This file describes how to interpret the contents of files train.txt, dataDesc.txt and deDomain.txt.

The file train.txt is a 6x140 matrix, where the rows are attributes and columns are customers. Rows 1 – 6 correspond to attributes RISK. AGE, CRED_HIS, INCOME, RACE and HEALTH in that order. The class label attribute is RISK. The values for each attribute have been encoded as integers.

The file test.txt is a 6x70 matrix. The interpretation for its rows and columns is identical to that for train.txt.

The file dataDisc.txt contains a list that can be used to interpret train.txt and test.txt. It was saved by json.dump. You can read it using the following code:

```
with open('../data/dataDesc.txt') as f:
    m = json.load(f)
```

Then m refers to the following list:

```
[['RISK',(1,2)], ['AGE',(1,2,3)], ['CRED_HIS',(1,2)], ['INCOME',(1,2)], ['RACE',(1,2,3)], ['HEALTH',(1,2)]]
```

This list contains as its members six sub-lists corresponding to six attributes in train.txt/test.txt. Note that these sub-lists are listed in the same order as the corresponding rows are listed in the matrixes for files train.txt and test.txt. The tuple following each attribute name is the domain for that attribute.

The file deDomain.txt contains a nested dictionary that can be used to decode the values for the attributes. It is used only by my program in disp.py to make the plotting readable, i.e., to use English words rather than their integer encodings in the plotting. Since it's adequate for your program to use the encoding only, your program code will never need to access this file.