Bias-Variance Tradeoff in Statistical Learning

March 27, 2025

Outline

Fundamental Concepts: Bias and Variance

Definition (Bias)

The bias of an estimator $\hat{\theta}$ for a parameter θ is:

$$\text{Bias}(\hat{\theta}) = \mathbb{E}[\hat{\theta}] - \theta$$

Definition (Variance)

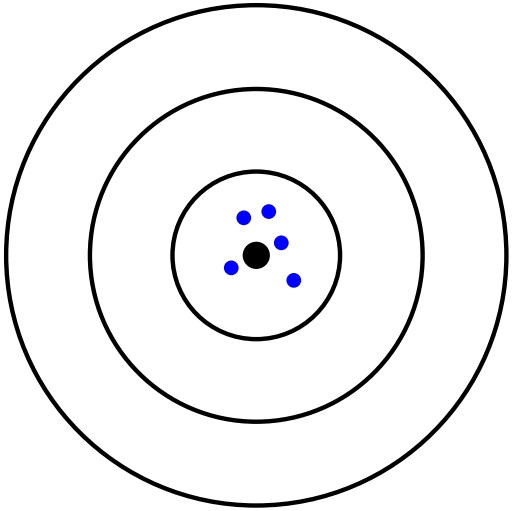
The variance of an estimator $\hat{\theta}$ is:

$$\text{Var}(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \mathbb{E}[\hat{\theta}])^2]$$

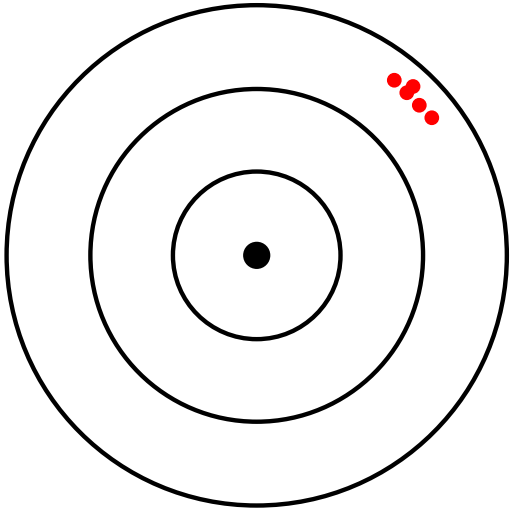
Interpretation

- **Bias:** Systematic error; how far predictions are from true values on average
- **Variance:** Statistical dispersion; how much predictions fluctuate

Target Shooting Analogy

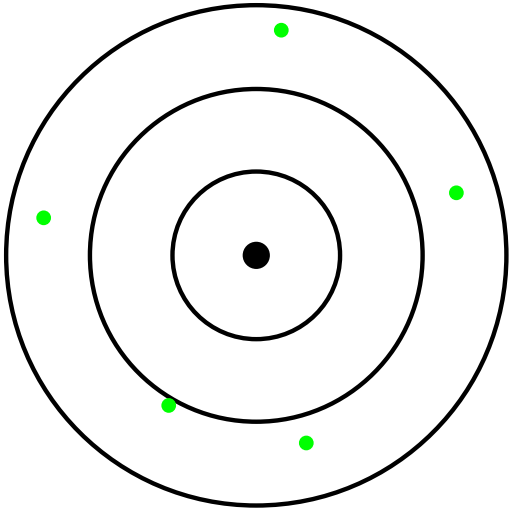


Low Bias, Low Variance
(Ideal)

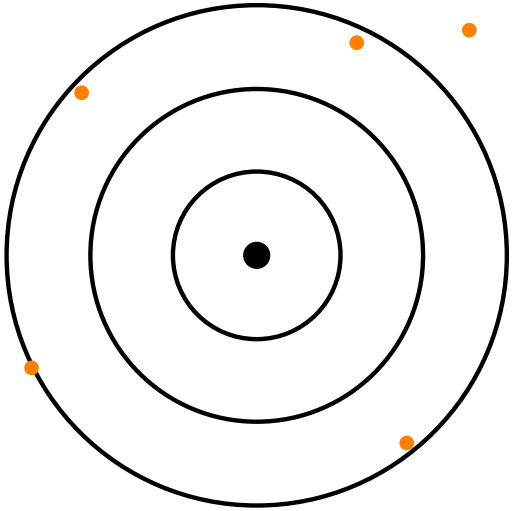


High Bias, Low Variance
(Underfitting)

Target Shooting Analogy (continued)



Low Bias, High Variance
(Overfitting)



High Bias, High Variance
(Worst Case)

The Bias-Variance Decomposition

Theorem

For a given point x, the expected mean squared error can be decomposed as:

$$\mathbb{E}[(y - \hat{f}(x))^2] = \underbrace{(f(x) - \mathbb{E}[\hat{f}(x)])^2}_{\text{Bias}^2} + \underbrace{\mathbb{E}[(\hat{f}(x) - \mathbb{E}[\hat{f}(x)])^2]}_{\text{Variance}} + \underbrace{\sigma_\varepsilon^2}_{\text{Irreducible Error}}$$

- $f(x)$ is the true function
- $\hat{f}(x)$ is our model's prediction
- σ_ε^2 is the noise variance (can't be reduced)

Proof Sketch of the Decomposition

Let $y = f(x) + \varepsilon$ where $\mathbb{E}[\varepsilon] = 0$ and $\text{Var}(\varepsilon) = \sigma_\varepsilon^2$.

1. Start with the expected squared error:

$$\mathbb{E}[(y - \hat{f}(x))^2] = \mathbb{E}[(f(x) + \varepsilon - \hat{f}(x))^2]$$

2. Expand and use $\mathbb{E}[\varepsilon] = 0$:

$$= \mathbb{E}[(f(x) - \hat{f}(x))^2] + \sigma_\varepsilon^2$$

3. Add and subtract $\mathbb{E}[\hat{f}(x)]$:

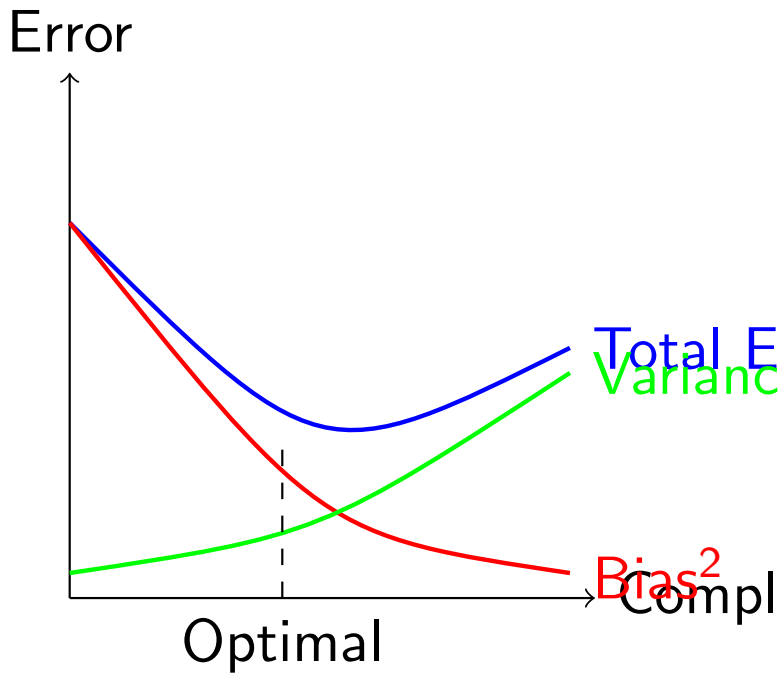
$$= \mathbb{E}[(f(x) - \mathbb{E}[\hat{f}(x)] + \mathbb{E}[\hat{f}(x)] - \hat{f}(x))^2] + \sigma_\varepsilon^2$$

4. Expand and use $\mathbb{E}[\mathbb{E}[\hat{f}(x)] - \hat{f}(x)] = 0$:

$$= (f(x) - \mathbb{E}[\hat{f}(x)])^2 + \mathbb{E}[(\hat{f}(x) - \mathbb{E}[\hat{f}(x)])^2] + \sigma_\varepsilon^2$$

The Tradeoff

- Simple models: **High bias**, **low variance**
- Complex models: **Low bias**, **high variance**
- The goal: Find optimal complexity that minimizes total error



Sample Mean Estimator

Properties

For i.i.d. random variables $X_1, X_2, \ldots, X_n$ with mean μ and variance σ^2, the sample mean $\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ has:

$$\text{Bias}(\bar{X}_n) = \mathbb{E}[\bar{X}_n] - \mu = 0$$

$$\text{Var}(\bar{X}_n) = \frac{\sigma^2}{n}$$

Key Insights

- The sample mean is unbiased
- Its variance decreases with sample size ($1/n$ rate)
- Larger samples provide more precise estimates

Variance Estimators

Two common estimators for the population variance:

Biased (Maximum Likelihood) Estimator

$$\hat{\sigma}^2_{\text{biased}} = \frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

Properties: $\mathbb{E}[\hat{\sigma}^2_{\text{biased}}] = \frac{n-1}{n}\sigma^2$ (underestimates true variance)

Unbiased Estimator

$$\hat{\sigma}^2_{\text{unbiased}} = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

Properties: $\mathbb{E}[\hat{\sigma}^2_{\text{unbiased}}] = \sigma^2$ (no bias)

Despite being biased, the MLE might have lower MSE for small samples (bias-variance tradeoff)

Introduction to Shrinkage

Concept

Shrinkage estimators deliberately introduce bias to reduce variance, potentially achieving lower overall error

- "Shrink" estimates toward a target value
- Trade off increased bias for reduced variance
- Can outperform unbiased estimators in terms of MSE

General Form

$$\hat{\theta}_\alpha = (1 - \alpha)\hat{\theta}_{\text{unbiased}} + \alpha\,\theta_{\text{target}}$$

where $\alpha \in [0, 1]$ is the shrinkage parameter

Properties of Shrinkage Estimators

For the shrinkage estimator $\hat{\theta}_\alpha = (1-\alpha)\hat{\theta}_{\text{unbiased}} + \alpha\,\theta_{\text{target}}$:

Bias and Variance

$$\text{Bias}(\hat{\theta}_\alpha) = \alpha(\theta_{\text{target}} - \theta)$$
$$\text{Var}(\hat{\theta}_\alpha) = (1-\alpha)^2\text{Var}(\hat{\theta}_{\text{unbiased}})$$

Mean Squared Error

$$\begin{aligned}\text{MSE}(\hat{\theta}_\alpha) &= \text{Bias}(\hat{\theta}_\alpha)^2 + \text{Var}(\hat{\theta}_\alpha) \\ &= \alpha^2(\theta_{\text{target}} - \theta)^2 + (1-\alpha)^2\text{Var}(\hat{\theta}_{\text{unbiased}})\end{aligned}$$

Optimal Shrinkage Parameter

Derivation

To find the optimal α^*, differentiate MSE with respect to α and set to zero:

$$\frac{d}{d\alpha}\text{MSE}(\hat{\theta}_\alpha) = 2\alpha(\theta_{\text{target}} - \theta)^2 - 2(1-\alpha)\text{Var}(\hat{\theta}_{\text{unbiased}}) = 0$$

Solving for α:

$$\alpha^* = \frac{\text{Var}(\hat{\theta}_{\text{unbiased}})}{(\theta_{\text{target}} - \theta)^2 + \text{Var}(\hat{\theta}_{\text{unbiased}})}$$

Insights

- Stronger shrinkage when variance is high
- Stronger shrinkage when target is close to true value
- Optimal α^* depends on unknown parameters (need estimation in

Example: James-Stein Estimator

Setting

Estimating means $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_p)^T$ of a multivariate normal with $p \geq 3$ components

James-Stein Estimator

$$\hat{\boldsymbol{\mu}}^{JS} = \left(1 - \frac{(p-2)\sigma^2}{\|\bar{\mathbf{x}}\|^2}\right) \bar{\mathbf{x}}$$

"Paradox"

The James-Stein estimator has lower expected squared error than the MLE (sample mean) for *any* true value of $\boldsymbol{\mu}$, despite introducing bias

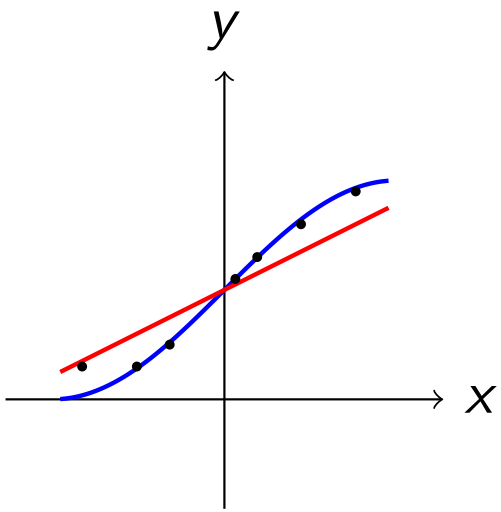
This example shows that unbiased estimators are not always optimal in terms of MSE

Underfitting vs. Overfitting

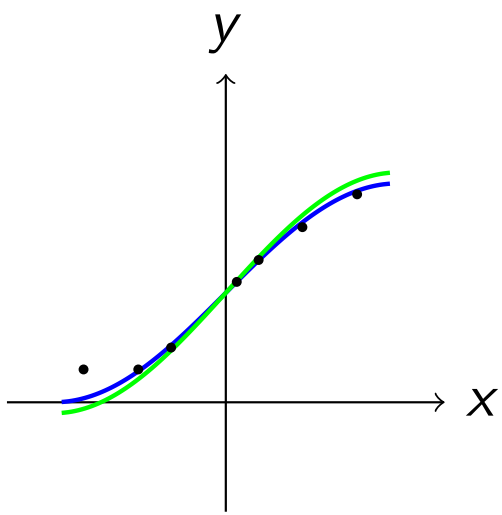
- **Underfitting:** Model is too simple (high bias, low variance)
  - Misses important patterns in the data
  - Performs poorly on both training and test data
- **Overfitting:** Model is too complex (low bias, high variance)
  - Captures noise in the training data
  - Performs well on training data but poorly on test data
- **Good fit:** Appropriate complexity (balanced bias and variance)
  - Captures main patterns without fitting noise
  - Generalizes well to new data

Polynomial Regression Example

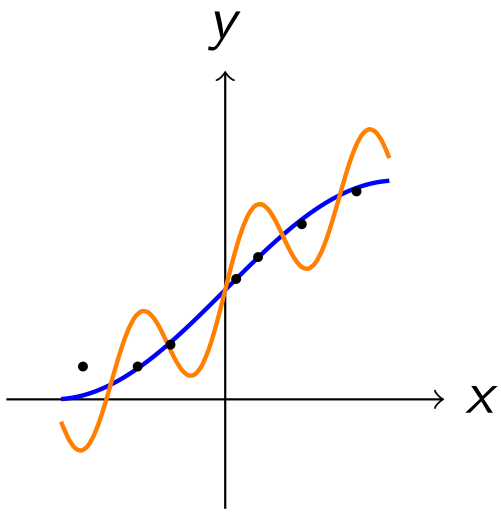
Consider modeling data with polynomials of different degrees:



Underfitting



Good Fit



Overfitting

Regularization Techniques

Ridge Regression (L2 regularization)

$$\min_{\beta} \|y - X\beta\|^2 + \lambda\|\beta\|^2$$

- Shrinks all coefficients toward zero
- Solution: $\hat{\beta}^{\text{ridge}} = (X^T X + \lambda I)^{-1} X^T y$

Lasso Regression (L1 regularization)

$$\min_{\beta} \|y - X\beta\|^2 + \lambda\|\beta\|_1$$

- Can set some coefficients exactly to zero (feature selection)
- No closed-form solution

Elastic Net

Sample Size Considerations

- **Small datasets:**
  - Higher risk of overfitting
  - Use simpler models (higher bias, lower variance)
  - Apply stronger regularization
  - Consider data augmentation
- **Large datasets:**
  - Can use more complex models (lower bias, manageable variance)
  - Less need for regularization
  - More capacity to learn intricate patterns
- **Rule of thumb:** Model complexity should increase with sample size

Ensemble Methods

Bagging (Bootstrap Aggregating)

- Train multiple high-variance, low-bias models on bootstrap samples
- Average predictions to reduce variance
- Example: Random Forests

Boosting

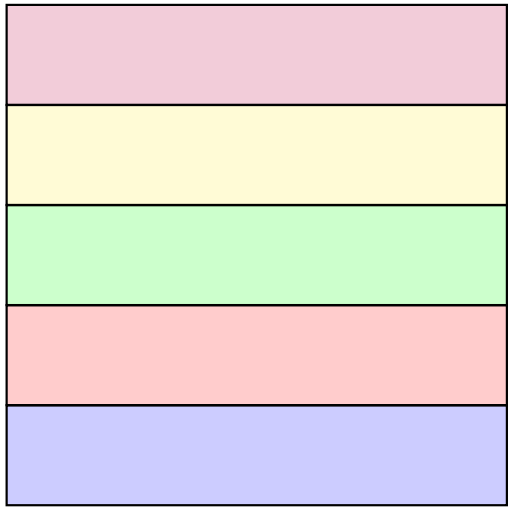
- Train sequence of weak models (high bias, low variance)
- Each model focuses on errors of previous models
- Reduces bias while controlling variance
- Examples: AdaBoost, Gradient Boosting

Stacking

- Combine predictions from different types of models
- Meta-learner optimizes the combination

Cross-Validation for Model Selection

1. Split data into k folds
2. For each model complexity:
   - Train on $k-1$ folds
   - Evaluate on remaining fold
   - Repeat for all folds
   - Average results
3. Choose model with best validation performance

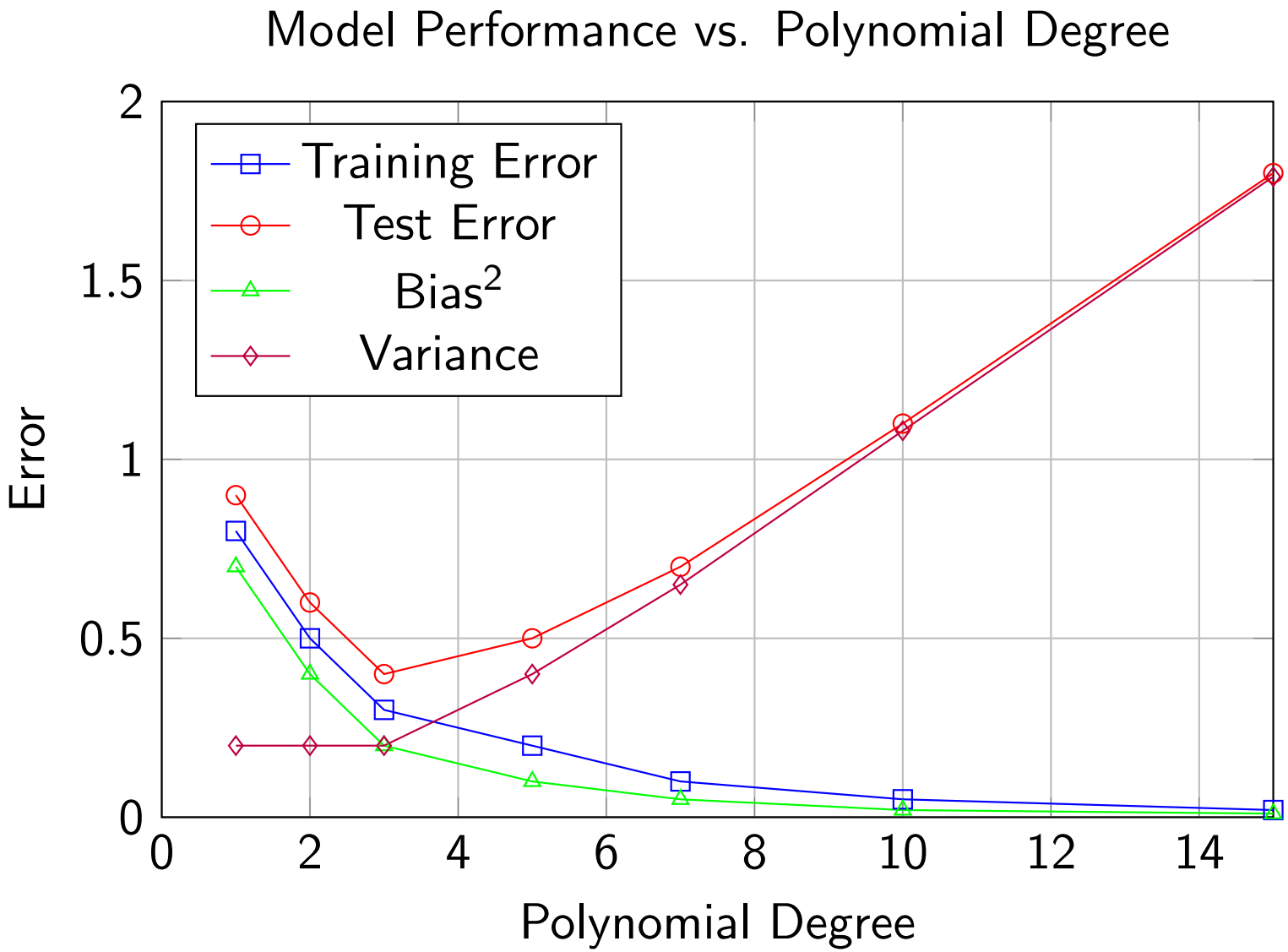


5-fold Cross-Validation

Benefits

- More reliable estimate of model performance
- Helps find optimal complexity without using separate test set
- Common choices: $k = 5$ or $k = 10$

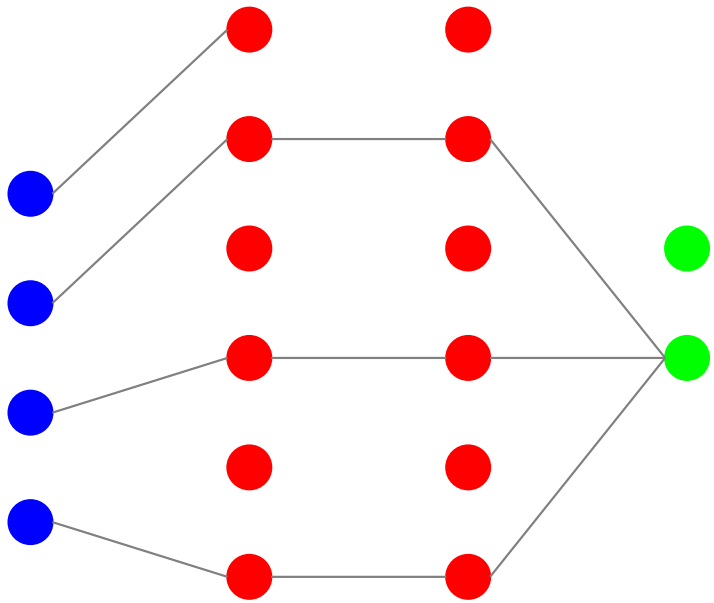
Experimental Results



- Training error consistently decreases with model complexity

Neural Network Depth and Width

- **Network depth:** Number of layers
  - Deeper networks can represent more complex functions (lower bias)
  - But harder to train and more prone to overfitting (higher variance)
- **Network width:** Neurons per layer
  - Wider networks have more parameters (lower bias)
  - But potentially higher variance
- **Regularization techniques:**
  - Dropout: randomly deactivate neurons
  - Weight decay: penalize large weights
  - Early stopping: stop training before overfitting



Practical Guidelines

1. **Start simple and gradually increase complexity**
   - Begin with a baseline model
   - Incrementally increase complexity while monitoring validation performance
2. **Use cross-validation to tune model complexity**
   - Find the sweet spot in the bias-variance tradeoff
3. **Consider regularization to control variance**
   - L1/L2 regularization, dropout, early stopping
4. **Adjust approach based on dataset size**
   - Small data: prioritize variance reduction
   - Large data: focus on bias reduction
5. **Use ensemble methods when appropriate**
   - Bagging for high-variance models
   - Boosting for high-bias models

Key Takeaways

- The bias-variance tradeoff is fundamental to understanding model performance
- Total error = Bias^2 + Variance + Irreducible Error
- Simple models: high bias, low variance
- Complex models: low bias, high variance
- Optimal model complexity balances bias and variance
- Regularization provides a way to control the tradeoff
- Shrinkage estimators can outperform unbiased ones
- Sample size affects the optimal complexity
- Cross-validation helps find the sweet spot
- Ensemble methods optimize the tradeoff

Further Reading

- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning*. Springer.
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural Networks and the Bias/Variance Dilemma. *Neural Computation*, 4(1), 1-58.
- Belkin, M., Hsu, D., Ma, S., & Mandal, S. (2019). Reconciling Modern Machine-Learning Practice and the Classical Bias-Variance Trade-Off. *Proceedings of the National Academy of Sciences*, 116(32), 15849-15854.