ML5 Ensemble Learning

October 27, 2023

[1]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
#import category_encoders as ce
from sklearn.metrics import accuracy_score, confusion_matrix

data = pd.read_csv("C:/Users/hp/Downloads/Practical_Data/car_evaluation.csv")
data.head()
```

[1]:

	vhigh	vhigh.1	2	2.1	small	low	unacc
0	vhigh	vhigh	2	2	small	med	unacc
1	vhigh	vhigh	2	2	small	high	unacc
2	vhigh	vhigh	2	2	med	low	unacc
3	vhigh	vhigh	2	2	med	med	unacc
4	vhigh	vhigh	2	2	med	high	unacc

[2]:
```
pip install category_encoders
```

```
Requirement already satisfied: category_encoders in
c:\users\hp\anaconda3\lib\site-packages (2.6.2)
Requirement already satisfied: patsy>=0.5.1 in c:\users\hp\anaconda3\lib\site-
packages (from category_encoders) (0.5.2)
Requirement already satisfied: scipy>=1.0.0 in c:\users\hp\anaconda3\lib\site-
packages (from category_encoders) (1.7.3)
Requirement already satisfied: statsmodels>=0.9.0 in
c:\users\hp\anaconda3\lib\site-packages (from category_encoders) (0.13.2)
Requirement already satisfied: scikit-learn>=0.20.0 in
c:\users\hp\anaconda3\lib\site-packages (from category_encoders) (1.0.2)
Requirement already satisfied: pandas>=1.0.5 in c:\users\hp\anaconda3\lib\site-
packages (from category_encoders) (1.4.2)
Requirement already satisfied: numpy>=1.14.0 in c:\users\hp\anaconda3\lib\site-
packages (from category_encoders) (1.22.4)
Requirement already satisfied: pytz>=2020.1 in c:\users\hp\anaconda3\lib\site-
packages (from pandas>=1.0.5->category_encoders) (2021.3)
Requirement already satisfied: python-dateutil>=2.8.1 in
c:\users\hp\anaconda3\lib\site-packages (from pandas>=1.0.5->category_encoders)
(2.8.2)
Requirement already satisfied: six in c:\users\hp\anaconda3\lib\site-packages
```

```
(from patsy>=0.5.1->category_encoders) (1.16.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in
c:\users\hp\anaconda3\lib\site-packages (from scikit-
learn>=0.20.0->category_encoders) (2.2.0)
Requirement already satisfied: joblib>=0.11 in c:\users\hp\anaconda3\lib\site-
packages (from scikit-learn>=0.20.0->category_encoders) (1.1.0)
Requirement already satisfied: packaging>=21.3 in
c:\users\hp\anaconda3\lib\site-packages (from
statsmodels>=0.9.0->category_encoders) (21.3)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in
c:\users\hp\anaconda3\lib\site-packages (from
packaging>=21.3->statsmodels>=0.9.0->category_encoders) (3.0.4)
Note: you may need to restart the kernel to use updated packages.
```

[3]:
```
import category_encoders as ce
```

[4]:
```
col_names = ['buying', 'maint', 'doors', 'persons', 'lug_boot', 'safety',␣
 ↪'class']
data.columns = col_names

data.head()
```

[4]:

	buying	maint	doors	persons	lug_boot	safety	class
0	vhigh	vhigh	2	2	small	med	unacc
1	vhigh	vhigh	2	2	small	high	unacc
2	vhigh	vhigh	2	2	med	low	unacc
3	vhigh	vhigh	2	2	med	med	unacc
4	vhigh	vhigh	2	2	med	high	unacc

[5]:
```
X =data.drop(['class'],axis=1)
y = data['class']

X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.
 ↪3,random_state=42)
X_train.shape,X_test.shape
```

[5]:
```
((1208, 6), (519, 6))
```

[6]:
```
encoder = ce.OrdinalEncoder(cols=['buying', 'maint', 'doors', 'persons',␣
 ↪'lug_boot', 'safety'])
X_train = encoder.fit_transform(X_train)
X_test = encoder.transform(X_test)
```

[7]:
```
rfc=RandomForestClassifier(random_state=0)
rfc.fit(X_train,y_train)
```

[7]:
```
RandomForestClassifier(random_state=0)
```

[8]:
```
y_pred = rfc.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)

print("Accuracy:", accuracy,"\n")

print("Confusion Matrix:\n", conf_matrix)
```

```
Accuracy: 0.928709055876686

Confusion Matrix:
 [[107   2   8   1]
 [  8   6   2   1]
 [  7   0 354   0]
 [  7   1   0  15]]
```

[ ]: