DAY-1

27-12-2016

-------------------------------------------------------

**1. To import packages**

**import** **pandas** **as** **pd**

**import** **numpy** **as** **np**

**import** **matplotlib.pyplot** **as** **plt**

**---------------------------------**

Steps to load data in ipython notebook

---------------------------------

2. To read users.dat file with delimiter as “::”

users = pd.read\_csv("users.dat",sep="::")

3. To display top 5 data with column header

users.head()

4. To add column name/s in the users.dat file

users = pd.read\_csv("users.dat",sep="::",names = ['userID','gender','age','occupation','zipcode'])

5. To display top 5 data with column header

users.head()

-----------------------------

6. To read movies.dat file with delimiter as “::”

movies = pd.read\_csv("movies.dat",sep="::")

7. To display top 5 data with column header

movies.head()

8. To add column name/s in the movies.dat file

movies = pd.read\_csv("movies.dat",sep="::",header=None,names=['MovieID','MovieName','GenreFull'])

9. To display top 5 data with column header

movies.head()

-----------------------------------------------

10. To read ratings.dat file with delimiter as “::”

ratings = pd.read\_csv("ratings.dat",sep="::")

11. To display top 5 data with column header

ratings.head()

12. To add column name/s in the ratings.dat file

ratings = pd.read\_csv("ratings.dat",sep="::",header=None, names = ['UserID','MovieID','Rating','TimeStamp'])

13. To display top 5 data with column header

ratings.head()

----------------------------------------------

----------Problem Solving--------

1. To split data

*###credit: stackoverflow*

**def** tidy\_split(df, column, sep='|', keep=False):

*"""*

*Split the values of a column and expand so the new DataFrame has one split*

*value per row. Filters rows where the column is missing.*

*Params*

*------*

*df : pandas.DataFrame*

*dataframe with the column to split and expand*

*column : str*

*the column to split and expand*

*sep : str*

*the string used to split the column's values*

*keep : bool*

*whether to retain the presplit value as it's own row*

*Returns*

*-------*

*pandas.DataFrame*

*Returns a dataframe with the same columns as `df`.*

*"""*

indexes = list()

new\_values = list()

df = df.dropna(subset=[column])

**for** i, presplit **in** enumerate(df[column].astype(str)):

values = presplit.split(sep)

**if** keep **and** len(values) > 1:

indexes.append(i)

new\_values.append(presplit)

**for** value **in** values:

indexes.append(i)

new\_values.append(value)

new\_df = df.iloc[indexes, :].copy()

new\_df[column] = new\_values

**return** new\_df

1. To split genres

movies\_splitGenre = tidy\_split(movies,'GenreFull',sep="|")

1. To check top 5 data with header

movies\_splitGenre.head()

1. Merge movies and ratings using inner join

movieRating = movies\_splitGenre.merge(ratings,how='inner',on = 'MovieID')

1. To show top 5 data with header

movieRating.head()

1. GenreFull column and rating column count size and mean od each genre

GenreRatingStats = movieRating.groupby('GenreFull').agg({'Rating': [np.size, np.mean]})

1. To display top 5 data with header

GenreRatingStats.head()

1. To display genre ratings stats minimum value

GenreRatingStats['Rating','size'].min()

1. Sorting the values

GenreRatingStats.sort\_values([('Rating','mean')],ascending=False)[:10]

1. Plotting

%**matplotlib** inline

1. To display histogram plot

users.age.plot.hist(bins=30)

plt.title("Distribution of users' ages")

plt.ylabel('count of users')

plt.xlabel('age');

1. To display count in users

users.count()

1. To display age >10 and <15

users[(users['age']>10) & (users['age']<15)]