Lecture 1. 01 / 12 (M). 9:30 – 10:45 am. #1.

# Overview of Supervised Learning.

## 1 – Quick Gothrough. : Terminology & Variable Types.

Inputs $\xrightarrow{\text{predict}}$ Outputs

Stat predictors independent vars.

Pattern Recognition. features responses

–

Variable types : quantitative measurements , qualitative vars. , ordered categorical vars.

categorical / discrete. ↑ (qualitative vars.)

Output. Regression classification. Chap 4 ?

## – Two Simple Approaches to prediction.

Least Squares & Nearest Neighbors.

Training set $\mathcal{T}$.

* Linear models and least squares

$$\hat{Y} = \hat{\beta}_0 + \sum_{j=1}^{p} X_j \hat{\beta}_j$$

↑ intercept / bias (in ML).

include $\mathbb{1}$ in X. $\Rightarrow$ $\hat{\beta}_0$ in $\hat{\beta}$

$$\hat{Y} = X^T \hat{\beta}$$

low variance: Decision boundary smooth. Rely heavily on assumption. potentially high bias.

Least squares : pick the coefficients $\beta$ to minimize the "residual sum of squares"

$$RSS(\beta) = \sum_{i=1}^{N} (y_i - x_i^T \beta)^2 .$$ – quadratic function of the parameters.

$$= (\vec{y} - \vec{X}\beta)^T (\vec{y} - \vec{X}\beta)$$

$\vec{y}$: N-vector, $\vec{X}$: N × p.

$\xrightarrow[\text{w.r.t. } \beta]{\text{Differentiate}}$ $\vec{X}^T(\vec{y} - \vec{X}\beta) = 0$

$\Downarrow$ $\vec{X}^T\vec{X}$ is non-singular.

$$\hat{\beta} = (\vec{X}^T\vec{X})^{-1} \vec{X}^T \vec{y} .$$

*. Nearest – Neighbor Methods.

Adaptive (no strigent assumptions). Decision boundary wiggly & unstable. – low bias, high variance.

$N_k(x)$ – neighbourhood of $x$ defined by $k$ closest points $x_i$ in $\mathcal{T}$.

KNN : $$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i .$$

– Effective # of ~~pra~~ para : $N/k$ .

A large subset of the most popular techniques are variants of the two simple procedures #2.

E.g. • Kenerl methods : use weights. decrease smoothly to zero, with distance.
v.s. "0/1" weights in KNN.

• In high-dimensional spaces. the distance kernels are modified to emphasize some vars. more than others.

• Local regression

• Linear models fit to a basis expansion.

• Projection pursuit & Neural network. → consist of sum of non-linearly transformed models.

2. Statistical Decision Theory.

Let $X \in \mathbb{R}^p$. $Y \in \mathbb{R}$. with joint distribution $Pr(X, Y)$.

We seek a function $f(x)$ for predicting $Y$ given values. of $X$.

require ↳ a loss function. $L(Y, f(x))$. for penalizing errors in prediction.

$$L(Y, f(x)) = (Y - f(x))^2 \quad \text{— squared error loss.}$$

⇒ Expected prediction error (EPE) as a criterion for choosing $f$.

$$EPE(f) = E(Y - f(x))^2.$$

$$= \int [y - f(x)]^2 \, Pr(dx, dy).$$

linear regression
$f(x) = x^T \beta$.

$P(X, Y) = P(Y|X) P(X)$.

$$= E_X E_{Y|X} ([Y - f(x)]^2 | X).$$

model-based approach.
solve $\beta$ theoretically
$\beta = [E(XX^T)]^{-1} E(XY)$.

suffice to minimize EPE pointwise.

$$f(x) = \text{argmin}_c \, E_{Y|X} ([Y - c]^2 | X = x).$$

$$\Rightarrow f(x) = E(Y | X = x)$$ i.e., the conditional expectation. aka. the regression function.

★ Thus the best prediction of $Y$ at any point $X = x$ is the conditional mean, when best is measured by average squared error.

—

NN methods. $\hat{f}(x) = Ave(y_i | x_i \in N_k(x))$. — Expectations is approx. by averaging over sample data.

— conditioning at a point
↓ relaxed to
on some region "close" to target point.

under mild regularity. $N, k \to \infty$ s.t. $k/N \to 0$.
$\hat{f}(x) \to E(Y | X = x)$.

$p \uparrow$. rate of convergence $\downarrow$. "curse of dimension"

Both KNN and least squares approx. conditional expectations by average. # 3.

Diff : · Least squares assume $f(x)$ is well approx. by a global linear function.

· k-NN assumes $f(x)$ is well approx. by a locally constant func.

Other Loss function : $L1 : E|Y - f(x)|$. $\Rightarrow$ conditional median $\hat{f}(x) = \text{median}(Y|X=x)$

discontinuities in derivatives, hinder the widespread use.

↳ squared error is analytically convenient, popular.

Categorical varibles :

Output : categorical variable $G$.

Loss function : $L_{K\times K}$ , $K = \text{card}(G)$. → cardinality.

$$L_{K\times K} = \begin{bmatrix} 0 & L(k,l) \\ & \ddots \\ & & 0 \end{bmatrix}$$

$L(k,l) \geq 0$ , zero-one loss function. (most often)

price paid for classifying an obs. belonging to $G_k$ as $G_l$.

$$EPE = E[L(G, \hat{G}(X))].$$

$$= E_X \sum_{k=1}^{K} L[G_k, \hat{G}(X)] Pr(G_k|X)$$

$\Rightarrow$ It suffices to minimize EPE pointwise.

$$\hat{G}(x) = \underset{g\in G}{\text{argmin}} \sum_{k=1}^{K} L(G_k, g) Pr(G_k|X=x).$$

$$\overset{0-1 \text{ loss}}{=} \underset{g\in G}{\text{argmin}} [1 - Pr(g|X=x)]. \Leftrightarrow$$

$\boxed{\text{Bayes classifier}}$

$$\hat{G}(x) = G_k \text{ if } Pr(G_k|X=x) = \max_{g\in G} Pr(g|X=x).$$

* KNN classifier directly approximate this

3. Statistical Models, Supervised Learning & Function Approximation

Goal : Find a useful approximation $\hat{f}(x)$ to $f(x)$. that underlies the predictive relationship between the inputs and outputs.

- Squared error loss → regression function $f(x) = E(Y|X=x)$ for a quantitative response

- Nearest-neighbor methods → direct estimates of the conditional expectation. but fails in high dimension setting (2.5). & when special structure exists. (2.7)

- Other classes of models for $f(x)$.

– Discuss a framework for incorporating them into the prediction problem.

#4.

## △ Statistical Model.

$Y = f(X) + \varepsilon$. $\varepsilon$. random error. $E(\varepsilon) = 0$. ind. of $X$. — Additive error model.

- Input – output pairs $(X, Y)$. not deterministic. ( $Y = f(X)$ ).
  unmeasured variables. through $\varepsilon$. ↳ can be handled by techniques.

- Assumption. $\varepsilon$ iid.
  Simple modifications : to avoid independence assumption, e.g. $Var(Y \mid X = x) = \sigma(x)$

- Quantitative response.
  Other data types → e.g. Generalized linear models.

## △ Machine Learning. – Supervised Learning.

Learn $f$ by example through a " teacher "
⇓
training set.. learning algorithm.

– Function Approximation.

$\{x_i, y_i\}$ $(p+1)$ - dimensional Euclidean space.

$$X_{p\times 1} \xrightarrow{f} Y$$
$$R^p \qquad R.$$

- Many approximations. a set of parameters. $\theta$.
  - linear models $f(x) = x^T\beta$. , $\theta = \beta$.
  - linear basis expansions.
    $$f_\theta(x) = \sum_{k=1}^{K} h_k(x)\,\theta_k.$$
    $h(\cdot)$. set of functions.
    e.g. polynomial
    trigonometric expansions.
    sigmoid. transformation.

∘ Least squares to estimate the parameters $\theta$ in $f_\theta$. ⟺ maximum
by minimizing the residual sum-of-squares. $Pr(Y \mid X, \theta) = N(f_\theta(X), \sigma^2)$.

$$RSS(\theta) = \sum_{i=1}^{N} (y_i - f_\theta(x_i))^2.$$

Let's look at it together (p 31).

∘ A more general principle for estimation : maximum likelihood estimation.

Supp. $y_i$ $i = 1, \cdots, N$ $\sim Pr_\theta(y)$.

$$L(\theta) = \sum_{i=1}^{N} \log Pr_\theta(y_i).$$

- ~~Nearest~~ Nearest - neighboor Methods
( & other local ).

#5.

Face problems : <1> High dimensions → " curse of dimensionality ". More 2.5.

Consider a p- dimension unit hypercube.
Supp. We send out a hypercubical neiborhood. about a target point to capture fraction r of the observations.

⇒. a fraction r of the unit volume. , the expected edge length
$e_p(r) = r^{1/p}$.

$p = 10$ , $r = 0.01$ ~~$e_{10}(r)$~~ $e_{10}(0.01) = 0.63$

$r = 0.1$ $e_{10}(0.1) = 0.80$.

<2>. More structured approaches can make more efficient use of the data.

$RSS(f) = \sum_{i=1}^{N} (y_i - f(x_i))^2$. any. $f(\cdot)$.

- minimize this → infinitely many solutions
if. multiple obs. $x_i$, $y_{il}$ $l=1, \cdots N_i$. → N sufficienty large.

- obtain useful results for finite N. ⇒. restrictions
e.g. parametric representation.
or build into the learning method.

Classes of Restricted Estimators

*. Roughness penalty and Bayesian methods.
~~RRS~~ $PRSS(f;\lambda) = RSS(f) + \lambda J(f)$.
explicitly penalize RSS (f) with a roughness penalty.

*. Kernel methods and local regression.
explicitly providing estimates of the regression function or conditional expectation by specifying the nature of the local neibonhood.

* Basis function and Dictionary methods.