

First we define a figure hook:

This simple code provided to test volume sensitivity of Olfactory Receptors.
 Ref article: Olfactory receptors are sensitive to molecular volume of odorants Authors: Majid Saberi, Hamed Seyed-allaei doi: <http://dx.doi.org/10.1101/013516>
 January 6, 2015

```
> rm ( list=ls() )
>
> #setwd("~/") #Path of files location setting
```

At first, we extract molecular volumes and responses from supplementary data.

```
> DoOR <- as.matrix( read.csv("proposed_odorants.csv") )
> rownames( DoOR ) <- DoOR[, 1]
> DoOR <- DoOR[, 2:33]
> mode( DoOR ) <- "numeric"
```

DoOR represent response of odor receptors(columns) to odorants(rows), for example:

```
> DoOR[ 1:5 , 1:5 ]
```

	Or10a	Or1a	Or22a	Or22b	Or23a
7732-18-5	NA	0.024878414	NA	NA	NA
7664-41-7	0.06296634	NA	0.10187126	NA	NA
124-38-9	0.05848141	0.007108118	0.06253786	0	0.023889161
67-56-1	0.05113509	NA	0.11791586	NA	0.013155924
64-18-6	0.12837361	NA	0.05348513	NA	0.002192654

```
> odorants <- as.matrix( read.csv("odorants.csv") )
> volume <- as.numeric( odorants[, 4] )
> names( volume ) <- odorants[, 1]
```

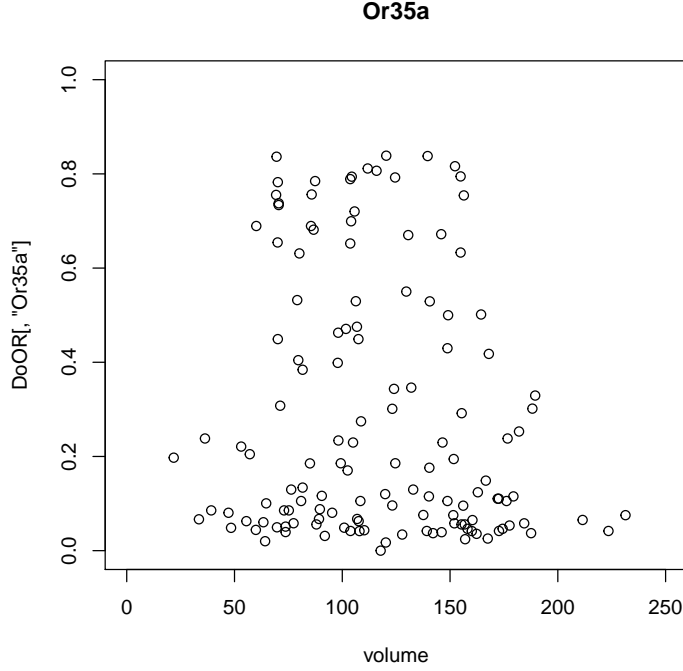
We have vector of molecular volumes:

```
> volume[1:10]
```

7732-18-5	7664-41-7	124-38-9	67-56-1	64-18-6	75-07-0	79-09-4	64-17-5
19.8	21.8	33.5	36.3	39.2	47.2	48.4	53.1
64-19-7	124-40-3						
55.5	57.1						

We can plot response of ORs versus molecular volumes for every receptors:

```
> plot(volume, DoOR[ , "Or35a" ], ylim=c(0,1), xlim=c(0,250), main="Or35a" )
```



To find out $f(v)$, volume and flexibility of binding packet of ORs we define useful functions. Weighted mean and weighted standard deviation functions:

```
> wt.mean <- function (x, wt)                                #Weighted mean function
+ {
+   s = which(!is.na(x * wt))
+   wt = wt[s]
+   x = x[s]
+   return(sum(wt * x)/sum(wt))
+ }
> wt.sd <- function (x, wt)                                    #Weighted sd function
+ {
+   s = which(!is.na(x + wt))
+   wt = wt[s]
+   x = x[s]
+   xbar = wt.mean(x, wt)
+   return(sqrt( sum(wt * (x^2))/sum(wt) - xbar^2 ) )
+ }
```

Gaussian function:

```
> gauss <- function (x, mu=0, sig=1)                         #Gaussian function
+ {
+   return( exp( -(x - mu)^2 / (2 * sig^2) ) )
+ }
```

$$f_n(v) = e^{-\frac{(v-v_n)^2}{2\sigma_n^2}}, \quad (1)$$

$$\frac{1}{\sigma_n^2} = \frac{1}{\sigma_{h_n}^2} - \frac{1}{\sigma_{g_n}^2} \quad (2)$$

âĖĖ

$$v_n = \sigma_n^2 \left(\frac{\mu_{h_n}}{\sigma_{h_n}^2} - \frac{v_{g_n}}{\sigma_{g_n}^2} \right). \quad (3)$$

âĖĖ

```
> parameter <- function(OR){
+   response = DoOR[, OR]                                #OR's response selection
+   sig_g = sd(volume, na.rm=T)                            #SD of g(v)
+   mu_g = mean(volume, na.rm=T)                          #MEAN of g(v)
+   sig_h = wt.sd(volume, response)                       #SD of h(v), 10th formula
+   mu_h = wt.mean(volume, response)                      #MEAN of h(v), 9th formual
+
+   sig_f = sqrt( sig_g^2 * sig_h^2 / (sig_g^2 - sig_h^2) )  #SD of f(v)
+   mu_f = sig_f^2 * (mu_h/sig_h^2 - mu_g/sig_g^2)         #MEAN of f(v)
+ return(c(mu_f, sig_f))
+ }
```

we see volume sensitivity of ORs. Scatterplot shows response versus molecular volume and f(v) lines as volume dependancy or volume profile of OR.

below we show receptors which their responses significantly are dependent to molecular volume and validated by bootstrapping test.

```
> colnames(DoOR)

[1] "Or10a" "Or1a"  "Or22a" "Or22b" "Or23a" "Or24a" "Or2a"  "Or30a" "Or33a"
[10] "Or35a" "Or42a" "Or42b" "Or43a" "Or43b" "Or45b" "Or46a" "Or47a" "Or49a"
[19] "Or49b" "Or59a" "Or59b" "Or67a" "Or67b" "Or67c" "Or71a" "Or85a" "Or85b"
[28] "Or85c" "Or85f" "Or94b" "Or98a" "Or9a"
```

Everyone can choose one of them and see volume sensitivity of OR. Scatterplot shows response versus molecular volume and f(v) lines as volume dependancy or volume profile of OR.

```
> OR = "Or35a"
> para <- parameter(OR)
> mu_f <- para[1]
> sig_f <- para[2]
> f_v = gauss(volume, mu=mu_f, sig=sig_f)                #Making f(v) as a gaussian function
> plot(volume, DoOR[,OR], ylim=c(0,1), xlim=c(0,250) , ylab="response" , main=OR , col="red")
> lines(volume, f_v )
```

