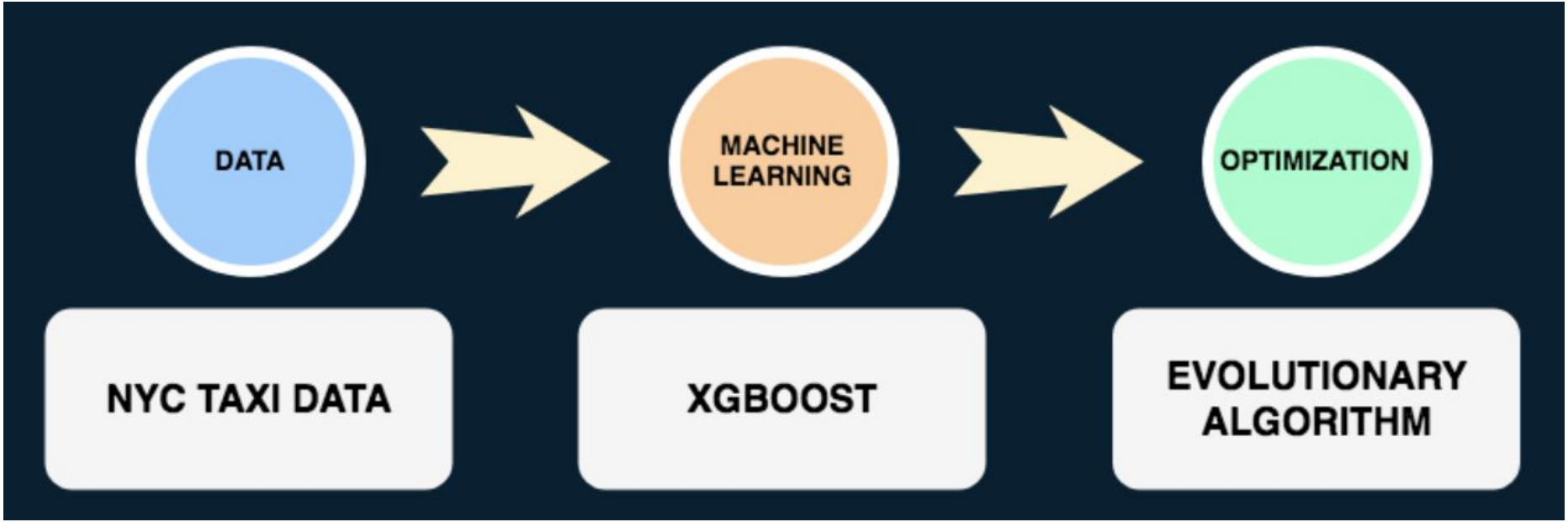
Travel Time Optimization Problem via Ant Colony and Genetic Evolution

Michael Peechatt and James Le

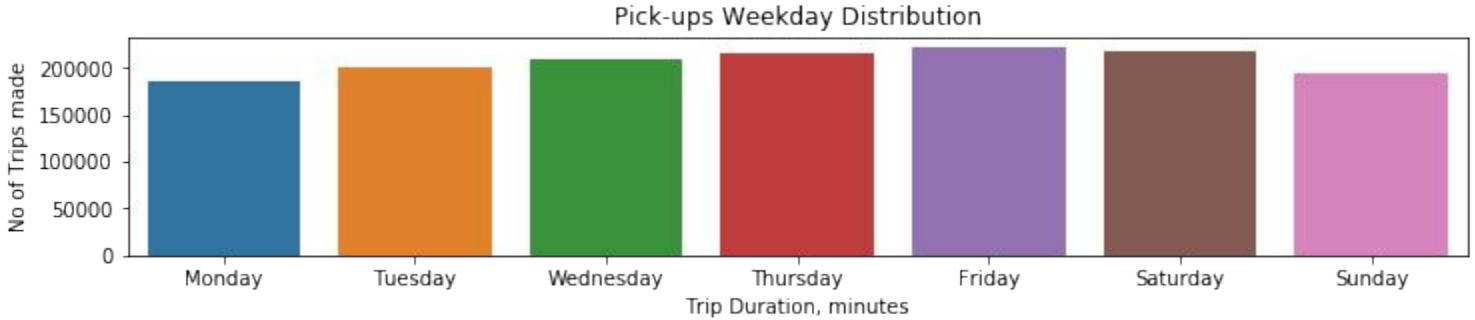

UBER
lyft
Lyft


DATA
MACHINE LEARNING
OPTIMIZATION
NYC TAXI DATA
XGBOOST
EVOLUTIONARY ALGORITHM

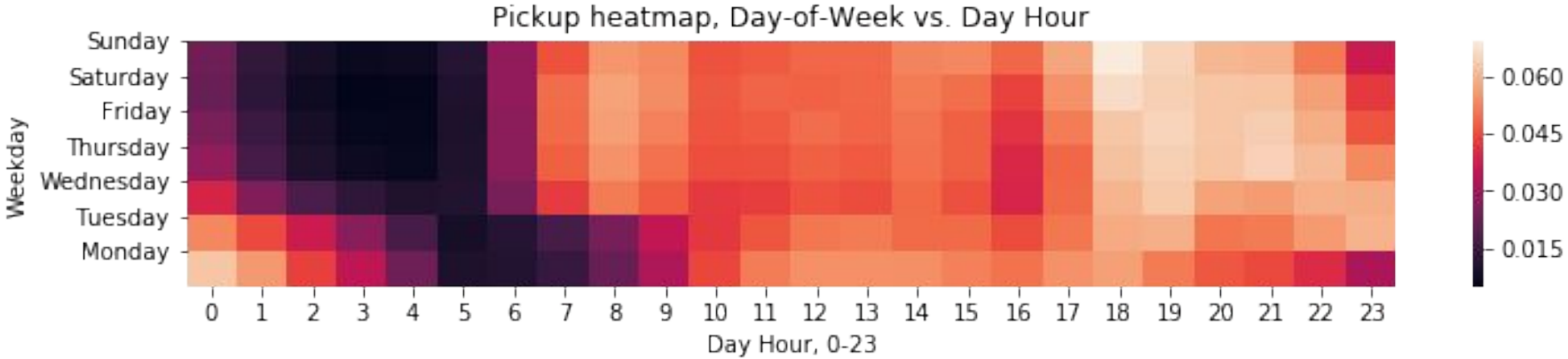
NYC Taxi and Limousine Commission Trip Record

- 1.5 Million Trips
- Year 2016
- 11 Attributes:
    - Id
    - Vendor_id
    - Pickup_datetime
    - Dropoff_datetime
    - Passenger_count
    - Pickup_longitude
    - Pickup_latitude
    - Dropoff_longitude
    - Dropoff_latitude
    - Store_and_fwd_flag
    - Trip_duration

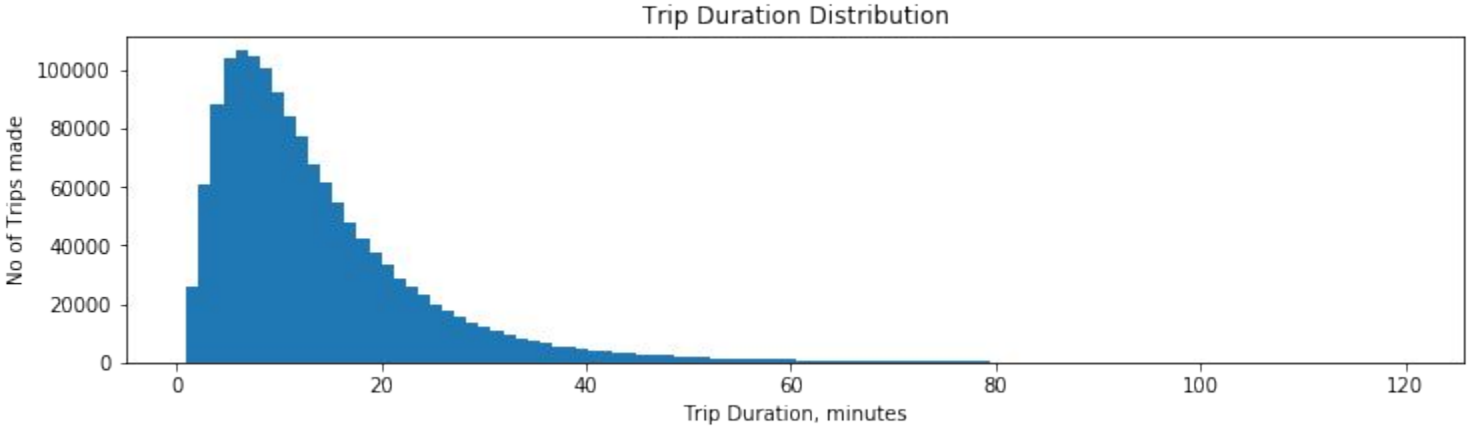
Pick-Up Times

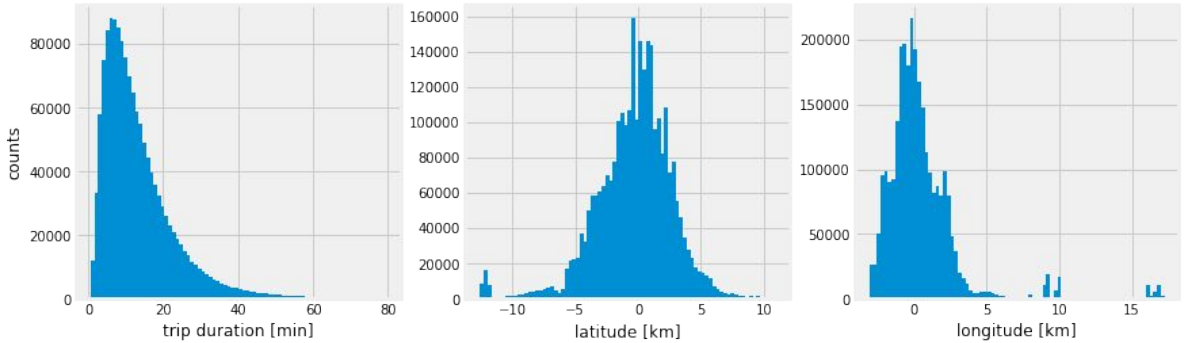


Pick-Up Times

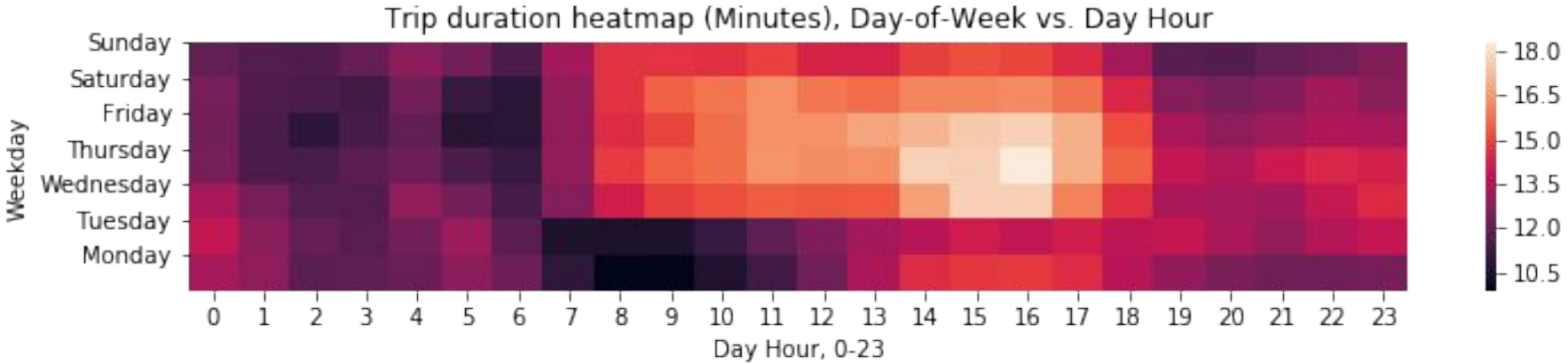


Trip Duration

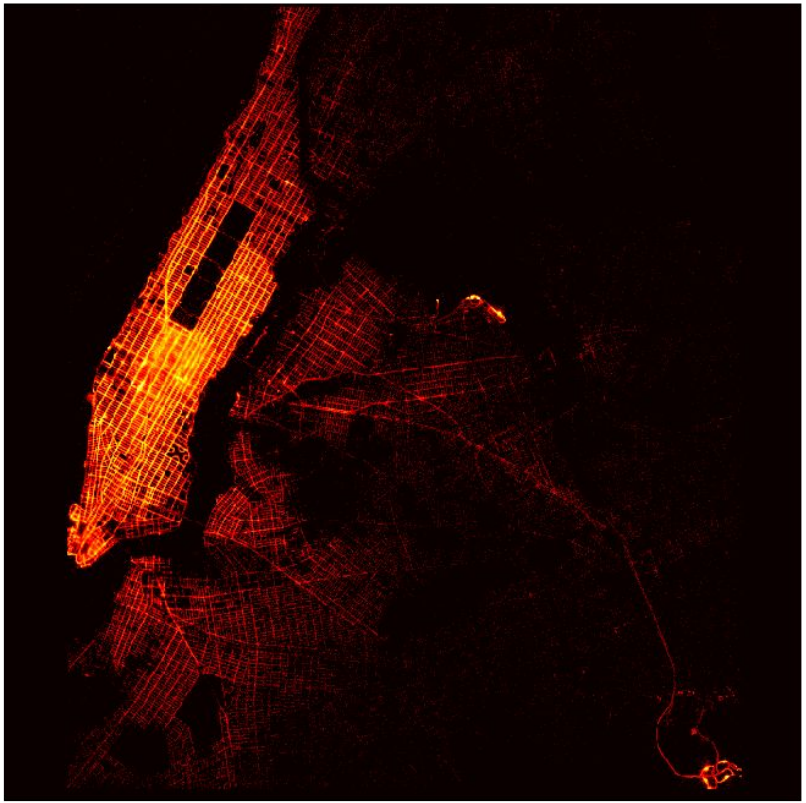


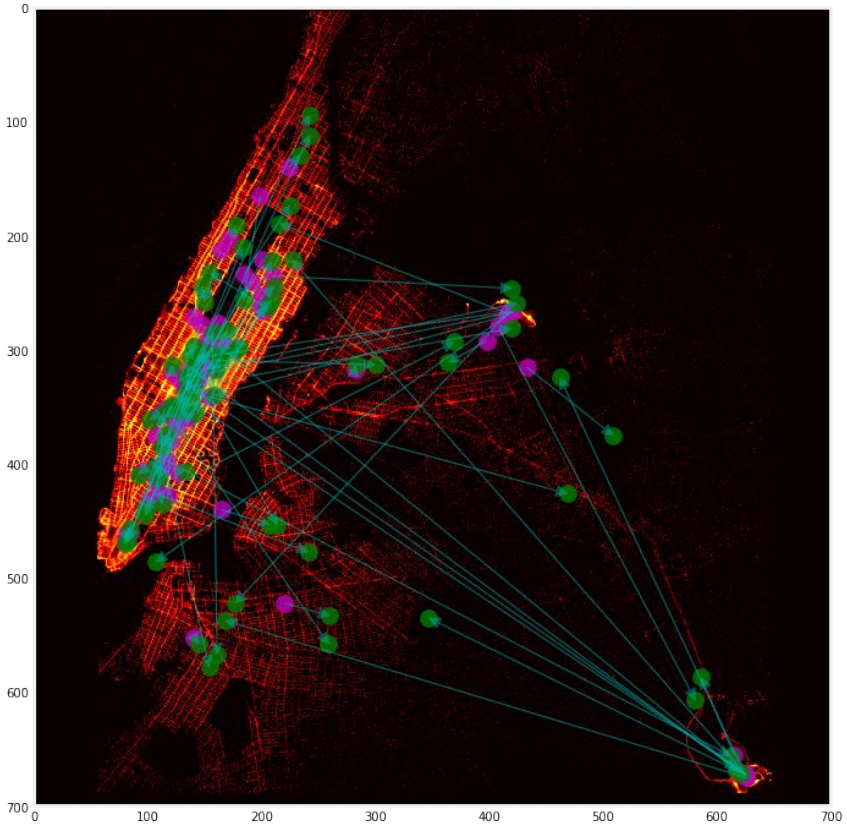


Trip Duration



Pickup and Dropoff Locations





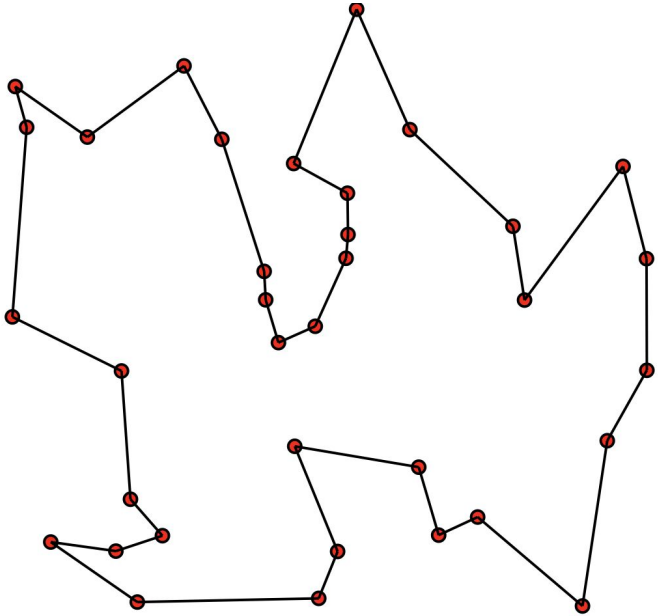
Machine Learning

- Features: passenger_count, pickup_longitude, pickup_latitude, dropoff_longitude, dropoff_latitude, store_and_fwd_flag
- Target: trip_duration
- XGBoost's Hyperparameters:
  - Learning_rate = 0.05
  - Max_depth = 14
  - Subsample = 0.9
  - Silent = 1
  - Feval = rmsle

Proposed Solutions

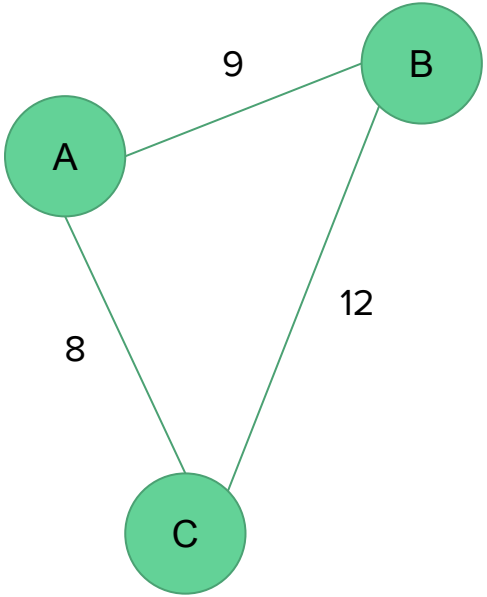
Ant Colony & Genetic Evolution Optimization

Traveling Salesman Problem (TSP)



$$O(n) = \left[\frac{1}{2}(n-1)!\right]$$

Ant Colony Optimization



$$\tau = \begin{pmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{pmatrix} \quad \eta = \begin{pmatrix} 0 & \frac{1}{9} & \frac{1}{8} \\ \frac{1}{9} & 0 & \frac{1}{12} \\ \frac{1}{8} & \frac{1}{12} & 0 \end{pmatrix}$$

Ant Colony Optimization

$$p_{ij} = \frac{[\tau_{ij}]^{\alpha} [\eta_{ij}]^{\beta}}{\sum_{h \in \mathcal{E}} [\tau_{ih}]^{\alpha} [\eta_{ih}]^{\beta}}$$

Ant Colony Optimization

```
# Update phermones based on total path cost
self.pheromone_delta[i][j] =
    self.colony.Q /
    self.total_cost
```

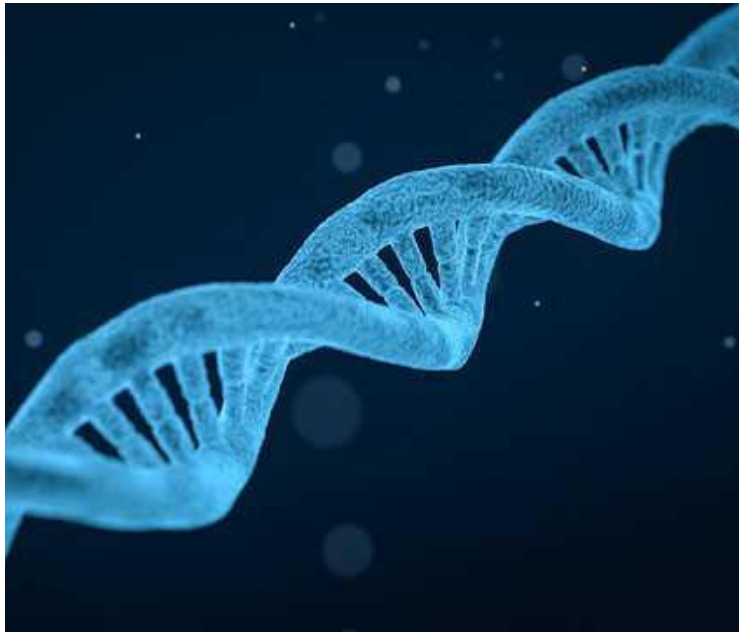
```
graph.pheromone[i][j] *= self.rho
for ant in ants:
    graph.pheromone[i][j] += ant.pheromone_delta[i][j]
```

Ant Colony Optimization

$$p_{ij} = \frac{[\tau_{ij}]^{\alpha} [\eta_{ij}]^{\beta}}{\sum_{h \in \mathcal{E}} [\tau_{ih}]^{\alpha} [\eta_{ih}]^{\beta}}$$

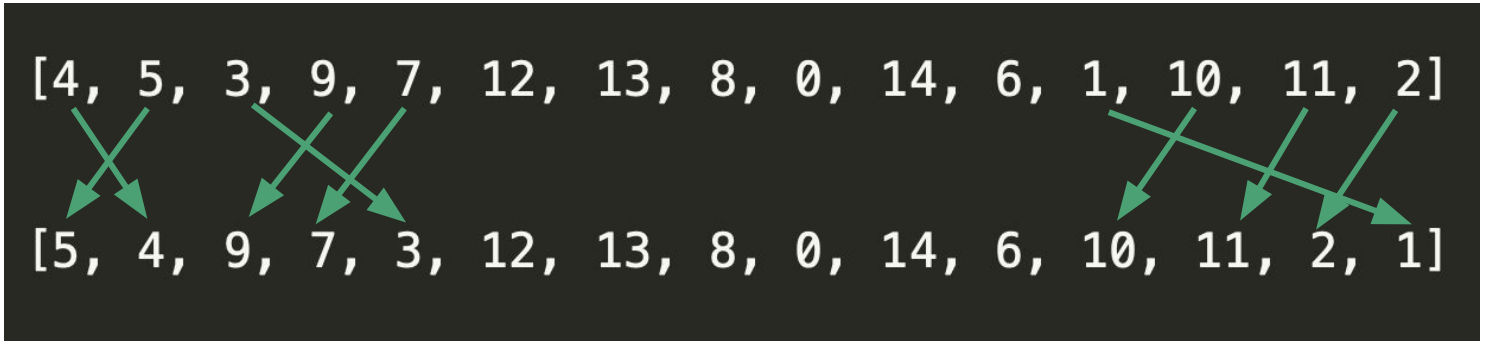
Genetic Evolution Optimization

1. Create a population of routes
2. Mutate
3. Crossover
4. Determine the fitness (travel time)
5. Select parent for next generation
6. *Repeat from step 2*



Mutation

```
Original Path:
[4, 5, 3, 9, 7, 12, 13, 8, 0, 14, 6, 1, 10, 11, 2]
```




```
Mutated Path:
[5, 4, 9, 7, 3, 12, 13, 8, 0, 14, 6, 10, 11, 2, 1]
```

Crossover

```
Current Path:
[2, 6, 10, 0, 7, 3, 14, 11, 1, 9, 8, 4, 5, 13, 12]

Mutated Path:
[14, 12, 13, 10, 11, 8, 0, 3, 7, 1, 5, 4, 2, 6, 9]

Offspring Path:
[14, 12, 13, 0, 7, 8, 14, 11, 1, 1, 5, 4, 2, 6, 12]
```

Crossover

```
Offspring path with duplicates removed:
[14, 12, 13, 0, 7, 8, 11, 1, 5, 4, 2, 6]

Missing locations:
[3, 9, 10]

Final offspring path:
[14, 12, 13, 0, 7, 8, 11, 1, 5, 4, 2, 6, 3, 9, 10]
```

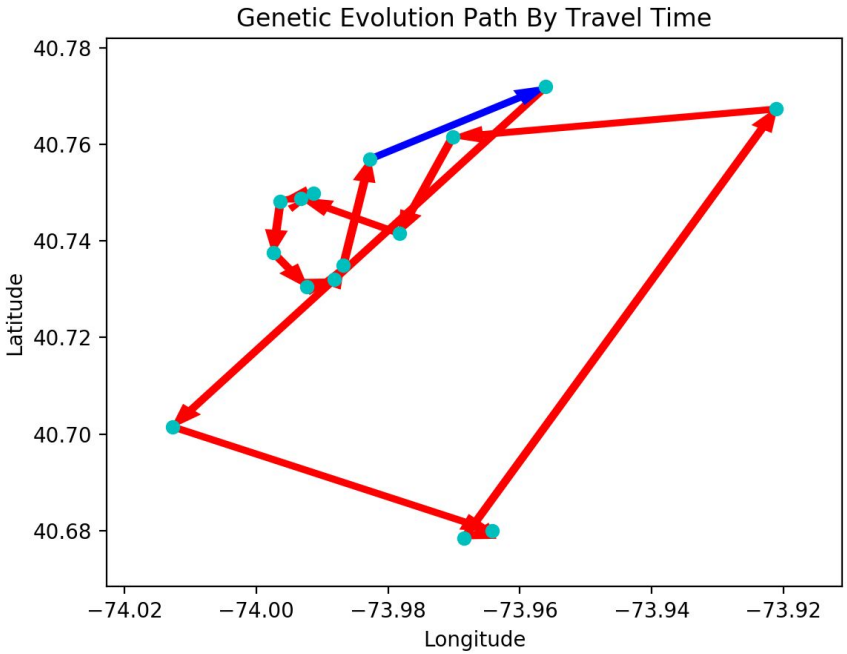
Evaluate Fitness

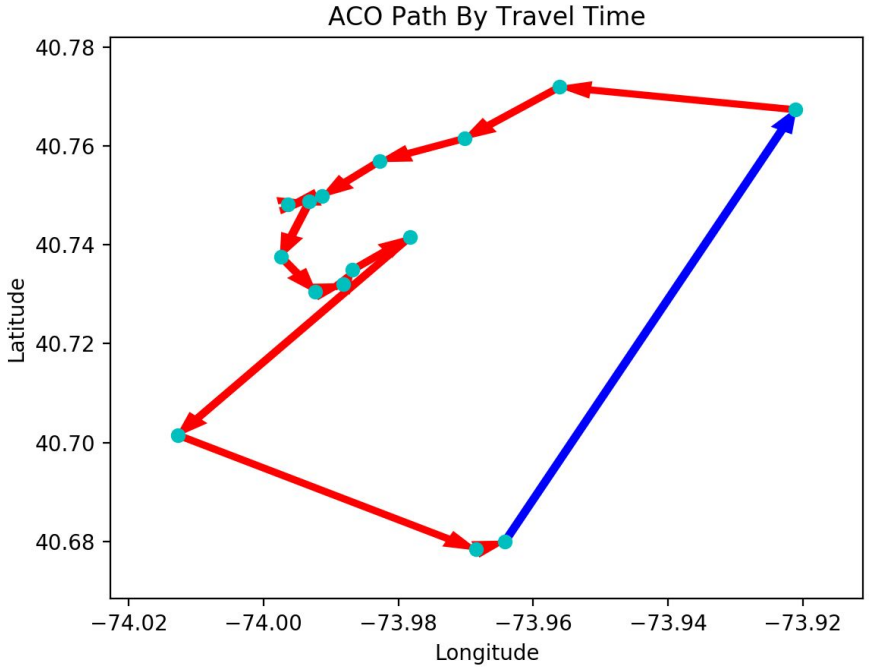
```
# Selection
# Select parent based on smaller path cost
candidate_cost = total_cost_from_path(candidate)
curr_cost = total_cost_from_path(curr_element)

if candidate_cost < curr_cost:
    population[i] = copy.copy(candidate)
```

Experimental Results

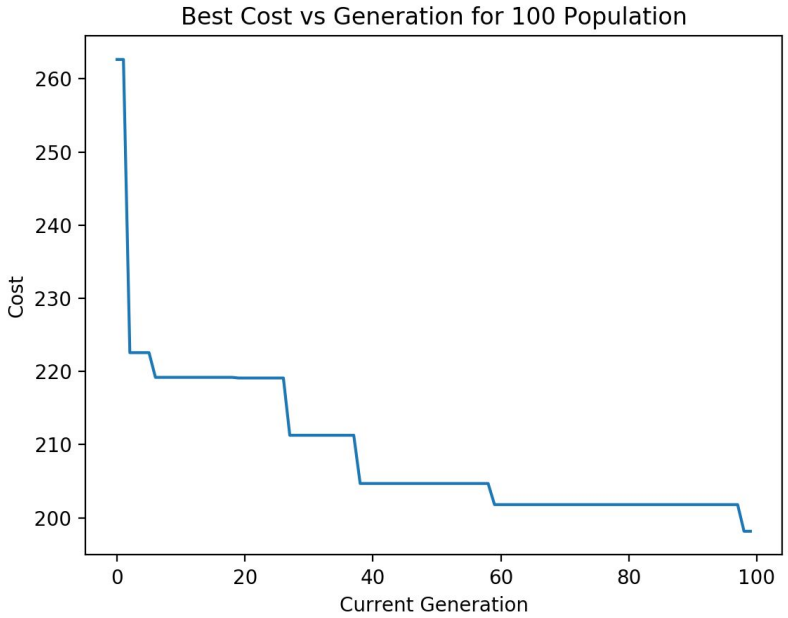
Ant Colony & Genetic Evolution Optimization

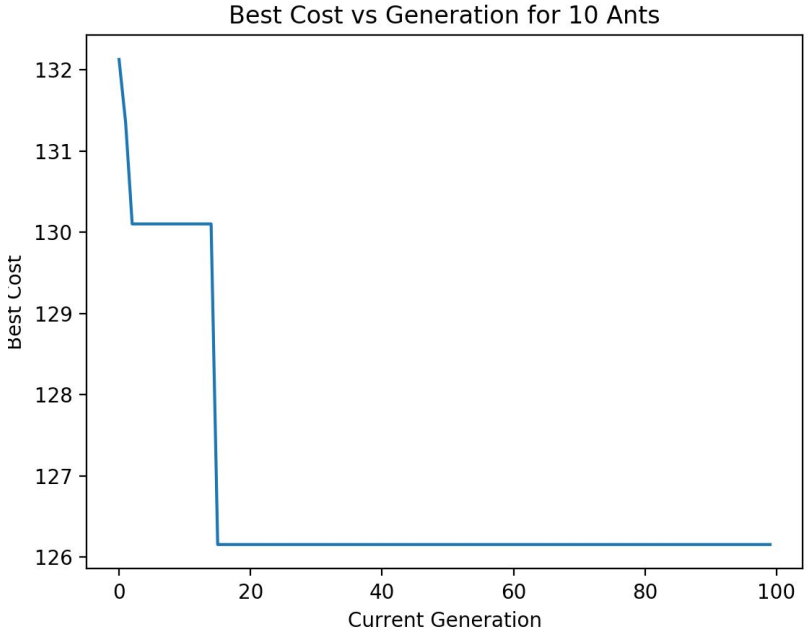
Generated Optimal Paths



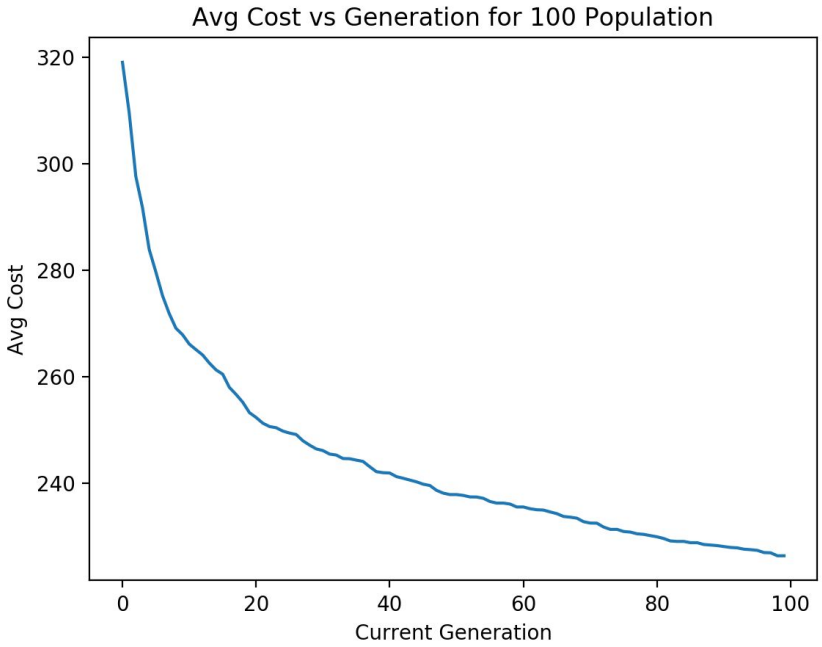


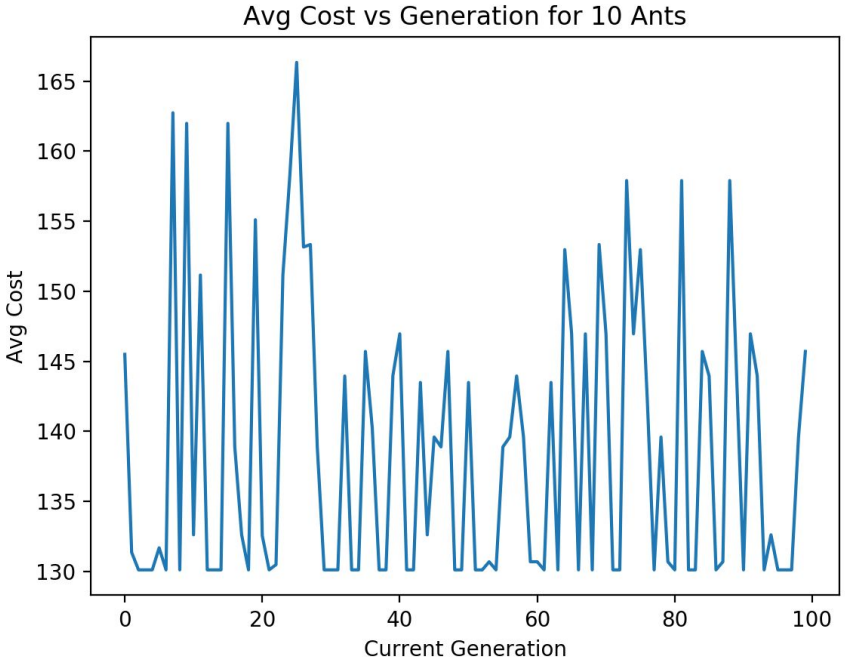
Optimal Cost Per Generation



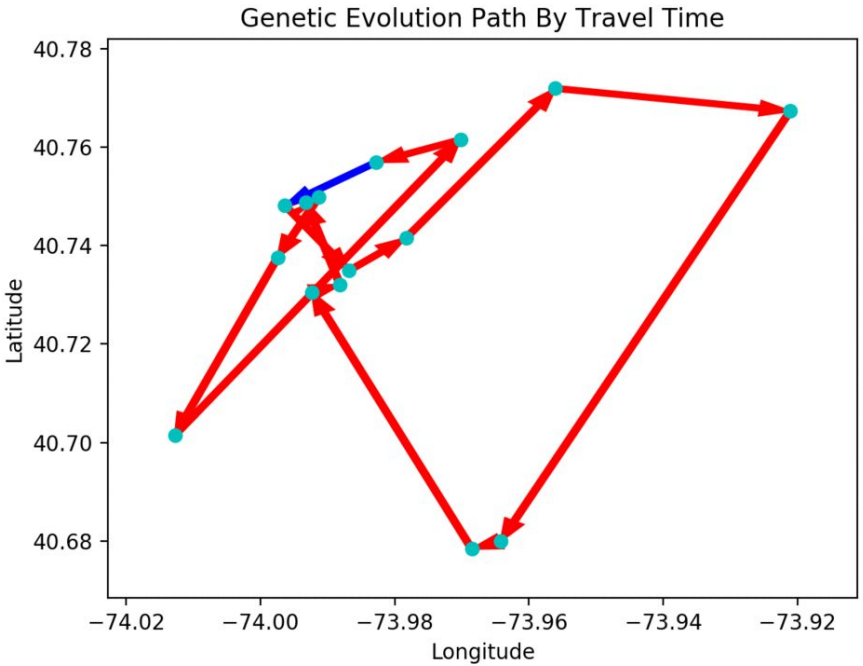


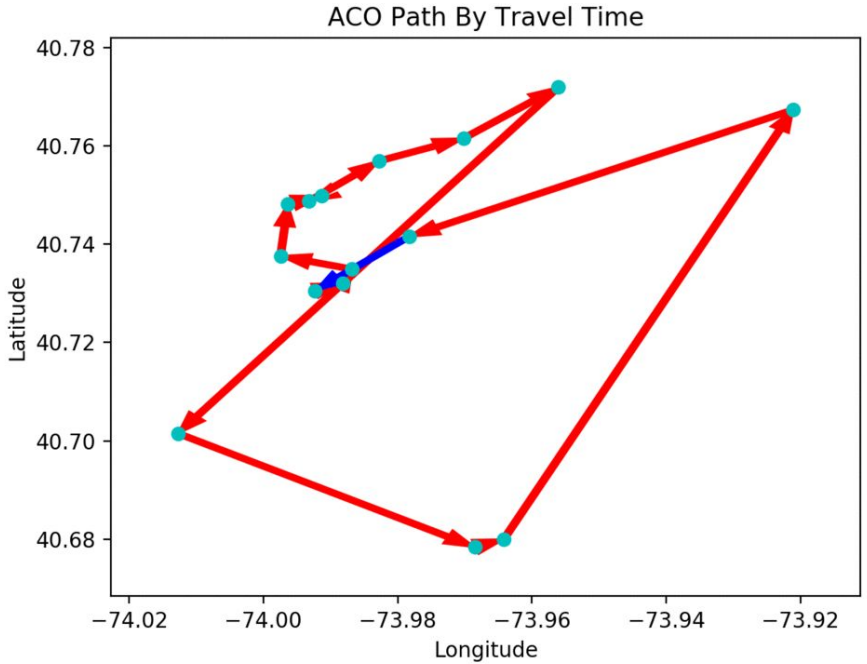
Average Cost Per Generation





Generated Paths






AC636
T615787C
3DAYS TO KILL