05_table1

randy

2022-03-22

```
`%!in%` <- Negate(`%in%`)

demog <- here::here("data", "epic", "Demog.csv") %>%
  read.csv() %>%
  janitor::clean_names() %>%
  dplyr::select(id = cffidno, sex,
         ethnic, mutation1,
         mutation2, race) %>%
  # Tue Apr 10 09:14:48 2022 ------------------------------
  mutate(mutation1 =
           case_when(mutation1 == "" ~ "Unknown",
                     TRUE ~ as.character(mutation1)),
         mutation2 =
           case_when(mutation2 == "" ~ "Unknown",
                     TRUE ~ as.character(mutation2))) %>%
  mutate(genotype =
           case_when(mutation1 == "F508del" &
                       mutation2 == "F508del" ~ "Two alleles F508del",
                     # mutation1 == "Unknown" &
                     # mutation2 == "F508del" ~ "Unknown",
                     # mutation2 == "Unknown" &
                     # mutation1 == "F508del" ~ "Unknown",
                     mutation1 %!in% c("Unknown", "F508del") &
                       mutation2 == "F508del" ~ "One allele F508del",
                     mutation2 %!in% c("Unknown", "F508del") &
                       mutation1 == "F508del" ~ "One allele F508del",
                     # mutation1 %!in% c("Unknown", "F508del") &
                     # mutation2 %!in% c("Unknown", "F508del") ~ "Other",
                     TRUE ~ "Others or Unknown")) %>%
  mutate(genotype = factor(genotype,
                           levels = c("Two alleles F508del",
                                      "One allele F508del",
                                      "Others or Unknown")))

# View(demog)
# levels(factor(demog$mutation1))
## Two alleles F508del
## One allele F508del
## Other or unknown

epic <- here::here("data", "epic",
                   "registration_age_min_3_4.csv") %>%
```

```
  read.csv(row.name = 1) %>%
  janitor::clean_names()
```

```
data <- left_join(epic, demog, by = "id") %>%
  mutate(sex = as.factor(sex))

ID <- unique(data$id)
length(ID) / 3
```

```
## [1] 456.6667
```

```
test <- sample(ID, 457, replace = FALSE)

data0 <- data %>%
  mutate(group =
           case_when(id %in% test ~ "testing",
                     TRUE ~ "training"))
# View(data0)
```

```
data1 <- data0 %>%
  group_by(id, group) %>%
  summarize(age_mean = mean(age),
            age_min = min(age),
            age_max = max(age),
            age_n = length(age),
            visitn = n(),
            h_mean = mean(ht),
            h_max = max(ht),
            h_min = min(ht),
            # Tue Apr  5 11:09:20 2022 ------------------------------
            ## add BMI at the baseline
            w_mean = mean(wt),
            w_max = max(wt),
            w_min = min(wt),
            sex = sex,
            genotype = genotype,
            ethnic = ethnic,
            race = race) %>%
  ungroup() %>%
  unique()
```

```
## `summarise()` has grouped output by 'id', 'group'. You can override using the
## `.groups` argument.
```

```
# data0 <- data %>%
#   group_by(id, sex, ethnic, genotype) %>%
#   nest()


data2 <- full_join(data1, data) %>%
  as.data.frame() %>%
```

```
  mutate(time = age - age_min,
         age_diff = age_max - age_min,
         BMI = wt / (0.1 * ht)^2)
```

```
## Joining, by = c("id", "sex", "genotype", "ethnic", "race")
```

```
# head(data2)
write.csv(data2, file = "data/epic_clean_randy.csv")
```

```
table1 <- data1 %>%
  unique() %>%
  dplyr::select(-id) %>%
  mutate(
    ethnic = case_when(ethnic == 1 ~ "Hispanic",
                       ethnic == 2 ~ "Non-Hispanic"),
    race = case_when(race == 1 ~ "White",
                     race != 1 ~ "Other"),
    sex = case_when(sex == "F" ~ "Female",
                    sex == "M" ~ "Male"),
    age_diff = age_max - age_min) %>%
  dplyr::select(group,

         Genotype = genotype,
         Gender = sex,
         Race = race,
         Ethnicity = ethnic,
         "Visit number" = visitn,
         "Age mean" = age_mean,
         "Age min" = age_min,
         "Age diff" = age_diff,
         "Age max" = age_max,
         "Height mean" = h_mean,
         "Height baseline" = h_min,
         "Weight mean" = w_mean) %>%
  ## select all the variables for table1
  tbl_summary(by = group) %>%
  ## just display all the variables in one column
  modify_header(label = "**Variable**") %>%
  # update the column header
  bold_labels() %>%
  italicize_labels() %>%
  as_flex_table() %>%
  flextable::bold(part = "header") %>%
  ## auto adjust the column widths
  flextable::autofit()
```

```
table1
```

```
## Warning: Warning: fonts used in ‘flextable‘ are ignored because the ‘pdflatex‘
## engine is used and not ‘xelatex‘ or ‘lualatex‘. You can avoid this warning
## by using the ‘set_flextable_defaults(fonts_ignore=TRUE)‘ command or use a
## compatible engine by defining ‘latex_engine: xelatex‘ in the YAML header of the
## R Markdown document.
```

Variable	testing, N = 457[1]	training, N = 913[1]
Genotype		
Two alleles F508del	256 (56%)	487 (53%)
One allele F508del	161 (35%)	316 (35%)
Others or Unknown	40 (8.8%)	110 (12%)
Gender		
Female	246 (54%)	447 (49%)
Male	211 (46%)	466 (51%)
Race		
Other	22 (4.8%)	40 (4.4%)
White	435 (95%)	873 (96%)
Ethnicity		
Hispanic	22 (5.0%)	21 (2.4%)
Non-Hispanic	417 (95%)	857 (98%)
Unknown	18	35
Visit number	44 (31, 60)	45 (32, 59)
Age mean	8.10 (6.64, 10.08)	8.12 (6.69, 9.93)
Age min	3.12 (3.05, 3.21)	3.14 (3.06, 3.23)
Age diff	9.7 (6.8, 12.8)	9.6 (7.1, 12.7)
Age max	12.9 (9.9, 16.0)	12.7 (10.3, 16.0)
Height mean	125 (116, 134)	125 (117, 135)
Height baseline	93.6 (91.0, 96.5)	94.0 (91.5, 97.0)
Weight mean	27 (22, 34)	27 (22, 34)

[1]n (%); Median (IQR)

```
## save pptx ------------------------------------------------
## flextable can be saved directly to powerpoints
flextable::save_as_pptx(
  table1,
  path = "figure/01_table1.pptx")
```

2 knots and bspline with cubic terms

- new table1 with two columns
  - include age_min & age_diff
  - time follow up time
  - training and testing
  - as different label for visit times
  - BMI, ht, wt

  - gender, r/eth, genotype

- split the data as 1/3 for testing and 2/3 for training
  - predicted value and the real obs for each subject in the testing set.
  - at least one observation for that individual

- cross validation GCV; as extra methodology for the model fitting
- use the predictive (dynamic predcition) as well as the marginal mean
- use the PML methods.