100-Determine_data_by_year

June 22, 2020

1 Final Project: Admission Prediction from NHAMCS

1.1 Data exploration notebook

1.1.1 DS5559: Big Data Analysis

1.1.2 Thomas Hartka, Alicia Doan, Michael Langmayr

Created: 6/21/20

In this notebook we determine which years contain data for certain variable and write this to a CSV.

1.2 Configure

[1]:
```
# set data directory
data_dir = "../data"
```

[2]:
```
# import python libraries
import os
import pandas as pd
import numpy as np
from functools import reduce
```

[3]:
```
# set up pyspark
from pyspark.sql import *
from pyspark.sql import SparkSession
from pyspark.sql.functions import *

spark = SparkSession.builder.getOrCreate()
```

1.3 Read in data

```
[4]: %%time
     NHAMCS = spark.read.parquet(data_dir + "/NHAMCS.2007-2017")
```

```
CPU times: user 2.02 ms, sys: 3.08 ms, total: 5.1 ms
Wall time: 7.07 s
```

```
[5]: NHAMCS.count()
```

```
[5]: 305897
```

```
[7]: # number of columns
     len(NHAMCS.columns)
```

```
[7]: 1219
```

1.4 Look for missing data

```
[8]: %%time
     # create dataframe with year and counts
     years_null = pd.DataFrame(NHAMCS.groupBy("YEAR").agg(count('YEAR').alias('N')).
      ↪collect(), columns=["YEAR","N"])

     # find col
     for col in NHAMCS.columns:
         if col !='YEAR':
             #print(col)
             n = NHAMCS.select('YEAR',col).subtract(NHAMCS.select('YEAR',col).
      ↪dropna()).groupBy("YEAR").agg(count('YEAR')).collect()
             #print(n)
             col_nulls = pd.DataFrame(n,columns=["YEAR",col])
             #print(col_nulls)
             years_null = years_null.merge(col_nulls, how='left', on="YEAR")
```

```
CPU times: user 1min 37s, sys: 1.91 s, total: 1min 39s
Wall time: 51min 31s
```

```
[10]: # change ALL NULL flag from 1 to 0, and NOT ALL NULL from null to 1
      years_data = years_null.replace(1.0,int(0)).fillna(int(1)).astype(int) \
                             .sort_values('YEAR').reset_index(drop=True)
```

```
[11]: years_data
```

```
[11]:    YEAR      N  VMONTH  VYEAR  VDAYR  AGE  ARRTIME  WAITTIME  LOV  RESIDNCE  \
      0  2007  35490       1      1      1    1        1         1    1         1
```

```
1   2008  34134       1      1      1    1        1         1    1         1
2   2009  34942       1      1      1    1        1         1    1         1
3   2010  34936       1      0      1    1        1         1    1         1
4   2011  31084       1      0      1    1        1         1    1         1
5   2012  29453       1      0      1    1        1         1    1         1
6   2013  24777       1      0      1    1        1         1    1         1
7   2014  23844       1      0      1    1        1         1    1         1
8   2015  21061       1      0      1    1        1         1    1         1
9   2016  19467       1      0      1    1        1         1    1         1
10  2017  16709       1      0      1    1        1         1    0         1

    …  EXCHSUM2E  BLANK7  BLANK8  EWHONOTE  EWHOPRACE  EWHOOTHE  EWHOPRACER  \
0   …          0       0       0         0          0         0           0
1   …          0       0       0         0          0         0           0
2   …          0       0       0         0          0         0           0
3   …          0       0       0         0          0         0           0
4   …          1       1       1         1          1         1           1
5   …          0       0       0         0          0         0           0
6   …          0       0       0         0          0         0           0
7   …          0       0       0         0          0         0           0
8   …          0       0       0         0          0         0           0
9   …          0       0       0         0          0         0           0
10  …          0       0       0         0          0         0           0

    EXCHSUM4E  EWHOUNKE  EXCHSUME
0           0         0         0
1           0         0         0
2           0         0         0
3           0         0         0
4           1         1         1
5           0         0         0
6           0         0         0
7           0         0         0
8           0         0         0
9           0         0         0
10          0         0         0

[11 rows x 1220 columns]
```

1.5 Write out varaibles table

[14]:

```
years_data.to_csv("../results/NHAMCS_vars_by_year.csv")
```

[ ]:

```

```