题目： Linear Regression

姓名:张胤民 学号:201694069

一、 实现功能简介

主要功能:

1. 实现单变量和多变量的线性回归的**误差计算(quadratic loss function)**

2. 实现单参数和多参数的**梯度更新算法**

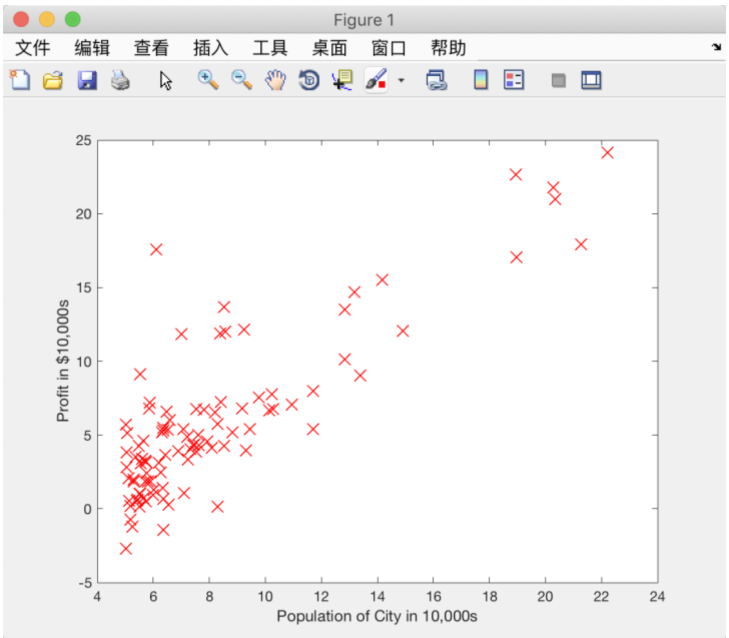
3. 实现特征的**标准化(normalization)**

4. 实现线性回归的**闭式解**(最小二乘法)

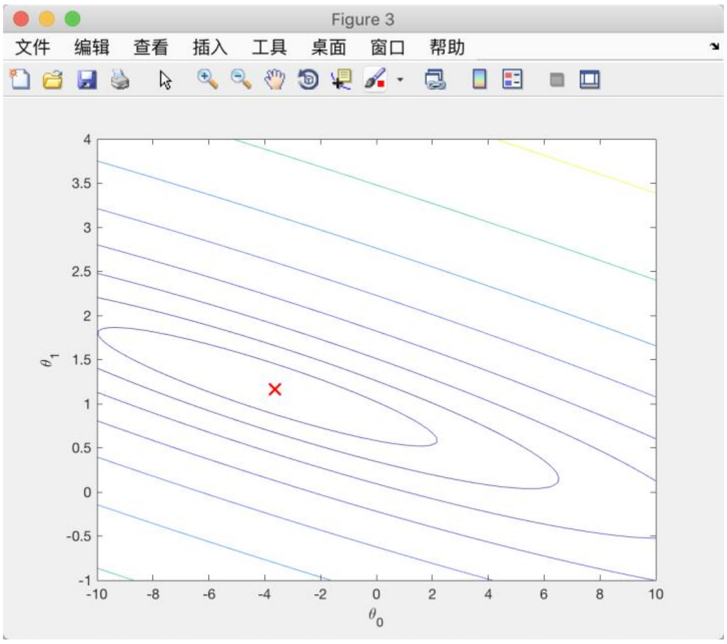
5. 实现数据显示

二、 具体编写代码及结果展示以及代码功能描述

1. 结果展示:




```
Theta found by gradient descent: -3.630291 1.166362
For population = 35,000, we predict a profit of 4519.767868
For population = 70,000, we predict a profit of 45342.450129
```



2. 代码编写以及代码功能描述

a) computeCost.m

代码：`J=sum((X*theta-y).^2)/(2*m);`

实现功能:单变量误差计算

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

b) computeCostMulti.m

代码：`J = ((X * theta - y)' * (X * theta - y))/(2*m);`

实现功能:多变量误差计算

$$J(\theta_0, \theta_1, \ldots, \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

c) featureNormalize.m

代码：

```
mu = mean(X);
sigma = std(X);
for i = 1:size(X,2)
X_norm(:,i) = (X(:,i) - mu(i)) / sigma(i);
end
```

实现功能: Z-score normalization

$$x_{normalization} = \frac{x - \mu}{\sigma}$$

d) gradientDescent.m

代码：

```
x=X(:,2);
theta0=theta(1);
theta1=theta(2);
theta0=theta0-alpha/m*sum(X*theta-y);
theta1=theta1-alpha/m*sum((X*theta-y).*x);
theta=[theta0;theta1];
```

实现功能:单变量的梯度下降

$$j=0:\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1)=\frac{1}{m}\sum_{i=1}^{m}(\theta_0+\theta_1x^{(i)}-y^{(i)})$$

$$j=1:\frac{\partial}{\partial\theta_1}J(\theta_0,\theta_1)=\frac{1}{m}\sum_{i=1}^{m}(\theta_0+\theta_1x^{(i)}-y^{(i)})\cdot x^{(i)}$$

e) gradientDescentMulti.m

代码：

```
% Perform a single gradient step on the parameter vector theta.
theta = theta - alpha / m * X' * (X * theta - y);
% Save the cost J in every iteration
J_history(iter) = computeCostMulti(X, y, theta);
```

实现功能:多变量的梯度下降(包含学习率以及迭代次数)

Gradient descent:

Repeat $\{$

$$\theta_j := \theta_j - \alpha\frac{\partial}{\partial\theta_j}J(\theta_0,\ldots,\theta_n)$$

$\}$ (simultaneously update for every $j=0,\ldots,n$)

f) normalEqn.m

代码：`theta = pinv(X' * X) * X' * y;`

实现功能:求解最小二乘法线性假设下的闭式解

$$\theta=(X^TX)^{-1}X^Ty$$

g) plotData.m

代码

```
figure; % open a new figure window
plot(x, y, 'rx', 'MarkerSize', 10); % Plot the data
ylabel('Profit in $10,000s'); % Set the y-axis label
xlabel('Population of City in 10,000s'); % Set the x-axis label
```

三、　小结（包括通过本内容的认识以及其他）

1. 认识:

线性回归: 线性回归是利用数理统计中回归分析，来确定两种或两种以上变量间相互依赖的定量关系的一种统计分析方法，运用十分广泛。其表达形式为 y = w'x+e，e 为误差服从均值为 0 的正态分布。

只包括一个自变量和一个因变量，且二者的关系可用一条直线近似表示，这种回归分析称为一元线性回归分析。如果回归分析中包括两个或两个以上的自变量，且因变量和自变量之间是线性关系，则称为多元线性回归分析。

最小二乘法: 在统计学中，线性回归（Linear regression）是利用称为线性回归方程的最小二乘函数对一个或多个自变量和因变量之间关系进行建模的一种回归分析。

最小二乘法的目标：求误差的最小平方和，对应有两种：线性和非线性。线性最小二乘的解是 closed-form 即

$$\theta = (X^T X)^{-1} X^T y$$

而非线性最小二乘没有 closed-form，通常用迭代法求解。

2. 心得:

1) 适用场景:数均分布符合线性性,误差项的独立性,同方差性,正态性;
2) 损失函数及其梯度下降法:影响梯度下降法优化误差的主要因素有学习率(learning rate), 迭代次数(iterations), 参数初始化(Initialization parameter)以及数据标准化(normalization).

   a) 学习率

   过小会导致梯度更新慢,长时间算法不能收敛,训练时间过长.

过大会导致梯度可能在最小值附近来回震荡,甚至可能无法收敛.

b) 迭代次数

过小,算法无法收敛; 过大在线性回归中不会出现过拟合现象.

c) 参数初始化

在学习率合理的情况下,采用随机初始化的方式时准确率最高.

d) 数据标准化

多变量的情况下, 如果各变量间尺度相似,那么标准化与否对收敛影响不大

如果给变量间尺度相差较大,不进行标准化,梯度下降过程中会出现震荡现象,进行标准化能加快收敛速度.

3) 最小二乘法的原则是以“残差平方和最小”确定直线位置。用最小二乘法除了计算比较方便外，得到的估计量还具有优良特性。这种方法对异常值非常敏感。