Data Science Clutter Project
Lab Notebook

Mark McFarland, P.E.*
Institute for Telecommunication Sciences
Boulder, CO

March 30, 2021

Abstract

This document is my lab notebook, containing my analysis for the project.

Contents

*Contact: mark@its.bldrdoc.gov, 303/497-4132

1 Data Overview

I examine the measurement data located in the project folder.

1.1 Phoenix Data

Summarize the Arizona data. There were Phoneix area measurements and "North Lot to Downtown" measurements.

1. Look at the data files we have

```
tree -fi data
```

```
## data
## data/AZ_LincolnHeight_Dwntwn_Run1.csv
## data/AZ_LincolnHeight_Dwntwn_Run1.txt
## data/AZ_LincolnHeight_Dwntwn_Run2.csv
## data/AZ_LincolnHeight_Dwntwn_Run2.txt
## data/AZ_LincolnHeight_Suburb_Run1.csv
## data/AZ_LincolnHeight_Suburb_Run1.txt
## data/AZ_LincolnHeight_Suburb_Run2.csv
## data/AZ_LincolnHeight_Suburb_Run2.txt
## data/AZ_LincolnHeight_Suburb_Run3.csv
## data/AZ_LincolnHeight_Suburb_Run3.txt
## data/AZ_OpenArms_Pima.csv
## data/AZ_OpenArms_Pima.txt
## data/AZ_OpenArms_Sub_Run1.csv
## data/AZ_OpenArms_Sub_Run1.txt
## data/AZ_OpenArms_Sub_Run2.csv
## data/AZ_OpenArms_Sub_Run2.txt
## data/AZ_Phoenix_North_Dwntwn.csv
## data/AZ_Phoenix_North_Dwntwn.txt
## data/AZ_Phoenix_North_North.csv
## data/AZ_Phoenix_North_North.txt
## data/AZ_Phoenix_South_Dwntwn.csv
## data/AZ_Phoenix_South_Dwntwn.txt
## data/AZ_Phoenix_South_South.csv
## data/AZ_Phoenix_South_South.txt
## data/NorthLot_to_Downtown
## data/NorthLot_to_Downtown/~$NorthLot_to_DowntownPhoenix.pptx
## data/NorthLot_to_Downtown/~$rthLot_to_Dwntwn_ElapsedTimeAreaDescriptions.docx
## data/NorthLot_to_Downtown/BasicTransmissionGainPlots.png
```

```
## data/NorthLot_to_Downtown/NorthLot_to_DowntownPhoenix.pptx
## data/NorthLot_to_Downtown/NorthLot_to_DowntownPhoenix.qgz
## data/NorthLot_to_Downtown/NorthLot_to_Dwntwn_ElapsedTimeAreaDescriptions.docx
## data/NorthLot_to_Downtown/NorthLottoDwntwnPhoenix.csv
## data/NorthLot_to_Downtown/NorthLotXmitInfo.csv
##
## 1 directory, 32 files
```

14 CSV files (or acquisitions), some TXT.

2. Number of observations in each CSV:

```
wc -l data/*.csv
```

```
##    8100 data/AZ_LincolnHeight_Dwntwn_Run1.csv
##    7991 data/AZ_LincolnHeight_Dwntwn_Run2.csv
##    5541 data/AZ_LincolnHeight_Suburb_Run1.csv
##    1848 data/AZ_LincolnHeight_Suburb_Run2.csv
##    1849 data/AZ_LincolnHeight_Suburb_Run3.csv
##    7679 data/AZ_OpenArms_Pima.csv
##    5662 data/AZ_OpenArms_Sub_Run1.csv
##    1982 data/AZ_OpenArms_Sub_Run2.csv
##    6188 data/AZ_Phoenix_North_Dwntwn.csv
##    6255 data/AZ_Phoenix_North_North.csv
##    6684 data/AZ_Phoenix_South_Dwntwn.csv
##    5804 data/AZ_Phoenix_South_South.csv
##   65583 total
```

From about 2K–8K observations in each run.

3. Examine one of the TXT files

```
cat data/AZ_LincolnHeight_Dwntwn_Run1.txt
```

```
## Phoenix AZ,N/A
## Date_Time,N/A 0:00
## TX_Lat,33.537482
## TX_Long,-112.034833
## TX_Power,0
## TX_Height,18.3
## RX_Height,3
## f__mhz,1756
```

The remaining TXT files have similar information.

4. Read in data in CSV files

   Note that I exclude the data file NorthLottoDwntwnPhoenix.csv.

5. Dimensions of data:

```
## [1] 65571     7
```

6. Number of observations per route, run, & location:

```
## $city
## city
## LincolnHeight      OpenArms      PhoenixN
##         25324         15320         12441
##      PhoenixS
##         12486
##
## $region
## region
## downtown   suburb     Pima    north    south
##    28959    16877     7678     6254     5803
##
## $run
## run
##     1     2     3
## 51905 11818  1848
##
## $`city:region`
##                region
## city            downtown suburb Pima north south
##   LincolnHeight    16089   9235    0     0     0
##   OpenArms             0   7642 7678     0     0
##   PhoenixN          6187      0    0  6254     0
##   PhoenixS          6683      0    0     0  5803
##
## $`city:run`
##                run
## city                1    2    3
##   LincolnHeight 13639 9837 1848
##   OpenArms      13339 1981    0
##   PhoenixN      12441    0    0
##   PhoenixS      12486    0    0
##
```

```
## $`region:run`
##           run
## region         1    2    3
##   downtown 20969 7990    0
##   suburb   11201 3828 1848
##   Pima      7678    0    0
##   north     6254    0    0
##   south     5803    0    0
##
## $`city:region:run`
## , , run = 1
##
##                region
## city            downtown suburb Pima north south
##   LincolnHeight     8099   5540    0     0     0
##   OpenArms             0   5661 7678     0     0
##   PhoenixN          6187      0    0  6254     0
##   PhoenixS          6683      0    0     0  5803
##
## , , run = 2
##
##                region
## city            downtown suburb Pima north south
##   LincolnHeight     7990   1847    0     0     0
##   OpenArms             0   1981    0     0     0
##   PhoenixN             0      0    0     0     0
##   PhoenixS             0      0    0     0     0
##
## , , run = 3
##
##                region
## city            downtown suburb Pima north south
##   LincolnHeight        0   1848    0     0     0
##   OpenArms             0      0    0     0     0
##   PhoenixN             0      0    0     0     0
##   PhoenixS             0      0    0     0     0
```

7. Summarize data:

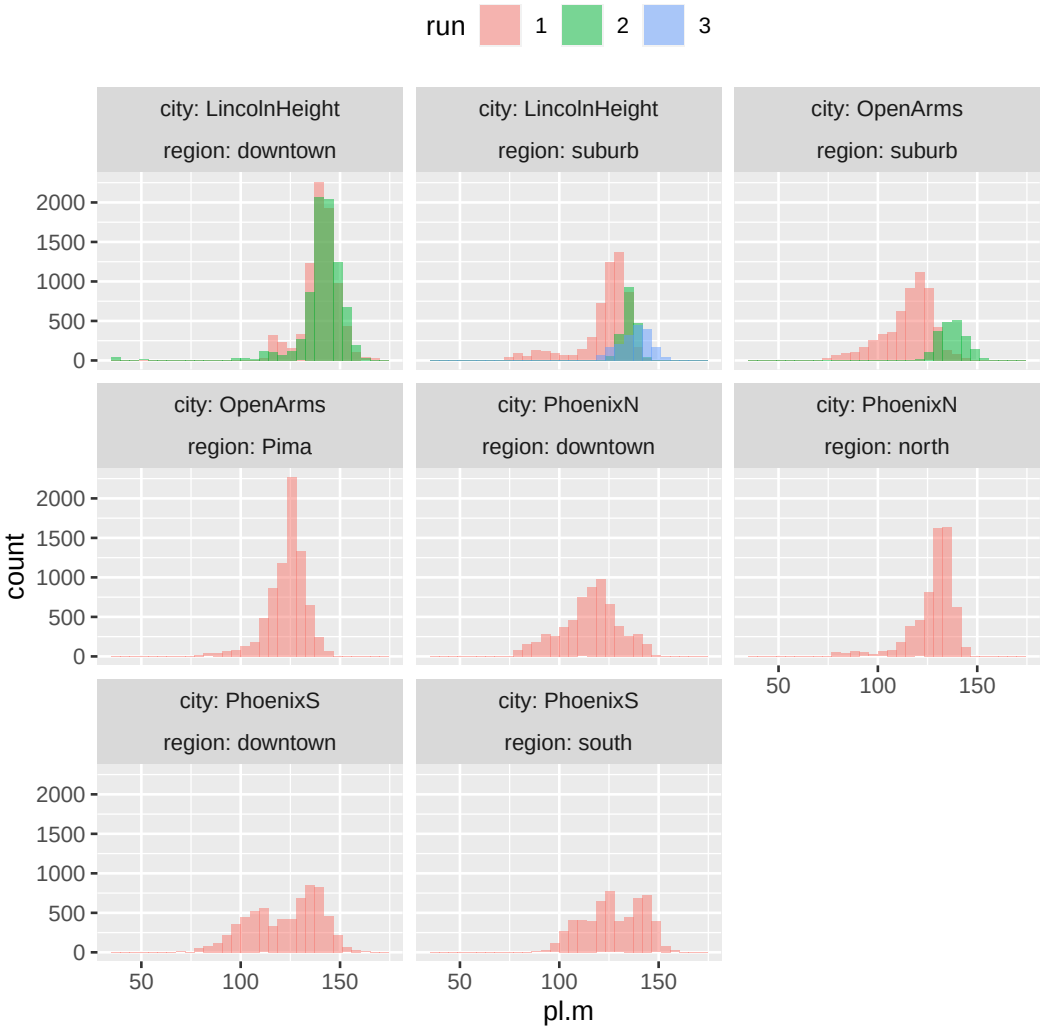
```
##       time           lat             lon
##  Min.   :   0   Min.   :33.43   Min.   :-112.1
```

```
##  1st Qu.:1318   1st Qu.:33.45   1st Qu.:-112.1
##  Median :2896   Median :33.47   Median :-112.1
##  Mean   :3065   Mean   :33.48   Mean   :-112.0
##  3rd Qu.:4629   3rd Qu.:33.51   3rd Qu.:-112.0
##  Max.   :7999   Max.   :33.55   Max.   :-111.8
##       pl.m                  city
##  Min.   : 35.45   LincolnHeight:25324
##  1st Qu.:119.51   OpenArms     :15320
##  Median :130.14   PhoenixN     :12441
##  Mean   :127.96   PhoenixS     :12486
##  3rd Qu.:139.08
##  Max.   :170.39
##       region      run
##  downtown:28959   1:51905
##  suburb  :16877   2:11818
##  Pima    : 7678   3: 1848
##  north   : 6254
##  south   : 5803
##
```

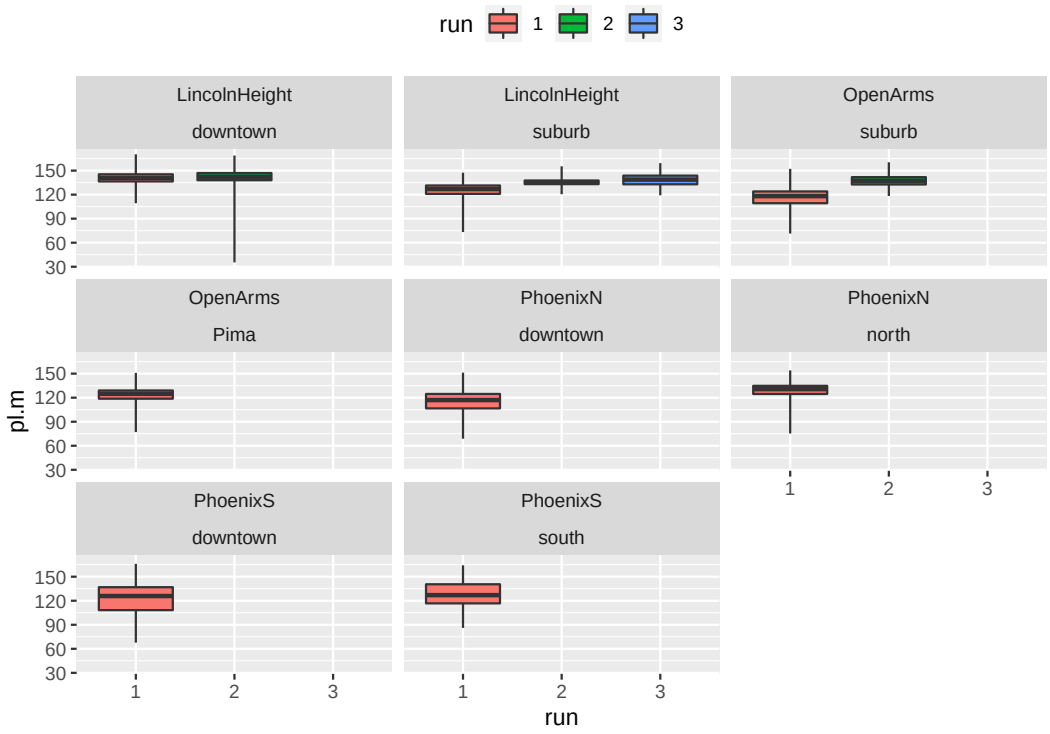
The variable `pl.m` is the measured path loss. Not sure if *sub* region and *suburb* region refer to the same region... both suburb? I'll assume so.

Why were repeated runs made?

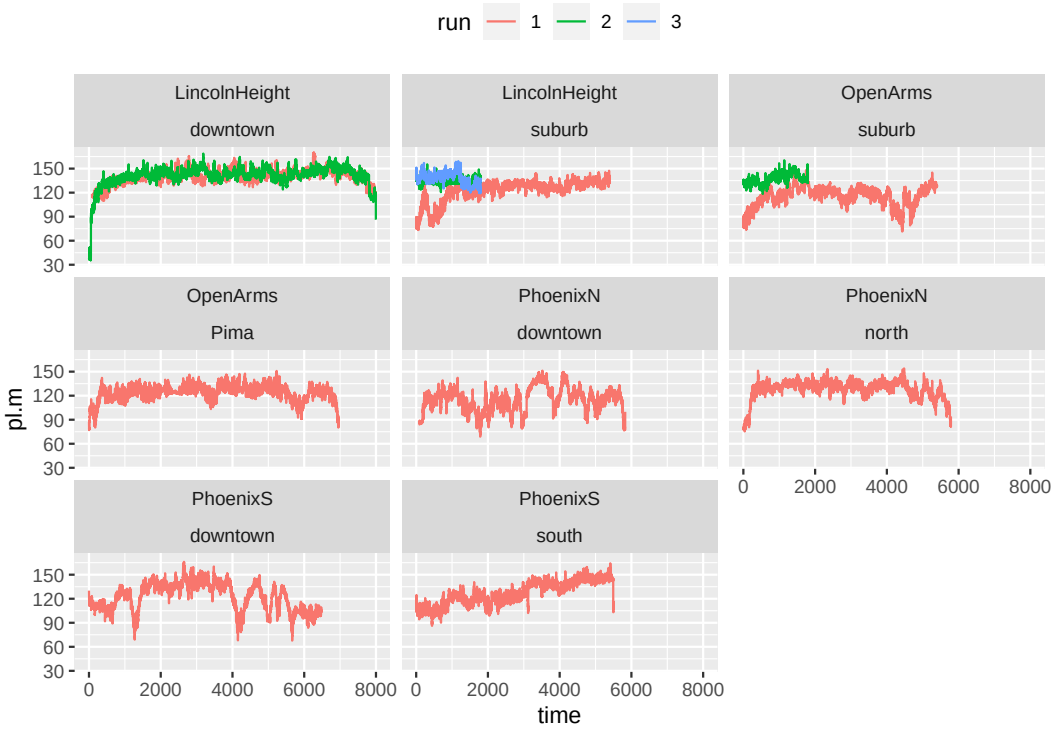
8. View unstacked histograms of these data:



9. View boxplots:

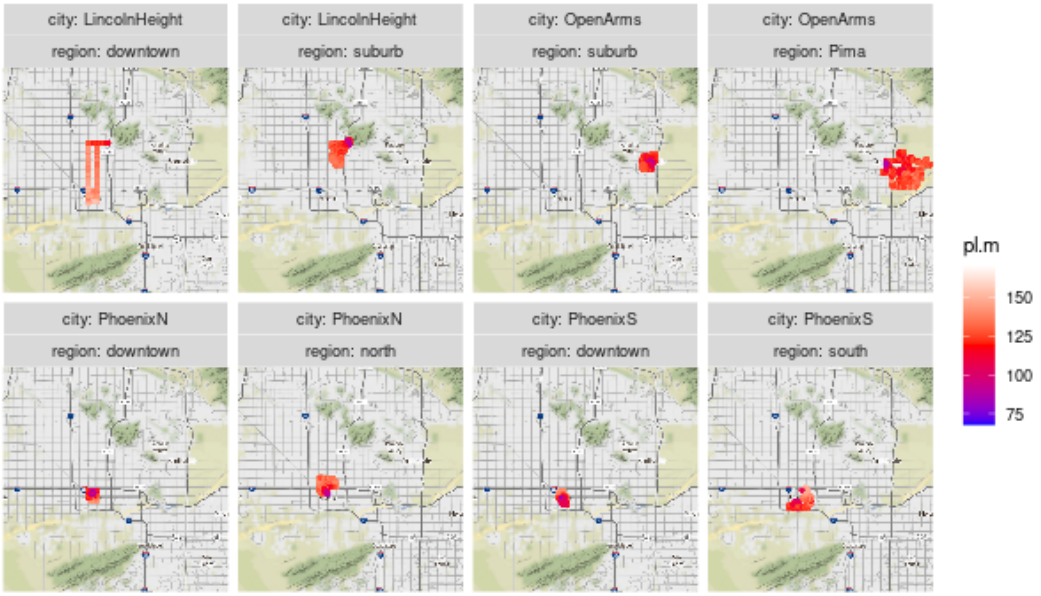


10. Examine time series plots

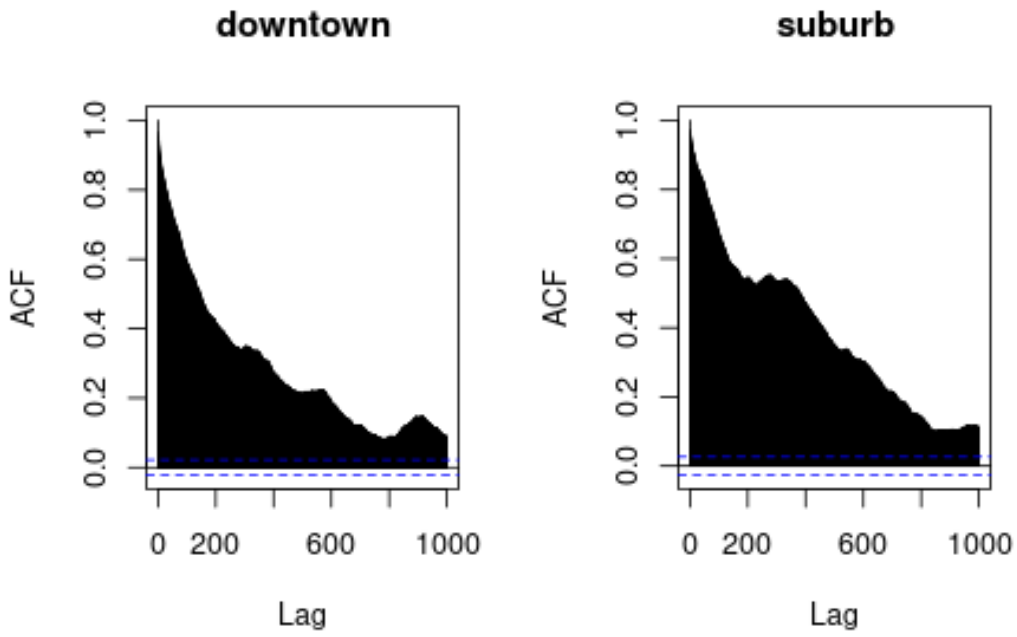


Not sure what the purpose of runs 2 & 3 are. Moving forward, I'll just include run 1 in my analysis.

11. Plot measurement data from Phoenix



12. Examine serial correlation via ACF plots for Lincoln Heights, downtown and suburban regions (run one only):



There is extensive serial correlation in the data. This is likely due to two main reasons:

(a) Oversampling while collecting data (i.e. machine collecting more observations than necessary). That is, too many observations were collected in one location.

   i. Just because a measurement device can acquire, say, 1M observations per second, doesn't mean one *should* collect 1M observations per second. More data doesn't mean more information.

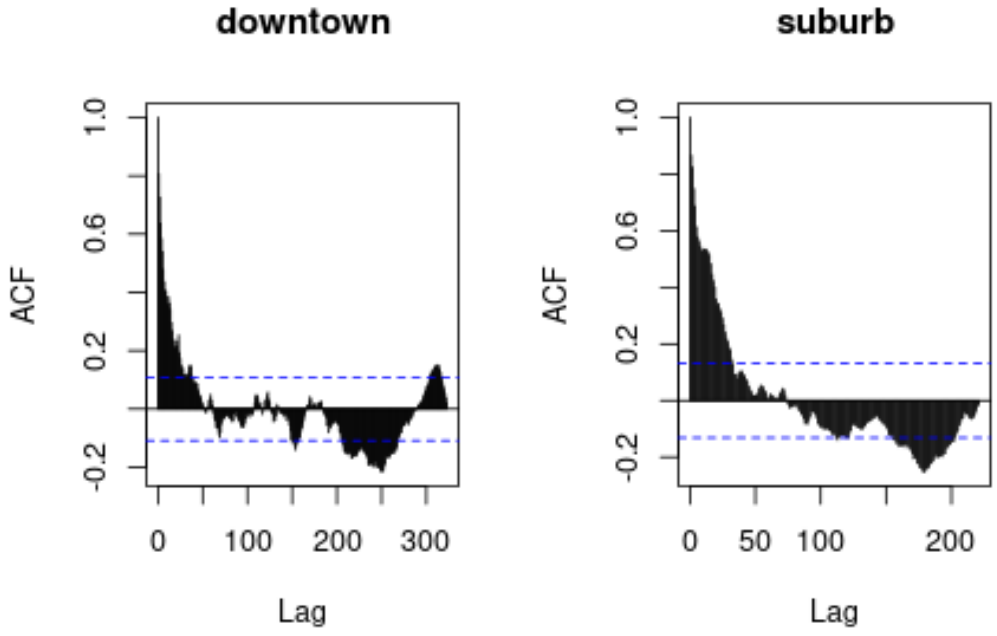
   ii. But this in itself isn't a problem. It can easily be addressed by downsampling the raw data. This is done by selecting every second, third, fifth, tenth, 25th, etc. observation. Ideally, the raw data observations should be independent of each other (not autocorrelated) before computing the slow fading characteristics.

(b) "Smoothing" (or computing a running average of) already oversampled observations to obtain slow-fading characteristics.

   i. Smoothing or averaging should be performed using uncorrelated

or independent observations. Otherwise, the dependency among observations becomes even more inveterate.

13. Address serial correlation. Downsample, taking every 100th observation and plotting the ACF:



Definitely an improvement. Suburban data could be downsampled even further.

1.2 Grand Junction

1. Read in Grand Junction - Grand Mesa data, four runs

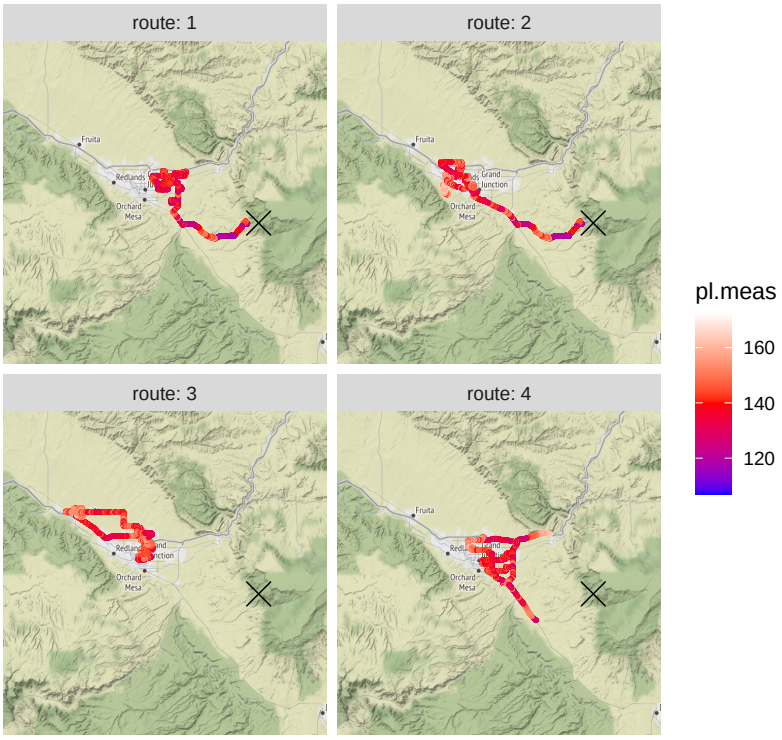
```
##    time      lat       lon  pl.meas   pl.itm     fspl
## 1 0.000 38.99077 -108.2708 109.8705 105.8411 105.8416
## 2 1.713 38.99077 -108.2708 107.0105 105.8411 105.8416
## 3 2.703 38.99077 -108.2708 108.7295 105.8411 105.8416
## 4 3.688 38.99076 -108.2708 107.7625 105.8420 105.8425
## 5 4.687 38.99076 -108.2708 108.1225 105.8444 105.8449
## 6 5.670 38.99074 -108.2709 108.5965 105.8500 105.8505
##       dist      angle route   location
## 1 2.638524 -0.1393199     1 Grand Mesa
## 2 2.638524 -0.1393199     1 Grand Mesa
## 3 2.638524 -0.1393199     1 Grand Mesa
## 4 2.638794 -0.1393057     1 Grand Mesa
## 5 2.639517 -0.1392675     1 Grand Mesa
```

```
## 6 2.641240 -0.1391766     1 Grand Mesa
```

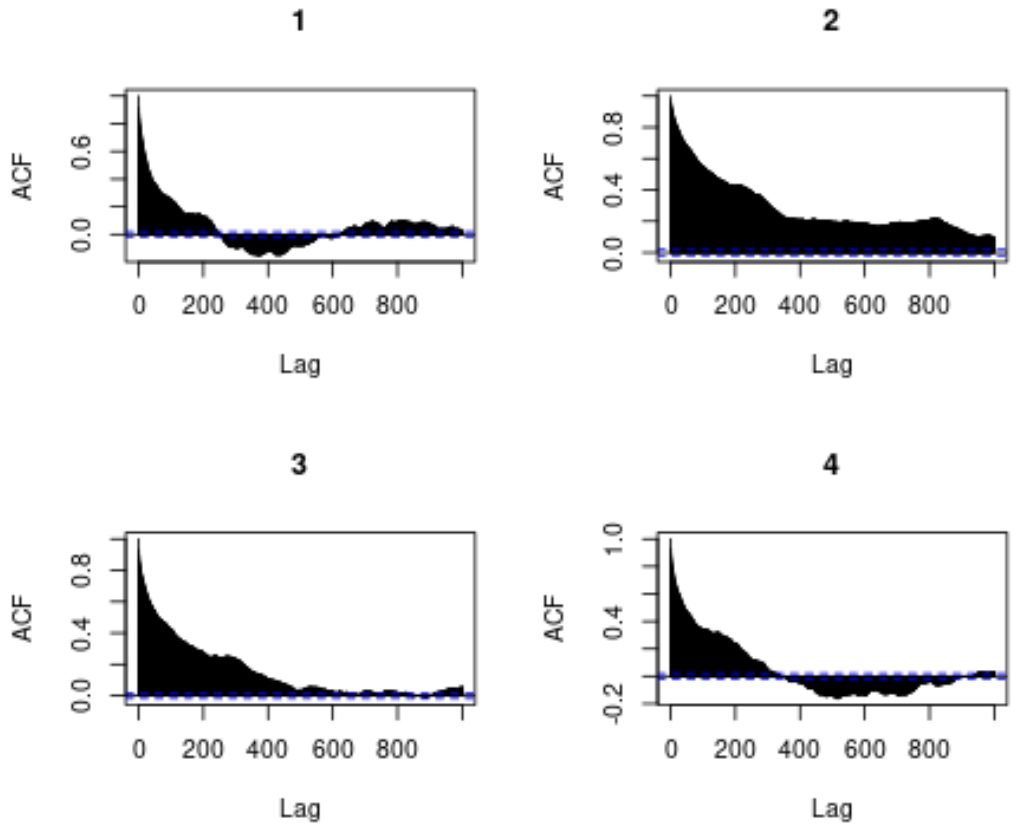
Dimensions of data:

```
## [1] 39416    10
```

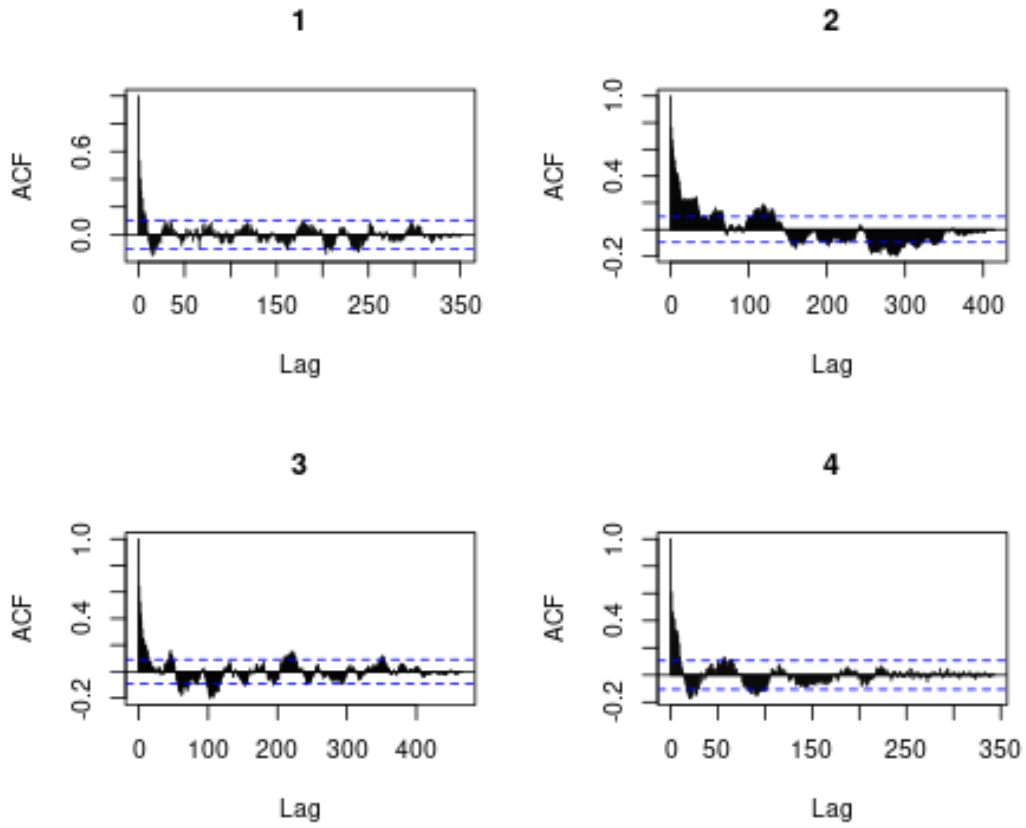
2. Plot measurement data from Grand Junction - Grand Mesa, four routes. Transmitter is marked with an *X*.



3. Examine serial correlation via ACF plots for Grand Junction data, routes 1–4:



4. Address serial correlation. Downsample, taking every 25th observation and plotting the ACF:



Again, a big improvement—this time by taking every 25th observation. Still some autocorrelation. Each route and/or segment of each route should be further uncorrelated.

1.3 Salt Lake City

1. Read in Salt Lake City data, three runs

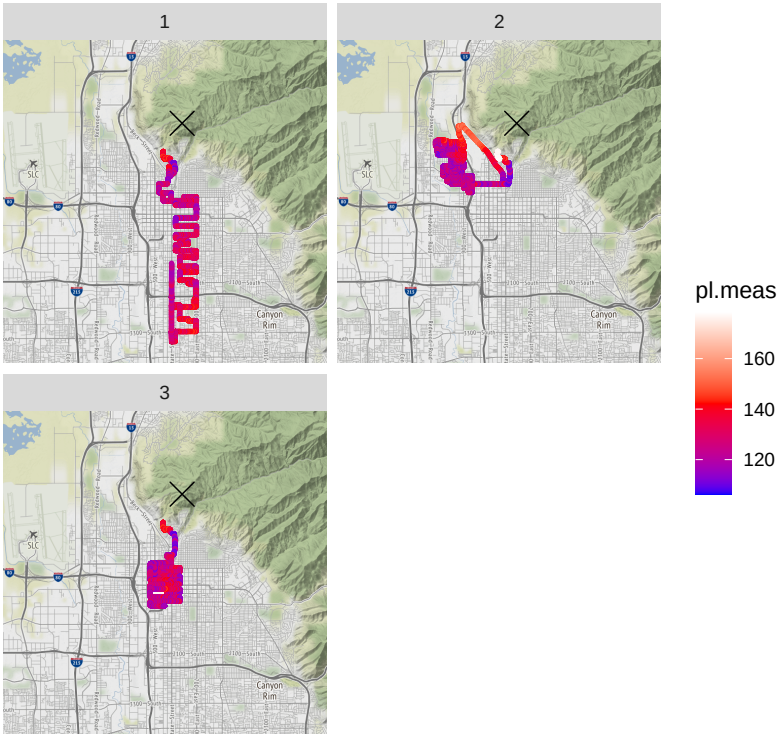
```
##    time      lat       lon  pl.meas   pl.itm     fspl
## 1 0.000 40.79243 -111.8938 136.1921 147.6119 103.2853
## 2 1.599 40.79243 -111.8938 135.7842 147.6030 103.2861
## 3 2.555 40.79243 -111.8938 135.6170 147.6030 103.2861
## 4 3.486 40.79243 -111.8938 135.6504 147.6030 103.2861
## 5 4.444 40.79243 -111.8938 135.5326 147.6030 103.2861
## 6 5.397 40.79243 -111.8938 136.3673 147.6030 103.2861
##       dist     angle route location
## 1 1.963621 -0.190404     1      SLC
## 2 1.963806 -0.190386     1      SLC
## 3 1.963806 -0.190386     1      SLC
```

```
## 4 1.963806 -0.190386     1      SLC
## 5 1.963806 -0.190386     1      SLC
## 6 1.963806 -0.190386     1      SLC
```

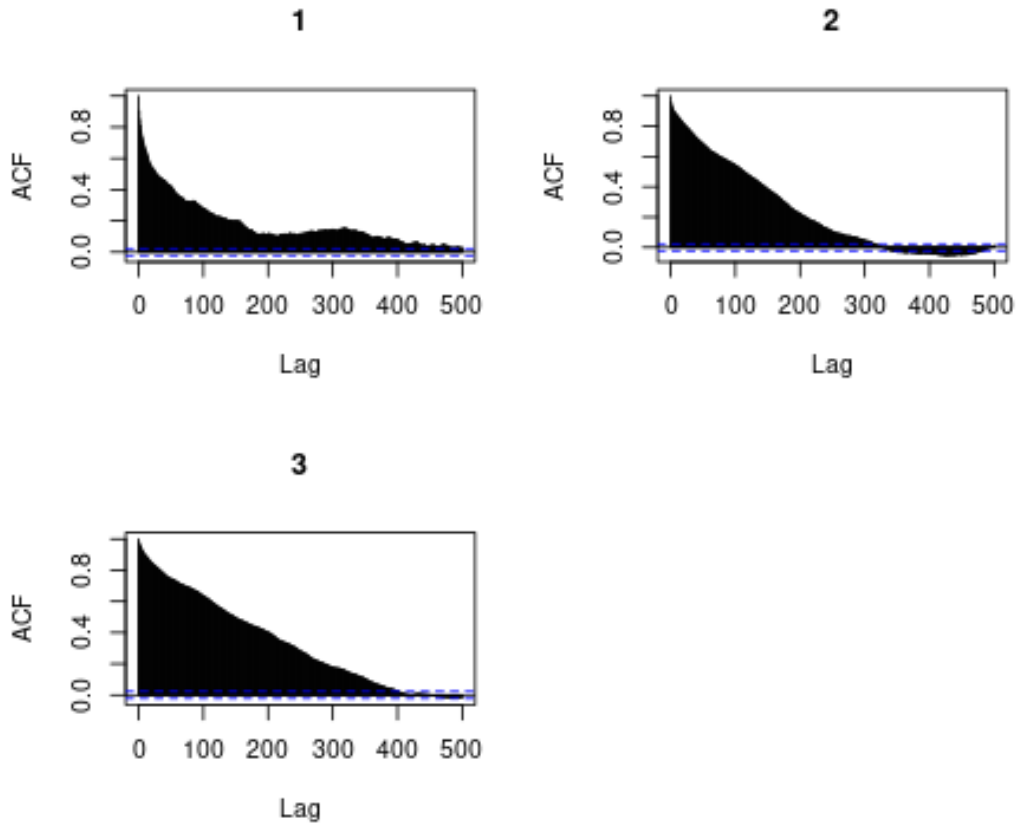
Dimensions of data:

```
## [1] 22000    10
```

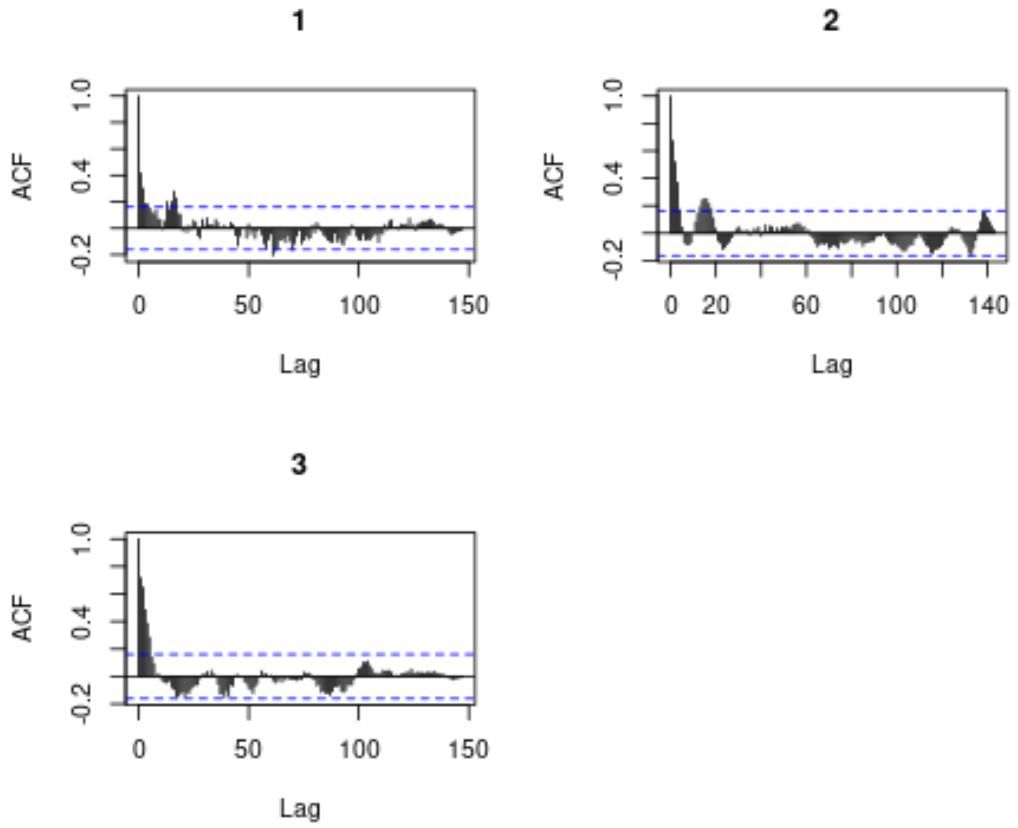
2. Plot measurement data from Salt Lake City, three routes. Transmitter is marked with an *X*.



3. Examine serial correlation via ACF plots for SLC data, routes 1–3:



4. Address serial correlation. Downsample, taking every 50th observation and plotting the ACF:



Again, an improvement—this time by taking every 50th observation. Still some autocorrelation. Each route would have to be further addressed individually.

2 K-factor and LOS Conditions in Martin Acres

Examine the Martin Acres data from the screening experiment, all data from all roads in the neighborhood, including Lashley & Moorhead. (Frequency = 1760 MHz.)

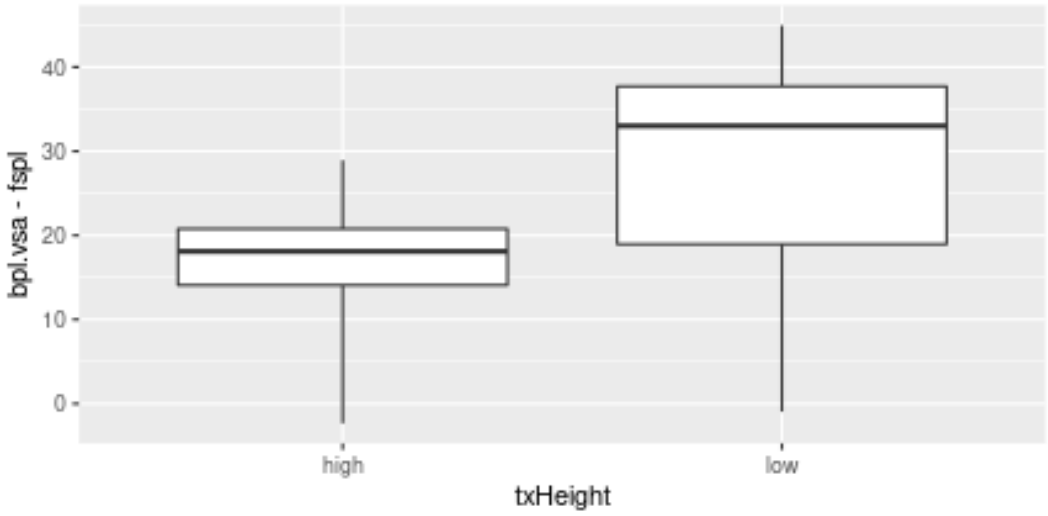
My goal is to see how well the Rician K factor can predict LOS conditions.

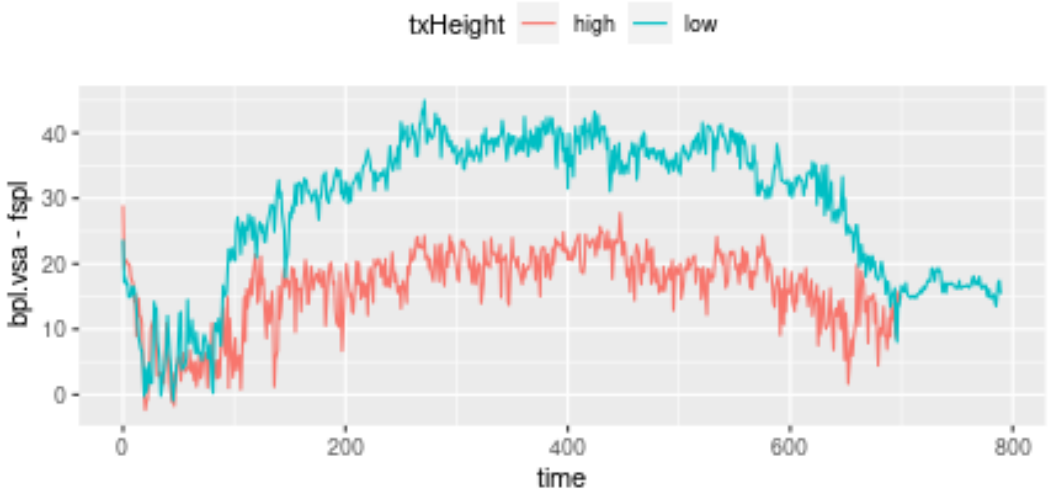
The Rician K factor, AKA Rician Factor, AKA K factor, is the ratio of the powers of the LoS component to the diffuse, or non-LoS component. The relative power of the LoS component, represented by the K factor, is a useful measure of com-

munication link quality. When $K = 0$, there is no LoS component. When $K = \infty$, there is no diffuse component.

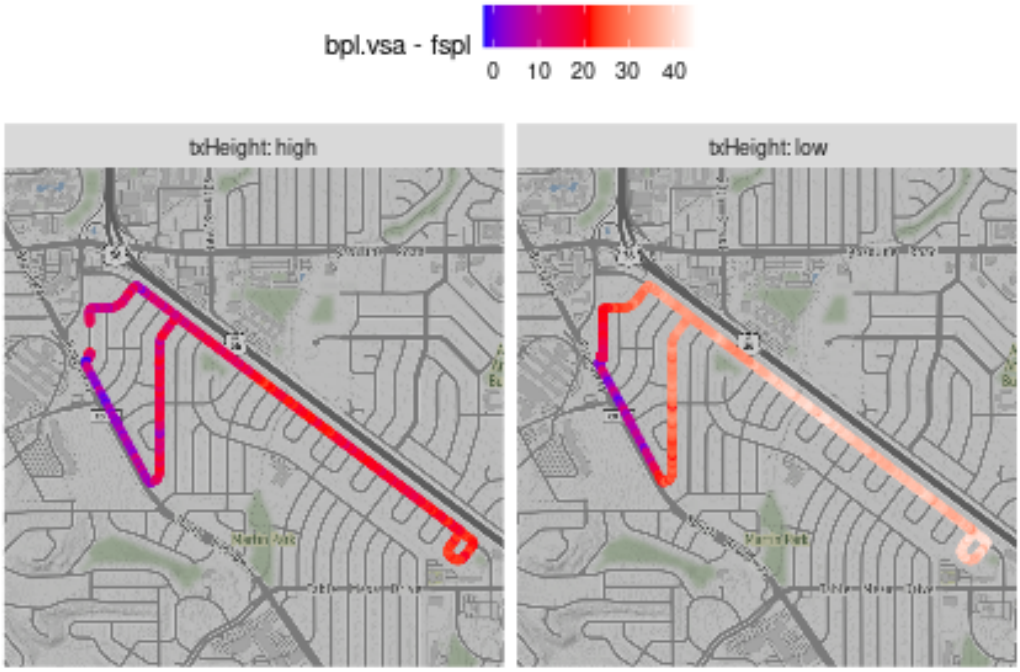
$$K = \frac{\text{LoS component power}}{\text{diffuse component power}} \tag{1}$$

1. Read in data and organize
2. First look at clutter (basic path loss minus free space path loss). Examine uncategorized clutter measurements WRT LOS conditions. Recall that clutter has been defined as the measured path loss (AKA basic path loss) minus the free space path loss. Clutter is measured as power in dB units.

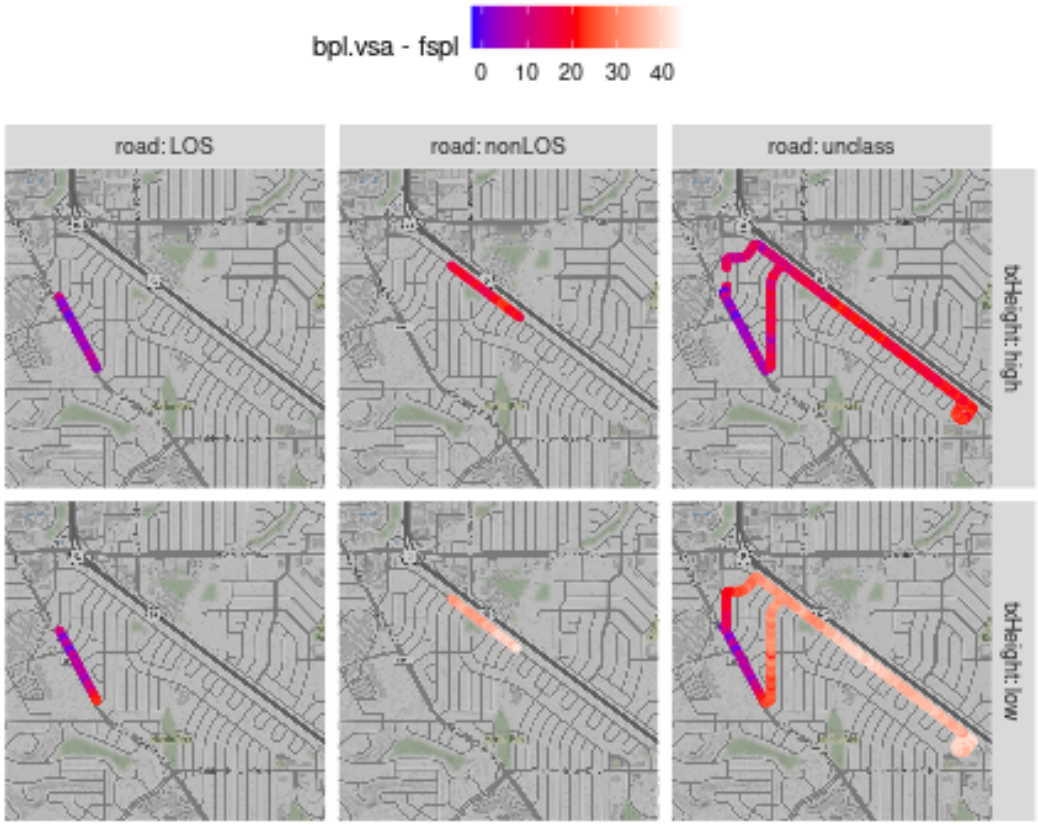




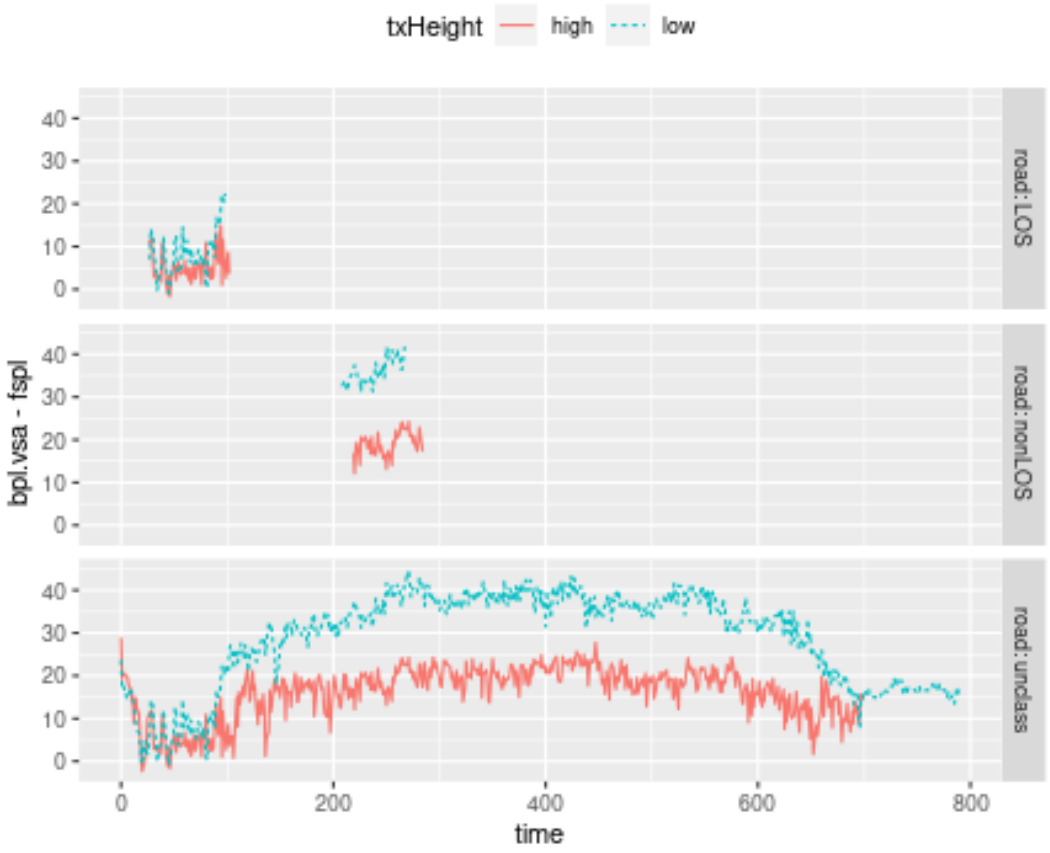
3. On a map:



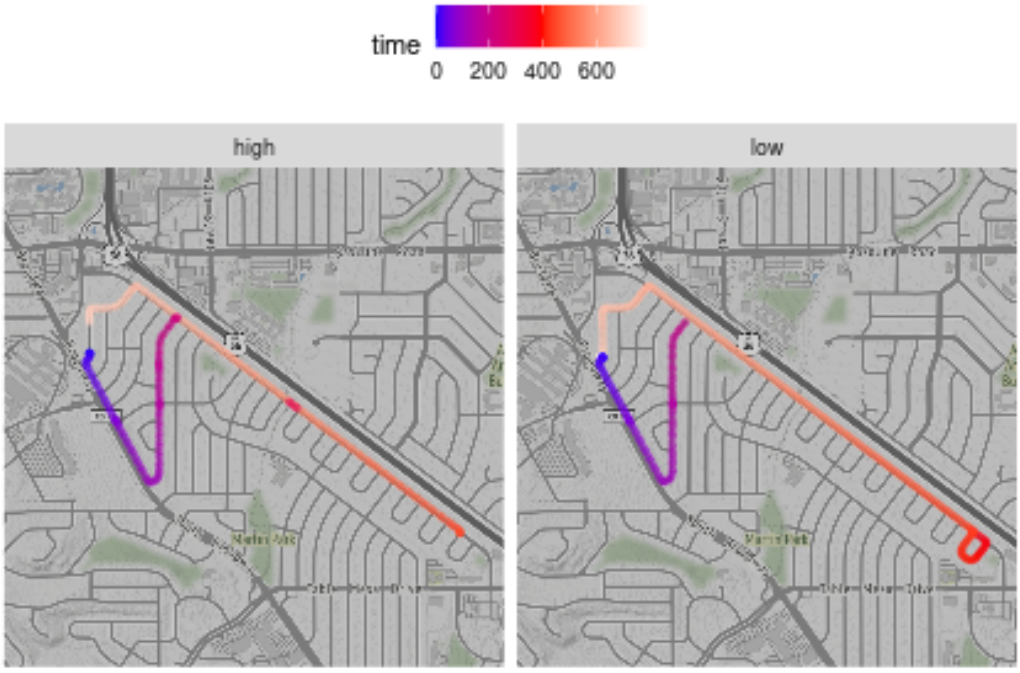
4. The LOS conditions occur on Lahsley Ln, and the non-LOS conditions occur on Moorhead Ave.



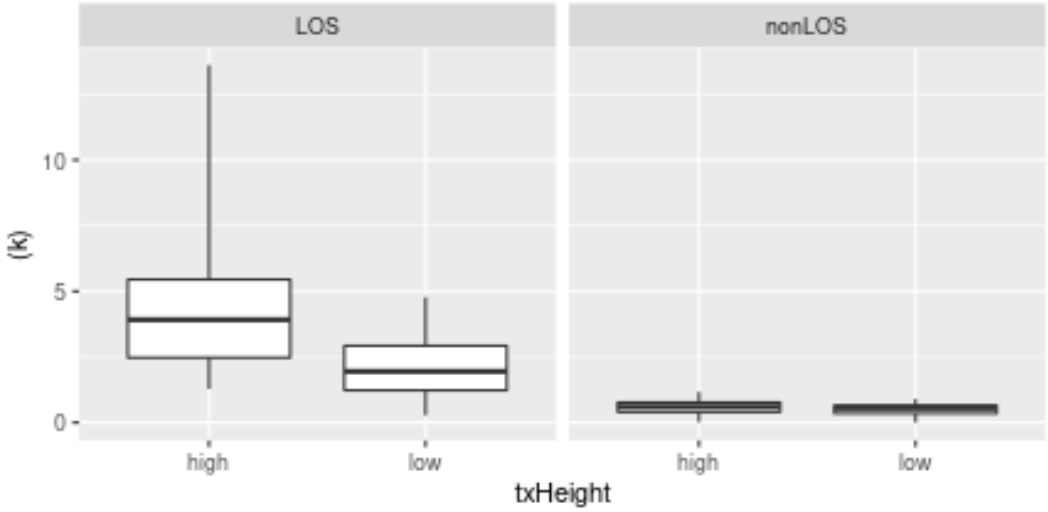
As lineplots:



5. Remove repeated data from when roads were traversed twice or more. Do for both antenna locations.
6. View final data set as collected over time. Recall that only one direction on Moorhead was used (not the back-and-forth data).

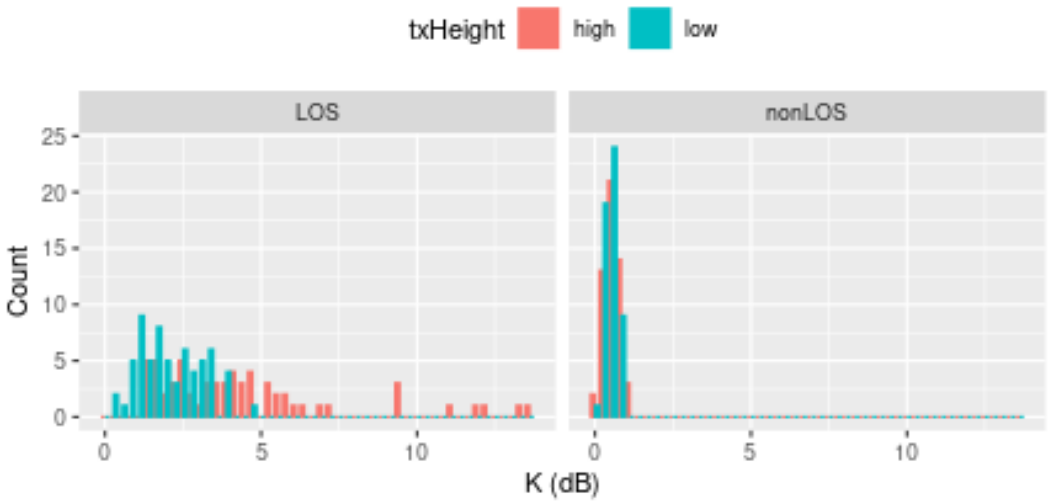


7. Examine K-factor boxplots for each tx height and LOS condition (using classified data).



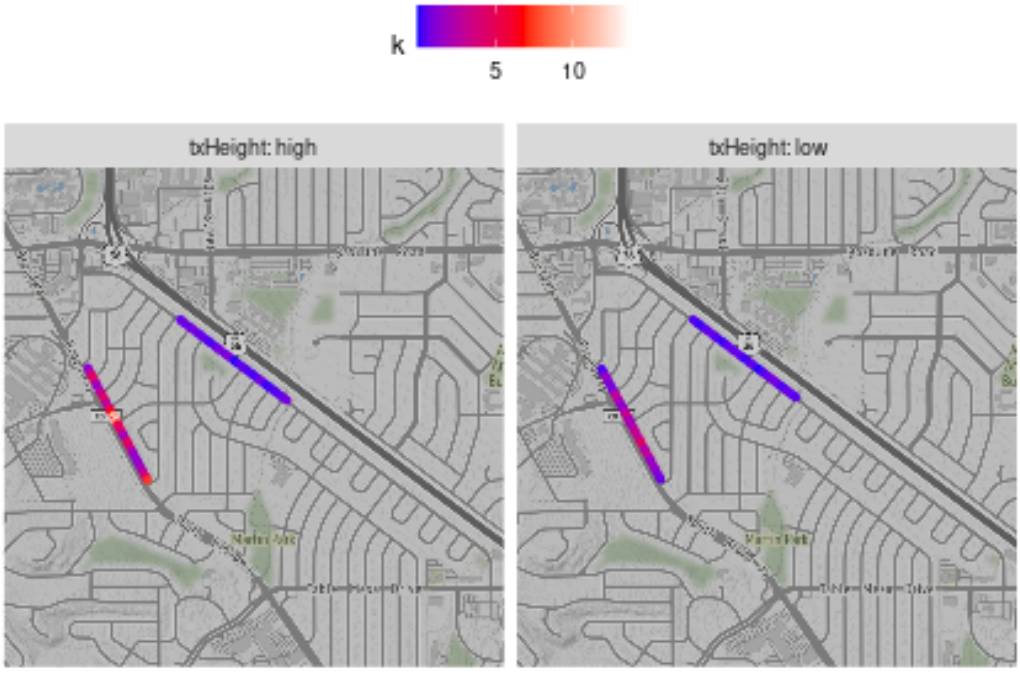
LOS conditions seem to have some effect on K central tendency and dispersion. Tx Height has a bigger effect in LOS conditions.

8. Dodged histograms of K factor:



   Notice similarities and discrepancies between and within each variable. It's useful to be able to visualize how this works.

9. Examine K factors on the map



10. Analyze how the k-factor associates with LOS conditions. Compute the point-biserial correlation between LOS conditions and k. (Using the catagorized data.) The point-biserial correlation is a measure of association between a dichotomous and a continuous variable.

```
library(ltm)
biserial.cor(dd$k, dd$road)

## [1] 0.5854125
```

Almost 60%. This means K is point-biserially correlated to LOS condition. I did expect a larger value. Perhaps our nonLOS conditions weren't extreme enough.

11. How does the standard deviation of the machine-measured path loss samples which were binned associate with the LOS conditions?

```
biserial.cor(dd$std.dbm, dd$road)

## [1] 0.8751885
```

MUCH higher than K factor.

(a) Out of curiosity, How do the K factor associate with standard deviation

of the machine-measured path loss samples?

```
cor.test(dd$std.dbm, dd$k, method='pearson')
```

```
##
##  Pearson's product-moment correlation
##
## data:  dd$std.dbm and dd$k
## t = 7.972, df = 230, p-value = 7.261e-14
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3579451 0.5605009
## sample estimates:
##       cor
## 0.4652925
```

Lower than I expected.

12. Perform some hypothesis tests for to better understand the spread and location of K based upon the two variables: LoS condition and transmitter height

```
dd.aov=aov(k~txHeight*road, data=dd)
summary(dd.aov)
```

```
##                Df Sum Sq Mean Sq F value   Pr(>F)
## txHeight        1  107.8   107.8   38.11 3.03e-09 ***
## road            1  441.7   441.7  156.09  < 2e-16 ***
## txHeight:road   1   84.0    84.0   29.68 1.32e-07 ***
## Residuals     228  645.3     2.8
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This tells us that the transmitter height, road, and their interaction have a significant (meaning not zero) effect on K.

How much effect do these variables have?

```
library(effectsize)
omega_squared(dd.aov, partial=F, ci=0.95)
```

```
## Parameter     | Omega2 |       95% CI
## -------------------------------------
## txHeight      |   0.08 | [0.02, 0.16]
```

```
## road           |   0.34 | [0.25, 0.43]
## txHeight:road |   0.06 | [0.01, 0.13]
```

The road (LoS conditions) accounts for 34% of the variability in K. Transmitter height only accounted for 8%, and the interaction accounted for 6%.

2.1 Conclusion

sub:conclusion

- LOS conditions and transmitter height, and their interaction do indeed have an effect on K factor. LoS conditions had the biggest effect (34%), followed by transmitter height (8%), and their interaction (6%).

  I was able to quantify 48% of the variability in K factor from two variables.

- K factor may be a good predictor of LOS conditions. Standard deviation may be even better...

3 Standard Deviation and LOS Conditions in Martin Acres

Examine the association between the signals' standard deviations and the LOS conditions.

TODO