Ginny Zhu
December 23, 2017

Data Wrangling: Json Mini Project

This mini project is to handle a json file (world_bank_projects.json) of world bank funded projects. The problems to be answered are:

1. Find the 10 countries with most projects
2. Find the top 10 major project themes (using column 'mjtheme_namecode')
3. In 2. above you will notice that some entries have only the code and the name is missing. Create a dataframe with the missing names filled in.

Procedures:

Detailed python code and outputs are included in the file 'DataWrangling_Json.ipynb'

First we load the file with Pandas to a data frame 'json_df' and inspect the shape, column names and head. In general the data frame has 500 records, with 50 attributes including id, approval information, country information, project name, theme, status etc. based on column names.

In order to find the 10 countries with most projects, we could group the data frame records by country names, and count how many records each country has, and order them from the most to the fewest. Top 10 in the output should be the answer to question 1.

Here is the main result:

```
    countryname
People's Republic of China                    19
Republic of Indonesia                         19
Socialist Republic of Vietnam                 17
Republic of India                             16
Republic of Yemen                             13
Nepal                                         12
People's Republic of Bangladesh               12
Kingdom of Morocco                            12
Africa                                        11
```

```
Republic of Mozambique                    11
```

For question 2, I first experimented with using .head(10) method on the sub-data frame of. json_df[[‘mjtheme_namecode’]], but it didn’t give me the full display of records (there were ‘…’ in the outputs as shown in the .ipynb file). So I used the loop to print the first 10 elements of ‘mjtheme_namecode’ series to display the full content of the each record. Each records is a list of varying lengths of dictionary elements that contain code and name information. The top 10 are:

```
[{u'code': u'8', u'name': u'Human development'}, {u'code': u'11',
u'name': u’'}]
[{u'code': u'1', u'name': u'Economic management'}, {u'code': u'6',
u'name': u'Social protection and risk management'}]
[{u'code': u'5', u'name': u'Trade and integration'}, {u'code': u'2',
u'name': u'Public sector governance'}, {u'code': u'11', u'name':
u'Environment and natural resources management'}, {u'code': u'6',
u'name': u'Social protection and risk management'}]
[{u'code': u'7', u'name': u'Social dev/gender/inclusion'}, {u'code':
u'7', u'name': u'Social dev/gender/inclusion'}]
[{u'code': u'5', u'name': u'Trade and integration'}, {u'code': u'4',
u'name': u'Financial and private sector development'}]
[{u'code': u'6', u'name': u'Social protection and risk management'},
{u'code': u'6', u'name': u''}]
[{u'code': u'2', u'name': u'Public sector governance'}, {u'code':
u'4', u'name': u'Financial and private sector development'}]
[{u'code': u'11', u'name': u'Environment and natural resources
management'}, {u'code': u'8', u'name': u''}]
[{u'code': u'10', u'name': u'Rural development'}, {u'code': u'7',
u'name': u''}]
[{u'code': u'2', u'name': u'Public sector governance'}, {u'code':
u'2', u'name': u'Public sector governance'}, {u'code': u'2', u'name':
u'Public sector governance'}]
```

For question 3, it’s logical to assume there are a limited number of code-name pairs according to results in question 2. So in order to fill the missing values, we first construct the code-name pairs based on the information already exists in the data frame. And here is the dictionary that we’ve created:

```
[(u'11', u'Environment and natural resources management'),
 (u'10', u'Rural development'),
 (u'1', u'Economic management'),
 (u'3', u'Rule of law'),
 (u'2', u'Public sector governance'),
 (u'5', u'Trade and integration'),
 (u'4', u'Financial and private sector development'),
 (u'7', u'Social dev/gender/inclusion'),
 (u'6', u'Social protection and risk management'),
 (u'9', u'Urban development'),
 (u'8', u'Human development')]
```

Indeed there are only 11 code-name pairs for the whole dataset. Based on this dictionary, we can then create a new data frame with the missing names filled in. To further check the results, we print the top 10 records of the new data frame to compare with the previous one.

```
[{u’code’: u'8', u'name': u'Human development'}, {u'code': u'11',
u'name': u'Environment and natural resources management'}]
[{u'code': u'1', u'name': u'Economic management'}, {u'code': u'6',
u'name': u'Social protection and risk management'}]
[{u'code': u'5', u'name': u'Trade and integration'}, {u'code': u'2',
u'name': u'Public sector governance'}, {u'code': u'11', u'name':
u'Environment and natural resources management'}, {u'code': u'6',
u'name': u'Social protection and risk management'}]
[{u'code': u'7', u'name': u'Social dev/gender/inclusion'}, {u'code':
u'7', u'name': u'Social dev/gender/inclusion'}]
[{u'code': u'5', u'name': u'Trade and integration'}, {u'code': u'4',
u'name': u'Financial and private sector development'}]
[{u'code': u'6', u'name': u'Social protection and risk management'},
{u'code': u'6', u'name': u'Social protection and risk management'}]
[{u'code': u'2', u'name': u'Public sector governance'}, {u'code':
u'4', u'name': u'Financial and private sector development'}]
[{u'code': u'11', u'name': u'Environment and natural resources
management'}, {u'code': u'8', u'name': u'Human development'}]
[{u'code': u'10', u'name': u'Rural development'}, {u'code': u'7',
u'name': u'Social dev/gender/inclusion'}]
```

```
[{u'code': u'2', u'name': u'Public sector governance'}, {u'code':
u'2', u'name': u'Public sector governance'}, {u'code': u'2', u'name':
u'Public sector governance'}]
```

Here you can see, all the previous “ u’name’: u’ ‘ ” are filled.

Issues:

1. I’m not quite sure about what the ‘top 10 major project themes’ in question 2 refer to? Does ‘top’ simply mean ‘head’ of the data or as in number? (in which case I’ll just create a dictionary of code-count pairs.
2. Is there any better way to display the whole content of each record other than mine? (Even though my code does the work, I’m not quite sure about the quality)