Intro to Statistical Learning

What is Statistical Learning?

Modeling

Every analysis we will do assumes a structure like:

(output) = f(input) + (noise)

... or, if you prefer...

(response variables) = f(explanatory variables) + (noise)

(dependent variables) = f(independent variables) + (noise)

(target) = f(predictors) + (noise)

Modeling

In any case: we are trying to reconstruct information in **data**, and we are hindered by **random noise**.

The function f might be very simple...

Y = X + (\epsilon)

... or very complex

$$
\begin{aligned}
z_i &= b_0 + b_1 x_i \\
q_i &= \frac{1}{1 + \exp(-z_i)} \\
y_i &\sim \mathrm{Bern}(q_i)
\end{aligned}
$$

This or That?

Statistical Learning vs. Machine Learning

You will often hear people refer to **machine learning** in reference to the topics in this class.

My opinion:

Statistical learning is more concerned with the *model structure*, *interpretation of estimates*, and *understanding error*.

Machine learning is more concerned with *model implementation* and *computational demands*.

Quantitative (numeric) vs. qualitative (categorical)

Often, the nature of our models will differ depending on the types of data involved!

Regression vs. Classification

regression = the response variables are *quantitative*

classification = the response variables are *categorical*

Supervised vs. Unsupervised

supervised learning = our data includes observations of the output variable

> What drug treatments are associated with better disease outcomes?

unsupervised learning = our data does NOT include any observations of the output variable

> What social groups already exist among the Stat 434 students?

Prediction vs. Inference

So, why do we care about estimating f?

prediction: We are trying to use future **inputs** to guess about future **outputs**.

> Which advertisements is Dr. B. most likely to click on Instagram?

inference: We are trying to tell a story about the **relationship** between variables.

> Which genes are more activated when breast cancer is present?

What do we need to learn?

Why not just "plug-and-chug"?

It is important to think carefully about:

- **Assumptions:** What do various models assume to be true about the data structure? Are these justified?
- **Interpretations:** What can we learn by estimating f for a particular model? Is that information what we are looking for?
- **Estimation:** How is each f being approximated? Will this be a close approximation?
- **Usage:** What are we going to do once we estimate f? Do certain models lend themselves better than others?

Estimation

If we are doing **prediction**, we mostly don't care about *assumptions*.

The "best" model is the model that predicts most accurately.

If we are doing **inference**, we care a lot about *assumptions*.

The "best" model is the one that matches the truth.

In this Class

You will learn:

- To apply many different models to real data using R or python.
- To interpret the output of these model estimates
- To use *cross-validation* to compare models
- To explain the general structure and philosophy behind each model
- To select an appropriate "best" model for a data analysis, and make a well-reasoned argument for your choice.