```
!pip install tensorflow
```

```
Requirement already satisfied: tensorflow in /usr/local/lib/python3.7/dist-packages (2.7
Requirement already satisfied: astunparse>=1.6.0 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: protobuf>=3.9.2 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: h5py>=2.9.0 in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: keras<2.8,>=2.7.0rc0 in /usr/local/lib/python3.7/dist-pac
Requirement already satisfied: wrapt>=1.11.0 in /usr/local/lib/python3.7/dist-packages (
Requirement already satisfied: typing-extensions>=3.6.6 in /usr/local/lib/python3.7/dist
Requirement already satisfied: wheel<1.0,>=0.32.0 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: absl-py>=0.4.0 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: libclang>=9.0.1 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: tensorboard~=2.6 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: flatbuffers<3.0,>=1.12 in /usr/local/lib/python3.7/dist-p
Requirement already satisfied: tensorflow-estimator<2.8,~=2.7.0rc0 in /usr/local/lib/pyt
Requirement already satisfied: gast<0.5.0,>=0.2.1 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: numpy>=1.14.5 in /usr/local/lib/python3.7/dist-packages (
Requirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: google-pasta>=0.1.1 in /usr/local/lib/python3.7/dist-pack
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.21.0 in /usr/local/lib/py
Requirement already satisfied: keras-preprocessing>=1.1.1 in /usr/local/lib/python3.7/di
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /usr/local/lib/python3.7/dist-pack
Requirement already satisfied: cached-property in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: google-auth-oauthlib<0.5,>=0.4.1 in /usr/local/lib/python
Requirement already satisfied: google-auth<3,>=1.6.3 in /usr/local/lib/python3.7/dist-pa
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in /usr/local/lib/python3.7
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: tensorboard-data-server<0.7.0,>=0.6.0 in /usr/local/lib/p
Requirement already satisfied: setuptools>=41.0.0 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.7/dist-pack
Requirement already satisfied: werkzeug>=0.11.15 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: cachetools<5.0,>=2.0.0 in /usr/local/lib/python3.7/dist-p
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.7/dist-pa
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.7/dist-packages (
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.7/dist
Requirement already satisfied: importlib-metadata>=4.4 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in /usr/local/lib/python3.7/dist-pac
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local/lib
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.7/dist-packages
```

```
from google.colab import drive
```

```
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mour
```

```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import classification_report
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.model_selection import train_test_split, cross_val_score


forest=pd.read_csv('/content/drive/MyDrive/Colab Notebooks/forestfires.csv')
forest
```

	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area	dayfri	daymon	da
0	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.00	1	0	
1	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.00	0	0	
2	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0.00	0	0	
3	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0.00	1	0	
4	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0.00	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
512	aug	sun	81.6	56.7	665.6	1.9	27.8	32	2.7	0.0	6.44	0	0	
513	aug	sun	81.6	56.7	665.6	1.9	21.9	71	5.8	0.0	54.29	0	0	
514	aug	sun	81.6	56.7	665.6	1.9	21.2	70	6.7	0.0	11.16	0	0	
515	aug	sat	94.4	146.0	614.7	11.3	25.6	42	4.0	0.0	0.00	0	0	
516	nov	tue	79.5	3.0	106.7	1.1	11.8	31	4.5	0.0	0.00	0	0	

517 rows × 31 columns

```
forest.shape
```

```
(517, 31)
```

```
forest.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 517 entries, 0 to 516
Data columns (total 31 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   month          517 non-null    object
 1   day            517 non-null    object
```

```
 2   FFMC           517 non-null    float64
 3   DMC            517 non-null    float64
 4   DC             517 non-null    float64
 5   ISI            517 non-null    float64
 6   temp           517 non-null    float64
 7   RH             517 non-null    int64
 8   wind           517 non-null    float64
 9   rain           517 non-null    float64
 10  area           517 non-null    float64
 11  dayfri         517 non-null    int64
 12  daymon         517 non-null    int64
 13  daysat         517 non-null    int64
 14  daysun         517 non-null    int64
 15  daythu         517 non-null    int64
 16  daytue         517 non-null    int64
 17  daywed         517 non-null    int64
 18  monthapr       517 non-null    int64
 19  monthaug       517 non-null    int64
 20  monthdec       517 non-null    int64
 21  monthfeb       517 non-null    int64
 22  monthjan       517 non-null    int64
 23  monthjul       517 non-null    int64
 24  monthjun       517 non-null    int64
 25  monthmar       517 non-null    int64
 26  monthmay       517 non-null    int64
 27  monthnov       517 non-null    int64
 28  monthoct       517 non-null    int64
 29  monthsep       517 non-null    int64
 30  size_category  517 non-null    object
dtypes: float64(8), int64(20), object(3)
memory usage: 125.3+ KB
```

```
forest.describe()
```

	FFMC	DMC	DC	ISI	temp	RH	wind
count	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000
mean	90.644681	110.872340	547.940039	9.021663	18.889168	44.288201	4.017602
std	5.520111	64.046482	248.066192	4.559477	5.806625	16.317469	1.791653
min	18.700000	1.100000	7.900000	0.000000	2.200000	15.000000	0.400000
25%	90.200000	68.600000	437.700000	6.500000	15.500000	33.000000	2.700000
50%	91.600000	108.300000	664.200000	8.400000	19.300000	42.000000	4.000000
75%	92.900000	142.400000	713.900000	10.800000	22.800000	53.000000	4.900000
max	96.200000	291.300000	860.600000	56.100000	33.300000	100.000000	9.400000

```
forest.isna().sum()
```

```
month           0
day             0
FFMC            0
DMC             0
DC              0
ISI             0
temp            0
RH              0
wind            0
rain            0
area            0
dayfri          0
daymon          0
daysat          0
daysun          0
daythu          0
daytue          0
daywed          0
monthapr        0
monthaug        0
monthdec        0
monthfeb        0
monthjan        0
monthjul        0
monthjun        0
monthmar        0
monthmay        0
monthnov        0
monthoct        0
monthsep        0
size_category   0
dtype: int64
```

```
forest2=forest.drop(['dayfri','daymon','daysat','daysun','daythu','daytue','daywed','monthapr
forest2
```

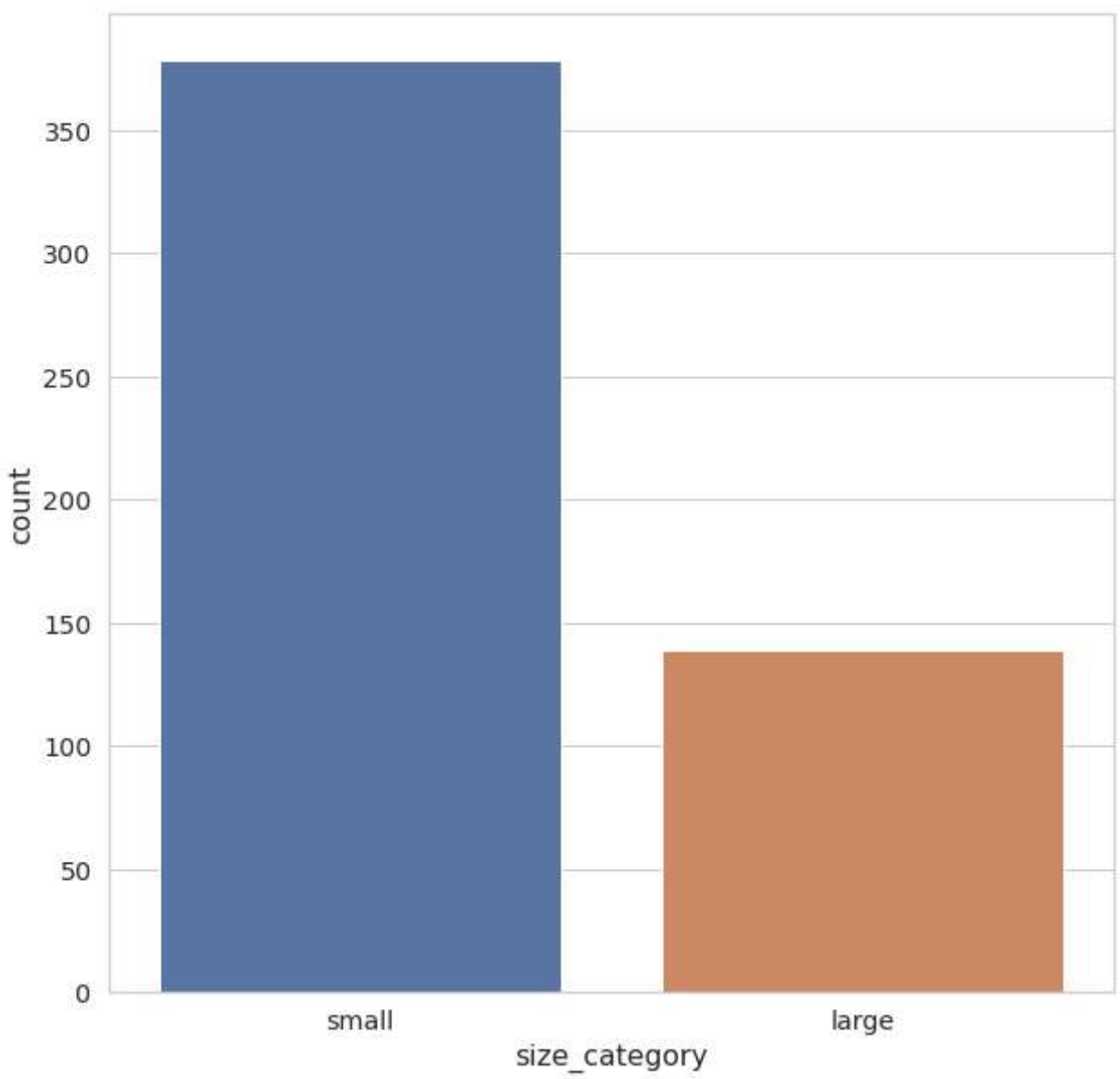
	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area	size_category
0	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.00	small
1	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.00	small

```
forest.size_category.value_counts()
```

```
small    378
large    139
Name: size_category, dtype: int64
```

```
import seaborn as sns
import matplotlib.pyplot as plt

sns.countplot(x='size_category', data=forest)
plt.show()
```

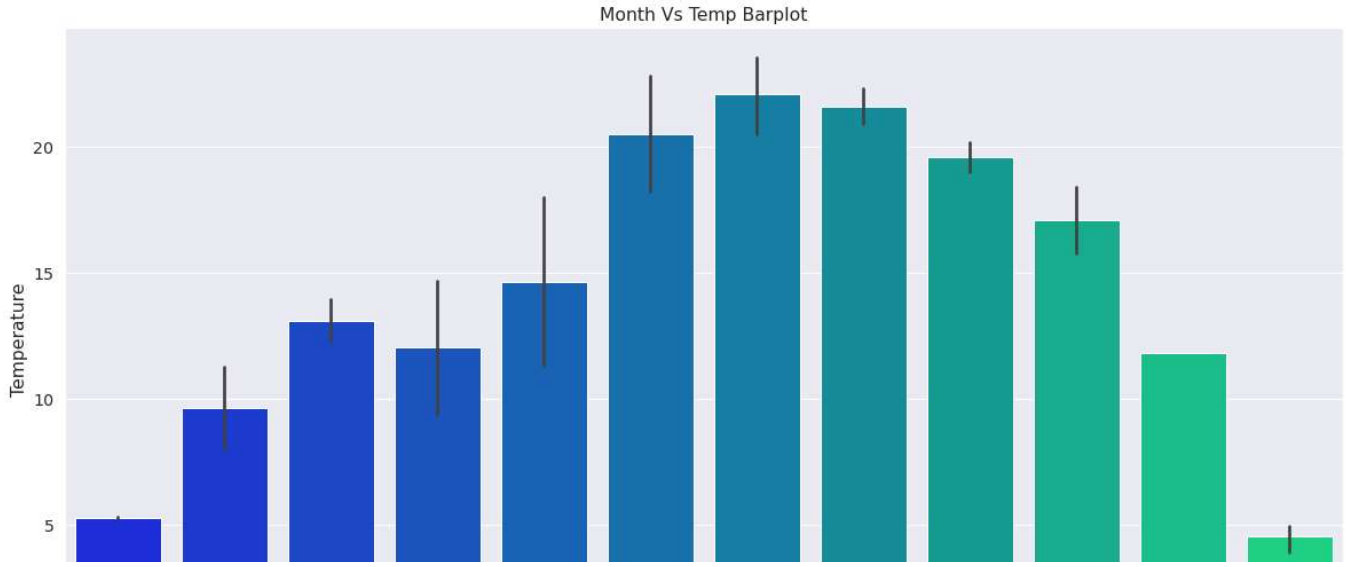


```
# checking for which value of area is categorised into large and small by creating crosstab b
pd.crosstab(forest.area, forest.size_category)
```

size_category	large	small
area		
0.00	0	247
0.09	0	1
0.17	0	1
0.21	0	1
0.24	0	1
...	...	...
200.94	1	0
212.88	1	0
278.53	1	0
746.28	1	0
1090.84	1	0

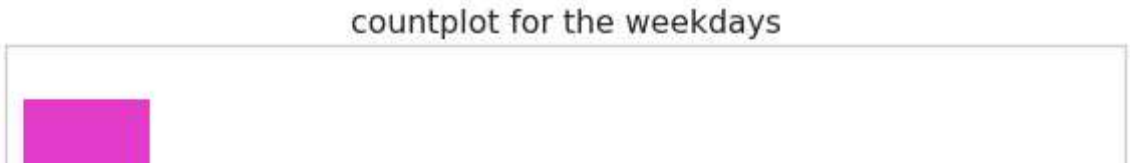
251 rows × 2 columns

```
plt.rcParams['figure.figsize']=[20,10]
sns.set(style="darkgrid", font_scale=1.3)
month_temp=sns.barplot(x='month', y='temp', data=forest, order =['jan','feb','mar','apr','may
month_temp.set(title='Month Vs Temp Barplot', xlabel='Month',ylabel='Temperature');
```

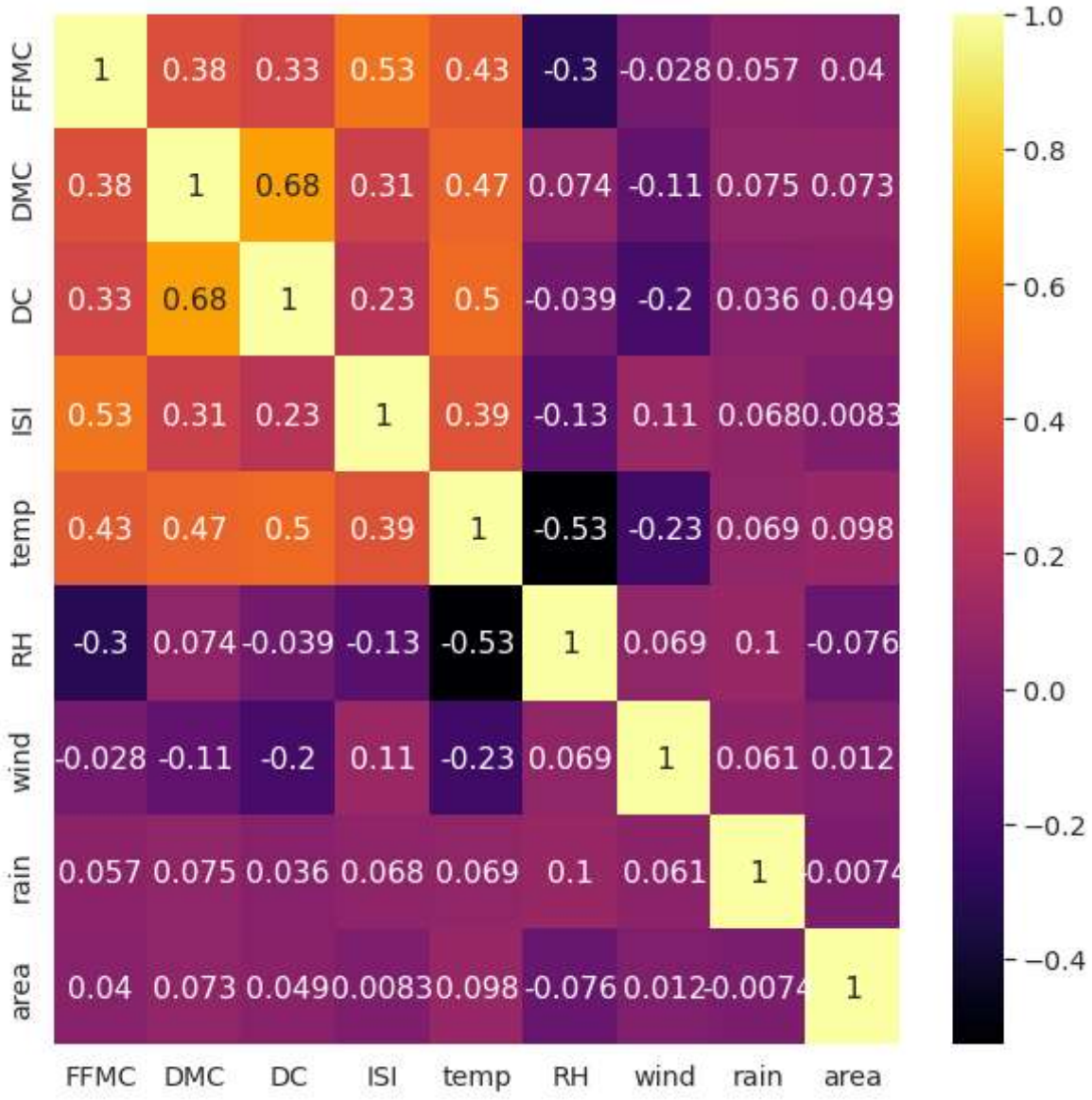



```
plt.rcParams['figure.figsize']=[10,10]
sns.set(style='whitegrid',font_scale=1.3)
day=sns.countplot(forest['day'], order=['sun','mon','tue','wed','thu','fri','sat'], palette='
day.set(title='countplot for the weekdays', xlabel ='days',ylabel='count');
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass th
  FutureWarning
```




```
sns.heatmap(forest2.corr(), annot = True,cmap="inferno")
plt.show()
```




```
# encoding month and day features
forest2.month.replace(('jan','feb','mar','apr','may','jun','jul','aug','sep','oct','nov','dec
forest2.day.replace(('mon','tue','wed','thu','fri','sat','sun'),(1,2,3,4,5,6,7), inplace=True
forest2.head(10)
```

	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area	size_category
0	3	5	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.0	small
1	10	2	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.0	small
2	10	6	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0.0	small
3	3	5	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0.0	small
4	3	7	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0.0	small
5	8	7	92.3	85.3	488.0	14.7	22.2	29	5.4	0.0	0.0	small
6	8	1	92.3	88.9	495.6	8.5	24.1	27	3.1	0.0	0.0	small
7	8	1	91.5	145.4	608.2	10.7	8.0	86	2.2	0.0	0.0	small

```
# encoding target variable 'size category'
forest2.size_category.replace(('small','large'),(0,1), inplace=True)
forest2
```

	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area	size_category
0	3	5	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.00	0
1	10	2	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.00	0
2	10	6	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0.00	0
3	3	5	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0.00	0
4	3	7	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0.00	0
...	...	...	...	...	...	...	...	...	...	...	...	...
512	8	7	81.6	56.7	665.6	1.9	27.8	32	2.7	0.0	6.44	1
513	8	7	81.6	56.7	665.6	1.9	21.9	71	5.8	0.0	54.29	1
514	8	7	81.6	56.7	665.6	1.9	21.2	70	6.7	0.0	11.16	1
515	8	6	94.4	146.0	614.7	11.3	25.6	42	4.0	0.0	0.00	0
516	11	2	79.5	3.0	106.7	1.1	11.8	31	4.5	0.0	0.00	0

517 rows × 12 columns

```
X = forest2.drop('size_category', axis=1)
y= forest2['size_category']


scaler=StandardScaler()


scaled_X = scaler.fit(X)
scaled_X
```

```
StandardScaler()
```

```
scaled_features=scaler.transform(X)
forest_head=pd.DataFrame(scaled_features, columns=forest2.columns[:-1])
forest_head
```

	month	day	FFMC	DMC	DC	ISI	temp	RH
0	-1.968443	0.357721	-0.805959	-1.323326	-1.830477	-0.860946	-1.842640	0.411724
1	1.110120	-1.090909	-0.008102	-1.179541	0.488891	-0.509688	-0.153278	-0.692456
2	1.110120	0.840597	-0.008102	-1.049822	0.560715	-0.509688	-0.739383	-0.692456
3	-1.968443	0.357721	0.191362	-1.212361	-1.898266	-0.004756	-1.825402	3.233519
4	-1.968443	1.323474	-0.243833	-0.931043	-1.798600	0.126966	-1.291012	3.356206
...	...	...	...	...	...	...	...	...
512	0.230531	1.323474	-1.640083	-0.846648	0.474768	-1.563460	1.536084	-0.753800
513	0.230531	1.323474	-1.640083	-0.846648	0.474768	-1.563460	0.519019	1.638592
514	0.230531	1.323474	-1.640083	-0.846648	0.474768	-1.563460	0.398350	1.577248
515	0.230531	0.840597	0.680957	0.549003	0.269382	0.500176	1.156839	-0.140366
516	1.549915	-1.090909	-2.020879	-1.685913	-1.780442	-1.739089	-1.222058	-0.815143

517 rows × 11 columns

```
X_train,X_test,y_train,y_test = train_test_split(forest_head,y,test_size=0.20, random_state=1
```

```
X_train.shape,y_train.shape
```

```
((413, 11), (413,))
```

```
X_test.shape, y_test.shape
```

```
((104, 11), (104,))
```

```
import tensorflow as tf
from keras.models import Sequential
from keras.layers import Dense
model = Sequential()
model.add(Dense(units=12, activation='relu'))
model.add(Dense(units=10, activation='tanh'))
model.add(Dense(units=1, activation= 'softmax'))
```

```
model.compile(optimizer='adam', loss = 'binary_crossentropy', metrics=['accuracy'])
```

```
model.fit(X_train, y_train, epochs= 10, batch_size=10)
```

```
Epoch 1/10
42/42 [==============================] - 0s 1ms/step - loss: 0.0121 - accuracy: 0.2518
Epoch 2/10
42/42 [==============================] - 0s 1ms/step - loss: 0.0115 - accuracy: 0.2518
Epoch 3/10
42/42 [==============================] - 0s 2ms/step - loss: 0.0105 - accuracy: 0.2518
Epoch 4/10
42/42 [==============================] - 0s 2ms/step - loss: 0.0108 - accuracy: 0.2518
Epoch 5/10
42/42 [==============================] - 0s 2ms/step - loss: 0.0100 - accuracy: 0.2518
Epoch 6/10
42/42 [==============================] - 0s 2ms/step - loss: 0.0104 - accuracy: 0.2518
Epoch 7/10
42/42 [==============================] - 0s 2ms/step - loss: 0.0095 - accuracy: 0.2518
Epoch 8/10
42/42 [==============================] - 0s 2ms/step - loss: 0.0094 - accuracy: 0.2518
Epoch 9/10
42/42 [==============================] - 0s 1ms/step - loss: 0.0090 - accuracy: 0.2518
Epoch 10/10
42/42 [==============================] - 0s 2ms/step - loss: 0.0089 - accuracy: 0.2518
<keras.callbacks.History at 0x7f24a83afb10>
```

```
scores=model.evaluate(X_test,y_test)
scores
```

```
4/4 [==============================] - 0s 2ms/step - loss: 0.2233 - accuracy: 0.3365
[0.2232731282711029, 0.3365384638309479]
```

```
print('Model Accuracy : ', round(scores[1],2))
print('Loss           : ', round(scores[0],2))
```

```
Model Accuracy :  0.34
Loss           :  0.22
```

✓ 2s completed at 7:25 PM