Carbon

Jiayu Wang

Tutorial 0: data preprocessing

1. Prepare the data

```
#All data should be prepared in a comma delimitated file, with
first row as variable names and each succeeding row as an
observation. The first column is used as an unique for each
observation. The last column is the class label for supervised
learning. No missing value is tolerated.
The raw sample data look like this:
more sample.txt
Sample id,Clump Thickness,Uniformity of Cell Size,Uniformity of
Cell Shape,Marginal Adhesion,Single Epithelial Cell Size,Bare
Nuclei,Bland Chromatin,Normal Nucleoli,Mitoses,Class
1000025,5,1,1,1,2,1,3,1,1,2
1002945,5,4,4,5,7,10,3,2,1,2
1015425,3,1,1,1,2,2,3,1,1,2
1016277,6,8,8,1,3,4,3,7,1,2
1017023,4,1,1,3,2,1,3,1,1,2
1017122,8,10,10,8,7,10,9,7,1,4
1018099,1,1,1,1,2,10,3,1,1,2
#To start loading data, use the carbon folder as working
directory, put the data file in it, too. Open terminal and type
in the following command, then press enter.
python3
```

2. Load data

```
#load data module
import data
#create a dataset object and read data from file sample.txt
sample = data.DataSet()
#To read as nominal data, you have to add argument 'nominal',
default is 'numeric'
#If there's class labels, argument suprv should be True as
default, if the data is unsupervised, suprv should be set to
False
#Read data from 'sample.txt'
sample.read('sample.txt', type='numeric',suprv=True)
#data are stored in attribute x, numerical data are in an array
while nominal data are in a list
sample.x
array([[  5.,   1.,   1., ...,   3.,   1.,   1.],
       [  5.,   4.,   4., ...,   3.,   2.,   1.],
       [  3.,   1.,   1., ...,   3.,   1.,   1.],
       ...,
       [  5.,  10.,  10., ...,   8.,  10.,   2.],
       [  4.,   8.,   6., ...,  10.,   6.,   1.],
       [  4.,   8.,   8., ...,  10.,   4.,   1.]])
#view class labels, they are stored as list of string
#if suprv is False, y will be all '0'
```

```
sample.y[:10]
['2', '2', '2', '2', '2', '4', '2', '2', '2', '2']
#view dataset dimension, (numberOfRow,numberOfCol)
sample.dim()
(699, 9)
#view feature names
sample.label
['Clump Thickness', 'Uniformity of Cell Size', 'Uniformity of
Cell Shape', 'Marginal Adhesion', 'Single Epithelial Cell Size',
'Bare Nuclei', 'Bland Chromatin', 'Normal Nucleoli', 'Mitoses']
#view subject ids
sample.key[:5]
['1000025', '1002945', '1015425', '1016277', '1017023']
```

3. View data

```
#TODO
```

4. Transformation

```
#Scale the data to range (0,1)
sample.scale(a=0,b=1)
```

5. Create train and test datasets

```
#shuffle the data
sample.shuffle()
#create train dataset and test dataset using 1:10 hold out
train,test = data.holdOut(sample,0.1)
```

```
#fast start
import data
sample = data.DataSet()
sample.read('sample.txt')
train,test = data.holdOut(sample,0.1)
from imp import reload

reload(CART)
clf = CART.build()
clf.train(train,tolS=1,tolN=4,model=False)
clf.classify([5, 4, 4, 5, 7, 10, 3, 2, 1])
clf.view('Uniformity of Cell Size','Uniformity of Cell
Shape',train)
```