Misclassification Rate and Minimizing Expected Loss

Authors: Kaiwen Zhang, Christina Zhao, Qinwen Zhou. (PDF)

This lecture went over the Misclassification Rate for the Bayes Classifier and how to minimize the expected loss. The Misclassification Rate is the probability that one observation (an instance) is mislabeled by the classifier- essentially the accuracy of the model. The Expected Loss is the cost of a particular outcome. When minimizing the expected loss of the Bayes Classifier, thresholds are selected that the class label assigned an instance will minimize the error. In a symmetric loss function, all errors are equally costly, whereas in an asymmetric loss function, different errors have different losses. Thus, the choice of threshold is to increase the probability of the low cost outcomes (minimizing the expected loss). Minimizing loss may not necessarily mean that the Misclassification rate is minimized, so the boundary choice is a trade-off.

Bayes Theorem and its applications in Machine Learning

The notes below referenced Bayes Theorem topic from Machine Learning Mastery.

Bayes Theorem:

Principled way of calculating a conditional probability without the joint probability.

$P(A|B) = P(B|A) * P(A)/P(B)$

Maximum a Posteriori(MAP)

Maximum a posteriori is a use of Bayes Theorem for modeling hypotheses. It is defined as the optimization or seeking the hypothesis with the maximum posterior probability (note: $P(h|D)$ in eq. 1).

Hypothesis: Fitting different machine learning algorithms or models on the data to map inputs to outputs can be thought of as testing several hypotheses about the relationships in data. For example, based on the width and length of petals, we can predict the categories of iris flowers. More on Hypothesis in Machine Learning.

Bayes Theorem provides a probabilistic framework to describe the relationship between data and a hypothesis.

eq 1: $(h|D) = P(D|h) * P(h)/P(D)$

h stands for hypothesis, D stands for data

Using MAP can help us find the hypothesis that best describes the observed data. It provides a Bayesian foundation for common machine learning algorithms like linear regression. Another probabilistic framework that achieves the same goal is Maximum Likelihood Estimation(MLE). Further reading on the differences between MLE and MAP can be found at https://machinelearningmastery.com/maximum-a-posteriori-estimation/.
Bayes Optimal Classifier and Naive Bayes Classifier
Both classifiers are uses of Bayes Theorem for classification. The difference between Bayes Optimal Classifier and MAP is that MAP framework seeks the most probable model while Bayes Optimal Classifier looks for the best classification of the new instance.

In real-life, Bayes Classifier is computationally expensive especially when the number of input variables increases. In class, the example we are shown was that given the volume of a fruit observed, predict whether it is an apple or orange, which is quite straightforward as we were only considering one feature. However, computing the posterior probability of real data is not feasible as it is hard to get the conditional probability of the observation(which could be represented by 30+ features) based on the class.

$P(X1, X2, ..., Xn|class) * P(class)/P(observation)$

Naive Bayes Classifier simplifies the calculation by making the assumption that each input variable is independent from each other. Hence the calculation becomes

P(class | X1, X2, ..., Xn) = P(X1|class) * P(X2|class) * ... * P(Xn|class) * P(class) / P(data)

Misclassification Rate

Resource: Pattern Recognition and Machine Learning, Christopher M. Bishop

The Misclassification Rate is the probability of an instance being assigned the incorrect class label by the classifier. To start, we need a rule that will divide the input space (for the observations) into Decision Regions (R_k), one for each class (k), where if a given value of x is in the region, it will be assigned the label of class k.

As an example, we can look at a 2 class problem, with a Class 1 (C_1) and Class 2. (C_2)

The probability of making a misclassification is denoted as the sum of the integral of the probability of x in R_1 belonging to Class 2 (the red and the green areas

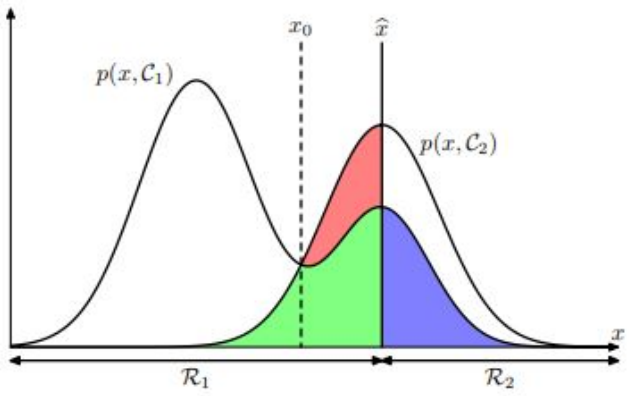

Figure 1: alt text

under the curve) and the probability of x in R_2 belonging to Class 1(the blue area under the curve).

$$p(\mathcal{C}_k|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{p(\mathbf{x})}.$$

Figure 2: alt text

In order to minimize the probability of a mistake occuring (p(mistake)), the decision rule should be chosen so that x is assigned to the class with an area under the curve with the smallest value (the smallest loss) at the point x. For example, if $p(C_1, x) < p(C_2, x)$, then x should be assigned to Class 1.

Another way of looking at how to minimize the probability of a mistake occuring is in terms of posterior probability of a class.

Using the product rule of probability (the probability of two independent events occurring together can be found by multiplying the individual probabilities of each event occurring alone), we get $p(C_k, x) = p(C_k|x) * p(x)$: that the probability of x being in the class is equal to the posterior probability multiplied by the probability of x occurring on its own.

$$p(\mathcal{C}_k|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{p(\mathbf{x})}.$$

Since: we can ignore $p(x)$, as it

is common to both equations. We can then think of it as $p(C_k, x) = p(C_k|x)$, where misclassification can be minimized by choosing to assign x to the class where the posterior probability $p(C_k|x)$ is the largest.

Minimizing Expected Loss

Loss matrix is introduced to encode a penalty for misclassification errors. Sometimes we will see a negative entry in the loss matrix indicating a reward for making the correct prediction.

A loss matrix is introduced to encode a penalty for misclassification errors for any given combination of true class and classified class. The convention is that a penalty is indicated by positive values; however, sometimes we will see negative entries in the loss matrix which indicate a reward for making the correct classification (same true and classified class).

The mathematical expression for the expected loss is the following equation:

$$\mathbb{E}[L] = \sum_k \sum_j \int_{\mathcal{R}_j} L_{kj} p(\mathbf{x}, \mathcal{C}_k) \, \mathrm{d}\mathbf{x}$$

where a k represents a true class and a j represents a classified class. R_j is the decision region on the x domain for classifying a class j. The equation illustrates the following process: the product of a penalty (or sometimes reward) value and the joint probability of x being of a particular true class k should be evaluated and aggregated across all values of x that are to be classified as class j, do that across all the different classified states, then more broadly across all the different true states.

Intuitively, if we treat the penalty/reward value as a constant, the focus of the calculus becomes $p(x, C_k)$ integrated across the decision region for a certain class j, and this can be seen as the probability for classifying j while actually k. The two summations to the left of the integration, therefore, are just the summing up of the products of penalty/reward value and probability of true-classified-states combinations, across all true-classified state combinations.

Truth \ Decision	cancer	normal
cancer	0	1000
normal	1	0

The matrix above shows the penalties for (mis-)detecting cancer. The expected loss in this case will be 0*p(detect cancer & true cancer) + 1000*p(detecting normal & true cancer) + 1*p(detecting cancer & true normal) + 0*p(detecting normal and true normal). This example is an application of the logic described in the previous paragraph.

However, since $p(x, C_k)$ is essentially the same as $p(C_k|x)$ because of the presence of $p(x)$ common to all terms, we can establish the expected-loss optimization as follows:

Regions $\mathcal{R}_j$ are chosen to minimize

$$\mathbb{E}[L] = \sum_k L_{kj} p(\mathcal{C}_k|\mathbf{x})$$

Translated into English, and returning to the cancer example, the optimization story is as follows: For each column (classified class), sum the products of the penalties and posterior probabilities of true classes. For example, if x is such that $p(C_{cancer} \mid x) = 0.4$ and $p(C_{normal} \mid x) = 1$ - $p(C_{cancer} \mid x) = 0.6$, then the sum of the products for the "cancer" column would be the "expected value" 0*0.4+1*0.6=0.6 and for the "normal" column the expected value would be 1000*0.4+0*0.6=400.
The optimal choice here is the classification that minimizes loss for this particular x – cancer in this case. Doing this for all values of x, all x's with the same smaller-expected-value column being "cancer" will form a decision region, and all x's having "normal" being the smaller expected-value producer will form the other decision region. These two decision regions constitute the optimal rules for minimizing expected loss. Of course, this process can be generalized to more than two classes/regions as well.