EE445 Section 4: Least Squares Data Fitting

References:

- [VMLS]: Chapter 13

Plan

- Univariate Data Fitting
- Polynomial fitting
- Validation

Univariate Data Fitting

Motivation: Goal

Let's say you are an executive at Uber.
You want to understand how the prices x of Lyft affect the demand y on your app.
This can help you decide where to place drivers.

Motivation: Modelling the Problem

We believe that x and y are related approximately by a function

$$y = f(x)$$

We want to "estimate" or "learn" this function.

Assumptions You assume that there will be a direct linear relationship between Lyft's prices and your demand.
The higher Lyft's prices are, the less people will use Lyft, and the more people will use Uber.

$$\hat{f}(x) = \theta_1 + \theta_2 x$$

Collect Data You collect data of Lyft's average price over many days $\{x_1 \dots x_n\}$ and your demand on each of those days $\{y_1 \dots y_n\}$.

Motivation: Formulate as LS problem

$$\min_{\theta} \sum_{i=1}^{n} (\theta_1 + \theta_2 x_i - y_i)^2 = \min_{\theta} \|\theta_1 \mathbf{1} + \theta_2 x^d - y^d\|^2$$

This problem of "estimating" a function can now be thought of as finding an approximate solution to a system of linear equations, which we saw last week in lecture.

In the above $x^d = (x_1, \ldots x_d)$ and $y^d = (y_1, \ldots y_d)$ as in the textbook notation.

Univariate function fitting: Interpretable solution

$$\hat{f}(x) = \operatorname{avg}\left(y^{\mathrm{d}}\right) + \rho \frac{\operatorname{std}\left(y^{\mathrm{d}}\right)}{\operatorname{std}\left(x^{\mathrm{d}}\right)} \left(x - \operatorname{avg}\left(x^{\mathrm{d}}\right)\right)$$

We can rewrite the above as

$$\hat{f}(x) - \operatorname{avg}\left(y^{\mathrm{d}}\right) = \rho \frac{\operatorname{std}\left(y^{\mathrm{d}}\right)}{\operatorname{std}\left(x^{\mathrm{d}}\right)} \left(x - \operatorname{avg}\left(x^{\mathrm{d}}\right)\right)$$

VMLS 13.2: Regression to the mean

Consider a data set in which the (scalar) $x^{(i)}$ is the parent's height (average of mother's and father's height), and $y^{(i)}$ is their child's height. Assume that over the data set the parent and child heights have the same mean value μ, and the same standard deviation σ. We will also assume that the correlation coefficient ρ between parent and child heights is (strictly) between zero and one. (These assumptions hold, at least approximately, in real data sets that are large enough.) Consider the simple straight-line fit or regression model given by (13.3), which predicts a child's height from the parent's height. Show that this prediction of the child's height lies (strictly) between the parent's height and the mean height μ (unless the parent's height happens to be exactly the mean μ). For example, if the parents are tall, i.e., have height above the mean, we predict that their child will be shorter, but still tall. This phenomenon, called regression to the mean, was first observed by the early statistician Sir Francis Galton (who indeed, studied a data set of parent's and child's heights).

Polynomial Fitting

Polynomial Regression

A simple extension beyond the straight-line fit is a polynomial fit, with

$$f_i(x) = x^{i-1}, \quad i = 1, \ldots, p,$$

so $\hat{f}$ is a polynomial of degree at most $p - 1$,

$$\hat{f}(x) = \theta_1 + \theta_2 x + \cdots + \theta_p x^{p-1}$$

In this case the matrix A has the form

$$A = \begin{bmatrix} 1 & x_1 & \cdots & (x_1)^{p-1} \\ 1 & x_2 & \cdots & (x_2)^{p-1} \\ \vdots & \vdots & & \vdots \\ 1 & x_N & \cdots & (x_N)^{p-1} \end{bmatrix}$$

LS Problem and solution

$$\min_{\theta \in \mathbb{R}^n} \sum_{i=1}^{n} (\hat{f}(x_i) - y_i)^2 \; = \min_{x \in \mathbb{R}^n} \|A\theta - y^d\|^2 \tag{1}$$

In lecture, we saw that the optimal solution to this problem is

$$\hat{x} = (A^\top A)^{-1} A^\top y$$

Move to part (a) of the notebook.

Validation

Intuitive overfitting: Setup

This problem is meant to be a short toy problem that explores the trade-off between the complexity of a model and its fit to the training data. If we make our models complicated enough, then we can fit pretty much anything.

Julia frequently goes to the pool to prepare for her upcoming triathlon. In the past month, she went to the pool 19 times. We want to look at the data and predict when she will go to the pool next month. Here is the data from this month:
Week 1: Sunday, Monday, Tuesday, Thursday, Friday
Week 2: Monday, Tuesday, Thursday, Saturday
Week 3: Sunday, Monday, Thursday, Friday, Saturday
Week 4: Monday, Tuesday, Thursday, Friday, Saturday

Intuitive overfitting: Models

Consider the following four explanations for her swimming habits:

1. "She goes to the pool every day"
2. "She goes to the pool every day except Wednesdays"
3. "She goes to the pool every day except Wednesdays and even Sundays"
4. "She goes to the pool every day except Wednesdays and even Sundays and the Tuesday of third week of the month and the Saturday of the first week of the month and the Friday of the second week of the month."

Intuitive overfitting: Interpretation

(a) Out of the 28 days in the four weeks, how many does each rule get right? That is, on how many days does the rule predict whether or not she will be at the pool?

(b) Which rule is the "best" rule if we only measure how well it does on the training data?

(c) Now, we have an intuitive notion that the later rules are more complex than the earlier rules. This is a rather difficult notion to formalize, but let's give it a shot by trying to measure the number of "cases" created by each rule. If we wanted to implement these rules in code, how many "case" statements (if/elif/elif/elif/else) would it take? Let's assume that we're only allowed to use AND statements, not OR statements (so that we can't combine cases).

(d) On Week 5, the schedule was "Sunday, Monday, Tuesday, Thursday, Friday, Saturday" and on Week 6 it was "Tuesday, Thursday, Friday, Saturday". How does this change your conclusions?