Machine Learning

Andrew Ng

Exercise 2: Linear Regression

This course consists of videos and programming exercises to teach you about machine learning. The ex designed to give you hands-on, practical experience for getting these algorithms to work. To get the mos course, you should watch the videos and complete the exercises in the order in which they are listed.

This first exercise will give you practice with linear regression. These exercises have been extensively te but they should also work in Octave, which has been called a "free version of Matlab." If you are using C install the **Image** package as well (available for Windows as an option in the installer, and available for L Octave-Forge).

Data

Download ex2Data.zip, and extract the files from the zip file.

The files contain some example measurements of heights for various boys between the ages of two anc y-values are the heights measured in meters, and the x-values are the ages of the boys corresponding t

Each height and age tuple constitutes one training example $(x^{(i)}, y^{(i)}$ in our dataset. There are $m =$

examples, and you will use them to develop a linear regression model.

Supervised learning problem

In this problem, you'll implement linear regression using gradient descent. In Matlab/Octave, you can loa using the commands

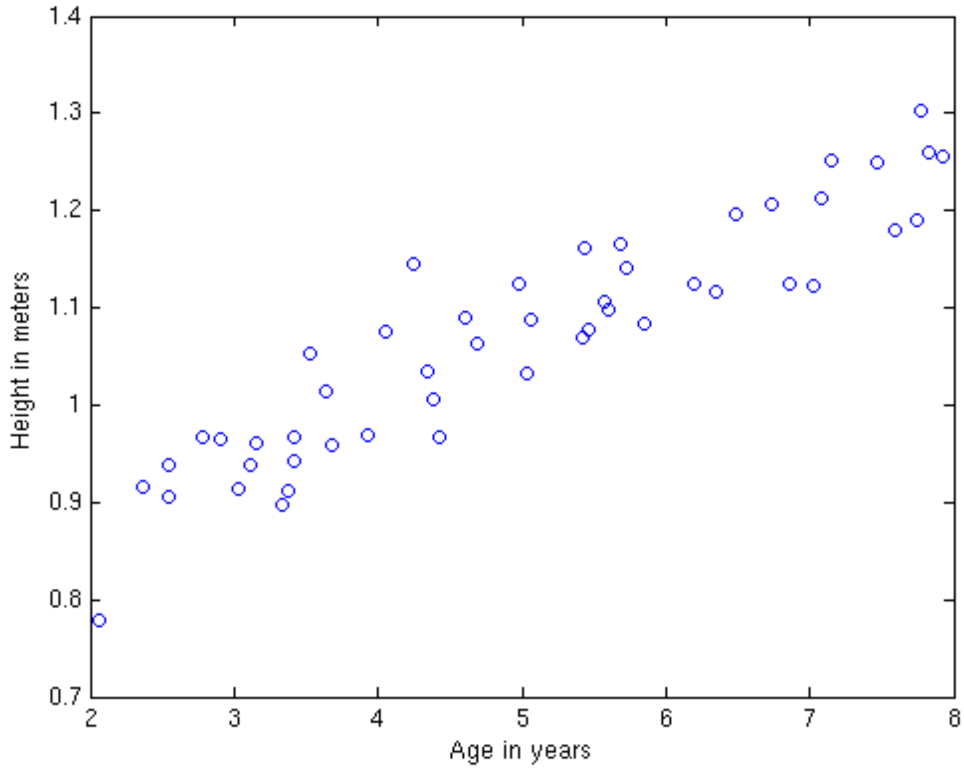
```
x = load('ex2x.dat');
y = load('ex2y.dat');
```

This will be our training set for a supervised learning problem with $n = 1$ features (in addition to the u

$x \in R^2$). If you're using Matlab/Octave, run the following commands to plot your training set (and lab

```
figure % open a new figure window
plot(x, y, 'o');
ylabel('Height in meters')
xlabel('Age in years')
```

You should see a series of data points similar to the figure below.



Before starting gradient descent, we need to add the $x_0 = 1$ intercept term to every example. To do th

Matlab/Octave, the command is

```
m = length(y); % store the number of training examples
x = [ones(m, 1), x]; % Add a column of ones to x
```

From this point on, you will need to remember that the age values from your training data are actually in column of x. This will be important when plotting your results later.

Linear regression

Now, we will implement linear regression for this problem. Recall that the linear regression model is

$$h_\theta(x) = \theta^T x = \sum_{i=0}^{n} \theta_i x_i,$$

and the batch gradient descent update rule is

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

1. Implement gradient descent using a learning rate of $\alpha = 0.07$. Since Matlab/Octave and Octave in starting from 1 rather than 0, you'll probably use `theta(1)` and `theta(2)` in Matlab/Octave to represe

Initialize the parameters to $\theta = \vec{0}$ (i.e., $\theta_0 = \theta_1 = 0$), and run one iteration of gradient descent fron

point. Record the value of of θ_0 and θ_1 that you get after this first iteration. (To verify that your impleme

later we'll ask you to check your values of θ_0 and θ_1 against ours.)

2. Continue running gradient descent for more iterations until θ converges. (this will take a total of abou After convergence, record the final values of θ_0 and θ_1 that you get.

When you have found θ, plot the straight line fit from your algorithm on the same graph as your training commands will look something like this:

```
hold on % Plot new data without clearing old plot
plot(x(:,2), x*theta, '-') % remember that x is now a matrix with 2 columns
                           % and the second column contains the time info
legend('Training data', 'Linear regression')
```

Note that for most machine learning problems, x is very high dimensional, so we don't be able to plot h

in this example we have only one feature, being able to plot this gives a nice sanity-check on our result.

3. Finally, we'd like to make some predictions using the learned hypothesis. Use your model to predict th boys of age 3.5 and age 7.

Debugging If you are using Matlab/Octave and seeing many errors at runtime, try inspecting your matri check that you are multiplying and adding matrices in ways that their dimensions would allow. Remembe Matlab/Octave by default interprets an operation as a matrix operation. In cases where you don't intend definition of an operator but your expression is ambiguous to Matlab/Octave, you will have to use the 'dc specify your command. Additionally, you can try printing x, y, and theta to make sure their dimensions ar

Understanding $J(\theta)$

We'd like to understand better what gradient descent has done, and visualize the relationship between tl $\theta \in R^2$ and $J(\theta)$. In this problem, we'll plot $J(\theta)$ as a 3D surface plot. (When applying learning alg

usually try to plot $J(\theta)$ since usually $\theta \in R^n$ is very high-dimensional so that we don't have any sim

visualize $J(\theta)$. But because the example here uses a very low dimensional $\theta \in R^2$, we'll plot $J(\theta)$

intuition about linear regression.) Recall that the formula for $J(\theta)$ is

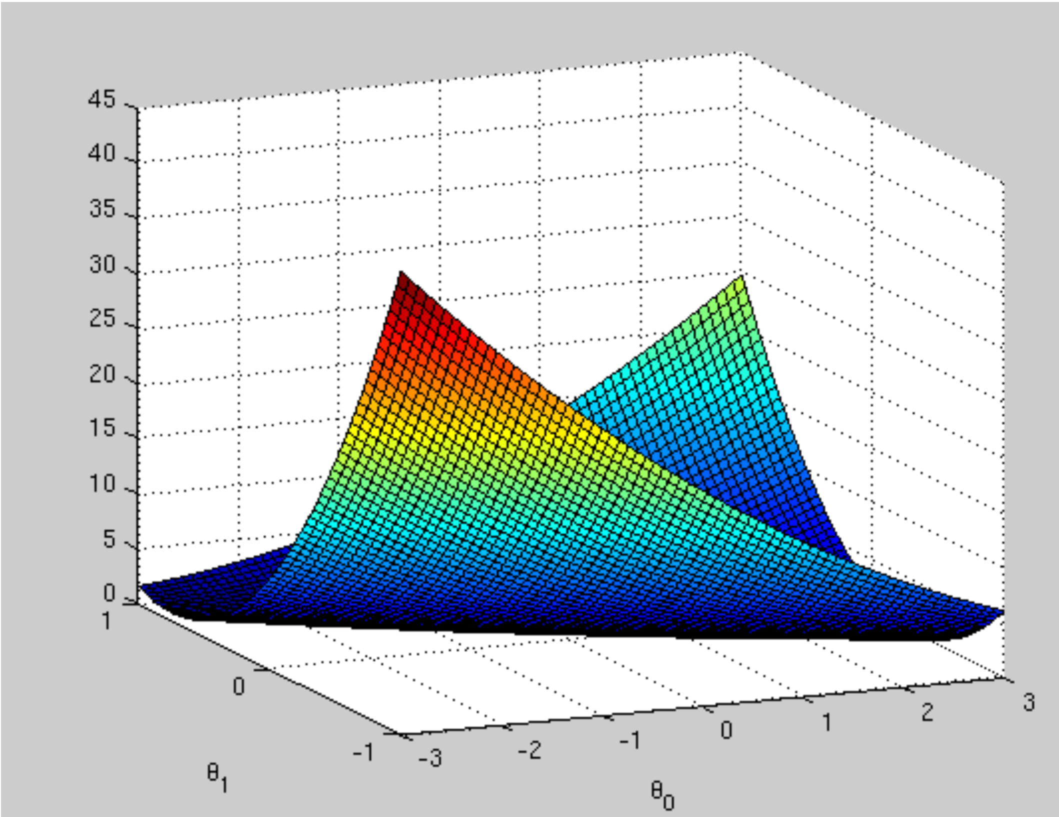
$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$$

To get the best viewing results on your surface plot, use the range of theta values that we suggest in the
below.

```
J_vals = zeros(100, 100);   % initialize Jvals to 100x100 matrix of 0's
theta0_vals = linspace(-3, 3, 100);
theta1_vals = linspace(-1, 1, 100);
for i = 1:length(theta0_vals)
          for j = 1:length(theta1_vals)
          t = [theta0_vals(i); theta1_vals(j)];
          J_vals(i,j) = %% YOUR CODE HERE %%
    end
end

% Plot the surface plot
% Because of the way meshgrids work in the surf command, we need to
% transpose J_vals before calling surf, or else the axes will be flipped
J_vals = J_vals'
figure;
surf(theta0_vals, theta1_vals, J_vals)
xlabel('\theta_0'); ylabel('\theta_1')
```

You should get a figure similar to the following. If you are using Matlab/Octave, you can use the orbit too
from different viewpoints.



What is the relationship between this 3D surface and the value of θ_0 and θ_1 that your implementation descent had found?

Show Solution
