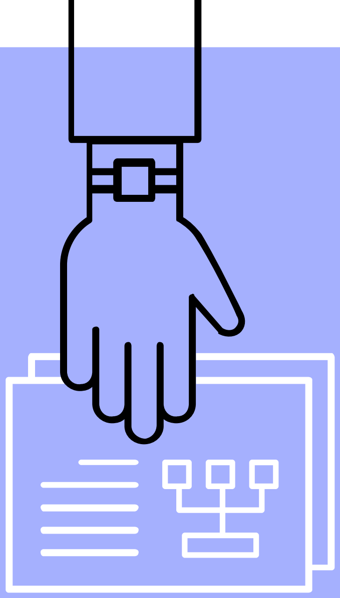
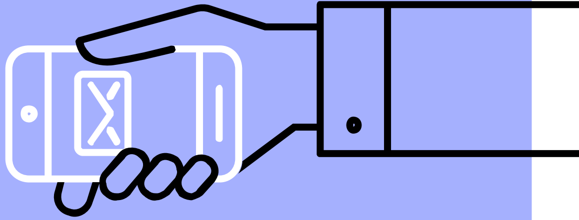
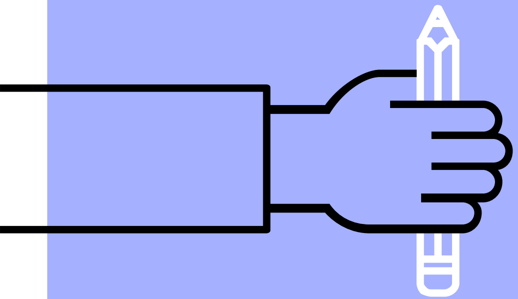
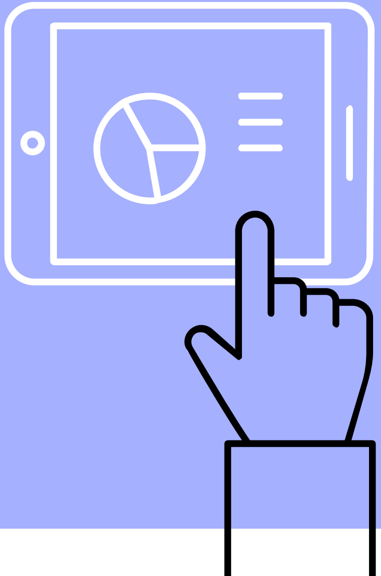




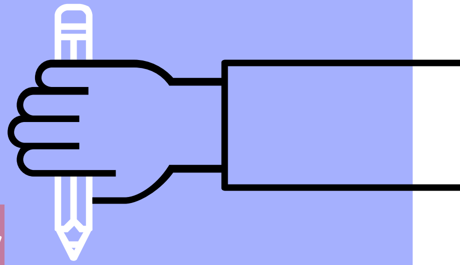
Olist Delivery Service Optimisation





INTRODUCTION

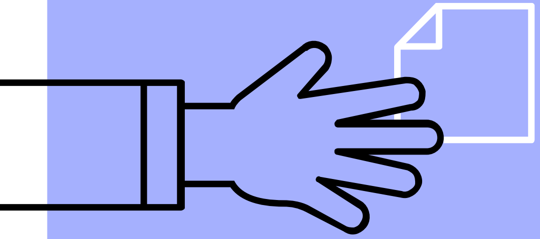
- O list data set
- online e-commerce site for sellers
- merchants + consumers —> main marketplaces
- Brazil



How long it takes for product to reach consumer?

To what extent is delivery service affected by other factors?

1. Weight
2. Freight value
3. Distance between buyer and seller
4. Volume
5. Review score
6. Price



To what extent is delivery service affected by other factors?

1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression

5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

How can OList potentially improve its delivery service?

To what extent is delivery service affected by other factors?

1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression

5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

- Filtered through to find datasets relevant to the problem
- Merged datasets to create a main data frame
  - 10 dataset → 1 dataframe
- Ensure data integrity
  - Used primary keys + composite keys
  - Eliminate duplicates → reduce redundancy
    - Eg. Order_ID
- Eliminate unnecessary fields
- Renaming columns for readability
- Checking against Null values

Cleaning Of Dataset - An overview

Merging

```
In [12]: maindf = pd.merge(orders, order_items, how='left', left_on='order_id', right_on='order_id')
         maindf = pd.merge(maindf, reviews, how='left', left_on='order_id', right_on='order_id')
         maindf = pd.merge(maindf, products, how='left', left_on='product_id', right_on='product_id')
         maindf = pd.merge(maindf, customers, how='left', left_on='customer_id', right_on='customer_id')
         maindf = pd.merge(maindf, location, how='left', left_on='customer_zip_code_prefix', right_on='geolocation_zip_code_p
         maindf = pd.merge(maindf,translation, how='left', left_on='product_category_name', right_on='product_category_name')
         maindf = pd.merge(maindf,order_payments, how='left', left_on='order_id', right_on='order_id')
         maindf.drop_duplicates(subset='order_id', inplace=True) ## for simplicity we want to have one of each order_id.
```

Dropping Duplicates

```
In [14]: maindf.drop_duplicates(subset='order_id', inplace=True) ## for simplicity we want to have one of each order_id.
```

Renaming Columns

```
In [15]:
         maindf = maindf.rename(columns = {'geolocation_lat_x': 'customer_lat', 'geolocation_lng_x': 'customer_lng', 'geoloca
```

Drop unnecessary columns

```
In [17]: maindf = maindf.drop(columns=['order_approved_at','order_item_id','review_id','review_comment_title', 'review_commen
                'review_creation_date', 'review_answer_timestamp','product_name_lenght',
                'product_description_lenght', 'product_photos_qty','order_approved_at', 'order_delivered_carrier_date','shipp
                'geolocation_zip_code_prefix_y','customer_state', 'geolocation_zip_code_prefix_x','customer_zip_code_prefix',
                'geolocation_state_x', 'product_category_name_english','payment_sequential', 'payment_type', 'payment_install
```

Drop Nan Values

```
In [18]: maindf = maindf.dropna()
         maindf.reset_index(drop=True,inplace=True)
```

[redacted]

To what extent is delivery service affected by other factors?

1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression

5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

- Calculate actual “Delivery Days”
- Date+time → Day + Month + Year + Time
- Use python’s datetime module
- Chose to limit the scope of testing
  - From 100k rows → 30k rows

order_purchase_timestamp	order_delivered_customer_date	order_estimated_delivery_date
2017-10-02 10:56:33	2017-10-10 21:25:13	2017-10-18 00:00:00
2018-07-24 20:41:37	2018-08-07 15:27:45	2018-08-13 00:00:00
2018-08-08 08:38:49	2018-08-17 18:06:29	2018-09-04 00:00:00

month	day	year2	month2	day2	year3	month3	day3	delivery_days
10	02	2017	10	10	2017	10	18	8.0
07	24	2018	08	07	2018	08	13	14.0
08	08	2018	08	17	2018	09	04	9.0

To what extent is delivery service affected by other factors?

1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression
5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

- After cleaning - We have calculated actual delivery days taken, estimated delivery days taken, and volume.

review_score	product_weight_g	customer_lat	customer_lng	Types of products	payment_value	seller_lat	seller_lng	volume	year	delivery_days	estimated_days
4	500.0	-23.574809	-46.587471	Household	18.12	-23.680114	-46.452454	1976.0	2017	8.0	16.0
4	400.0	-12.169860	-44.988369	Fashion	141.46	-19.810119	-43.984727	4693.0	2018	14.0	20.0
5	420.0	-16.746337	-48.514624	Electronics	179.12	-21.362358	-48.232976	9576.0	2018	9.0	27.0
5	450.0	-5.767733	-35.275467	Household	72.20	-19.840168	-43.923299	6000.0	2017	14.0	27.0
5	250.0	-23.675037	-46.524784	Household	28.62	-23.551707	-46.260979	11475.0	2018	3.0	13.0
...	...	...	...	...	...	...	...	...	...	...	...
5	1400.0	-15.832476	-48.010334	Household	118.63	-22.931256	-43.178813	22500.0	2018	7.0	13.0
5	300.0	-30.024860	-51.223432	Tools	53.60	-22.852758	-47.055102	1188.0	2017	10.0	27.0
5	1850.0	-19.612724	-46.924422	Tools	308.24	-20.802436	-49.395624	32560.0	2017	16.0	31.0
4	450.0	-22.912294	-43.382198	Household	132.25	-27.209811	-49.632920	8000.0	2017	13.0	21.0
5	400.0	-22.915062	-43.552655	Fashion	207.94	-15.847734	-48.113206	2100.0	2018	6.0	39.0

To what extent is delivery service affected by other factors?

1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression
5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

Basic Bivariate Analysis of our factors.

Factors	Correlation Against Delivery Days
Freight Value	0.22
Volume	0.08
Weight	0.09
Review Score	-0.34
Price	0.06

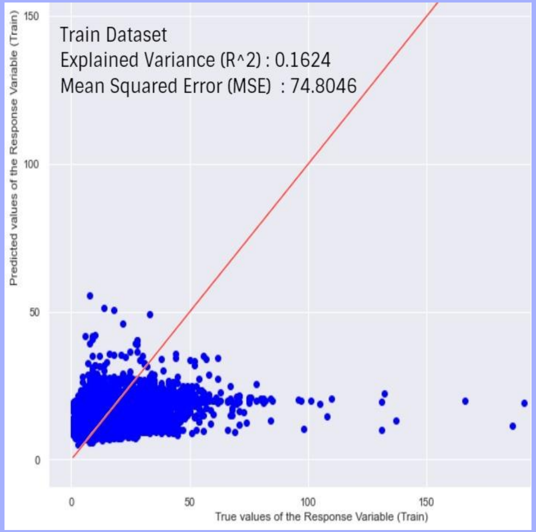
To what extent is delivery service affected by other factors?

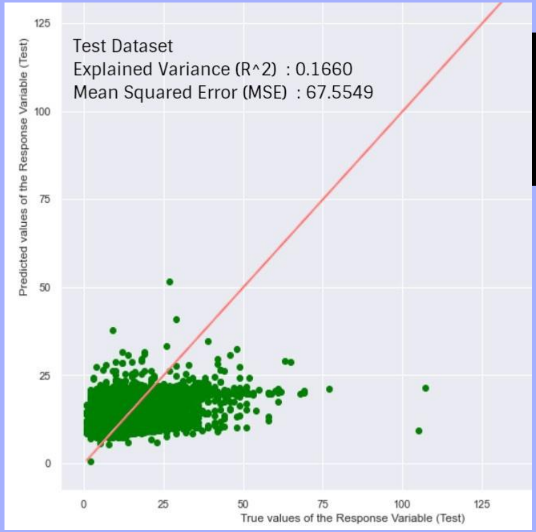
1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression
5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

- Only 1 variable is inversely related to delivery days - review score.
- Most of our correlation values are low. The exceptions are freight value (0.22) and review score (—0.34) .

1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression
5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

Correlation matrix of delivery days and other variables

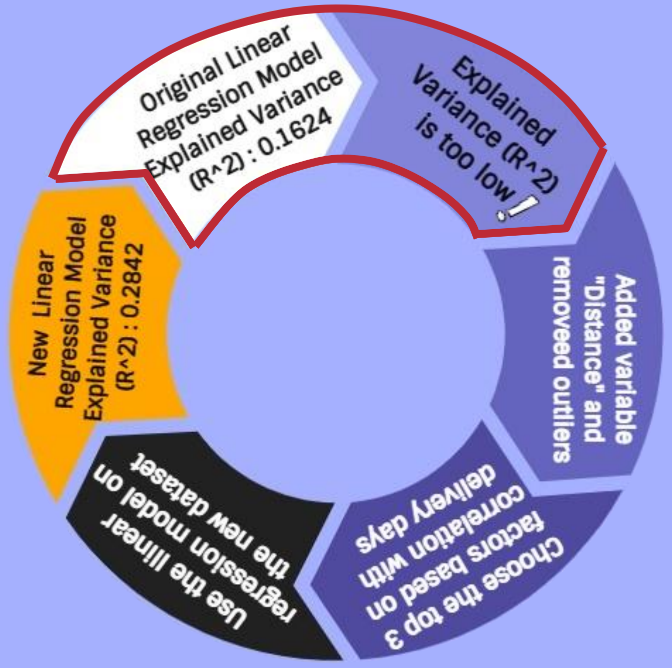




Linear Regression using all variables

Machine Learning

- Linear Regression
- We used linear regression to predict delivery days taken so we can identify any areas which we can use to optimize the delivery process.

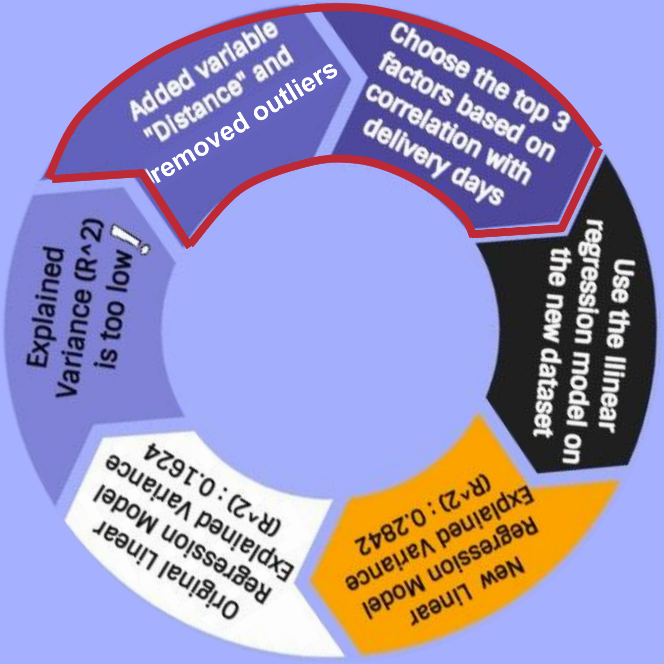


1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression

5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

Machine Learning

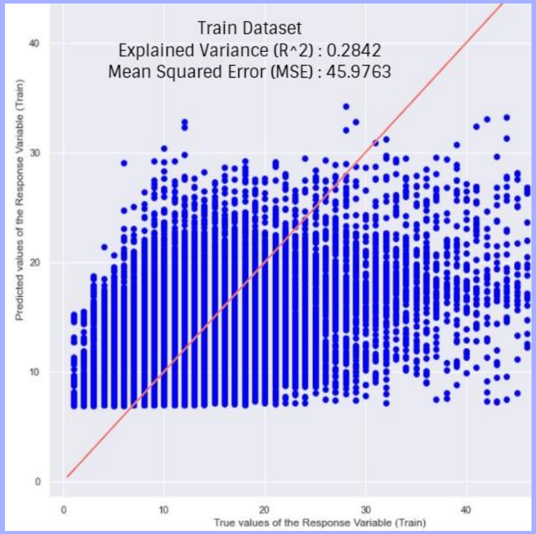
- Linear Regression
- We used linear regression to predict delivery days taken so we can identify any areas which we can use to optimize the delivery process.

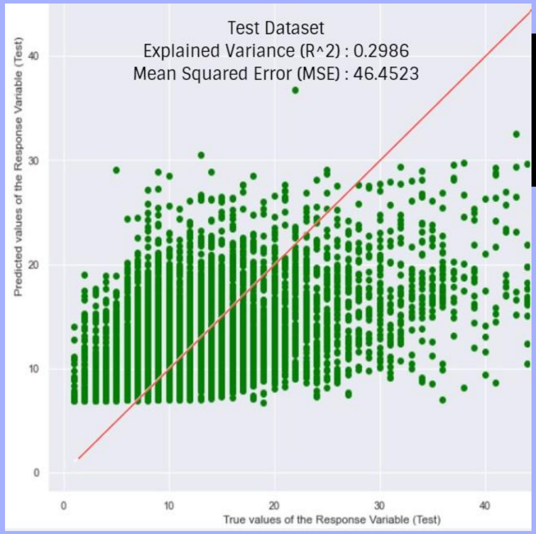


1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression

5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

Choosing the top 3 factors based on correlation matrix





Linear Regression using top 3 variables

1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression

5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

Machine Learning

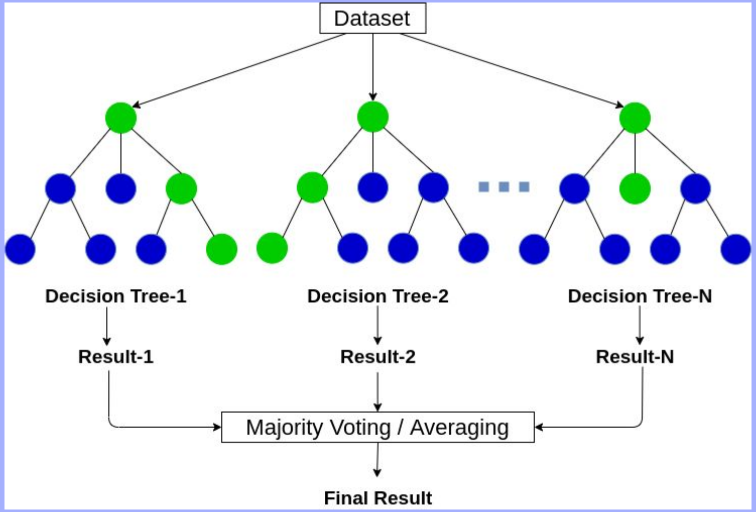
- Linear Regression
- We used linear regression to predict delivery days taken so we can identify any areas which we can use to optimize the delivery process.

1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression

5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

Random Forest Regression

- a supervised learning algorithm that uses ensemble learning method for regression
- Ensemble learning method: a technique that combines predictions from multiple machine learning algorithms to make a more accurate prediction than a single model



To what extent is delivery service affected by other factors?

1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression

5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

- Prediction
  - Our model predicted the delivery dates better than estimated days given by O-list. This can be seen from the difference in estimated days and actual delivery days.
  - This shows than O-list can do much more in giving their customers a better gauge in estimated delivery days taken. If our model with a low explained variance can do better, there is surely more that a company can do to provide better and more precise data to its customers.

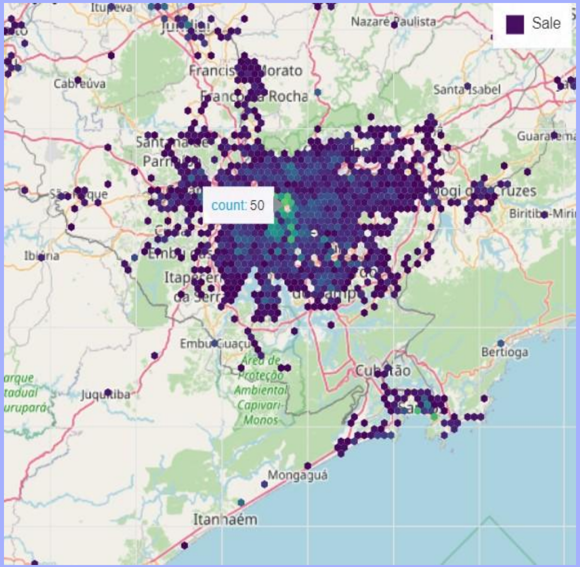
```
In [30]: sum = 0
         sum2 = 0
         for i in range(0,28948):
             given_estimate_difference = abs(jointDF.loc[i,'estimated_days'] - jointDF.loc[i,'delivery_days'])
             ML_estimate_difference = abs(jointDF.loc[i,'estimated_ML'] - jointDF.loc[i,'delivery_days'])
             sum = sum + given_estimate_difference
             sum2 = sum2 + ML_estimate_difference

         print("Total sum of difference between delivery days and estimated delivery days = ",sum)
         print("Total sum of difference between delivery days and estimated delivery from ML =",sum2)

         Total sum of difference between delivery days and estimated delivery days =  377937.0
         Total sum of difference between delivery days and estimated delivery from ML = 145410.25822320714
```

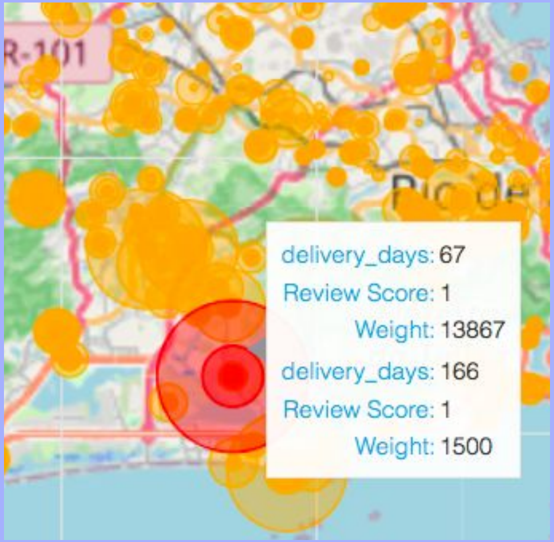
1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression

5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

Univariate Geospatial analysis

- GeoViews, geopandas, bokeh
- Similar to Heatmap onto map
- Dynamic bubble radius - space as a visual cue to encode data



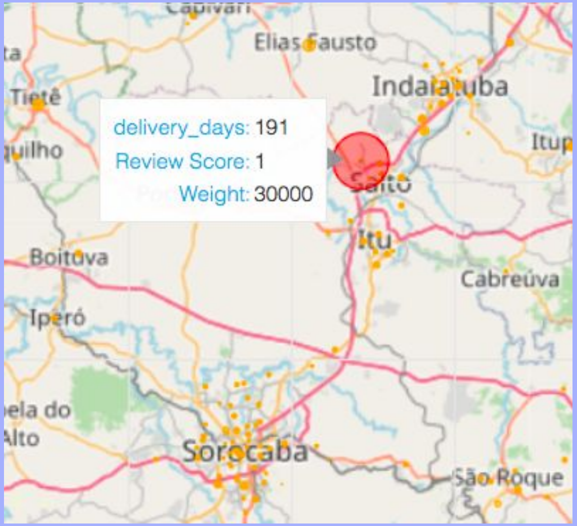


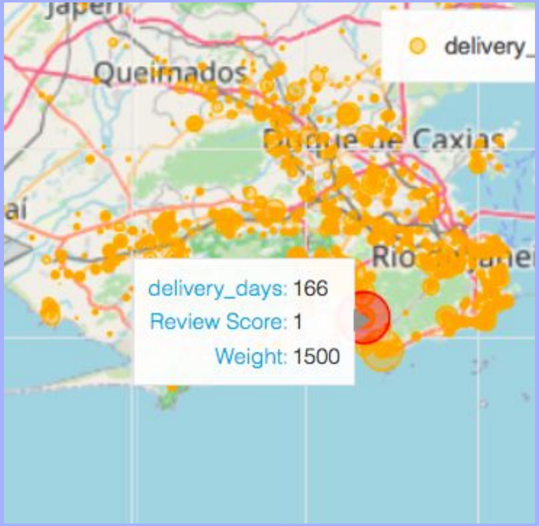
1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression

5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

Bivariate Geospatial analysis

- Weight
- Review rating





1. Clean data set
2. Explore variables
3. Find correlation
4. Plot linear regression
5. Machine Learning
6. Choose top 3 factors
7. Something new
8. Prediction
9. Recommendations + suggestion

Recommendations

1. Improve accuracy of estimated delivery → more transparency → more customer satisfaction

2. Increase its distribution network → setting up more centers outside of the city

3. More warehouses in locations such as Saito, West of Floresta da Tijura, Saito, Salvador

Contribution

Ananya

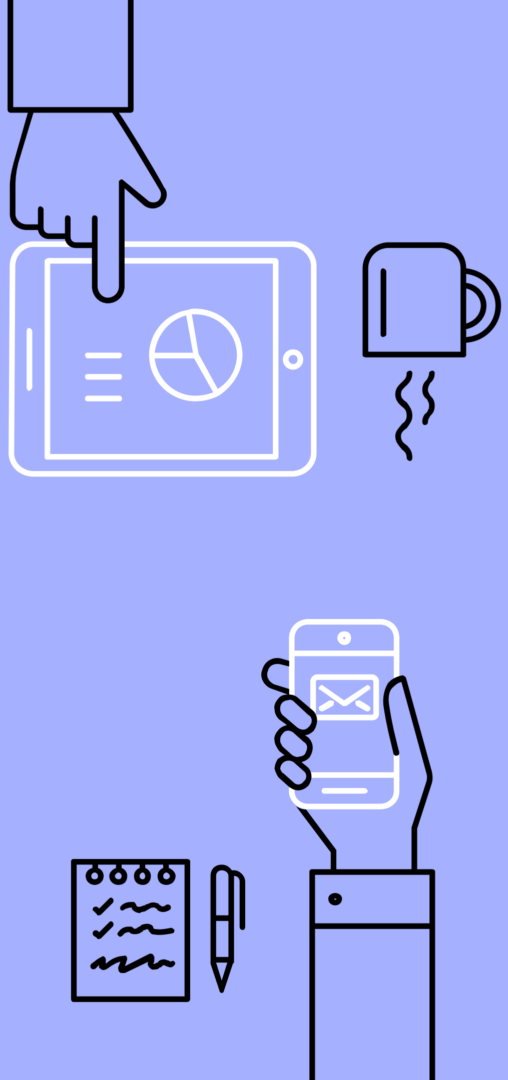
- Geospatial Analysis
- Geospatial Visualization
- Slides and Video
- Recommendations

Charlene

- Basic Linear Regression Model
- Improving using Random Forest Regression
- Recommendations

Tai Ann

- Data cleaning
- Exploratory data analysis
- Improving Linear Regression Model
- Comparison of ML and given estimates



Bibliography

https://pro.arcgis.com/en/pro-app/latest/tool-reference/spatial-statistics/generalized-linear-regression.htm

https://pysal.org/spreg/notebooks/Panel_FE_example.html

https://stackabuse.com/random-forest-algorithm-with-python-and-scikit-learn/

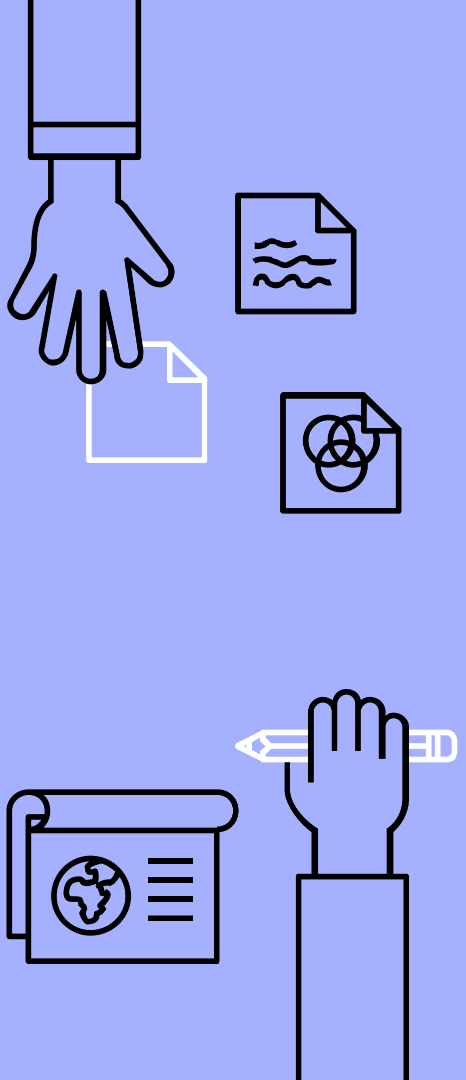
https://gis.stackexchange.com/questions/239436/spatial-weight-for-pysal-from-a-geojson-file-or-geodataframe

http://darribas.org/gds_scipy16/ipynb_md/02_geovisualization.html

https://pro.arcgis.com/en/pro-app/latest/arcpy/main/arcgis-pro-arcpy-reference.htm

https://towardsdatascience.com/calculating-distance-between-two-geolocations-in-python-26ad3afe287b

https://levelup.gitconnected.com/random-forest-regression-209c0f354c84



Thank you!