Classification and Decision Theory

The classification problem

In this lecture we turn from regression to classification, in which the goal is to estimate a function $f : X \rightarrow Y$ where $Y$ has finitely many elements (usually just a handful).

**Fish Classification.** Recall the example of fish classification where the output space is $Y = \{\text{Salmon, Bass}\}$ and the input space is $X = \mathbb{R}^3$, where the three components of a feature point in $X$ might correspond to the length, weight, and skin brightness of a fish, for example.

**Spam Detection.** Another example is spam detection. In this case the output label set is $Y = \{\text{spam, not spam}\}$ and the input space is $X = \{\text{emails represented as bags of words}\}$. The bag of words model is commonly used to represent text documents. It collapses an entire document down to a vector of word counts. For example, the short document

"the pacific ocean is the best ocean"

would be represented as

```
the : 2
pacific : 1
ocean : 2
is : 1
best : 1
```

More generally, a feature point $x \in X$ is of the form $x = (c_1, c_2, \ldots, c_k)$ where $c_i$ is the number of times word $i$ appears in the email.

These are examples of binary classification problems, but the number of classes can be greater than two.
News Topic Classification. One might wish to create a news article classifier, in which case the input and output spaces could be

\[ Y = \{ \text{politics, science, economics, arts, gossip} \} \]
\[ X = \{ \text{news articles represented as bags of words} \} \]

Bayesian decision theory

Decision theory provides us with a framework for making optimal decisions in the face of uncertainty. Clearly this is a subject worth exploring if we are to build good classifiers.

Assume there is a distribution (or density) \( p(x, y) \) over the product space \( X \times Y \). We introduce a loss function \( L : Y \times Y \to \mathbb{R} \) that will measure the penalty of misclassifications. \( L(y, \hat{y}) \) is the loss incurred for predicting \( \hat{y} \) when the true label is \( y \). We define the 0, 1-loss as

\[
L(y, \hat{y}) = \begin{cases} 
1 & y \neq \hat{y} \\
0 & \text{otherwise.}
\end{cases}
\]

Minimizing the 0, 1-loss is equivalent to minimizing the overall misclassification rate. 0, 1-loss is an example of a symmetric loss function: all errors are penalized equally. In certain applications, asymmetric loss functions are more appropriate. Consider a finger print scanner used to grant entry to the FBI headquarters. There are two types of errors such a scanner could make: it could incorrectly deny entry to an authorized agent, or it could incorrectly grant entry to a random person on the streets. Clearly the latter error should be penalized more heavily, since such a mistake could have dire consequences.

Once we have chosen an appropriate loss function, the natural next step is to minimize the loss. In particular, we seek to minimize the expected loss with respect to the probability distribution \( p(x, y) \) mentioned above.

Definition: We say \( f : X \to Y \) is a Bayes optimal classifier if \( f \) minimizes \( E[L(y, f(x))] \) where \( (x, y) \sim p(x, y) \).
The expected 0,1-loss is precisely the probability of making a mistake, i.e.

\[ E[L(y, f(x))] = P(y \neq f(x)) = \sum_{x \in X} \sum_{y : y \neq f(x)} P(x, y) \]
\[ = \sum_{x \in X} \sum_{y : y \neq f(x)} P(y|x)P(x) \]
\[ = \sum_{x \in X} P(x) \sum_{y : y \neq f(x)} P(y|x) \]
\[ = \sum_{x \in X} P(x)[1 - P(f(x)|x)] \]

Recall that the goal is to choose \( f \) so as to minimize this sum. Thus, for each \( x \), we should assign \( x \) to the class \( f(x) = \text{argmax}_y P(y|x) \), i.e. the most probable value for \( y \) conditional on \( x \). This is the the Bayes optimum classifier.

If this discussion seemed a little abstract, we recommend consulting Section 1.5.1 of the textbook for a more concrete example. Also, feel free to ask about this at TA hours!

\textit{Back to the Fish Example.} Suppose instead that \( Y = \{ \text{Bass, Salmon} \} \) and \( X = \mathbb{R} \) (the length of the fish). Then

\[ p(x, y) = p(y)p(x|y) \]

where \( p(y) \) is the frequency of each fish type in the river and \( p(x|y) \) is the distribution of the lengths conditional on a specific type of fish. Assume for now that \( p(x|y) \) is normal with mean and variance that depend on the fish type. In particular, let \( \mu_S \) and \( \sigma_S \) denote the mean and variance for salmon, and \( \mu_B \) and \( \sigma_B \) denote the mean and variance for sea bass. Further, assume that \( \sigma_S = \sigma_B \).

Let’s find the Bayes optimum classifier using the 0,1-loss function. Using Bayes’ rule, we have

\[ f(x) = \text{argmax}_y p(y|x) = \text{argmax}_y \frac{p(x|y)p(y)}{p(x)} = \text{argmax}_y p(x|y)p(y). \]

We say \( f(x) = \text{salmon} \) if

\[ P(x|\text{salmon})P(\text{salmon}) > P(x|\text{bass})P(\text{bass}). \]

For the sake of simplicity, assume \( P(\text{salmon}) = P(\text{bass}) \). Then the above reduces to guessing salmon whenever

\[ P(x|\text{salmon}) > P(x|\text{bass}). \]
To find the region of length over which this constraint holds, we solve for $x$ below

$$\frac{1}{\sqrt{2\pi\sigma_S^2}} \exp\left\{ -\frac{(x - \mu_S)^2}{2\sigma_S^2} \right\} > \frac{1}{\sqrt{2\pi\sigma_B^2}} \exp\left\{ -\frac{(x - \mu_B)^2}{2\sigma_B^2} \right\}.$$

Taking the log of both sides and solving for $x$, we find a threshold $t$ such that when $x < t$ we guess one category and when $x > t$ we guess the other.

Now suppose that $X = \mathbb{R}^2$ (length, weight). Then $p(x|y)$ is a bivariate normal distribution, which is a special case of the multivariate normal distribution. A multivariate normal distribution in $\mathbb{R}^d$ is denoted $\mathcal{N}(\mu, \Sigma)$, where $\mu$ is a $d$-dimensional vector and $\Sigma$ is a $d \times d$ matrix. The pdf $f$ of a multivariate normal distribution is defined by

$$f(x) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp\left\{ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right\},$$

where $|\Sigma|$ denotes the determinant of $\Sigma$. For $d = 2$, the decision boundary is the curve at which the Gaussian surfaces intersect. 

\qed