

Shuyan Bian – SEC01 (NUID 002309442)

Big Data System Engineering with Scala

Spring 2024

Assignment No. 7



- GitHub Repo URL -

- List of Tasks Implemented

- Code

FileName.Scala (Either text or screenshot can be added below of your code snippet, don't add the whole code just the part you completed)

- Unit tests / Results

1. Exploratory Data Analysis

```
scala> titanic.groupBy("Pclass").avg("Fare").orderBy("Pclass").show();
```

Pclass	avg(Fare)
1	84.15468749999992
2	20.66218315217391
3	13.675550101832997

First class tickets are far more expensive than others.

```
scala> titanic.groupBy("Pclass").agg((sum("Survived")/count("*")).as("Survival_rate")).orderBy("Pclass").show();
```

Pclass	Survival_rate
1	0.6296296296296297
2	0.47282608695652173
3	0.24236252545824846

The survival rate of first-class passengers is significantly higher than that of second-class passengers, and the survival rate of second-class passengers is significantly higher than that of third-class passengers.

```
scala> titanic.withColumn("Group", ceil(col("Age")/10)).groupBy(
"Group").agg(avg("Fare"), (sum("Survived")/count("*")).as("Survival_rate")).orderBy("Group").show();
```

Group	avg(Fare)	Survival_rate
NULL	22.158566666666673	0.2937853107344633
1	30.434439062500008	0.59375
2	29.529531304347838	0.3826086956521739
3	28.306718695652194	0.3652173913043478
4	42.496100000000002	0.44516129032258067
5	41.16318139534884	0.38372093023255816
6	44.77480238095238	0.40476190476190477
7	45.91078235294117	0.23529411764705882
8	25.936680000000003	0.2

The survival rate of teenagers is significantly higher than that of other age groups.

2. Feature Engineering

There are only two entries with Embarked = NULL, so I remove them.

Because the Cabin column has too many missing values, I remove it.

The PassengerId column is not used in the process of training, but it needs to be preserved for producing submission.

I drop the Ticket column, since I think it is hard to use.

SibSp and Parch can be combined to produce a FamilySize column. (Of course, Sibsp and Parch columns can be removed.)

There are many missing values in the Age column. I fill them with the average.

Finally, I find that the test data also has to be preprocessed in a slightly different manner. I choose to fill the missing Fare value with average. (It is integrated with the pipeline.)

Maybe there is a more right and modular way to organize the preprocess procedure, but now it works for this certain data. I'll not bother with it.

3. Prediction

```

val embarkedIndexer = new StringIndexer().setInputCol("Embarked").setOutputCol("EmbarkedIndex")
val embarkedEncoder = new OneHotEncoder().setInputCol("EmbarkedIndex").setOutputCol("EmbarkedVec")
val sexIndexer = new StringIndexer().setInputCol("Sex").setOutputCol("SexIndex")
val sexEncoder = new OneHotEncoder().setInputCol("SexIndex").setOutputCol("SexVec")
val assembler = new VectorAssembler()
    .setInputCols(Array("Pclass", "Age", "FamilySize", "Fare", "SexVec", "EmbarkedVec"))
    .setOutputCol("features")
val lr = new LogisticRegression().setLabelCol("Survived").setFeaturesCol("features")
val pipeline = new Pipeline().setStages(Array(
    preTransformer,
    sexIndexer, sexEncoder,
    embarkedIndexer, embarkedEncoder,
    assembler,
    lr))

```

I use logistic regression for training and predicting. The details are obvious from the code. I organize them separately and integrate them as a whole pipeline.

i Plugins supporting *.csv files found.		Install plugins	Ignore extension
1	PassengerId, Survived		✓
2	892,0		
3	893,0		
4	894,0		
5	895,0		
6	896,1		
7	897,0		
8	898,1		
9	899,0		
10	900,1		
11	901,0		
12	902,0		
13	903,0		
14	904,1		
15	905,0		
16	906,1		
17	907,1		
18	908,0		
19	909,0		
20	910,1		
21	911,1		
22	912,0		
23	913,0		
24	914,1		
25	915,1		
26	916,1		
27	917,0		

I produce my prediction also as a csv file.

Titanic - Machine Learning from Disaster

Start here! Predict survival on the Titanic and get familiar with ML basics



[Overview](#) [Data](#) [Code](#) [Models](#) [Discussion](#) [Leaderboard](#) [Rules](#) [Team](#) [Submissions](#)

Submissions

All

Successful

Errors

Recent ▼

Submission and Description

Public Score ⓘ



submission.csv

Complete · 24m ago

0.77272