2020 Applied Exam Q1

November

(Note: For clarity of the report, the output results are partially presented. R packages and figures are shown in the Appendix. Throughout this report, I will describe the methods and analyses, and summarize the findings in the beginning of each section or each step, and include relevant R code and results afterwards. Some hidden R results are summarized in the comment for the related code.)

(a)

Step 1: To start with, check the class of the columns and we don't need to do factorization. Create a new variable `instrument` as a factor, with 0 representing the old instrument and 1 representing the new one.

I then check the outliers in two ways. (1) check whether the data contains some collection error or resulting from instrument failure. I find that Patient 7 who enrolled in 2015 has 3 measurements for sample 1. According how the data is collected, prior to 2017, only two aliquots will be extracted. Therefore, the corresponding 24th observation is likely to be a collection error. Also, I plot out how `CD34pct` changes with `Stored` by Patient and Sample(Figure 1) and find that Patient 19, 20, 22, 29, 35 have `CD34pct` increase over time, which should be considered as instrument failure according to the description of the problem. A closer look at the data implies the 90th, 50th, 87th, 92th, and 84th observation should be outliers from instrument failure. For this type of outliers, since they don't match with the data rules, I decide to remove them. (2) Still in Figure 1, I find an anomaly in Patient 23, corresponding to the 30th observation. For this type of outlier, I try models with and without them and check the change in inference, and use `halfnormal` plot to help validate. As shown in the later analysis, it should be removed.

As for how to combine the measures prior to 2017 and starting from 2017, I have done three trials, with `d`, `d3`, and `d1`. Using 3 as threshold in `d3` is because the new instrument would have at most 3-year storage till now. Using 1 in `d1` as the threshold is because in the available dataset, the maximum of `Stored` starting from 2017 is 1, not 3. The details will be discussed in the following steps.

```
load('CD34.rda')
d <- data.frame(CD34)
# check types of the dataframe
sapply(d, class)
# create a new variable instrument as a factor
d$instrument <- as.factor(ifelse(d$Coll.year > 2016, 1, 0))
d <- d[-c(24, 50, 84, 87, 90, 92), ] # remove outliers identified by method (1)
rownames(d) <- NULL  # reset index
d3 <- d %>% filter(Stored <= 3)  # d3
d1 <- d %>% filter(Stored <= 1)  # d1
```

Step 2: Next, try fitting a large longitudinal mixed model on `d` and do model selection. I start with a model `m`, including all covariates in `d` for variance reduction purpose. Also, this is not a randomized experiment, so it is necessary to fit an adjusted model. The within-object correlation is specified as Compound Symmetry and I consider both random and fixed slope for `Stored`. Note that there is a nested group structure: `Sample` is nested in `Patient`, so I include it in the model as well. I have tried including `I(Stored^2)` in the model, but it will cause convergence error in estimation, also, including this term will finally lead to an upward fitting curve, and this is not proper given the decaying characteristic of the CD34+cells. So I don't include it the model shown as below.

But sadly, singularity and convergence errors keep showing up. After a few trials, I find that only after I remove `Coll.year`, would I fix the singularity errors. And as for `Viability` and `CD34pct`, they are sensitive to datasets: if I include them in the model, then removing some data points will make the model suffer convergence and singularity errors, while removing some other points will not. Also, checking the anova tables that could be estimated, I find that the effects of `Viability` and `CD34pct` are so apparent. Recalling that `viableCD34pct` is actually the product of these two, this phenomenon could then be understood. Considering all these, I decide to remove these three covariates from the model.

```
m <- lme(viableCD34pct ~ instrument * Stored + Sample + factor(Coll.year) + Viability + CD34pct,
         data = d, random = ~ Stored | Patient/Sample, correlation = corCompSymm())
```

Step 3: Then, I refit the model as shown in `m0`, and perform model comparison to check the structure of random effect and fixed effect. `anova(m0, m1)` shows that we can just consider random intercept and identity within-object correlation structure. `anova(m2, m3)` shows that we can drop `Sample`. The halfnorm plot of `m4` residuals(Figure 5(a)) suggests the 29th(the 30th in the original `d`) as an outlier. After removing it, the the halfnormal plot(Figure 5(b)) looks fine.

Checking the summary and Anova results of `m5`, I find that the slopes of `Stored` prior to 2017 and after 2017 are so different(with a 0.5695914 difference, and p-value is 1.244e-14). Combining with the pattern showed in Figure 2, I think it may be that the viable CD34+ cells will decay quickly in the first few years and then slows down decaying, and for observations prior to 2017, the storage time could be very long and there are only two data points for each patient, and hence the estimated slope would be small. While for observations starting from 2017, the length of storage is short and the estimated slope would be larger. In this scenario, since `instrument` interacts with `Stored` strongly, we can not distinguish the instrument effect from the effect of long storage time, and the estimation of the trajectory over time will not be trustworthy. Therefore, I think simply using `d` is not a good way to combine the two sources of information. Performing similar procedures on `d3`(omitted in in the report), and checking the `xyplot` as shown in Figure 3, I find that `d3` suffers from similar issues, and hence is not a good choice either.

```
m0 <- lme(viableCD34pct ~ instrument * Stored + Sample, data = d,
          random = ~ Stored | Patient/Sample, correlation = corCompSymm())
m1 <- lme(viableCD34pct ~ instrument * Stored + Sample,
          data = d, random = ~ 1 | Patient)
anova(m0, m1)
```

```
##    Model df      AIC      BIC    logLik   Test   L.Ratio p-value
## m0     1 13 149.2937 180.8943 -61.64685
## m1     2  7 137.5370 154.5527 -61.76849 1 vs 2 0.2432804  0.9997
```

```
m2 <- update(m1, method = "ML")
m3 <- update(m2, fixed = viableCD34pct ~ instrument * Stored)
anova(m2, m3)  # suggests m3 is good enough
```

```
##    Model df      AIC      BIC    logLik   Test   L.Ratio p-value
## m2     1  7 122.1971 139.6175 -54.09854
## m3     2  6 120.8220 135.7538 -54.41099 1 vs 2 0.6249154  0.4292
```

```
m4 <- update(m3, method = "REML") # outlier checking is in the appendix
m5 <- update(m4, data = d[-c(29),])  # outlier checking is in the appendix
fixef(m5) # check the estimated fixed effects
```

```
##        (Intercept)        instrument1             Stored instrument1:Stored
##          0.8838532         -0.2413573         -0.1035980         -0.5695914
```

```
Anova(m5, type = 2) # check the significance of the terms
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: viableCD34pct
##                      Chisq Df Pr(>Chisq)
## instrument          27.094  1  1.938e-07 ***
## Stored             117.219  1  < 2.2e-16 ***
## instrument:Stored   59.467  1  1.244e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Step 4: As for `d1`, it doesn't have the same issues as `d` and `d3`. For Patients enrolled in prior to 2017, they only have 1 observation per sample in `d1` and hence would not have a slope of `Stored`. In this case, we then don't need to consider

the `instrument:Stored` effect. Specifically, we are using the points starting from 2017 to estimate the trajectory, and use points before 2017 for comparing the instruments and obtaining a better estimate of the variance terms. Also, there are still many(72) observations in `d1`, so using it will not lose too much information. Therefore, `d1` is my preferred choice for combining the old and new measurement data. Shown below is my modeling on `d1`.

As mentioned before, the trajectory over time is decaying, so fitting a quadratic term of `Stored` is not proper, even if including it will give significant result. I have also tried including `exp(-Stored)` to model the decaying pattern, but the estimated curve doesn't match with intuition, so I don't use it in the end. In `d1`, as mentioned above, we can omit the `instrument:Stored` effect. I have also tried including this interaction term, but the model would then get singularity errors. Therefore, `instrument:Stored` is not included in the model. The previous analysis on `d` has justified the random effect structure, so I would directly use random intercept and identity within-object correlation structure here.

Starting with `ma.0`, `anova(ma.0, ma.1)` shows that we can drop `Sample`. As shown in Figure 6, the halfnormal plot for `ma.2` shows no apparent outliers. The residual plot suggests some variance heterogeneity, so I fit a linear model and try using boxcox to find a plausible transformation. Box-Cox result suggests a 0.3 power transformation. With such a transformation, the homogeneity of variance gets better, but the normality pattern of the residuals gets worse(Figure 7). Also, interpretation will get harder with this power transformation. I have also tried log transformation, and not as good either. So I would just stick with the original form and use `ma.2` as the final model. But we should keep in mind that the homogeneous variance assumption is not satisfied in this analysis, and this is a compromise I make.

```
d1 <- d[-c(29), ] %>% filter(Stored <= 1)  # update d1
ma.0 <- lme(viableCD34pct ~ instrument + Stored + Sample,
            data = d1, random = ~ 1 | Patient, method = "ML")
ma.1 <- update(ma.0, fixed = viableCD34pct ~ instrument + Stored)
anova(ma.0, ma.1)
```

```
##      Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## ma.0     1  6 22.76459 36.42459 -5.382294
## ma.1     2  5 22.87721 34.26054 -6.438602 1 vs 2 2.112616  0.1461
```

```
ma.2 <- update(ma.1, method = "REML") # use REML for inference, and diagnostics is in the appendix
Anova(ma.2, type = 2)
```

```
...
##              Chisq Df Pr(>Chisq)
## instrument  8.2598  1   0.004053 **
## Stored     80.6546  1  < 2.2e-16 ***
...
```

```
# try boxcox
lmod <- lm(viableCD34pct ~ instrument + Stored, data = d1)
bc <- boxcox(lmod, plotit = FALSE)
i <- which.max(bc$y)
bc$x[i] # suggests 0.3 power transformation
```

```
## [1] 0.3
```

```
ma.3 <- update(ma.2, fixed = (viableCD34pct)^0.3 ~instrument+ Stored) # diagnostics is in the appendix
```

```
# chose ma.2 as the final model, and check the estimated fixed effects
summary(ma.2)
```

```
...
## Fixed effects: viableCD34pct ~ instrument + Stored
##                   Value  Std.Error DF   t-value p-value
## (Intercept)  0.9197850 0.07566984 35 12.155239  0.0000
## instrument1 -0.2772822 0.09647989 35 -2.873990  0.0068
## Stored      -0.6717928 0.07480329 34 -8.980792  0.0000
```

...

Step 5: Use `fit.contrast` to estimate the effect of the instrument change. As shown below, the new instrument would give lower measurement of the percentage of cell that are viable CD34+ cells than the old instrument, by about 0.27728. And the 95% CI is [-0.4731468, -0.08141761]. With `Stored` held fixed, the estimateed conversion equation is then

$$viableCD34pct_{new} = viableCD34pct_{old} - 0.2772822.$$

```
fit.contrast(ma.2, "instrument", coeff = c(-1, 1), conf.int = 0.95)
```

```
##                          Estimate Std. Error  t-value    Pr(>|t|)   lower CI    upper CI
## instrument c=( -1 1 ) -0.2772822 0.09647989 -2.87399 0.006848187 -0.4731468 -0.08141761
## attr(,"class")
## [1] "fit_contrast"
```

Throughout the analysis, the assumptions are: (1) the random intercepts and the error terms are independent; (2) the random intercept terms are identically independently normally distributed; (3) the error terms are identically independently normally distributed. It would be hard to check assumption (1). And as for (2) and (3), the normality is acceptable, and the variance homogeneity is doubtful.

(b)

Since I use `d1` for analysis and there would be no decaying pattern for years prior to 2017(only one observation per patient per sample), I would describe the trajectory of decay for points starting from 2017, using `instrument=1`.

The estimated trajectory is as follows. And it tells us that with 1 year increase in storage, the percentage of cells that are viable CD34+ cells would decrease by 0.6717928. And as mentioned in part (a), we need to note that the decaying speed will decrease as time goes by and that the change in later years would be smaller than the first few years. Given the limited amount of data, I am not able to model such a pattern, but we can further work on this as more data is obtained.

$$viableCD34pct = 0.9197850 - 0.2772822 - 0.6717928 * Stored = 0.6425028 - 0.6717928 * Stored.$$

(c)

Solving the inequality $viableCD34pct = 0.6425028 - 0.6717928 * Stored > 0.05$, I get $Stored < 0.8819725$. So I predict that the product will retain effective for 0.8819725 years.

I consider two methods to quantify the uncertainty. The first method is to treat `effective` as binomially distributed, then we can fit a GLMM model and use the estimated probability at `Stored=1` to quantify the uncertainty. For simplicity, I just use the selected variables in (a). The second method is to get the prediction interval of `viableCD34pct` at `Stored=1`. And since there is no available function for getting the interval, I use `lmer{lme4}` and bootstrap(as suggested in Faraway (2016)) to get the prediction interval and estimate the probability of effective.

As shown below, `m1.gee` predicts the probability of remaining effective as 0.5837, while `m.lmer` and bootstrap predicts that after 1 year, the 95% CI of the percentage would be [-0.6217 0.5491], and the probability of remaining effective is 0.402. I would trust the one given by bootstrap more.

```
d1 <- within(d1, {
  effective <- ifelse(viableCD34pct > 0.05, 1, 0)
})
# newdata for prediction
newdata <- data.frame(instrument="1", Stored = 1, Patient = "40")
# geeglm model
m1.gee <- geeglm(formula=effective ~ instrument + Stored, id = Patient,
```

```
                 data = d1, corstr = "exchangebable", family = binomial)
predict(m1.gee, newdata)
```

```
##         1
## 0.5837094
```

```
# lmer model
m.lmer <- lmer(viableCD34pct ~ instrument + Stored + (1|Patient), data = d1)
# bootstrap for prediction
group.sd <- as.data.frame(VarCorr(m.lmer))$sdcor[1]
resid.sd <- as.data.frame(VarCorr(m.lmer))$sdcor[2]
pv <- numeric(1000)
for(i in 1:1000){
 set.seed(i)
 y <- unlist(simulate(m.lmer))
 bmod <- refit(m.lmer, y)
 pv[i] <- predict(bmod, re.form=~0, newdata = newdata)[1] +
   rnorm(n=1,sd=group.sd) + rnorm(n=1,sd=resid.sd)
}
# 95% prediction interval
quantile(pv, c(0.025, 0.975))
```

```
##       2.5%      97.5%
## -0.6216701  0.5491255
```

```
# estimate effective probability
mean(pv > 0.05)
```

```
## [1] 0.402
```

References

Faraway, Julian J. 2016. *Extending the Linear Model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models*. CRC press.

Appendix

Packages

All R packages used in this project are listed below.

```
library(gmodels)
library(MASS)
library(car)
library(dplyr)
library(nlme)
library(lattice)
library(faraway)
library(geepack)
library(lme4)
```

Figures

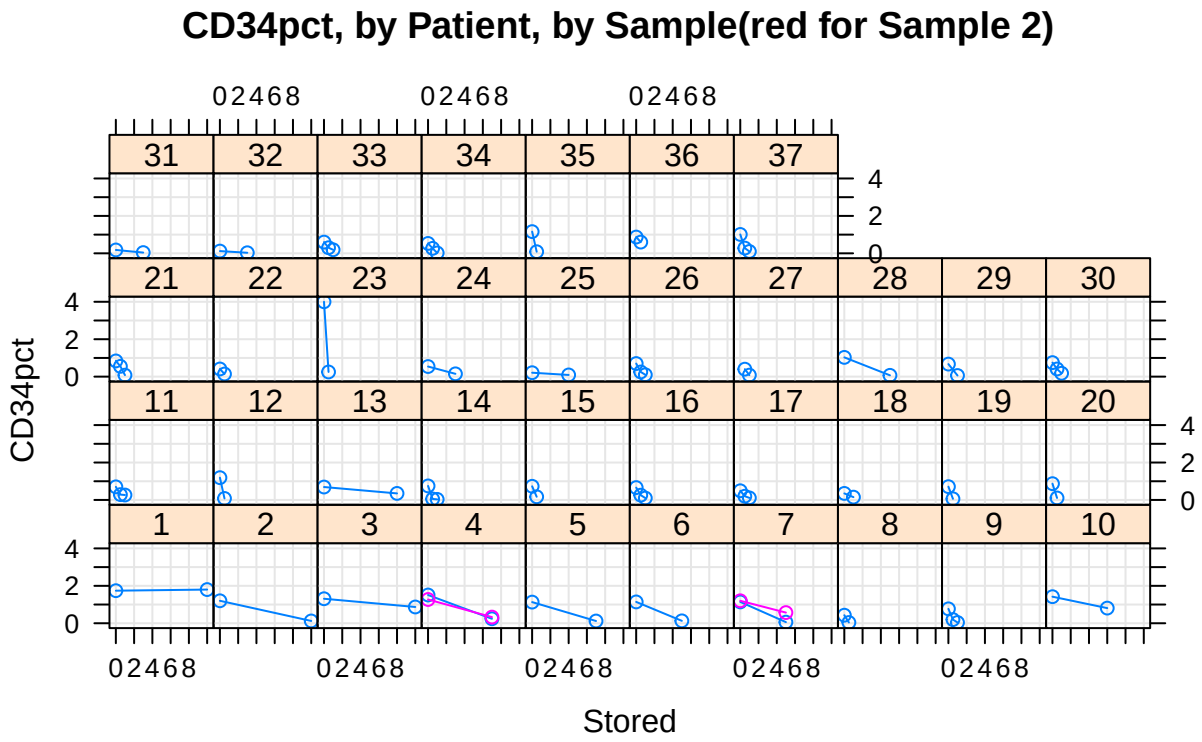


Figure 1: CD34pct vs. Stored in original d(used to identify outliers)

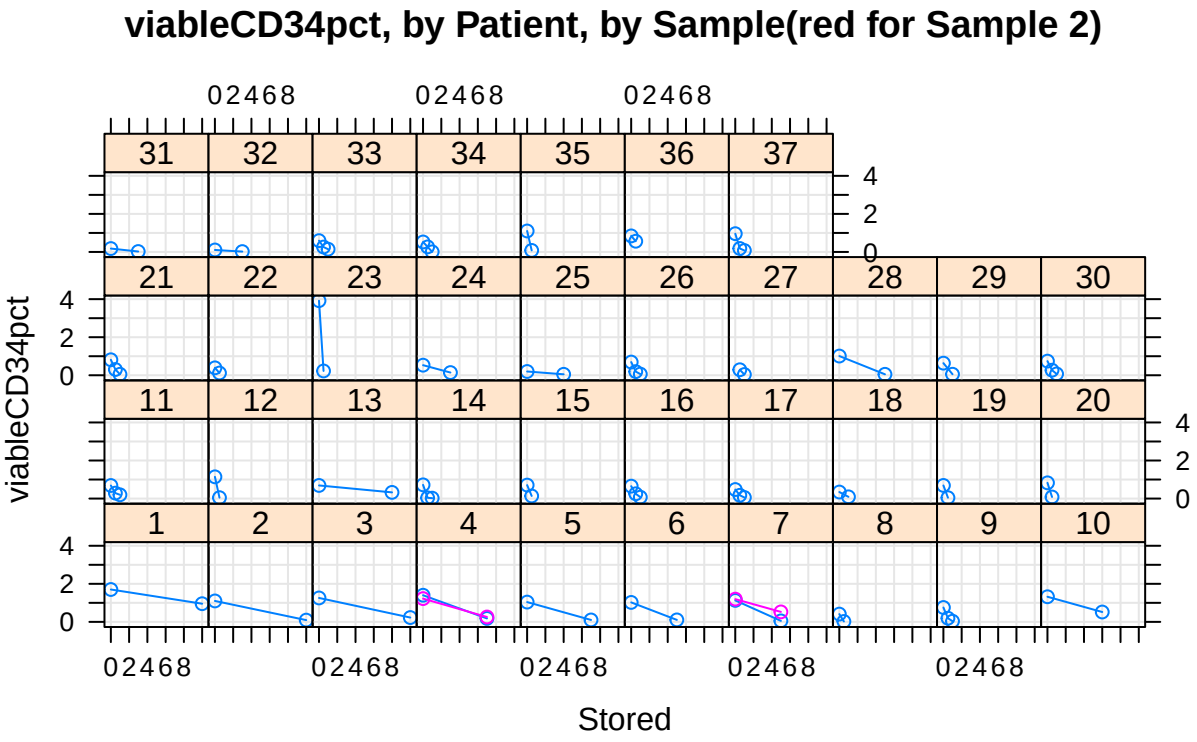


Figure 2: viableCD34pct vs. Stored in d(after outlier removing)

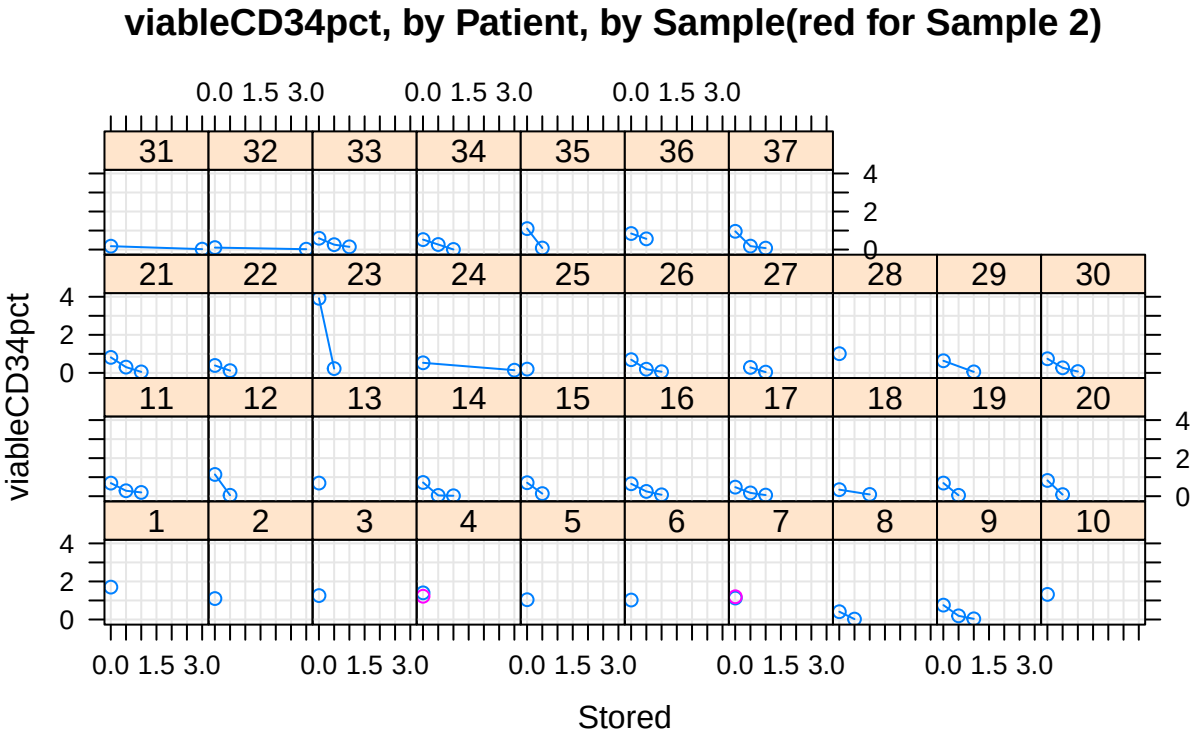


Figure 3: viableCD34pct vs. Stored in d3(Pay attention to Patient 24, 31, and 32)

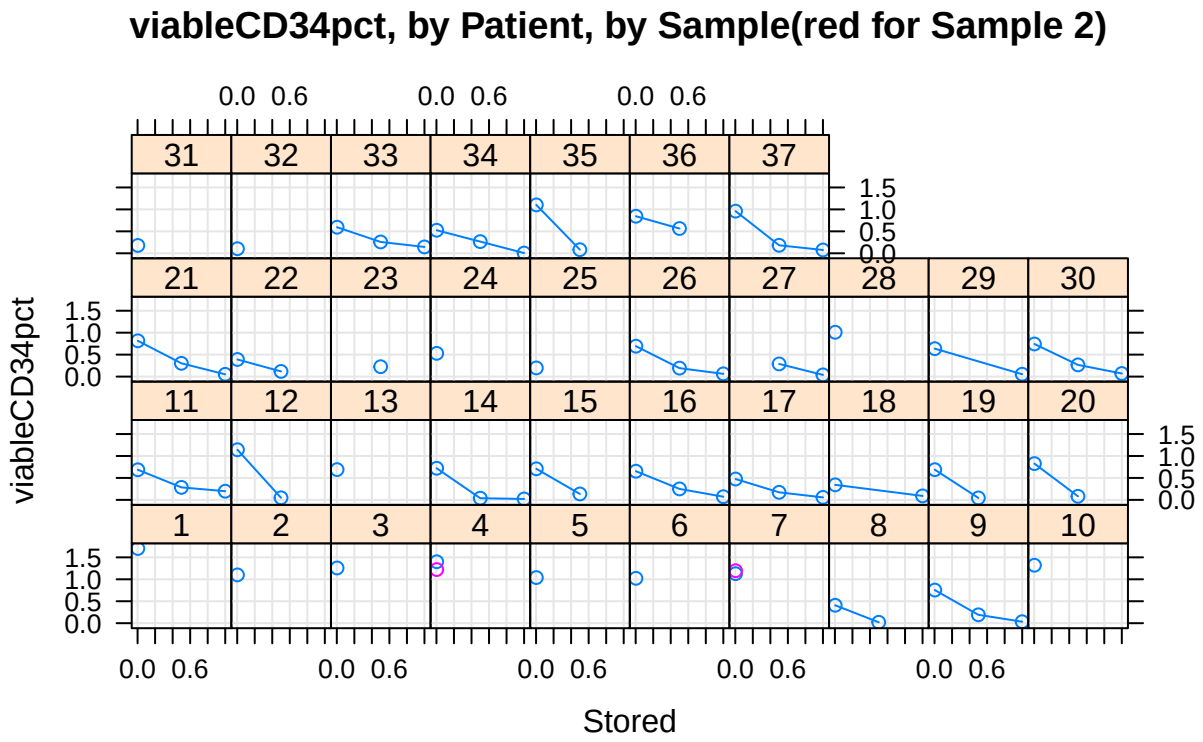
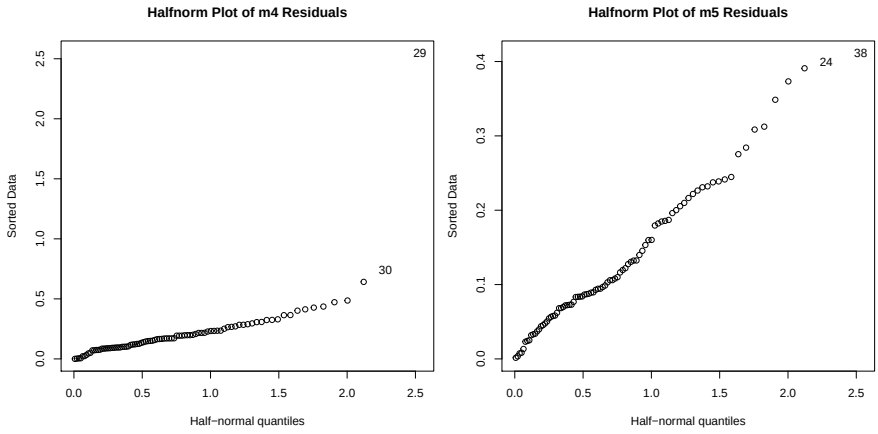
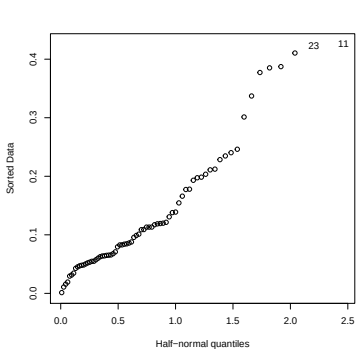


Figure 4: viableCD34pct vs. Stored in d1

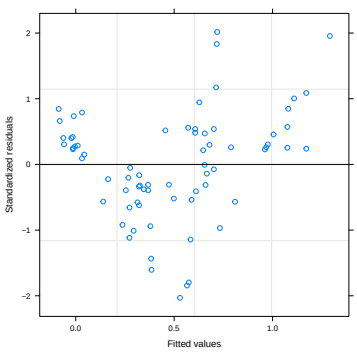


(a) Halfnorm Plot of m4 Residuals (b) Halfnorm Plot of m5 Residuals

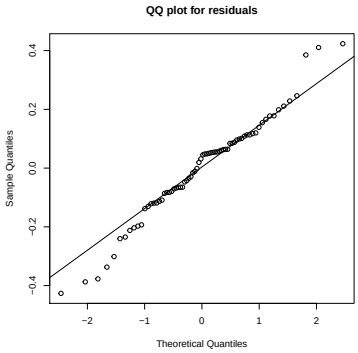
Figure 5: Residual Diagnostics for m4 and m5



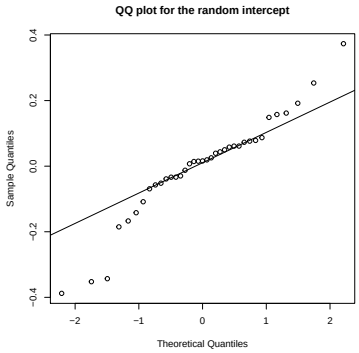
(a) Halfnormal plot of ma.2



(b) Residual Plot of ma.2

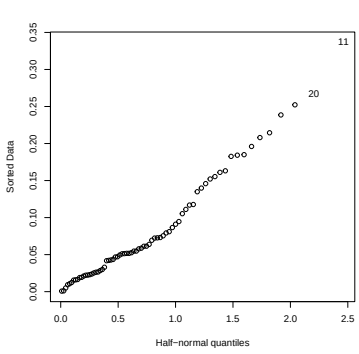


(c) Q-Q plot of residuals of ma.2

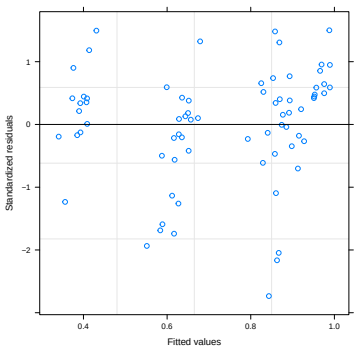


(d) Q-Q plot of the random intercept of ma.2

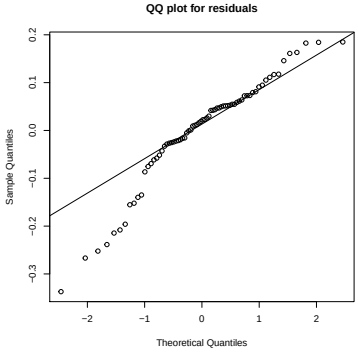
Figure 6: Model Diagnostics for ma.2



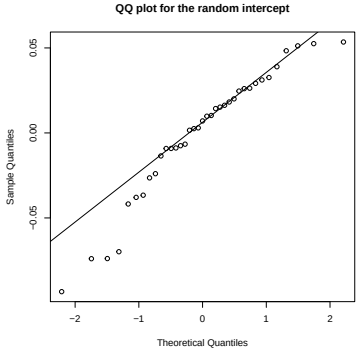
(a) Halfnormal plot of ma.3



(b) Residual Plot of ma.3



(c) Q-Q plot of residuals of ma.3



(d) Q-Q plot of the random intercept of ma.3

Figure 7: Model Diagnostics for ma.3