Naive Bayes Classifier

Objective: To build Naïve Bayes classifier model on 'Titanic' dataset as part of Lab Migration Project.

Methods:

(i) Import and load the dataset and view information about it using str() function.
(ii) Check for any missing values in the dataset and clean it.
(iii) Check the independence of attributes in the dataset by creating pair plots.
(iv) Split the data into training and testing set.
(v) Build Naïve Bayes model using naive_bayes () function.
(vi) Make predictions and check model accuracy.
(vii) Conclusion

```
#To clear the environment
rm(list=ls())

#Import the required libraries
library(naivebayes)

## naivebayes 0.9.7 loaded

library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(ggplot2)
library(psych)

##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
## 
##     %+%, alpha
```

```
#Import and load the dataset
data <- read.csv('titanic.csv')
str(data)
```

```
## 'data.frame':    891 obs. of  12 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley
(Florence Briggs Thayer)" "Heikkinen, Miss. Laina" "Futrelle, Mrs. Jacques
Heath (Lily May Peel)" ...
##  $ Sex        : chr  "male" "female" "female" "female" ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : chr  "A/5 21171" "PC 17599" "STON/O2. 3101282" "113803"
...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : chr  "" "C85" "" "C123" ...
##  $ Embarked   : chr  "S" "C" "S" "S" ...
```

```
#Check for missing values in dataset
sum(is.na(data))
```

```
## [1] 177
```

```
#Cleaning NA values
data_clean <- na.omit(data)
sum(is.na(data_clean))
```

```
## [1] 0
```

Inference: We checked for any missing values in the given dataset, and found that there are 177 NA values. So, in order to clean these NA values, we used na.omit() function and removed them.

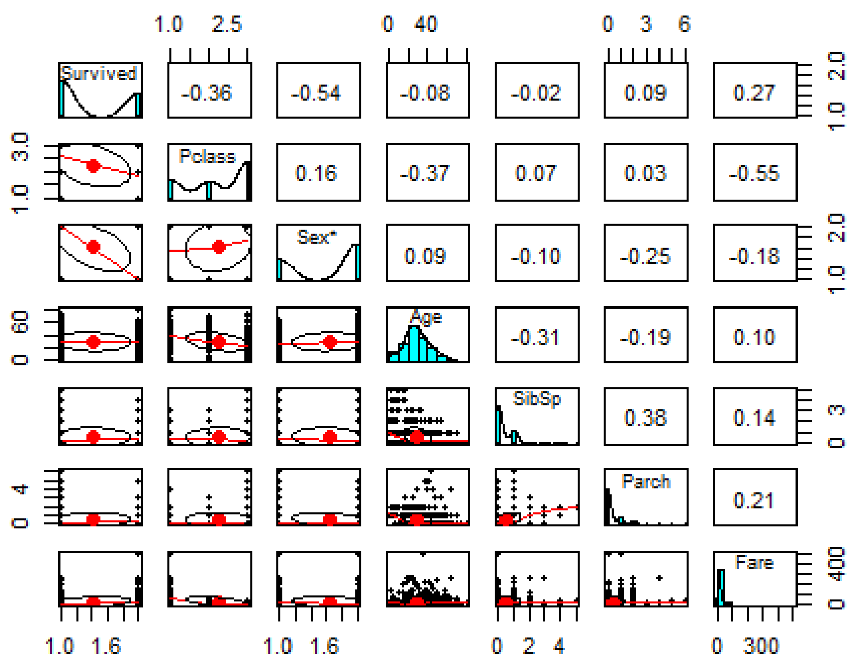
```
#To convert int in 'Survived' column to factor
data_clean$Survived <- as.factor(data_clean$Survived)

#To convert int in 'Pclass' column to factor
data_clean$Pclass <- as.factor(data_clean$Pclass)
data_clean <- select(data_clean,-c(PassengerId,Name,Ticket,Cabin,Embarked))
str(data_clean)
```

```
## 'data.frame':    714 obs. of  7 variables:
##  $ Survived: Factor w/ 2 levels "0","1": 1 2 2 2 1 1 1 2 2 2 ...
##  $ Pclass  : Factor w/ 3 levels "1","2","3": 3 1 3 1 3 1 3 3 2 3 ...
```

```
##  $ Sex     : chr  "male" "female" "female" "female" ...
##  $ Age     : num  22 38 26 35 35 54 2 27 14 4 ...
##  $ SibSp   : int  1 1 0 1 0 0 3 0 1 1 ...
##  $ Parch   : int  0 0 0 0 0 0 1 2 0 1 ...
##  $ Fare    : num  7.25 71.28 7.92 53.1 8.05 ...
##  - attr(*, "na.action")= 'omit' Named int [1:177] 6 18 20 27 29 30 32 33
37 43 ...
##   ..- attr(*, "names")= chr [1:177] "6" "18" "20" "27" …
```

Inference: Feature selection is carried out where the unwanted columns like PassengerId, Name, Ticket, Cabin and Embarked are removed from the clean dataset, and only the required ones are kept.

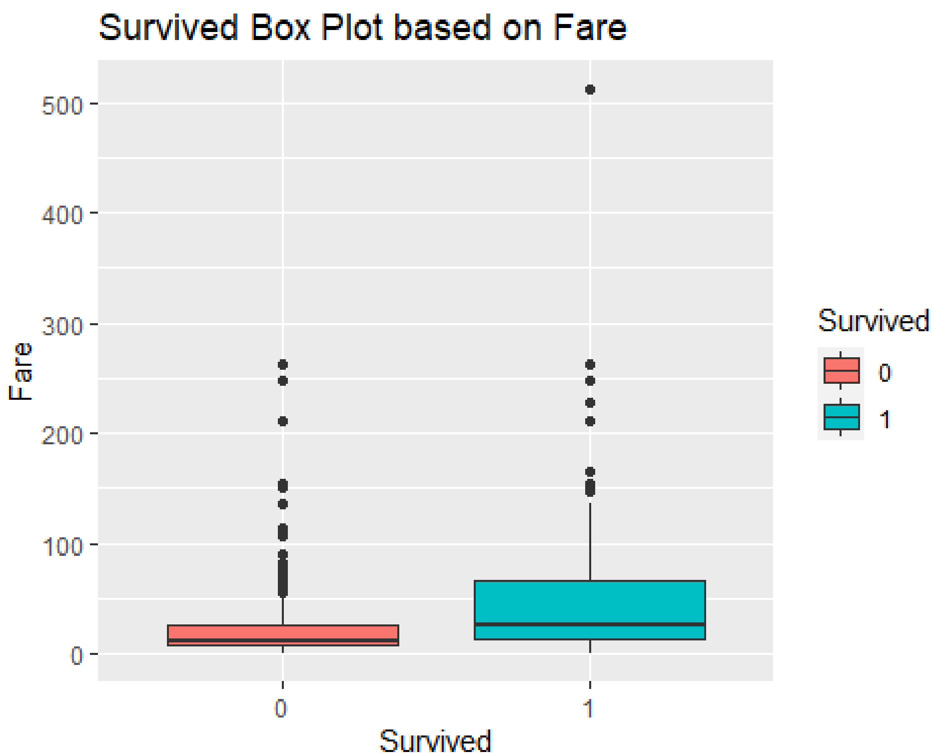
```
#Check the independence of attributes
pairs.panels(data_clean)
```



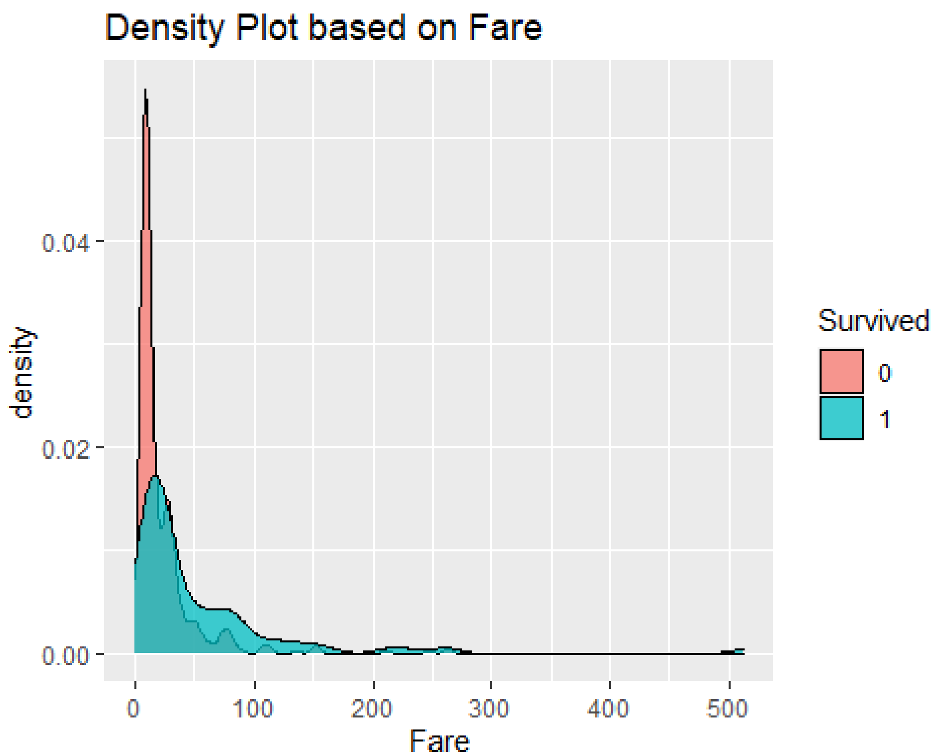
Naive Bayes models expect the features to be independent. So, we've created pair plots using the pairs() function to get an idea about how independent they are from the others.

Inference: About the correlation between the features, we can see that "Fare" and "Pclass" seem to be highly related (-0.55). Also features like "Sex", "Pclass" and "Fare" should be good predictors. The graphs show that "Fare", "Parch" and "SibSp" have a distribution close to normal, but with a left side skew. "Age" has a distribution that is close enough to Gaussian.

```
data_clean %>%
  ggplot(aes(x=Survived,y=Fare,fill=Survived))+
  geom_boxplot()+
  ggtitle('Survived Box Plot based on Fare')
```




```
data_clean %>%
  ggplot(aes(x=Fare,fill=Survived))+
  geom_density(alpha=0.75,color='black')+
  ggtitle('Density Plot based on Fare')
```



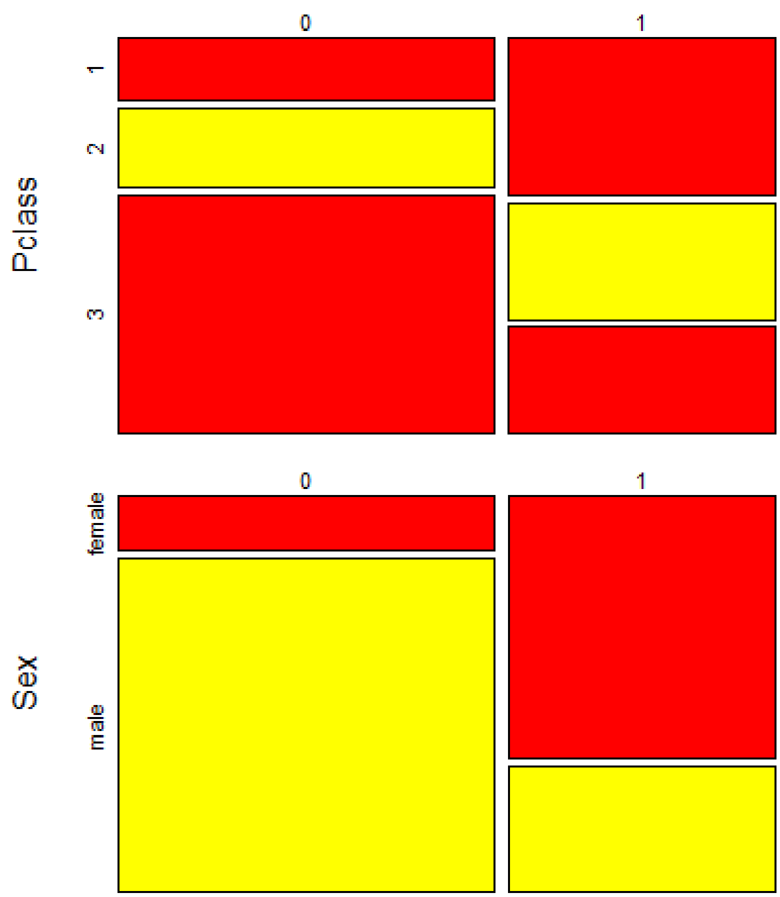

```
#Split dataset into training and testing data
set.seed(234)
smpl<-sample(2,nrow(data_clean),replace=T,prob=c(0.8,0.2))
train<-data_clean[smpl==1,]
test<-data_clean[smpl==2,]

mdl<-naive_bayes(Survived~ .,data=train)
mdl

##
## ================================== Naive Bayes
==================================
##
##  Call:
## naive_bayes.formula(formula = Survived ~ ., data = train)
##
## --------------------------------------------------------------------------
-------
##
## Laplace smoothing: 0
##
## --------------------------------------------------------------------------
-------
##
##  A priori probabilities:
##
##          0         1
```

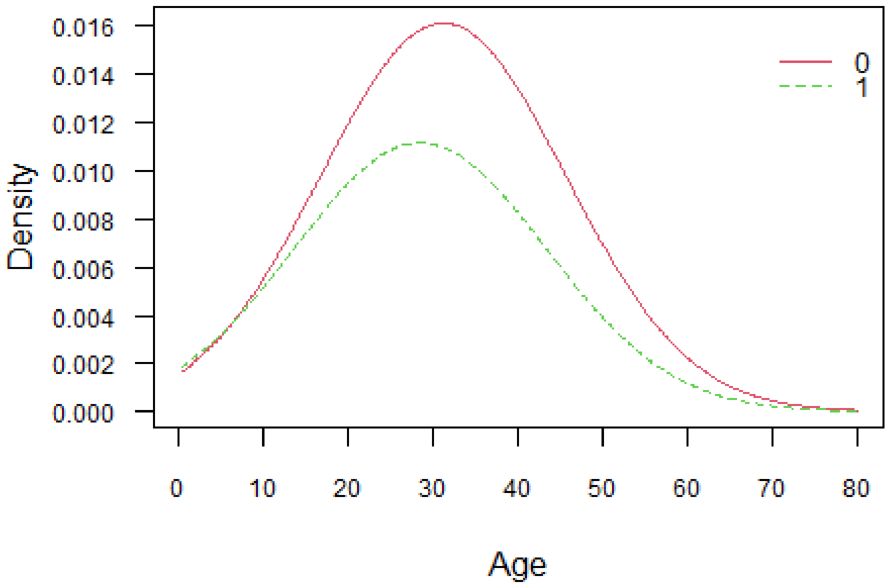
```
## 0.5847458 0.4152542
##
## ---------------------------------------------------------------------------
-------
##
##  Tables:
##
## ---------------------------------------------------------------------------
-------
##  ::: Pclass (Categorical)
## ---------------------------------------------------------------------------
-------
##
## Pclass         0         1
##      1 0.1652174 0.4163265
##      2 0.2115942 0.3061224
##      3 0.6231884 0.2775510
##
## ---------------------------------------------------------------------------
-------
##  ::: Sex (Bernoulli)
## ---------------------------------------------------------------------------
-------
##
## Sex              0         1
##   female 0.1449275 0.6775510
##   male   0.8550725 0.3224490
##
## ---------------------------------------------------------------------------
-------
##  ::: Age (Gaussian)
## ---------------------------------------------------------------------------
-------
##
## Age           0        1
##   mean 31.26812 28.53539
##   sd   14.46155 14.84708
##
## ---------------------------------------------------------------------------
-------
##  ::: SibSp (Gaussian)
## ---------------------------------------------------------------------------
-------
##
## SibSp          0         1
##   mean 0.5565217 0.4816327
##   sd   1.0717675 0.7162269
##
## ---------------------------------------------------------------------------
-------
```

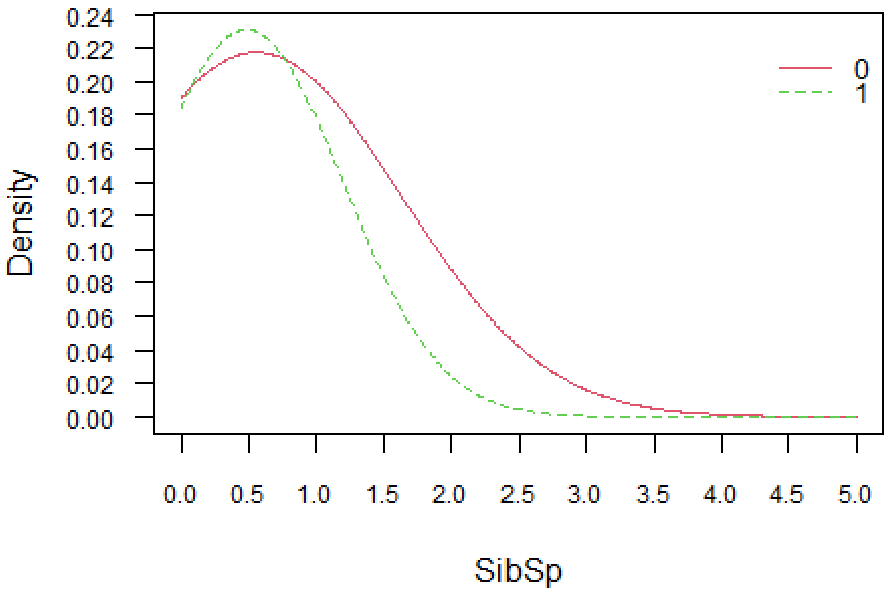
```
##  ::: Parch (Gaussian)
## ---------------------------------------------------------------------------
-------
##
## Parch          0         1
##   mean 0.3768116 0.5183673
##   sd   0.8941257 0.7766240
##
## ---------------------------------------------------------------------------
-------
##
## # ... and 1 more table
##
## ---------------------------------------------------------------------------
-------
```

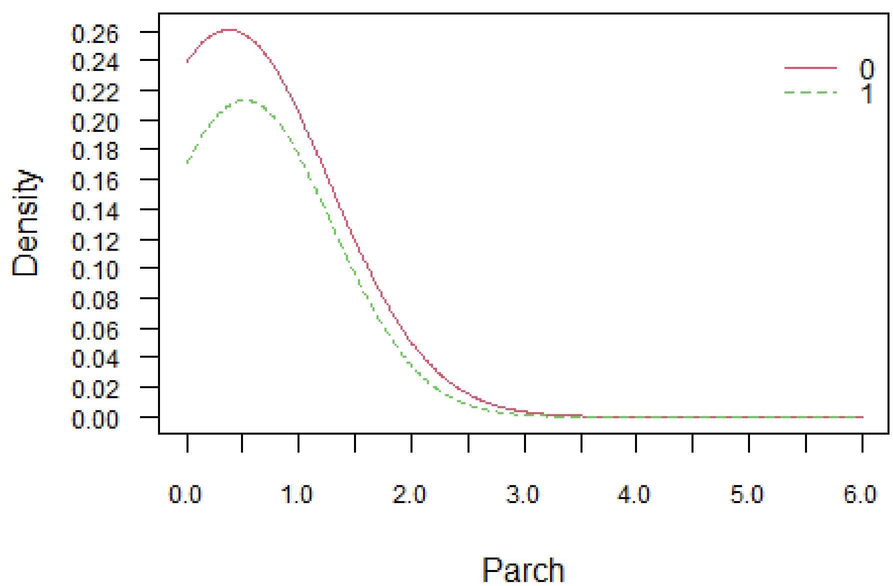
Inference: Here the apriori probabilities are calculated which indicates the distribution of the data. Then conditional probability for each variable is computed by the naïve bayes model separately.

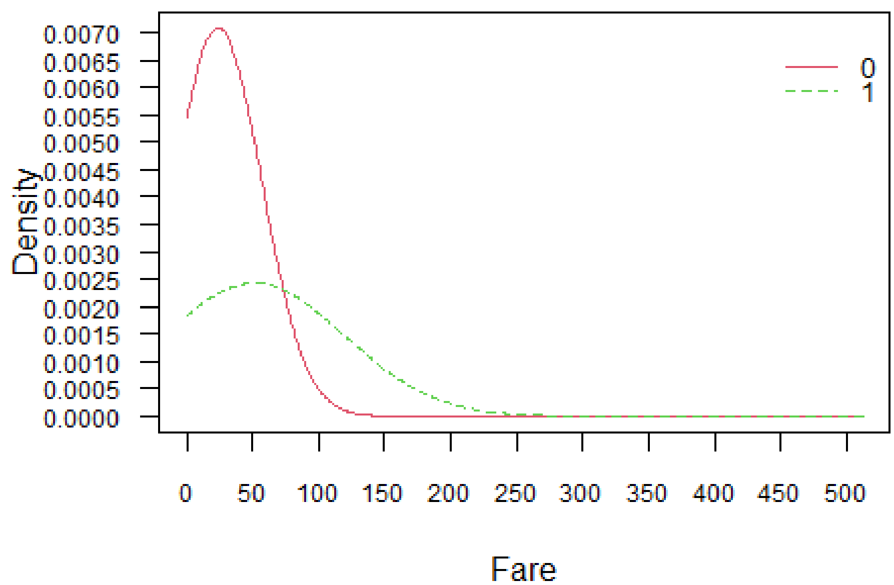
```
plot(mdl)
```




0.016
0.014
0.012
0.010
0.008
0.006
0.004
0.002
0.000
Density
0
1
0
10
20
30
40
50
60
70
80
Age


0.24
0.22
0.20
0.18
0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
Density
0
1
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
SibSp






```
p<-predict(mdl,train,type='prob')

## Warning: predict.naive_bayes(): more features in the newdata are provided
as
## there are probability tables in the object. Calculation is performed based
on
## features to be found in the tables.

head(cbind(p,train))

##            0          1 Survived Pclass    Sex Age SibSp Parch    Fare
## 1 0.93206085 0.06793915        0      3   male  22     1     0  7.2500
## 2 0.07595037 0.92404963        1      1 female  38     1     0 71.2833
## 3 0.51919955 0.48080045        1      3 female  26     0     0  7.9250
## 4 0.12638298 0.87361702        1      1 female  35     1     0 53.1000
```

```
## 5 0.93751292 0.06248708        0       3    male  35      0      0  8.0500
## 7 0.66936769 0.33063231        0       1    male  54      0      0 51.8625
```

```
#To find the accuracy of prediction
p1<-predict(mdl,train)
```

```
## Warning: predict.naive_bayes(): more features in the newdata are provided
as
## there are probability tables in the object. Calculation is performed based
on
## features to be found in the tables.
```

```
(tab1<-table(p1,train$Survived))
```

```
##
## p1    0   1
##   0 316  90
##   1  29 155
```

Inference: The confusion matrix for model is displayed here. Out of 345 not survived, 316 are correctly classified as not survived, and 29 are classified as survived. Out of 245 survived, 155 are correctly classified as survived and 90 are classified as not survived.

```
1-sum(diag(tab1))/sum(tab1)
```

```
## [1] 0.2016949
```

Inference: The model achieved 20.17% accuracy.

Conclusion:

Naive Bayes algorithm is based on Bayes theorem. Bayes theorem gives the conditional probability of an event A given another event B has occurred.

$$P(A/B) = [P(B/A)*P(A)]/P(B)$$

Apriori probabilities:

0	**1**
0.5847458	0.4152542

Conditional probabilities:

Pclass	**0**	**1**
1	0.1652174	0.4163265
2	0.2115942	0.3061224
3	0.6231884	0.2775510

Sex	0	1
female	0.1449275	0.6775510
male	0.8550725	0.3224490

Age	0	1
mean	31.26812	28.53539
sd	14.46155	14.84708

SibSp	0	1
mean	0.5565217	0.4816327
sd	1.0717675	0.7162269

Parch	0	1
mean	0.3768116	0.5183673
sd	0.8941257	0.7766240

The model achieved only 20.17% accuracy, using the given dataset.