Understanding Underfitting, Overfitting, and the Bias–Variance Trade-Off Through Polynomial Regression

Student ID: 24095044

Student Name: Surya Teja Nallapu

1. Introduction

One of the central challenges in machine learning is ensuring that models generalize well—that is, perform accurately not only on training data but also on unseen examples. When a model is **too simple**, it fails to capture important structure in the data, resulting in consistently poor predictions. This problem is known as **underfitting**. Conversely, when a model becomes **too flexible**, it may begin to learn random noise in the training data rather than the true underlying pattern, leading to excellent training performance but weak test performance—this is **overfitting**. As described by Mucci (2024), achieving effective generalization requires finding the right balance between these two extremes, a tension often explained through the **bias–variance trade-off**.

Polynomial regression provides a clean, intuitive way to observe these behaviours because model complexity can be controlled directly:

- increasing the polynomial degree makes the model more flexible, and
- decreasing the degree restricts the model, making it more rigid.

By experimenting with polynomial models of different degrees, we can build a deeper understanding of:

- what underfitting and overfitting look like in practice,
- how training and validation errors change as complexity varies, and
- how the bias–variance trade-off explains these behaviours.

This tutorial aims not only to explain the concepts but to help you develop an intuition for how model flexibility influences generalization, a skill crucial for real-world machine-learning practice.

2. Dataset and Experimental Setup

To explore underfitting, overfitting, and the bias–variance trade-off in a controlled and interpretable way, we construct a synthetic dataset based on the nonlinear function:

$$\boldsymbol{y = \sin(x) + \epsilon},$$

where $\boldsymbol{\epsilon}$ represents Gaussian noise.

Using synthetic data is advantageous for this type of analysis because the true underlying function is known. This makes it easy to distinguish between signal (the true sine curve) and noise (random deviations), allowing us to clearly observe how different models behave.

We generate 50 evenly spaced points in the interval $[0, 6]$, apply the sine function, and add Gaussian noise to simulate natural measurement variability found in real-world systems. The dataset is then split into 70% training data and 30% test data to enable a fair evaluation of generalization.

To study the effect of model complexity, we train polynomial regression models with degrees **1, 3, 5, and 15**. These models are constructed using *scikit-learn's PolynomialFeatures* transformer, which generates polynomial combinations of the input feature, and the *LinearRegression estimator* (Scikit-learn, n.d.). A fixed random seed ensures reproducibility so that results remain consistent across different runs and allow for meaningful comparisons between polynomial degrees.

3. Underfitting and Overfitting Through Polynomial Models

Polynomial regression provides a clear and intuitive lens for examining how model complexity influences generalization. By gradually increasing the polynomial degree, we can observe how the fitted curves transition from underfitting to well-balanced modelling and, eventually, overfitting. The models with degrees 1, 3, 5, and 15 reveal these behaviours vividly.

Degree 1 — Underfitting (High Bias):

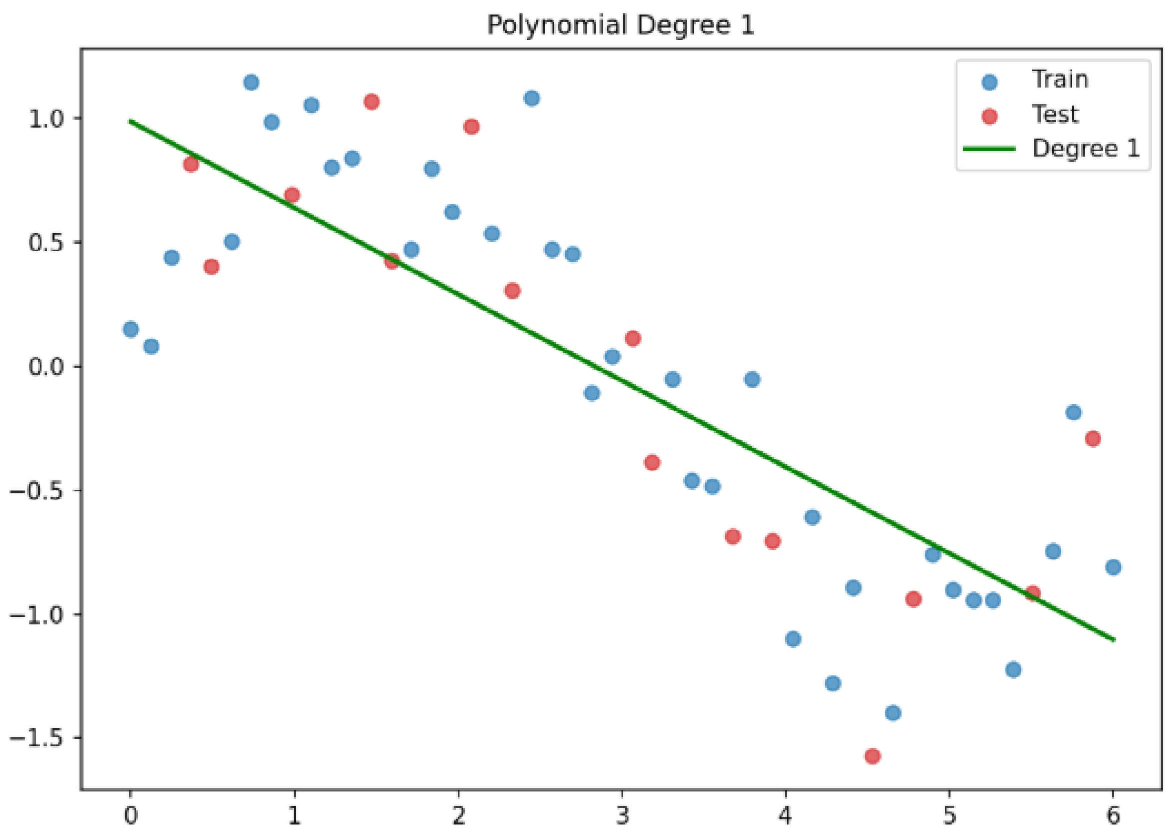

__Figure 1__. Degree-1 polynomial regression illustrating underfitting due to a rigid linear form.

The degree-1 model produces a simple straight-line fit that fails to capture the nonlinear structure of the sine function. Because the true relationship oscillates, the linear model can only approximate a rough average trend. As a result:

- It misses all curvature—both the peaks and troughs.
- Training points and test points lie far from the fitted line.
- Both training and test errors are large and similar.

This is a classic example of underfitting:

the model is too simple, leading to high bias and persistent predictive error. Even if we provide more data, the linear model cannot learn the true pattern because the model class lacks the flexibility required.

__Degree 3 — Moderate Complexity with Reduced Bias:__

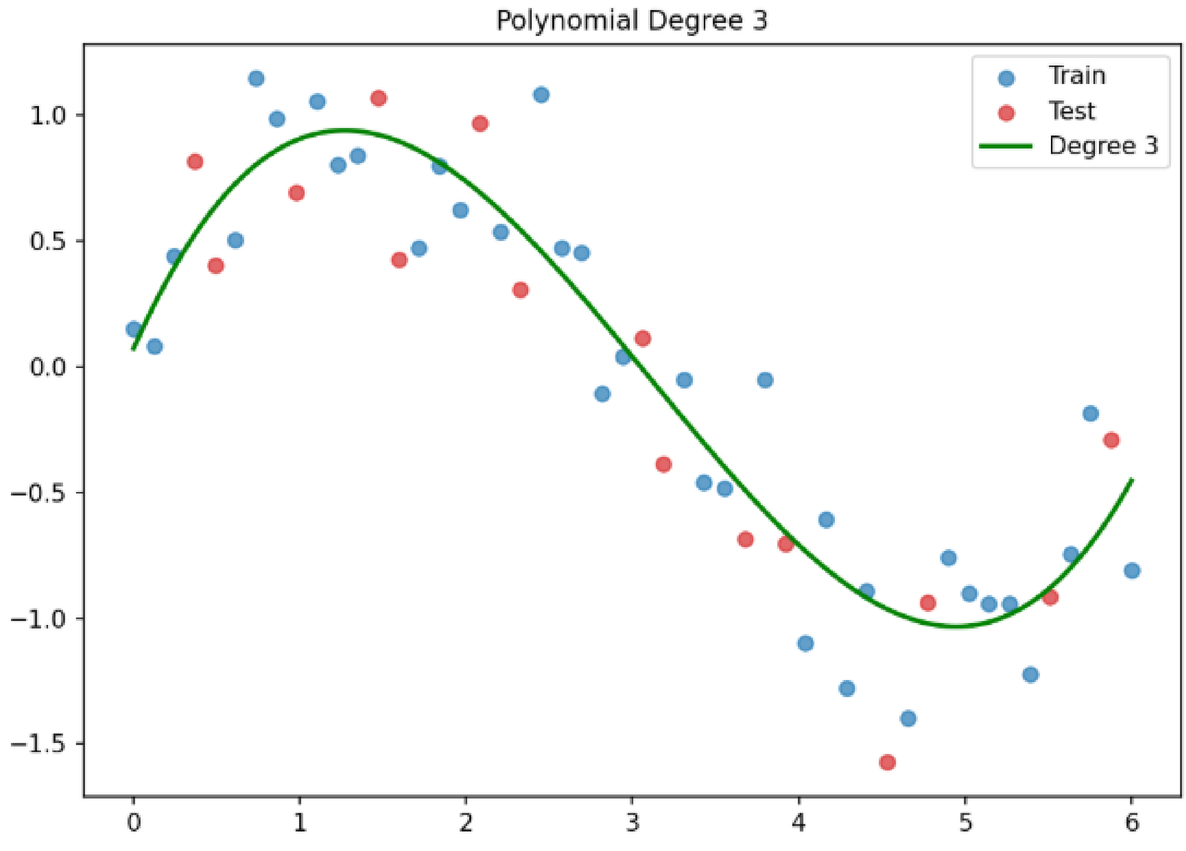

***Figure 2.** Degree-3 polynomial regression capturing the main nonlinear trend.*

The degree-3 polynomial offers a substantial improvement over the linear model. The fitted curve is flexible enough to follow the general sinusoidal shape without reacting strongly to noise. Key observations:

- It tracks the main upward and downward movements.

- The curve is smooth and stable.
- Bias is significantly lower compared to the degree-1 model.
- Variance remains relatively low, yielding strong generalization.

This degree demonstrates a healthy balance between model flexibility and stability, resulting in improved performance on both training and test sets.

Degree 5 — Balanced Flexibility with Strong Generalization:

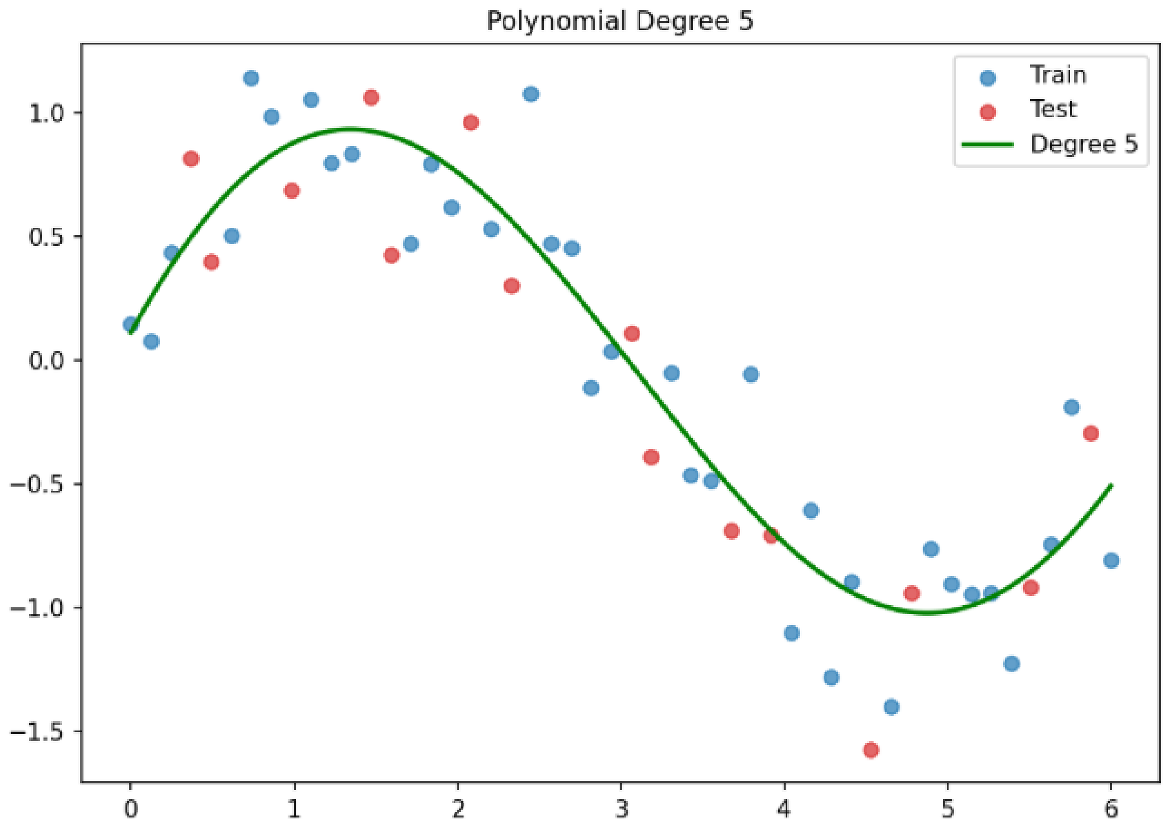

Figure 3. *Degree-5 polynomial regression achieving a strong balance between flexibility and stability.*

The degree-5 polynomial captures the sinusoidal structure even more effectively than the degree-3 model. It adapts well to regions of sharp curvature, tracking both the peaks and troughs with notable accuracy. Importantly, it does so without introducing unnecessary oscillations or becoming overly sensitive to individual data points.

This suggests that the model has reached a point where it is flexible enough to represent the underlying function while still maintaining stability. Training and test errors remain low and closely aligned, indicating a favourable balance between bias and variance.

For many real-world datasets, models of this level of complexity often deliver the best generalization performance, especially when the underlying pattern is moderately nonlinear.

Degree 15 — Overfitting Due to High Variance:

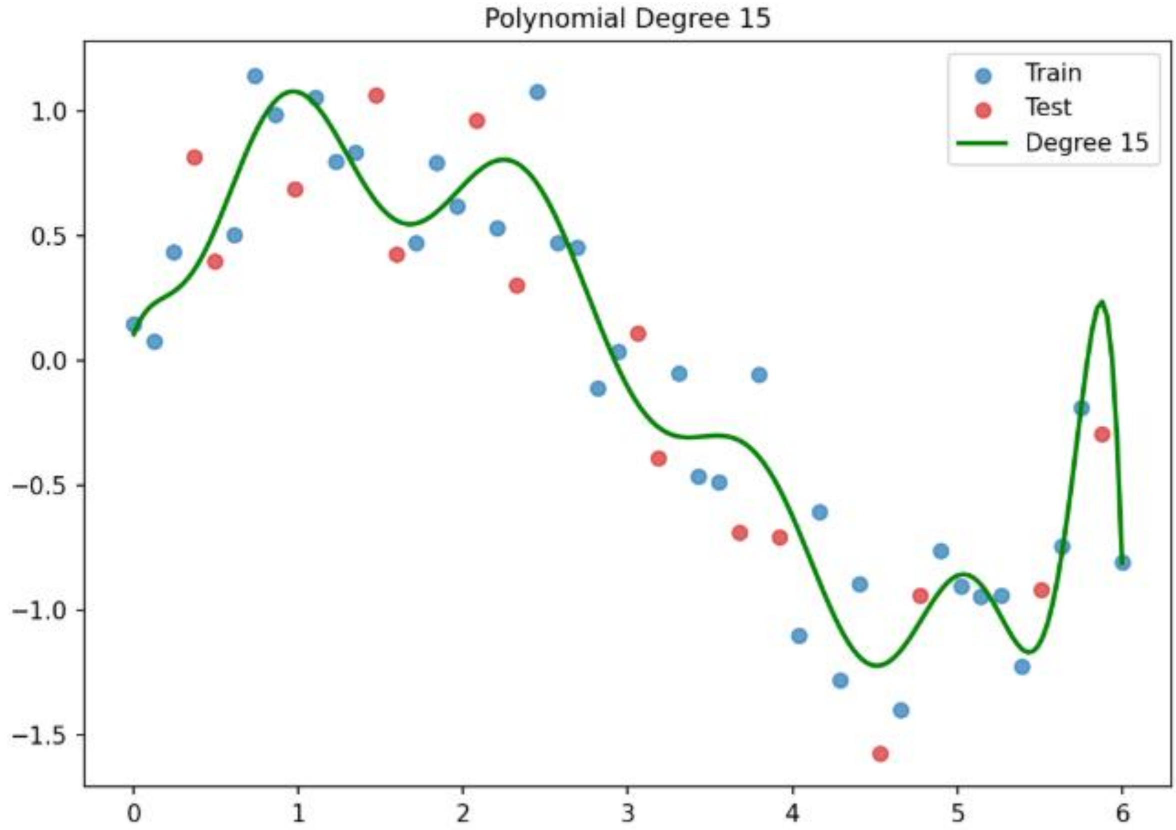

Figure 4. *Degree-15 polynomial regression illustrating pronounced overfitting.*

The degree-15 model demonstrates classic overfitting behaviour. While it successfully follows the general sinusoidal trend, it goes much further—attempting to match the random noise present in the training set. This leads to sharp oscillations, unstable curvature, abrupt changes in direction, and extreme sensitivity to individual data points, which are all characteristic signs of overfitting described in machine-learning literature (Choudhary, 2022).

Toward the right side of the plot, the model bends aggressively to accommodate a few isolated points—an unmistakable sign of high variance. Although this results in very low training error, the model fails to generalize, producing significantly larger errors on the test set.

The contrast between the smooth sine curve and the highly contorted fitted polynomial captures the essence of overfitting: too much flexibility can be just as harmful as too little.

Summary of Observations (Figures 1–4)

- Low-degree models (e.g., degree 1) underfit due to high bias and insufficient flexibility.
- High-degree models (e.g., degree 15) overfit by capturing noise, resulting in high variance and poor generalization.
- Intermediate models (degrees 3–5) strike the best balance, capturing the true pattern without reacting to random fluctuations.

Key Takeaway: *Optimal generalization occurs at a middle level of complexity, where bias and variance are balanced.*

4. Training vs Validation Error Curve

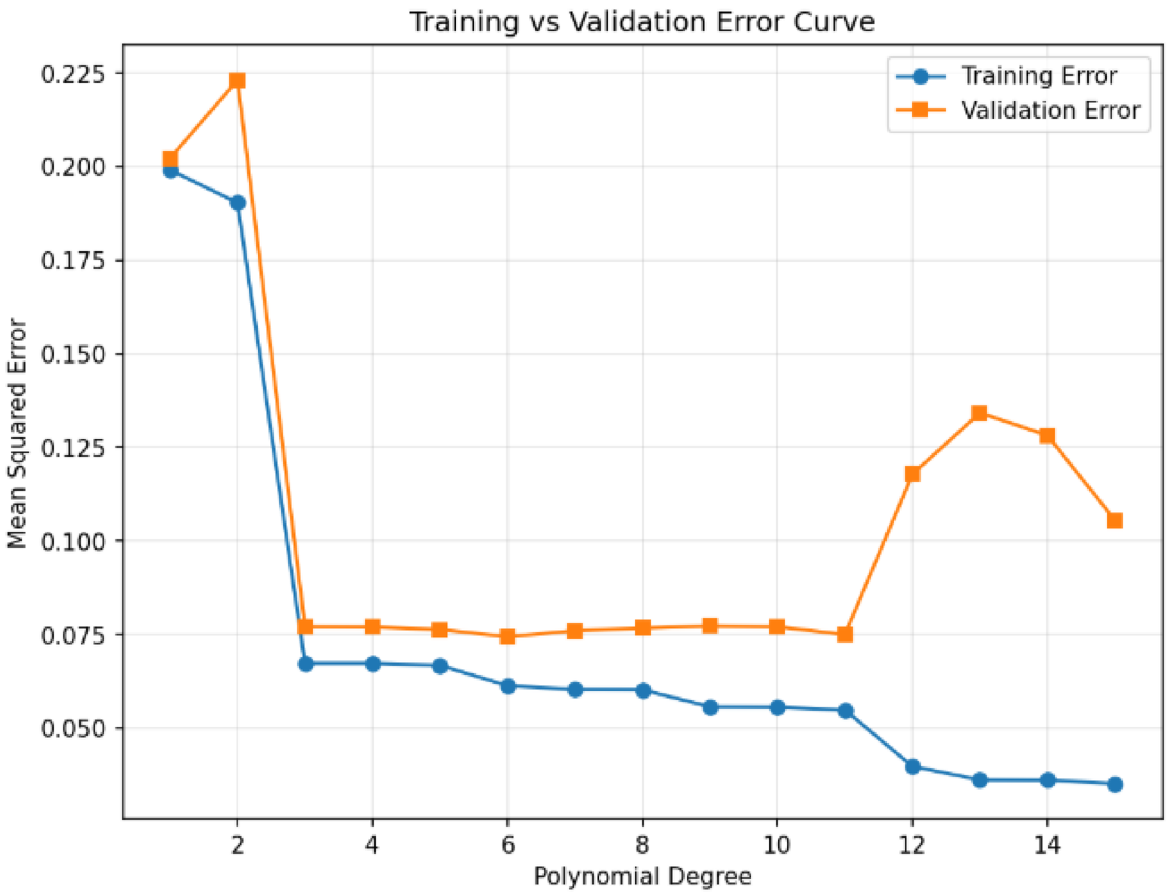

Figure 5. *Training and validation mean squared error (MSE) across polynomial degrees 1–15, illustrating the characteristic U-shaped validation curve.*

Figure 5 shows how model complexity affects generalization by plotting training and validation MSE for polynomial degrees 1–15. Training error decreases steadily with higher degrees because more flexible models fit the training data more closely—even fitting noise.

The validation error forms the classic U-shaped curve:

- Degrees 1–2: High training and validation error → underfitting.
- Degrees 3–6: Lowest validation error → optimal generalization.
- Degrees 10+: Validation error rises → overfitting as the model memorizes noise.

A widening gap between training and validation error at higher degrees reveals the bias–variance trade-off: bias decreases with complexity, but variance rises sharply.

Key Insight: The U-shaped validation curve is a powerful diagnostic—it shows the balance between underfitting and overfitting and helps select an appropriate model.

5. The Bias–Variance Trade-Off

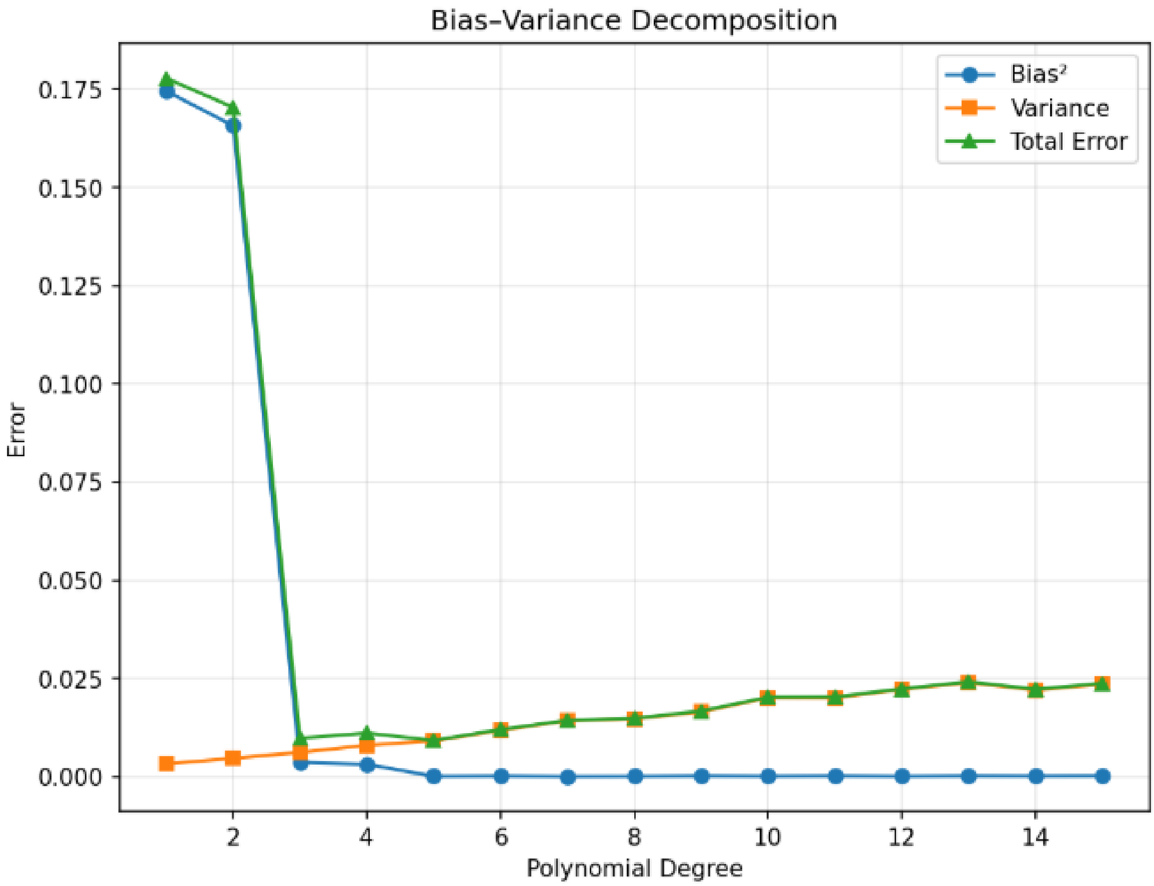

Figure 6*. Bias–variance decomposition illustrating how bias² decreases and variance increases with polynomial degree, shaping the total error curve.*

Figure 6 illustrates how bias and variance jointly shape prediction error as polynomial degree increases. Bias is highest for low-degree models (e.g., degrees 1–2), which are too rigid to capture the nonlinear sine function. This results in large bias² values that drop sharply once the model becomes sufficiently flexible (around degree 4 and above).

Variance, however, increases with complexity. Higher-degree polynomials fit the training data closely and become sensitive to small variations in the sample, causing variance to rise—particularly beyond degree 10, where it becomes the dominant source of error.

The total error—the sum of bias² and variance—follows the familiar U-shape also seen in the validation error curve. Error decreases initially as bias falls, but rises again once variance grows, highlighting the core principle of the bias–variance trade-off: Optimal generalization occurs not at minimal bias or minimal variance, but where their combined error is lowest.

In this experiment, polynomial degrees 3–6 achieve this balance, which explains their strong performance on unseen data.

Tip: Use bias–variance patterns to guide decisions about model complexity, regularization strength, or whether more training data is needed.

Insight: Although demonstrated with polynomials, the same trade-off governs decision trees, neural networks, and most modern ML models.

6. Learning Curves for a High-Complexity Model

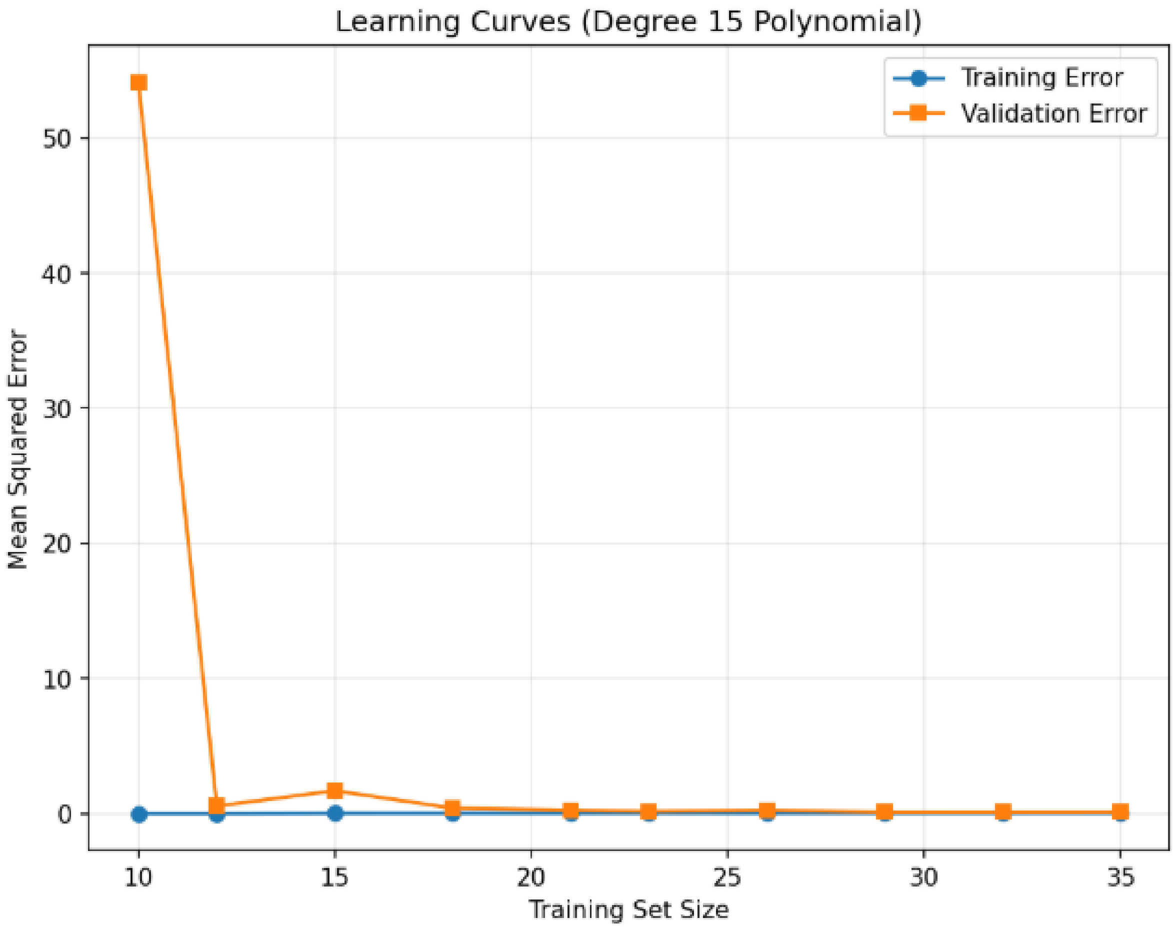

Figure 7. *Learning curves for the degree-15 polynomial model, illustrating the impact of training-set size on generalization.*

Figure 7 highlights how the performance of a high-variance, highly flexible model changes as the amount of training data increases. In this experiment, we examine the behaviour of the degree-15 polynomial, which we have already identified as prone to overfitting.

When trained on very small subsets of the data, the model achieves near-zero training error because it can perfectly interpolate a small number of points. However, this perfect fit is misleading: the validation error spikes dramatically. With so little data, the degree-15 polynomial models' noise, irregularities, and idiosyncrasies rather than the true underlying structure—resulting in severe overfitting.

As the training-set size increases, the behaviour changes significantly:

- Validation error decreases and eventually stabilizes.
- The model generalizes better because additional data “anchors” it, reducing the influence of noise.
- Training error increases slightly, but remains relatively low—expected for a model with substantial flexibility.

This pattern illustrates a key practical lesson: high-variance models benefit greatly from larger datasets. When too little data is available, complex models behave erratically; with more data, they become more stable and reliable.

This insight generalizes well beyond polynomial regression. Many modern learning systems—most notably deep neural networks—exhibit similar behaviour, requiring large datasets to achieve strong generalization.

Tip: If a model overfits persistently, even with regularization, consider increasing the dataset size. Sometimes, more data is the most powerful form of regularization.

Insight: The shape of learning curves is often more informative than raw accuracy scores. They reveal whether collecting more data will help, whether the model is too complex, or whether underfitting is unavoidable with the chosen architecture.

7. Practical Lessons and Conclusions

The experiments in this tutorial illustrate several foundational principles of machine learning:

- Underfitting occurs when the model is too simple to represent the underlying pattern; this appears as high training and validation error.
- Overfitting occurs when the model is overly flexible and begins to memorize noise rather than learn meaningful structure.
- The training vs. validation error curves reveals the optimal complexity level, where generalization error is minimized.
- The bias–variance decomposition explains why this optimum exists: as bias decreases with complexity, variance increases, and the best models strike a balance between the two.
- Learning curves show how the amount of training data affects generalization, particularly for complex, high-variance models.

Although polynomial regression is used here as a pedagogical tool due to its clear interpretability, the lessons apply widely across machine-learning practice—including neural networks, gradient-boosted trees, kernel methods, and ensemble models.

NOTE: Understanding underfitting, overfitting, and the bias–variance trade-off is key to informed decisions in model selection, hyperparameter tuning, data strategy, and regularization. These principles form the core of effective machine-learning practice.

All code, visualizations, and the Jupyter notebook used in this tutorial can be found in the accompanying GitHub repository: https://github.com/Surya353535/underfitting-overfitting-tutorial

References

Mucci, Ti. (2024). *Overfitting vs. Underfitting*. [online] Ibm.com. Available at: https://www.ibm.com/think/topics/overfitting-vs-underfitting.

CHOUDHARY, A.S. (2022). *Regularization in Machine Learning*. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2022/08/regularization-in-machine-learning/.

scikit-learn.org. (n.d.). *sklearn.preprocessing.PolynomialFeatures — scikit-learn 0.23.2 documentation*. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.PolynomialFeatures.html.