

# Dr. Waldrop's Odor-capture Notebook

## Intro

```
## Loading required package: viridisLite
```

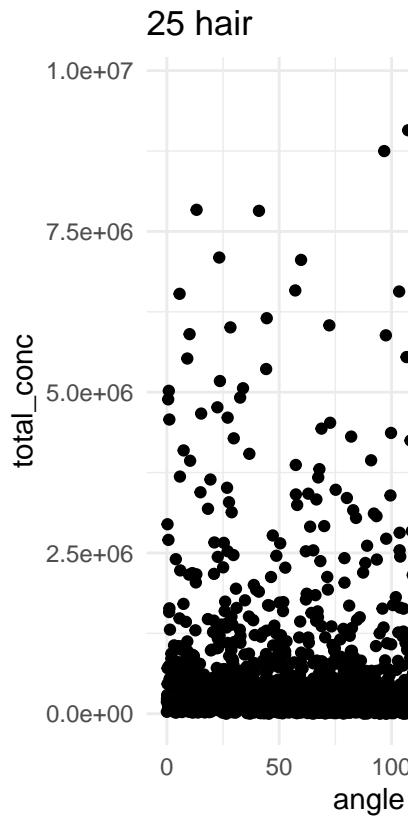
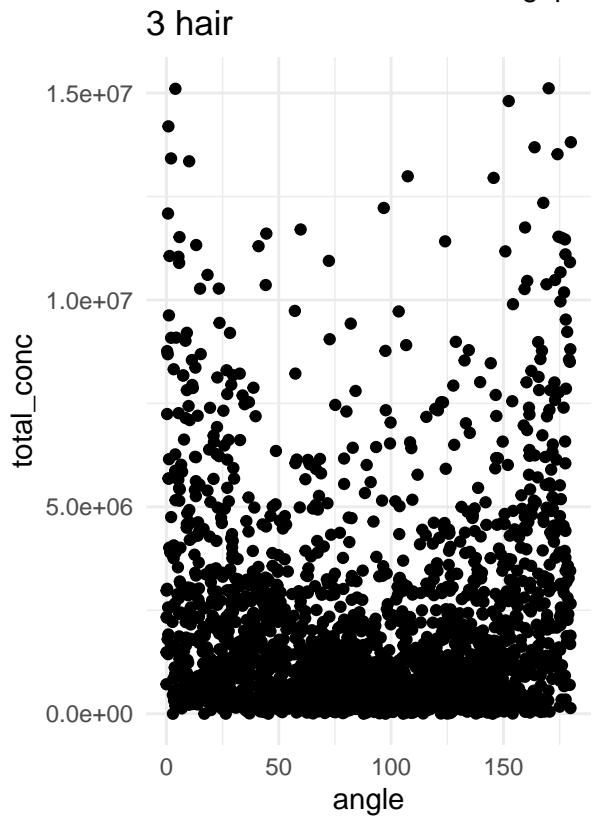
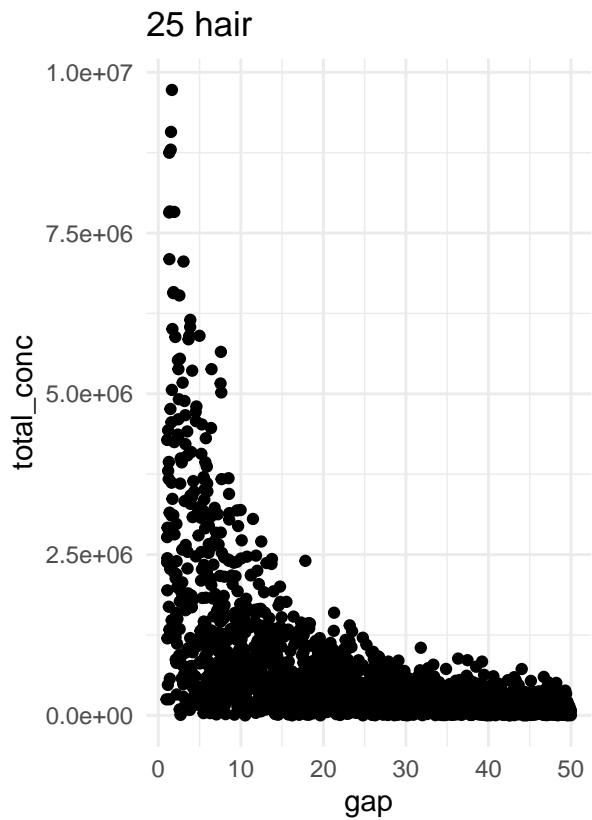
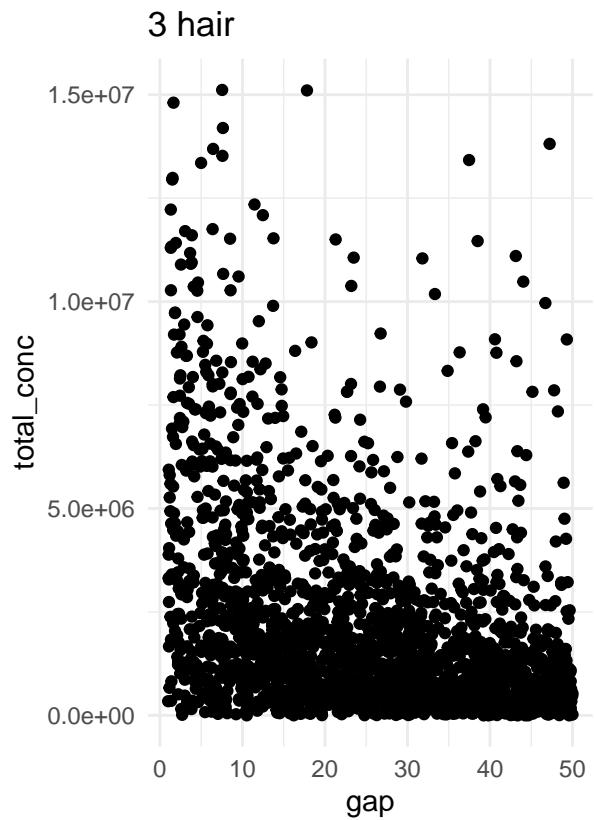
The entcode master project is a computational fluid dynamic simulation of odor capture by a chemosensory hair array consisting of three hairs and a supporting, non-chemosensory antenna. The simulations are in two dimensions, representing a cross-section of the array, and represent arrays with either 3 or 25 hairs. The project consists of setup for 2,000 individual simulations, each representing a unique combination of the following parameters:

- angle: Angle of array to oncoming flow: 0 to 180 degrees
- gap: Gap-to-hair-diameter ratio of the gaps between chemosensory hairs to their widths: 1 to 50
- ant: Antenna-to-hair-diameter ratio of the diameter of a chemosensory hair to the diameter of the supporting antenna: 1 to 50
- dist: Distance of the center hair from the antenna: 0.001 to 0.02 m
- re: Reynolds number: 0.1 to 10
- diff\_coef: Odor diffusion coefficient:  $1e-9$  to  $1e-4 \text{ m}^2 \text{ s}^{-1}$
- stink\_width: Width of the initial odor stripe: 0.5 to 5 % of the domain width
- init\_conc: Initial odor concentration:  $1e3$  to  $1e7$  (no units)

Total concentration captured (total\_conc) is a raw concentration number captured by the array only normalized by the capture area. Normal concentration captured (norm\_conc) is normalized by the cmax value in the individual simulation.

## Preliminary Results

**Total concentration graphs** Plotting gap width and total concentration.

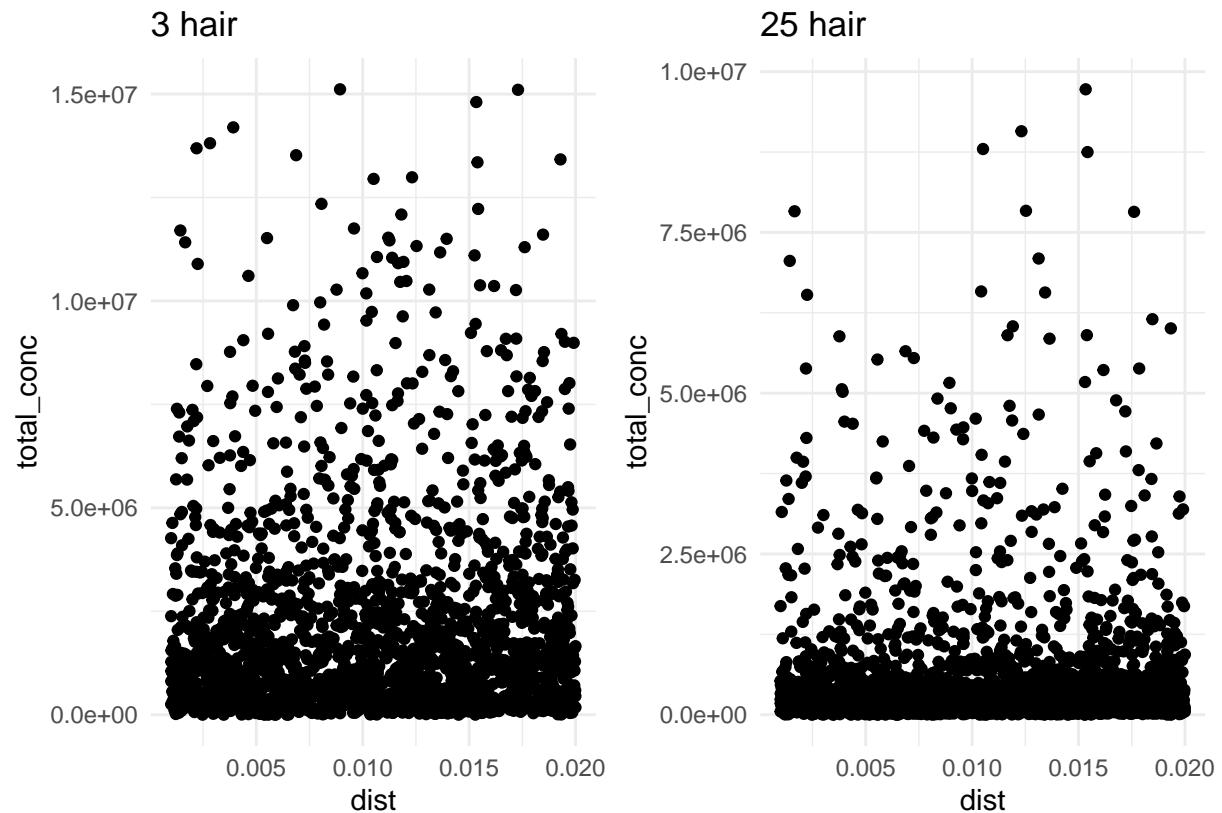


Plotting angle and total concentration

Plotting antenna width and total concentration

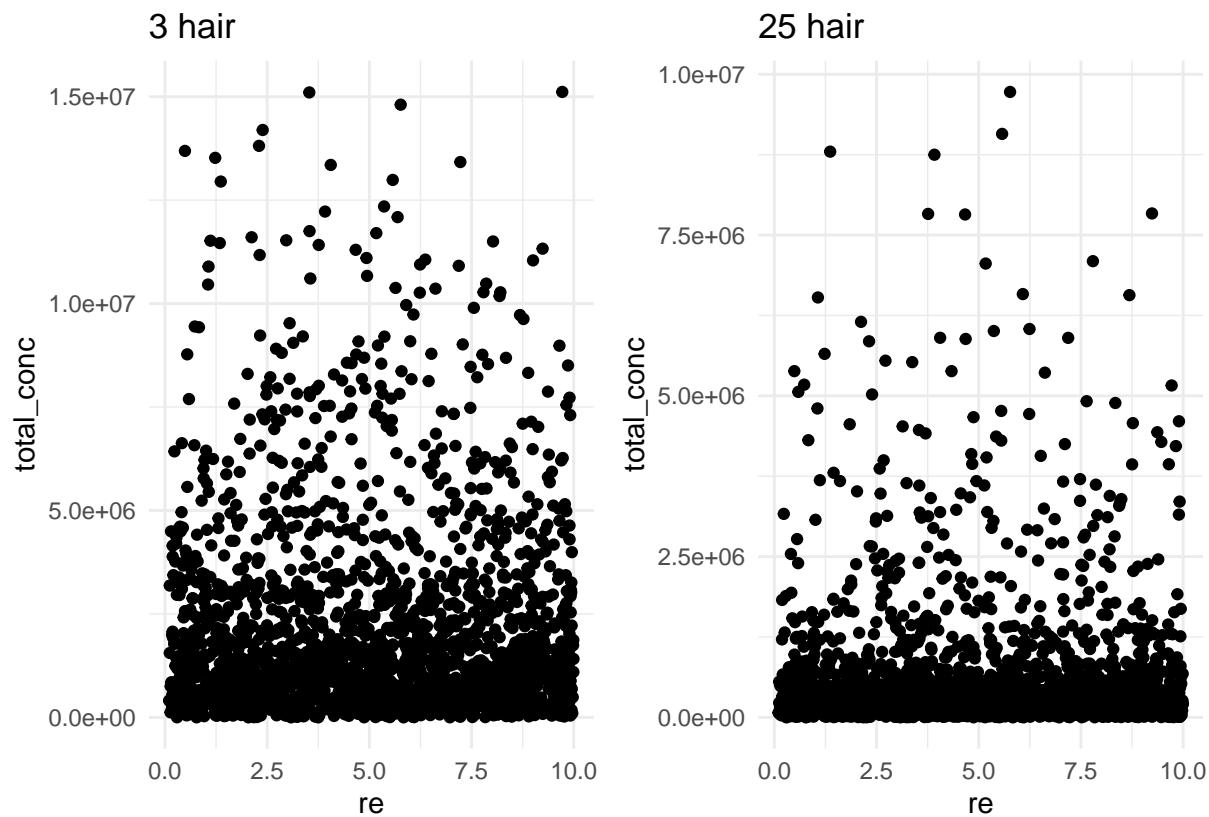
Plotting distance from antenna and total concentration

```
plot_parameter(three_dat, twofive_dat, "dist", "total_conc")
```



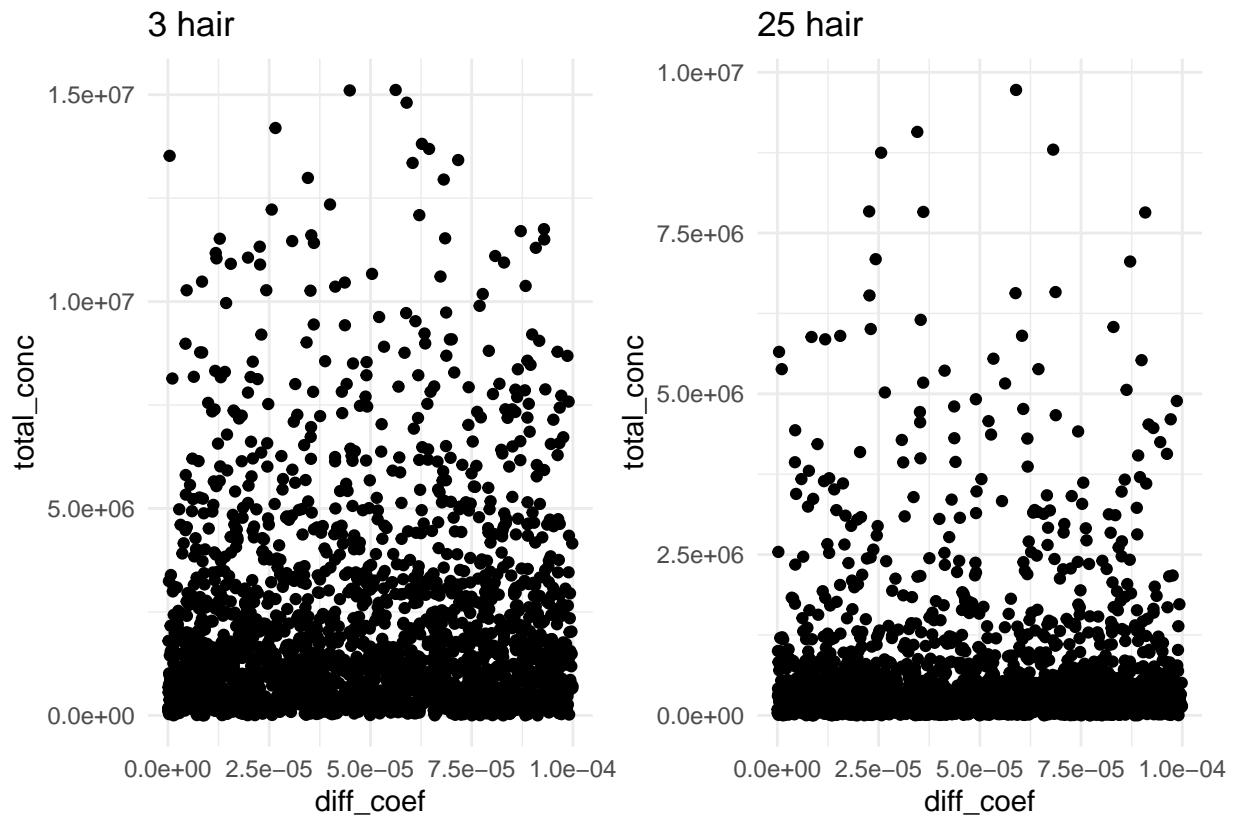
Plotting Reynolds number and total concentration

```
plot_parameter(three_dat, twofive_dat, "re", "total_conc")
```



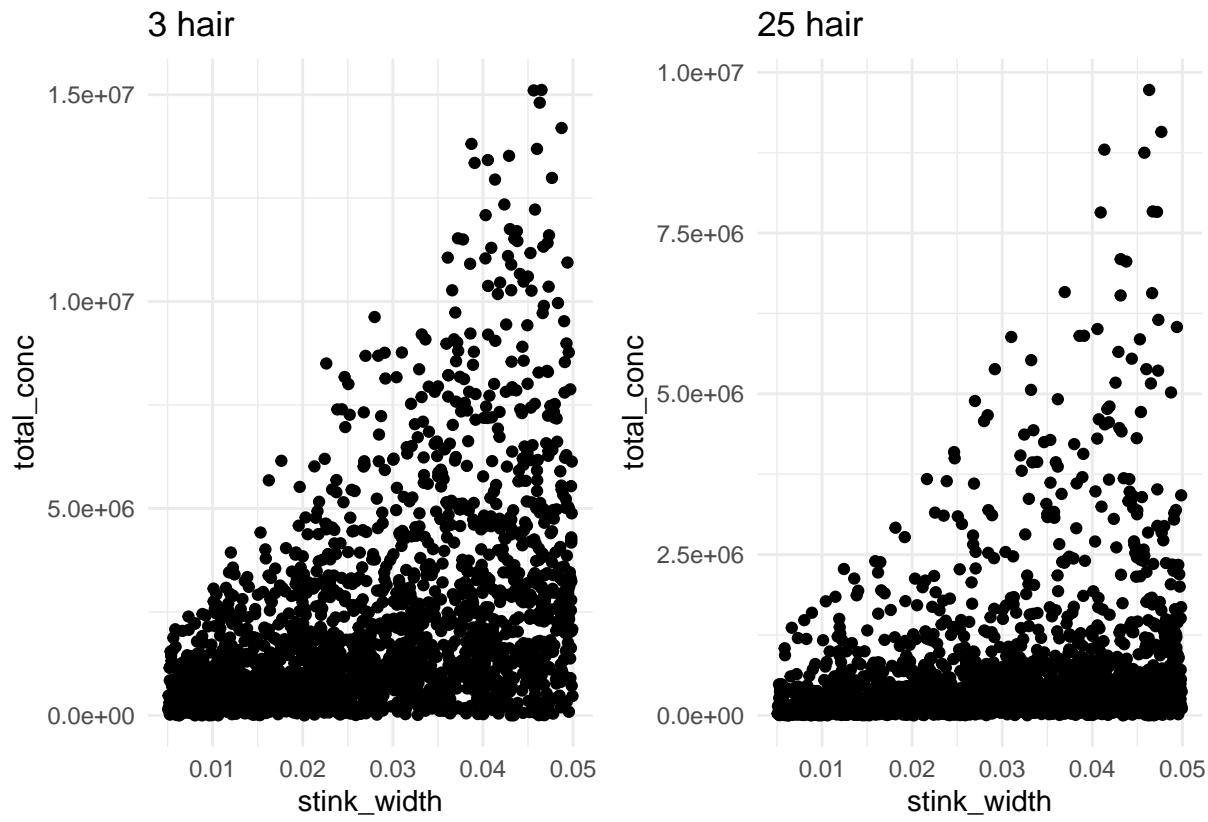
Plotting diffusion coefficient and total concentration

```
plot_parameter(three_dat, twofive_dat, "diff_coef", "total_conc")
```



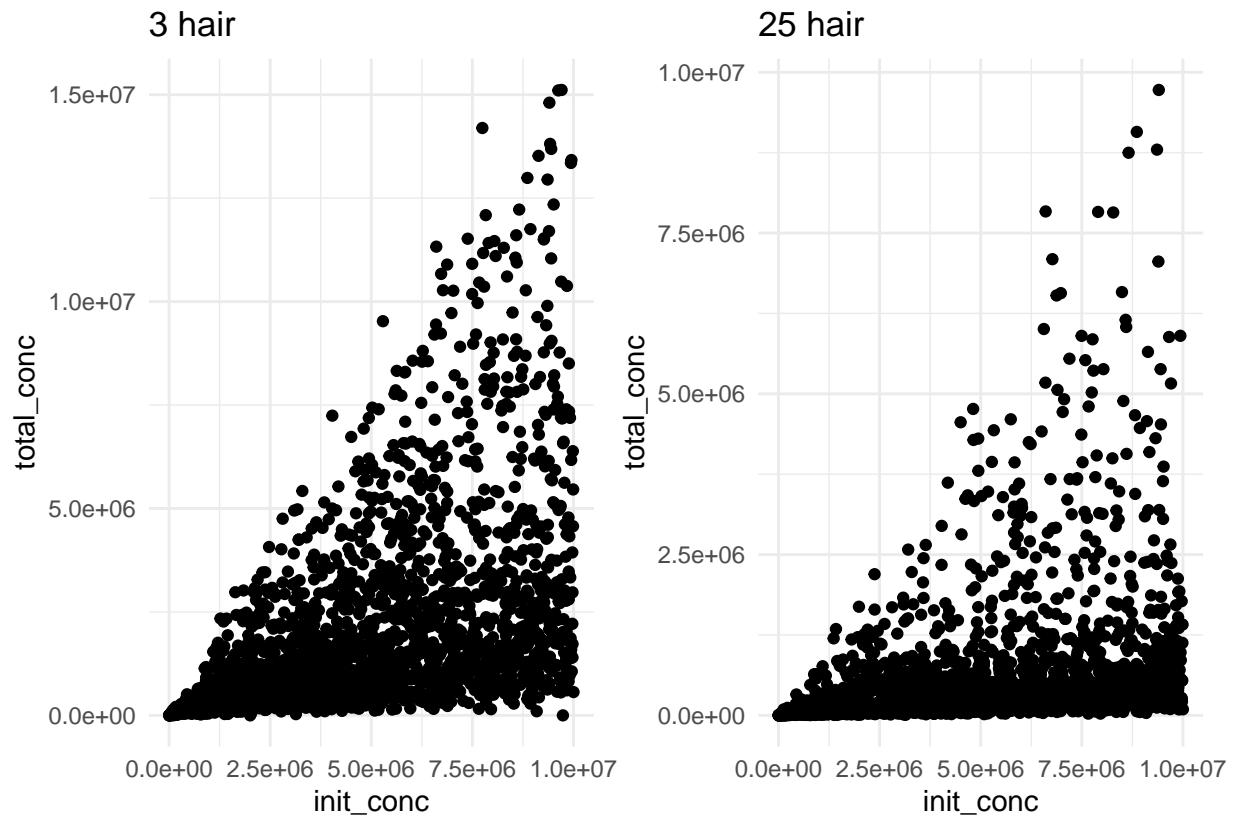
Plotting stripe width and total concentration

```
plot_parameter(three_dat, twofive_dat, "stink_width", "total_conc")
```



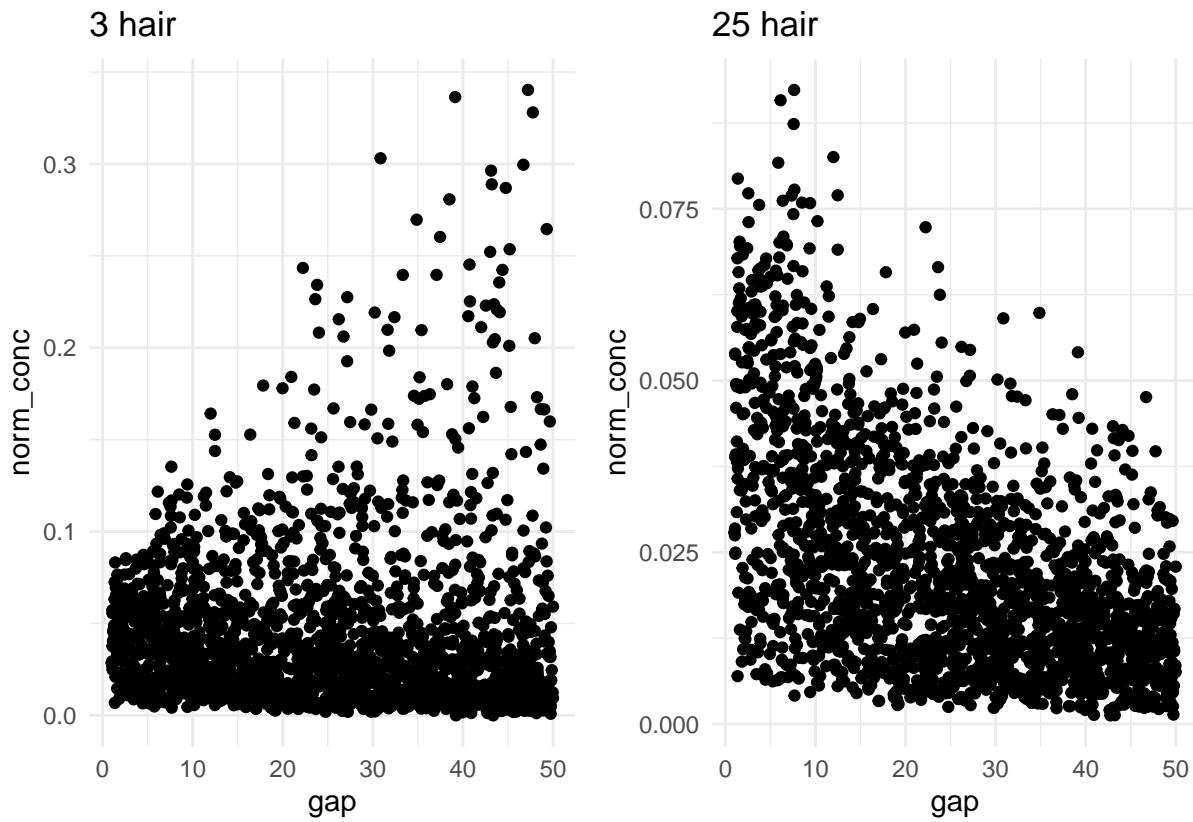
Plotting initial concentration and total concentration

```
plot_parameter(three_dat, twofive_dat, "init_conc", "total_conc")
```



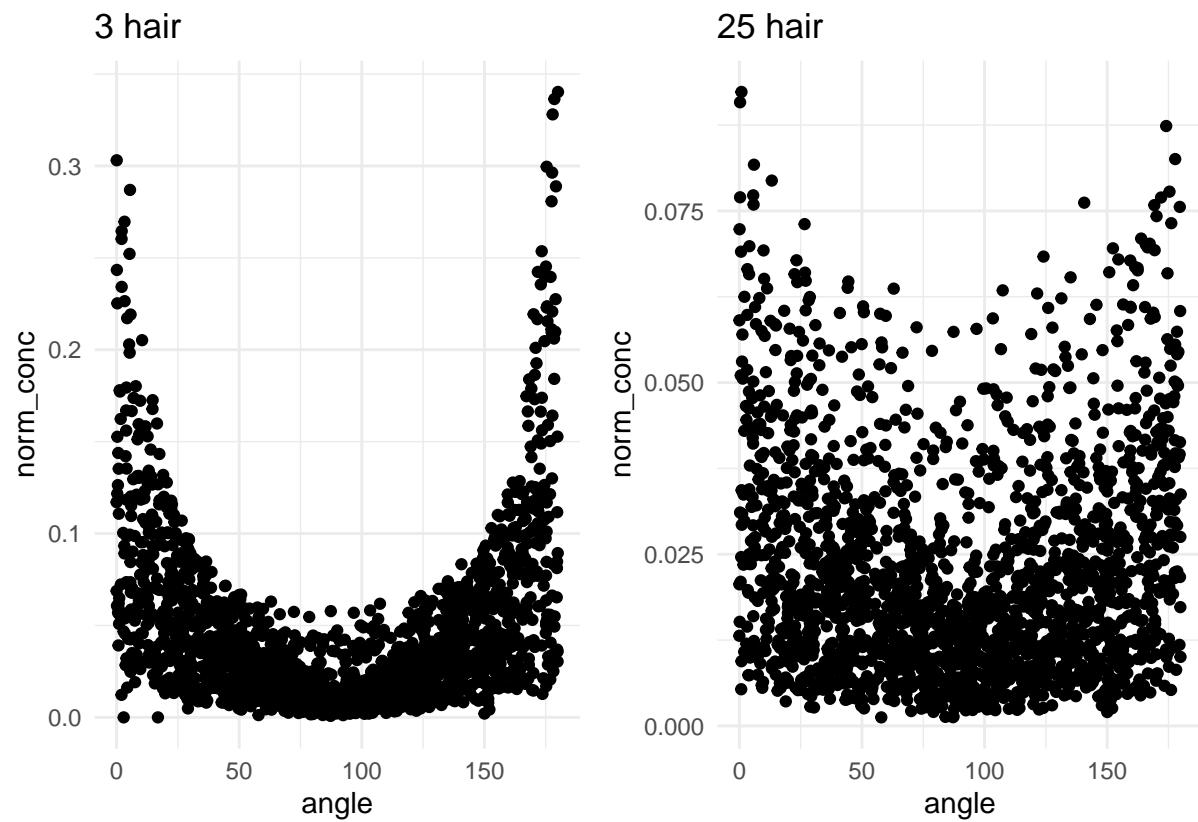
**Normalized concentration graphs** Plotting gap width and normalized concentration.

```
plot_parameter(three_dat, twofive_dat, "gap", "norm_conc")
```



Plotting angle and normalized concentration.

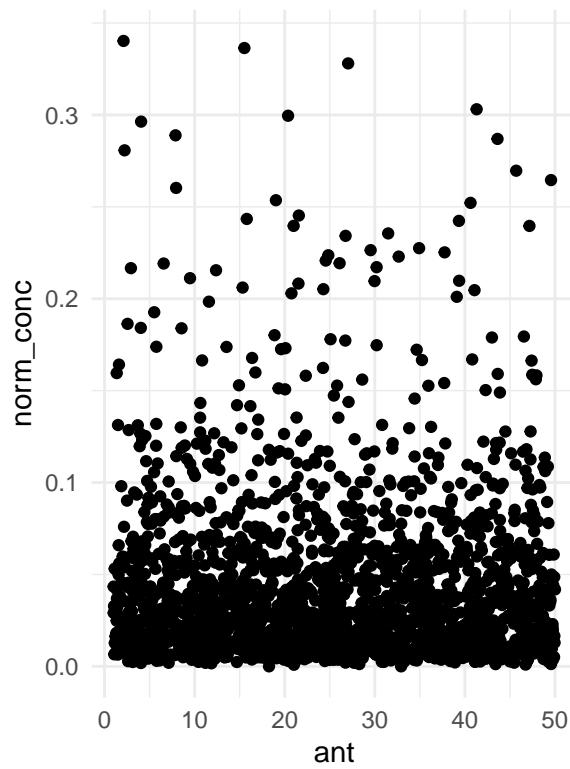
```
plot_parameter(three_dat, twofive_dat, "angle", "norm_conc")
```



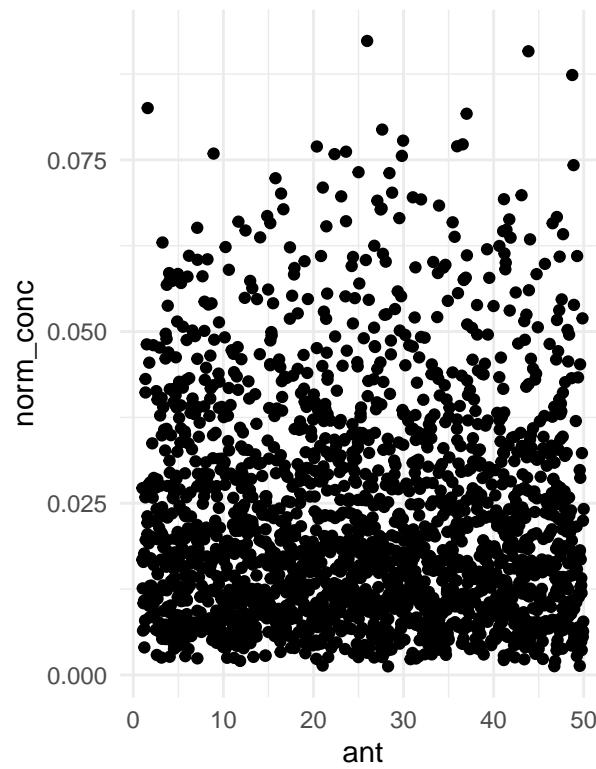
Plotting antenna width and normalized concentration.

```
plot_parameter(three_dat, twofive_dat, "ant", "norm_conc")
```

3 hair



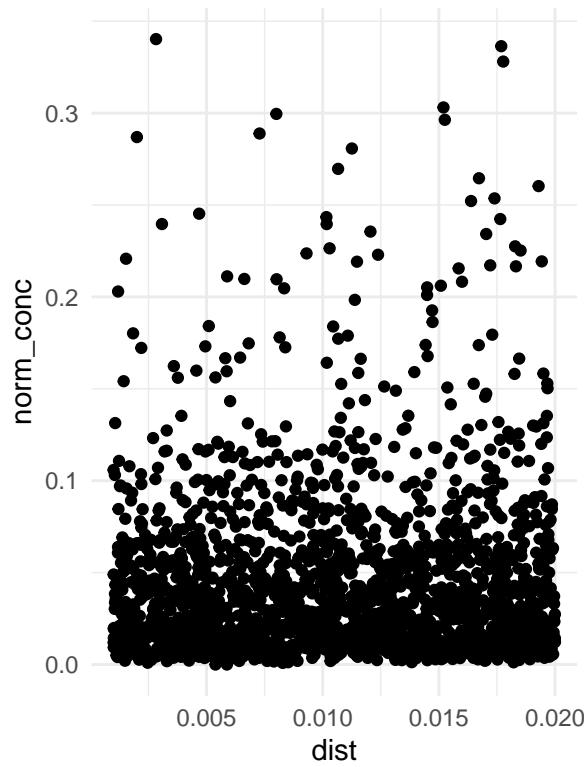
25 hair



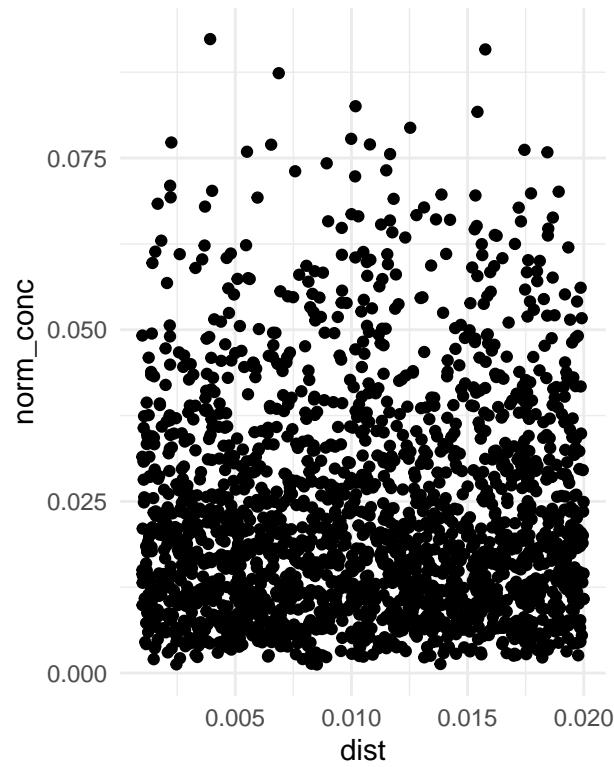
Plotting distance from antenna and normalized concentration.

```
plot_parameter(three_dat, twofive_dat, "dist", "norm_conc")
```

3 hair

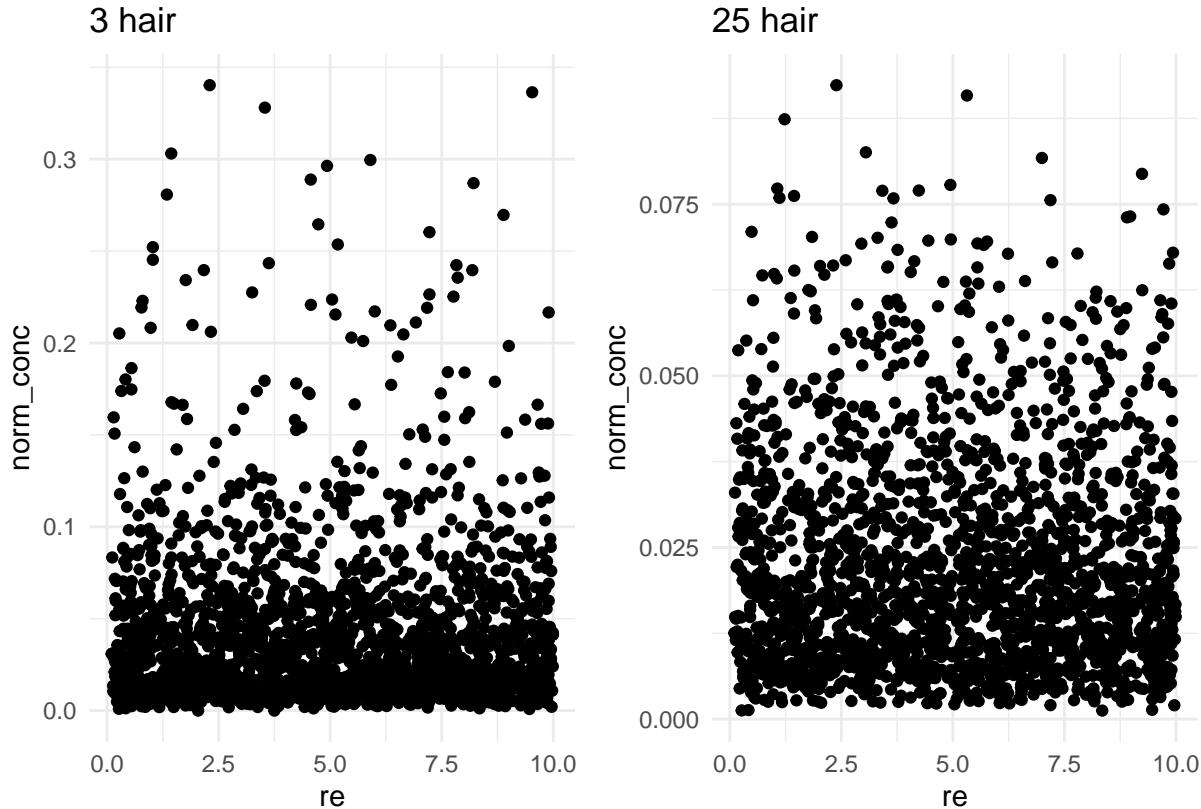


25 hair



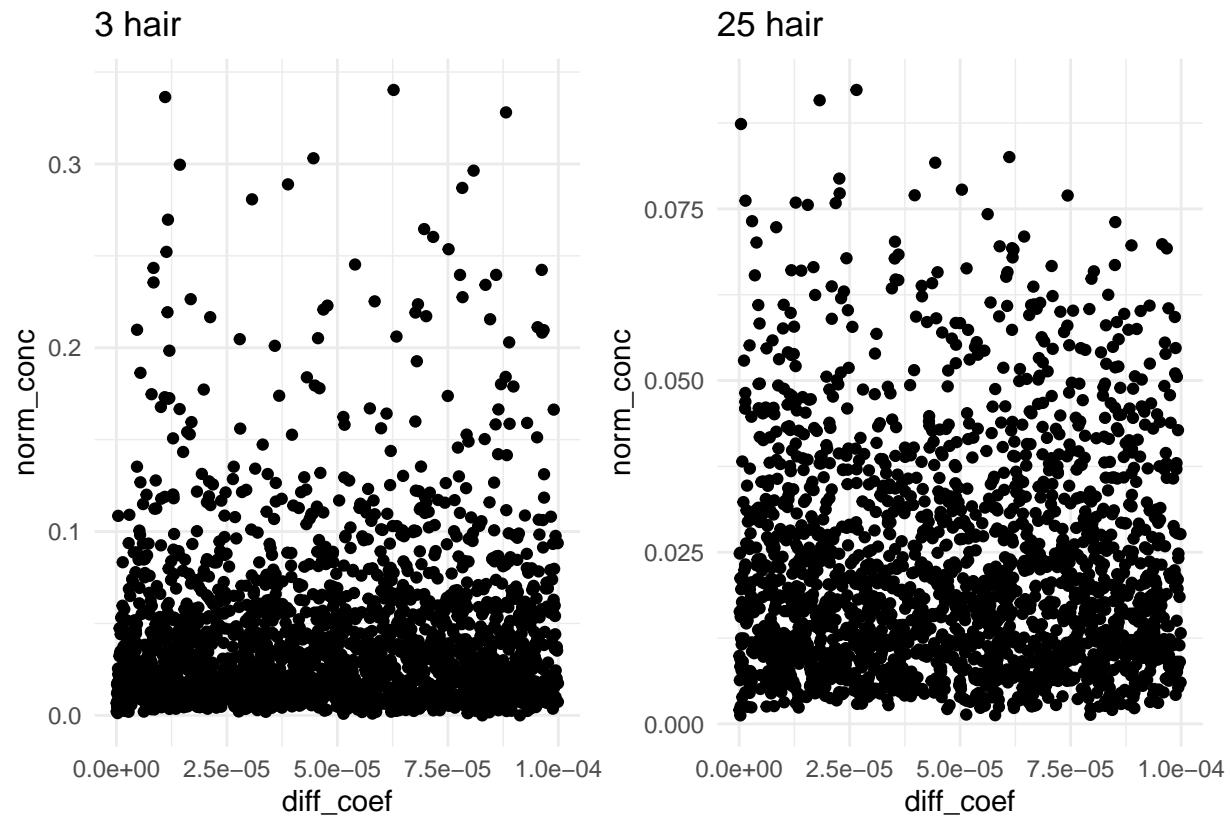
Plotting Reynolds number and normalized concentration.

```
plot_parameter(three_dat, twofive_dat, "re", "norm_conc")
```



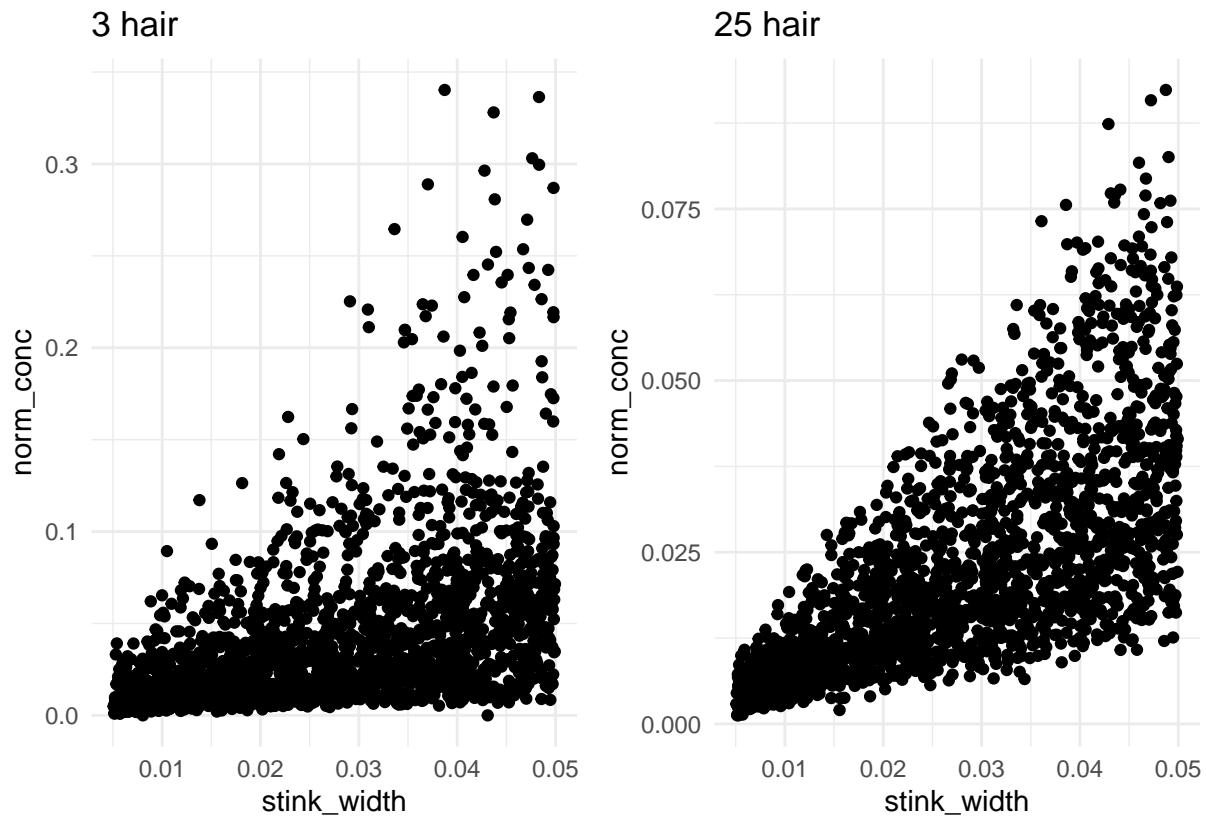
Plotting diffusion coefficient and normalized concentration.

```
plot_parameter(three_dat, twofive_dat, "diff_coef", "norm_conc")
```



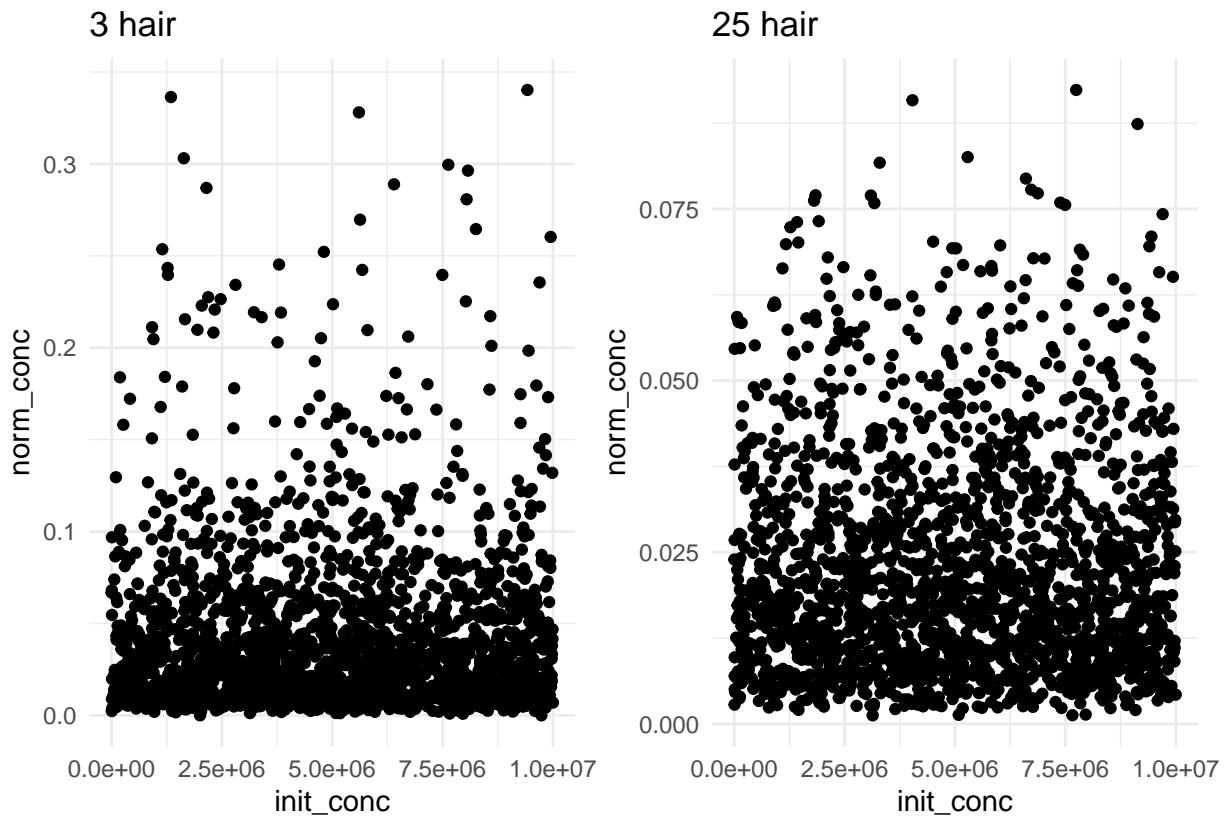
Plotting stripe width and normalized concentration.

```
plot_parameter(three_dat, twofive_dat, "stink_width", "norm_conc")
```



Plotting initial concentration and normalized concentration.

```
plot_parameter(three_dat, twofive_dat, "init_conc", "norm_conc")
```



**Individual Hairs** Plotting angle, gap width and individual hairs.

```

hairs_3dat <- tidyverse::pivot_longer(three_dat, cols = starts_with("hair"), names_to = "hair_number", values_to = "hair_value")
hair_names <- paste0(rep("hair", 3), seq(1,3))
hairs_3dat$hair_number <- factor(hairs_3dat$hair_number, levels = hair_names, ordered = T)
hairs_3dat$row_number <- rep(1, nrow(hairs_3dat))

hairs_25dat <- tidyverse::pivot_longer(twofive_dat, cols = starts_with("hair"), names_to = "hair_number", values_to = "hair_value")
hair_names <- paste0(rep("hair", 25), seq(1,25))
hairs_25dat$hair_number <- factor(hairs_25dat$hair_number, levels = hair_names, ordered = T)
hairs_25dat$row_number <- rep(NA, nrow(hairs_25dat))
for(i in 1:nrow(hairs_25dat)){
  hairs_25dat$row_number[i] <- switch(as.character(hairs_25dat$hair_number[i]),
    "hair1" = 1, "hair2" = 1, "hair3" = 1,
    "hair4" = 2, "hair5" = 2, "hair6" = 2, "hair7" = 2,
    "hair8" = 3, "hair9" = 3, "hair10" = 3, "hair11" = 3, "hair12" = 3,
    "hair13" = 4, "hair14" = 4, "hair15" = 4, "hair16" = 4, "hair17" = 4,
    "hair19" = 5, "hair20" = 5, "hair21" = 5, "hair22" = 5, "hair23" = 5,
    NA)
}

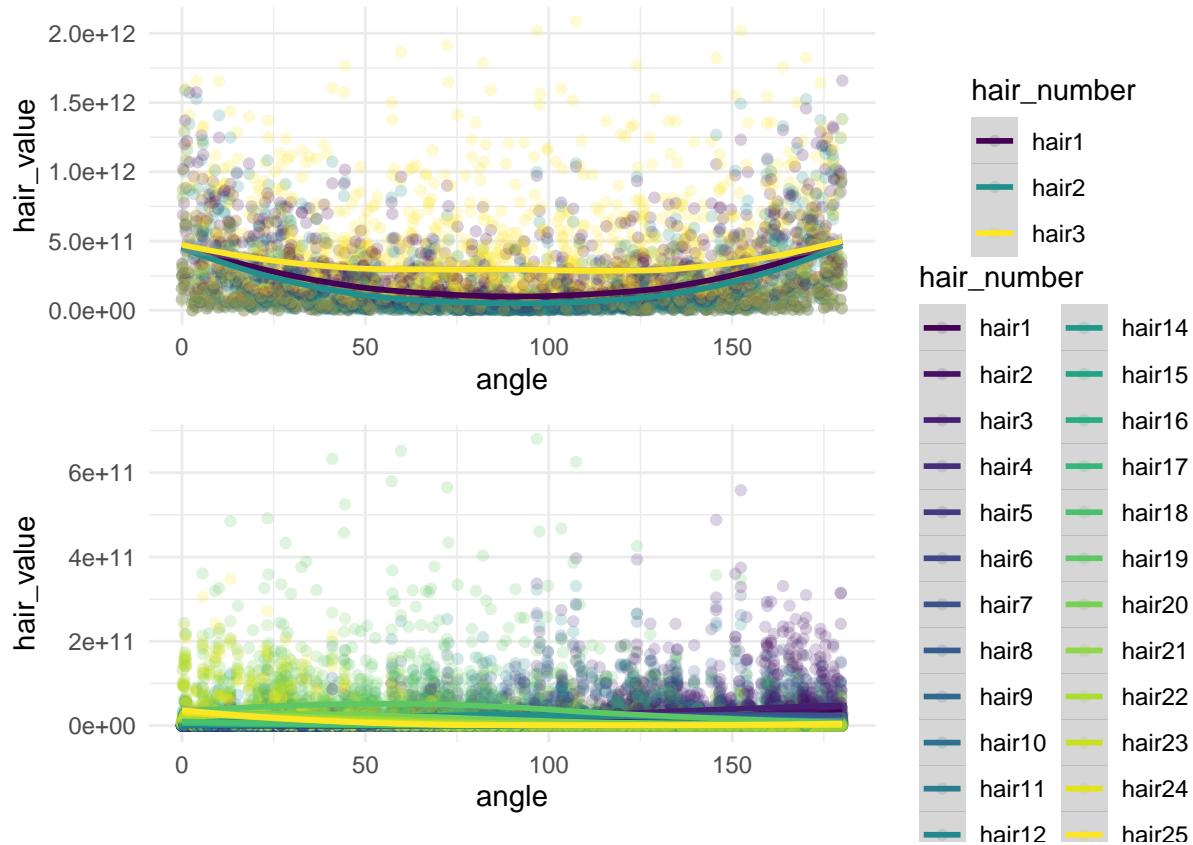
p3_1 <- ggplot(hairs_3dat, aes(angle, hair_value, color = hair_number)) +
  geom_point(alpha = 0.2, pch = 19) + geom_smooth(method = "loess") +
  scale_color_viridis(discrete = T) +
  theme_minimal()

```

```
p25_1 <- ggplot(hairs_25dat, aes(angle, hair_value, color = hair_number))+
  geom_point(alpha=0.2, pch = 19) + geom_smooth(method = "loess") +
  scale_color_viridis(discrete = T) +
  theme_minimal()
```

p3\_1 / p25\_1

```
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
```



## Individual rows

### Comparing to previous paper

Subsetting data to just the “water” and “air” conditions:

```
# Restricting data to less than a specific diffusion coefficient and stripe width
water_twofivesub_dat <- twofive_dat[twofive_dat$diff_coef < 5e-6 &
                                         twofive_dat$stink_width < 0.009,]
water_threesub_dat <- three_dat[three_dat$diff_coef < 5e-6 &
                                         three_dat$stink_width < 0.009,]

# Restricting data to greater than a specific diffusion coefficient and stripe width
air_twofivesub_dat <- twofive_dat[twofive_dat$diff_coef > 5e-5 &
                                         twofive_dat$stink_width > 0.04,]
air_threesub_dat <- three_dat[three_dat$diff_coef > 5e-5 &
                                         three_dat$stink_width > 0.04,]
```

Plotting just water conditions:

Plotting just air conditions: