Data Structuring_Part 2

March 1, 2021

Objectives

1. Structure the data for association between each individual meal and: (a) seafood item (b) side dish items (c) seafood item descriptions

[10]:
```
import pandas as pd
import numpy as np
import re

#Read filtered dataframe
nhanes = pd.read_pickle('../../Data/nhanes_post.pkl')

'''
This section structures the data by meal. First using seafood grouping, then␣
↪side dish grouping
'''

#Obtain only required variables
group_nhanes = nhanes[['SEQN', 'DR1.030Z', 'DR1.020', 'DESCRIPTION', 'species']]
#Define the grouping key by meal. Participant ID, Meal ID, and time of␣
↪consumption
meal_key = ['SEQN', 'DR1.030Z', 'DR1.020']

'''
Seafood grouping
'''

#Obtain seafood items
sf_nhanes = group_nhanes[group_nhanes['species'].notna()]

#Group the seafood df by meal
sf_meal_group = sf_nhanes.groupby(meal_key)

#Obtain the unique seafood item for each meal
sf_group = sf_meal_group.apply(lambda x: x['species'].unique())
sf_group = sf_group.apply(pd.Series)

#Rename the series columns and convert both grouping indecies to columns
```

```
sf_group = sf_group.rename({0: 'SF1', 1: 'SF2', 2: 'SF3', 3: 'SF4', 4: 'SF5', 5:
 ↪ 'SF6', 6: 'SF7'}, axis=1)
sf_group.reset_index(level=0, inplace=True)
sf_group.reset_index(level=0, inplace=True)
sf_group.reset_index(level=0, inplace=True)

#Obtain the seafood item count in each column. Result can be used as a␣
 ↪statistic to count
#the number of seafood species per meal
meal_fish_count = sf_group.count()


'''
Seafood description grouping
'''

#Group the seafood df by meal
sf_meal_group = sf_nhanes.groupby(meal_key)

#Obtain the unique seafood item for each meal
sf_des_group = sf_meal_group.apply(lambda x: x['DESCRIPTION'].unique())
sf_des_group = sf_des_group.apply(pd.Series)

#Rename the series columns and convert both grouping indecies to columns
sf_des_group = sf_des_group.rename({0: 'SFD1', 1: 'SFD2', 2: 'SFD3', 3: 'SFD4',␣
 ↪4: 'SFD5', 5: 'SFD6',
                                    6: 'SFD7', 7: 'SFD8', 8: 'SFD9'}, axis=1)
sf_des_group.reset_index(level=0, inplace=True)
sf_des_group.reset_index(level=0, inplace=True)
sf_des_group.reset_index(level=0, inplace=True)


'''
Side dish grouping
'''

#Obtain non-seafood items
not_sf_group = group_nhanes[group_nhanes['species'].isnull()]

#Group the side dish df by meal
not_sf_group = not_sf_group.groupby(meal_key)

#Obtain the unique side dish descriptions for each meal
not_sf_group = not_sf_group.apply(lambda x: x['DESCRIPTION'].unique())

#Rename the series columns and convert both grouping indecies to columns
not_sf_group = not_sf_group.apply(pd.Series)
```

```
not_sf_group = not_sf_group.rename({0: 'SD1', 1: 'SD2', 2: 'SD3', 3: 'SD4', 4:␣
 ↪'SD5', 5: 'SD6', 6: 'SD7',
                  7: 'SD8', 8: 'SD9', 9: 'SD10', 10: 'SD11', 11: 'SD12', 12:␣
 ↪'SD13', 13: 'SD14',
                  14: 'SD15', 15: 'SD16', 16: 'SD17', 17: 'SD18', 18: 'SD19',␣
 ↪19: 'SD20', 20: 'SD21',
                  21: 'SD22'}, axis=1)
not_sf_group.reset_index(level=0, inplace=True)
not_sf_group.reset_index(level=0, inplace=True)
not_sf_group.reset_index(level=0, inplace=True)

#Obtain the first word in description item for each column
for i in range(22):
    idx_string = 'SD' + str(i+1)
    not_sf_group[idx_string] = not_sf_group[idx_string].fillna('None')
    not_sf_group[idx_string] = not_sf_group[idx_string].apply(lambda x: re.
 ↪search(r'^([^,])+', x).group(0) if re.search((r','), x) else x)

#Re-apply NaNs for counting purposes
not_sf_group = not_sf_group.replace('None', np.nan)

#Obtain count of side dish item in each column. This can be used as a statistic␣
 ↪to describe
#the number of side dishes per meal.
side_dish_count = not_sf_group.count()


#Join the seafood species, seafood description, and derived side dish in a␣
 ↪structured dataframe
df1 = pd.merge(sf_group, not_sf_group, how='left', on=meal_key)
df_final = pd.merge(df1, sf_des_group, how='left', on=meal_key)

print(df_final.head())
```

```
   DR1.020  DR1.030Z   SEQN         SF1      SF2  SF3  SF4  SF5  SF6  SF7  \
0    82800         3  31131        fish      NaN  NaN  NaN  NaN  NaN  NaN
1    60300         2  31135        fish      NaN  NaN  NaN  NaN  NaN  NaN
2    90000         3  31139      shrimp      NaN  NaN  NaN  NaN  NaN  NaN
3    63900         2  31142  tuna-mixed      NaN  NaN  NaN  NaN  NaN  NaN
4    86400         3  31152      shrimp  seafood  NaN  NaN  NaN  NaN  NaN

   … SD22                                                SFD1  \
0  …  NaN  fish stick, patty, or fillet, ns as to type, f…
1  …  NaN  fish stick, patty, or fillet, ns as to type, b…
2  …  NaN                              lo mein, with shrimp
3  …  NaN          tuna, canned, ns as to oil or water pack
4  …  NaN                              lo mein, with shrimp
```

```
                                                  SFD2 SFD3 SFD4 SFD5 SFD6 SFD7  \
0                                                  NaN  NaN  NaN  NaN  NaN  NaN
1                                                  NaN  NaN  NaN  NaN  NaN  NaN
2                                                  NaN  NaN  NaN  NaN  NaN  NaN
3                                                  NaN  NaN  NaN  NaN  NaN  NaN
4  seafood soup with vegetables (including carrot…  NaN  NaN  NaN  NaN  NaN

  SFD8 SFD9
0  NaN  NaN
1  NaN  NaN
2  NaN  NaN
3  NaN  NaN
4  NaN  NaN

[5 rows x 41 columns]
```

Conclusions

The final dataframe contains an observation row for each meal, based on the participant ID number and meal name.