Lecture 3: Statistical Decision Theory (Part II)

Hao Helen Zhang

Outline of This Note

- Part I: Statistics Decision Theory (Classical Statistical Perspective - "Estimation")
  - loss and risk
  - MSE and bias-variance tradeoff
  - Bayes risk and minimax risk
- Part II: Learning Theory for Supervised Learning (Machine Learning Perspective - "Prediction")
  - optimal learner
  - empirical risk minimization
  - restricted estimators

Supervised Learning

Any supervised learning problem has three components:

- Input vector $\boldsymbol{X} \in R^d$.
- Output Y, either discrete or continuous valued
- Probability framework

$$(\boldsymbol{X}, Y) \sim P(\boldsymbol{X}, Y).$$

The goal is to estimate the relationship between $\boldsymbol{X}$ and Y, described by $f(\boldsymbol{X})$, for future prediction and decision-making.

- For regression, $f : \mathcal{R}^d \to R$
- For K-class classification, $f : \mathcal{R}^d \to \{1, \cdots, K\}$

Learning Loss Function

Similar to the learning theory, we use a *learning loss* function L

- to measure the discrepancy Y and $f(\boldsymbol{X})$
- to penalize the errors for predicting Y.

Examples:

- squared error loss (used in mean regression)

$$L(Y, f(\boldsymbol{X})) = (Y - f(\boldsymbol{X}))^2$$

- absolute error loss (used in median regression)

$$L(Y, f(\boldsymbol{X})) = |Y - f(\boldsymbol{X})|$$

- 0-1 loss function (used in binary classification)

$$L(Y, f(\boldsymbol{X})) = I(Y \neq f(\boldsymbol{X}))$$

Risk, Expected Prediction Error (EPE)

The risk of f is

$$R(f) = E_{\boldsymbol{X},Y} L(Y, f(\boldsymbol{X})) = \int L(y, f(\boldsymbol{x})) dP(\boldsymbol{x}, y)$$

In machine learning, $R(f)$ is called Expected Prediction Error (EPE).

- For squared error loss,

$$R(f) = EPE(f) = E_{\boldsymbol{X},Y}[Y - f(\boldsymbol{X})]^2$$

- For 0-1 loss

$$R(f) = EPE(f) = E_{\boldsymbol{X},Y} I(Y \neq f(\boldsymbol{X})) = Pr(Y \neq f(X)),$$

which is the probability of making a mistake.

Optimal Learner

We seek f which minimizes the average (long-term) loss over the entire population.

Optimal Learner: The minimizer of $R(f)$ is

$$f^* = \arg\min_f R(f).$$

- The solution f^* depends on the loss L.
- f^* is not achievable without knowing $P(\boldsymbol{x}, y)$ or the entire population

Bayes Risk: The optimal risk, or risk of the optimal learner

$$R(f^*) = E_{\boldsymbol{X},Y}(Y - f^*(\boldsymbol{X}))^2.$$

How to Find The Optimal Learner

Note that

$$\begin{aligned} R(f) &= E_{\boldsymbol{X},Y} L(Y, f(\boldsymbol{X})) = \int L(y, f(\boldsymbol{x})) dP(\boldsymbol{x}, y) \\ &= E_{\boldsymbol{X}} \left\{ E_{Y|\boldsymbol{X}} L(Y, f(\boldsymbol{X})) \right\} \end{aligned}$$

One can do the following,

- for each fixed $\boldsymbol{x}$, find $f^*(\boldsymbol{x})$ by solving

$$f^*(\boldsymbol{x}) = \arg\min_c E_{Y|\boldsymbol{X}=\boldsymbol{x}} L(Y, c)$$

Example 1: Regression with Squared Error Loss

For regression, consider $L(Y, f(\boldsymbol{X})) = (Y - f(\boldsymbol{X}))^2$. The risk is

$$R(f) = E_{\boldsymbol{X},Y}(Y - f(\boldsymbol{X}))^2 = E_{\boldsymbol{X}} E_{Y|\boldsymbol{X}}((Y - f(\boldsymbol{X}))^2|\boldsymbol{X}).$$

- For each fixed $\boldsymbol{x}$, the best learner solves the problem

$$\min_c E_{Y|\boldsymbol{X}}[(Y - c)^2|\boldsymbol{X} = \boldsymbol{x}].$$

  The solution is $E(Y|\boldsymbol{X} = \boldsymbol{x})$, so

$$f^*(\boldsymbol{x}) = E(Y|\boldsymbol{X} = \boldsymbol{x}) = \int y dP(y|\boldsymbol{x}).$$

- So the optimal leaner is $\underline{f^*(\boldsymbol{X}) = E(Y|\boldsymbol{X})}$.

Decomposition of EPE: Under Squared Error Loss

For any learner f, its EPE under the squared error loss is decomposed as

$$\begin{aligned}
EPE(f) &= E_{\boldsymbol{X},Y}[Y - f(\boldsymbol{X})]^2 = E_{\boldsymbol{X}} E_{Y|\boldsymbol{X}}([Y - f(\boldsymbol{X})]^2|\boldsymbol{X}) \\
&= E_{\boldsymbol{X}} E_{Y|\boldsymbol{X}}([Y - E(Y|\boldsymbol{X}) + E(Y|\boldsymbol{X}) - f(\boldsymbol{X})]^2|\boldsymbol{X}\} \\
&= E_{\boldsymbol{X}} E_{Y|\boldsymbol{X}}([Y - E(Y|\boldsymbol{X})]^2|\boldsymbol{X}) + E_{\boldsymbol{X}}[f(\boldsymbol{X}) - E(Y|\boldsymbol{X})]^2 \\
&= E_{\boldsymbol{X},Y}[Y - f^*(\boldsymbol{X})]^2 + E_{\boldsymbol{X}}[f(\boldsymbol{X}) - f^*(\boldsymbol{X})]^2 \\
&= EPE(f^*) + \text{MSE} \\
&= \text{Bayes Risk} + \text{MSE}
\end{aligned}$$

The Bayes risk (gold standard) gives a lower bound for any risk.

Example 2: Classification with 0-1 Loss

In binary classification, assume $Y \in \{0, 1\}$. The learning rule $f : R^d \to \{0, 1\}$.

Using 0-1 loss

$$L(Y, f(\boldsymbol{X})) = I(Y \neq f(\boldsymbol{X})),$$

the expected prediction error is

$$\begin{aligned} EPE(f) &= EI(Y \neq f(\boldsymbol{X})) \\ &= \Pr(Y \neq f(\boldsymbol{X})) \\ &= E_{\boldsymbol{X}}\left[P(Y = 1|\boldsymbol{X})I(f(\boldsymbol{X}) = 0) + P(Y = 0|\boldsymbol{X})I(f(\boldsymbol{X}) = 1)\right] \end{aligned}$$

Bayes Classification Rule for Binary Problems

For each fixed $\boldsymbol{x}$, the minimizer of $EPE(f)$ is given by

$$f^*(\boldsymbol{x}) = \begin{cases} 1 & \text{if} \quad P(Y=1|\boldsymbol{X}=\boldsymbol{x}) > P(Y=0|\boldsymbol{X}=\boldsymbol{x}) \\ 0 & \text{if} \quad P(Y=1|\boldsymbol{X}=\boldsymbol{x}) < P(Y=0|\boldsymbol{X}=\boldsymbol{x}) \end{cases}$$

It is called the *Bayes classifier*, denoted by ϕ_B.

The Bayes rule can be written as

$$\phi_B(\boldsymbol{x}) = f^*(\boldsymbol{x}) = I[P(Y=1|\boldsymbol{X}=\boldsymbol{x}) > 0.5].$$

- To obtain the Bayes rule, we need to know $P(Y=1|\mathbf{X}=\mathbf{x})$, or whether $P(Y=1|\mathbf{X}=\mathbf{x}=\mathbf{x})$ is larger than 0.5 or not.

Probability Framework

Assume the training set

$$(\boldsymbol{X}_1, Y_1), ..., (\boldsymbol{X}_n, Y_n) \sim \text{ i.i.d } P(\boldsymbol{X}, Y).$$

We aim to find the *best* function (optimal solution) from the model class $\mathcal{F}$ for decision making.

- However, the optimal learner f^* is not attainable without knowledge of $P(x, y)$.
- The training set $(x_1, y_1), ..., (x_n, y_n)$ generated from $P(x, y)$ carry information about $P(x, y)$
- We approximate the integral over $P(x, y)$ by the average over the training samples.

Empirical Risk Minimization

Key Idea: We can not compute $R(f) = E_{\boldsymbol{X},Y} L(Y, f(\boldsymbol{X}))$, but can compute its *empirical* version using the data!

Empirical Risk: also known as the *training error*

$$R_{\text{emp}}(f) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)).$$

- A reasonable estimator f should be close to f^*, or converge to f^* when more data are collected.

Using law of large numbers, $R_{\text{emp}}(f) \to EPE(f)$ as $n \to \infty$.

Learning Based on ERM

We construct the estimator of f^* by

$$\hat{f} = \arg\min_f R_{\text{emp}}(f).$$

This principle is called Empirical Risk Minimization. (ERM)

For regression,

- the empirical risk is known as Residual Sum of Squares (RSS)

$$R_{\text{emp}}(f) = RSS(f) = \sum_{i=1}^{n} [y_i - f(x_i)]^2.$$

- ERM gives the least mean squares:

$$\hat{f}^{ols} = \arg\min_f RSS(f) = \arg\min_f \sum_{i=1}^{n} [y_i - f(x_i)]^2.$$

Issues with ERM

Minimizing $R_{\text{emp}}(f)$ over all functions is not proper.

- Any function passing through all $(\boldsymbol{x}_i, y_i)$ has $RSS = 0$. Example:

$$f_1(\boldsymbol{x}) = \begin{cases} y_i & \text{if } \boldsymbol{x} = \boldsymbol{x}_i \text{ for some } i = 1, \cdots, n \\ 1 & \text{otherwise.} \end{cases}$$

  It is easy to check that $RSS(f_1) = 0$.

- However, it does not amount to any form of learning. For the future points not equal to $\boldsymbol{x}_i$, it will always predict $y = 1$. This phenomenon is called "overfitting".
- Overfitting is undesired, since a small $R_{\text{emp}}(f)$ does not imply a small $R(f) = \int L(y, f(\boldsymbol{x}))dP(\boldsymbol{x}, y)$.

Class of Restricted Estimators

We need to restrict the estimation process within a set of functions $\mathcal{F}$, to control *complexity* of functions and hence avoid over-fitting.

- Choose a simple model class $\mathcal{F}$.

$$\hat{f}_{ols} = \arg\min_{f} RSS(f \in \mathcal{F}) = \arg\min_{f \in \mathcal{F}} \sum_{i=1}^{n} [y_i - f(x_i)]^2.$$

- Choose a large model class $\mathcal{F}$, but use a roughness penalty to control model complexity: Find $f \in \mathcal{F}$ to minimize

$$\hat{f} = \arg\min_{f \in \mathcal{F}} \text{Penalized RSS} \equiv RSS(f) + \lambda J(f).$$

The solution is called **regularized empirical risk minimizer**.

How should we choose the model $\mathcal{F}$? Simple one or complex one?

Parametric Learners

Some model classes can be parametrized as

$$\mathcal{F} = \{f(\boldsymbol{X}; \alpha), \alpha \in \Lambda\},$$

where α is the index parameter for the class.

Parametric Model Class: The form of f is known except for **finitely** many unknown parameters

- Linear Models: linear regression, linear SVM, LDA

$$\begin{aligned} f(\boldsymbol{X}; \beta) &= \beta_0 + \boldsymbol{X}'\beta_1, \quad \beta_1 \in R^d, \\ f(\boldsymbol{X}; \beta) &= \beta_0 + sin(X_1)\beta_1 + \beta_2 X_2^2, \quad X \in R^2 \end{aligned}$$

- Nonlinear models

$$\begin{aligned} f(\boldsymbol{x}; \beta) &= \beta_1 x^{\beta_2} \\ f(\boldsymbol{X}; \beta) &= \beta_1 + \beta_2 \exp^{\beta_3 X^{\beta_4}} \end{aligned}$$

Advantages of Linear Models

Why are linear models in wide use?

- If linear assumptions are correct, it is more efficient than nonparametric fits.
- convenient and easy to fit
- easy to interpret
- providing the first-order Taylor approximation to $f(\mathbf{X})$

Example: In the linear regression, $\mathcal{F} = \{\text{all linear functions of } \mathbf{x}\}$.

The ERM is the same as solving the **ordinary least squares**,

$$(\hat{\beta}_0^{ols}, \hat{\boldsymbol{\beta}}^{ols}) = \arg \min_{(\beta_0, \boldsymbol{\beta})} \sum_{i=1}^{n} [y_i - \beta_0 - \mathbf{x}_i' \boldsymbol{\beta}]^2.$$

Drawback of Parametric Models

We never know whether linear assumptions are proper or not.

- Not flexible in practice, having all orders of derivatives and the same function form everywhere
- Individual observations have large influences on remote parts of the curve
- The polynomial degree can not be controlled continuously

More Flexible Learners

Nonparametric Model Class $\mathcal{F}$:
the function form f is unspecified, and the parameter set could be infinitely dimensional.

- $\mathcal{F} = \{\text{all continuous functions of } \boldsymbol{x}\}$.
- $\mathcal{F} =$ the second-order Sobolev space over $[0, 1]$.
- $\mathcal{F}$ is the space spanned by a set of Basis functions (dictionary functions)

$$\mathcal{F} = \{f : f_\theta(\boldsymbol{x}) = \sum_{m=1}^{\infty} \theta_m h_m(\boldsymbol{x})\}.$$

- Kernel methods and local regression:

  $\mathcal{F}$ contains all piece-wise constant functions,
  or, $\mathcal{F}$ contains all piece-wise linear functions.

Example of Nonparametric Methods

For regression problems,

- splines, wavelets, kernel estimators, locally weighted polynomial regression, GAM
- projection pursuit, regression trees, k-nearest neighbors

For classification problems,

- kernel support vector machines, k-nearest neighbors
- classification trees ...

For density estimation,

- histogram, kernel density estimation

Semiparametric Models: a compromise between linear models and nonparametric models

- partially linear models

Nonparametric Models

Motivation: The underlying regression function is so complicated that no reasonable parametric models would be adequate.
Advantages:

- **NOT** assume any specific form.
- Assume less, thus discover more
- infinite parameters: not no parameters
- Local more relying on the neighbors
- Outliers and influential points do not affect the results

Let the data speak for themselves and show us the appropriate functional form!

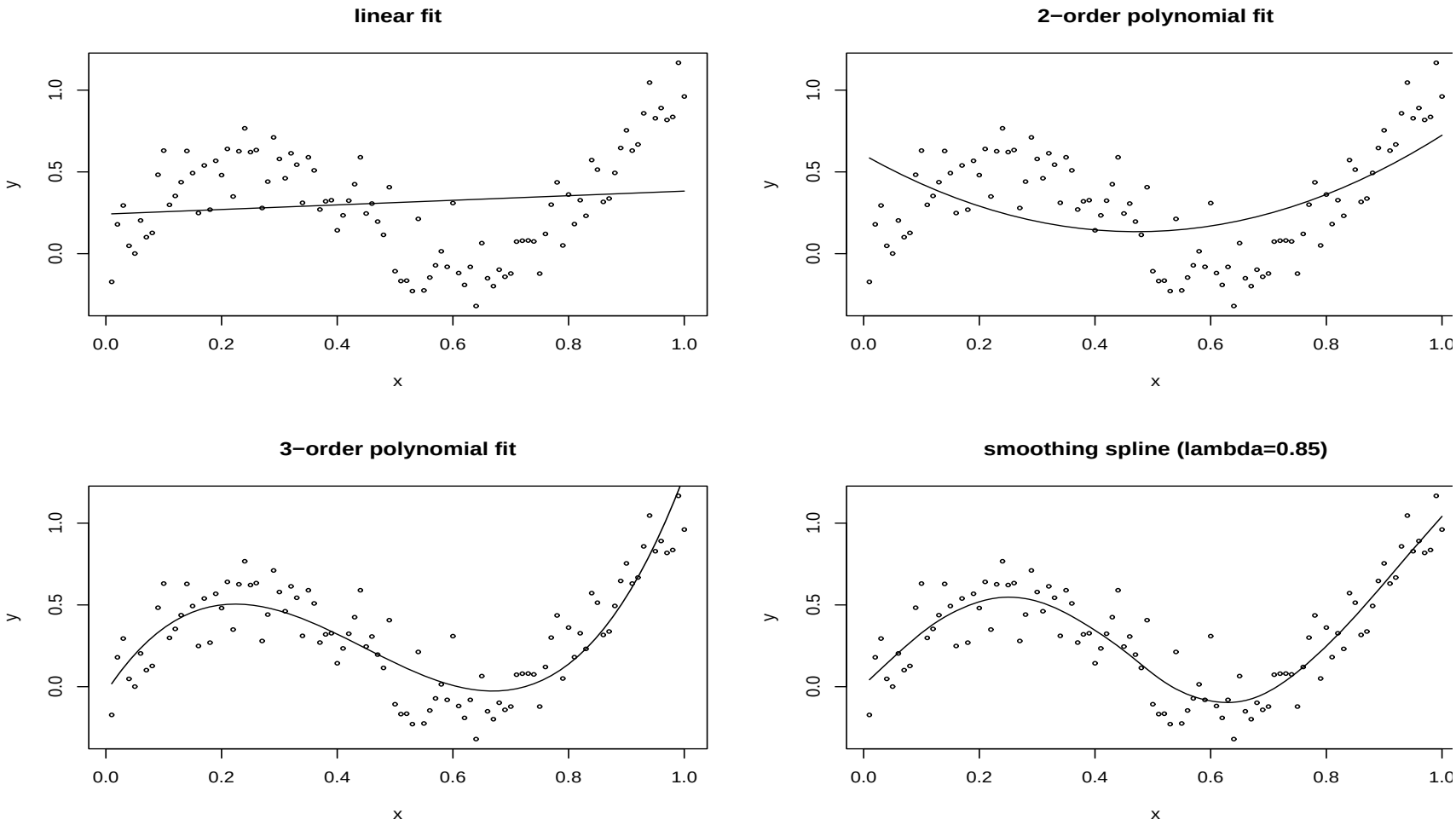
Price we pay: more computational effort, lower efficiency and slower convergence rate (if linear models happen to be reasonable)

Nonparametric vs Parametric Models

They are not mutually exclusive competitors

- A nonparametric regression estimate may suggest a simple parametric model
- In practice, parametric models are preferred for simplicity
- Linear regression line is *infinitely smooth* fit

method	# parameters	estimate	outliers
parametric	finite	global	sensitive
nonparametric	infinite	local	robust


linear fit
2−order polynomial fit
3−order polynomial fit
smoothing spline (lambda=0.85)
x
y
0.0
0.2
0.4
0.6
0.8
1.0
0.5

Interpretation of Risk

Let $\mathcal{F}$ denote the restricted model space.

- Denote $f^* = \arg\min EPE(f)$, the optimal learner. The minimization is taken over all measurable functions.
- Denote $\tilde{f} = \arg\min_{\mathcal{F}} EPE(f)$, the best learner in the restricted space $\mathcal{F}$.
- Denote $\hat{f} = \arg\min_{\mathcal{F}} R_{\text{emp}}(f)$, the learner based on limited data in the restricted space $\mathcal{F}$.

$$\begin{aligned} & EPE(\hat{f}) - EPE(f^*) \\ = \; & [EPE(\hat{f}) - EPE(\tilde{f})] + [EPE(\tilde{f}) - EPE(f^*)] \\ = \; & \text{estimation error} + \text{approximation error}. \end{aligned}$$

Estimation Error and Approximation Error

- The gap $EPE(\hat{f}) - EPE(\tilde{f})$ is called the *estimation* error, which is due to the limited training data.
- The gap $EPE(\tilde{f}) - EPE(f^*)$ is called the *approximation* error. It is incurred from approximating f^* with a restricted model space or via regularization, since it is possible that $f^* \notin \mathcal{F}$.

This decomposition is similar to the bias-variance trade-off, with

- estimation error playing the role of variance
- approximation error playing the role of bias.