Runqiu Ye
Stanford CS299
Problem Set #2
07/07/2024

Problem Set #2: Supervised Learning II

Problem 1 Logistic Regression: Training stability

(a) The most notable difference in training the logistic regression model on datasets A and B is that the algorithm does not converge on dataset B.

(b) To investigate why the training procedure behaves unexpectedly on dataset B, but not on A, we print the value of θ after every 10000 iterations. We notice that for data set B, although the normalized $\frac{\theta}{\|\theta\|}$ almost stop changing after several tens of thousands of iterations, each componenet of the unnormalized θ keeps increasing. We also notice that dataset A is not linearly separable while dataset B is linearly separable.

From the code, we notice that the algorithm calculates the gradient of loss function as

$$\nabla_\theta J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\frac{y^{(i)}x^{(i)}}{1+\exp(y^{(i)}\theta^T x^{(i)})}.$$

From this, we know that the algorithm uses gradient descent to minimize the loss function

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\log\frac{1}{1+\exp(-y^{(i)}\theta^T x^{(i)})}.$$

Hence, for a dataset that is linearly separable, that is, $y^{(i)}\theta^T x^{(i)} > 0$ for all i, a θ with larger norm always leads to a smaller loss, preventing the algorithm from converging. However, on a dataset that is not linearly separable, there exists i such that $y^{(i)}\theta^T x^{(i)} < 0$. By plotting $f(z) = \log(1+e^{-z})$ in Figure 1, we notice that negative margin dominates when scaling θ to a larger norm. Hence, we cannot always increase θ to a larger norm while minimizing $J(\theta)$.

(c) Consider the following modifications

i. Using a different constant learning rate will not make the algorithm converge on dataset B, since scaling θ to larger norm still always decreases the loss.

ii. Decreasing the learning rate over time will make the algorithm converge for dataset B, since in this way the change of θ converge to 0.

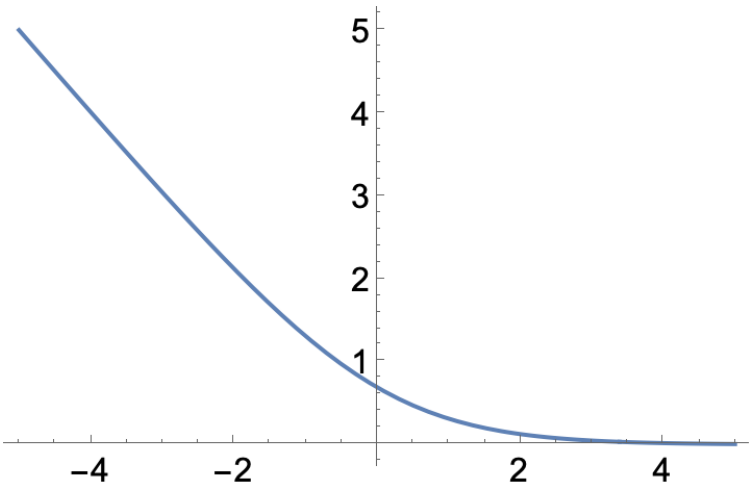

Figure 1: Plot of $f(z) = \log(1 + e^{-z})$ for $-5 \leq z \leq 5$.

iii. Linear scaling the input features does not help, since it does not change the dataset's linera separability.

iv. Adding a regularization term $\|\theta\|_2^2$ helps, since now scaling θ to larger norm penalize the algorithm.

v. Adding zero-mean Gaussian noise to the training data or labels helps as long as it makes the dataset not linearly separable.

(d) Support vector machines, which uses hinge loss, are not vulnerable to datasets like B. In SVM, geometric margin is considered, instead of functional margin considered here. In other words, θ is normalized, so for linearly separable datasets like B, the algorithm will still converge.

■

Problem 2 Model Calibration

Try to understand the output $h_\theta(x)$ of the hypothesis function of a logistic regression model, in particular why we might treat the output as a probability.

When probabilities outputted by a model match empirical observation, the model is *well-calibrated*. For example, if a set of examples $x^{(i)}$ for which $h_\theta(x^{(i)}) \approx 0.7$, around 70% of those examples should have positive labels. In a well-calibrated model, this property holds true at every probability value.

Suppose training set $\{x^{(i)}, y^{(i)}\}_{i=1}^m$ with $x^{(i)} \in \mathbb{R}^{n+1}$ and $y^{(i)} \in \{0, 1\}$. Assume we have an intercept term $x_0^{(i)} = 1$ for all i. Let θ be the maximum likelihood parameters learned after training logistic regression model. In order for model to be well-calibrated, given any range of probabilities (a, b) such that $0 \le a < b \le 1$, and trianing examples $x^{(i)}$ where the model outputs $h_\theta(x^{(i)})$ fall in the range (a, b), the fraction of positives in that set of examples should be equal to the average of the model outputs for those examples. That is,

$$\frac{\sum_{i \in I_{a,b}} P(y^{(i)} = 1 \mid x^{(i)}; \theta)}{|\{i \in I_{a,b}\}|} = \frac{\sum_{i \in I_{a,b}} \mathbb{I}\{y^{(i)} = 1\}}{|\{i \in I_{a,b}\}|},$$

where $P(y^{(i)} = 1 \mid x; \theta) = h_\theta(x) = 1/(1 + \exp(-\theta^T x))$, $I_{a,b} = \{i : h_\theta(x^{(i)}) \in (a, b)\}$.

(a) For the described logistic regression model over the range $(a, b) = (0, 1)$, we want to show the above equality holds. Recall the gradient of log-likelihood

$$\frac{\partial \ell}{\partial \theta_j} = \sum_{i=1}^{m} (y^{(i)} - h_\theta(x^{(i)})) x_j^{(i)}.$$

For a maximum likelihood estimation, $\frac{\partial \ell}{\partial \theta} = 0$. Hence $\frac{\partial \ell}{\partial \theta_0} = 0$. Since $x_0^{(i)} = 1$, we have

$$\sum_{i=1}^{m} y^{(i)} - h_\theta(x^{(i)}) = 0.$$

The desired equality follows immediately since $i \in I_{0,1}$ for all i.

(b) A perfectly calibrated model — that is, the equality holds for any $(a, b) \subset [0, 1]$ — does not imply that the model achieves perfect accuracy. Consider $(a, b) = (\frac{1}{2}, 1)$, the above equality implies

$$\frac{\sum_{i \in I_{a,b}} P(y^{(i)} = 1 \mid x^{(i)}; \theta)}{|\{i \in I_{a,b}\}|} = \frac{\sum_{i \in I_{a,b}} \mathbb{I}\{y^{(i)} = 1\}}{|\{i \in I_{a,b}\}|} < 1.$$

This shows that the model does not have perfect accuracy.

For the converse direction, a perfect accuracy does not imply perfectly calibrated. Consider again $(a, b) = (\frac{1}{2}, 1)$, then we have

$$\frac{\sum_{i \in I_{a,b}} \mathbb{I}\{y^{(i)} = 1\}}{|\{i \in I_{a,b}\}|} = 1 > \frac{\sum_{i \in I_{a,b}} P(y^{(i)} = 1 \mid x^{(i)}; \theta)}{|\{i \in I_{a,b}\}|}.$$

(c) Discuss what effect of L_2 regularization in the logistic regression objective has on model calibration. For L_2 regularization in logistic regression, the gradient becomes

$$\frac{\partial \ell}{\partial \theta_j} = \sum_{i=1}^{m} (y^{(i)} - h_\theta(x^{(i)}))x_j^{(i)} - 2C\theta_j = 0.$$

Hence, the equality does not hold unless $\theta_0 = 0$.

■

The interval $(0, 1)$ is the only range for which logistic regression is guaranteed to be calibrated. When GLM assumptions hold, all ranges $(a, b) \subset [0, 1]$ are well calibrated. In addition, when test set has same distribution and when model has not overfit or underfit, logistic regression are well-calibrated on test data as well. Thus logistic regression is popular when we are interested in level of uncertainty in the model output. △

Problem 3 Bayesian Interpretation of Regularization

In Bayesian statistics, almost every quantity is a random variable. Joint distribution of all the random variables are called *model* (e.g. $p(x, y, \theta)$). Every unknown quantity can be estimated by conditioning the model on all observed quantities. Such conditional distribution $p(\theta \mid x, y)$ is called *posterior distribution*. A consequence of this approach is that we are required to endow a *prior distribution* $p(\theta)$.

In purest Bayesian interpretation, we are required to keep the entire posterior distribution over the parameters all the way until prediction to come up with the *posterior predictive distribution*, and the final prediction will be the EV of the posterior predictive distribution. However, this is computationally very expensive.

The compromise is to estimate a point value of the parameters instead of the full distribution, which is the mode of the posterior distribution. Estimating the mode of posterior distribution is also called *maximum a posteriori estimation* (MAP). That is,

$$\theta_{\text{MAP}} = \arg\max_{\theta} p(\theta \mid x, y).$$

Compare this to the *maximum likelihood estimation* (MLE):

$$\theta_{\text{MLE}} = \arg\max_{\theta} p(y \mid x, \theta).$$

In this problem, explore connections between MAP estimation and common regularization techniques that are applied with MLE estimation. In particular, we will show how choice of prior distribution over θ is equivalent to different kinds of regularization.

(a) Assume that $p(\theta) = p(\theta \mid x)$, we have

$$p(\theta \mid x, y) = \frac{p(y \mid \theta, x)p(\theta \mid x)}{p(y \mid x)} = \frac{p(y \mid \theta, x)p(\theta)}{p(y \mid x)}.$$

Since $p(y \mid x)$ does not depend on θ,

$$\theta_{\text{MAP}} = \underset{\theta}{\text{argmax}}\, p(y \mid \theta, x)p(\theta).$$

Note that the assumption $p(\theta) = p(\theta \mid x)$ will be valid for models such as linear regression where the input x are not explicitly modeled by θ. Note also that this means x and θ are marginally independent, but not conditionally independent when y is given.

(b) Now we show that MAP estimation with a zero-mean Gaussian priori over θ, specifically $\theta \sim \mathcal{N}(0, \eta^2 I)$, is equivalent to applying L_2 regularization with MLE estimation. Specifically, we need to show

$$\theta_{\text{MAP}} = \underset{\theta}{\text{argmin}} - \log p(y \mid x, \theta) + \lambda \|\theta\|_2^2.$$

Recall the definition of multivariate normal, we have

$$\begin{aligned}
\theta_{\text{MAP}} &= \underset{\theta}{\text{argmax}}\, p(\theta \mid x, y) \\
&= \underset{\theta}{\text{argmax}}\, p(y \mid \theta, x)p(\theta) \\
&= \underset{\theta}{\text{argmax}}\, p(y \mid \theta, x) \exp\left(-\frac{\eta^2}{2}\|\theta\|_2^2\right),
\end{aligned}$$

where we have ignored some of the constants. Taking the negative log on both sides, it follows that

$$\theta_{\text{MAP}} = \underset{\theta}{\text{argmin}} - \log p(y \mid x, \theta) + \frac{\eta^2}{2}\|\theta\|_2^2,$$

as desired, where $\lambda = \frac{\eta^2}{2}$.

(c) Now consider a specific instance, a linear regression model given by $y = \theta^T x + \varepsilon$, where $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. Like before, assume a Gaussian prior on this model such that $\theta \sim \mathcal{N}(0, \eta^2 I)$. Let X be the design matrix of all training examples where each row is one example input, and y be the column vector of all the example outputs. We want to derive a closed form expression for θ_{MAP}.

■