


Optional Assignments

성신여자대학교 2023 년 1 학기 자료구조 수업

- 모든 문제는 Kaggle에서 인공지능을 이용해 생존율을 구하는 문제와 무관하게 만들었습니다. 생존 정보가 담긴 행은 문제풀이와 관련이 없어요. 다만 AI융합학부 학생들이 Kaggle에 관심을 가질 수 있도록 데이터에 남겨 둡니다.
 - 모든 사진, 또는 그림 자료는 과제와 무관합니다.
 - Input files
 - titanic.csv
 - A열: 승객 탑승 순서
 - B열: 이름
 - C열: 타이타닉 호 침몰 후 구조되었는지 여부(1은 생존, 0은 사망)
 - D열: 객실 등급(1등석, 2등석, 3등석)
 - E열: 성별
 - F열: 티켓 가격
 - G열: 식당 TOFFI까지 거리
 - H열: 침몰 전 승객의 이동 속도
 - test.dat
 - 1행: 승객 숫자, 승객 간 이동 가능한 길의 숫자
 - 2행~: 승객 탑승 번호, 승객 탑승 번호, 두 승객 간 거리
- 정보가 없는 경우 승객 간 직접적으로 이동이 불가능합니다.
- 892번 노드는 구명정을 의미합니다.


데이터 소개

- 본 문제는 LMS 에 제공한 데이터를 사용하며, Kaggle 데이터를 사용하지 않아도 됩니다.
- Kaggle 에 접속합니다.

 Competitions Datasets Models Code Discussions Courses ... Search Sign In Register

Start with more than a blinking cursor

Kaggle offers a no-setup, customizable, Jupyter Notebooks environment. Access GPUs at no cost to you and a huge repository of community published data & code.

 REGISTER WITH GOOGLE

Register with Email

Predict Malicious Websites: XGBoost

File Edit Insert Run View Help

```
data = pd.read_csv("../input/dataset.csv")

# clean up column names
data.columns = data.columns.\
    str.strip().\
    str.lower()

# remove non-numeric columns
data = data.select_dtypes(['number'])

# split data into training & testing
train, test = train_test_split(data, shuffle=True)

# peek @ dataframe
train.head()

# split training data into inputs & outputs
X = train.drop(["type"], axis=1)
Y = train["type"]

# specify model (xgboost defaults are generally fine)
model = xgb.XGBRegressor(free_method =)











# fit our model
model.fit(y=Y, X=X)

In[]: # split testing data into inputs & output
test_X = test.drop(["type"], axis=1)
test_Y = test["type"]



# predictions & actual values, from test set
predictions = model.predict(test_X) > 0
actual = test_Y
```



Console Draft Session (9m10s) CPU 45% GPU Off RAM 4.5/8GB Disk 32MB/128GB

- Data sets 에서 Titanic 을 검색합니다.







Datasets










 

 Titanic 

All datasets X Computer Science Education Classification Computer Vision NLP Data Visualization

Pre-Trained Model

 1,243 Datasets   

	Titanic Khashayar Baghizadeh Hosseini · Updated 6 years ago Usability 7.1 · 1 File (CSV) · 11 kB	 951  Gold ...
	Titanic dataset Brenda N · Updated a year ago Usability 10.0 · 1 File (CSV) · 12 kB	 499  Gold ...
	Titanic Azeem Bootwala · Updated 6 years ago Usability 8.2 · 2 Files (CSV) · 12 kB	 162  Bronze ...

- 과제에서 사용할 데이터는 이를 수업자료로 활용하기 위해 가공한 자료입니다. 다양한 데이터가 있으며, data card 를 보면 자세한 내용을 확인할 수 있습니다.

☰

+

🔍

🏆

📊

👤

🔗

📄

📖

📁

🏠

🔍 Search

RAHUL · UPDATED 3 YEARS AGO

▲ 66

New Notebook

Download (35 kB)

⋮

Titanic

Machine Learning from Disaster

Data Card

Code (339)

Discussion (0)

About Dataset

The sinking of the RMS Titanic is one of the most infamous shipwrecks in history. On April 15, 1912, during her maiden voyage, the Titanic sank after colliding with an iceberg, killing 1502 out of 2224 passengers and crew. This sensational tragedy shocked the international community and led to better safety regulations for ships.

One of the reasons that the shipwreck led to such loss of life was that there were not enough lifeboats for the passengers and crew. Although there was some element of luck involved in surviving the sinking, some groups of people were more likely to survive than others, such as women, children, and the upper-class.

In this challenge, we ask you to complete the analysis of what sorts of people were likely to survive. In particular, we ask you to apply the tools of machine learning to predict which passengers survived the tragedy.

Usability ⓘ
6.76

License
Unknown

Expected update frequency
Not specified

1. Ticket Price



출처: Quora

제공한 .csv 파일의 승객 순서는 승객의 탑승 순서이기도 합니다. 타이타닉 호는 2 등석 승객에 한해 먼저 도착한 승객에게 바우처를 제공합니다. i 번째 탑승한 승객이 제공받는 바우처 금액은 i 번째 탑승한 승객보다 나중에 탑승한 승객 중 가장 빠르게 탑승한 j 번째 승객($i < j$) 티켓 가격의 10%입니다. 단, j 번째 승객의 티켓 가격이 i 번째 승객의 티켓 가격보다 작은 경우에만 바우처가 발행됩니다($\text{fares}_j \leq \text{fares}_i$). 승객 별 바우처 가격을 콘솔창에 출력하세요.

예시 1: 예시는 이해를 돕기 위해 간략히 기술한 예시입니다. 문제에 맞게 변형하세요.

Input: `fares_ = [10, 2, 3, 1, 2]`

Output: `[0.2, 0.1, 0.1, 0, 0]`

설명: 0 번 승객(티켓 가격 10)에게 1 번 승객(티켓 가격 2)의 10%인 0.2 바우처가 발행됨. 1 번 승객(티켓 가격 2)에게 3 번 승객(티켓 가격 1)의 10%인 0.1 바우처가 발행됨. 2 번 승객(티켓 가격 3)에게 3 번 승객(티켓 가격 1)의 10%인 0.1 바우처가 발행됨. 4 번 승객은 5 번 승객의 티켓 가격이 더 비싸기 때문에 바우처가 발행되지 않음. 5 번 승객은 마지막에 탑승했기 때문에 바우처가 발행되지 않음.

예시 2:

Input: `fares_ = [10, 11, 12]`

Output: `[0, 0, 0]`

설명: 모든 승객은 바우처를 발행 받는 경우가 없음.

2. Number of Partners (어려운 문제)



출처: Getty Images

3 등석 승객들은 모두 댄스 파티에 참가합니다. 댄스 파티는 모든 승객들이 모였을 때 시작합니다. 1 등석 승객 중에서는 Rose 만 참석합니다. 파티 장소는 가장 아래층에 있는 3 등석 승객을 위한 식당 'TOFFI'입니다. 식당까지의 거리(dist_to_toffi, titanic.csv G 열)는 승객마다 다릅니다. TOFFI 까지 가는 속도(speed, titanic.csv H 열)도 승객마다 다릅니다. 단 식당까지의 거리, TOFFI 까지 가는 속도는 모두 정수형 변수(int)라고 가정할 수 있습니다.

댄스 파티에 참가할 때는 모두 기본적인 예절을 지킵니다. 걸음이 빠른 승객이 식당까지 가는 도중 다른 승객을 마주친다면, 앞질러 가지 않고 함께 손을 잡고 파트너가 되어 갑니다. 파트너가 될 수 있는 숫자에는 제한이 없으며, 파트너가 된 뒤에는 걸음이 느린 승객의 속도로 이동합니다. 나홀로 도착한 승객이 있을 경우에도 이를 파트너라 여깁니다. TOFFI 에 도착하는 순간에 다른 승객과 마주쳐도 손을 잡고 파트너가 됩니다.

댄스 파티에 참가한 파트너의 숫자를 구하세요.

예시 1: 예시는 이해를 돕기 위해 간략히 기술한 예시입니다. 문제에 맞게 변형하세요.

Input: distance_to_toffi_ = [2, 4, 10, 5, 7], speed = [2, 4, 1, 1, 3]

Output: 3

0 번 승객(speed_ = 2)과 1 번 승객(speed_ = 4)는 TOFFI 에서 만나 파트너가 됩니다. 3 번 승객(speed_ = 1)은 가장 뒤에서 출발하여 가장 늦게 홀로 도착합니다. 4 번 승객(speed_ = 1)과 5 번 승객(speed_ = 3)은 TOFFI 로부터 4 만큼 떨어진 곳에서 만나 파트너가 됩니다. 더 이상 승객이 없기 때문에 정답은 파트너 숫자는 셋입니다.

예시 2:

Input: distance_to_toffi_ = [20, 18, 16], speed_ = [4, 2, 1]

Output: 1

0 번 승객(speed_ = 4)과 1 번 승객(speed_ = 2)는 TOFFI 에서 16 만큼 떨어진 위치에서 파트너가 됩니다. 이렇게 파트너가 된 0 번 승객과 1 번 승객은 1 번 승객 속도(speed_ = 2)로 이동합니다. 그리고 2 번 승객과 TOFFI 에서 14 만큼 떨어진 위치에서 파트너가 됩니다. 0 번, 1 번, 2 번 승객으로 이루어진 파트너는 3 번 승객의 속도(speed_ = 1)로 TOFFI 까지 이동합니다.

3. Deliver Messages



출처: <https://www.historyextra.com/>(좌), <https://postalmuseum.si.edu/>(우)

3등석 승객 Watson은 모든 승객들에게 타이타닉 호의 침몰 소식을 알려야 합니다. Watson의 titanic.csv의 H열 침몰 전 이동 속도를 따릅니다. Watson은 자신의 객실에서 출발해 모두에게 침몰 소식을 알릴 때까지 얼마나 걸릴까요? 위급 상황이기 때문에 Watson이 승객에게 소식을 전하는 시간은 무시할 수 있을 정도로 짧습니다. 풀이 방법을 기술하고, 결과를 콘솔창에 보이세요.

4. Save Jack



출처: 레딧

Rose가 Jack을 구하러 출발하였습니다. 100분이 초과하면 타이타닉 호는 침몰하며, 구명정에 탈 수 없습니다. 지금 Jack은 수갑에 묶여 움직일 수 없는 상태입니다. Rose가 Jack을 찾은 뒤, 구명정을 탈 수 있을까요? 타이타닉 호가 물에 잠기며 Rose와 Jack 모두 거리 1을 움직이는데 1분이 걸립니다(titanic.csv speed_는 무시). 수갑을 푸는 시간은 10분이 걸립니다. 풀이 방법을 기술하고, 결과를 콘솔창에 보이세요.

5. Check Survival Rate

당신은 구명정에 누구를 태울지 결정할 수 있습니다. 4번 문제와 마찬가지로 물에 잠겨 가는 상태에서 승객의 이동 속도는 이제 의미가 없습니다(titanic.csv 파일의 speed_는 무시). 구명정 탑승 우선 순위는 1등석 여성, 다음은 2등석 여성입니다. 다른 승객은 구명정과 가까운 순서대로 탑승할 수 있습니다. 구명정의 정원은 600명입니다. 1등석 남성의 생존률을 구하세요.

구명정과 가까운 순서대로 정렬할 때 알고 있는 모든 정렬 방식을 확인해 보고, 이를 비교하여 가장 좋다고 판단하는 알고리즘을 적용하세요.



출처: <https://jamescamerontitanic.fandom.com/>