Cpts 570 Homework 3

Xinyu Liu

November 2021

Analytical

1

a.

$$\text{Prove } (\frac{1}{\sqrt{d}}\sum_{i=1}^{d} x_i - \frac{1}{\sqrt{d}}\sum_{i=1}^{d} z_i)^2 \leq \sum_{i=1}^{d}(x_i - z_i)^2$$

$$(\frac{1}{\sqrt{d}}(\sum_{i=1}^{d} x_i - \sum_{i=1}^{d} z_i))^2 \leq \sum_{i=1}^{d}(x_i - z_i)^2$$

$$\frac{1}{d}(\sum_{i=1}^{d} x_i - \sum_{i=1}^{d} z_i)^2 \leq \sum_{i=1}^{d}(x_i - z_i)^2$$

$$\frac{1}{d}(\sum_{i=1}^{d} x_i - z_i)^2 \leq \sum_{i=1}^{d}(x_i - z_i)^2$$

Jensen's inequality states that when $f(X)$ is convex, then $f(E[X]) \leq E[f(X)]$. In this case, we assume that $X = x_i - z_i$ and $f(X) = X^2$. Apply Jensen's inequality with:

$$f(E[X]) \leq E[f(X)]$$

$$(E[X])^2 \leq E[X^2]$$

$$(\frac{1}{d}\sum_{i=1}^{d} x_i - z_i)^2 \leq \frac{1}{d}\sum_{i=1}^{d}(x_i - z_i)^2$$

$$\frac{1}{d^2}(\sum_{i=1}^{d} x_i - z_i)^2 \leq \frac{1}{d}\sum_{i=1}^{d}(x_i - z_i)^2$$

$$\frac{1}{d}(\sum_{i=1}^{d} x_i - z_i)^2 \leq \sum_{i=1}^{d}(x_i - z_i)^2$$

b. This property gives a lower bound of Euclidean distance by only square once. Bigger the d, closer the result will be.

2

LSH stores data points in a data structure using a family of hash functions such that points that have close Euclidean distance have higher chance of collision. Thus, when finding the nearest points, it only needs to look for data points that are closer in the memory.

3

It's possible to construct a decision tree from a set of rules. It's hard to tell which rule is root, therefore it's much easier to build the tree from leaf nodes, which are pure nodes. Then find rules that consists of 2 leaf nodes and so on until there is no rule left.

4

In this experiment, it shows that discriminative model has a lower asymptotic error than generative model, but it takes a lot longer to converge than the latter.

5

a. Assuming the data fits Naive Bayes assumption, generative model and discriminative models should have similar results. However, Naive Bayes converges faster.

b. Logistic regression will perform better because naive bayes posterior belief is not correct.

c. No, Naive Bayes doesn't have learned parameters.

d. Yes, we could use coefficients of the classifier to find P(X)

6

We want to optimize the algorithm to have higher accuracy on test data without overfitting. Many training methods are NP-hard, but we could simply use undercomputing to solve the problem.

7

The article compares 5 methods with 2 algorithms and 3 datasets. It shows that McNemar's test have the best result for single run and 5x2cv for multiple runs.

8

a. Chess. We need to know from current state, what move(action) will gain the best reward.

b.

$$V^0(s) = 0, \forall s$$

$$V^k(s) = maxR(s,a) + \sum_{s'} T(s,a,s')V^{k-1}(s)$$

c. We keep track of (s,a) with a fake state q(s,a), then rewards function would look like $R(q(s,a)) = R(s,a)$, and transition function $T(s,a,s')$ would become $T(s,a,q(s,a))$.

9

$S' = S$

$A' = A$

$R'(s, s_{k-1}, ..., s_1) = R(s)$

$T'((s, s_{k-1}, ..., s_1), a, s') = Pr(s'|a, s_{k-1}, ..., s_1)$ or 0

Policy $\pi'(s_{k-1}, ..., s_1) = \pi(s_{k-1}, ..., s_1)$

K-order MDP has an equivalent M from normal MDP.

10

$R(s,a)$

$$V^k(s) = maxR(s,a) + \beta \sum T(s,a,s')V^{k-1}(s')$$

$R(s,a,s')$

$$V^k(s) = max \sum T(s,a,s')R(s,a) + \beta V^{k-1}(s')I'm$$

11

a.

$V_0 = R(s_0) + \beta V(s_1)$

$V_1 = R(s_1) + \beta V(s_1)$

Given $\beta = 1$, we get

$V_0 = V_1$

$V_1 = 1 + V_2$

No solution

b.

$V_0 = R(s_0) + \beta V(s_1)$

$V_1 = R(s_1) + \beta V(s_1)$

Given $\beta = 0.9$, we get

$V_0 = 0.9V_1$

$V_1 = 1 + 0.9V_2$

$V_0 = 9$, $V_1 = 10$

Programming and Empirical

Train: 0.953416149068323

Test: 0.8217821782178217

Below are results from sklearn:

Train sklearn: 0.9596273291925466

Test sklearn: 0.8316831683168316