Decision Theory

Matthew S. Shotwell, Ph.D.

Department of Biostatistics
Vanderbilt University School of Medicine
Nashville, TN, USA

January 18, 2024

Introduction

- need mechanism to quantify "goodness" of predictions
- need to use context: what is the purpose of making predictions

Loss function

- seek a model that predicts Y from X, denote $f(X)$
- how do we find $f(X)$?
- first need to specify what 'good' and 'bad' predictions look like
- encode this in a function that compares Y and $f(X)$
- a 'loss function' $L(Y, f(X))$

Minimizing loss

- $f(X)$ can be selected by minimizing the average loss, or 'expected loss' or 'expected prediction error': $EPE(f) = E[L(Y, f(X))]$
- the average, or 'expectation' is taken over the joint distribution of X and Y: $Pr(X, Y)$

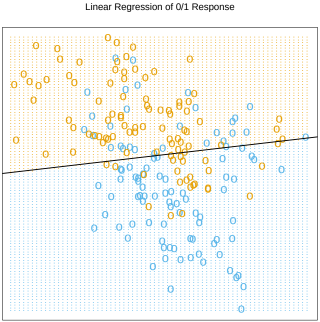


FIGURE 2.1. *A classification example in two dimensions. The classes are coded as a binary variable (*BLUE *= 0,* ORANGE *= 1), and then fit by linear regression. The line is the decision boundary defined by $x^T\hat{\beta} = 0.5$. The orange shaded region denotes that part of input space classified as* ORANGE*, while the blue region is classified as* BLUE.

Purposes of EPE

The EPE has two purposes:

1. To identify what $f(X)$ should look like: "decision theory" (today's focus)
2. To evaluate the predictive quality of a fitted model: The EPE can be approximated using testing data:

$$EPE(f) = E[L(Y, f(X))] \approx \frac{1}{N} \sum_{i=1}^{N} L(y_i, f(x_i))$$

Types of loss for predicting quantitative outputs

- squared-error (a.k.a L_2) loss:

$$L(Y, f(X)) = (Y - f(X))^2$$

- absolute-error loss (a.k.a. L_1) loss:

$$L(Y, f(X)) = |Y - f(X)|$$

Squared-error loss

- loss:
$$L(Y, f(X)) = (Y - f(X))^2$$
- expected loss:
$$\begin{aligned} EPE(f) &= E_{X,Y}[(Y - f(X))^2] \\ &= E_X[E_{Y|X}[(Y - f(X))^2]] \end{aligned}$$
- predictor:
$$\begin{aligned} f(X) &= \arg\min_C E_{Y|X}[(Y - C)^2 | X] \\ &= \mu_{Y|X} \end{aligned}$$
- $\hat{Y} = f(X) = \mu_{Y|X} = \hat{E}_{Y|X}[Y] =$ mean of Y given X

Squared-error loss

- If we specify squared error loss, the best predictor is always $\hat{Y} = f(X) =$ mean of Y given X.
- This is based on decision theory, which helps us determine what our predictor should look like, given the loss function we specify.
- We don't need data for this.

LS and NN predictors minimize squared error loss

- LS:

$$\hat{Y} = \hat{f}(X) = \hat{E}_{Y|X}[Y] = X\hat{\beta}$$

- NN:

$$\hat{Y} = \hat{f}(X) = \hat{E}_{Y|X}[Y] = \frac{1}{K} \sum_{x_i \in N_K(X)} y_i$$

Absolute error loss

- loss:
$$L(Y, f(X)) = |Y - f(X)|$$
- expected loss:
$$\begin{aligned} EPE(f) &= E_{X,Y}[|Y - f(X)|] \\ &= E_X[E_{Y|X}[|Y - f(X)|]] \end{aligned}$$
- predictor:
$$\begin{aligned} f(X) &= \arg\min_C E_{Y|X}[|Y - C|] \\ &= \text{median}(Y|X) \end{aligned}$$
- $\hat{Y} = f(X) = \text{median}(Y|X) =$ median of Y given X

Symmetric vs. Asymmetric loss

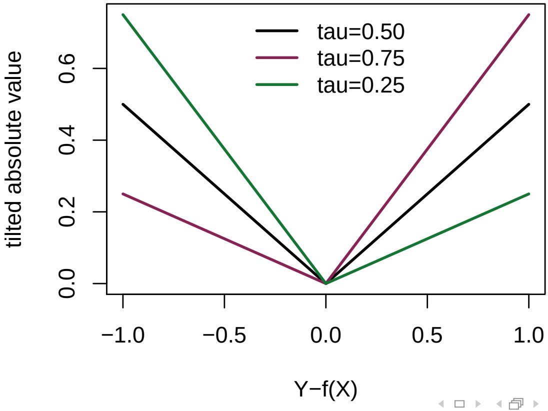
- Both L_1 and L_2 loss are symmetric: if $y = 0$ the loss is same whether $f(x) = 1$ or $f(x) = -1$
- Are certain types of bad predictions worse than others?

Symmetric vs. Asymmetric loss

- Example: Suppose my Tesla is using video to predict the distance between the car an a road barrier. If my prediction is too big, I may hit the barrier, but if my prediction is too small it may not matter much. May need an asymmetric loss function for making predictions.

Tilted absolute loss

$$L(Y, f(X), \tau) = \begin{cases} \tau(Y - f(X)) & (Y - f(X)) > 0 \\ (\tau - 1)(Y - f(X)) & (Y - f(X)) \leq 0 \end{cases}$$



Tilted absolute loss

- $f(X)$ is median (50th percentile) of Y given X for absolute loss
- $f(X)$ is $\tau \times$ 100 percentile of Y given X for tilted absolute loss

Discrete loss

- what if we are predicting a qualitative outcome?
- need different kind of loss function
- "discrete loss"

Discrete loss

- loss:

$$L(G, \hat{G}(X)) = \begin{bmatrix} 0 & l_{12} & \dots & l_{1K} \\ l_{21} & 0 & \dots & l_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ l_{K1} & l_{K2} & \cdots & 0 \end{bmatrix}$$

- expected loss:

$$\begin{aligned} EPE(\hat{G}) &= E_{X,G}[L(G, \hat{G}(X))] \\ &= E_X[E_{G|X}[L(G, \hat{G}(X))]] \\ &= E_X[\sum_{k=1}^{K} L(G_k, \hat{G}(X)) Pr(G = G_k|X)] \end{aligned}$$

Discrete loss

- loss:

$$L(G, \hat{G}(X)) = \begin{bmatrix} 0 & l_{12} & \dots & l_{1K} \\ l_{21} & 0 & \dots & l_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ l_{K1} & l_{K2} & \cdots & 0 \end{bmatrix}$$

- expected loss:

$$\begin{aligned} EPE(\hat{G}) &= E_{X,G}[L(G, \hat{G}(X))] \\ &= E_X[E_{G|X}[L(G, \hat{G}(X))]] \\ &= E_X[\sum_{k=1}^{K} L(G_k, \hat{G}(X))Pr(G = G_k|X)] \end{aligned}$$

Zero-one loss

- loss: $l_{ij} = 1$ for $i \neq j$ and $l_{ij} = 0$ if $i = j$
- predictor:

$$\begin{aligned}
\hat{G}(X) &= \arg\min_C \sum_{k=1}^{K} L(G_k, C) Pr(G = G_k|X) \\
&= \arg\min_C \left[1 - Pr(G = C|X)\right] \\
&= \arg\max_C \left[Pr(G = C|X)\right]
\end{aligned}$$

- $\hat{G}(X)$ is the class with highest probability given X
- $\hat{G}(X)$ is called the 'Bayes classifier'
- misclassification rate estimates EPE for zero-one loss

Other discrete loss

Several types of loss are based on target coding of categories; they provide different types of penalties for discrepancies between targets Y and $\hat{Y} = f(X) = Pr(Y = 1|X)$. Can also used regression loss (e.g., L_1 and L_2) on the target coded categories (that's what we did with the least-squares classifier).

- Cross-entropy loss; based on multinomial likelihood function
- Hinge loss; SVM classifier uses this
- Exponential loss; AdaBoost uses this