Bayes and Naive Bayes Classifiers

Statistical Machine Learning (ENGG*6600*02)

School of Engineering,
University of Guelph, ON, Canada

Course Instructor: Benyamin Ghojogh
Summer 2023

Bayes Classifier

Bayes Classifier

- According to Bayes rule, the *posterior* is:

$$\begin{aligned}\mathbb{P}(x \in \mathcal{C}_j \,|\, X = x) &= \frac{\mathbb{P}(X = x \,|\, x \in \mathcal{C}_j)\,\mathbb{P}(x \in \mathcal{C}_j)}{\mathbb{P}(X = x)} \\ &= \frac{f_j(x)\,\pi_1}{\sum_{k=1}^{|\mathcal{C}|} \mathbb{P}(X = x \,|\, x \in \mathcal{C}_k)\,\pi_k}, \end{aligned} \tag{1}$$

  where $|\mathcal{C}|$ is the number of classes.
- Posterior: $\mathbb{P}(x \in \mathcal{C}_j \,|\, X = x)$, likelihood (or class conditional): $\mathbb{P}(X = x \,|\, x \in \mathcal{C}_j)$, and prior: $\mathbb{P}(x \in \mathcal{C}_j)$
- The Bayes classifier maximizes the posteriors of the classes [1]:

$$\widehat{\mathcal{C}}(\boldsymbol{x}) = \arg\max_k \; \mathbb{P}(\boldsymbol{x} \in \mathcal{C}_k \,|\, X = \boldsymbol{x}). \tag{2}$$

- According to Eq. (1) and Bayes rule, we have:

$$\mathbb{P}(\boldsymbol{x} \in \mathcal{C}_k \,|\, X = \boldsymbol{x}) \propto \mathbb{P}(X = \boldsymbol{x} \,|\, \boldsymbol{x} \in \mathcal{C}_k) \underbrace{\mathbb{P}(\boldsymbol{x} \in \mathcal{C}_k)}_{\pi_k}, \tag{3}$$

  where the denominator of posterior (the marginal) which is:

$$\mathbb{P}(X = \boldsymbol{x}) = \sum_{r=1}^{|\mathcal{C}|} \mathbb{P}(X = \boldsymbol{x} \,|\, \boldsymbol{x} \in \mathcal{C}_r)\,\pi_r,$$

  is ignored because it is not dependent on the classes $\mathcal{C}_1$ to $\mathcal{C}_{|\mathcal{C}|}$.

Bayes Classifier

- The Eqs. (2) and (3) were:

$$\widehat{\mathcal{C}}(\boldsymbol{x}) = \arg\max_k \ \mathbb{P}(\boldsymbol{x} \in \mathcal{C}_k \,|\, X = \boldsymbol{x}),$$

$$\mathbb{P}(\boldsymbol{x} \in \mathcal{C}_k \,|\, X = \boldsymbol{x}) \propto \mathbb{P}(X = \boldsymbol{x} \,|\, \boldsymbol{x} \in \mathcal{C}_k) \underbrace{\mathbb{P}(\boldsymbol{x} \in \mathcal{C}_k)}_{\pi_k}.$$

- According to Eq. (3), the posterior can be written in terms of likelihood and prior; therefore, Eq. (2) can be restated as:

$$\widehat{\mathcal{C}}(\boldsymbol{x}) = \arg\max_k \ \pi_k \, \mathbb{P}(X = \boldsymbol{x} \,|\, \boldsymbol{x} \in \mathcal{C}_k). \tag{4}$$

- Note that the Bayes classifier does not make any assumption on the posterior, prior, and likelihood, unlike LDA and QDA which assume the uni-modal Gaussian distribution for the likelihood.
- Therefore, we can say the difference of Bayes and QDA is in assumption of *uni-modal* Gaussian distribution for the likelihood (class conditional); hence, if the likelihoods are already uni-modal Gaussian, the Bayes classifier reduces to QDA.
- Likewise, the difference of Bayes and LDA is in assumption of Gaussian distribution for the likelihood (class conditional) and equality of covariance matrices of classes; thus, if the likelihoods are already Gaussian and the covariance matrices are already equal, the Bayes classifier reduces to LDA.

Bayes Classifier

- It is noteworthy that the Bayes classifier is an optimal classifier because it can be seen as an ensemble of hypotheses (models) in the hypothesis (model) space and no other ensemble of hypotheses can outperform it (see Chapter 6, Page 175 in [2]).
- In simple words, it is optimal because it is optimizing the posterior of classes.
- In the literature, it is referred to as **Bayes optimal classifier**.
- In conclusion, the Bayes classifier is optimal. Therefore, if the likelihoods of classes are Gaussian, QDA is an optimal classifier and if the likelihoods are Gaussian and the covariance matrices are equal, the LDA is an optimal classifier.
- Often, the distributions in the natural life are Gaussian; especially, because of central limit theorem [3], the summation of independent and identically distributed (iid) variables is Gaussian and the signals usually add in the real world. This explains why LDA and QDA are very effective classifiers in machine learning.

Naive Bayes Classifier

Naive Bayes Classifier

- Implementing Bayes classifier is difficult in practice so we approximate it by **naive Bayes** [4]. If x_j denotes the j-th dimension (feature) of $\boldsymbol{x} = [x_1, \ldots, x_d]^\top$, Eq. (4) is restated as:

$$\widehat{\mathcal{C}}(\boldsymbol{x}) = \arg\max_k \ \pi_k \, \mathbb{P}(x_1, x_2, \ldots, x_d \,|\, \boldsymbol{x} \in \mathcal{C}_k). \tag{5}$$

- The term $\mathbb{P}(x_1, x_2, \ldots, x_d \,|\, \boldsymbol{x} \in \mathcal{C}_k)$ is very difficult to compute as the features are possibly correlated.
- By chain rule in probability, we have:

$$\begin{aligned}&\mathbb{P}(x_1, x_2, \ldots, x_d \,|\, \boldsymbol{x} \in \mathcal{C}_k) = \\ &\mathbb{P}(x_1 \,|\, \boldsymbol{x} \in \mathcal{C}_k)\, \mathbb{P}(x_2 \,|\, x_1, (\boldsymbol{x} \in \mathcal{C}_k))\, \mathbb{P}(x_3 \,|\, x_1, x_2, (\boldsymbol{x} \in \mathcal{C}_k)) \cdots \mathbb{P}(x_d \,|\, x_1, \ldots, x_{d-1}, (\boldsymbol{x} \in \mathcal{C}_k)).\end{aligned}$$

- Naive Bayes relaxes this possibility and naively assumes that the features are conditionally independent ($\perp\!\!\!\perp$) when they are conditioned on the class:

$$\mathbb{P}(x_1, x_2, \ldots, x_d \,|\, \boldsymbol{x} \in \mathcal{C}_k) \overset{\perp\!\!\!\perp}{\approx} \mathbb{P}(x_1 \,|\, \mathcal{C}_k)\, \mathbb{P}(x_2 \,|\, \mathcal{C}_k) \cdots \mathbb{P}(x_d \,|\, \mathcal{C}_k) = \prod_{j=1}^{d} \mathbb{P}(x_j \,|\, \mathcal{C}_k).$$

- Therefore, Eq. (5) becomes:

$$\widehat{\mathcal{C}}(\boldsymbol{x}) = \arg\max_k \ \pi_k \prod_{j=1}^{d} \mathbb{P}(x_j \,|\, \mathcal{C}_k). \tag{6}$$

Naive Bayes Classifier

- In **Gaussian naive Bayes**, univariate Gaussian distribution is assumed for the likelihood (class conditional) of every feature:

$$\mathbb{P}(x_j \,|\, \mathcal{C}_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2}\right), \tag{7}$$

where the mean and unbiased variance are estimated as:

$$\mathbb{R} \ni \widehat{\mu}_k = \frac{1}{n_k} \sum_{i=1}^{n} x_{i,j}\, \mathbb{I}\big(\mathcal{C}(\boldsymbol{x}_i) = k\big), \tag{8}$$

$$\mathbb{R} \ni \widehat{\sigma}_k^2 = \frac{1}{n_k - 1} \sum_{i=1}^{n} (x_{i,j} - \widehat{\mu}_k)^2\, \mathbb{I}\big(\mathcal{C}(\boldsymbol{x}_i) = k\big), \tag{9}$$

where $x_{i,j}$ denotes the j-th feature of the i-th training instance and $\mathbb{I}(.)$ is the indicator function which is one and zero if its condition is satisfied and not satisfied, respectively.

- The prior can again be estimated using:

$$\widehat{\pi}_k = \frac{n_k}{n}, \tag{10}$$

where n_k and n are the number of training instances in the k-th class and in total, respectively.

Naive Bayes Classifier

- According to Eqs. (6) and (7), Gaussian naive Bayes is equivalent to QDA where the covariance matrices are **diagonal**, i.e., the off-diagonal of the covariance matrices are ignored. Therefore, we can say that QDA is more powerful than Gaussian naive Bayes because Gaussian naive Bayes is a simplified version of QDA.
- Moreover, it is obvious that Gaussian naive Bayes and QDA are equivalent for **one** dimensional data.
- Comparing to LDA, the Gaussian naive Bayes is equivalent to LDA if the covariance matrices are diagonal and they are all equal, i.e., $\sigma_1^2 = \cdots = \sigma_{|\mathcal{C}|}^2$; therefore, LDA and Gaussian naive Bayes have their own assumptions, one on the off-diagonal of covariance matrices and the other one on equality of the covariance matrices.
- As Gaussian naive Bayes has some level of optimality [4], it becomes clear why LDA and QDA are such effective classifiers.

Acknowledgment

- Some slides of this slide deck are based on our tutorial paper: "Linear and quadratic discriminant analysis: Tutorial" [5]

References

[1] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[2] T. Mitchell, *Machine learning*. McGraw Hill Higher Education, 1997.

[3] M. Hazewinkel, "Central limit theorem," *Encyclopedia of Mathematics, Springer*, 2001.

[4] H. Zhang, "The optimality of naive Bayes," in *American Association for Artificial Intelligence (AAAI)*, 2004.

[5] B. Ghojogh and M. Crowley, "Linear and quadratic discriminant analysis: Tutorial," *arXiv preprint arXiv:1906.02590*, 2019.