C2SMART
CONNECTED CITIES WITH SMART TRANSPORTATION

NYC FREIGHT:

DATA ANALYSIS TO BUILD A SYNTHETIC POPULATION

Haggai Davis III

PhD Candidate at New York University Tandon School of Engineering

Outline

Data Input

- (Side) Data Visualization

Tour Generation

- Tour Length
- Origin Destination Sets
- Tour Sets

Entropy Maximization Model

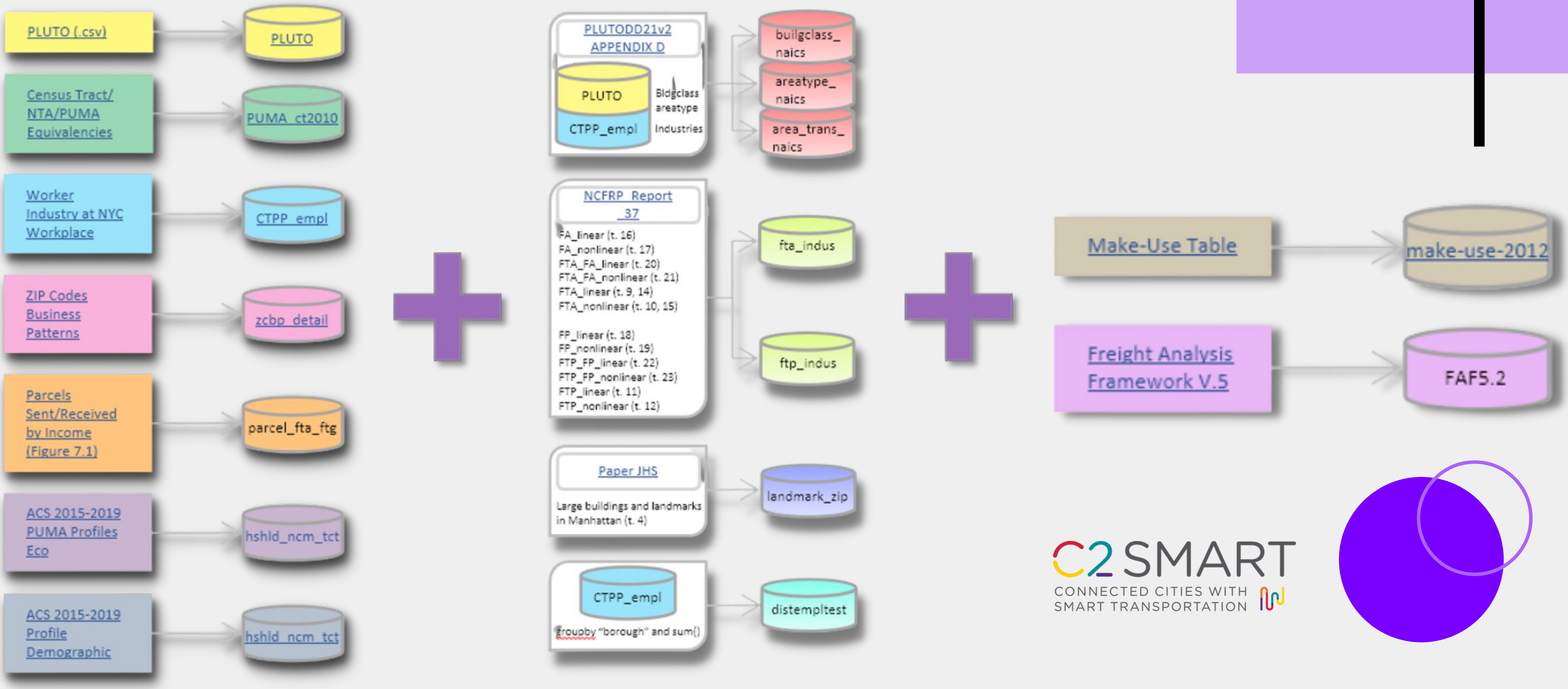
- Derivation
- Results

Analysis

(COMING SOON(TM)

DATA INPUT

Data Source



PLUTO (.csv) → PLUTO

Census Tract/ NTA/PUMA Equivalencies → PUMA_ct2010

Worker Industry at NYC Workplace → CTPP_empl

ZIP Codes Business Patterns → zcbp_detail

Parcels Sent/Received by Income (Figure 7.1) → parcel_fta_ftg

ACS 2015-2019 PUMA Profiles Eco → hshld_ncm_tct

ACS 2015-2019 Profile Demographic → hshld_ncm_tct

+

PLUTODD21v2 APPENDIX D

PLUTO

CTPP_empl

Bldgclass
areatype
Industries

→ builgclass_naics
→ areatype_naics
→ area_trans_naics

NCFRP_Report_37

FA_linear (t. 16)
FA_nonlinear (t. 17)
FTA_FA_linear (t. 20)
FTA_FA_nonlinear (t. 21)
FTA_linear (t. 9, 14)
FTA_nonlinear (t. 10, 15)

FP_linear (t. 18)
FP_nonlinear (t. 19)
FTP_FP_linear (t. 22)
FTP_FP_nonlinear (t. 23)
FTP_linear (t. 11)
FTP_nonlinear (t. 12)

→ fta_indus
→ ftp_indus

Paper JHS

Large buildings and landmarks in Manhattan (t. 4)

→ landmark_zip

CTPP_empl

groupby "borough" and sum()

→ distempltest

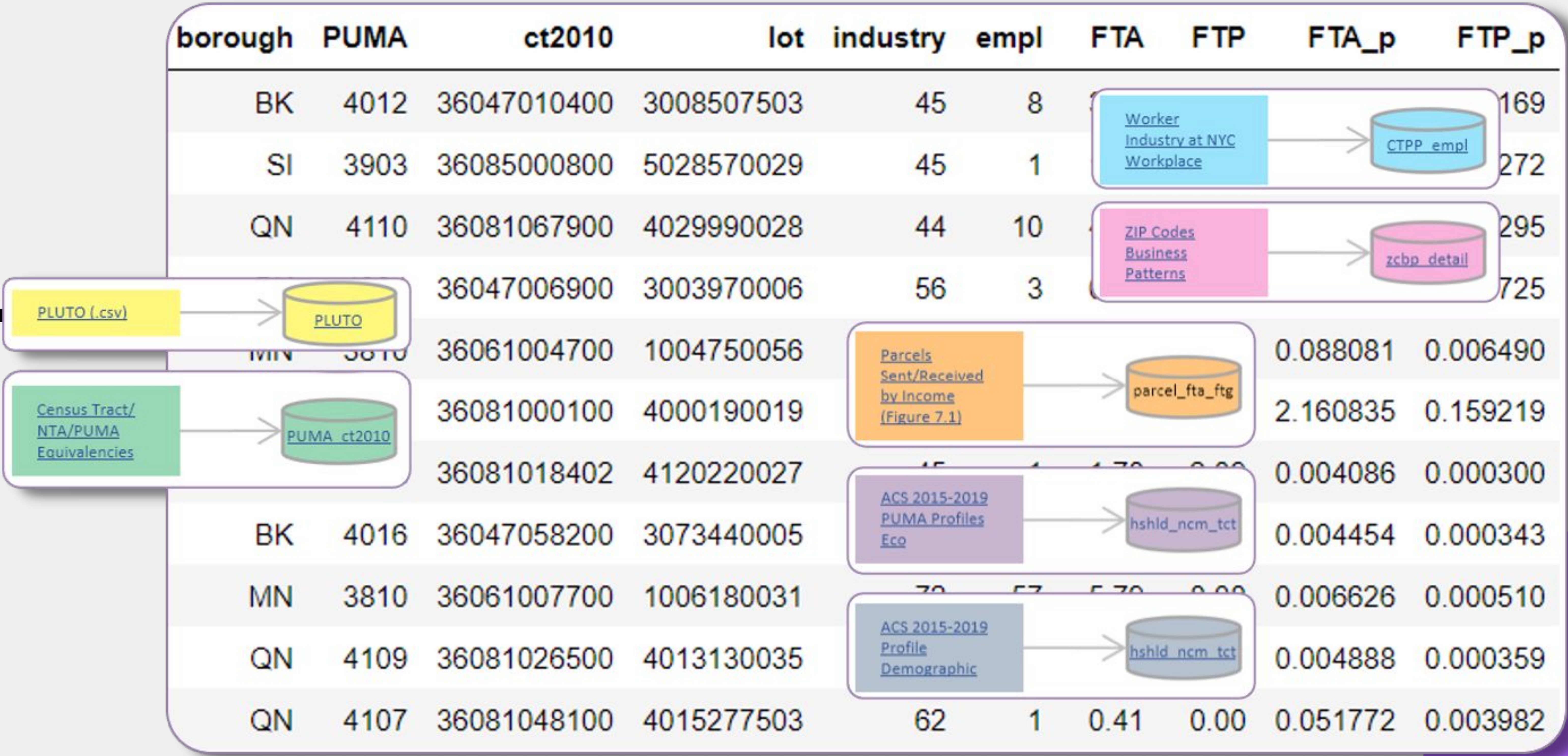
+

Make-Use Table → make-use-2012

Freight Analysis Framework V.5 → FAF5.2

C2SMART
CONNECTED CITIES WITH SMART TRANSPORTATION

Data Frame

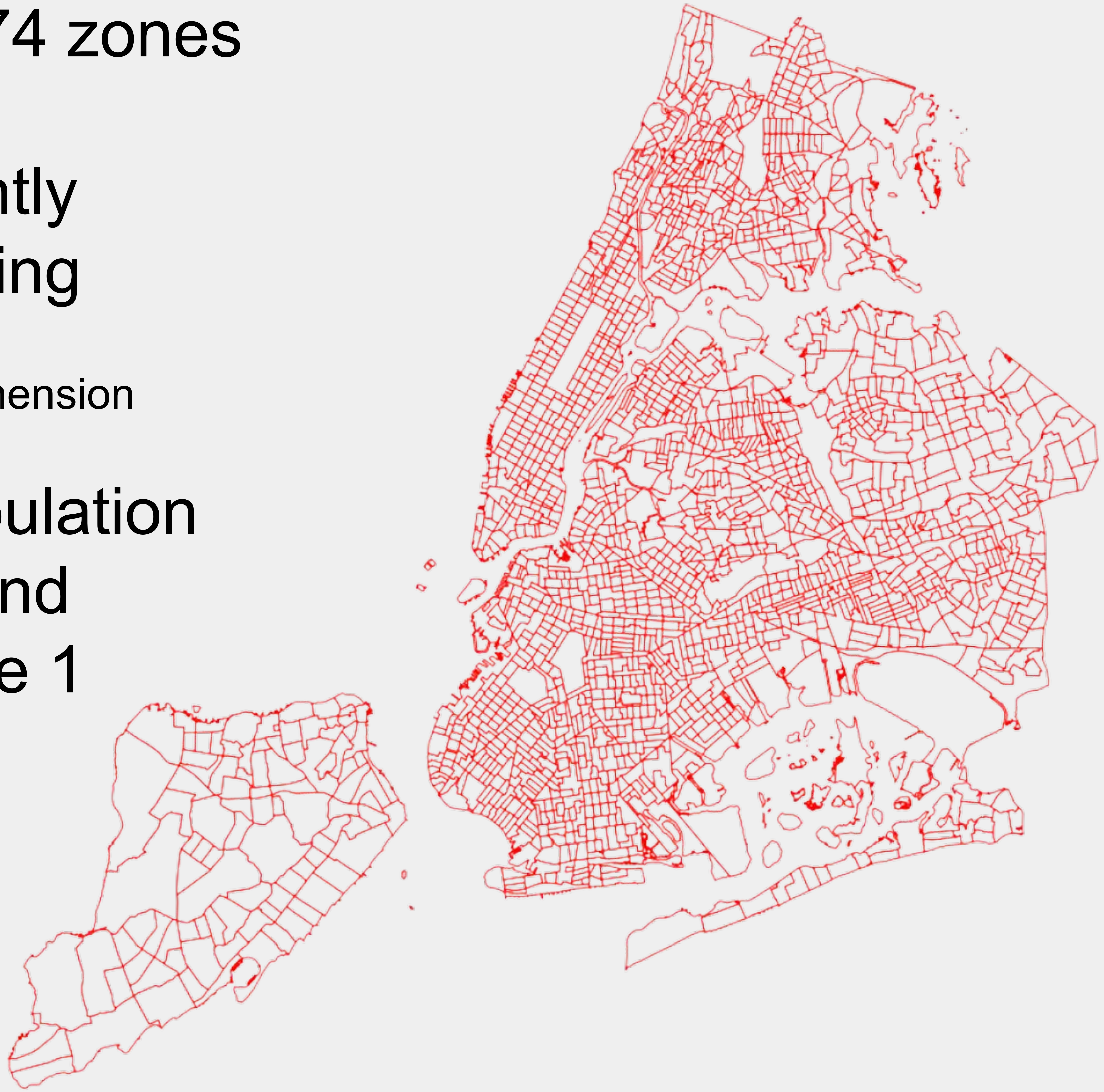


borough	PUMA	ct2010	lot	industry	empl	FTA	FTP	FTA_p	FTP_p
BK	4012	36047010400	3008507503	45	8	[illegible]	[illegible]	[illegible]	[illegible]169
SI	3903	36085000800	5028570029	45	1	[illegible]	[illegible]	[illegible]	[illegible]272
QN	4110	36081067900	4029990028	44	10	[illegible]	[illegible]	[illegible]	[illegible]295
[illegible]	[illegible]	36047006900	3003970006	56	3	[illegible]	[illegible]	[illegible]	[illegible]725
MN	3810	36061004700	1004750056	[illegible]	[illegible]	[illegible]	[illegible]	0.088081	0.006490
[illegible]	[illegible]	36081000100	4000190019	[illegible]	[illegible]	[illegible]	[illegible]	2.160835	0.159219
[illegible]	[illegible]	36081018402	4120220027	[illegible]	[illegible]	[illegible]	[illegible]	0.004086	0.000300
BK	4016	36047058200	3073440005	[illegible]	[illegible]	[illegible]	[illegible]	0.004454	0.000343
MN	3810	36061007700	1006180031	[illegible]	[illegible]	[illegible]	[illegible]	0.006626	0.000510
QN	4109	36081026500	4013130035	[illegible]	[illegible]	[illegible]	[illegible]	0.004888	0.000359
QN	4107	36081048100	4015277503	62	1	0.41	0.00	0.051772	0.003982

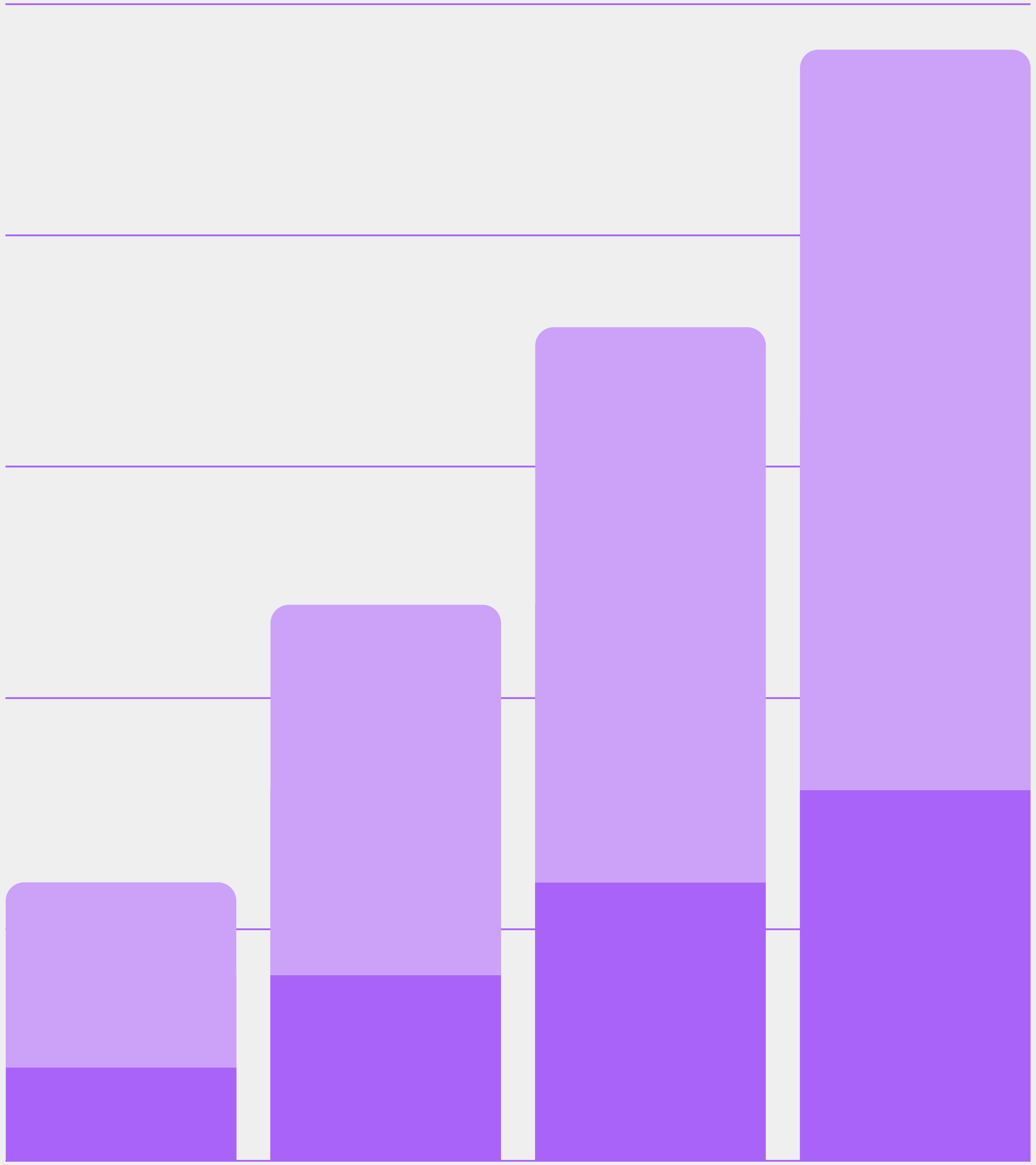
Worker Industry at NYC Workplace
CTPP_empl
ZIP Codes Business Patterns
zcbp_detail
PLUTO (.csv)
PLUTO
Census Tract/ NTA/PUMA Equivalencies
PUMA_ct2010
Parcels Sent/Received by Income (Figure 7.1)
parcel_fta_ftg
ACS 2015-2019 PUMA Profiles Eco
hshld_ncm_tct
ACS 2015-2019 Profile Demographic
hshld_ncm_tct

Equity 2010 Zones

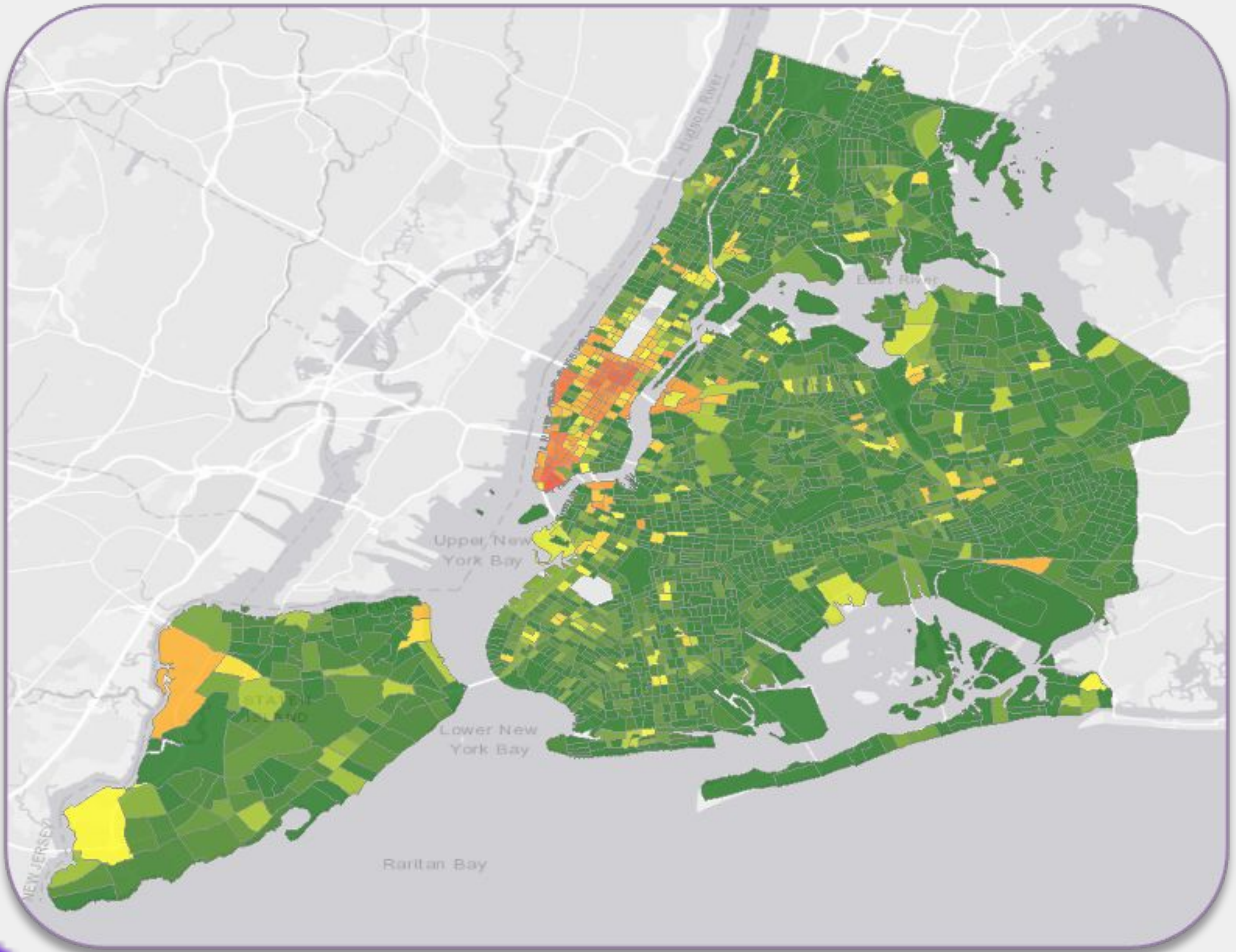
- Combined 2,165 census tracts into 574 zones
- Demographic data reliability significantly improved for minority groups by creating homogeneity in zones
  - The average % MoE was at least halved for each dimension
- Data source: ACS (3 dimensions: population above 67, population below poverty and population with a commute time above 1 hour)
- Method of Generation: Tabu Search based on pre-generated solutions

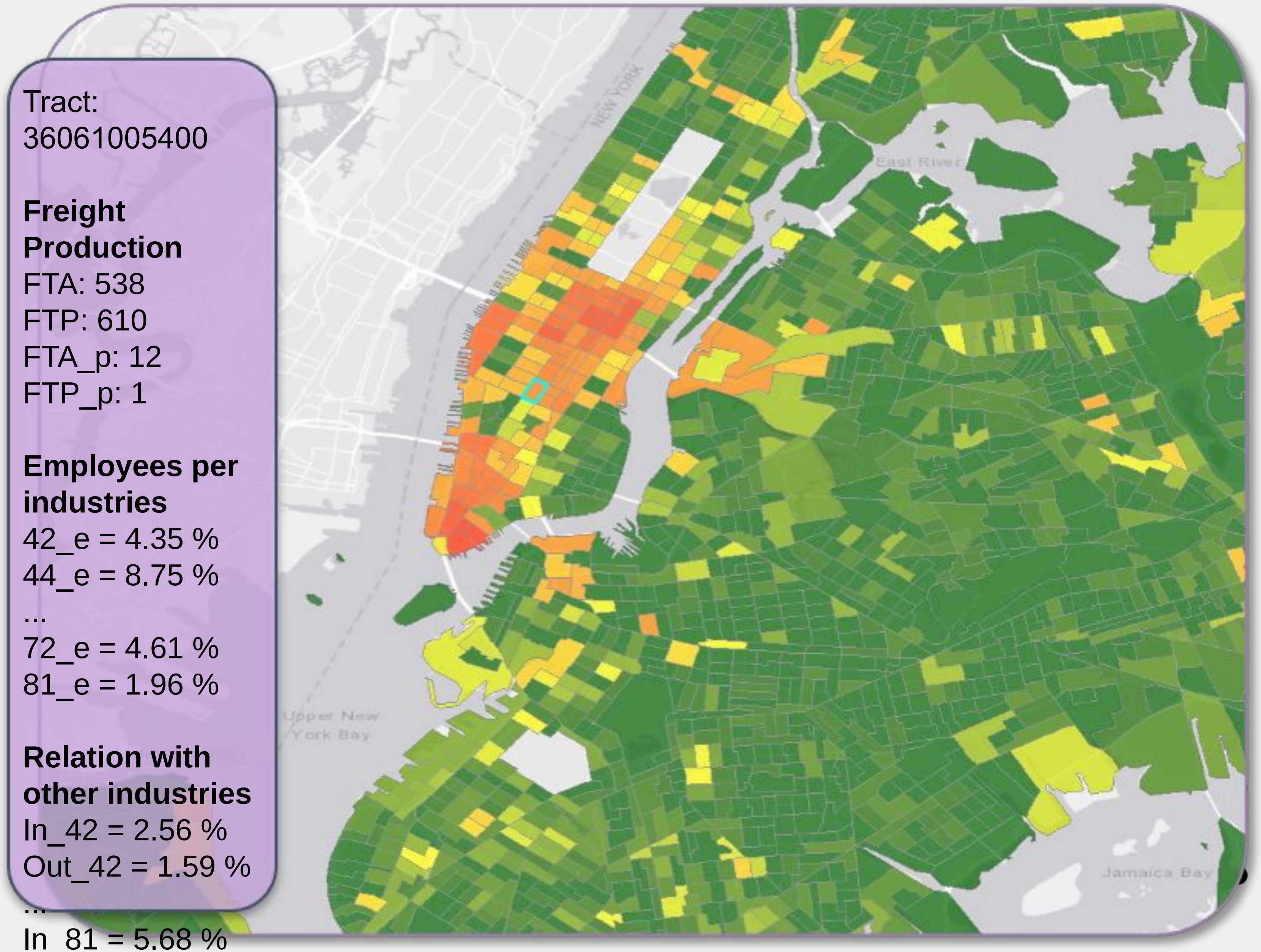


Data Visualization



Industry and Zonal Relationships





Tract:
36061005400

Freight Production
FTA: 538
FTP: 610
FTA_p: 12
FTP_p: 1

Employees per industries
42_e = 4.35 %
44_e = 8.75 %
...
72_e = 4.61 %
81_e = 1.96 %

Relation with other industries
In_42 = 2.56 %
Out_42 = 1.59 %
...
In_81 = 5.68 %
Out_81 = 0.56 %

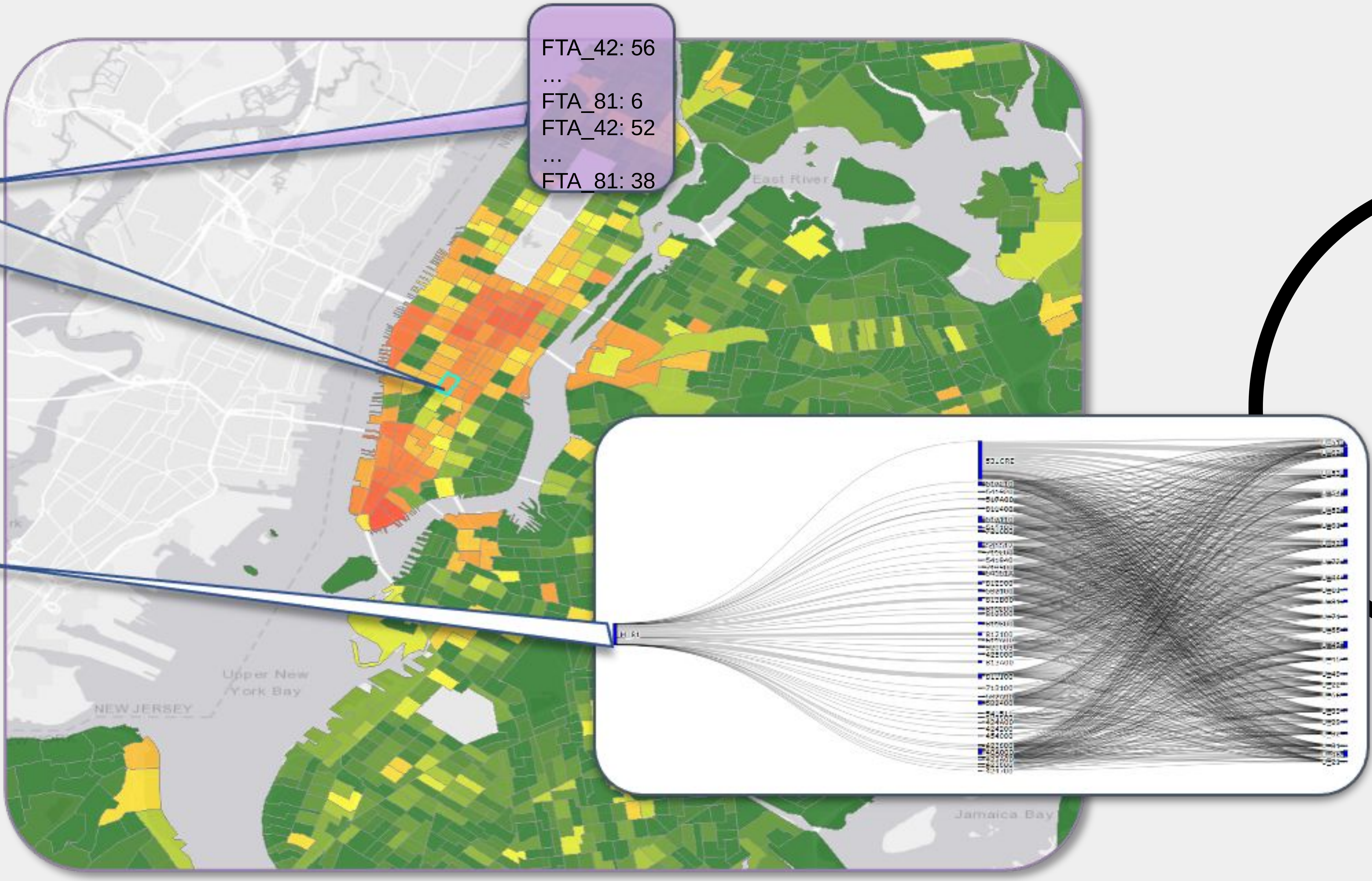
Industry and Zonal Relationships

Tract:
36061005400

Freight Production
FTA: 538
FTP: 610
FTA_p: 12
FTP_p: 1

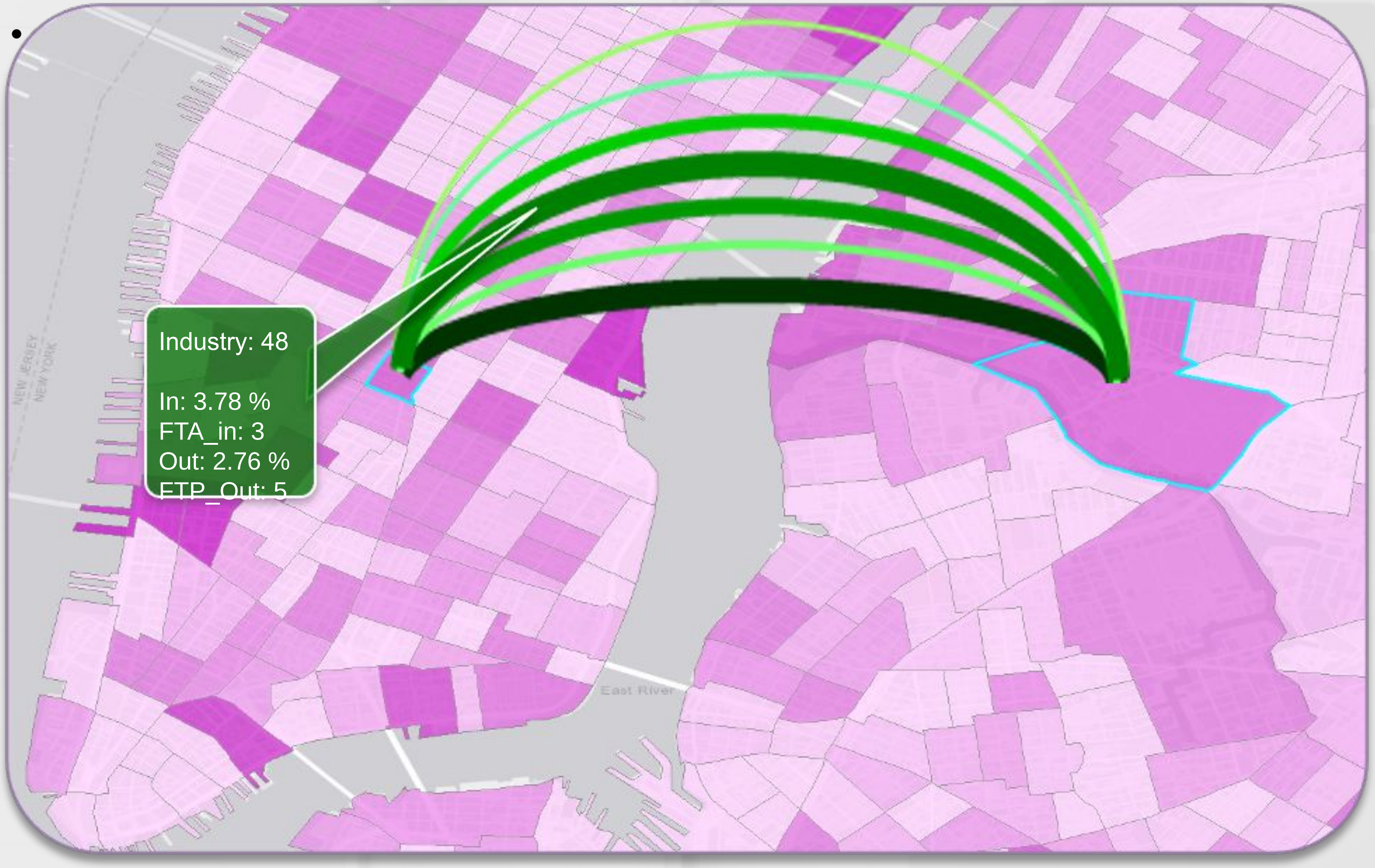
Employees per industries
42_e = 4.35 %
44_e = 8.75 %
...
72_e = 4.61 %
81_e = 1.96 %

Relation with other industries
In_42 = 2.56 %
Out_42 = 1.59 %
...
In_81 = 5.68 %
Out_81 = 0.56 %



FTA_42: 56
...
FTA_81: 6
FTA_42: 52
...
FTA_81: 38

Industry and Zonal Relationships

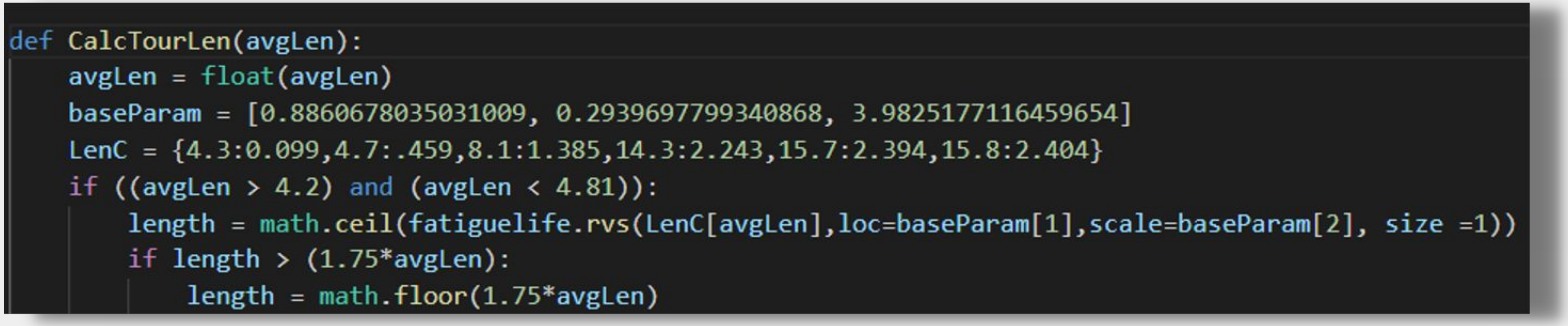


TOUR

GENERATION

Tour Length Generation

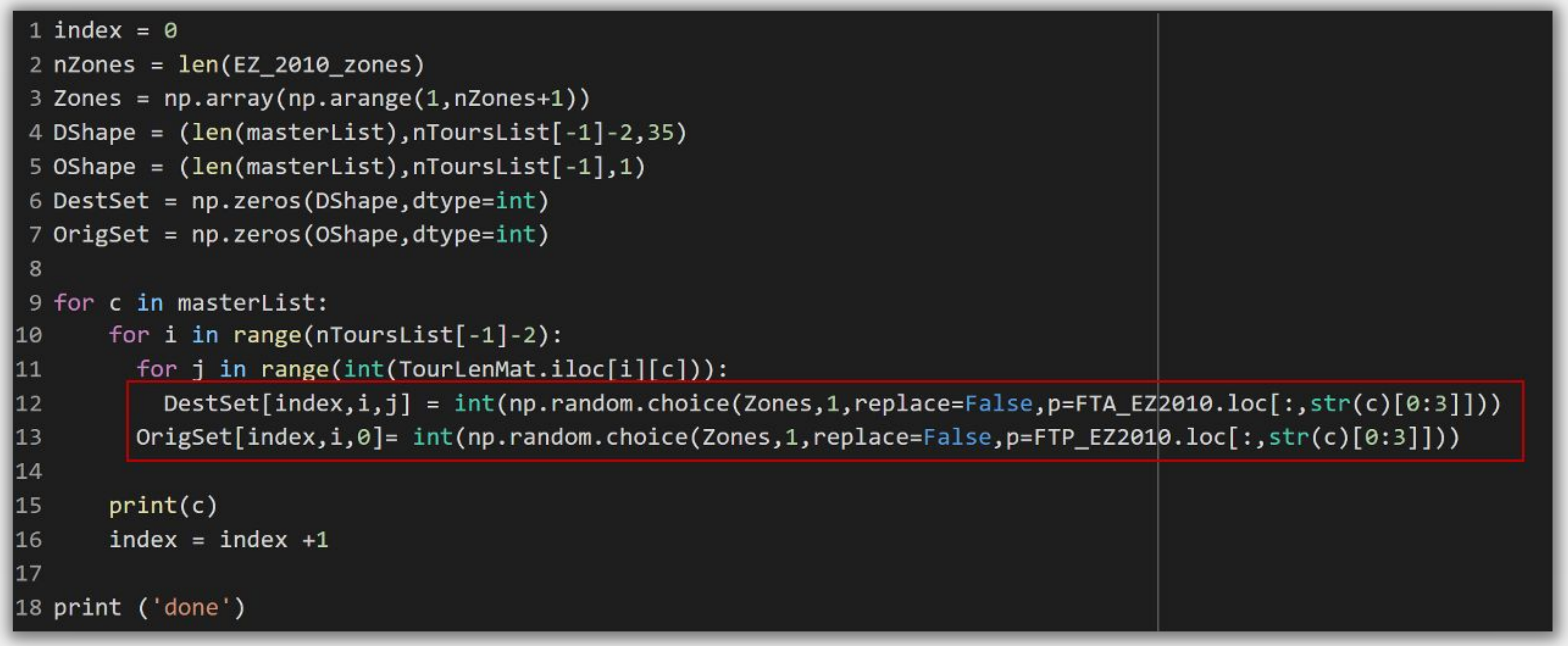
For Each Commodity → For 10,000 tours → Generate a tour length (in number of stops) based on distribution calibrated to the average number of stops for that commodity



```
def CalcTourLen(avgLen):
    avgLen = float(avgLen)
    baseParam = [0.8860678035031009, 0.2939697799340868, 3.9825177116459654]
    LenC = {4.3:0.099,4.7:.459,8.1:1.385,14.3:2.243,15.7:2.394,15.8:2.404}
    if ((avgLen > 4.2) and (avgLen < 4.81)):
        length = math.ceil(fatiguelife.rvs(LenC[avgLen],loc=baseParam[1],scale=baseParam[2], size =1))
        if length > (1.75*avgLen):
            length = math.floor(1.75*avgLen)
```

Origin Destination Set Generation

For Each Commodity → For the 10,000 tours → For the length of each tour (generated previously) →

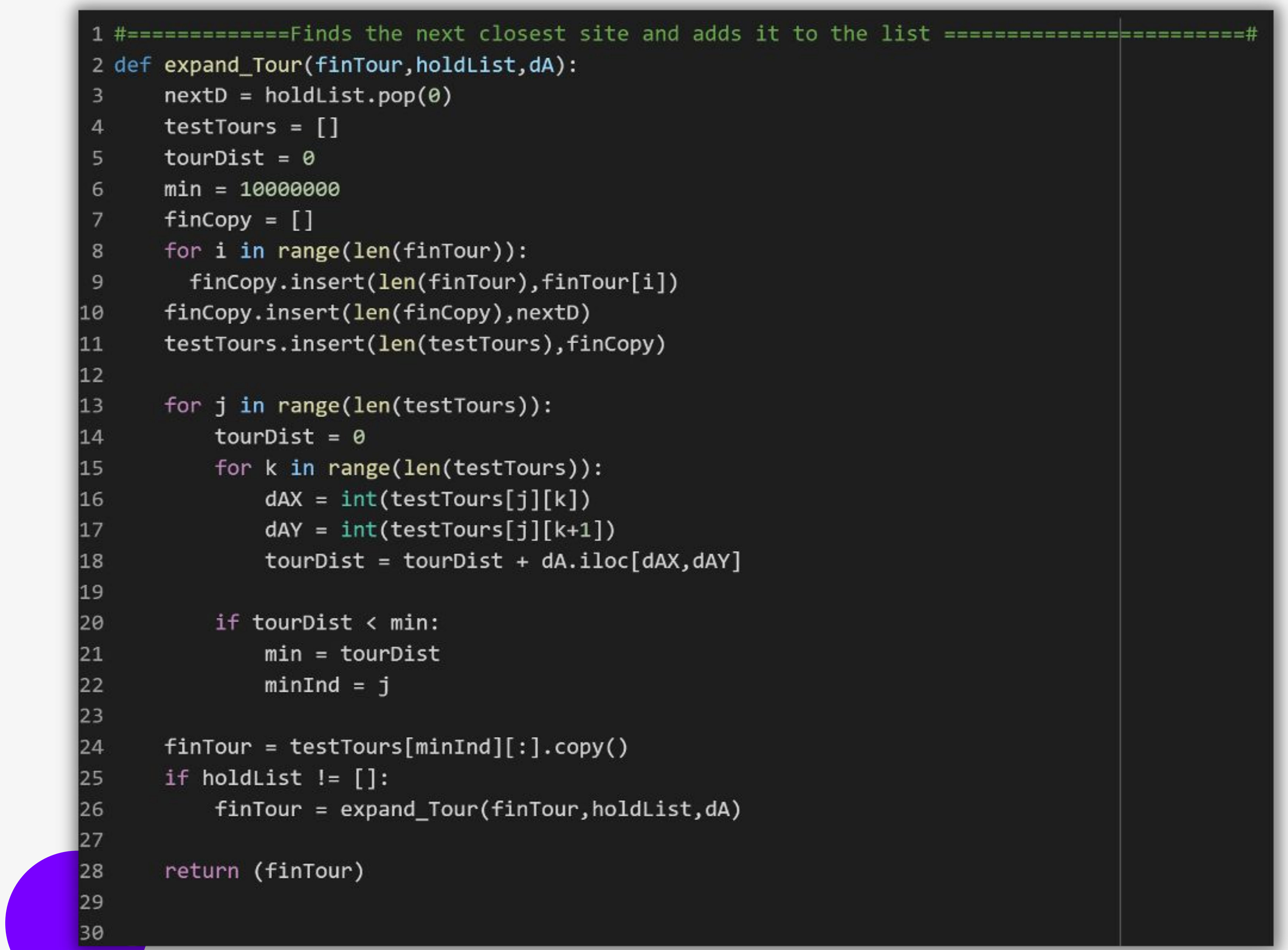


```python
index = 0
nZones = len(EZ_2010_zones)
Zones = np.array(np.arange(1,nZones+1))
DShape = (len(masterList),nToursList[-1]-2,35)
OShape = (len(masterList),nToursList[-1],1)
DestSet = np.zeros(DShape,dtype=int)
OrigSet = np.zeros(OShape,dtype=int)

for c in masterList:
    for i in range(nToursList[-1]-2):
      for j in range(int(TourLenMat.iloc[i][c])):
        DestSet[index,i,j] = int(np.random.choice(Zones,1,replace=False,p=FTA_EZ2010.loc[:,str(c)[0:3]]))
      OrigSet[index,i,0]= int(np.random.choice(Zones,1,replace=False,p=FTP_EZ2010.loc[:,str(c)[0:3]]))

    print(c)
    index = index +1

print ('done')
```

Weighted Randomly sample a location based upon the amount of Freight Produced or Attracted there

C2SMART
CONNECTED CITIES WITH SMART TRANSPORTATION

Tour Set Generation



```python
#=============Finds the next closest site and adds it to the list =========================#
def expand_Tour(finTour,holdList,dA):
    nextD = holdList.pop(0)
    testTours = []
    tourDist = 0
    min = 10000000
    finCopy = []
    for i in range(len(finTour)):
      finCopy.insert(len(finTour),finTour[i])
    finCopy.insert(len(finCopy),nextD)
    testTours.insert(len(testTours),finCopy)

    for j in range(len(testTours)):
        tourDist = 0
        for k in range(len(testTours)):
            dAX = int(testTours[j][k])
            dAY = int(testTours[j][k+1])
            tourDist = tourDist + dA.iloc[dAX,dAY]

        if tourDist < min:
            min = tourDist
            minInd = j

    finTour = testTours[minInd][:].copy()
    if holdList != []:
        finTour = expand_Tour(finTour,holdList,dA)

    return (finTour)
```

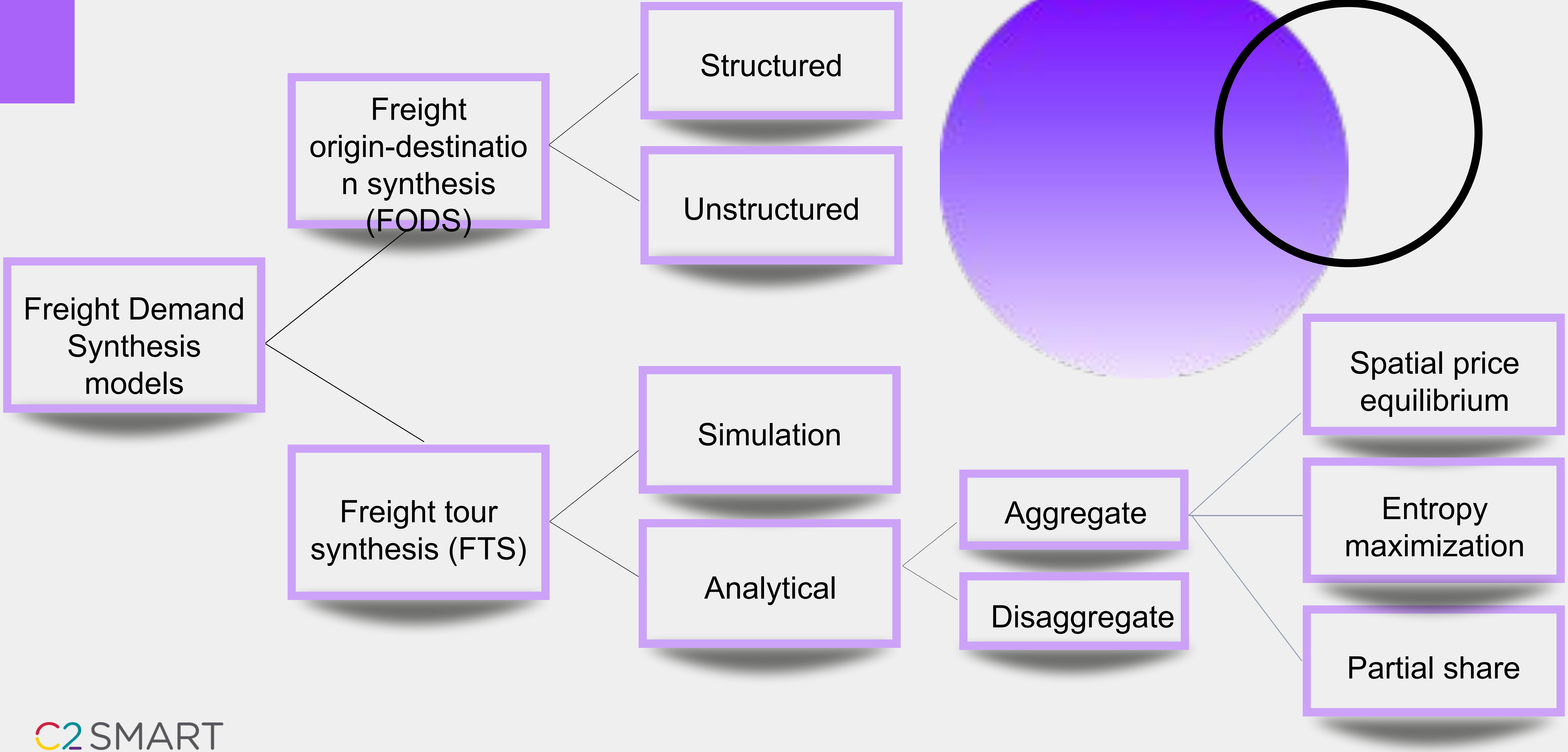
For Each OD Set (generated previously)

Scan through all the ways to connect the locations to find the shortest path required generating a distance matrix

C2SMART
CONNECTED CITIES WITH SMART TRANSPORTATION

ENTROPY MAXIMIZATION

Model Foundation



Freight Demand Synthesis models
Freight origin-destination synthesis (FODS)
Structured
Unstructured
Freight tour synthesis (FTS)
Simulation
Analytical
Aggregate
Disaggregate
Spatial price equilibrium
Entropy maximization
Partial share

Concepts of Entropy Maximization

$Max\ W\{T_{ij}\} = \frac{T!}{\prod_{ij} T_{ij}!}$ ▶ Mathematical Transformation ▶ $Min\ z = \sum_{ij}(T_{ij} log T_{ij} \ - T_{ij})$

1 $O_i \ = \sum_j T_{ij}$

2 $D_j = \sum_i T_{ij}$

3 $T_{ij} \ \geq 0$

4 $\sum_{ij} T_{ij} c_{ij} = C$

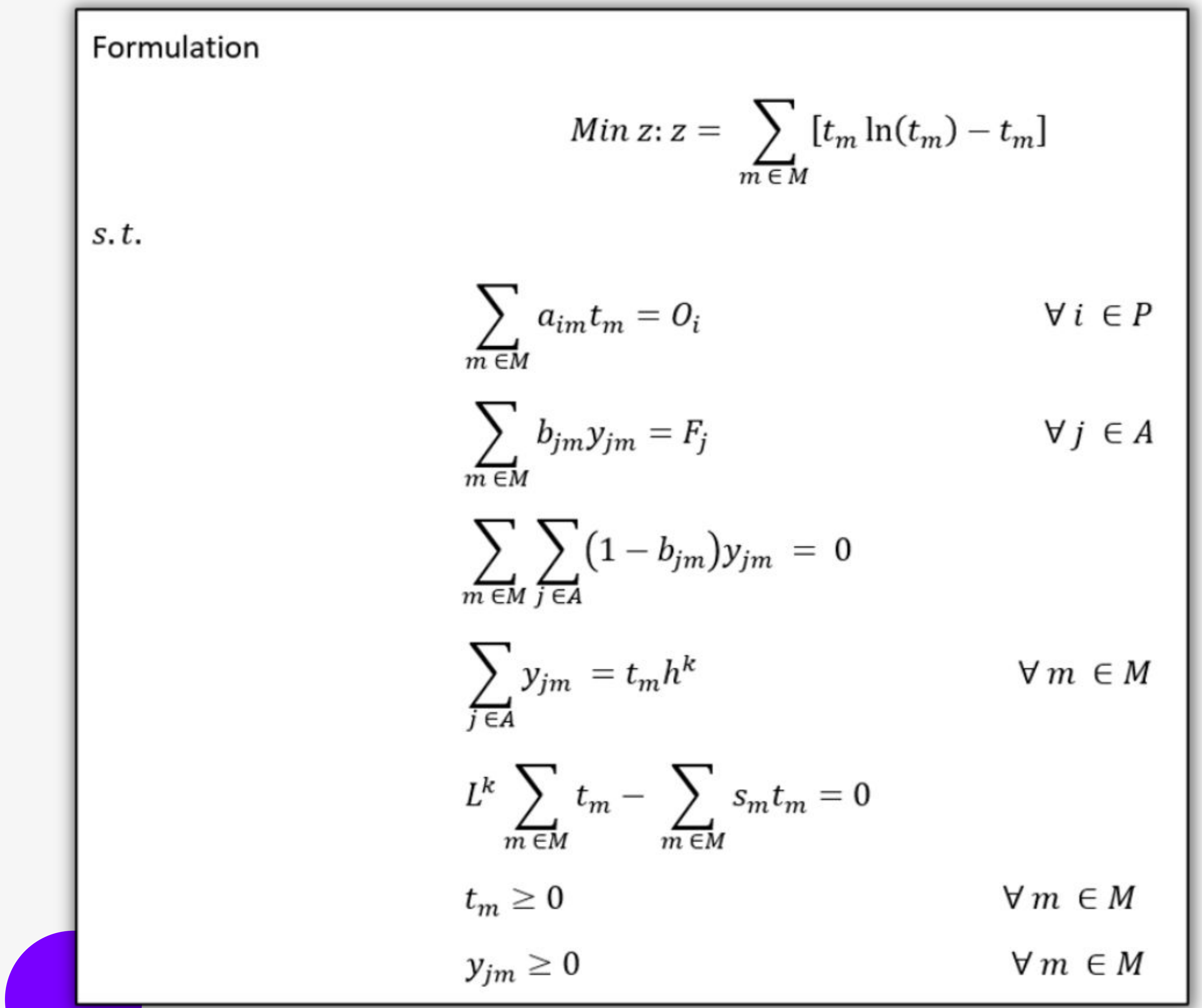
Generic Base Constraints

The Sum over all the trips departing from nodes should be equal to the total number of trips originating there (and vice versa)

Ensuring that the solution is sensible

The cost for each trip multiplied by the number of trips equals a generalized cost (which can be determined from other sources)

Model Formulation



Formulation

$$Min\ z{:}\ z = \sum_{m \in M} [t_m \ln(t_m) - t_m]$$

$s.t.$

$$\sum_{m \in M} a_{im} t_m = O_i \qquad \forall\, i \in P$$

$$\sum_{m \in M} b_{jm} y_{jm} = F_j \qquad \forall\, j \in A$$

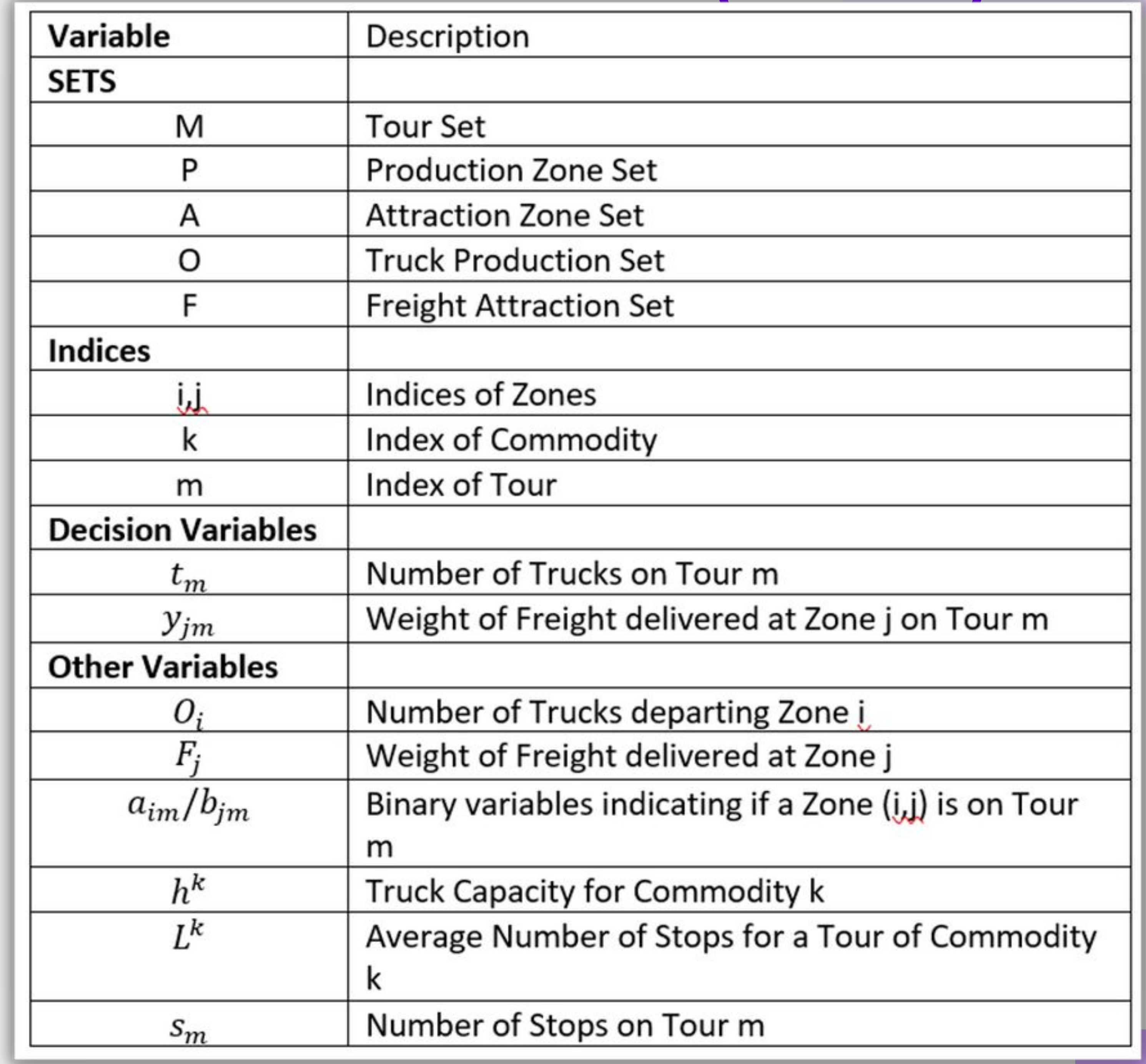
$$\sum_{m \in M} \sum_{j \in A} (1 - b_{jm}) y_{jm} = 0$$

$$\sum_{j \in A} y_{jm} = t_m h^k \qquad \forall\, m \in M$$

$$L^k \sum_{m \in M} t_m - \sum_{m \in M} s_m t_m = 0$$

$$t_m \geq 0 \qquad \forall\, m \in M$$

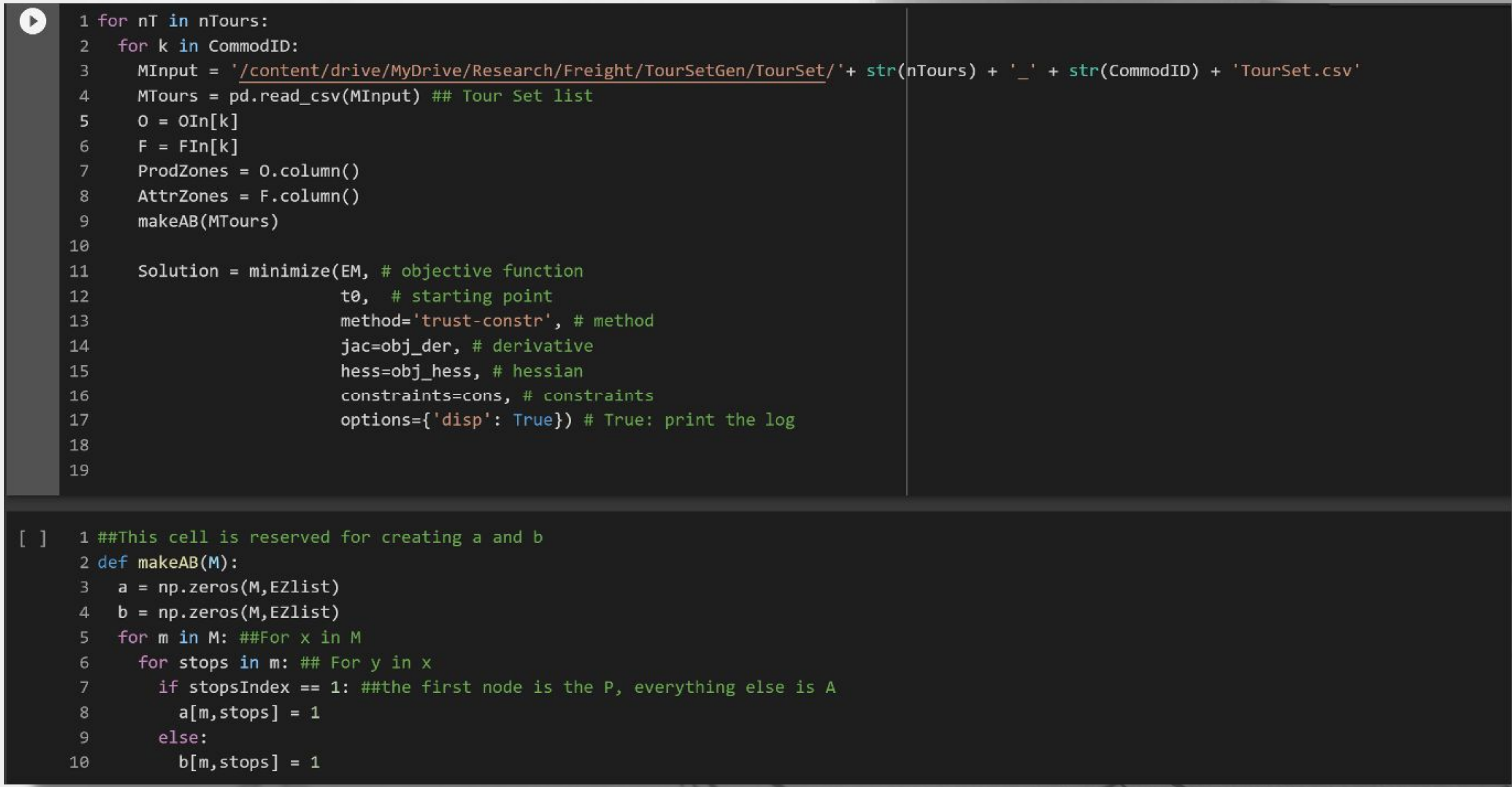
$$y_{jm} \geq 0 \qquad \forall\, m \in M$$



Variable	Description
SETS	
M	Tour Set
P	Production Zone Set
A	Attraction Zone Set
O	Truck Production Set
F	Freight Attraction Set
Indices	
i,j	Indices of Zones
k	Index of Commodity
m	Index of Tour
Decision Variables	
t_m	Number of Trucks on Tour m
y_{jm}	Weight of Freight delivered at Zone j on Tour m
Other Variables	
O_i	Number of Trucks departing Zone i
F_j	Weight of Freight delivered at Zone j
a_{im}/b_{jm}	Binary variables indicating if a Zone (i,j) is on Tour m
h^k	Truck Capacity for Commodity k
L^k	Average Number of Stops for a Tour of Commodity k
s_m	Number of Stops on Tour m

Using the minimize function of the SciPy library

- ‘trust-constr’ method
    - Calculate the 1st and 2nd Derivatives



```
for nT in nTours:
  for k in CommodID:
    MInput = '/content/drive/MyDrive/Research/Freight/TourSetGen/TourSet/'+ str(nTours) + '_' + str(CommodID) + 'TourSet.csv'
    MTours = pd.read_csv(MInput) ## Tour Set list
    O = OIn[k]
    F = FIn[k]
    ProdZones = O.column()
    AttrZones = F.column()
    makeAB(MTours)

    Solution = minimize(EM, # objective function
                        t0,  # starting point
                        method='trust-constr', # method
                        jac=obj_der, # derivative
                        hess=obj_hess, # hessian
                        constraints=cons, # constraints
                        options={'disp': True}) # True: print the log
```

```
##This cell is reserved for creating a and b
def makeAB(M):
  a = np.zeros(M,EZlist)
  b = np.zeros(M,EZlist)
  for m in M: ##For x in M
    for stops in m: ## For y in x
      if stopsIndex == 1: ##the first node is the P, everything else is A
        a[m,stops] = 1
      else:
        b[m,stops] = 1
```

ANALYSIS

Results

Where we can grow and how we can help the field:

Specific Tours

that are heavily loaded per commodity

Validations

to external data

Uniqueness

of contribution to the field

Applicability

Ways that this work can be used to help practitioners

C2SMART
CONNECTED CITIES WITH SMART TRANSPORTATION

C2SMART

CONNECTED CITIES WITH SMART TRANSPORTATION

c2smart.engineering.nyu.edu
c2smart@nyu.edu

Haggai Davis III, PhD Candidate
C2SMART Center
New York University
Tandon School of Engineering
6 MetroTech Center, Brooklyn, NY 11201

Thank you!