Time-series analysis - T_c approach

Take-home messages.

1. Compositional change is too fast in the simulations.
2. Some combinations of parameters are too extreme (20 sp. 20000 individuals, or 200 species 2000 individuals).
3. If we do not take into account our previous knowledge of the species pool, colonization-extinction rates become biased at small durations and, so, the T_c approach. However, the estimator is asymptotically unbiased and effective, meaning that converges fast to the unbiased estimates (using our knowledge of the species pool). LONG TIME-SERIES ARE LESS BIASED.

Loading the data

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
metadata <- read.csv("../../data/simulations/mobsim/IFYAH3130E_jitter_metadata.csv")
load("future_good_solutions.RData")

tst_w_metadata <- inner_join(metadata, tst)
```

```
## Joining, by = "parameter_id"
```

```
# tst_w_metadata <- tst_w_metadata %>% mutate(facet = paste0("SP: ", S_POOL, " N: ", N_SIM,
#                                             " SAD: ", SAD_COEF, " SIGMA : ", SIGMA))

tst_w_metadata <- tst_w_metadata %>% mutate(facet = paste0(S_POOL, "-", N_SIM,
                                                           "-", SAD_COEF, "-", SIGMA))
```

T_c plots

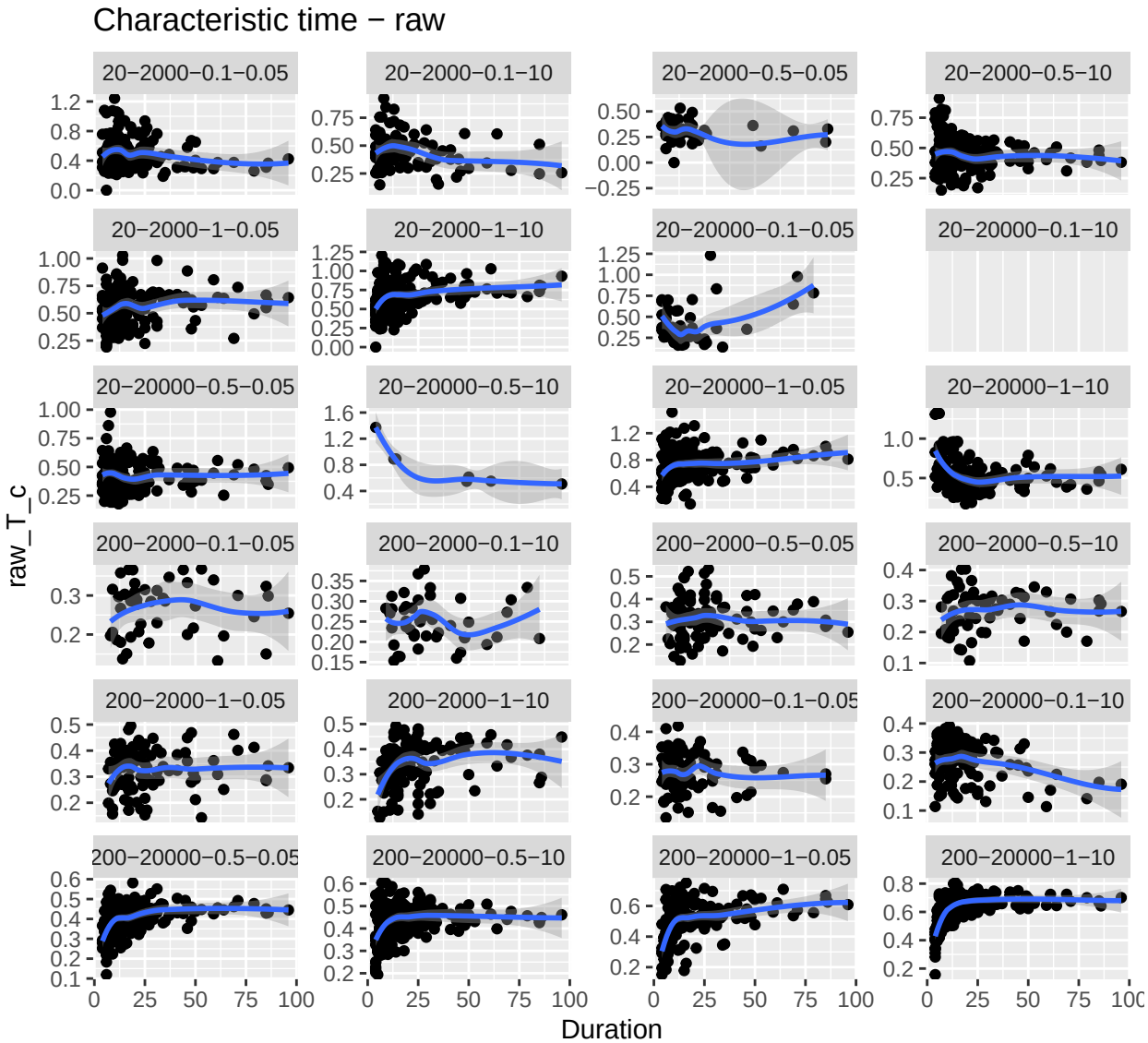
Remember that T_c measures how fast communities change. We can see here that the communities are changing quite fast (most times at a pace faster that T_c).

```
ggplot(tst_w_metadata, aes(x = Duration, y = raw_T_c)) + geom_point() +
   facet_wrap(~facet, scales = "free_y", ncol = 4) +
   ggtitle("Characteristic time - raw") +
   geom_smooth()
```

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

```
## Warning: Removed 2346 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 2346 rows containing missing values (geom_point).
```

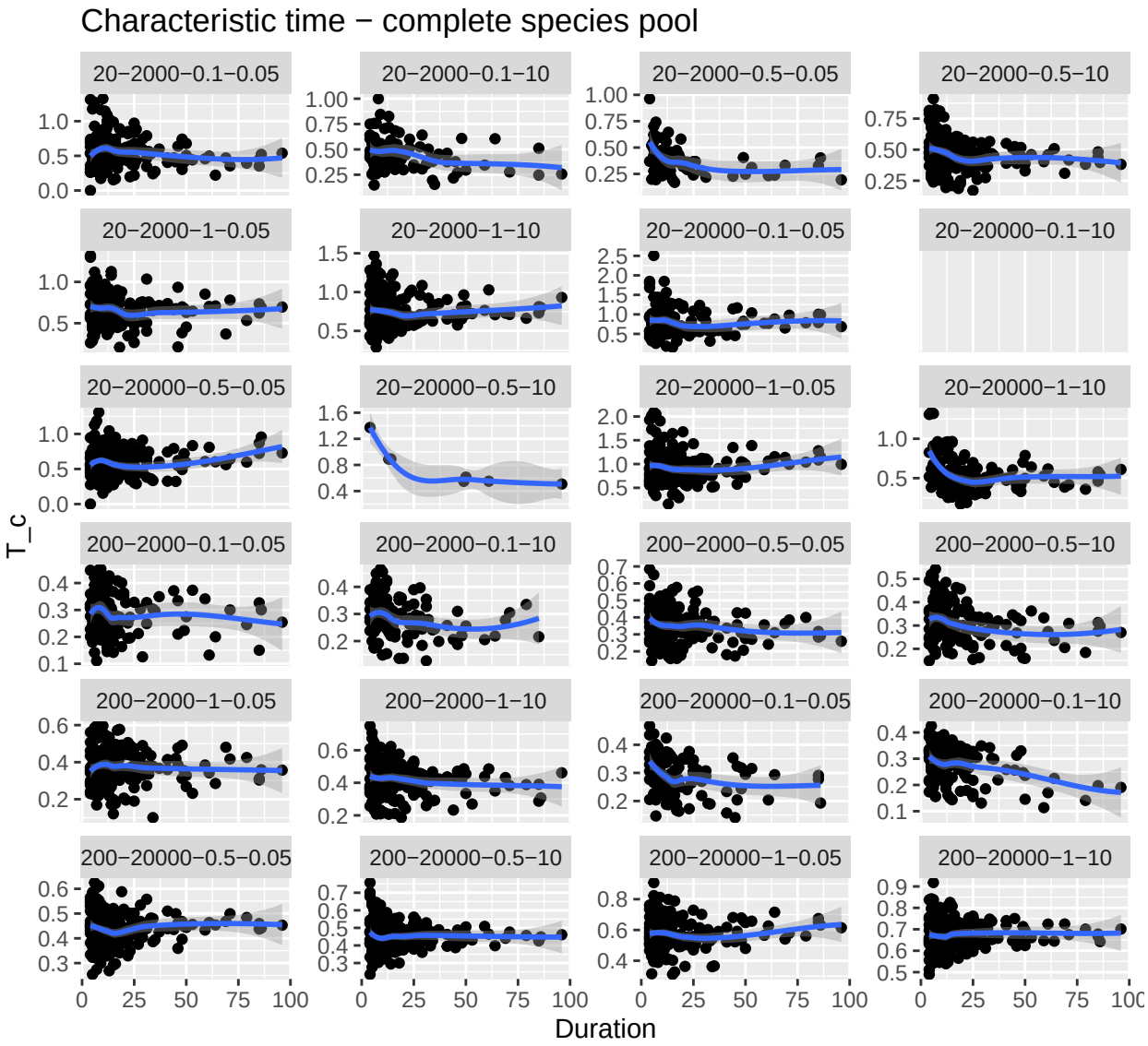


We see above that T_c is a bit biased for small time-series, but asymptotically unbiased. This is so because I used the observed species pool in the timeseries. Once a big proportion of the species in the pool have been recorded, we lost that bias.

```
ggplot(tst_w_metadata, aes(x = Duration, y = T_c)) + geom_point() +
   facet_wrap(~facet, scales = "free_y", ncol = 4) +
```

```
   ggtitle("Characteristic time - complete species pool") +
   geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## Warning: Removed 1482 rows containing non-finite values (stat_smooth).
## Warning: Removed 1482 rows containing missing values (geom_point).
```



However, if we use our knowledge of the species pool, we get unbiased estimates of T_c.

Asymptotic Jaccard

Asymptotic Jaccard corresponds to the expected Jaccard dissimilarity between two samples far in time, that is, effectively independent. Here we see three things:

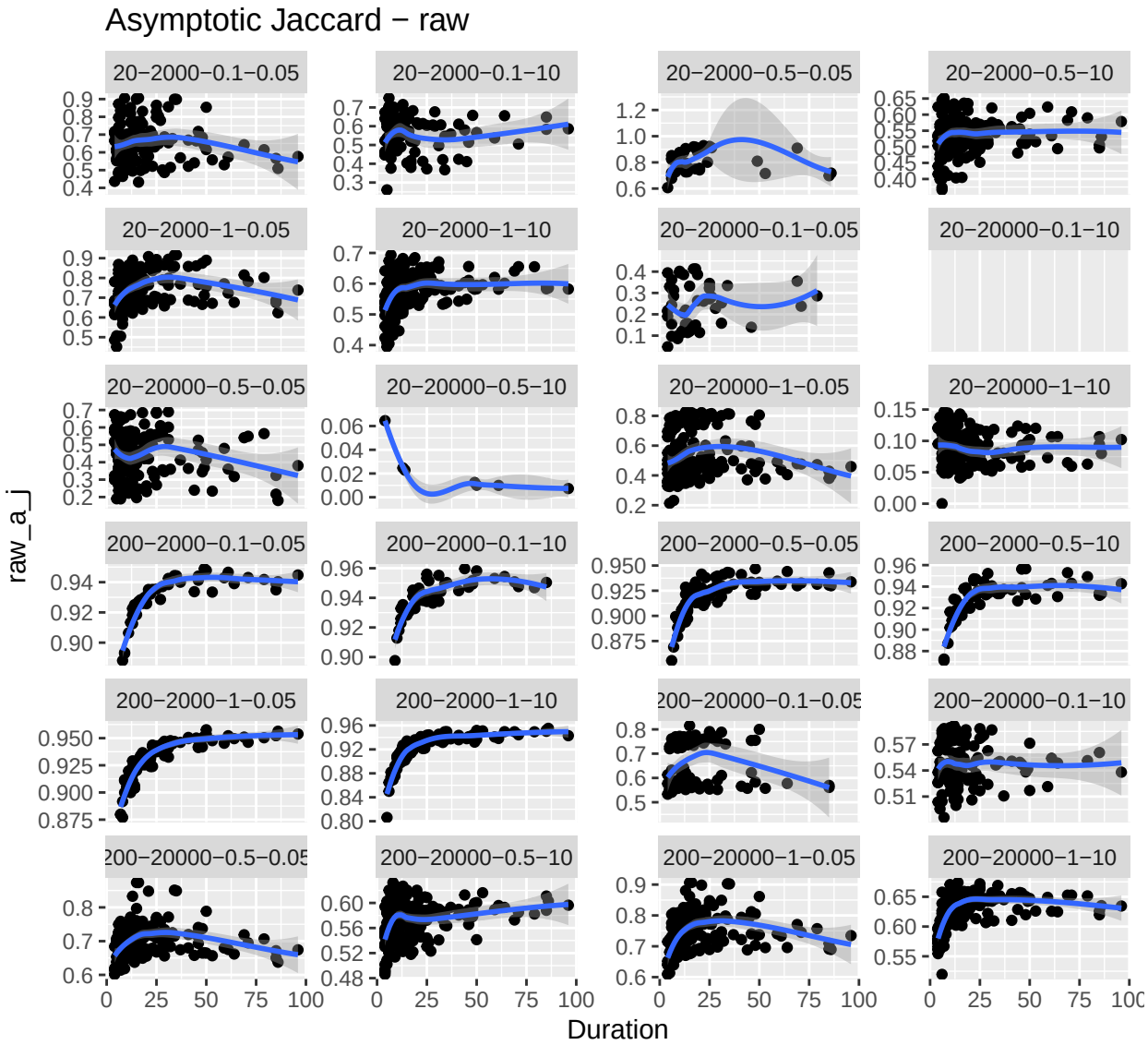
1. A bias in the raw measures.
2. The combination of 200 species in the pool and only 2000 individuals produces very high turnover.
3. The combination of 20 species and 20000 produces small turnover.

```
ggplot(tst_w_metadata, aes(x = Duration, y = raw_a_j)) + geom_point() +
   facet_wrap(~facet, scales = "free_y", ncol = 4) +
   ggtitle("Asymptotic Jaccard - raw") +
   geom_smooth()
```

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

```
## Warning: Removed 2349 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 2349 rows containing missing values (geom_point).
```

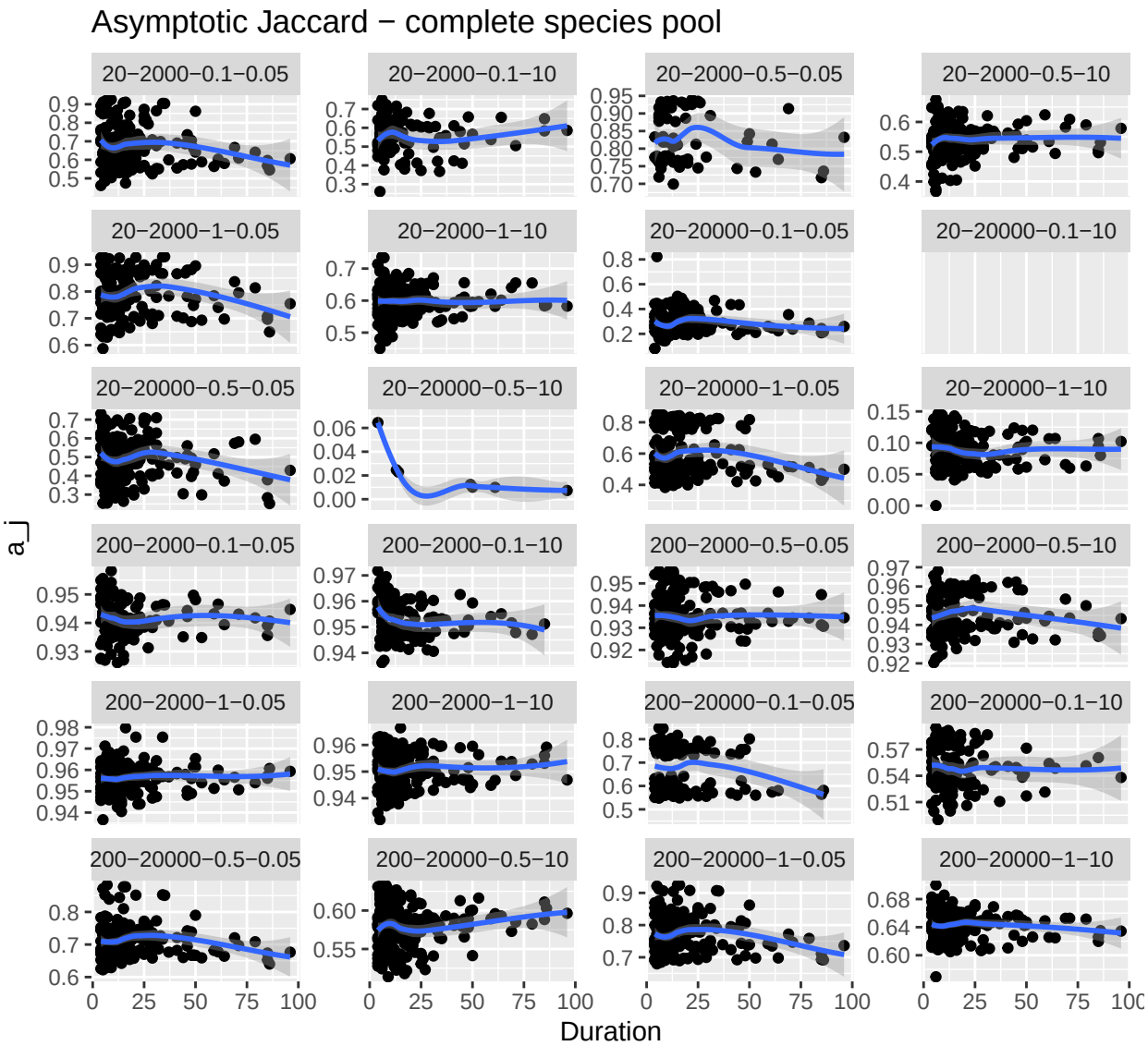


```
ggplot(tst_w_metadata, aes(x = Duration, y = a_j)) + geom_point() +
   facet_wrap(~facet, scales = "free_y", ncol = 4) +
   ggtitle("Asymptotic Jaccard - complete species pool") +
   geom_smooth()
```

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

```
## Warning: Removed 1484 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 1484 rows containing missing values (geom_point).
```



Other measures

Halving time

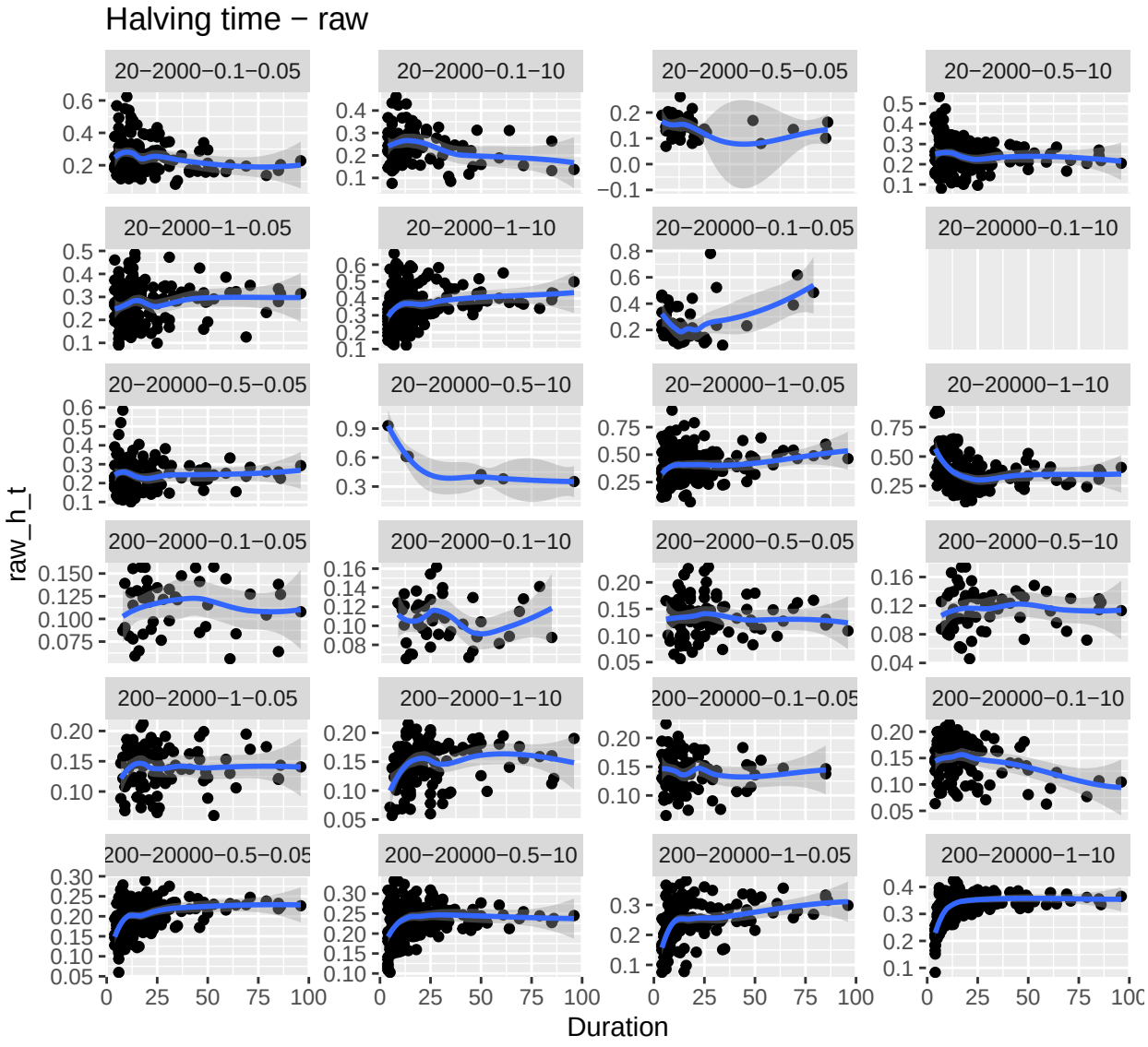
Akin to Helmut's half-saturation. It is too fast.

```
ggplot(tst_w_metadata, aes(x = Duration, y = raw_h_t)) + geom_point() +
   facet_wrap(~facet, scales = "free_y", ncol = 4) +
   ggtitle("Halving time - raw") +
   geom_smooth()
```

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

```
## Warning: Removed 2349 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 2349 rows containing missing values (geom_point).
```



Affinity

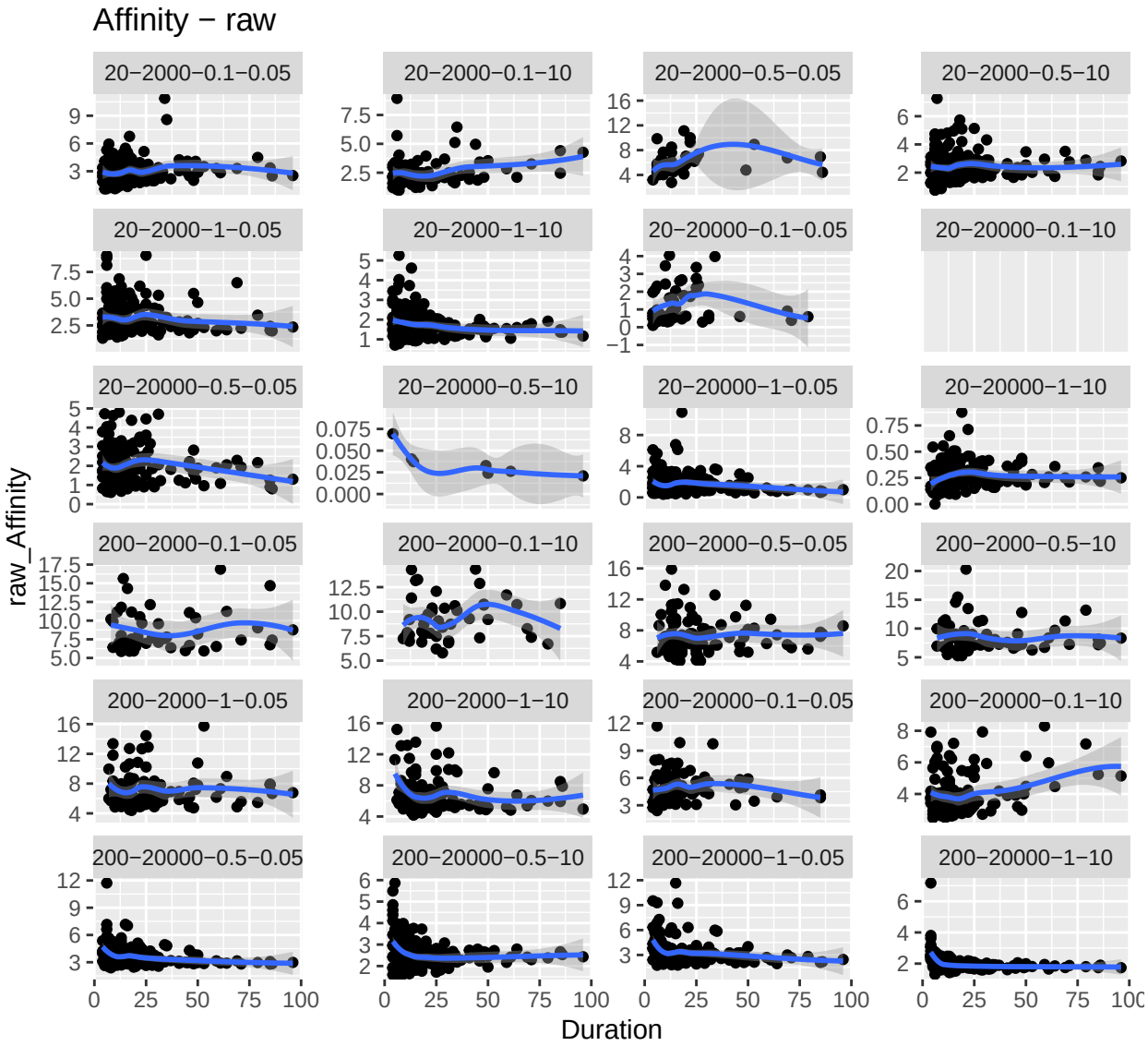
Akin to Helmut's approach. It is less biased than other measures. Low values in the 20 species, 20000 individuals scenario might indicate that there is a fast and small change in these simulations.

```
ggplot(tst_w_metadata, aes(x = Duration, y = raw_Affinity)) + geom_point() +
   facet_wrap(~facet, scales = "free_y", ncol = 4) +
   ggtitle("Affinity - raw") +
   geom_smooth()
```

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

```
## Warning: Removed 2349 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 2349 rows containing missing values (geom_point).
```



Jaccard at the Characteristic time.

Indicates dissimilarity at short time-scales. It is asymptotically unbiased.

```
ggplot(tst_w_metadata, aes(x = Duration, y = raw_J_t_c)) + geom_point() +
   facet_wrap(~facet, scales = "free_y", ncol = 4) +
   ggtitle("Jaccard at the Characteristic Time - raw") +
   geom_smooth()
```

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

```
## Warning: Removed 2349 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 2349 rows containing missing values (geom_point).
```