Matlab Homework Assignment #1

- Using MATLAB, do the following procedure:
  1. Download the dataf file to your directory, name it as hw8.mat, and load it into your MATLAB session by: `>> load hw8;`
  2. Check what variables (i.e., arrays) are defined in this data file by running: `>> whos`
  3. Plot the data by: `>> plot(x,y); grid;`
  4. Create the Vandermonde matrix by: `>> A=[x.^0 x.^1];`
  5. Compute the least squares line over the given data by: `>> sol = inv(A'*A)*A'*y;`
  6. Overlay the least squares line over the current plot by: `>> hold on; plot(x, sol(1)+sol(2)*x, '--');`
  7. Put title, axis labels by: `>> title('Least squares line fit'); xlabel('x'); ylabel('y');`
  8. Print out this plot and submit the hardcopy of the plot.

Matlab Homework Assignment #2

- Using MATLAB, do the following procedure:
  1. Download the data file to your directory, name it as hw10.mat, and load it into your MATLAB session as in HW8 (except the filename change). This is exactly the same dataset as before but now I used older MATLAB data format.
  2. Repeat the same procedure as in HW8, i.e., draw the original data and the least squares line.
  3. Now, compute the least squares parabola (i.e., the second order polynomial), and overlay it on the plot so that you can compare it with the least squares line. Use '.' option in the plot function for this parabola to see the difference instead of '--' option used for the least

squares line.

matlab commands

```
load hw10;
who;
figure(1);plot(x,y);grid
A=[x.^0 x.^1 x.^2];
sol = inv(A'*A)*A'*y;
hold on; plot(x, sol(1)+sol(2)*x+sol(3)*x.^2, '--');
hold on; plot(x, sol(1)+sol(2)*x+sol(3)*x.^2, 'r');
```

4. Submit the hardcopy of the plot.

Matlab Homework Assignment #3

- Using MATLAB, do the following exercises:
  1. Load the image called mandrill.mat, via:
     `>> load mandrill;`
     This loads a matrix X containing a face of mandrill, and a map containing the colormap of the image. If you cannot load this data in your MATLAB, then download this data from this link. Then, do the `load` command again. Display this matrix on your screen by:
     `>> image(X); colormap(map)`
  2. Compute the SVD of this mandrill image and plot the distribution of its singular values on your screen (Note that the MATLAB `svd` function returns three matrices *U, S, V* for a given input matrix. So, the singular values are plotted by:
     `>> plot(diag(S));`
     Then print this figure.

3. Let σ_j, $\boldsymbol{u}_j$, $\boldsymbol{v}_j$ be a singular value, the left and right singular vectors of the mandrill image, respectively. In other words, they are *S(j,j), U(:,j), V(:,j)* of the SVD of *X* in MATLAB. Let us define the rank *k* approximation of the image $\boldsymbol{X}$ as $\boldsymbol{X}_k := \sigma_1 \boldsymbol{u}_1 \boldsymbol{v}'_1 + \ldots + \sigma_k \boldsymbol{u}_k \boldsymbol{v}'_k$, where $\boldsymbol{v}'_j$ is a transpose of $\boldsymbol{v}_j$.

   Then, for *k=1,6,11,31*, compute $\boldsymbol{X}_k$ of the mandrill, and display the results. Fit these four images in one page by using `subplot` function in MATLAB (i.e., use `subplot(2,2,1)` to display the first image, `subplot(2,2,2)` to display the second image, etc.)
4. For *k=1,6,11,31*, display the residuals, i.e., $\boldsymbol{X}-\boldsymbol{X}_k$, fit them in one page, and print them.

matlab commands

```
load mandrill;
who;

image(X);
colormap(map);
size(X)
[U,S,V]=svd(X);
figure(2)
plot(diag(S));
semilogy(diag(S));
grid
figure(3)
image(S(1,1)*U(:,1)*V(:,1)');
image(X);
figure(3);image(S(1,1)*U(:,1)*V(:,1)');
colormap(map)
app=S(1,1)*U(:,1)*V(:,1)';
```

```
image(app);
app=app+S(2,2)*U(:,2)*V(:,2)';
image(app);
app=app+S(3,3)*U(:,3)*V(:,3)';
image(app)
for k=4:31
    app=app+S(k,k)*U(:,k)*V(:,k)';
   end
image(app)
colormap(map)
```

5. Finally, submit those three printouts as HW.

Thanks to Professor Naoki Saito for providing these assignments