Programming Exercise 8: Anomaly Detection and Recommender Systems

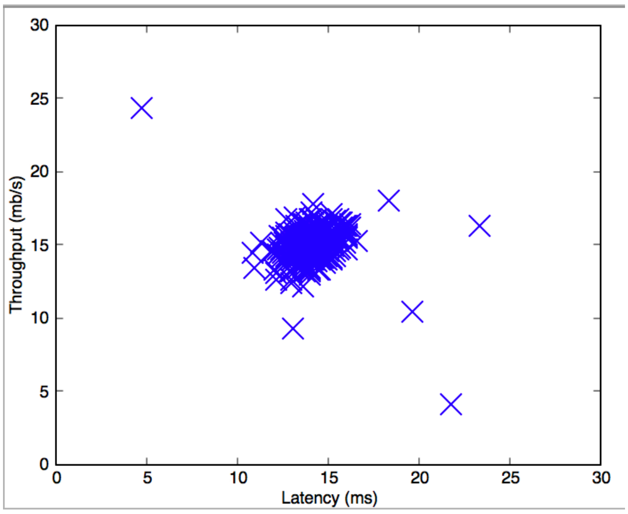
```
>> submit()
== Submitting solutions | Anomaly Detection and Recommender Systems...
Use token from last successful submission (WxxShirley@outlook.com)? (Y/n): Y
warning: division by zero

==
==                                   Part Name |     Score | Feedback
==                                   --------- |     ----- | --------
==              Estimate Gaussian Parameters |  15 /  15 | Nice work!
==                          Select Threshold |  15 /  15 | Nice work!
==              Collaborative Filtering Cost |  20 /  20 | Nice work!
==          Collaborative Filtering Gradient |  30 /  30 | Nice work!
==                          Regularized Cost |  10 /  10 | Nice work!
==                      Regularized Gradient |  10 /  10 | Nice work!
==                                   --------------------------------
==                                             | 100 / 100 |
==
>>
```

1 Anomaly detection

Task: Implement an anomaly detection algorithm to detect anomalous behavior in server computers.The features measure the throughput and latency of response of each server.

First,visualize the dataset.



1.1 Gaussian distribution

For each feature,we need to find parameters mu(i) and sigma2(i) that fit the data in the i-th dimension.

The Gaussian distribution is given by

$$p(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}},$$

where μ is the mean and σ^2 controls the variance.

1.2 Estimating parameters for a Gaussian

$$\mu_i = \frac{1}{m}\sum_{j=1}^{m} x_i^{(j)},$$

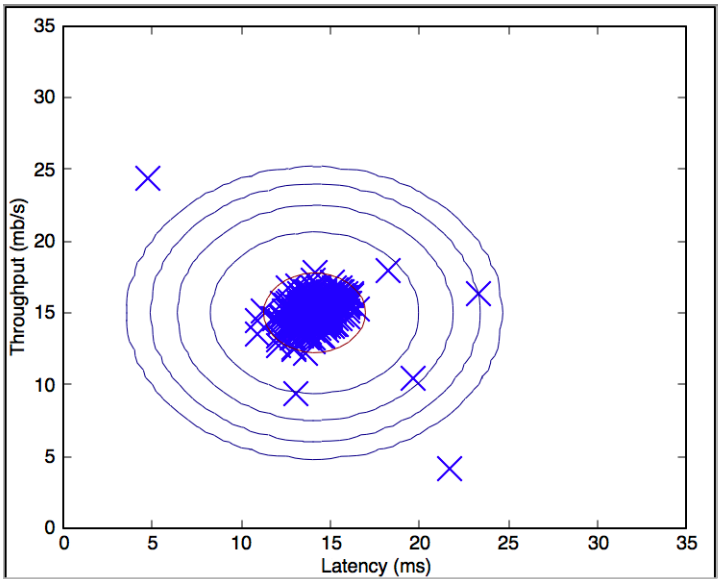
$$\sigma_i^2 = \frac{1}{m}\sum_{j=1}^{m}(x_i^{(j)} - \mu_i)^2.$$

So, the code to calculate mu and sigma(use a for loop):

```
for j=1:m,
sum=0;
for i=1:m,
sum+=X'(j,i);
end
mu(j)=sum/m;

X_ = X';
for j=1:n,
 sum=0;
 for i=1:m,
 X_(j,i)-=mu(j);
 sum+=X_(j,i)^2;
 end
 sigma2(j)=sum/m;
 end
```

And the Gaussian distribution contours of the distribution fit to the dataset.



1.3 Selecting the threshold,epsilon

Choosing the best threshold based on the F1-Score.

Definition:

The F_1 score is computed using precision ($prec$) and recall (rec):

$$F_1 = \frac{2 \cdot prec \cdot rec}{prec + rec}, \tag{3}$$

You compute precision and recall by:

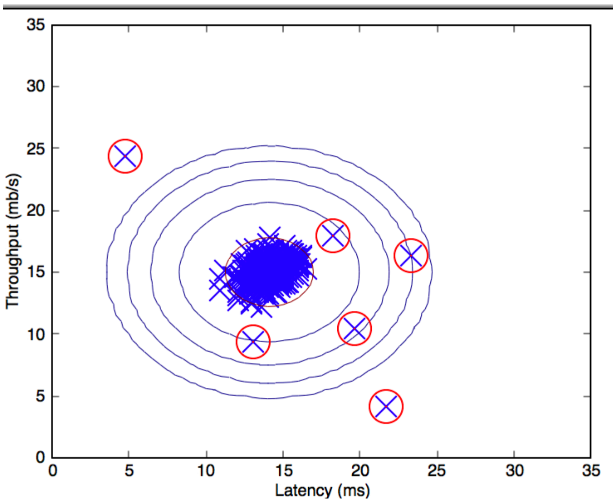
$$prec = \frac{tp}{tp + fp} \tag{4}$$

$$rec = \frac{tp}{tp + fn}, \tag{5}$$

```
function [bestEpsilon bestF1] = selectThreshold(yval,pval)
bestEpsilon=0;
bestF1=0;
F1=0;
stepsize = (max(pval)-min(pval))/1000;
for epsilon=min(pval):stepsize:max(pval)
predictions = (pval<epsilon);
fp = sum((predictions==1)&(yval==0));
tp = sum((predictions==1)&(yval==1));
fn = sum((predictions==0)&(yval==1));
prec = tp/(tp+fp);
rec = tp/(tp+fn);
F1 = 2*prec*rec/(prec+rec);
```

```
if F1>bestF1
bestF1 = F1;
bestEpsilon=epsilon;
end
end
end
```

The classified anomalies:



2 Recommender Systems

Task:Implement the collaborative filtering learning algorithm and apply it to a dataset of movie ratings.

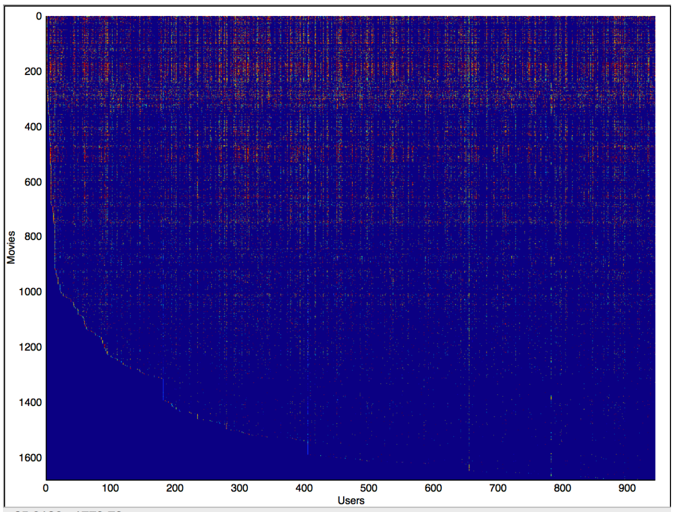
2.1 Movie ratings dataset

```
X - num_movies  x num_features matrix of movie features
Theta - num_users  x num_features matrix of user features
Y - num_movies x num_users matrix of user ratings of movies
R - num_movies x num_users matrix, where R(i, j) = 1 if the
    i-th movie was rated by the j-th user
```

$$\mathtt{X} = \begin{bmatrix} — (x^{(1)})^T — \\ — (x^{(2)})^T — \\ \vdots \\ — (x^{(n_m)})^T — \end{bmatrix}, \quad \mathtt{Theta} = \begin{bmatrix} — (\theta^{(1)})^T — \\ — (\theta^{(2)})^T — \\ \vdots \\ — (\theta^{(n_u)})^T — \end{bmatrix}.$$

2.2 Collaborative filtering learning algorithm

Load the dataset:



2.2.1 Collaborative filtering cost function

$$J(x^{(1)}, ..., x^{(n_m)}, \theta^{(1)}, ..., \theta^{(n_u)}) = \frac{1}{2} \sum_{(i,j):r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2.$$

```
%Two ways
% -1- Use 'for' loop
temp_matrix = X * Theta';
for i=1:size(R,1),
for j=1:size(R,2),
if R(i,j)==1
J+=(temp_matrix(i,j)-Y(i,j))^2;
end
end
end
J/=2;

% -2-
J = sum(sum(((X*Theta'-Y).*R).^2))/2;
% Pay Atthention to this Trick:
    %Instead of checking whether R(i,j)==1,
    %Use the '.*'
```

2.2.2 Collaborative filtering gradient

The gradients of the cost function is given by:

$$\frac{\partial J}{\partial x_k^{(i)}} = \sum_{j:r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})\theta_k^{(j)}$$

$$\frac{\partial J}{\partial \theta_k^{(j)}} = \sum_{i:r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})x_k^{(i)}.$$

```
temp_matrix = (X*Theta'-Y).*R;
X_grad = temp_matrix*Theta;
Theta_grad = temp_matrix'*X;
```

2.2.3 Regularized cost function

$$J(x^{(1)}, ..., x^{(n_m)}, \theta^{(1)}, ..., \theta^{(n_u)}) = \frac{1}{2} \sum_{(i,j):r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \left(\frac{\lambda}{2} \sum_{j=1}^{n_u} \sum_{k=1}^{n} (\theta_k^{(j)})^2\right) + \left(\frac{\lambda}{2} \sum_{i=1}^{n_m} \sum_{k=1}^{n} (x_k^{(i)})^2\right).$$

```
J += (sum(sum(Theta.^2)))+sum(sum(X.^2)))*lambda/2;
```

2.2.4 Regularized gradient

$$\frac{\partial J}{\partial x_k^{(i)}} = \sum_{j:r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})\theta_k^{(j)} + \lambda x_k^{(i)}$$

$$\frac{\partial J}{\partial \theta_k^{(j)}} = \sum_{i:r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})x_k^{(i)} + \lambda \theta_k^{(j)}.$$

```
X_grad += lambda*X;
Theta_grad += lambda*Theta;
```

Output:

```
Top recommendations for you:
Predicting rating 5.0 for movie Someone Else's America (1995)
Predicting rating 5.0 for movie Star Kid (1997)
Predicting rating 5.0 for movie Entertaining Angels: The Dorothy Day Story (1996)
Predicting rating 5.0 for movie Aiqing wansui (1994)
Predicting rating 5.0 for movie They Made Me a Criminal (1939)
Predicting rating 5.0 for movie Great Day in Harlem, A (1994)
Predicting rating 5.0 for movie Marlene Dietrich: Shadow and Light (1996)
Predicting rating 5.0 for movie Saint of Fort Washington, The (1993)
Predicting rating 5.0 for movie Prefontaine (1997)
Predicting rating 5.0 for movie Santa with Muscles (1996)


Original ratings provided:
Rated 4 for Toy Story (1995)
Rated 3 for Twelve Monkeys (1995)
Rated 5 for Usual Suspects, The (1995)
Rated 4 for Outbreak (1995)
Rated 5 for Shawshank Redemption, The (1994)
Rated 3 for While You Were Sleeping (1995)
Rated 5 for Forrest Gump (1994)
Rated 2 for Silence of the Lambs, The (1991)
Rated 4 for Alien (1979)
Rated 5 for Die Hard 2 (1990)
Rated 5 for Sphere (1998)
>>
```