NAME: Wairiuko Samuel

COURSE: Python Research Paper

TITLE: Encoding

Encoding

Encoding is the process of converting data from one form to another .eg changing the labels into numeric form so as to convert it into the machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated.

How it is used

Encoding is an important pre-processing step for the structured dataset in supervised learning. Converts categorical column into numeric column. Encoding can either be Label or One Hot Coding

Benefits

- ✓ There are automation tools you can use to encode and archive your files as they are created. This is a solution you should explore if you need to have back-ups of your files.
- ✓ Encoding keeps your data safe since the files are not readable. This is a good way of protecting data from theft since any stolen file would not be usable.
- ✓ It’s an ideal solution if you need third party to access your files and you want to limit access to sensitive files that contain vital information.
- ✓ Encoding removes redundancies from data, through the dummy variables the files become smaller and faster to access and process.
- ✓ Encoded data reduces the storage space in the computer thus an ideal way to store massive amount of data.
- ✓ Encoded data is easy to manage thus becoming the easiest way to organize your data in an automated way.
- ✓ There are automation tools that can be used to encode and store files as soon as they are created. this makes exploration and access easier.

Example

Label Encoding

```
import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pylab as plt

#%matplotlib inline
```

```
import seaborn as sns

tips= pd.read_csv('restaurant_tips.csv')

tips. head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

To encode sex column

METHOD 1

Dummy encoding

```
data=pd.get_dummies(tips["sex"])
```

results

	Female	Male
0	1	0
1	0	1
2	0	1
3	0	1

When building the model create a new column sex_encoded and drop sex column to avoid overfitting the model.

METHOD 2

Label Encoding

```
from sklearn. preprocessing import LabelEncoder

l3=LabelEncoder()

label=le.fit_transform(tips["day"])
```

After encoding a machine learning algorithm like Logistic Regression can predict 0 or 1 and each prediction may actually have been a 0 or 1. Predictions for 0 that were actually 0 appear in the cell for prediction=0 and actual=0, whereas predictions for 0 that were actually 1 appear in the cell for prediction = 0 and actual=1. And so on.

One Hot Coding

In customer churn dataset I used One-Hot-Coding to creating dummy variables for geographical variable that has 3 categories – ‘France, ‘Germany’ and ‘Spain’. One hot coding removed this variable and generated 3 new dummy variables 0 and 1 using ‘get dummies’ function of Pandas.

```
df = pd.read_csv('churn.csv')

Geography_dummies = pd.get_dummies(DATA1.Geography)

J=pd.concat([DATA1, Geography_dummies], axis=1)
```

Resources

1. Machine learning, Data Mining and Big Data Analytics Lecture Notes by Gitimoni Saikia
2. Python Project Lecture Notes by Vijay Kumar