Deep Learning and Labeled Patterns

表格 1

>> A= reshape(1:9,3,3)			
A =			
	1	4	7
	2	5	8
	3	6	9
>> B = reshape(1:16,4,4)			
B =			
	1	5	9	13
	2	6	10	14
	3	7	11	15
	4	8	12	16

```
>> conv2(B,A,'valid')

ans =

   192   372
   237   417
```

```
>> AA=A([3 2 1],:)

AA =

     3     6     9
     2     5     8
     1     4     7

>> BB = AA(:,[3 2 1]);
>> BB

BB =

     9     6     3
     8     5     2
     7     4     1
```

```
>> sum(sum(B(1:3,1:3).*BB))

ans =

   192

>> sum(sum(B(2:4,1:3).*BB))

ans =

   237
```

```
>> sum(sum(B(1:3,2:4).*BB))

ans =

   372

>>

>> sum(sum(B(2:4,2:4).*BB))

ans =

   417
```

HOME | PLOTS | APPS | EDITOR | PUBLISH | VIEW

Search Documentation | Sign In

New | Open | Save | Find Files | Compare | Print — FILE
Go To | Find — NAVIGATE
Insert | Comment | Indent — EDIT
Breakpoints — BREAKPOINTS
Run | Run and Advance | Run Section | Advance | Run and Time — RUN

/ ▸ Users ▸ apple ▸ Desktop ▸ Jiann-Ming Wu ▸ code2019 ▸ course2020_codes

Current Folder

Name

- flip_all.m
- my_valid_conv2_matlab.m
- my_valid_conv2.m

my_valid_conv2.m (Function)

my_valid_conv2(x, K)

Editor - /Users/apple/Desktop/Jiann-Ming Wu/code2019/course2020_codes/flip_all.m

+16 | my_valid_conv2_matlab.m | plot_nen.m | SeparableTest.m | gen_module.m | flip_all.m

```
1   function M = flip_all(K)
```

Command Window

```
>> A = reshape(1:1:9,3,3);
>> B = reshape(1:16,4,4);
>> my_valid_conv2_matlab(B,A)

ans =

   192   372
   237   417

>> conv2(B,A,'valid')

ans =

   192   372
   237   417

fx >>
```

Name

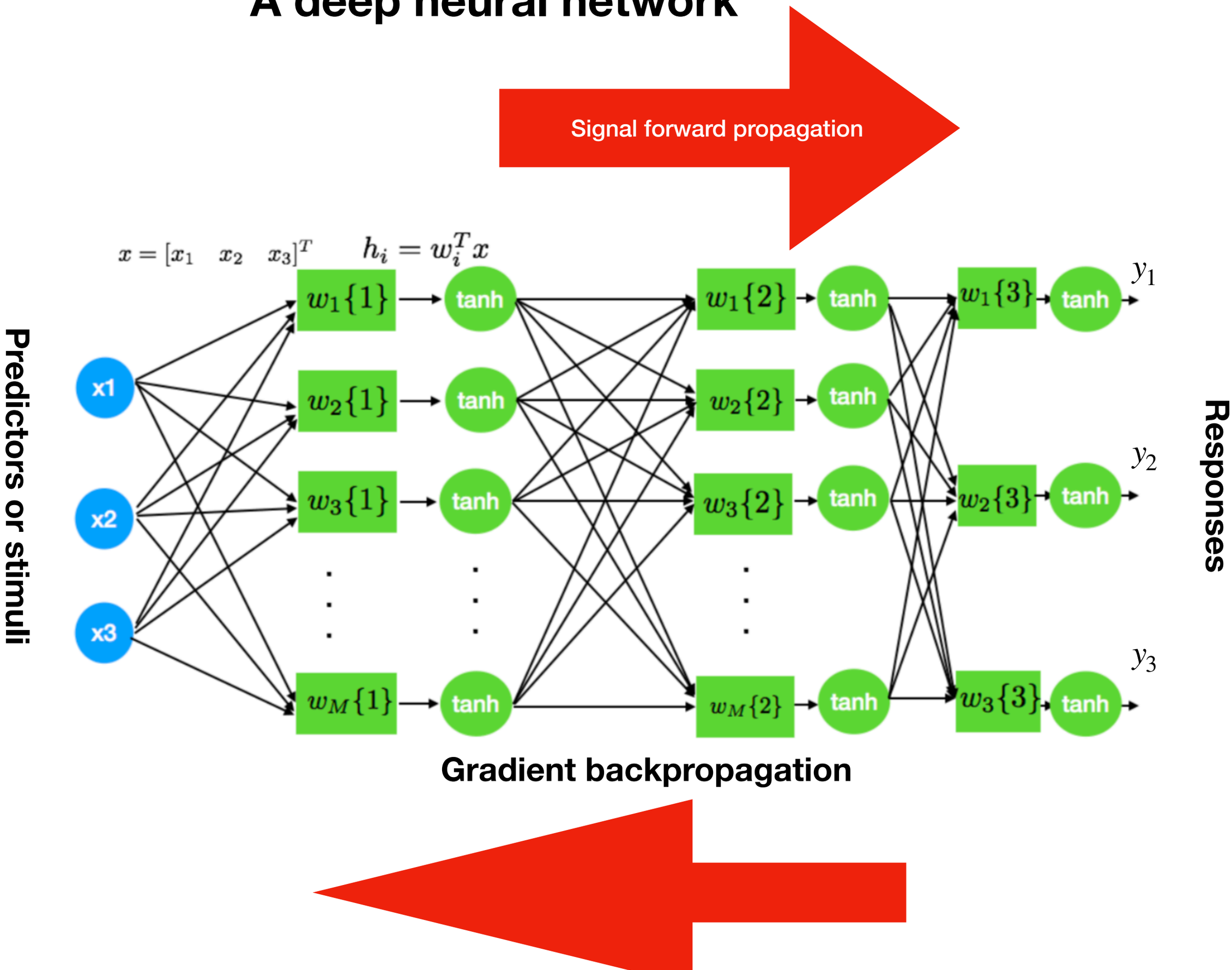
- A
- ans
- B

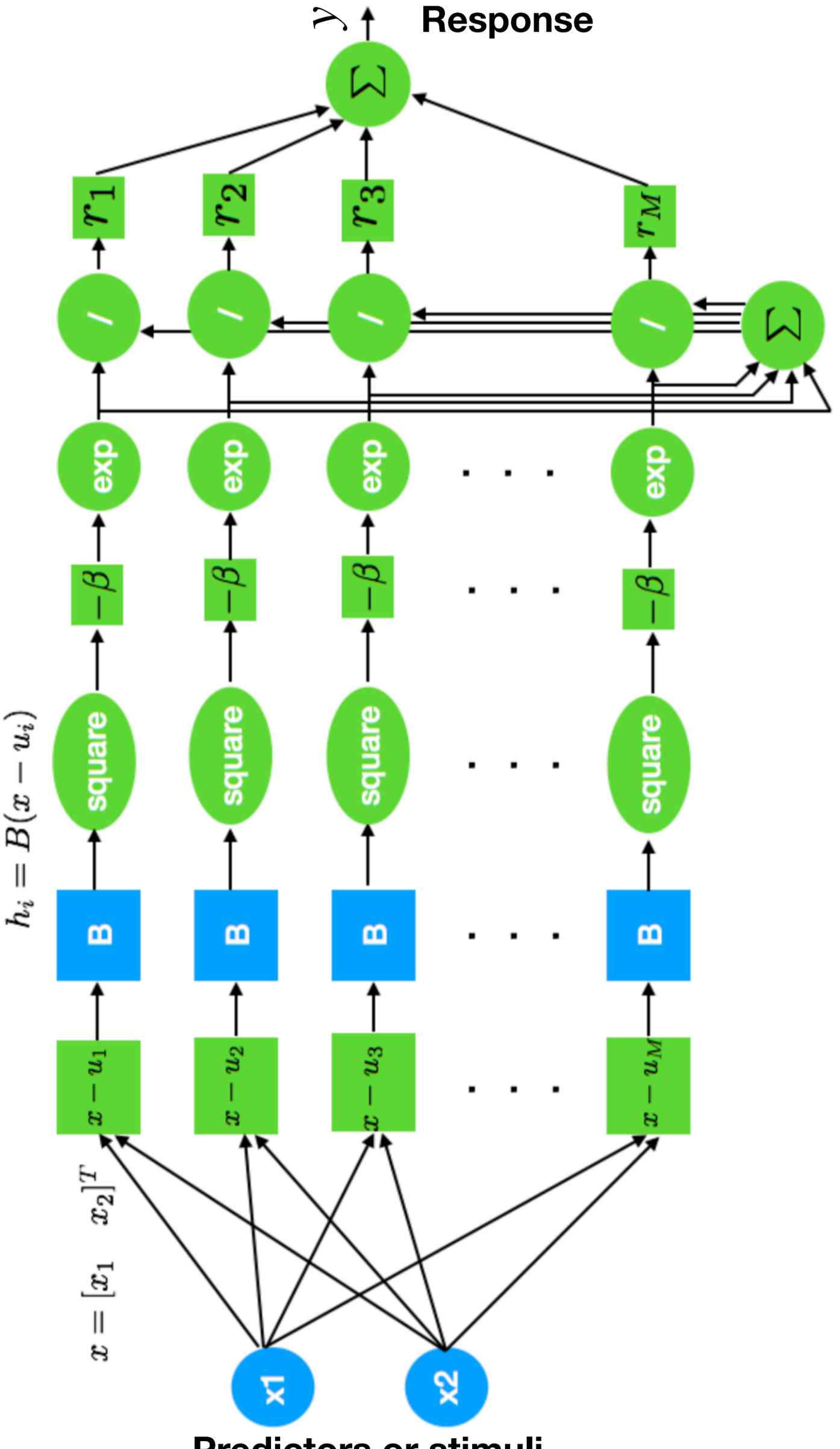
my_valid_conv2_matlab
and
conv2 produce the
same result

Exercise I

Try to write a Matlab function that is equivalent to conv2 for valid convolution.

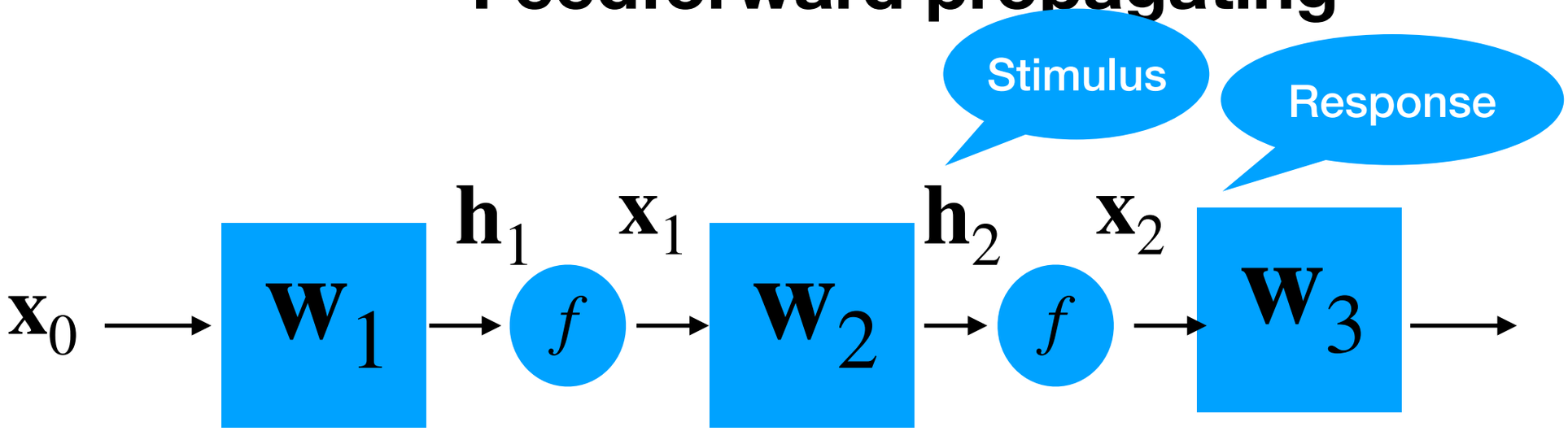
A deep neural network




Response
$\hat{y}$
Σ
r_1
r_2
r_3
r_M
/
Σ
exp
$-\beta$
square
B
$h_i = B(x - u_i)$
$x - u_1$
$x - u_2$
$x - u_3$
$x - u_M$
$x = [x_1 \quad x_2]^T$
x1
x2
Predictors or stimuli

A Deep Neural Network with two hidden layers

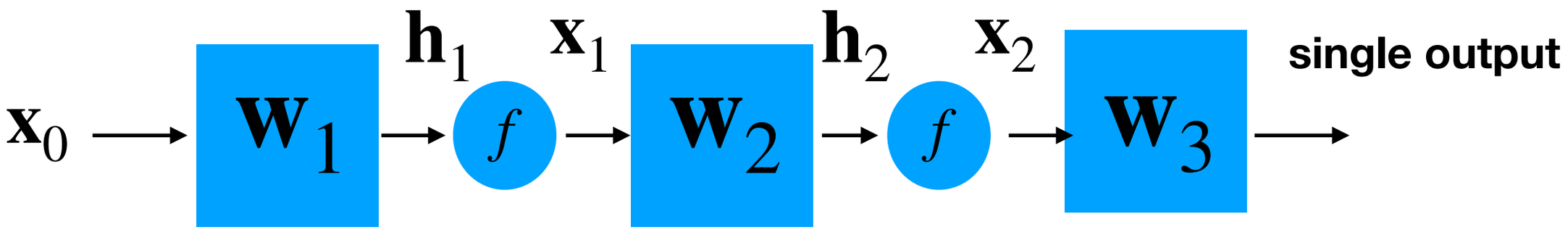
Feedforward propagating



$$\mathbf{h}_1 = \mathbf{w}_1\mathbf{x}_0 \qquad \mathbf{h}_2 = \mathbf{w}_2\mathbf{x}_1 \qquad y = h_3 = \mathbf{w}_3^T\mathbf{x}_2$$

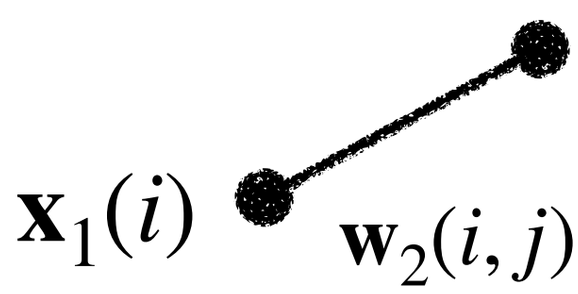
$$\mathbf{x}_1 = f(\mathbf{h}_1) \qquad \mathbf{x}_2 = f(\mathbf{h}_2)$$

Gradients back-propagation

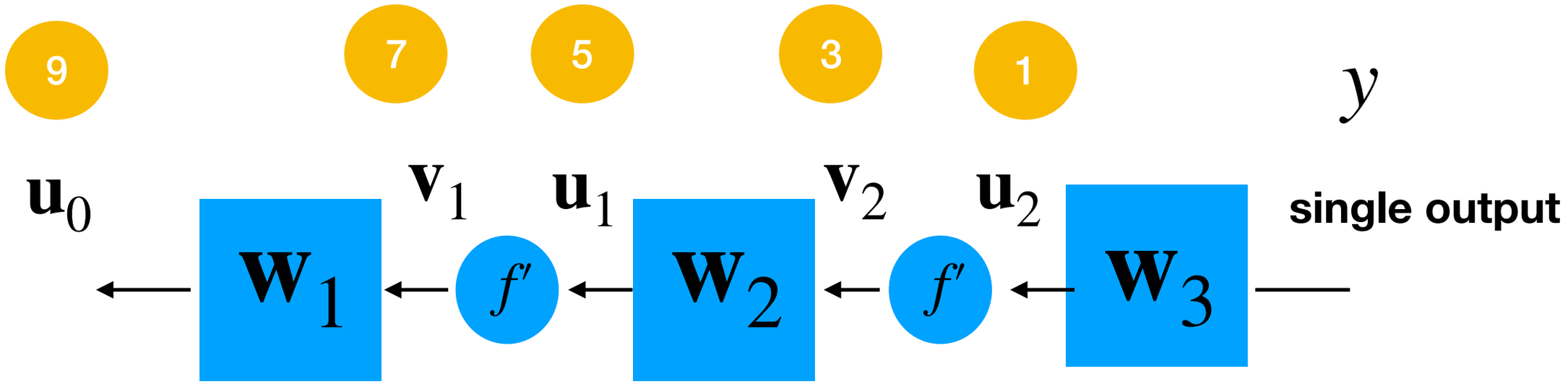


$$\mathbf{h}_1 = \mathbf{w}_1\mathbf{x}_0 \qquad \mathbf{h}_2 = \mathbf{w}_2\mathbf{x}_1 \qquad y = h_3 = \mathbf{w}_3^T\mathbf{x}_2$$

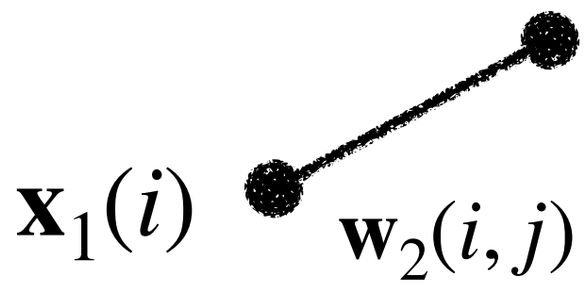
$$\mathbf{x}_1 = f(\mathbf{h}_1) \qquad \mathbf{x}_2 = f(\mathbf{h}_2)$$

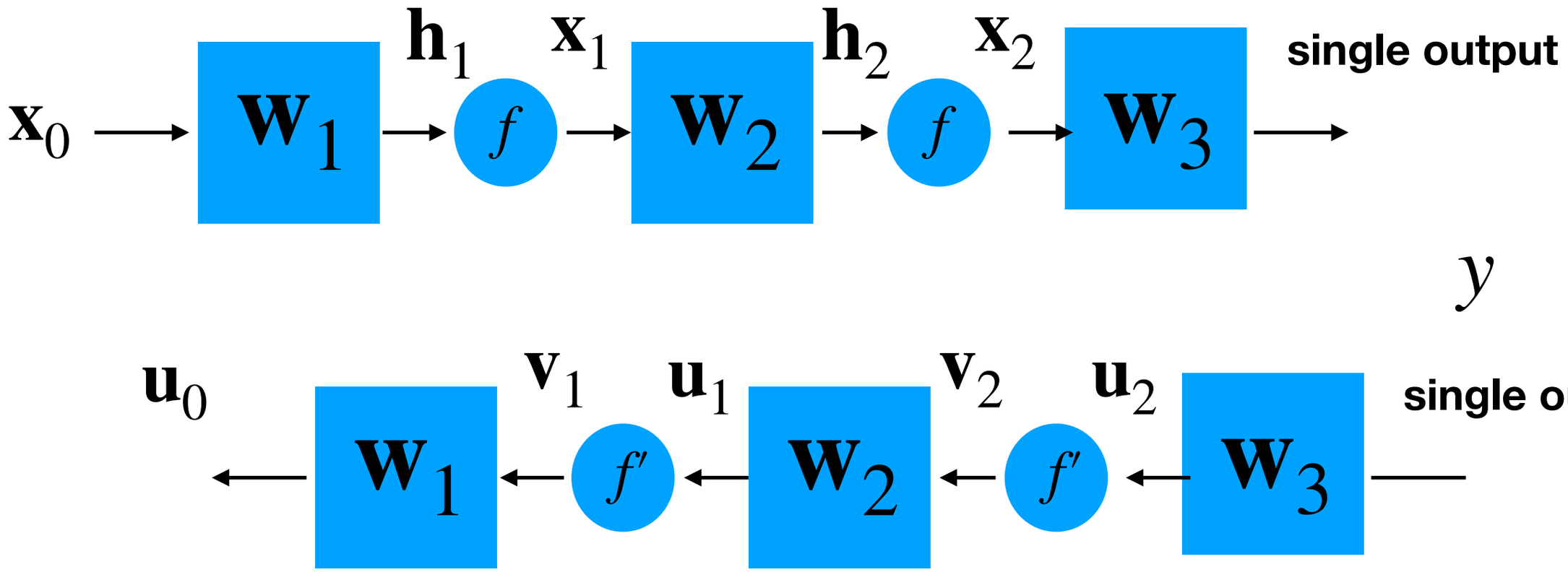


$$\mathbf{h}_2(j) = \sum_k \mathbf{x}_1(k) w_2(k,j)$$

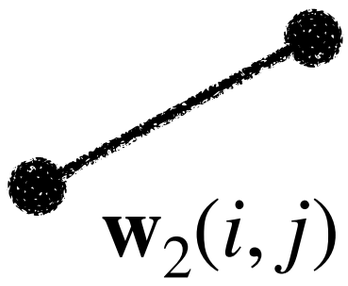


$$\mathbf{h}_2(j) = \sum_k \mathbf{x}(k) w_2(k, j)$$





$$\mathbf{h}_2(j) = \sum_k \mathbf{x}_1(k) w_2(k, j)$$



Exercise II.A

Let $\mathbf{v}_2(j) = \dfrac{dy}{d\mathbf{h}_2(j)}$. $\quad \dfrac{dy}{d\mathbf{w}_2(i,j)} = ?$

Exercise II.B

Let $\mathbf{v}_2(j) = \dfrac{dy}{d\mathbf{h}_2(j)}$. $\qquad \dfrac{dy}{d\mathbf{w}_2(i,j)} = ?$

Let $e = (t - y)^2$, where t denotes the desired target

$$\frac{de}{d\mathbf{w}_2(i,j)} = ?$$

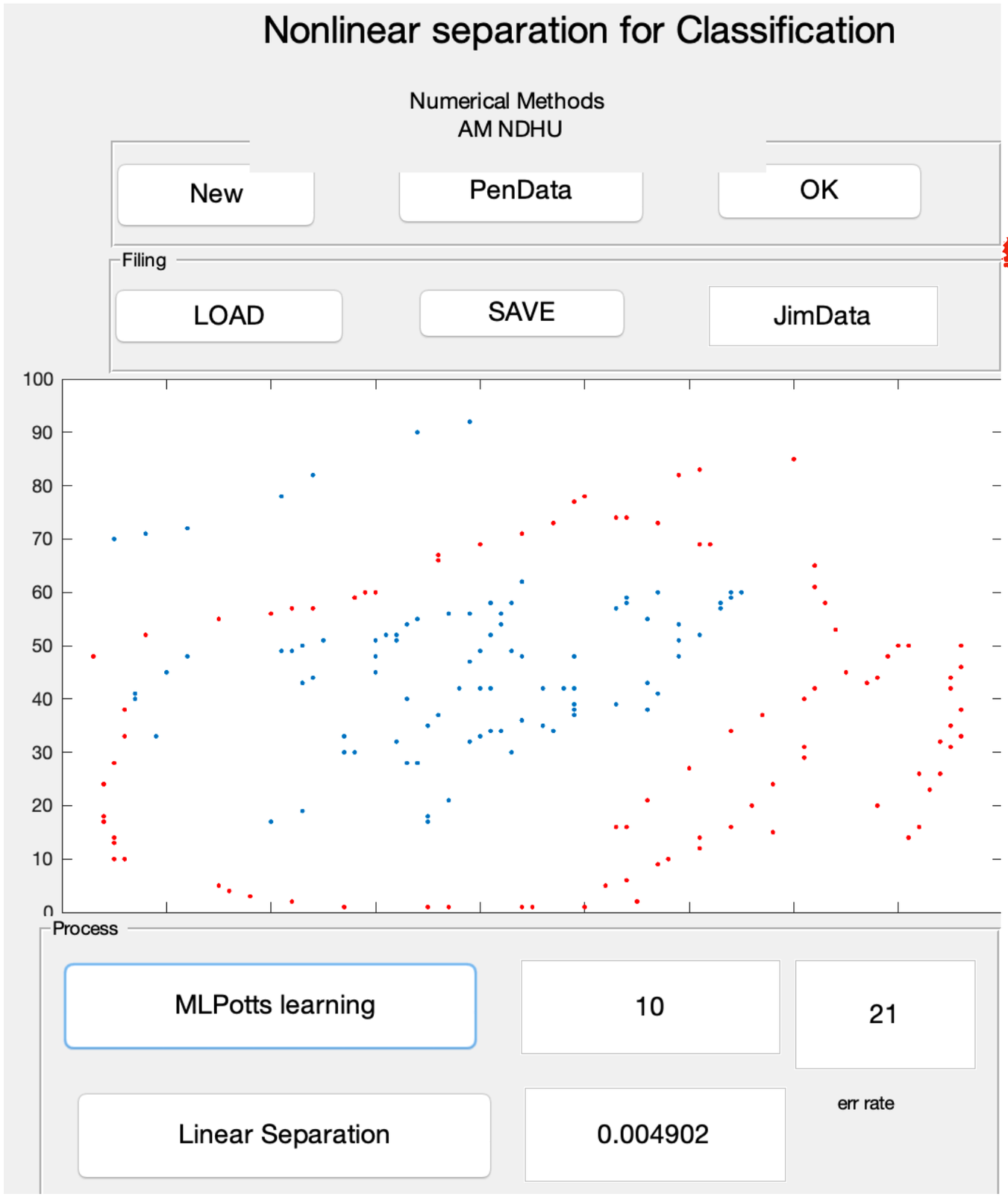
Exercise II.C Show convergence of e for gradient descent learning.

Gradient descent learning

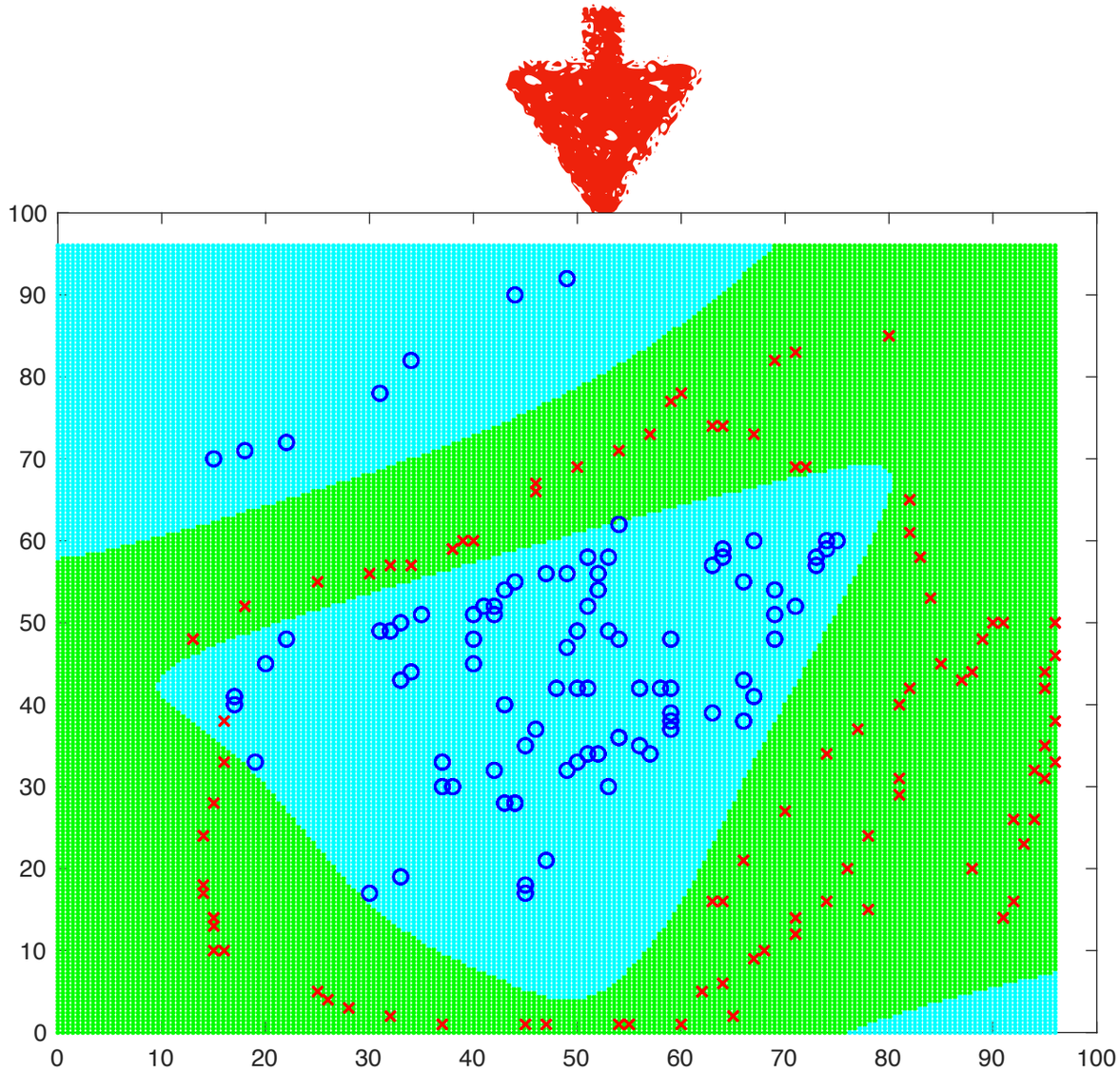
$$\Delta \mathbf{w}_2(i,j) = -\eta \frac{de}{d\mathbf{w}_2(i,j)}$$

where η is a small positive value

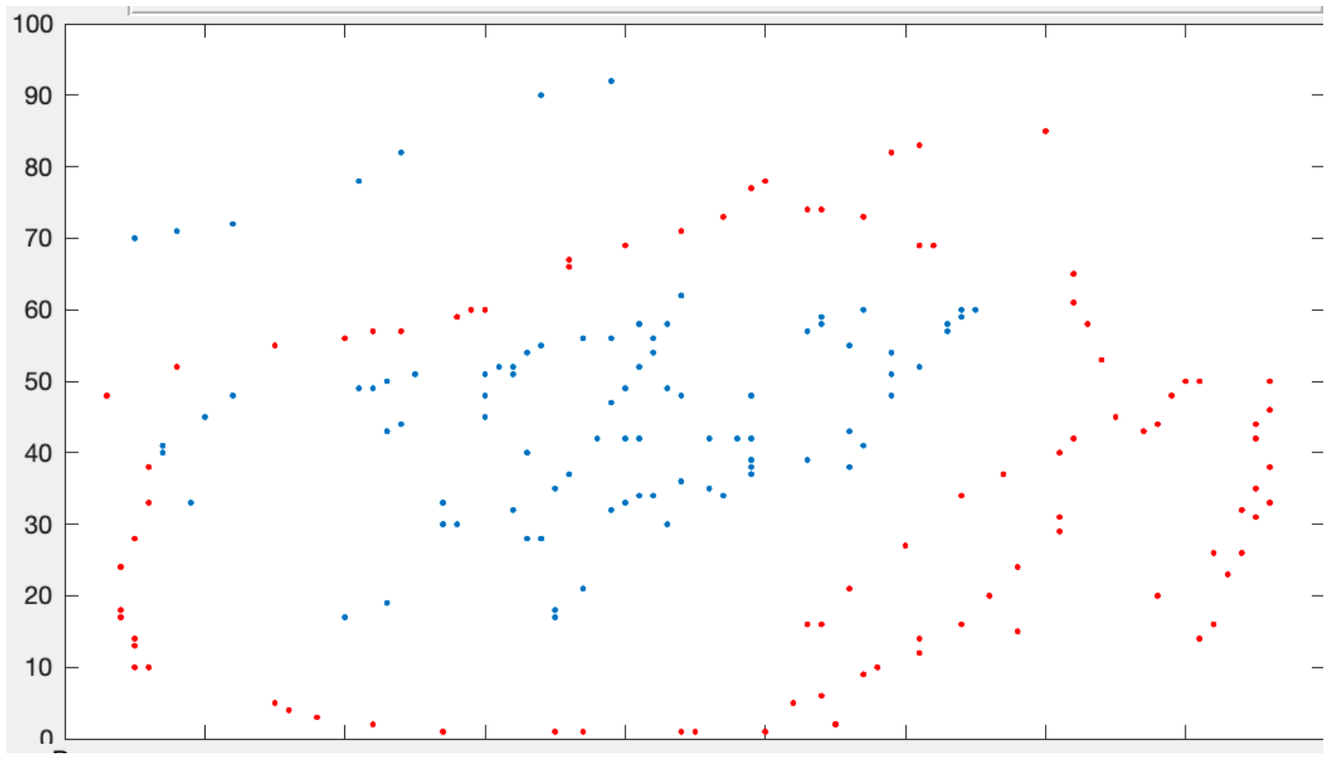
Multilayer Potts Perceptrons With Levenberg–Marquardt Learning



Supervised Learning for classification



Each point has a label. Its color is either red or blue.



Predictors or stimuli: $X = \{x[t] \in R^2\}_{t=1}^{N}$

Responses: $Y = \{y[t] \in \{0,1\}\}_{t=1}^{N}$

The training data are composed of *labeled patterns*.

$$S = \{(x[t], y[t]) \,|\, x[t] \in R^2, y[t] \in \{0,1\}\}_{t=1}^{N}$$

The data space has been well partitioned to two exclusive regions, in each of which training data are with the same label

/ ▸ Users ▸ apple ▸ Desktop ▸ Jiann-Ming Wu ▸ code2019 ▸ code2006 ▸ Apps ▸ FunAppr ▸ Separation_LMMLPotts_2013 ▸

Current Folder

Name ▲

back_up

data

ClusteringTest.fig

ClusteringTest.m

gen

gen

plot

Sep

Sep

Sep

Sep

Sep

Sep

sep

sep

test

Separabl

addpath(

ods\Fun/

otts_201

Separ

Separ

SeparableTest_...

Editor - /Users/apple/Desktop/Jiann-Ming Wu/code2019/code2006/Apps/FunAppr/Separation_LMMLPotts_2013/S...

+15 | demo_chaos_ts_4LNRBF.m | my_valid_conv2.m | plot_nen.m | SeparableTest.m | gen_module.m | +

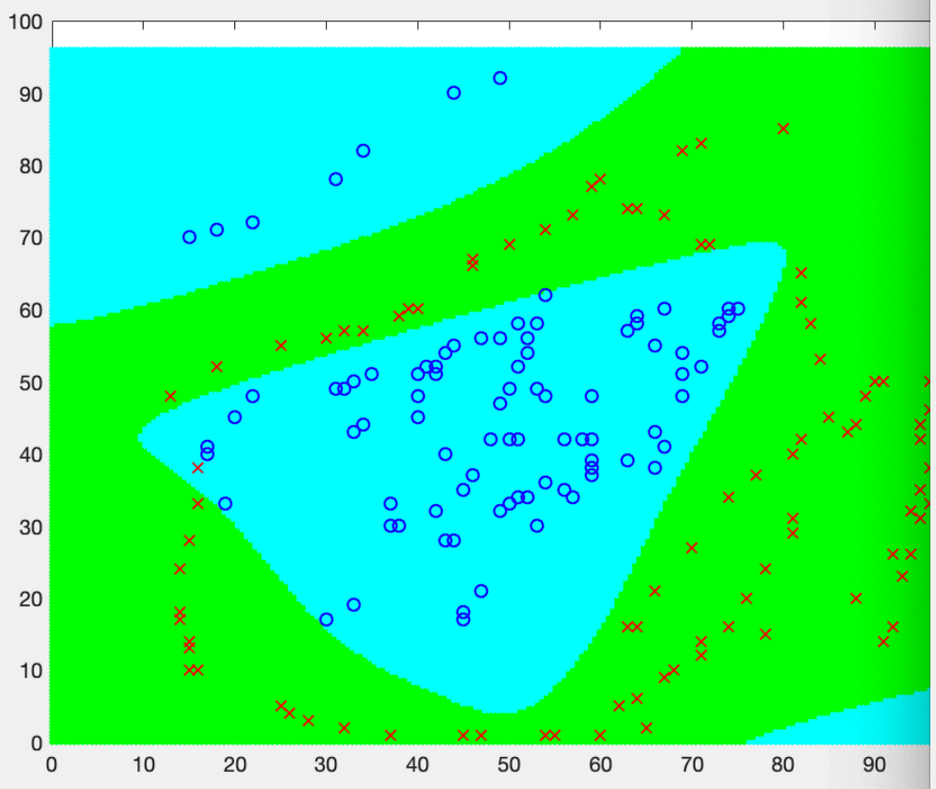
```
425    % handles    empty
426
```

Workspace

Name ▲

Figure 3

File Edit View Insert Tools Desktop Window Help



SeparableTest

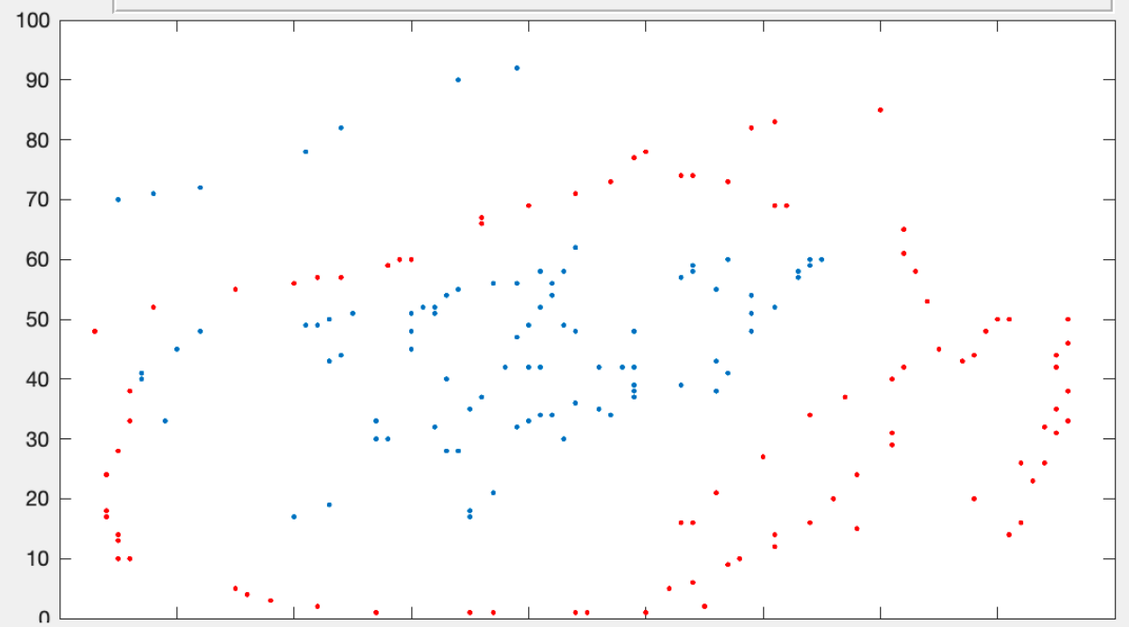
Nonlinear separation for Classification

Numerical Methods
AM NDHU

New | PenData | OK

Filing

LOAD | SAVE | JimData



Process

MLPotts learning | 10 | 21

What is the learning error or training error?

Handwritten digit Classification

$F(x \mid \theta_{random})$

Training data set: labeled digits

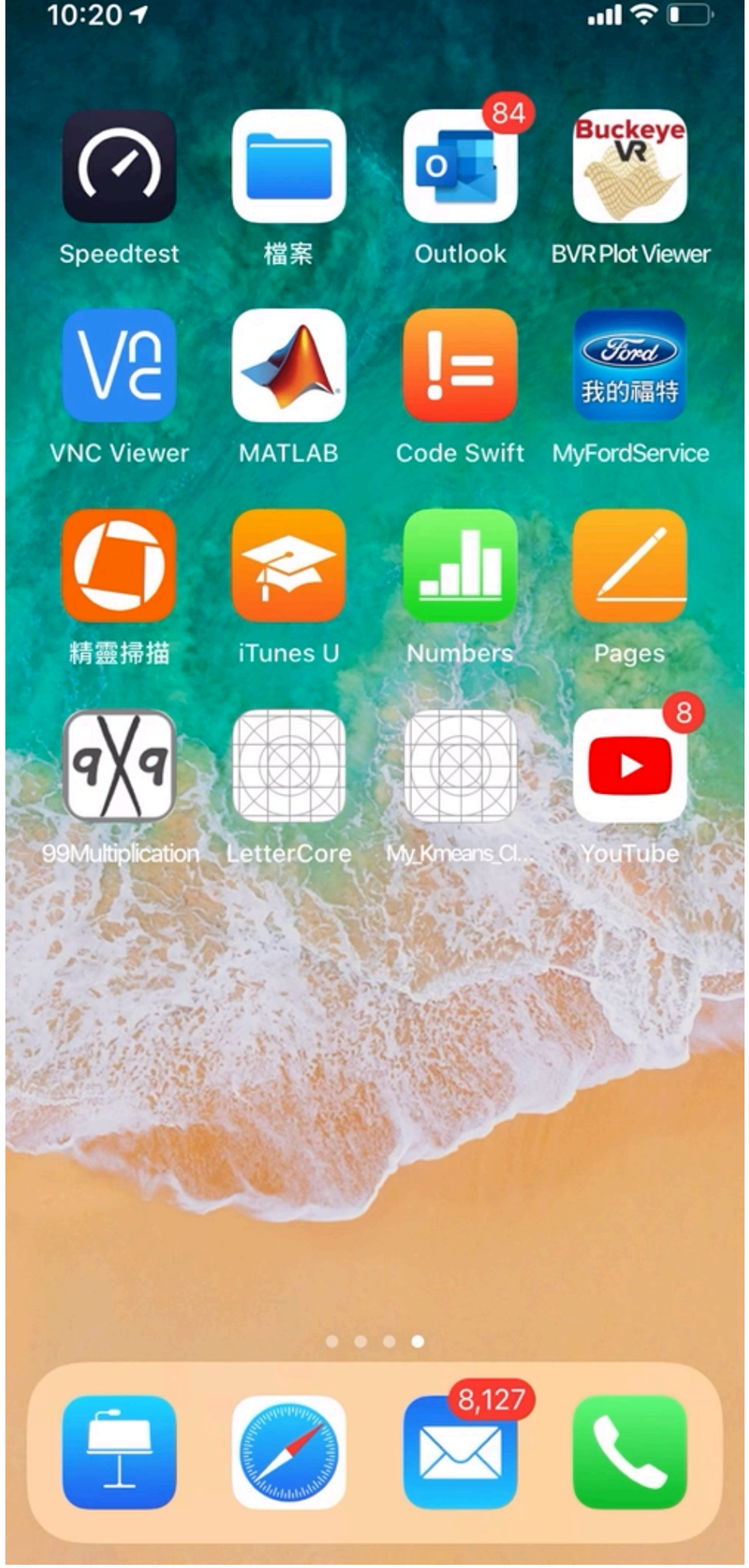
Train a classifier

$F(x \mid \theta_{opt})$

Testing data set

Test $F(x \mid \theta_{opt})$

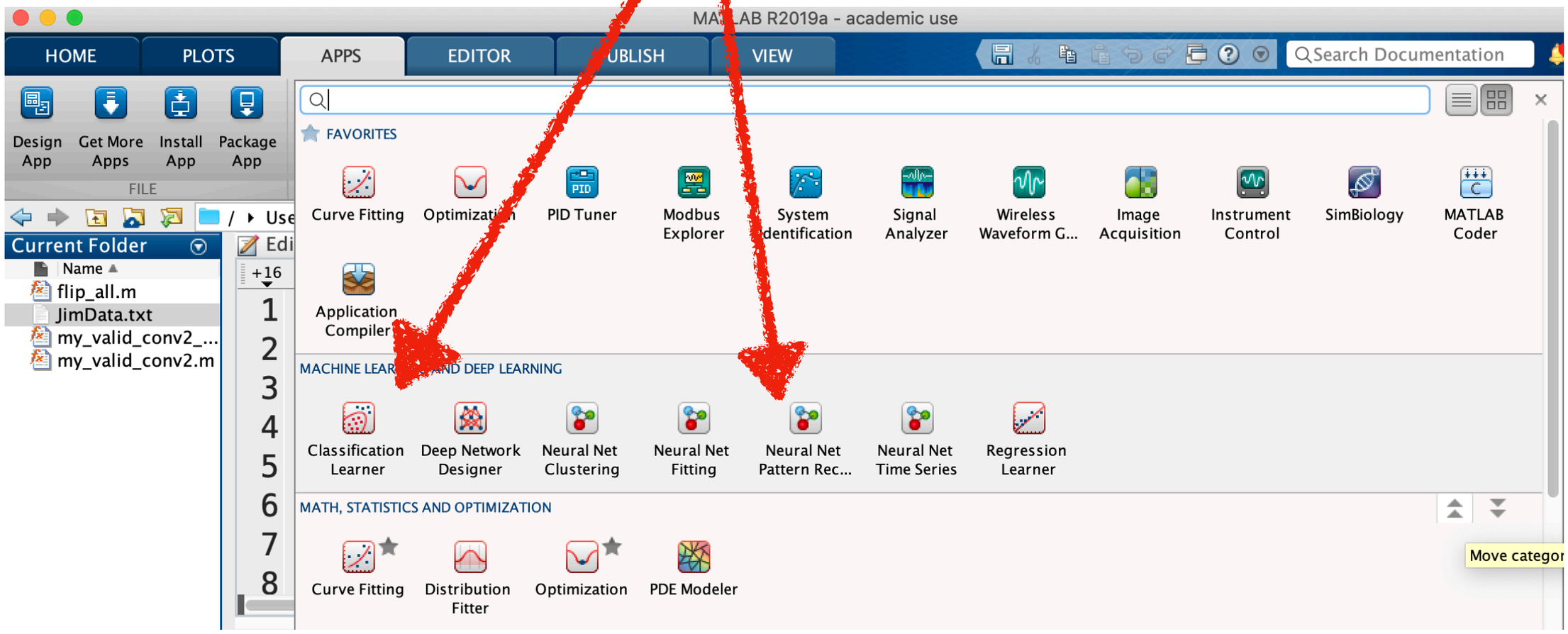
Real application of $F(x \mid \theta_{opt})$



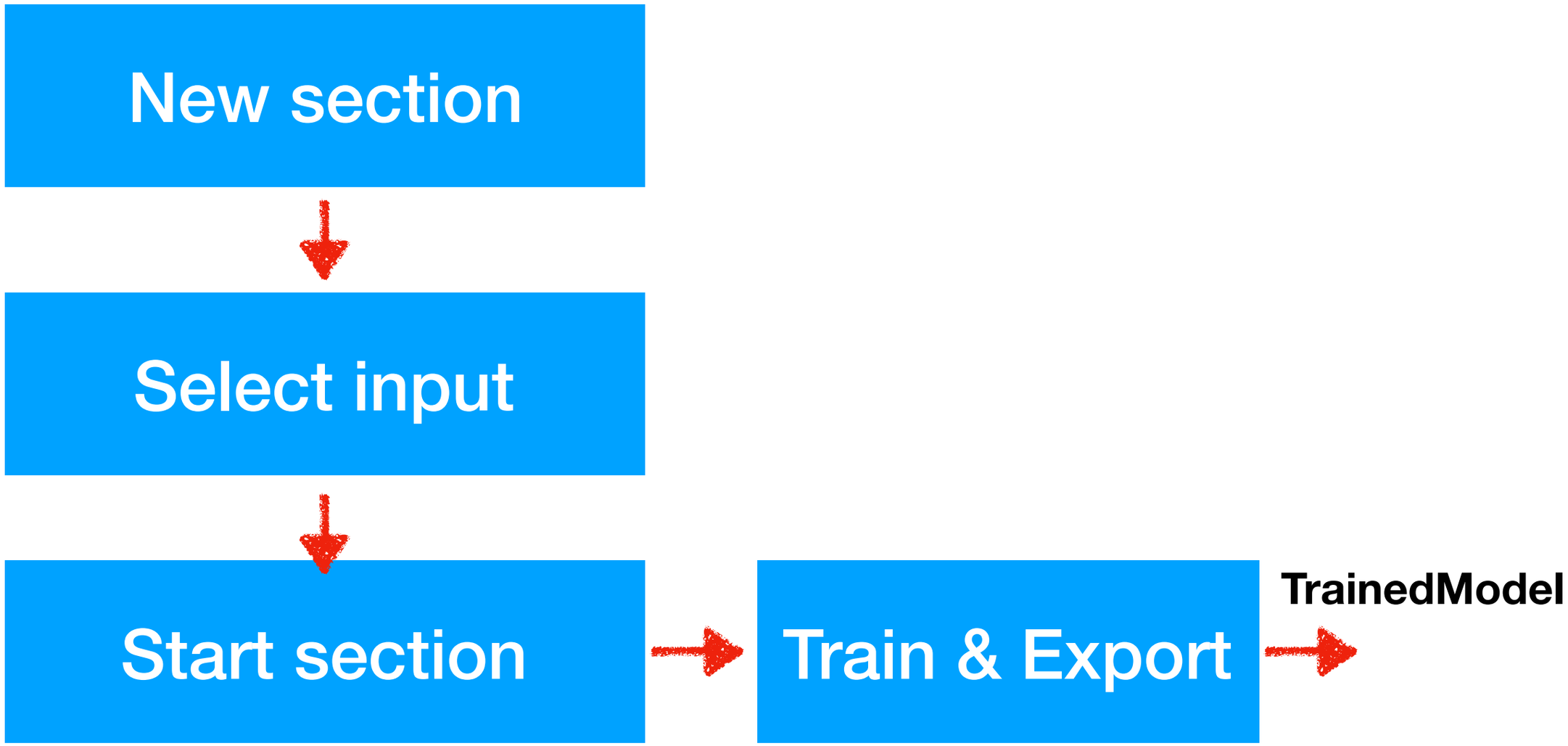
iPhone screen recording

Classifier Learner App

load JimData.txt



Classier Learner



Classification Learner - Scatter Plot

CLASSIFICATION LEARNER | VIEW

New Session | Feature Selection | PCA | Linear Discrimi... | Quadratic Discrimi... | All Discrimi... | Logistic Regression | Advanced | Use Parallel | Train | Scatter Plot | Confusion Matrix | ROC Curve | Parallel Coordinates Plot | Export Plot to Figure | Generate Function | Export Model

FILE | FEATURES | MODEL TYPE | TRAINING | PLOTS | EXPORT

Data Browser

History

1 Tree	Accuracy: **100.0%**		
Last change: Disabled PCA	2/2 features		
2 Logistic Regression	Accuracy: 65.2%		
Last change: Logistic Regression	2/2 features		
3.1 Naive Bayes	Accuracy: 84.5%		
Last change: Gaussian Naive Bayes	2/2 features		
3.2 Naive Bayes	Accuracy: 86.6%		
Last change: Kernel Naive Bayes	2/2 features		
4 Quadratic Discriminant	Accuracy: 83.8%		
Last change: Quadratic Discriminant	2/2 features		

Current Model

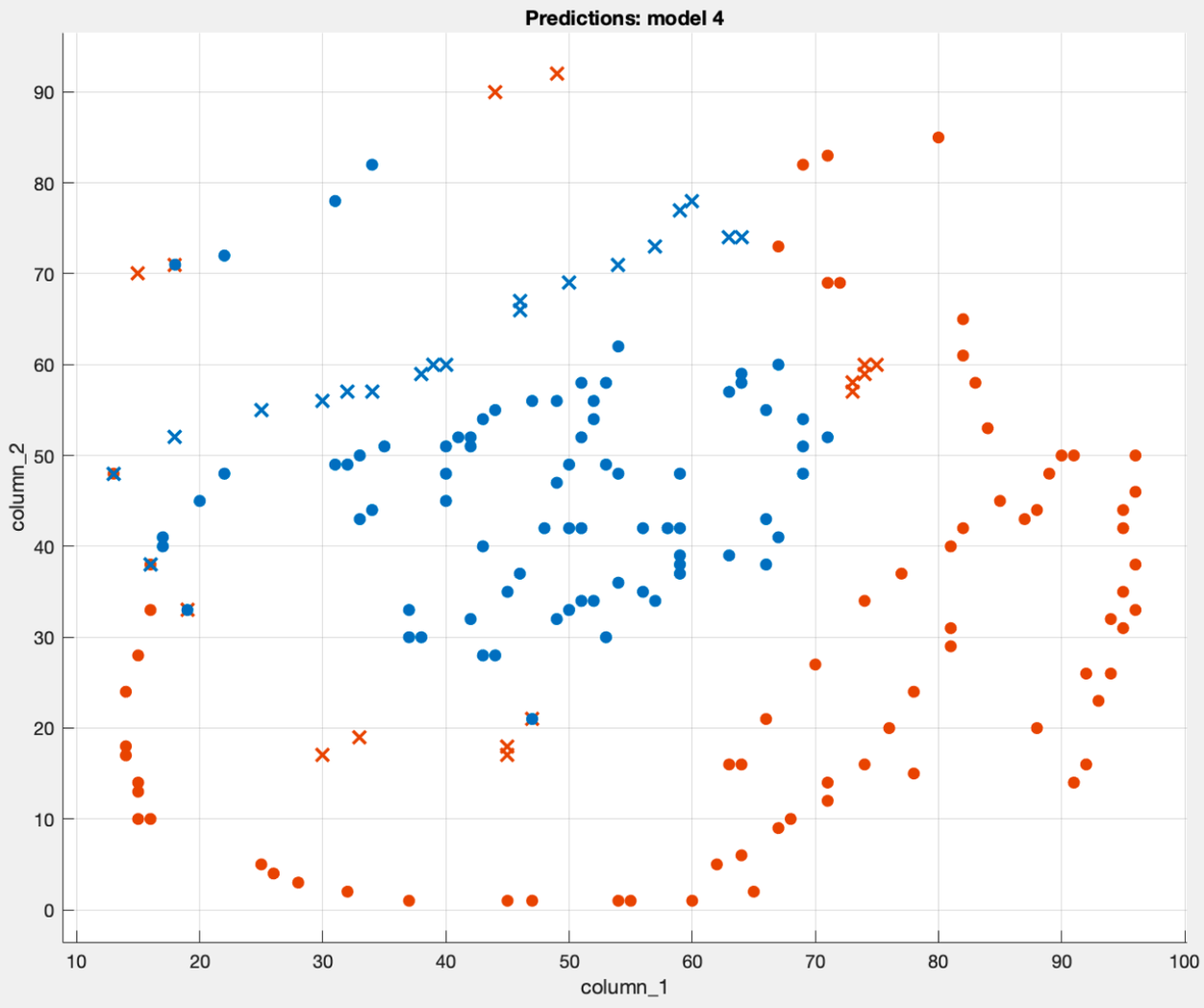
Model 4: Trained

Results

Accuracy	83.8%
Prediction speed	~16000 obs/sec
Training time	4.941 sec

Model Type
Preset: Quadratic Discriminant

Scatter Plot



Plot
- Data
- Model predictions
  - Correct
  - Incorrect

Predictors
X: column_1
Y: column_2

Classes | Move to Front

Show	Order
☑	0
☑	1

How to investigate features

Data set: **JimData** Observations: **1428** Size: **35 kB** Predictors: **2** Response: **column_3** Response Classes: **2**

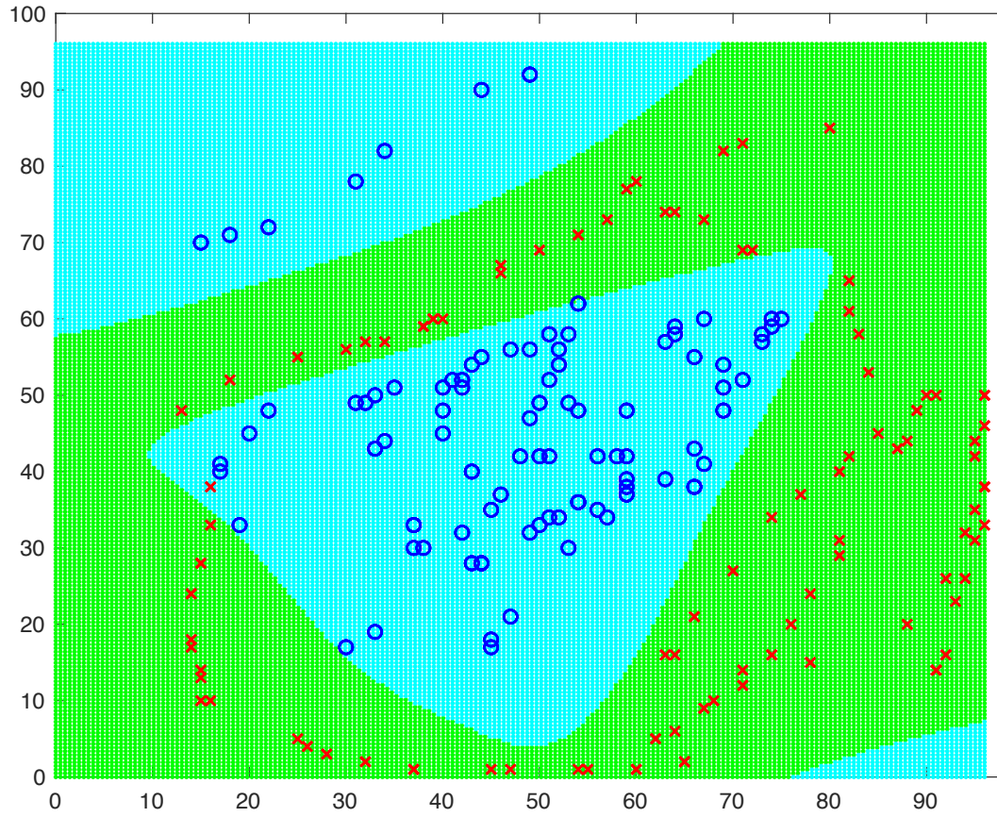
Validation: **5-fold Cross-Validation**

Exercise III
Select a learner.
Train a classifier subject to JimData and export the TrainedModel.

Exercise IV
Write a Matlab function to evaluate the accuracy of the TrainedModel.

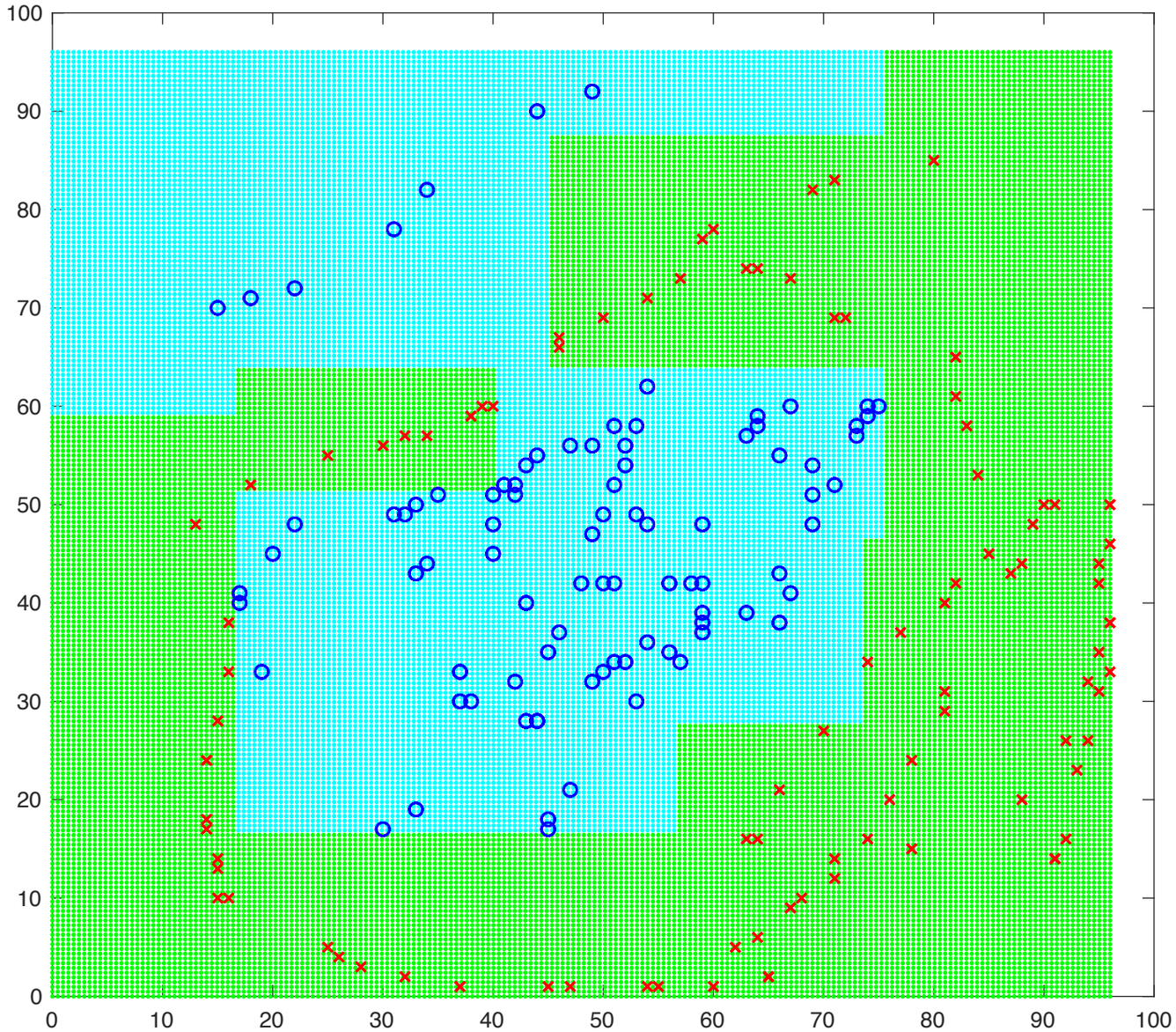
```
function  accuracy = my_fit(JimData,trainedModel)


yfit = trainedModel.predictFcn(X);
```

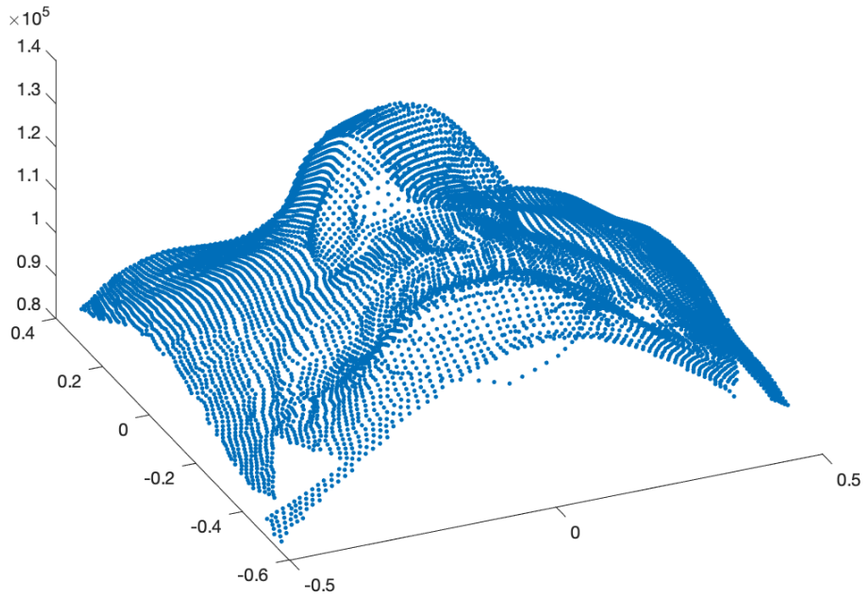
Exercise V
Draw the partitioned exclusive regions of the TRainedModel



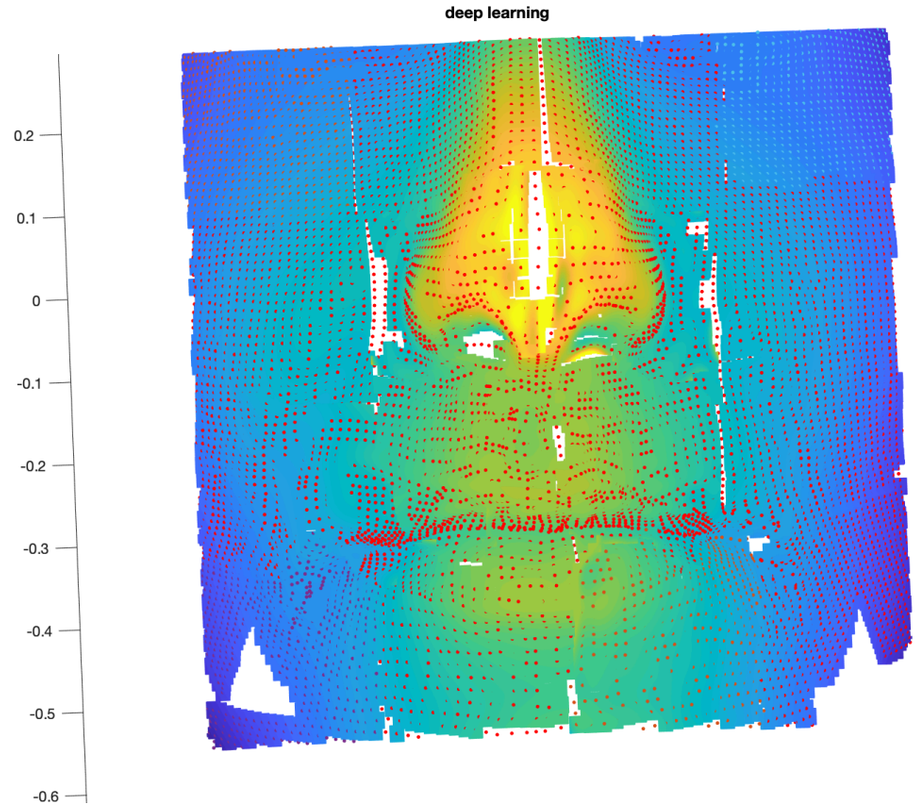

```
n=200;
x1=linspace(0,Net.max_x,n);   % revise codes
x2=x1;
xx=[repmat(x1',n,1) reshape(repmat(x2,n,1),n*n,1)];
CC=                    % write codes
figure
ind=find(CC>0.5);
plot(xx(ind,1),xx(ind,2),'g.');hold on
ind=find(CC<0.5);
plot(xx(ind,1),xx(ind,2),’c.'); pc = JimData(:,3);x_train=JimData(:,1:2)
plot(x_train(find(pc==0),1),x_train(find(pc==0),2),'oB');
plot(x_train(find(pc==1),1),x_train(find(pc==1),2),'xR');
```

```
n=200;
x1=linspace(0,max(max(X)),n);
x2=x1;
% revise codes
xx=[repmat(x1',n,1) reshape(repmat(x2,n,1),n*n,1)];
CC= trainedModel.predictFcn(xx);% write codes
figure
ind=find(CC>0.5);
plot(xx(ind,1),xx(ind,2),'g.');hold on
ind=find(CC<0.5);
plot(xx(ind,1),xx(ind,2),'c.');
pc = JimData(:,3);
x_train=JimData(:,1:2);
plot(x_train(find(pc==0),1),x_train(find(pc==0),2),'oB');
plot(x_train(find(pc==1),1),x_train(find(pc==1),2),'xR');
```


100
90
80
70
60
50
40
30
20
10
0
0
10
20
30
40
50
60
70
80
90
100



圖一、3D人臉的原始資料點



圖九a、完整的臉頰、鼻子、嘴、下巴3D人臉平面幾何重構與近似