Greening Seattle

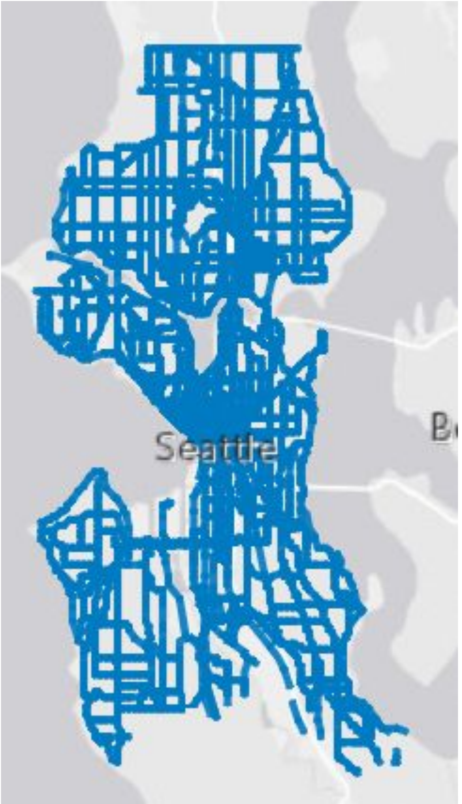
GIS Visualizations

Final Presentation

Sarah Pristash, Shaun Gallagher, Wenqi Cui

Problems

- Transportation accounts for 60% of total core emissions in Seattle, 61% percent of which is attributed to gasoline/diesel sources
- Population increased 25% from 2008 - 2018, projected to continue and intensify city-wide traffic burden
- Citizen survey data indicates current transportation must be more robust and equitable, especially for BIPOC communities
- Large data sets make it difficult to visualize



From Seattle.gov: 2018 Traffic Flow Counts

1. 2018 Seattle Greenhouse Gas Emissions Inventory

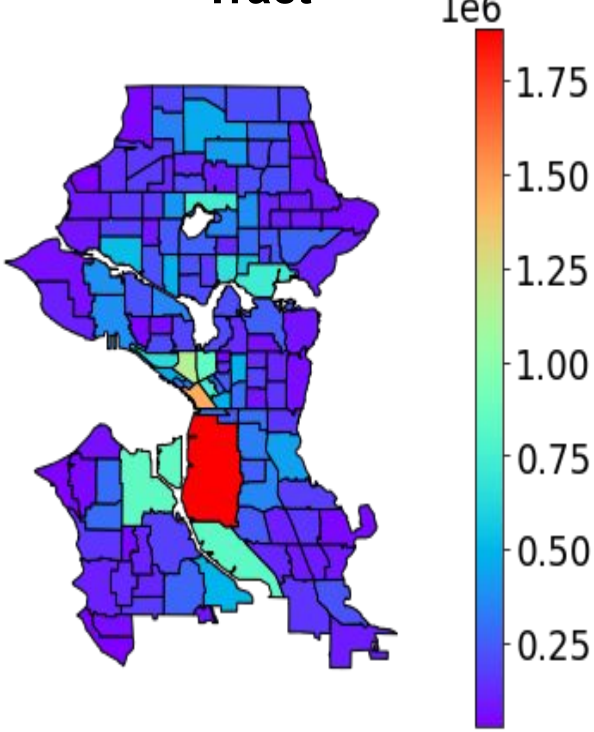
Background and Scope

- Modeling the impact on traffic can give an idea of which features most affect flow

- Visualizing these changes regionally allow for a regional approach to discover the best solutions for specific areas of Seattle

- **Questions:**
    - How can we use data science to predict traffic volumes based on urban features?
    - How can data from traffic flow be effectively visualized?

2015 City of Seattle Traffic Data by Census Tract



Chloropleth Plot Generated in geoplot

Technology Overview: Geographic Information Systems

Objective: Use python to create interactive plots to visualize traffic and input feature data over geographic areas

ArcGIS

GeoPandas

Most useful packages:

Geopandas: geographical data can be stored in Pandas-like dataframes. Geographies are represented in terms of Shapely shapes - points, linestrings, polygons, etc. Shapely spatial functions can then be performed while maintaining the link to the data

Shapely: allows us to merge data based on geographic location - i.e. point in polygon, line intersecting polygon, etc.

Folium: Creates visually pleasing, interactive, detailed maps of geospatial data

Data Processing Workflow

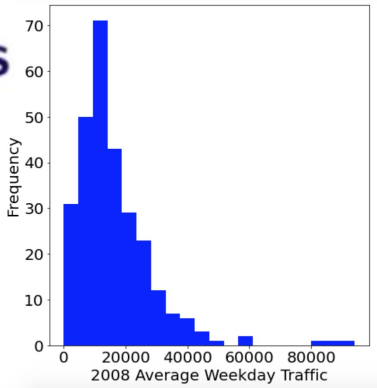
1

2008_Traffic_Flow_Counts

OBJECTID	GEOBASID	DOWNTOWN	COMPKEY	STNAME	OID_	NAME	YEAR	SEGKEY	AAWDT	IN
1	94250370	N	12522	RENTON AVE S	196	RENTON AVE S, N/O S CLOVERDALE ST	2008	12522	8700	
2	50000650	N	8702	AIRPORT WAY S	152	AIRPORT WAY S, N/O S NORFOLK ST	2008	8702	12700	
3	127150040	N	16114	N NORTHGATE WAY	31	N NORTHGATE WAY, W/O ASHWORTH AVE N	2008	16114	26700	
4	123450090	N	15794	N 65TH ST	166	N 65TH ST, W/O LINDEN AVE N	2008	15794	8800	
5	9550200	Y	2762	2ND AVE	90	2ND AVE, NW/O LENORA ST	2008	2762	11900	
6	80150420	N	11190	LAKE CITY WAY NE	161	LAKE CITY WAY NE, S/O NE 145TH ST	2008	11190	35300	
7	78850080	Y	11055	JAMES ST	63	JAMES ST, NE/O 7TH AVE	2008	11055	31900	
8	71050870	N	10795	GREENWOOD AVE N	98	GREENWOOD AVE N, S/O N 145TH ST	2008	10795	22900	
9	94250400	N	12525	RENTON AVE S	197	RENTON AVE S, SE/O S HENDERSON ST	2008	12525	7200	
10	31300190	Y	6138	4TH AVE	54	4TH AVE, NW/O LENORA ST	2008	6138	13600	
11	63200340	N	9992	DELRIDGE WAY SW	125	DELRIDGE WAY SW, NW/O SW CAMBRIDGE ST	2008	9992	13600	
12	300310	N	1221	1ST AVE S	87	1ST AVE S, S/O S LUCILE ST	2008	1221	17400	
13	118650110	N	15163	N 130TH ST	192	N 130TH ST, W/O ASHWORTH AVE N	2008	15163	19400	
14	26200100	N	5324	34TH AVE W	91	34TH AVE W, N/O W BARRETT ST	2008	5324	5100	
15	133750350	N	17051	NE 55TH ST	135	NE 55TH ST, E/O 35TH AVE NE	2008	17051	5700	
16	20950400	N	4471	3RD AVE NW	245	3RD AVE NW, S/O NW 80TH ST	2008	4471	8700	
17	21550520	N	4641	30TH AVE NE	244	30TH AVE NE, S/O NE 145TH ST	2008	4641	6300	

Datasets from public records

2



Traffic Flow Distributions

3

ML (DNN)

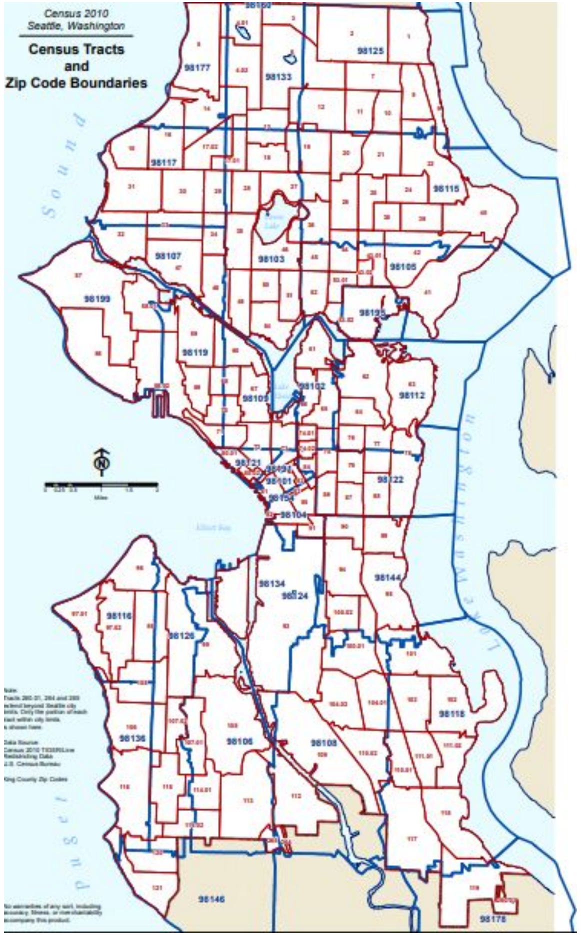
4

Final Visualization

Key Questions:

-What geographic level should we be looking at for our dataset?
-How can we deal with different reporting between different datasets?

Zip Codes as a Geographic Marker



-Looking at every individual street could obscure area-level trends, the same is true for a city-wide view

-Here several census tracts are contained within zip codes, we can sum up these data to get a representative intermediate-level pictures

-This can be accomplished through Shapely sjoin Feature, which can group the data sets into larger pre-defined features

Data Cleaning: A geographic approach

Objective: Use spatial aspects of geographical data to organize input feature data into geographic locations

Zip

Data Cleaning: A geographic approach

Objective: Use spatial aspects of geographical data to organize input feature data into geographic locations

Traffic

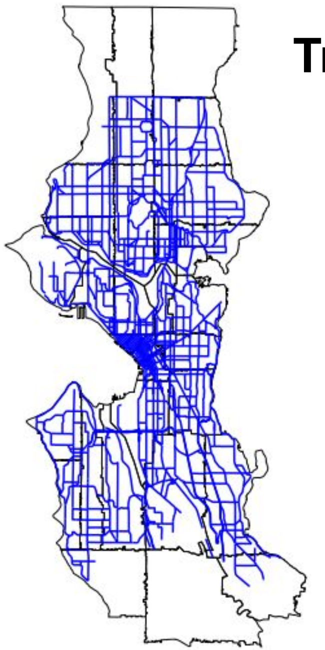
```
city_by_zip = gpd.sjoin(zip_bounds, gdf_15, op='intersects')
```

Data Cleaning: A geographic approach

Objective: Use spatial aspects of geographical data to organize input feature data into geographic

Census

\+



Traffic

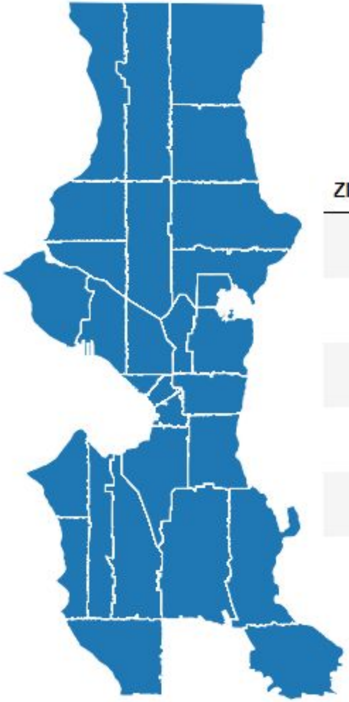
=

```
city_by_zip = gpd.sjoin(zip_bounds, gdf_15, op='intersects')
```

Combined

Data Cleaning: A geographic approach

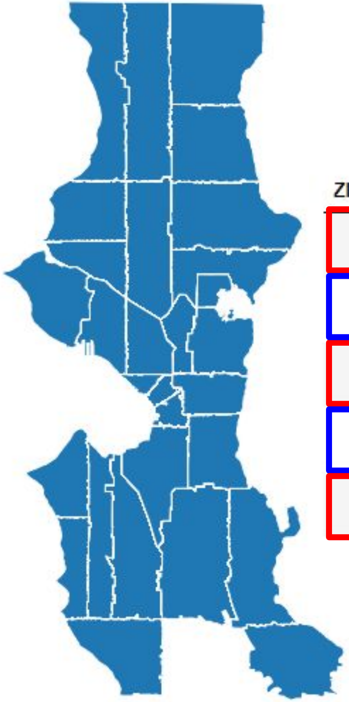
Objective: Use spatial aspects of geographical data to organize input feature data into geographic locations



ZIPCODE	geometry	NAME10	SHAPE_Area_left	index_right	YEAR	AAWDT	GEOBASID	STNAME	SHAPE_Length
98101	POLYGON ((-122.34598 47.60892, -122.34490 47.6...	74.02	1.470012e+07	1163	2015	10639.335836	800.0	ALASKAN WAY	1245.144941
98121	POLYGON ((-122.36110 47.61854, -122.36095 47.6...	81.00	1.225219e+07	1163	2015	10639.335836	800.0	ALASKAN WAY	1245.144941
98101	POLYGON ((-122.34598 47.60892, -122.34490 47.6...	74.02	1.470012e+07	1186	2015	2500.000000	823.0	VIRGINIA ST	372.729354
98121	POLYGON ((-122.36110 47.61854, -122.36095 47.6...	81.00	1.225219e+07	1186	2015	2500.000000	823.0	VIRGINIA ST	372.729354
98101	POLYGON ((-122.34598 47.60892, -122.34490 47.6...	74.02	1.470012e+07	1197	2015	11000.000000	705.0	ALASKAN WAY	613.438180

Data Cleaning: A geographic approach

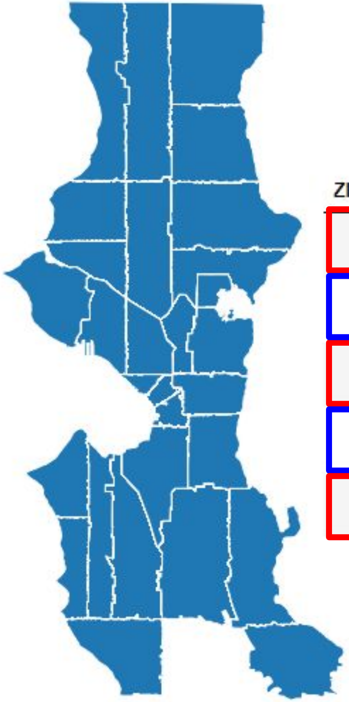
Objective: Use spatial aspects of geographical data to organize input feature data into geographic locations



ZIPCODE	geometry	NAME10	SHAPE_Area_left	index_right	YEAR	AAWDT	GEOBASID	STNAME	SHAPE_Length
98101	POLYGON ((-122.34598 47.60892, -122.34490 47.6...	74.02	1.470012e+07	1163	2015	10639.335836	800.0	ALASKAN WAY	1245.144941
98121	POLYGON ((-122.36110 47.61854, -122.36095 47.6...	81.00	1.225219e+07	1163	2015	10639.335836	800.0	ALASKAN WAY	1245.144941
98101	POLYGON ((-122.34598 47.60892, -122.34490 47.6...	74.02	1.470012e+07	1186	2015	2500.000000	823.0	VIRGINIA ST	372.729354
98121	POLYGON ((-122.36110 47.61854, -122.36095 47.6...	81.00	1.225219e+07	1186	2015	2500.000000	823.0	VIRGINIA ST	372.729354
98101	POLYGON ((-122.34598 47.60892, -122.34490 47.6...	74.02	1.470012e+07	1197	2015	11000.000000	705.0	ALASKAN WAY	613.438180

Data Cleaning: A geographic approach

Objective: Use spatial aspects of geographical data to organize input feature data into geographic locations



ZIPCODE	geometry	NAME10	SHAPE_Area_left	index_right	YEAR	AAWDT	GEOBASID	STNAME	SHAPE_Length
98101	POLYGON ((-122.34598 47.60892, -122.34490 47.6...	74.02	1.470012e+07	1163	2015	10639.335836	800.0	ALASKAN WAY	1245.144941
98121	POLYGON ((-122.36110 47.61854, -122.36095 47.6...	81.00	1.225219e+07	1163	2015	10639.335836	800.0	ALASKAN WAY	1245.144941
98101	POLYGON ((-122.34598 47.60892, -122.34490 47.6...	74.02	1.470012e+07	1186	2015	2500.000000	823.0	VIRGINIA ST	372.729354
98121	POLYGON ((-122.36110 47.61854, -122.36095 47.6...	81.00	1.225219e+07	1186	2015	2500.000000	823.0	VIRGINIA ST	372.729354
98101	POLYGON ((-122.34598 47.60892, -122.34490 47.6...	74.02	1.470012e+07	1197	2015	11000.000000	705.0	ALASKAN WAY	613.438180

```
traffic_zones = city_by_zip.dissolve(by='ZIPCODE', aggfunc = sum)
```

Data Cleaning: A geographic approach

Objective: Use spatial aspects of geographical data to organize input feature data into geographic locations

ZIPCODE	NAME10	geometry	AAWDT
98101	13323.60	POLYGON ((-122.34598 47.60892, -122.34490 47.6...	1.877668e+06
98102	2738.74	POLYGON ((-122.33574 47.64203, -122.33108 47.6...	4.782904e+05
98103	5670.00	POLYGON ((-122.35808 47.69966, -122.35741 47.6...	1.880369e+06
98104	13944.00	POLYGON ((-122.34105 47.59627, -122.34031 47.5...	1.847450e+06
98105	6560.00	MULTIPOLYGON (((-122.32859 47.66646, -122.3285...	1.762661e+06

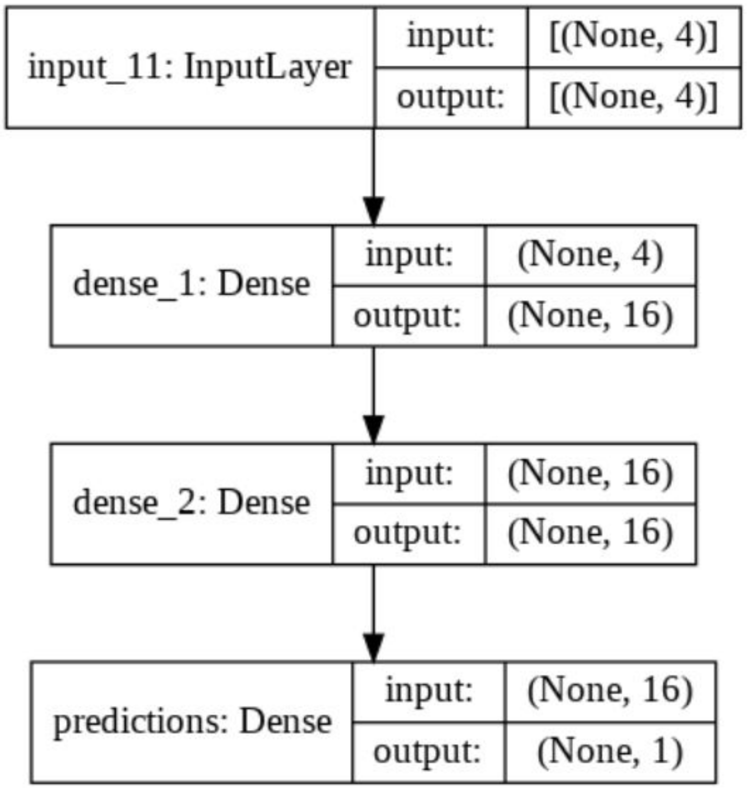
```
traffic_zones = city_by_zip.dissolve(by='ZIPCODE', aggfunc = sum)
```

Demo: Interactive plots using folium

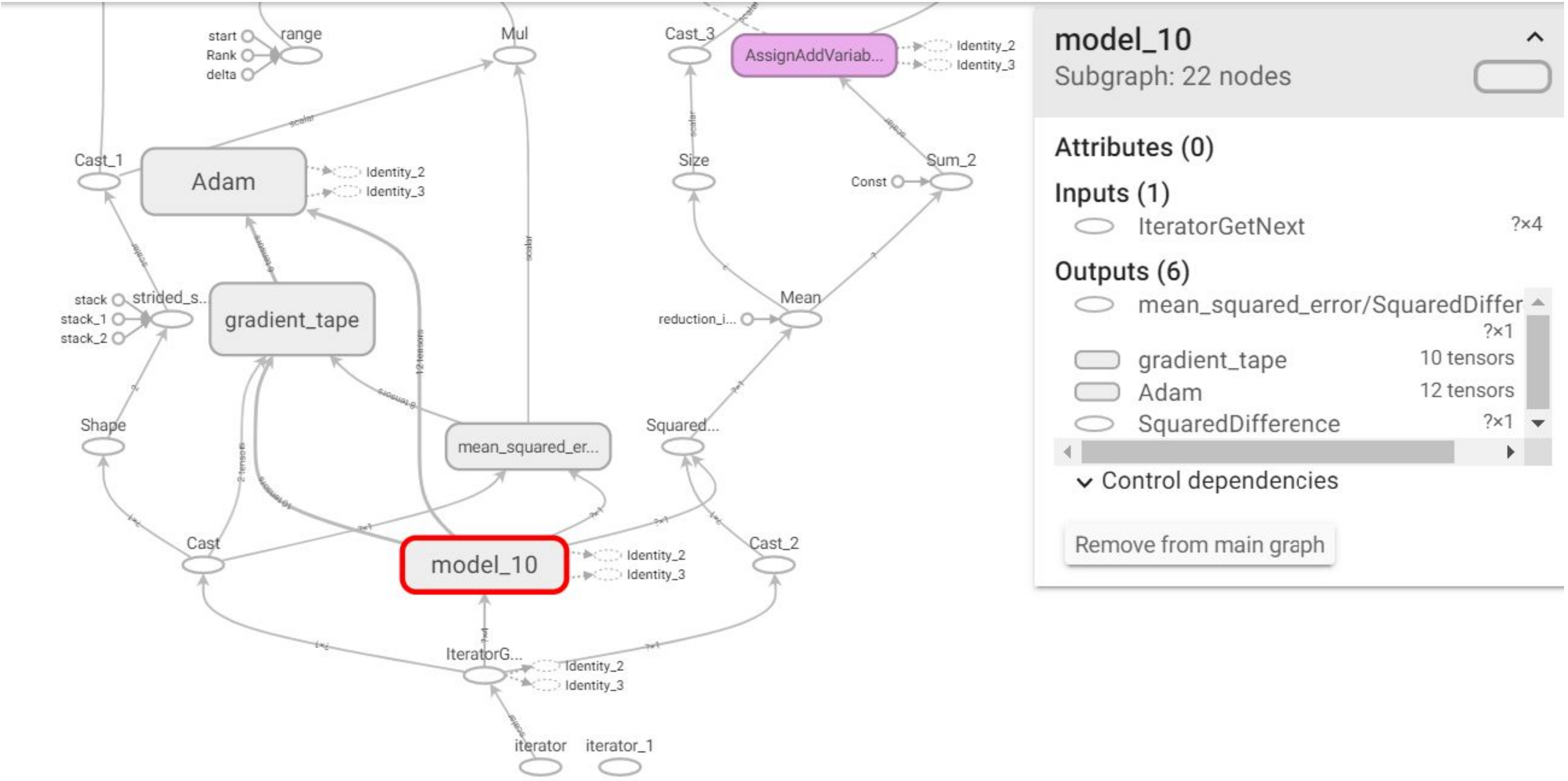
http://localhost:8888/view/map_html/choropleth_map_v3.html

Visualize the model

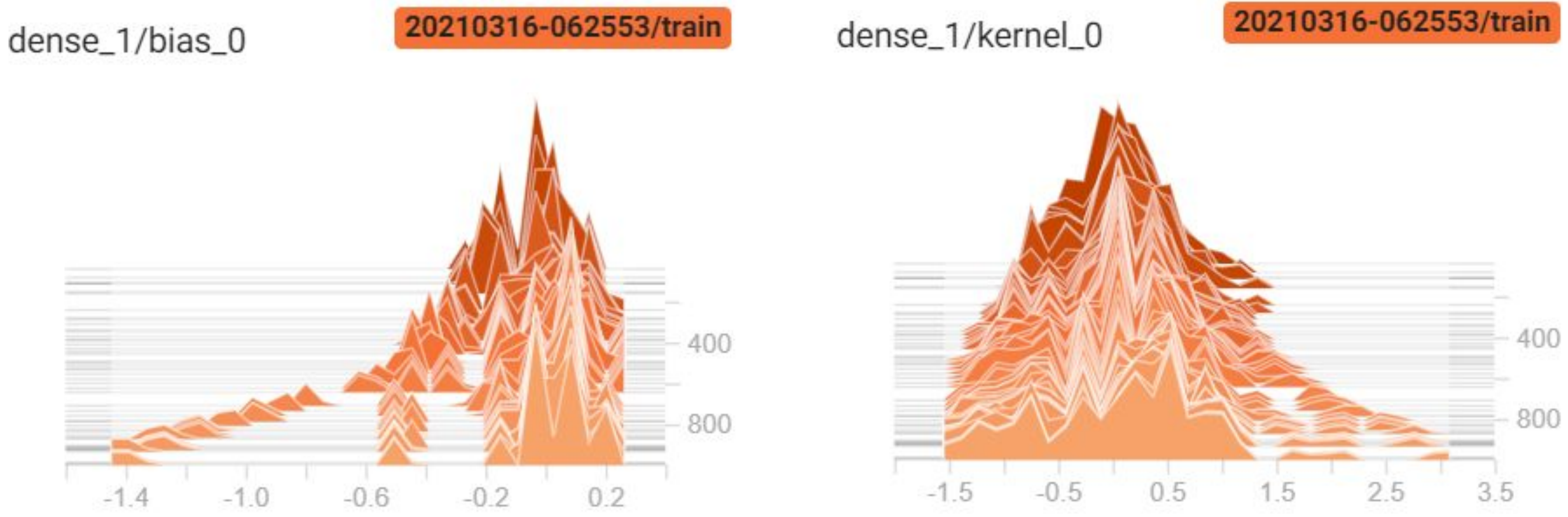
- Make use of the ‘keras.utils.plot_model’ function in Tensorflow
- The structure is of two dense layer, each has 16 neurons, with tanh activation



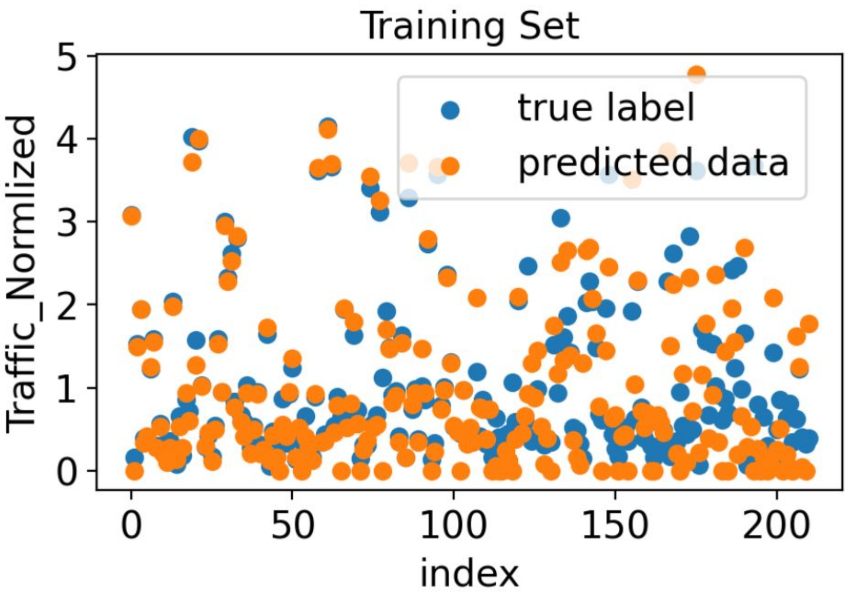
Visualize the Neural Network

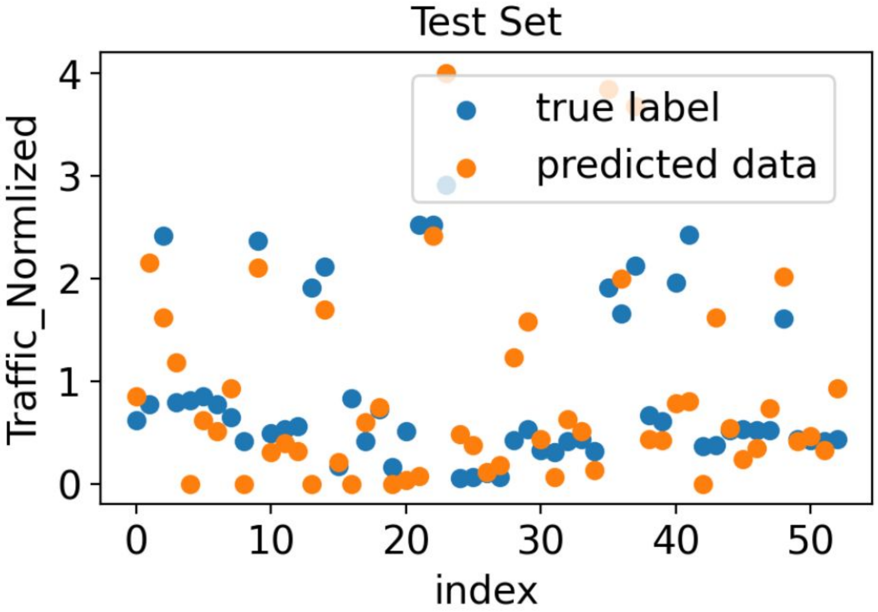


Visualize the Weight through epochs

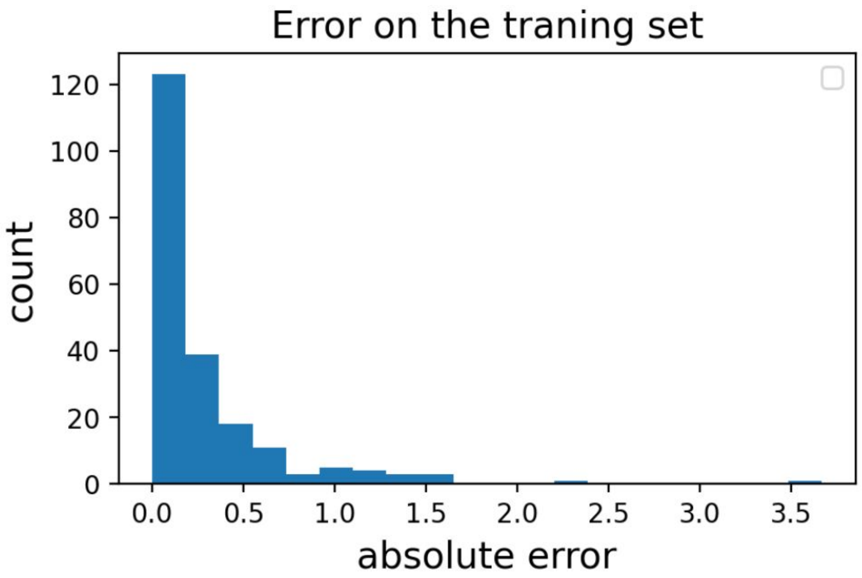


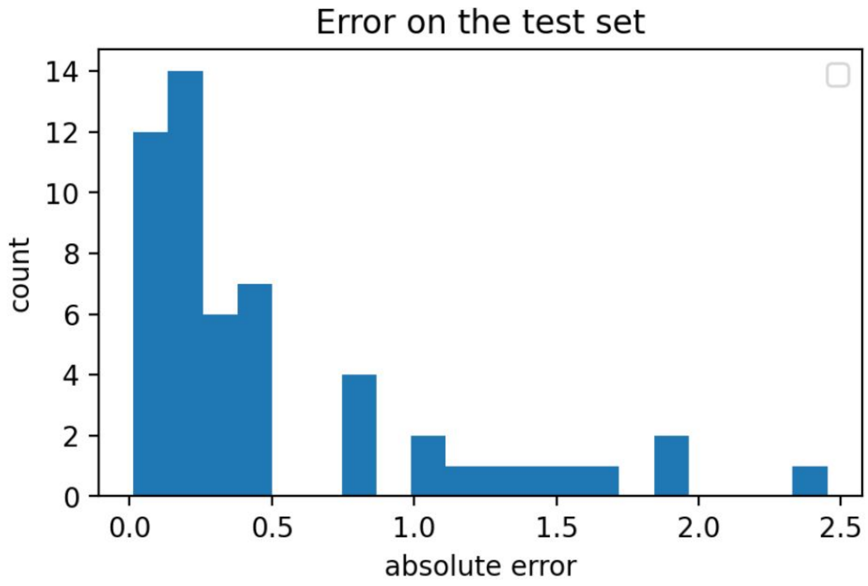
Compare the true label with the predicted label





Compare the error on the training set and the test set





Loss verses epochs