mergeTables

May 9, 2018

```
In [42]: import pandas as pd
         import numpy as np
```

```
In [43]: df = pd.read_csv("/Users/abhinavgarg/Projects/Data-Science/stage4/data/matchedTuples.
```

```
In [44]: platformDict = {"PC":"PC", "Xbox 360":"Xbox", "Xbox One":"Xbox", "Xbox":"Xbox",
         "PlayStation 4":"PlayStation","PlayStation 3":"PlayStation",
         "PlayStation 2":"PlayStation","PlayStation":"PlayStation","Wii":"Wii",
         "Macintosh":"Macintosh","PSP":"PlayStation","Game Boy Advance":"Nintendo",
         "PlayStation Vita":"PlayStation","Linux":"Linux","DS":"Nintendo","Android":"Mobile",
         "iOS (iPhone/iPad)":"Mobile","GameCube":"Nintendo","Wii U":"Wii","Mac":"Macintosh",
         "Nintendo 3DS":"Nintendo","Nintendo 64":"Nintendo", "Super Nintendo":"Nintendo",
         "Nintendo DS":"Nintendo", "NES":"Nintendo","Nintendo Switch":"Nintendo",
         "Nintendo GameCube":"Nintendo","Dreamcast":"Sega","Mobile":"Mobile",
         "Game Boy Color":"Nintendo","iOS":"Mobile","3DS":"Nintendo","Game Boy":"Nintendo",
         "Sega Mega Drive/Genesis":"Sega", "Saturn":"Sega","Genesis":"Sega",
         "Windows Mobile":"Mobile","Sega Master System":"Sega","Sega Saturn":"Sega",
         "Nintendo 2DS":"Nintendo","PlayStation VR":"PlayStation","Commodore 64":"Commodore",
         "Amiga":"Commodore","BlackBerry":"Mobile","N-Gage":"Mobile","Sega CD":"Sega",
         "PlayStation Now":"PlayStation","Sega Game Gear":"Sega","Sega 32X":"Sega",
         "Atari 2600":"Atari","Atari ST":"Atari", "Atari 8-bit":"Atari", "Atari 7800":"Atari",
         "Atari 5200":"Atari","Atari Jaguar":"Atari", "Atari Lynx":"Atari","NeoGeo":"Neo Geo",
         "NeoGeo Pocket Color":"Neo Geo","Neo Geo":"Neo Geo", "Neo Geo Pocket Color":"Neo Geo"
         "Neo Geo CD":"Neo Geo","Apple II":"Macintosh","Neo-Geo CD":"Neo Geo",
         "Windows Mobile":"Mobile","HTC Vive":"Mobile"
         }
```

```
In [45]: def getMergedPlatform(lplatform,rplatform):
             mergedPlatString = lplatform+"|"+rplatform
             uniquePlatforms = set()
             for it in mergedPlatString.split('|'):
                 uniquePlatforms.add(platformDict.get(it,"Other"))
             return '|'.join(list(uniquePlatforms))

         def getLargerString(lString,rString):
             if len(lString) > len(rString):
                 return lString
             return rString
         def getMergedSet(lplatform,rplatform):
             mergedPlatString = lplatform+"|"+rplatform
             return '|'.join(list(set(mergedPlatString.split('|'))))

         # def getCommonDate(ldate,rdate):
```

```
In [47]: def checkNullValue(left,right):
             if str(left) != "nan" and str(right) != "nan":
                 return 0
             if str(left) != "nan" and str(right) == "nan":
                 return 1
             if str(left) == "nan" and str(right) != "nan":
                 return 2
             return 3

         def getMergedValue(left,right,parameter_type=None):
             ret = checkNullValue(left,right)
             if ret == 3:
                 return ""
             if ret == 2:
                 return right
             if ret == 1:
                 return left
             if parameter_type == 'title':
                 return getLargerString(left,right)
             if parameter_type == 'platform':
                 return getMergedPlatform(left,right)
             return getMergedSet(left,right)
```

```
In [56]: dfMerged = pd.DataFrame()
         dfMerged['ltable_ID'] = pd.Series(dtype=str)
         dfMerged['rtable_ID'] = pd.Series(dtype=str)
         dfMerged['Title'] = pd.Series(dtype=str)
         dfMerged['Developer'] = pd.Series(dtype=str)
         dfMerged['Publisher'] = pd.Series(dtype=str)
         dfMerged['Platform'] = pd.Series(dtype=str)
         dfMerged['Genre'] = pd.Series(dtype=str)
         dfMerged['ReleaseDate'] = pd.Series(dtype=str)
         dfMerged['lRating'] = pd.Series(dtype=float)
         dfMerged['rRating'] = pd.Series(dtype=float)

         for idx, row in df.iterrows():
         #     print(row)
             title = getMergedValue(row['ltable_Title'], row['rtable_Title'],'title')
```

```
    developer = getMergedValue(row['ltable_Developer'], row['rtable_Developer'])
    publisher = getMergedValue(row['ltable_Publisher'], row['rtable_Publisher'])
    platform = getMergedValue(row['ltable_Platform'],
                              row['rtable_Platform'],'platform')
    genre = getMergedValue(row['ltable_Genre'], row['rtable_Genre'])

    relDateRet = checkNullValue(row['ltable_CleanRDate'],row['rtable_CleanRDate'])
    releasedate = ""
    if relDateRet == 0:
        releasedate = row['ltable_CleanRDate']
    elif relDateRet == 1:
        releasedate = row['ltable_CleanRDate']
    elif relDateRet == 2:
        releasedate = row['rtable_CleanRDate']
    dfMerged.loc[idx] = [row['ltable_ID'],row['rtable_ID'],title,developer,publisher,
                         platform,genre,releasedate,row['ltable_Rating'],
                         row['rtable_Rating']]

#     row[ltable_ID],row[rtable_ID],row[ltable_ID]
```

```
In [59]: dfMerged.head()
         dfMerged.to_csv("/Users/abhinavgarg/Projects/Data-Science/stage4/data/tableE.csv",
                         index=False)
```