Data Wrangling

Hospital Compare Data

Hanieh Mirzaee

Data Import

The data for this project was manually downloaded from Medicare.in a zip format. It contains 68 flat files providing information on Medicare hospitals. For proper data importing using the Pandas read_csv, I used the "ISO-8859-1" encoding. An example is given below:

```
# Load 'HCAHPS - Hospital.csv' which contains the summary star ratings
d = os.path.dirname(os.getcwd())
data_file = os.path.join(d, 'data','raw','HCAHPS - Hospital.csv')
survey_df = pd.read_csv(data_file, encoding = "ISO-8859-1")
```

Detailed information on the files and description of measures are provided in the pdf file that accompanies the raw data. In the following I briefly discuss the .csv files I chose for data wrangling and constructing the final Pandas data frame along with major cleaning steps. Further detail is provide in the relevant notebook.

HCAHPS - Hospital.csv

This file contains the survey questions and the summary star ratings computed from the patients' answers. These ratings which are computed based on the patient survey results are available under the 'Patient Survey Star Rating' column. I extracted the corresponding rows, therefore removing other unnecessary values in that column, and renamed the column to 'Summary Star Rating' in the resulting data frame. The values, which are integers between 1 and 5 inclusive, will be used as labels later on for predictive modelling. After using a pie chart to understand the proportion of the ratings, I noticed that there are around 1322 'Not Available' values.

Upon further examination I realized that these values were due to insufficient number of survey results for which a rating could not be calculated. Therefore, I removed those hospitals.

Hospital General Information.csv

This file contains some general information about the hospitals, such as the city, state, ZIP code, hospital type, and ownership. It additionally includes national comparisons regarding measures such as safety of care, timeliness of care, efficient use of medical imaging among others. I selected the relevant columns and performed a merge with the initial data frame.

```
Data columns (total 16 columns):
Provider ID                                                  3490 non-null int64
Hospital Name                                                3490 non-null object
Summary star rating                                          3490 non-null object
ZIP Code                                                     3490 non-null int64
City                                                         3490 non-null object
State                                                        3490 non-null object
Hospital Type                                                3490 non-null object
Hospital Ownership                                           3490 non-null object
Emergency Services                                           3490 non-null object
Meets criteria for meaningful use of EHRs                    3464 non-null object
Mortality national comparison                                3490 non-null object
Safety of care national comparison                           3490 non-null object
Readmission national comparison                              3490 non-null object
Effectiveness of care national comparison                    3490 non-null object
Timeliness of care national comparison                       3490 non-null object
Efficient use of medical imaging national comparison         3490 non-null object
dtypes: int64(2), object(14)
memory usage: 463.5+ KB
None
```

Remaining flat files

Among the remaining flat files I picked the hospital specific .csv files (most have the –hospital in their file names). These files generally contain a 'Measure ID' and a 'Score' column. I looped over the unique Measure Ids from these columns, for each measure I extracted the rows containing only that measure, renamed the Score column to that measure and merged with the original data frame. In other words in the updated data frame each measure represents a column whose values are the corresponding scores. The flat files are:

1. Complications and Deaths- Hospital.csv
2. Healthcare associated infections- Hospital.csv
3. Hospital Returns - Hospital.csv
4. Medicare Hospital Spending per Patient.csv
5. Outpatient Imaging Efficiency- Hopital.csv
6. Structural Measures- Hospital.csv
7. Timely and Effective Care-Hospitals.csv

To merge these additional measures a left outer join was necessary to retain all the hospitals even if not all the measures was reported for them.

Further cleaning steps

I replaced the 'Not available' values with np.nan, and made sure all the numeric columns have numeric types. Finally I wrote the new data frame into a new processed .csv file.