Statistical Learning Theory

[(https://github.com/learn-co-curriculum/dsc-stat-learning-theory-v2-5)](https://github.com/learn-co-curriculum/dsc-stat-learning-theory-v2-5) [(https://github.com/learn-co-curriculum/dsc-stat-learning-theory-v2-5/issues/new/choose)](https://github.com/learn-co-curriculum/dsc-stat-learning-theory-v2-5/issues/new/choose)

Introduction

Before you get into building machine learning models, a basic understanding of statistical learning theory is essential.

Objectives

You will be able to:

- Explain the difference between modeling for inference and prediction
- Explain generalization in the context of statistical modeling

Statistical Learning Theory

Statistical learning theory is based on the idea of using data along with statistics to provide a framework for learning.

In statistical learning theory, the main idea is to construct a **model** to draw certain conclusions from data, and next, to use this model to make **predictions.**

This builds on statistical modeling, which represents the relationship between independent and dependent variables as a mathematical equation. For parametric statistical models such as linear regression, this means that the model learns **parameters** that will be used for future predictions.

Inference vs. Prediction

There are two different modeling approaches to statistical modeling: modeling for ***inference*** and modeling for ***prediction***. A "perfect" model might be useful for both, but often your modeling strategy will need to be calibrated based on the goal of the model.

Inference

When you are modeling for inference, you are asking the question:

> What is the relationship between `x` and `y` ?

and sometimes, if you have good reason to infer a causal relationship:

> How does `x` affect `y` ?

where `x` is your collection of independent variables (i.e. features) and `y` is your dependent variable (i.e. target). The focus is on *understanding*. Most of the history of statistics and all of the linear regression content so far has been focused on this approach.

When modeling for inference, it is important for the model to be **statistically significant** and **interpretable**, sometimes at the expense of overall model fit. Every feature used in the model should be carefully chosen based on underlying domain understanding.

Prediction

When you are modeling for prediction, you are asking the question:

> How well can I use `x` to predict `y` ?

`x` is still your collection of independent variables, and `y` is still your dependent variable. But you are less concerned about how and which features impact `y` as opposed to how you can efficiently use them to predict `y` .

When modeling for prediction, it is important for the model to ***generalize*** to unseen data. This means that the overall model fit is more important than the coefficients of features or statistical significance, and that you will often use all available features rather than carefully choosing them. Both in terms of the number of features and in terms of the type of model used, predictive models tend to be more ***complex*** than inferential models.

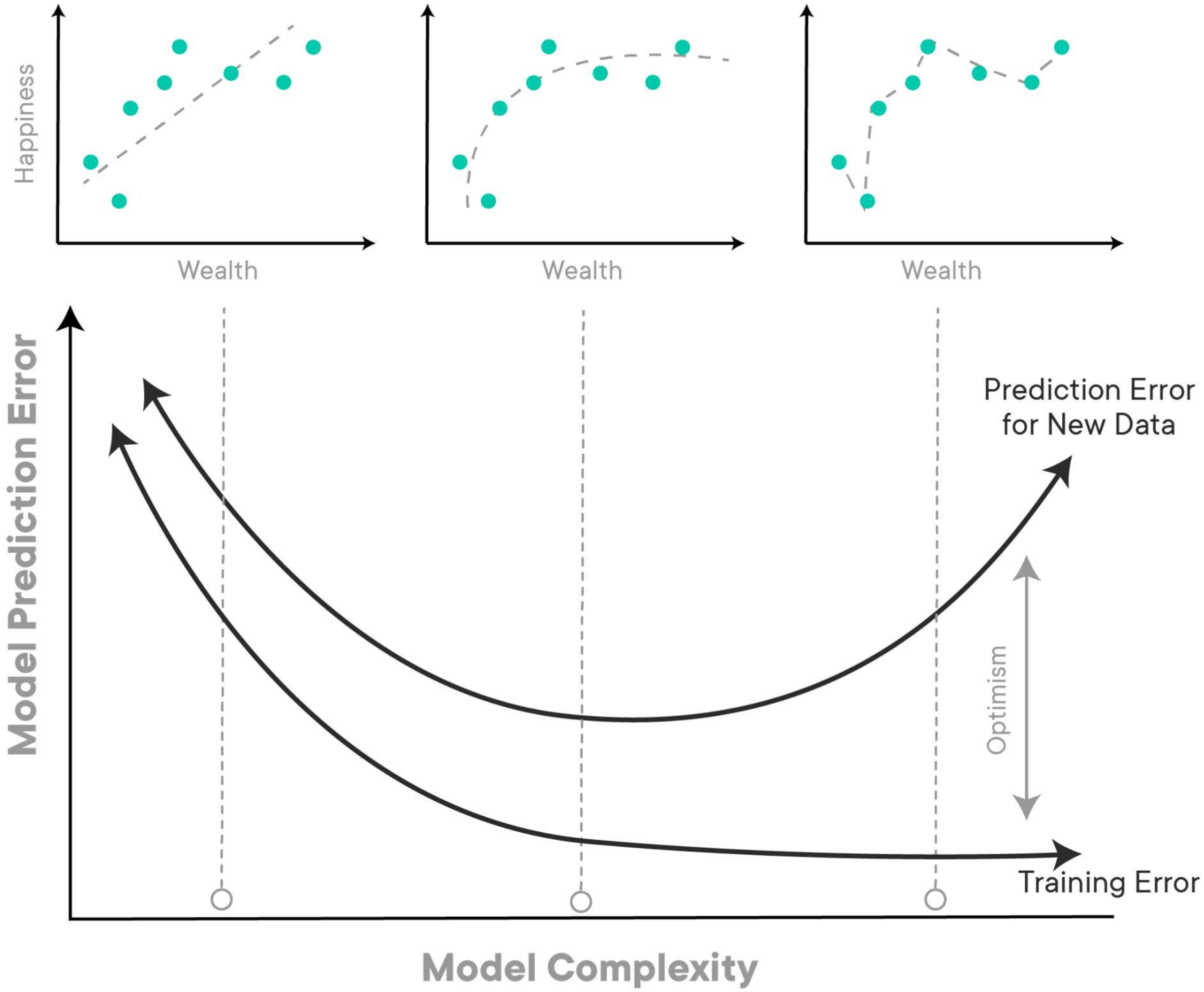
Model Generalization

The model learns about the data during the ***training*** stage. Examples are presented to the model and the model tweaks its parameters to better understand the data.

Once the training is over, the model is unleashed upon new data and then uses what it has learned to make predictions with that data. This is where problems can emerge. If we over-train the model on the training data -- i.e. make the model memorize every detail of the data it is shown -- it will be able to identify all the relevant information in the training data, but will fail miserably when presented with the new data.

We then say that the **model is not capable of generalizing**, or that the **model is over-fitting the training data.**

Let's take a look at an example of the phenomenon: modeling happiness as a function of wealth.



In the top three diagrams, we have data and models (dashed curves). From left to right the models have been trained longer and longer on the training data. The training error curve in the bottom box shows that the training error gets better and better as we train longer (increasing model complexity).

You may think that if we train longer we'll get better! Well, yes, but **only better at describing the training data**. The top right box shows a very complex model that hits all the data points. This model does great on the training data, but when presented with new data (examine the prediction error curve in the bottom box) then it does worse! The gap between the training error and prediction error for new data (labeled "optimism") is growing as model complexity increases, which means that we are getting *worse* at generalizing.

In order to create good predictive models in machine learning that are capable of generalizing, one needs to know when to stop training the model so that it doesn't over-fit.

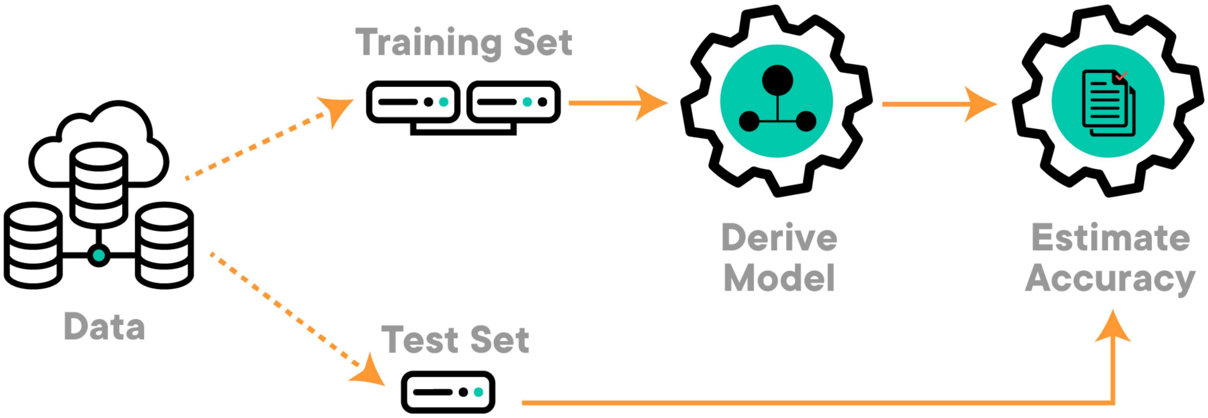
Model Validation

As the data which is available to us for modeling is finite, the available data needs to be used very effectively to build and **validate** a model. Model validation is a process of measuring overfitting and indicates the degree of generalizability.

Here is how we perform validation, in its simplest form:

- Split the data into two parts with a 70/30, 80/20, or a similar split
- Use the larger part for **training** so the model learns from it
- Use the smaller part for ***testing*** the model

This setup looks like as shown below:



This is called a ***train-test split*** and means that you can compare the model performance on training data vs. testing data using a given **metric.** The metric can be R-Squared or it can be an error-based metric like RMSE.

If the metric is much better on the training data than the test data, this indicates overfitting. A model that generalizes well will have similar metrics on the two datasets.

Another approach to validation is ***cross-validation***. This involves splitting the data multiple times and training multiple models, to get more of a distribution of possible metrics rather than relying on metrics from a single train-test split.

Additional Resources

- Youtube: Introduction to Statistical Learning Theory (https://www.youtube.com/watch?v=rqJ8SrnmWu0)

  (https://www.youtube.com/watch?v=rqJ8SrnmWu0)

- An Overview of Statistical Learning Theory with examples (https://www.princeton.edu/%7Ekulkarni/Papers/Journals/j077_2011_KulHar_WileyTutorial.pdf)

Summary

In this lesson, you briefly looked at statistical learning theory and its main components. When modeling for inference rather than prediction, some additional conceptual considerations become important. In particular, predictive models should generalize to unseen data. Model validation is used to measure how well a model will generalize.

How do you feel about this lesson?

👍 👎

Have specific feedback?

Tell us here! (https://github.com/learn-co-curriculum/dsc-stat-learning-theory-v2-5/issues/new/choose)