

Application of Python Scripting for Major Work Destination Analysis

Yunkyung Choi, Jung Seo, and Tom Vo

Research & Analysis

Southern California Association of Governments

2014 ESRI User Conference | July 14-18, 2014 | San Diego Convention Center



Contents

§ Introduction of SCAG

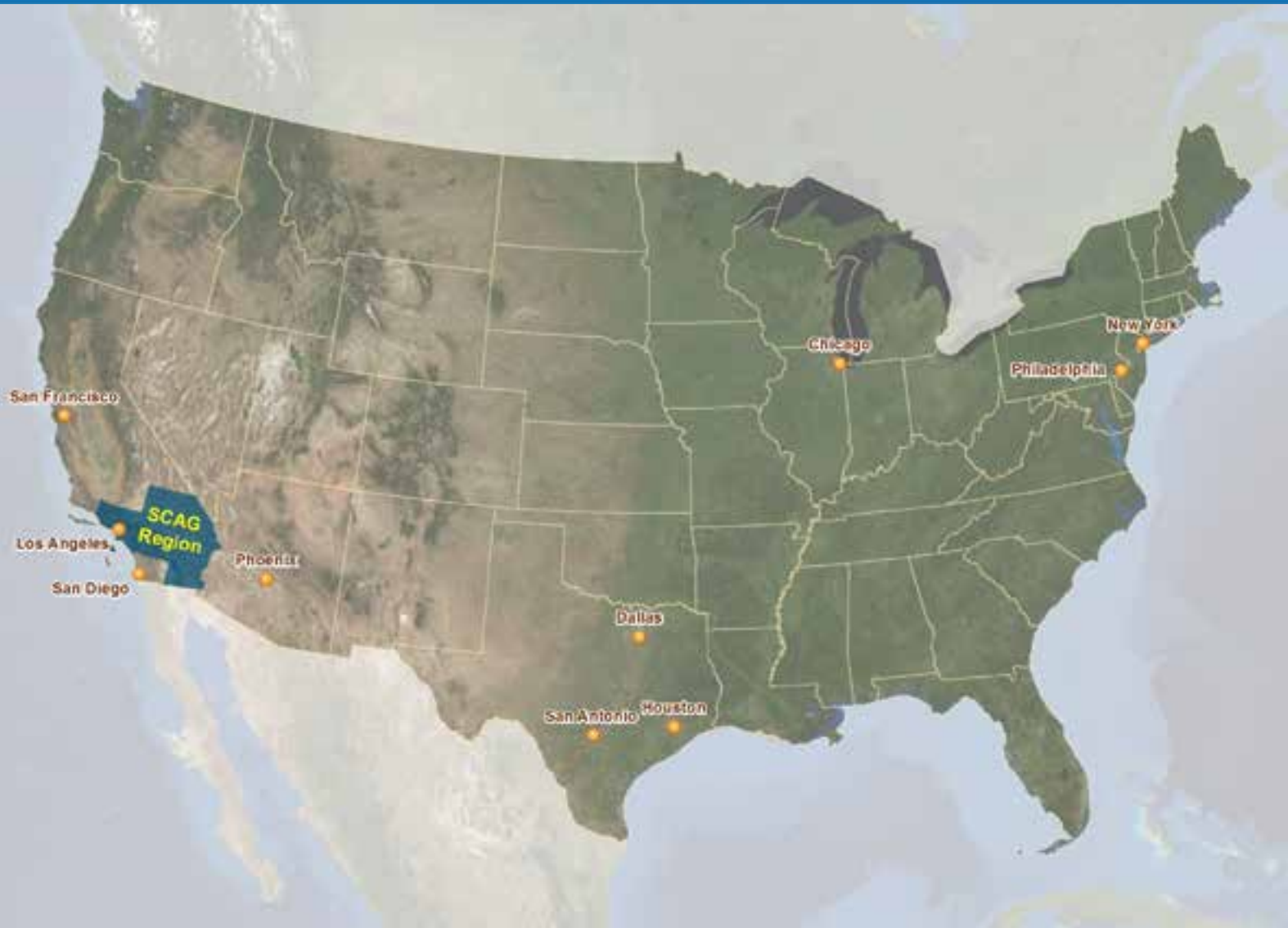
§ Objectives

§ Methodology

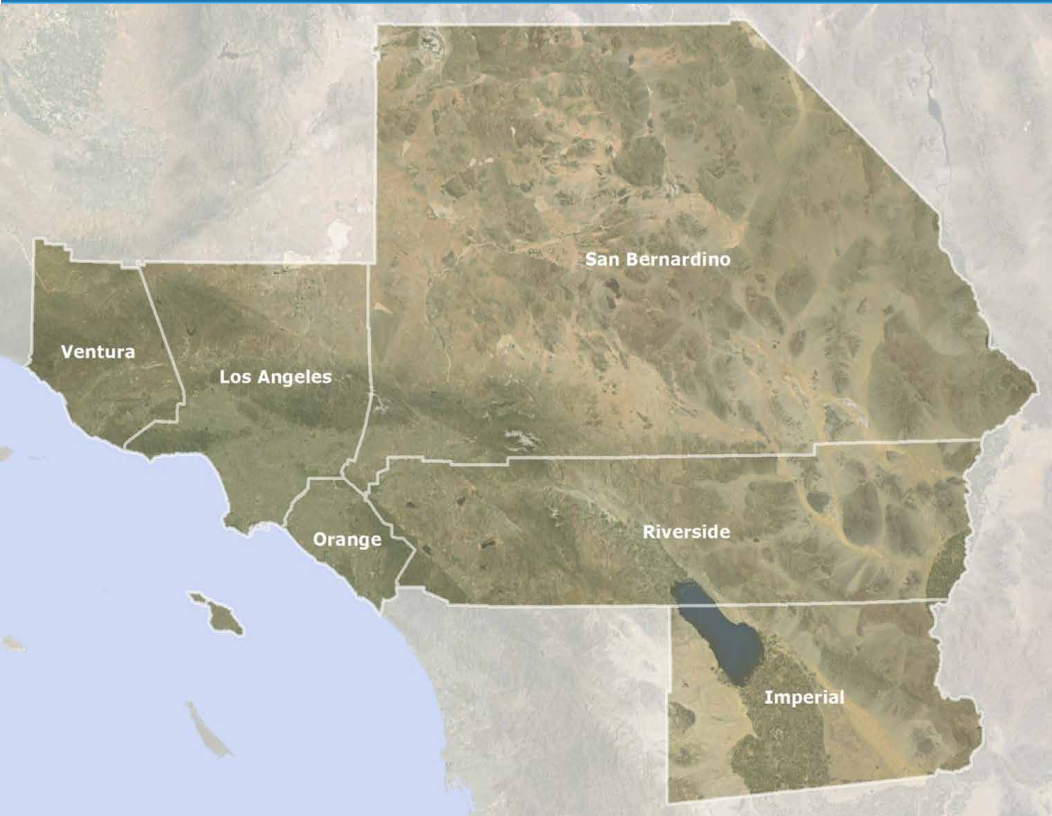
§ Results

§ Conclusions

Overview of SCAG



Quick Facts of SCAG



- § Nation's largest Metropolitan Planning Organization (MPO)
- § 6 counties and 191 cities
- § 15 sub-regions
- § 18.4 million people (2012)
- § 38,000 square miles
- § 16th largest economy in the world (GRP: \$924 Billion in 2012)

Objectives

- § Identify the major work destinations for each jurisdiction in the SCAG region
 - **191 cities and 6 counties**
- § Visualize the spatial patterns of the major work destinations for each jurisdiction
 - To understand where residents of each jurisdiction are employed
- § Provide informational data resources to local jurisdictions for planning purposes
 - E.g. 2013 Local Profiles: Planning data reports

Methodology for identifying work destination

§ Using *OnTheMap* application

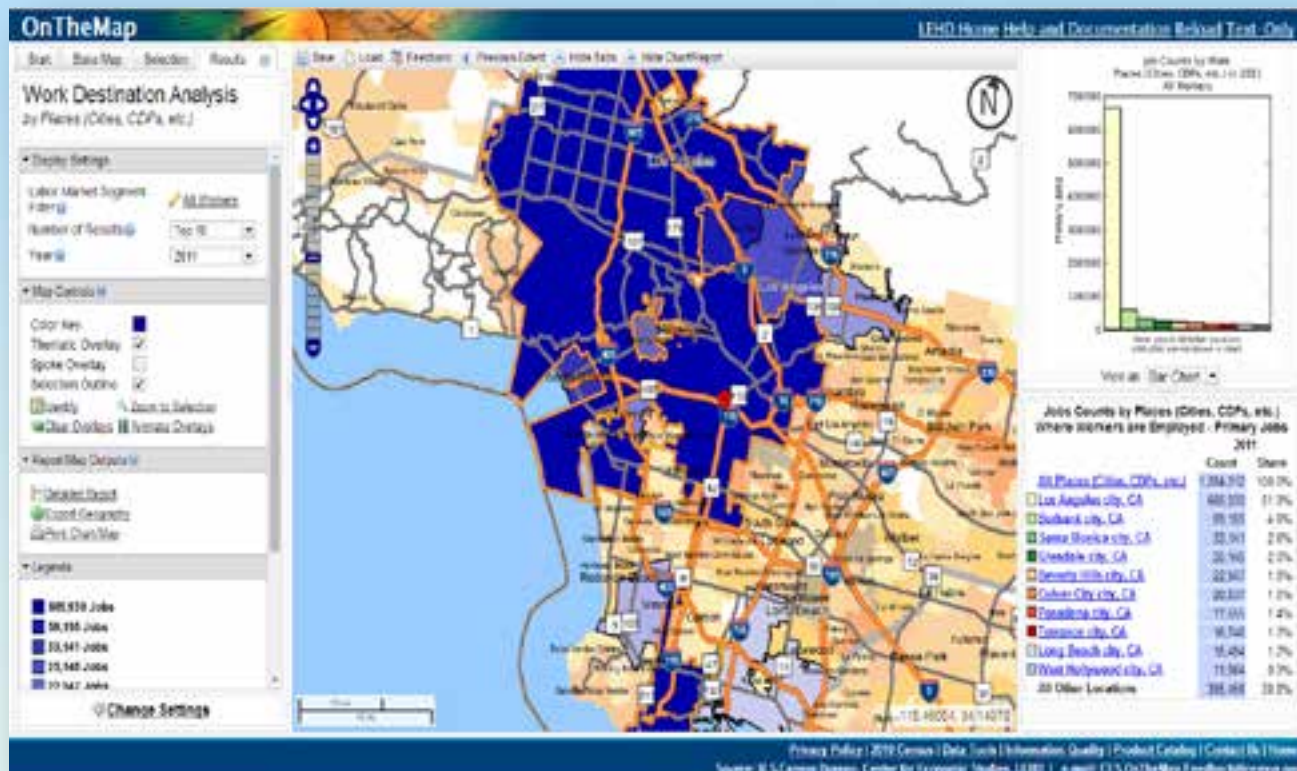
- Available on <http://onthemap.ces.census.gov/>

§ Using SAS (Statistical Analysis System) with LEHD Origin-Destination Employment Statistics (*LODES*) raw datasets

- Available on <http://lehd.ces.census.gov/data/>

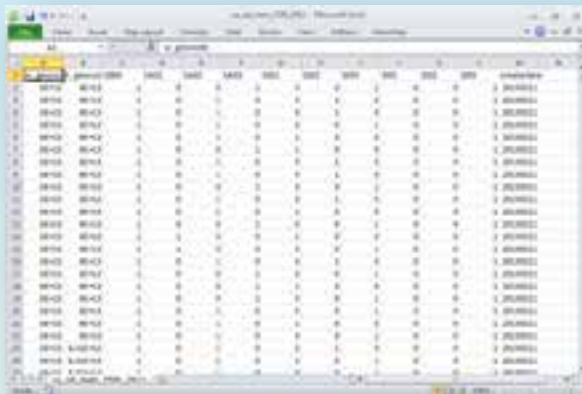
OnTheMap application

- § Web-based mapping and reporting application
- § Based on 2002-2011 *LODES* data

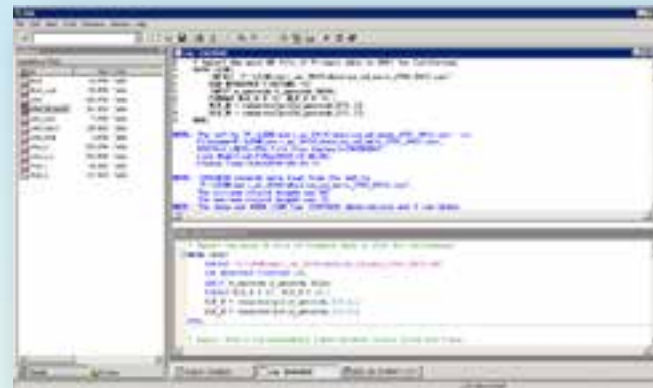


Programming scripts with *LODES* raw datasets

- § State-based annual data from 2002-2011
- § 3 types of CSV data files
 - Origin-Destination (OD)
 - Residence Area Characteristics (RAC)
 - Workplace Area Characteristics (WAC)
- § Enumerated by 2010 census blocks



A screenshot of a spreadsheet application displaying a large table of data. The table has many columns and rows, with some cells containing numerical values and others containing text. The spreadsheet is titled 'LODES' and is open in a window.



A screenshot of a code editor showing a script with various lines of code. The code is written in a programming language and includes comments and function calls. The code editor is titled 'LODES' and is open in a window.

Pros and Cons

	<i>OnTheMap</i>	Programming Scripts
Pros	<ul style="list-style-type: none">§ Easy-to-use interface and able to visualize data instantly§ Support analyses at multiple levels of geography	<ul style="list-style-type: none">§ Able to manipulate data for multiple areas at once§ Improve efficiency of managing and processing big data
Cons	<ul style="list-style-type: none">§ Not able to search/download for multiple locations	<ul style="list-style-type: none">§ Incorporate with ArcGIS to visualize the data

LODES raw data manipulation using SAS

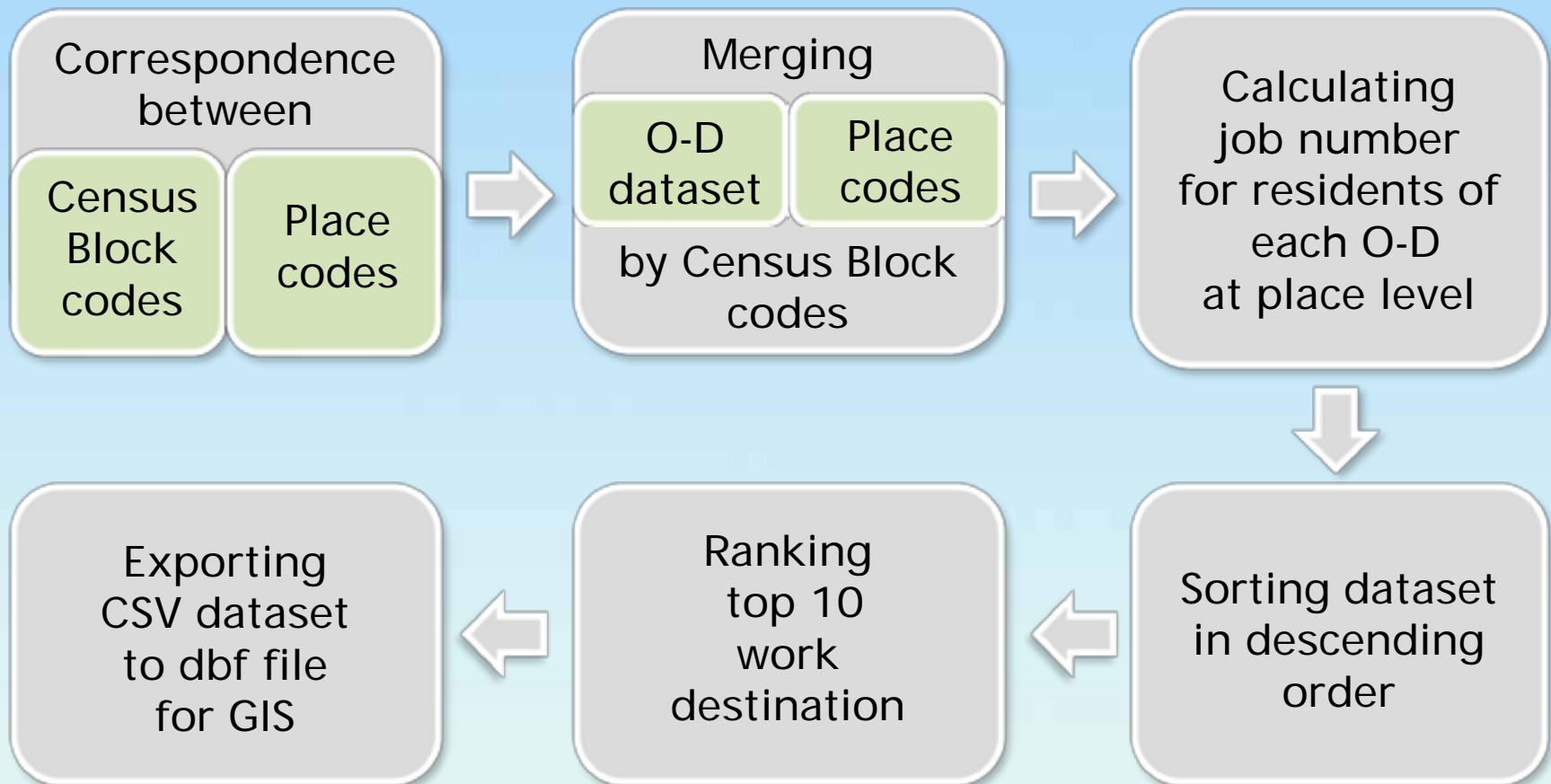
§ LODES raw dataset description

- LODES CA OD dataset (Version 7, 2011)
- 3 Variables: w_geocode, h_geocode, S000

Pos	Variable	Type	Explanation
1	w_geocode	Char15	Workplace Census Block Code
2	h_geocode	Char15	Residence Census Block Code
3	S000	Num	Total number of jobs
4	SA01	Num	Number of jobs of workers age 29 or younger
5	SA02	Num	Number of jobs of workers age 30 to 54
6	SA03	Num	Number of jobs of workers age 55 or older
7	SE01	Num	Number of jobs with earnings \$1250/month or less
8	SE02	Num	Number of jobs with earnings \$1251/month to \$3333/month
9	SE03	Num	Number of jobs with earnings greater than \$3333/month
10	SI01	Num	Number of jobs in Goods Producing industry sectors
11	SI02	Num	Number of jobs in Trade, Transportation, and Utilities industry sectors
12	SI03	Num	Number of jobs in All Other Services industry sectors
13	createdate	Char	Date on which data was created, formatted as YYYYMMDD

LODES raw data manipulation using SAS (cont.)

§ O-D Analysis Processing Steps



LODES raw data manipulation using SAS (cont.)

The screenshot displays the SAS software interface. The main window shows a log of SAS code and its execution output. The code includes data import, processing, and export steps. The output provides details on the number of observations and variables, and the time taken for each step.

```
Log - (Untitled)
120  PERRIN_CITY_H CITTYMVE_H DES_1 JOB_1 DES_2 JOB_2 DES_3 JOB_3 DES_4 JOB_4 DES_5 JOB_5
1201 DES_6 JOB_6 DES_7 JOB_7 DES_8 JOB_8 DES_9 JOB_9 DES_10 JOB_10;
121  SET LEHD_OD_TOP10;
122  RUN;

NOTE: There were 352 observations read from the data set WORK.LEHD_OD_TOP10.
NOTE: The data set WORK.LEHD_OD_TOP10 has 352 observations and 22 variables.
NOTE: DATA statement used (Total process time):
      real time           0.01 seconds
      cpu time            0.01 seconds

123
124  * Export data set;
125  PROC EXPORT OUTFILE='P:\LEHD\esri_uc_2014\docs\lehd_od_top10.dbf'
126  DATA=LEHD_OD_TOP10
127  DBMS=ODBC REPLACE;
128  RUN;

NOTE: Exported 22 columns and 352 rows of data to the dBase table file.
NOTE: 'P:\LEHD\esri_uc_2014\docs\lehd_od_top10.dbf' file was successfully created.
NOTE: PROCEDURE EXPORT used (Total process time):
      real time           0.03 seconds
      cpu time            0.01 seconds

LEHD_OD_PERRIN_CITY
* Import the esri OD files of Perryway Jobs in 2011 for California.
DATA LEHD;
  INFILE 'P:\LEHD\esri_uc_2014\docs\oa_od_main_3701_2011.csv'
  DSD MISSOVER FIRSTOBS=2;
  INPUT v geocode h geocode $COO;
  FORMAT BLK_W $ 15. BLK_H $ 15.;
  BLK_W = compress(put(v_geocode,15.));
  BLK_H = compress(put(h_geocode,15.));
RUN;

* Import BGA's correspondence table between Census Block and Place;
PROC IMPORT DATAFILE='P:\LEHD\esri_uc_2014\docs\11_2010_wrag_tabblockto_place.dbf'
  OUT=BLOCK_CODE
  USEDFST Replace;
RUN;

* Create Block ID variables for merge purpose;
DATA BLOCK;
  SET BLOCK_CODE;
  FORMAT BLK_W $ 15. BLK_H $ 15.;
```

Visualizing Major Work Destinations

§ Using *ESRI ArcGIS* application

- *Data Driven Pages* – To create a multi-page map series from a single map document

§ Using *Python* programming language

- *Python* – Interpreted, object-oriented, high-level general-purpose programming language

Arcpy Module

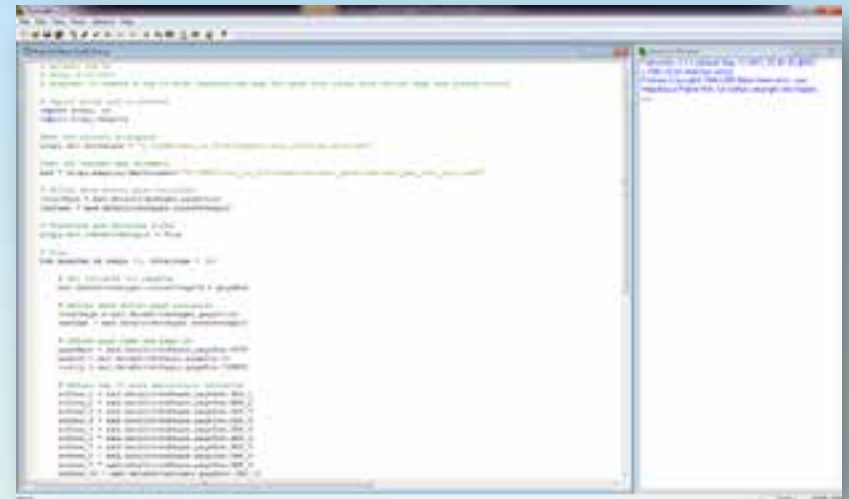
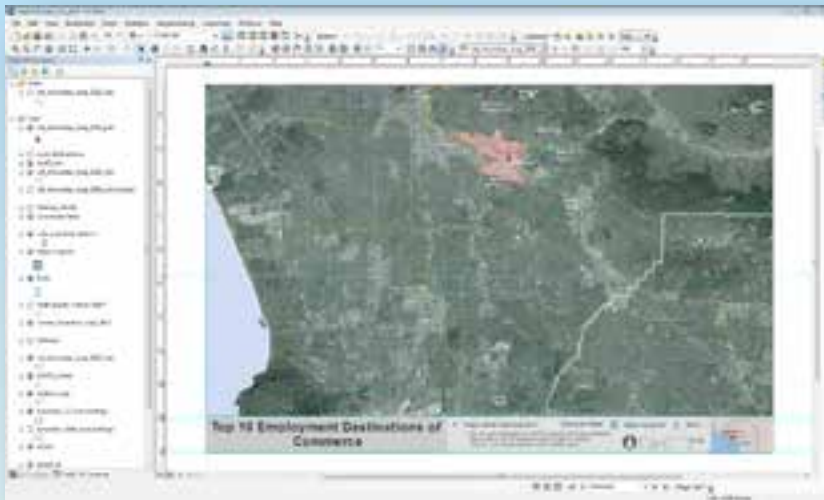
§ *Python* site package for performing GIS functions available in *ArcGIS*

- Encompassing the *arcgisscripting* module
- E.g. *arcpy.mapping* (mapping module)

§ Useful and productive way to perform geographic data analysis, data conversion, data management, and map automation with Python

Visualizing Major Work Destinations (Automated Mapping Workflow)

§ Using ArcGIS and Python to create a series of work destination maps for 191 cities and 6 counties



```
import arcpy
import os
import sys

# Set the workspace to the current directory
arcpy.env.workspace = os.getcwd()

# Define the input and output paths
input_path = "Input"
output_path = "Output"

# Create a list of cities and counties
cities = ["City 1", "City 2", "City 3"]
counties = ["County 1", "County 2", "County 3"]

# Loop through each city and county
for city in cities:
    for county in counties:
        # Create a unique name for the output map
        output_name = os.path.join(output_path, city + county + ".png")

        # Run the mapping process
        arcpy.ExecuteTool("ToolPath", output_name)
```

A screenshot of a Python script editor showing code for an automated mapping workflow. The code includes imports for arcpy, os, and sys. It defines input and output paths, lists cities and counties, and uses loops to iterate through each city and county, creating unique output map names and running a mapping tool.



Python Script to Visualize Top 10 Work Destinations

§ Importing ArcPy and ArcPy.mapping modules in Python to automatically create work destination maps

- Data Driven Page, Definition Query, and Exporting functions in ArcGIS
 - mxd.dataDrivenPages.currentPageID
 - lyr.definitionQuery = sqlExp
 - arcpy.mapping.ExportToJPEG

```
# Import ArcPy and os modules
import arcpy, os
import arcpy.mapping

#Set the current workspace
arcpy.env.workspace = "P:/LEHD/esri_uc_2014/shapes/local_profiles_2013.gdb"

#Set the current map document
mxd = arcpy.mapping.MapDocument("P:/LEHD/esri_uc_2014/mxds/regional_maps/regional_map_city_gnrl.mxd")

# Define data driven page variables
totalPage = mxd.dataDrivenPages.pageCount
curPage = mxd.dataDrivenPages.currentPageID
```


Top 10 Work Destinations Data

§ Identified top 10 work destinations where residents of each jurisdictions commute to work (191 cities and 6 counties)

City_code	City_name	Des_1	Job#_1	Des_2	Job#_2	...	Des_10	Job#_10
1001	Alhambra city	754 Los Angeles city	111 San Bernardino city	538 Los Angeles city	157 Apple Valley city		190 Anaheim city	
1002	Agoura city	2025 Thousand Oaks city	830 Agoura Hills city	425 Calabasas city	579 Woodlake Village city		340 Burbank city	
1003	Alhambra city	754 Los Angeles city	111 San Bernardino city	538 Los Angeles city	157 Apple Valley city		190 Anaheim city	
1004	Alhambra city	754 Los Angeles city	111 San Bernardino city	538 Los Angeles city	157 Apple Valley city		190 Anaheim city	
1005	Alhambra city	754 Los Angeles city	111 San Bernardino city	538 Los Angeles city	157 Apple Valley city		190 Anaheim city	
1006	Alhambra city	754 Los Angeles city	111 San Bernardino city	538 Los Angeles city	157 Apple Valley city		190 Anaheim city	
1007	Alhambra city	754 Los Angeles city	111 San Bernardino city	538 Los Angeles city	157 Apple Valley city		190 Anaheim city	
1008	Alhambra city	754 Los Angeles city	111 San Bernardino city	538 Los Angeles city	157 Apple Valley city		190 Anaheim city	
1009	Alhambra city	754 Los Angeles city	111 San Bernardino city	538 Los Angeles city	157 Apple Valley city		190 Anaheim city	
1010	Alhambra city	754 Los Angeles city	111 San Bernardino city	538 Los Angeles city	157 Apple Valley city		190 Anaheim city	

Visualization of Top 10 Work Destinations


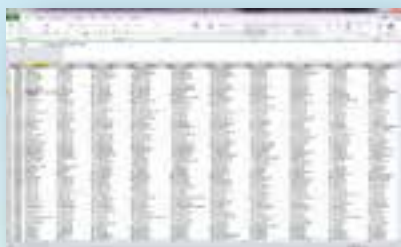
- § Created a series of maps that depict top 10 work destinations for 191 cities and 6 counties, based on automated workflow using ArcGIS and Python.



Incorporation of Python and MS Excel

§ Utilize modules in Python to create Top 10 Destination tables for each jurisdiction

- E.g. xlrd and xlwt module
 - `citynames = sheet.cell_value(i,2)`
 - `s.write(0,0,citynames,header_style1)`



Alhambra city	
Cities	Job #
Los Angeles city	9812
Alhambra city	3335
Pasadena city	1691
Monterey Park city	1115
San Gabriel city	767
Glendale city	672
Rosemead city	614
Industry city	554
El Monte city	534
Burbank city	513
Total	19607

Conclusions

- § Significantly efficient in processing O-D analysis with SAS for numerous jurisdictions
- § Some inconsistencies issue in the total number of job at place level between SAS programming and *OnTheMap* application
- § Data limitation of *LODES* datasets
- § Incorporation of Python and ArcGIS improves processing time and accuracy of map production

Future Studies

- § Work destinations at different geographical level
 - City to City à Sub-region to Sub-region
 - Top 10 work destinations à Top 20 work destinations
- § Produce major work destination data using Census Transportation Planning Products (CTPP)
- § Desktop application or ArcGIS toolbox to generate maps based on a selection of jurisdictions

Thank you!

Yunkyung Choi (ChoiY@scag.ca.gov)

Tom M. Vo (Vo@scag.ca.gov)

