Lecture 12: Introduction to Machine Learning

So far we seen two major topics in the course, signal processing & dimensionality reduction.

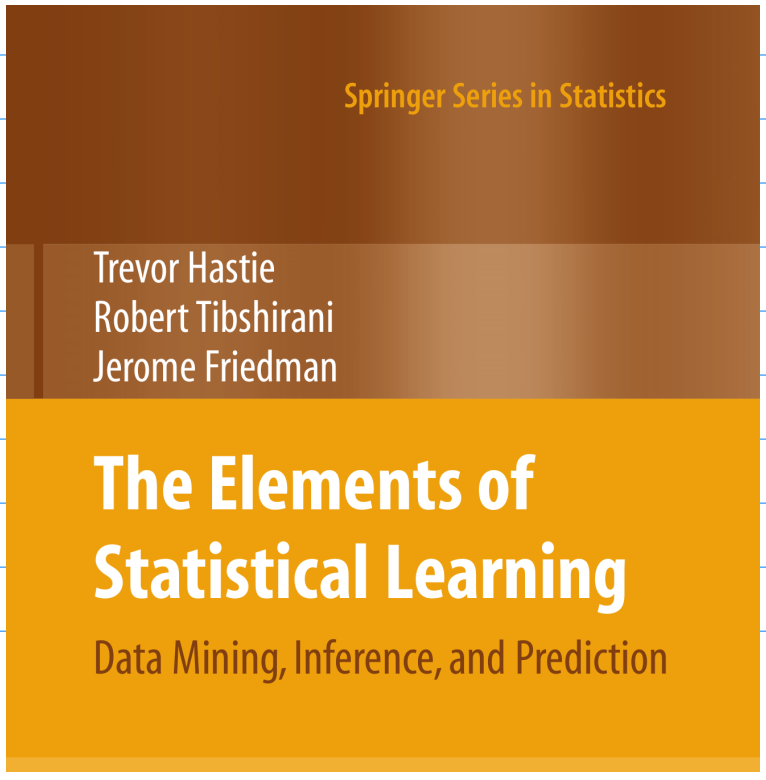
We will now briefly describe Machine Learning (ML) as a broad collection of tools & ideas for analysis of data.

You will see that many of the topics we have seen so far will reappear in ML under different names. These connections are not by coincidence & are due to share mathematical underpinnings among these methods.

ML is a large field & a deep treatment of it requires an entire curriculum. We will only see the basics over the next three lectures.

I strongly recommend the following if you would like to know more about ML.

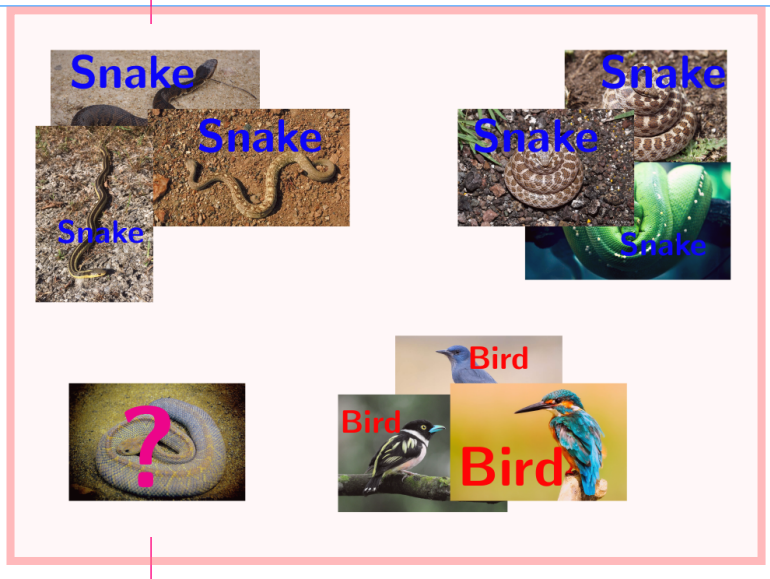
You can download the ebook for free.



12.1 Preliminaries & Terminology

Broadly speaking we consider two categories of problems in ML:

Supervised Learning	Un-Supervised Learning
Goal: Predict/classify data given sotraining dataset	Find meaningful structure in dataset
ex: Regression, classification, function approximation, etc	clustering, dimensionality reduction, Generative modeling



In standard ML terminology we consider a set of points $X = \{x_0, \dots, x_{N-1}\}$ of features/inputs. $x_j \in X$ (images, audio, medical history etc) which for all intents & purposes can be taken as $X \equiv \mathbb{R}^d$. In which case we can think of X as a matrix $X \in \mathbb{R}^{d \times N}$ as we did for PCA.

Associated to these features/inputs we may have a collection of outputs / Responses $Y = \{y(x_0), \dots, y_{N-1}(x_{N-1})\}$ have $y_j \in Y$ that is often thought of as either a Euclidean space $Y \equiv \mathbb{R}^m$ or space of categories ex $Y = \{\text{dogs}, \text{cats}\}$.

We refer to the pair of input & output sets $\{X, Y\}$ as a training dataset.

Goal of supervised Learning:

Given the traing set $\{X, Y\}$ predict $y(x)$ for any new input/feature $x \in X$.

Observe that we have already seen this in the context of function approx. & signal processing.

Consider a function $f:[0,2\pi) \longrightarrow \mathbb{R}$ that is observed on a uniform grid $x_n = n\delta x$ for $n=0,\dots,N-1$ so $X=\{x_0,\dots,x_{N-1}\}$ & $Y=\{f_0,\dots f_{N-1}\}$ (Recall notation $f_n = f(x_n)$).

Then we can approx. f at a new point $x\in[0,2\pi)$ using its DFT,

$$\tilde{f}(x) = \frac{1}{N}\sum_{k=-N/2}^{N/2-1} \tilde{c}_k \exp(ikx)$$

($\tilde{c}_k$: DFT of Y)

*Indeed Supervised learning is deeply connected to function approx.

In the setting of unsupervised Learning we only have access to the features X & no output Y is specified.

Goal of unsupervised Learning:
Find meaningful structure/clusters/reduced dimension in the dataset X.

In this light PCA/POD/DMD methods can be viewed as unsupervised learning methods.

12.2 Intro to Supervised Learning (SL)

For the next few weeks we will primarily focus on supervised learning, post-poning unsupervised learning until the second half of the course.

The well-established paradigm in SL is the functional model where we assume there exists a function

$$f^\dagger : X \to Y \quad \text{so that} \quad y_j = f(x_j) + \varepsilon_j$$

where ε_j are some noise that may be in the output or our observation of the $f(x_j)$.

By far the most common assumption is Gaussian additive noise

$$\varepsilon_j \sim N(0, \sigma^2)$$ ← Noise variance

This implies $y_j | x_j \sim N(f^\dagger(x_j), \sigma^2)$

PDF of y for fixed x_j

$$\pi(y_j | x_j) \propto \exp\left(-\frac{1}{2\sigma^2} |f^\dagger(x_j) - y_j|^2\right)$$ ← Likelihood of y given x_j

Proportional to

Repeating the same calculation for the entire vector $Y \in \mathbb{R}^N$ we get

$$Y|X \sim N(f(X), \sigma^2 I)$$

where we used shorthand notation $f(X) = (f(x_0), \dots, f(x_{N-1})) \in \mathbb{R}^N$. Then Y is a multivariate Gaussian with PDF

$$p(Y|X) \propto \exp\left(\frac{-1}{2\sigma^2} \|f(X) - Y\|^2\right)$$

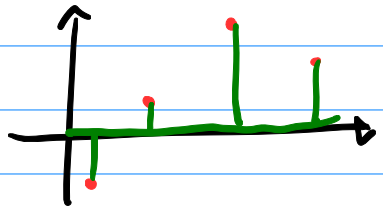
Euclidean norm

Then we can formulate an optimization problem for finding f^+ by maximizing the likelihood of $Y|X$. This is called a maximum Likelihood estimator (MLE)

$$f_{MLE} = \underset{f}{\operatorname{argmin}} \frac{1}{2\sigma^2} \|f(X) - Y\|^2$$

At the moment the MLE problem above is impossible to solve since many functions f exist that will achieve zero loss.

ex $f(\underline{x}) = \begin{cases} y_j & \text{if } \underline{x} = \underline{x}_j \\ 0 & \text{otherwise} \end{cases}$



so we need to make some assumptions on the form of f. A particularly simple model is linear regression

$$f(\underline{x}) = \beta_0 + \sum_{j=1}^{d} \beta_j x_j$$

ie, we assume f is an affine transformation of $\underline{x}$. Then, our MLE takes the form

$$f_{MLE} \equiv \beta_{MLE} = \underset{\beta \in \mathbb{R}^{d+1}}{\operatorname{argmin}} \frac{1}{2\sigma^2} \|A\beta - Y\|^2$$

where $A = \begin{bmatrix} 1 & \underline{x}_0^T \\ \vdots & \vdots \\ 1 & \underline{x}_{N-1}^T \end{bmatrix} = \left[1 \,\middle|\, X^T \right] \in \mathbb{R}^{N \times (d+1)}$

Therefore, MLE is nothing but a least squares solution to the problem $A\beta = Y$.

Typically $N \gg d$ so this system is over-determined.

Solution is given by solving the normal equations,

Differentiate cost function

$$\frac{\partial}{\partial \beta} \left(\frac{1}{2\sigma^2} (A\beta - Y)^T (A\beta - Y) \right) = \frac{1}{\sigma^2} A^T (A\beta - Y)$$

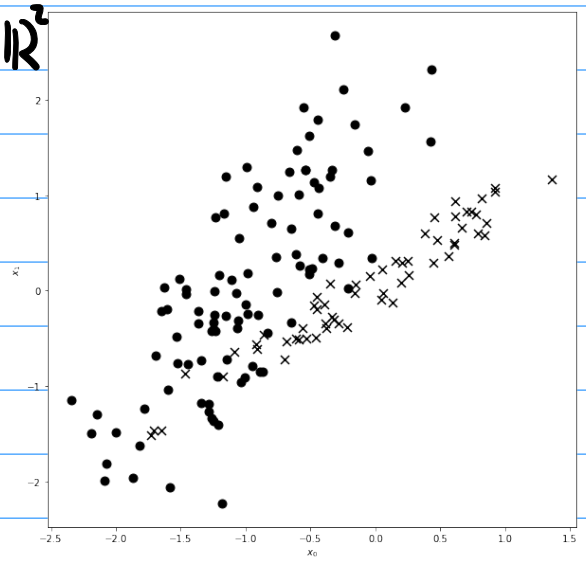
$$\Rightarrow A^T (A\beta_{MLE} - Y) = 0 \Rightarrow \beta_{MLE} = (A^T A)^{-1} A^T Y$$

ex training a classifier to distinguish points in 2D

$\{X, Y\}$ – mixture of points $\underline{x}_j \in \mathbb{R}^2$ with Labels $\{-1, +1\}$.

$X \in \mathbb{R}^{2 \times N}$, $Y \in \{-1, +1\}^N \subset \mathbb{R}^N$

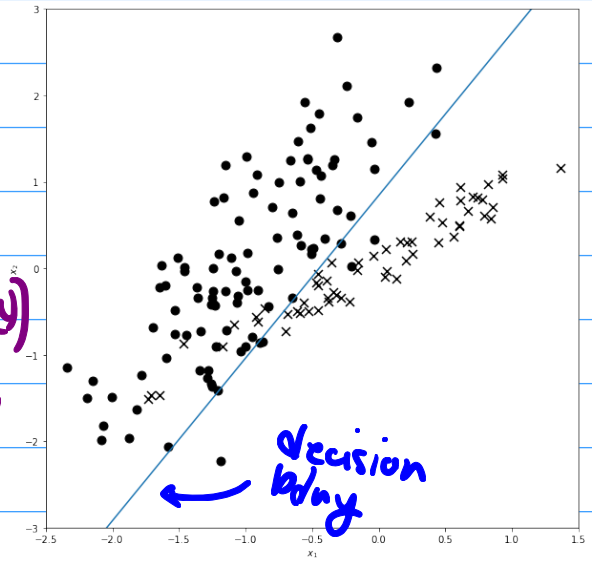
* we treat Y as a real output as opposed to binary.



linear model

$$f(\underline{x}) = \beta_0 + \beta_1 x_0 + \beta_2 x_1$$

we find $\underline{\beta} \in \mathbb{R}^3$ by solving the MLE problem (least squares $A\beta = Y$.



Thus, f_{MLE} is a 3D hyperplane.

We can simply take $\text{sign}(f_{MLE}(\underline{x}))$ as a classifier for any new point $\underline{x} \in \mathbb{R}^2$.

Thus, the set $\{\underline{x} \mid f_{MLE}(\underline{x}) = 0\}$ is precisely our decision boundary! which is the line

$$x_1 = -\frac{\beta_0^{MLE} + \beta_1^{MLE} x_0}{\beta_2^{MLE}}$$