PROBLEMS

Due Friday Feb. 24, 2023 @ 4pm

1. Should the following be tackled as regression or classification problems? Some are a bit tricky. Think carefully and explain your answers.

   (a) Given a person's music library and number of times they have listened to each song, predict how many stars (from 1 to 5) they will rate a new song.

   (b) Given a person's credit rating, predict whether they will file for bankrupcy or not in the next year.

   (c) Given a family's combined salary and home value, predict how many cars they will own.

   (d) Given a partially obscured picture of a vehicle, predict how many wheels it has.

   (e) Given an MRI image, identify the locations (expressed in pixel coordinates) of tumors.

2. Suppose we collect data for a group of students in a machine learning class with variables x_1 = hours studied, x_2 = undergrad GPA, and $y = 1$ if receive an A or $y = 0$ if not. We fit a logistic regression model and produce the coefficients $b = -6,\ w_1 = 0.1,\ w_2 = 1$.

   (a) Using the variables $y, x_1, x_2, b, w_1, w_2, z$ (where z is the score), write an equation for the probability of receiving an A given the numbers of hours studied and the undergrad GPA.

   (b) Estimate the probability that a student who studies for 40 hours and has an undergrad GPA of 3.5 gets an A in the class.

   (c) How many hours would the student in part (a) need to study to have a 50% chance of getting an A in the class?

3. Suppose that a logistic model for a binary class label $y \in \{0, 1\}$ is given by

$$\Pr\{y = 1|\mathbf{x}, b, w_1, w_2\} = \frac{1}{1 + e^{-z}}, \quad z = b + w_1x_1 + w_2x_2,$$

   where $[b, w_1, w_2] = [2, 4, 6]$. Describe the following sets:

   (a) The set of $\mathbf{x} = [x_1, x_2]^\mathsf{T}$ such that $\Pr\{y = 1|\mathbf{x}, b, w_1, w_2\} > \Pr\{y = 0|\mathbf{x}, b, w_1, w_2\}$. *Hint*: First write these probabilities in terms of z, then write z in terms of x_1, x_2, b, w_1, w_2.

   (b) The set of $\mathbf{x}$ such that $\Pr\{y = 1|\mathbf{x}, b, w_1, w_2\} > 0.75$.

   (c) The set of x_1 such that $\Pr\{y = 1|\mathbf{x}, b, w_1, w_2\} > 0.75$ and $x_2 = 1$.

4. A data scientist is hired by the IRS to screen taxpayers for potential tax fraud. The data scientist decides to use two features for each taxpayer:

- x_1 = the taxpayer's income (in thousands of dollars), and
- x_2 = the number of foreign bank accounts they hold.

To train the model, the data scientist uses the results of tax investigations from previous years. Imagine that she has the following training data:

Income (thousands \$), x_{i1}	300	500	700	800	1000
# foreign accounts, x_{i2}	0	1	1	2	1
Fraud (1=yes, 0=no), y_i	0	1	0	1	1

(a) Draw a scatter plot of the data, using different markers for each of the two classes.

(b) Find a linear classifier that makes at most one error on the training data. The classifier should be of the form

$$\widehat{y}_i = \begin{cases} 1 & \text{if } z_i > 0 \\ 0 & \text{if } z_i < 0, \end{cases} \qquad z_i = b + [w_1, w_2]\mathbf{x}_i \;\text{ for }\; \mathbf{x}_i = \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}.$$

What are the weights $[b, w_1, w_2]$ for your classifier? (There is more than one correct answer, but a simple classifier will make the next part easier!)

(c) Now consider a logistic model of the form

$$\Pr\{y_i = 1 | \mathbf{x}_i, b, w_1, w_2\} = \frac{1}{1 + e^{-z_i}}, \quad z_i = b + [w_1, w_2]\mathbf{x}_i.$$

Using $[b, w_1, w_2]$ from the previous part, which sample i is the *least* likely (i.e., which i gives the smallest $p(y_i|\mathbf{x}_i, b, w_1, w_2)$)? If you do the calculations correctly, you should not need a calculator.

(d) Now consider a new set of parameters

$$[\bar{b}, \overline{w}_1, \overline{w}_2] = [\alpha b, \alpha w_1, \alpha w_2],$$

where $\alpha > 0$ is an arbitrary positive scalar. Would using the new parameters change the prediction $\widehat{y}$ relative to part (b)? Would they change the likelihood value $p(y_i|\mathbf{x}_i, \bar{b}, \overline{w}_1, \overline{w}_2)$ relative to the likelihood value $p(y_i|\mathbf{x}_i, b, w_1, w_2)$ from part (c)? If they do not change, state why. If they do change, qualitatively describe the change as a function of α.

1. Should the following be tackled as regression or classification problems? Some are a bit tricky. Think carefully and explain your answers.

   (a) Given a person's music library and number of times they have listened to each song, predict how many stars (from 1 to 5) they will rate a new song.

   (b) Given a person's credit rating, predict whether they will file for bankrupcy or not in the next year.

   (c) Given a family's combined salary and home value, predict how many cars they will own.

   (d) Given a partially obscured picture of a vehicle, predict how many wheels it has.

   (e) Given an MRI image, identify the locations (expressed in pixel coordinates) of tumors.

a) regression, because 1-5 is ordinal

b) classification, because the output is yes or no

c) regression, because # cars is not a distinct category

d) classification, with input as a photo, the values of wheel count are discrete categories

e) regression because the location is needed
Classification would only yield contains or does not contain predictions

2. Suppose we collect data for a group of students in a machine learning class with variables x_1 = hours studied, x_2 = undergrad GPA, and $y = 1$ if receive an A or $y = 0$ if not. We fit a logistic regression model and produce the coefficients $b = -6$, $w_1 = 0.1$, $w_2 = 1$.

   (a) Using the variables $y, x_1, x_2, b, w_1, w_2, z$ (where z is the score), write an equation for the probability of receiving an A given the numbers of hours studied and the undergrad GPA.

   (b) Estimate the probability that a student who studies for 40 hours and has an undergrad GPA of 3.5 gets an A in the class.

   (c) How many hours would the student in part (a) need to study to have a 50% chance of getting an A in the class?

Logistic regression model

$$\ln(\text{odds}) = B_0 + B_1 X_1 + B_2 X_2 \rightarrow \text{odds} = \frac{P(x)}{1-P(x)}$$

$$\downarrow$$

$$\text{odds} = e^{B_0 + B_1 X_1 + B_2 X_2}$$

$$\frac{P(x)}{1-P(x)} = e^{B_0 + B_1 x_1 + B_2 x_2}$$

$$\Rightarrow P(x) = \frac{e^{B_0 + B_1 X_1 + B_2 X_2}}{1 + e^{B_0 + B_1 X_1 + B_2 X_2}}$$

b) Given, B_0, B_1, B_2, X_1, X_2

$$\Rightarrow \ln(\text{odds}) = -6 + 0.05(40) + 1(3.5)$$

$$= [-6 + \cancel{0.05(40)}^{\,2} + 1(3.5)]$$

e^x (

$$\frac{P(x)}{1-P(x)} = e^{-.5}$$

$$P(x) = e^{-.5}(1 - P(x))$$

$$= e^{-.5} - e^{-.5}P(x)$$

$$P(x) \cdot (1 + e^{-.5}) = e^{-.5}$$

$$P(x) = \frac{e^{-.5}}{1 + e^{-.5}} = 0.378$$

c) Solve for X_1

$$X_1 = [\ln(\text{odds}) - B_0 - B_2 X_2] \frac{1}{B_1}$$

$$X_1 \left(\text{odds} = \frac{1/2}{1 - 1/2} = 1,\ B_0 = -6,\ B_1 = 0.05,\ B_2 = 1,\ X_2 = 3.5\right)$$

$$= (\cancel{\ln(1)}^{\,0} + 6 - 3.5)20 = 50 \text{ hours}$$

Sigmoid function

3. Suppose that a logistic model for a binary class label $y \in \{0, 1\}$ is given by

$$\Pr\{y = 1|\mathbf{x}, b, w_1, w_2\} = \frac{1}{1 + e^{-z}}, \quad z = b + w_1x_1 + w_2x_2,$$

where $[b, w_1, w_2] = [2, 4, 6]$. Describe the following sets:

(a) The set of $\mathbf{x} = [x_1, x_2]^\mathsf{T}$ such that $\Pr\{y = 1|\mathbf{x}, b, w_1, w_2\} > \Pr\{y = 0|\mathbf{x}, b, w_1, w_2\}$. *Hint*: First write these probabilities in terms of z, then write z in terms of x_1, x_2, b, w_1, w_2.

(b) The set of $\mathbf{x}$ such that $\Pr\{y = 1|\mathbf{x}, b, w_1, w_2\} > 0.75$.

(c) The set of x_1 such that $\Pr\{y = 1|\mathbf{x}, b, w_1, w_2\} > 0.75$ and $x_2 = 1$.

describe the set

3a) given by $P(y=1|x) > P(y=0|x)$

$$\begin{cases} P(y=1) = \dfrac{1}{1+e^{-z}} \\ P(y=0) = 1 - P(y=1) = \dfrac{e^{-z}}{1+e^{-z}} \end{cases} \Rightarrow \quad \frac{1}{1+e^{-z}} > \frac{e^{-z}}{1+e^{-z}}$$

or

$$\frac{1}{1+e^{-z}} > 0.5 \quad \leftarrow \text{better} \quad — [1]$$

$$\frac{1}{1+e^{-z}} > 0.5 \quad — [1]$$

$$2 > (1 + e^{-z})$$

$$1 > e^{-z}$$

$$\ln(1) > -z$$

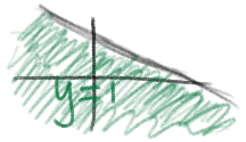
$$0 > -z \quad — [2]$$

where $z = b + w_1 x_1 + w_2 x_2$

$$= 2 + 4x_1 + 6x_2 \quad — [3]$$

[2] and [3] $\Rightarrow \quad 0 > -(2 + 4x_1 + 6x_2)$

$2 + 4x_1 + 6x_2 > 0 \Rightarrow$ this expression describes the decision boundry in the X space which resembles



3b) $\frac{1}{1+e^{-z}} > 0.75$

$\frac{4}{3} > 1 + e^{-z}$

$\ln\frac{4}{3} > \cancel{\ln(1)}^{0} - z$

$\ln\frac{4}{3} > -z$ ———— [4]

from [3] and [4]

$\ln\frac{4}{3} > -(2 + 4x_1 + 6x_2)$

$\ln\frac{4}{3} < 2 + 4x_1 + 6x_2$ — the decision boundry

3c) if $x_2 = 1$, Continuing from 3b)

$\ln\frac{4}{3} < 8 + 4x_1$

$x_1 > \ln\frac{4}{3} - 8$

> -7.71

$\Rightarrow$ 

4. A data scientist is hired by the IRS to screen taxpayers for potential tax fraud. The data scientist decides to use two features for each taxpayer:

- x_1 = the taxpayer's income (in thousands of dollars), and
- x_2 = the number of foreign bank accounts they hold.

To train the model, the data scientist uses the results of tax investigations from previous years. Imagine that she has the following training data:

Income (thousands \$), x_{i1}	300	500	700	800	1000
# foreign accounts, x_{i2}	0	1	1	2	1
Fraud (1=yes, 0=no), y_i	0	1	0	1	1

(a) Draw a scatter plot of the data, using different markers for each of the two classes.

(b) Find a linear classifier that makes at most one error on the training data. The classifier should be of the form

$$\hat{y}_i = \begin{cases} 1 & \text{if } z_i > 0 \\ 0 & \text{if } z_i < 0, \end{cases} \qquad z_i = b + [w_1, w_2]\mathbf{x}_i \quad \text{for} \quad \mathbf{x}_i = \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}.$$

What are the weights $[b, w_1, w_2]$ for your classifier? (There is more than one correct answer, but a simple classifier will make the next part easier!)

(c) Now consider a logistic model of the form

$$\Pr\{y_i = 1|\mathbf{x}_i, b, w_1, w_2\} = \frac{1}{1+e^{-z_i}}, \qquad z_i = b + [w_1, w_2]\mathbf{x}_i.$$

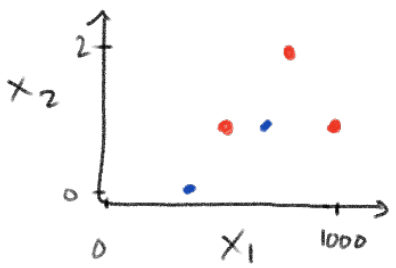
Using $[b, w_1, w_2]$ from the previous part, which sample i is the *least* likely (i.e., which i gives the smallest $p(y_i|\mathbf{x}_i, b, w_1, w_2)$)? If you do the calculations correctly, you should not need a calculator.

(d) Now consider a new set of parameters

$$[\overline{b}, \overline{w}_1, \overline{w}_2] = [\alpha b, \alpha w_1, \alpha w_2],$$

where $\alpha > 0$ is an arbitrary positive scalar. Would using the new parameters change the prediction $\hat{y}$ relative to part (b)? Would they change the likelihood value $p(y_i|\mathbf{x}_i, \overline{b}, \overline{w}_1, \overline{w}_2)$ relative to the likelihood value $p(y_i|\mathbf{x}_i, b, w_1, w_2)$ from part (c)? If they do not change, state why. If they do change, qualitatively describe the change as a function of α.

4a) scatter plot



b) linear classifier

$b = -1$

$w_1 = 0$

$w_2 = 2$

$\Rightarrow$

$z_i = b + [w_1, w_2]\, x_i$

$z = [-1, 1, 1, 3, 1]$ ← score

only 1 error for $i = 3$, false positive

ben

c) for the z_i's generated, which yields the smallest Probability $y=1$ given the logistic model

$$Pr(y=1) = \frac{1}{1+e^{-z_i}}$$

z_1 yields the smallest probability

because it maximizes e^{-z} as the only negative z

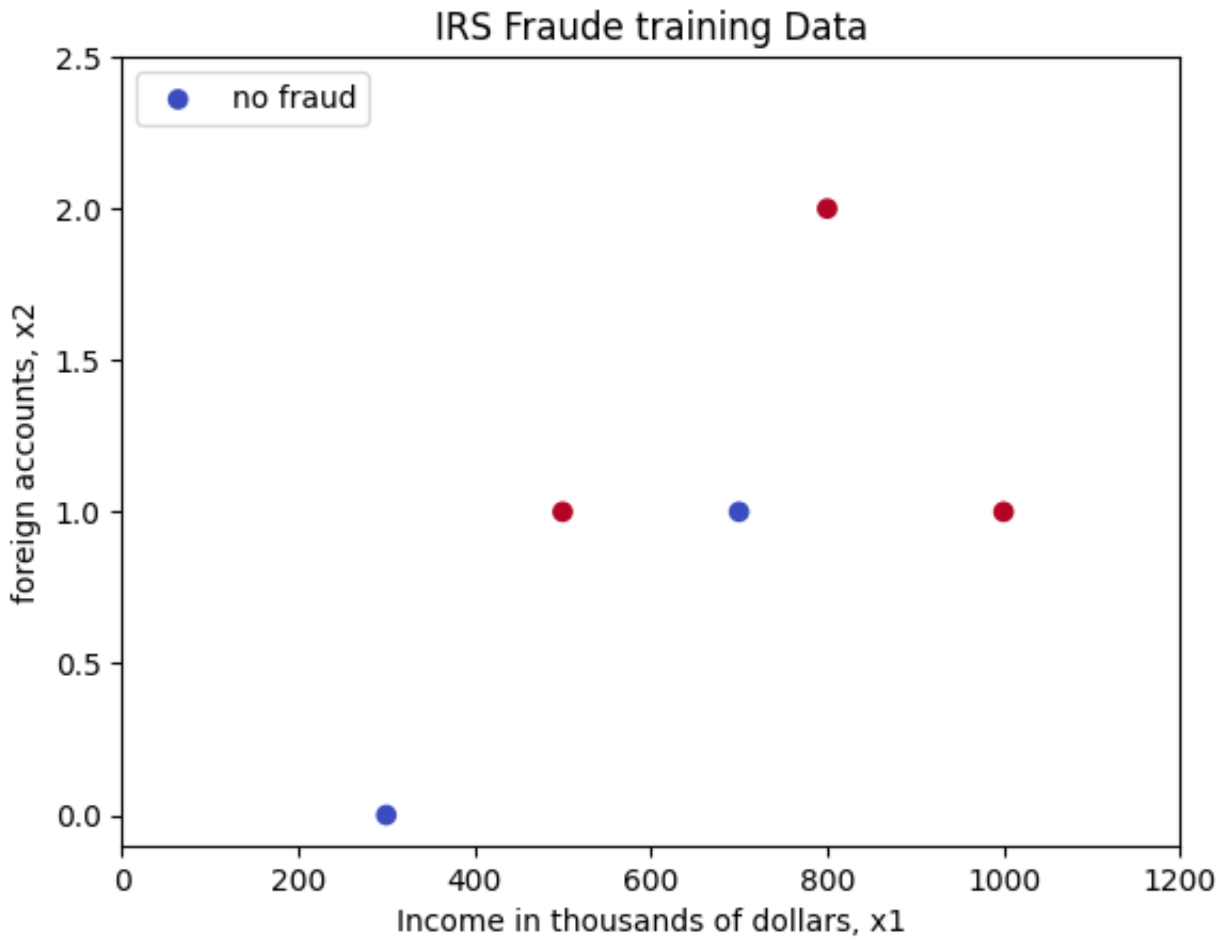
d) the linear model would not change its Predictions because the coeffients' ratios are what determine $>$ or $<$ 0 the same goes for the logistic model because z of < 0 will always result in a smaller probability than $z >$

ECE 5307 Excerise Set 5

```
Authored by Brian Lesko, a Graduate Researcher and Teaching
Associate, a Masters of mechanical engineering student,
studying mechatronic controls, robotics, and machine
learning.

This documenet is originally an interactive python notebook

Contents:
Coding Verification for problem 4
```

a) Plot the IRS fraud data given

In [ ]:
```
import matplotlib.pyplot as plt
import numpy as np

x1 = np.array([300,500,700,800,1000])
x2 = np.array([0,1,1,2,1])
y = np.array([0,1,0,1,1])

# define a function to plot the data
def plot_data(x1, x2, y):
    plt.scatter(x1, x2, c=y, cmap=plt.cm.coolwarm)
    plt.title('IRS Fraude training Data')
    plt.xlabel('Income in thousands of dollars, x1')
    plt.ylabel('foreign accounts, x2')
    plt.legend(['no fraud', 'fraud'], loc='upper left')
    #center the axes at 0
    plt.xlim(0, 1200)
    plt.ylim(-.1, 2.5)


plot_data(x1, x2, y)
```



b) Find a linear classifier that makes at most one error on the training data

In [ ]:
```
# find a linear classifier and plot the decision boundary that makes at most
import numpy as np

b = -1
w1 = 0
w2 = 2

z = w1*x1 + w2*x2 + b

# define a vector correct that is 1 if the classifier is correct and 0 other
correct = np.zeros(5)
for i in range(5):
    if y[i] == 1 and z[i] > 0:
        correct[i] = 1
    elif y[i] == 0 and z[i] < 0:
        correct[i] = 1
    else:
        correct[i] = 0

# if z>0 then yhat =1 and if z<0 then yhat =0
yhat = np.zeros(5)
for i in range(5):
    if z[i] > 0:
        yhat[i] = 1
    else:
        yhat[i] = 0
print(y)
print(yhat)
print('The classifier made', 5 - np.sum(correct), 'errors on the training da
```

```
[0 1 0 1 1]
[0. 1. 1. 1. 1.]
The classifier made 1.0 errors on the training data
```

In [ ]:
```
print(z)
```

```
[-1  1  1  3  1]
```

In [ ]:
```
# using the logistic model with z
def logistic(z):
    return 1/(1+np.exp(-z))

probabilities = logistic(z)
print(probabilities)
```

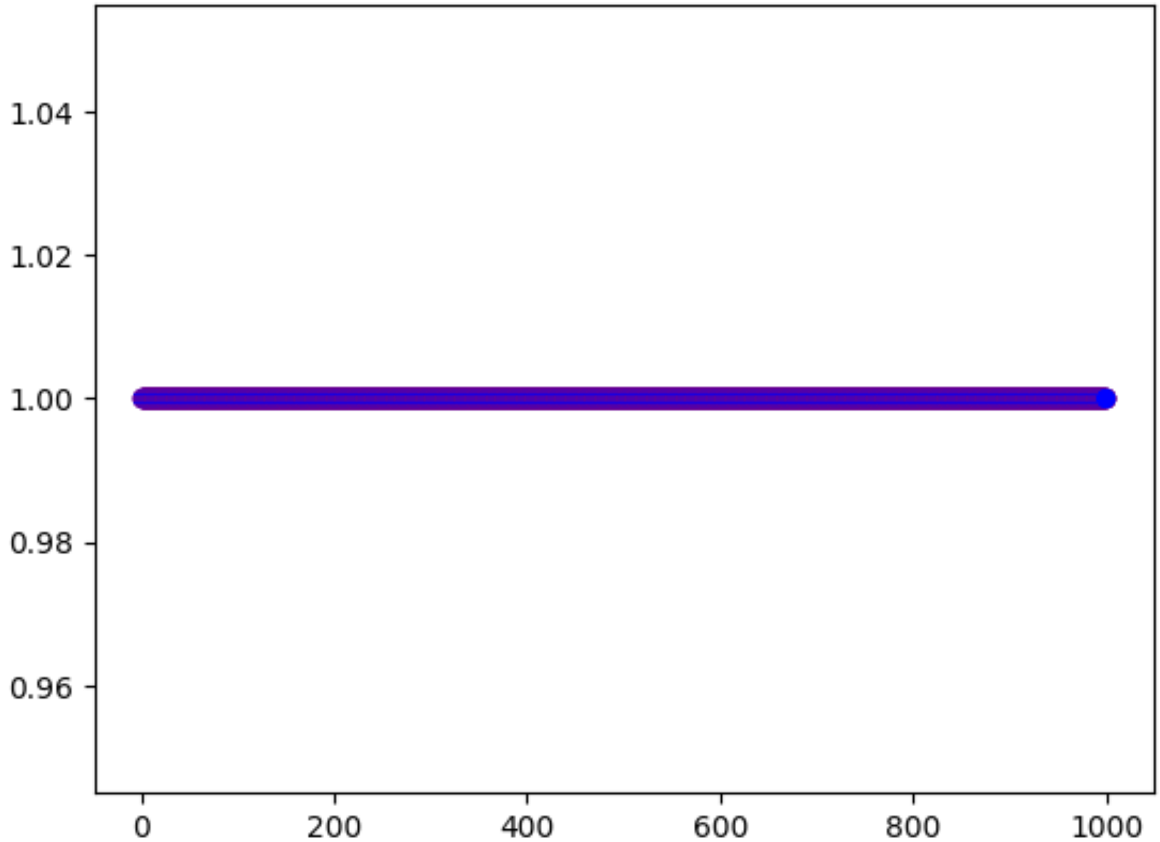
```
[0.26894142 0.73105858 0.73105858 0.95257413 0.73105858]
```

In [ ]:
```
#add a scaling parameter alpha
alpha = 0.1

z_alpha = (w1*x1 + w2*x2 + b) * alpha
print(z_alpha)

# define a function linear_errors that takes z and y as inputs and returns t
def linear_errors(z, y):
    errors = 0
    for i in range(5):
        if y[i] == 1 and z[i] > 0:
            errors += 0
        elif y[i] == 0 and z[i] < 0:
            errors += 0
        else:
            errors += 1
    return errors
errors = linear_errors(z, y)
print(errors)

# define a function logistic_errors that takes probabilities and outputs the
def logistic_errors(probabilities, y):
    errors = 0
    for i in range(5):
        if probabilities[i] > 0.5:
            yhat = 1
        else:
            yhat = 0
        if y[i] != yhat:
            errors += 1
    return errors
```

```
[-0.1  0.1  0.1  0.3  0.1]
1
```

In [ ]:
```
#for a range of alphas make a for loop
for alpha in np.arange(1, 1000, 1):
    z_alpha = (w1*x1 + w2*x2 + b) * alpha
    errors_linear = linear_errors(z_alpha, y)
    probabilities = logistic(z_alpha)
    errors_logistic = logistic_errors(probabilities, y)
    #plot the alpha vs errors_logistic
    plt.scatter(alpha, errors_logistic, c='r')
    plt.scatter(alpha, errors_linear, c='b')
```

```
/var/folders/7y/6b82r8j576s0j4p84sz0f5dc0000gn/T/ipykernel_5233/1910266207.
py:3: RuntimeWarning: overflow encountered in exp
  return 1/(1+np.exp(-z))
```


1.04
1.02
1.00
0.98
0.96
0
200
400
600
800
1000