THINKFUL

Similarity Measures

DATA SCIENCE

Agenda

How to measure dissimilarity with non-numeric data

- Categorical measures:
  - Simple Matching (Hamming) Distance
  - Jaccard
  - Dice
- Mixed data:
  - Gower

Warm Up

Consider the following dataset, representing five different individuals in the context of credit risk:

ID	Age	Age of Oldest Account (Years)	Region	Number of Late Payment in Past 12 Months
1	21	2	West	4
2	24	3	Northwest	3
3	35	12	South	1
4	52	20	East	0
5	55	18	Southeast	0

Which individuals are intuitively similar to each other? What is the quantitative basis behind this intuition?

UP NEXT

Categorical Data

Vector Representations

Most distance and similarity measures for nominal categorical data (i.e. non-binary, more than two levels) expect “dummy” binary vector representations

ID	Any late payment in past 12 months?	Region
1	Y	West
2	N	South
3	Y	West
4	N	East
5	Y	East

Vector Representations

Most distance and similarity measures for nominal categorical data (i.e. non-binary, more than two levels) expect “dummy” binary vector representations

ID	Yes, had a late payment in past 12?	No, no late payment in past 12?	Region West?	Region South?	Region East?
1	1	0	1	0	0
2	0	1	0	1	0
3	1	0	1	0	0
4	0	1	0	0	1
5	1	0	0	0	1

Vector Representations

When comparing binary vectors, four types of matches and mismatches can occur (“a”, “b”, “c”, and “d” signify the matching style):

a: Both values are 1

b: The first value is 1 and the second is 0

c: The first value is 0 and the second is 1

d: Both values are 0

ID	Yes, had a late payment in past 12?	No, no late payment in past 12?	Region West?	Region South?	Region East?
1	1	0	1	0	0
2	0	1	0	1	0
3	1	0	1	0	0
4	0	1	0	0	1
5	1	0	0	0	1

Compare ID 1 to 2:
2 b, 2 c, 1 d

Compare ID 4 to 5:
1 a, 1 b, 1 c, 2 d

Simple Matching Distance

The numerator is sometimes called the *Hamming Distance*

ID	Yes, had a late payment in past 12?	No, no late payment in past 12?	Region West?	Region South?	Region East?
1	1	0	1	0	0
2	0	1	0	1	0
3	1	0	1	0	0
4	0	1	0	0	1
5	1	0	0	0	1

	1	2	3	4	5
1	-	4/5	0	4/5	2/5
2		-	4/5	2/5	4/5
3			-	4/5	2/5
4				-	2/5
5					-

$$\frac{b+c}{a+b+c+d}$$

Jaccard Dissimilarity

In set theory parlance, the corresponding similarity measure can be interpreted as “intersection over union”, or IoU

ID	Yes, had a late payment in past 12?	No, no late payment in past 12?	Region West?	Region South?	Region East?
1	1	0	1	0	0
2	0	1	0	1	0
3	1	0	1	0	0
4	0	1	0	0	1
5	1	0	0	0	1

	1	2	3	4	5
1	-	1	0	1	2/3
2		-	1	2/3	1
3			-	1	2/3
4				-	2/3
5					-

$$\frac{b + c}{a + b + c}$$

Dice Dissimilarity

Co-occurrence proportion; closely related to F_1 score

Similar to Jaccard, with the type “a” matches double-weighted

ID	Yes, had a late payment in past 12?	No, no late payment in past 12?	Region West?	Region South?	Region East?
1	1	0	1	0	0
2	0	1	0	1	0
3	1	0	1	0	0
4	0	1	0	0	1
5	1	0	0	0	1

	1	2	3	4	5
1	-	1	0	1	1/2
2		-	1	1/2	1
3			-	1	1/2
4				-	1/2
5					-

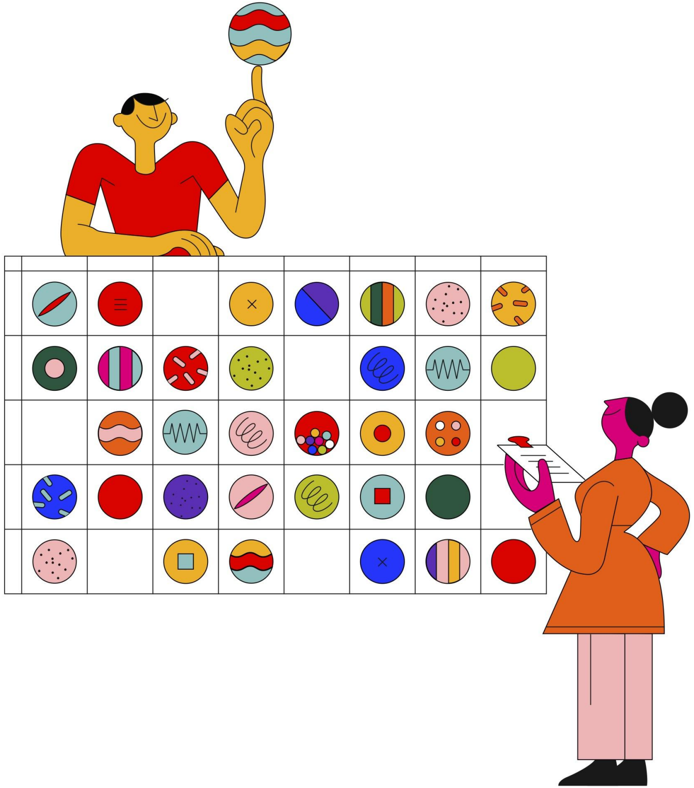
$$\frac{b+c}{2a+b+c}$$

Comparing Categorical Measures

- Simple Matching Distance/Hamming vs. Jaccard/Dice
  Key difference: inclusion vs. exclusion of type “d” (0-0) matches
- Simple Matching Distance: binary variables where 0-0 matches are informative
- Jaccard: binary variables where 0-0 matches are not informative
- Dice: non-binary categorical variables

UP NEXT

Mixed Data



Gower Distance

Far and away the most popular distance measure for mixed data; a weighted average of distances calculated on individual variables

$$Gower_{xy} = \frac{\sum_{z=1}^{n} w_{xyz} Gower_{xyz}}{\sum_{z=1}^{n} w_{xyz}}$$

- All distances are normalized on a [0,1] scale
- For numeric/quantitative variables, the Gower distance is the absolute difference in the values divided by the range; alternatively, a range-normalized version of Manhattan distance
- For ordinal variables, the categories are ranked, converted into numeric values, and then processed in the same manner as numeric values
- For binary variables, distance is equivalent to Jaccard
- For non-binary categorical variables, distance is equivalent to Dice

Gower Distance

- For numeric data -> apply range normalized Manhattan distance metric

```python
df=pd.DataFrame({'age':[21, 24, 35, 52, 55],
'age of oldest account':[2, 3, 12, 20, 18],
'region':['West', 'South', 'West', 'East','East'],
'late payments':['Y', 'N', 'Y', 'N', 'Y']})
df.head()
```

	age	age of oldest account	region	late payments
0	21	2	West	Y
1	24	3	South	N
2	35	12	West	Y
3	52	20	East	N
4	55	18	East	Y

```python
from sklearn.neighbors import DistanceMetric
man = np.asarray(df.select_dtypes(exclude=['object']))
DistanceMetric.get_metric('manhattan').pairwise(man)/max(np.ptp(man),1)
```

```
array([[0.        , 0.0754717 , 0.45283019, 0.9245283 , 0.94339623],
       [0.0754717 , 0.        , 0.37735849, 0.8490566 , 0.86792453],
       [0.45283019, 0.37735849, 0.        , 0.47169811, 0.49056604],
       [0.9245283 , 0.8490566 , 0.47169811, 0.        , 0.09433962],
       [0.94339623, 0.86792453, 0.49056604, 0.09433962, 0.        ]])
```

Gower Distance

- For categorical data -> convert to dummies and apply Jaccard distance metric

```python
df=pd.DataFrame({'age':[21, 24, 35, 52, 55],
'age of oldest account':[2, 3, 12, 20, 18],
'region':['West', 'South', 'West', 'East','East'],
'late payments':['Y', 'N', 'Y', 'N', 'Y']})
df.head()
```

	age	age of oldest account	region	late payments
0	21	2	West	Y
1	24	3	South	N
2	35	12	West	Y
3	52	20	East	N
4	55	18	East	Y

```python
from sklearn.neighbors import DistanceMetric
jac = df.select_dtypes(include=['object'])
DistanceMetric.get_metric('jaccard').pairwise(pd.get_dummies(jac))
```

```
array([[0.        , 1.        , 0.        , 1.        , 0.66666667],
       [1.        , 0.        , 1.        , 0.66666667, 1.        ],
       [0.        , 1.        , 0.        , 1.        , 0.66666667],
       [1.        , 0.66666667, 1.        , 0.        , 0.66666667],
       [0.66666667, 1.        , 0.66666667, 0.66666667, 0.        ]])
```

Write a Gower Distance Function

- Put both data types together

```
df=pd.DataFrame({'age':[21, 24, 35, 52, 55],
'age of oldest account':[2, 3, 12, 20, 18],
'region':['West', 'South', 'West', 'East','East'],
'late payments':['Y', 'N', 'Y', 'N', 'Y']})
df.head()
```

	age	age of oldest account	region	late payments
0	21	2	West	Y
1	24	3	South	N
2	35	12	West	Y
3	52	20	East	N
4	55	18	East	Y

```
from sklearn.neighbors import DistanceMetric

def gower_distance(X):
    variable_dist = []

    for i in range(X.shape[1]):

        feature = X.iloc[:,[i]]
        if feature.dtypes.values == np.object:

            feature_dist = DistanceMetric.get_metric('jaccard').pairwise(pd.get_dummies(feature))
        else:

            feature_dist = DistanceMetric.get_metric('manhattan').pairwise(feature)/max(np.ptp
(feature.values),1) #manhattan distance normalized with numpy peak to peak for range

        variable_dist.append(feature_dist)

    return np.array(variable_dist).mean(0) #return row means
```

```
gower_distance(df)
```

```
array([[0.        , 0.53594771, 0.24183007, 0.97794118, 0.72222222],
       [0.53594771, 0.        , 0.70588235, 0.69199346, 0.93627451],
       [0.24183007, 0.70588235, 0.        , 0.73611111, 0.48039216],
       [0.97794118, 0.69199346, 0.73611111, 0.        , 0.2998366 ],
       [0.72222222, 0.93627451, 0.48039216, 0.2998366 , 0.        ]])
```

Compare to the Gower Module

[Gower Module](#)

```
df=pd.DataFrame({'age':[21, 24, 35, 52, 55],
'age of oldest account':[2, 3, 12, 20, 18],
'region':['West', 'South', 'West', 'East','East'],
'late payments':['Y', 'N', 'Y', 'N', 'Y']})
df.head()
```

	age	age of oldest account	region	late payments
0	21	2	West	Y
1	24	3	South	N
2	35	12	West	Y
3	52	20	East	N
4	55	18	East	Y

```
gower_distance(df)
```

```
array([[0.        , 0.53594771, 0.24183007, 0.97794118, 0.72222222],
       [0.53594771, 0.        , 0.70588235, 0.69199346, 0.93627451],
       [0.24183007, 0.70588235, 0.        , 0.73611111, 0.48039216],
       [0.97794118, 0.69199346, 0.73611111, 0.        , 0.2998366 ],
       [0.72222222, 0.93627451, 0.48039216, 0.2998366 , 0.        ]])
```

```
import gower
gower.gower_matrix(df)
```

```
array([[0.        , 0.53594774, 0.24183007, 0.97794116, 0.7222222 ],
       [0.53594774, 0.        , 0.7058824 , 0.6919935 , 0.9362745 ],
       [0.24183007, 0.7058824 , 0.        , 0.7361111 , 0.48039216],
       [0.97794116, 0.6919935 , 0.7361111 , 0.        , 0.2998366 ],
       [0.7222222 , 0.9362745 , 0.48039216, 0.2998366 , 0.        ]],
      dtype=float32)
```

Summary

- Observations consisting of categorical data have similarity measures that are analogous to quantitative measures after conversion to binary representation: Simple Matching Distance, Jaccard, Dice
- For mixed quantitative and categorical data, Gower distance is the predominant measure of similarity

EXERCISE

1. [Jupyter notebook]
2. [Data set: Student life survey data]
3. Isolate the subset of questions related to stress
4. Calculate the dissimilarity matrix using a measure that is appropriate for categorical data
5. Identify the pairs of students with no answers in common and all answers in common
6. Determine which student's responses had the highest/lowest average similarity with other students

THINKFUL

Thank You