

# Rajeev\_Singleton\_Names

February 10, 2018

## 1 Find singletons

1.0.1 The database only shows names that appear at least five times in a given year. This script will find names that appeared only in a single year.

```
In [30]: last_year = 2017 #change this when Social Security database is updated  
        save_path = "rajeev_data/single_names" # files created by this notebook will be saved
```

```
import time  
import os  
if not os.path.isdir(save_path): # creates path if it does not exist  
    os.makedirs(save_path)
```

```
import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
%run Rajeev_download_process_files.py
```

```
# used to round limit of y axis up to second-most-significant digit  
def determine_y_limit(x):  
    significance = int(floor((log10(x))))  
    val = floor(x / (10 ** (significance - 1))) + 1  
    val = val * (10 ** (significance - 1))  
    return val
```

Data already downloaded.

Data already extracted.

Reading from pickle.

Tail of dataframe 'yob':

Tail of dataframe 'names':

Tail of dataframe 'years':

Tail of dataframe 'yob1900':

Tail of dataframe 'names1900':

Tail of dataframe 'years1900':

```

In [31]: df_oneyear = names[names.year_min == names.year_max]
df_oneyear = df_oneyear[['name', 'sex', 'year_min', 'pct_max']]
df_oneyear.columns = ['name', 'sex', 'year', 'pct']

In [32]: oneyearnames = list(df_oneyear.name.unique())
yobcopy = yob.copy()
yobcopy = yobcopy[yobcopy.name.isin(oneyearnames)]

In [33]: df_oneyear['births'] = 0
for i in range(len(df_oneyear)):
    df_oneyear.births.loc[i] = yobcopy[(yobcopy.name == df_oneyear.name.iloc[i]) &
                                         (yobcopy.sex == df_oneyear.sex.iloc[i])].births

In [34]: df_oneyear.to_pickle(save_path+'/df_oneyear.pickle')

In [35]: df_oneyear = pd.read_pickle(save_path+'/df_oneyear.pickle')

In [36]: df_oneyear.sort_values(by='births', inplace=True, ascending=False)
df_oneyear.head(50)

```

```

Out[36]:

```

	name	sex	year	pct	births
51868	Jocelynmarie	F	2011	0.000285	0
98981	Kinganthony	M	2014	0.000261	0
98979	Nazr	M	1998	0.000262	0
98978	Emareon	M	2007	0.000241	0
98977	Nashan	M	2003	0.000304	0
98976	Yaritza	M	1993	0.000255	0
98975	Rowden	M	2012	0.000264	0
98974	Alaijah	M	2004	0.000252	0
98973	Bean	M	1981	0.000279	0
98972	Keiten	M	2006	0.000292	0
98971	Fennell	M	1960	0.000234	0
98970	Delmonta	M	1998	0.000262	0
98969	Marwaan	M	2016	0.000319	0
98968	Saher	M	2001	0.000258	0
98967	Adrianno	M	2010	0.000418	0
98966	Kingzton	M	2016	0.000319	0
98965	Jociah	M	2014	0.000314	0
98964	Amaire	M	2008	0.000295	0
98963	Connan	M	1985	0.000271	0
98962	Xylar	M	2009	0.000252	0
98961	Azhaan	M	2015	0.000524	0
98980	Quennell	M	2002	0.000258	0
98982	Chiquita	M	1989	0.000300	0
98914	Beckman	M	2016	0.000266	0
98983	Jantsen	M	1989	0.000250	0
99002	Deziah	M	2014	0.000261	0
99001	Yeruchem	M	2007	0.000289	0
99000	Tama	M	2012	0.000317	0

98999	Kymarley	M	2008	0.000245	0
98998	Yahushua	M	2015	0.000262	0
98997	Kenuel	M	2016	0.000266	0
98996	Martravius	M	2006	0.000244	0
98995	Josheua	M	1985	0.000325	0
98994	Irael	M	2012	0.000264	0
98993	Brentten	M	1992	0.000250	0
98992	Zaydren	M	2014	0.000261	0
98991	Koorosh	M	2000	0.000255	0
98990	Darone	M	1990	0.000244	0
98989	Kyann	M	2012	0.000264	0
98988	Raynathan	M	1995	0.000368	0
98987	Jahvel	M	2010	0.000313	0
98986	Quvondo	M	1990	0.000292	0
98985	Lendol	M	1931	0.000481	0
98984	Traice	M	1998	0.000262	0
98960	Xu	M	1988	0.000261	0
98959	Jeshwa	M	2007	0.000241	0
98958	Hafford	M	1916	0.001011	0
98957	Dav	M	1959	0.000234	0
98934	Patrocinio	M	1927	0.000532	0
98933	Chartez	M	2002	0.000258	0

```
In [37]: %matplotlib inline
dictionary = {0:1000, 1:20, 2:15, 3:0, 4:5}
xmax = df_oneyear.year.max()
plt.figure() # <- makes a new figure and sets it active (add this)
plt.hist(list(df_oneyear.year),xmax) # <- finds the current active axes/figure and pl
plt.title('Histogram of names appearing only once in the Social Security database')
plt.xlabel('Year')
plt.ylabel('Number of names')
plt.show()
```

Histogram of names appearing only once in the Social Security database

