Building decision trees

Data mining lab 1

Python code for decision trees

- Read code for the building of decision trees (see ***buildtree.py***)
- Pay attention how function ***classify*** handles missing attribute values. You are to implement similar recursive calls for missing values in training data.

Input data (training sets)

We extract the model from the following inputs (training sets):

1. The data about subscriptions ***signupdata.txt*** (delimiter '\t')

2. Weather data ***weatherdata.txt*** (delimiter ',')

3. Contact lenses data ***lensesdata.txt*** (delimiter ',')

4. Training set obtained from ***zillow.com*** website about real estate data ***propertydata.txt*** (delimiter '\t')

Obtaining data from the WEB

In order to see how the real estate data was obtained using Zillow API

(http://www.zillow.com/howto/api/APIOverview.htm):

1. Download file ***zillowNew.py***

2. Download addresses for which the data was obtained: ***addresslist.txt***

3. Run the following Python commands:

```
import zillowNew
housedata=zillowNew.getpricelist(  )
import files
files.printdatatofile( housedata,'propertydata.txt', '\t')
```

Building decision trees

1. Install graphics library PIL Python Imaging Library 1.1.6 for Python 2.6 from http://www.pythonware.com/products/pil/ [Already installed on Lab machines]

2. Download the following files: buildtree.py, displaytree.py, files.py to your home directory.

Then in the interpreter type the following sample code:

```
#imports
import files, buildtree, displaytree
```

Example 1. Sign up data

- In order to build the model underlying the sign up data run the following commands

```
#read data into an array
my_data=files.readdatafile ('signupdata.txt','\t')
tree=buildtree.buildtree(my_data)
displaytree.printtree(tree)
displaytree.drawtree(tree,jpeg='signuptree.jpg')
```

Example 2. Weather data

- In order to build model underlying the weather data run

```
my_data=files.readdatafile ('weatherdata.txt',',')
tree=buildtree.buildtree(my_data)
displaytree.printtree(tree)
displaytree.drawtree(tree,jpeg='weathertree.jpg')
```

Lab assignment 1

Build decision tree for lenses data and for real estate data.

Tip: for real estate data the class attribute is numeric, so use ***variance*** instead of entropy

```
tree=buildtree.buildtree(my_data, buildtree.variance)
```

Lab assignment 2. Prediction

The module ***classify*** in ***buildtree.py*** classifies new instances (possibly, with missing attributes):

Predict the price of the following real estate unit:

[single family house built in 1920, 2 bathrooms, 3 bedrooms, 2 rooms].

What is the most likely price for this unit? How good is the prediction?

```
buildtree.classify([None, 'SingleFamily',1920, 2, 3, 2],tree)
```