IoT Lab 3 Report

Team name: The Gradient Ascenders

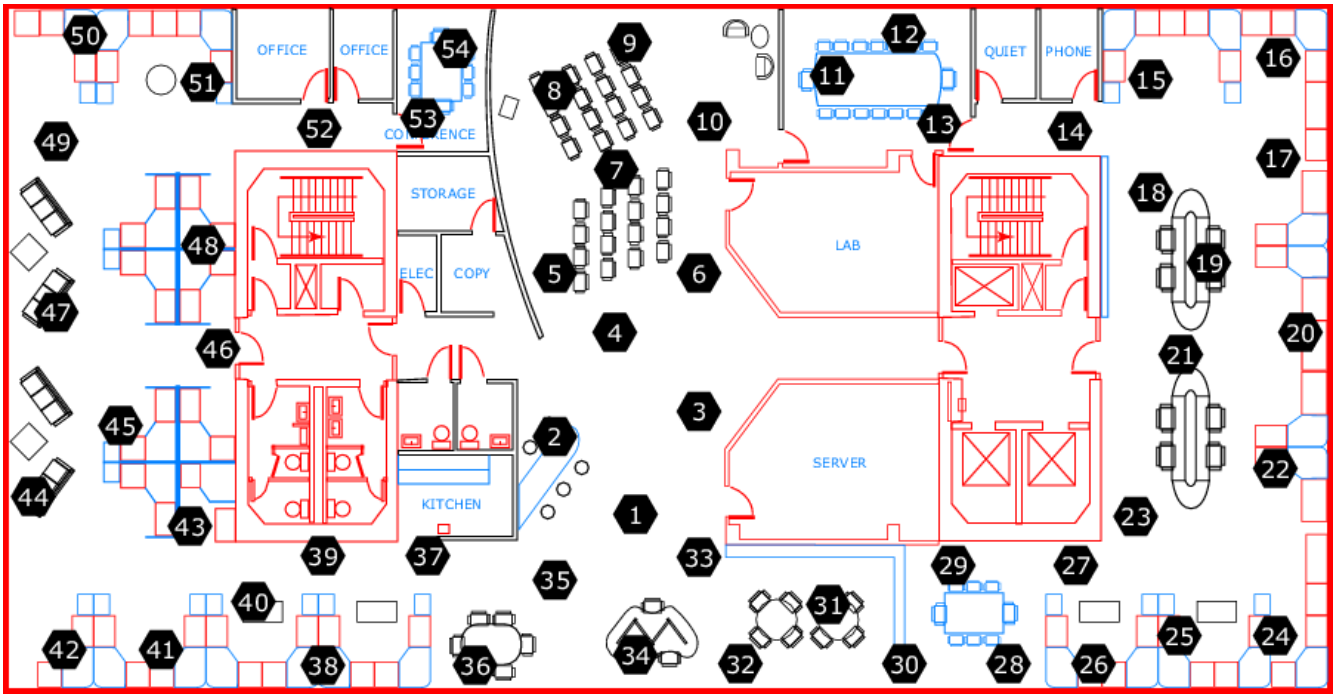
Kanika Narang	Meghna Srivastava	Shivani Naik
MT2016069	MT2016082	MT2016088

Problem:

The objective of this lab is to work on available sensor data and determine the statistical correlation measure of the same.

Refer to the link provided in the reference section. It provides information on a sensor testbed deployment and data collection from various environmental sensors. Understand the sensor placement in the floor plan, the types and data reading frequency. Download the data file and understand the schema.

Sensor Placement:



As we can observe, there are 54 sensors placed at various places on the floor.

date:yyyy-mm-dd	time:hh:mm:ss.xxx	epoch:int	moteid:int	temperature:real	humidity:real	light:real	voltage:real

Data was collected over a span of 28th February to 5th April. One epoch corresponds to 30 seconds and is a monotonically increasing sequence number for every sensor. Every sensor has a unique mote id. Temperature is in degrees Celsius. Humidity is temperature corrected relative humidity, ranging from 0-100%. Light is in Lux (a value of 1 Lux corresponds to moonlight, 400 Lux to a bright office, and 100,000 Lux to full sunlight.)

We have used R to perform this lab.

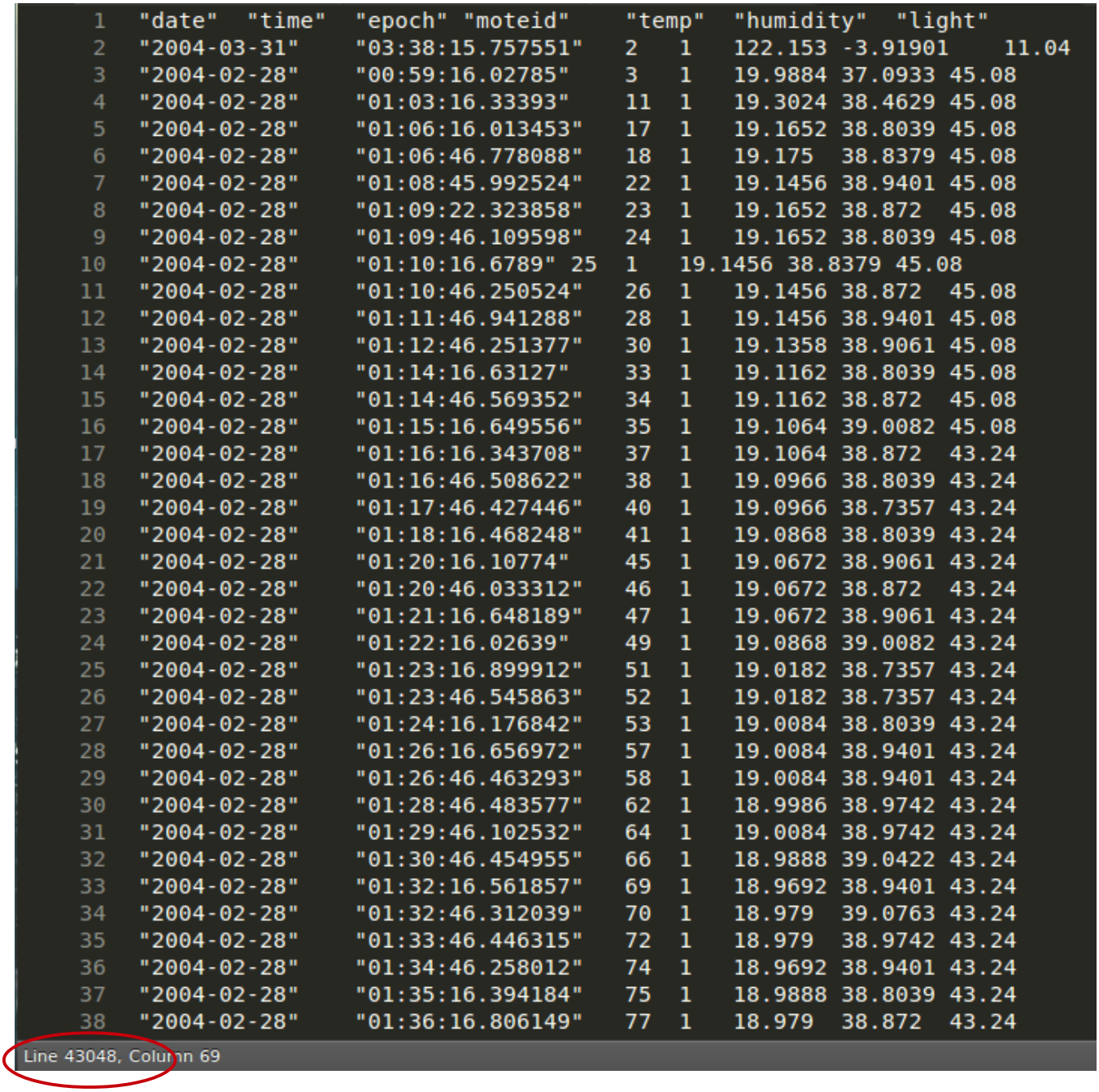
Part 1: Save sensor 1's data (Temperature, humidity, light).

We separated the data collected from sensor 1 and saved it in another file called sensor1.txt.

```
#Read data from the txt file as a table
data <- read.table("data.txt", sep=" ")
#Form a data frame from the table
dataframe1 <- data.frame((data))

#Renaming columns
colnames(dataframe1)<-("date","time","epoch","moteid","temp","humidity","light","volt")

#Form a subset and leave out voltage column (8)
sub <- subset(dataframe1, moteid==1, select=c(-8))

#Write data to sensor1.txt
write.table(sub,"sensor1.txt", sep="\t", row.names=FALSE)
```

Observations:

1. As we can see from the output file's screenshot, there are 43048 entries for sensor 1.
2. There are missing epochs.
3. We also have some outliers, with temperature in the range of 122 degrees.



	"date"	"time"	"epoch"	"moteid"	"temp"	"humidity"	"light"	
2	"2004-03-31"	"03:38:15.757551"	2	1	122.153	-3.91901	11.04	
3	"2004-02-28"	"00:59:16.02785"	3	1	19.9884	37.0933	45.08	
4	"2004-02-28"	"01:03:16.33393"	11	1	19.3024	38.4629	45.08	
5	"2004-02-28"	"01:06:16.013453"	17	1	19.1652	38.8039	45.08	
6	"2004-02-28"	"01:06:46.778088"	18	1	19.175	38.8379	45.08	
7	"2004-02-28"	"01:08:45.992524"	22	1	19.1456	38.9401	45.08	
8	"2004-02-28"	"01:09:22.323858"	23	1	19.1652	38.872	45.08	
9	"2004-02-28"	"01:09:46.109598"	24	1	19.1652	38.8039	45.08	
10	"2004-02-28"	"01:10:16.6789"	25	1	19.1456	38.8379	45.08	
11	"2004-02-28"	"01:10:46.250524"	26	1	19.1456	38.872	45.08	
12	"2004-02-28"	"01:11:46.941288"	28	1	19.1456	38.9401	45.08	
13	"2004-02-28"	"01:12:46.251377"	30	1	19.1358	38.9061	45.08	
14	"2004-02-28"	"01:14:16.63127"	33	1	19.1162	38.8039	45.08	
15	"2004-02-28"	"01:14:46.569352"	34	1	19.1162	38.872	45.08	
16	"2004-02-28"	"01:15:16.649556"	35	1	19.1064	39.0082	45.08	
17	"2004-02-28"	"01:16:16.343708"	37	1	19.1064	38.872	43.24	
18	"2004-02-28"	"01:16:46.508622"	38	1	19.0966	38.8039	43.24	
19	"2004-02-28"	"01:17:46.427446"	40	1	19.0966	38.7357	43.24	
20	"2004-02-28"	"01:18:16.468248"	41	1	19.0868	38.8039	43.24	
21	"2004-02-28"	"01:20:16.10774"	45	1	19.0672	38.9061	43.24	
22	"2004-02-28"	"01:20:46.033312"	46	1	19.0672	38.872	43.24	
23	"2004-02-28"	"01:21:16.648189"	47	1	19.0672	38.9061	43.24	
24	"2004-02-28"	"01:22:16.02639"	49	1	19.0868	39.0082	43.24	
25	"2004-02-28"	"01:23:16.899912"	51	1	19.0182	38.7357	43.24	
26	"2004-02-28"	"01:23:46.545863"	52	1	19.0182	38.7357	43.24	
27	"2004-02-28"	"01:24:16.176842"	53	1	19.0084	38.8039	43.24	
28	"2004-02-28"	"01:26:16.656972"	57	1	19.0084	38.9401	43.24	
29	"2004-02-28"	"01:26:46.463293"	58	1	19.0084	38.9401	43.24	
30	"2004-02-28"	"01:28:46.483577"	62	1	18.9986	38.9742	43.24	
31	"2004-02-28"	"01:29:46.102532"	64	1	19.0084	38.9742	43.24	
32	"2004-02-28"	"01:30:46.454955"	66	1	18.9888	39.0422	43.24	
33	"2004-02-28"	"01:32:16.561857"	69	1	18.9692	38.9401	43.24	
34	"2004-02-28"	"01:32:46.312039"	70	1	18.979	39.0763	43.24	
35	"2004-02-28"	"01:33:46.446315"	72	1	18.979	38.9742	43.24	
36	"2004-02-28"	"01:34:46.258012"	74	1	18.9692	38.9401	43.24	
37	"2004-02-28"	"01:35:16.394184"	75	1	18.9888	38.8039	43.24	
38	"2004-02-28"	"01:36:16.806149"	77	1	18.979	38.872	43.24	

Line 43048, Column 69

Original data from sensor 1

Prediction of missing data:

We have tried two ways of filling the missing data:

1] Temporal correlation: We know that 5 epochs correspond to 2.5 minutes and there is no observable change in the various parameters that we are capturing during this duration. Thus, we have used a window of previous 5 epochs and calculated their average to predict the missing epoch values.
We found that the autocorrelation with a lag 5 (we are taking 5 epoch window) was very high, 0.998.

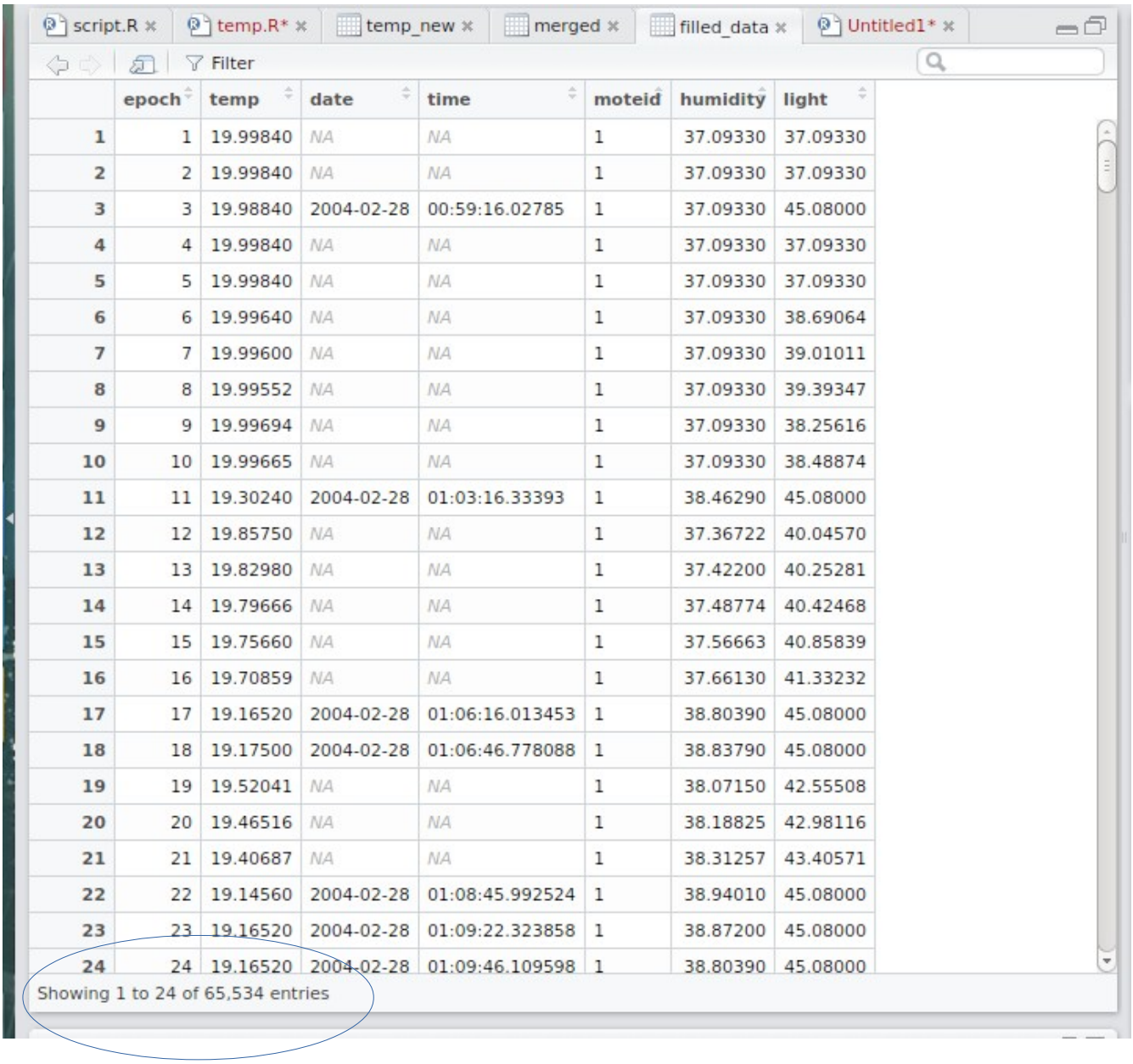
2] Spatial Correlation: According to the floor plan the nearest sensor to sensor 1 was sensor 33. So we calculated the correlation coefficient between the two sensors and the value was .7482. And we filled the data of the missing epochs with the data available from sensor 33.

We can see that temporal correlation is more accurate and the correlation coefficient is lesser than the autocorrelation. Also we can see there is biasing among different sensors due to the environment in which they are placed, hence the predicted values of the temporal correlation method seems to be more accurate than the spatial correlation method and we are pasting those results here.

Code:

```
#Function which will replace NA (absent epoch values) with average of previous 5 epochs

aver = function(x){
  for(i in 6:nrow(x))
  {

    if(is.na(x[i,2]))
    {

      x[i, 2] = (x[i - 1, 2] + x[i - 2, 2] + x[i - 3 , 2] + x[i - 4 , 2] + x[i - 5 , 2]) / 5
    }
  }
  return(x)
}

#Get the readings of sensor 1 into “sub”, removing outliers outside range of [10, 60] degrees
sub <- subset(dataframe1, (moteid==1) & (temp < 60) & (temp>10), select=c (-8))

#Remove duplicate epochs (erroroneous epoch values) present in data
sub = sub[!duplicated(sub[3]), ]

#new_table contains auto-generated increasing epoch numbers
epoch = c(1:65534)
new_table <- data.frame(epoch)

#Outer join the epoch numbers table and sensor 1 readings. The rows of missing epoch values will
#have NA values
merged = merge(x=new_table,y=sub,all=TRUE, by="epoch")

#Select required rows into new dataframes
temp_new = subset(merged, select=c(1,5))
humidity_new = subset(merged, select = c(1,6))
light_new = subset(merged, select = c(1,7))

#Calculate missing values
temp_new = aver(temp_new)
humidity_new = aver(humidity_new)
light_new = aver(light_new)
```

```
#Form the final merged data frame by taking join over epoch values
filled_data = merge(x=temp_new, y = merged[, c("epoch", "date", "time","moteid")], by= "epoch" )
filled_data = merge(x=filled_data, y = humidity_new, by="epoch")
filled_data = merge(x=filled_data, y = light_new, by="epoch")
filled_data[,5] = 1

#Write new data to file
write.table(filled_data,"filled_data.txt",sep="\t",row.names=FALSE)
```



script.R × | temp.R* × | temp_new × | merged × | filled_data × | Untitled1* ×

Filter

	epoch	temp	date	time	moteid	humidity	light
1	1	19.99840	NA	NA	1	37.09330	37.09330
2	2	19.99840	NA	NA	1	37.09330	37.09330
3	3	19.98840	2004-02-28	00:59:16.02785	1	37.09330	45.08000
4	4	19.99840	NA	NA	1	37.09330	37.09330
5	5	19.99840	NA	NA	1	37.09330	37.09330
6	6	19.99640	NA	NA	1	37.09330	38.69064
7	7	19.99600	NA	NA	1	37.09330	39.01011
8	8	19.99552	NA	NA	1	37.09330	39.39347
9	9	19.99694	NA	NA	1	37.09330	38.25616
10	10	19.99665	NA	NA	1	37.09330	38.48874
11	11	19.30240	2004-02-28	01:03:16.33393	1	38.46290	45.08000
12	12	19.85750	NA	NA	1	37.36722	40.04570
13	13	19.82980	NA	NA	1	37.42200	40.25281
14	14	19.79666	NA	NA	1	37.48774	40.42468
15	15	19.75660	NA	NA	1	37.56663	40.85839
16	16	19.70859	NA	NA	1	37.66130	41.33232
17	17	19.16520	2004-02-28	01:06:16.013453	1	38.80390	45.08000
18	18	19.17500	2004-02-28	01:06:46.778088	1	38.83790	45.08000
19	19	19.52041	NA	NA	1	38.07150	42.55508
20	20	19.46516	NA	NA	1	38.18825	42.98116
21	21	19.40687	NA	NA	1	38.31257	43.40571
22	22	19.14560	2004-02-28	01:08:45.992524	1	38.94010	45.08000
23	23	19.16520	2004-02-28	01:09:22.323858	1	38.87200	45.08000
24	24	19.16520	2004-02-28	01:09:46.109598	1	38.80390	45.08000

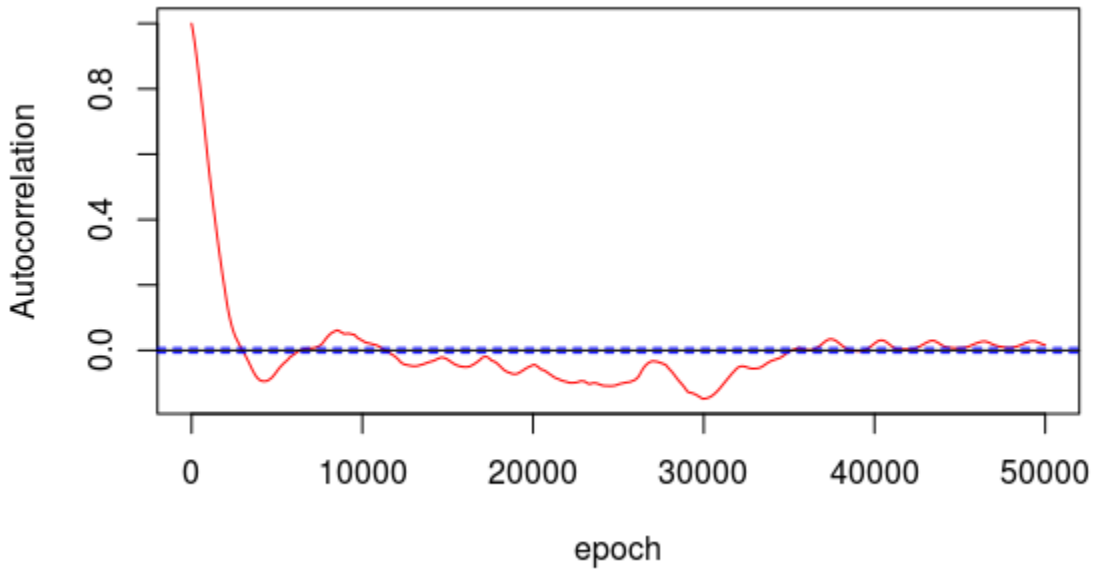
Showing 1 to 24 of 65,534 entries

Predicted Data Using Temporal Correlation

As we don't have the time and date for the missing epochs, we can see NA values in those columns. Since each epoch is of 31 seconds , so for a time period from 28 February to 5 April we will have around 65,534 epochs, and hence we have predicted data for 65,534 entires.

Part 2: Autocorrelation for temperature data.

```
#Store temperature in temp_new
temp_new <- subset(sub, select = c (temp))

#Find autocorrelation using acf
temp_autocorr = acf(temp_new$temp,type=c("correlation"),plot=TRUE, lag.max = 50000)
print(temp_autocorr)

#Plot autocorrelation against epoch shifts
plot(temp_autocorr,type="l",col="red", xlab="epoch", ylab="Autocorrelation")
```



Autocorrelation of sensor 1 temperature

```
Autocorrelations of series 'temp_new$temp', by lag

    0     1     2     3     4     5     6     7     8     9    10    11    12    13    14
1.000 0.999 0.998 0.998 0.998 0.998 0.997 0.997 0.997 0.997 0.997 0.996 0.996 0.996 0.996
   15    16    17    18    19    20    21    22    23    24    25    26    27    28    29
0.995 0.995 0.995 0.994 0.994 0.994 0.994 0.993 0.993 0.993 0.993 0.992 0.992 0.992 0.992
   30    31    32    33    34    35    36    37    38    39    40    41    42    43    44
0.991 0.991 0.991 0.990 0.990 0.990 0.990 0.989 0.989 0.989 0.988 0.988 0.988 0.988 0.987
   45    46    47    48    49    50    51    52    53    54    55    56    57    58    59
0.987 0.987 0.987 0.986 0.986 0.986 0.985 0.985 0.985 0.984 0.984 0.984 0.984 0.983 0.983
   60    61    62    63    64    65    66    67    68    69    70    71    72    73    74
0.983 0.982 0.982 0.982 0.981 0.981 0.981 0.980 0.980 0.980 0.980 0.979 0.979 0.979 0.978
   75    76    77    78    79    80    81    82    83    84    85    86    87    88    89
0.978 0.978 0.977 0.977 0.977 0.976 0.976 0.976 0.976 0.975 0.975 0.975 0.974 0.974 0.974
   90    91    92    93    94    95    96    97    98    99   100   101   102   103   104
0.973 0.973 0.973 0.972 0.972 0.972 0.971 0.971 0.971 0.970 0.970 0.970 0.969 0.969 0.969
  105   106   107   108   109   110   111   112   113   114   115   116   117   118   119
0.968 0.968 0.968 0.967 0.967 0.967 0.966 0.966 0.966 0.965 0.965 0.965 0.964 0.964 0.964
  120   121   122   123   124   125   126   127   128   129   130   131   132   133   134
0.963 0.963 0.963 0.962 0.962 0.962 0.961 0.961 0.961 0.960 0.960 0.960 0.959 0.959 0.959
  135   136   137   138   139   140   141   142   143   144   145   146   147   148   149
0.958 0.958 0.958 0.957 0.957 0.957 0.956 0.956 0.956 0.955 0.955 0.955 0.954 0.954 0.953
  150   151   152   153   154   155   156   157   158   159   160   161   162   163   164
0.953 0.953 0.952 0.952 0.952 0.951 0.951 0.951 0.950 0.950 0.950 0.949 0.949 0.949 0.948
  165   166   167   168   169   170   171   172   173   174   175   176   177   178   179
0.948 0.947 0.947 0.947 0.946 0.946 0.946 0.945 0.945 0.945 0.944 0.944 0.943 0.943 0.943
  180   181   182   183   184   185   186   187   188   189   190   191   192   193   194
0.942 0.942 0.942 0.941 0.941 0.941 0.940 0.940 0.939 0.939 0.939 0.938 0.938 0.938 0.937
  195   196   197   198   199   200   201   202   203   204   205   206   207   208   209
0.937 0.936 0.936 0.936 0.935 0.935 0.935 0.934 0.934 0.933 0.933 0.933 0.932 0.932 0.931
  210   211   212   213   214   215   216   217   218   219   220   221   222   223   224
0.931 0.931 0.930 0.930 0.930 0.929 0.929 0.928 0.928 0.928 0.927 0.927 0.926 0.926 0.926
  225   226   227   228   229   230   231   232   233   234   235   236   237   238   239
0.925 0.925 0.925 0.924 0.924 0.923 0.923 0.923 0.922 0.922 0.921 0.921 0.921 0.920 0.920
  240   241   242   243   244   245   246   247   248   249   250   251   252   253   254
0.919 0.919 0.919 0.918 0.918 0.917 0.917 0.917 0.916 0.916 0.915 0.915 0.915 0.914 0.914
  255   256   257   258   259   260   261   262   263   264   265   266   267   268   269
```

Autocorrelation with different lags

Observations :

1. When lag = 0 we get maximum autocorrelation of 1.
2. As the lag increases , the autocorrelation value decreases respectively.
3. We can see some peaks in the autocorrelation plot which indicates periodicity in the data.

Part 3: Correlation coefficient between temperature-light,temperature-relative humidity and relative humidity-light .

Observations:

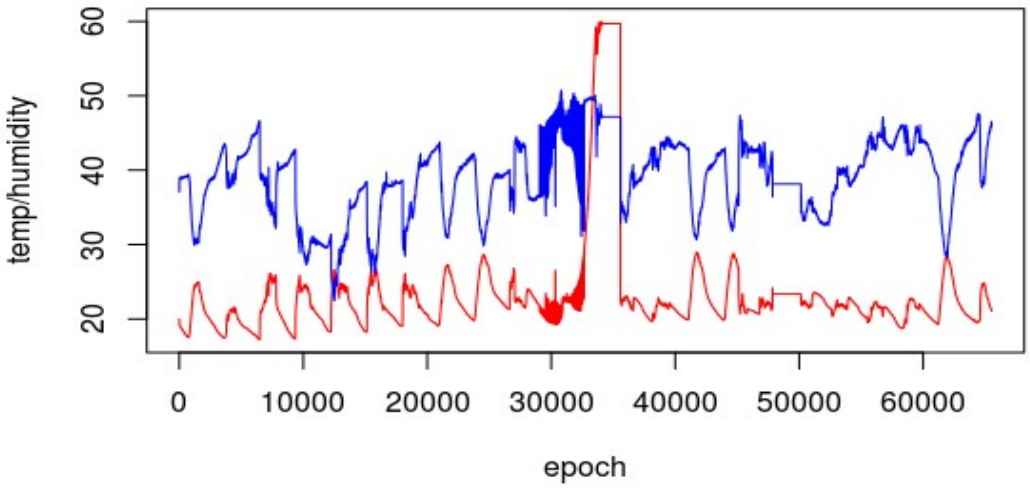
1. As we see from the plot of temperature and relative humidity , there is a strong negative correlation between them up to 30,000 entries.
2. Around 30,000th entry we see a sudden peak in the temperature data which might suggest a problem in the sensor or any mishap in the environment.
3. Even after this entry we find the data to be strongly negative correlated.

```
> light_humidity = cor(filled_data$light, filled_data$humidity)
> print(light_humidity)
[1] -0.2686249
> print(light_temp)
            temp
light 0.2602594
> temp_humidity = cor(filled_data$temp, filled_data$humidity)
> print(temp_humidity)
[1] 0.1648557
> light_humidity = cor(filled_data$light, filled_data$humidity)
> print(light_humidity)
[1] -0.2686249
```

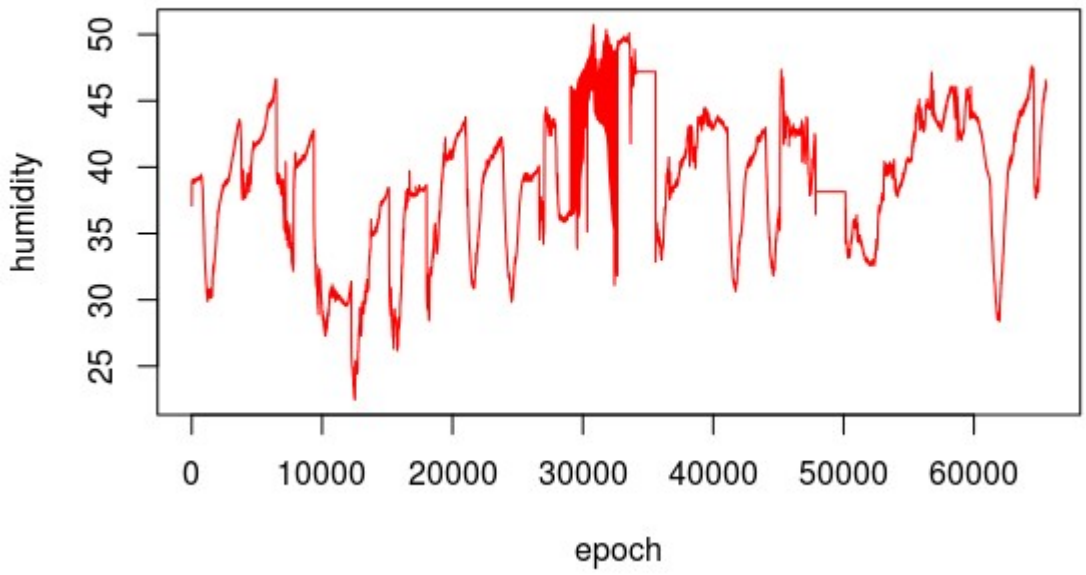
Correlation coefficients

```
> temp_humidity1 = cor(filled_data$temp[1:30000], filled_data$humidity[1:30000])
> print(temp_humidity1)
[1] -0.6387945
>
```

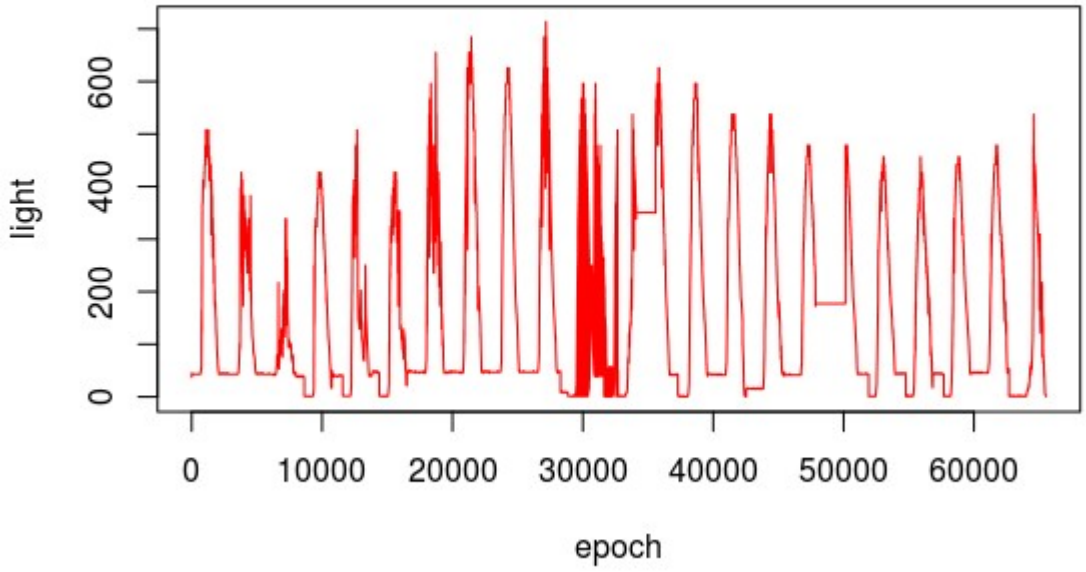
Correlation coefficient for first 30000 epochs between temperature and relative humidity



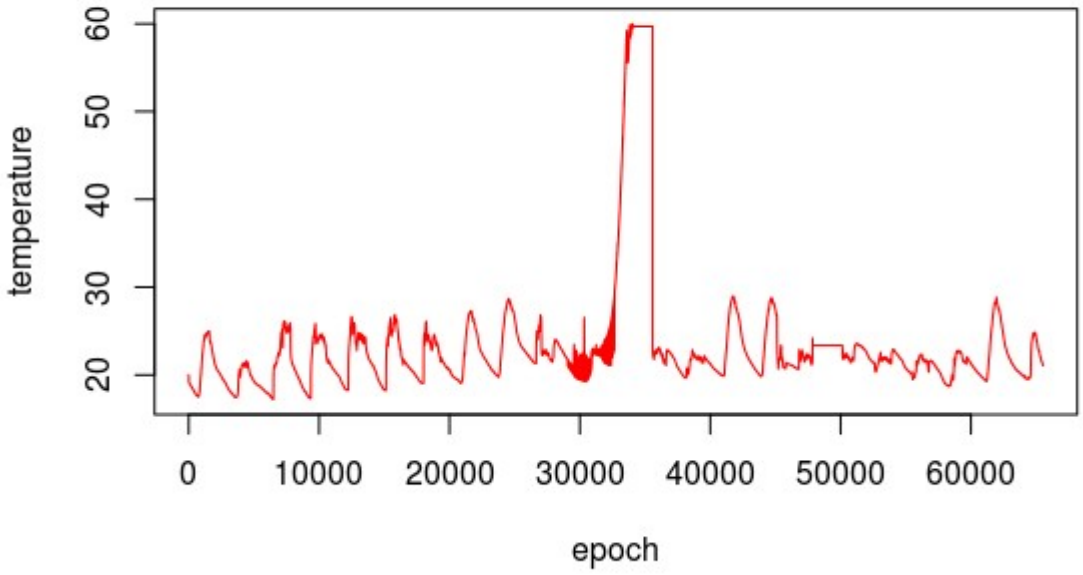
Temperature humidity plot to observe correlation. Red: Temperature Blue: Humidity



Humidity



Light



Temperature

Part 4: Correlation coefficient between temperature data of sensor 1 and sensor 6.

Observations:

1.According to the floor plan the two sensors(1 and 6)seem quite close spatially and hence we expect a higher correlation coefficient between them.

2. But we find the data to be less correlated with the coefficient value of .4085 ,which suggests that there might be a different environment near the sensors assuming the server room area and the lab area works on different temperature values.

```
> sensor_1_6 = cor(temp1, temp6)
> print(sensor_1_6)
          temp.y
temp.x 0.4085199
> sensor_1_50 = cor(temp11, temp50)
> print(sensor_1_50)
          temp.y
temp.x 0.6856734
```

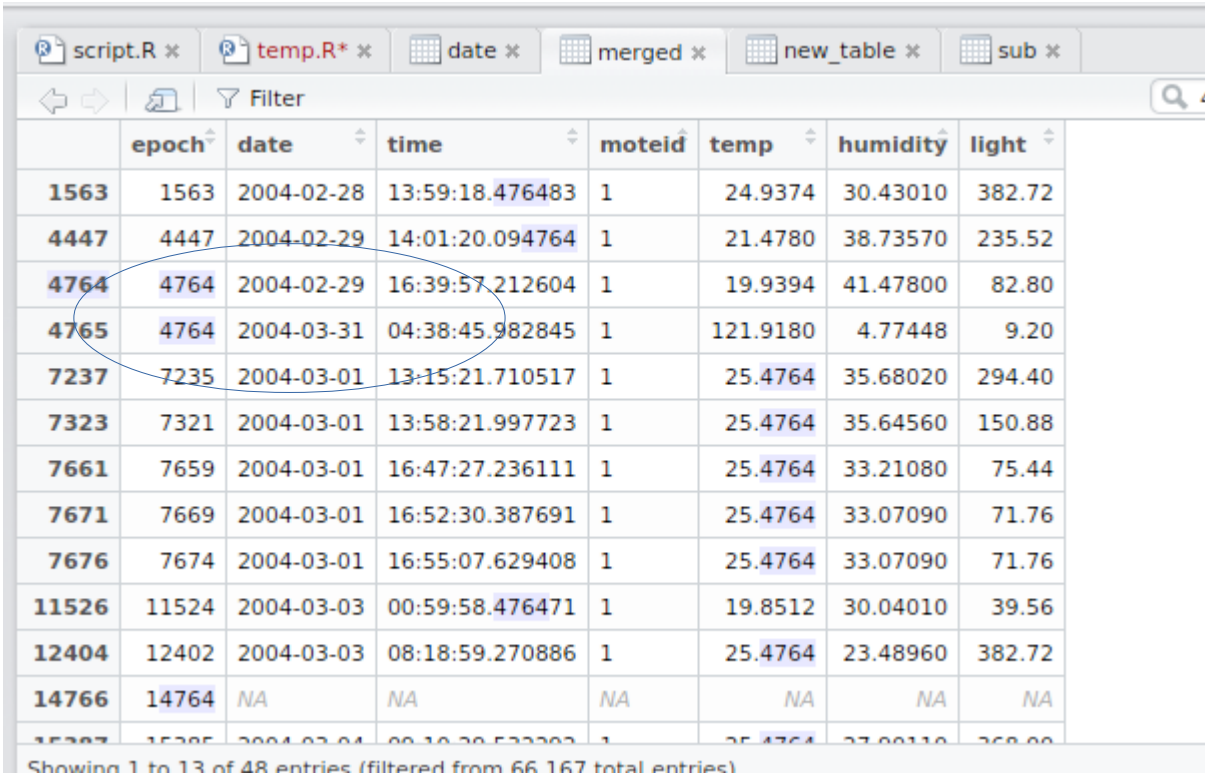
Correlation coefficient of temperature data between Sensor 1 and Sensor 6, and Sensor 1 and Sensor 50.

Part 5: Correlation coefficient between temperature data of sensor 1 and sensor 50.

Observations:
1.Much the same way as Part 4, Sensor 1 and Sensor 50 are far apart and are expected to be less correlated and the coefficient value is .6856.

The image above captures the coefficientvalue this part as well.

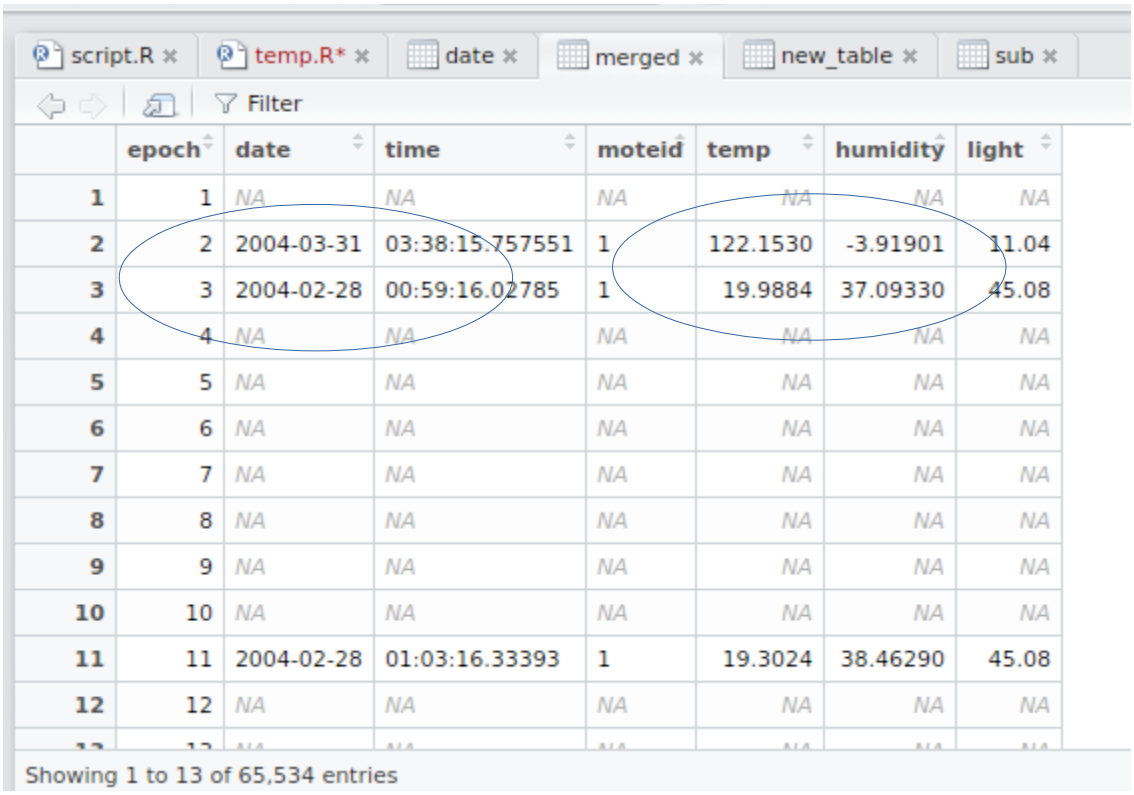
Few Extra Observations:
1. We found the epoch numbers to be repeating for the same sensor with different dates which is not supposed to happen because the numbers are monotonically increasing a per the Lab description.
633 such epochs found.
2.We found Epoch 2 for date 31 March and Epoch 3 for date 29 February, hence the epochs dont seem to be in order.
3. We found temperature at some epochs to be around 122 degrees and others around as low as -38 degrees. We considered these as outliers and tried to cleanse the data.



	epoch	date	time	moteid	temp	humidity	light
1563	1563	2004-02-28	13:59:18.476483	1	24.9374	30.43010	382.72
4447	4447	2004-02-29	14:01:20.094764	1	21.4780	38.73570	235.52
4764	4764	2004-02-29	16:39:57.212604	1	19.9394	41.47800	82.80
4765	4764	2004-03-31	04:38:45.982845	1	121.9180	4.77448	9.20
7237	7235	2004-03-01	13:15:21.710517	1	25.4764	35.68020	294.40
7323	7321	2004-03-01	13:58:21.997723	1	25.4764	35.64560	150.88
7661	7659	2004-03-01	16:47:27.236111	1	25.4764	33.21080	75.44
7671	7669	2004-03-01	16:52:30.387691	1	25.4764	33.07090	71.76
7676	7674	2004-03-01	16:55:07.629408	1	25.4764	33.07090	71.76
11526	11524	2004-03-03	00:59:58.476471	1	19.8512	30.04010	39.56
12404	12402	2004-03-03	08:18:59.270886	1	25.4764	23.48960	382.72
14766	14764	NA	NA	NA	NA	NA	NA

Showing 1 to 13 of 48 entries (filtered from 66,167 total entries)

Epoch Duplication



	epoch	date	time	moteid	temp	humidity	light
1	1	NA	NA	NA	NA	NA	NA
2	2	2004-03-31	03:38:15.757551	1	122.1530	-3.91901	11.04
3	3	2004-02-28	00:59:16.02785	1	19.9884	37.09330	45.08
4	4	NA	NA	NA	NA	NA	NA
5	5	NA	NA	NA	NA	NA	NA
6	6	NA	NA	NA	NA	NA	NA
7	7	NA	NA	NA	NA	NA	NA
8	8	NA	NA	NA	NA	NA	NA
9	9	NA	NA	NA	NA	NA	NA
10	10	NA	NA	NA	NA	NA	NA
11	11	2004-02-28	01:03:16.33393	1	19.3024	38.46290	45.08
12	12	NA	NA	NA	NA	NA	NA

Showing 1 to 13 of 65,534 entries

Epoch Order Mismatch and Outliers Identification