Regularization and model selection

February 3, 2021

1 Cross validation and k-fold cross validation

Cross validation: Split train data S into S_train, and S_dev, and select the hypothesis h_i which has smallest error $\hat{\epsilon}_{dev}(h_i)$
k-fold cross validation: Split train data S into k disjoint subsets. For each hypothesis h_i, gets error $\hat{\epsilon}_j(h_i)$, where $j \in 1, .., k$. For each hypothesis h_i, we need to compute the average error across $j \in 1, .., k$, and select the hypothesis with lowest error.

2 Feature Selection

When number of features, n ¿¿ m, the VC dimension of the hypothesis would be $O(n)$, and hence, over-fitting would be a potential problem. In such settings, we can select relevant features subset from 2^n possible subsets.

2.1 Forward search

Begin with an empty feature set, add a feature which gives highest accuracy, incrementally one at a time, and thus, select the best feature subset. It has $O(n^2)$ complexity.

2.2 Mutual Information

One naive way is to select the feature which has highest correlation with the label, i.e. using spearman correlation coefficient. More robust way is to select features with higher Mutual Information (MI).

$$\mathrm{MI}\left(x_i, y\right) = \sum_{x_i \in \{0,1\}} \sum_{y \in \{0,1\}} p\left(x_i, y\right) \log \frac{p\left(x_i, y\right)}{p\left(x_i\right) p(y)}$$

For non binary features and labels, it would be sum over its domains. Mutual Information can also be expressed as Kullback-Leibler (KL) divergence.

$$\mathrm{MI}\left(x_i, y\right) = \mathrm{KL}\left(p\left(x_i, y\right) \| p\left(x_i\right) p(y)\right)$$

If the feature x_i is independent from label y, KL divergence and MI would be zero, as $p(x_i, y) = p(x_i)p(y)$, which represents that the feature is "non-informative".

3 Bayesian statistics and regularization

We have been talking about fitting a parameter using maximum likelihood (ML) approach, and choose parameters according to following:

$$\theta_{\mathrm{ML}} = \arg\max_{\theta} \prod_{i=1}^{m} p\left(y^{(i)} \mid x^{(i)}; \theta\right)$$

Where, θ is estimated using gradient descent, normal equation, or newton's method etc. When we view θ as a constant valued but unknown, it is a frequentist world view.
In a bayesian settings, θ is viewed as an unknown random variable, with a prior distribution $p(\theta)$. So, probability distribution of θ over training data is as below.

$$\begin{aligned} p(\theta \mid S) &= \frac{p(S \mid \theta) p(\theta)}{p(S)} \\ &= \frac{\left(\prod_{i=1}^{m} p\left(y^{(i)} \mid x^{(i)}, \theta\right)\right) p(\theta)}{\int_{\theta} \left(\prod_{i=1}^{m} p\left(y^{(i)} \mid x^{(i)}, \theta\right) p(\theta)\right) d\theta} \end{aligned}$$

Where, $p(y|x,\theta)$ is computed using the hypothesis of the model, i.e. for logistic regression, $p\left(y^{(i)} \mid x^{(i)},\theta\right) = h_\theta\left(x^{(i)}\right)^{y^{(i)}}\left(1-h_\theta\left(x^{(i)}\right)\right)^{\left(1-y^{(i)}\right)}$
Now, probability distribution over labels would be,

$$p(y \mid x, S) = \int_\theta p(y \mid x, \theta) p(\theta \mid S) d\theta$$

And, final prediction would be the expectation,

$$\mathrm{E}[y \mid x, S] = \int_y y p(y \mid x, S) dy$$

This is prediction using "fully bayesian" approach. but it is computationally expensive to compute the posterior distribution requiring integral over high dimensional θ. Instead we can get single point estimate pf θ using **MAP (maximum a posteriori)**, as below,

$$\theta_{\mathrm{MAP}} = \arg\max_\theta \prod_{i=1}^{m} p\left(y^{(i)} \mid x^{(i)}, \theta\right) p(\theta)$$

Note that, this is the same formula as **Maximul Likhlihood** estimate with prior terms at the end.
Generally priors distibution is assumed as, $\theta \sim \mathcal{N}\left(0, \tau^2 I\right)$, with this choise $_{MAP}$ would have smaller norm than that selected by maximum likelihood, and hence bayesian MAP estimate is less susceptible to overfitting than ML estimate of the parameters.