Model Optimization
Bias and Variance

\[ E[(y - \hat{f}(x))^2] = \text{Bias}[\hat{f}(x)]^2 + \text{Var}[\hat{f}(x)] + \sigma^2 \]

\[ \text{Bias}[\hat{f}(x)] = E[\hat{f}(x) - f(x)] \]

\[ \text{Var}[\hat{f}(x)] = E[\hat{f}(x)^2] - E[\hat{f}(x)]^2 \]

Error = (expected loss of accuracy)^2 + flexibility of model + irreducible error

https://en.wikipedia.org/wiki/Bias%E2%80%93variance_tradeoff
Question:
Why would there be a trade-off between bias and variance?
Underfitting

Underfitting means we have **high bias** and **low variance**.

- Lack of relevant variables/factor
- Imposing limiting assumptions
  - Linearity
  - Assumptions on distribution
  - Wrong values for parameters
Overfitting

Overfitting means we have **low bias** and **high variance**.

- Model fits too well to specific cases
- Model is over-sensitive to sample-specific noise
- Model introduces too many variables/complexities than needed
A Tale of Two Datasets
Parsimonious (adj.) - unwilling to spend money or use resources; stingy, frugal.

In data science, it pays to be parsimonious. (Occam’s Razor)
Model Goals

When training a model we want our models to:

- Capture the trends of the training data
- Generalize well to other samples of the population
- Be moderately interpretable

The first two are especially difficult to do simultaneously! The more sensitive the model, the less generalizable and vice versa.
Question:
Why is overfitting more difficult to control than underfitting?
Variance Reduction

Avoiding overfitting is a variance reduction problem.

Variance of the model is a function of the variances of each variable.

- Reduce the number of variables to use [Subset Selection]
- Reduce the complexity of the model [Pruning]
- Reduce the coefficients assigned to the variables [Regularization]

Cross-validation is used to test the relative predictive power of each set of parameters and subset of features.
Validation - Traditional

About 30% of the training set is reserved as a validation set.

Error on validation set serves as a good estimate of the test error.

- Advantage: useful especially if a test-set is not available
- Disadvantage: reduces size of available training data

Hastie et al. “Elements of Statistical Learning.”
More Generally: Cross Validation (CV)

Set of validation techniques that uses the training dataset itself to validate model

- Allows maximum allocation of training data from original dataset
- Efficient due to advances in processing power

Cross validation is used to test the effectiveness of any model or its modified forms.
Leave-p-Out Validation

For each data point:

- Leave out p data points and train learner on the rest of the data.
- Compute the test error for the p data points.

Define average of these \( \binom{n}{p} \) error values as validation error.
**K-fold Validation**

Create equally sized $k$ partitions, or **folds**, of training data.

For each fold:

- Treat the $k-1$ other folds as training data.
- Test on the chosen fold.

The average of these errors is the validation error.

Often used in practice with $k=5$ or $k=10$. 
Question:
How are $k$-fold and leave-$p$-out different?
Subset Selection

- **Best subset selection**: Test all $2^p$ subset selections for best one
- **Forward subset selection**
  - Iterate over $k = 0 \ldots (p-1)$ predictors
  - At each stage, select the best model with $(p-k)$ predictors
  - Find best model out of the p-1 selected candidates with CV
- **Backward selection** - Reverse of forward subset selection
  - Start from $p$ predictors and work down

In practice, best subset selection method is rarely used, why?
Regularization

We defined our error up until now as:

\[ SS_{(residuals)} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \]

Minimizing this equation on training data = minimizing training loss.
But we can often do better!
Regularization

To avoid overfitting, we add a penalty term independent of the data, known as regularization.

\[
\text{Error} = (\text{Training Loss})^2 + \text{Regularization}
\]

- Ridge Regression
- Lasso Regression
Ridge Regression

Uses $L_2$ - regularization penalty:

$$
\sum_{i=1}^{n} \left( y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = RSS + \lambda \sum_{j=1}^{p} \beta_j^2,
$$

$\lambda$ is the penalty threshold constant and controls sensitivity.

- Useful for non-sparse, correlated predictor variables
- Used when predictor variables have small individual effects
- Limits the magnitudes of the coefficient terms, but not to 0
Lasso Regression

Uses $L_1$ regularization penalty:

$$
\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| = \text{RSS} + \lambda \sum_{j=1}^{p} |\beta_j|.
$$

This time the penalty term uses absolute value rather than squaring.

- Useful for sparse, uncorrelated variables
- Used when there are few variables with medium to high effects
- Drives coefficients to 0 when $\lambda$ sufficiently large (performs feature selection)
Training Accuracy vs Test Accuracy

Key idea: Regularization and cross-validation are techniques to limit the model’s sensitivity.

In practice, if CV error is high:

- Compare with training
- If significantly lower:
  - Raise penalty constant
  - Try different subset
  - Try different parameters
Coming Up

Your problem set: None

Next week: Things are going to get meta.

See you then!