Darya Filipchyk

09/27/2020

NY Taxi

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:
```
data_filename = 'nyc_data.csv'
data = pd.read_csv(data_filename,
parse_dates=['pickup_datetime', 'dropoff_datetime'])
# see documentation string
```

In [3]:
```
data_filename = 'nyc_data.csv'
data = pd.read_csv(data_filename,
parse_dates=['pickup_datetime', 'dropoff_datetime'])
pd.read_csv?
```

In [8]:
```
data.head()
```

Out[8]:

	medallion	hack_license	vendor_id
0	76942C3205E17D7E7FE5A9F709D16434	25BA06A87905667AA1FE5990E33F0E2E	VTS
1	517C6B330DBB3F055D007B07512628B3	2C19FBEE1A6E05612EFE4C958C14BC7F	VTS
2	ED15611F168E41B33619C83D900FE266	754AEBD7C80DA17BA1D81D89FB6F4D1D	CMT
3	B33E704CC189E80C9671230C16527BBC	6789C77E1CBDC850C450D72204702976	VTS
4	BD5CC6A22D05EB2D5C8235526A2A4276	5E8F2C93B5220A922699FEBAFC2F7A54	VTS

In [ ]:
```

```

In [7]:
```
data.describe()
```

Out[7]:

	rate_code	passenger_count	trip_time_in_secs	trip_distance	pickup_longitude	pic
count	846945.000000	846945.000000	8.469450e+05	8.469450e+05	846945.000000	84
mean	1.026123	1.710272	8.125239e+02	9.958211e+00	-73.975155	
std	0.223480	1.375266	1.609831e+04	6.525205e+03	0.035142	
min	0.000000	0.000000	-1.000000e+01	0.000000e+00	-74.098305	
25%	1.000000	1.000000	3.610000e+02	1.050000e+00	-73.992371	
50%	1.000000	1.000000	6.000000e+02	1.800000e+00	-73.982094	
75%	1.000000	2.000000	9.600000e+02	3.200000e+00	-73.968048	
max	6.000000	6.000000	4.294796e+06	6.005123e+06	-73.028473	

In [9]:
```
p_lng = data.pickup_longitude
p_lat = data['pickup_latitude']
```

In [19]:
```
# returns the first 5 rows
p_lng.head()
```

Out[19]:
```
0   -73.955925
1   -74.005501
2   -73.969955
3   -73.991432
4   -73.966225
Name: pickup_longitude, dtype: float64
```

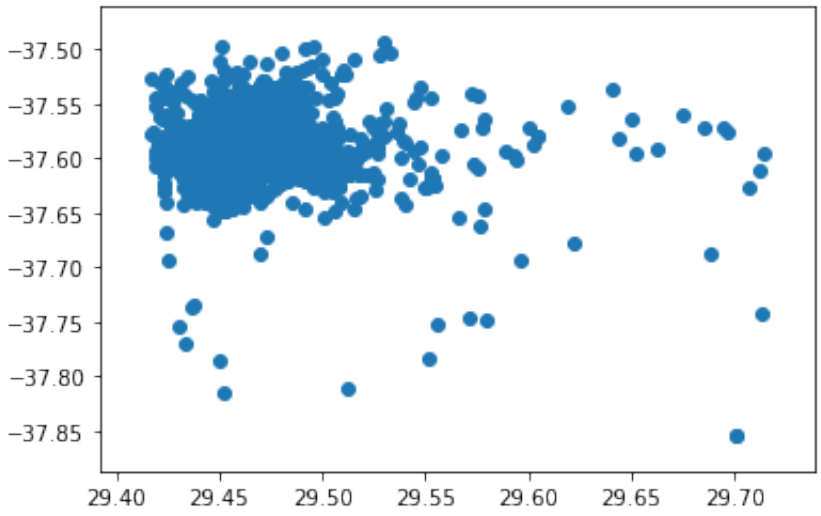
In [11]:
```
# Get the coordinates of points in pixels from geographical coordinate
s.
def lat_lng_to_pixels(lat, lng):
    lat_rad = lat * np.pi / 180.0
    lat_rad = np.log(np.tan((lat_rad + np.pi / 2.0) / 2.0))
    x = 100 * (lng + 180.0) / 360.0
    y = 100 * (lat_rad - np.pi) / (2.0 * np.pi)
    return (x, y)
```

In [12]:
```
# Get pickup coordinates from pickup latitude and longitude
px, py = lat_lng_to_pixels(p_lat, p_lng)
#py.head()
type(py)
```

Out[12]:
```
pandas.core.series.Series
```

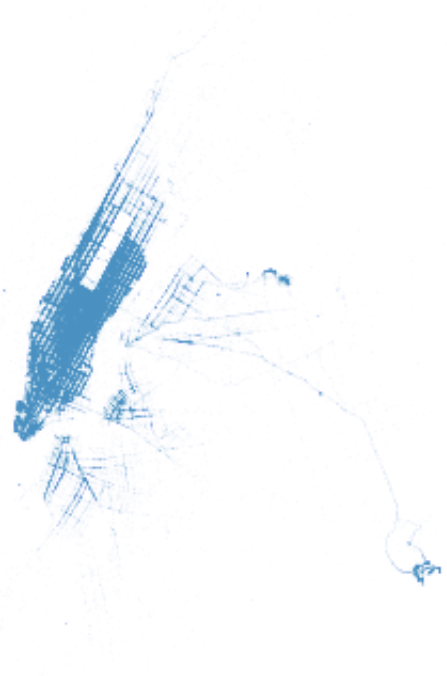
In [13]:
```
plt.scatter(px, py)
```

Out[13]:
```
<matplotlib.collections.PathCollection at 0x7fbf19543090>
```



In [15]:
```
# Specify the figure size
plt.figure(figsize=(8, 6))
# equal aspect ratio
plt.axis('equal')
# zoom in
plt.xlim(29.40, 29.55)
plt.ylim(-37.63, -37.54)
# remove the axes
plt.axis('off')
# s argument is used to make the marker size smaller
# alpha specifies opacity
plt.scatter(px, py, s=.1, alpha=0.03)
```

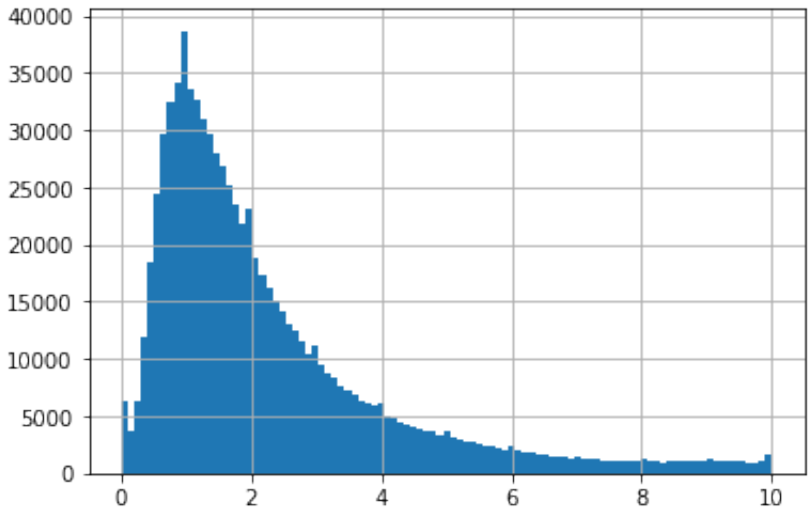
Out[15]: <matplotlib.collections.PathCollection at 0x7fbf082c8e90>



In [16]:
```
bin_array = np.linspace(start=0., stop=10., num=100)
```

In [17]:
```
data.trip_distance.hist(bins=bin_array)
```

Out[17]: <matplotlib.axes._subplots.AxesSubplot at 0x7fbefcde4e90>



In [2]:
```
data.trip_distance.head()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-2-09c5e2d519d1> in <module>
----> 1 data.trip_distance.head()

NameError: name 'data' is not defined
```

In [9]:
```
(data.trip_distance > 100).head()
```

Out[9]:
```
0    False
1    False
2    False
3    False
4    False
Name: trip_distance, dtype: bool
```

In [10]:
```
data.loc[data.trip_distance > 100]
```

Out[10]:

	medallion	hack_license	vendor_
504497	7237EC7ABD6114EDDC87A3AA846F8418	D52502537E2DF62C9BFFECF5A387E7E9	CM
507107	50DA72F510E2F84A42712E13744FAC7B	EA9D03A766C1D32A6668FFF0C1EB4E4B	CM
548988	A978A0AAE9B2CFEE310FACD97A09C319	CE56A27F53ABF411094B6CD708BFBA96	CM
558665	5A5C516A820FE476E9D3E14101B669AC	C24585AA866FC76A4E09A05F55DC7E54	CM

In [1]:
```
from ipywidgets import interact
#@interact is a decorator to create a widget.
@interact
def show_nrows(distance_threshold=(0, 100)):
    return len(data.loc[data.trip_distance > distance_threshold])
```

In [ ]:
```
```