

# Eerste nadere analyse van de sensordata ingezet in het vuurwerkexperiment.

RIVM, Peter Zandveld -8 maart 2017

## 1. sensor data

De data van alle sensoren is in een Access database gezet. In de database staat verder:

- LML pm10 meetdata
- Meteo data
- Enquête gegevens.

Deze gegevens kunnen op allerlei manieren worden gekoppeld. Hieronder de eerste resultaten. Het betreft oriënterend onderzoek. In een later stadium kan besloten worden bepaalde groepen data af te keuren of apart te behandelen.

## 2.Koppeling met enquête data.

Doel van deze analyse is te onderzoeken wat de invloed van de ophanging/positionering van de sensor op het signaal is.

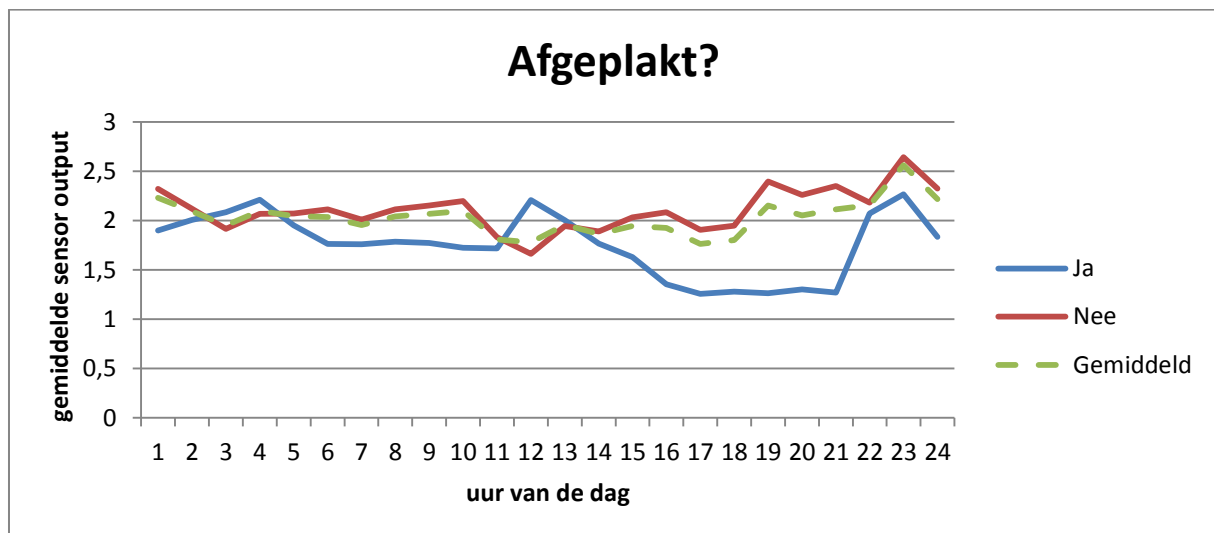
Op de meetdata zijn 2 bewerkingen uitgevoerd:

- Met een eenvoudig filter algoritme is een deel van de scherpe pieken weg gefilterd. (Zie \$4 )
- Dit signaal is uur-gemiddeld.

Vervolgens is de meetdata van alle sensoren zonder verdere selectie gekoppeld aan de enquête gegevens

### 2.1 Is de sensor afgeplakt?

Uit de literatuur is bekend dat er storing door strooilicht kan ontstaan door de opening aan de bovenkant van de sensor. Een deel van de deelnemers heeft de opening afgeplakt. In de enquête is aangegeven wie dit heeft gedaan. Er zijn dus twee groepen sensors. Het gemiddelde per uur van de dag is bepaald:



Respondenten

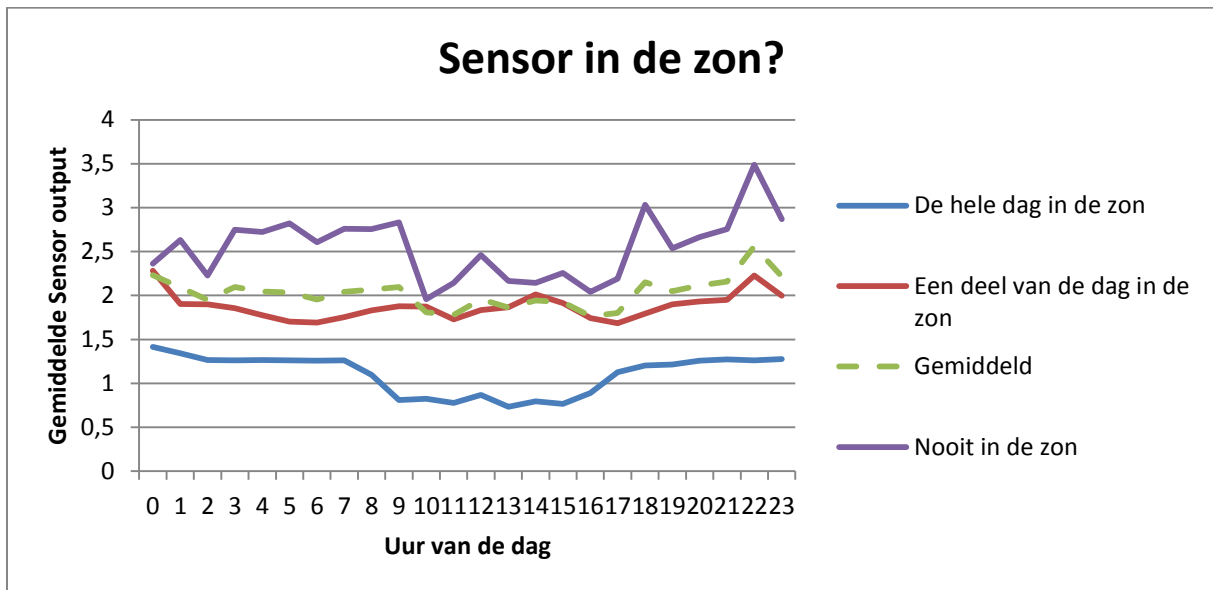
Afgeplakt: 8

Niet afgeplakt 39

Niet ingevuld: 1

Voorlopige conclusie: Er is een effect aan het eind van de dag. Verklaring?

## 2.2. Hangt de sensor in de zon

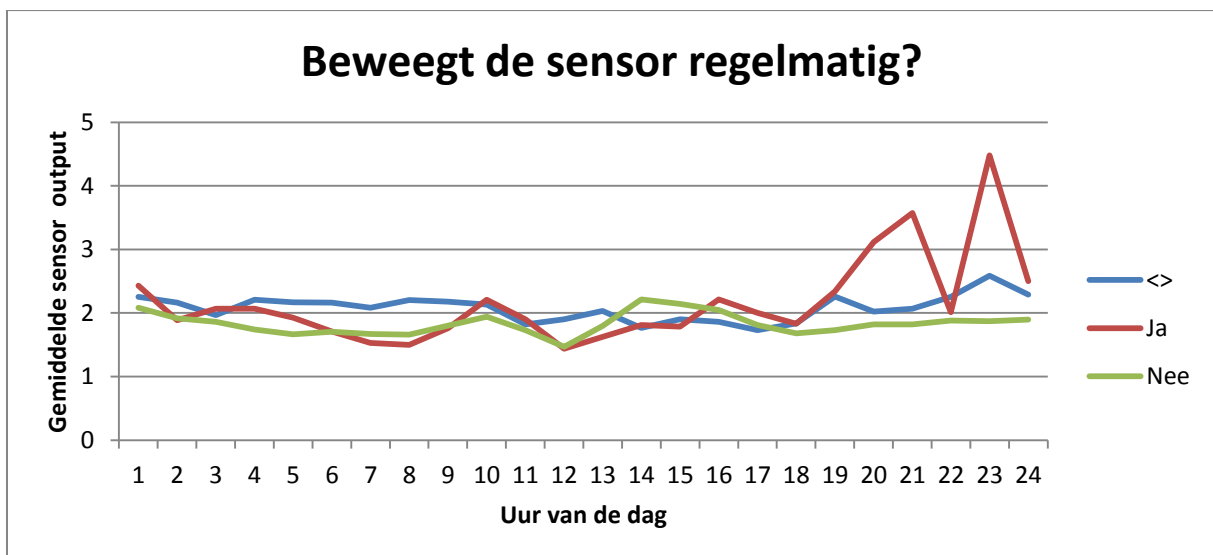


Respondenten

Hele dag in de zon:	4
Een deel van de dag in de zon:	29
Nooit in de zon:	14
Geen antwoord	1

Conclusie: Dit lijkt een belangrijke parameter, maar waarom ook in de nacht verschillen, offsets?

## 2.3. Is de sensor gevoelig voor beweging?

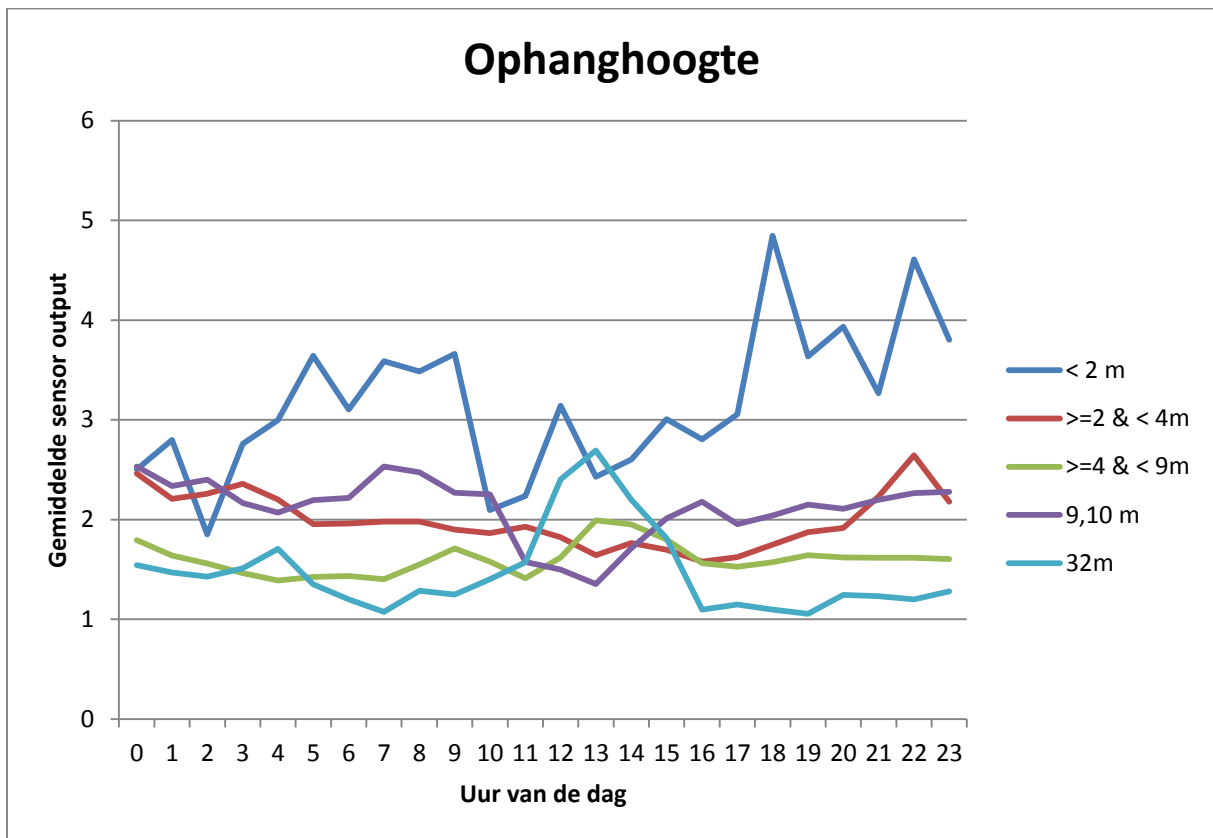


Respondenten

Geen antwoord	33
Ja	6
Nee	9

Conclusie: deze vraag is waarschijnlijk te lastig, en te weinig precies. Vermoedelijk een kleine invloed op het uurgemiddelde

## 2.4 Hoe hoog hangt de sensor?



Respondenten

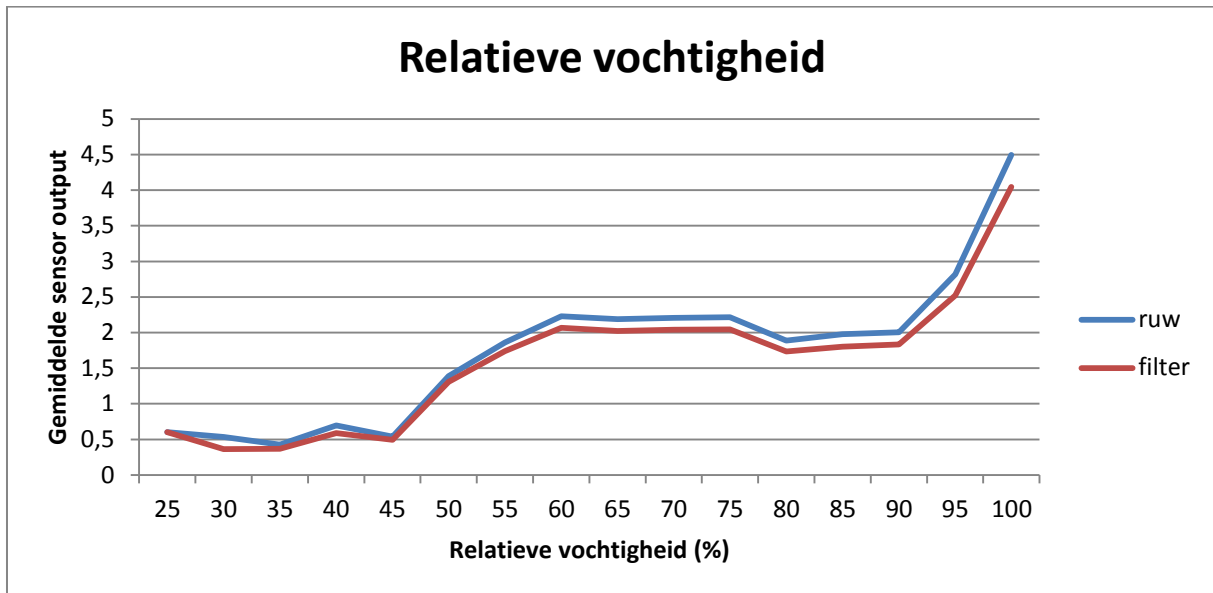
< 2 meter	8
>=2 < 4 meter	22
>=4 < 9 meter	14
>=9 <= 10 meter	2
32 meter	1

Conclusie: laag opgehangen sensoren meten een hoger signaal

### 3 Koppeling met meteo data

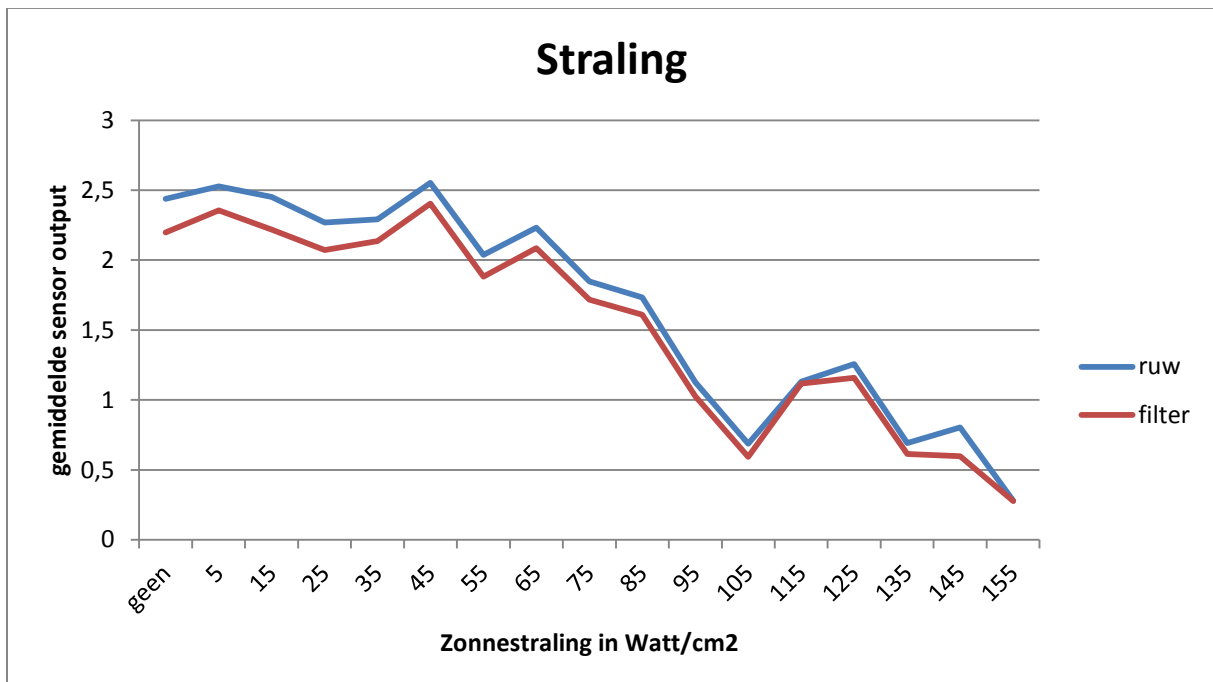
Voor iedere sensor is bepaald welk meteo station het dichtst bij ligt. De sensor is aan dit meteostation gekoppeld.

#### 3.1 Relatieve vochtigheid



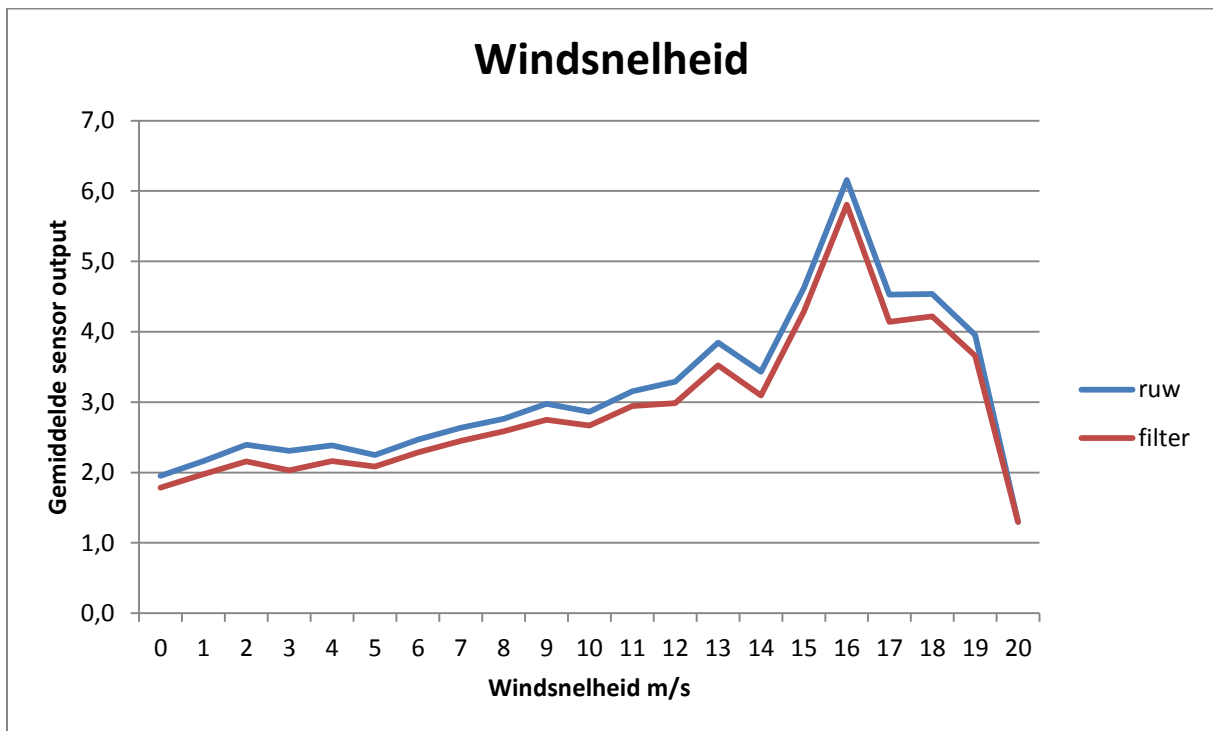
Conclusie: Relatieve vochtigheid is een belangrijke parameter!

#### 3.2 Zon



Conclusie: straling van de zon is een belangrijke parameter

### 3.3 Windsnelheid



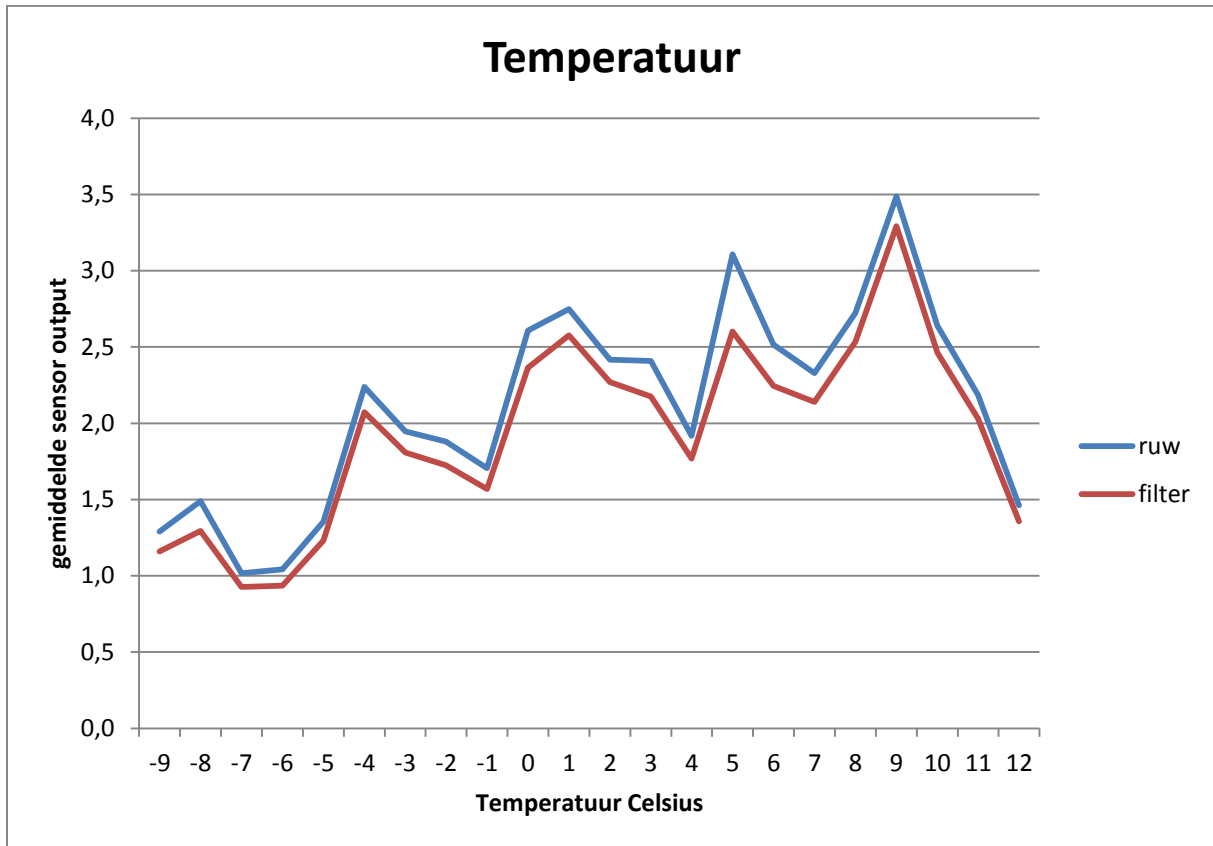
Indien je voornamelijk een lokale bron zou bemeten verwacht je een lager signaal met oplopende windsnelheid. Misschien kijken we naar opwaaiend stof, of beweging van de sensor.

NB niet te veel aandacht besteden aan de piek bij 16 m/s. Er is weinig data bij de hoge windsnelheden.

ws	ruw	filter	freq
0	2,0	1,8	1110
1	2,2	2,0	7260
2	2,4	2,2	13134
3	2,3	2,0	11411
4	2,4	2,2	10006
5	2,3	2,1	7283
6	2,5	2,3	4842
7	2,6	2,4	2813
8	2,8	2,6	1629
9	3,0	2,7	1163
10	2,9	2,7	834
11	3,2	2,9	518
12	3,3	3,0	254
13	3,8	3,5	159
14	3,4	3,1	100
15	4,6	4,3	56
16	6,2	5,8	58

17	4,5	4,1	45
18	4,5	4,2	18
19	4,0	3,7	11
20	1,3	1,3	3
21	2,4	2,4	1

### 3.4 Temperatuur



Ook de temperatuur is een belangrijke parameter

T	ruw	filter	n
-9	1,3	1,2	15
-8	1,5	1,3	39
-7	1,0	0,9	181
-6	1,0	0,9	614
-5	1,4	1,2	817
-4	2,2	2,1	2337
-3	1,9	1,8	2409
-2	1,9	1,7	2827
-1	1,7	1,6	3390
0	2,6	2,4	7222

1	2,7	2,6	4417
2	2,4	2,3	4586
3	2,4	2,2	5509
4	1,9	1,8	5325
5	3,1	2,6	5889
6	2,5	2,2	5736
7	2,3	2,1	3868
8	2,7	2,5	1865
9	3,5	3,3	1443
10	2,6	2,5	803
11	2,2	2,0	809
12	1,5	1,4	324
13	0,2	0,2	8
14	3,4	3,4	1

#### 4. Het filter algoritme.

Het filter is een blokje van 7 opeenvolgende metingen dat over de tijdreeks schuift. De meetwaarde van het middelste tijdstip wordt vergeleken met het gemiddelde van de 6 omliggende waarden.

```

for ( itijd=3;itijd<ntijd-3;itijd++){
    gefilterd[itijd] = ruwe_meting_5min[itijd];
    m0 = ruwe_meting_5min[itijd-3];
    m1 = ruwe_meting_5min [itijd-2];
    m2 = ruwe_meting_5min [itijd-1];
    m3 = ruwe_meting_5min [itijd+1];
    m4 = ruwe_meting_5min [itijd+2];
    m5 = ruwe_meting_5min [itijd+3];
    gem = (m0 + m1 + m2 + m3 + m4 + m5)/6;
    if (ruwe_meting_5min[itijd]/gem > 3){
        gefilterd[itijd] = gem;
    }
}

```