Multilayer Perceptron Analysis of freight data for NRC

the freight dataset from https://https://www150.statcan.gc.ca/n1/pub/50-503-x/50-503-x2018001-eng.htm (https://https://www150.statcan.gc.ca/n1/pub/50-503-x/50-503-x2018001-eng.htm)) is used. This dataset contains information about more than 50,000 cases which were recorded from 2011 to 2016. The Canadian Freight Analysis Framework (CFAF) integrates data from several sources to create a comprehensive picture of freight flows across the country by geography, commodity and mode of transport. The framework database estimates tonnage, value, and tonne-kilometers by origin and destination, by commodity type, and by mode. They recommmend that the database can be used in a variety of analyses such as assessing highway capacity and forecasting traffic, evaluating investments in infrastructure, examining trade flows, and analyzing policies including road pricing and multimodal freight programs. Informaiton on the variables available in the database is provided. Using the information, the goal is to build a model to predict which case is profitable (for example, revenue is bigger that the specific value) and which case is not.

Data type Length Description Values

Year character 4 Reference year

Mode character 2 The mode by which the shipment(s) moved. AR = Air RL = Rail TF = Truck (for-hire)

SCTGGroup character 50 The type of commodity shipped. Groupings based on the Standard Classification of Transported Goods (SCTG) 2-digit level.

OrigCMA character 3 In Canada, the sub-provincial area (census agglomeration (CA), census metropolitan area (CMA), rest of province (ROP)) or province/territory which is the origin of the shipment(s). Outside Canada, the United States and Mexico are grouped and all other countries are grouped.

OrigProv character 2 In Canada, the province or territory which is the origin of the shipment(s). Outside Canada, the United States and Mexico are grouped and all other countries are grouped.

OrigCtry character 2 The country which is the origin of the shipment(s).

DestCMA character 3 In Canada, the sub-provincial area (census agglomeration (CA), census metropolitan area (CMA), rest of province (ROP)) or province/territory which is the destination of the shipment(s). Outside Canada, the United States and Mexico are grouped and all other countries are grouped.

DestProv character 2 In Canada, the province or territory which is the destination of the shipment(s). Outside Canada, the United States and Mexico are grouped and all other countries are grouped.

DestCtry character 2 The country which is the destination of the shipment(s).

Shipments numeric 38 The aggregate number of shipments transported. For air and truck, a shipment represents the movement of a single commodity from an origin to a destination. For rail this represents the number of cars. Blank cells contain data which have been suppressed to meet the confidentiality requirements of the Statistics Act.

Weight numeric 38 The aggregate weight of the shipments, in kilograms (kg). Blank cells contain data which have been suppressed to meet the confidentiality requirements of the Statistics Act.

Revenue numeric 38 The aggregate revenue that the carrier earned from transporting the shipments, in dollars. Blank cells contain data which have been suppressed to meet the confidentiality requirements of the Statistics Act.

Distance numeric 38 The aggregate distance that the shipments were transported, in kilometres (km). Blank cells contain data which have been suppressed to meet the confidentiality requirements of the Statistics Act.

TonneKm numeric 38 The weight of each shipment multiplied by the distance transported and then aggregated. Blank cells contain data which have been suppressed to meet the confidentiality requirements of the Statistics Act.

Value numeric 38 The aggregate value of the shipments, in dollars. Blank cells contain data which have been

Read in Data

In [1]:
```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
%matplotlib inline

freight = pd.read_csv('../../../freight.csv')
freight.head()
```

Out[1]:

	Year	Mode	Commodity	OrigCMA	OrigProv	OrigCtry	DestCMA	DestProv	DestCtry	Shi
0	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Edmonton	Alberta	CANADA	
1	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Halifax	Nova Scotia	CANADA	
2	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Hamilton	Ontario	CANADA	
3	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Montreal	Quebec	CANADA	
4	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	New Brunswick	New Brunswick	CANADA	

Clean continuous variables

Fill missing values

In [2]:
```
freight.isnull().sum()
```

Out[2]:
```
Year            0
Mode            0
Commodity       0
OrigCMA         0
OrigProv        0
OrigCtry        0
DestCMA         0
DestProv        0
DestCtry        0
Shipments    1318
Weight       1318
Revenue      1318
Distance     1318
TonneKm      1318
Value        1318
dtype: int64
```

In [3]:
```
freight['Shipments'].fillna(freight['Shipments'].mean(), inplace=True)
freight.head(10)
```

Out[3]:

	Year	Mode	Commodity	OrigCMA	OrigProv	OrigCtry	DestCMA	DestProv	Dest
0	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Edmonton	Alberta	CAN
1	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Halifax	Nova Scotia	CAN
2	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Hamilton	Ontario	CAN
3	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Montreal	Quebec	CAN
4	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	New Brunswick	New Brunswick	CAN
5	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Newfoundland and Labrador	Newfoundland and Labrador	CAN
6	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Northwest Territories	Northwest Territories	CAN
7	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Other international	Other international	C internat
8	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Prince Edward Island	Prince Edward Island	CAN
9	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Quebec City	Quebec	CAN

In [4]:
```
freight['Weight'].fillna(freight['Weight'].mean(), inplace=True)
freight.head(10)
```

Out[4]:

	Year	Mode	Commodity	OrigCMA	OrigProv	OrigCtry	DestCMA	DestProv	Dest
0	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Edmonton	Alberta	CAN
1	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Halifax	Nova Scotia	CAN
2	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Hamilton	Ontario	CAN
3	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Montreal	Quebec	CAN
4	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	New Brunswick	New Brunswick	CAN
5	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Newfoundland and Labrador	Newfoundland and Labrador	CAN
6	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Northwest Territories	Northwest Territories	CAN
7	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Other international	Other international	(internat
8	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Prince Edward Island	Prince Edward Island	CAN
9	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Quebec City	Quebec	CAN

In [5]:
```
freight['Distance'].fillna(freight['Distance'].mean(), inplace=True)
freight.head(10)
```

Out[5]:

	Year	Mode	Commodity	OrigCMA	OrigProv	OrigCtry	DestCMA	DestProv	Dest
0	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Edmonton	Alberta	CAN
1	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Halifax	Nova Scotia	CAN
2	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Hamilton	Ontario	CAN
3	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Montreal	Quebec	CAN
4	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	New Brunswick	New Brunswick	CAN
5	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Newfoundland and Labrador	Newfoundland and Labrador	CAN
6	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Northwest Territories	Northwest Territories	CAN
7	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Other international	Other international	C internat
8	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Prince Edward Island	Prince Edward Island	CAN
9	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Quebec City	Quebec	CAN

In [6]:
```
freight['TonneKm'].fillna(freight['TonneKm'].mean(), inplace=True)
freight.head(10)
```

Out[6]:

	Year	Mode	Commodity	OrigCMA	OrigProv	OrigCtry	DestCMA	DestProv	Dest
0	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Edmonton	Alberta	CAN
1	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Halifax	Nova Scotia	CAN
2	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Hamilton	Ontario	CAN
3	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Montreal	Quebec	CAN
4	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	New Brunswick	New Brunswick	CAN
5	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Newfoundland and Labrador	Newfoundland and Labrador	CAN
6	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Northwest Territories	Northwest Territories	CAN
7	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Other international	Other international	(internat
8	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Prince Edward Island	Prince Edward Island	CAN
9	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Quebec City	Quebec	CAN

In [7]:
```
freight['Value'].fillna(freight['Value'].mean(), inplace=True)
freight.head(10)
```

Out[7]:

	Year	Mode	Commodity	OrigCMA	OrigProv	OrigCtry	DestCMA	DestProv	Dest
0	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Edmonton	Alberta	CAN
1	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Halifax	Nova Scotia	CAN
2	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Hamilton	Ontario	CAN
3	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Montreal	Quebec	CAN
4	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	New Brunswick	New Brunswick	CAN
5	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Newfoundland and Labrador	Newfoundland and Labrador	CAN
6	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Northwest Territories	Northwest Territories	CAN
7	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Other international	Other international	C internat
8	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Prince Edward Island	Prince Edward Island	CAN
9	2011	Air	Miscellaneous products	Calgary	Alberta	CANADA	Quebec City	Quebec	CAN

Combine

Drop unnnecessary variables

In [8]:
```
freight.drop(['OrigCMA', 'OrigCtry','DestCMA','DestCtry'], axis=1, inplace=Tru
e)
```

In [9]:
```
freight.head()
```

Out[9]:

	Year	Mode	Commodity	OrigProv	DestProv	Shipments	Weight	Revenue	
0	2011	Air	Miscellaneous products	Alberta	Alberta	8479.342545	1.219892e+08	NaN	6.497
1	2011	Air	Miscellaneous products	Alberta	Nova Scotia	311.000000	2.272400e+04	31504.10	1.164
2	2011	Air	Miscellaneous products	Alberta	Ontario	620.000000	3.560913e+06	4846957.66	1.672
3	2011	Air	Miscellaneous products	Alberta	Quebec	2086.000000	2.446410e+05	334898.57	6.270
4	2011	Air	Miscellaneous products	Alberta	New Brunswick	8479.342545	1.219892e+08	NaN	6.497

Clean categorical variables

Fill in missing & create indicator for `Revenue'

In [10]:
```
freight.isnull().sum()
```

Out[10]:
```
Year            0
Mode            0
Commodity       0
OrigProv        0
DestProv        0
Shipments       0
Weight          0
Revenue      1318
Distance        0
TonneKm         0
Value           0
dtype: int64
```

Convert `Mode` to numeric

In [12]:
```
mode_num = {'Air': 1, 'Rail': 2, 'Truck for-hire':3}
freight['Mode'] = freight['Mode'].map(mode_num)
freight.head()
```

Out[12]:

	Year	Mode	Commodity	OrigProv	DestProv	Shipments	Weight	Revenue	
0	2011	1	Miscellaneous products	Alberta	Alberta	8479.342545	1.219892e+08	6.665525e+06	6.4
1	2011	1	Miscellaneous products	Alberta	Nova Scotia	311.000000	2.272400e+04	3.150410e+04	1.1
2	2011	1	Miscellaneous products	Alberta	Ontario	620.000000	3.560913e+06	4.846958e+06	1.6
3	2011	1	Miscellaneous products	Alberta	Quebec	2086.000000	2.446410e+05	3.348986e+05	6.2
4	2011	1	Miscellaneous products	Alberta	New Brunswick	8479.342545	1.219892e+08	6.665525e+06	6.4

In [13]:
```
origprov_num = {'Alberta': 1, 'British Columbia': 2, 'Manitoba':3, 'New Brunswick':4, 'Newfoundland and Labrador':5, 'Northwest Territories':6, 'Nova Scotia':7, 'Nunavut':8, 'Ontario':9, 'Other international':10,'Prince Edward Island':11, 'Quebec':12, 'Saskatchewan':13, 'United States and Mexico':14, 'Yukon': 15}
freight['OrigProv'] = freight['OrigProv'].map(origprov_num)
freight.head()
```

Out[13]:

	Year	Mode	Commodity	OrigProv	DestProv	Shipments	Weight	Revenue	
0	2011	1	Miscellaneous products	1	Alberta	8479.342545	1.219892e+08	6.665525e+06	6.4
1	2011	1	Miscellaneous products	1	Nova Scotia	311.000000	2.272400e+04	3.150410e+04	1.1
2	2011	1	Miscellaneous products	1	Ontario	620.000000	3.560913e+06	4.846958e+06	1.6
3	2011	1	Miscellaneous products	1	Quebec	2086.000000	2.446410e+05	3.348986e+05	6.2
4	2011	1	Miscellaneous products	1	New Brunswick	8479.342545	1.219892e+08	6.665525e+06	6.4

In [14]:
```
destprov_num = {'Alberta': 1, 'British Columbia': 2, 'Manitoba':3, 'New Brunswick':4, 'Newfoundland and Labrador':5, 'Northwest Territories':6, 'Nova Scotia':7, 'Nunavut':8, 'Ontario':9, 'Other international':10,'Prince Edward Island':11, 'Quebec':12, 'Saskatchewan':13, 'United States and Mexico':14, 'Yukon':15}
freight['DestProv'] = freight['DestProv'].map(destprov_num)
freight.head()
```

Out[14]:

	Year	Mode	Commodity	OrigProv	DestProv	Shipments	Weight	Revenue	
0	2011	1	Miscellaneous products	1	1	8479.342545	1.219892e+08	6.665525e+06	6.4
1	2011	1	Miscellaneous products	1	7	311.000000	2.272400e+04	3.150410e+04	1.1
2	2011	1	Miscellaneous products	1	9	620.000000	3.560913e+06	4.846958e+06	1.6
3	2011	1	Miscellaneous products	1	12	2086.000000	2.446410e+05	3.348986e+05	6.2
4	2011	1	Miscellaneous products	1	4	8479.342545	1.219892e+08	6.665525e+06	6.4

In [15]:
```
commodity_num = {'Agricultural products': 1, 'Automobiles and other Transportation Equipment': 2, 'Base metals and Articles of Base metals':3, 'Coal':4, 'Food':5, 'Forest products':6, 'Fuel Oils and crude petroleum':7, 'Minerals':8, 'Miscellaneous products':9, 'Other Manufactured goods':10,'Plastic and Chemical products':11, 'Waste and Scrap':12}
freight['Commodity'] = freight['Commodity'].map(commodity_num)
freight.head()
```

Out[15]:

	Year	Mode	Commodity	OrigProv	DestProv	Shipments	Weight	Revenue	
0	2011	1	9	1	1	8479.342545	1.219892e+08	6.665525e+06	6.497
1	2011	1	9	1	7	311.000000	2.272400e+04	3.150410e+04	1.164
2	2011	1	9	1	9	620.000000	3.560913e+06	4.846958e+06	1.672
3	2011	1	9	1	12	2086.000000	2.446410e+05	3.348986e+05	6.270
4	2011	1	9	1	4	8479.342545	1.219892e+08	6.665525e+06	6.497

Drop unnecessary variables

In [16]:
```
freight.drop(['Revenue'], axis=1, inplace=True)
```

Write out cleaned data

In [17]:
```
freight.to_csv('../../../freight_cleaned.csv', index=False)
```

In [18]:
```
import pandas as pd
from sklearn.model_selection import train_test_split

freight = pd.read_csv('../../../freight_cleaned.csv')
freight.head()
```

Out[18]:

	Year	Mode	Commodity	OrigProv	DestProv	Shipments	Weight	Distance	T
0	2011	1	9	1	1	8479.342545	1.219892e+08	6.497016e+06	9.254
1	2011	1	9	1	7	311.000000	2.272400e+04	1.164384e+06	8.507
2	2011	1	9	1	9	620.000000	3.560913e+06	1.672140e+06	9.603
3	2011	1	9	1	12	2086.000000	2.446410e+05	6.270484e+06	7.351
4	2011	1	9	1	4	8479.342545	1.219892e+08	6.497016e+06	9.254

In [19]:
```
ffeatures = freight.drop('Revenue_ind', axis=1)
flabels = freight['Revenue_ind']

X_train, X_test, y_train, y_test = train_test_split(ffeatures, flabels, test_s
ize=0.4, random_state=42)
X_val, X_test, y_val, y_test = train_test_split(X_test, y_test, test_size=0.5,
random_state=42)
```

In [20]:
```
for dataset in [y_train, y_val, y_test]:
    print(round(len(dataset) / len(flabels), 2))
```

```
0.6
0.2
0.2
```

In [21]:
```
X_train.to_csv('../../../train_ffeatures.csv', index=False)
X_val.to_csv('../../../val_ffeatures.csv', index=False)
X_test.to_csv('../../../test_ffeatures.csv', index=False)

y_train.to_csv('../../../train_flabels.csv', index=False)
y_val.to_csv('../../../val_flabels.csv', index=False)
y_test.to_csv('../../../test_flabels.csv', index=False)
```

In [22]:
```
import joblib
import pandas as pd
from sklearn.model_selection import GridSearchCV
from sklearn.neural_network import MLPClassifier
import warnings
warnings.filterwarnings('ignore', category=FutureWarning)
warnings.filterwarnings('ignore', category=DeprecationWarning)

tr_features = pd.read_csv('../../../train_ffeatures.csv')
tr_labels = pd.read_csv('../../../train_flabels.csv')
```

In [23]:
```
def print_results(results):
    print('BEST PARAMS: {}\n'.format(results.best_params_))

    means = results.cv_results_['mean_test_score']
    stds = results.cv_results_['std_test_score']
    for mean, std, params in zip(means, stds, results.cv_results_['params']):
        print('{} (+/-{}) for {}'.format(round(mean, 3), round(std * 2, 3), pa
rams))
```

In [24]:
```
mlp = MLPClassifier()
parameters = {
    'hidden_layer_sizes': [(10,), (50,), (100,)],
    'activation': ['relu', 'tanh', 'logistic'],
    'learning_rate': ['constant', 'invscaling', 'adaptive']
}

cv = GridSearchCV(mlp, parameters, cv=5)
cv.fit(tr_features, tr_labels.values.ravel())

print_results(cv)
```

```
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
```

```
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
```

```
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
```

```
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
```

```
BEST PARAMS: {'activation': 'relu', 'hidden_layer_sizes': (100,), 'learning_r
ate': 'invscaling'}

0.892 (+/-0.113) for {'activation': 'relu', 'hidden_layer_sizes': (10,), 'lea
rning_rate': 'constant'}
0.916 (+/-0.08) for {'activation': 'relu', 'hidden_layer_sizes': (10,), 'lear
ning_rate': 'invscaling'}
0.911 (+/-0.08) for {'activation': 'relu', 'hidden_layer_sizes': (10,), 'lear
ning_rate': 'adaptive'}
0.929 (+/-0.023) for {'activation': 'relu', 'hidden_layer_sizes': (50,), 'lea
rning_rate': 'constant'}
0.936 (+/-0.018) for {'activation': 'relu', 'hidden_layer_sizes': (50,), 'lea
rning_rate': 'invscaling'}
0.936 (+/-0.029) for {'activation': 'relu', 'hidden_layer_sizes': (50,), 'lea
rning_rate': 'adaptive'}
0.911 (+/-0.045) for {'activation': 'relu', 'hidden_layer_sizes': (100,), 'le
arning_rate': 'constant'}
0.942 (+/-0.02) for {'activation': 'relu', 'hidden_layer_sizes': (100,), 'lea
rning_rate': 'invscaling'}
0.933 (+/-0.032) for {'activation': 'relu', 'hidden_layer_sizes': (100,), 'le
arning_rate': 'adaptive'}
0.755 (+/-0.012) for {'activation': 'tanh', 'hidden_layer_sizes': (10,), 'lea
rning_rate': 'constant'}
0.763 (+/-0.021) for {'activation': 'tanh', 'hidden_layer_sizes': (10,), 'lea
rning_rate': 'invscaling'}
0.757 (+/-0.028) for {'activation': 'tanh', 'hidden_layer_sizes': (10,), 'lea
rning_rate': 'adaptive'}
0.771 (+/-0.005) for {'activation': 'tanh', 'hidden_layer_sizes': (50,), 'lea
rning_rate': 'constant'}
0.775 (+/-0.014) for {'activation': 'tanh', 'hidden_layer_sizes': (50,), 'lea
rning_rate': 'invscaling'}
0.774 (+/-0.011) for {'activation': 'tanh', 'hidden_layer_sizes': (50,), 'lea
rning_rate': 'adaptive'}
0.779 (+/-0.014) for {'activation': 'tanh', 'hidden_layer_sizes': (100,), 'le
arning_rate': 'constant'}
0.778 (+/-0.008) for {'activation': 'tanh', 'hidden_layer_sizes': (100,), 'le
arning_rate': 'invscaling'}
0.778 (+/-0.013) for {'activation': 'tanh', 'hidden_layer_sizes': (100,), 'le
arning_rate': 'adaptive'}
0.696 (+/-0.078) for {'activation': 'logistic', 'hidden_layer_sizes': (10,),
'learning_rate': 'constant'}
0.717 (+/-0.126) for {'activation': 'logistic', 'hidden_layer_sizes': (10,),
'learning_rate': 'invscaling'}
0.697 (+/-0.068) for {'activation': 'logistic', 'hidden_layer_sizes': (10,),
'learning_rate': 'adaptive'}
0.753 (+/-0.023) for {'activation': 'logistic', 'hidden_layer_sizes': (50,),
'learning_rate': 'constant'}
0.772 (+/-0.01) for {'activation': 'logistic', 'hidden_layer_sizes': (50,),
'learning_rate': 'invscaling'}
0.765 (+/-0.018) for {'activation': 'logistic', 'hidden_layer_sizes': (50,),
'learning_rate': 'adaptive'}
0.761 (+/-0.05) for {'activation': 'logistic', 'hidden_layer_sizes': (100,),
'learning_rate': 'constant'}
0.775 (+/-0.016) for {'activation': 'logistic', 'hidden_layer_sizes': (100,),
'learning_rate': 'invscaling'}
0.772 (+/-0.018) for {'activation': 'logistic', 'hidden_layer_sizes': (100,),
'learning_rate': 'adaptive'}
```

```
C:\Users\purit\anaconda3\lib\site-packages\sklearn\neural_network\_multilayer
_perceptron.py:571: ConvergenceWarning: Stochastic Optimizer: Maximum iterati
ons (200) reached and the optimization hasn't converged yet.
  % self.max_iter, ConvergenceWarning)
```

Check if there's any null value.

In [25]:
```
freight.isnull().sum()
```

Out[25]:
```
Year           0
Mode           0
Commodity      0
OrigProv       0
DestProv       0
Shipments      0
Weight         0
Distance       0
TonneKm        0
Value          0
Revenue_ind    0
dtype: int64
```

In [26]:
```
cv.best_estimator_
```

Out[26]:
```
MLPClassifier(activation='relu', alpha=0.0001, batch_size='auto', beta_1=0.9,
              beta_2=0.999, early_stopping=False, epsilon=1e-08,
              hidden_layer_sizes=(100,), learning_rate='invscaling',
              learning_rate_init=0.001, max_fun=15000, max_iter=200,
              momentum=0.9, n_iter_no_change=10, nesterovs_momentum=True,
              power_t=0.5, random_state=None, shuffle=True, solver='adam',
              tol=0.0001, validation_fraction=0.1, verbose=False,
              warm_start=False)
```

I have identified the best estimator as above, which gives me 94.2% accuracy.

In [27]:
```
joblib.dump(cv.best_estimator_, '../../../MLP_model.pkl')
```

Out[27]:
```
['../../../MLP_model.pkl']
```