Generalization Error

ML Instruction Team, Fall 2022

CE Department
Sharif University of Technology

ML Cycle

- In every ML project:
  - You study the data.
  - You select a model.
  - You train it on the training data (i.e., it searches for model parameters that minimize a cost function).
  - As a final step, you apply the model to predict new cases, which is called inference, and you expect the model to generalize well.

- In addition to predicting the training examples correctly, the model should also be capable of generalizing to new cases.
  - It is only through the application of a model to new cases that we can determine how well it will generalize.
  - Putting your model into production and monitoring how well it performs is one way to do that.
  - The more suitable strategy would be to divide your data into two sets: a Training set and a Test set.

Measuring Generalization

- Training Set: which is used to train the model.
- Validation Set: which is used to tune the hyperparameters of the model.
- Test Set: which is used to measure the generalization performance.
- The losses on these subsets are called training , validation , and test loss, respectively.
- Cost Function: the average loss over the training set :

$$\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(y_i, \hat{y}_i)$$

- What is the purpose of the hyperparameter tuning in ML projects?

Bias + Variance

- What are Bias and Variance:
  - Bias: is commonly defined as the difference between the expected value of the estimator and the parameter that we want to estimate.
  - Variance: is defined as the difference between the expected value of the squared estimator minus the squared expectation of the estimator.

$$\mathrm{Bias}(\hat{\theta}) = \mathbb{E}[\hat{\theta}] - \theta, \quad \mathrm{Var}(\hat{\theta}) = \mathbb{E}[(\mathbb{E}[\hat{\theta}] - \hat{\theta})^2].$$

- Bias-Variance Decomposition:

$$\begin{aligned}
&\mathrm{MSE} := \mathbb{E}[(y-\hat{y})^2] \\
&= \mathbb{E}[y^2 + \hat{y}^2 - 2y\hat{y}] = \mathbb{E}[y^2] + \mathbb{E}[\hat{y}^2] - \mathbb{E}[y\hat{y}] \\
&= \mathrm{Var}(y) + \mathbb{E}[y]^2 + \mathrm{Var}[\hat{y}] + \mathbb{E}[\hat{y}]^2 - 2y\mathbb{E}[\hat{y}] \\
&= \mathrm{Var}(y) + \mathrm{Var}(\hat{y}) + (y^2 - 2yE[\hat{y}] + \mathbb{E}[\hat{y}]^2) \\
&= \mathrm{Var}(y) + \mathrm{Var}(\hat{y}) + (y - \mathbb{E}[\hat{y}])^2 \\
&= \varepsilon^2 + \mathrm{Var}[\hat{y}] + \mathrm{Bias}[\hat{y}]^2
\end{aligned}$$

Overfitting

- Overfitting means the model works well on training data, but it doesn't generalize well.
- Overfitting occurs when there is too much complexity in the model in comparison to the amount and noise in the training data.

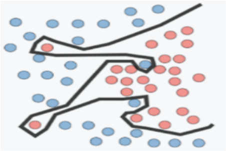

Figure: High Variance, Source

- How to fix this problem?
  - Simplify the model by selecting one with fewer parameters, reducing the number of attributes in the training data, or constraining the model.
  - Gather more training data.
  - Reduce the noise in the training data.

Underfitting

- Underfitting is the opposite of overfitting: it occurs when your model is too simple to learn the underlying structure of the data.

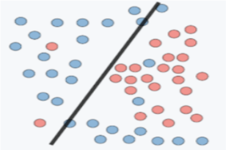

Figure: High Variance, Source

- The main options to fix this problem:
  - Selecting a more powerful model, with more parameters
  - Feeding better features to the learning algorithm (feature engineering)
  - Reducing the constraints on the model (e.g., reducing the regularization hyperparameter)

Overview

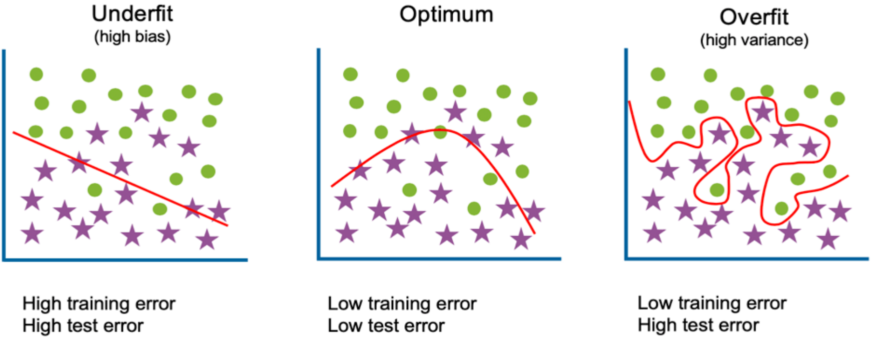

Figure: Overfitting vs Underfitting, Source

Overview

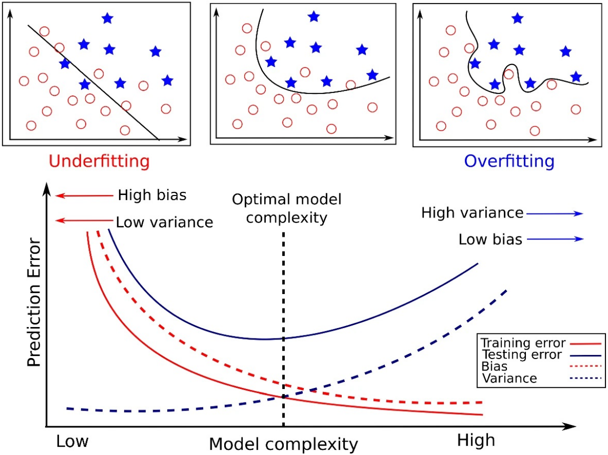

Figure: Overfitting vs Underfitting, Source

Thank You!

Any Question?