bi_cricket (1)

November 24, 2019

[ ]:
```
# from google.colab import drive
# drive.mount('/gdrive')
# %cd /gdrive
```

[282]:
```
import pandas as pd
import os
```

[283]:
```
files = [file for dirpath, directory, file in os.walk('./all_csv/')][0]
# files=[file for dirpath,directory,file in os.walk(r'/gdrive/My Drive/all_csv/
 ↪')][0]
```

[284]:
```
match_data = pd.DataFrame(data=None)
odi_scorecard = pd.DataFrame(data=None)
ttwenty_scorecard = pd.DataFrame(data=None)
odi_info = pd.DataFrame(data=None)
ttwenty_info = pd.DataFrame(data=None)
```

[285]:
```
def rename_date_umpire(index_list):
    n = 0
    for i in range(index_list.__len__()):
        if str.lower(index_list[i]).strip() == 'date':
            index_list[i] += '_'+str(n)
            n += 1
    n = 0
    for i in range(index_list.__len__()):
        if str.lower(index_list[i]).strip() == 'umpire':
            index_list[i] += '_'+str(n)
            n += 1
    n = 0
    for i in range(index_list.__len__()):
        if str.lower(index_list[i]).strip() == 'team':
            index_list[i] += '_'+str(n)
            n += 1
    return index_list
```

[286]:
```
def find_game(df_game,df_info):
    if 'series' in df_info.columns:
        if 'odi' in str.lower(df_info.iloc[0]['series']):
            return 'odi'
```

```
        if 't20i' in str.lower(df_info.iloc[0]['series']) or 't20' in str.
↪lower(df_info.iloc[0]['series']) or 'indian premier league' in str.
↪lower(df_info.iloc[0]['series'])  or 'indian premier league' in str.
↪lower(df_info.iloc[0]['competition']):
            return 'twenty'
    if max(df_game['balls-bowled'])<=24:
        return 'twenty'
    if 24<max(df_game['balls-bowled'])<=60:
        return 'odi'
    return
```

[287]:
```
def append_file(temp_df, temp_info_df, type_game):
    global odi_scorecard
    global ttwenty_scorecard
    global odi_info
    global ttwenty_info
    if type_game == 'odi':
        odi_scorecard = odi_scorecard.append(temp_df, ignore_index=True)
        odi_info = odi_info.append(temp_info_df, ignore_index=True)
    elif type_game == 'twenty':
        ttwenty_scorecard = ttwenty_scorecard.append(
            temp_df, ignore_index=True)
        ttwenty_info = ttwenty_info.append(temp_info_df, ignore_index=True)
```

[288]:
```
def get_extras_type(match_data):
    list_extras = []
    for index, row in match_data.iterrows():
        ov = str(row['over'])
        if '.' in ov:
            ov = str(row['over']).split('.')
            ball_no = int(ov[1])
            over_no = int(ov[0])
        else:
            continue
        if row['extras'] != 0:
            if row['runs'] != 0:
                match_data.loc[index, 'extras_type'] = 'w'
            list_extras.append(index)
        if ball_no > 6:
            if len(list_extras) > 0:
                match_data.loc[list_extras.pop(-1), 'extras_type'] = 'w'
    for i in list_extras:
        match_data.loc[i, 'extras_type'] = 'b'
    return match_data
```

[289]:
```
def prepare_scorecard(match_data,team_0,team_1):
    match_data = get_extras_type(match_data)
#     print(match_data[match_data['bowler']==        'Mashrafe Mortaza'])
```

```
    teams=['','']
    players = list((match_data['striker'].append(
        match_data['non-striker']).append(match_data['bowler'])).unique())
    #     to make 22 players if any player has not played
#     for i in range(len(players),22):
#         players.append('p_'+str(i))
#         fow
#    ␣
 ↪player_stats=['match-id','innings','name','batting-position','over-batsman','runs-scored','
    player_stats = ['match-id', 'team-name', 'innings', 'name',␣
 ↪'batting-position', 'over-batsman', 'runs-scored', 'balls-played', 'dots',␣
 ↪'ones', 'twos', 'threes', 'fours', 'sixes',
                    'wicket-method', 'balls-bowled', 'maiden-overs',␣
 ↪'runs-given', 'wickets', 'extras', 'fall-of-wicket-score',␣
 ↪'fall-of-wicket-overs', 'fall-of-wicket-no', 'fall-of-wicket-bowler']
    player_data = {key: {key_type: 0 for key_type in player_stats}
                   for key in players}
    for p in players:
        player_data[p]['match-id'] = match_data.loc[0, 'file_no']
        player_data[p]['name'] = p
    team_score = 0
    balls = 0
    pos = 1
    inning = False
    w = 1
    p_no = 1
    w_no = 1
    extras_over = 0
    for index, row in match_data.iterrows():
        ov = str(row['over'])
        if '.' in ov:
            ov = ov.split('.')
            ball_no = int(ov[1])
            over_no = int(ov[0])
        else:
            continue
        if over_no > 50:
            player_data = [value for key, value in player_data.items()]
            scorecard = pd.DataFrame(data=player_data)
            scorecard = scorecard[player_stats]
            return scorecard
        if ball_no == 1 and over_no == 0:
            pos = 1
            w = 1
            team_score = 0
            w_no = 1
            runs_over = 0
```

```
            if row['innings']==1 and teams[0]=='':
                teams[0]=row['batting-team']
                if teams[0]==team_0:
                    teams[1]=team_1
                elif teams[0]==team_1:
                    teams[1]=team_0
            # if row['innings']!=1:
            #     p_no=12
        if ball_no == 1:
            extras_over = 0
        if row['runs'] == 1:
            player_data[row['striker']]['ones'] += 1
        elif row['runs'] == 2:
            player_data[row['striker']]['twos'] += 1
        elif row['runs'] == 3:
            player_data[row['striker']]['threes'] += 1
        elif row['runs'] == 4:
            player_data[row['striker']]['fours'] += 1
        elif row['runs'] == 6:
            player_data[row['striker']]['sixes'] += 1
        elif row['extras'] == 0:
            player_data[row['striker']]['dots'] += 1
        if player_data[row['striker']]['batting-position'] == 0:
            player_data[row['striker']]['batting-position'] = pos
#             print(type(row['over']),type(extras_over))
#             print(row['over'])
            player_data[row['striker']
                        ]['over-batsman'] = float(row['over'])-extras_over
            pos += 1
        if player_data[row['non-striker']]['batting-position'] == 0:
            player_data[row['non-striker']]['batting-position'] = pos
            player_data[row['non-striker']
                        ]['over-batsman'] = float(row['over'])-extras_over
            pos += 1
    # wicket
#         print(row['out-player'])
        if not pd.isna(row['out-player']):
            player_data[row['out-player']]['wicket-method'] = row['out']
#             fow
#             player_data[players[p_no-1]]['fow']=w
#             player_data[players[p_no-1]]['fow_runs']=team_score
#             player_data[players[p_no-1]]['fow_overs']=row['over']
#             player_data[players[p_no-1]]['fow_batsman']=row['out-player']
#             player_data[players[p_no-1]]['fow_bowler']=row['bowler']
            p_no += 1
            w += 1
            if row['out'] != 'run out':
```

```
                player_data[row['bowler']]['wickets'] += 1
            player_data[row['out-player']]['fall-of-wicket-score'] = team_score
            player_data[row['out-player']
                        ]['fall-of-wicket-overs'] =␣
↪float(row['over'])-extras_over
            player_data[row['out-player']]['fall-of-wicket-no'] = w_no
            player_data[row['out-player']
                        ]['fall-of-wicket-bowler'] = row['bowler']
            w_no += 1
        team_score += row['runs']+row['extras']
        runs_over += row['runs']
        if row['extras'] != 0 and row['extras_type'] == 'w':
            runs_over += 1
            player_data[row['bowler']]['extras'] += 1
            player_data[row['striker']]['runs-scored'] += row['extras']-1
            player_data[row['bowler']]['runs-given'] += row['extras']-1
            extras_over += 0.1
        elif row['extras'] != 0:
            #             print(row)
            player_data[row['bowler']]['balls-bowled'] += 1
            player_data[row['striker']]['balls-played'] += 1
        else:
            player_data[row['striker']]['balls-played'] += 1
            player_data[row['bowler']]['balls-bowled'] += 1
            player_data[row['bowler']]['runs-given'] += row['runs']
        player_data[row['striker']]['runs-scored'] += row['runs']
        if ball_no >= 6:
            if ball_no == 6 and runs_over == 0:
                player_data[row['bowler']]['maiden-overs'] += 1
            runs_over = 0
        player_data[row['striker']]['innings'] = row['innings']
        player_data[row['striker']]['team-name']=row['batting-team']
        player_data[row['non-striker']]['team-name']=row['batting-team']
        if row['innings'] == 1:
            player_data[row['bowler']]['innings'] = 2
            player_data[row['bowler']]['team-name']=teams[1]
#             print(teams,row['bowler'])
        elif row['innings']==2:
            player_data[row['bowler']]['innings'] = 1
            player_data[row['bowler']]['team-name']=teams[0]
#             print(teams,row['bowler'])
#     print(player_data)

    player_data = [value for key, value in player_data.items()]
    scorecard = pd.DataFrame(data=player_data)
    scorecard = scorecard[player_stats]
    return scorecard
```

[290]:
```
i = 0
for file in files[:5]:
    print(i, '--', file)
    i += 1
    count = 0
    df_index = []
    df_row = []
    add = r'./all_csv/'+file
#     add=r'/gdrive/My Drive/all_csv/'+file
    df_index = ['file_no']
    file_no = file.split('.')[0]
    df_row = [file_no]
    with open(add) as f:
        new_f = f.readlines()
        for line in new_f:
            if 'version' in line:
                count += 1
            elif 'info' in line:
                count += 1
                line = line.strip().split(',')
                df_index.append(line[1])
                df_row.append(line[2])
            else:
                df_index = rename_date_umpire(df_index)
                df_dic = dict(zip(df_index, df_row), index=[0])
                temp_info_df = pd.DataFrame(df_dic)
#                 df_info=df_info.append(temp_info_df,ignore_index=True)
                # gender=df_info['gender'].iloc[0]
                # gender=str.lower(gender.strip())
                break
    temp_df = pd.read_csv(add, skiprows=count, names=[
                          0, 'innings', 'over', 'batting-team', 'striker',␣
 ↪'non-striker', 'bowler', 'runs', 'extras', 'out', 'out-player'])
    temp_df = temp_df.drop([0], axis=1)
    temp_df['file_no'] = [file_no]*(temp_df.shape[0])
    temp_sc = prepare_scorecard(temp_df,temp_info_df['team_0'].
 ↪values[0],temp_info_df['team_1'].values[0])
    # print(temp_sc)
#     append_file(temp_df,gender,type_game)
    append_file(temp_sc, temp_info_df, find_game(temp_sc, temp_info_df))
```

```
0 -- 1019975.csv
1 -- 682919.csv
2 -- 952191.csv
3 -- 1043961.csv
4 -- 565820.csv
```

[291]:
```
odi_info.columns
```

[291]:
```
Index(['city', 'competition', 'date_0', 'file_no', 'gender', 'index',
       'match_number', 'match_referee', 'player_of_match', 'reserve_umpire',
       'season', 'series', 'team_0', 'team_1', 'toss_decision', 'toss_winner',
       'tv_umpire', 'umpire_0', 'umpire_1', 'venue', 'winner', 'winner_runs'],
      dtype='object')
```

[292]:
```
ttwenty_info.columns
```

[292]:
```
Index(['city', 'competition', 'date_0', 'file_no', 'gender', 'index',
       'match_number', 'match_referee', 'neutralvenue', 'player_of_match',
       'reserve_umpire', 'season', 'series', 'team_0', 'team_1',
       'toss_decision', 'toss_winner', 'tv_umpire', 'umpire_0', 'umpire_1',
       'venue', 'winner', 'winner_runs', 'winner_wickets'],
      dtype='object')
```

[280]:
```
odi_info.to_csv('./odi_info.csv', index=False)
ttwenty_info.to_csv('./ttwenty_info.csv', index=False)
odi_scorecard.to_csv('./odi_scorecard.csv', index=False)
ttwenty_scorecard.to_csv('./ttwenty_scorecard.csv', index=False)
```