In [4]:
```
import numpy as np
import matplotlib.pyplot as plt
from sklearn import datasets
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix, classification_repor
```

In [40]:
```
import pandas as pd
data = pd.read_csv(r"C:\Users\bharath\OneDrive\Desktop\fish 2.csv")
```

In [41]:
```
data.head(10)
```

Out[41]:

	nofish	livebait	camper	persons	child	xb	zg	count
0	1	0	0	1	0	-0.896315	3.050405	0
1	0	1	1	1	0	-0.558345	1.746149	0
2	0	1	0	1	0	-0.401731	0.279939	0
3	0	1	1	2	1	-0.956298	-0.601526	0
4	0	1	0	1	0	0.436891	0.527709	1
5	0	1	1	4	2	1.394485	-0.707535	0
6	0	1	0	3	1	0.184717	-3.398022	0
7	0	1	0	4	3	2.329107	-5.450902	0
8	1	0	1	3	2	0.188386	-1.527418	0
9	0	1	1	1	0	0.287690	1.393890	1

In [54]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score
X = data[['nofish', 'livebait', 'camper']]
y = data['persons']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_st

scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
svm_model = SVC()
svm_model.fit(X_train, y_train)
y_pred = svm_model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy of the SVM model: {accuracy}")
```

```
Accuracy of the SVM model: 0.2
```

In [56]:

```
from sklearn.datasets import load_iris
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

data = load_iris()
X = data.data
y = data.target
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_st

clf = RandomForestClassifier(n_estimators=100, random_state=42)
clf.fit(X_train, y_train)

predictions = clf.predict(X_test)

accuracy = accuracy_score(y_test, predictions)
print(f"Accuracy of Random Forest Classifier: {accuracy}")
```

```
Accuracy of Random Forest Classifier: 1.0
```

In [60]:

```
from sklearn.decomposition import PCA
import numpy as np
data = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
pca = PCA(n_components=2)
transformed_data = pca.fit_transform(data)
print(transformed_data)
```

```
[[-5.19615242e+00  3.62353582e-16]
 [ 0.00000000e+00  0.00000000e+00]
 [ 5.19615242e+00  3.62353582e-16]]
```

In [66]:

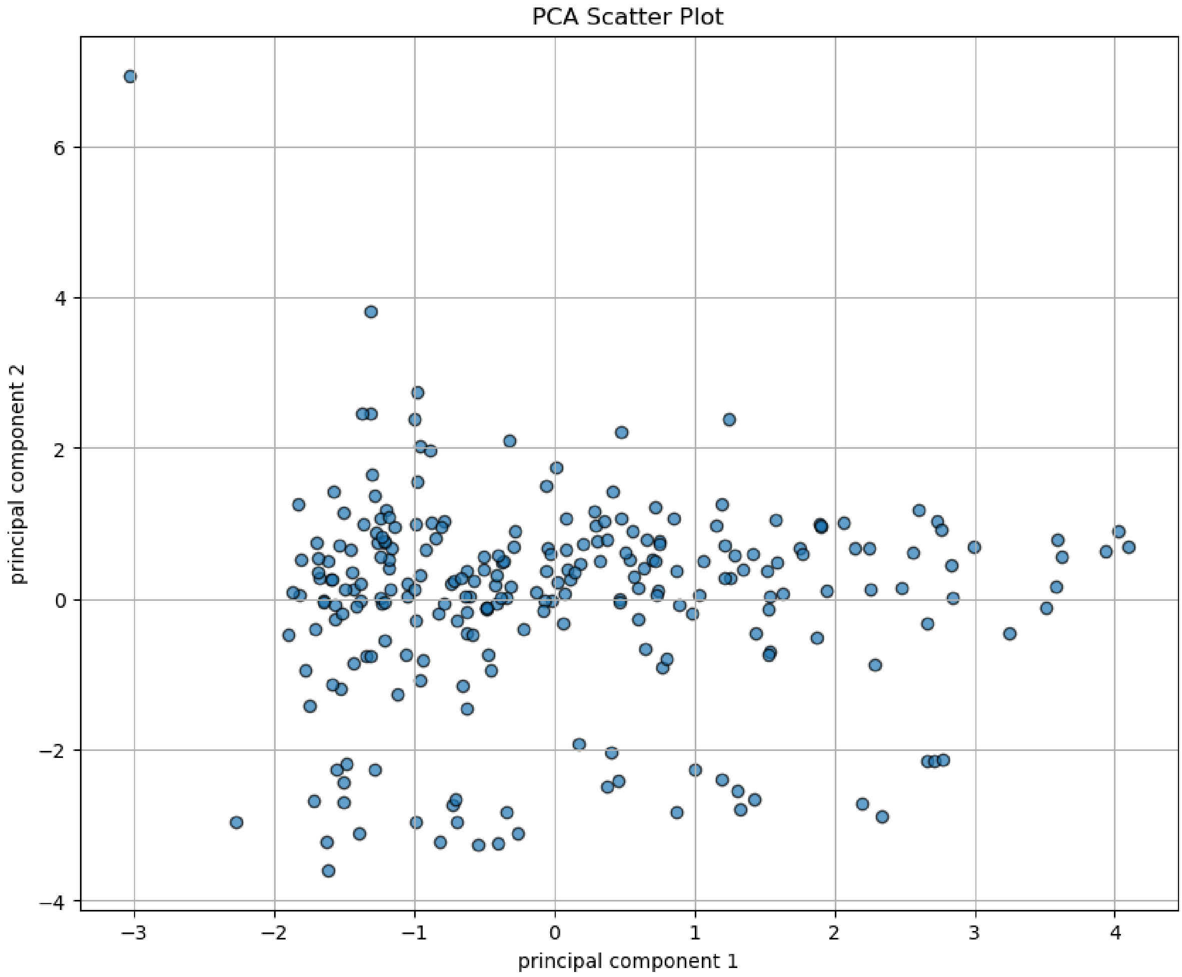
```
import pandas as pd
import numpy as np
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
```

In [69]:

```
data = data = pd.read_csv(r"C:\Users\bharath\OneDrive\Desktop\fish 2.csv")
X = data.drop('persons', axis=1)
X_standardized = (X - X.mean()) / X.std()
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_standardized)
```

In [73]:

```
pca_df = pd.DataFrame(data=X_pca, columns=['PC1', 'PC2'])
plt.figure(figsize=(10, 8))
plt.scatter(pca_df['PC1'], pca_df['PC2'], edgecolors='k', alpha=0.7)
plt.title('PCA Scatter Plot')
plt.xlabel("principal component 1")
plt.ylabel("principal component 2")
plt.grid(True)
plt.show()
```



In [74]:

```
data.head(10)
```

Out[74]:

	nofish	livebait	camper	persons	child	xb	zg	count
0	1	0	0	1	0	-0.896315	3.050405	0
1	0	1	1	1	0	-0.558345	1.746149	0
2	0	1	0	1	0	-0.401731	0.279939	0
3	0	1	1	2	1	-0.956298	-0.601526	0
4	0	1	0	1	0	0.436891	0.527709	1
5	0	1	1	4	2	1.394485	-0.707535	0
6	0	1	0	3	1	0.184717	-3.398022	0
7	0	1	0	4	3	2.329107	-5.450902	0
8	1	0	1	3	2	0.188386	-1.527418	0
9	0	1	1	1	0	0.287690	1.393890	1

In [ ]: