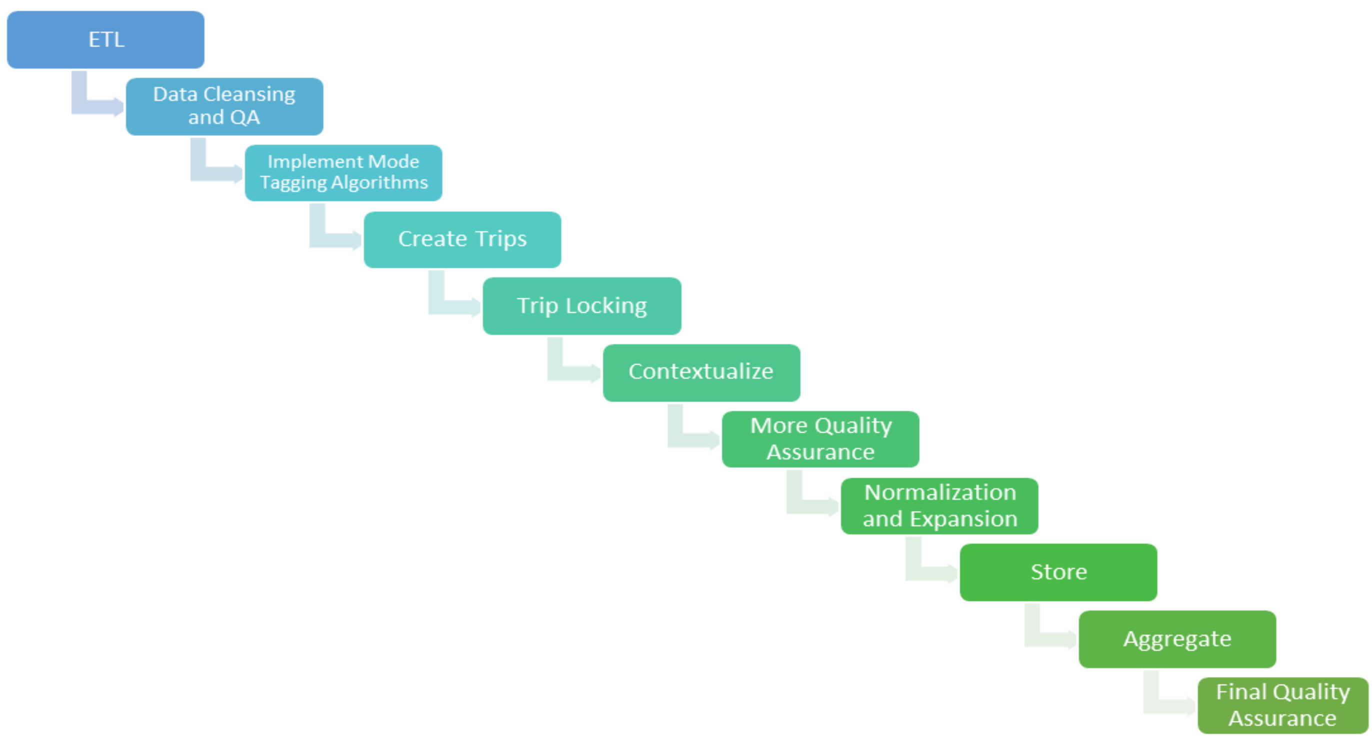
Methodological Document for Traffic Congestion

Streetlight General Information and Training

Data Collection and Resources:

StreetLight obtains its data from third-party data suppliers. Location-based services (LBS) data is the type of source data (for personal vehicles). StreetLight also gathers information about commercial trucks from their connected transportation management systems.

Data Processing Methodology:



11- Steps Data Processing Methodology

1. **ETL (Extract, Transform, and Load):**

First, StreetLight gather data in large batches from the safe cloud environments of their suppliers. Depending on the source, this can happen every day, every week, or every month.

Before being acquired and processed by StreetLight, the data was de-identified by the vendors, therefore StreetLight is not in possession of any data that contains personal information.

The ETL process transfers data safely from one environment to another, removes spurious or corrupted points, rearranges data, and indexes it for quicker access and more effective storage.

2. **Data Cleaning and Quality Assurance:**

Following the ETL process, StreetLight run several automated quality assurance tests to determine key data parameters. To name a few examples, they run tests on:

- Confirm that the data volume has not changed unexpectedly.
- Ensure that the data is correctly geolocated.
- Confirm that the data follows the same patterns as the previous batch of data from that supplier.

Furthermore, StreetLight personnel review key statistics about each data set both visually and manually. If any anomalies or flaws are discovered, StreetLight thoroughly examines the data. Any issues are escalated to our suppliers for discussion.

3. **Implement Mode Tagging Algorithms:**

StreetLight developed the following approach to infer the probable mode of travel after extensive research. Although they are not currently available in the StreetLight InSight® platform, they isolate air and boat/ferry trips first (available for custom analyses through our Services team), and then isolate the remaining modes as follows:

- Based on the fact that rail trips occur along a uniquely identifiable rail network, isolate rail trips using a set of heuristic rules.
- Separate and organize points into walking or stationary trips. A device continuously pinging at rest, for example, has a distinct signature when compared to modes of travel.
- Based on the movement patterns and spatial features described below, the remaining unassigned LBS pings are assigned mode probabilities for cars, bikes, and buses.

Finally, StreetLight investigated a subset of the features to determine what was most influential in the training of the random forest classifier. A model's features can be considered attributes or explanatory variables. This included assessing many ping-specific characteristics such as time, distance, speed, acceleration, circuity, and angular velocity for each ping as well as its preceding and subsequent pings, day of the week, and an hour of the day, and so on. Streetlight also took into account contextual factors like road classification, road network density, the presence of bike and bus lanes, the presence of rail lines, and the proximity to parks.

4. **Create Trips:**

This approach to trip creation is more complicated with LBS data due to the presence of multiple modes. We examine the pings in chronological order and assign a stream of points to a rail, stationary, or walk trip — or leave them unassigned. We then use the machine learning, model-based classifier to predict the travel mode of each remaining ping that has not been assigned to rail, stationary, or walk (as described in Step 3). After that, the stream of mode-tagged pings is routed to trip creation algorithms. When we see a sequence in which the mode probability changes, we terminate the current trip and begin the new trip with the new mode.

5. **Trip Locking:**

A trip from an LBS device comprises a series of linked pings. Because these pings do not necessarily originate on the network (road or otherwise), StreetLight must associate them with network links. This is known as trip locking (map matching). If the traveler turns a corner but the device only pings every 10 seconds, the intersection may be missed when all of the pings from the device are combined to form a trip. StreetLight uses OpenStreetMap (OSM) network information, including route types, speed limits, and directionality, to lock the trip to the network for walk, car, bike, bus, and rail trips. This locking process ensures that the entire route of the vehicle, bus, train, etc. is protected.

6. **Contextualize:**

StreetLight then integrates additional contextual data sets to add richness and improve accuracy. This included information about the road network, such as speed limits and directional cues, as well as geospatial features such as the presence of bus routes and bike lanes, proximity to parks, and road network density. StreetLight also use land use data, parcel data, census data, and other sources.

7. **More Quality Assurance:**

Because of the nature of our mode-tagging algorithms, mode-based quality assurance is more involved and includes the additional checks listed below:

- Individual ping mode classification: To test the classifier, we used standard machine learning algorithm testing techniques. We trained the model on 80% of the training data and tested it against the 20% we kept back and did not expose to the model.
- Trip unit-level testing: To test the start and end of the trips, we hand-picked 100+ unit test trips against which we verified the trip boundaries, overall trip travel mode, and individual ping mode as determined by the system.
- Meso-level trip testing: We examined specific regions, primarily metropolitan areas, to ensure that trip distributions appeared reasonable. We concentrated on areas with well-known behavior, such as bus and bike lanes, to ensure that our trips mirrored reality.
- Trip system testing: We ran a variety of tests on the total number of trips generated in each month of data, including visual checks, statistical checks, spatial and spatiotemporal checks, and real-world data comparisons.

8. **Normalization and Expansion to Estimated Trip Counts:**

The data is then normalized and expanded to generate a StreetLight Volume estimate or a StreetLight Index value. StreetLight need to design approaches to normalize our sample data to

make it comparable over time because we expect fluctuation in our suppliers' data sample month to month. For example, a 20% increase in our sample size one month may not imply that traffic has increased by 20%.
We can expand our sample to estimate the actual flow of travel for our LBS trips tagged as vehicle, bike, and walk. This procedure entails implementing machine learning models that generate monthly estimates for each mode at a given location.

StreetLight assessed that calculating monthly expansion ratios based on agency counts would be problematic due to the unpredictability of timely, available truth data due to the restricted availability of agency data and the high frequency of monthly updates in StreetLight's data processing pipeline. Instead, StreetLight calculates expansion ratios for each mode in a seed month in 2019, then adjusts scaling values month over month based on variation in our LBS vehicular penetration rates. As a result, to account for changes in our sample size, the StreetLight sample trips for bus and rail are normalized to the StreetLight Index.

9. **Store Clean Data in Secure Data Repository:**

The data is kept in a proprietary format once it has been transformed into patterns, verified for quality assurance, normalized, expanded, and contextualized. This allows for lightning-fast responses to queries via the StreetLight InSight® platform. When the data reaches this stage, it occupies less than 5% of the initial area of the data before ETL.

10. **Aggregate in Response to Queries:**

When a user runs an analysis in StreetLight InSight®, the platform automatically retrieves and aggregates the appropriate data from the data repository. For example, if a user wishes to know the percent of trips made from origin zone A to destination zone B versus destination zone C in September 2021, they can enter these values into StreetLight InSight®. Trips that began in origin zone A and finished in either destination zone B or destination C in September 2021 will be retrieved from the data repositories, aggregated and enlarged as needed, and sorted into the relevant metrics.

11. **Final Metric Quality Assurance:**

Final Metric quality assurance actions are completed automatically before sending results to the user. First, StreetLight InSight® assesses whether the analysis zones are suitable. The zone will be highlighted for review if it is a nonviable polygon shape, outside of the coverage region (for example, in an ocean), or too small (for example, evaluating journeys that finish at a single residence). A Metric will be warned if it delivers a result with too few trips or activities to be statistically valid or to safeguard privacy. When results are flagged, StreetLight's support team evaluates them to see if they are sufficient from a statistical or privacy protection standpoint. The support team then consults with the user to determine the appropriate next steps.

Sample Size:

The sample size and penetration rate for a given analysis are determined by the study's specific criteria. The reason for this is that some data is valuable for some analysis but not for others. Supplier data, for example, may provide high-quality, clean location data for one study but dirty, unusable location data — or no data at all — for another. Efficiently identifying data that is useful for a certain analysis is a fundamental component of StreetLight's data science value. Because penetration rates vary, sample sizes are provided for practically all StreetLight InSight® assessments automatically.

The sample size for LBS analyses is currently provided as the number of unique devices and/or journeys, depending on the type of analysis. For navigation-GPS analysis, the sample size is given as the number of trips. These values are conceptually comparable to automobile journeys.

Commercial vehicles that use up-to-date fleet management solutions are more likely to be included in StreetLight's navigation-GPS data set than unincorporated trucks for commercial navigation-GPS data analysis.

Streetlight Training

StreetLight InSight Training:

It is a platform of traffic congestion analysis. According to the clients' requirements, we can choose conditions including zones, model, vehicles, period, time. After running, we can gain metrics and visualized results. The metrics are used for an analysis.

In the training part, the platform supplies the beginner, medium and advanced levels.

After the training, you will be able to create analyses and retrieve relevant metrics to help answer transportation questions.

Zone Activity: type of analysis that allows us to capture the flow of vehicles and the demographics of travelers on a road without knowing their origin or destination

Workflow Tabs of Zone Activity:

1. Basic Info: Name, Mode of Travel, Country of Analysis, Unit of Measurement, Tag (optional), Output type, Description (optional)
2. Time Period: Date Period, preset data periods, day types (recommended is all days), day parts (times of day)
3. Zones: OSM Line Segments are for roads, Location Search bar, Vehicle Network (Road Classification)
4. Add Ons: Trip Attributes (Travel Time, Speed, Circuity), Traveler Attributes (Household income, Education, Race/Ethnicity, Family Status, Trip Purpose (*must go back to time period and change it to get most updated survey data for traveler attributes)), Home and Work Location

After analysis is available, click actions → open in Viz3D

Result Filters

1. Time Controls
2. Zone Selection

3. Metric Controls: Default is All vehicles volume but can also see percentage
4. Map Layers
5. Advanced Visualization Controls

Top right → click Zone Distribution Widget to get results
Below is Time Distribution Widget: can click select all day parts on the left for a line graph distribution (shows counts by the hour)
Below is Traveler Attribute Widget, can expand to see histograms/pie charts; can change day type attributes for different graphs

- Can click 3 dots to take screenshot or download csv file for graph
- Analysis details allows you to show all the chosen attributes as well as the sample size
- Download button (top right close to exit button) allows you to download metrics
- Can view different analysis for varying survey year data for traveler attributes (2020 census) → traveler attributes graphs will be different and new categories/info will be there for 2020 census data graphs
    - Data must be 2022 in order to retrieve 2020 US Census Metrics

Shapefile Links

https://www.census.gov/quickfacts/fact/table/rialtocitycalifornia,US/PST045221

https://www.arcgis.com/home/item.html?id=11a91a31e9fd40b3b2a56d88a7773304#overview

https://gisdata-scag.opendata.arcgis.com/datasets/city-boundaries-scag-region/explore?location=33.788849%2C-116.908250%2C8.82

https://ucr.maps.arcgis.com/home/item.html?id=bb0a77377d67482f90f8c7416e7c87f2&sublayer=0

COUNTY files
https://ucr.maps.arcgis.com/home/item.html?id=9859918e235f47ce90deff19bed75010

Steps for Zone Centroids for ArcGIS: 7/14

1. Turn on 2 layers: Rialto Boundary and Census Block Groups (from SCAG)
2. Create a new field to associate GEOID's to a specific geography named Jurisdiction
3. Perform a Select by Location where the target layers are the census block groups: this looks at any block groups that have their census within Rialto. Source Layer is the Rialto Boundary. Spatial Selection method is chosen as "have their centroid within the source Layer feature". This shows all census block groups that have their centroid within the boundary. Can set Jurisdiction to "Rialto".

    a. The above steps create essentially a lookup table that corresponds Census Block Group GEOIDs to our boundary.

Steps for Streetlight Trips to or from Pre-set geography: 7/14
to upload a new zone set:

1. go to zones
2. upload ARCgis folder with zip shapefile of Rialto
    1. for zone name choose city, for zone ID choose shapeID, for passthrough choose always no (OD zone), for direction leave as blank, calibrate data and hit save
3. you can now view zone set under zone tab

for analysis

1. choose trips to or from pre-set geography
2. for basic info
    1. name the analysis
    2. choose all vehicles as the mode of travel
    3. unit of measurement is files
    4. tags keep as blank
    5. output type: sample trip counts (device trip)
3. for time periods
    1. can do a custom date range (he did 2019 start to end)
    2. for day types, wrcog did weekday Tue-Thurs. and selected start and end
    3. for day parts he left it as is
4. for zones
    1. click choose from user generated zone set and add rialto zone
    2. select 2020 us census block groups (make sure its consistent with what u use in ARCgis!)
5. add ons: select trip and traveler attributes

Python Steps

1. Import the pandas and numpy library. Import the reduce function from the functools modules.
2. Read the CSVs exported from ArcGIS as DataFrames using the read_csv function.
3. In order to make clear which block groups are from Rialto, create a new dummy column.
4. Clean the total block groups dataframe. (Get rid of unnecessary columns)
5. Load in the csv files for the zone distributions from Streetlight. (All day, PM from, and PM to)
6. Merge all day from traffic analysis data with Block group data, then clean and remove similar columns. Do this for PMFrom and PMTo.

7. Drop any block groups that don’t have any traffic percentages.
8. Load in the county labels and clean the data by removing the first digit of the county FIPS.
9. Join the county labels with the data and create a separate data frame for Rialto.
10. Make a list for the county percentages and create the Rialto block group percentages.
11. Merge all county percentages and drop any unnecessary counties.
12. Add an index as a column and add the Rialto information.
13. Output the county percentages table showing totalTrafficAllDayFrom, totalTrafficPMTo, totalTrafficPMFrom, and location.
14. Export the relevant tables.

(**More detailed Python data cleaning Steps)**

Go to Python

1. Merge Trip Percentages from StreetLight along with the City assigned Block Group tables created in ArcGIS by their FIPS code
2. Clean and Remove any NA numbers from Block groups where Trip Percentages are NA or 0 in ALL traffic percentages
3. Aggregate the sums of each of the City’s traffic percentages in a loop to form the dataframe of Trip Distribution Metrics (Group by the City)
4. Clean the data once aggregated, and include County aggregations to the the dataframe
5. Find the Top 5 Destinations in the All-Day Metric

Steps for creating a Trip Purpose CSV in Python (home to work, home to other, non-home-based trips)

1. Merge the Trip Purpose percentages (when reading the dataframe make sure turn low-memory = False due to the large size of the dataframe) with the City assigned Block Group tables
2. Clean the produced table by removing all block groups with NA or 0 values in each Trip Purpose, and filter “All-Day” in the Day Parts column
3. Find the average for each City’s block groups trip purpose percentages to get the proportion of trips that were for Home to Work, Home to Other, and Non-Home based Trips and produce a Dataframe assigning these with the corresponding location
4. Merge the produced dataframe with the Percentage distribution table and re-assign the Home to Work, Home to Other, and Non-Home Based Trips column by multiplying their values with the percentage distribution tables to find the true percentage volume for each trip purpose
5. Export as a .csv for Excel Visualization purposes

Community Profile 2019

https://www.census.gov/quickfacts/fact/table/rialtocitycalifornia/BZA010220#BZA010220
https://censusreporter.org/profiles/16000US0660466-rialto-ca/

Population

103,549

Household

26,134

Income

$ 65,538

Employment

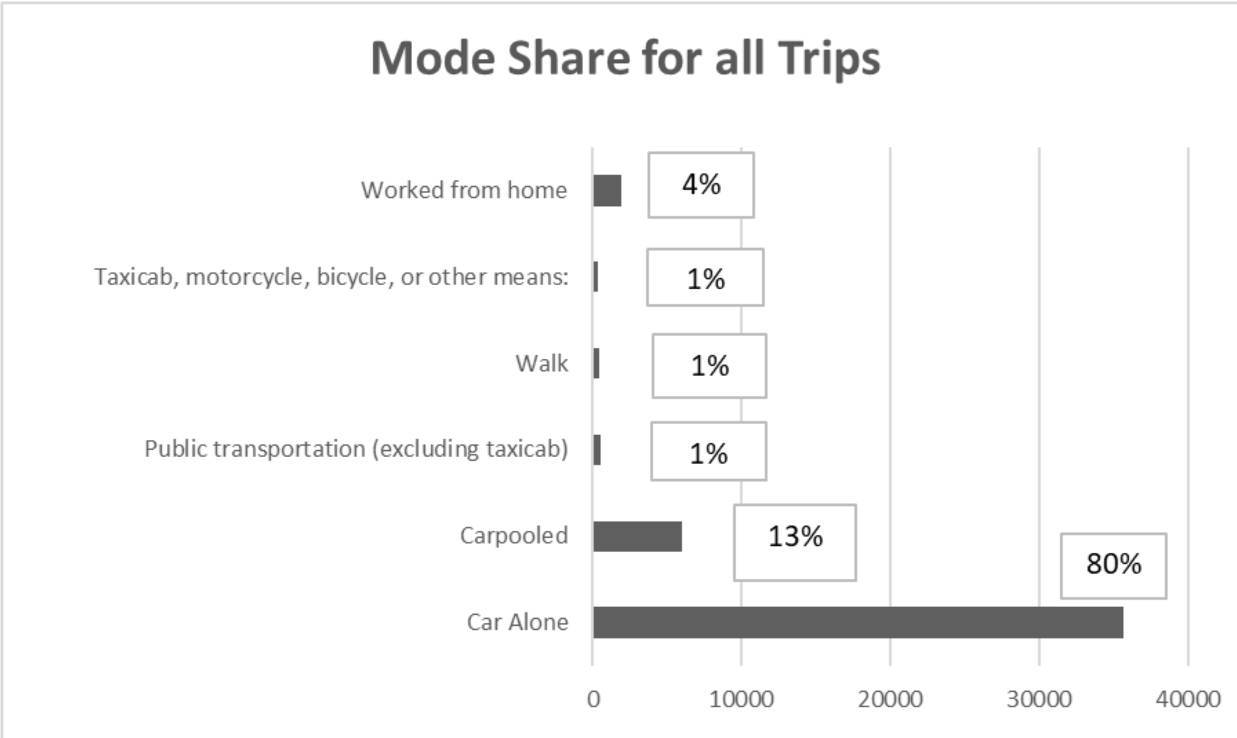
44,920

Mode Share All the Trips (2019):
Data Sources: Streetlight Data Analysis (Trip to/from pre-set geography analysis) and https://data.census.gov/cedsci/table?q=rialto%20transportation&y=2019&tid=ACSDT5Y2019.B08141

Methodology:
Utilized the US census Bureau data and filtered their lists with “2019” “transportation” & “Rialto city, CA”. This can be used for every city, county, and state as the data can be aggregated and combined easily.
Reasoning:
This data is important to us as it gives us insight into what type of vehicles are going through the city of Rialto and if there are any clues or pattern that might aid us in identifying what is causing the traffic congestion.

How to create bar chart in Excel

1. Using the Trip Purpose distribution table create a bar chart, the chart should be horizontal and include percentages (Chart Design, select your chart)
2. Click on the Plot area and make the Vertical Grid Lines white, and change horizontal and vertical axis colors if necessary (I chose black)
3. Change fonts to match previous fonts
4. Make charts stacked or separated by for each Trip Purpose
5. You can inverse order in Chart’s format options to make it from the top destination to the smallest destination
6. Choose font sizes and color options



Commute Distance and Direction (2019)
Data Source: https://lehd.ces.census.gov/data/
Note: Had to ask Streetlight with how to interpret this data

Steps to create radar chart for Commute Trip Distance and Direction with Live and Work share

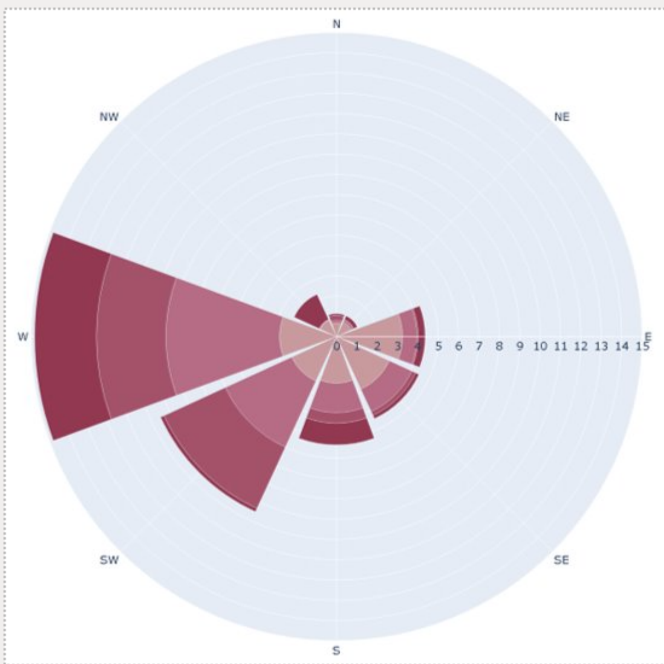
1. Go to the US Census Bureau called OnTheMap (https://onthemap.ces.census.gov/)
2. Enter the name of the City you want to pull the data for (In our case we're looking for Rialto, CA)
3. Select the type of analysis you would like to perform. For example, this analysis, we have select the Home/Work Area, Distance/Direction, the latest year and Job Type (we selected All Jobs)
4. Press Go

You can pull the detailed report in excel format and recreate those Radar Chart.

Spencer created these graphs using the plotly express library in Python

https://stackoverflow.com/questions/63713418/offset-polar-bar-radial-origin-python-plotly-express
https://towardsdatascience.com/improving-plotlys-polar-bar-charts-43f6eec867b7

https://community.plotly.com/t/using-overlay-barmode-with-px-bar-polar/51513

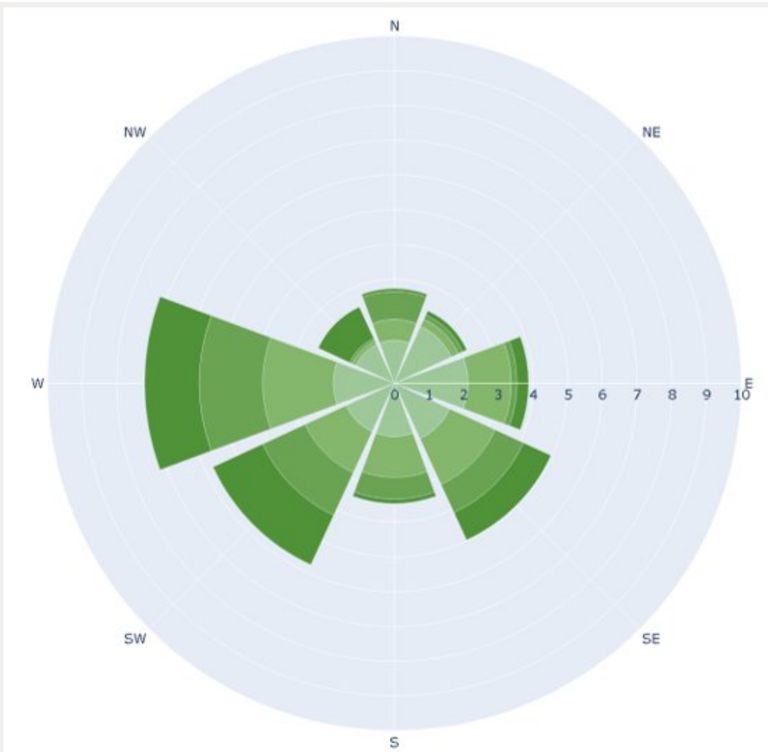


Commute Distance and Direction from Rialto

Less Than 10 miles	15,970	37.1%
10-24 miles	12,817	29.8%
25-50 miles	7,800	18.1%
Greater than 50 miles	6,428	14.9%

```python
import plotly.graph_objects as go
categories = ['E', 'NE','N','NW','W','SW','S','SE']
fig = go.Figure()

fig.add_trace(go.Barpolar(
    r=[3.238, 0.804, 0.676, 0.884, 2.797,2.419,2.294,2.858],
    name='Less Than 10 miles',
    theta=categories,
    marker_color='rgb(200,154,158)'
))
fig.add_trace(go.Barpolar(#base=0,
    r=[0.714,0.111,0.126,0.028,5.579,3.587,1.456,1.216],
    name='10-24 miles',
    theta=categories,
    marker_color='rgb(182,108,132)'
))

fig.add_trace(go.Barpolar(#base=0,
    r=[0.059, 0.017, 0.243, 0.019, 3.41, 3.326, 0.523, 0.203, 0.059],
    name='25-50 miles',
    theta=categories,
    marker_color='rgb(164,82,106)'
))
```

```
fig.add_trace(go.Barpolar(#base=0,
    r=[0.34, 0.191, 0.085, 1.357, 3.034, 0.175, 1.068, 0.178],
    name='Greater than 50 miles',
    theta=categories,
    marker_color='rgb(146,56,80)'
))


fig.update_layout(
  polar=dict(
    radialaxis=dict(
      visible=True,
      range=[0, 15]
    )),
    font_size=40,
  showlegend=True
)

fig.show()
```



Commute Distance and Direction to Rialto

■	Less Than 10 miles	12,985	41.5%
■	10-24 miles	8,147	26.1%
■	25-50 miles	5,878	18.8%
■	Greater than 50 miles	4,242	13.6%

```python
import plotly.graph_objects as go
categories = ['E', 'NE','N','NW','W','SW','S','SE']
fig = go.Figure()

fig.add_trace(go.Barpolar(
    r=[2.124, 1.805, 1.237, 1.228, 1.763, 1.536,1.535,1.757],
    name='Less Than 10 miles',
    theta=categories,
    marker_color='rgb(158,200,154)'
))
fig.add_trace(go.Barpolar(#base=0,
    r=[1.245, 0.228, 0.599, 0.111, 2.038, 1.3, 1.177, 1.449],
    name='10-24 miles',
    theta=categories,
    marker_color='rgb(132,182,108)'
))

fig.add_trace(go.Barpolar(#base=0,
    r=[0.162, 0.131, 0.813, 0.079, 1.825, 1.347, 0.628, 0.893],
    name='25-50 miles',
    theta=categories,
    marker_color='rgb(106,164,82)'
))

fig.add_trace(go.Barpolar(#base=0,
    r=[0.335, 0.131, 0.076, 1, 1.572, 1.572, 0.125, 0.870],
    name='Greater than 50 miles',
    theta=categories,
    marker_color='rgb(80,146,56)'
))

#fig.update_traces(text=['E', 'NE','N','NW','W','SW','S','SE'],)


fig.update_layout(
  polar=dict(
    radialaxis=dict(
      visible=True,
      range=[0, 10]
    )),
  font_size=40,
  showlegend=True
)

fig.show()
```

Steps ArcGIS visualization for Heat Maps

1. Using the Scag City Boundaries layer, create a layer from only San Bernardino county
2. Upload the created layer to the projects geodatabase (gdb) as well as the tables created for the trip distribution percentages
3. Merge the Layer Attribute table and the Stand-alone table by location (City either FIPS or Location)
4. Create a heat map by clicking symbology and choosing heat map in the Appearance tab, split by Standard Deviation, and clean the percentages so they only have 5 decimal places, you can change color in these properties as well as filter out the City of Interest (Rialto) by placing an expression where the Location = ‘City of interest’
5. In the labels tab, you can also place labels on the locations that have a certain percentage by creating a filter for the label only containing 4 digits (right function, 4 digits) for the Trip Distribution percentage and treating the decimal as a percentage
6. Create a layout and insert a map frame with the created map, you can ACTIVATE the map frame to zoom in to the location by right clicking the map frame and selecting that option, Note to insert a map WITHOUT labels on the same layout you must create another map and duplicate the map you created and turn off the labels and add it to the layout with the map frame.
7. On the insert tab, you can insert the North Arrow, Distance ruler, and a Legend
8. If you need to Reshape your mapframe, choose the Reshape option in top tab and choose what shape you need to reshape it to, make sure to remove mapframe borders by making their size 0 or removing their color
9. Repeat this process for each different Trip distribution percentage time slot