A Glimpse of the NYC Citibike Data

Xiaoxiao Wang

June, 2014

The NYC Citibike - what is it?

- A public bike sharing system
  - An additional transportation option for new yorkers and visitors
  - Launched since May 2013, available 24 hours a day, all year around
  - Thousands of bikes, hundreds of stations at **Manhattan and Brooklyn**
  - 3 traveling options: Subscriber - Annual Membership (45 min per ride)

    Customer - 24-hour Pass (30 min per ride)

    Customer - 7-day Pass (30 min per ride)

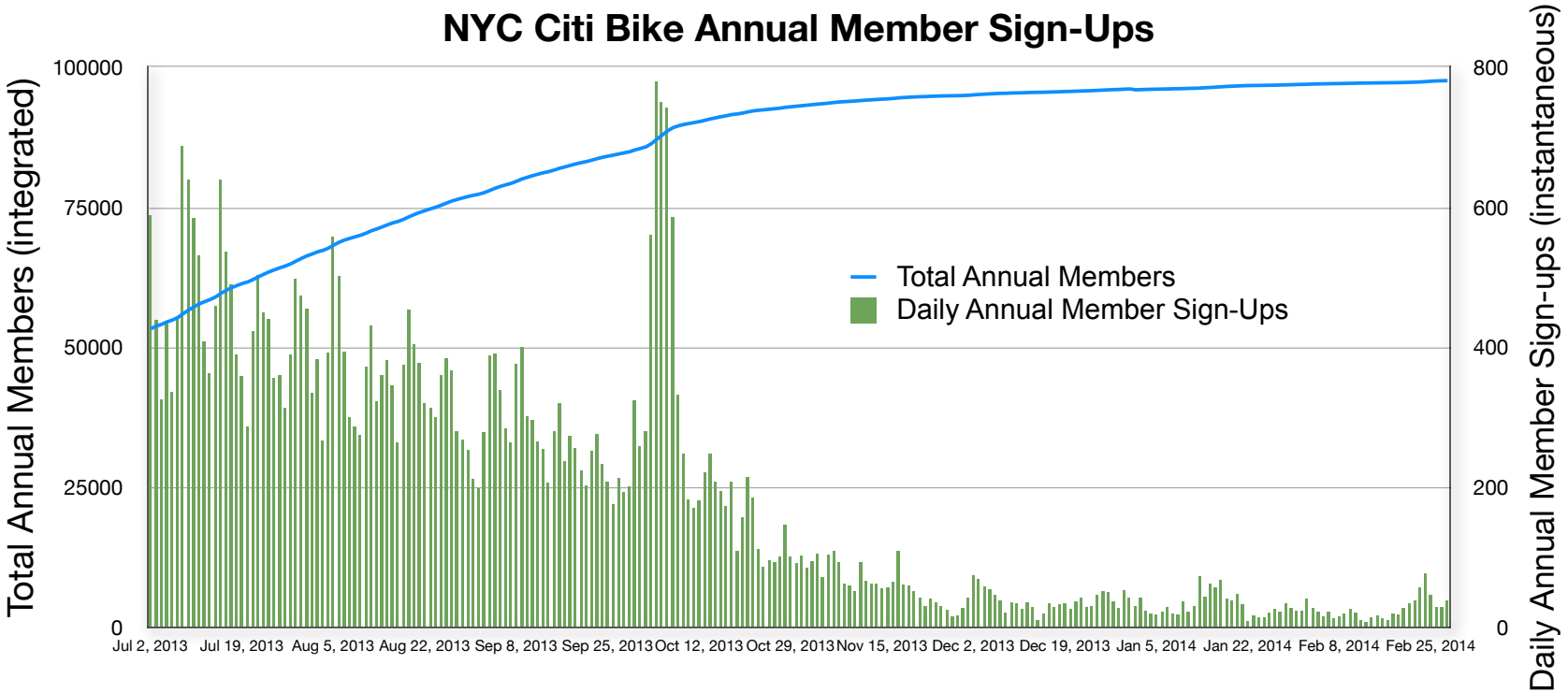
- A fantastic example of public data sharing
  - Trip histories in .CSV format since last July: https://www.citibikenyc.com/system-data
  - Aggregated/daily ridership & membership data since launch are also available
  - Encourage interested public to use the data for analysis, development, and fun!

Data and Analysis Tools

- Data I used
  - **Citibike Data**
    - Timeline: July, 2013 to February, 2014: https://www.citibikenyc.com/system-data (.csv)
    - Statistics: **5,600,000** trips in total.
    - Data Attributes: User Type (Subscriber/Customer), User Year of Birth, User Gender, Start/End Station, Start/End Time, Trip Duration, Station Lat/Long, Bike ID, etc.
  - **Weather data** (same time period): from an weather API website http://www.wunderground.com/
  - **New York City geographical data by zip code** (shapefile): from http://www.columbia.edu/acis/eds/gis/images/nyc_zipcta.zip, original source at *U.S. Department of Commerce, U.S. Census Bureau, Geography Division (2008)*

- Tools and Methods
  - **Code Documentation at GitHub: https://github.com/XiaoxiaoWang87/CitiBike**
  - **General: use Python and ROOT** (a standard statistical data analysis framework in experimental particle physics)
  - **Data cleaning and preparation:** Python csv.DictReader, save data into a data structure called .root / **n-tuple file** (analogy: "table" - records saved in rows, attributes saved in columns; "root file" - records saved as tree entries, attributes saved as branches/variables)
  - **Data processing: Shell scripting**, run parallel jobs on batch system **@hep.hpc.yale.edu**
  - **Data analysis & visualization: ROOT, Pandas** (Data frame and series), **Numpy, IPython, Basemap, PySAL, Fiona, Shapely, GDAL** (ogr2ogr) for geographical file conversion, **Matplotlib, Numbers** (mac) for time series

Citibike Membership



* Some Estimates Used:

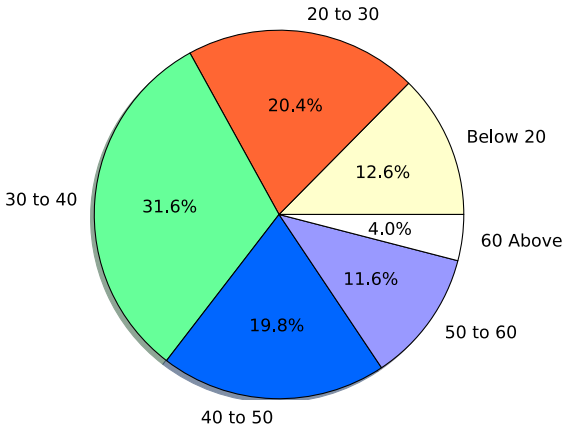
- A 150 lb person cycling at 14mph burns 50 calories per mile

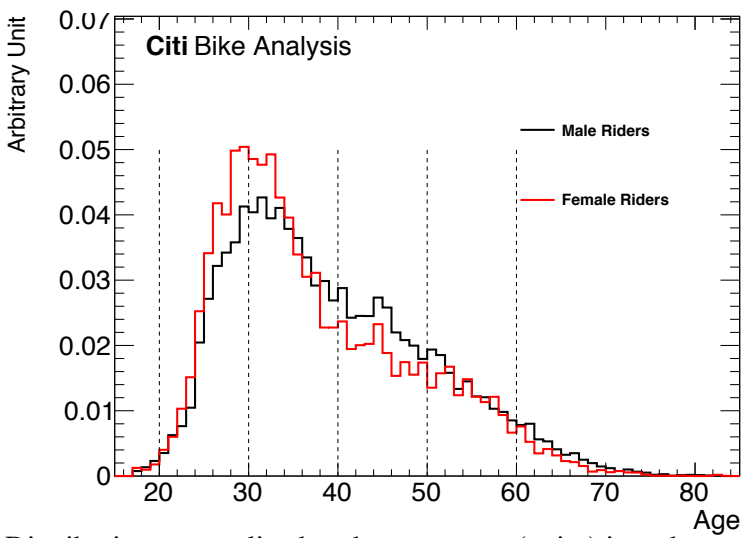
- Cars and light-weight trucks: $4.2*10^{-4}$ ton CO_2 per mile; 21.4 miles per gallon gasoline

- Generally it's been quite successful
  - Over 97,000 annual member sign-ups by the end of Feb, 2014
  - 12,023,391 miles traveled, corresponding to $6*10^8$ calories burned, $6*10^5$ gallons less of gasoline consumption, and **$5*10^3$ tons less of CO_2 emission**
  - The daily annual member sign-ups (green histogram above) periodically peaks at every Monday; there was also a large increase in sign-ups last October; in winter, people are more reluctant to start an annual membership. It would be beneficial for marketing directors to identify the causes of the surge in membership.

Diversity

User Age Composition

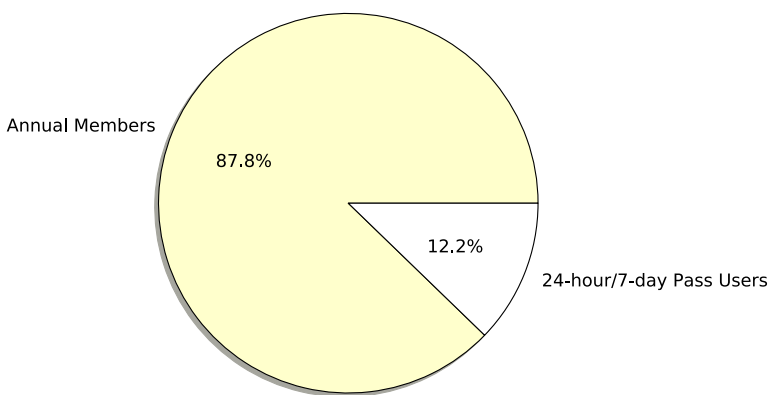




* Distributions normalized to the same area (unity) in order to compare shapes

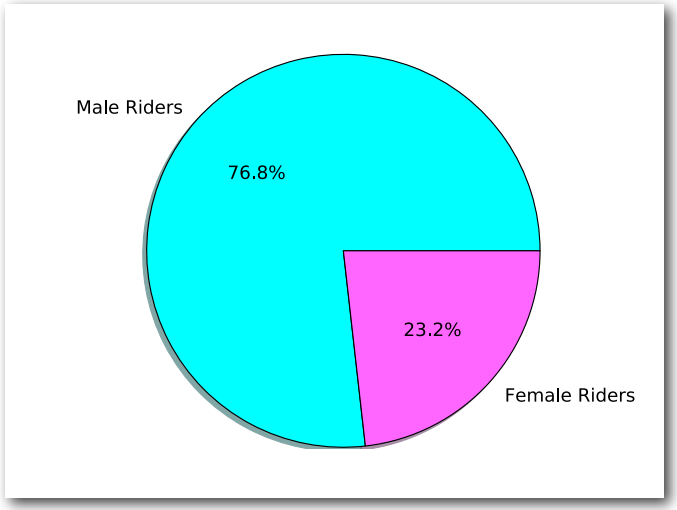
- Citibike riders exhibit a wide range of ages, with 30-to-40-year-old being the most enthusiastic user group (~30%)
- On average, female riders are slightly younger than male riders
- With the ratio 7:1, annual members make up the majority of Citibike users

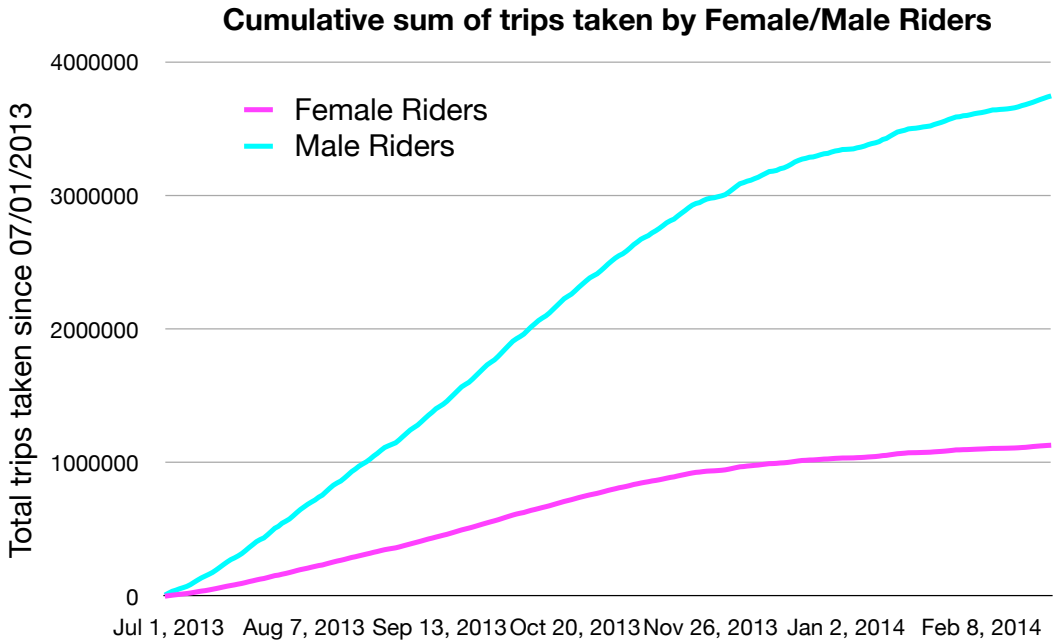
User Type



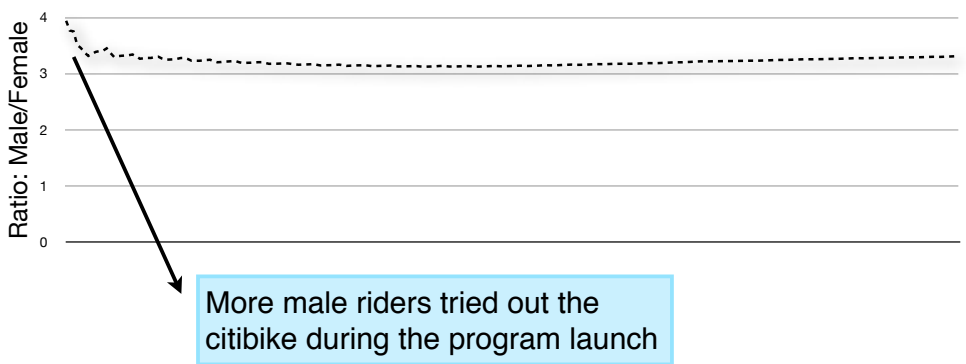
Diversity...?

User Gender Composition



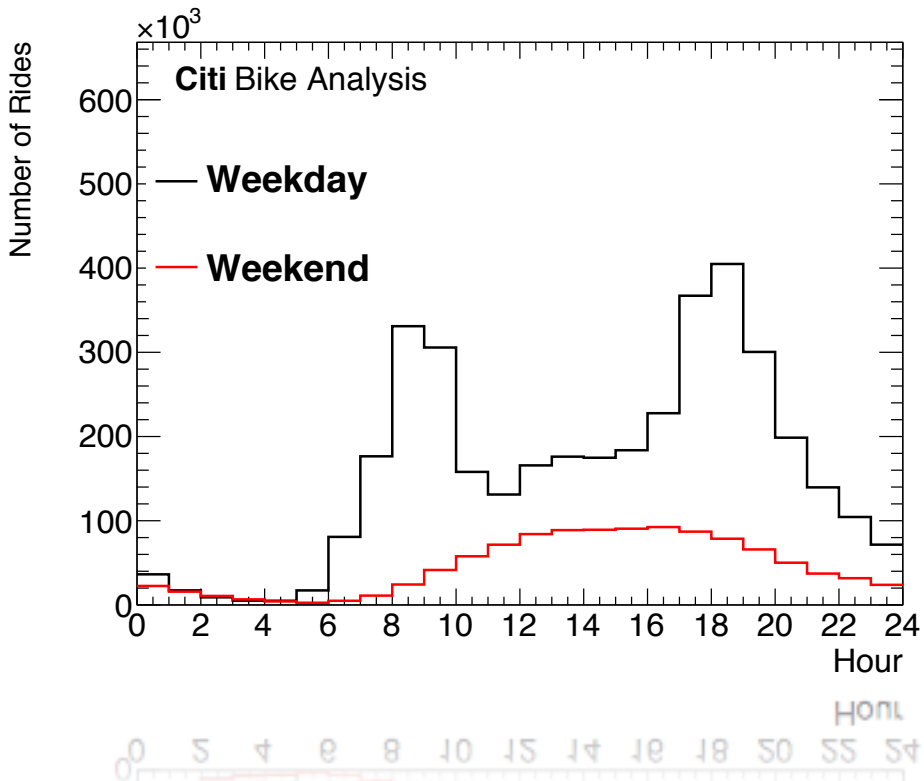


- There is a large gender gap in the usage of citibikes
- Out of all trips, over 70% are taken by male riders
- With a ratio of 3:1, there are consistently more male riders than female riders



Citibike - Commuting to Work

Number of Rides Taken By the Annual Subscribers Within a Day



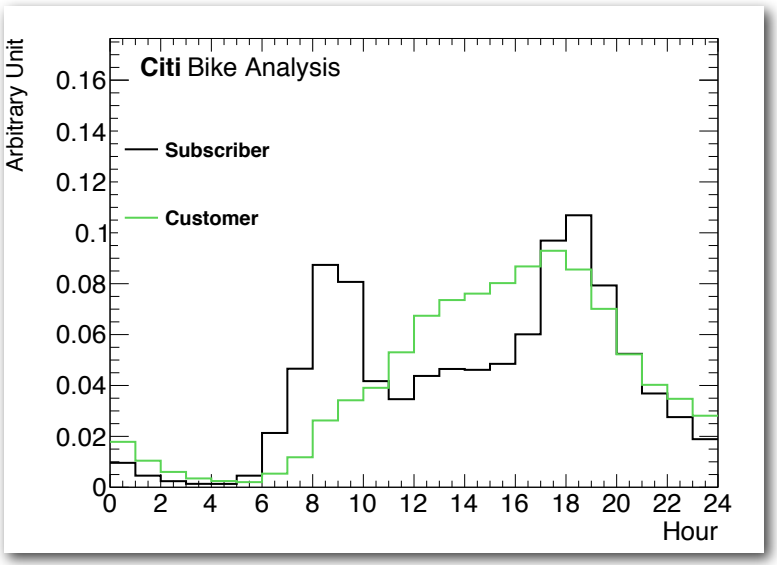
* Hour corresponds to the end time of each trip

- The trip peaks around 9 am and 6 pm, which mirrors new yorkers' average time of starting work and arriving home, respectively
- On weekends, a smoother distribution which peaks from 1 pm to 5 pm is observed. People may use bikes to work out in a park, go to a coffee shop, or go shopping.

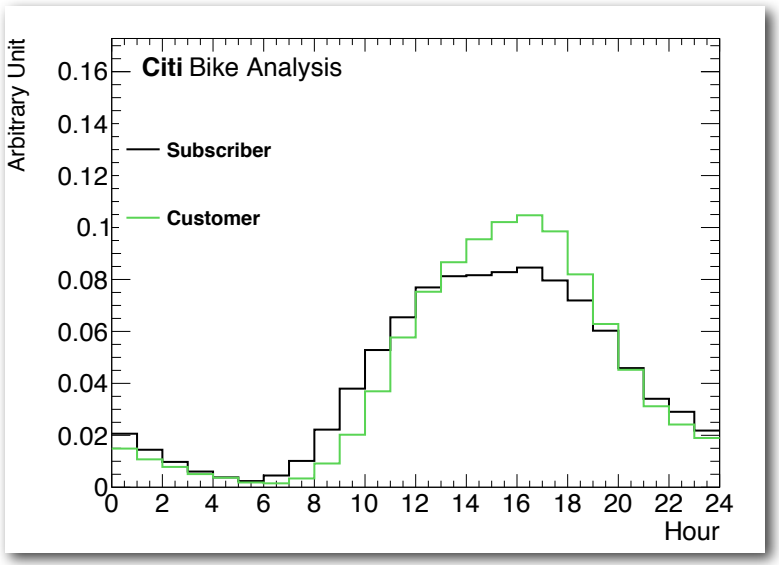
Citibike - Commuting to Work

Annual Subscribers and 24-hour/7-day Customers Traveling Habit Comparisons

Weekdays



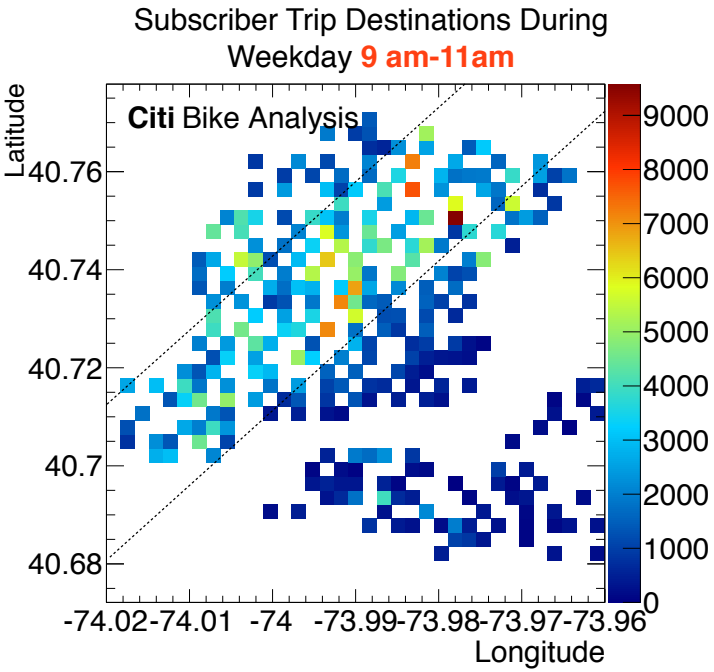
Weekends

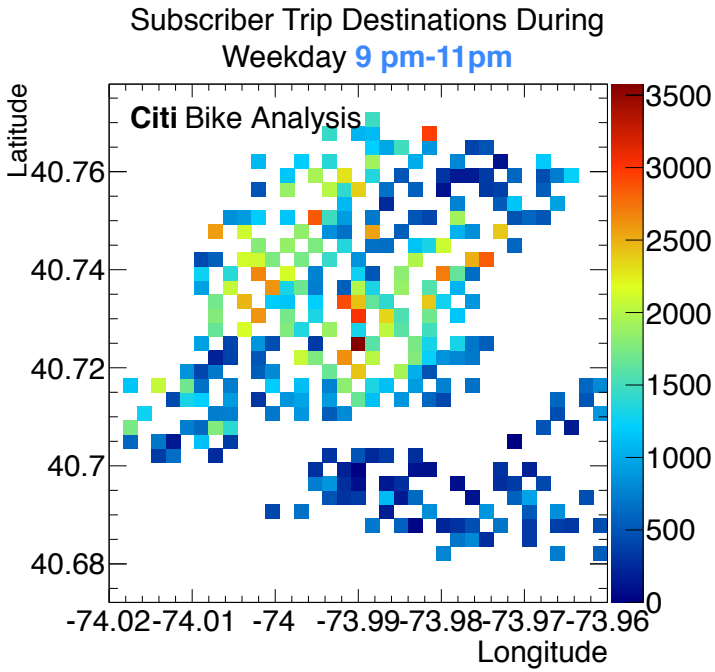


*Both plots: Distributions normalized to the same area (unity) in order to compare shapes

- There is an obvious difference between subscribers and customers on weekdays, as the majority of customers are expected to be visitors who use bikes for recreational purposes.
- On weekends, the traveling patterns of subscribers and customers are similar (with customers' average traveling time being just slightly later than subscribers).

Citibike - Traveling Patterns

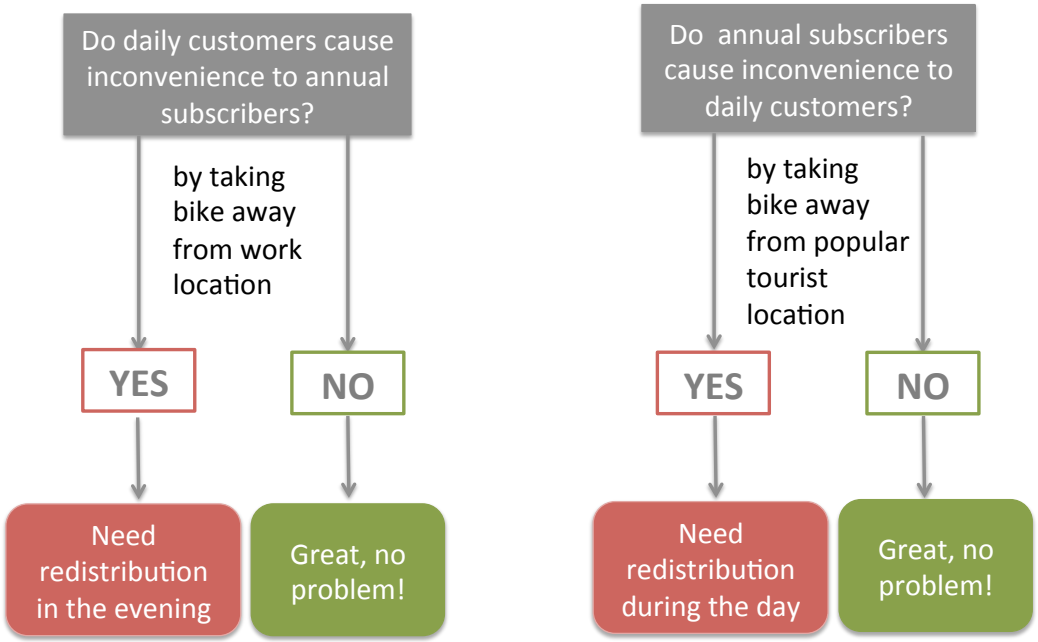




- Without using any geographical data, by simply plotting the number of trips on a longitude-latitude 2D plane, the outlines of Manhattan and Brooklyn emerge (above plots)
- In the morning, most trips end in central (between east & west) manhattan (denoted with dashed line on the left plot)
- In the evening, residential neighborhoods and places with a vibrant nightlife such as east/west villages attract more travelers (right plot)

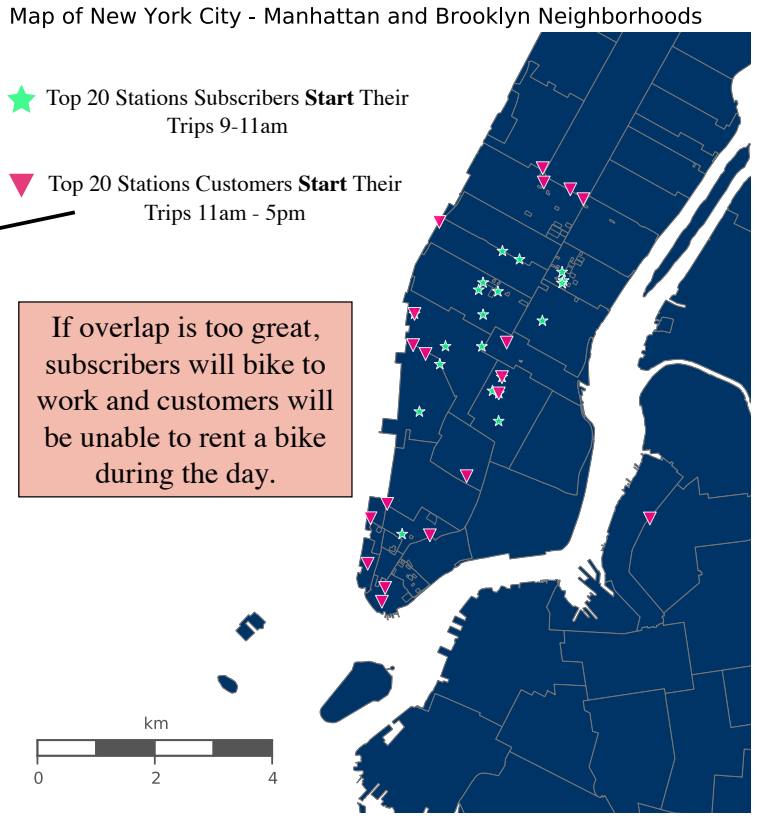
Bike Shortage?

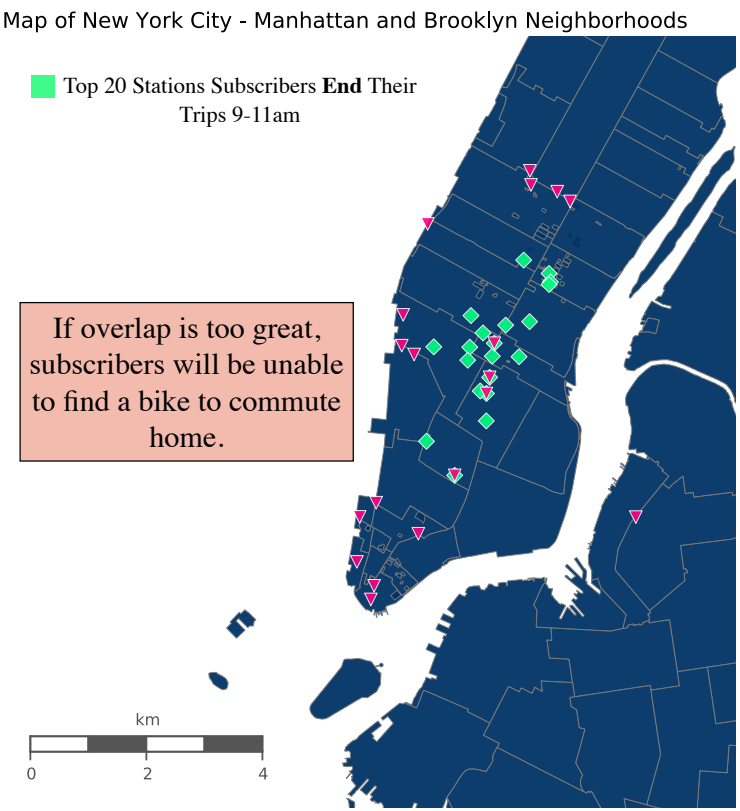
- A common concern about citibike is during rush hours some places run out of bike while other bike stations are full (see the previous slide). **But is it really a problem?**
- The majority of users are annual subscribers using bikes for work. For them the redistribution of bikes by Citibike crews may make things worse, as they need to bike home in the evening as well.
- **Two questions to address:**



A close look of the subscriber/customer location overlap:

Weekend data used for customer. Explained next slide.





- Both the table on the previous slide and the maps here show that the traveling patterns for subscribers and customers are quite different, with little overlap.
- Subscribers need bikes from Mid-Manhattan, while customers prefer to start their trips from places such as Central Park, High Line, and Hudson River Park, etc..

Bike Shortage? (cont.)

Location

trips/ hr

★ Top 5 Stations *Subscribers Start* Their Trips 9-11am (weekDAYS)	■ Top 5 Stations *Subscribers End* Their Trips 9-11am (weekDAYS)	▼ Top 5 Stations *Customers Start* Their Trips 11am-5pm (weekEND)
8 Ave & W 31 St, **22.7**	E 17 St & Broadway, **19.6**	Central Park S & 6 Ave, **10.6**
Pershing Square N, **19.2**	W 21 St & 6 Ave, **18.4**	Centre St & Chambers St, **8.4**
E 43 St & Vanderbilt Ave, **18.9**	E 43 St & Vanderbilt Ave, **16.8**	Grand Army Plaza & Central Park S, **7.8**
Lafayette St & E 8 St, **17.7**	W 27 St & 7 Ave, **16.6**	West St & Chambers St, **7.5**
Pershing Square S, **17.4**	Lafayette St & E 8 St, **16.2**	Broadway & W 58 St, **5.6**

Why did I use weekend data for customer?

- Because the weekday data of customers suffers from the biases caused by subscribers' commuting activities.

What important assumption did I make?

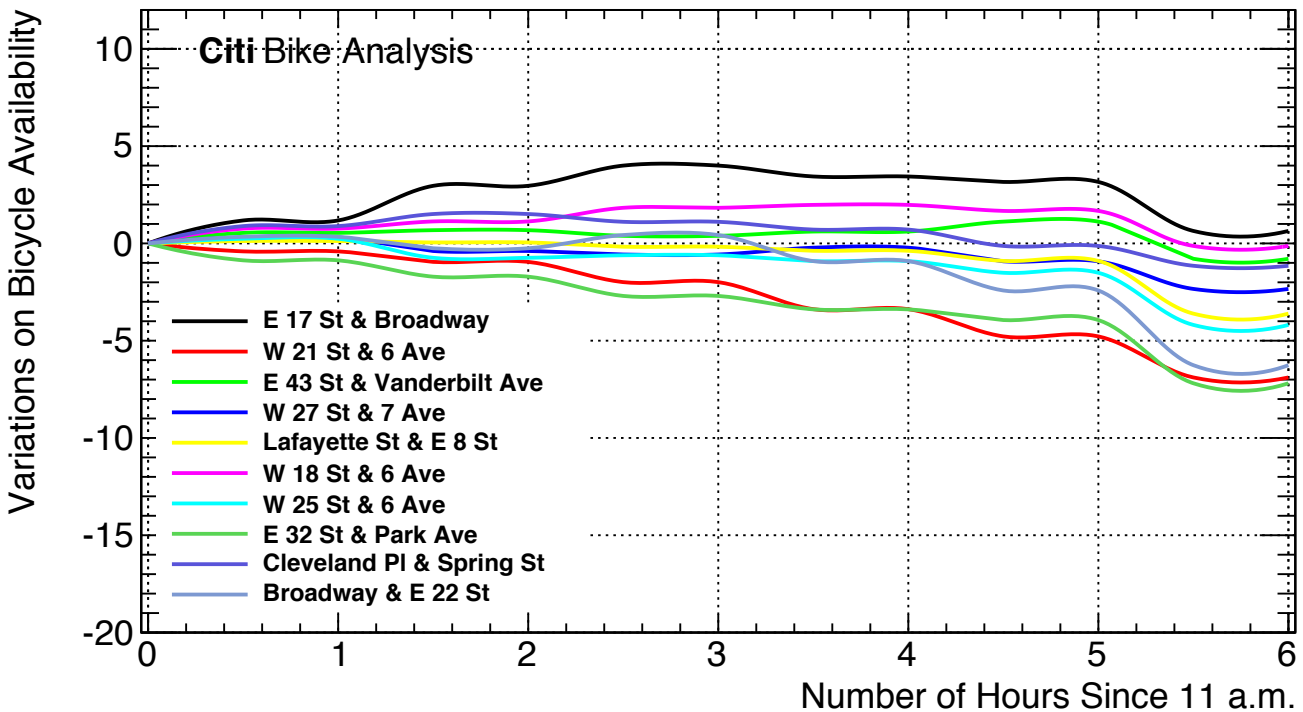
- The weekdays/weekends traveling habits of customers are similar, i.e. popular bike sites for travelers stay the same throughout the week.

What conclusion can I draw?

The location overlap (subscriber start vs. customer start, customer start vs. subscriber end) is small. There is no immediate need for redistribution of bikes.

Another way to study bike shortage:

- Here I picked the top 10 stations where subscribers leave their bikes in the morning (9-11 am), and observe the fluctuations of their bike availability during the day (11 am - 5 pm)



*Bike availability:

0: number of bikes at 11 am

+: number of inbound bikes

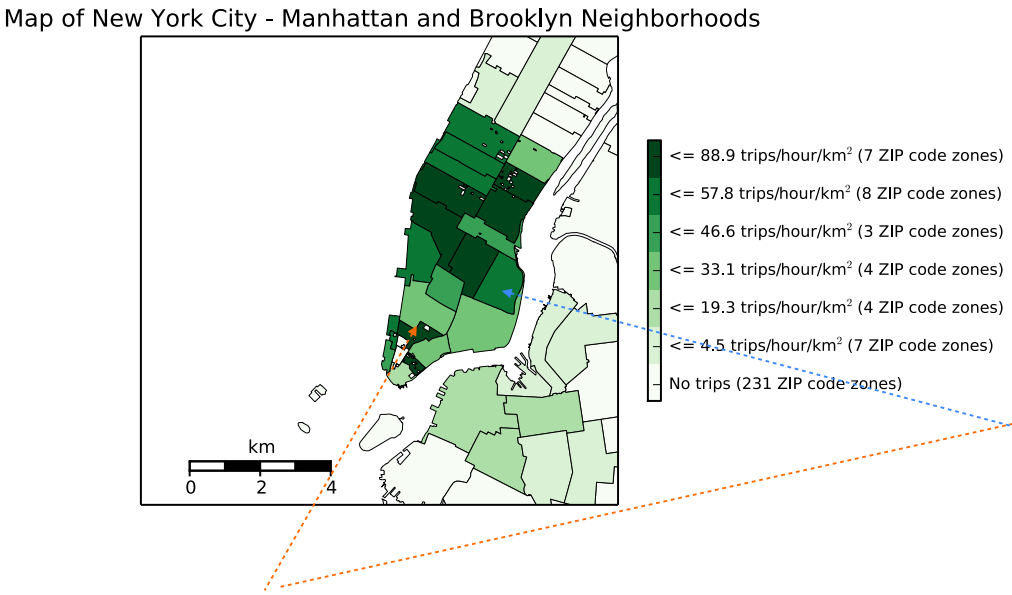
—: number of outbound bikes

- The plot shows until 4 pm, for most stations the bike availability is good. After 4 pm the availability begins to fall, but this is likely caused by subscribers themselves commuting home, and therefore not a problem

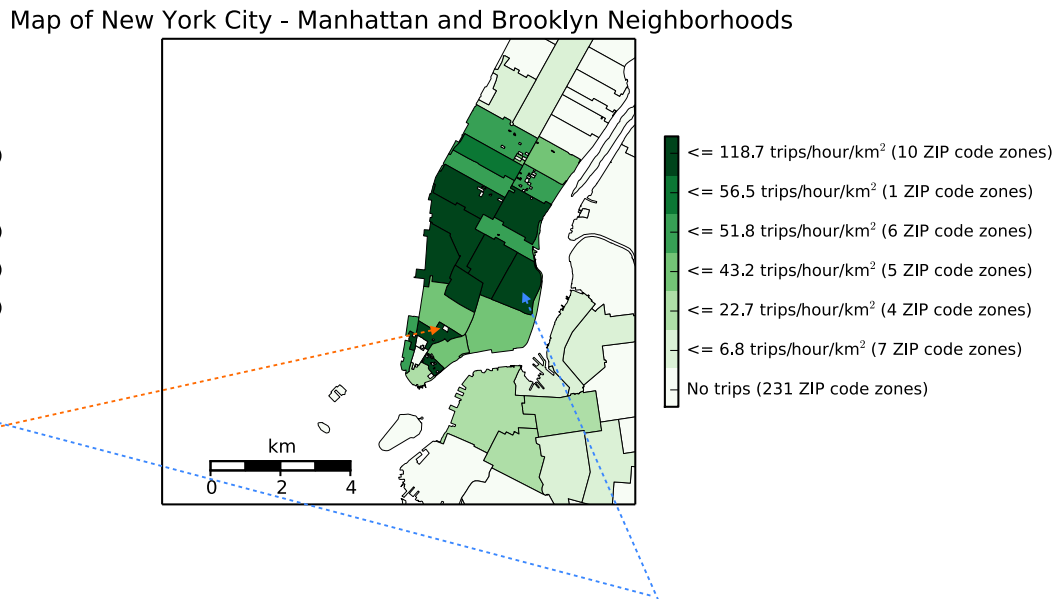
- “W 21St & 6 Ave” and “E 32 St & Park Ave” have the tendency to run out of bikes quickly - may need redistribution of bikes

Traffic Pressure

Annual Subscriber Starting Trip Densities by Zip Code Area
During Weekdays **6am-12pm**



Annual Subscriber Ending Trip Densities by Zip Code Area
During Weekdays **6pm-12am**

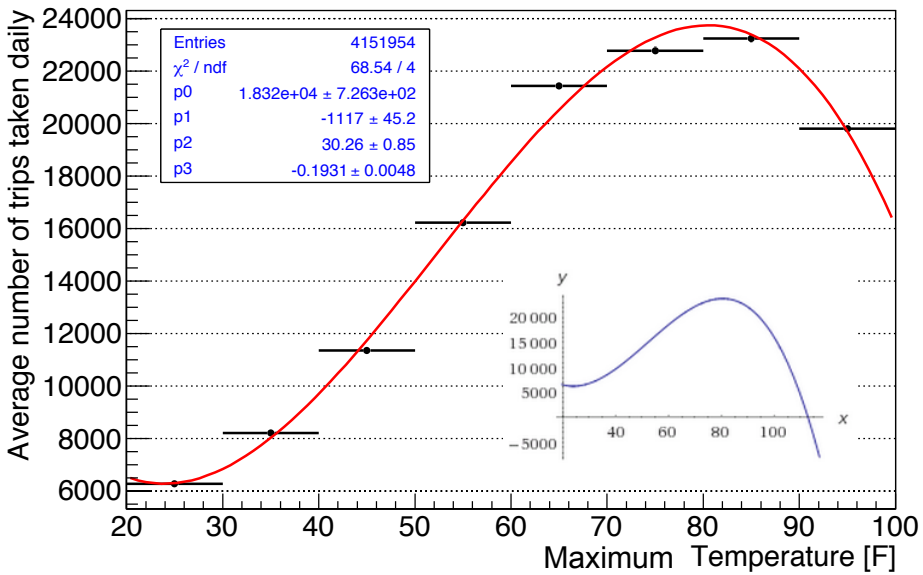


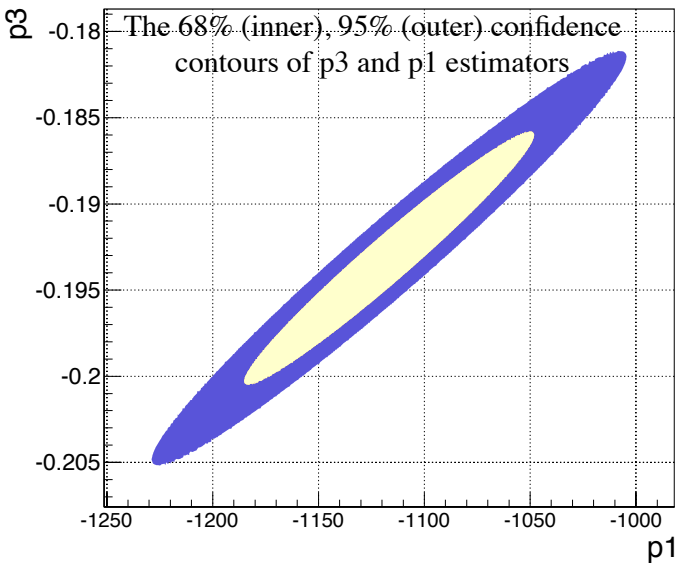
- **SOHO surrounding areas** always suffer from high citibike traffic pressure (> 60 trips/hour/km²). **East Village** has larger traffic pressure in the evening.
- Compared to Manhattan, Brooklyn has comparatively less citibike traffic density (~20 trips/hour/km²)

Impact of Temperature

- Below I plotted the average number of daily trips as a function of maximum temperature. Then I fit it using a simple 3-order polynomial ($y = p0 + p1*x + p2*x^2 + p3*x^3$) with chi^2 method : $\sum_{i=1}^{N}\left[\frac{y_i - f(x_i;\theta)}{\sigma_i}\right]^2$

*Note that to avoid biases from uneven number of days in a specific temperature range, I plot here the trips per day, by re-weighting histograms





- The fitted chi^2/ndf ~ 10, suggesting a reasonably good fit (with ~1 being an ideal fit). Also, a 95% confidence interval of the estimator will fully cover all data points. Data in each bin is modeled with Poisson errors (shown on the plot).
- The solution of $0 = p0 + p1*x + p2*x^2 + p3*x^3$ with best fitted estimators is at **x = 113 F (45 C)**, predicting that in such a high temperature no one will use bikes.
- As temperature increases, more people start using citibikes. The peak of bike usage is at max temperature ~ 80 F, after which the bike usage begins to drop (perhaps because of the intolerable temperature).

Summary

- I analyzed 8 months of citibike data, more than $5.6*10^6$ entries of trip information.
- It was an extremely fun experience! Particle physics analysis methods + real-world problem.
- I used Python based analysis code and performed analysis mostly with PyROOT and Pandas.
- Parallel computing on my university batch systems made my data processing extremely efficient.

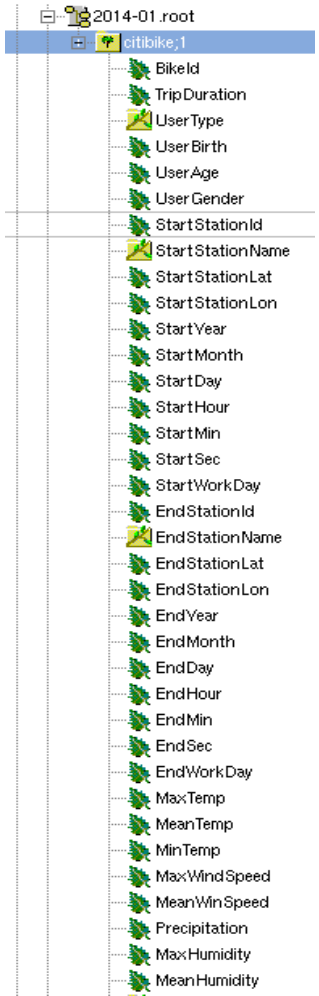
- **Highlights of Results**
  - A large gender gap in Citibike users exists. (We could use it as a starting point to identify potential causes and make Citi-biking more appealing to female users).
  - New Yorkers use Citibike for work - a good sign!
  - No immediate need for redistributing bikes, despite some places are short of bikes during weekdays
  - During the past week New York City had an average maximum temperature around 80 F, which indicates it is the peak season for Citibike!

Backup

References

- ***"So you'd like to make a map using Python"***, Sensitive Cities, October 2013, http://sensitivecities.com/so-youd-like-to-make-a-map-using-python-EN.html#.U5zp2o1dWD7
- ***"Posts Tagged Coordinate Systems"***, Gothos, August 2012, http://gothos.info/tag/coordinate-systems/
- ***"New York City Numeric and Spatial Data",*** Columbia University Libraries, http://library.columbia.edu/locations/dssc/data/nyc.html
- ***"New York City Neighborhood Tabulation Areas"***, The Official Website of the City of New York, http://www.nyc.gov/html/dcp/pdf/bytes/nynta_metadata.pdf
- ***"Interactive: A Month of Citi Bike",*** The New Yorker, MICHAEL GUERRIERO**,** July 2013, http://www.newyorker.com/online/blogs/newsdesk/2013/07/month-of-citi-bike.html
- ***"Snow, Ice and Wind No Issue for Citi Bike's Die-Hards",*** The New York Times, MATT FLEGENHEIMER, Jan 2014, http://www.nytimes.com/2014/01/29/nyregion/snow-ice-and-wind-no-issue-for-citi-bikes-die-hards.html
- ***"The Citibike Blog",*** https://www.citibikenyc.com/blog
- ***"Weather API: Introduction",*** http://www.wunderground.com/weather/api/d/docs?MR=1
- ***"GDAL/OGR Tracker and Wiki"*****,** http://trac.osgeo.org/gdal/wiki
- ***"NYC Open Data",*** https://nycopendata.socrata.com/

Data Structure



- PyROOT
  - A Python extension module that allows the user to interact with any ROOT classes from the Python interpreter
  - ROOT splits each event (in the Citibike case is “trip”) into its pieces (the variables, or the "columns" of the table representation) and builds a file by putting together all columns (left plot)
  - Access variables by looping over all entries of Citibike trips and call the variable name such as “UserAge”
  - For large dataset reading and writing much faster than tables
  - Easy to apply selection criteria such as “StartHour>=9 && StartHour<=11”
  - Easy to combine different files together (e.g. I ran one batch job for every month’s data and obtained 8 root files. In the end I combined the 8 root files to get the complete dataset information)

Geo Data

- Geographical data projection
  - **The biggest complication I faced when making a heat map**
  - The New York City Zip Code shapefiles are easy to get, e.g. here: http://www.columbia.edu/acis/eds/gis/images/nyc_zipcta.zip
  - However, all available shapefiles online are already projected into X and Y coordinates (the NAD 83 coordinate system is frequently used in North America, the US Census Bureau in particular)
  - But the Python module Basemap only takes geographical coordinates which are in latitude and longitude.
  - So I had to first find a way to convert the projected shapefile back into geographical coordinates
  - Here I found an easy way to do it (first need to install GDAL):
    - ogr2ogr -t_srs EPSG:4326 new.shp original.shp
    - “EPSG” is the code to identify the coordinate system we want to project into. Here 4326 means WGS84, World Geodetic System of 1984, which is a geographical coordinate system using latitude and longitude