

▼ Topic02a : Prelim Problem Set I

▼ Case 1

Represent the following representations into its vectorized form using LaTeX.

Problem 1.a. System of Linear Equations

$$\begin{cases} -y + z = \frac{1}{32} \\ \frac{1}{2}x - 2y = 0 \\ -x + \frac{3}{7}z = \frac{4}{5} \end{cases}$$

Problem 1.b. Linear Combination

$$\cos(\theta)\hat{i} + \sin(\theta)\hat{j} - \csc(2\theta)\hat{k}$$

Problem 1.c. Scenario

A conference has 200 student attendees, 45 professionals, and has 15 members of the panel. There is a team of 40 people on the organizing committee. Represent the percent composition of each attendee type of the conference in matrix form.

Express your answers in LaTeX in the answer area.

```
1 import numpy as np
2 import math
```

▼ Problem 1.a: System of Linear Equations

$$A = \begin{bmatrix} 0 & -1 & 1 & \frac{1}{32} \\ \frac{1}{2} & -2 & 0 & 0 \\ -1 & 0 & \frac{3}{7} & \frac{4}{5} \end{bmatrix}$$

▼ Problem 1.b: Linear Combination

$$B = \begin{bmatrix} \sin(\theta) \\ \cos(\theta) \\ -\csc(2\theta) \end{bmatrix}$$

▼ Problem 1.c: Scenario

$$Percentage = \frac{No. of Attendee type \times 100}{Total Number of Attendees}$$

$$Students = \frac{200 \times 100}{300} = 66.67\%$$

$$Professionals = \frac{45 \times 100}{300} = 15.00\%$$

$$Panel\ Members = \frac{15 \times 100}{300} = 13.33\%$$

$$Organizing\ Committee = \frac{40 \times 100}{300} = 5.00\%$$

▼ **Matrix form of percentage composition of each attendee type**

$$Attendees = \begin{bmatrix} 66.67\% \\ 15.00\% \\ 13.33\% \\ 5.00\% \end{bmatrix}$$

▼ **Case 2**

**Problem 2.a: Vector Magnitude**

The magnitude of a vector is usually computed as:

$$||v|| = \sqrt{a_0^2 + a_1^2 + \dots + a_n^2}$$

Whereas  $v$  is any vector and  $a_k$  are its elements wherein  $k$  is the size of  $v$ . Re-formulate  $||v||$  as a function of an inner product. Further discuss this concept and provide your user-defined function.

**Problem 2.b: Angle Between Vectors**

Inner products can also be related to the Law of Cosines. The property suggests that:

$$u \cdot v = ||u|| \cdot ||v|| \cos(\theta)$$

Whereas  $u$  and  $v$  are vectors that have the same sizes and  $\theta$  is the angle between  $u$  and  $v$ .

Explain the behavior of the dot product when the two vectors are perpendicular and when they are parallel.

▼ **Problem 2.a: Vector Magnitude**

An inner product space makes a **norm/ length**, that is, a notion of length of a vector.

Let  $V$  be an inner product space and let  $v \in V$ . The **norm/length** of  $v$  is denoted  $||v||$  and is defined by

$$||v|| = \sqrt{\langle v, v \rangle}$$

the square root is defined by the property (i) of the inner product.

**Example:** Find  $||vect||$  where,

$$vect = \begin{bmatrix} 2 & 4 \\ -3 & 7 \end{bmatrix}$$

**Solution:**

$$\begin{aligned} ||vect|| &= \sqrt{\langle vect, vect \rangle} \\ &= \sqrt{\langle \begin{bmatrix} 2 & 4 \\ -3 & 7 \end{bmatrix}, \begin{bmatrix} 2 & 4 \\ -3 & 7 \end{bmatrix} \rangle} \\ &= \sqrt{(2)(2) + (4)(4) + (-3)(-3) + (7)(7)} \\ &= \sqrt{4 + 16 + 9 + 49} \\ vect &= \sqrt{78} \text{ or } 8.831760866327848 \end{aligned}$$

```
1 vect = np.array([2,4,-3,7])
2 vect_2 = np.array([2,4,-3,7])
3
4 answer = np.matmul(vect,vect_2)
5 answer
```

78

```
1 magnitude = np.sqrt(np.matmul(vect,vect_2))
2 print('The magnitude is the square root of', answer, 'or', magnitude)
```

The magnitude is the square root of 78 or 8.831760866327848

▼ **Problem 2.b: Angle Between Vectors**

The **dot product** is a method for determining when two vectors are perpendicular and it will give another method for determining when two vectors are parallel. The term **orthogonal** is also used in place of perpendicular.

If two vectors are **orthogonal (perpendicular)**, then, the angle between them is 90 degrees. From the Law of Cosines theorem, this tells that if two vectors are orthogonal then,

$$u \cdot v = 0$$

With that being said, two vectors are considered to be orthogonal if their dot product **equals to 0**.

Additionally, if two vectors are **parallel**, then the angle between them is either 0 degrees—pointing in the same direction, or 180 degrees—pointing in the opposite direction. Again, using the Law of Cosine theorem, this would mean that one of the following would have to be *true*.

$$\begin{aligned} u \cdot v &= ||u|| \cdot ||v|| \ (\theta = 0^\circ) \\ &\text{OR} \\ u \cdot v &= -||u|| \cdot ||v|| \ (\theta = 180^\circ) \end{aligned}$$

**Example 1:**

$$u = \langle 3, 4 \rangle, v = \langle -8, 6 \rangle$$

**Solution:**

$$\begin{aligned} &\langle 3, 4 \rangle \cdot \langle -8, 6 \rangle \\ &= (3 \times (-8)) + 4 \times 6 \end{aligned}$$

$$\begin{aligned} &= -24 + 24 \\ &= 0 \end{aligned}$$

Therefore, this makes the vector **perpendicular**.

**Example 2:**

$$u = \langle -9, 18 \rangle, v = \langle -3, 6 \rangle$$

**Solution:**

$$u = -3v$$

Making the vector **parallel**:

$$3\langle -3, 6 \rangle = \langle -9, 18 \rangle$$

▼ **Case 3**

For the final cases analysis we will be looking at series of equations building up a single feed-forward computation of a logistic regression. The case will not require you to learn fully what is logistic regression.

$$X = \begin{bmatrix} \text{---} (x^{(1)})^T \text{---} \\ \text{---} (x^{(2)})^T \text{---} \\ \vdots \\ \text{---} (x^{(m)})^T \text{---} \end{bmatrix}, Y = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix}, \text{ and } \theta = \begin{bmatrix} \theta^{(1)} \\ \theta^{(2)} \\ \vdots \\ \theta^{(m)} \end{bmatrix}$$

The dataset  $X$  has  $m$  entries with  $n$  features while  $Y$  is the vector containing the ground truths of the entries of  $X$ , and  $\theta$  are the parameters or weights of the vectors. We first compute the vector product of the dataset and the parameters as:

$$z = x^{(i)} \theta^{(i)} = X \cdot \theta$$

Eq. 3.1

We then solve for the hypothesis of the logistic regression algorithm as:

$$h_{\theta}(x) = g(z)$$

Eq. 3.2

Where  $g$  is an activation function that maps the values of the hypothesis vector between a range of 0 and 1. We computed the activation as a sigmoid function:

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

Eq. 3.3

Finally we compute the loss of the logistic regression algorithm using  $J$ . Whereas  $J(\theta)$  is a function that computes the logistic loss of the hypothesis with respect to the ground truths  $y$ . it is then computed as:

$$J(\theta) = \frac{1}{m} \sum_{i=0}^m = [-y^{(i)} \log(h_{\theta}(x^{(i)})) - (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)}))]$$

Eq. 3.4

**Problem 3.a: Matrix Equivalences**

In Eq. 1,  $z$  can also be solved as  $X \cdot \theta$  which is the vectorized form of  $x^{(i)} \theta^{(i)}$ . However, it can also be expressed as  $\theta^T \cdot X$ . Prove the equality of  $X \cdot \theta$  with  $\theta^T \cdot X$  in this case.

**Problem 3.b: Matrix Shapes**

Determine the shape of  $h_{\theta}$  if  $X$  has a shape of  $(300, 5)$ .

**Problem 3.c: Vectorization**

Express  $J(\theta)$  into its vectorized form.

**Problem 4.c: Computational Programming (Also Laboratory 2)**

Encode Equations 3.1 to 3.4 as the class `LRegression` wherein:

- `LRegression` should be instantiated with a dataset  $X$ , a ground truth vector  $y$ , and a parameter vector  $\theta$ . Each parameter should have a data type of `numpy.array`.
- It should further have methods reflecting to at least the four (4) aforementioned equations. Each should have a return value.

▼ **Problem 3.a: Matrix Equivalences**

$$X = \begin{bmatrix} \text{---} (x^{(1)})^T \text{---} \\ \text{---} (x^{(2)})^T \text{---} \\ \vdots \\ \text{---} (x^{(m)})^T \text{---} \end{bmatrix}, \theta = \begin{bmatrix} \theta^{(1)} \\ \theta^{(2)} \\ \vdots \\ \theta^{(m)} \end{bmatrix}$$
$$X \cdot \theta = \begin{bmatrix} \text{---} (x^{(1)})^T \theta \text{---} \\ \text{---} (x^{(2)})^T \theta \text{---} \\ \vdots \\ \text{---} (x^{(m)})^T \theta \text{---} \end{bmatrix}$$
$$\theta^T \cdot X = \begin{bmatrix} \text{---} \theta^T (x^{(1)} \text{---}) \\ \text{---} \theta^T (x^{(2)} \text{---}) \\ \vdots \\ \text{---} \theta^T (x^{(m)} \text{---}) \end{bmatrix}$$

using element-wise multiplication:

$$X \cdot \theta = \theta^T \cdot X$$
$$[(x^{(1)})^T \theta^{(1)} + (x^{(2)})^T \theta^{(2)} \dots + (x^{(m)})^T \theta^{(m)}] = [(\theta^{(1)})^T x^{(1)} + (\theta^{(2)})^T x^{(2)} \dots + (\theta^{(m)})^T x^{(m)}]$$

▼ **Problem 3.b: Matrix Shapes**

$$X = \begin{bmatrix} a^1 & a^2 & a^3 & a^4 & a^5 \\ a^6 & a^7 & a^8 & a^9 & a^{10} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a^{296} & a^{297} & a^{298} & a^{299} & a^{300} \end{bmatrix}, \theta = \begin{bmatrix} (\theta^1) \\ (\theta^2) \\ \vdots \\ (\theta^k) \end{bmatrix}$$

The sigmoid function definition is:

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

The process can be written with matrix operation:

$$z = X \cdot \theta$$

$$h_{\theta}(x) = g(X \cdot \theta)$$

$$h_{\theta}(x) = g(z)$$

After applying multiplication between  $(k \cdot 1)$  and  $(m \cdot k)$  dimension we get a number  $z$ , then the  $z$  is passed to sigmoid function and we will have the final result of  $h_{\theta}(x)$ . The equation can be merged into one therefore the matrix shape of the  $h_{\theta}(x)$  would be the same as the matrix  $X$ .

$$h_{\theta}(x) = \begin{bmatrix} \theta^1(a) & \theta^2(a) & \theta^3(a) & \theta^4(a) & \theta^5(a) \\ \theta^6(a) & \theta^7(a) & \theta^8(a) & \theta^9(a) & \theta^{10}(a) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \theta^{296}(a) & \theta^{297}(a) & \theta^{298}(a) & \theta^{299}(a) & \theta^{300}(a) \end{bmatrix}$$

▼ Problem 3.c: Vectorization

The partial derivatives of  $(\theta_j)$

$$\begin{bmatrix} \frac{\partial J}{\partial \theta_0} \\ \frac{\partial J}{\partial \theta_1} \\ \frac{\partial J}{\partial \theta_2} \\ \vdots \\ \frac{\partial J}{\partial \theta_n} \end{bmatrix} = \frac{1}{m} \begin{bmatrix} \sum_{i=1}^m [-y^{(i)} \log(h_{\theta}(x^{(i)})) - (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)}))] \\ \sum_{i=1}^m [-y^{(i)} \log(h_{\theta}(x^{(i)})) - (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)}))] \\ \sum_{i=1}^m [-y^{(i)} \log(h_{\theta}(x^{(i)})) - (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)}))] \\ \vdots \\ \sum_{i=1}^m [-y^{(i)} \log(h_{\theta}(x^{(i)})) - (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)}))] \end{bmatrix}$$

Since the partial derivatives were expressed, the gradient vector is formed as:

$$\frac{\partial J(\theta)}{\partial \theta_0} = \frac{1}{m} \sum_{i=1}^m (h_0(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

Therefore the vectorized form of  $J(\theta)$  is:

$$= \frac{1}{m} X^T (h_{\theta}(x) - y)$$

▼ Problem 4.c: Computational Programming (Also Laboratory 2)

1

✓ 0s completed at 11:27 PM

● ✕