This exercise contains small data sets which enable making some calculation by hand.
It is recommended to perform some of these calculations by hand, when you are done coding the solution, or when you are debugging it, to make sure it works properly.

Linear regression:

1. A new research tries to find a relation between ages of two siblings and the the number of times they talk to each other in an average week. Here is the data collected so far:

Older sibling	Younger sibling	Times talked
31	22	2
22	21	3
40	37	8
26	25	12

   A. Assuming the research hypothesis is that there is a linear connection i.e. $h_\theta(x_1, x_2) = \theta_1 x_1 + \theta_2 x_2$, find the best θ_1, θ_2.
   B. One research assistant suggested to another feature: The age difference between the siblings i.e. $x_3 = x_1 - x_2$. Can this improve the fitting? If it can, calculate the new $\theta_1, \theta_2, \theta_3$, if it will not, explain why.
   C. Another research assistant suggested to add the square of the age difference as a feature i.e. $x_3 = (x_1 - x_2)^2$. Can this improve the fitting? If it will, calculate the new $\theta_1, \theta_2, \theta_3$, if it will not, explain why.
   D. A third research assistant suggested to add a strange feature: a vector of ones, i.e. $x_{3,i} = 1 \ \forall i$. Can this improve the fitting? if it will not, explain why. If it will, calculate the new If it will, calculate the new $\theta_1, \theta_2, \theta_3$, and also explain what is the meaning of this new feature. (hint: start by writing what is the new $h_\theta(x_1, x_2, x_3)$).
   E. Try adding or combining features to see if you can make a better prediction.

Gradient descent:

1. Consider a supervised learning model with only one input feature and a standard MSE loss function.
   Let the hypothesis class be $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$.
   And we have 3 data points in our training set:

X	Y
0	1
1	3
2	7

We want to optimize the weights using full batch gradient descent. Starting point is (2,2,0).

A. Find the loss at the starting point and after 100 iterations, using two different learning rates: 1, 0.1, 0.01
B. For each learning rate, explain why did the gradient descent succeed/fail?
C. Try finding the best learning rate for the model to converge, think what wrong with both bigger and smaller learning rates.

A good summary of gradient descent algorithms
http://ruder.io/optimizing-gradient-descent/

C**.Repeat the process using LR=0.01, but this time with momentum $\gamma = 0.9$.

D**.Repeat the process using LR=0.01, but this time with Nesterov accelerated gradient $\gamma = 0.9$