# Longitudinal Data Analysis Practice using the Multi-Center AIDS Cohort Study (MACS) dataset

Miles Xi

May, 2025

## Contents

## 1. Data Preparation

We model CD4 cell counts and viral load data from subjects who experienced seroconversion.

```
# Load libraries
library(haven)  # For reading Stata files

# Load lab result data, select CD4 cell count and viral load for each subject
lab = read_dta("~/Desktop/lab_rslt.dta")
lab = as.data.frame(lab[, c("CASEID", "VISIT", "LEU3N", "VLOAD")])  # "RO2VL", "R2SVL", "RTQ2VL", "RCOB

# Load HIV status data, select subjects who experienced seroconversion
status = as.data.frame(read_dta("~/Desktop/hivstats.dta"))
seroconversion = status[status$STATUS==4, ]

# From all subjects in 'lab', select those with seroconversion
data_match = lab[lab$CASEID %in% seroconversion$CASEID, ]

# Add a column, POSVIS, indicating the first positive visit
data_match$POSVIS = seroconversion$POSVIS[match(data_match$CASEID, seroconversion$CASEID)]

# Keep visits after or at seroconversion only
lab_post_sroconv = data_match[data_match$VISIT >= data_match$POSVIS, ]

# Get all subjects
subjects = unique(lab_post_sroconv$CASEID)
```

## 1.1 Subject Selection

Now, we'll filter out subjects whose viral load at seroconversion is missing (NA).

```
# Remove subjects whose viral load at seroconversion is NA
valid_subjects = c()  # Keep these
for (id in subjects){
  data_subset = lab_post_sroconv[lab_post_sroconv$CASEID == id, ]  # Select a subset of data
  viral_load_at_posvis = data_subset[1, 'VLOAD']   # The first (1) row
  if (!is.na(viral_load_at_posvis)){
    valid_subjects = c(valid_subjects, id)
  }
}

lab_post_sroconv = lab_post_sroconv[lab_post_sroconv$CASEID %in% valid_subjects, ]
```

## 1.2 Categorizing Viral Load

We'll categorize subjects based on their initial viral load into low, medium, and high groups.

```
# Determine the category (low, medium, high) of the viral load of each subject
get_category = function(vl){
  if (vl < 15000){
    return('low')
  } else if (vl > 46000){
    return('high')
  } else {
    return('medium')
  }
}

vl_class = data.frame(
  CASEID = numeric(),
  vload = numeric(),
  category = character()
)

for (id in valid_subjects){
  data_subset = lab_post_sroconv[lab_post_sroconv$CASEID == id, ]  # Get a subset by id
  viral_load = data_subset[1, 'VLOAD']  # Get the first viral load (vl)
  category = get_category(viral_load)
  vl_class = rbind(vl_class, data.frame(CASEID = id, vload=viral_load, category=category))
}

# Merge back to the main dataset, 'lab_post_sroconv'
lab_post_sroconv = merge(lab_post_sroconv, vl_class, by='CASEID')
```

### 1.3 Processing Time Information

Converting visit numbers to years since seroconversion for easier interpretation.

```
# Convert the visit number (e.g., 10, 20) to year (0, 1, 2)
lab_post_sroconv$year = ((lab_post_sroconv$VISIT - lab_post_sroconv$POSVIS) / 10) / 2
lab_post_sroconv$year = round(lab_post_sroconv$year * 2) / 2   # Round values such as 0.45 to 0.5
lab_post_sroconv$year_group = floor(lab_post_sroconv$year)  # Group two consecutive visits (e.g., 0.0 a
write.csv(lab_post_sroconv, "lab_post_sroconv.csv", row.names = FALSE)
```

## 2. Exploratory Data Analysis

We'll now analyze the processed data to understand CD4 count trends over time and by viral load category.

```
# Clear environment and load the processed data
rm(list = ls())
data = read.csv("lab_post_sroconv.csv")
data = data[data$year < 5, ]   # Only use observations of the first 4 years

# Exclude observations with less than 3 observations
data = subset(data, ave(CASEID, CASEID, FUN=length) > 2)
```

### 2.1 Group Means Over Time

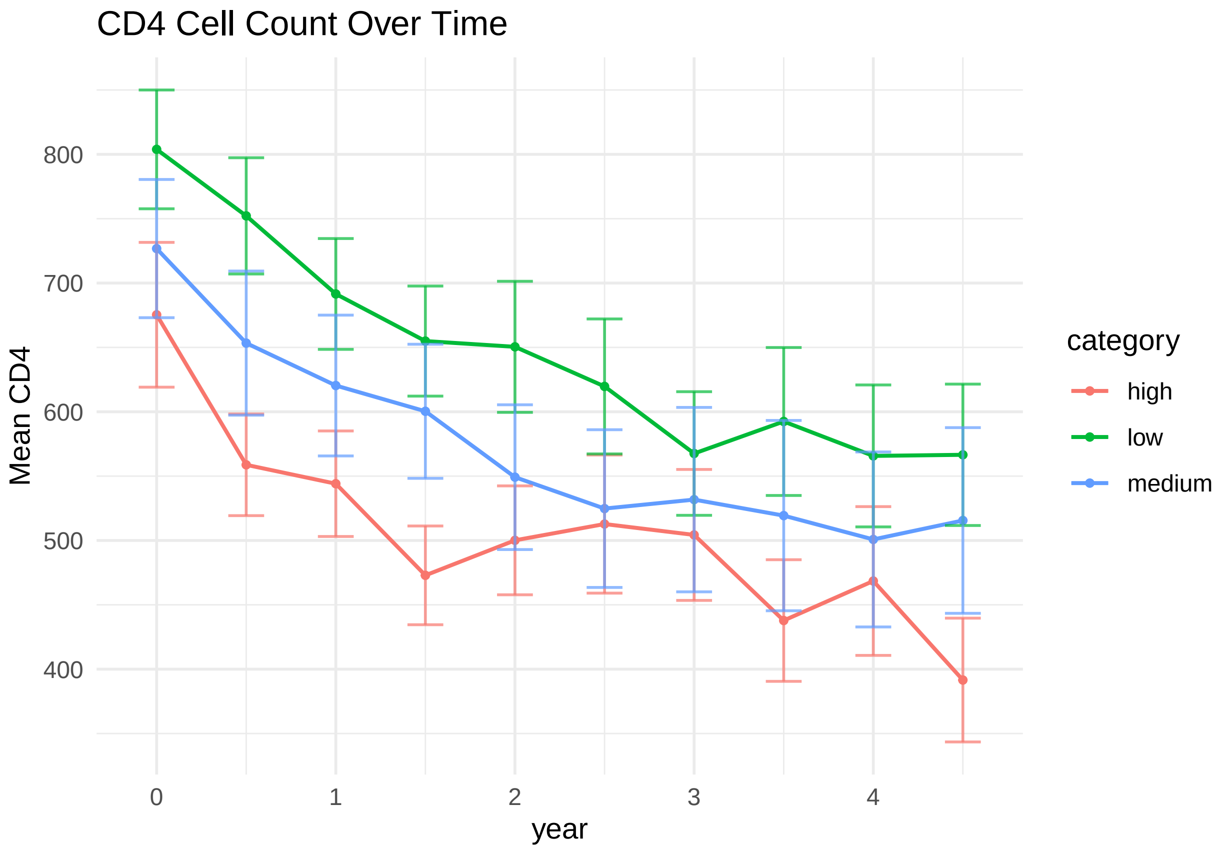
Examining how average CD4 counts change over time by viral load category.

```
# Average response (mean CD4 count), grouped by year & initial viral load
get_mean_and_se = function(x){
  mean = mean(x, na.rm=TRUE)
  se = sd(x) / sqrt(sum(!is.na(x)))
  n = sum(!is.na(x))
  return(c(mean=mean, se=se, n=n))
}
summary_tbl = aggregate(LEU3N ~ year + category,
                        data = data,
                        FUN = get_mean_and_se)
summary_tbl$mean_CD4 = summary_tbl$LEU3N[, "mean"]
summary_tbl$se_CD4 = summary_tbl$LEU3N[, "se"]
summary_tbl$n = summary_tbl$LEU3N[, "n"]
summary_tbl$LEU3N = NULL  # Drop the column

# Load ggplot2 for visualization
library(ggplot2)
```

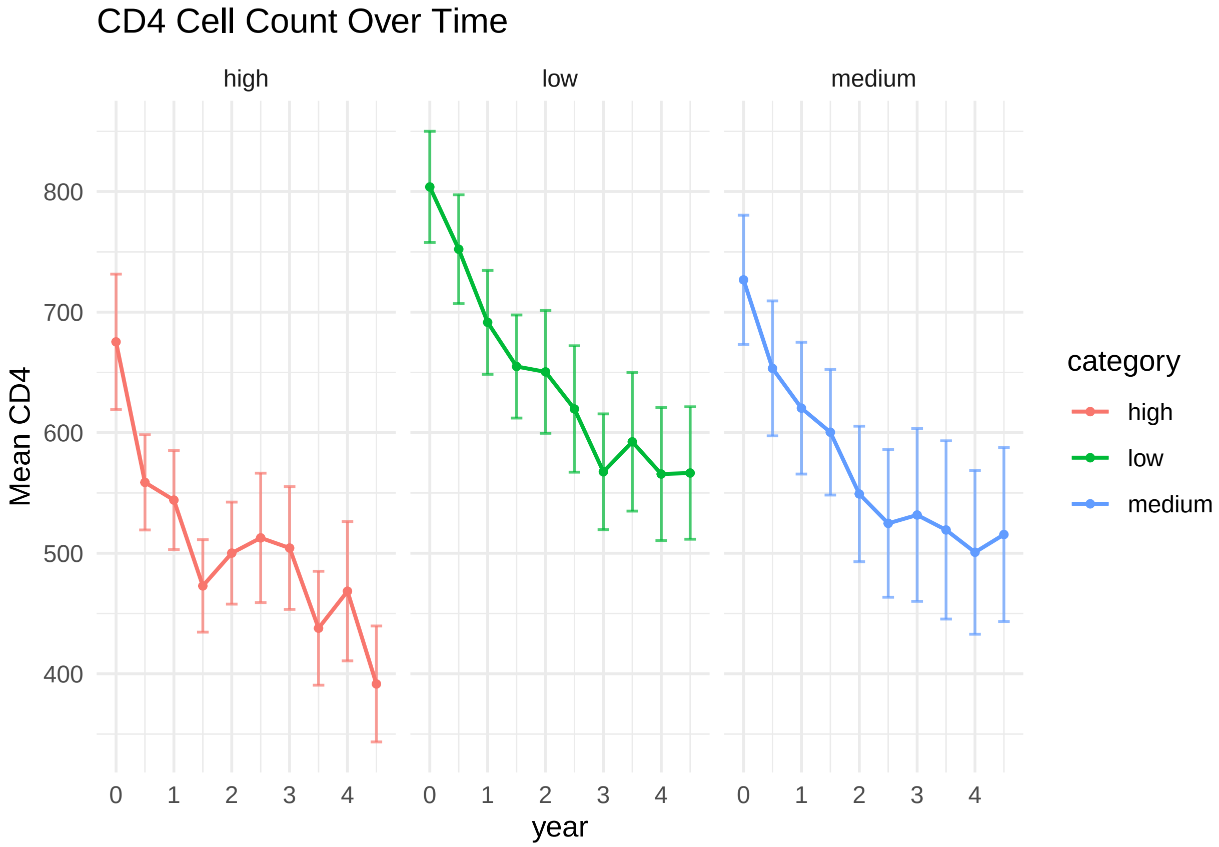
### Single Plot with Three Lines

```
ggplot(summary_tbl, aes(x=year, y=mean_CD4, color=category)) +
  geom_line(linewidth=0.7) +
  geom_point(size=1) +
  geom_errorbar(aes(ymin=mean_CD4 - 1.96*se_CD4, ymax=mean_CD4 + 1.96*se_CD4), width=0.2, alpha=0.7) +
  labs(title = "CD4 Cell Count Over Time", y='Mean CD4') +
  theme_minimal()
```



### Three Subplots for Each Category

```
ggplot(summary_tbl, aes(x=year, y=mean_CD4, color=category)) +
  geom_line(linewidth=0.7) +
  geom_point(size=1) +
  geom_errorbar(aes(ymin=mean_CD4 - 1.96*se_CD4, ymax=mean_CD4 + 1.96*se_CD4), width=0.2, alpha=0.7) +
  facet_wrap(~category) +
  labs(title = "CD4 Cell Count Over Time", y='Mean CD4') +
  theme_minimal()
```



### Summary Table

```
knitr::kable(summary_tbl[order(summary_tbl$year, summary_tbl$category), ],
             caption = "CD4 Count Summary by Year and Viral Load Category")
```

Table 1: CD4 Count Summary by Year and Viral Load Category

| | year | category | mean_CD4 | se_CD4 | n |
|---|---|---|---|---|---|
| 1 | 0.0 | high | 675.3462 | 28.70453 | 156 |
| 11 | 0.0 | low | 803.8614 | 23.54579 | 166 |
| 21 | 0.0 | medium | 726.7800 | 27.39069 | 100 |

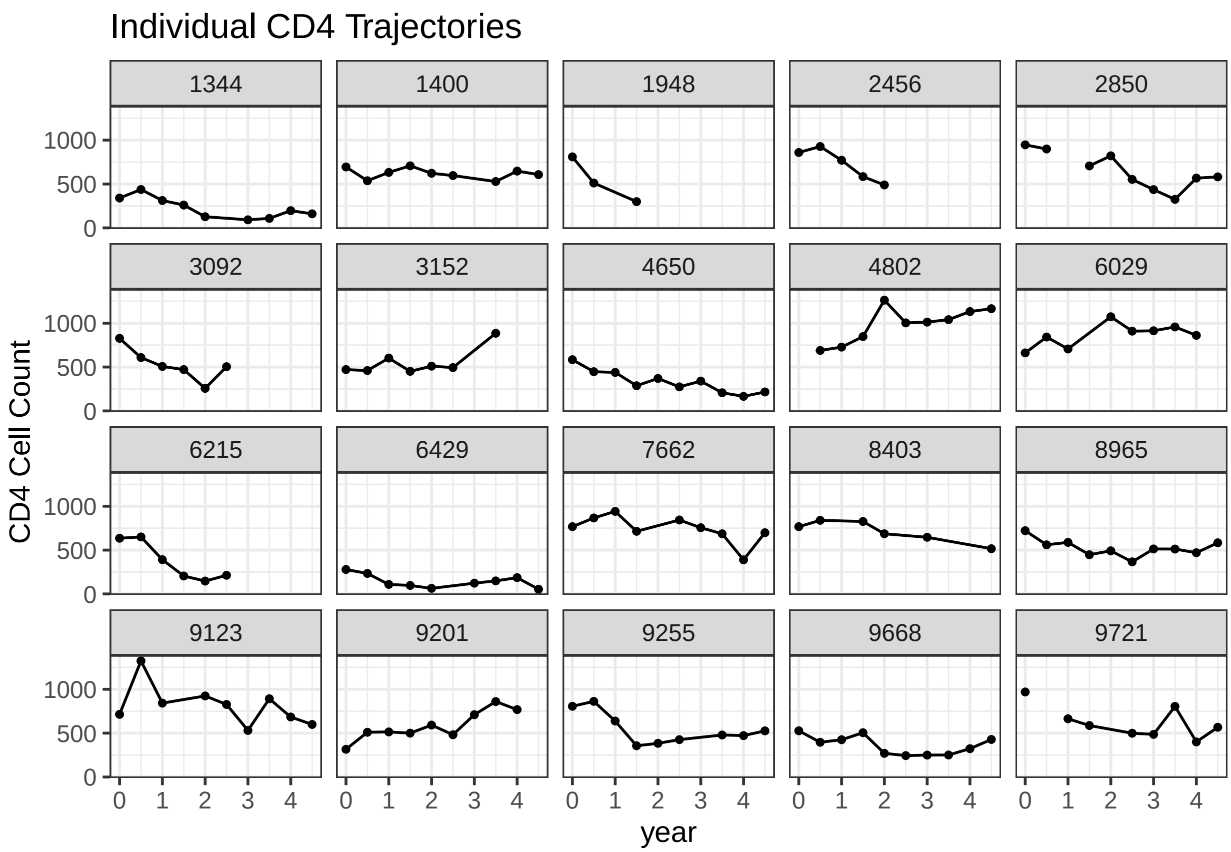
|  | year | category | mean_CD4 | se_CD4 | n |
|---|---|---|---|---|---|
| 2 | 0.5 | high | 558.7748 | 20.14670 | 151 |
| 12 | 0.5 | low | 752.1975 | 23.03967 | 162 |
| 22 | 0.5 | medium | 653.3474 | 28.56043 | 95 |
| 3 | 1.0 | high | 544.0921 | 20.91783 | 152 |
| 13 | 1.0 | low | 691.5329 | 21.96627 | 152 |
| 23 | 1.0 | medium | 620.3778 | 27.90526 | 90 |
| 4 | 1.5 | high | 472.8958 | 19.57039 | 144 |
| 14 | 1.5 | low | 654.9133 | 21.81179 | 150 |
| 24 | 1.5 | medium | 600.3617 | 26.57541 | 94 |
| 5 | 2.0 | high | 500.1049 | 21.58890 | 143 |
| 15 | 2.0 | low | 650.4521 | 25.97388 | 146 |
| 25 | 2.0 | medium | 549.1839 | 28.69277 | 87 |
| 6 | 2.5 | high | 512.7578 | 27.38669 | 128 |
| 16 | 2.5 | low | 619.6835 | 26.74527 | 139 |
| 26 | 2.5 | medium | 524.7733 | 31.26264 | 75 |
| 7 | 3.0 | high | 504.3025 | 25.95313 | 119 |
| 17 | 3.0 | low | 567.5672 | 24.49994 | 134 |
| 27 | 3.0 | medium | 531.7403 | 36.54966 | 77 |
| 8 | 3.5 | high | 437.7983 | 24.10839 | 119 |
| 18 | 3.5 | low | 592.4380 | 29.32318 | 121 |
| 28 | 3.5 | medium | 519.3194 | 37.72474 | 72 |
| 9 | 4.0 | high | 468.5000 | 29.49013 | 114 |
| 19 | 4.0 | low | 565.6935 | 28.13600 | 124 |
| 29 | 4.0 | medium | 500.7867 | 34.68386 | 75 |
| 10 | 4.5 | high | 391.5505 | 24.53584 | 109 |
| 20 | 4.5 | low | 566.5517 | 28.01962 | 116 |
| 30 | 4.5 | medium | 515.5286 | 36.80610 | 70 |

## 2.2 Individual Variation

Examining how CD4 counts vary among individuals.

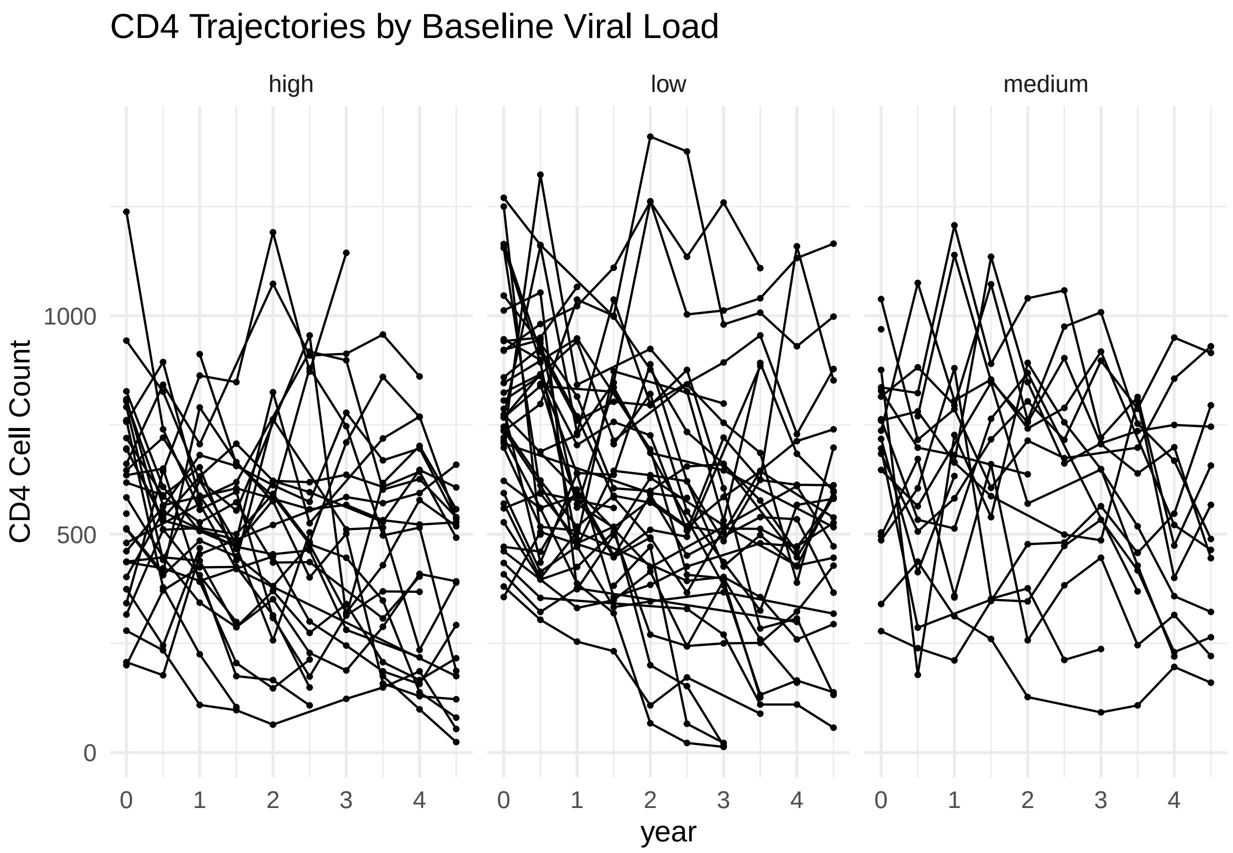
### Individual Trajectories

```r
# Plot the CD4 trajectories for randomly selected subjects
set.seed(19890604)
temp = subset(data,
              CASEID %in% sample(unique(data$CASEID), 20))
ggplot(temp, aes(x=year, y=LEU3N)) +
  geom_line(linewidth=0.5) +
  geom_point(size=0.9) +
  facet_wrap(~ CASEID) +
  labs(title = "Individual CD4 Trajectories", y='CD4 Cell Count') +
  theme_bw()
```



### Trajectories by Viral Load Category

```
# Plot individual series stratified by covariate group
set.seed(19890604)
temp = subset(data,
              CASEID %in% sample(unique(data$CASEID), 90))
ggplot(temp, aes(x=year, y=LEU3N, group=CASEID)) +
  geom_line(linewidth=0.4) +
  geom_point(size=0.5) +
  facet_wrap(~ category, ncol=3) +
  labs(title = "CD4 Trajectories by Baseline Viral Load", y='CD4 Cell Count') +
  theme_minimal()
```

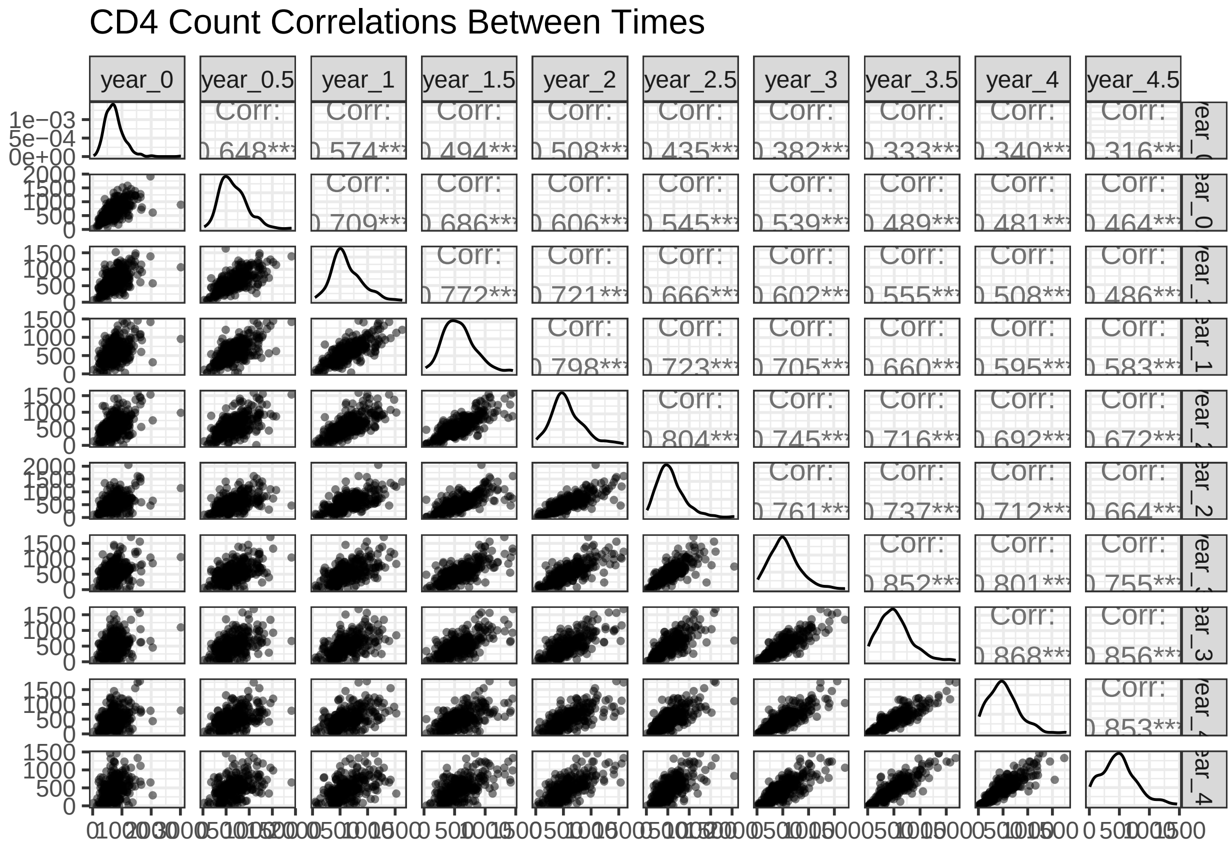


## 2.3 Correlation Analysis

Examining correlations between CD4 counts at different time points.

```
# Create an array of scatter plots showing Y's at year j vs. Y's at year k
library(tidyr)
temp = data[, c('CASEID', 'LEU3N', 'year')]
data_wide = pivot_wider(temp,
                        names_from = year,
                        values_from = LEU3N,
                        names_prefix = 'year_')
# data_wide = na.omit(data_wide)  # Not necessary

library(GGally)  # For ggpairs
ggpairs(data_wide[, -1],
        lower = list(continuous = wrap('points', size = 0.8, alpha = 0.5)),
        title = "CD4 Count Correlations Between Times"
        ) +
  theme_bw()
```


CD4 Count Correlations Between Times

The diagonal plots show the marginal distribution of the variables. The plot shows within-person correlations are high for observations close together in time, but the correlation tends to decrease with increasing time separation between the measurement times.

# 3. Derived Variable Analysis

## 3.1 Individual Slopes Analysis

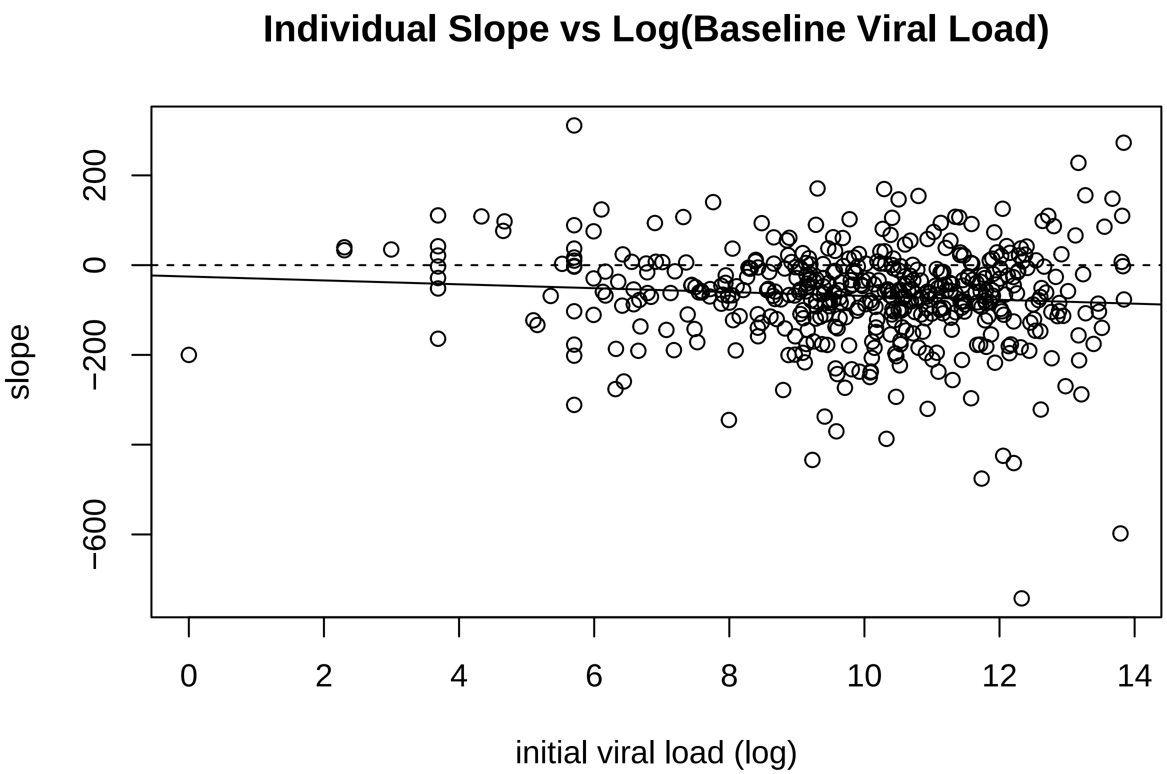
Regressing CD4 on time for each subject to analyze individual rate of change.

```r
# CD4_ij = beta0_i + beta1_i * t_ij + epsilon_ij
ids = unique(data$CASEID)
slopes = data.frame(
  CASEID = ids,
  slopes = numeric(length = length(ids))
  )

for (i in seq_along(ids)){
  temp = data[data$CASEID %in% ids[i], ]
  lm = lm(LEU3N ~ year, data=temp)
  slopes[i, 'slopes'] = coef(lm)[2]
}

slopes = merge(slopes, data[, c('CASEID', 'vload', 'category')], by='CASEID')
slopes = unique(slopes)

plot(slopes ~ log(vload), data=slopes,
     xlab = "initial viral load (log)", ylab = "slope",
     main = "Individual Slope vs Log(Baseline Viral Load)"
     )
abline(h = 0, lty = 2)
lm = lm(slopes ~ log(vload), data=slopes)  # Significant at 10%
abline(lm)
```



### 3.2 Mean Slopes by Category

Analyzing the average rate of CD4 decline by viral load category.

```
# The mean slopes grouped by the baseline viral load category
summarize_slopes_by_category = function(x){
  n = length(x)
  m = mean(x)
  se = sd(x) / sqrt(n)  # Fixed: srqt to sqrt
  return(c(mean=m, se=se, n=n))
}

slopes_tbl = aggregate(
  slopes ~ category, data=slopes, FUN = summarize_slopes_by_category
)
print(slopes_tbl)
```

```
##   category slopes.mean  slopes.se   slopes.n
## 1     high  -72.921144   9.620864 175.000000
## 2      low  -61.581050   7.513668 182.000000
## 3   medium  -71.699915   9.182269 104.000000
```

### 3.3 Data Attrition Analysis

Examining how many observations we have for each subject.

```
# Data attrition
obs_per_subject = table(data$CASEID)
obs_summary = table(obs_per_subject)
obs_summary = data.frame(
  'Number_of_obs' = as.integer(names(obs_summary)),
  'Number_of_subjects' = as.integer(obs_summary)
)
obs_summary
```

```
##   Number_of_obs Number_of_subjects
## 1             3                 28
## 2             4                 33
## 3             5                 34
## 4             6                 34
## 5             7                 37
## 6             8                 43
## 7             9                 96
## 8            10                157
```

## 4. Mixed Effects Regression Models

Using linear mixed effects models to account for individual variations while examining the effect of viral load category on CD4 count over time.

### 4.1 Random Intercept Model

```
# mu_ij := E(Y_ij | x_ij) = (beta0 + beta2 * L_ij + beta3 * M_ij) +
#                           (beta1 + beta4 * L_ij + beta5 * M_ij) * month_ij
# where M_ij, H_ij are dummies indicating the baseline viral load
# The first model is a random intercept model
# Y_ij = mu_ij + b_i0 + epsilon_ij

library(lme4)
data$category = as.factor(data$category)
lme1 = lmer(LEU3N ~ category*year + (1|CASEID), data=data)
summary(lme1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: LEU3N ~ category * year + (1 | CASEID)
##    Data: data
## 
## REML criterion at convergence: 48553.6
## 
## Scaled residuals: 
##     Min      1Q  Median      3Q     Max 
## -3.6592 -0.5648 -0.0790  0.4797  9.5467 
## 
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  CASEID   (Intercept) 48811    220.9   
##  Residual             33492    183.0   
## Number of obs: 3580, groups:  CASEID, 462
## 
## Fixed effects:
##                     Estimate Std. Error t value
## (Intercept)          603.001     18.975  31.779
## categorylow          167.557     26.545   6.312
## categorymedium        80.506     30.853   2.609
## year                 -48.511      3.698 -13.120
## categorylow:year      -6.000      5.128  -1.170
## categorymedium:year   -7.962      5.888  -1.352
## 
## Correlation of Fixed Effects:
##             (Intr) ctgryl ctgrym year   ctgryl:
## categorylow -0.715                             
## categorymdm -0.615  0.440                      
## year        -0.382  0.273  0.235               
## catgrylw:yr  0.276 -0.382 -0.169 -0.721        
## ctgrymdm:yr  0.240 -0.172 -0.374 -0.628  0.453 
```

### 4.2 Random Intercept and Slope Model

```r
# The 2nd model: Y_ij = mu_ij + b_i0 + b_i1 * month_ij + epsilon_ij
lme2 = lmer(LEU3N ~ category*year + (1+year|CASEID), data=data)
summary(lme2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: LEU3N ~ category * year + (1 + year | CASEID)
##    Data: data
## 
## REML criterion at convergence: 47929.6
## 
## Scaled residuals: 
##     Min      1Q  Median      3Q     Max 
## -4.1625 -0.4998 -0.0468  0.4558  9.4499 
## 
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr 
##  CASEID   (Intercept) 58094    241.0         
##           year         5761     75.9    -0.41
##  Residual             22244    149.1
## Number of obs: 3580, groups:  CASEID, 462
##
## Fixed effects:
##                      Estimate Std. Error t value
## (Intercept)           609.532     19.769  30.833
## categorylow           160.920     27.638   5.822
## categorymedium         80.226     32.139   2.496
## year                  -56.109      6.813  -8.235
## categorylow:year       -1.046      9.462  -0.111
## categorymedium:year    -6.420     11.028  -0.582
##
## Correlation of Fixed Effects:
##             (Intr) ctgryl ctgrym year   ctgryl:
## categorylow -0.715
## categorymdm -0.615  0.440
## year        -0.477  0.341  0.293
## catgrylw:yr  0.344 -0.476 -0.211 -0.720
## ctgrymdm:yr  0.295 -0.211 -0.471 -0.618  0.445
```

Note that the t-values for the interaction terms categorylow/medium:year dramatically decreased after adding the random slope.

### 4.3 Model Comparison

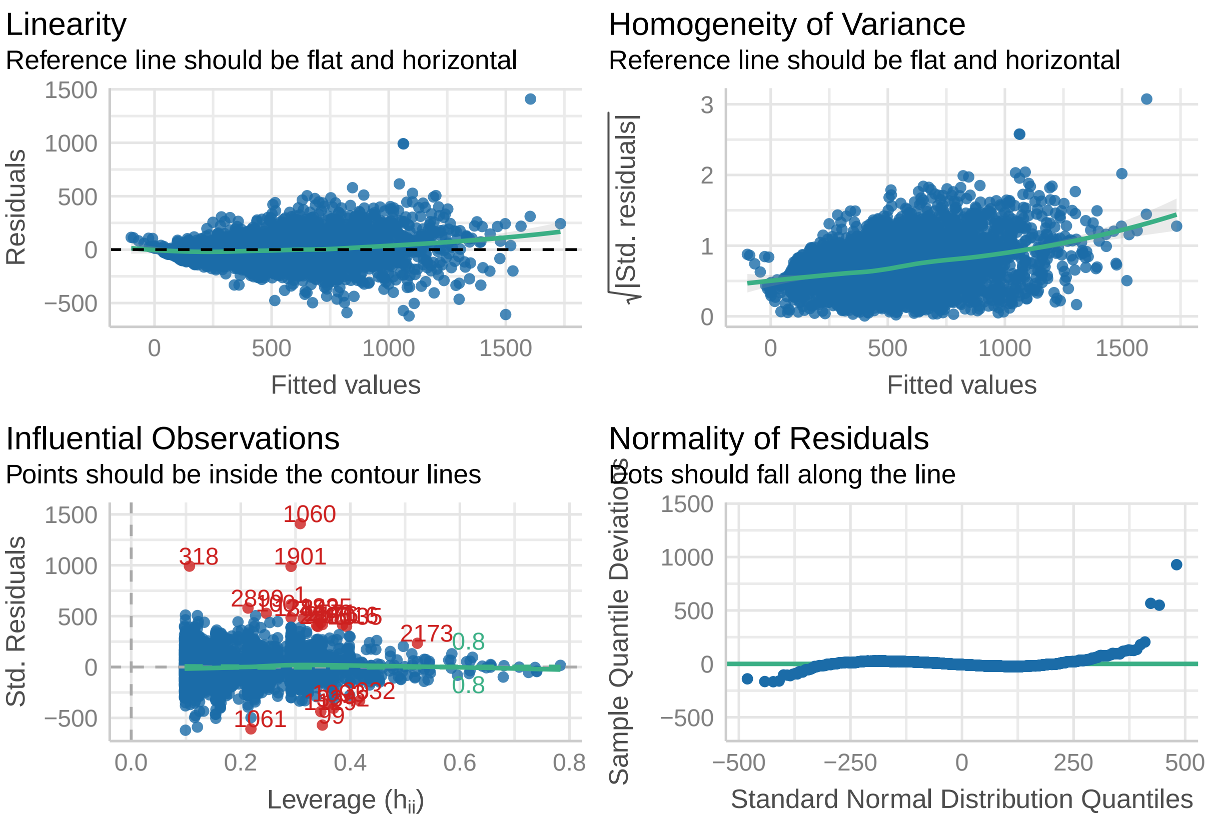
```
# Overall model significance test & test the random effect
lm_null = lm(LEU3N ~ 1, data=data)
anova(lme2, lme1, lm_null)  # The two mixed effects models are highly significant
```

```
## Data: data
## Models:
## lm_null: LEU3N ~ 1
## lme1: LEU3N ~ category * year + (1 | CASEID)
## lme2: LEU3N ~ category * year + (1 + year | CASEID)
##         npar   AIC   BIC logLik deviance   Chisq Df Pr(>Chisq)
## lm_null    2 51093 51106 -25545    51089
## lme1       8 48606 48656 -24295    48590 2498.83  6  < 2.2e-16 ***
## lme2      10 47990 48052 -23985    47970  620.36  2  < 2.2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

### 4.4 Model Diagnostics

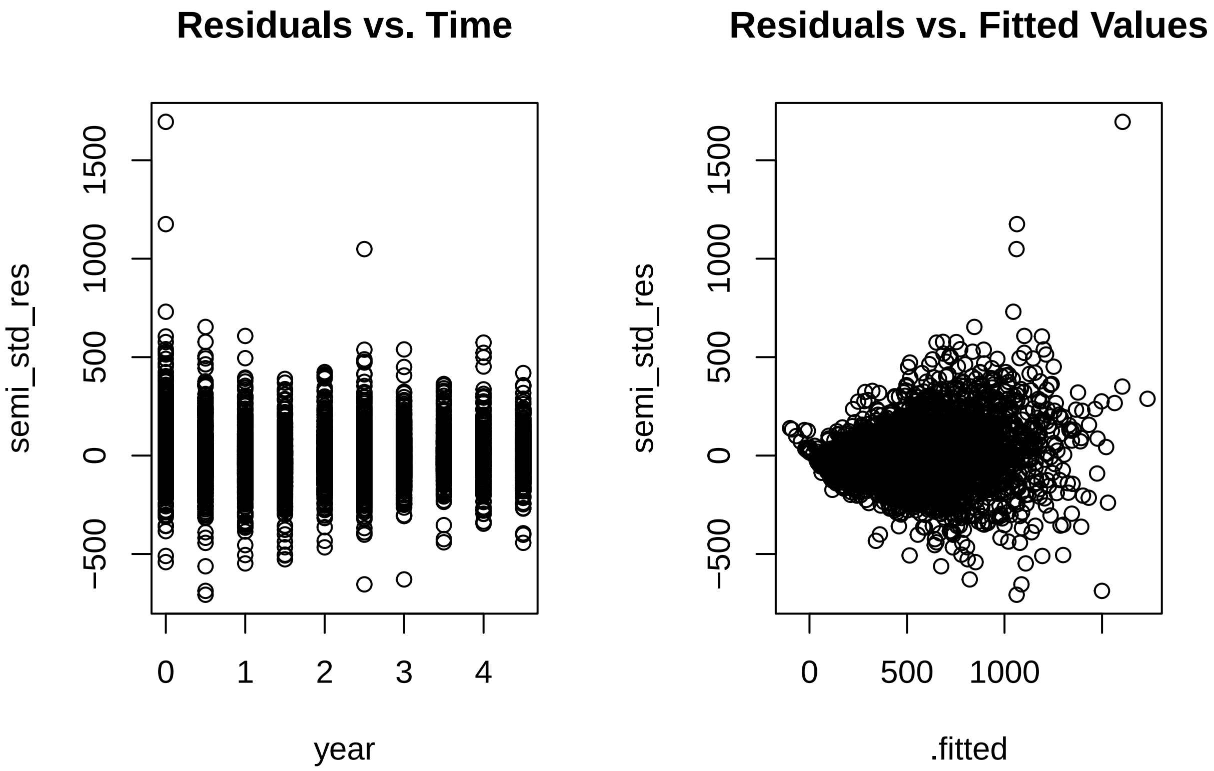
Checking model assumptions to ensure valid inference.

```
# Check model assumptions
library(performance)  # To visually check assumptions
check_model(lme2, check=c('linearity', 'homogeneity', 'qq', 'outliers'))
```



### Homogeneity of Variance

```
# Homogeneity of variance
library(broom.mixed)
data = augment(lme2)
data$semi_std_res = data$.resid / sqrt(1 - data$.hat)
par(mfrow=c(1,2))
plot(semi_std_res ~ year, data=data, main="Residuals vs. Time")
plot(semi_std_res ~ .fitted, data=data, main="Residuals vs. Fitted Values")
```



### Influential Points

```
# Influential points
influential = as.data.frame(data[order(data$.cooksd, decreasing=TRUE), ][1:10, c(2:10)])
influential
```

```
##    LEU3N category year CASEID   .fitted    .resid      .hat  .cooksd   .fixed
## 1   3015     high  0.0   3210 1605.5953 1409.4047 0.3086153 9.609018 609.5324
## 2   2053     high  0.0   5600 1063.6584  989.3416 0.2918011 4.266764 609.5324
## 3    492      low  0.5   1293 1062.2661 -570.2661 0.3490226 2.006810 741.8751
## 4   1660     high  0.0   1007 1045.2060  614.7940 0.2918011 1.647652 609.5324
## 5   1684      low  0.0   5218 1191.5573  492.4427 0.3379588 1.400989 770.4525
## 6   1250      low  0.0   4690  836.5208  413.4792 0.3857444 1.309599 770.4525
## 7   1613     high  0.0   3143 1213.1184  399.8816 0.3931474 1.279032 609.5324
## 8    414     high  0.0   3453  851.6165 -437.6165 0.3461652 1.161889 609.5324
## 9   1210     high  0.0   4603  773.0764  436.9236 0.3461652 1.158212 609.5324
## 10  1163      low  0.5   7163  684.6683  478.3317 0.3140543 1.144230 741.8751
```

The rows 1, 2, 4, 5, and 7 have more than 1500 CD4 cells which are beyond the normal range (500 - 1500). Rows 3 and 8 have less than 500 CD4 cells.

## 5. Conclusion

This analysis explored CD4 cell count trajectories in HIV patients after seroconversion, categorized by their initial viral load. We observed that CD4 counts tend to decline over time; however the rate of decline are not significantly associated with the initial viral load level. Mixed effects models were used to account for individual variability while examining the overall trend.

We found that a) patients across all viral load categories show declining CD4 counts over time. b) Individual trajectories show substantial variability. c) The random slope model better accounts for individual differences in CD4 decline rates. d) Some outliers with unusually high or low CD4 counts were identified

Future work could explore additional predictors of CD4 decline.