CS 524: Introduction to Optimization
Lecture 29 : Regression and tradeoffs

Michael Ferris

Computer Sciences Department
University of Wisconsin-Madison

November 10, 2023

Data Science

- Extract meaning from data: learning
- Use this knowledge to make predictions: inference
- Optimization provides tools for modeling / formulation / algorithms
- Modeling and domain-specific knowledge is vital in practice: "80% of data analysis is spent on the process of cleaning and preparing the data."

Typical Setup

After cleaning and formatting, obtain a data set of m objects:

- Vectors of features: a_j, $j = 1, 2, \ldots, m$
- Outcome / observation / label y_j for each feature vector

$$A = \begin{bmatrix} a_1^T \\ a_2^T \\ \vdots \\ a_m^T \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$$

The outcomes y_j could be:

- a real number: regression
- a label indicating the a_j lies in one of M classes (for $M \geq 2$): classification. (M can be very large)
- no labels (y_j is null):
  - subspace identification: locate low-dimensional subspaces that approximately contain the (high-dimensions) vectors a_j
  - clustering: partition the a_j into clusters; each cluster groups objects with similar features.

Fundamental Data Analysis

Seek a function ϕ that:

- approximately maps a_j, to y_j for each j; $\phi(a_j) \approx y_j$, $j = 1, 2, \ldots, m$
- satisfies additional properties to make it "plausible" for the application, robust to perturbations in the data, generalizable to other data samples from the same distribution.

Can usually define ϕ in terms of some parameter vector x - thus identification of ϕ becomes a data-fitting problem:

- Find a nice x such that $\phi(a_j; x) \approx y_j$ for $j = 1, 2, \ldots, m$
- Objective function in this problem often built up of m terms that capture mismatch between predictions and observations for data item (a_j, y_j)
- The process of finding ϕ is called learning or training.

The process to generate ϕ

We are given a set of examples $a_j \in \mathbb{R}^n$ and we wish to predict either:

- a function output y_j using $y_j = \phi(a_j)$ i.e. learn function ϕ, or
- a class to which a_j belongs, i.e. y_j is a label for that class, e.g. benign, malignant

- These are broadly defined as regression or classification problems
- We aim to build a regression function $\phi : \mathbb{R}^n \mapsto \mathbb{R}$ or a classifier function $\phi : \mathbb{R}^n \mapsto \{benign, malignant\}$

- Use a set of examples, the training set, to build (learn) the function, and
- a set of new (unseen) data, the testing set, to see how well the function performs (generalize)

What's the use of the mapping ϕ?

- Prediction: Given new data vector a_k, predict outputs $y_k \leftarrow \phi(a_k; x)$.
- Analysis: ϕ (more particularly the parameter x) reveals structure in the data

Many possible complications:

- Noise or errors in a_j and y_j
- Missing data:
- Overfitting: ϕ exactly fits the set of training data (a_j, y_j) but predicts poorly on "out-of-sample" data (a_k, y_k)

Regression (see 29regression.ipynb)

Introduce the idea of linear regression. Predictor variables, response variables / observations.
Example. See file `iowheat.dat`. In wheat farming, predictor variables, are rainfalls and average temps in the weeks of the growing season, amounts of fertilizer used, chemicals in soil, etc. Response variable is amount of harvest. Each observation corresponds to a single growing season.
Each observation consists of a vector $a_i \in \mathbb{R}^n$ of predictor variables, and a scalar response variable y_i. Have $i = 1, 2, \ldots, m$ observations. Seek a vector $x \in \mathbb{R}^n$ of coefficients such that

$$a_i^T x \approx y_i, \quad i = 1, 2, \ldots, m.$$

(Usually make one of the predictor variables a constant 1 to allow for any systematic shift.)

- Talk about standardization of the columns of A prior to solving. A scaling issue to ensure that components of x are similar in size.
- If $A_{\cdot j}$ is multiplied by c, its coefficient is divided by c
- If y is multiplied by c, *all* coeffs are multiplied by c
- Neither t not F statistics are affected by changing the units of measurement of any variable
- Idea is to replace y and each A variable with a standardized version - subtract mean and divide by standard deviation.
  - Coeffs reflect standard deviation of y for one standard deviation change in $A_{\cdot j}$
  - Can compare the magnitudes of x to conclude importance

Loss functions

Achieve this by minimizing an objective function of the form

$$L(x) = \sum_{i=1}^{m} \rho_i(a_i^T x - y_i) = \sum_{i=1}^{m} \rho_i(r_i),$$

where $r_i \stackrel{\text{def}}{=} a_i^T x - y_i$ is the *residual* and ρ_i is a *loss function*. Loss functions have the property that $\rho(r) = 0$ when $r = 0$, and $\rho \geq 0$ otherwise. Also $\rho(r)$ increases with $|r|$.

Most common loss function is least squares: $\rho(r) = \frac{1}{2}r^2$. To write the corresponding $L(x)$ more compactly, we can assemble a matrix and vector from the observations:

$$A = \begin{bmatrix} a_1^T \\ a_2^T \\ \vdots \\ a_m^T \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}.$$

Now can write

$$\begin{aligned} L(x) &= \frac{1}{2}\sum_{i=1}^{m}(a_i^T x - y_i)^2 = \frac{1}{2}\|Ax - y\|_2^2 \\ &= \frac{1}{2}x^T(A^T A)x + x^T(A^T y) + y^T y. \end{aligned}$$

Minimizer (obtained by setting $L'(x) = 0$) is at

$$x = -(A^T A)^{-1}A^T y$$

May have (e.g. positivity) constraints on x. Cannot in general write solution in closed form; must solve an optimization problem to find x. If the constraints are linear, the problem is a QP.

Least-squares is sensitive to *outliers*: if some residual r_i is large it can affect the solution x significantly. Other loss functions are more "robust" in the presence of outliers. ℓ_1 loss function is $\rho(r) = |r|$. Then get

$$L_1(x) = \sum_{i=1}^{m} |a_i^T x - y_i|.$$

Can formulate $\min_x L_1(x)$ as an LP, by splitting r_i into positive and negative parts. If linear or bound constraints are present it remains an LP.

Huber is a hybrid of least-squares and ℓ_1. Behaves like least squares (ℓ_2) for r near 0, otherwise like ℓ_1. Define

$$\rho_H(r) = \begin{cases} \frac{1}{2}r^2 & |r| \leq \sigma \\ \sigma|r| - \frac{1}{2}\sigma^2 & |r| \geq \sigma. \end{cases}$$

Here $\sigma > 0$ is a given parameter. The function ρ_H is C^1 but not C^2 (i.e. there is a discontinuity in the second derivative at $|r| = \sigma$). Still, an NLP solver will usually work.

The Huber regression problem is thus:

$$\min_x L_H(x) := \sum_{i=1}^{m} \rho_H(a_i^T x - y_i)$$

Surprisingly, can also formulate Huber estimation as a QP! We omit the details; the derivation requires duality. The formulation is:

$$\min_w \ \frac{1}{2} w^T w + y^T w \ \text{ s.t. } \ A^T w = 0, \ \ -\sigma e \leq w \leq \sigma e.$$

The marginal variables for the constraint $A^T w = 0$ are identical to the solution x of the Huber problem in primal form.
Constraints, for example may know that some components of x must be nonnegative. Then the problem becomes a bona fide optimization problem.

Optimal tradeoffs

We often want to optimize several different objectives simultaneously, but these objectives are **conflicting**.

- risk vs expected return (finance)
- power vs fuel economy (automobiles)
- quality vs memory (audio compression)
- space vs time (computer programs)
- mittens vs gloves (winter)

Optimal tradeoffs

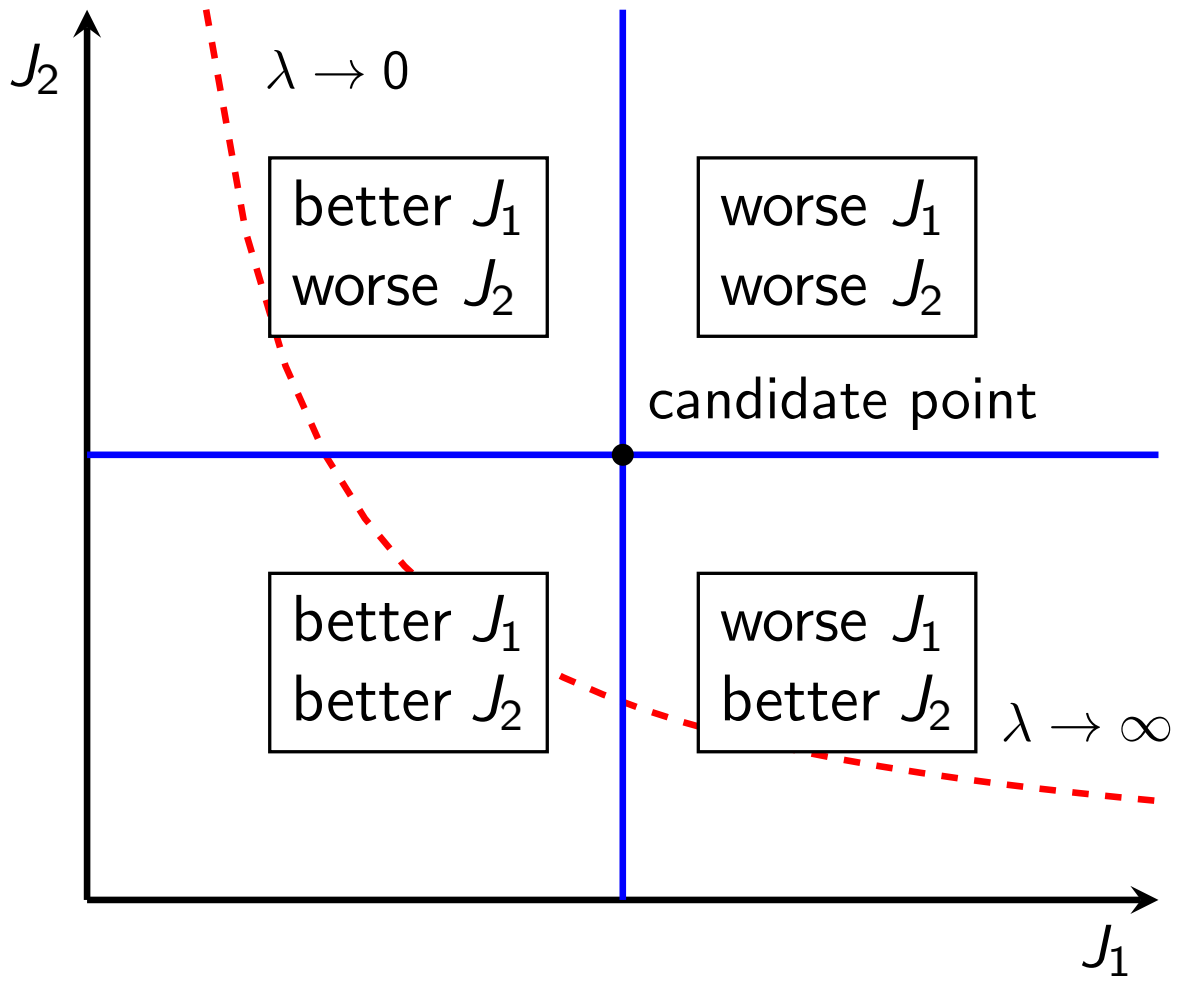
- Suppose $J_1 = \|Ax - b\|^2$ and $J_2 = \|Cx - d\|^2$.
- We would like to make **both** J_1 and J_2 small.
- A sensible approach: solve the optimization problem:

$$\underset{x}{\text{minimize}} \quad J_1 + \lambda J_2$$

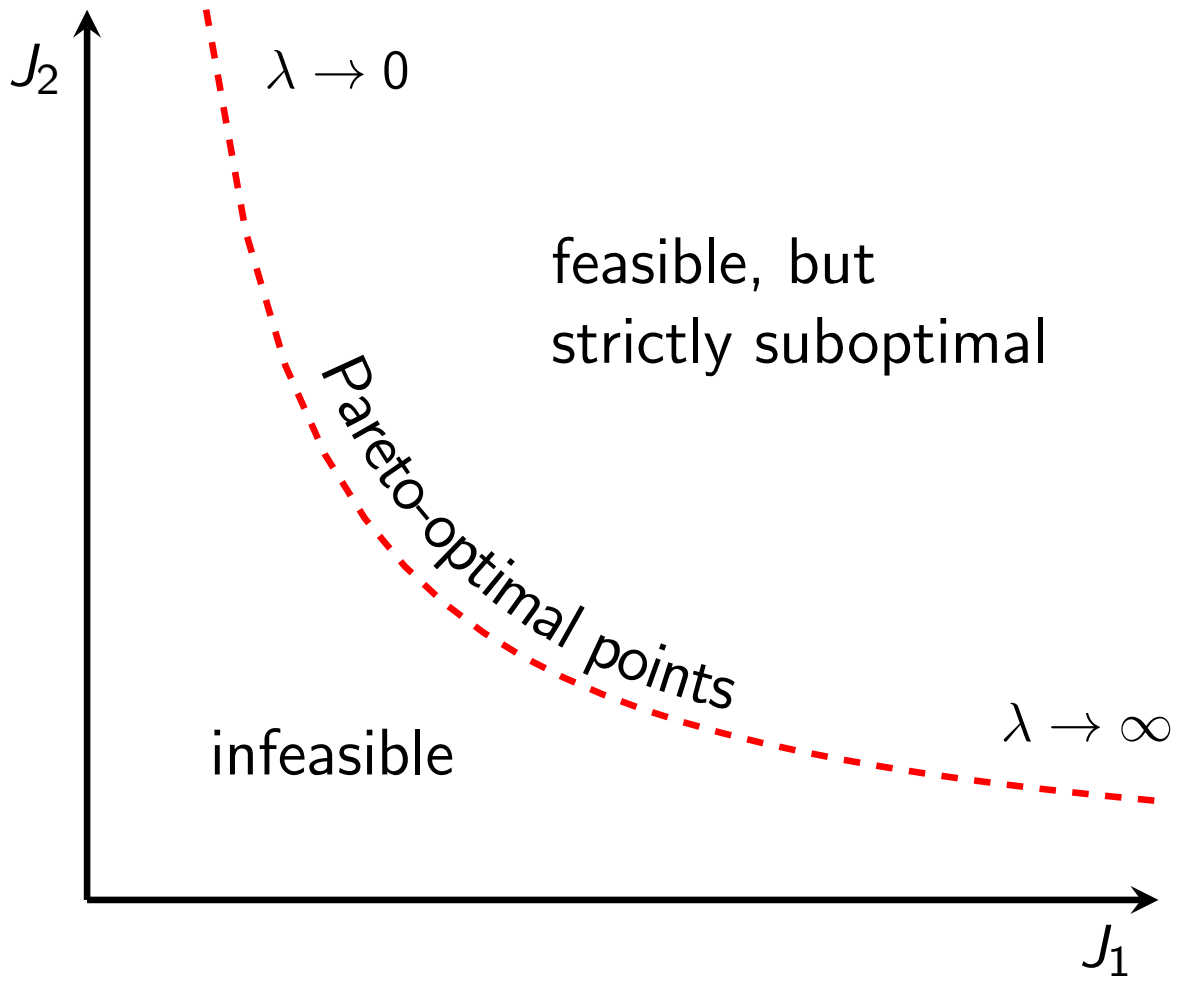
  where $\lambda > 0$ is a (fixed) **tradeoff parameter**.

- Then tune λ to explore possible results.
  - When $\lambda \to 0$, we place more weight on J_1
  - When $\lambda \to \infty$, we place more weight on J_2

Pareto curve

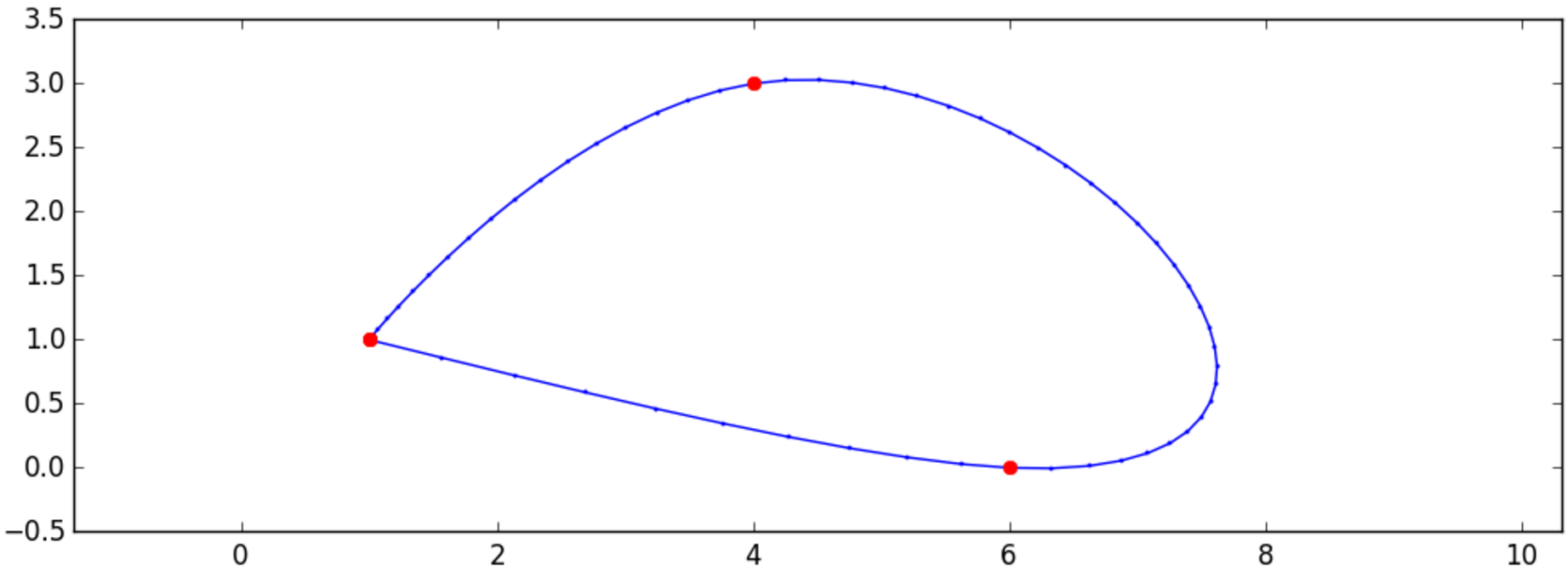


Pareto curve



See hovercraft example

We are in command of a hovercraft. We are given a set of k waypoint locations and times. The objective is to hit the waypoints at the prescribed times while minimizing fuel use.



Tradeoff fuel cost versus exactness of hitting waypoints