```
import numpy as np
import numpy.ma as ma
from scipy.interpolate import griddata
from numpy.random import uniform, seed
from matplotlib import cm

def gen_label(X):
      return np.random.binomial(1, 1.0 / (1.0 + np.exp(-np.matmul(X, (-2.0, 1.0)))))

def IRLS(X, y):
      weight = np.zeros(shape=(2))
      # bias = np.log(np.mean(y) / (1.0 - np.mean(y)))
      bias = 0
      #print(bias)
      threshold = 1e-6
      change = 1e6
      while(change >= threshold):
            ita = np.matmul(X, weight) + bias
            #print(ita)
            miu = 1.0 / (1.0 + np.exp(-ita))
            #print(miu)
            s = np.multiply(miu, 1.0 - miu)
            z = ita + (y - miu) / s
            #print(s, z)
            s = np.diag(s).copy()
            XTSX_inv = np.linalg.inv(np.matmul(np.matmul(np.transpose(X), s), X))
            XTSz = np.matmul(np.matmul(np.transpose(X), s), z)
            weight_new = np.matmul(XTSX_inv, XTSz)
            change = np.linalg.norm(weight_new - weight)
            weight = weight_new
      return weight

def get_fisher_information(X, beta):
      prob = 1.0 / (1.0 + np.exp(-np.matmul(X, (-2.0, 1.0))))
      w = np.multiply(prob, 1 - prob)
      w = np.diag(w).copy()
      return np.matmul(np.matmul(np.transpose(X), w), X)
```