Chapter 3

Naïive Bayes Learning

Contents

3.1 Direct Learning

- Consider a distribution D

- X - Instance space, Y - Set of labels. (e.g. ± 1)
- Given a sample $\{(\mathbf{x}, y)\}_1^n$ and a loss function $L(\mathbf{x}, y)$, find a hypothesis $h \in H$ that minimizes $\sum_{i=1\dots n} L(h(\mathbf{x}_i), y_i)$.

Table 3.1: Losses

$0-1$ loss:	$L(h(\mathbf{x}), y) = 1, h(\mathbf{x}) \neq y$ otherwise $L(h(\mathbf{x}), y) = 0$
L_2 Loss:	$L(h(\mathbf{x}), y) = (h(\mathbf{x}) - y)^2$
Hinge Loss:	$L(h(\mathbf{x}), y) = \max\{0, 1 - yh(\mathbf{x})\}$
Exponential Loss:	$L(h(\mathbf{x}), y) = e^{-yh(\mathbf{x})}$

3.2 Probabilistic Model

Paradigm:

- Learn a probability distribution of the dataset.
- Use it to estimate which outcome is more likely.

Instead of learning $h : X \to Y$, learn $P(Y|X)$.

- Estimate probability from data
  - Maximum Likelihood Estimate (MLE)
  - Maximum Aposteriori Estimation (MAP)

3.3 Probability Recap

$$0 \leq P(A) \leq 1$$
$$P(true) = 1, P(false) = 0$$
$$P(A \vee B) = P(A) + P(B) + P(A \wedge B)$$
$$P(A|B) = \frac{P(A \wedge B)}{P(B)}$$

3.4 Joint Distribution

Making a joint distribution of d variables

- Make a truth table listing all combinations of values of your variables (if there are d boolean variables then the table will have 2^d rows)

- For each combination of values, say how probable it is.
- The probability must sum up to 1.

Once we have the Joint Distribution, we find probability of any logical expression involving these variables.

$$\begin{aligned} P(E) &= \sum_{rows\ matching\ E} P(row) \\ P(E_1 \mid E_2) &= \frac{E_1 \wedge E_2}{P(E_2)} \\ &= \frac{\sum_{\text{rows matching } E_1 \text{ and } E_2} P(row)}{\sum_{\text{rows matching } E_2} P(row)} \end{aligned} \tag{3.1}$$

3.5 Probability Distribution

3.5.1 Bernoulli Distribution

Random Variable X takes values $\{0, 1\}$ such that

$$P(X = 1) = p = 1 - P(X = 0)$$

Example: Tossing a coin.

3.5.2 Binomial Distribution

Random Variable X takes values $\{1, 2, \ldots, n\}$ representing the number of successes $X = 1$ in n Bernoulli trials.

$$P(X = k) = f(n, p, k) = C_n^k p^k (1 - p)^{n-k}$$

Example: Tossing a coin n times.

3.5.3 Categorical Distribution

Random Variable X takes on values in $\{1, 2, \ldots, k\}$ such that

$$P(X = i) = p_i \text{ and } \sum_1^k p_i = 1$$

Example: Rolling a die.

3.5.4 Multinomial Distribution

- Let the random variables $X_i (i = 1, 2, \ldots, k)$ indicates the number of times outcome i was observed over the n trials.
- The vector $X = (X_1, X_2, \ldots, X_k)$ follows a multinomial distribution (n, p) where $p = (p_1, p_2, \ldots, p_k)$ and $\sum_1^k = 1$

$$\begin{aligned} f(x_1, x_2, \ldots, x_k, n, p) &= P(X_1 = x_1, X_2 = x_2, \ldots, X_k = x_k) \\ &= \frac{n!}{x_1! \times x_2! \times \cdots \times x_k!} \times p_1^{x_1} \times p_2^{x_2} \times \cdots \times p_k^{x_k} \text{ where } \sum_{i=1}^{k} x_i = n \end{aligned}$$

Example: Rolling a die n times.

3.6 Independence

When two events do not affect each other's probabilities, they are called independent events

$$\begin{aligned} A \perp\!\!\!\perp B &\leftrightarrow P(A \wedge B) = P(A) \times P(B) \\ &\leftrightarrow P(A \mid B) = P(A) \end{aligned}$$

The conditional independence of events A and B, given C is:

$$\begin{aligned} A \perp\!\!\!\perp B|C \leftrightarrow P(A \mid B, C) &= \frac{P(A \wedge B|C)}{P(B|C)} = \frac{P(A|C) \times P(B|C)}{P(B|C)} \\ &= P(A|C) \end{aligned}$$

3.7 Bayes' Rule

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \tag{3.2}$$

where A and B are events and $P(B) \neq 0$. Applying Bayes' rule for machine learning –

$$P(hypothesis \mid evidence) = \frac{P(evidence \mid hypothesis) \times P(hypothesis)}{P(evidence)} \tag{3.3}$$

3.8 Bayesian Learning

- Goal: find the best hypothesis from some space H of hypotheses, given the observed data (evidence) D.
- Define the most probable hypothesis in H to be the best.
- In order to do that, we need to assume a probability distribution over the class H.
- In addition, we need to know something about the relation between the evidence and the hypotheses.

$P(h)$ – Prior Probability of the hypothesis h. Reflects the background knowledge, before data is observed.

$P(D)$ – Probability that this sample of the data is observed.

$P(D|h)$ – Probability of observing the sample D, given that hypothesis h is the target, also referred to as likelihood.

$P(h|D)$ – Posterior probability of h. The probability that h is the target, given that D has been observed.

- $P(h|D)$ increases with $P(h)$ and $P(D|h)$.
- $P(h|D)$ decreases with $P(D)$.

3.9 Maximum APosteriori Estimate

$$P(h|D) = \frac{P(D|h) \times P(h)}{P(D)} \tag{3.4}$$

- The learner considers a set of candidate hypotheses H (models) and attempts to find the most probable one $h \in H$, given the observed data.
- Such maximally probable hypothesis is called maximum a posterior estimate (MAP). Bayes theorem is used to compute it:

$$\begin{aligned} h_{MAP} &= \arg\max_{h \in H} P(h|D) \\ &= \arg\max_{h \in H} \frac{P(D|h) \times P(h)}{P(D)} \\ &= \arg\max_{h \in H} P(D|h) \times P(h) \end{aligned}$$

3.10 Maximum Likelihood Estimate

- We may assume that a priori, hypotheses are equally probable.

$$P(h_i) = P(h_j) \forall h_i, h_j \in H$$

- With that assumption, we can treat $\frac{P(h)}{P(D)}$ as a constant. We get the maximum likelihood estimate (MLE):

$$\begin{aligned} h_{MLE} &= \arg\max_{h \in H} \frac{P(D|h) \times P(h)}{P(D)} \\ &= \arg\max_{h \in H} P(D|h) \end{aligned} \tag{3.5}$$

- Here we just look for the hypothesis that best explains the data.

3.11 Bayesian Classifier

- $f : \mathbf{X} \to Y$ where, instances $\mathbf{x} \in \mathbf{X}$ is a collection of inputs –

$$\mathbf{x} = (x_1, x_2, \ldots, x_n)$$

- Given an example, assign it the most probable value in Y.

$$\begin{aligned} y_{MAP} &= \arg\max_{y_j \in Y} P(y_j|x) \\ &= \arg\max_{y_j \in Y} P(y_j|x_1, x_2, \ldots, x_n) \\ &= \arg\max_{y_j \in Y} \frac{P(x_1, x_2, \ldots, x_n|y_j)P(y_j)}{P(x_1, x_2, \ldots, x_n)} \\ &= \arg\max_{y_j \in Y} P(x_1, x_2, \ldots, x_n|y_j)P(y_j) \end{aligned} \tag{3.6}$$

- Given the training data, we have to estimate the two terms.
- Estimating $P(y)$ is easy, e.g., under the binomial distribution assumption, count the number of times y appears in the training data.
- However, it is not feasible to estimate $P(x_1, x_2, \ldots, x_n|y)$
- In this case, we have to estimate for each target value, the probability of each instance (some of which might now ever occur).
- In order to use a Bayesian classifiers in practice, we need to make assumptions that will allow us to estimate these quantities.

3.12 Naïve Bayes Classifier

Assumption: Input feature values are independent, given the target value.

$$\begin{aligned}
P(x_1, x_2, \ldots, x_n | y_j) &= P(x_1|y_j) \times P(x_2, \ldots, x_n, x_j) \\
&= P(x_1|y_j) \times P(x_2|y_j) \times P(x_3, \ldots, x_n, x_j) \\
&= P(x_1|y_j) \times P(x_2|y_j) \times P(x_3|y_j) \times \cdots \times P(x_n|x_j) \\
&= \prod_{i=1}^{n} P(x_i|y_j)
\end{aligned} \tag{3.7}$$

$$\begin{aligned}
Y_{NB} &= \arg\max_{y_j \in Y} P(x_1, x_2, \ldots, x_n | y_j) P(y_j) \\
&= \arg\max_{y_j \in Y} P(y_j) \prod_{i=1}^{n} P(x_i|y_j)
\end{aligned} \tag{3.8}$$

3.13 Estimating Probabilities

How do we estimate $P(x_i|y)$?

$$P(x_i|y) = \frac{\text{number of } x_i \text{ labeled as } y}{\text{total number of label } y} = \frac{n_i}{n} \tag{3.9}$$

Sparsity of data is a problem –

- If n is small, the estimate is not accurate.
- If $n_i = 0$, we will never accurately predict Y if an instance that never appeared in the training appears in the test data.

3.14 Laplace Smoothing

$$P(x_i|y) = \frac{n_i + \alpha}{n + \alpha d} \tag{3.10}$$

- Also known as additive smoothing.
- $\alpha > 0$ is a smoothing parameter.
- d is the dimension of the input.

3.15 Continuous Features

- Assume $P(x_i|y)$ has a Gaussian (normal) distribution.
- It is a continuous distribution with probability density function:

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{3.11}$$

  - μ is the mean of the distribution.
  - σ^2 is the variance of the distribution.
  - x is a continuous variance $(-\infty \leq x \leq \infty)$

3.16 Gaussian Naïve Bayes

Table 3.2: Naïve Bayes Example

X_1	X_2	X_3	Y
2	3	1	1
-1.2	2	0.4	1
1.2	0.3	0	0
2.2	1.1	0	1

Compute the mean and standard deviation to estimate the likelihood.

$$\mu_1 = E[X_1 \mid Y = 1] = \frac{2 + (-1.2) + 2.2}{3} = 1$$

$$\sigma_1^2 = E\left[(X_1 - \mu_1)^2 | Y = 1\right] = \frac{(2-1)^2 + (-1.2-1)^2 + (2.2-1)^2}{3} = 2.43$$

$$P(x_1|Y = 1) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_1-\mu_1)^2}{2\sigma^2}} = \frac{1}{3.91} e^{-\frac{(x_1-1)^2}{4.86}}$$

3.17 Bayesian Belief Network

- Naïve Bayes classifier works with the assumption that the values of the input features are conditionally independent given the target value.
- This assumption dramatically reduces the complexity of learning the target function.
- Bayesian Belief Network describes the probability distribution governing a set of variables by specifying a set of conditional independence assumptions along with a set of conditional probabilities. Conditional independence assumptions here apply to subsets of the variables.

$$P(x_1, x_2, \ldots, x_l \mid {x_1}', {x_2}`, \ldots, {x_m}`, y_1, y_2, \ldots, y_n) = P(x_1, x_2, \ldots, x_l | y_1, y_2, \ldots, y_n)$$

3.18 Training Bayesian Classifier

During training, typically log-space is used.

$$
\begin{aligned}
y_{NB} &= \arg\max_y \left[\log P(y) \prod_{i=1}^{n} P(x_i|y) \right] \\
&= \arg\max_y \left[\log P(y) + \sum_{i=1}^{n} \log P(x_i|y) \right]
\end{aligned}
$$

3.19 Text Classification

Algorithm 3.1 Text-based Naïve Bayes Classification

```
function TRAIN-NAIVE-BAYES(D, C) returns log P(c) and log P(w|c)
    for all class c ∈ C do                                   ▷ Calculate P(c) terms
        N_doc ← number of documents in D
        N_c ← number of documents from D in class c
        logprior[c] ← log N_c / N_doc
        V ← vocabulary of D
        bigdoc[c] ← APPEND(d) for d ∈ D with class c
        for all word w in V do                               ▷ Calculate P(w|c) terms
            COUNT(w, c) ← # of occurrences of w in bigdoc[c]
            loglikelihood[w, c] ← log (COUNT(w,c)+1) / (Σ_{w' in V} (COUNT(w',c)+|V|))
        end for
    end for
    return logprior, loglikelihood, V
end function
```

Algorithm 3.2 Test Naïve Bayes

```
function TEST-NAIVE-BAYES(testdoc, logprior, loglikelihood, C, V) returns best c
    for all class c ∈ C do
        sum[c] ← logprior[c]
        for all position i in testdoc do
            word ← testdoc[i]
            if word ∈ V then
                sum[c] ← sum[c] + loglikelihood[word, c]
            end if
        end for
    end for
    return argmax_c, sum[c]
end function
```

The word with doesn't occur in the training set, so we drop it completely (we don't use unknown word models for Naïve Bayes)

3.20 Evaluating Classifiers

- Gold Label is the correct output class label of an input.
- Confusion Matrix is a table for visualizing how a classifier performs with respect to the gold labels, using two dimensions (system output and gold labels), and each cell labeling a set of possible outcomes.
- True Positives and True Negatives are correctly classified outputs belonging to the positive and negative class, respectively.
- False Positives and False Negatives are incorrectly classified outputs.

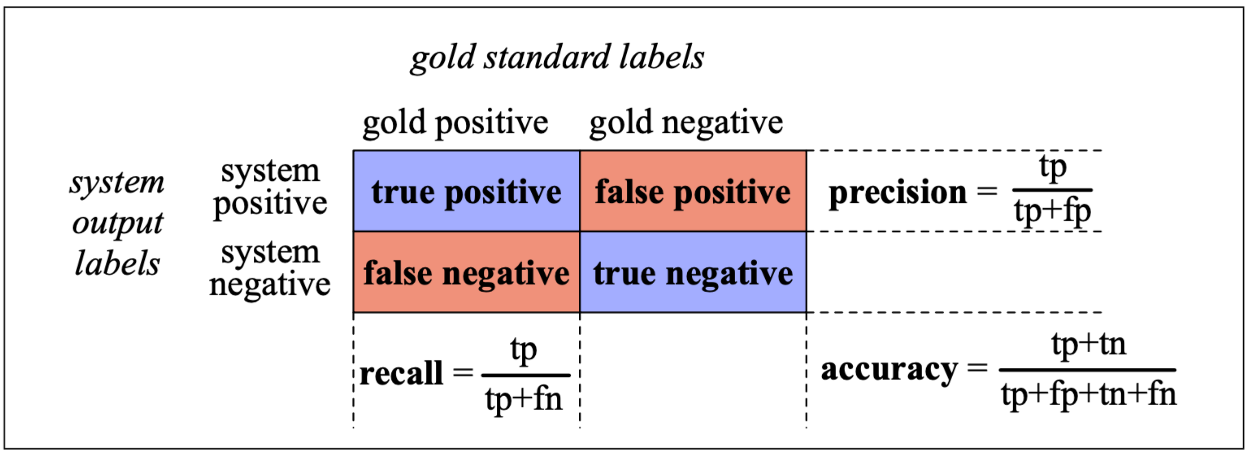

Figure 3.1: An example of a Confusion Matrix.

3.21 Precision, Recall, F-Measure

$$\textbf{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \tag{3.12}$$

$$\textbf{Recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \tag{3.13}$$

$$\textbf{F} - \textbf{measure} = F_\beta = \frac{(\beta^2 + 1)PR}{\beta^2 P + R} \tag{3.14}$$

3.22 ROC Curve

- A receiver operating characteristic curve (ROC curve) is a graphical plot that illustrates the performance of a binary classifier model.
- The ROC curve is the plot of the true positive rate (recall) (TPR) (3.13) against the false positive rate (FPR).

$$\mathbf{FPR} = \frac{\text{false positives}}{\text{false positives} + \text{true negatives}} \tag{3.15}$$

- ROC curve plots TPR vs. FPR at different classification thresholds.
- Classification threshold is used to convert the output of a probabilistic classifier into class labels.
- The threshold determines the minimum probability required for a positive class.
- Lowering the classification threshold classifiers more items as positive, thus increasing both False Positives and True Positives.

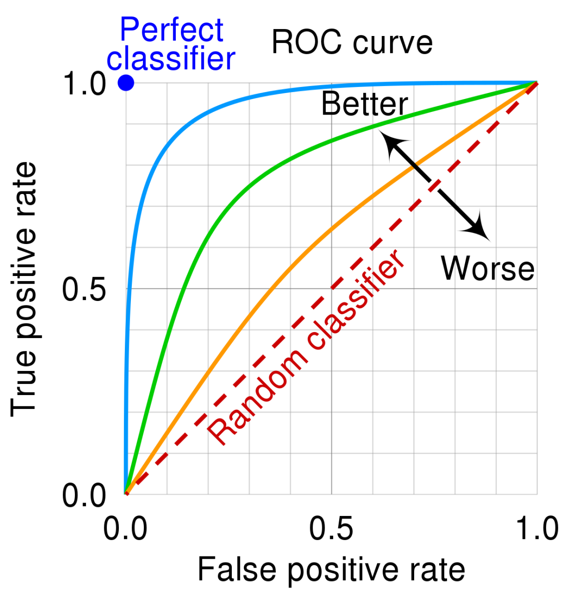

Figure 3.2: ROC Curve

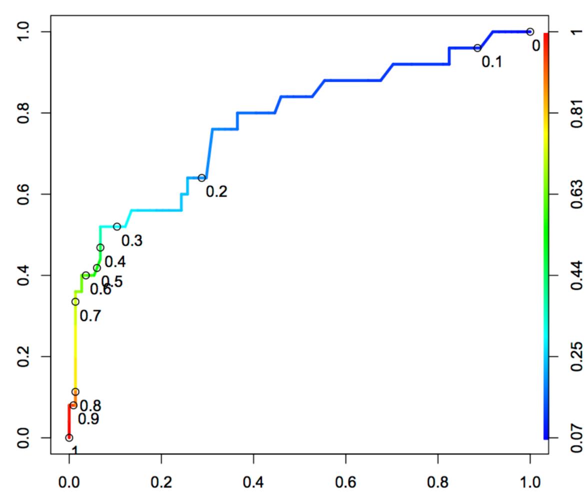

Figure 3.3: ROC Curve with defined thresholds

3.23 AUC

- The Area Under the Curve (AUC) provides an aggregate measure of performance across all possible classification thresholds.
- Between two ROC curves plotted based on two learning models, the model with the higher AUC learned better than the other.

3.24 Naïve Bayes: Two Classes

- Naïve Bayes classifier gives a method for predicting the most likely class rather than an explicit class.
- In the case of two classes, $y \in \{0, 1\}$ we predict that $y = 1$ iff

$$\frac{P(y_j = 1) \times \prod_{i=1}^{n} P(x_i | y_j = 1)}{P(y_j = 0) \times \prod_{i=1}^{n} P(x_i | y_j = 0)} > 1$$

- $p_i = P(x_i | y_j = 1)$, $q_i = P(x_i | y_j = 0)$. Assuming Bernoulli Naïve Bayes,

$$\frac{P(y_j = 1) \times \prod_{i=1}^{n} p_i^{x_i} (1 - p_i)^{1 - x_i}}{P(y_j = 0) \times \prod_{i=1}^{n} q_i^{x_i} (1 - q_i)^{1 - x_i}} > 1$$
$$\Rightarrow \frac{P(y_j = 1) \times \prod_{i=1}^{n} (1 - p_i)(p_i / 1 - p_i)^{x_i}}{P(y_j = 0) \times \prod_{i=1}^{n} (1 - q_i)(q_i / 1 - q_i)^{x_i}} > 1$$

Take logarithm; we predict $y = 1$ iff

$$\log \frac{P(y_j = 1)}{P(y_j = 0)} + \sum_i \log \frac{1 - p_i}{1 - q_i} + \sum_i \left(\log \frac{p_i}{1 - p_i} - \log \frac{q_i}{1 - q_i} \right) x_i > 0$$

- We get that Naïve Bayes is a linear separator with –

$$w_i = \log \frac{p_i}{1 - p_i} - \log \frac{q_i}{1 - q_i} = \log \frac{p_i (1 - q_i)}{q_i (1 - p_i)} \tag{3.16}$$

- NB classifier corresponds to a linear classifier if the likelihood is from exponential family distributions i.e., Bernoulli, binomial, Gaussian etc.
- In the case of two classes, we can say:

$$\log \frac{P(y_j = 1 | x)}{P(y_j = 0 | x)} = \sum_i \mathbf{w}_i \mathbf{x}_i + b$$

- but since $P(y_j = 1 | x) = 1 - P(y_j = 0 | x)$, we get:

$$P(y_j = 1 | x) = \frac{1}{1 + e^{-(\sum_i w_i x_i + b)}}$$

- This is simply the logistic function.