2020

MachineLearning-CAP5610

ASSIGNMENT-1

BHUVANESH JEEVARATHINAM

UNIVERSITY OF CENTRAL FLORIDA | DATA ANALYTICS

Task 1
Attributes

Attributes	Type 1	Type 2
Rating of an Amazon product by a person on a scale of 1 to 5	Discrete	Ordinal
The Internet Speed	Continuous	Interval
Number of customers in a store.	Discrete	Nominal
your Student ID	Discrete	Nominal
Distance	Continuous	Ratio
your letter grade (A, B, C, D)	Discrete	Ordinal
The temperature in the campus	Continuous	Interval

Task 2
Distance/Similarity Measures

Proximity Measure for Shape:

We can use *ratio of (Length / width) for* determining the shape of the boxes. For eg: Rectangular boxes will have a different ratio of length to width whereas the Square boxes will have a similar ratio. In the given problem., The ratio of boxes with vector scale (1,1) and (3,3) are 1, whereas the ratio of boxes with vector sale (1,2) and (3,6) is not 1 but it is ½. Hence from this proximity measure we can identify that the group shape each box belongs.

Proximity Measure for Size:

We can use *ratio of (Length*width)* for determining the size of the boxes. Size of a box is usually represented with the are occupied by the box. Area is determined basically by the Product of Length and its width. Hence in the given vector the Product of the elements in the vector will give us the area/Size occupied by the box. When we compare the ratio of the area of different boxes, we can get the classification for similar box size. For e.g.: Box with vector (1,1) has area = 1 unit whereas Box with vector (3,3) has area of 9 units. Hence the ratio of them is not equal to 1 which proves that the area of the two boxes are not similar. Hence, they both belong to different groups.

Task 3 - Data Preprocessing of Titanic – Part 1

I have pushed my code to a repository named CAP5610-MachineLearning. Kindly please access the code from the repo link provided.

Codelink:

https://github.com/bhuvaneshkj/CAP5610-MachineLearning-Assignment.git

Subtask 1: Analyze by describing data

```
train_df=pd.read_csv(r"C:\Users\chat2\Downloads\titanic\train.csv")
test_df=pd.read_csv(r"C:\Users\chat2\Downloads\titanic\test.csv")
df=[train_df,test_df]
data_frame=pd.concat(df,sort='True')
data_frame.head()
```

	Age	Cabin	Embarked	Fare	Name	Parch	PassengerId	Pclass	Sex	SibSp	Survived	Ticket
0	22.0	NaN	S	7.2500	Braund, Mr. Owen Harris	0	1	3	male	1	0.0	A/5 21171
1	38.0	C85	C	71.2833	Cumings, Mrs. John Bradley (Florence Briggs Th...	0	2	1	female	1	1.0	PC 17599
2	26.0	NaN	S	7.9250	Heikkinen, Miss. Laina	0	3	3	female	0	1.0	STON/O2. 3101282
3	35.0	C123	S	53.1000	Futrelle, Mrs. Jacques Heath (Lily May Peel)	0	4	1	female	1	1.0	113803
4	35.0	NaN	S	8.0500	Allen, Mr. William Henry	0	5	3	male	0	0.0	373450

Q1: Which features are available in the dataset?

```
train_df.columns
```

```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

Features which are available in dataset are PassengerId, Survived, Pclass, Name, Sex, Age, SibSp, Parch, Ticket, Fare, Cabin, Embarked respectively.

Q2: Which features are categorical?

```
dataframe_columns=data_frame.columns
cat=[]
cat_df=pd.DataFrame()
cols = ['Feature', 'Datatype','Unique Count']
dat = pd.DataFrame(columns = cols)
for k in dataframe_columns:
    dat = dat.append({'Feature': str(k), 'Datatype': str(data_frame[k].dtype) ,'Unique Count':str(data_frame[k].nunique())},ignore_index=True)
dat
```

	Feature	Datatype	Unique Count
0	Age	float64	98
1	Cabin	object	186
2	Embarked	object	3
3	Fare	float64	281
4	Name	object	1307
5	Parch	int64	8
6	PassengerId	int64	1309
7	Pclass	int64	3
8	Sex	object	2
9	SibSp	int64	7
10	Survived	float64	2
11	Ticket	object	929

From the Dataset we can find that each feature has its respective unique value counts. From the observation we can see that Features like Survived, Pclass, Sex, SibSp, Parch, Embarked etc has unique value count of less than 10. For a feature to be considered Categorical it should have a count which can be in a countable and measure level. Features like Cabin have 186 unique value which cannot be considered as categorical as 186 categories of cabins cannot exist.

Q3: Which features are numerical?

```
#Code to detect numerical datatype. select_dtypes function helps
numerical_columns = data_frame.select_dtypes([np.number]).columns
numerical_columns
```

```
Index(['Age', 'Fare', 'Parch', 'PassengerId', 'Pclass', 'SibSp', 'Survived'], dtype='object')
```

Features such as PassengerId, Survived, Pclass, Age, SibSp, Parch, Fare etc are numerical columns.

Q4: Which features are mixed data types?

```
from pandas.api.types import infer_dtype
a=["q",2]
infer_dtype(a,skipna='True')
```

```
'mixed-integer'
```

```
for col in features:
    print(infer_dtype(data_frame[col],skipna='True'))
```

```
integer
floating
integer
string
string
floating
integer
integer
string
floating
string
string
```

We tend to find the mixed data types in columns of dataset by using a function in Pandas Api which checks for it. Here we see that none of the features have mixed datatype.

Q5: Which features contain blank, null or empty values?

```
for k in dataframe_columns:
    if(data_frame[k].isnull().values.any()):
        print(k+"-"+"Yes has null values")
    else:
        print(k+"-"+"No null Values")
```

```
PassengerId-No null Values
Pclass-No null Values
Name-No null Values
Sex-No null Values
Age-Yes has null values
SibSp-No null Values
Parch-No null Values
Ticket-No null Values
Fare-Yes has null values
Cabin-Yes has null values
Embarked-Yes has null values
```

We can infer from code that features like Age, Fare, Cabin, Embarked etc. has null values.

Q6: What are the data types (e.g., integer, floats or stringsfor various features?

```
dataframe_columns=data_frame.columns
cat=[]
cat_df=pd.DataFrame()
cols = ['Feature', 'Datatype','Unique Count']
dat = pd.DataFrame(columns = cols)
for k in dataframe_columns:
    dat = dat.append({'Feature': str(k), 'Datatype': str(data_frame[k].dtype) ,'Unique Count':str(data_frame[k].nunique())},ignore_index=True)
dat
```

	Feature	Datatype	Unique Count
0	Age	float64	98
1	Cabin	object	186
2	Embarked	object	3
3	Fare	float64	281
4	Name	object	1307
5	Parch	int64	8
6	PassengerId	int64	1309
7	Pclass	int64	3
8	Sex	object	2
9	SibSp	int64	7
10	Survived	float64	2
11	Ticket	object	929

Features and its Datatypes Listed:

Passenger – Integer

Survived – float

Pclass – Integer

Name – Object

Sex – Object

Age – Float

SibSp - Int

Parch – Int

Ticket – Object

Fare – Float

Cabin – Object

Embarked – Object

Q7: To understand what is the distribution of numerical feature values across the samples, please list the properties (count, mean, std, min, 25% percentile, 50%percentile, 75% percentile, max) of numerical features?

```
data_frame.describe()
```

	Age	Fare	Parch	PassengerId	Pclass	SibSp	Survived
count	1046.000000	1308.000000	1309.000000	1309.000000	1309.000000	1309.000000	891.000000
mean	29.881138	33.295479	0.385027	655.000000	2.294882	0.498854	0.383838
std	14.413493	51.758668	0.865560	378.020061	0.837836	1.041658	0.486592
min	0.170000	0.000000	0.000000	1.000000	1.000000	0.000000	0.000000
25%	21.000000	7.895800	0.000000	328.000000	2.000000	0.000000	0.000000
50%	28.000000	14.454200	0.000000	655.000000	3.000000	0.000000	0.000000
75%	39.000000	31.275000	0.000000	982.000000	3.000000	1.000000	1.000000
max	80.000000	512.329200	9.000000	1309.000000	3.000000	8.000000	1.000000

The above tabular description shows us the Statistics of the Titanic Training Dataset.

Q8: To understand what is the distribution of categorical features, we define: count is the total number of categorical values per column; unique is the total number of unique categorical values per column; top is the most frequent categorical value; freq is the total number of the most frequent categorical value. Please the properties (count, unique, top, freq) of categorical features?

To find:

Number of Categorical Values: count() is used.

Number of Unique Categories – nunique() is used

Maximum/Top Occurring – mode() is used

Frequency of Top Occurring – value_counts().values[0] is used

	Survived	Pclass	Sex	SibSp	Parch	Embarked
Count of Categories	891	1309	1309	1309	1309	1307
Unique Categories	2	3	2	7	8	3
Top Most Occuring Category Value	0.0	3	male	0	0	S
Frequency Most Occuring Category Value	549	709	843	891	1002	914

```
cat_col=['Survived','Pclass','Sex','SibSp','Parch','Embarked']
number=[]
unique_count=[]
top=[]
freq=[]
for col in cat_col:
    number.append(data_frame[col].count())
    unique_count.append(data_frame[col].nunique())
    top.append(data_frame[col].mode().values[0])
    freq.append(data_frame[col].value_counts().values[0])
```

```
cat_index=["Count of Categories","Unique Caetgories","Top Most Occuring Category Value","Frequency Most Occuring Category Value"]
freq_df=pd.DataFrame((np.array([number,unique_count,top,freq])),columns=cat_col,index=cat_index)
freq_df
```

	Survived	Pclass	Sex	SibSp	Parch	Embarked
Count of Categories	891	1309	1309	1309	1309	1307
Unique Caetgories	2	3	2	7	8	3
Top Most Occuring Category Value	0.0	3	male	0	0	S
Frequency Most Occuring Category Value	549	709	843	891	1002	914

Codelink :

https://github.com/bhuvaneshkj/CAP5610-MachineLearning-Assignment.git