

t2yjsw0lc

February 6, 2025

```
[19]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, classification_report, \
    confusion_matrix
from sklearn.ensemble import AdaBoostClassifier
import warnings
warnings.filterwarnings('ignore')
```

```
[22]: #load the data
data = pd.read_csv(r"C:\Users\91703\Downloads\accident.csv")
```

```
[23]: data.head()
```

```
[23]:
```

	Age	Gender	Speed_of_Impact	Helmet_Used	Seatbelt_Used	Survived
0	56	Female	27.0	No	No	1
1	69	Female	46.0	No	Yes	1
2	46	Male	46.0	Yes	Yes	0
3	32	Male	117.0	No	Yes	0
4	60	Female	40.0	Yes	Yes	0

```
[24]: data.tail()
```

```
[24]:
```

	Age	Gender	Speed_of_Impact	Helmet_Used	Seatbelt_Used	Survived
195	69	Female	111.0	No	Yes	1
196	30	Female	51.0	No	Yes	1
197	58	Male	110.0	No	Yes	1
198	20	Male	103.0	No	Yes	1
199	56	Female	43.0	No	Yes	1

```
[25]: data.shape
```

```
[25]: (200, 6)
```

```
[8]: data.isnull().sum()
```

```
[8]: Age                0
     Gender             1
     Speed_of_Impact    3
     Helmet_Used        0
     Seatbelt_Used      0
     Survived           0
     dtype: int64
```

```
[26]: data['Speed_of_Impact'] = data['Speed_of_Impact'].
      ↪ fillna(data['Speed_of_Impact'].mean())
```

```
[27]: data['Gender'] = data['Gender'].fillna(data['Gender'].mode()[0])
```

```
[28]: data.isnull().sum()
```

```
[28]: Age                0
     Gender             0
     Speed_of_Impact    0
     Helmet_Used        0
     Seatbelt_Used      0
     Survived           0
     dtype: int64
```

```
[29]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 6 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Age                   200 non-null   int64
1   Gender                200 non-null   object
2   Speed_of_Impact       200 non-null   float64
3   Helmet_Used           200 non-null   object
4   Seatbelt_Used         200 non-null   object
5   Survived              200 non-null   int64
dtypes: float64(1), int64(2), object(3)
memory usage: 9.5+ KB
```

```
[30]: le = LabelEncoder()

data['Gender'] = le.fit_transform(data['Gender'])
data['Helmet_Used'] = le.fit_transform(data['Helmet_Used']).map({'Yes':1, 'No':
      ↪ 0}))
```

```
data['Seatbelt_Used'] = le.fit_transform(data['Seatbelt_Used'].map({'Yes':
↪1, 'No':0}))
data['Survived'] = le.fit_transform(data['Survived'].map({'Yes':1, 'No':0}))
```

```
[31]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 6 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Age                   200 non-null   int64
1   Gender                200 non-null   int32
2   Speed_of_Impact       200 non-null   float64
3   Helmet_Used           200 non-null   int64
4   Seatbelt_Used         200 non-null   int64
5   Survived              200 non-null   int64
dtypes: float64(1), int32(1), int64(4)
memory usage: 8.7 KB
```

```
[40]: scaler = StandardScaler()
data_scaled = scaler.fit_transform(data)
```

```
[38]: data[['Age', 'Speed_of_Impact']] = scaler.fit_transform(data[['Age',
↪'Speed_of_Impact']])
```

```
[41]: data.head()
```

```
[41]:
```

	Age	Gender	Speed_of_Impact	Helmet_Used	Seatbelt_Used	Survived
0	0.843704	0	-1.456672	0	0	0
1	1.715924	0	-0.819569	0	1	0
2	0.172767	1	-0.819569	1	1	0
3	-0.766547	1	1.561182	0	1	0
4	1.112080	0	-1.020760	1	1	0

```
[44]: data_scaled = pd.DataFrame(data_scaled, columns = data.columns)

X = data_scaled.drop('Survived', axis=1)
y = data_scaled['Survived']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
↪random_state=42)
```

```
[46]: X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

```
[47]: adaboost_model = AdaBoostClassifier(n_estimators=100, random_state=42)
adaboost_model.fit(X_train_scaled, y_train)
```

```
[47]: AdaBoostClassifier(n_estimators=100, random_state=42)
```

```
[49]: X_test_scaled
y_pred = adaboost_model.predict(X_test_scaled)
print(y_pred)
```

```
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
```

```
[50]: print("Accuracy:", accuracy_score(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
```

Accuracy: 1.0

Classification Report:

	precision	recall	f1-score	support
0.0	1.00	1.00	1.00	40
accuracy			1.00	40
macro avg	1.00	1.00	1.00	40
weighted avg	1.00	1.00	1.00	40

Confusion Matrix:

```
[[40]]
```

```
[51]: plt.figure(figsize=(8, 6))
sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, cmap='Blues')
plt.xlabel("Predicted labels")
plt.ylabel("True labels")
plt.show()
```

