week8_code

Created: 18/11/28 10:25 **Updated:** 18/11/28 11:16

partial codes in week_8 code assignment.

> written by *VincentX3*, Nov.28.18
> In this assignment we try to compress image by using K-Means Algorithm. What's more we also done dimensional reduction using PCA.
>
> *the example images are quite, emm, don't know how to describe.*

K-Means

kMeansInitCentroids.m

random initialize

```
% Initialize the centroids to be random examples
% Randomly reorder the indices of examples
randidx = randperm(size(X, 1));
% Take the first K examples as centroids
centroids = X(randidx(1:K), :);
```

findClosestCentroids.m

__ cluster assignment__

```
m=size(X,1);
dist=zeros(K,1);
for i=1:m
    for k=1:K
        dist(k)=sum((X(i,:)-centroids(k,:)).^2);
    end
     [value, idx(i)] = min(dist);
```

```
    %idx(i)=find(dist==min(dist));
end
```

`Catuion` here:
because in some cases may have the same minimum distance to **more than one centroid.** This situation led `find()` failed to implement. So be careful, use `[value, idx(i)] = min(dist);` is a smart choice.

computeCentroids.m

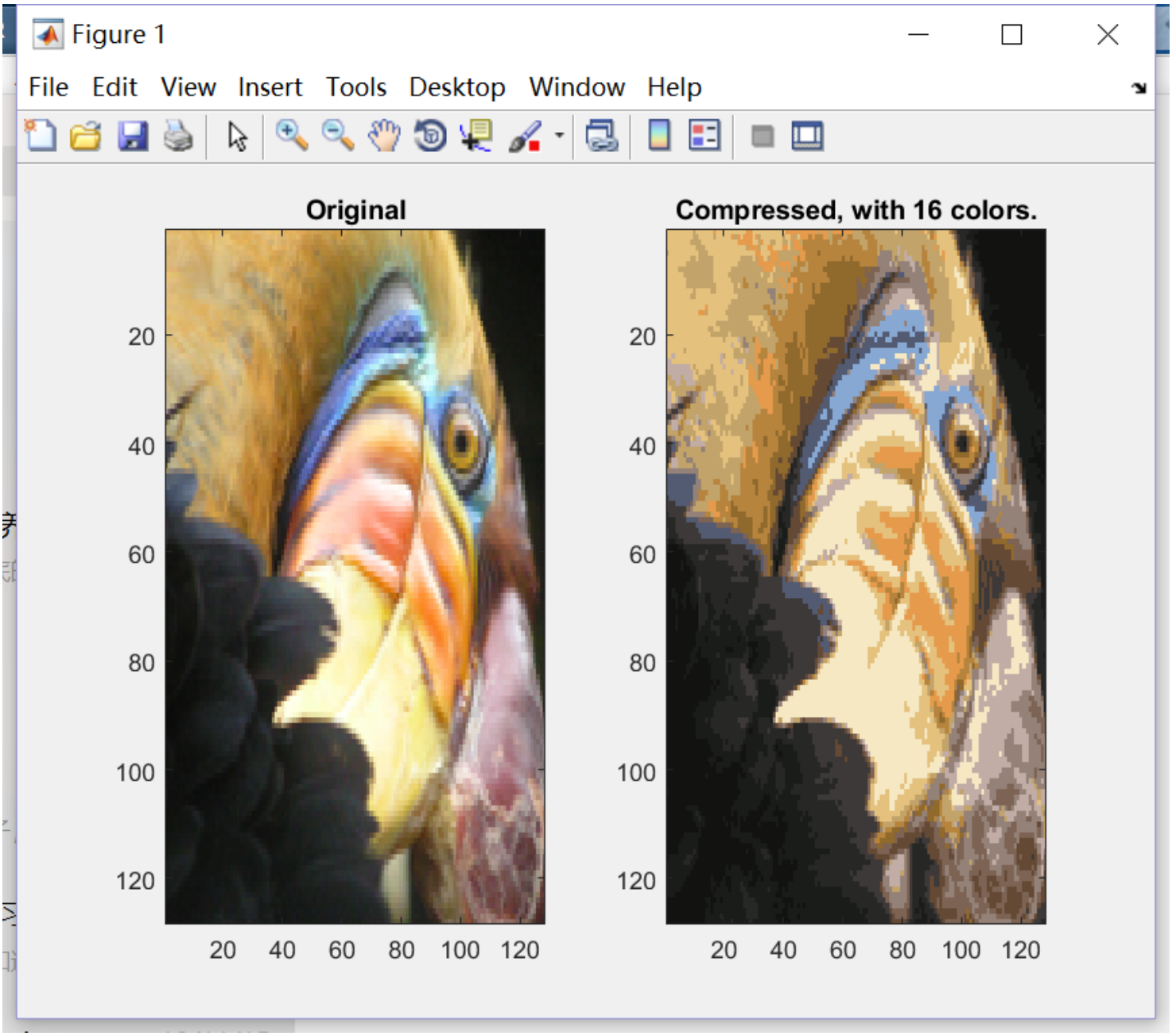
move centroids

```
for k=1:K
    %sum up the number
    num=sum(idx==k);
    %move centroids
    centroids(k,:)=sum(X(idx==k,:))/num;
end
```

---

when implementing K-Means, remember to iterate in order to get better result, or we may stuck in bad local optima.

let's enjoy our works.



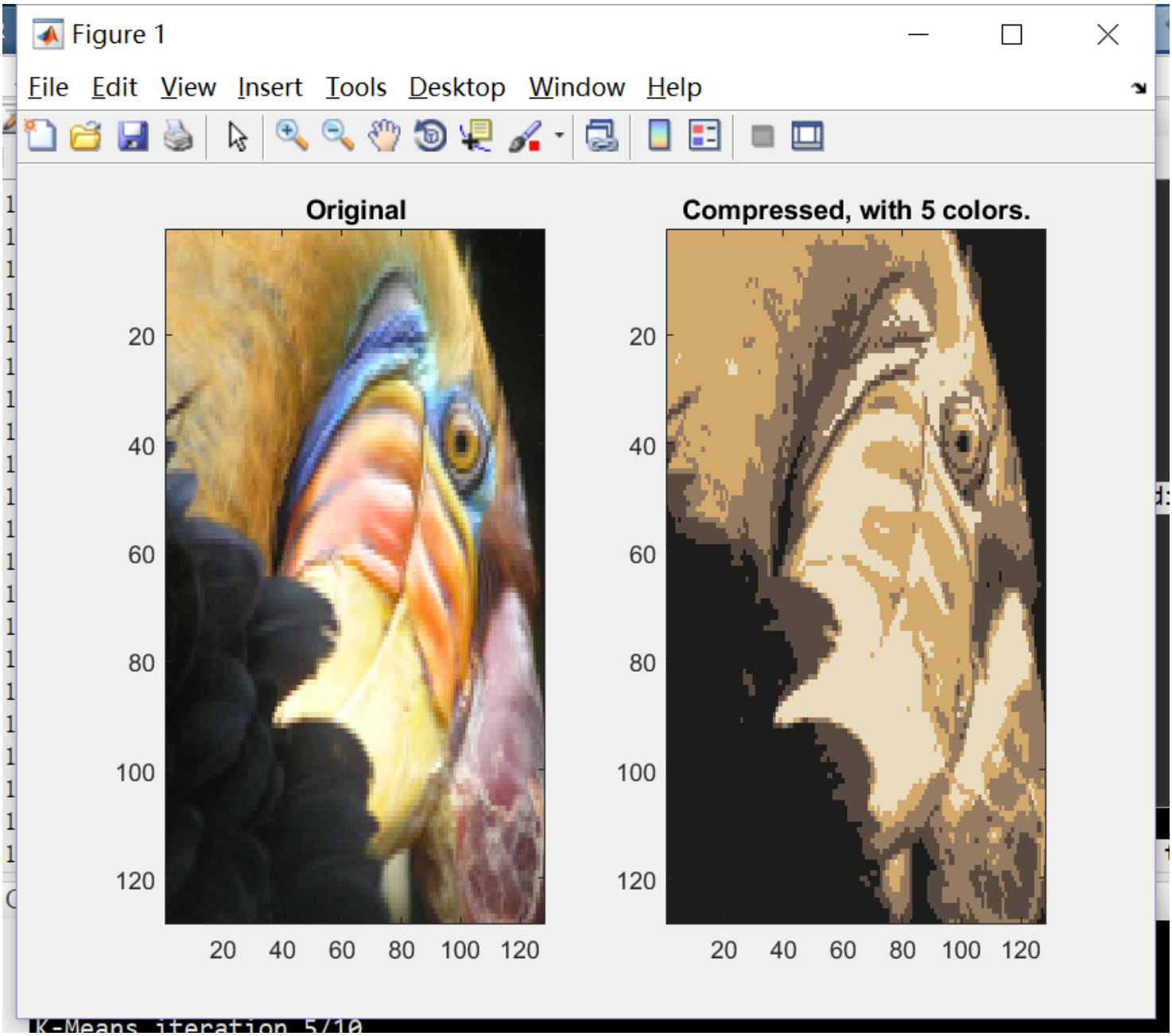
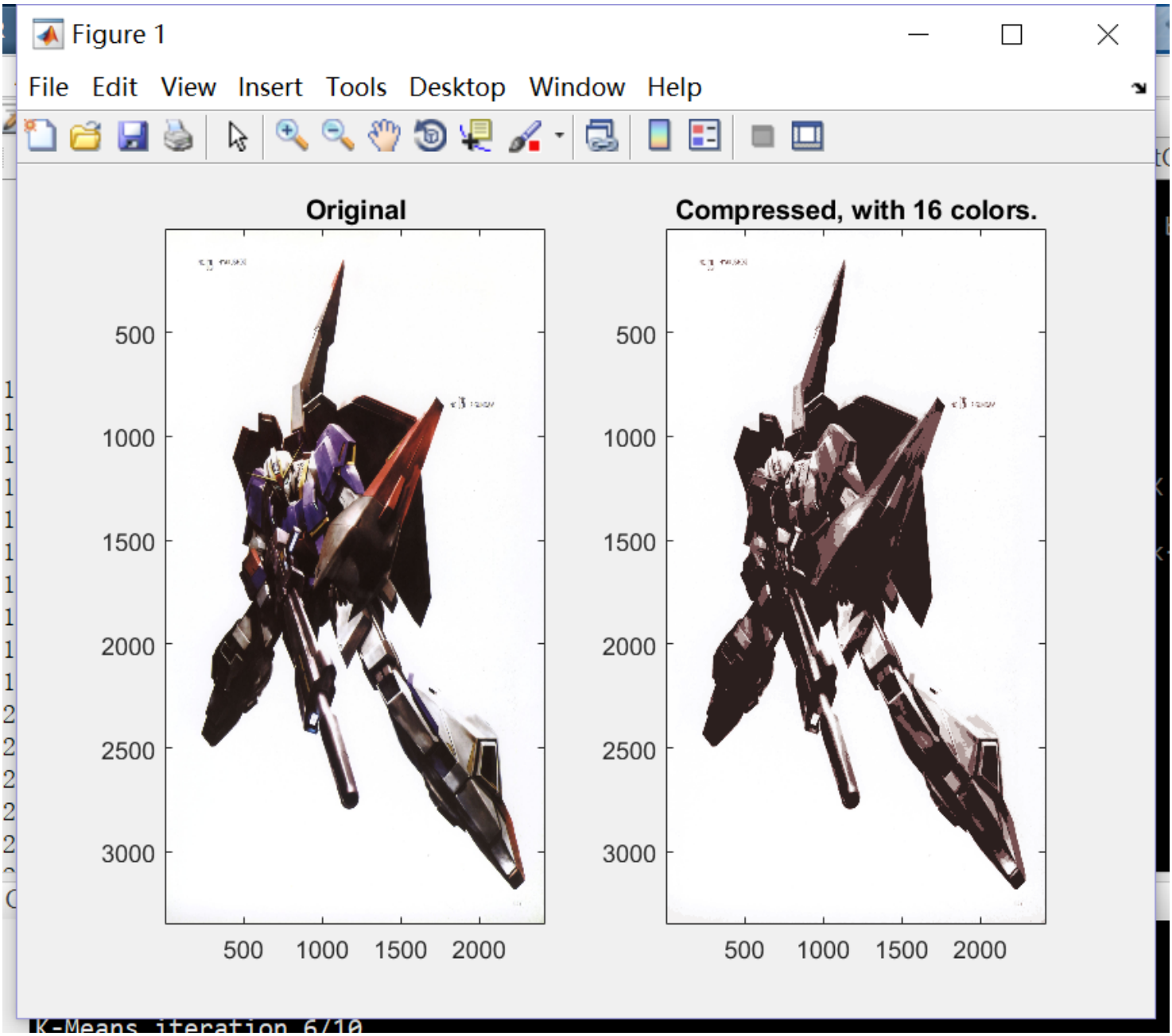


Figure 1
File Edit View Insert Tools Desktop Window Help
Original
Compressed, with 5 colors.
20
40
60
80
100
120
20 40 60 80 100 120
K-Means iteration 5/10



PCA

featureNormalize.m

computing co-variance matrix need normalizing first.

```
mu = mean(X);
X_norm = bsxfun(@minus, X, mu);
sigma = std(X_norm);
X_norm = bsxfun(@rdivide, X_norm, sigma);
```

pca.m

```
% X had already normalized
% Σ is a nXn matrix
Sigma=X'*X/m;
[U, S, V] = svd(Sigma);
```

projectData.m

```
Ureduce = U(:,1:K);     % take the first k directions
Z = X * Ureduce;        % compute the projected data points
```

recoverData.m

```
U_reduce = U(:, 1:K);
X_rec = Z * U_reduce';
```

Eigenfaces

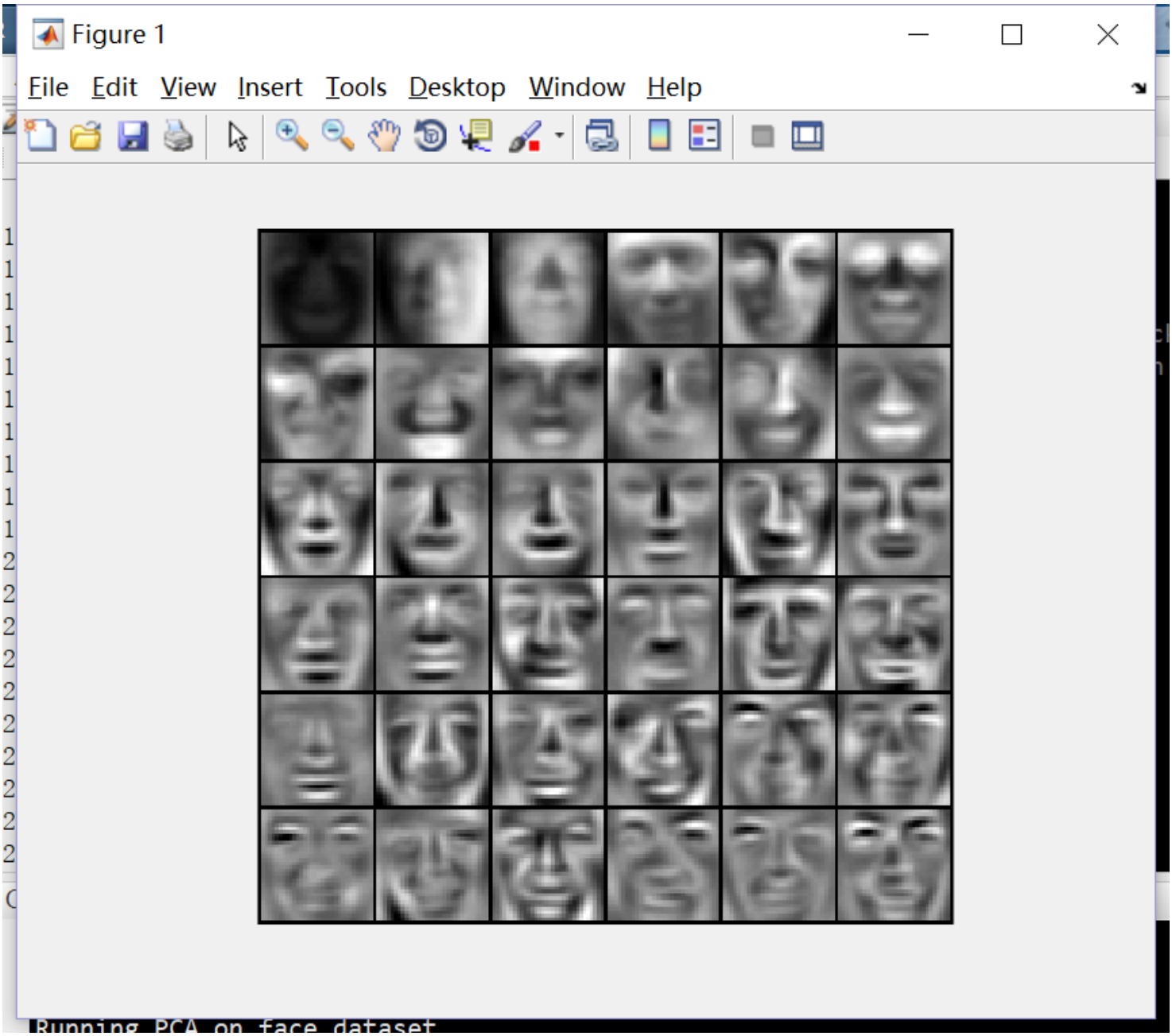


Figure 1
File Edit View Insert Tools Desktop Window Help
Running PCA on face dataset