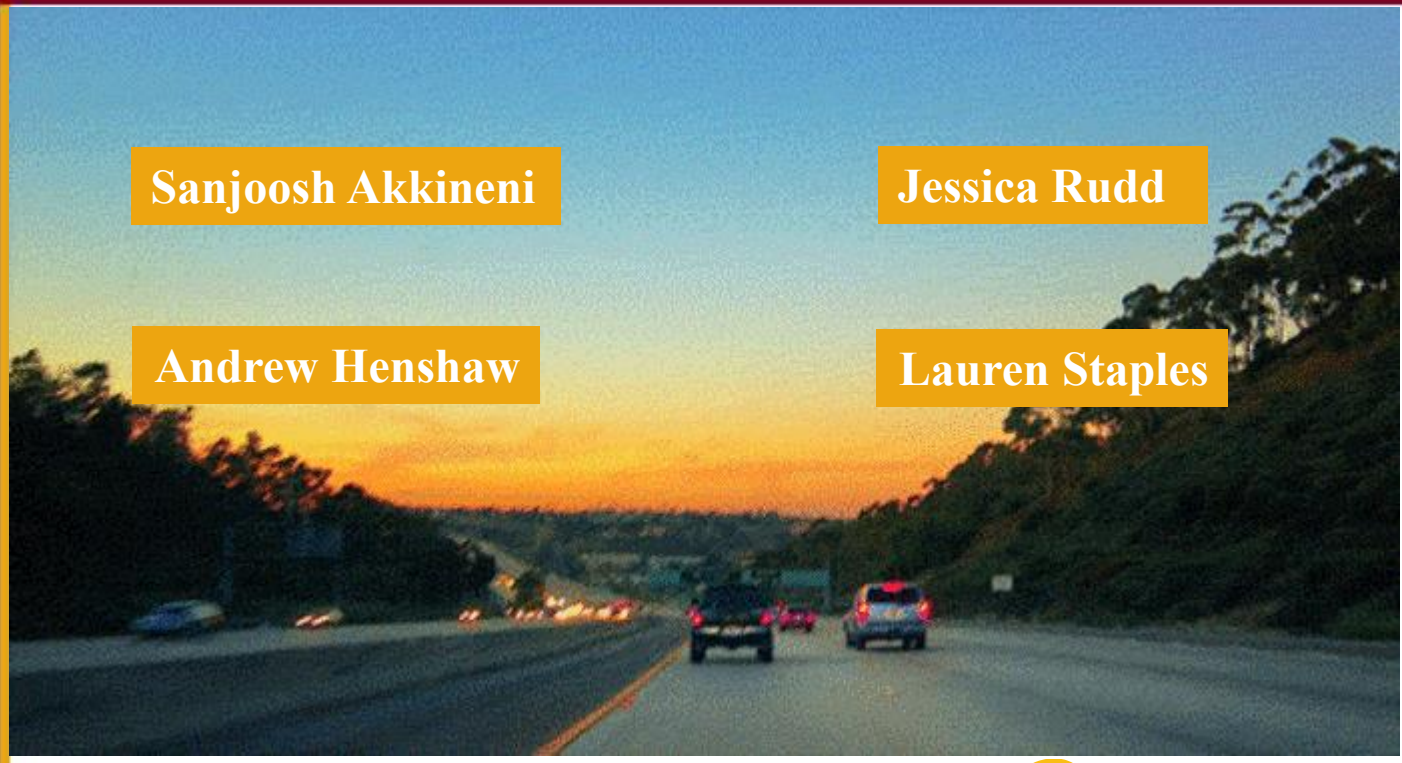
An Optimized Route for Q100's Bert and Kristin to Visit all Jersey Mike's Subs in Atlanta for Charity: *Proof of Concept in JMP and SAS*



Sanjoosh Akkineni

Jessica Rudd

Andrew Henshaw

Lauren Staples

Kennesaw State University

The Graduate College

Ph.D. in Analytics and Data Science

About Lauren and JMP

- Masters in Statistics from University of New Hampshire
- Prior member of NEJUG
- Inspired by Dr. Phil Ramsey
- Still my go-to for quick once-overs
- 100% my go-to for Design of Experiments
- Mostly use SAS, R, Python now
- Spoiler alert: this project mostly used Python
    - Took first stabs with JMP/SAS
    - Will try to relate back to possibilities with JMP at the end.

The Bert Show : Popular morning show on Atlanta's Q100 radio station

Bert's Big Adventure®

Magical Moments...VIP Kids

Problem

- Have two popular radio show hosts Bert and Kristin visit each of these 37 locations.
- How do we optimally determine how to divide the locations into two sets, one for each radio host?
- How do we get each host through Atlanta traffic to visit their respective locations as fast as possible?

Step by Step!

- Step 1: get latitude and longitude for each location.
  - Python library geopy

```python
from geopy.geocoders import Nominatim
geolocator = Nominatim()
#from gmplot import gmplot

location = geolocator.geocode('780 Johnson Ferry Rd. NE, Atlanta, GA')

print(location.latitude,location.longitude)
```

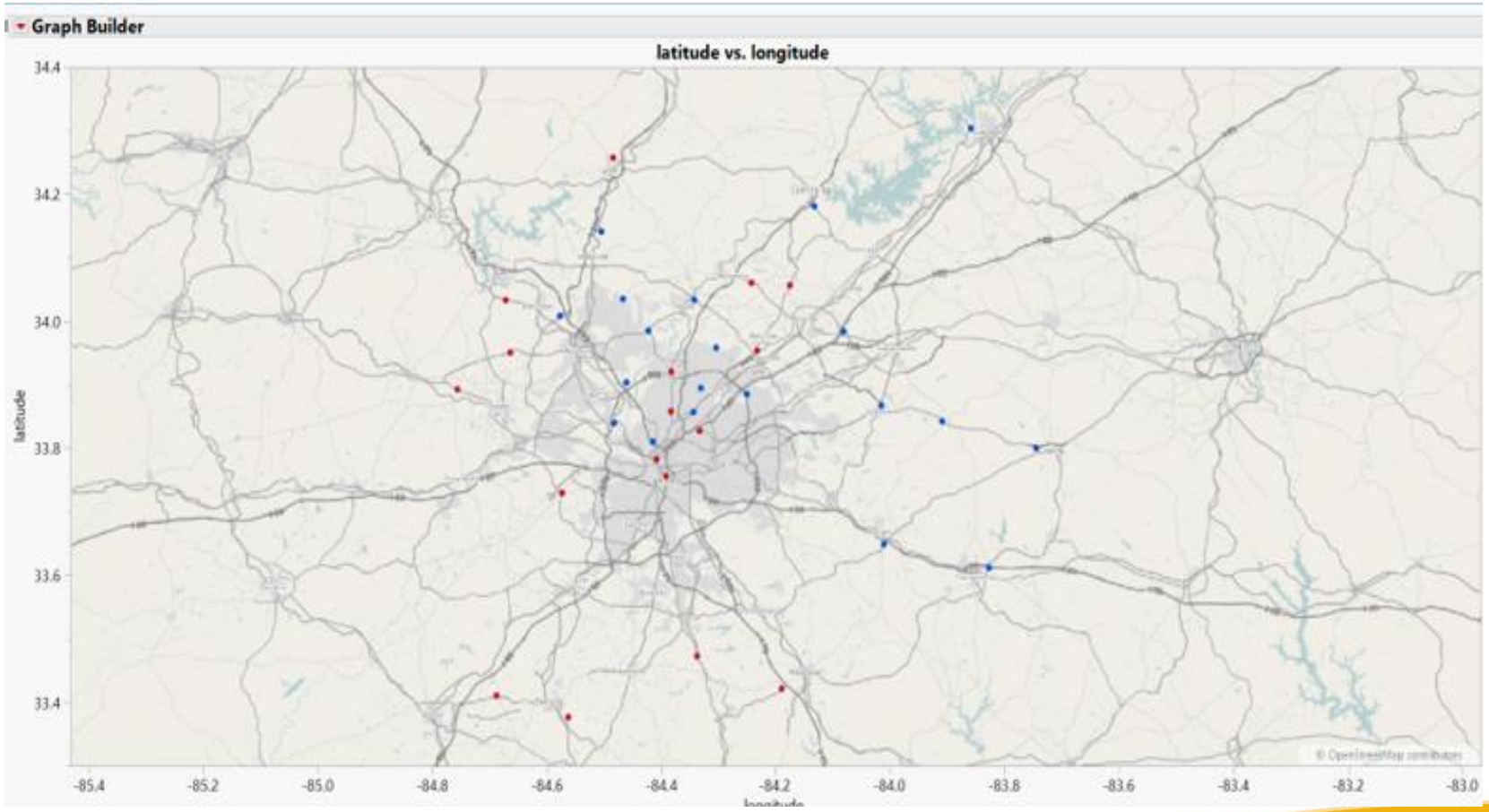
```
>pythonw -u "radio station geocoords.py"
33.9680476169321 -84.4133994768412
>Exit code: 0
```

- Step 2: get drive time between all pairs of stations
  - distancematrix with Google Maps API
  - 666 pairs of locations

Step by Step!

- Step 3: create a visual prototype….here I used JMP!
  - Fast
  - Easy
  - GraphBuilder
  - ***By the way….I COULD have used Geocoder Addin to get the lat/long***
  - https://www.jmp.com/en_ch/events/ondemand/mastering-jmp/using-geographic-maps.html
    - 2:28

Step 3 JMP Output



Step by Step!

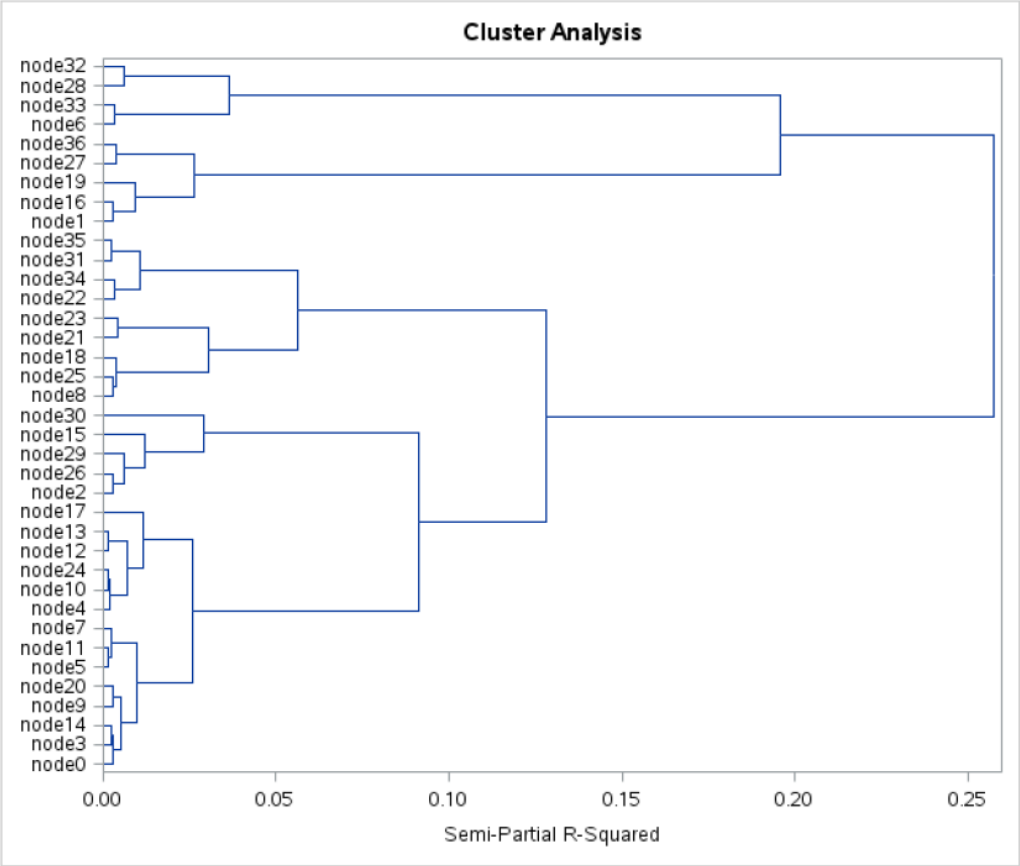
- Step 4: determine an optimal split into two routes
  - Best judgement
  - Cluster by Euclidean Distance
  - Cluster by drive time
  - Ultimately used a Genetic Algorithm
- Step 5: determine the optimal route for each split
  - Traveling Salesman Problem
  - SAS™ PROC OPTGRAPH

Step 4 Code-By Travel Time

```
data transposed2(type=distance);
infile cards missover;
   input name $6. (node0 node1 node2 node3 node4 node5 node6 node7 node8 node9 node10
   node11 node12 node13 node14 node15 node16 node17 node18 node19 node20 node21
   node22 node23 node24 node25 node26 node27 node28 node29 node30 node31 node32
   node33 node34 node35 node36)(5.);
cards;
node0   0
node1   3025    0
node2   1295    3414    0
node3   892     2887    1199    0
node4   1849    3063    2183    1776    0
node5   1443    3181    1821    1283    1244    0
node6   3493    4425    3827    3420    2445    2948    0
node7   1354    3320    1631    1254    1155    677     2898    0
node8   1198    3602    1043    1494    1885    1509    3628    1567    0
node9   1089    3055    1526    989     1416    657     3158    922     1190    0
node10  1545    3256    1879    1472    777     1000    2519    1048    1543    1301    0
```

- Flying Mileages Example
- What's up with that type=distance?

Step 4 Code



- PROC CLUSTER
  - PROC FASTCLUS not for distance data type
- Unbalanced

Step 5

```
proc optgraph
   loglevel    = moderate
   data_links = splittimes2
   out_nodes   = TSPTourNodesa;
   data_links_var
      from     = from
      to       = to
      weight   = weight;
   tsp
      out      = TSPTourLinksb;
run;

proc sql;
title 'drivetime in minutes of Split 1';
select sum(weight)/3600 as drivetime from tsptourlinksa;
quit;
```

drivetime in minutes of Split 1

drivetime
6.820556

drivetime in minutes of Split 2

drivetime
5.7075

****should have been using digraphs to achieve Hamiltonian paths

Solution

Combined approach (CA) to the Multiple Traveling Salesman Problem (mTSP)

Custom Genetic Algorithm (GA)

+

Google's Combinatorial Optimization Solver

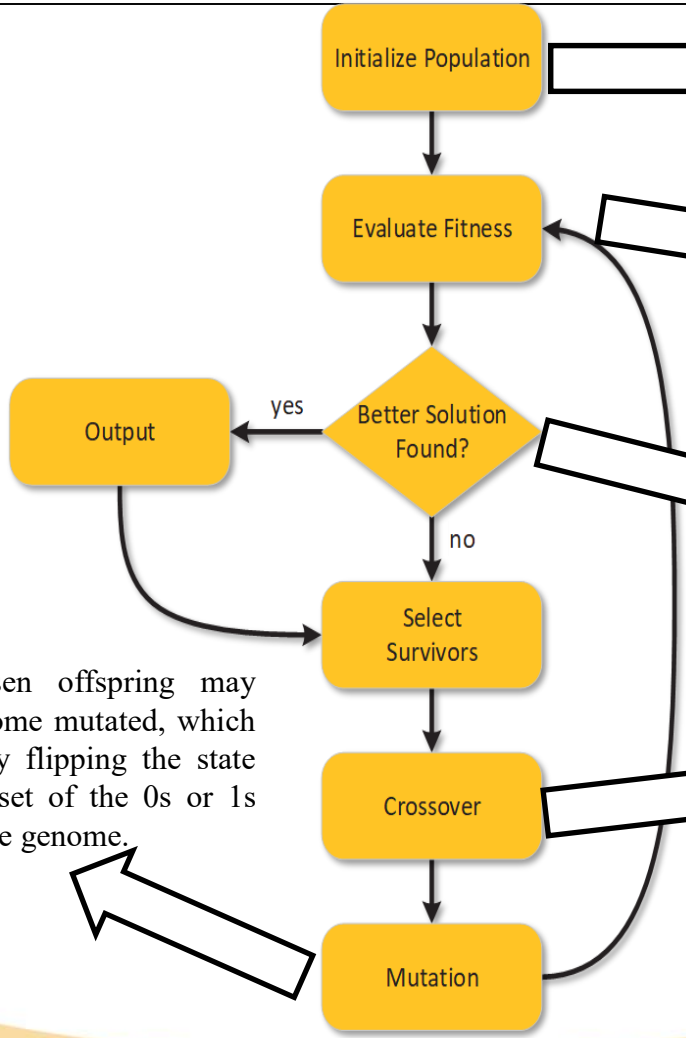
What is a Genetic Algorithm?

Optimization strategy that mimics natural selection

“ Survival of the fittest”

- GA determines the assignment of cities to each radio host

Genetic Algorithm Work Flow



Initialization of the population consists of creating 20 individuals with random genomes. The genome for any individual is just a vector of 37 0s and 1s that define to which route each of the 37 destinations belongs.

"Process" the genome of each individual (building two, separate, unordered lists of destinations). Each list is then passed to a constraint-optimization-based route planner. *In our case, the negated maximum of the time for each route is returned as the fitness*

If the fitness of an individual is better than any that has preceded it, that genome is logged and the fitness is reported. This genome becomes the new benchmark.

A crossover operation is performed on pairs of genomes such that a randomly-selected portion of each genome (vector of 0s and 1s) is used in the construction of new offspring.

Randomly-chosen offspring may have their genome mutated, which involves simply flipping the state of a small subset of the 0s or 1s that make up the genome.

Example

The genotype of one individual might look like this:

1	2	3	4	5		34	35	36	37
0	0	1	0	0	.....	1	1	0	0

Another one might look like this:

1	2	3	4	5		34	35	36	37
1	1	1	0	1	.....	0	0	0	1

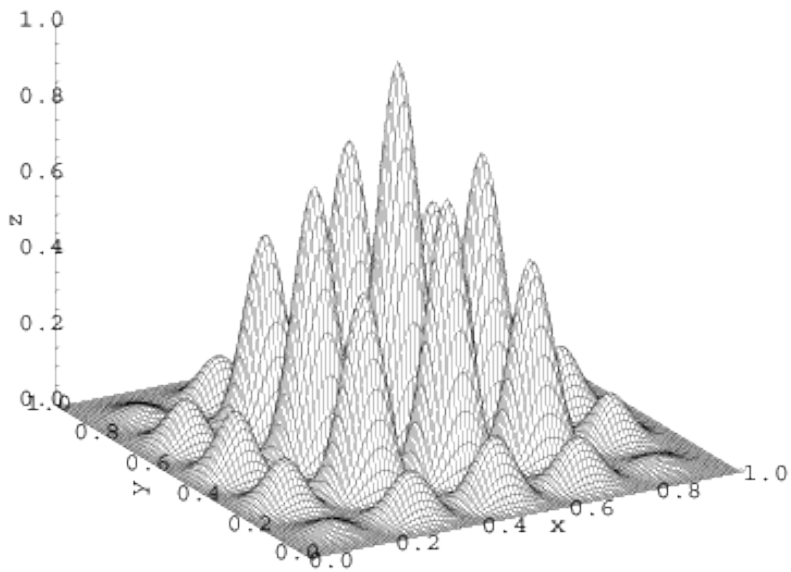
The crossover point is randomly selected. Let's assume index 5 for this example. The child that would be produced would have the following genotype:

1	2	3	4	5		34	35	36	37
0	0	1	0	1	.....	0	0	0	1

So, stops 1, 2, 4, … 34, 35, 36 would be assigned to one route and 3,5, …, 37 would belong to the other.

Search Space

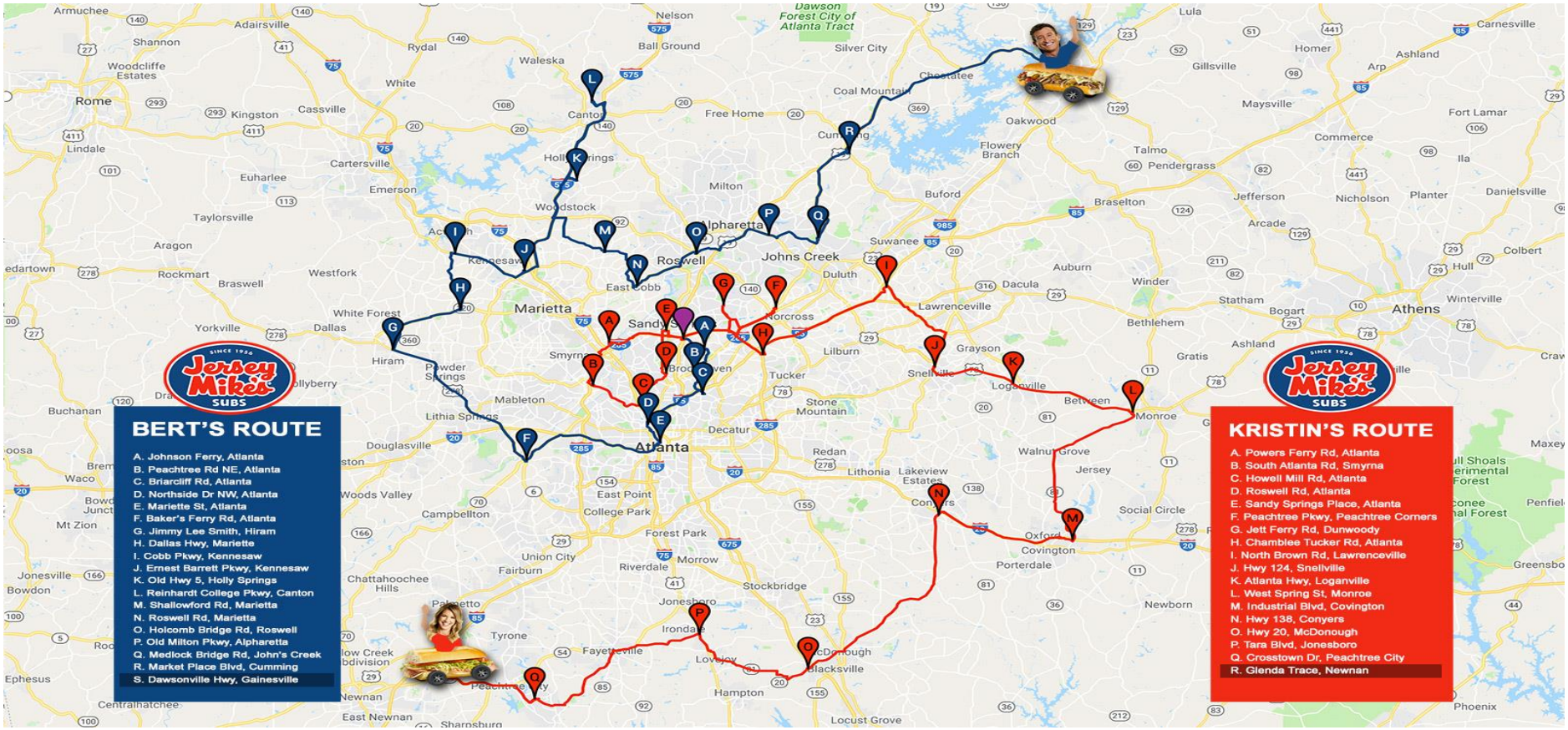
- The space for all possible feasible solutions



Results

- CA method produced a good solution in 900 generations (15 minutes of run time). Brute force would have required 13.8 tredecillion options:
    - There are $(37 \mid 19) \cdot 19! \cdot 18! = 1.38 \times 10^{43}$ (13.8 tredecillion) different ways to split and visit the 37 locations.
- The solution divided the locations into groups of 19 and 18 locations. With a budget of 10 minutes to spend at each location.
- We estimated the completion time for Bert's route (19 locations) to be 8 hrs and 51 mins, and Kristin's route (18 locations) to be 8 hrs and 33 minutes.
- With most Jersey Mike's opening at 10 AM and closing at 9 PM, these solutions more than suffice with plenty of time to spare.

Color-Coded mapped routes



KENNESAW STATE UNIVERSITY

The Graduate College

Ph.D. in Analytics and Data Science

Results

		Predicted			Actual		
	# Locations	Total Time (including 10 min at each location)	Predicted Start Time	Predicted End Time	Total Time	Actual Start Time	Actual End Time
Bert	19	8 hrs 51 min	10:00 AM	6:51 PM	8 hrs 15 min	10:03 AM	6:18 PM
Kristin	18	8 hrs 33 min	10:00 AM	6:33 PM	7 hrs 22 min	10:05 AM	5:27 PM

Conclusion

- CA-produced solution with average drive times from the Google Maps API produced a real-world-validated successful result.

- Radio show hosts Bert and Kristin visited each location well-within their predicted time.

Raised a record

$165,557

for these kids!

Could we have done this in JMP or SAS?

- SAS PROC OPTMODEL
  - TSP
    - Shortest Hamiltonian Path
  - Genetic Algorithm
  - Connect the two with loops
- JMP
  - Clustering is available, and for this datatype
  - Genetic Algorithm available through an interface to MATLAB
  - Scripting example found for associative arrays
    - https://www.jmp.com/support/help/14/associative-arrays-in-graph-theory.shtml#223591
  - JMP Genetics example
    - https://support.sas.com/resources/papers/proceedings12/160-2012.pdf


Thank You