
Rideshare Service
Demand Model
...

Agenda

- Research Question
- Dataset
- Introduction to Approaches
- Modeling

Problem of Interest

Goal

Construct transportation demand model. Use ride characteristics (time requested, weather, origin and destination) to predict demand for rides in any US city.

Motivation

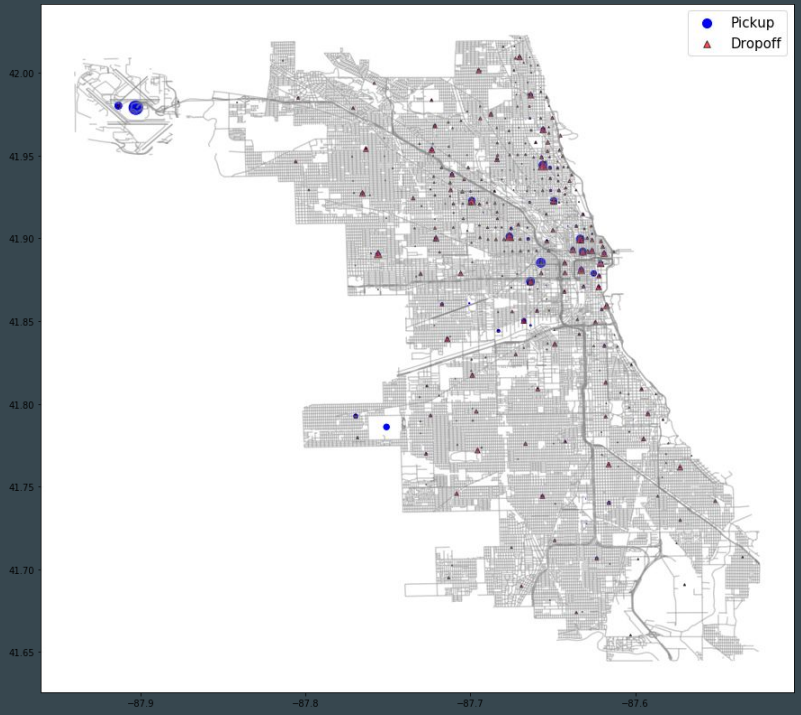
- Given information regarding an area's geospatial characteristics and demographics:
  - Location of airports, hospitals, postal services, universities, restaurants etc.
  - Population, gender ratio, age
  - Weather and time of day
- Important to whom?
  - City planners - estimating traffic flow and road usage, required width
  - Transportation providers

Overview of Data

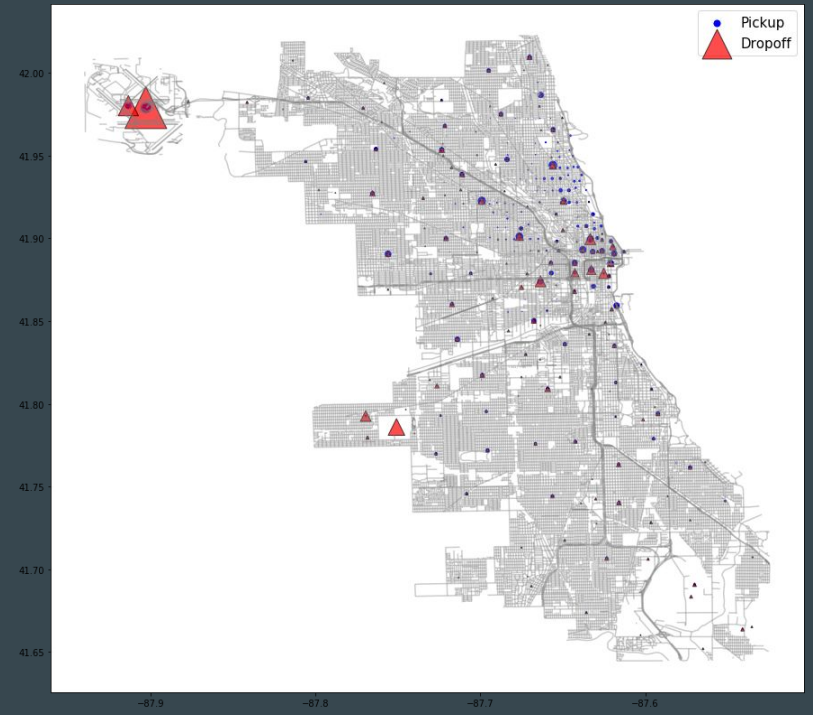
CDP: Transportation Network Providers - Trips

- 4.3 Million rideshare trips by major companies (Uber, Lyft).
  - All rides taken in Chicago in September 2019
- Relevant data:
  - Trip pickup and dropoff locations & times

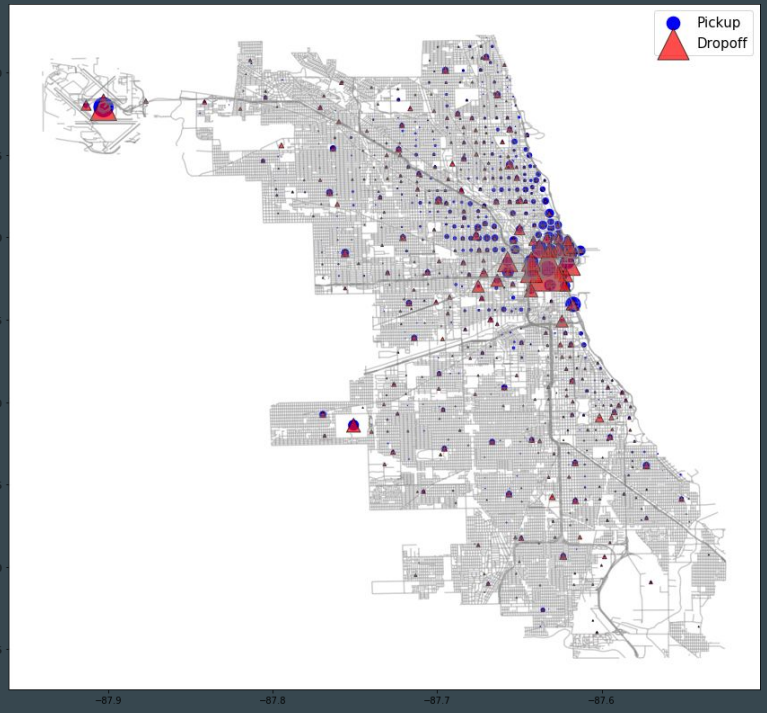
Motivation



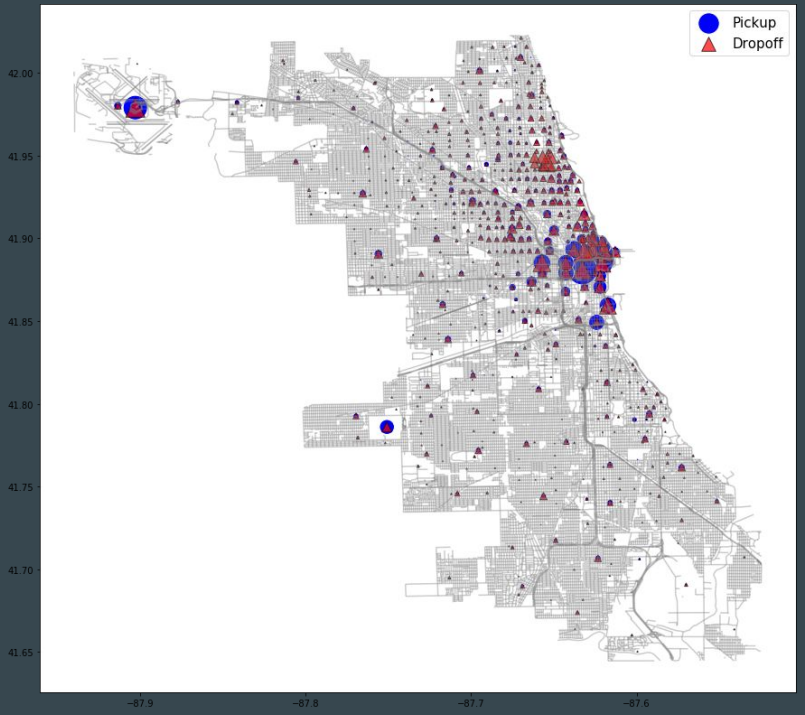
12am



5am


Pickup
Dropoff
42.00
41.95
41.90
41.85
41.80
41.75
41.70
41.65
−87.9
−87.8
−87.7
−87.6

8am


Pickup
Dropoff
42.00
41.95
41.90
41.85
41.80
41.75
41.70
41.65
−87.9
−87.8
−87.7
−87.6

6pm

Overview of Data

US Census: Density and Demographic stats on a tract level:

- Car ownership, Age Stats, Population density, Use of public transit, Average income of the area

World Weather Online: Weather Data

- Weather data for the city of Chicago

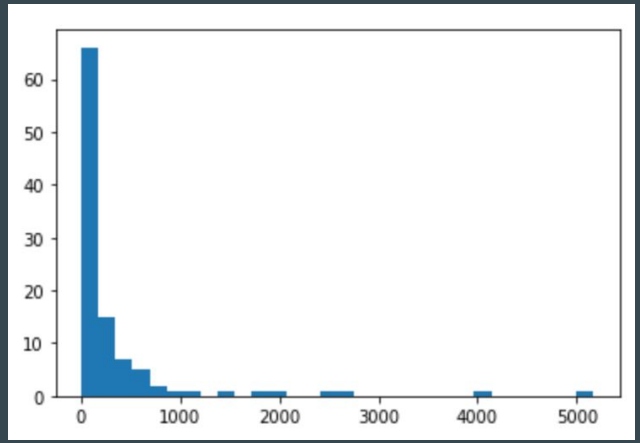
Overview of Data

Google Cloud

Google Maps Platform

Google Maps Places API:

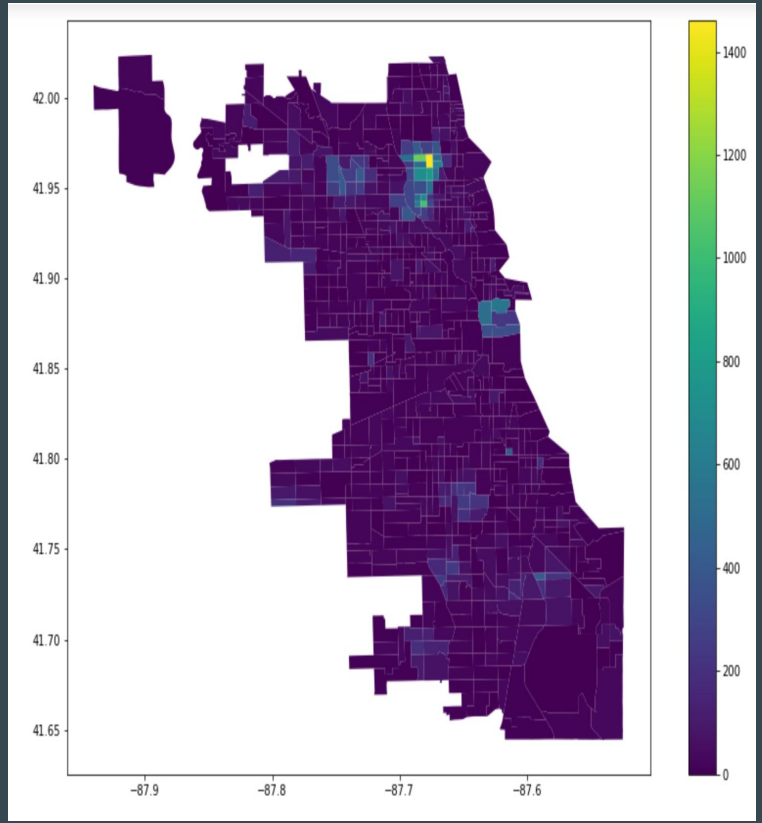
- Pulled all 34k listed Places Of Interest (POI) in Chicago.
- Places spanned 104 categories including:
  - Museums, restaurants, office buildings, bus stops, etc.



POI investigation

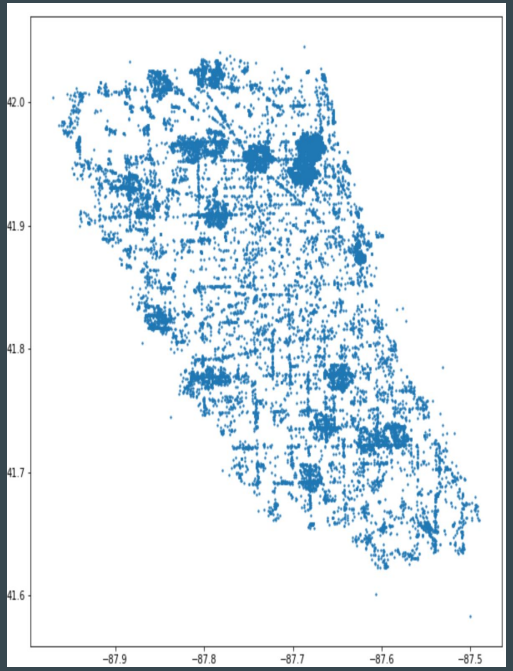
Top two tracts (POIs/area).

- Both in Lincoln Square
- No data errors inflating numbers
- Area very close to average for tracts



Data aggregation

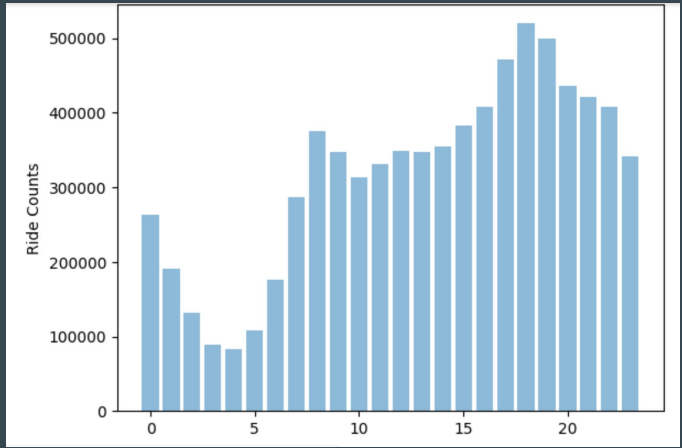
- **Points Of Interest (POIs):**
  - Calculated area of tract
  - Created summary metrics including total POIs in tract, total transportation POIs in tract, density and transportation sites
- **Aggregated rides, weather, demographic information and POI data by:**
  - Origin tract, destination tract, and hour.
  - 4M x 400



Initial Data

	origin	dest	date	hour	Trip ID	geoid10_x	Area_x	accounting_x	airport_x	amusement_park_x	...	transit_station_y	travel_agency_y
0	010100	010100	09/01/2019	4	1	1.703101e+10	383167.215833	0.0	0.0	0.0	...	0.0	1.0
1	010100	010100	09/01/2019	8	1	1.703101e+10	383167.215833	0.0	0.0	0.0	...	0.0	1.0
2	010100	010100	09/02/2019	4	1	1.703101e+10	383167.215833	0.0	0.0	0.0	...	0.0	1.0
3	010100	010100	09/03/2019	6	1	1.703101e+10	383167.215833	0.0	0.0	0.0	...	0.0	1.0
4	010100	010100	09/03/2019	9	1	1.703101e+10	383167.215833	0.0	0.0	0.0	...	0.0	1.0

time	temp	windspeedMiles	precipMM	humidity	visibility	WindChillF	uvIndex
400.0	66.0	11.0	0.3	80.0	28.0	66.0	0.0
800.0	66.0	9.0	0.2	76.0	28.0	66.0	4.0
400.0	68.0	5.0	0.0	82.0	28.0	68.0	0.0
600.0	70.0	14.0	0.0	80.0	28.0	70.0	0.0



Feature Engineering:

- POIs
  - Deleted high-count generic POIs (examples: 'point_of_interest', 'establishment', 'route').
  - Limited outliers to increase power (80th & 20th percentile)
- Weather
  - Created indicator of weather into good and bad by precipitation, temp and wind chill **(Weather G/B)**.

Pre-processing

- Normalization: Normalized values of each predictor column
  new value = (original val - mean) / std dev
- Grouped rides by tract OD and:
  - **Day In Week** (Sun, Mon, etc)
  - **Weekdays** (Grouped Mon-Thurs Only)
  - **Time-Of-Day** (morning, midday, evening, midnight)
- Performed dimensionality reduction using PCA
- Performed Log transformation on Y

PCA (95% variance explained)

```
[0.15305225 0.15039558 0.02466934 0.02267719 0.02159068 0.02092735
 0.01685274 0.01569661 0.01522639 0.01486177 0.01442448 0.0123659
 0.01114031 0.01041473 0.01034536 0.01006102 0.00986572 0.00964621
 0.00919015 0.0090341  0.00829029 0.00813758 0.00807007 0.0079741
 0.00728948 0.00713199 0.00688121 0.0068354  0.00663899 0.00652379
 0.00633406 0.00621579 0.00615393 0.00600875 0.00577673 0.00571802
 0.00557245 0.00547037 0.00539157 0.0053415  0.00525577 0.00518537
 0.00514744 0.00506722 0.0050228  0.004937   0.00477523 0.00471079
 0.00466216 0.00462025 0.00460519 0.00455902 0.00454344 0.00449385
 0.00447984 0.00442354 0.00426418 0.00420622 0.00413415 0.00406064
 0.00404097 0.00400147 0.00395814 0.00391964 0.00382443 0.003773
 0.00364757 0.00355225 0.00352429 0.00348272 0.00346004 0.00340273
 0.00336685 0.00331273 0.00326511 0.00324677 0.00311998 0.00305417
 0.00302006 0.0029353  0.00291164 0.00286677 0.00281897 0.00277989
 0.00277221 0.00268548 0.00261925 0.00256093 0.00250328 0.00247149
 0.00241733 0.00239625 0.00237484 0.00234011 0.00230534 0.00226381
 0.00223907 0.00209697 0.00206381 0.00203159 0.00202399 0.00198806
 0.00197819 0.00195167 0.00191875 0.001914   0.00189188 0.00186983
 0.0018558  0.00181441 0.00178965 0.00177691 0.00169146 0.00167126
 0.00162894 0.0015907  0.00155203 0.00153487 0.00147269 0.00146682
 0.00144513 0.00140886 0.00137587 0.00134612 0.00129982 0.00126884]
126
```

	0	1	2	3	4	5	6	7	8	9	...	120	121	122	123	124	125	origin	dest	hour	day_of_week
0	5.472259	9.165699	-2.453141	-3.833495	3.265549	-2.255301	-1.374904	-0.644396	-1.656844	1.893699	...	0.083614	1.199540	-0.268145	0.316705	-0.751488	0.012051	250400	650500	5	6
1	-0.814974	-3.685809	-0.979362	1.414877	0.079769	-0.296090	-0.354687	-0.143862	0.229276	-0.721776	...	-0.026242	-0.110486	0.055299	0.256682	0.062465	-0.754331	221100	610300	12	6
2	-4.653852	-2.236128	0.707865	-0.005786	-0.374371	-0.016147	-0.203579	0.198490	-0.164207	0.218632	...	-0.067409	-0.027586	0.099063	0.036727	0.012741	0.108532	62800	240600	17	0
3	-3.485493	2.374225	1.947933	0.663628	-0.877530	4.166300	4.705410	6.360843	3.115046	9.430581	...	-0.121109	-0.006650	0.157086	-0.000304	0.017208	0.034160	330100	832900	23	4
4	-1.496863	-3.626220	-0.183288	1.238121	-0.139024	-0.227487	0.556357	-0.132157	1.450837	-0.637493	...	0.169295	1.091841	0.325649	-0.368395	-0.257768	0.343953	60800	243500	21	2

Models

Radiation Model

- Applied the radiation model algorithm
- Used Population, POIs, distance and existing data as inputs.

MLP

- Multi-layer Perceptron classifier.
- This model optimizes the log-loss function with stochastic gradient descent.

Boosting

- AdaBoost, Gradient Boosting
- Boosting is a method of converting weak learners into strong learners. Increase weight of “weak” points
- While the AdaBoost model identifies the shortcomings by using high weight data points, gradient boosting performs the same by using gradients in the loss function

Model Results

		Extended Radiation Model	Classifier*	Classifier + MLP Regressor*	MLP Regressor	MLP Regressor*	Adaboost Regressor*	**Gradient Boosted Regressor**
Grouping (In addition to OD tract pairs)	Day in Week		X	X	X	X	X	**X**
	Weekdays	X						
	Time-of-day		X	X		X	X	
	Hour				X			**X**
	Weather G/B							
Data Transformation	Outlier Limiting		X	X	X	X		**X**
	PCA		X	X	X	X		**X**
	Log(Y) (base e)		X	X	X	X		
Results	CPC (Avg)	0.35	.35	.40	0.702	0.659	.61	**0.735**

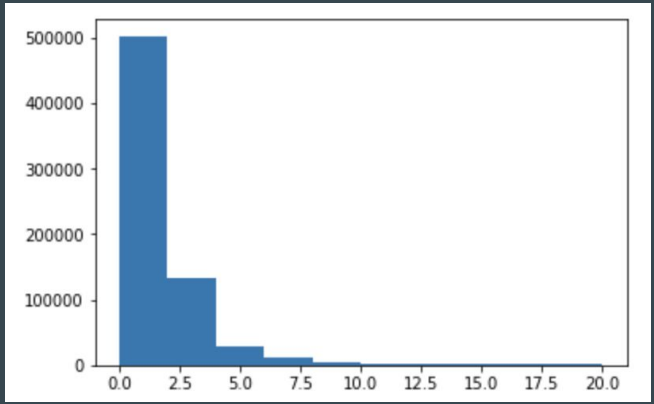
* Ran models without Log transformation with lower CPC. Also ran models on Log e, log 2, log 10.

Future Work

- Getting better separation of predicted small values (0 - 1)
- Skewed distribution (many 1s)
  - Changing metric away from CPC
  - Sampling Based Mitigation
- Tuning Gradient Boosting parameters

Related Work

- Short-Term Forecasting of Passenger Demand under On-Demand Ride Services: A Spatio-Temporal Deep Learning Approach
- Supply-demand Forecasting For a Ride-Hailing System
- Forecasting Uber Demand in NYC (Not a paper but great resource)
- Forecasting demand with limited information using Gradient Tree Boosting



Thank you!