SISCER 2022: Longitudinal Data Analysis, MACS Data (partial solution)

Katie Wilson, Anna Plantinga (Instructors)   Yiqun Chen (TA)

Contents

This file provides an analysis in R of longitudinal data collected from the Multi-center Aids Cohort Study (MACS), a study that aims to characterize the time course of CD4 cell depletion (Kaslow et al. 1987; see also Fitzmaurice et al. 2018). This document is the "exercise" version where you get to practice analyzing the dataset (the designed exercises are indicated by the phrase **fill in your answers**).

Data Overview

This data set contains repeated measures on n=307 subjects that are living with HIV, the virus that causes AIDS. The data contains measurements of CD4 cells, an immunologic marker of the impact of HIV, taken at approximately semi-annual follow-up visits. The data contains measurements obtained in the first 4 years after acquisition and excludes subjects that died within the first 4 years.

There are several variables that give time information. First, the variable MONTHS measures the number of months since acquisition of HIV infection. The variable VTIME is the number of calendar months since January 1984. ATIME is the calendar time (months since 1/1984) at which a subject is diagnosed with AIDS, the disease caused by HIV. Note that although at MONTHS=0 a subject is diagnosed with HIV, there may be a long delay until the serious consequences of disease manifest – this latter time being the time of AIDS diagnosis. The variable ATIME is missing if a subject was not observed to be diagnosed with AIDS. Finally, DTIME is the calendar time of death (months since 1/1984) or follow-up time depending on the value of IDEATH.

Some subjects miss scheduled visits or choose to discontinue study participation and thus have fewer than the expected number of measurements.

Variables: (columns of the data file)

- ID = subject ID
- MONTHS = months since seroconversion (detection of HIV)
- AGE = age of subject
- CD4-COUNT = # of CD4 positive cells (helper cells) per mm^3
- CD8-COUNT = # of CD8 positive cells (suppressor cells) per mm^3
- VLOAD0 = viral load at baseline (copies per ml)
- AIDSCASE = 1 if no AIDS observed; 2 if AIDS observed; 3 if died prior to AIDS
- VTIME = calendar time of study visit in months since January 1984
- SCTIME = calendar time of seroconversion (detection of HIV) in months since 1/1984
- ATIME = calendar time of AIDS diagnosis in months since 1/1984
- DTIME = calendar time of death in months since 1/1984, or follow-up time
- IDEATH = indicator of death at DTIME (1=death, 0=censored)

Note 1: ATIME is missing (NA) if the time was not observed during study follow-up (i.e., subjects remained AIDS free and alive).

Note 2: There is a lower limit of detection for viral load and thus measurements at 300 reflect this detection limit.

Note 3: The ability to measure viral loads actually became available many years after the study was started, and for many subjects this measurement needed to be obtained from stored samples. Thus, not all subjects have a viral load at baseline (perhaps due to limited blood samples).

For the rest of the analysis, we will consider the baseline viral load categorized into the following groups (see Chapter 18 of van Belle et al.):

- Low viral load: baseline value less than 15×10^3
- Medium viral load: baseline value between 15×10^3 and 46×10^3
- High viral load: baseline value greater than 46×10^3

R packages

We first load the packages we will need for this module.

```
library(dplyr) # used for data manipulation
library(lattice) # used for visualizing longitudinal data
library(ggplot2) # used for plotting
library(corrplot) # used for visualizing cor mat
library(lmtest)
library(sandwich)
library(car)
library(nlme) # run mixed model
library(multcomp)
library(geepack) # run gee model
library(doBy) #
library(broom) # inferential results (CI) for gee
library(broom.mixed) # inferential results (CI) for mixed models
```

If you need to install any of the listed packages (e.g., if you do not have the `geepack` package installed, you might get an error message "Error in library(geepack) : there is no package called `geepack`". In this case, you can install the missing package by

```
install.packages(c('geepack'), repos='https://cloud.r-project.org')
```

Scientific Question

Is baseline viral load associated with rate of decline in CD4+ cells among men who are HIV+?

Exploratory Analyses

EDA approach

First up, we will look at baseline viral load. What is the distribution (mean, standard deviation, etc.)? What is the distribution of baseline viral load — how many individuals are in each category (low, medium, high)?

Next, we will look at CD4 counts, specifically focusing on how CD4 counts change over time. Subject-specific trajectories are helpful for seeing individual-level patterns in change. We could also look at the distribution of CD4 counts by year (we have 4 years of data) and at the correlation between them.

Finally, we will consider CD4 both over time and by baseline viral load. We could look at the distribution of CD4 counts by year by baseline viral load category (e.g., boxplots, table of mean and standard deviation).

Another important consideration is missing data. While the details of properly accounting for missing data are beyond the scope of this module, we will explore some aspects of it in exploratory data analysis. First, it is important to note how much is missing, when is it missing, specific patterns of missingness (e.g., dropout, where once patients have a missing outcome, all future outcomes are missing) and then investigate relationships with other variables. For example, does it appear that subjects with missing CD4 measurements tend to have higher baseline viral loads? Are they younger? Are the CD4 measurements that they do have lower than the CD4 measurements of the individuals who do complete the study?

Loading and Processing the Data

First we load in the data. Note that it is in the "long format," where each line corresponds to one measurement on an individual, so there are multiple rows per individual.

```
macs <- read.csv("../Datasets/macs.csv")[,-1]
head(macs, 5)
```

```
##     id months age cd4 cd8 vload0 aidscase vtime sctime atime dtime ideath
## 1 1022      6  27 391 300  70737        3    18     12    NA    66      1
## 2 1022     12  27 361 596  70737        3    24     12    NA    66      1
## 3 1022     16  28 288 845  70737        3    28     12    NA    66      1
## 4 1022     27  29 378 774  70737        3    39     12    NA    66      1
## 5 1022     33  29 197 868  70737        3    45     12    NA    66      1
```

```
## check the number of unique participants
length(unique(macs$id))
```

```
## [1] 307
```

Baseline Viral Load

For the purpose of this analysis, we will create a new variable for baseline viral load category. Note that because some participants are missing baseline viral load, we have to be careful with defining the categories.

```
macs$vload0_cat <- NA
macs$vload0_cat[macs$vload0 < 15000] <- "Low"
macs$vload0_cat[(macs$vload0 >= 15000) & (macs$vload0 < 46000)] <- "Medium"
macs$vload0_cat[macs$vload0 >= 46000] <- "High"
macs$vload0_cat[is.na(macs$vload0_cat)] <- "Missing"
macs$vload0_cat <- factor(macs$vload0_cat, levels=c("Low", "Medium", "High", "Missing"))
```

```
unique_vload0 <- unique(macs[, c("id", "vload0_cat")])
table(unique_vload0$vload0_cat)
```

```
##
##     Low  Medium    High Missing
##      75      76      77      79
```

```
prop.table(table(unique_vload0$vload0_cat))
```

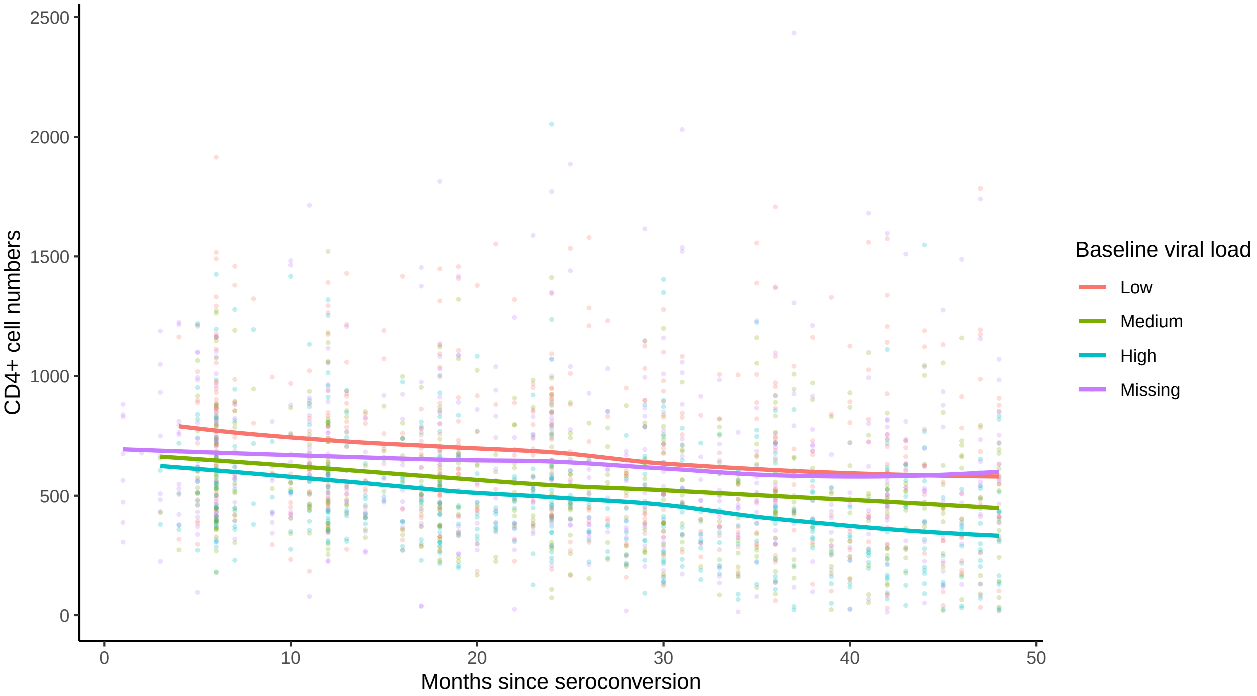
```
##
##       Low    Medium      High   Missing
## 0.2442997 0.2475570 0.2508143 0.2573290
```

From the table above, we see that 75 people (24.4%) are categorized as having low, 76 people (24.8%) having medium, and 77 people (25.1%) having high baseline viral load. The remaining 79 people (25.7%) are missing their baseline viral load.

Trajectory of CD4+ cells

We could first consider a simple time plot with a scatterplot smoother added by baseline viral load category:

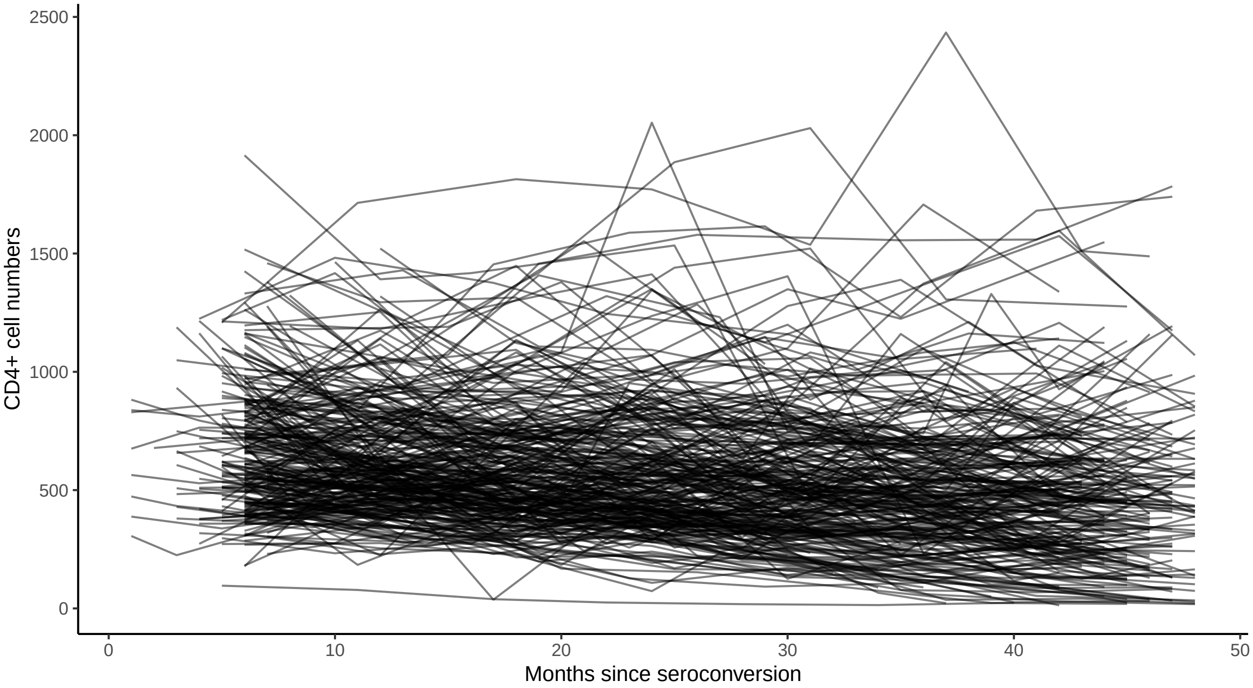
```
ggplot(macs, aes(x=months, y=cd4, group = vload0_cat, color = vload0_cat)) +
  geom_point(size=0.5, alpha = 0.25) +
  geom_smooth(se=F) +
  theme_classic() +
  ylab("CD4+ cell numbers") +
  xlab("Months since seroconversion")+
  scale_color_discrete(name = "Baseline viral load")
```



It appears there is a decrease in CD4+ cells over time (and it appears fairly linear). In addition, subjects with higher baseline viral loads tend to have a lower count of CD4+ cells on average. However, since the points for the same individual are not connected, we cannot determine right away whether the decreasing pattern holds for most individuals.

We can use the following "spaghetti plot" to visualize the individual trajectories.

```
ggplot(macs, aes(x = months, y=cd4, group=id)) +
  geom_line(alpha=0.5) +
  xlab("Months since seroconversion") +
  ylab("CD4+ cell numbers") +
  theme_classic()
```

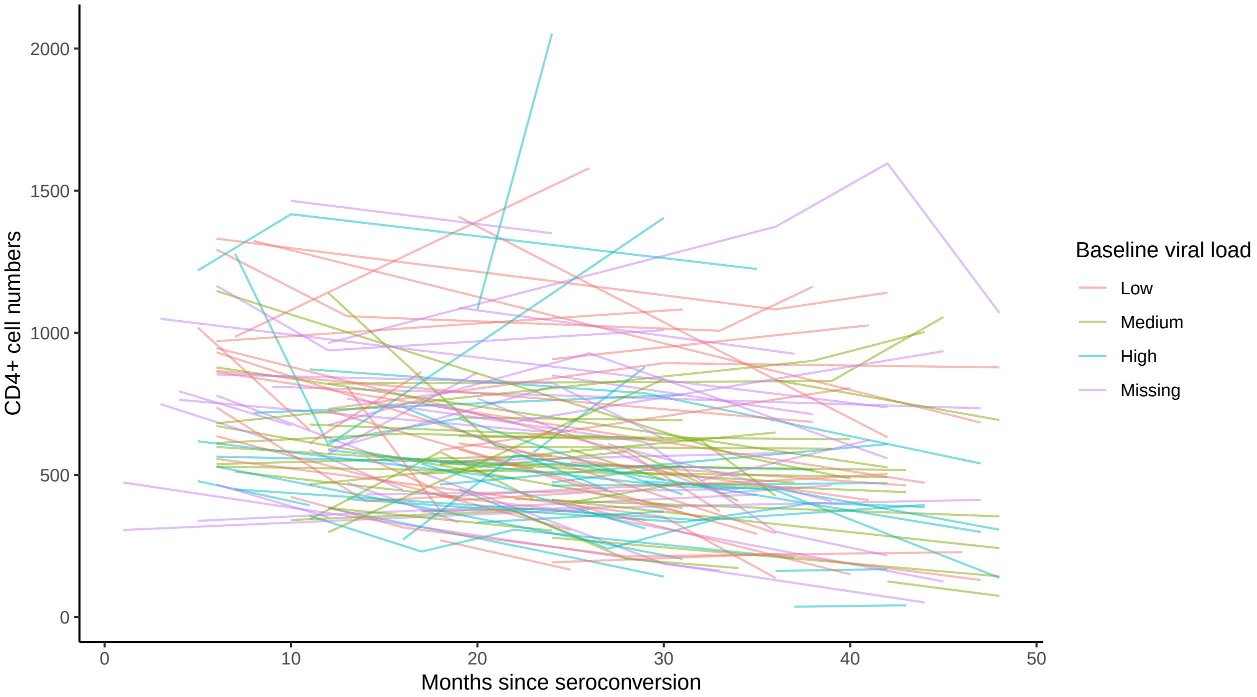


Given the large number of individuals, this plot is very busy and unhelpful. Instead, we will only highlight a few individuals. Which ones should we choose? We could select a random group or highlight ones with specific features. Here, for simplicity, we will randomly sample 20% of the participants from each of the baseline viral group.

```
set.seed(2022)

macs_subsample <- macs %>%
  group_by(vload0_cat) %>%
  sample_frac(0.2) %>%
  ungroup()

ggplot(macs_subsample, aes(x = months, y=cd4,
                           group=id, colour=vload0_cat)) +
  geom_line(alpha=0.5) +
  xlab("Months since seroconversion") +
  ylab("CD4+ cell numbers") +
  theme_classic()+
  scale_color_discrete(name = "Baseline viral load")
```



In general, we see "tracking" of the individual curves for these participants (the idea of "between-person variability" where participants tend to maintain their relative level of CD4 cell numbers, e.g., participants with higher levels tend to stay high); however, there does appear to be a bit of variability within an individual.

Distribution of CD4+ cells in years 1, 2, 3, and 4 following seroconversion

First, we create a variable `year` that indicates which year the measurement came from and obtain some summary statistics:

```
macs$year <- cut(macs$months, c(0, 12, 24, 36, 48))
table(macs$year, useNA = "always") # number of measurements in each year
```

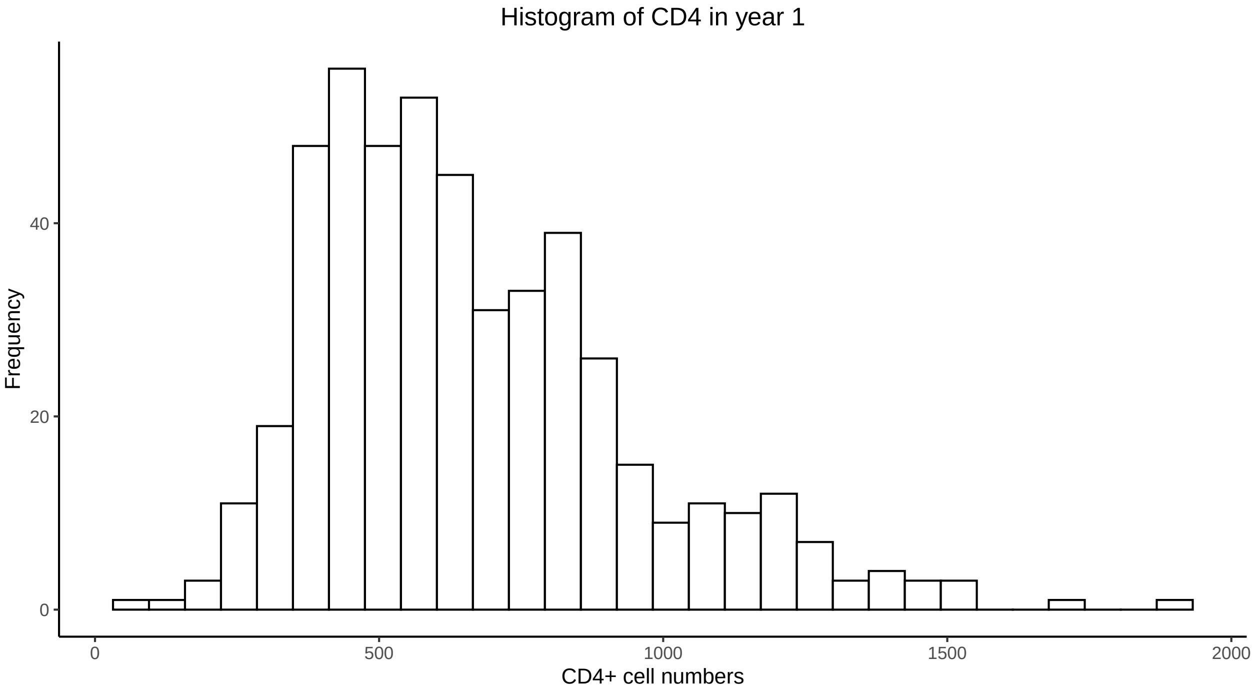
```
##
##  (0,12] (12,24] (24,36] (36,48]    <NA>
##     506     529     515     471       0
```

```
macs %>%
  group_by(year) %>%
  summarise(nobs = sum(!is.na(cd4)),
            nmiss = sum(is.na(cd4)),
            cd4_mean = mean(cd4, na.rm=T),
            cd4_sd = sd(cd4, na.rm=T))
```

```
## # A tibble: 4 x 5
##   year     nobs nmiss cd4_mean cd4_sd
##   <fct>   <int> <int>    <dbl>  <dbl>
## 1 (0,12]    493    13     668.   282.
## 2 (12,24]   526     3     613.   289.
## 3 (24,36]   513     2     546.   307.
## 4 (36,48]   469     2     499.   331.
```

We can also use histograms to visualize the distribution of CD4 in each year (see example for year 1 in the following code block).

```
ggplot(macs %>% filter(year=="(0,12]"),
       aes(x=cd4)) +
  geom_histogram(color="black", fill="white")+
  xlab("CD4+ cell numbers") +
  ylab("Frequency") +
  ggtitle("Histogram of CD4 in year 1")+
  theme_classic()+
  theme(plot.title = element_text(hjust = 0.5)) # center title
```



Next, we can obtain summary statistics (mean, sd) of CD4+ cell counts by baseline viral load category:

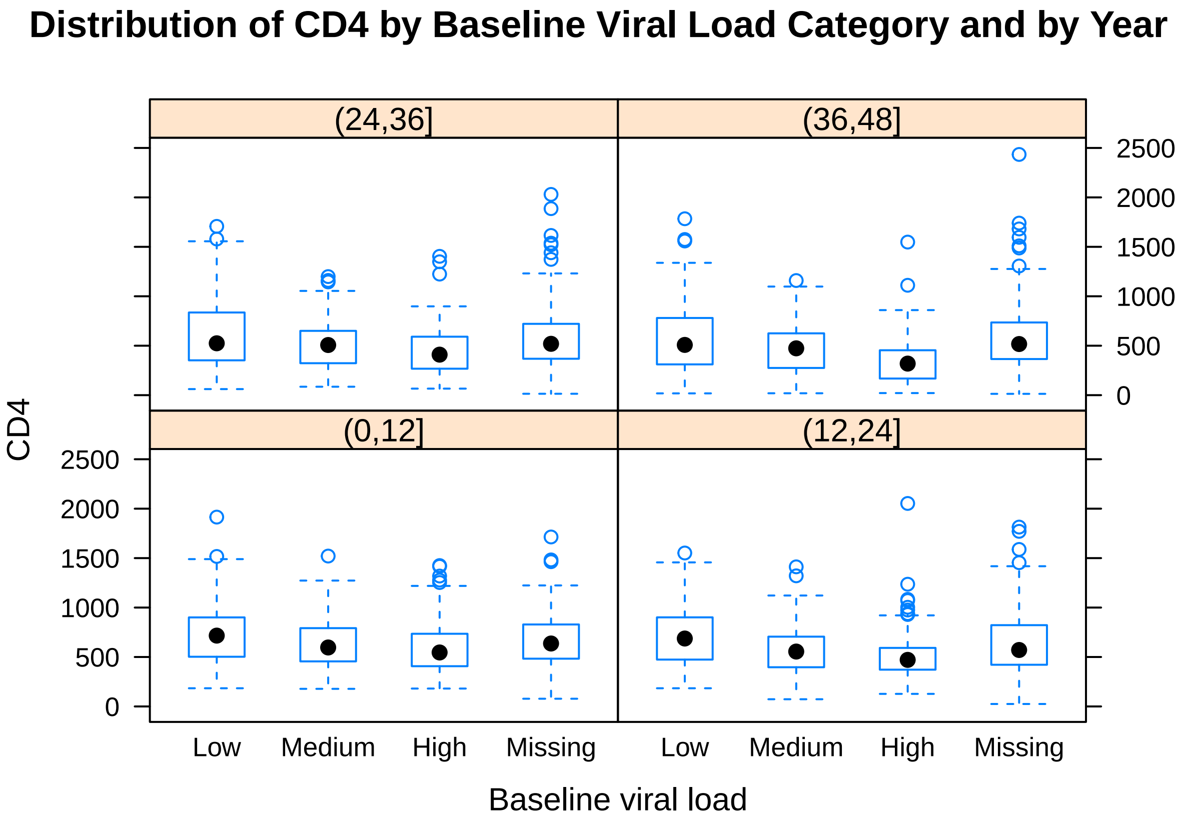
```
macs %>% group_by(id, year, vload0_cat) %>%
  summarise(cd4_year = mean(cd4,na.rm=T)) %>%
  group_by(year, vload0_cat) %>%
  summarise(nobs = sum(!is.na(cd4_year)),
            nmiss = sum(is.na(cd4_year)),
            cd4_mean = mean(cd4_year, na.rm=T),
            cd4_sd = sd(cd4_year, na.rm=T))
```

```
## # A tibble: 16 x 6
## # Groups:   year [4]
##    year    vload0_cat  nobs nmiss cd4_mean cd4_sd
##    <fct>   <fct>      <int> <int>    <dbl>  <dbl>
##  1 (0,12]  Low           73     2     761.   305.
##  2 (0,12]  Medium        76     0     663.   256.
##  3 (0,12]  High          76     1     612.   261.
##  4 (0,12]  Missing       78     1     687.   265.
##  5 (12,24] Low           70     0     703.   287.
##  6 (12,24] Medium        72     0     574.   226.
##  7 (12,24] High          74     1     510.   203.
##  8 (12,24] Missing       71     0     664.   311.
##  9 (24,36] Low           65     1     634.   318.
## 10 (24,36] Medium        69      0      510.   219.
## 11 (24,36] High          71      0      446.   222.
## 12 (24,36] Missing       73      0      600.   342.
## 13 (36,48] Low           61      0      575.   351.
## 14 (36,48] Medium        64      0      471.   246.
## 15 (36,48] High          65      0      349.   250.
## 16 (36,48] Missing       71      0      566.   339.
```

From this we can see that mean CD4+ cell count tends to be lower at later time points. Within a given year, those in the lower viral load category have the highest mean CD4+ cell counts, followed by medium, with those in the high category having the lowest mean CD4+ cell counts. Those with missing baseline viral load have a mean CD4+ cell count similar to the low and medium groups. Spread (as measured by standard deviation) tends to be greater with larger mean values.

Below, we visualize the distribution of CD4 counts by baseline viral category and year.

```
bwplot(cd4 ~ vload0_cat | year, data=macs,
       xlab="Baseline viral load", ylab="CD4",
       main="Distribution of CD4 by Baseline Viral Load Category and by Year")
```



Characterizing the correlation among CD4+ cell measurements

We will compute the pairwise correlations across CD4 measurements by year. Recall that some individuals have multiple measurements in a given year; for those individuals, we will first obtain the mean CD4 per person per year. Ultimately we want a dataset in the "wide format" where each row corresponds to an individual and there are columns for each year containing the mean CD4+ cell number.

```
mac_mean_cd4 <- macs %>% group_by(id, vload0_cat) %>%
  summarise(cd4_1 = mean(cd4[year == "(0,12]"], na.rm = TRUE),
            cd4_2 = mean(cd4[year == "(12,24]"], na.rm = TRUE),
            cd4_3 = mean(cd4[year == "(24,36]"], na.rm = TRUE),
            cd4_4 = mean(cd4[year == "(36,48]"], na.rm = TRUE))
head(mac_mean_cd4)
```

```
## # A tibble: 6 x 6
## # Groups:   id [6]
##      id vload0_cat cd4_1 cd4_2 cd4_3  cd4_4
##   <int> <fct>      <dbl> <dbl> <dbl>  <dbl>
## 1  1022 High        376   288   288.   39
## 2  1049 High        722.  430   326   204.
## 3  1120 Low        1144. 1305  1108. 1142
## 4  1164 Missing     714.  477   NaN   NaN
## 5  1214 Medium      847   766.  717   661.
## 6  1235 High        384   220   158.   28.5
```

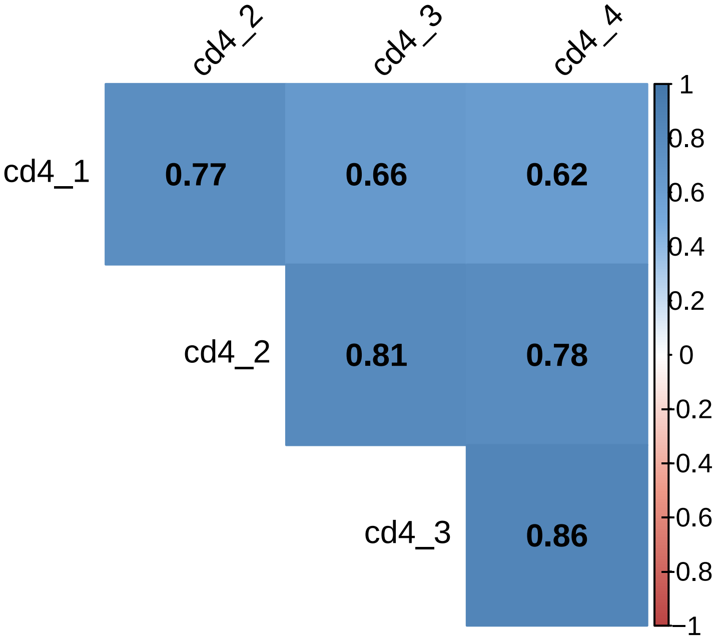
Next, we compute the correlation matrix using the `cor` function in `R`.

```
cor_mat <- cor(mac_mean_cd4[,c("cd4_1", "cd4_2", "cd4_3", "cd4_4")],
               use="pairwise.complete.obs")
cor_mat
```

```
##           cd4_1     cd4_2     cd4_3     cd4_4
## cd4_1 1.0000000 0.7663944 0.6578078 0.6222791
## cd4_2 0.7663944 1.0000000 0.8080873 0.7820780
## cd4_3 0.6578078 0.8080873 1.0000000 0.8581819
## cd4_4 0.6222791 0.7820780 0.8581819 1.0000000
```

Here is some code for creating a color-coded correlation plot:

```
col <- colorRampPalette(c("#BB4444", "#EE9988", "#FFFFFF", "#77AADD", "#4477AA"))
corrplot(cor_mat, method="color", col=col(200),
         type="upper", order="hclust",
         addCoef.col = "black", # Add coefficient of correlation
         tl.col="black", tl.srt=45, # Text label color and rotation
         diag=FALSE)
```



Overall, there appears to be fairly strong positive correlation among the CD4 counts for the same individual, with some evidence that it decreases with increasing time lag.

Missing Responses

Note that aside from those missing baseline viral load (a covariate), we have not investigated those missing CD4 (response). According to the data description, measurements of CD4 cells were taken approximately semi-annually for the first 4 years following HIV acquisition. In other words, a subject with "complete" observations should have roughly 8 CD4 counts. Additionally, the description notes that some subjects missed scheduled visits or choose to discontinue study participation and thus have fewer than the expected number of measurements. There are many things to consider when studying missing CD4 measurements. The easiest and first one would be to look at a simple count of the number of measurements each person has:

```
visit_counts <- macs %>% group_by(id) %>% summarise(n=n(), nobs=sum(!is.na(cd4)))
visit_counts
```

```
## # A tibble: 307 x 3
##       id     n  nobs
##    <int> <int> <int>
##  1  1022     6     6
##  2  1049     8     8
##  3  1120     8     8
##  4  1164     4     4
##  5  1214     8     8
##  6  1235     8     8
##  7  1259     6     6
##  8  1290     8     8
##  9  1308     7     6
## 10  1350     7     7
## # ... with 297 more rows
```

```
# number of visits with cd4 measurement available
table(visit_counts$nobs)
```

```
##
##   0   1   2   3   4   5   6   7   8   9
##   2   4  12   9   9  29  46  82 111   3
```

From this, we note that there are 114 individuals that have 8 or 9 CD4 measurements.

Bonus question: what could be a next step for examining the missing responses using exploratory analyses?

Inferential Analyses

Recall our scientific question of interest: is there an association between **baseline viral load** and **rate of decline** of CD4+ cell depletion?

Note: for the remainder of our analysis, we will consider what's known as the available data analysis; that is, we will simply ignore the observations with missing baseline viral loads.

Let Y_{ij} be the CD4+ cell counts for measurement j on individual i, t_{ij} be the month for individual i's jth measurement since sero-conversion, and vload0_i be the baseline viral load category for individual i. We will consider the following model:

$$E[Y_{ij}|t_{ij}, \text{vload0}_i] = \beta_0 + \beta_1 t_{ij} + \beta_2 I(\text{vload0}_i = \text{medium}) + \beta_3 I(\text{vload0}_i = \text{high}) + \quad (1)$$

$$\beta_4 t_{ij} \times I(\text{vload0}_i = \text{medium}) + \beta_5 t_{ij} \times I(\text{vload0}_i = \text{high}), \quad (2)$$

where our reference category is “low” baseline viral load and $I(A)$ is an indicator function that equals 1 if event A holds, and 0 otherwise.

Note that in the model above we include two interaction terms (corresponding to β_4 and β_5) because we are interested in understanding whether the **rate of decline** of CD4 is different for individuals who have medium (or high) viral load at baseline when compared to individuals with low viral load at baseline.

To help us understand the model better, consider the following questions:

- What is the rate of change/decline in CD4 cells for the group with low viral load at baseline? **Fill in your answers**
- What is the rate of change/decline in CD4 cells for the group with medium viral load at baseline? **Fill in your answers**
- What is the rate of change/decline in CD4 cells for the group with high viral load at baseline? **Fill in your answers**
- What is the null hypothesis corresponding to the scientific question of interest? **Fill in your answers**

Linear Mixed Models

In this section, we will demonstrate how to analyze the data using linear mixed models (LMM), which *explicitly* distinguishes between *between-subject* and *within-subject* sources of variability. In this case, our results from an available data analysis are valid provided that (i) data are missing at random (MAR); and (ii) the likelihood function of LMM is correctly specified.

As covered in lectures, there are many choices to make when fitting a linear mixed effects model:

- Random intercepts? Random slopes? Both?
- Maximum likelihood? REML?

For demonstration purposes we will use REML, which is also the default in the `lme` function in `R`. **Note**: recall that with REML, we *cannot* use the likelihood ratio test to compare two models with different fixed effects.

In terms of random intercepts vs slopes, recall that random intercepts allows each subject to have their own level, but the rate of change is the same. Random slopes allows each subject to have their own rate of change.

A reasonable starting point to model the data here might be to allow each participant to have their own level of CD4 and rate of change (i.e., we are fitting a model with random intercepts and slopes).

We will fit the model using categorized baseline viral load, as discussed earlier:

$$E[Y_{ij}|t_{ij}, \text{vload0}_i, b_{0i}, b_{1i}] = \beta_0 + b_{0i} + (\beta_1 + b_{1i})t_{ij} + \beta_2 I(\text{vload0}_i = \text{medium}) + \beta_3 I(\text{vload0}_i = \text{high}) + \beta_4 t_{ij} \times I(\text{vload0}_i = \text{medium}) + \beta_5 t_{ij} \times I(\text{vload0}_i = \text{high}),$$

where

$$\begin{pmatrix} b_{0i} \\ b_{1i} \end{pmatrix} \overset{i.i.d.}{\sim} \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix} \right). \tag{3}$$

The following code fits the model and generates inferential summaries (e.g., CI).

```
model_lmm <- lme(fixed=cd4 ~ months*vload0_cat,
            method="REML",
            random=reStruct(~1 + months| id, REML=TRUE),
            data=macs,
            subset=vload0_cat != "Missing",
            na.action=na.omit)
summary(model_lmm)
```

```
## Linear mixed-effects model fit by REML
##   Data: macs
##   Subset: vload0_cat != "Missing"
##        AIC      BIC    logLik
##   19691.47 19744.42 -9835.733
##
## Random effects:
##  Formula: ~1 + months | id
##  Structure: General positive-definite, Log-Cholesky parametrization
##             StdDev     Corr
## (Intercept) 247.133247 (Intr)
## months        5.719359 -0.439
## Residual    142.230867
##
## Fixed effects:  cd4 ~ months * vload0_cat
##                              Value Std.Error   DF   t-value p-value
## (Intercept)               800.0181  32.19760 1250 24.847135  0.0000
## months                     -5.2886   0.87136 1250 -6.069411  0.0000
## vload0_catMedium        -120.3072  45.20554  223 -2.661337  0.0083
## vload0_catHigh          -132.0101  45.16791  223 -2.922652  0.0038
## months:vload0_catMedium    0.2614   1.21698 1250  0.214769  0.8300
## months:vload0_catHigh     -2.3827   1.21438 1250 -1.962042  0.0500
##  Correlation:
##                         (Intr) months vld0_M vld0_H mn:0_M
## months                  -0.548
## vload0_catMedium        -0.712  0.390
## vload0_catHigh          -0.713  0.391  0.508
## months:vload0_catMedium  0.392 -0.716 -0.548 -0.280
## months:vload0_catHigh    0.393 -0.718 -0.280 -0.549  0.514
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -4.03856600 -0.51978455 -0.05957685  0.42249719  6.86257270
##
## Number of Observations: 1479
## Number of Groups: 226
```

```r
tidy(model_lmm, conf.int = TRUE)
```

```
## # A tibble: 10 x 10
##    effect   group   term  estimate std.error    df statistic    p.value conf.low
##    <chr>    <chr>   <chr>    <dbl>     <dbl> <dbl>     <dbl>      <dbl>    <dbl>
##  1 fixed    <NA>    (Int~  800.       32.2    1250    24.8    4.38e-111  737.
##  2 fixed    <NA>    mont~   -5.29      0.871  1250    -6.07    1.70e-  9   -7.00
##  3 fixed    <NA>    vloa~ -120.       45.2     223    -2.66    8.35e-  3 -209.
##  4 fixed    <NA>    vloa~ -132.       45.2     223    -2.92    3.83e-  3 -221.
##  5 fixed    <NA>    mont~    0.261     1.22   1250     0.215  8.30e-  1   -2.13
##  6 fixed    <NA>    mont~   -2.38      1.21   1250    -1.96    5.00e-  2   -4.77
##  7 ran_pars id      sd_(~  247.       NA        NA    NA     NA          220.
##  8 ran_pars id      cor_~   -0.439    NA        NA    NA     NA           -0.572
##  9 ran_pars id      sd_m~    5.72     NA        NA    NA     NA            4.91
## 10 ran_pars Residu~ sd_O~  142.       NA        NA    NA     NA           NA
## # ... with 1 more variable: conf.high <dbl>
```

From this model we can easily see what the estimated association between CD4 and time is for the low baseline viral load category: $\hat{\beta}_1$. To obtain estimates for the "Medium" and "High" categories, one option is to use the `multcomp` package:

```
model_lmm_med <- glht(model_lmm, linfct=c("months + months:vload0_catMedium = 0"))
confint(model_lmm_med)
```

```
##
##   Simultaneous Confidence Intervals
##
## Fit: lme.formula(fixed = cd4 ~ months * vload0_cat, data = macs, random = reStruct(~1 +
##     months | id, REML = TRUE), subset = vload0_cat != "Missing",
##     method = "REML", na.action = na.omit)
##
## Quantile = 1.96
## 95% family-wise confidence level
##
##
## Linear Hypotheses:
##                                        Estimate lwr     upr
## months + months:vload0_catMedium == 0 -5.0273  -6.6924 -3.3621
```

```
model_lmm_high <- glht(model_lmm, linfct=c("months + months:vload0_catHigh = 0"))
confint(model_lmm_high)
```

```
##
##   Simultaneous Confidence Intervals
##
## Fit: lme.formula(fixed = cd4 ~ months * vload0_cat, data = macs, random = reStruct(~1 +
##     months | id, REML = TRUE), subset = vload0_cat != "Missing",
##     method = "REML", na.action = na.omit)
##
## Quantile = 1.96
## 95% family-wise confidence level
##
##
## Linear Hypotheses:
##                                      Estimate lwr     upr
## months + months:vload0_catHigh == 0 -7.6713  -9.3291 -6.0135
```

Thus, we find the estimated rate of change in mean CD4 counts per month is:

- -5.3 (95% CI: -7.0, -3.6) among those with low baseline viral loads;
- -5.0 (95% CI: -6.7, -3.4) among those with medium baseline viral loads;
- -7.7 (95% CI: -9.3, -6.0) among those with high baseline viral loads.

Note that $\hat{\beta}_4$ and $\hat{\beta}_5$ are the estimated differences between the rates of change in the medium vs low (i.e. $-5.0-(-5.3)=\hat{\beta}_4$) and high vs low (i.e. $-7.7-(-5.3)=\hat{\beta}_5$).

Finally, we test for a statistically significant association between baseline viral load (category) and rate of change in CD4; that is, **Fill in your answers**.

```
print(anova(model_lmm)) # Wald test
```

```
##                   numDF denDF   F-value p-value
## (Intercept)           1  1250 1482.1123  <.0001
## months                1  1250  148.7498  <.0001
## vload0_cat            2   223   11.7130  <.0001
## months:vload0_cat     2  1250    2.9405  0.0532
```

We do not have sufficient evidence that the rate of change in mean CD4 over time depends on baseline viral load category (p = **Fill in your answers**).

Alternatively, if we fitted our models using ML (as opposed to REML), we can also apply a LRT:

```
model_lmm_ML <- lme(fixed=cd4 ~??,
            method="ML",
            random=??,
            data=macs,
            subset=vload0_cat != "Missing",
            na.action=na.omit)
model_lmm_ML_reduced <- lme(fixed=cd4 ~ ??,
                    method="ML",
                    random=??,
                    data=macs,
                    subset=vload0_cat != "Missing",
                    na.action=na.omit)

print(anova(model_lmm_ML, model_lmm_ML_reduced)) # LRT
```

Note: if you force anova to compare two REML-fitted objects with different fixed effects, you will get a warning message stating that "REML comparisons are not meaningful"!

Sensitivity Analysis: A sensitivity analysis could be done where we compare models with the same systematic trend, but different covariance structures. Then, using the "best fitting" model, see how our inference changes.

Here are some models that we may consider:

1. Random intercepts only
2. Random intercepts + random slopes, uncorrelated
3. Random intercepts + random slopes, correlated (`model_lmm`)

We will use AIC to determine the best fitting model: **Fill in your answers**

```
model_lmm_1 <- lme(fixed=??,
            method="REML",
            random=??,
            data=macs,
            subset=vload0_cat != "Missing",
            na.action=na.omit)
model_lmm_2 <- lme(fixed=??,
                   method="REML",
              random=??,
              data=macs,
              subset=vload0_cat != "Missing",
              na.action=na.omit)
print(anova(model_lmm_1, model_lmm_2, model_lmm))
```

It turns out that the third model is the best fitting one according to AIC or BIC (lowest value), which is the one we had selected in the beginning.

GEE

As an alternative to LME, we could consider modeling our data using GEE. The idea of GEE is that we treat the correlation structure as a nuisance – i.e., something that exists in the data but not of primary interest, and we need to acknowledge, but don't want to make assumptions about it. Our default will be to use empirical (also known as sandwich/robust) standard errors. Note that there is also a different approach to model fitting. GEE does *not* use a likelihood so there is no distinction of REML or ML, and likelihood ratio tests are *not* applicable.

Recall that we need to specify a *working covariance model* for GEE, which does not need to match the true covariance model for the regression coefficient estimate to be correct (in large samples). But if the working covariance is close to the true covariance, we can get efficiency gains (i.e., higher power). Recall that the parameter estimates and estimated standard errors do depend on the working covariance matrix (so you will get different, but valid results, depending on the working covariance model you choose). There is a stricter assumption regarding the missing data — for valid inference, data need to be missing completely at random (MCAR).

So what working covariance model should you use? As with pretty much every modeling choice, it depends on the application. If you have knowledge of the true correlation structure, it makes sense to include it. If not, working independence is *usually* OK for many practical purposes.

```
# geeglm assumes that the input data is organized by subject id and then time
macs <- macs %>%
  arrange(id, months)
```

Next we will fit the model using GEE with working independence model after removing the subjects with missing baseline viral data.

```
mod_gee_ind <- geeglm(cd4 ~ months*vload0_cat, id = id,
                data = macs %>%
                  filter(vload0_cat!="Missing") %>%
                  mutate(vload0_cat=factor(vload0_cat)),
                corstr = "independence")
print(anova(mod_gee_ind))
```

```
## Analysis of 'Wald statistic' Table
## Model: gaussian, link: identity
## Response: cd4
## Terms added sequentially (first to last)
##
##                   Df      X2 P(>|Chi|)
## months             1 108.346 < 2.2e-16 ***
## vload0_cat         2  19.549 5.688e-05 ***
## months:vload0_cat  2   4.405    0.1106
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Again we would conclude that we do not have evidence that baseline viral load (categorized) is associated with rate of decline in CD4 counts ($p = 0.11$). As we did with the linear mixed model formulation we can use the `glht()` function to get estimates and confidence intervals for linear combinations of the parameters (that way we can get estimates for the rate of change in mean CD4 in the medium and high groups):

```
tidy(mod_gee_ind, conf.int=T)
```

```
## # A tibble: 6 x 7
##   term                   estimate std.error statistic p.value conf.low conf.high
##   <chr>                     <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
## 1 (Intercept)             791.         36.8  462.     0          719.     864.
## 2 months                   -4.84        1.13  18.3     1.93e-5    -7.06     -2.62
## 3 vload0_catMedium       -123.         46.4    7.04    7.97e-3  -214.     -32.2
## 4 vload0_catHigh         -142.         46.3    9.34    2.24e-3  -232.     -50.8
## 5 months:vload0_catMedi~    0.122       1.40   0.00759 9.31e-1    -2.62      2.86
## 6 months:vload0_catHigh    -1.93        1.33   2.12    1.46e-1    -4.54      0.671
```

```
mod_gee_ind_med <- glht(mod_gee_ind, linfct=c("months + months:vload0_catMedium = 0"))
confint(mod_gee_ind_med)
```

```
##
##   Simultaneous Confidence Intervals
##
## Fit: geeglm(formula = cd4 ~ months * vload0_cat, data = macs %>% filter(vload0_cat !=
##     "Missing") %>% mutate(vload0_cat = factor(vload0_cat)), id = id,
##     corstr = "independence")
##
## Quantile = 1.96
## 95% family-wise confidence level
##
##
## Linear Hypotheses:
##                                         Estimate lwr     upr
## months + months:vload0_catMedium == 0 -4.7213  -6.3283 -3.1144
```

```
mod_gee_ind_high <- glht(mod_gee_ind, linfct=c("months + months:vload0_catHigh = 0"))
confint(mod_gee_ind_high)
```

```
##
##   Simultaneous Confidence Intervals
##
## Fit: geeglm(formula = cd4 ~ months * vload0_cat, data = macs %>% filter(vload0_cat !=
##     "Missing") %>% mutate(vload0_cat = factor(vload0_cat)), id = id,
##     corstr = "independence")
##
## Quantile = 1.96
## 95% family-wise confidence level
##
##
## Linear Hypotheses:
##                                       Estimate lwr     upr
## months + months:vload0_catHigh == 0 -6.7754  -8.1326 -5.4182
```

As a sensitivity analysis, we will fit the same model using GEE with working exchangeable model. **Fill in your answers**

```
mod_gee_exch <- geeglm(???)
print(anova(mod_gee_exch))
```

Not surprisingly, we do not have evidence that baseline viral load (categorized) is associated with rate of decline in CD4 counts (p = ?) **Fill in your answers**.

```
summary(mod_gee_exch)
```

In this case, we can also extract the estimated correlation between any two observations within a subject: 0.686 (given by the `alpha` estimate under "Estimated Correlation Parameters"). This indicates that the observations within the same subject are quite correlated empirically.

Note: Working independence or exchangeable correlation models are quite easy to apply to most study designs. In this case, because our data are unbalanced (i.e., measurement times for each individual are not the same) and there are many observations within each subject (most individuals have 4-8 observations), we cannot directly use other working covariance models such as the AR-1 model or the unstructured model.

Comparison of results: For categorized baseline viral load, the estimated rates of change in CD4 counts over time are included as below:

	LMM (rand int&slope)	GEE (working ind)
Low	-5.3 (-7.0, -3.6)	-4.8 (-7.1, -2.6)
Medium	-5.0 (-6.7, -3.4)	-4.7 (-6.3, -3.1)
High	-7.7 (-9.3, -6.0)	-6.8 (-8.1, -5.4)

Bonus: Fill in the same table but for the models we considered in the sensitivity analysis: **Fill in your answers**

	LMM (rand int) | G	EE (exchangeable)
Low		
Medium		
High		

References

Fitzmaurice, G. M., Laird, N. M., and Ware, J. H. (2018), Applied Longitudinal Analysis, Wiley & Sons, Limited, John.

Kaslow, R. A., Ostrow, D. G., Detels, R., Phair, J. P., Polk, B. F., and Rinaldo, C. R., Jr (1987), “The Multicenter AIDS Cohort Study: rationale, organization, and selected characteristics of the participants,” American Journal of Epidemiology, 126, 310–318. https://doi.org/10.1093/aje/126.2.310.

Van Belle, G., Fisher, L. D., Heagerty, P. J., & Lumley, T. (2004), Biostatistics: a methodology for the health sciences, John Wiley & Sons.