Pattern Recognition and Machine Learning: Homework 1

Qingru Hu 2020012996

February 26, 2023

Problem 1

T_1: Playing Go game

E: games played with its opponent

P: the winning rate after a certain amount of games

T_2: Making medical decisions by CT images.

E: all the CT images that the model has seen through training

P: classification accuracy

T_3: Controlling a robot walk.

E: all the pictures or other forms of information it gathers from the environment during controlling the robot walking 训练时间

P: the reward function to measure control quality, which, in example, may encourage moving forward and discourage moving astray

T_4: Autonomous driving.

E: the pictures or other forms of information it gathers during driving 行驶的路程数

P: the reward function to measure driving quality, which, in example, may encourage driving on the track and discourage driving astray

T_5: Generate realistic images.

E: all the realistic images it has seen 判别器能区分的准确率

P: scores of how realistic the pictures it generates are, evaluated by human

T_6: ChatGPT chatting with human.

E: all the chats it had with human

P: scores of the quality of these chats (like the fluentness, the use of words and so on), evaluated by the person who chats with ChatGPT

Problem 2

2.1

$$\text{Sensitivity} = \frac{\text{TP}}{\#\text{ Actual Positives}} = \frac{999}{1000} = 0.999$$

$$\text{Specificity} = \frac{\text{TN}}{\#\text{ Actual Negatives}} = \frac{980}{1000} = 0.980$$

2.2

Let A be the event "getting a cancer", and B be the event "tested positive". From the question we know that $P(A) = 0.1\% = 0.001$. From 2.1 we know that $P(B|A) = 0.999$. Besides, the probability of a person being tested positive when he actually does not get the disease is:

$$P(B|\bar{A}) = \frac{\text{FP}}{\#\text{ Actual Negatives}} = \frac{20}{1000} = 0.020$$

From the Bayesian equation we can get the probability of Alice getting the disease after being tested positive is:

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)} = \frac{P(A)P(B|A)}{P(A)P(B|A) + P(\bar{A})P(B|\bar{A})}$$

$$P(A|B) = \frac{0.001 \times 0.999}{0.001 \times 0.999 + 0.999 \times 0.020} = 0.0476$$

2.3 错误率要比人群中患病的概率再低一点

The precision α is:

$$\alpha = \frac{\text{TP}}{\#\text{ Estimated Positives}}$$

, which is close to $P(A|B)$ when the testing sample is very large. If the prior probability for getting a cancer in population is p, then α can be caculated as:

$$\alpha = P(A|B) = \frac{P(A)P(B|A)}{P(B)} = \frac{P(A)P(B|A)}{P(A)P(B|A) + P(\bar{A})P(B|\bar{A})}$$

Given the currrent testing ability, if we want α, that is the chance of Alice really got a cancer, to be more than 99%, then p must satisfy:

$$\frac{p \times 0.999}{p \times 0.999 + (1-p) \times 0.020} > 99\%$$

$$p > 0.665$$

That is a horrible probability for getting a cancer in population. Therefore, even if the sensitivity and specificity of the testing techique is very prospective, the precision of the test can still be very low due to the extremely small prior probability.

Problem 3

3.1

Since the volume V_c of a high-dimensional unit cube is always 1, the probability P of sampling a data $x \sim \nu$ that successfully becomes a reliable neighbor for query point $x_c = (0.5, 0.5, ..., 0.5)^\top \in \mathbb{R}^d$ is just:

$$P = \frac{V_d(\epsilon)}{V_c} = V_d(0.2) = \frac{(\sqrt{\pi} \times 0.2)^d}{\Gamma(d/2+1)}$$

For d equals 2, 5, and 10, the relevant probability P is listed in Tab.1.

d	2	5	10
P	1.257e-01	1.684e-03	2.611e-07

Table 1: P for $d = 2, 5, 10$

3.2

Because we keep sampling data points to create a dataset until we get a reliable neighbor for x_c in $\mathbb{R}^d$, the number of sampling times N obeys the geometric distribution $N \sim G(P)$, where P is the same as in 3.1.

Therefore, from probability theory, we can know that the expectation value of N is simply $\frac{1}{P}$. Calculate P as in 3.1 and plot $N/logN \sim d$ in Fig.1. (Here log refers to log10.)

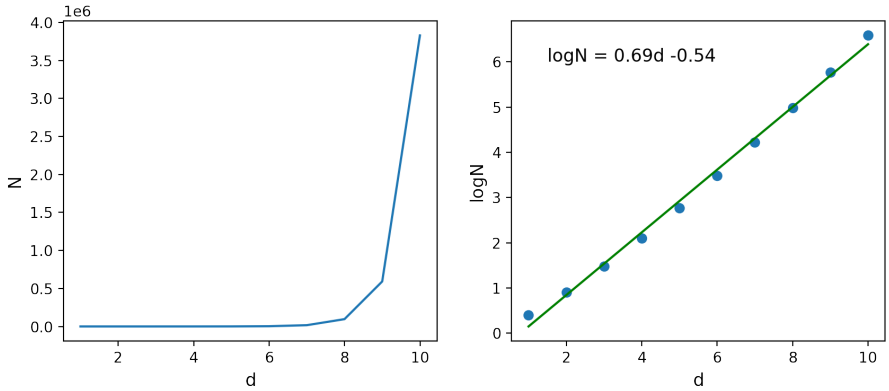

Figure 1: Plot of N/logN over d and the best linear fit

Using the least square algorithm, the best linear fit of $\log N \sim d$ is

$$\log N = 0.69d - 0.54$$

Thus, $\log N$ for $d = 100$ will be 68.769. It tells us that the num of data points sampled before we get a reliable neighbor for x_c increases exponentially with the dimension of space. It can be safely inferred that all the data points become very far from each other in very high dimensions. It is a vivid demonstration of the curse of high dimensionality.