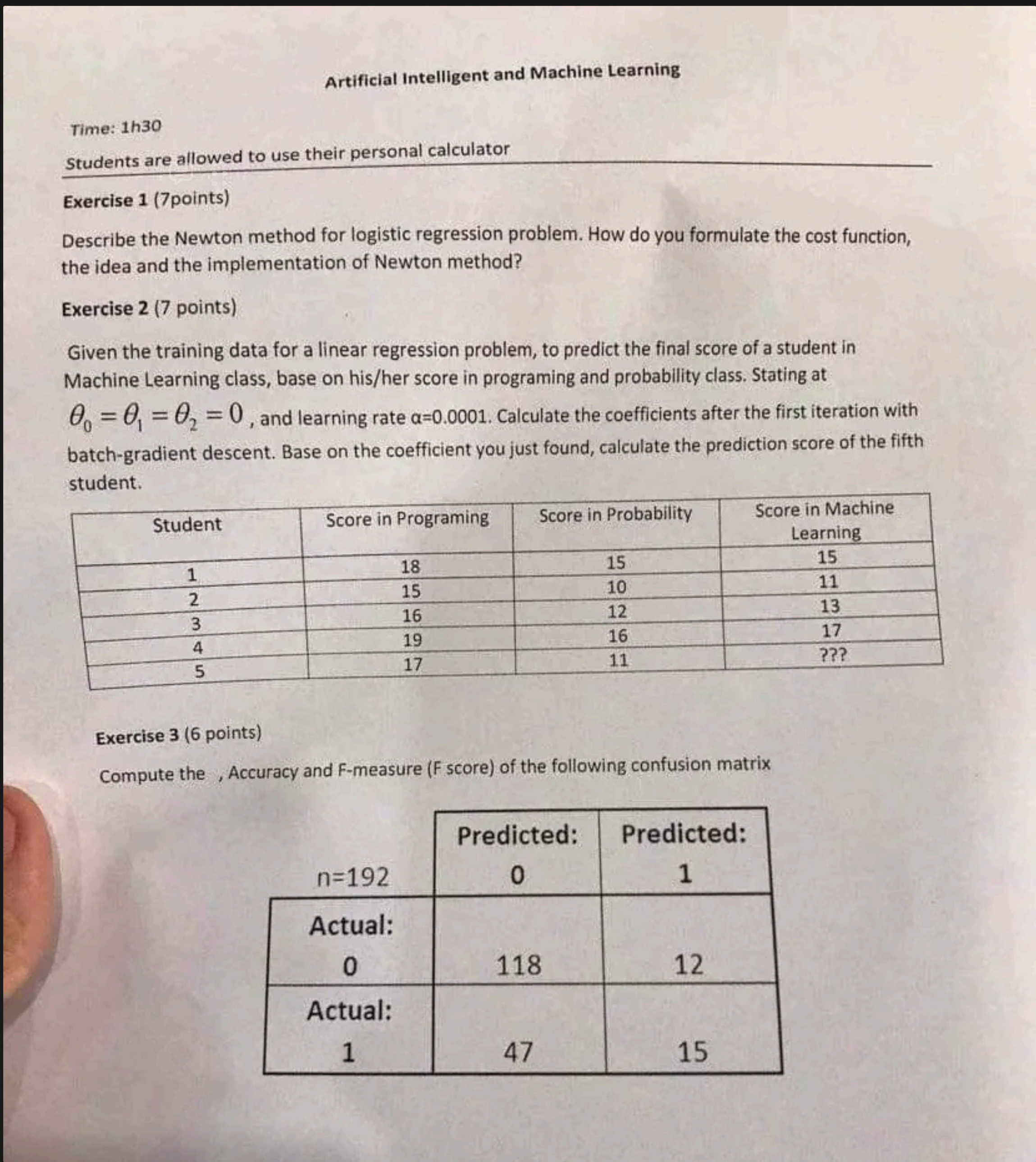
2022 Gen 11 Final

▼ Source



Artificial Intelligent and Machine Learning

Time: 1h30

Students are allowed to use their personal calculator

Exercise 1 (7points)

Describe the Newton method for logistic regression problem. How do you formulate the cost function, the idea and the implementation of Newton method?

Exercise 2 (7 points)

Given the training data for a linear regression problem, to predict the final score of a student in Machine Learning class, base on his/her score in programing and probability class. Stating at $\theta_0 = \theta_1 = \theta_2 = 0$, and learning rate α=0.0001. Calculate the coefficients after the first iteration with batch-gradient descent. Base on the coefficient you just found, calculate the prediction score of the fifth student.

Student	Score in Programing	Score in Probability	Score in Machine Learning
1	18	15	15
2	15	10	11
3	16	12	13
4	19	16	17
5	17	11	???

Exercise 3 (6 points)

Compute the , Accuracy and F-measure (F score) of the following confusion matrix

n=192	Predicted: 0	Predicted: 1
Actual: 0	118	12
Actual: 1	47	15

Exercise 1

Describe the Newton method for logistic regression problem. How do you formulate the cost function, the idea and the implementation of Newton method?

✅ Logistic regression is a statistical model used in machine learning for binary classification problems. It uses the concept of the log-likelihood of the Bernoulli distribution and a transformation called the sigmoid function.

The cost function in logistic regression is formulated using the concept of maximum likelihood estimation. The idea is to find the parameters that maximize the likelihood of making the observations given the parameters.

The sigmoid function, also known as the logistic function, is used to transform the linear combination of features and weights into a probability between 0 and 1. It is defined as:

$$h(x) = \frac{1}{1 + e^{-z}}$$

where $z = \theta_1 x + \theta_2$.

Newton's method, also known as Newton-Raphson method, is an optimization algorithm used to find the maximum (or minimum) of a function. In the context of logistic regression, it's used to find the parameters that maximize the log-likelihood.

The implementation of Newton's method involves iteratively updating the parameters using the gradient (first derivative) and the Hessian (matrix of second derivatives) until convergence. The update rule in Newton's method is:

$$\theta := \theta - H^{-1} \nabla_\theta J(\theta)$$

where H is the Hessian matrix, $\nabla_\theta J(\theta)$ is the gradient of the cost function $J(\theta)$ with respect to θ, and H^{-1} is the inverse of the Hessian matrix.

This method has been proven to be more efficient than gradient descent for logistic regression as it converges faster. However, it requires calculating and inverting the Hessian matrix in each iteration, which can be computationally expensive for large datasets.

Exercise 2

Given the training data for a linear regression problem, to predict the final score of a student in Machine Class, based on his/her score in programming probability class. Starting at $\theta_0 = \theta_1 = \theta_2 = 0$, and learning rate $\alpha = 0.0001$. Calculate the coefficient after the first iteration with batch-gradient descent. Base on the coefficient you just found, calculate the prediction score the 5th student.

Student	Score in Programming	Score in Probability	Score in Machine Learning
1	18	15	15
2	15	10	11
3	16	12	13
4	19	16	17
5	17	11	

The batch gradient descent algorithm updates the parameters in the direction of the negative gradient of the cost function. The update equations for the parameters are as follows:

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

where:

- m is the number of training examples,
- $h_\theta(x)$ is the hypothesis function, defined as $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$,
- $x^{(i)}$ is the i-th training example,
- $y^{(i)}$ is the actual output of the i-th training example,
- α is the learning rate.

For this problem, we'll use 1 for male and 0 for female. Let's calculate the new coefficients after one iteration:

1. For θ_0:

$$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})$$

1. For θ_1 (Score in Programming):

$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_1^{(i)}$$

1. For θ_2 (Score in Probability):

$$\theta_2 := \theta_2 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_2^{(i)}$$

You can substitute the given values into these equations to get the new coefficients after one iteration. Note that you'll need to calculate $h_\theta(x)$ for each training example in each sum.

After calculating the new coefficients, you can use them to predict the score of the 5th student in Machine Learning class by substituting their scores in Programming and Probability into the hypothesis function:

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

Exercise 3

Compute the Accuracy and F-measure (F-score) of the following confusion matrix.

n = 192	Predicted: 0	Predicted: 1
Actual: 0	118	12
Actual: 1	47	15

The confusion matrix is a useful tool for evaluating the performance of a classification model. It contains information about actual and predicted classifications done by the classifier. Here's how we can calculate Accuracy and F-measure (F-score) from the given confusion matrix:

n = 192	Predicted: 0	Predicted: 1