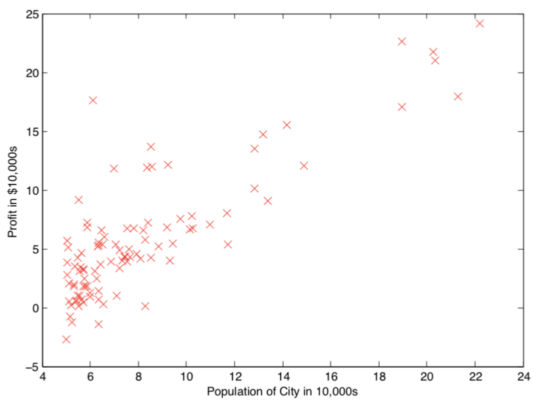
# Programming Exercise:
# Linear and Logistic Regressions

## 1 Exercise 1 : Linear Regression

### 1.1 Linear regression with one variable

Suppose you are the CEO of a restaurant franchise and are considering different cities for opening a new outlet. The chain already has trucks in various cities and you have data for profits and populations from the cities.

### Data




In the exercise, $x$ is 'Population', $y$ is 'Profit'

### Gradient Descent

You will fit the linear regression parameters $\mathbf{w}$ to your dataset using gradient descent.

The objective of linear regression is to minimize the cost function:

$$E(\mathbf{w}) = \frac{1}{2N} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2$$

where the hypothesis $y(x, \mathbf{w})$ is given by the linear model.
$y(x, \mathbf{w}) = w_0 + w_1 x$

Question (please answer in the notebook): in this dataset, $N =$?

The parameters of your model are the $w_j$ values which you will adjust to minimize cost $E(\mathbf{w})$. One way to do this is to use the batch gradient descent algorithm. In batch gradient descent, each iteration performs the update:

$$\mathbf{w} = \mathbf{w} - \eta \frac{1}{N} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\} x_n$$

With each step of gradient descent, your parameters $w_j$ come closer to the optimal values that will achieve the lowest cost $E(\mathbf{w})$.

## 1.2 Linear regression with multiple variables

You will implement linear regression with multiple variables to predict the prices of houses. Suppose you are selling your house and you want to know what a good market price would be. One way to do this is to first collect information on recent houses sold and make a model of housing prices.

### Feature Normalization

The notebook will start by loading and displaying some values from this dataset.

You then complete the code to:

- Subtract the mean value of each feature from the dataset.

- After subtracting the mean, additionally scale (divide) the feature values by their respective "standard deviations."

### Gradient Descent

Previously, you implemented gradient descent on a univariate regression problem. The only difference now is that there is one more feature in the matrix X. The hypothesis function and the batch gradient descent update rule remain unchanged. Repeat your pre-processing steps from last part and run the linear regression procedure on the new data set.

### Selecting learning rates

Try out different learning rates for the dataset and find a learning rate that converges quickly.

### Normal Equations

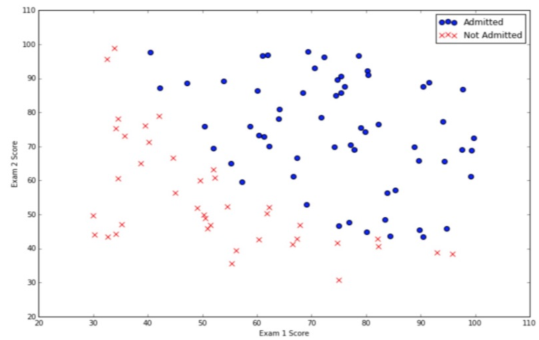
In the lecture, you learned the closed-form solution to linear regression.

You can implement this solution (bonus!!!).

# 2 Exercise 2 : Logistic Regression

## 2.1 Logistic Regression

### Data



### Feature mapping

### Sigmoid function

Recall the that the logistic regression hypothesis is defined as:

$y(x, \mathbf{w}) = \sigma(\mathbf{w}^T x)$

where $\sigma$ is a sigmoid function:

$\sigma(z) = \dfrac{1}{1 + e^{-z}}$

### Cost function and gradient

The cost function in logistic regression is:

$$E(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} \{-t_n ln(y(x_n, \mathbf{w})) - (1 - t_n) ln(1 - y(x_n, \mathbf{w}))\}$$

and the gradient of the cost is a vector of the same length as $\mathbf{w}$ where the $j$th element is defined as follows:

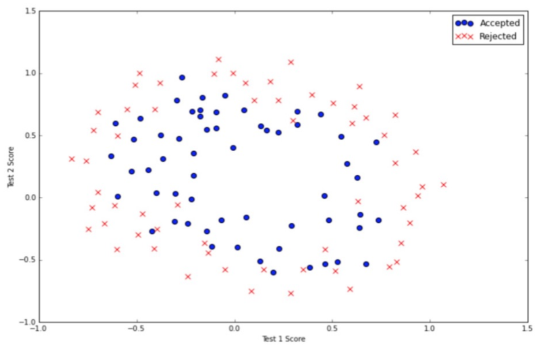
$$\frac{\Delta E(w_j)}{\Delta w_j} = \frac{1}{N} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\} x_n^{(j)}$$

($x_n^{(j)}$: the $j$th element of the input $x_n$)

While this gradient looks identical to the linear regression gradient, the formula is actually different because linear and logistic regression have different definitions of $y(x, \mathbf{w})$.

## 2.2 Regularized Logistic Regression

## Data




## Feature mapping

This data looks a bit more complicated than the previous example. There is no linear decision boundary that will perform well on this data. One way to deal with this is to construct features that are derived from polynomials of the original features.

- Create a bunch of polynomial features.
- Modify the cost and gradient functions to include the regularization term.