Question 2. (6 marks)

(1) Suppose a bank classifies customers as either good or bad credit risks. On the basis of extensive historical data, the bank has observed that 1% of good credit risks and 10% of bad credit risks overdraw their account in any given month. A new customer opens a cheque account at this bank. On the basis of a check with a credit bureau, the bank believe that there is a 70% chance that the customer will turn out to be a good credit risk. Suppose that this customer's account is overdrawn in the first month. How does this alter the bank's opinion of this customer's creditworthiness? (3 marks)

This problem can be solved using Bayes' theorem, which is a fundamental principle in probability theory and statistics that describes how to update the probabilities of hypotheses (in this case, the hypothesis that the customer is a good or bad credit risk) when given evidence (in this case, the evidence is that the customer overdrew their account).

Here is how to apply Bayes' theorem to this problem. First, let's define some events:

- A: The customer is a good credit risk.
- B: The customer is a bad credit risk.
- C: The customer overdraws their account.

The bank has initially estimated the probabilities as follows:

- P(A) = 0.70 (the probability that the customer is a good credit risk)
- P(B) = 1 - P(A) = 0.30 (the probability that the customer is a bad credit risk)
- P(C|A) = 0.01 (the probability that a good credit risk customer overdraws their account)
- P(C|B) = 0.10 (the probability that a bad credit risk customer overdraws their account)

The bank wants to find P(A|C) and P(B|C), the probabilities that the customer is a good or bad credit risk given that they overdraw their account. According to Bayes' theorem:

- P(A|C) = P(C|A) * P(A) / P(C)
- P(B|C) = P(C|B) * P(B) / P(C)

First, we need to find P(C), the total probability that the customer overdraws their account, which can be obtained using the law of total probability:

- P(C) = P(C and A) + P(C and B) = P(C|A)P(A) + P(C|B)P(B) = 0.01*0.70* + *0.10*0.30 = 0.037.

Now, we can substitute P(C) into the Bayes' theorem equations:

- P(A|C) = 0.01 * 0.70 / 0.037 ≈ 0.189
- P(B|C) = 0.10 * 0.30 / 0.037 ≈ 0.811

So, after the customer overdraws their account, the bank's updated belief is that there is approximately an 18.9% chance that the customer is a good credit risk and an 81.1% chance that they are a bad credit risk. In other words, the event of overdrawing the account significantly alters the bank's opinion of this customer's creditworthiness, making it much more likely in their view that the

customer is a bad credit risk.

(2) The logistic sigmoid function is defined as:

$\sigma(a)$ =1/ (1 + exp(− a))

and the 'tanh' function is denoted as:

$tanh$(a) = 2σ(2a) − 1

Prove that a general linear combination of logistic sigmoid functions of the form

$y(x, \boldsymbol{w})$ = $\omega 0$ + ∑ωj*σ((x − μi)/s)

is equivalent to a linear combination of 'tanh' functions of the form

$y(x, \boldsymbol{w})$ = u_0 + ∑u_j*tanh((x − μi)/s)

and **find** expressions to relate the new parameters {$u1$, … , uM} to the original

parameters {$\omega 1$, … , ωM}.

Answer

To prove this, let's first observe the relationship between the logistic sigmoid function and the tanh function. As given, the tanh function can be written in terms of the logistic sigmoid function:

tanh(a) = 2σ(2a) - 1

This can be rewritten as:

σ(a) = 0.5(tanh(a/2) + 1) [let's call this Equation 1]

Now, let's consider a general linear combination of logistic sigmoid functions:

y(x, w) = w0 + ∑wjσ((x - μj) / s)

Substituting Equation 1 into this gives:

y(x, w) = w0 + ∑wj*0.5(tanh((x - μj) / 2s) + 1)

Let's express this as a linear combination of tanh functions:

y(x, w) = (w0 + ∑0.5wj) + ∑0.5wj*tanh((x - μj) / 2s)

This has the form of:

y(x, u) = u0 + ∑uj*tanh((x - μi) / s)

By comparing the two, it's clear that:

u0 = w0 + ∑0.5wj

uj = 0.5wj

This means that you can express the parameters of the linear combination of tanh functions {u1, ..., uM} in terms of the original parameters {w1, ..., wM} as:

u0 = w0 + ∑0.5wj (j from 1 to M)

uj = 0.5wj (j from 1 to M)

So, we have proven that a general linear combination of logistic sigmoid functions can be expressed as a linear combination of tanh functions, and we have found expressions to relate the parameters of the two.

Question 3. (10 marks)
This question explores the topic classifier performance evaluation.
(1) Explain the significance of area under the receiver operating characteristics (ROC) curve as it pertains to classifier performance measure. (3 marks)
Answer: The ROC curve is plotted with "True Positive Rate" (TPR, or sensitivity) on the y-axis against the "False Positive Rate" (FPR, or 1 - specificity) on the x-axis. The curve plots the tradeoff between sensitivity and specificity for every possible cut-off.The Area Under the ROC curve (AUC-ROC) provides a single scalar value that represents the overall performance of the model across all possible classification thresholds.Values for AUC range from 0 to 1. A model with a score of 0.5 has no discriminative power and is as good as random guessing, while a model with an AUC of 1 is a perfect model. Typically, a value above 0.7 is considered acceptable,

(2) Suppose that we have a data set and each data item describes objects that can be classified as either belonging to class $\omega 1$ or $\omega 2$. Two classifiers (X andY) were designed using 500 training samples from the data set. Some samples were set aside for testing and performance measurement. The result of the test is shown in Table 1 below.
Calculate the respective AUC for the ROC of the two classifiers and comment on their relative performance.
Also calculate the apparent error rates of the two classifiers and comment on the implication.
The area, A, under the ROC curve (AUC) is given by
A = 1/(N_1*N_2){ S_0 - 1/2*N1*(N_1+1) }
where $N1$ and $N2$ are the number of samples from $\omega 1$ and $\omega 2$ respectively. S_0 is the sum of the ranks of class $\omega 1$ test patterns. (3 marks)

Table1:

Test sample	Classifier X	Classifier Y	True Class	Ranking of Likelihood ratios
$x1$	$\omega 2$	$\omega 1$	$\omega 2$	1
$x2$	$\omega 2$	$\omega 2$	$\omega 2$	2
$x3$	$\omega 2$	$\omega 2$	$\omega 2$	3
$x4$	$\omega 2$	$\omega 2$	$\omega 2$	4
$x5$	$\omega 1$	$\omega 2$	$\omega 2$	5
$x6$	$\omega 2$	$\omega 1$	$\omega 1$	6
$x7$	$\omega 1$	$\omega 1$	$\omega 1$	7
$x8$	$\omega 1$	$\omega 1$	$\omega 1$	8
$x9$	$\omega 1$	$\omega 1$	$\omega 1$	9
$x10$	$\omega 1$	$\omega 2$	$\omega 1$	10

Answer:
Based on the table you provided, we can calculate the performance of Classifier X and Y. The apparent error rates can be calculated as the fraction of misclassified instances. The AUC for the ROC curve can be calculated using the formula you provided.

1. **Classifier X**:
    - The true class of $x1$, $x2$, $x3$, $x4$, and $x5$ is $\omega2$, and Classifier X correctly classified all except $x5$.
    - The true class of $x6$, $x7$, $x8$, $x9$, and $x10$ is $\omega1$, and Classifier X correctly classified all except $x6$.
    - Therefore, the apparent error rate for Classifier X is 2/10 = 20%.

    To calculate AUC for Classifier X:

    - S0 is the sum of the ranks of class $\omega1$ test patterns. For Classifier X, this would be 6+7+8+9+10 = 40.
    - $N1$ *and* $N2$ are the number of samples from $\omega1$ and $\omega2$ respectively. In this case, $N1$*=5 and* $N2$=5.
    - Plug these into the formula: A = 1/(5*5){ 40 - 1/2*5*(5+1) } = 0.6
2. **Classifier Y**:
    - The true class of $x1$, $x2$, $x3$, $x4$, and $x5$ is $\omega2$, and Classifier Y correctly classified all.
    - The true class of $x6$, $x7$, $x8$, $x9$, and $x10$ is $\omega1$, and Classifier Y correctly classified all except $x10$.
    - Therefore, the apparent error rate for Classifier Y is 1/10 = 10%.

    To calculate AUC for Classifier Y:

    - S0 for Classifier Y would be 1+6+7+8+9 = 31.
    - $N1$ *and* $N2$ are the same as for Classifier X.
    - Plug these into the formula: A = 1/(5*5){ 31 - 1/2*5*(5+1) } = 0.2

In summary, Classifier X has a higher AUC-ROC (0.6 vs 0.2), meaning it is better at ranking positive instances higher than negative instances, despite having a higher error rate (20% vs 10%). This discrepancy could suggest that Classifier X made more errors on harder-to-classify instances. On the other hand, Classifier Y has a lower error rate, but it's worse at ranking instances, suggesting it may be overfitting to the negative class. The optimal classifier would depend on the specific cost trade-off between false positives and false negatives.

(3) The McNemar's or Gillick test can be used to compare two classifiers with respect to the classification errors they make. Compute the statistic z and state its implication (with respect to the null hypothesis) for the data shown in Table 1. Also compare your result with that of part (2). The z statistic is computed using the following formula:
z = (|n01 − n10| − 1)/(2次根号(n01 + n10))
where
n01 = number of samples misclassified by but not by)
n10= number of samples misclassified by) but not by

answer：
Based on the information you provided, the McNemar's test can be used to compare the performance of Classifier X and Classifier Y.

The McNemar's test is used to determine whether the row and column marginal frequencies in a 2x2 contingency table are equal (i.e., the null hypothesis is that they are equal).

The test statistic is computed as:

z = (|n01 − n10| − 1)/sqrt(n01 + n10)

Where:

- n01 = number of samples misclassified by Classifier X but not by Classifier Y
- n10 = number of samples misclassified by Classifier Y but not by Classifier X

From Table 1, we see:

- n01 = number of samples misclassified by X but not by Y = 1 (x6)
- n10 = number of samples misclassified by Y but not by X = 1 (x5)

So, substituting these values into the formula, we get:

z = (|1 - 1| - 1)/sqrt(1 + 1) = (-1)/sqrt(2) ≈ -0.707

The value of the test statistic, z, falls within the range of -1.96 to 1.96 (assuming a 95% confidence level), so we cannot reject the null hypothesis, implying that there's no significant difference between the errors made by the two classifiers.

Comparing this with the result from part (2), we can see that although Classifier X has a higher error rate and a higher AUC-ROC than Classifier Y, the difference in the types of errors they make is not statistically significant according to the McNemar’s test. This indicates that the two classifiers perform similarly on this particular dataset.

Four measurements are made on each of a random sample of 500 animals. The first three variables were different linear dimensions, measured in centimeters, while the fourth variable was the weight of the animal measured in grams. The sample covariance was calculated and its four eigenvalues found to be 14.1, 4.5, 1.2, and 0.2. The eigenvectors corresponding to the first and second eigenvalues were:
$u^{1}t$ = *[0.39 0.42 0.44 0.69]*
$u^{2}t$ = [0.40 0.39 0.42 −0.72]
where t denotes transpose.
(1) What is percentage of the variance in the original data accounted for by the first two principal components? Describe the results? (3 marks)
(2) Comment on the use of these two principal values, instead of the original four measurements, as input in a machine learning algorithm. (2 marks)

answer:
(1) The percentage of variance in the original data accounted for by the first two principal components is calculated by adding the eigenvalues corresponding to these components and dividing by the sum of all eigenvalues. This sum is then multiplied by 100 to express it as a percentage.

Eigenvalues: 14.1, 4.5, 1.2, 0.2

Sum of all eigenvalues = 14.1 + 4.5 + 1.2 + 0.2 = 20

Sum of the first two eigenvalues = 14.1 + 4.5 = 18.6

Percentage of variance explained by the first two principal components = (18.6 / 20) * 100 = 93%

The first two principal components account for 93% of the variance in the original data. This implies that the first two principal components capture the majority of the information in the original four dimensions. The data seems to be mostly spread along these two directions.

(2) Using these two principal values, instead of the original four measurements, as input in a machine learning algorithm has several potential advantages:

- **Dimensionality Reduction**: Reducing the dimensionality of the data from four dimensions to two can help to speed up training times and reduce the computational resources required.
- **Noise Reduction**: PCA can also help to eliminate noise in the data by focusing on the components of the data with the largest variance, which are often the most informative. This can help to improve the accuracy and generalization ability of a machine learning algorithm.
- **Visualization**: Reducing the data to two dimensions can also make it possible to visualize the data, which can provide useful insights.

However, it's important to keep in mind that while the first two principal components account for 93% of the variance, they don't capture all the information in the original data. The remaining 7% could potentially be important for some applications, and using only the first two principal components could lead to a loss of important information in these cases.

Question 5. (9 marks)
(1) Explain the purpose of clustering and when it shall be used in machine learning. (2 marks)
The process of dividing a collection of physical or abstract objects
into multiple classes composed of similar objects is called clustering. The cluster generated by clustering is a set of data objects, which
are similar to the objects in the same cluster and different from the
objects in other clusters.
when It is an unsupervised learning in Machine learning.(no labels)

(2) Describe the procedure of k-means clustering. (2 marks)

K-means is a super simple clustering method. The main reason for
its simplicity is that it only needs to set a k value when using it
(setting needs to aggregate data into several categories)

1. Choose k points as the center of each category.
2. Calculate the distance between all other points and the center k
   and classify the related points into the same category.

3. According to the points of each category, recalculate the center of each category.
4. Repeat the step 2 and 3 until the center stop changing.

(3) Describe the way(s) you can use to select the k value of k-means clustering. (1 mark)

1. quick choose:
2. Elbow method:
3. Gap statistic
4. Silhouette Coefficient
5. (4) Why we usually run the k-means clustering multiple times for a given k value? (1 mark)
   Because the k is decided by ourselves, Repeated computation prevents local optimization.
   (5) Explain back propagation algorithm, providing the necessary equations for the weight updates. (3 marks)

he backpropagation algorithm is used in training neural networks and involves the following steps:

Forward Propagation: Inputs are passed through the network, layer by layer, until the output layer is reached.
Compute Loss: The error (or loss) of the network is computed by comparing the predicted output to the true output.
Backward Propagation: The error is propagated back through the network. The gradient of the loss function with respect to the network weights is calculated using the chain rule of differentiation.
Update Weights: The weights are updated in the direction that minimizes the loss. This is typically done using a method such as stochastic gradient descent (SGD).
The general weight update equation using SGD is:

$w_{ij} = w_{ij} - \eta * (\partial \text{Loss} / \partial w_{ij})$

where:

w_{ij} is the weight between the i-th and j-th neuron,
η is the learning rate,
$\partial \text{Loss} / \partial w_{ij}$ is the gradient of the loss with respect to the weight w_{ij}.