Project 1: Hot Wire(less)

Yuding Ai
Penn ID: 31295008
CIS 553 - Networked Systems

February 20, 2017

1 Phase 1: Take 5

In Phase 1, we found 5 WAPs (Wifi access point) and recorded the BSSIDs and origin locations (lat./long.) However, I later realized that the origin locations I found last time are not accurate enough, so this time I choose the **location with strongest signal to be as the origin of the WAPs** and attach them again in this final report as below. (the BSSIDs are the same as in my Phase 1 email though.)
5 WAPs for Phase 1:

1. BSSID1 84:d4:7e:dc:2a:40, Latitude: 39.95233362, Longitude: -75.19346528
2. BSSID2 94:b4:0f:53:07:00, Latitude: 39.95322759, Longitude: -75.19217689
3. BSSID3 ac:86:74:57:d6:82, Latitude: 39.95303359, Longitude: -75.19318620
4. BSSID4 e2:55:7d:46:ba:70, Latitude: 39.95348435, Longitude: -75.19542690
5. BSSID5 e8:40:40:80:52:e8, Latitude: 39.95339425, Longitude: -75.19608680

2 Phase 2: Tower of Power

In Phase 2, we collected power levels (in dBm) for the 5 WAPs found in Phase 1 with more than 20 locations each. Furthermore, besides the required BSSID, Lat, Long and power(in dBm), I have also included the distance in meters (**Dis**)[1] from each location to its WAP origin and the 'ideal' signal strength in dBm (**Rpower**) in Free space, calculated through the idea of $Power \propto \frac{1}{r^2}$, where r is the distance from the location to its WAP origin. To calculate this so-called Rpower, we first convert the power of the origin from dBm into mW and compute the Rpower for the target location in mW, and then converted it back to dBm as to make comparison with the observed data(the actual signal I measured using my Android phone

[1]distance is calculated using the so called Harversine formula, see detailed calculation in python source code and see reference:http://www.movable-type.co.uk/scripts/latlong.html

with the wigle app). Equation 1 describes how dBm converted to mW, and vice versa where x is in unit of dBm and P is in unit of mW.

$$x = 10 \log_{10} \frac{P}{1mW}$$
$$P = 1mW \times 10^{\frac{x}{10}} \tag{1}$$

So for example: if the origin has a Power of -41 dBm, when converted into mW, it should be 0.000079432823472 mW. An location that is 7.04 meters away from the origin, should have $Rpower = 0.000079432823472 \times \frac{1}{7.04^2} = 1.6027 \times 10^{-6}\ mW = -57.95\ dBm$

Listing 1: WAP1_Library

```
BSSID                  Latitude     longitude     Power  Dis    Rpower
84:d4:7e:dc:2a:40      39.95233362  -75.19346528  -41    0.0    -41.0
84:d4:7e:dc:2a:40      39.95230982  -75.19354176  -55    7.04   -57.95
84:d4:7e:dc:2a:40      39.95228379  -75.19354936  -61    9.06   -60.14
84:d4:7e:dc:2a:40      39.95227624  -75.19355259  -61    9.81   -60.83
84:d4:7e:dc:2a:40      39.95232304  -75.19359157  -67    10.83  -61.69
84:d4:7e:dc:2a:40      39.95236796  -75.19357628  -68    10.21  -61.18
84:d4:7e:dc:2a:40      39.95237908  -75.19354717  -63    8.62   -59.71
84:d4:7e:dc:2a:40      39.9523782   -75.19348767  -58    5.31   -55.51
84:d4:7e:dc:2a:40      39.95237557  -75.19339785  -60    7.4    -58.39
84:d4:7e:dc:2a:40      39.9523449   -75.1933452   -70    10.32  -61.27
84:d4:7e:dc:2a:40      39.95231382  -75.19332773  -75    11.93  -62.54
84:d4:7e:dc:2a:40      39.95228106  -75.19327585  -76    17.18  -65.7
84:d4:7e:dc:2a:40      39.95228917  -75.19323223  -76    20.48  -67.23
84:d4:7e:dc:2a:40      39.95231623  -75.19326823  -70    16.91  -65.56
84:d4:7e:dc:2a:40      39.95232877  -75.19337275  -70    7.91   -58.96
84:d4:7e:dc:2a:40      39.9522439   -75.19339994  -82    11.43  -62.16
84:d4:7e:dc:2a:40      39.95221351  -75.1934048   -83    14.32  -64.12
84:d4:7e:dc:2a:40      39.95216466  -75.19332293  -78    22.37  -67.99
84:d4:7e:dc:2a:40      39.95214442  -75.1932546   -80    27.67  -69.84
84:d4:7e:dc:2a:40      39.95216179  -75.19316639  -85    31.86  -71.06
84:d4:7e:dc:2a:40      39.9523527   -75.1934005   -69    5.92   -56.44
84:d4:7e:dc:2a:40      39.95239043  -75.19345689  -62    6.36   -57.07
84:d4:7e:dc:2a:40      39.95242305  -75.19352098  -52    11.02  -61.85
84:d4:7e:dc:2a:40      39.95242487  -75.19343724  -61    10.43  -61.36
84:d4:7e:dc:2a:40      39.95240718  -75.19331876  -67    14.93  -64.48
84:d4:7e:dc:2a:40      39.95240224  -75.1932791   -69    17.61  -65.92
84:d4:7e:dc:2a:40      39.95232515  -75.19335194  -72    9.71   -60.74
84:d4:7e:dc:2a:40      39.95229446  -75.19333902  -66    11.61  -62.3
84:d4:7e:dc:2a:40      39.95233105  -75.19334085  -77    10.61  -61.52
```

Listing 2: WAP2_Starbucks

```
BSSID                  Latitude     longitude     Power  Dis    Rpower
94:b4:0f:53:07:00      39.95300328  -75.19218521  -67    24.96  -70.94
94:b4:0f:53:07:00      39.95314201  -75.19215558  -54    9.69   -62.73
94:b4:0f:53:07:00      39.95308746  -75.19214409  -54    15.84  -66.99
94:b4:0f:53:07:00      39.95322759  -75.19217689  -43    0.0    -43.0
94:b4:0f:53:07:00      39.95331679  -75.19219967  -68    10.11  -63.1
94:b4:0f:53:07:00      39.95341871  -75.19223796  -77    21.89  -69.8
94:b4:0f:53:07:00      39.95341624  -75.19221528  -69    21.24  -69.54
```

94:b4:0f:53:07:00	39.95336173	-75.19219009	-64	14.96	-66.5
94:b4:0f:53:07:00	39.95333451	-75.19220592	-67	12.15	-64.69
94:b4:0f:53:07:00	39.95329762	-75.19221843	-65	8.56	-61.65
94:b4:0f:53:07:00	39.95323437	-75.19222063	-55	3.8	-54.61
94:b4:0f:53:07:00	39.95316914	-75.19222198	-54	7.55	-60.56
94:b4:0f:53:07:00	39.95305603	-75.19222518	-56	19.52	-68.81
94:b4:0f:53:07:00	39.9530076	-75.19222164	-69	24.77	-70.88
94:b4:0f:53:07:00	39.95296533	-75.19222223	-68	29.43	-72.37
94:b4:0f:53:07:00	39.9529472	-75.19219909	-70	31.25	-72.9
94:b4:0f:53:07:00	39.95293145	-75.19221608	-68	33.11	-73.4
94:b4:0f:53:07:00	39.95293909	-75.19220119	-70	32.16	-73.15
94:b4:0f:53:07:00	39.95294534	-75.19199002	-70	35.21	-73.93
94:b4:0f:53:07:00	39.95302457	-75.19194761	-72	29.87	-72.5
94:b4:0f:53:07:00	39.95307636	-75.1918776	-68	30.56	-72.7
94:b4:0f:53:07:00	39.95329533	-75.19169873	-68	41.46	-75.35
94:b4:0f:53:07:00	39.95334517	-75.19191655	-73	25.76	-71.22
94:b4:0f:53:07:00	39.953197	-75.191923	-59	21.91	-69.81
94:b4:0f:53:07:00	39.95318632	-75.19193303	-59	21.29	-69.56
94:b4:0f:53:07:00	39.95323969	-75.19191041	-61	22.76	-70.14
94:b4:0f:53:07:00	39.95331454	-75.19192654	-65	23.43	-70.4
94:b4:0f:53:07:00	39.95331931	-75.19193617	-68	22.92	-70.2
94:b4:0f:53:07:00	39.95334841	-75.19195773	-67	23.02	-70.24
94:b4:0f:53:07:00	39.95335327	-75.19196058	-67	23.14	-70.29
94:b4:0f:53:07:00	39.95333676	-75.19197971	-71	20.74	-69.34
94:b4:0f:53:07:00	39.9531888	-75.19204021	-59	12.43	-64.89
94:b4:0f:53:07:00	39.95318109	-75.19197923	-70	17.63	-67.92
94:b4:0f:53:07:00	39.95315349	-75.19183212	-71	30.53	-72.69
94:b4:0f:53:07:00	39.95332706	-75.19149968	-73	58.79	-78.39
94:b4:0f:53:07:00	39.95334124	-75.1914769	-77	61.01	-78.71
94:b4:0f:53:07:00	39.9533159	-75.19142933	-75	64.49	-79.19
94:b4:0f:53:07:00	39.95329751	-75.19145057	-76	62.42	-78.91
94:b4:0f:53:07:00	39.95325922	-75.19149104	-76	58.58	-78.36

Listing 3: WAP3_United_by_blue

BSSID	Latitude	longitude	Power	Dis	Rpower
ac:86:74:57:d6:82	39.9531262	-75.19316856	-60	10.41	-61.35
ac:86:74:57:d6:82	39.95306805	-75.19322581	-61	5.11	-55.17
ac:86:74:57:d6:82	39.95303359	-75.1931862	-41	0.0	-41.0
ac:86:74:57:d6:82	39.95300903	-75.19329689	-59	9.83	-60.85
ac:86:74:57:d6:82	39.95308204	-75.19333511	-62	13.79	-63.79
ac:86:74:57:d6:82	39.95307472	-75.19350965	-64	27.96	-69.93
ac:86:74:57:d6:82	39.95305298	-75.19355542	-77	31.56	-70.98
ac:86:74:57:d6:82	39.95304666	-75.19358466	-69	34.01	-71.63
ac:86:74:57:d6:82	39.95309792	-75.19371891	-79	45.98	-74.25
ac:86:74:57:d6:82	39.95311104	-75.19376055	-75	49.72	-74.93
ac:86:74:57:d6:82	39.95314332	-75.19378227	-81	52.27	-75.36
ac:86:74:57:d6:82	39.95315069	-75.19381858	-82	55.47	-75.88
ac:86:74:57:d6:82	39.95318367	-75.1938578	-82	59.65	-76.51
ac:86:74:57:d6:82	39.95316839	-75.19387127	-82	60.31	-76.61
ac:86:74:57:d6:82	39.9529689	-75.19327191	-74	10.26	-61.22
ac:86:74:57:d6:82	39.95295116	-75.1932809	-70	12.22	-62.74
ac:86:74:57:d6:82	39.95298173	-75.19337015	-66	16.71	-65.46
ac:86:74:57:d6:82	39.95299407	-75.19344377	-64	22.4	-68.0

ac:86:74:57:d6:82	39.95297741	-75.19354724	-76	31.41	-70.94
ac:86:74:57:d6:82	39.9530528	-75.19362915	-67	37.83	-72.56
ac:86:74:57:d6:82	39.95305761	-75.19366132	-76	40.6	-73.17
ac:86:74:57:d6:82	39.95311303	-75.19377652	-74	51.1	-75.17
ac:86:74:57:d6:82	39.95304962	-75.19363119	-71	37.98	-72.59
ac:86:74:57:d6:82	39.95300972	-75.19348948	-71	26.0	-69.3
ac:86:74:57:d6:82	39.95293554	-75.19321491	-69	11.18	-61.97
ac:86:74:57:d6:82	39.95287939	-75.19300501	-79	23.08	-68.27

Listing 4: WAP4_Bluemercury

BSSID	Latitude	longitude	Power	Dis	Rpower
e2:55:7d:46:ba:70	39.95344475	-75.1953241	-67	9.81	-61.83
e2:55:7d:46:ba:70	39.95346047	-75.19534486	-68	7.48	-59.48
e2:55:7d:46:ba:70	39.95351389	-75.19536677	-62	6.09	-57.69
e2:55:7d:46:ba:70	39.9535116	-75.19539087	-62	4.32	-54.7
e2:55:7d:46:ba:70	39.95351465	-75.19540162	-56	4.0	-54.04
e2:55:7d:46:ba:70	39.95348435	-75.1954269	-42	0.0	-42.0
e2:55:7d:46:ba:70	39.95350455	-75.19552856	-61	8.95	-61.04
e2:55:7d:46:ba:70	39.95353418	-75.19554469	-61	11.47	-63.19
e2:55:7d:46:ba:70	39.95356256	-75.1956038	-68	17.41	-66.82
e2:55:7d:46:ba:70	39.95353174	-75.19586533	-69	37.75	-73.54
e2:55:7d:46:ba:70	39.95344449	-75.19582356	-68	34.11	-72.66
e2:55:7d:46:ba:70	39.95345135	-75.19575759	-69	28.43	-71.08
e2:55:7d:46:ba:70	39.95342285	-75.19569923	-57	24.21	-69.68
e2:55:7d:46:ba:70	39.95340621	-75.19565746	-69	21.49	-68.65
e2:55:7d:46:ba:70	39.95338485	-75.19551169	-60	13.22	-64.42
e2:55:7d:46:ba:70	39.9533588	-75.19546526	-60	14.34	-65.13
e2:55:7d:46:ba:70	39.95336648	-75.19540467	-72	13.25	-64.44
e2:55:7d:46:ba:70	39.95336993	-75.19533335	-72	15.02	-65.53
e2:55:7d:46:ba:70	39.95327412	-75.1955647	-65	26.17	-70.36
e2:55:7d:46:ba:70	39.95328151	-75.19562271	-65	28.07	-70.96
e2:55:7d:46:ba:70	39.95328667	-75.19567539	-76	30.53	-71.7
e2:55:7d:46:ba:70	39.95329931	-75.19576064	-71	35.12	-72.91
e2:55:7d:46:ba:70	39.95333245	-75.19582484	-70	37.9	-73.57
e2:55:7d:46:ba:70	39.95343911	-75.19562141	-69	17.33	-66.78
e2:55:7d:46:ba:70	39.95343569	-75.19557638	-69	13.85	-64.83
e2:55:7d:46:ba:70	39.95322091	-75.19541343	-66	29.32	-71.34
e2:55:7d:46:ba:70	39.95330729	-75.19556592	-66	22.99	-69.23
e2:55:7d:46:ba:70	39.95339435	-75.19585788	-70	38.09	-73.62
e2:55:7d:46:ba:70	39.95330516	-75.19589166	-71	44.36	-74.94

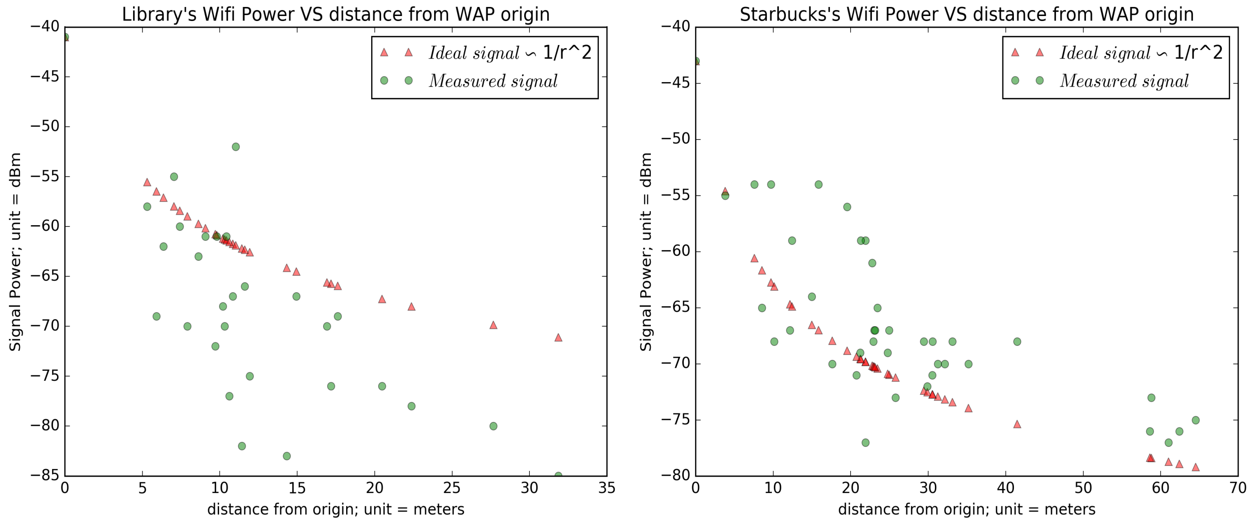
Listing 5: WAP5_PenneBar

BSSID	Latitude	longitude	Power	Dis	Rpower
e8:40:40:80:52:e8	39.95337588	-75.19609743	-70	2.24	-49.99
e8:40:40:80:52:e8	39.95337808	-75.19618277	-77	8.38	-61.46
e8:40:40:80:52:e8	39.95334292	-75.19610598	-72	5.94	-58.47
e8:40:40:80:52:e8	39.95331375	-75.19602633	-80	10.33	-63.28
e8:40:40:80:52:e8	39.95329611	-75.19600751	-72	12.84	-65.17
e8:40:40:80:52:e8	39.953289	-75.19598095	-76	14.78	-66.39
e8:40:40:80:52:e8	39.95329387	-75.1959173	-70	18.26	-68.23
e8:40:40:80:52:e8	39.95333236	-75.19589952	-70	17.39	-67.81
e8:40:40:80:52:e8	39.95327273	-75.19608164	-80	13.52	-65.62

e8:40:40:80:52:e8	39.95336876	-75.19570731	-70	32.48	-73.23
e8:40:40:80:52:e8	39.95344523	-75.19592838	-64	14.65	-66.32
e8:40:40:80:52:e8	39.95342261	-75.19593443	-82	13.37	-65.52
e8:40:40:80:52:e8	39.9534374	-75.19599858	-71	8.92	-62.01
e8:40:40:80:52:e8	39.9534569	-75.19608773	-72	6.97	-59.86
e8:40:40:80:52:e8	39.95350044	-75.19651072	-82	38.03	-74.6
e8:40:40:80:52:e8	39.95345781	-75.19636381	-78	24.65	-70.84
e8:40:40:80:52:e8	39.9534526	-75.19627185	-69	17.06	-67.64
e8:40:40:80:52:e8	39.95336277	-75.19591907	-65	14.72	-66.36
e8:40:40:80:52:e8	39.95336332	-75.19594091	-64	12.91	-65.22
e8:40:40:80:52:e8	39.95340967	-75.19603491	-64	4.75	-56.53
e8:40:40:80:52:e8	39.95338347	-75.19618058	-71	8.09	-61.15
e8:40:40:80:52:e8	39.95337122	-75.19617366	-70	7.84	-60.88
e8:40:40:80:52:e8	39.95338304	-75.19612399	-71	3.41	-53.65
e8:40:40:80:52:e8	39.95338875	-75.19603628	-63	4.35	-55.77
e8:40:40:80:52:e8	39.95339425	-75.1960868	-43	0.0	-43.0
e8:40:40:80:52:e8	39.95339517	-75.19612735	-66	3.46	-53.78
e8:40:40:80:52:e8	39.95340651	-75.19616441	-66	6.76	-59.59
e8:40:40:80:52:e8	39.95343159	-75.19620826	-76	11.16	-63.95
e8:40:40:80:52:e8	39.95349377	-75.19634645	-79	24.75	-70.87

Using the above data, we could simply plot Power VS Distance as to compare the measured Wifi signal strength with the ideal signal strength. Such result is depicted in Figure 1, Figure 2 and Figure 3.

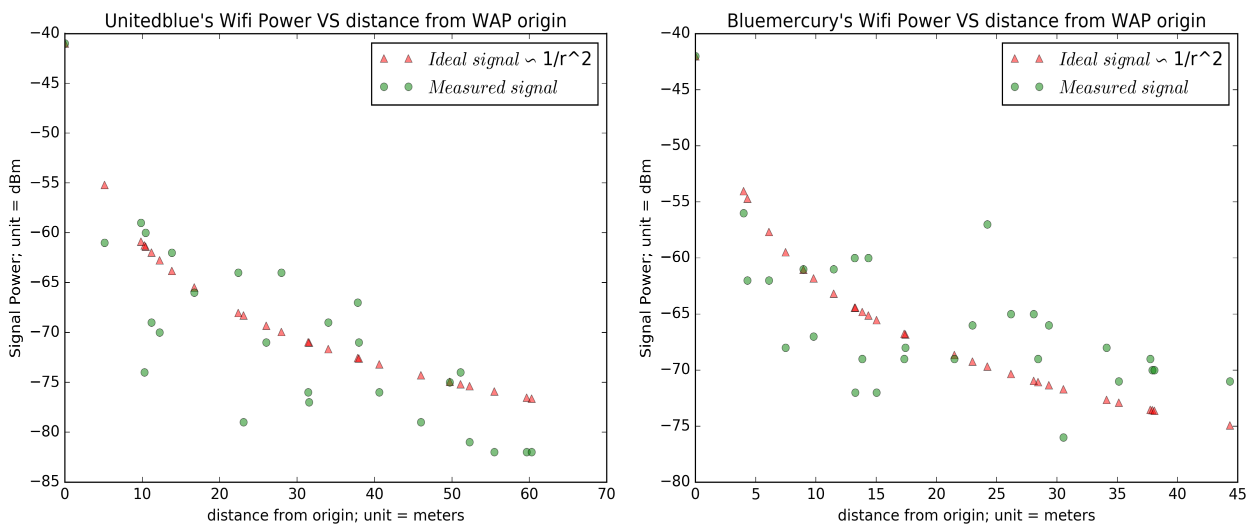
Next we compare the attenuation at each point to what the $1/r^2$ model would predict. As is shown in the Power VS Distance plots for Library, and Penne-bar, almost always, the measured Wifi signal strength is a bit weaker than the Ideal signal strength (indicates the actual signal attenuation is greater then $1/r^2$ model), such that the average bulk propagation loss is a little greater than what $1/r^2$ predicts and this result is expected due to the fact that the real world is not a "free space", so that signals will be reflected and bounced back or block by obstacles, thus that the actual signal is a bit weaker then the ideal case predicted by $Power \propto \frac{1}{r^2}$. Nevertheless, for some other plots, such like the starbucks one, we see that for about half of the times, the measured signal are actually greater than ideal case (and they are not supposed to be) and this could be explained by an inaccurate measurement of origin location and origin signal strength since the Wifi router of Starbucks is actually located inside the building, but once we get into a building, we lost the GPS signal so it cause the inaccuracy. And so does all my other WAPs except for the one from Library. My target Library router is located near the glass wall of the underground floor, which is very close to outside so that I could get a relatively more accurate location and signal strength from outside.



(a) Library Power vs Distance

(b) Starbucks Power vs Distance

Figure 1: Library and Starbucks



(a) United by Blue Power vs Distance

(b) Bluemercury Power vs Distance

Figure 2: United by Blue and Bluemercury Power vs Distance

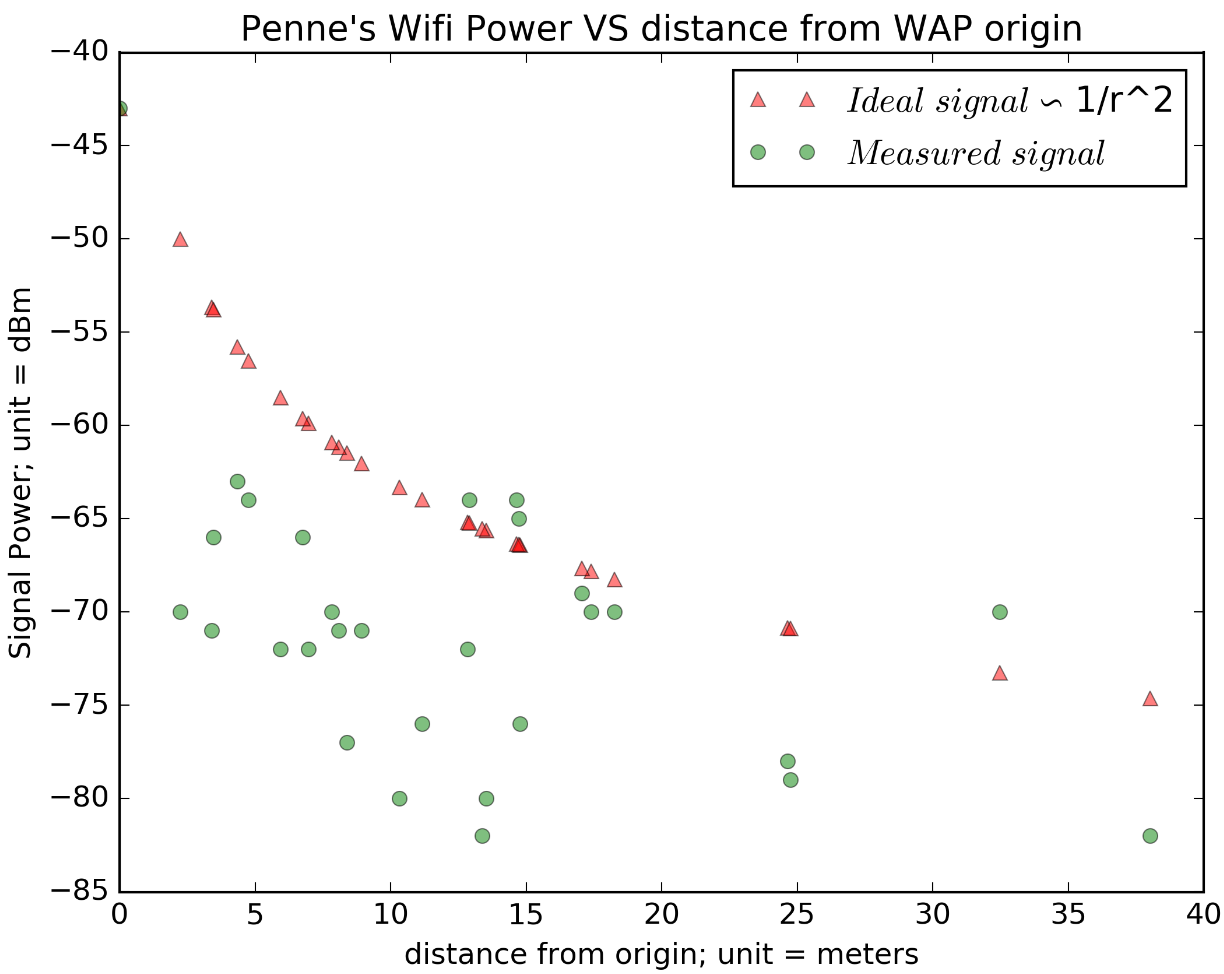


Figure 3: Penne-bar Power vs Distance

2.1 Calculate the Average bulk propagation loss

I add this section on Feb 20 as to include the calculation for average bulk propagation. All calculation is based on Piazza post @64 and assume the noise power to be $P_{noise} = -100dBm = 10^{-10}mW$.

To calculate the Signal power to noise ratio in dB, we use Equation 2.

$$SNR_{dB} = 10\log_{10}(P_r/P_{noise}) \tag{2}$$

such that we could obtain the SNR by plugging the signal power (in mW) for each WAP into P_r. (Applies to both the measured signal power and the theoretical signal power predicted by $1/r^2$, so my Rpower). Then we assume

$$SNR = \frac{k}{r^2} \tag{3}$$

thus that the average bulk propagation loss k_{ave} could then be obtained by taking the average of all k values. (k in unit of $dB * m^2$.).
Finally, we attached the calculated Average bulk propagation loss along with the propagation loss (k) for each location as below:

Listing 6: Propagation_Loss_Library

```
Average propagation loss is 27324.2784447 dB*m^2
Average propagation loss according to 1/r^2 is 25514.3967432 dB*m^2
For each location, the propagation loss k and the 1/r^2 prediction k_r is
    shown below along with the distance to the origin WAP
k = 5647.0 dB*m^2;      k_r = 5793.0 dB*m^2;    dis = 7.03802631076 m
k = 9854.0 dB*m^2;      k_r = 9784.0 dB*m^2;    dis = 9.0618833097 m
k = 11539.0 dB*m^2;     k_r = 11522.0 dB*m^2;   dis = 9.80596429778 m
k = 14785.0 dB*m^2;     k_r = 14162.0 dB*m^2;   dis = 10.832413402 m
k = 13229.0 dB*m^2;     k_r = 12519.0 dB*m^2;   dis = 10.206259806 m
k = 9067.0 dB*m^2;      k_r = 8823.0 dB*m^2;    dis = 8.62108585467 m
k = 3303.0 dB*m^2;      k_r = 3233.0 dB*m^2;    dis = 5.31344446139 m
k = 6525.0 dB*m^2;      k_r = 6436.0 dB*m^2;    dis = 7.40468297856 m
k = 13727.0 dB*m^2;     k_r = 12798.0 dB*m^2;   dis = 10.3153821261 m
k = 19082.0 dB*m^2;     k_r = 17307.0 dB*m^2;   dis = 11.9333880556 m
k = 39834.0 dB*m^2;     k_r = 36795.0 dB*m^2;   dis = 17.1775019374 m
k = 56608.0 dB*m^2;     k_r = 52928.0 dB*m^2;   dis = 20.4771923703 m
k = 36899.0 dB*m^2;     k_r = 35630.0 dB*m^2;   dis = 16.9127305601 m
k = 8067.0 dB*m^2;      k_r = 7377.0 dB*m^2;    dis = 7.90812628916 m
k = 18419.0 dB*m^2;     k_r = 15827.0 dB*m^2;   dis = 11.4293815818 m
k = 29121.0 dB*m^2;     k_r = 25249.0 dB*m^2;   dis = 14.3205614485 m
k = 68571.0 dB*m^2;     k_r = 63563.0 dB*m^2;   dis = 22.3721973827 m
k = 106416.0 dB*m^2;    k_r = 98638.0 dB*m^2;   dis = 27.6691633409 m
k = 146130.0 dB*m^2;    k_r = 131988.0 dB*m^2;  dis = 31.8558296901 m
k = 4482.0 dB*m^2;      k_r = 4042.0 dB*m^2;    dis = 5.91723931967 m
k = 4893.0 dB*m^2;      k_r = 4694.0 dB*m^2;    dis = 6.35933311975 m
k = 13487.0 dB*m^2;     k_r = 14683.0 dB*m^2;   dis = 11.0229156834 m
k = 13048.0 dB*m^2;     k_r = 13088.0 dB*m^2;   dis = 10.4275174315 m
```

```
k = 28101.0 dB*m^2;     k_r = 27540.0 dB*m^2;   dis = 14.934091421 m
k = 39714.0 dB*m^2;     k_r = 38758.0 dB*m^2;   dis = 17.6144506426 m
k = 12351.0 dB*m^2;     k_r = 11290.0 dB*m^2;   dis = 9.70991283476 m
k = 16859.0 dB*m^2;     k_r = 16360.0 dB*m^2;   dis = 11.6135225709 m
k = 15320.0 dB*m^2;     k_r = 13576.0 dB*m^2;   dis = 10.6135545399 m
```

Listing 7: Propagation_Loss_Starbucks

```
Average propagation loss is 125281.031958 dB*m^2
Average propagation loss according to 1/r^2 is 128739.704229 dB*m^2
For each location, the propagation loss k and the 1/r^2 prediction k_r is
    shown below along with the distance to the origin WAP
k = 77252.0 dB*m^2;     k_r = 79710.0 dB*m^2;   dis = 24.9600472545 m
k = 10424.0 dB*m^2;     k_r = 11244.0 dB*m^2;   dis = 9.69091351878 m
k = 27835.0 dB*m^2;     k_r = 31093.0 dB*m^2;   dis = 15.8355549803 m
k = 12777.0 dB*m^2;     k_r = 12275.0 dB*m^2;   dis = 10.1100370713 m
k = 64190.0 dB*m^2;     k_r = 60742.0 dB*m^2;   dis = 21.8866942924 m
k = 56829.0 dB*m^2;     k_r = 57073.0 dB*m^2;   dis = 21.2372858308 m
k = 27090.0 dB*m^2;     k_r = 27650.0 dB*m^2;   dis = 14.9627600728 m
k = 18298.0 dB*m^2;     k_r = 17957.0 dB*m^2;   dis = 12.1475524689 m
k = 8933.0 dB*m^2;      k_r = 8687.0 dB*m^2;    dis = 8.55688864768 m
k = 1622.0 dB*m^2;      k_r = 1616.0 dB*m^2;    dis = 3.8049882474 m
k = 6332.0 dB*m^2;      k_r = 6707.0 dB*m^2;    dis = 7.55308274775 m
k = 43064.0 dB*m^2;     k_r = 47946.0 dB*m^2;   dis = 19.521751758 m
k = 77277.0 dB*m^2;     k_r = 78429.0 dB*m^2;   dis = 24.7651582553 m
k = 108238.0 dB*m^2;    k_r = 112026.0 dB*m^2;  dis = 29.4261899113 m
k = 123985.0 dB*m^2;    k_r = 126812.0 dB*m^2;  dis = 31.245123373 m
k = 137023.0 dB*m^2;    k_r = 142941.0 dB*m^2;  dis = 33.1086609123 m
k = 131324.0 dB*m^2;    k_r = 134577.0 dB*m^2;  dis = 32.1566274803 m
k = 157417.0 dB*m^2;    k_r = 162291.0 dB*m^2;  dis = 35.2065434914 m
k = 115085.0 dB*m^2;    k_r = 115535.0 dB*m^2;  dis = 29.8685573791 m
k = 116772.0 dB*m^2;    k_r = 121167.0 dB*m^2;  dis = 30.564352689 m
k = 214875.0 dB*m^2;    k_r = 227515.0 dB*m^2;  dis = 41.4608434257 m
k = 86293.0 dB*m^2;     k_r = 85112.0 dB*m^2;   dis = 25.7641993674 m
k = 55705.0 dB*m^2;     k_r = 60898.0 dB*m^2;   dis = 21.9137838684 m
k = 52596.0 dB*m^2;     k_r = 57386.0 dB*m^2;   dis = 21.2935022869 m
k = 61133.0 dB*m^2;     k_r = 65871.0 dB*m^2;   dis = 22.7613297699 m
k = 67002.0 dB*m^2;     k_r = 69966.0 dB*m^2;   dis = 23.4349160037 m
k = 65670.0 dB*m^2;     k_r = 66828.0 dB*m^2;   dis = 22.9207063662 m
k = 65695.0 dB*m^2;     k_r = 67412.0 dB*m^2;   dis = 23.0172773286 m
k = 66414.0 dB*m^2;     k_r = 68175.0 dB*m^2;   dis = 23.1428980508 m
k = 55055.0 dB*m^2;     k_r = 54339.0 dB*m^2;   dis = 20.7392184607 m
k = 17914.0 dB*m^2;     k_r = 18823.0 dB*m^2;   dis = 12.427113184 m
k = 39471.0 dB*m^2;     k_r = 38826.0 dB*m^2;   dis = 17.6293424569 m
k = 119310.0 dB*m^2;    k_r = 120890.0 dB*m^2;  dis = 30.5305269247 m
k = 449359.0 dB*m^2;    k_r = 467978.0 dB*m^2;  dis = 58.7929281818 m
k = 498759.0 dB*m^2;    k_r = 505116.0 dB*m^2;  dis = 61.008857815 m
k = 549039.0 dB*m^2;    k_r = 566468.0 dB*m^2;  dis = 64.4933275324 m
k = 518141.0 dB*m^2;    k_r = 529462.0 dB*m^2;  dis = 62.4163064831 m
k = 456482.0 dB*m^2;    k_r = 464567.0 dB*m^2;  dis = 58.5849494862 m
```

Listing 8: Propagation_Loss_Unitedbyblue

```
Average propagation loss is 170483.573966 dB*m^2
```

```
Average propagation loss according to 1/r^2 is 167041.756226 dB*m^2
For each location, the propagation loss k and the 1/r^2 prediction k_r is
   shown below along with the distance to the origin WAP
k = 12896.0 dB*m^2;     k_r = 13043.0 dB*m^2;   dis = 10.4102239575 m
k = 3132.0 dB*m^2;      k_r = 2980.0 dB*m^2;    dis = 5.10865485862 m
k = 11392.0 dB*m^2;     k_r = 11570.0 dB*m^2;   dis = 9.82545213672 m
k = 23021.0 dB*m^2;     k_r = 23362.0 dB*m^2;   dis = 13.7932343381 m
k = 96129.0 dB*m^2;     k_r = 100763.0 dB*m^2;  dis = 27.9560121846 m
k = 135422.0 dB*m^2;    k_r = 129429.0 dB*m^2;  dis = 31.5555415094 m
k = 148020.0 dB*m^2;    k_r = 151063.0 dB*m^2;  dis = 34.0060065896 m
k = 291779.0 dB*m^2;    k_r = 281740.0 dB*m^2;  dis = 45.9819545238 m
k = 331314.0 dB*m^2;    k_r = 331145.0 dB*m^2;  dis = 49.7241962193 m
k = 382490.0 dB*m^2;    k_r = 367094.0 dB*m^2;  dis = 52.2691923266 m
k = 433862.0 dB*m^2;    k_r = 415035.0 dB*m^2;  dis = 55.471061968 m
k = 501659.0 dB*m^2;    k_r = 482134.0 dB*m^2;  dis = 59.6478702991 m
k = 512799.0 dB*m^2;    k_r = 493186.0 dB*m^2;  dis = 60.306485113 m
k = 13989.0 dB*m^2;     k_r = 12645.0 dB*m^2;   dis = 10.2558847431 m
k = 19255.0 dB*m^2;     k_r = 18171.0 dB*m^2;   dis = 12.2173914497 m
k = 34910.0 dB*m^2;     k_r = 34759.0 dB*m^2;   dis = 16.7117066249 m
k = 61703.0 dB*m^2;     k_r = 63711.0 dB*m^2;   dis = 22.3974640459 m
k = 133207.0 dB*m^2;    k_r = 128217.0 dB*m^2;  dis = 31.4121236523 m
k = 180308.0 dB*m^2;    k_r = 188259.0 dB*m^2;  dis = 37.828774679 m
k = 222522.0 dB*m^2;    k_r = 217858.0 dB*m^2;  dis = 40.5993782429 m
k = 347340.0 dB*m^2;    k_r = 350394.0 dB*m^2;  dis = 51.1036458651 m
k = 187564.0 dB*m^2;    k_r = 189861.0 dB*m^2;  dis = 37.9841948492 m
k = 87848.0 dB*m^2;     k_r = 86698.0 dB*m^2;   dis = 25.9952918857 m
k = 15992.0 dB*m^2;     k_r = 15113.0 dB*m^2;   dis = 11.1774452228 m
k = 73535.0 dB*m^2;     k_r = 67815.0 dB*m^2;   dis = 23.0837438308 m
```

Listing 9: Propagation_Loss_Blue_mercury

```
Average propagation loss is 73843.535684 dB*m^2
Average propagation loss according to 1/r^2 is 75564.4332745 dB*m^2
For each location, the propagation loss k and the 1/r^2 prediction k_r is
   shown below along with the distance to the origin WAP
k = 12029.0 dB*m^2;     k_r = 11532.0 dB*m^2;   dis = 9.80973643617 m
k = 7054.0 dB*m^2;      k_r = 6577.0 dB*m^2;    dis = 7.48246311331 m
k = 4450.0 dB*m^2;      k_r = 4290.0 dB*m^2;    dis = 6.08949774953 m
k = 2235.0 dB*m^2;      k_r = 2099.0 dB*m^2;    dis = 4.31564877964 m
k = 1825.0 dB*m^2;      k_r = 1793.0 dB*m^2;    dis = 4.00060685343 m
k = 9542.0 dB*m^2;      k_r = 9545.0 dB*m^2;    dis = 8.9545082264 m
k = 15659.0 dB*m^2;     k_r = 15948.0 dB*m^2;   dis = 11.471239934 m
k = 38201.0 dB*m^2;     k_r = 37843.0 dB*m^2;   dis = 17.4122278352 m
k = 181007.0 dB*m^2;    k_r = 187476.0 dB*m^2;  dis = 37.7525310503 m
k = 146605.0 dB*m^2;    k_r = 152025.0 dB*m^2;  dis = 34.1105933572 m
k = 102680.0 dB*m^2;    k_r = 104359.0 dB*m^2;  dis = 28.4342078513 m
k = 67387.0 dB*m^2;     k_r = 74816.0 dB*m^2;   dis = 24.2068879994 m
k = 58675.0 dB*m^2;     k_r = 58512.0 dB*m^2;   dis = 21.4943512713 m
k = 20621.0 dB*m^2;     k_r = 21394.0 dB*m^2;   dis = 13.2194603105 m
k = 24275.0 dB*m^2;     k_r = 25330.0 dB*m^2;   dis = 14.3428210776 m
k = 22813.0 dB*m^2;     k_r = 21486.0 dB*m^2;   dis = 13.2469664787 m
k = 29328.0 dB*m^2;     k_r = 27869.0 dB*m^2;   dis = 15.0199785993 m
k = 84237.0 dB*m^2;     k_r = 87905.0 dB*m^2;   dis = 26.1697526684 m
k = 96898.0 dB*m^2;     k_r = 101596.0 dB*m^2;  dis = 28.0675280353 m
```

```
k = 124939.0 dB*m^2;    k_r = 120926.0 dB*m^2;  dis = 30.5348967158 m
k = 159106.0 dB*m^2;    k_r = 161463.0 dB*m^2;  dis = 35.1195558082 m
k = 183903.0 dB*m^2;    k_r = 189037.0 dB*m^2;  dis = 37.9043569931 m
k = 38148.0 dB*m^2;     k_r = 37480.0 dB*m^2;   dis = 17.3314861883 m
k = 24351.0 dB*m^2;     k_r = 23551.0 dB*m^2;   dis = 13.8470300959 m
k = 106634.0 dB*m^2;    k_r = 111230.0 dB*m^2;  dis = 29.3248870774 m
k = 65518.0 dB*m^2;     k_r = 67225.0 dB*m^2;   dis = 22.9863962634 m
k = 185676.0 dB*m^2;    k_r = 190921.0 dB*m^2;  dis = 38.0866834995 m
k = 253823.0 dB*m^2;    k_r = 261574.0 dB*m^2;  dis = 44.3578598309 m
```

Listing 10: Propagation_Loss_Penne_Bar

```
Average propagation loss is 31520.6125474 dB*m^2
Average propagation loss according to 1/r^2 is 30289.1472939 dB*m^2
For each location, the propagation loss k and the 1/r^2 prediction k_r is
   shown below along with the distance to the origin WAP
k = 635.0 dB*m^2;       k_r = 535.0 dB*m^2;     dis = 2.23529551086 m
k = 9406.0 dB*m^2;      k_r = 8315.0 dB*m^2;    dis = 8.37822676698 m
k = 4550.0 dB*m^2;      k_r = 4073.0 dB*m^2;    dis = 5.93902744339 m
k = 14626.0 dB*m^2;     k_r = 12841.0 dB*m^2;   dis = 10.3323921229 m
k = 21268.0 dB*m^2;     k_r = 20142.0 dB*m^2;   dis = 12.8400894252 m
k = 29062.0 dB*m^2;     k_r = 26963.0 dB*m^2;   dis = 14.7820449067 m
k = 42359.0 dB*m^2;     k_r = 41769.0 dB*m^2;   dis = 18.2629710782 m
k = 38402.0 dB*m^2;     k_r = 37739.0 dB*m^2;   dis = 17.3891162276 m
k = 25056.0 dB*m^2;     k_r = 22427.0 dB*m^2;   dis = 13.5238079492 m
k = 133989.0 dB*m^2;    k_r = 137400.0 dB*m^2;  dis = 32.4812475957 m
k = 25968.0 dB*m^2;     k_r = 26465.0 dB*m^2;   dis = 14.6496548743 m
k = 24845.0 dB*m^2;     k_r = 21899.0 dB*m^2;   dis = 13.3693213252 m
k = 10191.0 dB*m^2;     k_r = 9475.0 dB*m^2;    dis = 8.92287164527 m
k = 6265.0 dB*m^2;      k_r = 5676.0 dB*m^2;    dis = 6.9690002025 m
k = 200996.0 dB*m^2;    k_r = 190298.0 dB*m^2;  dis = 38.026480423 m
k = 82060.0 dB*m^2;     k_r = 77707.0 dB*m^2;   dis = 24.6546561707 m
k = 36676.0 dB*m^2;     k_r = 36280.0 dB*m^2;   dis = 17.0610193774 m
k = 26449.0 dB*m^2;     k_r = 26744.0 dB*m^2;   dis = 14.7239315354 m
k = 20155.0 dB*m^2;     k_r = 20358.0 dB*m^2;   dis = 12.9063127408 m
k = 2725.0 dB*m^2;      k_r = 2556.0 dB*m^2;    dis = 4.74522533606 m
k = 8368.0 dB*m^2;      k_r = 7724.0 dB*m^2;    dis = 8.08556606645 m
k = 7799.0 dB*m^2;      k_r = 7239.0 dB*m^2;    dis = 7.83662390764 m
k = 1486.0 dB*m^2;      k_r = 1285.0 dB*m^2;    dis = 3.40734967681 m
k = 2272.0 dB*m^2;      k_r = 2135.0 dB*m^2;    dis = 4.35081875329 m
k = 1472.0 dB*m^2;      k_r = 1325.0 dB*m^2;    dis = 3.45901481032 m
k = 5615.0 dB*m^2;      k_r = 5323.0 dB*m^2;    dis = 6.7564739785 m
k = 16559.0 dB*m^2;     k_r = 15059.0 dB*m^2;   dis = 11.1580920212 m
k = 83324.0 dB*m^2;     k_r = 78344.0 dB*m^2;   dis = 24.7522618857 m
```

As is shown in those propagation loss (k) data, Once again, For Library, United by blue and Penne-bar, the result indicates that the average bulk propagation loss is a little greater than what $1/r^2$ predicts and this result is expected due to the fact that the real world is not a "free space", so that signals will be reflected and bounced back or block by obstacles, thus that the actual signal is a bit weaker then the ideal case predicted by $Power \propto \frac{1}{r^2}$. (same conclusion as before). And for Starbucks and Blue mercury, due to the reason of inaccurate measurement of WAP origin location and its signal power, the propagation loss seems to be a bit smaller then what $1/r^2$ predicts.

3 Phase 3: Peak Performance

First of all, utilizing the Latitude and Longitude data we recorded in phase 2, we draw those locations directly onto google map[2] as a scatter plot to show where those measured locations are. (I am actually a little bit surprised by the great accuracy of my GPS) Such plot is depicted in Figure 4, where the dark purple dots are for Van Pelt Library, Green for Starbucks, Red for a cafe called 'united by blue', Blue for a beauty retailer named Bluemurcury and Magenta for a restaurant named 'Penne-bar'.

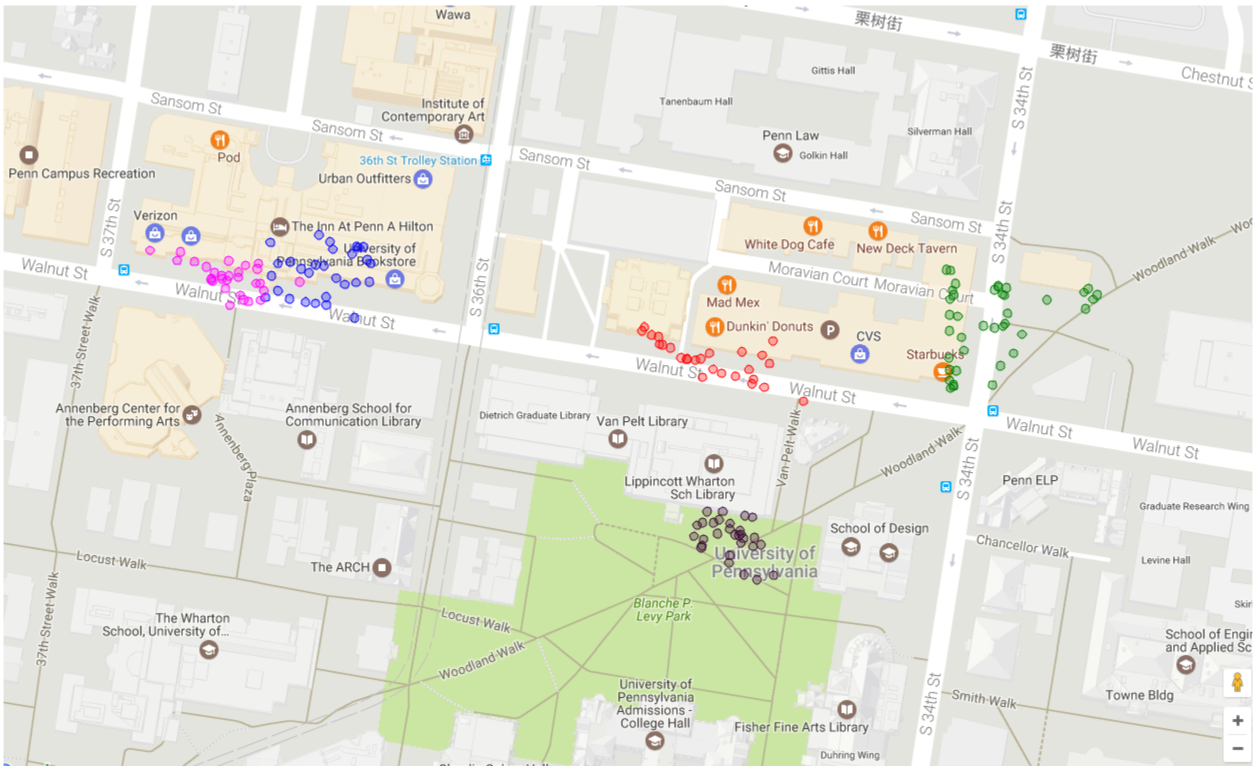
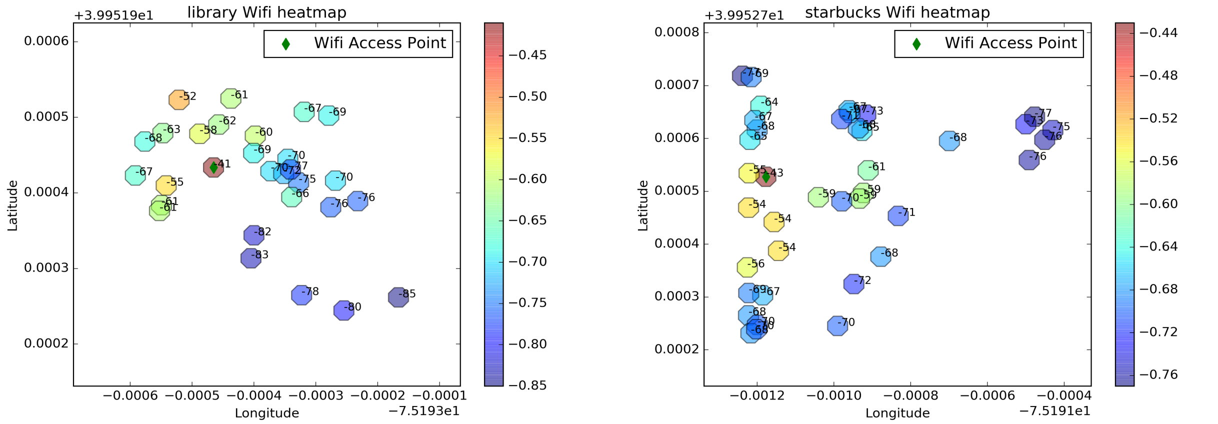


Figure 4: WAP Location Scatters on Google map

Then we plot the data from phase 2 as a heat map for each access points where the WAP origin location is marked as a green diamond symbol and a more reddish color indicates a stronger Wifi signal while a more bluish color stands for a weaker Wifi signal.(Notice that the 'pattern shape' of each plots matched with the scatter plots on google map) Such plots are draw in Figure 5 for Library and Starbucks, Figure 6 for united by blue and bluemercury and Figure 7 for penne-bar.

As is shown in those plots, we see that spots that are close to the origin remain strong enough for online surfing (more than -60 dBm) while the spots that are further away from the origin became weaker(more bluish) and weaker.(agree with the theory) Therefore, such behavior of the Wifi signal is as expected.

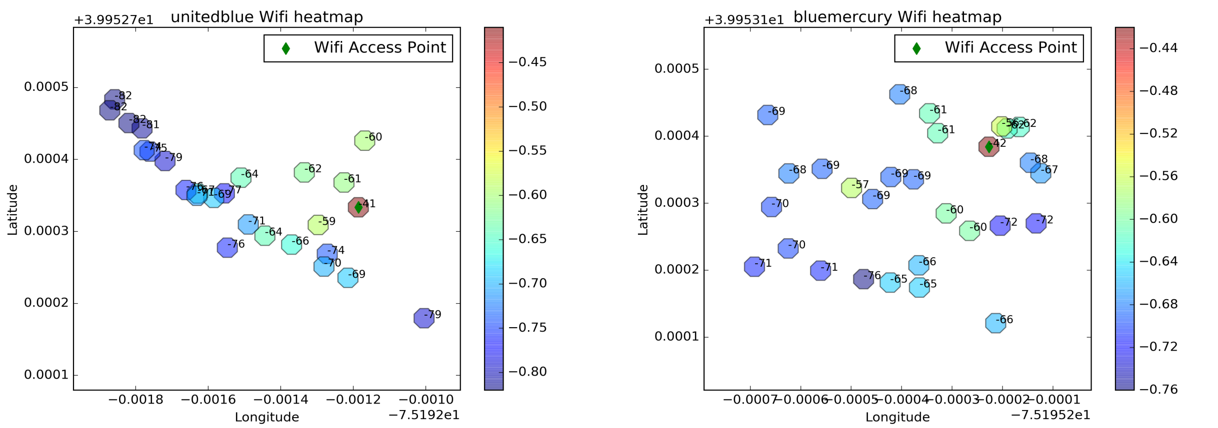
[2]Using a little python package called gmplot, see reference https://github.com/vgm64/gmplot



(a) Library Heat map

(b) Starbucks Heat map

Figure 5: Library and Starbucks Heat map



(a) United by Blue Heat map

(b) Bluemercury Heat map

Figure 6: United by Blue and Bluemercury Heat map

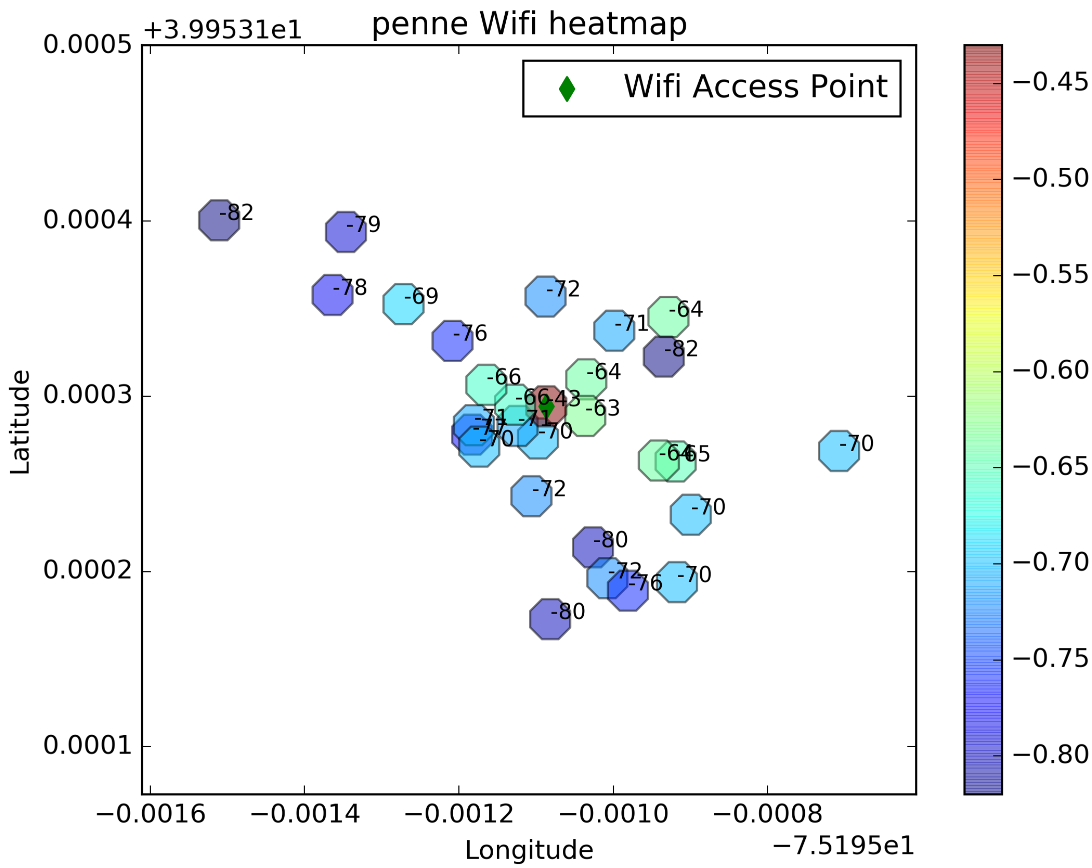


Figure 7: Penne-bar Heat map

4 Appendix: Python Source Code for data processing

In the Appendix, as for reference, we attached the python code that analyzed and visualized the data for this whole project.

Listing 11: WAP1_Library

```
# phase2.py
# CIS 553
# data process phase 2
# data visualization phase 3
# Author: Yuding Ai
# date: Feb 10
from mpl_toolkits.basemap import Basemap
import math
import gmplot
import numpy as np
import matplotlib.mlab as mlab
import matplotlib.pyplot as plt
from matplotlib import rc
from matplotlib import cm
import matplotlib.ticker as mtick
from matplotlib.ticker import LinearLocator, FormatStrFormatter


#==============================================================================
# Original data
#==============================================================================
# since I realized that the origin point of each access point I record at
    phase
#1 is not actually the accurate one, so I decided to redo the origin location
#of each access point, the BSSID is the same as in my phase 1 though

# the original data of lattitude and longitude for the library WAP 84:d4:7e:dc
    :2a:40
liboriginsignal = -41
liborigin=[39.95233362,-75.19346528]

liblat
    =[39.95233362,39.95230982,39.95228379,39.95227624,39.95232304,39.95236796,
        39.95237908,39.9523782,39.95237557,39.9523449,39.95231382,39.95228106,
        39.95228917,39.95231623,39.95232877,39.9522439,39.95221351,39.95216466,

        39.95214442,39.95216179,39.9523527,39.95239043,39.95242305,39.95242487,

        39.95240718,39.95240224,39.95232515,39.95229446,39.95233105]

liblon=[-75.19346528,-75.19354176,-75.19354936,-75.19355259,-75.19359157,
        -75.19357628,-75.19354717,-75.19348767,-75.19339785,-75.1933452,
        -75.19332773,-75.19327585,-75.19323223,-75.19326823,-75.19337275,
        -75.19339994,-75.1934048,-75.19332293,-75.1932546,-75.19316639,
        -75.1934005,-75.19345689,-75.19352098,-75.19343724,-75.19331876,
        -75.19327910,-75.19335194,-75.19333902,-75.19334085]
```

```
libsignal=[-41,-55,-61,-61,-67,-68,-63,-58,-60,-70,-75,-76,-76,-70,-70,
        -82,-83,-78,-80,-85,-69,-62,-52,-61,-67,-69,-72,-66,-77]

# the original data of lattitude and longitude (in degree) for the Starbucks
   WAP 94:b4:0f:53:07:00
staroriginsignal = -43
starorigin = [39.95322759,-75.19217689]
starlat =
   [39.95300328,39.95314201,39.95308746,39.95322759,39.95331679,39.95341871,
        39.95341624,39.95336173,39.95333451,39.95329762,39.95323437,39.95316914,

        39.95305603,39.9530076,39.95296533,39.9529472,39.95293145,39.95293909,
        39.95294534,39.95302457,39.95307636,39.95329533,39.95334517,39.953197,
        39.95318632,39.95323969,39.95331454,39.95331931,39.95334841,39.95335327,

        39.95333676,39.9531888,39.95318109,39.95315349,39.95332706,39.95334124,

        39.9533159,39.95329751,39.95325922]
starlon = [-75.19218521,-75.19215558,-75.19214409,-75.19217689,-75.19219967,
        -75.19223796,-75.19221528,-75.19219009,-75.19220592,-75.19221843,
        -75.19222063,-75.19222198,-75.19222518,-75.19222164,-75.19222223,
        -75.19219909,-75.19221608,-75.19220119,-75.19199002,-75.19194761,
        -75.1918776,-75.19169873,-75.19191655,-75.191923,-75.19193303,
        -75.19191041,-75.19192654,-75.19193617,-75.19195773,-75.19196058,
        -75.19197971,-75.19204021,-75.19197923,-75.19183212,-75.19149968,
        -75.1914769,-75.19142933,-75.19145057,-75.19149104]
starsignal = [-67,-54,-54,-43,-68,-77,-69,-64,-67,-65,-55,-54,-56,-69,-68,
        -70,-68,-70,-70,-72,-68,-68,-73,-59,-59,-61,-65,-68,-67,-67,-71,-59,
        -70,-71,-73,-77,-75,-76,-76]


# the original data of lattitude and longitude (in degree) for the united by
   blue  access point ac:86:74:57:d6:82
unitedoriginsignal = -41
unitedorigin = [39.95303359,-75.1931862]
unitedlat =
   [39.9531262,39.95306805,39.95303359,39.95300903,39.95308204,39.95307472,
        39.95305298,39.95304666,39.95309792,39.95311104,39.95314332,39.95315069,

        39.95318367,39.95316839,39.9529689,39.95295116,39.95298173,39.95299407,

        39.95297741,39.9530528,39.95305761,39.95311303,39.95304962,39.95300972,

        39.95293554,39.95287939]
unitedlon = [-75.19316856,-75.19322581,-75.1931862,-75.19329689,-75.19333511,
        -75.19350965,-75.19355542,-75.19358466,-75.19371891,-75.19376055,
        -75.19378227,-75.19381858,-75.1938578,-75.19387127,-75.19327191,
        -75.1932809,-75.19337015,-75.19344377,-75.19354724,-75.19362915,
        -75.19366132,-75.19377652,-75.19363119,-75.19348948,-75.19321491,
        -75.19300501]
unitedsignal = [-60,-61,-41,-59,-62,-64,-77,-69,-79,-75,-81,-82,-82,-82,
        -74,-70,-66,-64,-76,-67,-76,-74,-71,-71,-69,-79]
```

```
# the original data of lattitude and longitude (in degree) for the bluemercury
     access point e2:55:7d:46:ba:70
blueoriginsignal = -42
blueorigin = [39.95348435,-75.1954269]
bluelat =
    [39.95344475,39.95346047,39.95351389,39.9535116,39.95351465,39.95348435,
         39.95350455,39.95353418,39.95356256,39.95353174,39.95344449,39.95345135,

         39.95342285,39.95340621,39.95338485,39.9533588,39.95336648,39.95336993,

         39.95327412,39.95328151,39.95328667,39.95329931,39.95333245,39.95343911,

         39.95343569,39.95322091,39.95330729,39.95339435,39.95330516]
bluelon = [-75.1953241,-75.19534486,-75.19536677,-75.19539087,-75.19540162,
         -75.1954269,-75.19552856,-75.19554469,-75.1956038,-75.19586533,
         -75.19582356,-75.19575759,-75.19569923,-75.19565746,-75.19551169,
         -75.19546526,-75.19540467,-75.19533335,-75.1955647,-75.19562271,
         -75.19567539,-75.19576064,-75.19582484,-75.19562141,-75.19557638,
         -75.19541343,-75.19556592,-75.19585788,-75.19589166]
bluesignal = [-67,-68,-62,-62,-56,-42,-61,-61,-68,-69,-68,-69,-57,-69,-60,-60,
         -72,-72,-65,-65,-76,-71,-70,-69,-69,-66,-66,-70,-71]

# the original data of lattitude and longitude (in degree) for the Penne-bar
    access point e8:40:40:80:52:e8
penneoriginsignal = -43
penneorigin = [39.95339425,-75.1960868]
pennelat =
    [39.95337588,39.95337808,39.95334292,39.95331375,39.95329611,39.953289,
         39.95329387,39.95333236,39.95327273,39.95336876,39.95344523,39.95342261,

         39.9534374,39.9534569,39.95350044,39.95345781,39.9534526,39.95336277,
         39.95336332,39.95340967,39.95338347,39.95337122,39.95338304,39.95338875,

         39.95339425,39.95339517,39.95340651,39.95343159,39.95349377]
pennelon = [-75.19609743,-75.19618277,-75.19610598,-75.19602633,-75.19600751,
         -75.19598095,-75.1959173,-75.19589952,-75.19608164,-75.19570731,-75.19592838,

         -75.19593443,-75.19599858,-75.19608773,-75.19651072,-75.19636381,-75.19627185,

         -75.19591907,-75.19594091,-75.19603491,-75.19618058,-75.19617366,-75.19612399,

         -75.19603628,-75.1960868,-75.19612735,-75.19616441,-75.19620826,-75.19634645]

pennesignal =
    [-70,-77,-72,-80,-72,-76,-70,-70,-80,-70,-64,-82,-71,-72,-82,-78,-69,
         -65,-64,-64,-71,-70,-71,-63,-43,-66,-66,-76,-79]
#==============================================================================
# helper functions
#==============================================================================

def discalc(lat,lon,origin):
    """Calculate the distance (unit = meter) of all locations from the origin
       of one WAP"""
```

```
    # calculation following online sourse:http://www.movable-type.co.uk/
        scripts/latlong.html
    # using the Harversine formula
    R = 6373  #unit = km the radius of earth, reference: http://andrew.hedges.
        name/experiments/haversine/
    # create a list to store all the distances
    Distance = []
    for idx in range(len(lat)):
        #convert degree to radian using numpy
        deltalat = np.radians(lat[idx]-origin[0])
        deltalon = np.radians(lon[idx]-origin[1])
        rlat1 = np.radians(origin[0])
        rlat2 = np.radians(lat[idx])
        a = (math.sin(deltalat/2))**2 + math.cos(rlat1)*math.cos(rlat2)*(math.
            sin(deltalon/2))**2
        c = 2*math.atan2(math.sqrt(a),math.sqrt(1-a))
        d = R* c * 1000
        Distance.append(d)

    return Distance

#-------------------------------------------------------------------------------
#unit converter
def dbmtomw(signal):
    """Convert dBm to mW"""
    return 10.0**(signal/10.0)

def mwtodbm(power):
    """Convert mW to dBm"""
    return 10.0*math.log10(power/1.0)
#-------------------------------------------------------------------------------
#calculate Rpower (the "ideal power") in free space
def Rpower(dis,originsignal):
    """Calculate the ideal power for each access point and store
    them into a list"""
    rpower = []
    for r in dis:
        if r == 0:
            rpower.append(originsignal)
        else:
            result = dbmtomw(float(originsignal)) * (1.0)/(1.0*r**2)
            result = mwtodbm(float(result))
            rpower.append(result)
    return rpower
#-------------------------------------------------------------------------------


#===============================================================================
#Plot locations direct onto google map as for better visualization
#===============================================================================

#reference:https://github.com/vgm64/gmplot
#-------------------------------------------------------------------------------
# gmap = gmplot.GoogleMapPlotter(39.95211, -75.19368, 17)
```

```
# # gmap.plot(starlat, starlon, 'cornflowerblue', edge_width=10)
# gmap.scatter(liblat, liblon, '#3B0B39', size=5, marker=False)
# gmap.scatter(liblat, liblon, 'r', marker=True)
# gmap.draw("lib.html")

def gdraw(lat,lon,origin,name):
    """Draw all the measured location as scatters (based on
    their lat and long) onto google map and output the
    result as a html file"""
    gmap = gmplot.GoogleMapPlotter(origin[0], origin[1], 17)
    gmap.scatter(lat, lon, '#3B0B39', size=2, marker=False)
    # gmap.scatter(lat, lon, 'r', marker=True)
    gmap.draw(name)

#==============================================================================
#Plot Heatmap
#==============================================================================

def plotheatmap(lat,lon,origin,signal,name):
    """Plot the heap map for each WAP"""
    fig, ax = plt.subplots(1)
    maxs =1.000* max(signal)
    # color = [str(maxs/(1.00*item)) for item in signal]
    color = [(1.00*item)/(100.0) for item in signal]
    plt.scatter(lon,lat,s = 400,marker = '8', c = color,alpha=0.5)

    ax.scatter(origin[1],origin[0],marker = 'd',color = 'g', s = 60,alpha = 1,
       label = 'Wifi␣Access␣Point')
    ax.legend(scatterpoints = 1)

    # plt.gray()
    cbar = plt.colorbar()
    plt.ylim(min(lat)-1E-4,max(lat)+1E-4)
    plt.xlim(min(lon)-1E-4,max(lon)+1E-4)
    for i, txt in enumerate(signal):
        ax.annotate(txt,(lon[i],lat[i]),fontsize = 10)
    xlabel = r'Longitude'
    ylabel = r'Latitude'
    plt.xlabel(xlabel)
    plt.ylabel(ylabel)
    plt.locator_params(axis='y', nticks=3)
    plt.locator_params(axis='x', nticks=8)
    filename = name + '.png'
    titlename = name + '␣Wifi␣heatmap'
    plt.title(titlename)
    fig.savefig(filename,dpi=300)

#==============================================================================
#Plot Power VS Distance
#==============================================================================
def plotpower(dis,Rpower,signal,name):
    """Plot Power VS Distance as to compare the measured average bulk
       propagation loss
```

```
    with the ideal propagation loss for a free space"""

    fig2 = plt.figure()
    plt.plot(dis,Rpower,'^',color='r',alpha = 0.5, label = r'$Ideal\␣signal\␣\
       backsim$␣1/r^2')
    plt.plot(dis,signal,'o',color='g', alpha = 0.5, label = r'$Measured\␣
       signal$')
    ylabel = 'Signal␣Power;␣unit␣=␣dBm'
    plt.ylabel(ylabel)
    xlabel = 'distance␣from␣origin;␣unit␣=␣meters'
    plt.xlabel(xlabel)
    titlename = name + "'s␣Wifi␣Power␣VS␣distance␣from␣WAP␣origin"
    filename = name +"power.png"
    plt.legend()
    plt.title(titlename)
    fig2.savefig(filename, dpi=300, bbox_inches='tight')

#==============================================================================
#Plot Delta Power VS Distance
#==============================================================================
def deltaplotpower(dis,Rpower,signal,originsignal,name):
    """Plot Power VS Distance as to compare the measured average bulk
       propagation loss
    with the ideal propagation loss for a free space"""
    for i in range(len(dis)):
        Rpower[i] = originsignal - Rpower[i]
        signal[i] = originsignal - signal[i]

    fig2 = plt.figure()
    plt.plot(dis,Rpower,'^',color='r',alpha = 0.5, label = r'$Ideal\␣signal\␣\
       backsim$␣1/r^2')
    plt.plot(dis,signal,'o',color='g', alpha = 0.5, label = r'$Measured\␣
       signal$')
    ylabel = 'Propagation␣Loss;␣unit␣=␣dBm'
    plt.ylabel(ylabel)
    xlabel = 'distance␣from␣origin;␣unit␣=␣meters'
    plt.xlabel(xlabel)
    titlename = name + "'s␣Porpapgation␣loss␣VS␣distance␣(r)"
    filename = name +"deltapower.png"
    plt.legend(loc = 4)
    plt.title(titlename)
    fig2.savefig(filename, dpi=300, bbox_inches='tight')

#==============================================================================
# calculate k value the average propagation loss
#==============================================================================
# Assume noise power to be C  = -100dBm
def avepropaloss(signal,dis,Rpower,name):
    # calculation follows Piazza post @64
    C = -100
    K = []
    R = []
    ronly = []
    konly = []
```

```
    for i in range(len(signal)):
        snr = 10*math.log10(dbmtomw(signal[i])/dbmtomw(C)) #unit = dB
        rsnr = 10*math.log10(dbmtomw(Rpower[i])/dbmtomw(C)) #unit = dB
        if dis[i] !=0:
            r = rsnr*dis[i]**2
            k = snr*dis[i]**2
            lst = [k,dis[i],r]
            K.append(lst)
            ronly.append(r)
            konly.append(k)
    ave = sum(konly)/float(len(konly))
    rave = sum(ronly)/float(len(ronly))

    """out put the average propagation loss"""
    outfile = open(name,'w')
    outfile.write("Average␣propagation␣loss␣is␣"+ str(ave) +"␣dB*m^2\n")
    outfile.write("Average␣propagation␣loss␣according␣to␣1/r^2␣is␣" + str(rave
       ) +"␣dB*m^2\n")
    outfile.write("For␣each␣location,␣the␣propagation␣loss␣k␣and␣the␣1/r^2␣
       prediction␣k_r␣is␣shown␣below␣along␣with␣the␣distance␣to␣the␣origin␣
       WAP\n")

    for data in K:
        content = "k␣=␣"+ str(round(data[0])) +"␣dB*m^2;\t"+ "k_r␣=␣"+str(
           round(data[2])) +"␣dB*m^2;\t"+"dis␣=␣"+ str(data[1])+ "␣m\n"
        outfile.write(content)
    outfile.close()

    return K

#==============================================================================
# write the outputdata into a text file
#==============================================================================
def outputdata(lat,lon,signal,name,bssid,dis,rpower):
    """write the output data into a text file"""
    outfile = open(name,'w')
    d = []
    for i in range(len(lat)):
        t = (lat[i],lon[i],signal[i],dis[i],rpower[i])
        d.append(t)
    outfile.write("BSSID␣" + "␣␣\t\t" + "Latitude" + "␣\t" + "longitude" + "␣\
       t" + "Power" + "␣\t" +"Dis"+"␣\t"+"Rpower" + "\n")

    for data in d:
        content = bssid + "\t" + str(data[0])+ "\t"+ str(data[1]) +"\t" + str(
           data[2]) +"\t" + str(round(data[3],2)) + "\t"+str(round(data[4],2)
           )+"\n"
        outfile.write(content)

    outfile.close()



#------------------------------------------------------------------------------
```

```python
def main():

    # location visualization
    # locate each location onto googlemap one by one as html file
    gdraw(liblat,liblon,liborigin,"library.html")
    gdraw(starlat,starlon,starorigin,"starbucks.html")
    gdraw(unitedlat,unitedlon,unitedorigin,"united.html")
    gdraw(bluelat,bluelon,blueorigin,"bluemercury.html")
    gdraw(pennelat,pennelon,penneorigin,"penne.html")

    # locate all locations onto googlemap as one html file
    gmap = gmplot.GoogleMapPlotter(liborigin[0], liborigin[1], 17)
    gmap.scatter(liblat, liblon, '#3B0B39', size=2, marker=False)
    gmap.scatter(starlat,starlon,'g',size = 2 ,marker = False)
    gmap.scatter(bluelat,bluelon,'b',size = 2 ,marker = False)
    gmap.scatter(unitedlat,unitedlon,'r',size = 2 ,marker = False)
    gmap.scatter(pennelat,pennelon,'m',size = 2 ,marker = False)
    gmap.draw("all.html")

    # plot the heatmap
    plotheatmap(liblat,liblon,liborigin,libsignal,"library")
    plotheatmap(starlat,starlon,starorigin,starsignal,"starbucks")
    plotheatmap(unitedlat,unitedlon,unitedorigin,unitedsignal,"unitedblue")
    plotheatmap(bluelat,bluelon,blueorigin,bluesignal,"bluemercury")
    plotheatmap(pennelat,pennelon,penneorigin,pennesignal,"penne")

    # calculate the distance:
    libdis = discalc(liblat,liblon,liborigin)
    stardis = discalc(starlat,starlon,starorigin)
    unitedis = discalc(unitedlat,unitedlon,unitedorigin)
    bluedis = discalc(bluelat,bluelon,blueorigin)
    pennedis = discalc(pennelat,pennelon,penneorigin)

    # # calculate Rpower
    Rlib = Rpower(libdis,liboriginsignal)
    Rstar = Rpower(stardis,staroriginsignal)
    Runited = Rpower(unitedis,unitedoriginsignal)
    Rblue = Rpower(bluedis,blueoriginsignal)
    Rpenne = Rpower(pennedis,penneoriginsignal)

    #write the data for phase 2 in to a text file
    outputdata(liblat,liblon,libsignal,"library.txt","84:d4:7e:dc:2a:40",
       libdis,Rlib)
    outputdata(starlat,starlon,starsignal,"starbucks.txt","94:b4:0f:53:07:00",
       stardis,Rstar)
    outputdata(unitedlat,unitedlon,unitedsignal,"unitedblue.txt","ac:86:74:57:
       d6:82",unitedis,Runited)
    outputdata(bluelat,bluelon,bluesignal,"bluemercury.txt","e2:55:7d:46:ba:70
       ",bluedis,Rblue)
    outputdata(pennelat,pennelon,pennesignal,"penne.txt","e8:40:40:80:52:e8",
       pennedis,Rpenne)

    #Plot distance vs signal strength
    plotpower(libdis,Rlib,libsignal,"Library")
```

```
    plotpower(stardis,Rstar,starsignal,"Starbucks")
    plotpower(unitedis,Runited,unitedsignal,"Unitedblue")
    plotpower(bluedis,Rblue,bluesignal,"Bluemercury")
    plotpower(pennedis,Rpenne,pennesignal,"Penne")

    #plot the distance vs propagation loss
    deltaplotpower(libdis,Rlib,libsignal,liboriginsignal,"Library")
    deltaplotpower(stardis,Rstar,starsignal,staroriginsignal,"Starbucks")
    deltaplotpower(unitedis,Runited,unitedsignal,unitedoriginsignal,"
       Unitedblue")
    deltaplotpower(bluedis,Rblue,bluesignal,blueoriginsignal,"Bluemercury")
    deltaplotpower(pennedis,Rpenne,pennesignal,penneoriginsignal,"Penne")

    #calculate k
    avepropaloss(libsignal,libdis,Rlib,'libaveloss.txt')
    avepropaloss(starsignal,stardis,Rstar,'staraveloss.txt')
    avepropaloss(unitedsignal,unitedis,Runited,'unitedaveloss.txt')
    avepropaloss(bluesignal,bluedis,Rblue,'blueaveloss.txt')
    avepropaloss(pennesignal,pennedis,Rpenne,'penneaveloss.txt')


main()
```