**Linear Regression with One Variable**

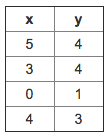
**TOTAL POINTS 5**

1.Question 1

Consider the problem of predicting how well a student does in her second year of college/university, given how well she did in her first year.

Specifically, let x be equal to the number of "A" grades (including A-. A and A+ grades) that a student receives in their first year of college (freshmen year). We would like to predict the value of y, which we define as the number of "A" grades they get in their second year (sophomore year).

Here each row is one training example. Recall that in linear regression, our hypothesis is h\_\theta(x) = \theta\_0 + \theta\_1x*hθ*​(*x*)=*θ*0​+*θ*1​*x*, and we use m*m* to denote the number of training examples.



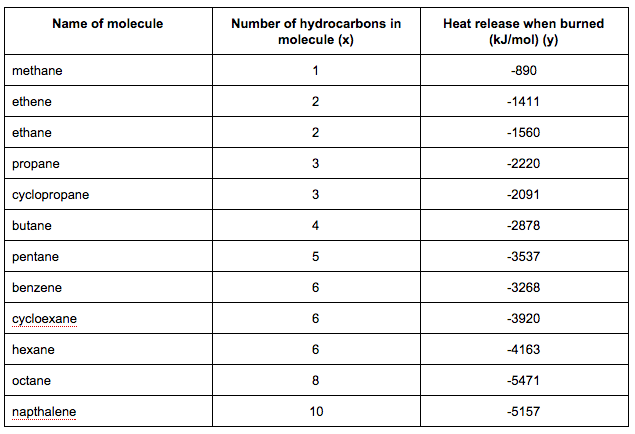
For the training set given above (note that this training set may also be referenced in other questions in this quiz), what is the value of m*m*? In the box below, please enter your answer (which should be a number between 0 and 10).

1 point



2.Question 2

Many substances that can burn (such as gasoline and alcohol) have a chemical structure based on carbon atoms; for this reason they are called hydrocarbons. A chemist wants to understand how the number of carbon atoms in a molecule affects how much energy is released when that molecule combusts (meaning that it is burned). The chemist obtains the dataset below. In the column on the right, “kJ/mol” is the unit measuring the amount of energy released.



You would like to use linear regression (h\_{\theta}(x) = \theta\_0 + \theta\_1 x*hθ*​(*x*)=*θ*0​+*θ*1​*x*) to estimate the amount of energy released (y) as a function of the number of carbon atoms (x). Which of the following do you think will be the values you obtain for \theta\_0*θ*0​and \theta\_1*θ*1​? You should be able to select the right answer without actually implementing linear regression.

1 point



\theta\_0 = -569.6, \theta\_1 = 530.9*θ*0​=−569.6,*θ*1​=530.9



\theta\_0 = -1780.0, \theta\_1 = -530.9*θ*0​=−1780.0,*θ*1​=−530.9



\theta\_0 = -569.6, \theta\_1 = -530.9*θ*0​=−569.6,*θ*1​=−530.9



\theta\_0 = -1780.0, \theta\_1 = 530.9*θ*0​=−1780.0,*θ*1​=530.9

Ans: \theta\_0 = -569.6, \theta\_1 = -530.9*θ*0​=−569.6,*θ*1​=−530.9

3.Question 3

Suppose we set \theta\_0 = -1, \theta\_1 = 0.5*θ*0​=−1,*θ*1​=0.5. What is h\_{\theta}(4)*hθ*​(4)?

1 point



4.Question 4

Let f*f* be some function so that

f(\theta\_0, \theta\_1)*f*(*θ*0​,*θ*1​) outputs a number. For this problem,

f*f* is some arbitrary/unknown smooth function (not necessarily the

cost function of linear regression, so f*f* may have local optima).

Suppose we use gradient descent to try to minimize f(\theta\_0, \theta\_1)*f*(*θ*0​,*θ*1​)

as a function of \theta\_0*θ*0​ and \theta\_1*θ*1​. Which of the

following statements are true? (Check all that apply.)

1 point



No matter how \theta\_0*θ*0​ and \theta\_1*θ*1​ are initialized, so long

as \alpha*α* is sufficiently small, we can safely expect gradient descent to converge

to the same solution.



If the first few iterations of gradient descent cause f(\theta\_0, \theta\_1)*f*(*θ*0​,*θ*1​) to

**increase** rather than decrease, then the most likely cause is that we have set the

learning rate \alpha*α* to too large a value.



If \theta\_0*θ*0​ and \theta\_1*θ*1​ are initialized at

the global minimum, then one iteration will not change their values.



Setting the learning rate \alpha*α* to be very small is not harmful, and can

only speed up the convergence of gradient descent.

5.Question 5

Suppose that for some linear regression problem (say, predicting housing prices as in the lecture), we have some training set, and for our training set we managed to find some \theta\_0*θ*0​, \theta\_1*θ*1​ such that J(\theta\_0, \theta\_1)=0*J*(*θ*0​,*θ*1​)=0.

Which of the statements below must then be true? (Check all that apply.)

1 point



For these values of \theta\_0*θ*0​ and \theta\_1*θ*1​ that satisfy J(\theta\_0, \theta\_1) = 0*J*(*θ*0​,*θ*1​)=0,

we have that h\_\theta(x^{(i)}) = y^{(i)}*hθ*​(*x*(*i*))=*y*(*i*) for every training example (x^{(i)}, y^{(i)})(*x*(*i*),*y*(*i*))



This is not possible: By the definition of J(\theta\_0, \theta\_1)*J*(*θ*0​,*θ*1​), it is not possible for there to exist

\theta\_0*θ*0​ and \theta\_1*θ*1​ so that J(\theta\_0, \theta\_1) = 0*J*(*θ*0​,*θ*1​)=0



For this to be true, we must have \theta\_0 = 0*θ*0​=0 and \theta\_1 = 0*θ*1​=0

so that h\_\theta(x) = 0*hθ*​(*x*)=0



We can perfectly predict the value of y*y* even for new examples that we have not yet seen.

(e.g., we can perfectly predict prices of even new houses that we have not yet seen.)

Week 2

## Linear Regression with Multiple Variables

**Linear Regression with Multiple Variables**

**TOTAL POINTS 5**

1.Question 1

Suppose *m*=4 students have taken some class, and the class had a midterm exam and a final exam. You have collected a dataset of their scores on the two exams, which is as follows:

|  |  |  |
| --- | --- | --- |
| midterm exam | (midterm exam)^2 | final exam |
| 89 | 7921 | 96 |
| 72 | 5184 | 74 |
| 94 | 8836 | 87 |
| 69 | 4761 | 78 |

You'd like to use polynomial regression to predict a student's final exam score from their midterm exam score. Concretely, suppose you want to fit a model of the form h\_\theta(x) = \theta\_0 + \theta\_1 x\_1 + \theta\_2 x\_2*hθ*​(*x*)=*θ*0​+*θ*1​*x*1​+*θ*2​*x*2​, where x\_1*x*1​ is the midterm score and x\_2 is (midterm score)^2. Further, you plan to use both feature scaling (dividing by the "max-min", or range, of a feature) and mean normalization.

What is the normalized feature x\_2^{(4)}*x*2(4)​? (Hint: midterm = 69, final = 78 is training example 4.) Please round off your answer to two decimal places and enter in the text box below.

1 point



2.Question 2

You run gradient descent for 15 iterations

with \alpha = 0.3*α*=0.3 and compute

J(\theta)*J*(*θ*) after each iteration. You find that the

value of J(\theta)*J*(*θ*) **decreases** quickly then levels

off. Based on this, which of the following conclusions seems

most plausible?

1 point



Rather than use the current value of \alpha*α*, it'd be more promising to try a smaller value of \alpha*α* (say \alpha = 0.1*α*=0.1).



\alpha = 0.3*α*=0.3 is an effective choice of learning rate.



Rather than use the current value of \alpha*α*, it'd be more promising to try a larger value of \alpha*α* (say \alpha = 1.0*α*=1.0).

Ans: \alpha = 0.3*α*=0.3 is an effective choice of learning rate.

3.Question 3

Suppose you have m = 28*m*=28 training examples with n = 4*n*=4 features (excluding the additional all-ones feature for the intercept term, which you should add). The normal equation is \theta = (X^TX)^{-1}X^Ty*θ*=(*XTX*)−1*XTy*. For the given values of m*m* and n*n*, what are the dimensions of \theta*θ*, X*X*, and y*y* in this equation?

1 point



X*X* is 28\times428×4, y*y* is 28\times128×1, \theta*θ* is 4\times44×4



X*X* is 28\times428×4, y*y* is 28\times128×1, \theta*θ* is4\times14×1



X*X* is 28\times528×5, y*y* is 28\times528×5, \theta*θ* is 5\times55×5



X*X* is 28\times528×5, y*y* is 28\times128×1, \theta*θ* is 5\times15×1

Ans: X*X* is 28\times528×5, y*y* is 28\times128×1, \theta*θ* is 5\times15×1

4.Question 4

Suppose you have a dataset with m = 1000000*m*=1000000 examples and n = 200000*n*=200000 features for each example. You want to use multivariate linear regression to fit the parameters \theta*θ* to our data. Should you prefer gradient descent or the normal equation?

1 point



Gradient descent, since (X^TX)^{-1}(*XTX*)−1 will be very slow to compute in the normal equation.



The normal equation, since gradient descent might be unable to find the optimal \theta*θ*.



The normal equation, since it provides an efficient way to directly find the solution.



Gradient descent, since it will always converge to the optimal \theta*θ*.

Ans: Gradient descent, since (X^TX)^{-1}(*XTX*)−1 will be very slow to compute in the normal equation.

5.Question 5

Which of the following are reasons for using feature scaling?

1 point



It prevents the matrix X^TX*XTX* (used in the normal equation) from being non-invertable (singular/degenerate).



It speeds up solving for \theta*θ* using the normal equation.



It speeds up gradient descent by making it require fewer iterations to get to a good solution.



It is necessary to prevent gradient descent from getting stuck in local optima.

## Octave/Matlab Tutorial

* 

Welcome Back,  
MD.!

* [Jahangirnagar University](https://www.coursera.org/programs/jahangirnagar-university-5wliw)
* [Jahangirnagar University](https://www.coursera.org/programs/jahangirnagar-university-na-qyc9d)
* [Explore](https://www.coursera.org/browse)
* [My Coursera](https://www.coursera.org/)

Account

* [Help Center](https://learner.coursera.help/)
* [For Enterprise](https://www.coursera.org/business)
* [Updates](https://www.coursera.org/updates)
* [Accomplishments](https://www.coursera.org/accomplishments)

[Machine Learning](https://www.coursera.org/learn/machine-learning/home/welcome)

[Week 2](https://www.coursera.org/learn/machine-learning/home/week/2)

Octave/Matlab Tutorial

[Prev](https://www.coursera.org/learn/machine-learning/supplement/AJ9VS/lecture-slides)

[Next](https://www.coursera.org/learn/machine-learning/programming/8f3qT/linear-regression)

**QUIZ • 10 MIN**

## Octave/Matlab Tutorial

**Submit your assignment**

**DUE**Jul 6, 12:59 PM +06

**ATTEMPTS**3 every 8 hours

Start

**Receive grade**

**TO PASS**80% or higher

**Grade**

—

#### **Octave/Matlab Tutorial**

Graded Quiz • 10 min

**Due** Jul 6, 12:59 PM +06

## Octave/Matlab Tutorial

**TOTAL POINTS 5**

1.Question 1

Suppose I first execute the following Octave/Matlab commands:



1

2

A = [1 2; 3 4; 5 6];

B = [1 2 3; 4 5 6];

Which of the following are then valid commands? Check all that apply. (Hint: A' denotes the transpose of A.)

1 point



C = A \* B;



C = B' + A;



C = A' \* B;



C = B + A;

2.Question 2

Let A =

⎡⎣⎢⎢16594211714310615138121⎤⎦⎥⎥

*A*=⎣⎢⎢⎢⎡​16594​211714​310615​138121​⎦⎥⎥⎥⎤​.

Which of the following indexing expressions gives B =

⎡⎣⎢⎢16594211714⎤⎦⎥⎥

*B*=⎣⎢⎢⎢⎡​16594​211714​⎦⎥⎥⎥⎤​? Check all that apply.

1 point



B = A(:, 1:2);



B = A(1:4, 1:2);



B = A(:, 0:2);



B = A(0:4, 0:2);

3.Question 3

Let A*A* be a 10x10 matrix and x*x* be a 10-element vector. Your friend wants to compute the product Ax*Ax* and writes the following code:



1

2

3

4

5

6

v = zeros(10, 1);

for i = 1:10

for j = 1:10

v(i) = v(i) + A(i, j) \* x(j);

end

end

How would you vectorize this code to run without any FOR loops? Check all that apply.

1 point



v = A \* x;



v = Ax;



v = A .\* x;



v = sum (A \* x);

4.Question 4

Say you have two column vectors v*v* and w*w*, each with 7 elements (i.e., they have dimensions 7x1). Consider the following code:



1

2

3

4

z = 0;

for i = 1:7

z = z + v(i) \* w(i)

end

Which of the following vectorizations correctly compute z? Check all that apply.

1 point



z = sum (v .\* w);



z = w' \* v;



z = v \* w;



z = w \* v;

5.Question 5

In Octave/Matlab, many functions work on single numbers, vectors, and matrices. For example, the sin function when applied to a matrix will return a new matrix with the sin of each element. But you have to be careful, as certain functions have different behavior. Suppose you have an 7x7 matrix X*X*. You want to compute the log of every element, the square of every element, add 1 to every element, and divide every element by 4. You will store the results in four matrices, A, B, C, D*A*,*B*,*C*,*D*. One way to do so is the following code:



1

2

3

4

5

6

7

8

for i = 1:7

for j = 1:7

A(i, j) = log(X(i, j));

B(i, j) = X(i, j) ^ 2;

C(i, j) = X(i, j) + 1;

D(i, j) = X(i, j) / 4;

end

end

Which of the following correctly compute A, B, C,*A*,*B*,*C*, or D*D*? Check all that apply.

1 point



C = X + 1;



D = X / 4;



A = log (X);



B = X ^ 2;



I, **MD. TAZEL HOSSAN**, understand that submitting work that isn’t my own may result in permanent failure of this course or deactivation of my Coursera account. Learn more about Coursera’s Honor Code