16. Data Science – ML – Dummy Variable & OneHotEncoding

Contents

1. Dataset : homeprices2.csv 2

2. Categorical data 3

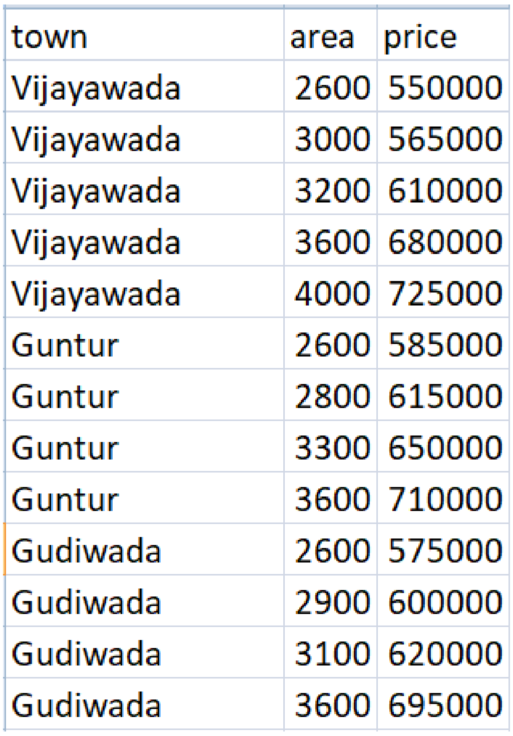
3. How to handle text data in town column 4

4. Model may behaves like below, ... 4

5. Dummy variables 5

16. Data Science – ML – Dummy Variable & OneHotEncoding

1. Dataset : homeprices2.csv

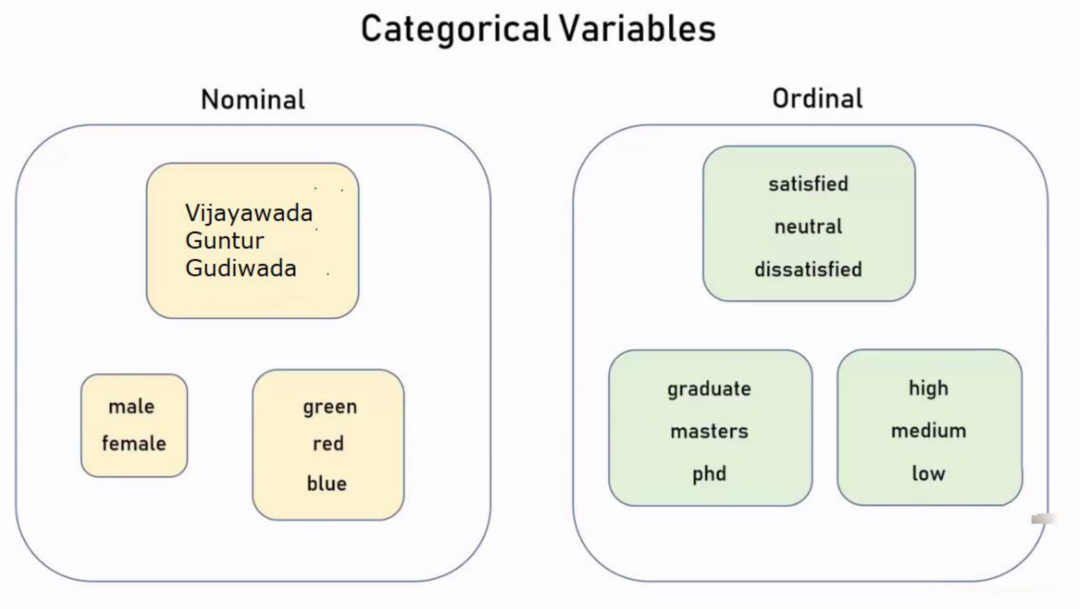


town	area	price
Vijayawada	2600	550000
Vijayawada	3000	565000
Vijayawada	3200	610000
Vijayawada	3600	680000
Vijayawada	4000	725000
Guntur	2600	585000
Guntur	2800	615000
Guntur	3300	650000
Guntur	3600	710000
Gudiwada	2600	575000
Gudiwada	2900	600000
Gudiwada	3100	620000
Gudiwada	3600	695000

Prediction

- ✓ With 3400 sqr ft area in Guntur
- ✓ With 2800 sqr ft area in Gudiwada

2. Categorical data



Categorical Variables

Nominal

Vijayawada
Guntur
Gudiwada

male
female

green
red
blue

Ordinal

satisfied
neutral
dissatisfied

graduate
masters
phd

high
medium
low

3. How to handle text data in town column

- ✓ Here town column having text data
- ✓ How to handle text data in numeric model?

4. Model may behaves like below,

- ✓ We can convert text data into number and can proceed to the next step.
  - Vijayawada - 1
  - Guntur - 2
  - Gudiwada - 3

- ✓ If we are giving data to the model then model assumes that the order like 1, 2, 3 and follow the below approach

  - Vijayawada < Guntur < Gudiwada

    - Or

  - Vijayawada + Guntur = Gudiwada

5. Dummy variables

- ✓ So, in this scenario we need to create dummy variable

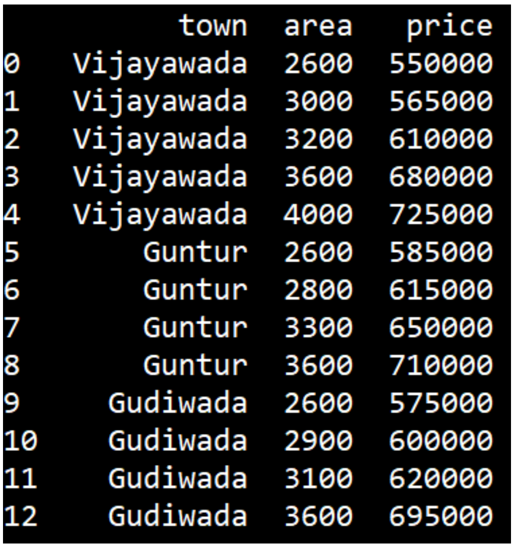
Town	Area	Price	Gudiwada	Guntur	Vijayawada
Gudiwada	2600	575000	1	0	0
Gudiwada	2900	600000	1	0	0
Gudiwada	3100	620000	1	0	0
Gudiwada	3600	695000	1	0	0
Guntur	2600	585000	0	1	0
Guntur	2800	615000	0	1	0
Guntur	3300	650000	0	1	0
Guntur	3600	710000	0	1	0
Vijayawada	2600	550000	0	0	1
Vijayawada	3000	565000	0	0	1
Vijayawada	3200	610000	0	0	1
Vijayawada	3600	680000	0	0	1
Vijayawada	4000	725000	0	0	1

Program Name

Loading dataset
demo1.py

```
import pandas as pd

df = pd.read_csv("homeprices2.csv")

print(df)
```

Output

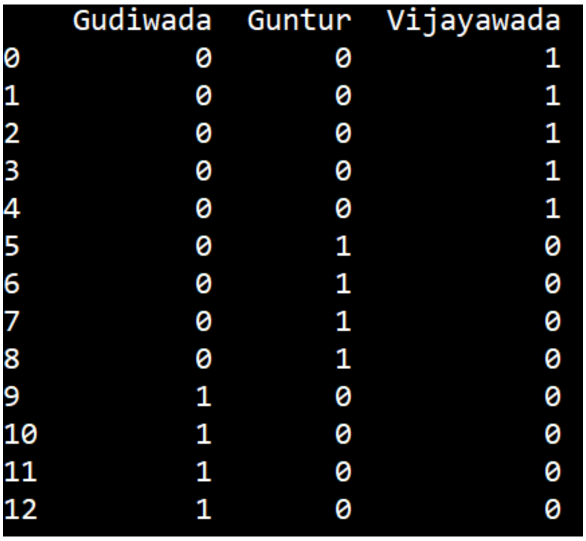


```
          town  area   price
0   Vijayawada  2600  550000
1   Vijayawada  3000  565000
2   Vijayawada  3200  610000
3   Vijayawada  3600  680000
4   Vijayawada  4000  725000
5       Guntur  2600  585000
6       Guntur  2800  615000
7       Guntur  3300  650000
8       Guntur  3600  710000
9     Gudiwada  2600  575000
10    Gudiwada  2900  600000
11    Gudiwada  3100  620000
12    Gudiwada  3600  695000
```

Program Name Creating dummy variables by using pandas
demo2.py

```
import pandas as pd

df = pd.read_csv("homeprices2.csv")
dummies = pd.get_dummies(df.town)

print(dummies)
```

Output



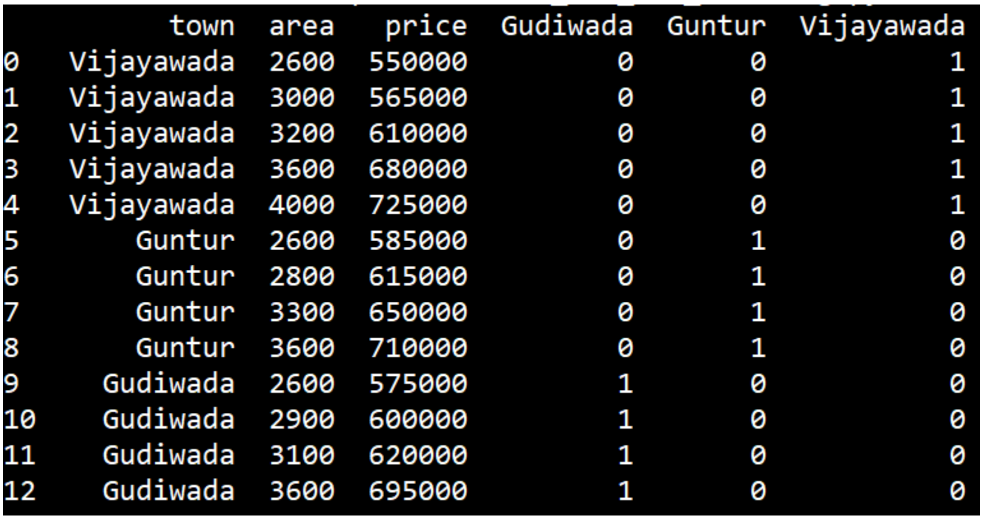
```
    Gudiwada  Guntur  Vijayawada
0          0       0           1
1          0       0           1
2          0       0           1
3          0       0           1
4          0       0           1
5          0       1           0
6          0       1           0
7          0       1           0
8          0       1           0
9          1       0           0
10         1       0           0
11         1       0           0
12         1       0           0
```

Program Name

Creating dummy variables and adding to the dataframe
demo3.py

```
import pandas as pd

df = pd.read_csv("homeprices2.csv")
dummies = pd.get_dummies(df.town)
merged = pd.concat([df, dummies], axis='columns')

print(merged)
```

Output



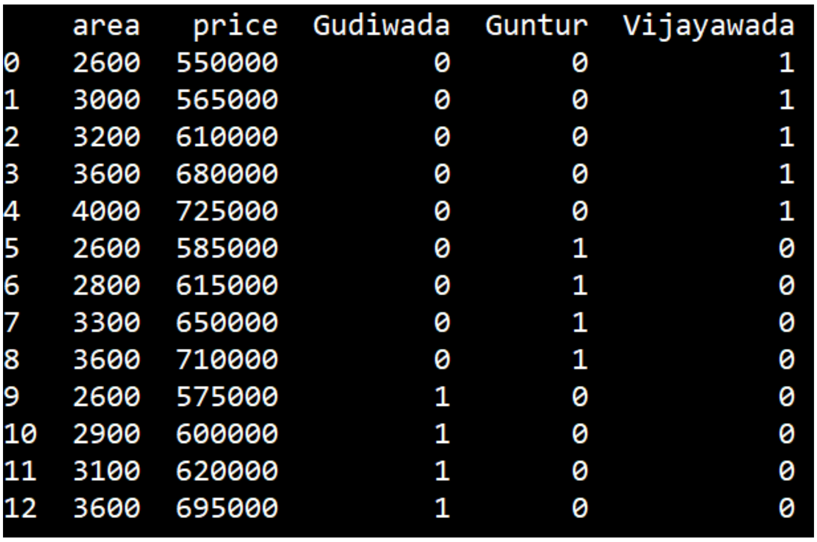
```
          town  area   price  Gudiwada  Guntur  Vijayawada
0   Vijayawada  2600  550000         0       0           1
1   Vijayawada  3000  565000         0       0           1
2   Vijayawada  3200  610000         0       0           1
3   Vijayawada  3600  680000         0       0           1
4   Vijayawada  4000  725000         0       0           1
5       Guntur  2600  585000         0       1           0
6       Guntur  2800  615000         0       1           0
7       Guntur  3300  650000         0       1           0
8       Guntur  3600  710000         0       1           0
9     Gudiwada  2600  575000         1       0           0
10    Gudiwada  2900  600000         1       0           0
11    Gudiwada  3100  620000         1       0           0
12    Gudiwada  3600  695000         1       0           0
```

Program Name

Creating dummy variables and preparing the dataframe
demo4.py

```
import pandas as pd

df = pd.read_csv("homeprices2.csv")
dummies = pd.get_dummies(df.town)

merged = pd.concat([df, dummies],axis='columns')

final = merged.drop(['town'], axis='columns')

print(final)
```

Output

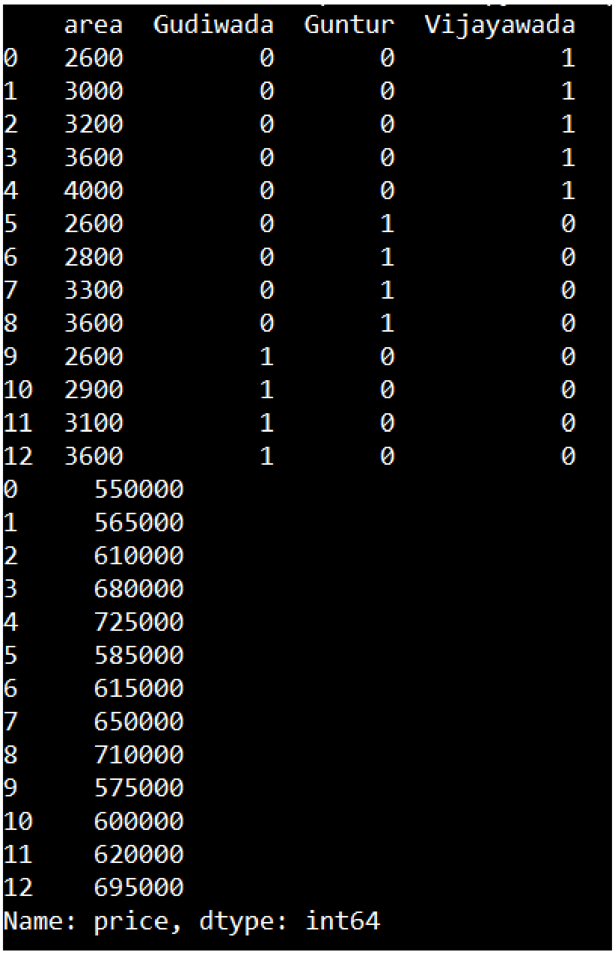


```
    area   price  Gudiwada  Guntur  Vijayawada
0   2600  550000         0       0           1
1   3000  565000         0       0           1
2   3200  610000         0       0           1
3   3600  680000         0       0           1
4   4000  725000         0       0           1
5   2600  585000         0       1           0
6   2800  615000         0       1           0
7   3300  650000         0       1           0
8   3600  710000         0       1           0
9   2600  575000         1       0           0
10  2900  600000         1       0           0
11  3100  620000         1       0           0
12  3600  695000         1       0           0
```

Program Name	Preparing X and y demo5.py

```python
import pandas as pd

df = pd.read_csv("homeprices2.csv")
dummies = pd.get_dummies(df.town)
merged = pd.concat([df, dummies],axis='columns')
final = merged.drop(['town'], axis='columns')

X = final.drop('price', axis='columns')
y = final.price

print(X)
print(y)
```

Output



```
    area  Gudiwada  Guntur  Vijayawada
0   2600         0       0           1
1   3000         0       0           1
2   3200         0       0           1
3   3600         0       0           1
4   4000         0       0           1
5   2600         0       1           0
6   2800         0       1           0
7   3300         0       1           0
8   3600         0       1           0
9   2600         1       0           0
10  2900         1       0           0
11  3100         1       0           0
12  3600         1       0           0
0     550000
1     565000
2     610000
3     680000
4     725000
5     585000
6     615000
7     650000
8     710000
9     575000
10    600000
11    620000
12    695000
Name: price, dtype: int64
```

Program Name

Creating a model
demo6.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

df = pd.read_csv("homeprices2.csv")
dummies = pd.get_dummies(df.town)
merged = pd.concat([df, dummies],axis='columns')
final = merged.drop(['town'], axis='columns')

X = final.drop('price', axis='columns')
y = final.price

model = LinearRegression()
model.fit(X, y)
print("Model got trained")
```

Output

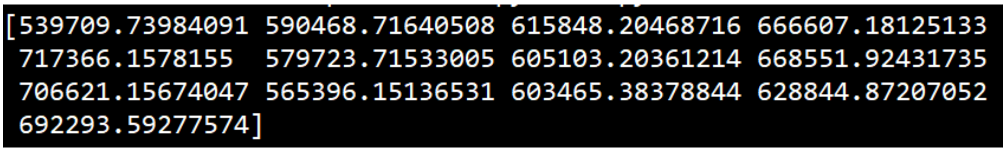
```
Model got trained
```

Program Name

Predicting the house prices
demo7.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

df=pd.read_csv("homeprices2.csv")
dummies = pd.get_dummies(df.town)
merged = pd.concat([df, dummies],axis='columns')
final = merged.drop(['town'], axis='columns')

X = final.drop('price', axis='columns')
y = final.price

model = LinearRegression()
model.fit(X, y)
print(model.predict(X))
```

Output



```
[539709.73984091 590468.71640508 615848.20468716 666607.18125133
 717366.1578155  579723.71533005 605103.20361214 668551.92431735
 706621.15674047 565396.15136531 603465.38378844 628844.87207052
 692293.59277574]
```

Program Checking the score

Name demo8.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

df=pd.read_csv("homeprices2.csv")
dummies = pd.get_dummies(df.town)
merged = pd.concat([df, dummies],axis='columns')
final = merged.drop(['town'], axis='columns')

X = final.drop('price', axis='columns')
y = final.price

model = LinearRegression()
model.fit(X, y)
print(model.score(X, y))
```

Output

```
0.9573929037221873
```

Program Predicting house price in Vijayawada

Name demo9.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

df=pd.read_csv("homeprices2.csv")
dummies = pd.get_dummies(df.town)
merged = pd.concat([df, dummies],axis='columns')
final = merged.drop(['town'], axis='columns')

X = final.drop('price', axis='columns')
y = final.price

model = LinearRegression()
model.fit(X, y)
print(model.predict([[3400, 0, 0, 1]]))
```

Output

```
[641227.69296925]
```

Program Predicting house price in Guntur
Name demo10.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

df=pd.read_csv("homeprices2.csv")
dummies = pd.get_dummies(df.town)
merged = pd.concat([df, dummies],axis='columns')
final = merged.drop(['town'], axis='columns')

X = final.drop('price', axis='columns')
y = final.price

model = LinearRegression()
model.fit(X, y)
print(model.predict([[3400, 0, 1, 0]]) )
```

Output

```
[681241.66845839]
```

Program Name

Predicting house price in Gudiwada
demo11.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

df=pd.read_csv("homeprices2.csv")
dummies = pd.get_dummies(df.town)
merged = pd.concat([df, dummies],axis='columns')
final = merged.drop(['town'], axis='columns')

X = final.drop('price', axis='columns')
y = final.price

model = LinearRegression()
model.fit(X, y)
print(model.predict([[3400, 1, 0, 0]]) )
```

Output

```
[666914.10449365]
```