1.A Walk Through Linear Models

(a)perceptron

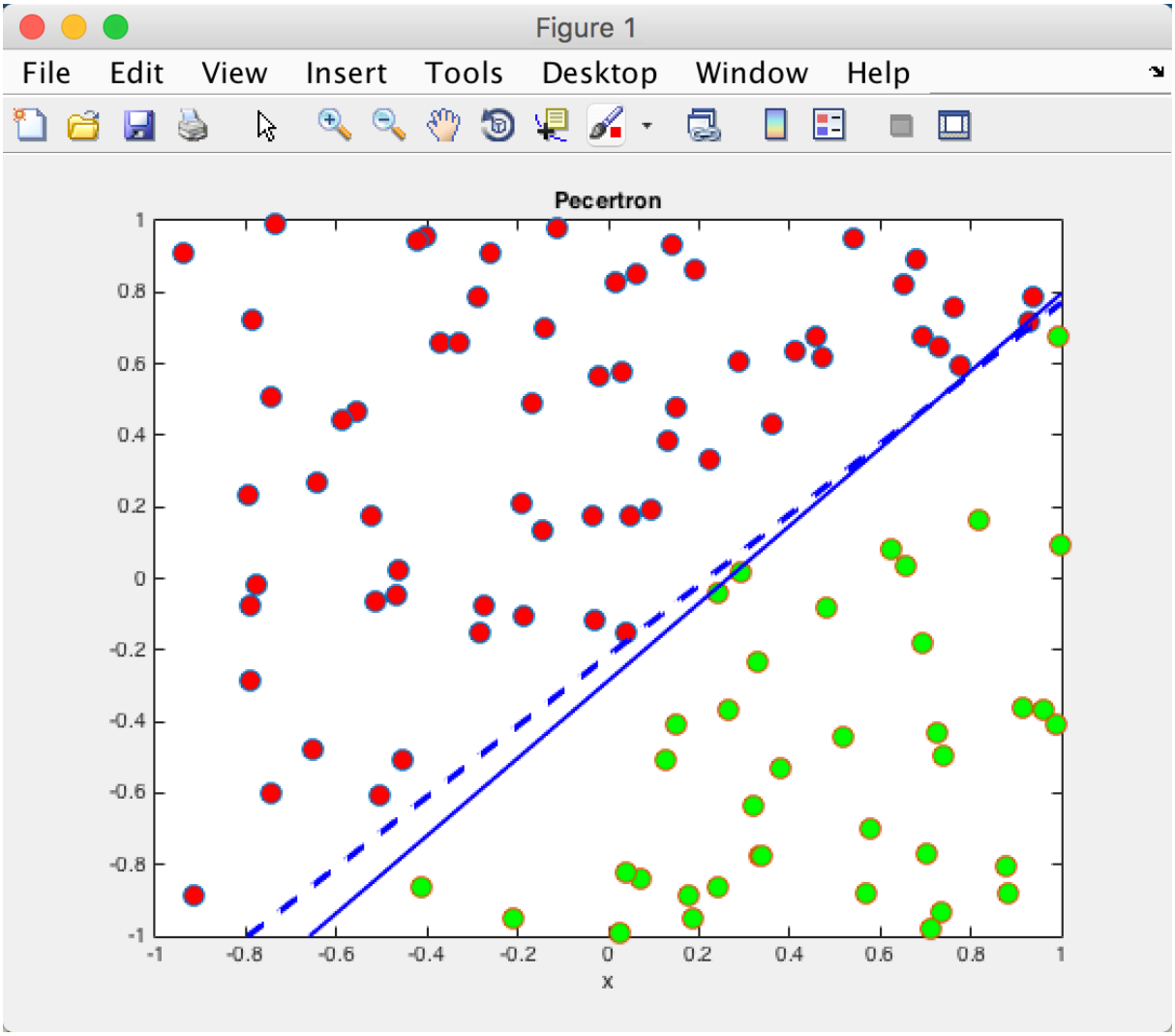
Since the signed distance from x_i to the decision boundary is $\frac{\beta^T x_i+\beta_0}{\|\beta\|}$, the distance from a misclassified x_i to the decision boundary is $\frac{-y_i(\beta^T x_i+\beta_0)}{\|\beta\|}$.

The goal is to minimize:

$$D(\beta,\beta_0) = -\sum_{i\in\mathcal{M}} y_i(\beta^T x_i + \beta_0) \ .$$

$$\frac{\partial D(\beta,\beta_0)}{\partial \beta} = -\sum_{i\in\mathcal{M}} y_i x_i$$

$$\frac{\partial D(\beta,\beta_0)}{\partial \beta_0} = -\sum_{i\in\mathcal{M}} y_i \ .$$



(i)perceptron result:

when training set is 10, testing set is 10000, the error rate is:

```
E_train is 0.000000, E_test is 0.408000.
E_train is 0.000000, E_test is 0.123700.
E_train is 0.000000, E_test is 0.107800.
E_train is 0.000000, E_test is 0.079100.
E_train is 0.000000, E_test is 0.012600.
E_train is 0.000000, E_test is 0.053800.
E_train is 0.000000, E_test is 0.065200.
E_train is 0.000000, E_test is 0.167500.
E_train is 0.000000, E_test is 0.042500.
E_train is 0.000000, E_test is 0.026600.
Average number of iterations is 5.300000e+00
```

```
expected E_train is 0.002000, expected E_test is 0.120472.
```

when training set is 100, testing set is 10000, the error rate is:

```
E_train is 0.000000, E_test is 0.004400.
E_train is 0.000000, E_test is 0.044000.
E_train is 0.000000, E_test is 0.066300.
E_train is 0.000000, E_test is 0.011700.
E_train is 0.000000, E_test is 0.018700.
E_train is 0.000000, E_test is 0.010300.
E_train is 0.000000, E_test is 0.008600.
E_train is 0.000000, E_test is 0.007700.
E_train is 0.000000, E_test is 0.004200.
E_train is 0.000000, E_test is 0.018000.
Average number of iterations is 1.370000e+01
```

```
expected E_train is 0.001220, expected E_test is 0.013978.
```

(ii)

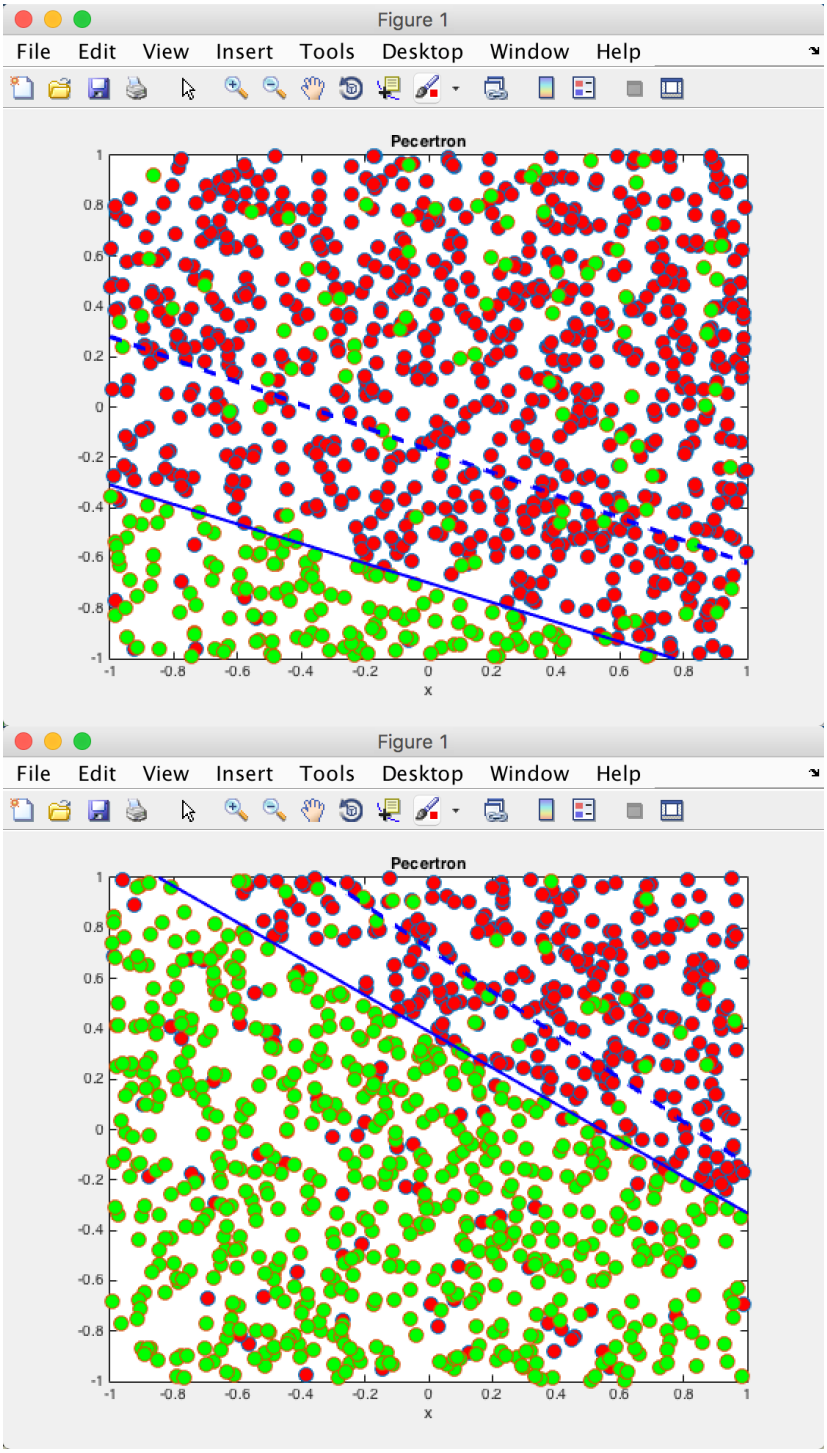
when the size of training set is 10:

```
Average number of iterations is 5.300000e+00
```

when the size of training set is 100:

```
Average number of iterations is 1.370000e+01
```

(iii)Perceptron but with noise:

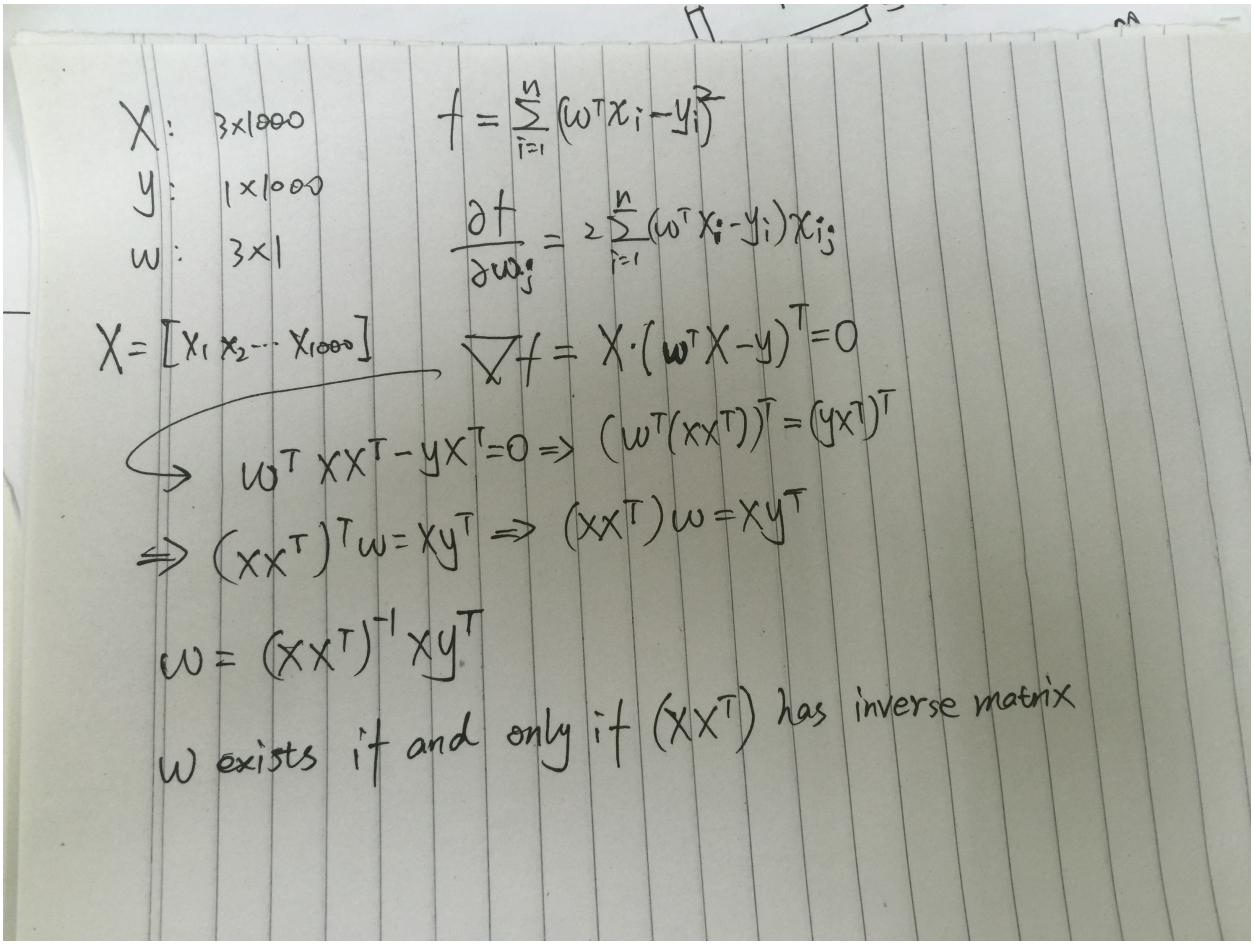


the perceptron could not separate data with noise correctly.

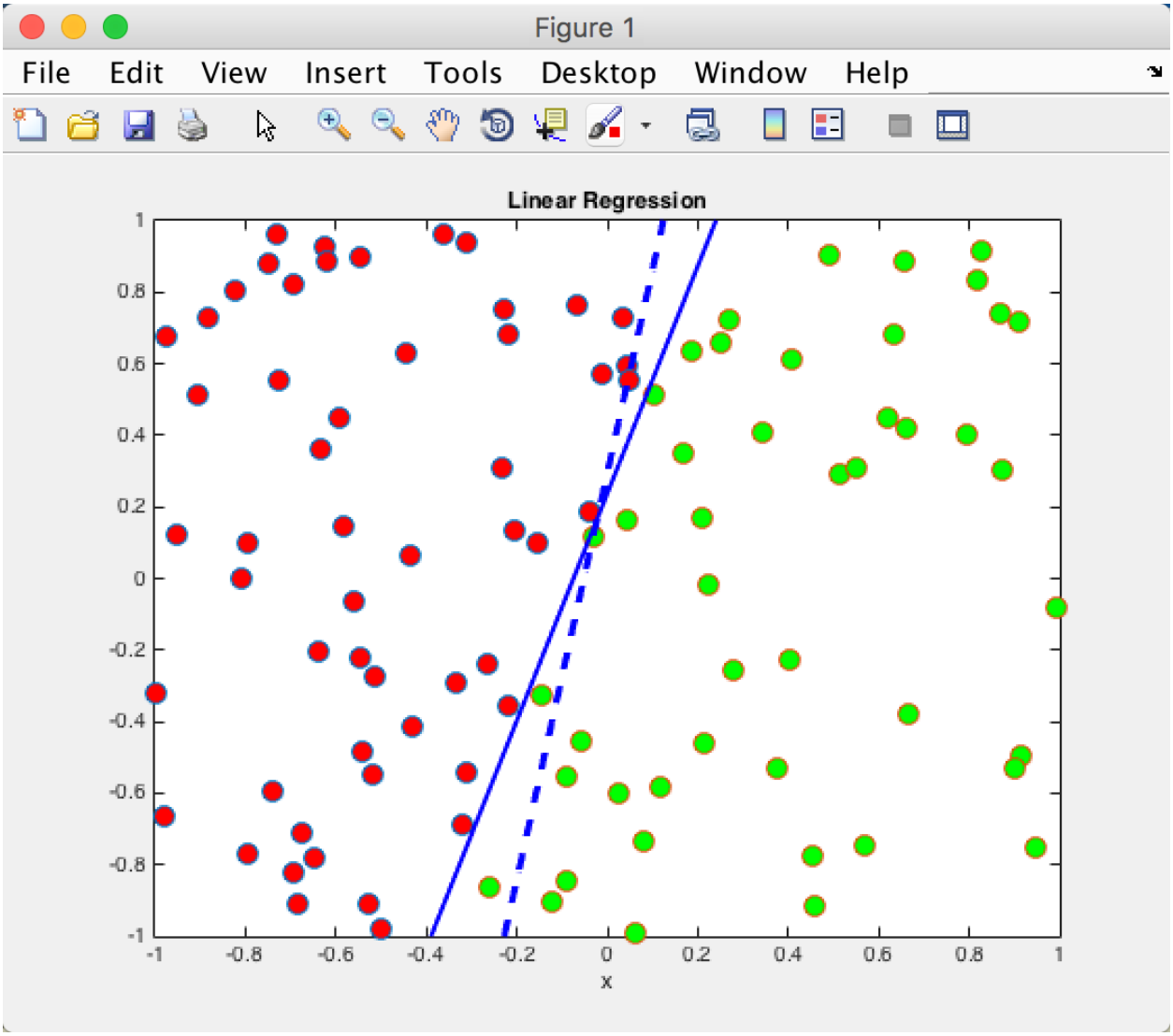
(b)linear regression

Our aim is to minimize cost function! I copied this picture from http://ufldl.stanford.edu/tutorial/supervised/LinearRegression/.

$$J(\theta) = \frac{1}{2}\sum_i \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2 = \frac{1}{2}\sum_i \left(\theta^\top x^{(i)} - y^{(i)}\right)^2$$



This is the result of Linear regression

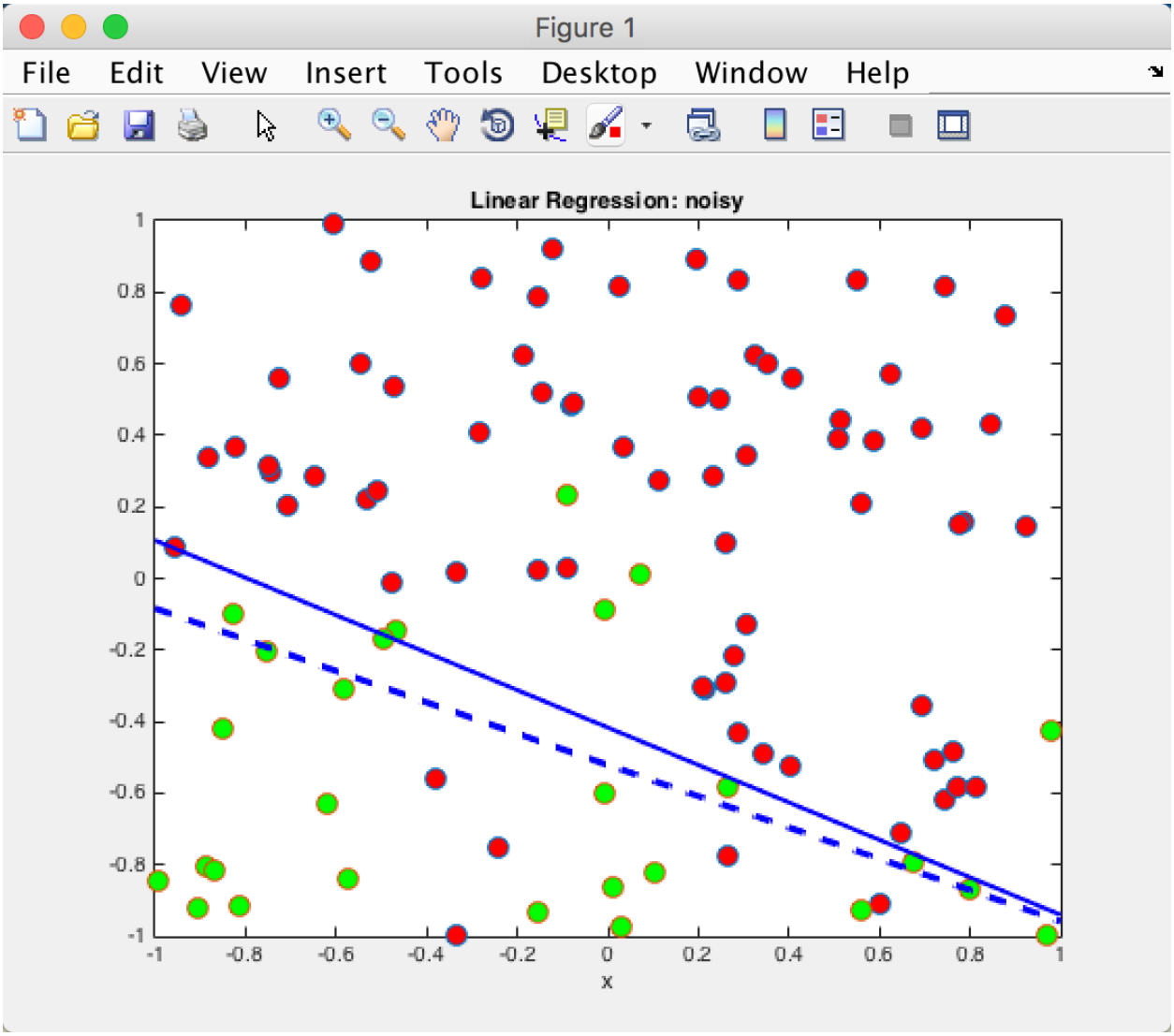


(i)training error and testing error when training size is 100, testing size is 10000:

```
E_train is 0.110000, E_test is 0.077400.
E_train is 0.020000, E_test is 0.025200.
E_train is 0.020000, E_test is 0.040100.
E_train is 0.070000, E_test is 0.072900.
E_train is 0.080000, E_test is 0.040400.
E_train is 0.040000, E_test is 0.069000.
E_train is 0.040000, E_test is 0.072900.
E_train is 0.040000, E_test is 0.075800.
E_train is 0.010000, E_test is 0.017100.
E_train is 0.000000, E_test is 0.003300.
```

```
expected E_train is 0.039200, expected E_test is 0.049483.
```

(ii)training error and testing error when training size is 100 and testing size is 10000 with noise:




```
E_train is 0.140000, E_test is 0.137100.
E_train is 0.130000, E_test is 0.178400.
E_train is 0.140000, E_test is 0.206400.
E_train is 0.110000, E_test is 0.128200.
E_train is 0.110000, E_test is 0.122000.
E_train is 0.130000, E_test is 0.117800.
E_train is 0.150000, E_test is 0.136300.
E_train is 0.190000, E_test is 0.153700.
E_train is 0.130000, E_test is 0.117000.
E_train is 0.210000, E_test is 0.174600.
```

expected E_train is 0.135540, expected E_test is 0.148853.

(iii)

```
E_train is 0.490000, E_test is 0.549600.
```

(iv) the second line is the result after the transformation:

$(1, x_1, x_2) \rightarrow (1, x_1, x_2, x_1x_2, x^2_1, x^2)$.

~~E_train is 0.490000, E_test is 0.549600.~~

```
E_train is 0.050000, E_test is 0.066000.
```

(c)logistic regression

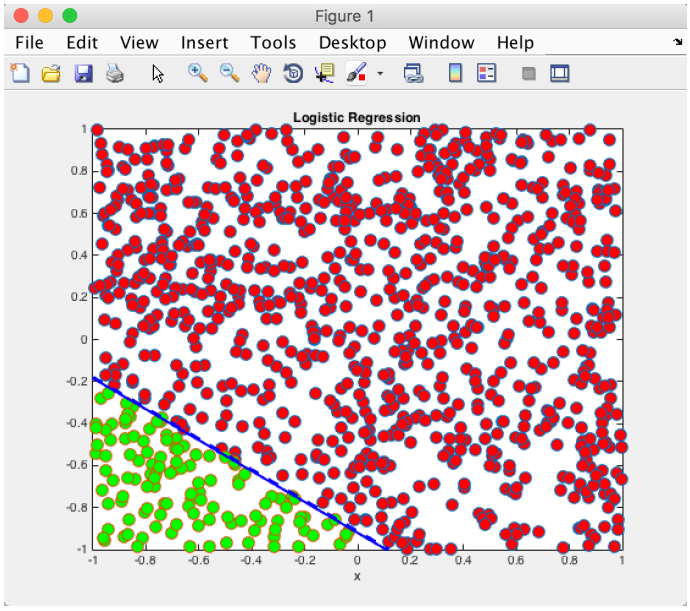
$$P(y = 1|x) = h_\theta(x) = \frac{1}{1 + \exp(-\theta^\top x)} \equiv \sigma(\theta^\top x),$$
$$P(y = 0|x) = 1 - P(y = 1|x) = 1 - h_\theta(x).$$

cost function is:

$$J(\theta) = -\sum_i \left(y^{(i)} \log(h_\theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))\right).$$

gradient is:

$$\nabla_\theta J(\theta) = \sum_i x^{(i)}(h_\theta(x^{(i)}) - y^{(i)})$$



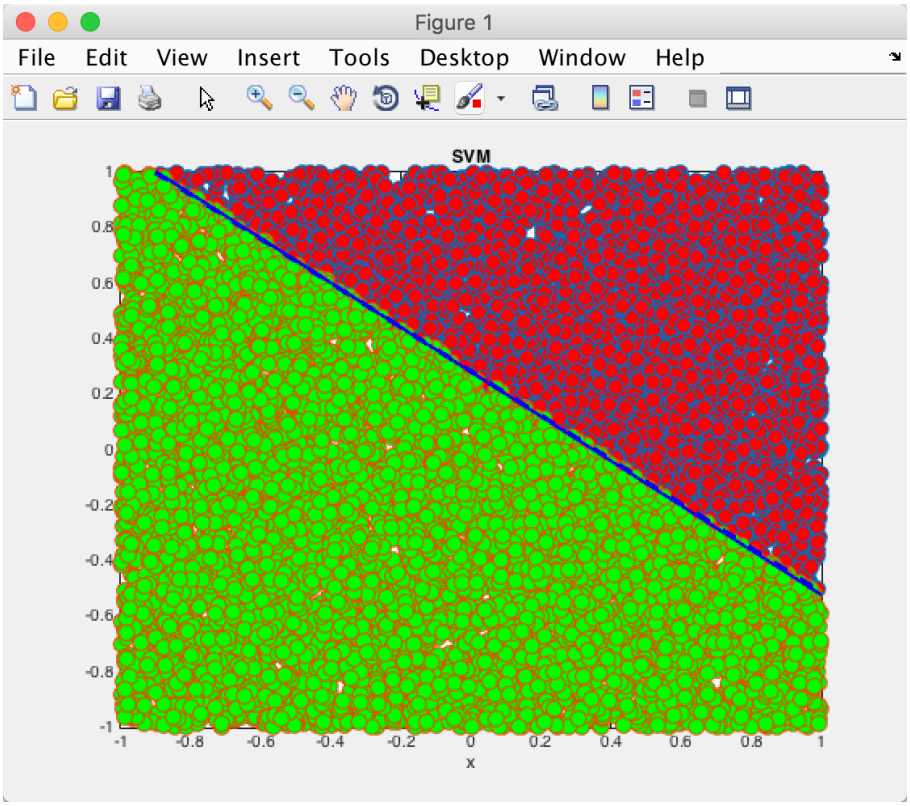
(i)the training error and testing error with the training size= 100 and testing size= 10000

```
E_train is 0.010000, E_test is 0.006700.
E_train is 0.000000, E_test is 0.005700.
E_train is 0.250000, E_test is 0.231200.
E_train is 0.000000, E_test is 0.012500.
E_train is 0.010000, E_test is 0.010400.
E_train is 0.000000, E_test is 0.004500.
E_train is 0.000000, E_test is 0.000900.
E_train is 0.000000, E_test is 0.002600.
E_train is 0.010000, E_test is 0.010700.
E_train is 0.000000, E_test is 0.005300.   ......

expected E_train is 0.006900, expected E_test is 0.015980.
```

(ii) error with the training size= 100 and testing size= 10000 **with noise**

```
E_train is 0.200000, E_test is 0.213300.
E_train is 0.140000, E_test is 0.141300.
E_train is 0.240000, E_test is 0.222000.
E_train is 0.170000, E_test is 0.175600.
E_train is 0.100000, E_test is 0.123400.
E_train is 0.220000, E_test is 0.277500.
E_train is 0.200000, E_test is 0.128600.
E_train is 0.200000, E_test is 0.161600.
E_train is 0.080000, E_test is 0.124900.
E_train is 0.180000, E_test is 0.161600.   ......

expected E_train is 0.152800, expected E_test is 0.170060.
```

(d)SVM



(i) training size=30 testing size=10000

```
E_train is 0.000000, E_test is 0.051400.
E_train is 0.000000, E_test is 0.033200.
```

.....

```
expected E_train is 0.000000, expected E_test is 0.038484.
```

(ii) training size=100 testing size=10000

```
E_train is 0.000000, E_test is 0.005800.
E_train is 0.000000, E_test is 0.001800.
```

.....

```
expected E_train is 0.000000, expected E_test is 0.011071.
```

(iii)training size=100

```
average number of support vectors is3.309000e+00
```

2.Regularization and Cross-Validation

result:

```
testing lambda:1
    12

testing lambda:2
    12

testing lambda:3
    12

testing lambda:4
    48

testing lambda:5
    11

testing lambda:6
    10

testing lambda:7
     8

testing lambda:8
     8

lambda=1000.000000 sigma w^2=0.029849
training error is0.010000
lambda=0.000000 sigma w^2=21.011096
training error is0.000000
lambda=1000.000000 testing error is0.059267
lambda=0.000000 testing error is0.297338
```

(a) the lambda chosen by LOOCV is 100 or 1000
(b) with lambda=1000
∑w^2=0.029849
with lambda=0
∑w^2=21.011096

(c) with lambda=1000
training error=0

testing error=0.059267
with lambda=0
training error=0
testing error=0.297338
(d) I regard 20 columns as one validation set. So in the training data, X is 784*200, there are 10 sets to do validation. Here is my LOOCV result:

So I should choose lambda=lambdas(6)=10

```
14
        for i = 1:length(lambdas)
24          E_val = 0;
            for j = 1:20:size(X, 2)
12              % take point j out of X
                X_ = [X(:,1:j-1), X(:,j+20:end)]; y_ = [y(1,1:j-1), y(1,j+20:end)];
12                  w = logistic_r(X_, y_, lambdas(i));
                    pred=h_theta(w,[X(:,j:j+19);ones(1,20)]);
14                  pred(pred>0.5)=1;
                    pred(pred<=0.5)=0;
 8                  pred=pred-y(j:j+19);
                    E_val=E_val+sum(pred~=0);
28          end
17
```

```
lambda=10.000000 sigma w^2=2.070765
training error is0.000000
lambda=0.000000 sigma w^2=55434.200601
training error is0.000000
lambda=10.000000 testing error is0.054746
lambda=0.000000 testing error is0.063285
finished
```

3.Bias Variance Trade-off

(i)false (ii)false (iii)true (iv)false (v)false

4.Neural Network vs. SVM

(a)

```
Error rate for NN is 0.024800.
```

```
Error rate for SVM is 0.116400.
```