

# Final Project: Titanic - Machine Learning from Disaster

! " : # \$ % & ' : 2201110122

( )

\* + , - ' . / 0 1 2 3 4 5 6 7 8 9 . : ; < = >

1912 ? 4 @ 15 A B C D E F G H I J K L % " M N / O " . O P Q R \* + , - ' C S T U V W X / O > N Y . 1 B Z 4 . [ \ ] ^ O \_ ` a . b c d B e f 2 2 2 4 7 g h i Z j k . 1502 ] l m >

noYp\_ = q r s t u B v w x \_ q ] y z { } | \_ } ~ • € • >

C , \ f „ k B ... † ‡ ^ % \_ = \ Š < € • • Ž • , \ • ' ' " " " • . ] | \_ } ~ c p - " — ~ g h ™ Š > œ • 7 ž ? Ÿ ž | ž € £ ¤ ¥ | § ¨ © >

™ Š a

! " # \$

« ¬ project ~ - . ™ Š a • @ kaggle. Titanic ° , € ± ² ³ ´ μ ... ¶ | ¯ • ¸ \ ™ Š ¹ º » data.zip ¼ » k

%&

C« ¬y ° k B...† \_ Û\ Vw. ™š ª Bz k ÚÛ• 7ž ? Ÿž | ž ₣ £ ₧ ¥ | § ¨ g  
h ÜÝ > = \ ™š ª 7% train.csv BƆ = \ ™š ª 7% test.csv >

train.csv ß Úà ¾4 á â gh> ã ä º å 1 891 ] ©. æç ÜÝ Bè ‡. 1 B ß é ê {  
† 1 ½ Y p B è ì % " í « î ï " >

test.csv ™š ª Úà ð w. ÜÝ B v O \_ ñ ò [ Ç gh. " í « î ï " > Š < , q ó ô 1 ...  
† õ € • . ö ! >

— ~ ÷ C ™š k ø - . ù ú ª train.csv B Š < ¾4 z { 418 7 gh> C k ø  
- test.csv © 1 ½ Y p >

' ( + ,

' (	) *	KEY
PassengerId	gh ID	!
Survived	c p	0=½, 1=1
Pclass	¾ ¿ ¨ À	1 = Á = B 2 = Á Â B 3 = Á Ã
Name	• 7	!
Sex	i	!
Age	Ä Å ™ % Æ Ç. ? Ÿ	!
SibSp	* + , - ' 4. È É Ê Ë Ì Í ™ Î	!
Parch	Ï 4 * + , - ' . Đ Ñ / Ò Ó	!
Ticket	¿ '	!
Fare	Ô h ¿ Õ	!
Cabin	h Ö '	!
Embarked	Ï Z × Ø	C = Cherbourg, Q = Queenstown, S = Southampton

Pclass' ¤ £ ¤ ¥² Ç> SES©. û ì  
1st = 4 §  
2nd = k ù  
3rd = ý §

Age' þ ô ? Ÿ ÿ ! 1B" ? Ÿ %ÿ™> þ ô ? Ÿ 1 # \$. B" %~ xx.5 . &'

sibsp'™ŠªÄ, ( ) ' \* + P, - . ...  
Sibling = Ê Ê BÊ Ê B / Ê B / Ê  
Spouse = O1 B2Ó> 34i 561l 78©

Parch :™ŠªÄ, ( ) ' \* + P, - . ...  
Parent = Ñ9ž Ð9  
Child =E : ž : Óž / E ž / Ó  
\_qÒÓ; i < == >Ô? Bt µ@{ † • å Parch=0>

ABCD

...†BCDEyFÄ€Ã(AB.óô

- 1. Logistic Regression
- 2. Random Forest
- 3. Xgboost

ûG\$H

0. - " . / O-1 203450

eI ÁÃ) J E K L • «™Š:

```

1 import pandas as pd
2 df_train = pd.read_csv('./data/train.csv')
3 df_test = pd.read_csv('./data/test.csv')

```

1.. / 67—89“Age”: “Cabin” 3 “Embarked”; < = > ?

1.1 @! AgeB...† MN4b c d O“E P i Ò Ó Q R”. í « S" B T  
 Ä...† V Ü ? Ÿ ž i i c p \_ U V C. W. >

C 1 2 O k B...† X (Sex, Survived) @ ù ú ™ š Y ? â Z B E [ \ ] V ^ . (Sex, Survived)  
 Z k — ~ bootstrap è % • ) B • \_ ` ? Ÿ . a b c

```

1 import random
2
3 age = {
4     ! ! 'male': {
5         ! ! ! ! 1: df_train['Age'][(df_train['Survived']==1) &
6         (df_train['Sex']=='male') & ~df_train['Age'].isnull()].values,
7         ! ! ! ! 0: df_train['Age'][(df_train['Survived']==0) &
8         (df_train['Sex']=='male') & ~df_train['Age'].isnull()].values
9     },
10    ! ! 'female': {
11        ! ! ! ! 1: df_train['Age'][(df_train['Survived']==1) &
12        (df_train['Sex']=='female') & ~df_train['Age'].isnull()].values,
13        ! ! ! ! 0: df_train['Age'][(df_train['Survived']==0) &
14        (df_train['Sex']=='female') & ~df_train['Age'].isnull()].values
15    }
16 }
17
18 for i in df_train['Age'][df_train['Age'].isnull()].index:
19     ! ! df_train['Age'][i] = random.choice(age[df_train['Sex'][i]]
20     [df_train['Survived'][i]])

```

C 4 5 O k B...† ß < d ™ š X i â Z B E ~ 1 2 O k e i Z . f g c \_ ` ? Ÿ . a b  
 c

















