Deadly Mushroom Classification

Garrett Ransom

October 2018

Abstract

Eating certain types of mushrooms can be the difference between life and death. In this paper, we evaluate the performance of various classification models that can assist us in distinguishing poisonous mushrooms from edible ones. Our dataset is composed of 8124 examples described by 22 categorical variables and one binary predictor variable.

1 Introduction

blah blah blah

2 Classification Models

2.1 Logistic Regression

In the case of Logistic Regression, we use the sigmoid function as a means to map our predictor variable as a probabilty such that $y \in \{0, 1\}$.

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$$

where

$$g(z) = \frac{1}{1 + e^{-z}}$$

We maximize the log likelihood function to find the best fit for θ.

$$\ell(\theta) = \sum_{i=1}^{m} y^{(i)} \log h(x^{(i)}) + (1 - y^{(i)}) \log(1 - h(x^{(i)}))$$

We maximize the log likelihood by using gradient descent given by $\theta := \theta - \alpha \nabla_\theta \ell(\theta)$. By taking the partial derivatives with respect to θ for one training example (x, y), we get the following equation:

$$\frac{\partial}{\partial \theta_j} \ell(\theta) = (y - h_\theta(x)) x_j$$

Therefore, using stochastic gradient descent rule gives us:

$$\theta_j := \theta_j - \alpha (y^{(i)} - h_\theta(x^{(i)})) x_j^{(i)}$$

2.2 Support Vector Machines

In the case of Support Vector Machines, we create a classifier by using the equation

$$h_{w,b} = g(w^T x + b).$$

Given a training example $(x^{(i)}, y^{(i)})$, we can form the idea of a functional margin of (w, b) with respect to the training example as

$$\hat{\gamma}^{(i)} = y^{(i)}(w^T x + b).$$

Given a training set $S = \{(x^{(i)}, y^{(i)}); i = 1, ..., m\}$, we can define the functional and geometric margin as the smallest margin of across all training examples

$$\gamma = \min_{i=1,...,m} \hat{\gamma}^{(i)}.$$

Ultimately, by doing some cool math, we can use the following problem for finding the optimal margin classifier

$$\min_{w,b} \frac{1}{2} ||w||^2 \tag{1}$$

$$\text{s.t} \quad y^{(i)}(w^T x + b) \geq 1, i = 1, ..., m \tag{2}$$

2.3 Decision Trees

Suppose we want to create a decision tree for a classification task. If we have N training examples composed of p inputs (x_i, y_i) for $i, ..., N$ with $x_i = (x_{i1}, x_{i2}, ..., x_{ip})$, then there exists a set of splitting points that best classify output variables. We can split the data into M regions $R_1, R_2, ..., R_M$ and we can model the response as a constant c_m in every region:

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m).$$

We can use the Gini Index, Misclassification Error, or Cross-Entropy to find the best splitting points.

$$\text{Gini Index: } \sum_{k \neq k'} \hat{p}_{mk} \hat{p}_{mk'} = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$

2.4 Random Forests

2.5 Naive Bayes

The Naive Bayes classifier is a model based on Bayes' theorem with the assumption of independence between features. Bayes' Rule tells us that

$$p(A|B) = \frac{p(A)p(B|A)}{p(B)}.$$

$p(A|B)$ is considered the posterior, while $p(A)$ is the prior andtio $p(B|A)$ is the likelihood and $p(B)$ is the evidence.

But how does this apply to classification? If we represent a vector $\mathbf{x} = (x_1, ..., x_n)$ by n independent features, we calculate the probability of $\mathbf{x}$ belonging to a class C_k as $p(C_k|x_1, ..., x_n)$. Referring back to Bayes' theorem, the conditional probability can be represented as

$$p(C_k|\mathbf{x}) = \frac{p(C_k)p(\mathbf{x}|C_k)}{p(\mathbf{x})}.$$

In practice, we only pay attention to the numerator of this equation. We can further break down the representation into the joint probability model

$$p(C_k, x_1, ..., x_n)$$

which can be further broken down by using the chain rule of probability

$$p(C_k, x_1, ..., x_n) = p(x_1|x_2, ..., x_n, C_k)p(x_2|x_3, ..., x_n, C_k)...p(x_{n-1}|x_n, C_k)p(x_n|C_k)p(C_k).$$

Now, conditional independence comes into play, where we can assume that for each feature x_i, every other feature x_j for $j \neq i$ is conditionally independent relative to x_i. This means that

$$p(x_i|x_i + 1, ..., x_n, C_k) = p(x_i|C_k).$$

Therefore, the joint model can be represented as

$$p(C_k|x_1, ..., x_n) = p(C_k) \prod_{i=1}^{n} p(x_i|C_k).$$

3 Experiment Results