Data Mining
Project - Titanic

Dr. Shahrzad Haddadan

Karson Cheng, Kai-Yin Huang, Pai Peng
Ken Lau, Garrett Tai, Feng Peiyuan

Table of contents

Master of Business and Science
RUTGERS

Introduction: Objective

Build some predictive models that answers the question: “what sorts of people were more likely to survive?” using passenger data like name, age, gender, etc.

And we will compare each supervised and unsupervised method to see which one performs better.

Introduction: How does it looks like?

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  891 non-null    int64
 1   Survived     891 non-null    int64
 2   Pclass       891 non-null    int64
 3   Name         891 non-null    object
 4   Sex          891 non-null    object
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64
 7   Parch        891 non-null    int64
 8   Ticket       891 non-null    object
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object
 11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

Total Entries: There are 891 entries in the dataset

Attributes:

- **PassengerId:** A unique identifier for each passenger
- **Survived:** Indicates whether the passenger survived (1) or not (0).
- **Pclass:** The class of the ticket the passenger purchased (1st, 2nd, or 3rd).
- **Name:** The name of the passenger.
- Sex: The gender of the passenger.
- **Age:** The age of the passenger.
- SibSp: The number of siblings or spouses the passenger had aboard the Titanic.
- **Parch:** The number of parents or children the passenger had aboard the Titanic.
- **Ticket:** The ticket number.
- **Fare:** The amount of money spent on the ticket.
- **Cabin:** The cabin number where the passenger stayed.
- **Embarked:** The port where the passenger embarked the Titanic (C = Cherbourg; Q = Queenstown; S = Southampton).

The dependent attribute (the one we would want to predict) is typically Survived since it indicates whether a passenger lived or died, which is a common target for classification models.

Master of Business and Science RUTGERS

Introduction:Statistics Results

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

Data Preprocessing

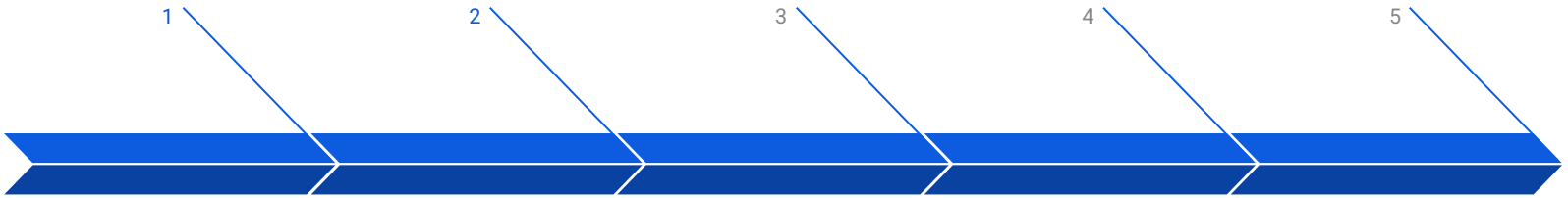
- Index: PassengerId
- Numerical (Continuous) variables: Age, Fare
- Binary Variables: Survived
- String Variable: Name
- Categorical Variables: PClass, Sex, ...

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

Data Preprocessing

- Checked for prediction class imbalance
- Dropped duplicated rows
- Dropped columns: Name, Ticket, Cabin
- Transformed Sex into Binary Variable
- Applied One-Hot-Encoding to Embarked
- Filled Age with median
- Normalized and mapped numerical variables into [-1, 1]
- Train/Test Split in 8:2 ratio

Classification Model 1 - Support Vector Machine(SVM)




```
# Create the SVM model
svm_model1 = SVC(kernel='linear', random_state=seed)
svm_model2 = SVC(kernel='rbf', gamma='scale', random_state=seed)

# Train the model
svm_model1.fit(X_train, y_train)
svm_model2.fit(X_train, y_train)

# Make predictions on the test set
y_pred1 = svm_model1.predict(X_test)
y_pred2 = svm_model2.predict(X_test)

# Accuracy
accuracy1 = sklearn.metrics.accuracy_score(y_test, y_pred1)
accuracy2 = sklearn.metrics.accuracy_score(y_test, y_pred2)
print("Linear SVM Accuracy:", accuracy1)
print("Nonlinear SVM Accuracy:", accuracy2)
```

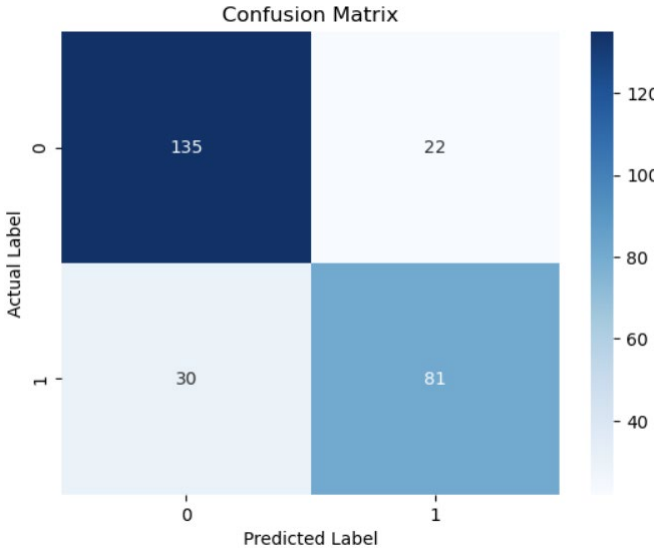
```
Linear SVM Accuracy: 0.7877094972067039
Nonlinear SVM Accuracy: 0.8044692737430168
```

Linear SVM Accuracy: 0.7877

Nonlinear SVM Accuracy: 0.8044

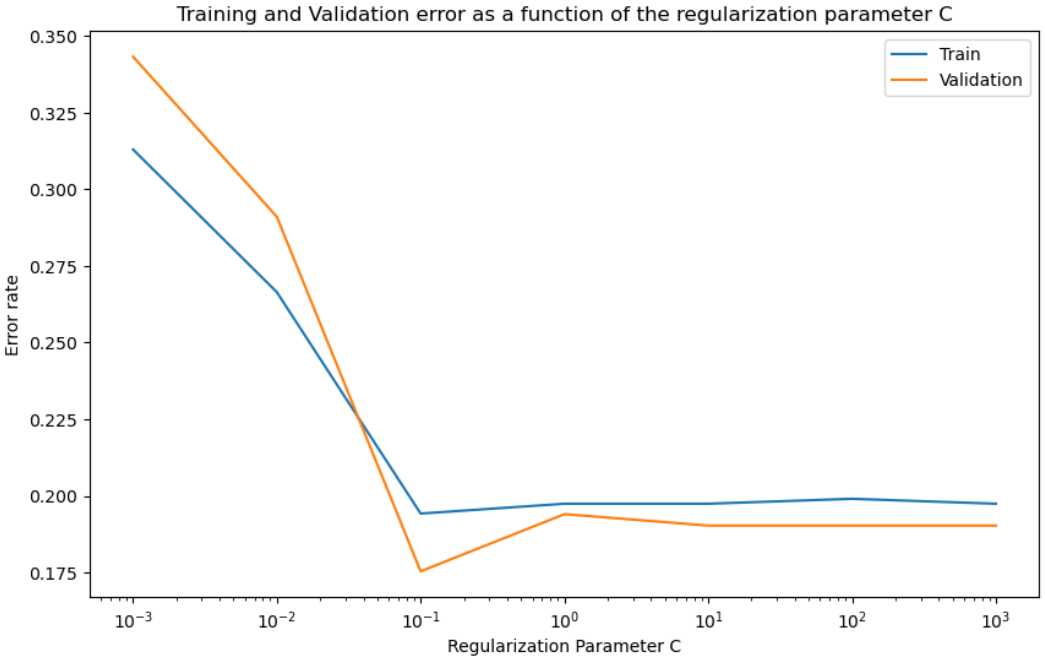
Master of Business and Science
RUTGERS

Classification Model 2 - Logistics LogisticRegression



- Training score:0.8047752808988764
- Test score: 0.8044692737430168
- Accuracy: 0.8044692737430168
- Sex had the most substantial effect on survival predictions

Classification Model 2 - Logistics LogisticRegression



- Best parameters: {'C': 1, 'penalty': 'l1'}
- Best cross-validated score: 0.81
- Test set score with best parameters: 0.80

Master of Business and Science
RUTGERS

Classification Model 3 - Gradient Tree Boosting (xgboost)

```
params = {
  'n_estimators': [100, 1000],
  'learning_rate':  [0.01, 0.1, 0.05],
  'max_depth': [4, 5, 6],
  'colsample_bytree':[0.8, 1],
#   'gamma':[0],
  # 'scale_pos_weight':[scale],
}

gbc = xgb.XGBClassifier(
    # tree_method='auto', device = "cuda",
    random_state=seed,
    early_stopping_rounds=42,
    eval_metric='error',
    objective='binary:logistic',
)

k_fold = KFold(n_splits=5, shuffle=True, random_state=seed)

grid_search = GridSearchCV(
    estimator=gbc,
    param_grid=params,
    scoring='accuracy',
    cv = k_fold,
    verbose=1,
    n_jobs=4,
)
```

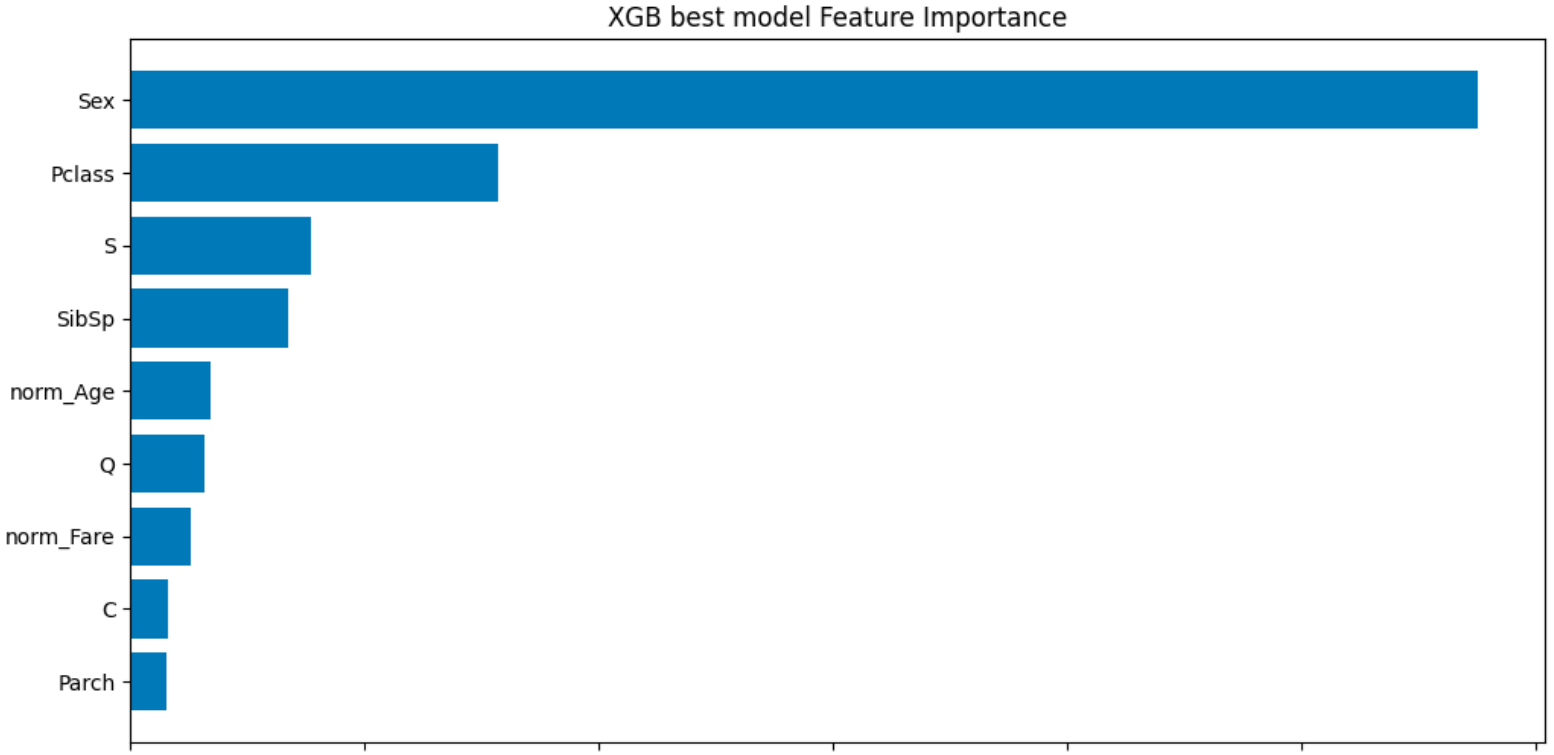
- Adopted GridSearchCV and KFold in model fitting and hyperparameters tuning.
- Prediction error used in eval_metric for Grid Search Early Stopping and accuracy score for scoring for Grid Search

Master of Business and Science
RUTGERS

Classification Model 3 - Gradient Tree Boosting (xgboost)

```
The best Parameters are {'colsample_bytree': 1, 'learning_rate': 0.05, 'max_depth': 5, 'n_estimators': 100}
In-sample Accuracy: 0.8806

          True      False
True   0.612360  0.023876
False  0.095506  0.268258

Out-of-sample Accuracy: 0.8212

          True      False
True   0.513966  0.022346
False  0.156425  0.307263
```

Classification Model 3 - Gradient Tree Boosting (xgboost)



Master of Business and Science
RUTGERS

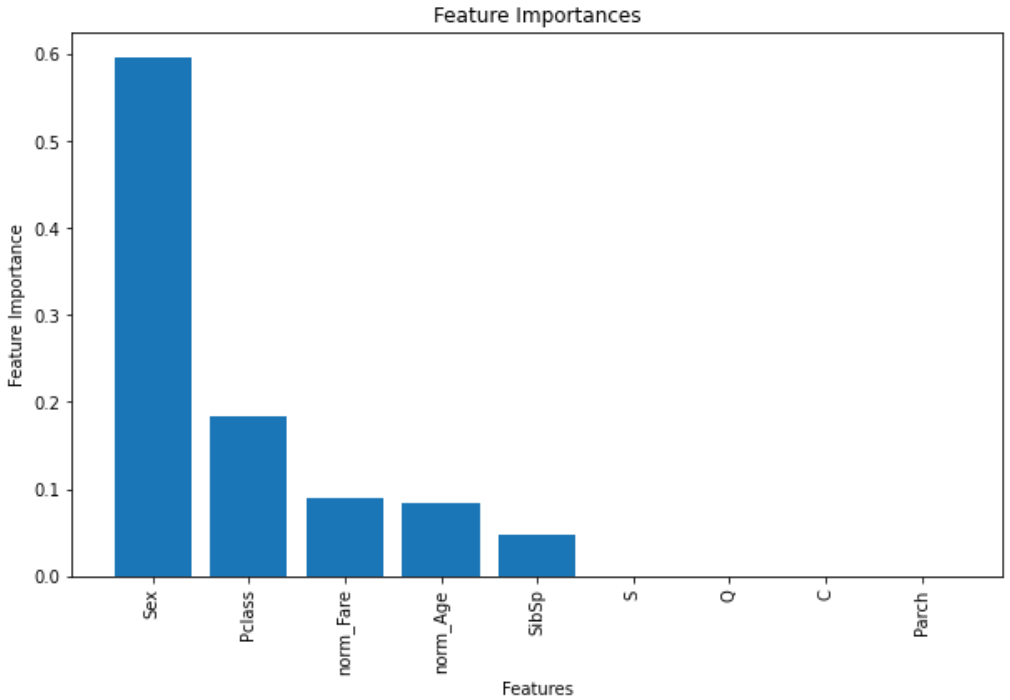
Classification Model 4
Decision Tree

```
# Define the hyperparameter grid
param_grid = {
    'criterion': ['gini'],
    'max_depth': [2, 3, 4, 5, 6, 7, 8, 9, 10],
    'min_samples_split': [2, 3, 4, 5, 6],
    'min_samples_leaf': [1, 2, 3, 4, 5],
    'max_features': [None],
    'splitter': ['best']
}

# Instantiate the DecisionTreeClassifier
dt = DecisionTreeClassifier(random_state=seed)


for k in range(2, 21):
    print(k, '-fold cross validation:')

    # Instantiate GridSearchCV
    grid_search = GridSearchCV(estimator=dt, param_grid=param_grid, cv=k, scoring='accuracy')
```

Classification Model 4
Decision Tree

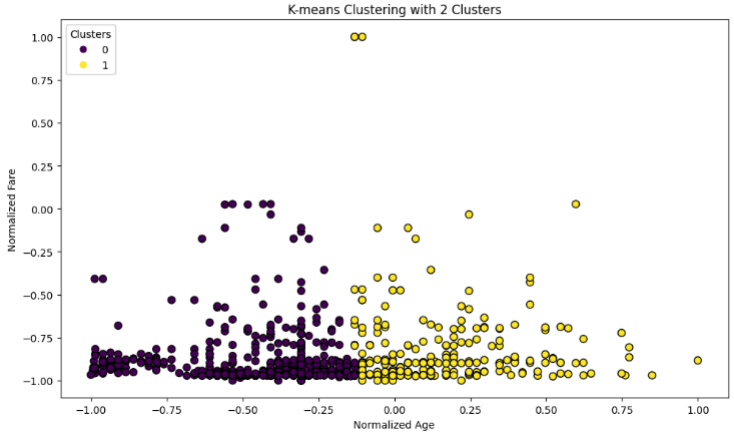
```
5-fold cross validation:
Best Hyperparameters: {'criterion': 'gini', 'max_depth': 3, 'max_features': None, 'min_samples_leaf': 3, 'min_samples_split': 2, 'splitter': 'best'}

Accuracy on Training Set: 83.15

Accuracy on Test Set: 80.97
```

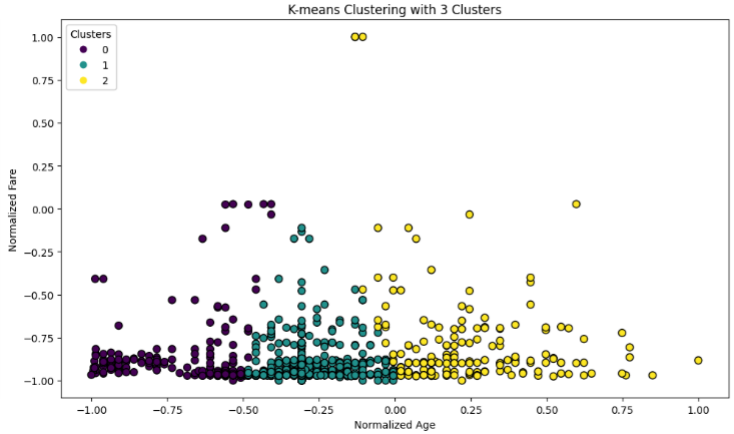
Classification Model 4
Decision Tree

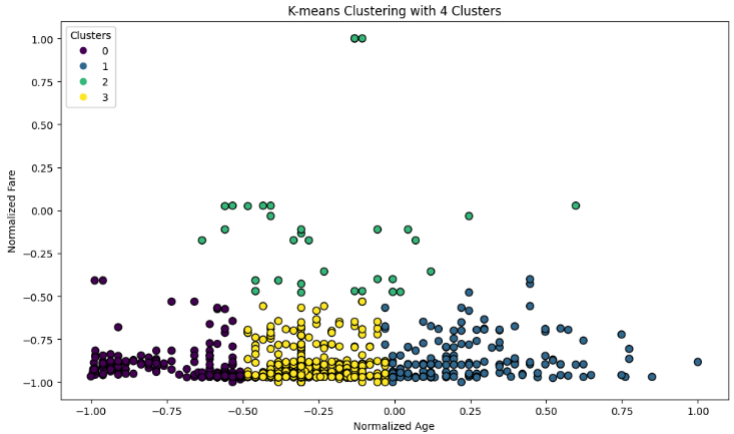


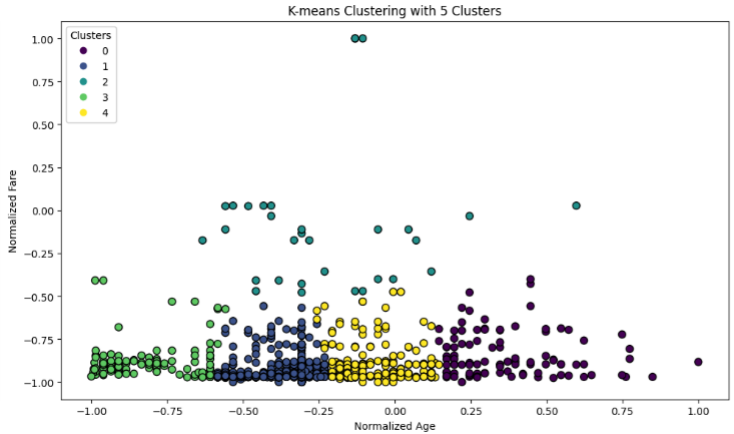
Master of Business and Science
RUTGERS

Clustering: K-means









Master of Business and Science
RUTGERS

Clustering: K-means

```
Crosstab with 2 clusters:
Survived     0    1
Cluster_2
0          415  252
1          134   90

Crosstab with 3 clusters:
Survived     0    1
Cluster_3
0          104   87
1          347  185
2           98   70

Crosstab with 4 clusters:
Survived     0    1
Cluster_4
0           96   79
1          101   58
2            9   24
3          343  181

Crosstab with 5 clusters:
Survived     0    1
Cluster_5
0           61   37
1          324  154
2            9   22
3           46   57
4          109   72
```

Are classes well separated into individual clusters?

True

Are some classes more cohesive and some less?

True

Do some classes correspond to several clusters?

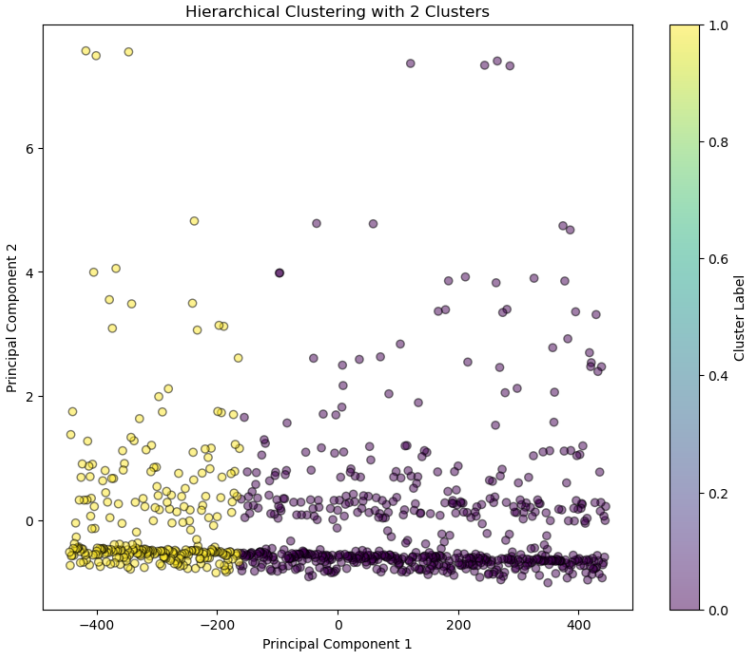
True

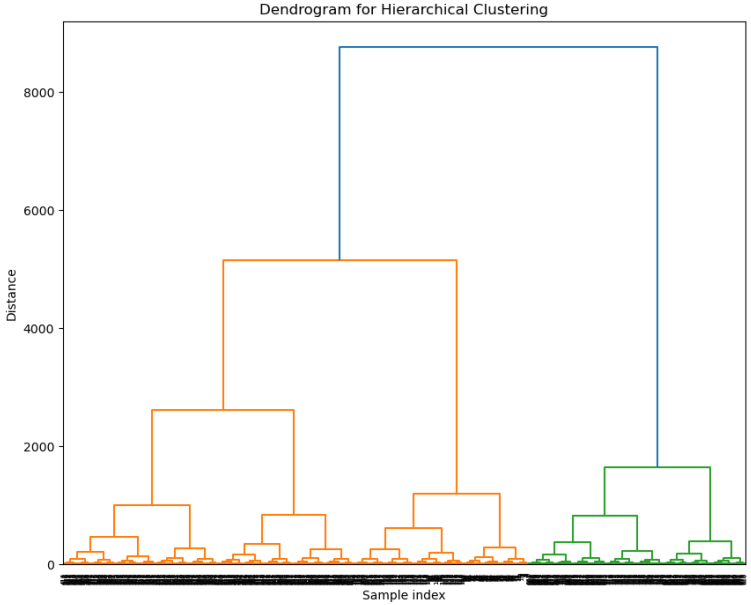
Are the class labels completely irrelevant to the clusters you see?

False

Master of Business and Science
RUTGERS

Clustering:Hierarchical clustering





Master of Business and Science
RUTGERS

Clustering: Hierarchical Summary

```
Cross-tabulation of labels and clusters:
 Clusters    0    1
Labels
0         369  180
1         239  103
Are classes well separated into individual clusters? True
Are some classes more cohesive and some less? True
Do some classes correspond to several clusters? True
Are the class labels completely irrelevant to the clusters you see? False
```

Clustering Summary

Cluster Separation

The classes seem to be reasonably well-separated into individual clusters, as indicated by the cross-tabulation. However, there is some overlap, particularly with Cluster 1 containing a mix of both classes.

Cohesion Variation

There appears to be variation in the cohesion of the classes within the clusters. Some classes are more cohesive than others, as evidenced by the difference in cluster sizes and the distribution of class labels within clusters.

Class-Correspondence to Clusters

Some classes correspond to several clusters, this could be due to overlapping features or insufficient feature representation in the clustering process.

Relevance of Class Labels

The class labels are not entirely irrelevant to the clusters observed. While there is some overlap and variation in cohesion, there is still a discernible relationship between the original class labels and the clusters formed.

Master of Business and Science
RUTGERS

Model Summary I

SVM

Advantages:

- high-dimensional data
- Find the maximum bounds of different features

Disadvantages:

- Sensitive to large datasets and noisy data
- Long training time

Gradient Boosting Trees

Advantages:

- Handling Missing Values
- Regression and Classification
- Intuitively explanation

Disadvantages:

- Long training time
- The Risk of Overfitting

Model Summary II

Decision tree

Advantages:

- EZ to understand
- Good visualization for decision-making process
- Less data preprocessing

Disadvantages:

- Overfitting
- Sensitive to data changes

Logistic regression

Advantages:

- Fast training speed
- Provide probabilities of predicted class membership

Disadvantages:

- Sensitive to feature selection
- Linear classifier

Conclusion

Best model: **XGboost**

Accuracy: **82.12%**

Key Influencing Factors

1. **Gender**: woman has higher survival rate
2.**Pclass**: survival rate is higher in first class
3.**Age**: child and elder will have high priority to evacuate
4.**Fare**: high price ticket has higher survival rate
5.**Embarked**: different boarding locations will affect the survival rate.

Thanks!

Do you have any questions?