CMPSC 448: Machine Learning and AI
Homework 3

Instruction

This HW only includes theory problems. Please note, You need to submit a report in PDF including the solution of problems.

Logistic Regression

Problem 1. [30 points] Consider a binary training data $\mathcal{S} = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)\}$ where the feature vectors are $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y_i \in \{0, 1\}, i = 1, 2, \ldots, n$. Note that in the lectures we assumed $y_i \in \{-1, +1\}$.

1. Show that
$$\mathbb{P}[y|\boldsymbol{x}; \boldsymbol{w}] = \mathbb{P}[y = 1|\boldsymbol{x}; \boldsymbol{w}]^y \cdot \mathbb{P}[y = 0|\boldsymbol{x}; \boldsymbol{w}]^{(1-y)} \tag{1}$$

2. Following the derivation of logistic regression in lectures, derive the log-likelihood for the training data when the label of each training example is set to be $y_i \in \{0, 1\}$ and sigmoid function is used to covert the linear predictions to probabilities.

3. Then, write down the gradient descent (GD) for obtained optimization problem and discuss the contribution of each training example to updated solution in every iteration of GD. In particular, compare the contribution of a misclassified example with the contribution of a correctly classified example to the gradient.

Decision Trees

Problem 2. [30 points] In this problem, you will investigate building a decision tree for a binary classification problem. The training data is given in Table 1 with 16 instances that will be used to learn a decision tree for predicting whether a mushroom is edible or not based on its attributes (Color, Size, and Shape). Please note the label set is a binary set {Yes, No}.

1. Which attribute would the algorithm choose to use for the root of the tree. Show the details of your calculations. Recall from lectures that if we let $\mathcal{S}$ denote the data set at current node, A denote the feature with values $v \in \mathcal{V}$, H denote the entropy function, and $\mathcal{S}_v$ denote the subset of $\mathcal{S}$ for which the feature A has the value v, the gain of a split along the feature A, denoted InfoGain(S, A) is computed as:
$$\text{InfoGain}(\mathcal{S}, A) = H(\mathcal{S}) - \sum_{v \in \mathcal{V}} \left(\frac{|\mathcal{S}_v|}{|\mathcal{S}|} \right) H(\mathcal{S}_v)$$
That is, we are taking the difference of the entropy before the split, and subtracting off the entropies of each new node after splitting, with an appropriate weight depending on the size of each node.

Instance	Color	Size	Shape	Edible
D1	Yellow	Small	Round	Yes
D2	Yellow	Small	Round	No
D3	Green	Small	Irregular	Yes
D4	Green	Large	Irregular	No
D5	Yellow	Large	Round	Yes
D6	Yellow	Small	Round	Yes
D7	Yellow	Small	Round	Yes
D8	Yellow	Small	Round	Yes
D9	Green	Small	Round	No
D10	Yellow	Large	Round	No
D11	Yellow	Large	Round	Yes
D12	Yellow	Large	Round	No
D13	Yellow	Large	Round	No
D14	Yellow	Large	Round	No
D15	Yellow	Small	Irregular	Yes
D16	Yellow	Large	Irregular	Yes

Table 1: Mushroom data with 16 instances, three categorical features, and binary labels.

2. Draw the full decision tree that would be learned for this data (assume no pruning and you stop splitting a leaf node when all samples in the node belong to the same class, i.e., there is no information gain in splitting the node).

Problem 3. [10 points] Handling real valued (numerical) features is totally different from categorical features in splitting nodes. This problem intends to discuss a simple way to decide good thresholds for splitting based on numerical features. Specifically, when there is a numerical feature in data, an option would be treating all numeric values of feature as discrete, i.e., proceeding exactly as we do with categorical data. What problems may arise when we use a tree derived this way to classify an unseen example?

Support Vector Machines

Problem 4. [30 points] Consider a data set with three data points in $\mathbb{R}^2$

$$\boldsymbol{X} = \begin{bmatrix} 0 & 0 \\ 0 & -1 \\ -2 & 0 \end{bmatrix}, \boldsymbol{y} = \begin{bmatrix} -1 \\ -1 \\ +1 \end{bmatrix}$$

Manually solve the following optimization problem for hard-margin SVM stated as

$$\begin{aligned} \min_{\boldsymbol{w},b} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1, i = 1, 2, \ldots, n \end{aligned}$$

to get the optimal hyperplane $(\boldsymbol{w}_*, b_*)$ and its margin