Using Pandas for Data Reduction

RONALD LOPEZ

Outline

- Problem at hand
  - Where do students go?
  - The challenges of answering this question
  - Data that is available
- How to approach solving the problem with Pandas
  - Why use Pandas
  - Which built-in functions can we use to solve the problem
- Use Pandas to answer the question

Where Do Students Go?

- What happens to our students?
  - Do some transfer to a 4-year university?
  - Do they pursue a graduate degree?
  - Do they enroll at another 2-year institution?
  - Did they enroll in a vocational program?

Why We Don't Know? And How Can We Find Out?

- Students are not required to inform us
  - Immediately transfer when they meet requirements
  - Stop enrolling
- How do schools get this data?
  - Post-graduation surveys
    - Send out surveys 1-year, 3-years, 5-years, and 10-years after

National Student Clearinghouse (NSC)

- Higher education institutions that receive or accept Federal Financial Aid
  - 98% of all students in public and private U.S. institutions
- Provide enrollment information to Department of Education's National Student Loan Data System (NSLDS) and education finance industry
  - Meet the requirements for deferment or forbearance

NSC Student Tracker

- "Research service that provides continuing collegiate enrollment and degree information on your current and former students as well as your former admission applicants"
- Report that tells you students' enrollment history
- Matches students records by birthdate and name

Example Student Tracker Report

First Name	Last Name	Student ID	College Name	Term Start Date	Term End Date	College Sequence	Graduated?	CIP	Graduation Date
Ronald	Lopez	0000001	Evergreen Valley	8/12/2015	12/3/2015	1			
Ronald	Lopez	0000001	Evergreen Valley	1/4/2016	5/15/2016	1			
Ronald	Lopez	0000001	Evergreen Valley	1/4/2016	5/15/2016	1			
Ronald	Lopez	0000001	Evergreen Valley	8/12/2016	12/3/2016	1			
Ronald	Lopez	0000001	Evergreen Valley	1/4/2017	5/15/2017	1			
Ronald	Lopez	0000001	Evergreen Valley				Y	111.1	6/8/2017

Problems With NSC

- Files are too large
  - You can't open them in Excel
- Files contains a lot of noise
- Requires a lot of processing

Desired Report

First Name	Last Name	Student ID	College Name	First Term Start Date	Last Term End Date	College Sequence	Graduated?	CIP	Graduation Date
Ronald	Lopez	0000001	Evergreen Valley	8/12/2015	5/15/2017	1	Y	111.1	6/8/2017

Python To The Rescue

- You can accomplish this with built-in functions
  - read_csv()
  - to_datetime()
  - groupby()
  - min()
  - max()
  - merge()
  - drop_duplicates()
  - loc

First Step: Loading The Data

```
1    import pandas as pd
2
3    data = pd.read_csv("File Path of the CSV")
4
5
6
7    data['Enrollment Begin'] = pd.to_datetime(data['Enrollment Begin'], format='%Y%m%d')
8
9    data['Enrollment End'] = pd.to_datetime(data['Enrollment End'], format='%Y%m%d')
10
11   data['Graduation Date'] = pd.to_datetime(data['Graduation Date'], format='%Y%m%d')
12
```

to_datetime()

- Only using two parameters
  - arg
    - This is the field we're passing
  - format
    - "%Y%m%d"
    - Turns the date into 2017/01/24
    - http://strftime.org/
  - Documentation

Group Enrollment Records by ID & College

```
1    group = data.groupby(['Requester Return Field', 'College Code/Branch'], as_index=False)
2
3
```

- What does Groupby do?
  - Group series using mapper (dict or key function, apply given function to group, return result as series) or by a series of columns
  - Documentation

Example

ID	College	Term Start Date	Term End Date	College Sequence
00001	Evergreen	9/1/2015	12/12/2015	1
	Evergreen	1/1/2016	5/25/2016	1
	Evergreen	9/1/2016	12/12/2016	1
00001	San Jose State	1/25/2016	6/3/2016	2
	San Jose State	9/5/2016	12/15/2017	2
	San Jose State	1/24/2018	5/24/2018	2
000002	De Anza	9/1/2015	12/12/2015	1
	De Anza	1/1/2016	5/25/2016	
000002	Foothill	9/1/2016	12/12/2016	2
000002	San Jose State	1/25/2016	6/3/2016	3
	San Jose State	9/5/2016	12/15/2017	

First Step: Find The Earliest and Latest Enrollment Dates

```
#Finding the earliest and latest enrollment date at each college
groupmin = group['Enrollment Begin'].min()
groupmax= group['Enrollment End'].max()

#Renaming the columns
groupmin = groupmin.rename( columns={"Enrollment Begin": "Earliest Enrollment"})
groupmax = groupmax.rename( columns={"Enrollment End": "Latest Enrollment"})
```

- Min and Max also work on dates!

Next Step: Join The Data

```
1
2   test = pd.merge(data, groupmin, how = 'left', on = ['Requester Return Field', 'College Code/Branc
3
4   test = pd.merge(test, groupmax, how = 'left', on = ['Requester Return Field', 'College Code/Branc
5
```

- Joining the earliest and latest enrollment records by ID and College

Result

ID	College	Term Start Date	Term End Date	College Sequence	Earliest Enrollment	Latest Enrollment
00001	Evergreen	9/1/2015	12/12/2015	1	9/1/2015	12/12/2016
	Evergreen	1/1/2016	5/25/2016	1	9/1/2015	12/12/2016
	Evergreen	9/1/2016	12/12/2016	1	9/1/2015	12/12/2016
00001	San Jose State	1/25/2016	6/3/2016	2	1/25/2016	5/24/2018
	San Jose State	9/5/2016	12/15/2017	2	1/25/2016	5/24/2018
	San Jose State	1/24/2018	5/24/2018	2	1/25/2016	5/24/2018
000002	De Anza	9/1/2015	12/12/2015	1	9/1/2015	5/25/2016
	De Anza	1/1/2016	5/25/2016	1	9/1/2015	5/25/2016
000002	Foothill	9/1/2016	12/12/2016	2	9/1/2016	12/12/2016
000002	San Jose State	1/25/2016	6/3/2016	3	1/25/2016	12/15/2017
	San Jose State	9/5/2016	12/15/2017	3	1/25/2016	12/15/2017

Identifying Graduates

```
1   grads = data.loc[data['Graduated?'] =='Y']
2
3   grads = grads[['Requester Return Field', 'College Code/Branch', 'Graduated?', 'Graduation Date', 'D
4
5   #Dropping duplicate columns before merging dataset with graduate dataset
6   test = test.drop(['Enrollment Begin', 'Enrollment End', 'Graduated?', 'Graduation Date', 'Degree T
7
8   #Joining the graduation records to the main dataset
9   test = pd.merge(test, grads, how = 'left', on = ['Requester Return Field', 'College Code/Branch'])
10
```

- .loc vs .iloc
  - .loc for labels
  - .iloc for integer base positions

Result

ID	College	Term Start Date	Term End Date	College Sequence	Earliest Enrollment	Latest Enrollment	Graduated?
00001	Evergreen	9/1/2015	12/12/2015	1	9/1/2015	12/12/2016	Yes
	Evergreen	1/1/2016	5/25/2016	1	9/1/2015	12/12/2016	Yes
	Evergreen	9/1/2016	12/12/2016	1	9/1/2015	12/12/2016	Yes
00001	San Jose State	1/25/2016	6/3/2016	2	1/25/2016	5/24/2018	
	San Jose State	9/5/2016	12/15/2017	2	1/25/2016	5/24/2018	
	San Jose State	1/24/2018	5/24/2018	2	1/25/2016	5/24/2018	
000002	De Anza	9/1/2015	12/12/2015	1	9/1/2015	5/25/2016	Yes
	De Anza	1/1/2016	5/25/2016	1	9/1/2015	5/25/2016	Yes
000002	Foothill	9/1/2016	12/12/2016	2	9/1/2016	12/12/2016	
000002	San Jose State	1/25/2016	6/3/2016	3	1/25/2016	12/15/2017	
	San Jose State	9/5/2016	12/15/2017	3	1/25/2016	12/15/2017	

Remove Duplicates

```
1   final_df = test.drop_duplicates(['Requester Return Field', 'College Cod
2
```

ID	College	College Sequence	Earliest Enrollment	Latest Enrollment	Graduated?
00001	Evergreen	1	9/1/2015	12/12/2016	Yes
00001	San Jose State	2	1/25/2016	5/24/2018	
000002	De Anza	1	9/1/2015	5/25/2016	Yes
000002	Foothill	2	9/1/2016	12/12/2016	
000002	San Jose State	3	1/25/2016	12/15/2017	

How DO We Find Direct Transfers?

- Can be easily done with .loc and .isin functions
- No need for list comprehensions

Finding EVC Transfers and Top Locations

```
1   #First college in EVC
2   evc_cohort = final_df.loc[(final_df['College Code/Branch'] =='012452-00') & (final_df['College Seque
3
4   #Transferred directly to a 4-year
5   transfers = final_df.loc[(final_df['2-year / 4-year'] == "4") & (final_df['College Sequence'] ==2)]
6
7
8   evc_transfers = transfers.loc[transfers["Requester Return Field"].isin(evc_cohort["Requester Return
9
10  colleges = evc_transfers.groupby(['College Name'], as_index=False)["Requester Return Field"].agg('co
11
12  top_colleges = colleges.sort_values(['Requester Return Field'], ascending = False)[:10]
13
```

Top Ten Transfer Schools

Institution	Number of Transfers
San Jose State University	1,211
University of Phoenix	131
San Francisco State	97
University of California, Davis	82
University of California, Santa Cruz	64
California State University, East Bay	57
National Hispanic University	43
University of California, Irvine	43
University of California, San Diego	39
Santa Clara University	36