Matlab Pattern Classification

Demo after Chapter 2 (Bayes)

Menu ->Help -> Product Help

Contents tab -> Statistics Toolbox -> Classification Check Discriminant Analysis Go to classify Study its syntax Study the description of *type, prior, class, err, coeff* See also mahal Search for knnclassify Study the description of *k, distance, rule*

Demos tab -> Toolboxes -> Statistics -> Multivariate Analysis -> Classification Copy and paste code Use knnclassify with *k*=1, 3, 5 instead of classify

Demo after Chapter 3 (Linear)

Solve Example 3.3 (Slide 22) using Matlab.

```
X = [.4 .5 1; .6 .5 1; .1 .4 1; .2 .7 1; .3 .3 1; ...
        .4 .6 1; .6 .2 1; .7 .4 1; .8 .6 1; .7 .5 1];
y = [1 1 1 1 1 -1 -1 -1 -1 -1]';
hold on;
gscatter(X(:,1), X(:,2), y,'rg','ox',10);
A = X' * X;
B = X' * y;
w = inv(A)*B;
f = sprintf('0 = %g*x+%g*y+%g',w(1), w(2), w(3));
ezplot(f, [0 1 0 1]);
```

Menu -> Help -> Product Help Search for svm Study svmclassify and svmtrain

Demo after Chapter 4 (Non-linear)

Contents tab -> Neural Network Toolbox -> Getting Started Check **Recognizing Patterns**

Using Command-Line Functions:

```
load cancer_dataset
net = newpr(cancerInputs, cancerTargets, 20);
net=train(net, cancerInputs, cancerTargets);
```

Using the Neural Network Toolbox[™] Pattern Recognition Tool GUI: nprtool

Demos tab -> Toolboxes -> Statistics -> Multivariate Analysis -> Classification Copy and paste code Study Decision Trees treefit and treeval

Demo after Chapter 5 (Feature Generation)

Contents tab -> Image processing Toolbox -> Getting Started Study Examples 1 and 2

Demos tab -> Toolboxes -> Image Processing -> Image Segmentation Study Detecting a Cell

Demo after Chapter 10 (System Evaluation)

Demos tab -> Statistics -> Multivariate Analysis -> Selecting Features for Classifying High-dimensional Data

Study "Loading the Data"

```
Study "Dividing Data into a Training Set and a Test Set"
```

Search for cvpartition

Study kfold, holdout, leavout, resubstitution See also crossval

Do the demo of classify using crossval

```
load fisheriris
fun = @(xT,yT,xt,yt)(sum(~strcmp(yt,classify(xt,xT,yT))));
rate = sum(crossval(fun, meas(:,1:2), species)) / 150
Or
c = cvpartition(species,'k',10);
rate = sum(crossval(fun, meas(:,1:2), species, 'partition', c)) /
sum(c.TestSize)
```