Naive Bayes Classifiers

Classification problems

1. discriminant function

   These approaches map each input directly onto a class label and probability plays no rule.

   ex. $f(x) = \mathbf{w}^T\mathbf{x} + w_0$

2. discriminative model

   These models solve the **posterior** $P(C_k \mid \mathbf{x})$ directly and then assign each new $\mathbf{x}$ to a class using a suitable loss function or other decision function.

   ex. Logistic regression

3. generative model

   These models first determine $P(\mathbf{x} \mid C_k)$ and $P(C_k)$ for each class individually. Then solve for $P(C_k \mid \mathbf{x})$. Afterwards, a decision function determines class membership. The generative methods model distribution of both input and output.

   ex. Naive Bayes

NBC

Input: $\mathbf{x} \in \{1, \ldots, K\}^D$

By applying Bayes rule to a generative classifier, we can classify a feature vector $\mathbf{x}$ of the form:

$p(y = c \mid \mathbf{x}, \theta) \propto p(\mathbf{x} \mid y = c, \theta) p(y = c \mid \theta)$

The key to using such models is specifying a suitable form for the class-conditional density $p(\mathbf{x} \mid y = c)$.

Assumption: the features are conditionally independent given the class label.

$p(\mathbf{x} \mid y = c, \theta) = \prod_{j=1}^{D} p(x_j \mid y = c, \theta_{jc})$

(Notice: We call the model "naive" since we do not always expect the features to be independent, even conditional on the class label. However, even if the assumption is not true, NBC still works well. The reason is that the model is quite simple and relatively immune to overfitting.)

The form of class-conditional density depends on the type of each feature.

- Real-valued features

  $p(\mathbf{x} \mid y = c, \theta) = \prod_{j=1}^{D} \mathcal{N}(x_j \mid \mu_{jc}, \sigma_{jc}^2) = \prod_{j=1}^{D} \{ \frac{1}{\sigma_{jc}\sqrt{2\pi}} \exp^{-\frac{(x_j - \mu_{jc})^2}{2\sigma_{jc}^2}} \}$

- Binary features ($x_j \in \{0, 1\}$)

  $p(\mathbf{x} \mid y = c, \theta) = \prod_{j=1}^{D} \mathrm{Ber}(x_j \mid p_{jc}) = \prod_{j=1}^{D} \{ p_{jc}^{x_j} (1 - p_{jc})^{1 - x_j} \}$

- Categorical features ($x_j \in \{1, \ldots, K\}$)

  $p(\mathbf{x} \mid y = c, \theta) = \prod_{j=1}^{D} \mathrm{Cat}(x_j \mid \mu_{jc})$ where μ_{jc} is a histogram over the possible values for x_j in class c.

MLE for NBC

The probability for a single data case:

$$p(\mathbf{x}_i, y_i \mid \theta) = p(y_i \mid \pi) \prod_j p(x_{ij} \mid \theta_j))$$

The log-likelihood:

$$\log p(\mathcal{D} \mid \theta) = \sum_{c=1}^{C} N_c \log \pi_c + \sum_{j=1}^{D} \sum_{c=1}^{C} \sum_{i:y_i=c} \log p(x_{ij} \mid \theta_{jc})$$

This expression decomposes into a series of terms:

- π
- DC terms containing θ_{jc}

We can optimise all these parameters separately:

- MLE for class prior is given by:

  $\hat{\pi}_c = \frac{N_c}{N}$
- MLE for the likelihood depends on the type of distribution we choose. Suppose all features are binary, so MLE is:

  $\hat{\theta}_{jc} = \frac{N_{jc}}{N_c}$

Tutorial: NBC for spam filter

Using Bayes' Rule:

$$p(\text{spam} \mid \text{email}) = \frac{p(\text{email}|\text{spam})p(\text{spam})}{p(\text{email}|\text{spam}) \times p(\text{spam}) + p(\text{email}|\text{ham}) \times p(\text{ham})}$$

Each email contains many words:

$$p(\text{email} \mid C_k) = p(w_1 \mid C_k) \times p(w_2 \mid C_k) \ldots$$