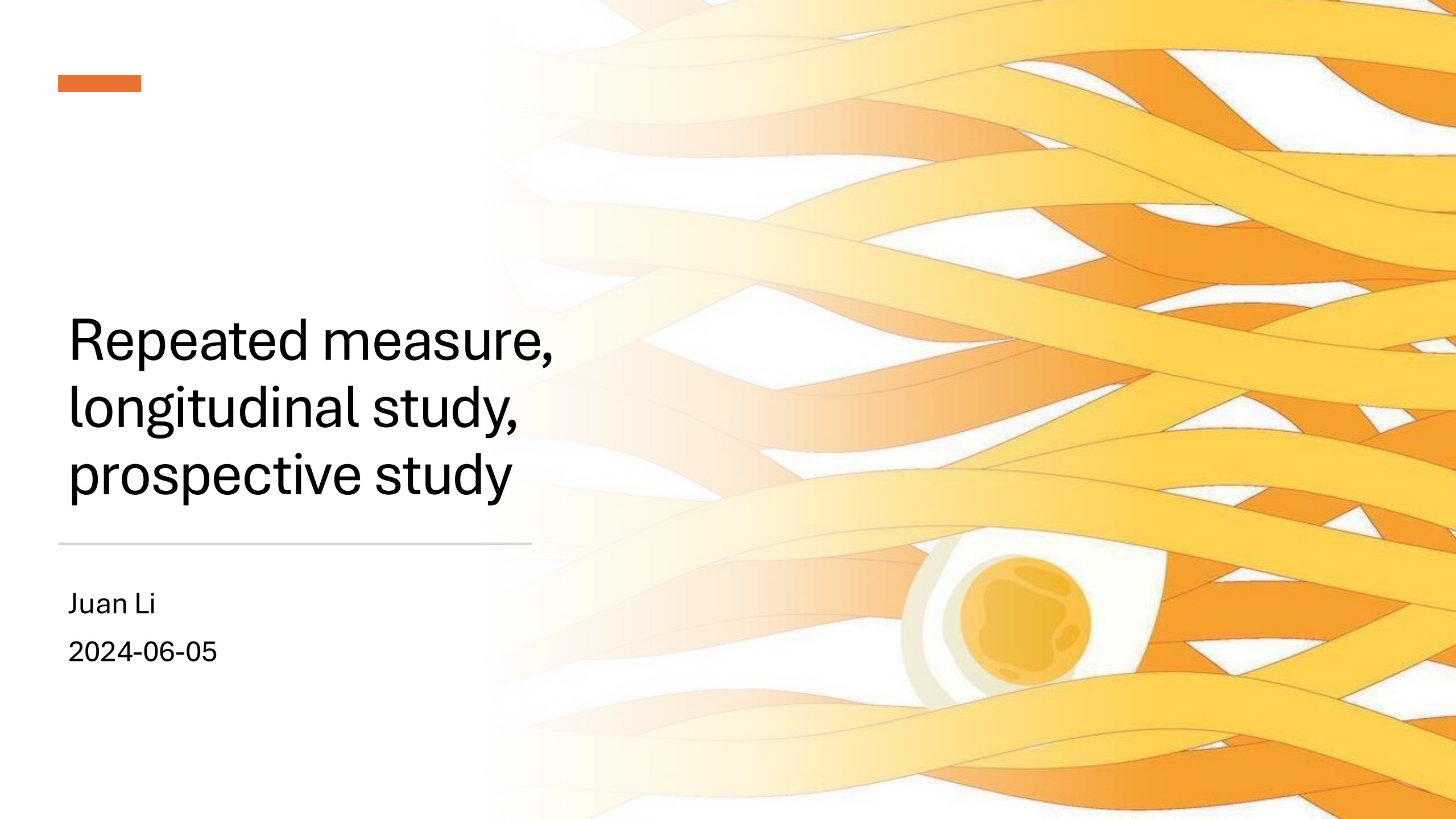
Repeated measure, longitudinal study, prospective study

Juan Li

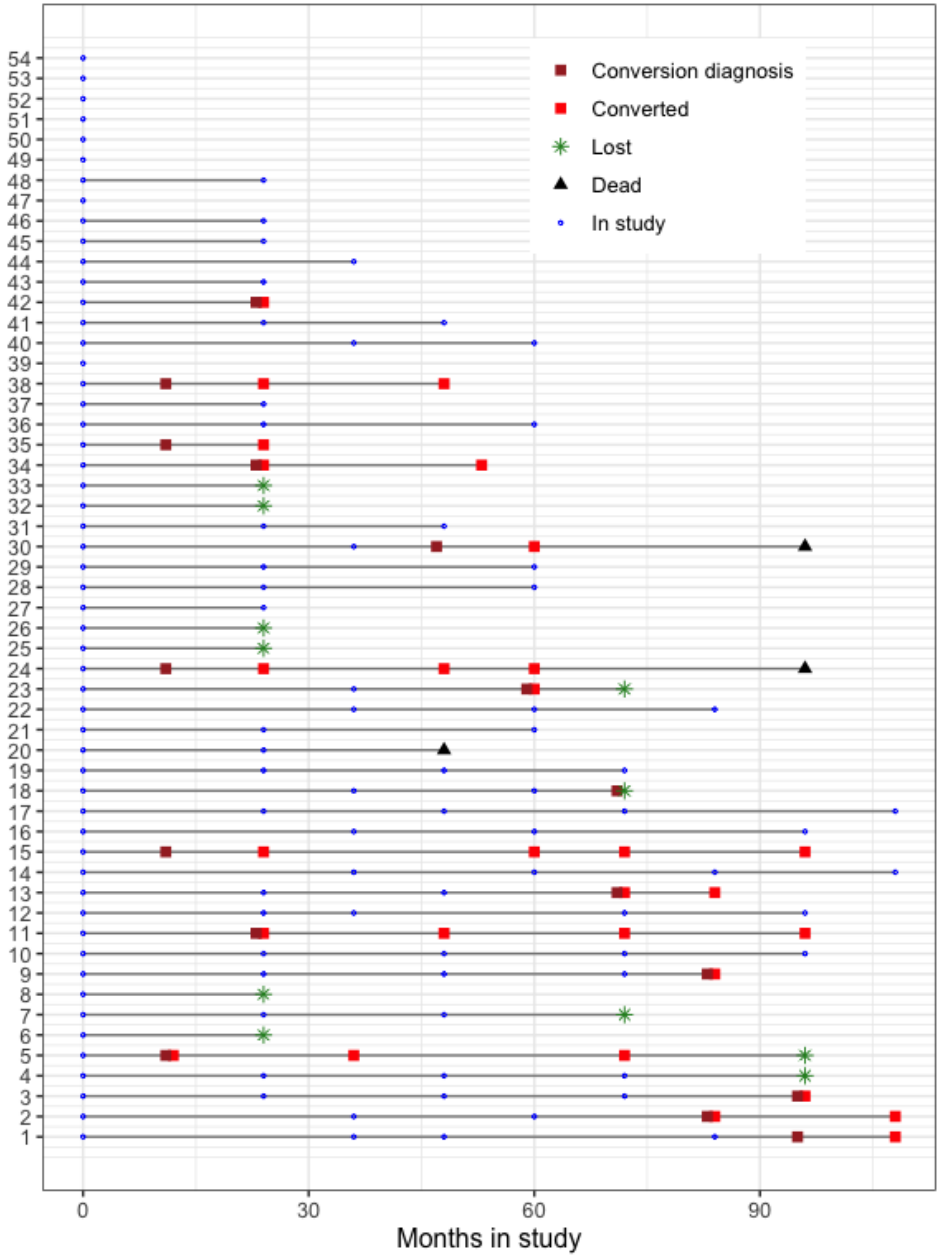
2024-06-05



Resources

1. **Code and data:**
    1. Slides/code/data variable sheet : https://github.com/JuanLiOHRI/longitudinal
    2. Data: can't share the real data, discuss about synthetic data.
2. **Linear Mixed Models:**
    1. https://rpubs.com/alecri/review_longitudinal
    2. https://cran.r-project.org/web/views/MixedModels.html
3. **Survival analysis:**
    1. https://stats.oarc.ucla.edu/wp-content/uploads/2025/02/survival_r_full.html
    2. https://www.sthda.com/english/wiki/survival-analysis-basics
    3. https://www.emilyzabor.com/survival-analysis-in-r.html
    4. https://rpkgs.datanovia.com/survminer/
4. **Cumulative case/dynamic control ROC (!! Inconsistent results, need to discuss)**
    1. `survivalROC`: https://datascienceplus.com/time-dependent-roc-for-survival-prediction-models-in-r/
    2. The tidymodels approach for survival analysis:
        1. https://www.tidyverse.org/blog/2024/04/tidymodels-survival-analysis/
        2. No data splitting: https://www.tidymodels.org/learn/statistics/survival-metrics/
        3. With data splitting and workflow: https://www.tidymodels.org/learn/statistics/survival-case-study/
        4. `roc_auc_survival`: https://yardstick.tidymodels.org/reference/roc_auc_survival.html

Kassel RBD: a longitudinal, prospective study



Variable	Not converted N = 38[1]	Converted N = 16[1]	p-value[2]
sex			0.2
Female	14 (37%)	3 (19%)	
Male	24 (63%)	13 (81%)	
age	69 (62, 74)	68 (65, 72)	>0.9

1. Prospectively following patients with confirmed REM behavior disorder (RBD) for pheno-conversion to Parkinson's disease (PD) or dementia with Lewy bodies (DLB).
2. Recruitment is on-going.
    1. Currently enrolled 54 participants.
    2. Follow-up (FU) every ~24 months.
    3. Longest follow-up: 108 months.
3. Within the 54 participants:
    1. 16 have converted: PD: n = 10; DLB: n = 5; unclear: n = 1. Majority males. And they remained in study after pheno-conversion.
    2. 11 have lost from the study:
        1. 3 lost *after* pheno-conversion
        2. 8 lost *before* pheno-conversion (censored)
    3. 3 have dead:
        1. 2 died *after* pheno-conversion
        2. 1 died *before* pheno-conversion (censored)
    4. 29 remain in the study without pheno-conversion (yet; censored).

Smell test scores in Kassel RBD

Questions:

1. How did the scores change over time?
2. Did the scores change differently for the converted patients from those who haven't?
3. Any sex differences?
4. Can we use smell test scores to predict 2-, 4-, 6- or 8-year conversion within the cohort?

Sniffin' Sticks test has 3 subtests: identification (ID), threshold (TH), and discrimination (DS).

SST-ID

- It consists of 16 blue pens with black numbers.
- Each pen is presented only once and an interval of at least 30 seconds is observed between each presentation to avoid olfactory desensitisation.
- For each odorant pen, the subject must make a forced choice from a list of 4 written proposals.
- This test does not require blinded condition.
- The identification score corresponds to the number of correct responses.
- **Integer score, score-range: 0-16. Higher score means better olfaction (sense of smell).**

Smell test scores in Kassel RBD: summary table & score distribution

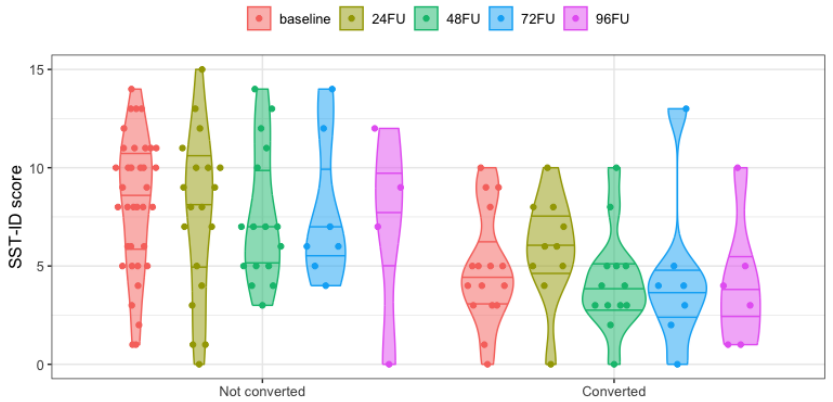
```r
df_tbl <- data %>% select(ID, converted, EVENT_ID, Sniffin.Sticks.ID) %>%
  # pivot the table to make scores in each visit a separate variable
  pivot_wider(names_from = EVENT_ID, values_from = Sniffin.Sticks.ID)
# update variable name
names(df_tbl)[c(3:ncol(df_tbl))] <- paste0("SST_ID_", names(df_tbl)[c(3:ncol(df_tbl))])

# gtsummary
df_tbl %>%
  select(-ID) %>%
  tbl_summary(by = converted,
              missing = "ifany",
              type = list("SST_ID_96FU" ~ "continuous")) %>%
  add_p() %>%
  modify_header(label = "**Variable**") %>% # update the column header
  bold_labels() %>%
  as_flex_table()
```

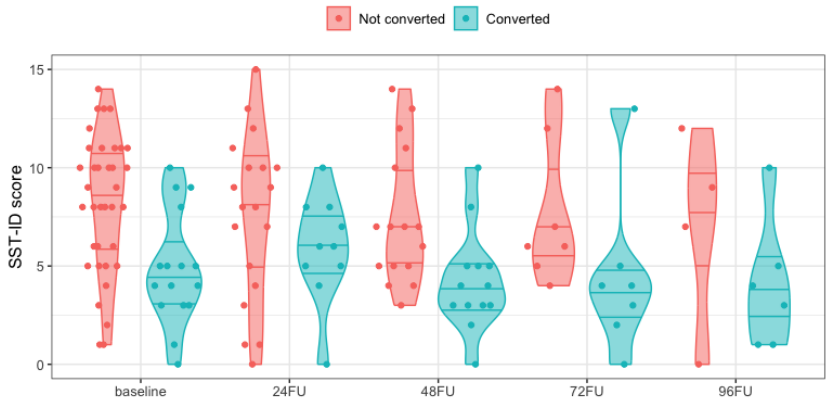
Variable	Not converted, N = 38[1]	Converted, N = 16[1]	p-value[2]
SST_ID_baseline	9 (6, 11)	5 (3, 6)	0.002
Unknown	2	0	
SST_ID_24FU	8 (5, 10)	6 (5, 8)	0.2
Unknown	19	6	
SST_ID_48FU	7 (5, 10)	4 (3, 5)	0.006
Unknown	22	3	
SST_ID_72FU	6 (6, 10)	4 (3, 5)	0.046
Unknown	31	9	
SST_ID_96FU	8 (5, 10)	4 (2, 5)	0.5
Unknown	34	10	

[1]Median (IQR)

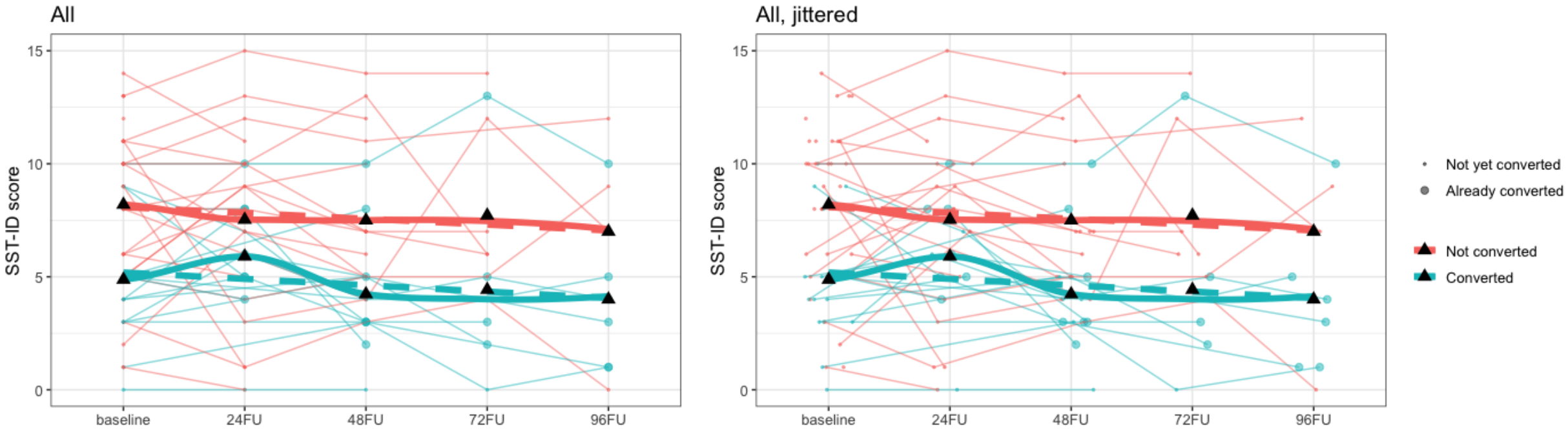
[2]Wilcoxon rank sum test

1. Over half (30/52) RBD patients had reduced sense of smell at baseline.
2. The converted group had even lower score (worse olfaction) than the not-yet-converted group.
3. Within each group, the smell test scores were somewhat stable across study visits, with slight decrease.





Spaghetti plot



1. For jittering:
    1. Consider jittering because the integer scores.
    2. Technical details for making the data points and lines aligned after jittering.
    3. Set random seed for reproducibility
2. Jittering doesn't affect `geom_smooth()`.
3. Mean values and fitted lines (both `lm` and `loess`) don't consider the linked data.

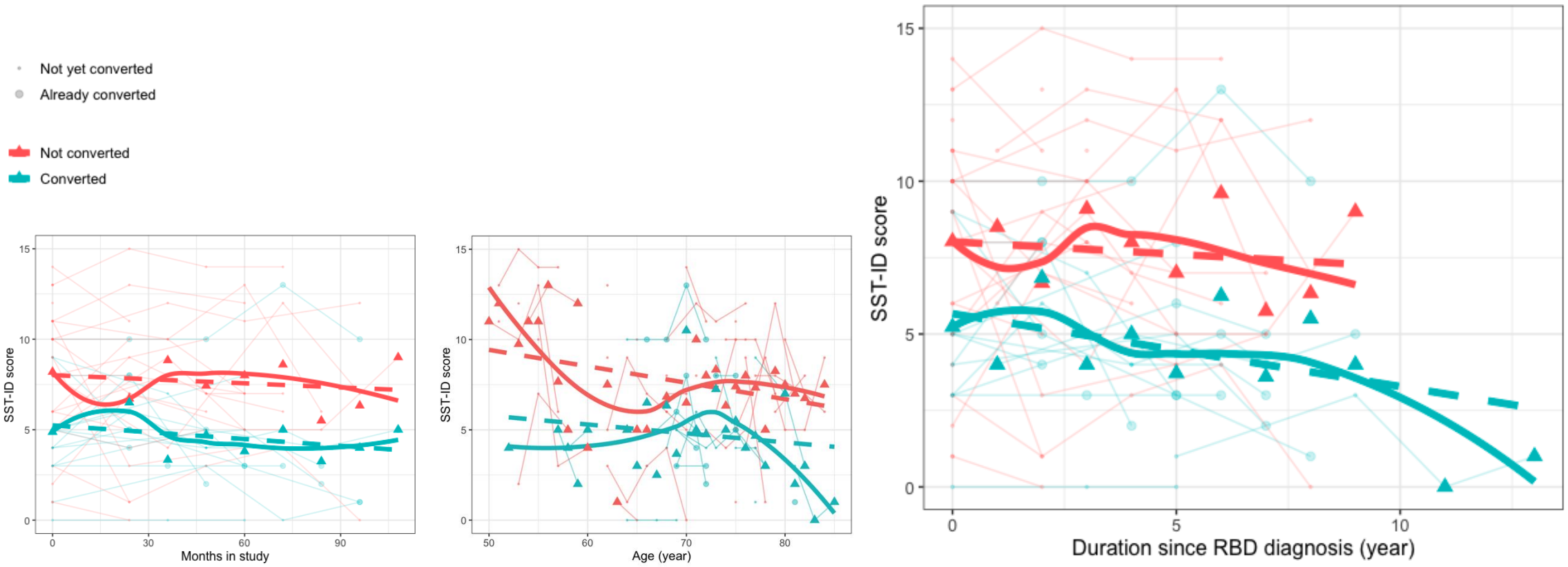
```
geom_smooth(aes(group = converted), se = FALSE, size=2)+
geom_smooth(aes(group = converted), se = FALSE, size=2, method = "lm", linetype = "dashed")+
stat_summary(aes(group = converted), geom = "point", fun.y = mean,
             shape = 17, size = 3, color = "black") +
```

Same conclusions as earlier:

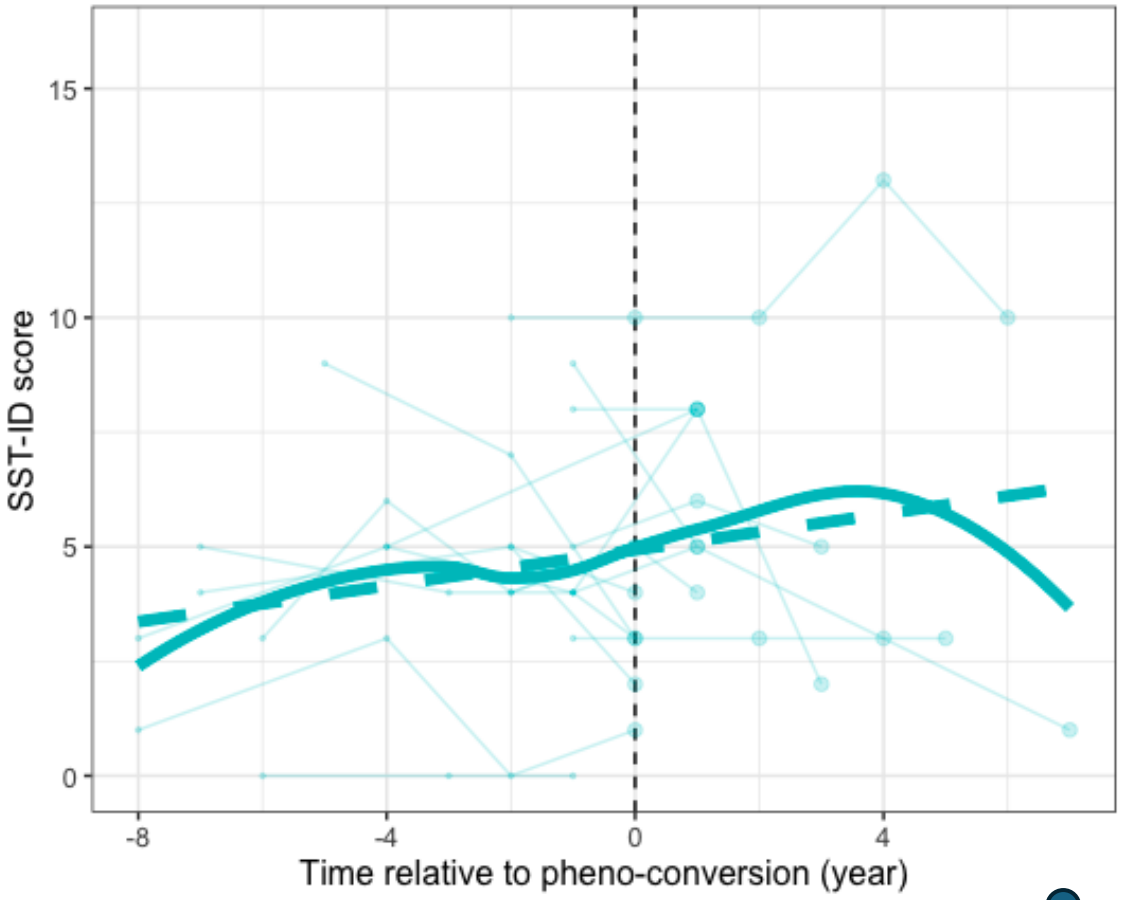
1. The converted group had lower score (worse olfaction) than the not-converted group.
2. Within each group, the smell test scores were somewhat stable across study visits, with slight decrease.

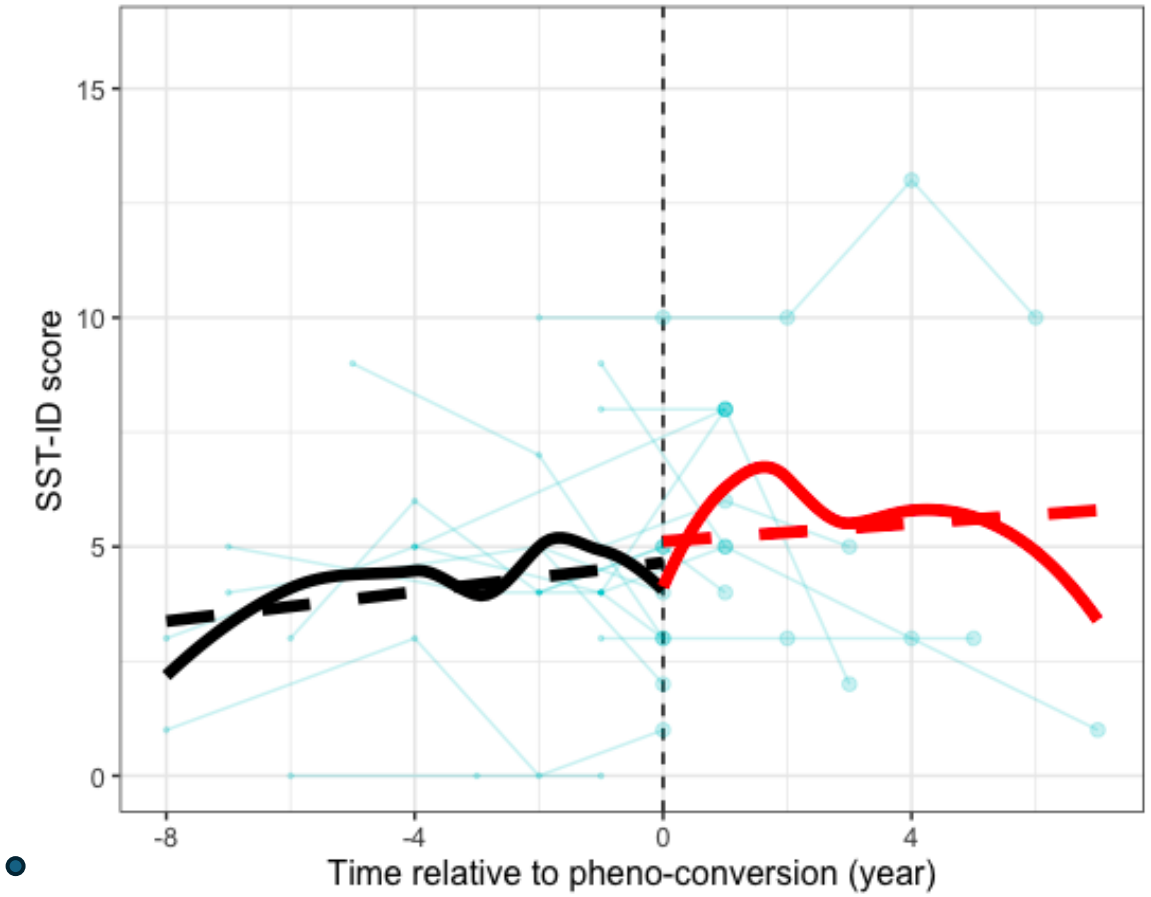
Li, *et al*. unpublished data

Spaghetti plot, better time scale



Spaghetti plot, better time scale – the converted group





Why increased??

1. Change of smell test score *before* PD/DLB conversion – marker for PD/DLB onset; prodromal symptom/risk factor
2. Change of smell test score *after* PD/DLB conversion – marker for PD/DLB progression

Linear Mixed Models: time effect (Is the mean SST-ID score varying with RBD duration?)

The mean response is modeled as a combination of **population characteristics (*fixed effects*) assumed to be shared by all individuals**, while **subject-specific effects (*random effects*) are unique to a particular individual**. Linear Mixed Models are a particular type of hierarchical models which contains both fixed and random effects.

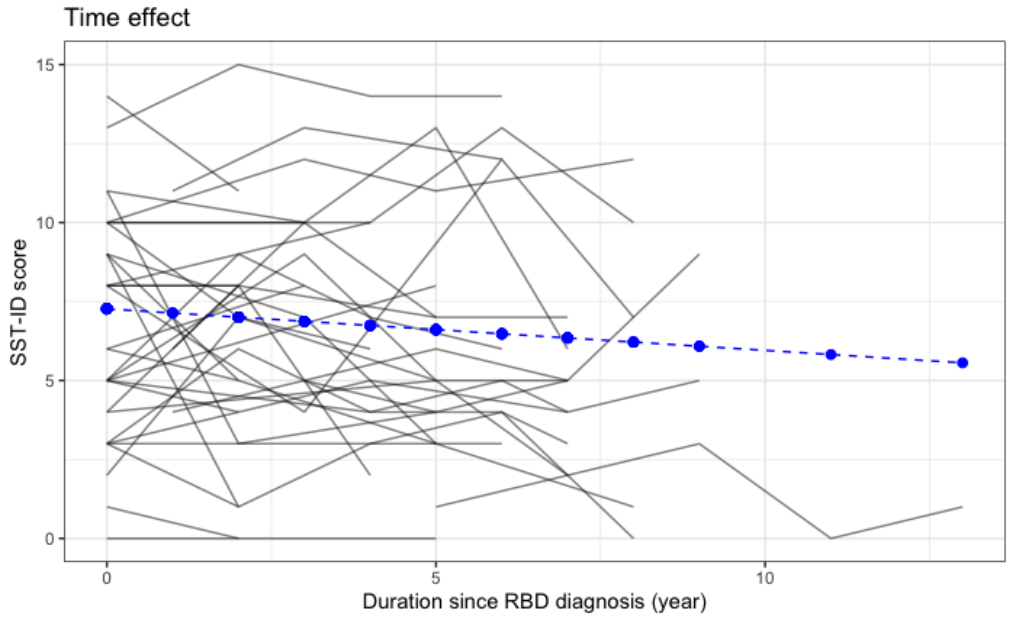
```
library(lme4)
library(Epi)
library(lmerTest)
library(emmeans)
library(rstatix)
```

```
lin_time <- lmer(Sniffin.Sticks.ID ~ RBD_duration + (1 | ID), data = data)
```

Fixed time effect Random effect

```
summary(lin_time)
```

```
Fixed effects:
             Estimate Std. Error       df t value Pr(>|t|)
(Intercept)   7.26679    0.49865 68.75384  14.573   <2e-16 ***
RBD_duration -0.13140    0.07434 96.76940  -1.768   0.0803 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
            (Intr)
RBD_duratin -0.376
```



- The mean SST-ID score at time of RBD diagnosis is 7.27.
- For the increasing of 1 year with RBD, the mean SST-ID score decreases 0.13 (p-value: 0.08).
- The negative trajectory of the mean score over RBD duration is not significant (p-value: 0.08).

Li, *et al.* unpublished data
https://rpubs.com/alecri/review_longitudinal

Linear Mixed Models: group effect (Is the mean SST-ID score varying in the two groups?)

```
lin_group <- lmer(Sniffin.Sticks.ID ~ RBD_duration + converted + (1 | ID), data = data)
```

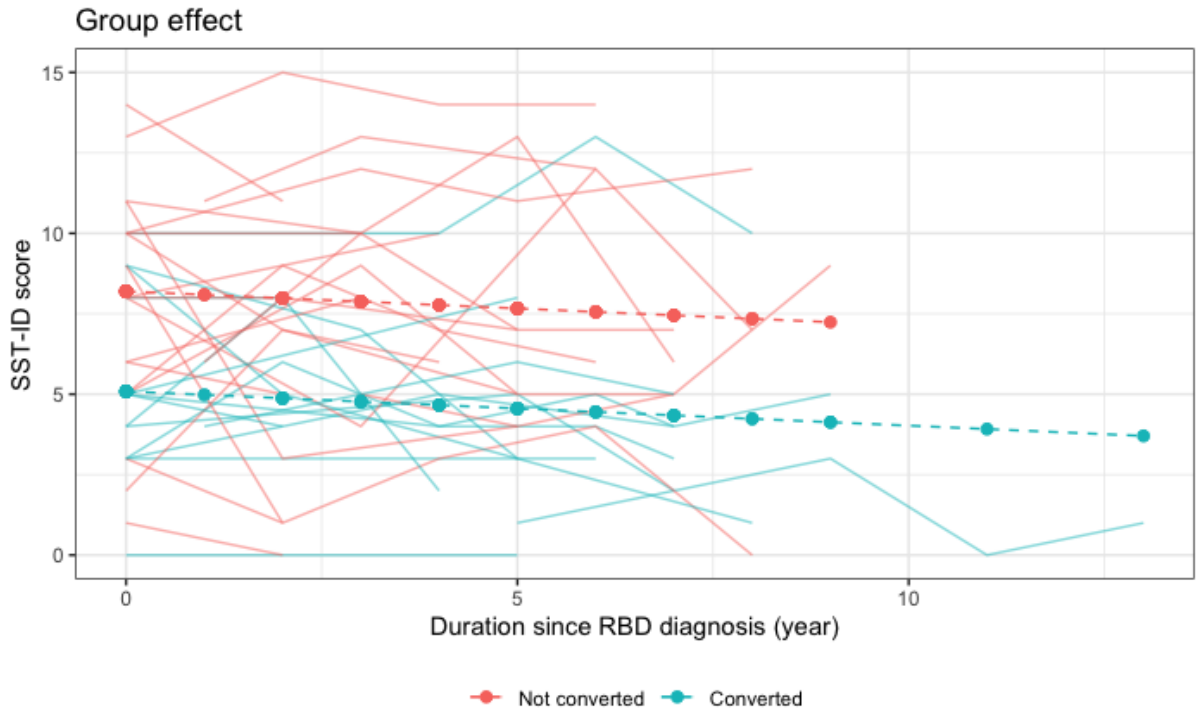
Fixed time and group effect
(different intercepts, same slope)

Random effect

```
summary(lin_group)
```

```
Fixed effects:
                   Estimate Std. Error      df t value Pr(>|t|)
(Intercept)          8.1979     0.5342 63.6940  15.345  < 2e-16 ***
RBD_duration        -0.1065     0.0741 98.1302  -1.437  0.15377
convertedConverted  -3.1093     0.9050 48.5542  -3.436  0.00122 **
```

The mean score of the converted group is significantly lower than the not-yet-converted group.



Li, *et al.* unpublished data
https://rpubs.com/alecri/review_longitudinal

Linear Mixed Models: interaction between time and group

(Is the change of the mean SST-ID score over time varying according to group?)

```
lin_inter <- lmer(Sniffin.Sticks.ID ~ RBD_duration*converted + (1 | ID), data = data)
```

Fixed time and group effect
(different intercepts, **different slopes**)

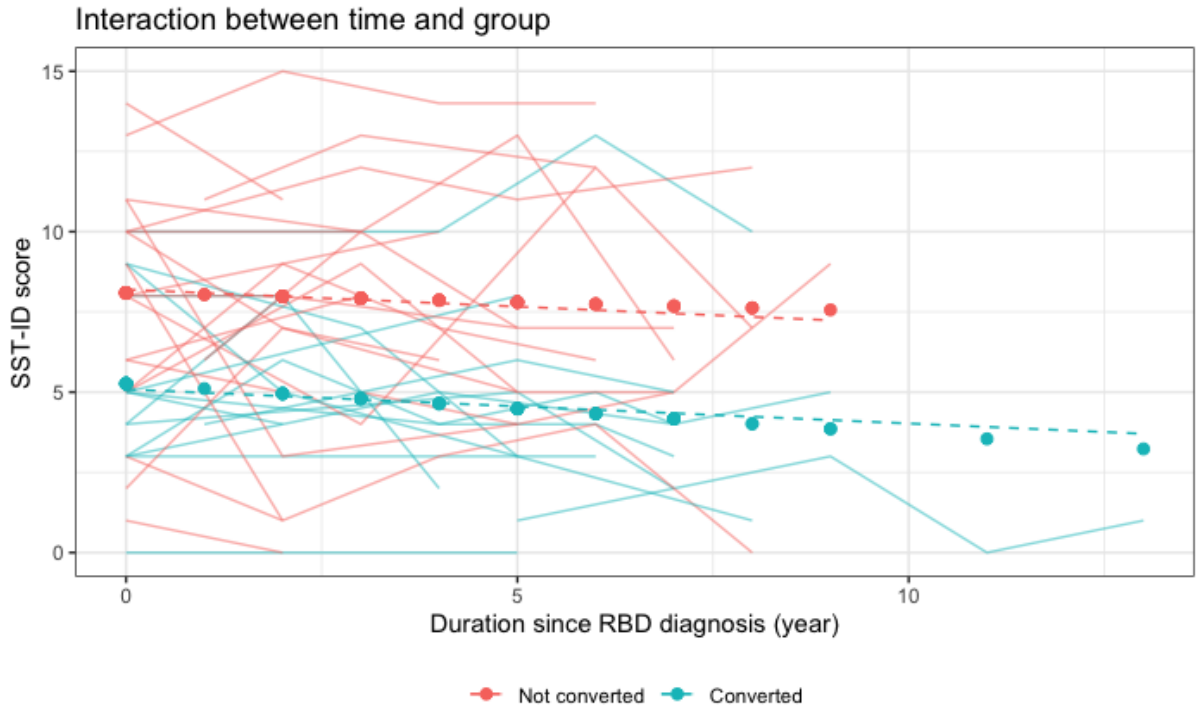
Random effect

```
summary(lin_inter)
```

```
Fixed effects:
                                Estimate Std. Error       df t value Pr(>|t|)
(Intercept)                      8.10069    0.55441 72.81761  14.611  < 2e-16 ***
RBD_duration                    -0.05965    0.10356 99.45617  -0.576  0.56591
convertedConverted              -2.83340    0.99790 69.69825  -2.839  0.00592 **
RBD_duration:convertedConverted -0.09697    0.14881 97.13519  -0.652  0.51615
```

```
Anova(lin_inter, type = 3)
```

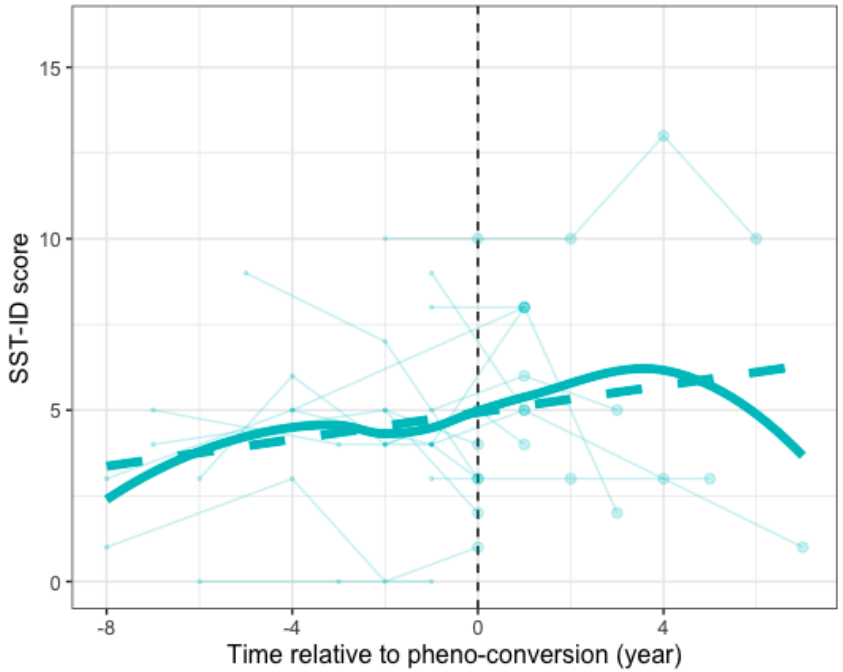
```
Analysis of Deviance Table (Type III Wald chisquare tests)

Response: Sniffin.Sticks.ID
                         Chisq Df Pr(>Chisq)
(Intercept)           213.4930  1    < 2e-16 ***
RBD_duration            0.3318  1    0.56461
converted               8.0620  1    0.00452 **
RBD_duration:converted  0.4247  1    0.51462
```



The Wald test detects no evidence of interaction between `RBD_duration` and `converted` (p = 0.51), which suggests that the mean response profiles is more likely to be parallel.

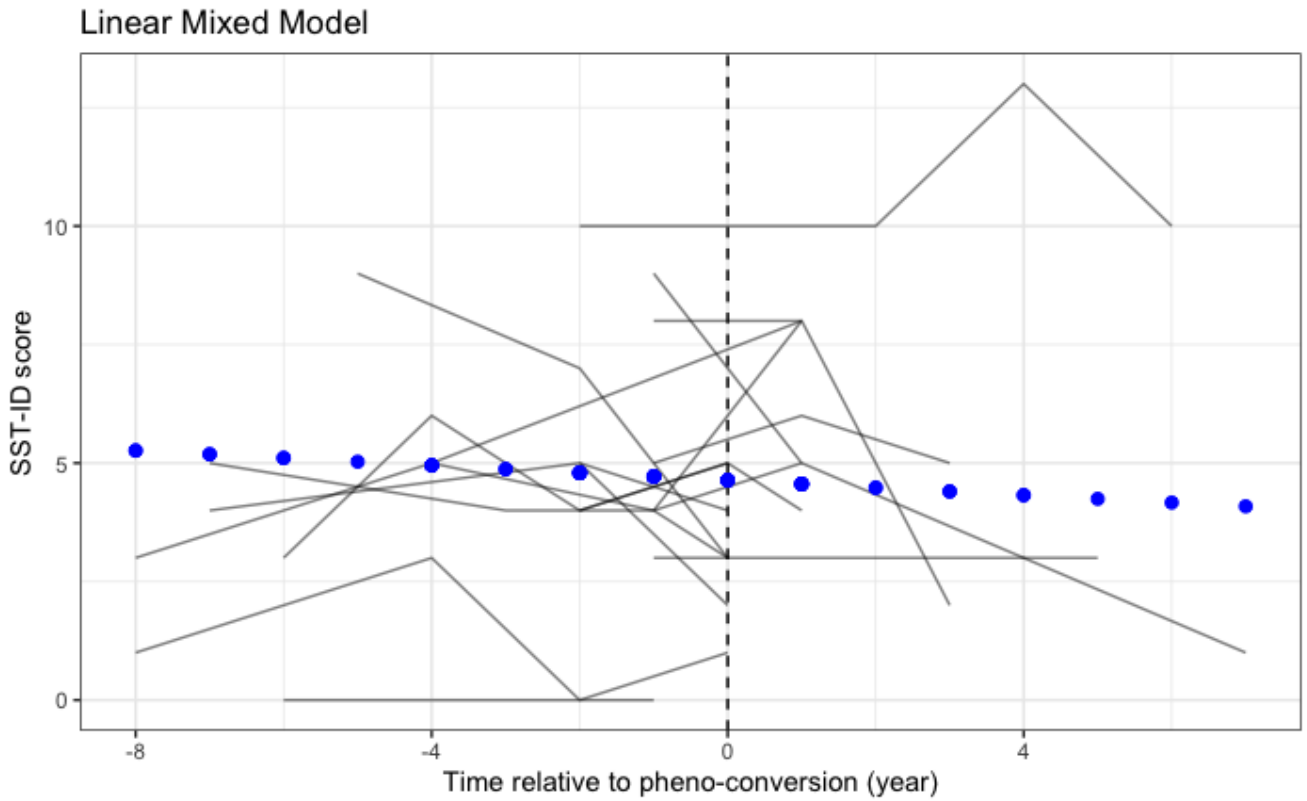
Li, *et al.* unpublished data
https://rpubs.com/alecri/review_longitudinal

Linear Mixed Models: time relative to pheno-conversion



Why increased??

```
df_conv <- data %>% filter(converted == "Converted") %>% droplevels()
lin_visit_conv <- lmer(Sniffin.Sticks.ID ~ time_to_conversion + (1 | ID), data = df_conv)
```

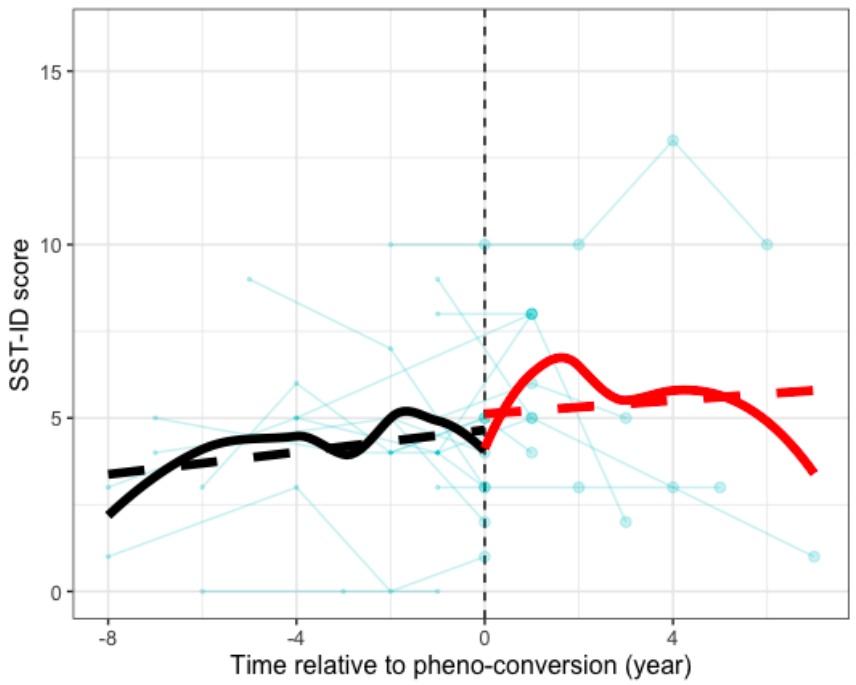



```
summary(lin_visit_conv)
```

```
Fixed effects:
                    Estimate Std. Error       df t value Pr(>|t|)
(Intercept)          4.63652    0.65569 15.29473   7.071 3.41e-06 ***
time_to_conversion -0.07839    0.08971 40.96639  -0.874    0.387
```

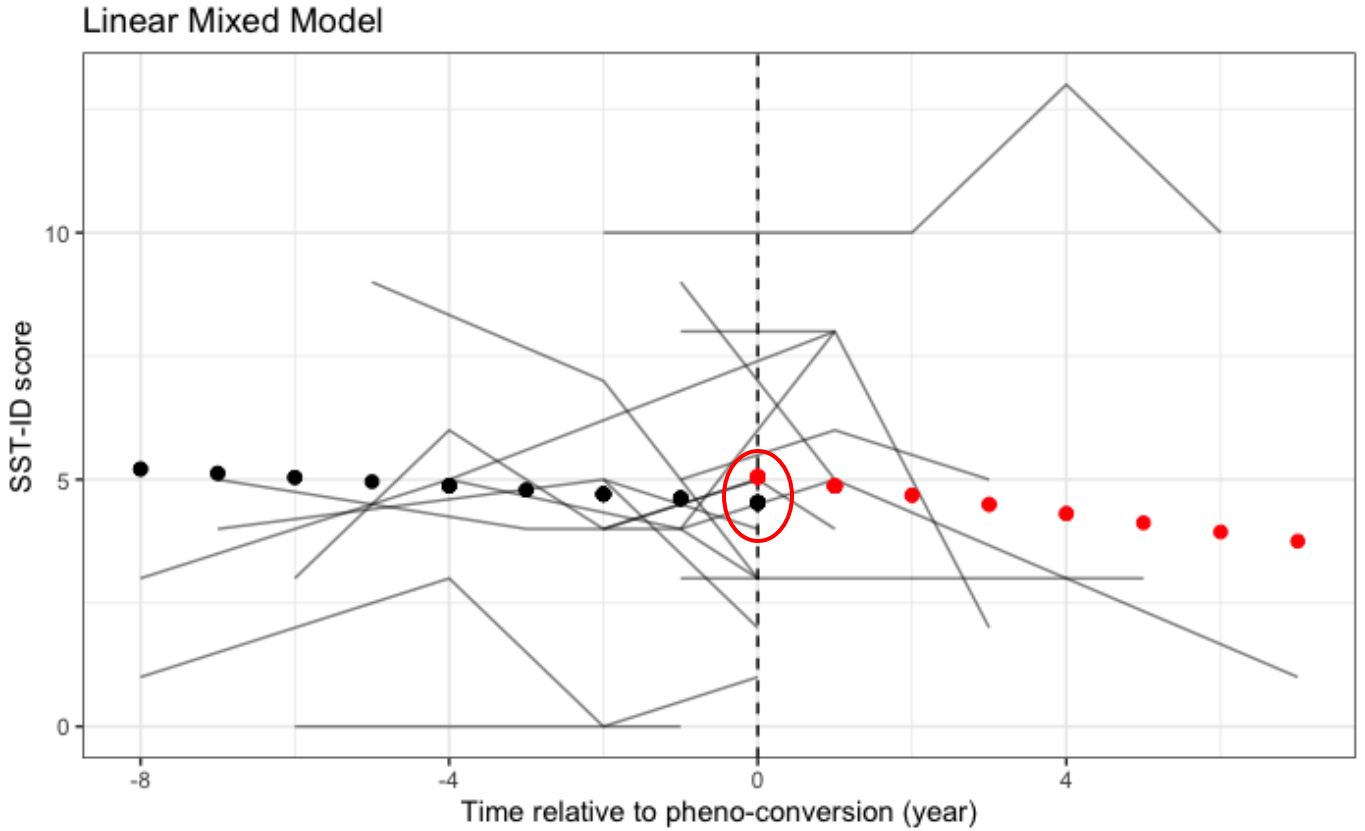
Li, *et al.* unpublished data
https://rpubs.com/alecri/review_longitudinal

Linear Mixed Models: time relative to pheno-conversion




```
df_conv_1 <- df_conv %>% filter(time_to_conversion <= 0)
lin_visit_conv_1 <- lmer(Sniffin.Sticks.ID ~ time_to_conversion + (1 | ID), data = df_conv_1)
```

```
df_conv_2 <- df_conv %>% filter(time_to_conversion >= 0)
lin_visit_conv_2 <- lmer(Sniffin.Sticks.ID ~ time_to_conversion + (1 | ID), data = df_conv_2)
```




```
summary(lin_visit_conv_1)
```

```
Fixed effects:
                  Estimate Std. Error       df t value Pr(>|t|)
(Intercept)        4.53414    0.70149 18.35210   6.464 4.03e-06 ***
time_to_conversion -0.08482    0.10404 20.46424  -0.815    0.424
```

```
summary(lin_visit_conv_2)
```

```
Fixed effects:
                  Estimate Std. Error      df t value Pr(>|t|)
(Intercept)         5.0550     0.7971 18.8692   6.342 4.52e-06 ***
time_to_conversion  -0.1861     0.2337 14.3294  -0.796    0.439
```

Li, *et al.* unpublished data
https://rpubs.com/alecri/review_longitudinal

Linear Mixed Models: linear splines

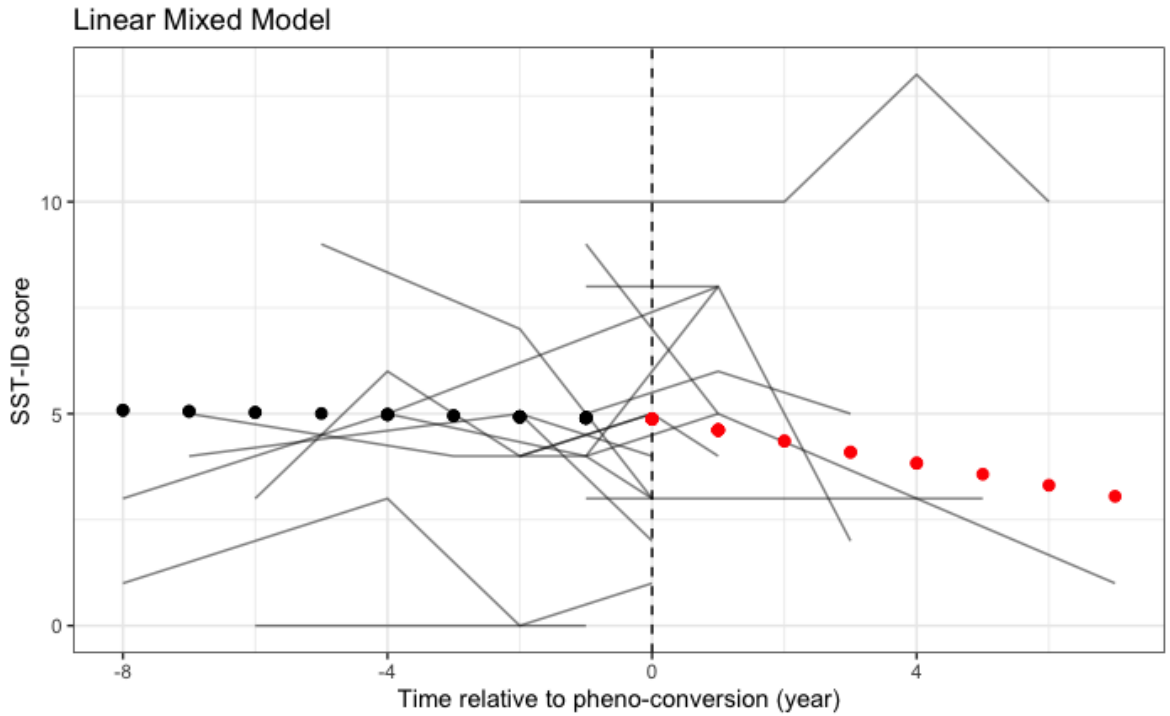
```
df_conv <- df_conv %>%
  mutate(timepost = pmax(time_to_conversion, 0))
lin_lspl0 <- lmer(Sniffin.Sticks.ID ~ time_to_conversion + timepost + (time_to_conversion + timepost | ID), data = df_conv)
```

```
summary(lin_lspl0)
```

```
Fixed effects:
                  Estimate Std. Error       df t value Pr(>|t|)
(Intercept)        4.87580    0.66404 14.88304   7.343 2.55e-06 ***
time_to_conversion -0.02579    0.12241  3.11992  -0.211    0.846
timepost           -0.23481    0.31355  8.37858  -0.749    0.474
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
K <- rbind(
  "population mean pre-conversion slope" = c(0, 1, 0),
  "population mean post-conversion slope" = c(0, 1, 1)
)
rstatix::tidy(multcomp::glht(lin_lspl0, linfct = K), conf.int = TRUE)
```

contrast <chr>	estimate <dbl>	conf.low <dbl>	conf.high <dbl>
population mean pre-conversion slope	-0.02579082	-0.2993519	0.2477703
population mean post-conversion slope	-0.26059769	-0.8681024	0.3469071



Estimate of the population mean pre-conversion slope is -0.026.

Estimate of population mean post-conversion slope is ((-0.026)+(-0.23)) = -0.26, which indicates that the decreasing rate after pheno-conversion is higher than that before conversion (not significant).

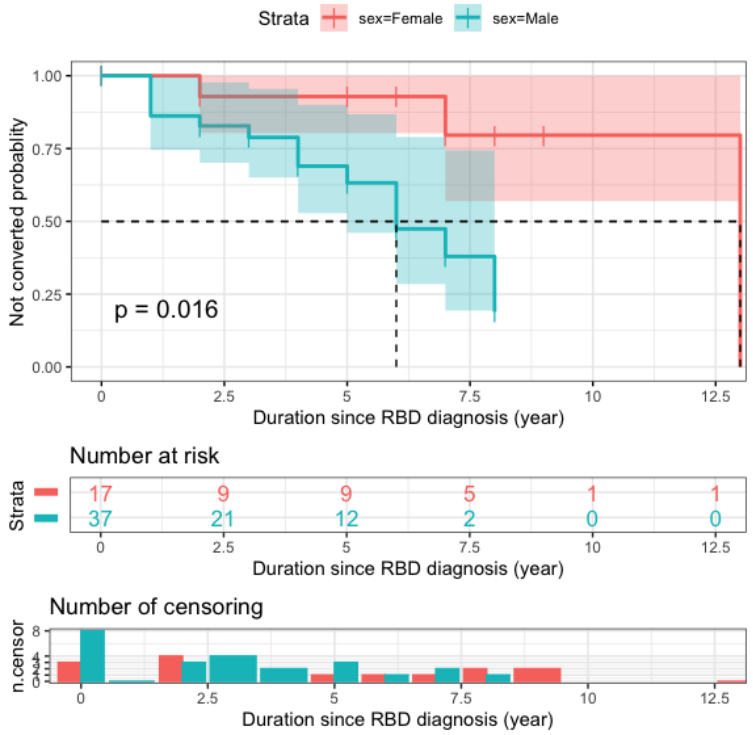
Li, *et al.* unpublished data
https://rpubs.com/alecri/review_longitudinal

Survival analysis

Li, *et al*. unpublished data

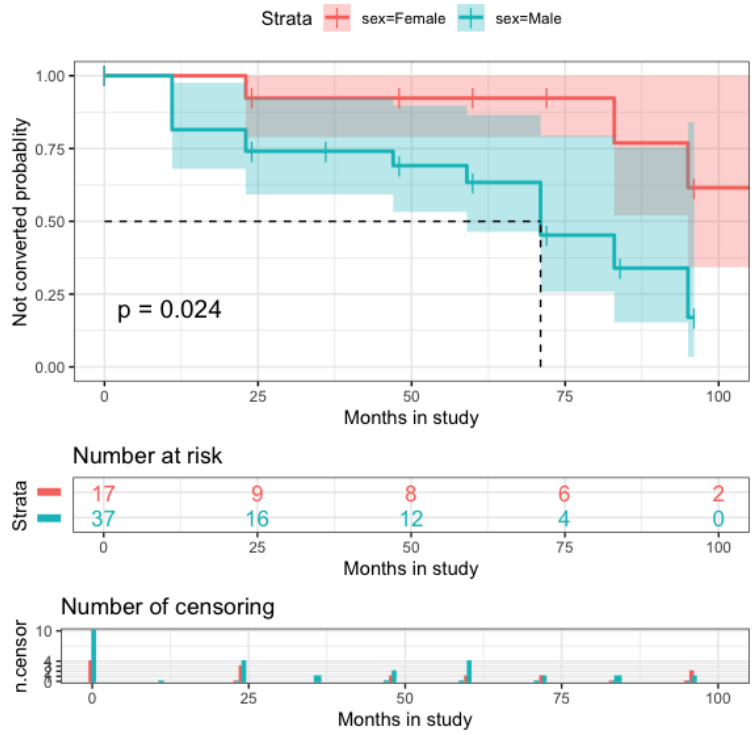
```
library(survival)
library(survminer)
library(broom)
```

Variable	Not converted N = 38[1]	Converted N = 16[1]	p-value[2]
sex			0.2
Female	14 (37%)	3 (19%)	
Male	24 (63%)	13 (81%)	
age	69 (62, 74)	68 (65, 72)	>0.9




```
# log rank test
survdiff(Surv(RBD_duration, status) ~ sex, data=df_surv)
```

```
             N Observed Expected (O-E)^2/E (O-E)^2/V
sex=Female 17        3     7.35      2.57      5.78
sex=Male   37       13     8.65      2.19      5.78

 Chisq= 5.8  on 1 degrees of freedom, p= 0.02
```

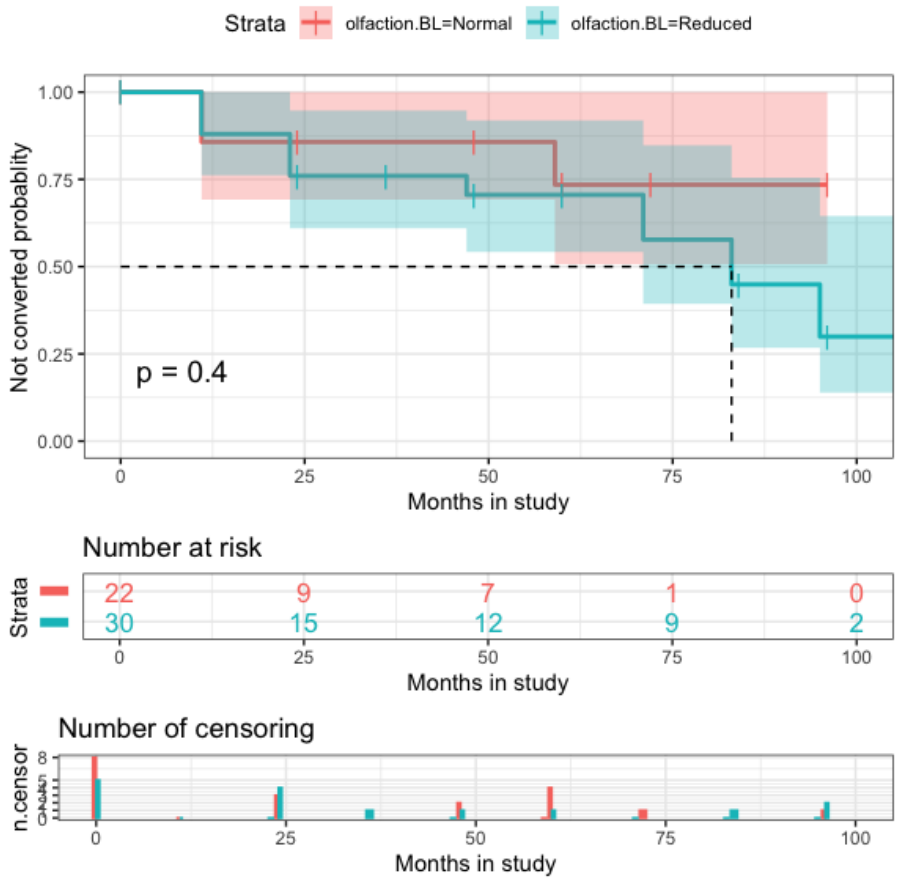



```
# log rank test
survdiff(Surv(months, status) ~ sex, data=df_surv)
```

```
             N Observed Expected (O-E)^2/E (O-E)^2/V
sex=Female 17        3     7.14      2.40      5.13
sex=Male   37       13     8.86      1.94      5.13

 Chisq= 5.1  on 1 degrees of freedom, p= 0.02
```

Survival analysis

Reduced olfaction = baseline SST-ID age- and sex-adjusted percentile <= 10%

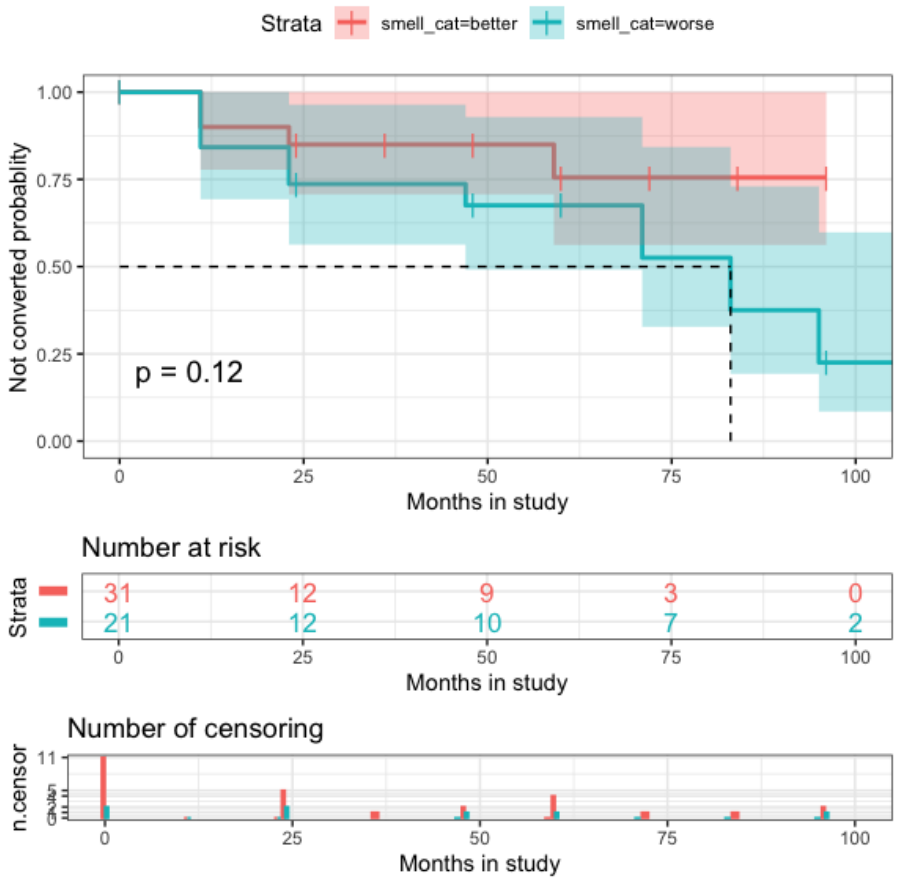


n=52, 2 observations deleted due to missingness.

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
olfaction.BL=Normal	22	3	4.41	0.449	0.721
olfaction.BL=Reduced	30	13	11.59	0.171	0.721

Chisq= 0.7 on 1 degrees of freedom, p= 0.4

Worse olfaction = baseline SST-ID score < 5.5 (optimal threshold by Youden index)



AUC: 0.776 (0.645-0.907)

Threshold: 5.5

Sensitivity: 0.75

Specificity: 0.75

n=52, 2 observations deleted due to missingness.

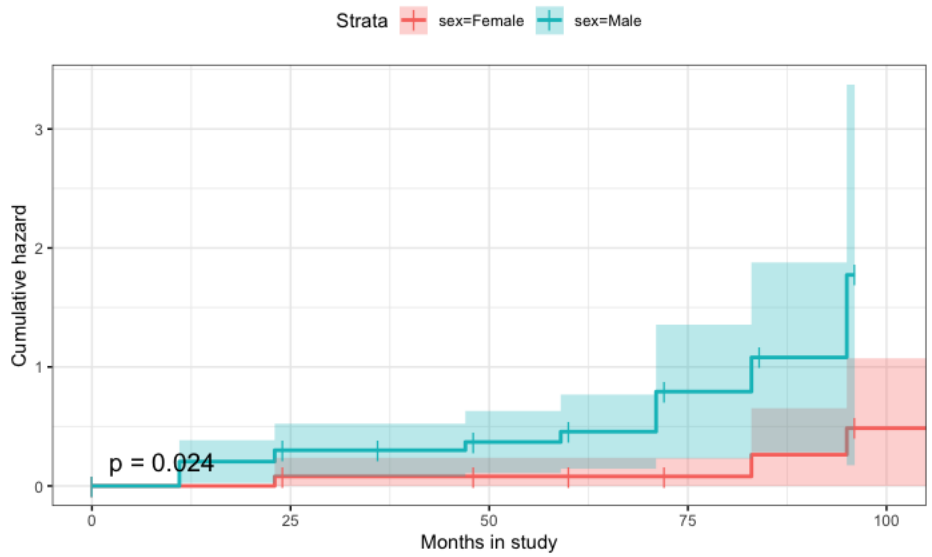
	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
smell_cat=better	31	4	6.89	1.213	2.44
smell_cat=worse	21	12	9.11	0.918	2.44

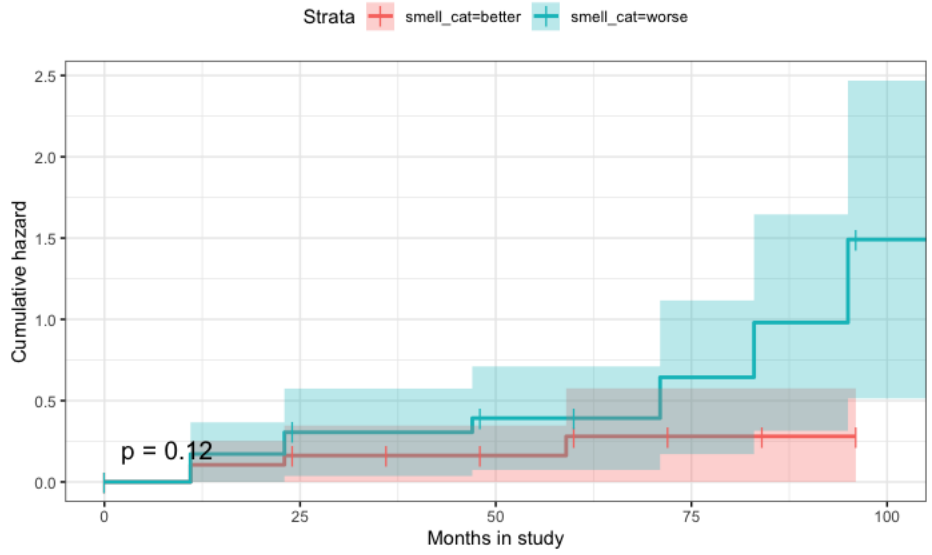
Chisq= 2.4 on 1 degrees of freedom, p= 0.1

Li, *et al*. unpublished data
https://stats.oarc.ucla.edu/wp-content/uploads/2025/02/survival_r_full.html

Survival analysis: hazard

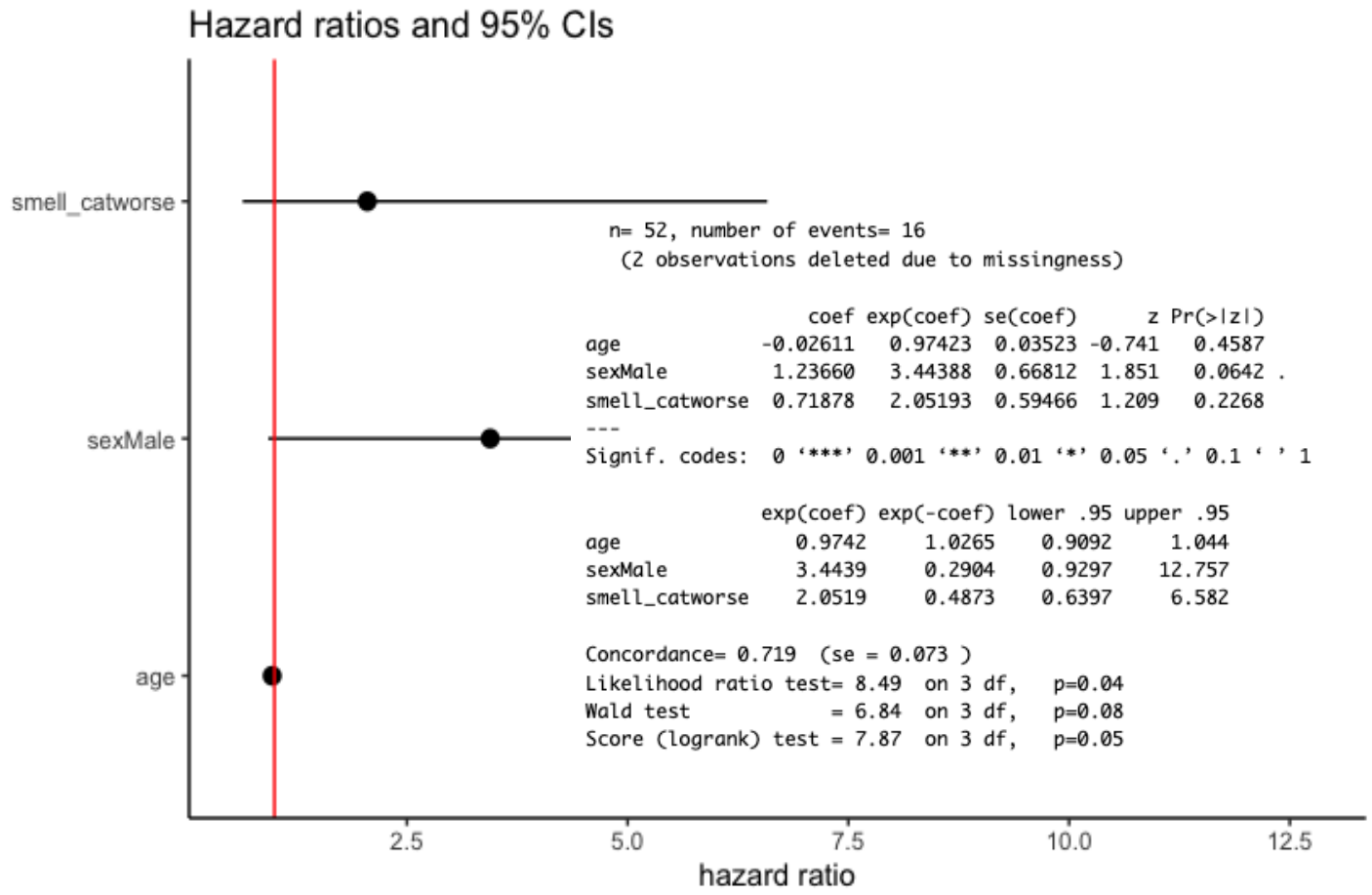
"cumhaz" plots the cumulative hazard function (f(y) = -log(y))





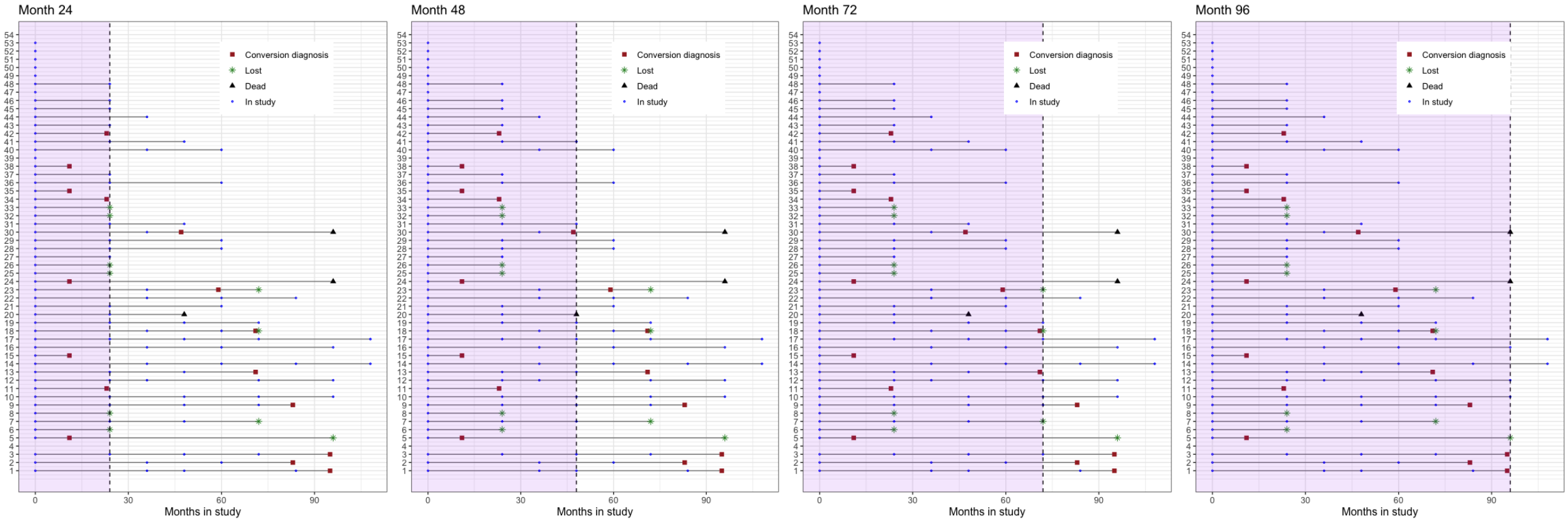
The Cox proportional hazards model

```
df_cph <- df_surv %>%
  mutate(status = status + 1) # the Surv() function accepts a status variable with 1=censored and 2=event

fit.cox <- coxph(Surv(months, status) ~ age + sex + smell_cat, data=df_cph)
# summary of results
summary(fit.cox)

# Get the Baseline Cumulative Hazard
baseline_hazard <- basehaz(fit.cox)
```




```
n= 52, number of events= 16
  (2 observations deleted due to missingness)

                coef exp(coef) se(coef)      z Pr(>|z|)
age          -0.02611   0.97423  0.03523 -0.741   0.4587
sexMale       1.23660   3.44388  0.66812  1.851   0.0642 .
smell_catworse 0.71878   2.05193  0.59466  1.209   0.2268
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

               exp(coef) exp(-coef) lower .95 upper .95
age               0.9742     1.0265    0.9092     1.044
sexMale           3.4439     0.2904    0.9297    12.757
smell_catworse    2.0519     0.4873    0.6397     6.582

Concordance= 0.719  (se = 0.073 )
Likelihood ratio test= 8.49  on 3 df,   p=0.04
Wald test            = 6.84  on 3 df,   p=0.08
Score (logrank) test = 7.87  on 3 df,   p=0.05
```

Cumulative case/dynamic control ROC – my implementation



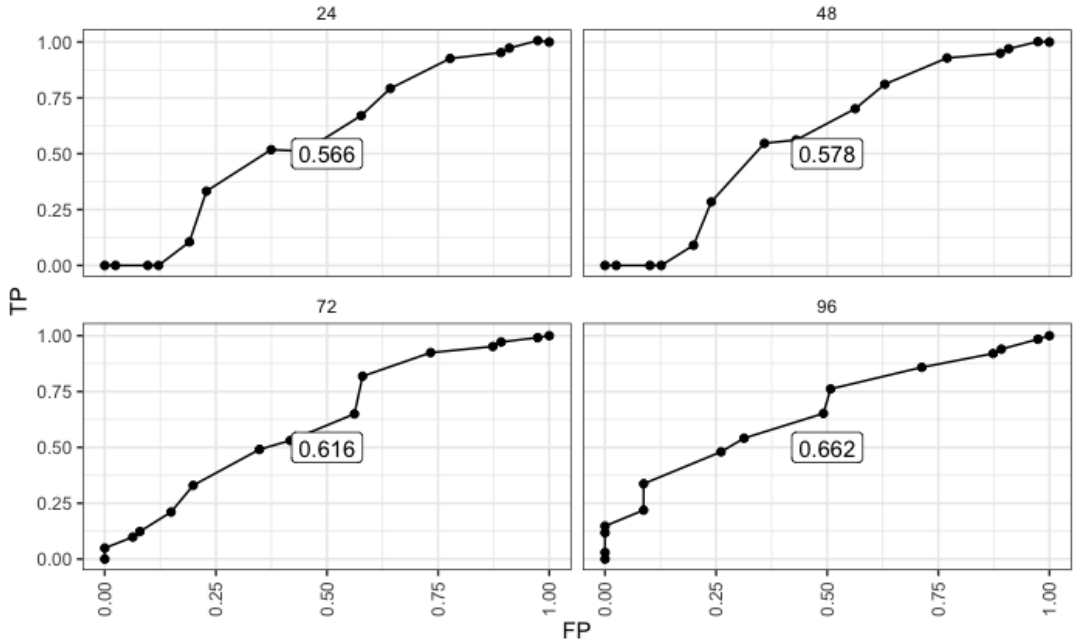
	24 months	48 months	72 months	96 months
Number at risk	39	31	23	21
Dynamic control	31	22	11	5
Cumulative case	8	9	12	16
Censored	13	21	29	31
SST-ID baseline: AUC (95% CI)	0.573 (0.372-0.773)	0.556 (0.354-0.757)	0.451 (0.204-0.697)	0.644 (0.361-0.927)

* Two participants are removed due to missing baseline SST-ID scores.

Li, *et al*. unpublished data

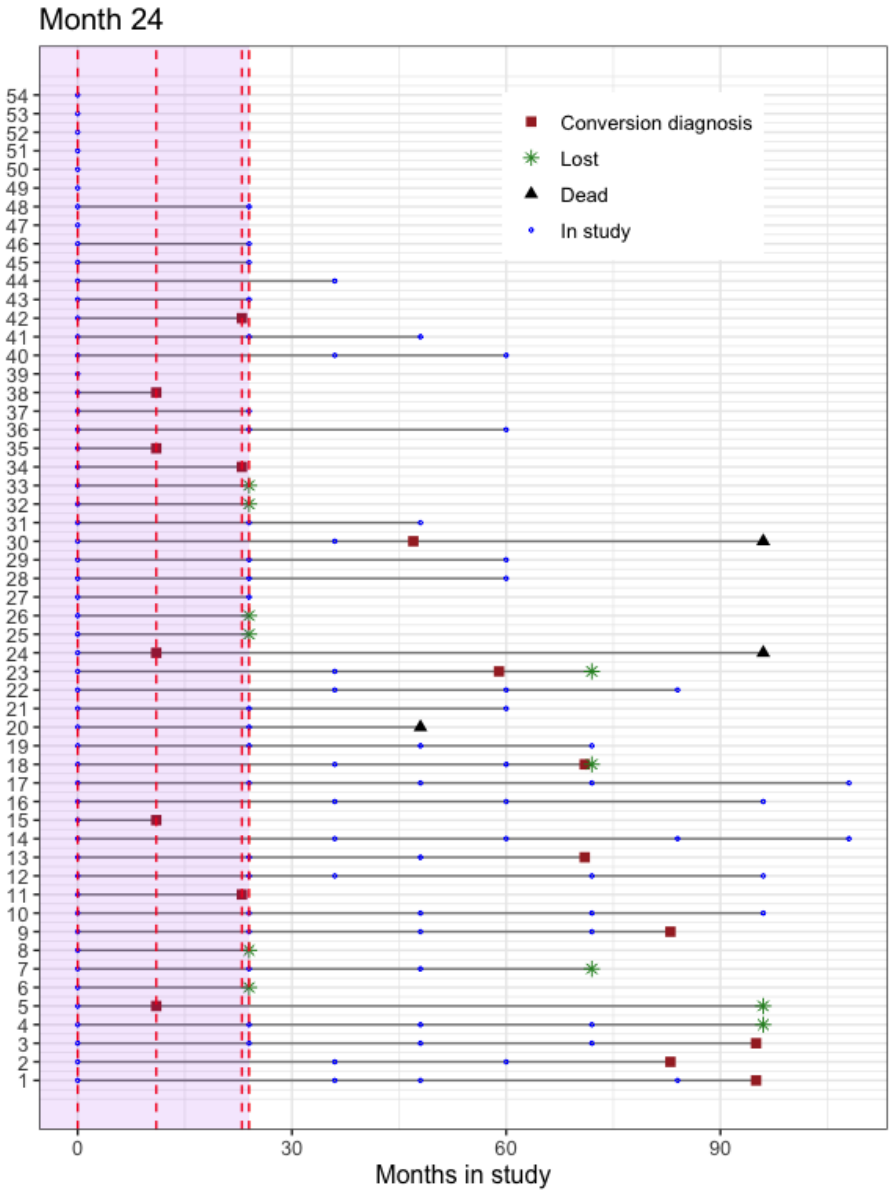
https://datascienceplus.com/time-dependent-roc-for-survival-prediction-models-in-r/

Cumulative case/dynamic control ROC – `survivalROC`



	0	11	23	24
Number at risk (n_t)	52	39	34	31
Number of conversion (d_t)	0	5	3	0
Survival ($s_0 = 1 - \sum_t d_t/n_t$)	1	0.87	0.79	0.79

Instead of just looking at the estimation time point (24 months), it checks every available time point before 24 months and pools the result (see View(survivalROC)).



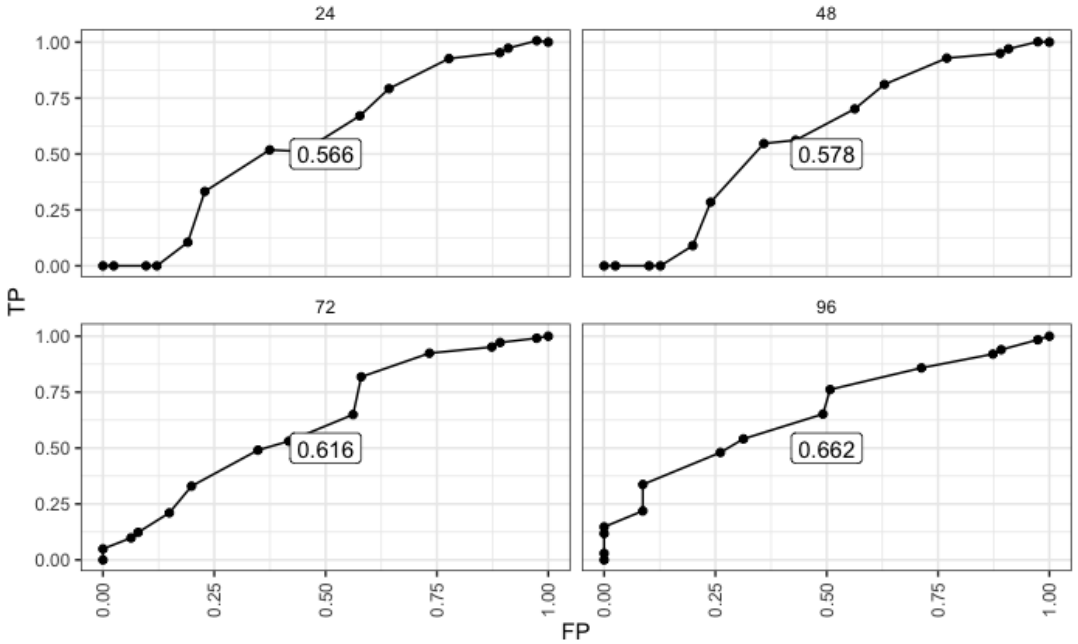
Cumulative case/dynamic control ROC – `survivalROC` , technical detail

Cumulative case/dynamic control ROC

The cumulative case/dynamic control ROC[2] defines sensitivity and specificity at time *t* at a threshold *c* as follows.

$$\text{sensitivity}^{\mathbb{C}}(c,t) = P(M_i > c | T_i \leq t)$$
$$\text{specificity}^{\mathbb{D}}(c,t) = P(M_i \leq c | T_i > t)$$

The cumulative sensitivity considers those who have died by time *t* as the denominator (disease) and those who have a marker value higher than *c* among them as true positives (positive in disease). The dynamic specificity regards those who are still alive at time *t* as the denominator (health) and those who have a marker value less than or equal to *c* among them as true negatives (negative in health). Varying the threshold value *c* from the lowest value to the highest value gives the entire ROC curve at time *t*. The definitions are based on the latent event time T_i, which is not observed for censored individuals. Estimation must account for censoring.[2] This is one gives the prediction performance for the *risk* (cumulative incidence) of events over the *t*-year period.




```r
library(survivalROC)
library(purrr) # map, map_dbl

## Define a helper functio nto evaluate at various t
survivalROC_helper <- function(t) {
    survivalROC(Stime        = df_surv$months,
                status       = df_surv$status,
                marker       = -df_surv$SST.ID.BL, # for the ROC direction
                predict.time = t,
                method       = "KM")
}

## Evaluate at evaluation_time
survivalROC_data <- data_frame(t = evaluation_time) %>%
    mutate(survivalROC = map(t, survivalROC_helper),
           ## Extract scalar AUC
           auc = map_dbl(survivalROC, magrittr::extract2, "AUC"),
           ## Put cut off dependent values in a data_frame
           df_survivalROC = map(survivalROC, function(obj) {
               as_data_frame(obj[c("cut.values","TP","FP")])
           })) %>%
    dplyr::select(-survivalROC) %>%
    unnest() %>%
    arrange(t, FP, TP)

## Plot
survivalROC_data %>%
    ggplot(mapping = aes(x = FP, y = TP)) +
    geom_point() +
    geom_line() +
    geom_label(data = survivalROC_data %>% dplyr::select(t,auc) %>% unique,
               mapping = aes(label = sprintf("%.3f", auc)), x = 0.5, y = 0.5) +
    facet_wrap( ~ t) +
    theme_bw() +
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5),
          legend.key = element_blank(),
          plot.title = element_text(hjust = 0.5),
          strip.background = element_blank())
```

https://datascienceplus.com/time-dependent-roc-for-survival-prediction-models-in-r/

Cumulative case/dynamic control ROC – the `tidymodels` approach

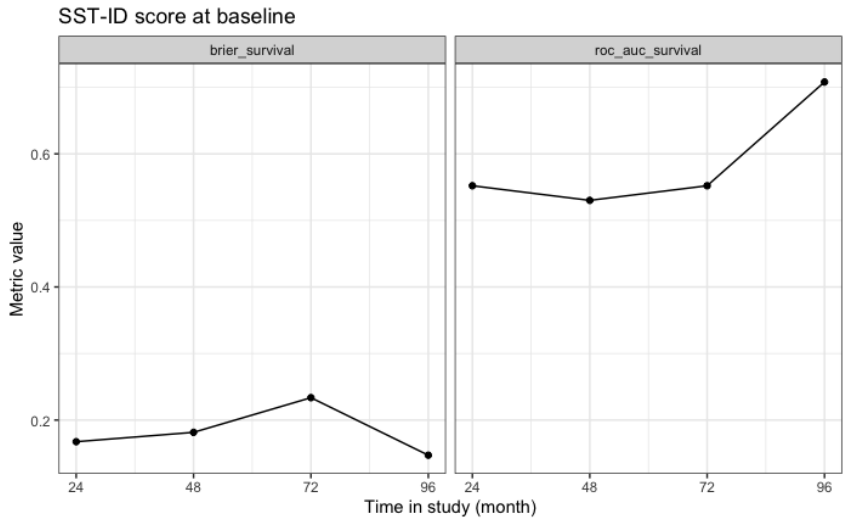
```
library(tidymodels)
library(censored)
```

```{r}
df_surv_tm <- df_surv %>%
  # remove months == 0
  # https://stats.stackexchange.com/questions/176376/invalid-survival-times-for-this-distribution
  filter(months > 0) %>%
  mutate(conversion_surv = Surv(months, status == 1))

# define model
survreg_spec <- survival_reg() %>%
  set_engine("survival") %>%
  set_mode("censored regression")

# performance metrics
survival_metrics <- metric_set(brier_survival_integrated, brier_survival,
                               roc_auc_survival, concordance_survival)
```

```{r}
score_fit <- survreg_spec %>% fit(conversion_surv ~ SST.ID.BL, data = df_surv_tm)

score_pred <- augment(score_fit, df_surv_tm, eval_time = evaluation_time)

res <- survival_metrics(score_pred, truth = conversion_surv, .pred, estimate = .pred_time)
res

res <- res %>%
  filter(!is.na(.eval_time)) %>%
  droplevels()
ggplot(res, aes(.eval_time, .estimate)) +
  geom_line()+
  geom_point()+
  scale_x_continuous(breaks = evaluation_time) +
  facet_grid(~.metric)+
  labs(title = "SST-ID score at baseline",
       x = "Time in study (month)",
       y = "Metric value") +
  theme_bw()
```

This formulation takes survival probability predictions at one or more specific *evaluation times* and, for each time, computes the area under the ROC curve. To account for censoring, inverse probability of censoring weights (IPCW) are used in the calculations. See equation 7 of section 4.3 in Blanche *at al* (2013) for the details.

https://yardstick.tidymodels.org/reference/roc_auc_survival.html



	24 months	48 months	72 months	96 months
brier_survival_integrated	0.143			
brier_survival	0.167	0.182	0.234	0.147
concordance_survival	0.594 (same as estimated using `coxph`)			
roc_auc_survival	0.552	0.53	0.552	0.708

Li, *et al.* unpublished data
https://www.tidymodels.org/learn/statistics/survival-metrics/

Thank you