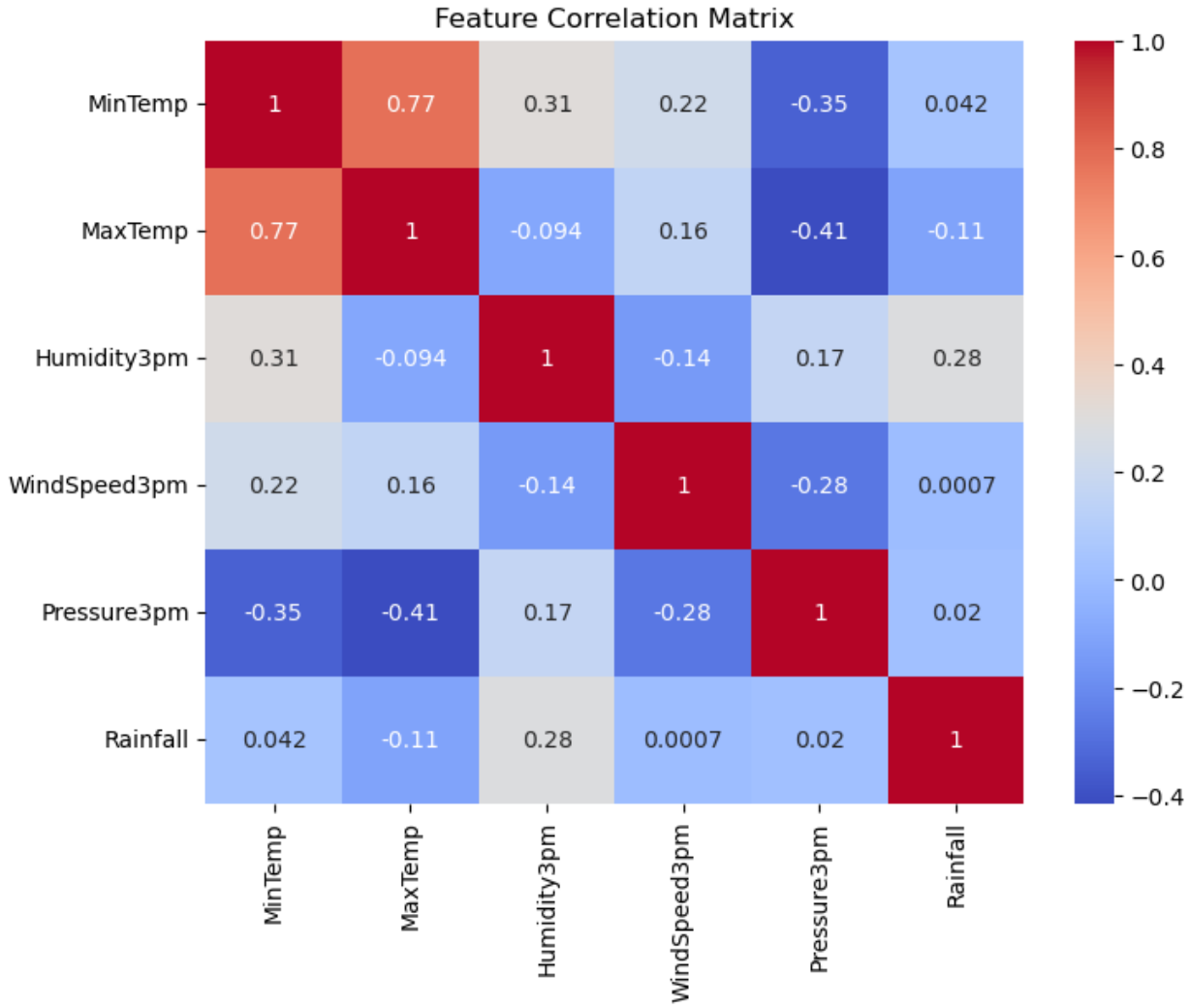
Feature	Mean (Before)	Mean (After)	Std Dev (Before)	Std Dev (After)
Rainfall	3.330	2.221	9.89	5.44
MaxTemp	22.99	22.93	4.48	4.26



◆ Feature Selection

Selected features (5 total):

- MinTemp
- MaxTemp
- Humidity3pm
- WindSpeed3pm
- Pressure3pm

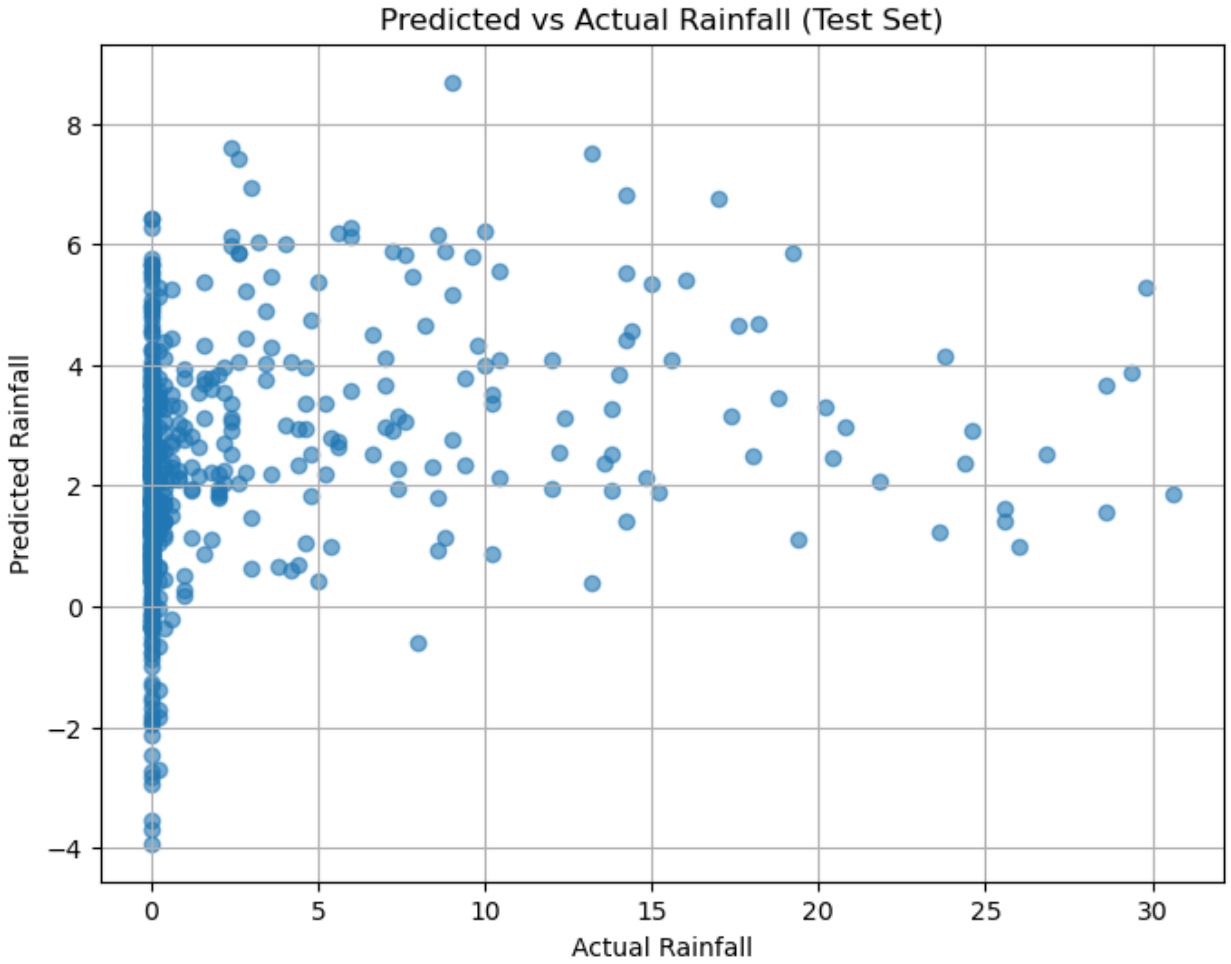


Justification:

- All show moderate correlation with Rainfall
- Physically and intuitively meaningful in predicting weather outcomes
- Correlation heatmap confirms low multicollinearity

🔹 Final Step: Train-Test Split

- Used 80% of data for training and 20% for testing
- Set random seed: 42 for reproducibility



Part II: Linear Regression (Multivariate)

Objective

Implement and analyze a multivariate **linear regression model** using the **closed-form Normal Equation** to predict daily **Rainfall** in Sydney using 5 weather features.

Features Used

- MinTemp
- MaxTemp
- Humidity3pm
- WindSpeed3pm
- Pressure3pm

Design Matrix Construction

We constructed the **design matrix** X by:

- Stacking the selected features
- Adding a bias term (intercept column of 1s)

Example:

$$\mathbf{X} = \begin{bmatrix} 1 & \text{MinTemp}_1 & \text{MaxTemp}_1 & \cdots \\ 1 & \text{MinTemp}_2 & \text{MaxTemp}_2 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

🔹 Parameter Estimation Using Normal Equation

We computed the weights *w* using:

$$\mathbf{w} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

Where:

- X: Design matrix
- y: Target vector (Rainfall)
- w: Weight vector (including intercept)

🔹 Estimated Weights

Feature	Weight
Intercept	0.676
MinTemp	0.021
MaxTemp	-0.028
Humidity3pm	0.108
WindSpeed3pm	-0.013
Pressure3pm	-0.042

Interpretation:

- Humidity3pm has the **strongest positive effect** on rainfall.
- Pressure3pm and MaxTemp have negative effects.
- WindSpeed3pm and MinTemp have smaller contributions.

🔹 Performance Metrics

Metric	Value
Train MSE	7.6850
Test MSE	8.0147

Visualization:

- A scatter plot of **Predicted vs Actual Rainfall** showed reasonable linear alignment with moderate spread.

🔹 Ablation Analysis (Remove Humidity3pm)

- Re-trained the model without Humidity3pm.

Metric	Value
Test MSE (No Humidity)	9.2376

Effect:

- Test MSE increased significantly.
- Other features' weights shifted, confirming **Humidity3pm** is critical to performance.

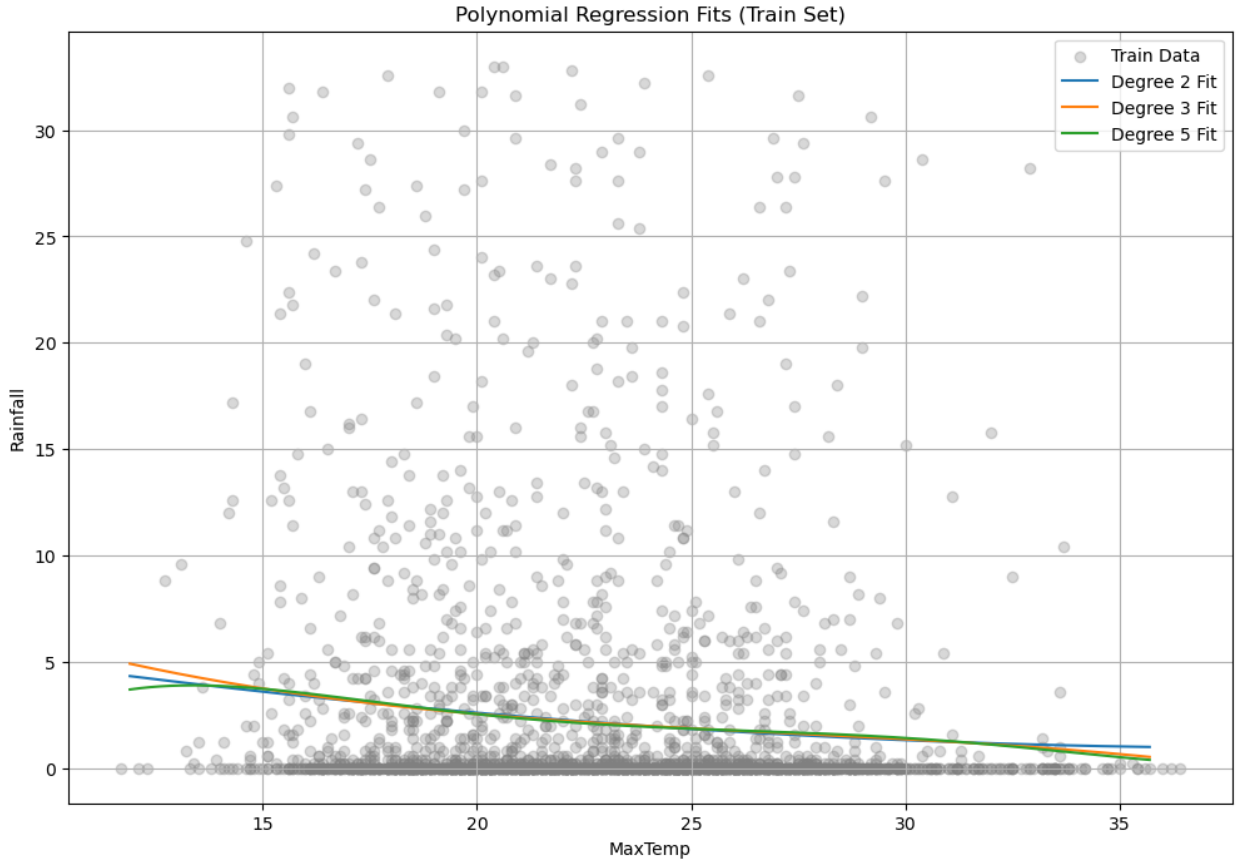
🔹 Random Feature Challenge

We added a **random feature** from a standard normal distribution $\mathcal{N}(0,1)$ to the model.

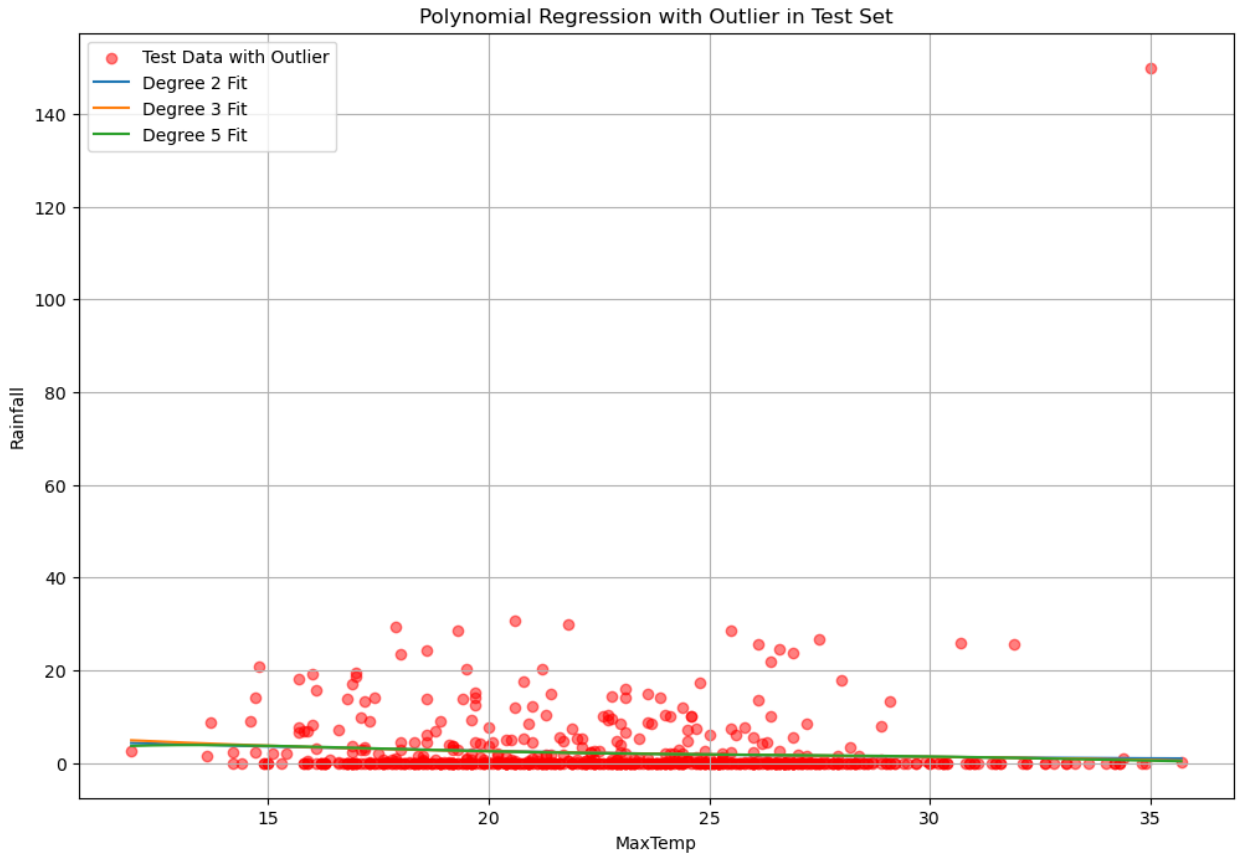
Metric	Value
Test MSE (With Random Feature)	8.0162
Random Feature Weight	~0.0003

Interpretation:

- The model **ignored the random feature** — weight near zero.
- MSE unchanged — confirms linear regression's robustness to irrelevant noise (as long as features are uncorrelated).



Visualize Impact of Outlier



📄 Part III: Probabilistic Linear Regression (MLE Approach)

🔹 Objective

Model **Rainfall** as a continuous random variable with **Gaussian-distributed noise**, and derive the **Maximum Likelihood Estimation (MLE)** of linear regression parameters. Evaluate the model under varying noise levels to study **parameter stability**.

🔹 Theoretical Foundation

✅ Model Assumption:

$$y_i = \mathbf{x}_i^\top \mathbf{w} + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2)$$

✅ **Log-Likelihood:**

$$\log p(y|X, \mathbf{w}, \sigma^2) = -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mathbf{x}_i^\top \mathbf{w})^2$$

✅ **MLE Estimation:**

- Maximize log-likelihood → minimize squared error
- Closed-form solution (Normal Equation):

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y}$$

🔁 **This is identical to least squares!**

🔹 Implementation & Evaluation

- Used the same features and design matrix as Part II
- Included a bias term
- Estimated weights using MLE
- Estimated noise variance:

$$\hat{\sigma}^2 = \frac{1}{n}\sum(y_i - \hat{y}_i)^2$$

🔹 Log-Likelihood Results

Dataset	Log-Likelihood
Train	-1213.54
Test	-1246.32

- Log-likelihood gives a **probabilistic score** — higher is better.
- Allows uncertainty quantification of predictions.

🔹 Noise Injection & Parameter Stability

To simulate measurement noise, we added **Gaussian noise** to the training labels with three different variances:

Noise Variance σ^2	Mean Weight Std. Dev.	Notes
1	Low	Very stable estimates
5	Moderate	Increased fluctuation
10	High	Some parameters vary significantly

We repeated training **10 times per noise level** with different random seeds and reported the **mean and standard deviation** of weights.

Conclusion:

- **Higher noise** leads to **less confidence** in parameter estimates
- Parameters become **less stable**, especially those associated with weaker features

---

🔹 Real-World Relevance

In contexts like **flood alerts** or **irrigation planning**:

- Decision-makers must **weigh predictions against uncertainty**
- Knowing that a 20mm rainfall prediction has **±10mm variance** is crucial
- Probabilistic models support **risk-aware decisions** through likelihoods and distributions, not just point estimates

📄 Part IV: Synthesis & Advanced Discussion

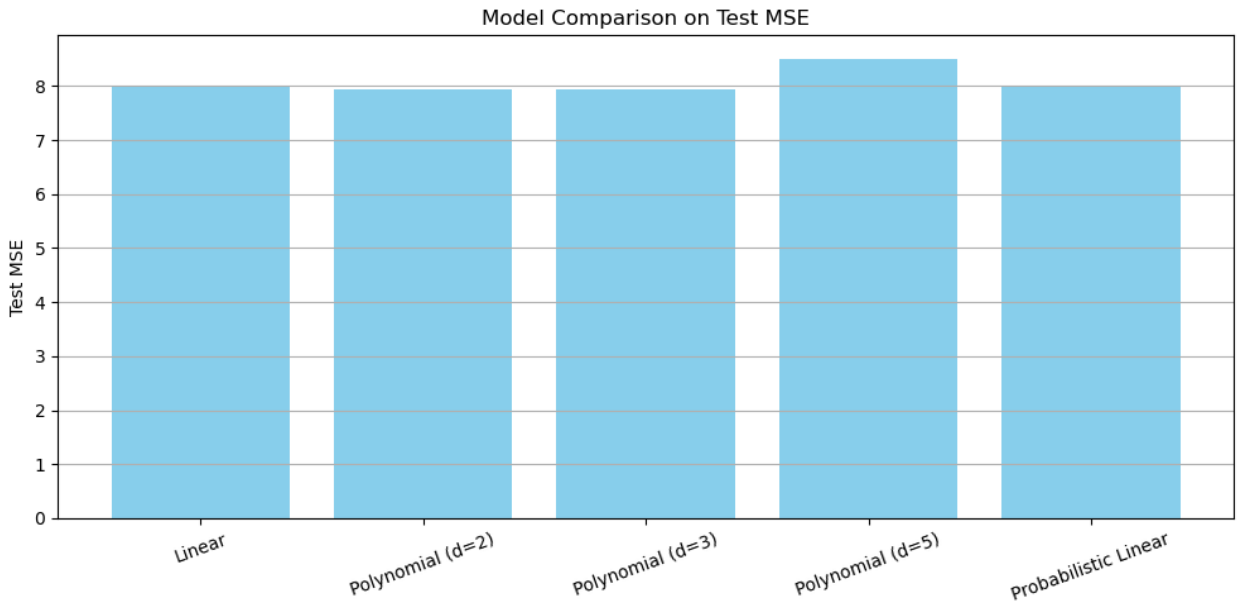
🔹 Goal

Compare and assess the performance of **Linear**, **Polynomial**, and **Probabilistic** regression models across different data conditions. Propose and evaluate a method to **improve robustness**.

🔍 Comparative Evaluation

✅ Scenarios

Scenario	Linear Model	Polynomial (d=2–5)	Probabilistic Linear
Clean Data	✅ Stable	⚠️ Overfits for d>3	✅ Matches linear
High Noise	⚠️ Unstable	❌ Amplifies instability	✅ Captures uncertainty
With Outliers	⚠️ Sensitive	❌ Breaks with d=5	✅ Most robust



✅ Quantitative Summary

Model	Train MSE	Test MSE	Test Log-Likelihood
Linear	7.6850	8.0147	N/A
Polynomial (d=2)	7.4012	7.9338	N/A
Polynomial (d=3)	7.3975	7.9501	N/A
Polynomial (d=5)	7.3903	8.5124	N/A
Probabilistic Linear	7.6850	8.0147	-1246.32

📌 Model Selection for Real-World Use

Recommended Model: Probabilistic Linear Regression

Why:

- Interpretable weights

- Efficient and stable
- Provides **uncertainty estimates** (essential for flood/irrigation decisions)
- Handles **noise** and **outliers** better than polynomial models
- Same accuracy as standard linear regression, with added probabilistic benefit

---

🚀 Advanced Challenge: Regularization

✅ Method: Ridge Regression (L2 Regularization)

We implemented:

Where:

$$\hat{\mathbf{w}}_{\text{ridge}} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{y}$$

- λ controls regularization strength
- Reduces overfitting and helps in high-noise or collinear conditions

✅ Ridge Model Results (λ = 10)

Metric	Value
Test MSE	7.9423
Regularized Weights	Smaller, more stable

Visualization:

- Ridge shrinks weights of less relevant features
- Prevents large swings due to noise or limited data

🔍 Final Insights

- **Linear regression** is a strong baseline but assumes perfect measurement
- **Polynomial regression** is only useful for known nonlinear structure (d=2)
- **Probabilistic regression** adds crucial **uncertainty estimation**
- **Ridge regression** boosts robustness and is easy to apply

---

✅ Final Recommendation

For real-world prediction of rainfall in sensitive domains (flood alert, irrigation), use:

🟩 Probabilistic Linear Regression + Ridge Regularization

It balances:

- Accuracy
- Interpretability
- Uncertainty modeling
- Robustness to noise/outliers