ML Homework 3 _ part 0

December 11, 2017

1 ML Homework 3 - Part 0

Ivan Puhachov

1.1 Transformation of data

1.1.1 How to use:

put into *filename* pathh to csv file

```
In [9]: import pandas as pd
        import numpy as np
```

```
In [10]: filename = 'Letters.csv'

         DATA = pd.read_csv(filename)

         feature_range = np.array([DATA[i].unique().tolist() for i in DATA]) # list of all featu

         inputvector_length = sum([len(feature_range[i]) for i in range(len(feature_range))])

         def encodeValue(value, column):
             """
             create a characteristics vector, encoding value from column in data
             """
             encodedValue = np.zeros(len(feature_range[column]))
             try:
                 index = feature_range[column].index(value)
                 encodedValue[index]=1
             except:
                 pass
             return encodedValue

         def deencodeValue(vector, column):
             """
             deencode characteristics vector given from column
             """
             index = np.argmax(vector)
```

```
            return feature_range[column][index]

        def transformInput(row):
            """
            return a transformed input query even if it has question marks (substitute with 0-i
            """
            transformedRow = np.array([])
            for i in range(len(row)):
                value = row[i]
                encodedValue = encodeValue(value,i)
                transformedRow = np.concatenate((transformedRow,encodedValue))
                #print(encodedValue)
            return transformedRow
```

Example

```
In [3]: DATA.iloc[0,0]

Out[3]: 'T'

In [4]: vector = encodeValue(DATA.iloc[0,0],0)
        print(vector)

[ 1.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.
  0.  0.  0.  0.  0.  0.  0.  0.]


In [5]: print(deencodeValue(vector,0))

T


In [6]: tst = DATA.iloc[100,:].values
        print(transformInput(tst))

[ 0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.
  0.  0.  0.  1.  0.  0.  0.  0.  1.  0.  0.  0.  0.  0.  0.  0.  0.  0.
  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.
  0.  0.  1.  0.  1.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.
  0.  0.  1.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.
  1.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.
  0.  0.  1.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.
  0.  0.  0.  0.  1.  0.  0.  0.  0.  0.  0.  0.  1.  0.  0.  0.  0.  0.
  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  1.  0.  0.  0.
  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  1.  0.  0.  0.  0.  0.
  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  1.  0.  0.
  0.  0.  0.  0.  1.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.
  0.  0.  0.  0.  0.  0.  1.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.
  0.  0.  0.  0.  0.  0.  0.  1.  0.  0.  0.  0.  0.  0.  0.  0.  1.  0.
```

```
  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  1.  0.  0.  0.
  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.]
```

```
In [7]: [len(i) for i in feature_range]
```

```
Out[7]: [26, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16]
```

```
In [8]: len(transformInput(tst))
```

```
Out[8]: 282
```