

## Lecture 4: Supervised learning introduction

---

CSI 5v93: Introduction to machine learning

Baylor University  
Computer Science Department

Dr. Greg Hamerly  
<http://cs.baylor.edu/~hamerly/>

CSI 5v93: Introduction to machine learning, Lecture 4 – p. 1/14

## Announcements

---

- Homework 1 due today

CSI 5v93: Introduction to machine learning, Lecture 4 – p. 2/14

## Questions?

---

CSI 5v93: Introduction to machine learning, Lecture 4 – p. 3/14

## Chapter 2: Overview of supervised learning

---

- 2.1 – Introduction
- 2.2 – Variable types and terminology
- 2.3 – Two simple approaches to prediction: Least squares and nearest neighbors
- 2.4 – Statistical decision theory
- 2.5 – Local methods in high dimensions
- 2.6 – Statistical models, supervised learning, and function approximation
- 2.7 – Structured regression models
- 2.8 – Classes of restricted estimators
- 2.9 – Model selection and the bias-variance tradeoff

CSI 5v93: Introduction to machine learning, Lecture 4 – p. 4/14

## Questions from last lecture: The Curse

---

Question: *In higher dimensions, don't relative distances stay the same, regardless of actual distance?*

Answer: No, not generally. Try the following in Matlab.

```
x = [0 1 3]'; % 3 points in 1 dimension
y = [[0 1 3]' rand(3, 9) ]; % 3 points in 10 dimensions

a = norm(x(2) - x(1)) % equals 1
b = norm(x(3) - x(1)) % equals 3
b / a % equals 3

c = norm(y(2,:) - y(1,:)) % I got 1.7363
d = norm(y(3,:) - y(1,:)) % I got 3.0972
d / c % I got 1.7838, less than 3
```

This shows two things:

- relative distances in higher dimensions don't stay the same
- relative distances are “washed out,” providing weaker discrimination

CSI 5v93: Introduction to machine learning, Lecture 4 – p. 5/14

## Questions from last lecture: Terminology

---

MODELS: things like *nearest-neighbor* and *linear model* are models.

PARAMETER ESTIMATION FRAMEWORKS: things like *least-squares* and *maximum likelihood estimation* are not models, they are ways of finding parameters of models.

Also, there is no difference between  $\beta$  for linear models and  $\theta$  that we discuss in general. Just different names for the parameters.

CSI 5v93: Introduction to machine learning, Lecture 4 – p. 6/14

## Questions from last lecture: When do we use what?

---

We have different models (linear, nearest-neighbor, etc.) and different learning methods (least-squares, maximum likelihood, etc.).

Use the model that makes sense for the problem.

Often we choose the learning technique that works best. Sometimes ML and least-squares are equivalent (under the additive Gaussian noise error model).

For some models, one learning technique is more natural. For example, for the linear model, least squares is often more natural.

## Questions from last lecture: log-likelihood

---

How do we get this?

$$\mathcal{L}(\theta|Y) = \log \Pr(\theta|Y) = \sum_{i=1}^n \log \Pr(y_i|\theta)$$

Starting with the definition, using Bayes' rule, and making some assumptions:

$$\begin{aligned} \mathcal{L}(\theta|Y) &\equiv \log \Pr(\theta|Y) && \text{(definition)} \\ &= \log \left[ \frac{\Pr(Y|\theta) \Pr(\theta)}{\Pr(Y)} \right] && \text{(by Bayes' rule)} \\ &\propto \log \Pr(Y|\theta) && \text{(by assuming constant prior/total probabilities)} \\ &= \log \prod_{i=1}^n \Pr(y_i|\theta) \\ &= \sum_{i=1}^n \log \Pr(y_i|\theta) \end{aligned}$$

## Structured regression models (2.7)

---

Consider the RSS criterion for an arbitrary function  $f$ :

$$RSS(f) = \sum_{i=1}^n (y_i - f(x_i))^2$$

If we have no restrictions on  $f$ , then minimizing this function leads to infinitely many functions.

There are an infinite number of functions that pass through the training points  $\{x_i, y_i\}$  (or through the average  $y_i$  values for the same  $x_i$ ).

In order to obtain useful solutions, we must restrict  $f$ .

## Constraints

---

Constraining a function can be described using complexity restrictions.

These complexity restrictions can be thought of as “regular behavior in a neighborhood” (e.g. constant, linear, or low-order polynomial).

The strength of the constraint and the size of the neighborhood are positively correlated.

## Classes of restricted estimators (2.8)

---

There are several methods of modelling constraints in regression-type problems, which we touch on briefly:

Roughness penalty (aka Bayesian method, aka regularization)

$$PRSS(f; \lambda) = RSS(f) + \lambda J(f)$$

Here we must pick  $\lambda$  and  $J$ , and they correspond to Bayesian priors on the variability of  $f$ .

For  $\lambda = 0$ , no penalty is imposed. For  $\lambda = \infty$ , only linear models are allowed.

## Kernel methods and local regression

---

A *kernel* is a function that explicitly defines the neighborhood around a point  $x_0$ :

$$K_\lambda(x_0, x)$$

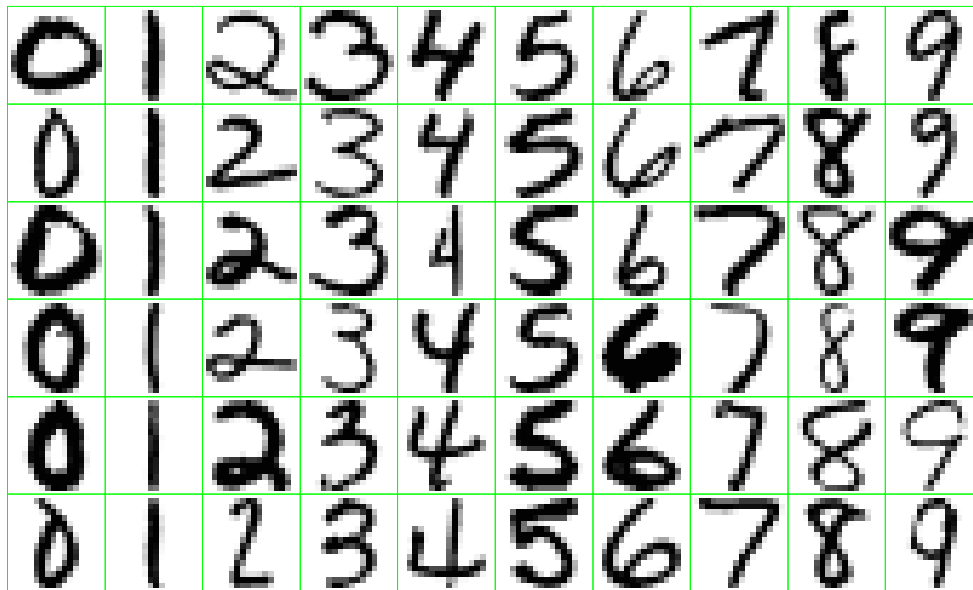
Examples: Gaussian kernel, constant-width kernel (on board)

The kernel's primary parameter is the *kernel width*, which defines the size of the neighborhood.

A kernel + a local model allow flexible models where the penalty is the kernel width.

## ZIP code demonstration with $k$ -nearest neighbors

---



CSI 5v93: Introduction to machine learning, Lecture 4 – p. 13/14

### 2-minute journal

---

Please write a response to the following on a piece of paper and hand it in immediately. Please make it anonymous (no names). Write about:

- major points you learned today
- areas not understood or requiring clarification

CSI 5v93: Introduction to machine learning, Lecture 4 – p. 14/14