Untitled

November 7, 2021

```
[1]: from pandas import *
```

```
[2]: df = read_csv("device_all.csv")
     df
```

```
[2]:       device  measure_no  rssi  measured_distance
     0      esp20           0   -65           0.667065
     1      esp20           0   -79           0.667065
     2      esp20           0   -65           0.667065
     3      esp20           0   -64           0.667065
     4      esp20           0   -82           0.667065
     ...      ...         ...   ...                ...
     2087   esp22           0  -103           5.377492
     2088   esp22           0   -95           5.377492
     2089   esp22           0   -95           5.377492
     2090   esp22           0   -98           5.377492
     2091   esp22           0  -100           5.377492

     [2092 rows x 4 columns]
```
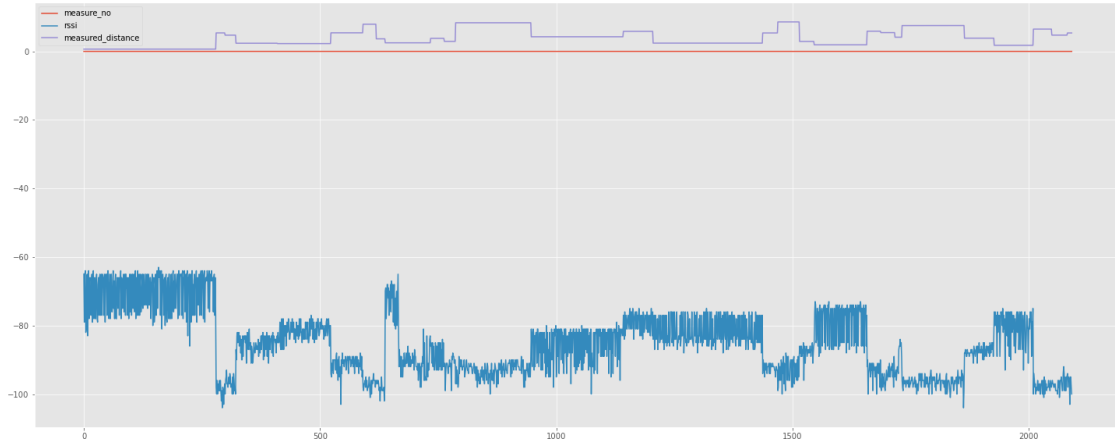
```
[3]: import matplotlib.pyplot as plt
     from pandas.plotting import register_matplotlib_converters
     register_matplotlib_converters()
     plt.style.use("ggplot")
```

```
[4]: df.plot(figsize=(25,10))
```

```
[4]: <AxesSubplot:>
```

```
[5]: esp20df = df[df["device"] == "esp20"]
```

```
[6]: esp20df
```

```
[6]:      device  measure_no  rssi  measured_distance
     0     esp20           0   -65           0.667065
     1     esp20           0   -79           0.667065
     2     esp20           0   -65           0.667065
     3     esp20           0   -64           0.667065
     4     esp20           0   -82           0.667065
     ..      ...         ...   ...                ...
     782   esp20           0   -89           2.924196
     783   esp20           0   -89           2.924196
     784   esp20           0   -91           2.924196
     785   esp20           0   -90           2.924196
     786   esp20           0   -90           2.924196

     [787 rows x 4 columns]
```
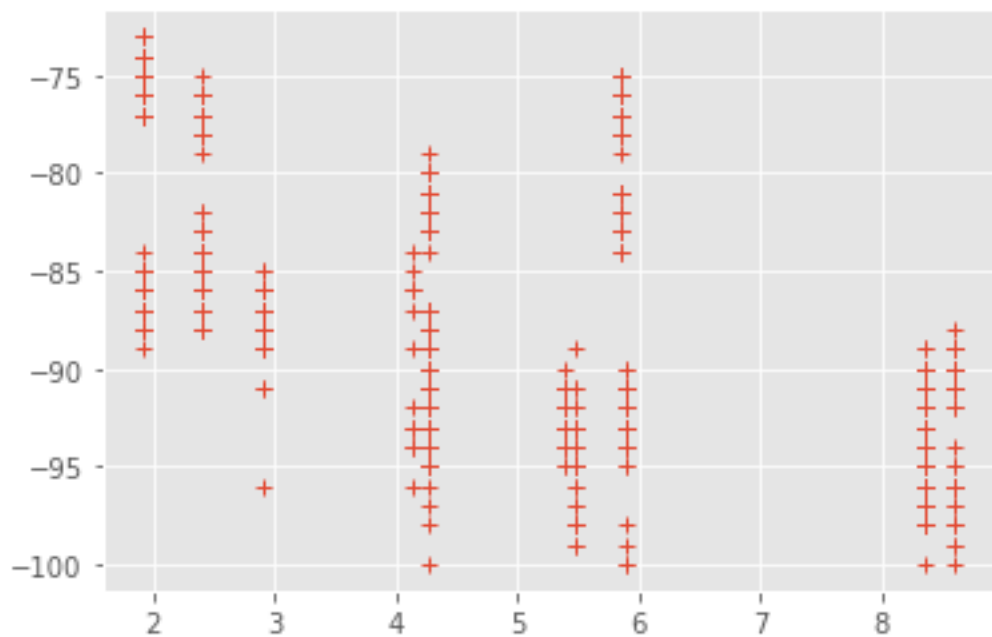
```
[7]: plt.plot(esp20df["measured_distance"].to_numpy(),esp20df["rssi"].to_numpy(),
     ↪"+")
     plt.show()
```
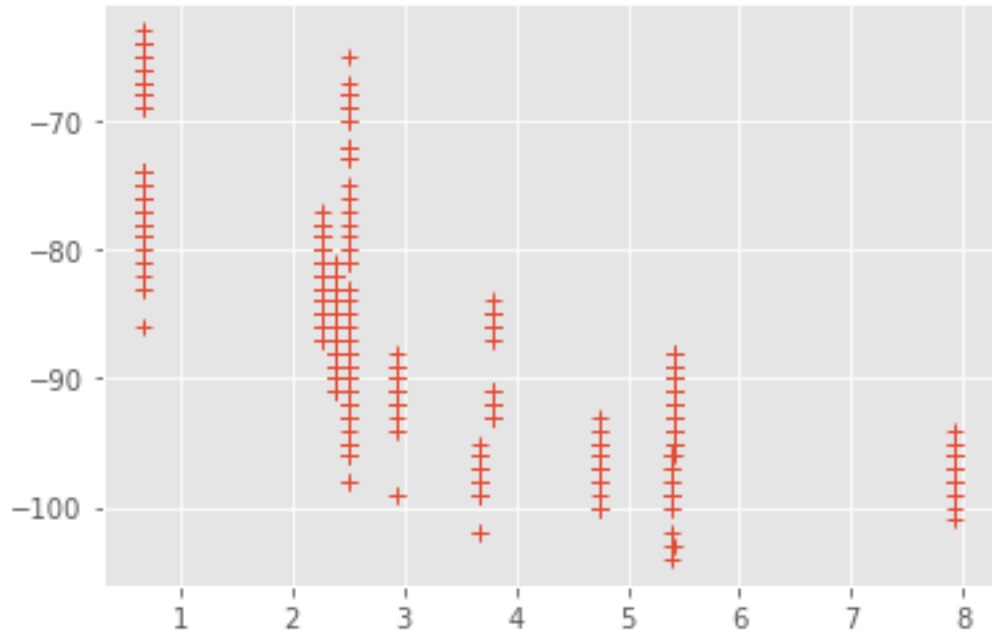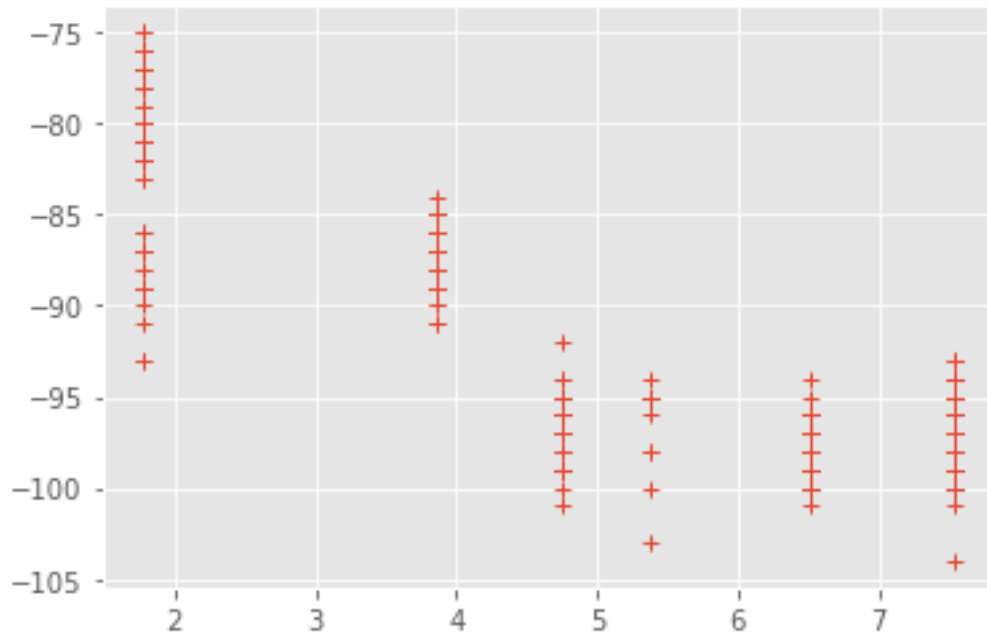
```
[8]: esp21df = df[df["device"] == "esp21"]
```

```
[9]: plt.plot(esp21df["measured_distance"].to_numpy(),esp21df["rssi"].to_numpy(),␣
     ↪"+")
     plt.show()
```

```
[10]: esp22df = df[df["device"] == "esp22"]
      plt.plot(esp22df["measured_distance"].to_numpy(),esp22df["rssi"].to_numpy(),␣
       ↪"+")
      plt.show()
```



```
[11]: import numpy as np
```

```
[12]: # RSSI = A * 10 * log( distance / d0 ) + B
```

```
[ ]:
```

```
[13]: #>>> x = numpy.array([1, 7, 20, 50, 79])
      #>>> y = numpy.array([10, 19, 30, 35, 51])
      #>>> scipy.optimize.curve_fit(lambda t,a,b: a+b*numpy.log(t),   x,   y)
      #(array([ 6.61867467,   8.46295606]),
      # array([[ 28.15948002,   -7.89609542],
      #        [ -7.89609542,    2.9857172 ]]))
      # y   6.62 + 8.46 log(x)
```

```
[14]: import scipy
      import scipy.optimize
```

```
[15]: # distance = d0 * exp((RSSI - b)/a) + d
```

```
[16]: def createPredicted(dataframe, p0):
          result = scipy.optimize.curve_fit(lambda t,a,b,c,d: c*np.exp((b+t)*a)+d, ⊔
       ↪dataframe["rssi"].to_numpy(),  dataframe["measured_distance"].to_numpy(), ⊔
       ↪p0=p0,  maxfev=10000)
          reg = result[0]
          dist = result[1]
          print(reg)
          dfret = dataframe.assign( predicted = (np.exp((dataframe["rssi"].to_numpy()⊔
       ↪+ reg[1]  ) * reg[0]) *  reg[2] + reg[3]))
          return (reg, dist, dfret)
```

```
[17]: esp20reg = createPredicted(esp20df, (-0.3, 0.1,0.1,0))
      esp20reg
```

```
[-0.04250667 -0.31412568  0.09530329 -0.95485682]
```

```
[17]: (array([-0.04250667, -0.31412568,  0.09530329, -0.95485682]),
       array([[ 2.39522094e-05, -2.14349594e+02, -8.68088196e-01,
                -1.82707211e-03],
              [-2.14349590e+02,  1.56792785e+11,  6.35170020e+08,
                1.77939865e+04],
              [-8.68088179e-01,  6.35170020e+08,  2.57308366e+06,
                7.20648196e+01],
              [-1.82707211e-03,  1.77939868e+04,  7.20648208e+01,
                1.48081696e-01]]),
           device  measure_no  rssi  measured_distance  predicted
      0     esp20           0   -65           0.667065   0.575644
      1     esp20           0   -79           0.667065   1.820247
      2     esp20           0   -65           0.667065   0.575644
      3     esp20           0   -64           0.667065   0.511951
      4     esp20           0   -82           0.667065   2.197682
      ..      ...         ...   ...                ...        ...
      782   esp20           0   -89           2.924196   3.290198
      783   esp20           0   -89           2.924196   3.290198
      784   esp20           0   -91           2.924196   3.666869
      785   esp20           0   -90           2.924196   3.474532
      786   esp20           0   -90           2.924196   3.474532

      [787 rows x 5 columns])
```

```
[18]: plt.plot(esp20reg[2]['predicted'])
      plt.plot(esp20reg[2]["measured_distance"])
```

```
[18]: [<matplotlib.lines.Line2D at 0x7f6d5829de80>]
```
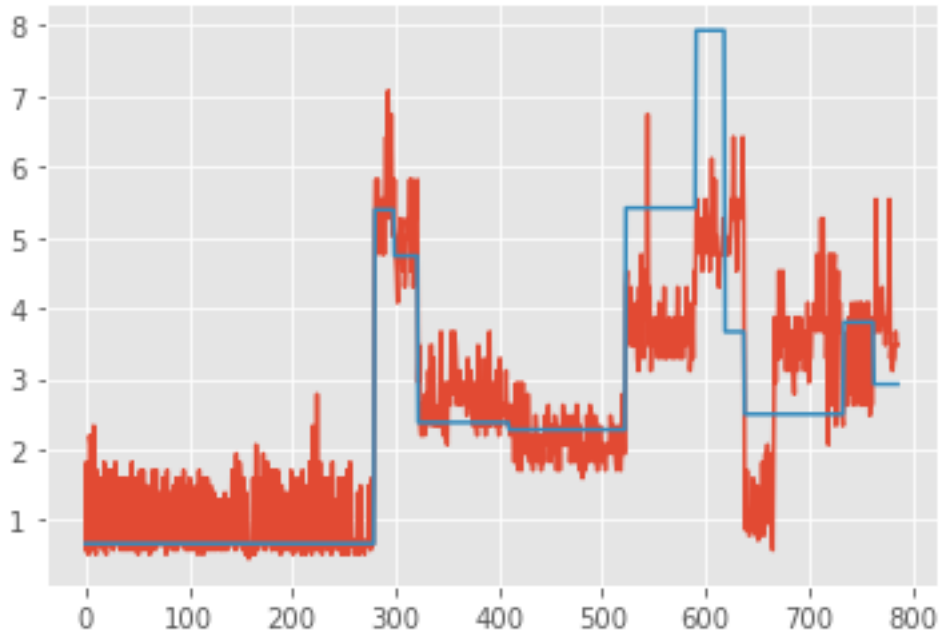
```
[19]: esp21reg = createPredicted(esp21df,(-0.2, 0.2,0.1,-4))
      plt.plot(esp21reg[2]['predicted'])
      plt.plot(esp21reg[2]["measured_distance"])
      esp21reg
```

```
[-0.05915429 -0.47845718  0.02246369  0.60301902]
```

```
[19]: (array([-0.05915429, -0.47845718,  0.02246369,  0.60301902]),
       array([[ 1.25109347e-04, -3.06391155e+02, -4.06843927e-01,
               -8.22277146e-03],
              [-3.06391145e+02,  4.30199848e+10,  5.71652850e+07,
                1.63353711e+04],
              [-4.06843913e-01,  5.71652850e+07,  7.59616689e+04,
                2.16872607e+01],
              [-8.22277146e-03,  1.63353717e+04,  2.16872616e+01,
                5.64806726e-01]]),
           device  measure_no  rssi  measured_distance  predicted
      787   esp21           0   -97           8.352429   7.776662
      788   esp21           0   -92           8.352429   5.939904
      789   esp21           0   -97           8.352429   7.776662
      790   esp21           0   -93           8.352429   6.265128
      791   esp21           0   -96           8.352429   7.364618
      ...     ...         ...   ...                ...        ...
      1727  esp21           0   -86           4.137273   4.345379
      1728  esp21           0   -84           4.137273   3.927814
      1729  esp21           0   -86           4.137273   4.345379
```

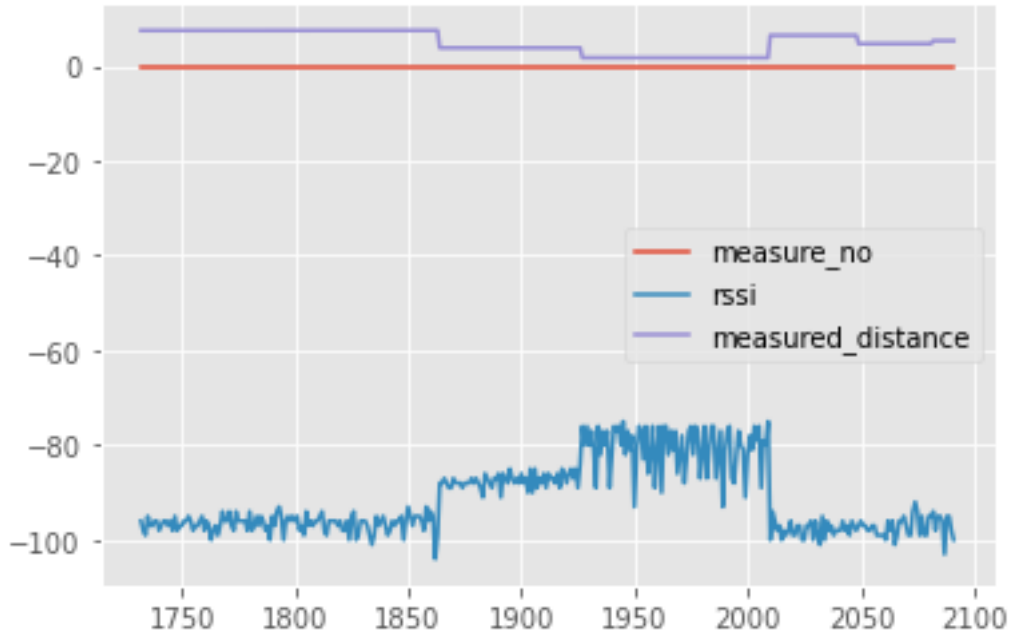```
1730   esp21   0   -85   4.137273   4.130423
1731   esp21   0   -89   4.137273   5.072085

[945 rows x 5 columns])
```
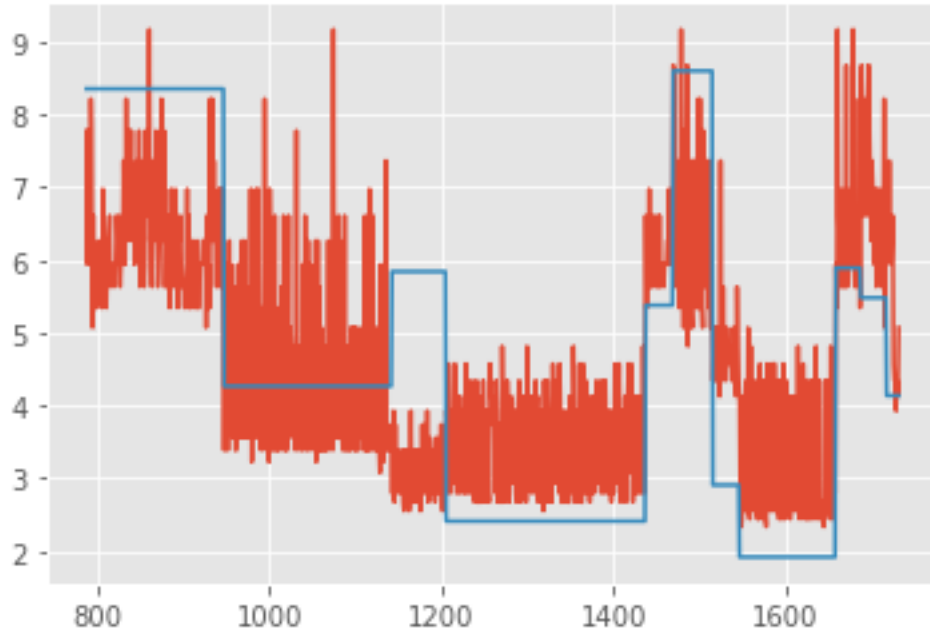


```
[20]: esp22df.plot()
```

```
[20]: <AxesSubplot:>
```

```
[21]: esp22reg = createPredicted(esp22df,(-0.2, 0.2,0.1,-4))
      plt.plot(esp22reg[2]['predicted'])
      plt.plot(esp22reg[2]["measured_distance"])
      esp22reg
```

```
[-9.67762835e-03 -1.91016056e+02  1.81913542e+00 -2.29341611e+01]
```

```
[21]: (array([-9.67762835e-03, -1.91016056e+02,  1.81913542e+00, -2.29341611e+01]),
       array([[ 9.70083248e-05,  6.35555014e+03,  1.11947055e+02,
               -2.72540083e-01],
              [ 6.35660343e+03,  3.41131389e+14,  6.00508869e+12,
               -1.18479867e+07],
              [ 1.11965596e+02,  6.00508869e+12,  1.05710267e+11,
               -2.08755065e+05],
              [-2.72540064e-01, -1.18450263e+07, -2.08702953e+05,
                7.66569139e+02]]),
            device  measure_no  rssi  measured_distance  predicted
      1732  esp22           0   -96           7.530922   6.319246
      1733  esp22           0   -98           7.530922   6.890969
      1734  esp22           0   -99           7.530922   7.181006
      1735  esp22           0   -95           7.530922   6.037508
      1736  esp22           0   -97           7.530922   6.603724
      ...     ...         ...   ...                ...        ...
      2087  esp22           0  -103           5.377492   8.369638
      2088  esp22           0   -95           5.377492   6.037508
      2089  esp22           0   -95           5.377492   6.037508
```

```
2090  esp22              0    -98           5.377492   6.890969
2091  esp22              0    -100          5.377492   7.473865

[360 rows x 5 columns])
```