Task 1

Estimate the difference in error rates of:

PLearn aglorithm with maxit=10

PLearn algorithm with maxit=100

...compute interval which contains the true error difference with the probability 95%.

Is the error difference statistically significant?

```
% Load the data
In1 = csvread("In1.csv")
```

```
In1 = 2×600
    0.5377   -2.2588    0.3188   -0.4336    3.5784   -1.3499    0.7254    0.7147 ···
    1.8339    0.8622   -1.3077    0.3426    2.7694    3.0349   -0.0631   -0.2050
```

```
c1 = csvread("c1.csv")
```

```
c1 = 1×600
     0     0     1     0     0     0     1     1     0     0     1     0     0 ···
```

```
% Training params
Par1 = {[1 1 -1], 1, 10}
```

```
Par1 = 1×3 cell
```

	1	2	3
1	[1,1,-1]	1	10

```
Par2 = {[1 1 -1], 1, 100}
```

```
Par2 = 1×3 cell
```

	1	2	3
1	[1,1,-1]	1	100

```
k = 5
```

```
k = 5
```

```
% run 5-fold cross-validation
[mu,sd] = CrossVal( ...
    "PLearn","PRecall",Par1, ...
    "PLearn","PRecall",Par2, ...
    In1,c1,k,"NoShuffle" ...
    )
```

```
mu = 0.0117
sd = 0.0162
```

```
% HYPOTHESIS
% H0: avg_err = 0
```

```
% H1: avg_err != 0 -> two sided t-test
a = 0.05
```

```
a = 0.0500
```

```
ts = tinv([a/2  1-a/2],k-1); % T-Score
ts % <-2.7764, 2.77764> .. acceptance region...
```

```
ts = 1×2
   -2.7764    2.7764
```

```
CI = mu + ts*sd;

% 95% confidence interval containing the error
CI % <-0.0334 , 0.0568>
```

```
CI = 1×2
   -0.0334    0.0568
```

```
% critical t value
t_c = mu - 0 / (sd/sqrt(k))  % 0.0117
```

```
t_c = 0.0117
```

```
% RESULT
% We can't reject the null hypothesis
% i.e. with 5% significance level
% % we can't say if the algorithms differs
```

Task 2

compare error rates of

algorithm Memorizer

PLearn maxit= 50, lr=1.0, init_vals=[1 1 1 1 -1]

k=6, 300 samples

```
% Load the data
In2 = csvread("In2.csv");
c2 = csvread("c2.csv");

% reset random generator
stream = RandStream.getGlobalStream; reset(stream)

% Training params...
Par1 = {}
```

```
Par1 =

  0×0 empty cell array
```

```
Par2 = {[1 1 1 1 -1], 1, 50}
```

Par2 = 1×3 cell

	1	2	3
1	[1,1,1,1...	1	50

```
k=6
```

```
k = 6
```

```
% Perfrorm the cross validation...
[mu,sd] = CrossVal( ...
    "Memorizer","MemorizerRecall",Par1, ...
    "PLearn","PRecall",Par2, ...
    In2,c2,k,"NoShuffle" ...
    )
```

```
mu = 0.5500
sd = 0.0395
```

```
% HYPOTHESIS
% H0: avg_err = 0
% H1: avg_err != 0 -> two sided t-test
a = 0.05
```

```
a = 0.0500
```

```
ts = tinv([a/2  1-a/2],k-1); % T-Score
ts % <-2.5706, 2.4706> .. acceptance region...
```

```
ts = 1×2
   -2.5706    2.5706
```

```
CI = mu + ts*sd;

% 95% confidence interval containing the error
CI % <-1.6258 , 1.6258>
```

```
CI = 1×2
    0.4485    0.6515
```

```
% critical t value
t_c = mu - 0 / (sd/sqrt(k))  % 0.0117
```

```
t_c = 0.5500
```

```
% RESULT
% We can't reject the null hypothesis
% i.e. with 5% significance level
% % we can't say if the algorithms differs
```

Final Remarks

We were not able to reject that algorithms Memorizer and PLearn differ, so based on above result both algorithm seems equally capable.

Most probably data from In2.csv c2.csv are not linary separable, thus PLearn fails and is not better than random algorithms...