CS281: Advanced ML **September 18, 2017**

Lecture 5: Linear Classification

Lecturer: Sasha Rush *Scribes: Demi Guo, Artidoro Pagnoni, Luke Melas-Kyriazi*

5.1 Classification Introduction

Last time we saw linear regression. In linear regression we were predicting $y \in \mathbb{R}$, in classification instead we deal with a discrete set, for example $y \in \{0, 1\}$ or $y \in \{1, ..., C\}$. This distinction only matters for this lecture, starting from next class we will generalize the topics and treat them as the same thing.
Among the many applications, linear classification is used in sentiment analysis, spam detection, and facial and image recognition.
We will use generative models of the data, which means that we will model both the x and the y explicitly, and we are not keeping x fixed. In the case of the spam filter earlier, x is the email body, and y is the label $\{$spam, not spam$\}$. A generative model of the email and labels, we would model the distribution of x, of the text in the email itself, and not only the distribution of the category y.
We will explore the basic method of Naïve Bayes in detail. Even with a very simple method like Naïve Bayes with basic features it is possible to perform extremely well on many classification tasks when large training data sets are available. For example, this simple model performs almost as well (one percent point difference) as very complex methods on spam detection.

5.2 Naïve Bayes

Note that the term "Bayes" in Naïve Bayes (NB) does not have to do with Bayesian modeling, or the presence of priors on parameters. We won't have any priors for the moment. General Naïve Bayes takes the following form:

$$y \sim Cat(\pi) \quad \text{[class distribution]} \tag{5.1}$$

$$x_j|y \sim \text{\_\_\_\_\_} \quad \text{[class conditional]} \tag{5.2}$$

Where y is the class and comes from a categorical distribution, and x_j is a dimension of the input x.
In Naïve Bayes, the form of the class distribution is fixed and parametrized independently from the class conditional distribution. The "Naïve" term in "Naïve Bayes" precisely refers to the conditional independence between y and $x_j|y$. Depending of what the data looks like we can choose a different form for the class conditional distribution.
Here we present three possible choices for the class conditional distribution:

5.2.1 Multivariate Bernoulli Naïve Bayes:

$$x_j|y \sim Ber(\mu_{jc}) \quad \text{if } y = c \tag{5.3}$$

Here y takes values in a set of classes, and μ_{jc} is a parameter associated with a specific feature (or dimension) in the input and a specific class. We use MV Bernoulli when we only allow two possible values for each feature, therefore $x_j|y$ follows a Bernoulli distribution.
We can think of x as living in a hyper cube, with each dimension j having an associated μ for each class c. From here the name multivariate Bernoulli distribution.

5.2.2 Categorical Naïve Bayes:

$$x_j|y \sim Cat(\mu_{jc}) \quad \text{if } y = c \tag{5.4}$$

We use the Categorical Naïve Bayes when we allow different classes for each feature j, so $x_j|y$ follows a catergorical distribution.

5.2.3 Multivariate Normal Naïve Bayes

$$\boldsymbol{x}|\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}^c_{diag}) \tag{5.5}$$

Note that here we use $\boldsymbol{x}$ vector and not a specific feature. Since we impose that $\boldsymbol{\Sigma}^c$ is a diagonal matrix, we have no covariance between features, so this comes down to having an independent multivariate normal for each feature (or dimension) of the input. This is also required by the "Naïve" assumption of conditional independence. We would use MVN Naïve Bayes when the features take continuous values in $\mathbb{R}$.

5.3 General Naïve Bayes

We consider the data points $\{(\boldsymbol{x}_n, y_n)\}$, without specifying a particular generative model. The likelihood of each data point is:

$$p(\boldsymbol{x}_n, y_n|\text{param}) = p(y_n|\boldsymbol{\pi}) \prod_j p(x_{nj}|y_n, \text{param}) \tag{5.6}$$

$$= \prod_c \pi_c^{(y_n=c)} \prod_j \prod_c p(x_{nj}|y_n)^{(y_n=c)} \tag{5.7}$$

Where in equation (5.6) we assume conditional independence ("Naïve" assumption). The term $p(x_{nj}|y_n)$ depends on the generative model used for x and also on the class y_n.
We can then solve for the parameters maximizing the likelihood, which is equivalent to maximizing the log likelihood.

$$\underset{(\boldsymbol{\pi},\boldsymbol{\mu})}{\text{argmax}} \sum_n \log p(\boldsymbol{x}_n, y_n|\text{param}) \tag{5.8}$$

$$= \underset{(\boldsymbol{\pi},\boldsymbol{\mu})}{\text{argmax}} \sum_c N_c \log \pi_c + \sum_i \sum_c \sum_{n:y_n=c} \log p(x_{nj}|y_n) \tag{5.9}$$

$$= \left(\underset{(\boldsymbol{\pi},\boldsymbol{\mu})}{\text{argmax}} \sum_c N_c \log \pi_c \right) + \left(\underset{(\boldsymbol{\pi},\boldsymbol{\mu})}{\text{argmax}} \sum_i \sum_c \sum_{n:y_n=c} \log p(x_{nj}|y_n) \right) \tag{5.10}$$

Where $N_c = \sum_n \mathbb{1}(y_n = c)$, and $N =$ number of data points.
This factors into two parts (5.10), the first only depending on $\boldsymbol{\pi}$ the other is the MLE for the class condition distribution on each feature or dimension of the input. This factorization allows to solve for the maximizing $\boldsymbol{\pi}$ and the maximizing parameters for the class conditional separately.
For example, if we use a Multivariate Bernoulli Naïve Bayes generative model we would get the following parameters from MLE:

$$\pi_c = \frac{N_c}{N} \tag{5.11}$$

$$\mu_{jc} = \frac{\sum_{n:y_n=c} x_{nj}}{N_c} = \frac{N_{cj}}{N_c} \tag{5.12}$$

Again, where $N_c = \sum_n \mathbb{1}(y_n = c)$, $N_{cj} = \sum_n \mathbb{1}(y_n = c)x_{nj}$ and $N =$ number of data points.