Name: Basil Ibrahim
ID: 20317669

Course: ACTSC 936
Semester: Winter 2009

Outline

Name: Basil Ibrahim
ID: 20317669
Course: ACTSC 936
Semester: Winter 2009

Longitudinal Modeling of dynCorr Data

1.Introduction

This project report analyzes a longitudinal dataset found in the *dynCorr* package of R. The dataset consists of three different medical measurements –related to kidney problems- of thirty four human subjects, at ten day intervals, for a maximum period of 230 days. Although the dataset is originally used for dynamical correlation analysis, this paper focuses on finding longitudinal models that could well represent the data.

The report is organized as follows:

As stated in the outline, before modeling the dataset and fitting the residuals, etcetera, the paper dedicates two sections for viewing the dataset in raw form, understanding its nature, posing minimal (but necessary) preliminary assumptions and analyses before posing any necessary modeling assumptions. This is a necessary procedure, in order to increase the effectiveness – or even make it possible - of model selection.

After obtaining a "good feel" of the data at hand, one proceeds to the modeling of the dataset. We begin by using linear mixed effects models (LME models for short), choosing the best candidate model for this dataset. We also check whether the LME assumptions are preserved, and attempt to correct for any violations. We then proceed to more complex models, namely GLMM models (which are subject specific models) and GEE models (which are marginal models), under different correlation structures for the GEE case.

Finally, the paper concludes with a general analysis of all the models used in this paper. We also indicate the main challenges in analyzing this dataset and state the main issues that need to be addressed for future research on similar (or the same) datasets. We also pose some possible recommendations and innovations that could be conducted, given the necessary time and academic resources.

2. Preliminary Issues

The dataset consists of five elements: the subject number (1,2,…,34), *time* (0,10,20,…,240), *resp1, resp2* and *resp3* which take on positive real values. In this report, we consider resp2 to be the response variable, with *time, resp1* and *resp3* as candidate covariates (we will consider models without time and models with *time* as a fixed covariate). Hence, the exploratory analysis conducted here will be focused on *resp2*, which is a measurement of "badness" of health, across time. Before proceeding to the analysis, the given dataset poses two issues that need to be addressed.

The first issue is the presence of dropouts in the study: most of the subjects do not get recorded beyond Day 200. The reasons for these dropouts are unknown to us. In the literature, there are several methods for dealing with such missing data. In our exploratory analysis and model fitting we will assume a missing completely at random type (MCAR), which will allow us to simply use the data we have without changing exploratory and modeling methodologies used on complete data. The main downfall of this approach is obtaining inaccurate results, due to not taking into account possible causes of dropouts. For instance, if patients dropped out due to death or vast improvement, it would be necessary to incorporate these pieces of information into our modeling and analyses. Thus the MCAR assumption may be a satisfactory starting point for data analysis, since we do not have any information explaining the subject dropouts; however, there are limitations using this method.

A possible remedy for that is, although we are assuming an MCAR case, we can still attempt to orient our modeling and analysis, at least partially, to see if there are differences between subjects who stayed longer in the study than those who dropped out earlier. And so, as an attempt to analyze possibly hidden factors between subjects that dropped out early and subjects that have continued, the dataset was partitioned once into two groups and once into three groups. For the two group partitioning case, subjects dropping out within 180 days were put into the first group, and the rest were put in the second group; this was done by calculating the median maximum time across all subjects (i.e. taking the last data point of each subject and calculating the median), which was 185 days. As for the three group partitioning case, the first group consisted of subjects who dropped out within 170 days, the second group was assigned for those who dropped out between and including 180 and 200 days, and the third group was assigned for those above 200 days. However, since we do not know the reasons of dropout, we believe it a reasonable starting point to assume an MCAR case. Thus, we handle this issue of data "missingness" by partitioning our subjects on the basis of dropout times and exploring any possible commonalities and differences within and between each group respectively.

The second issue is having nonsystematic or stochastically time varying covariates, namely *resp1* and *resp3*. As stated in Fitzmaurice et al. "when a covariate is both time-varying and stochastic, new issues arise concerning the interpretation and estimation of regression parameters in models for longitudinal data" (415). Not addressing this issue can lead to inaccurate inferences and causal interpretation of the models' coefficients (416).

A remedy for this is to assume the stochastically time-varying covariates as exogenous. Mathematically, one would assume that:

$f(X_{ij+1} \mid X_{i1}, X_{i2}, ..., X_{ij}, Y_{i1}, Y_{i2}, ..., Y_{ij}) = f(X_{ij+1} \mid X_{i1}, X_{i2}, ..., X_{ij})$ holds (418). With these assumptions and remedies, we proceed to the exploratory analysis section.

3. Exploratory Analysis

We begin with some summary statistics, in order to have an idea where the data values of the different variables lie:

	Resp2	**Resp1**	**Resp3**
Range	[0.60, 3.36]	[2.12, 5.89]	[103.90, 388.40]
1st Quartile	0.83	3.86	156.00
Median	0.93	4.08	179.30
Mean	1.11	4.08	181.90
3rd Quartile	1.29	4.32	203.00

Table 1: Summary Statistics of dynCorrData dataset (Rounded to Two Decimal Places)

One sees that *resp3* spans over a large range, relative to its own measurement units; however, since we do not know what the units of measurements are, nor those of the other variables, we cannot indicate whether or not the range in *resp3* is large or small relative to *resp1* and *resp2*. Thus, without further knowledge of the data set, we are constrained to analyzing each variable separately.

Nevertheless, one can still obtain useful information from these summary statistics. For instance, the statistics suggest that *resp2* is positively skewed since the mean greater than the median (this rule of thumb may fail though, since the data is multimodal). We also see that *resp2* takes on positive values in this dataset. This could help us model our response variable under certain nonnegative skewed distributions (such as the gamma distribution for a GLMM model, which will be discussed later). Hence, we attempt to use whatever numerical information we can obtain for choosing candidate models that can well represent the dataset.

Besides numerical statistical summaries, we examine the data graphically, to improve our understanding of the change of individual data across the different time points. We depict four figures, showing all individual trajectories of *resp2*, the response variable, across time under different partitions. In each figure, we add a locally weighted regression curve (LOESS curve), to give an overall mean profile of the trajectories in each graph.

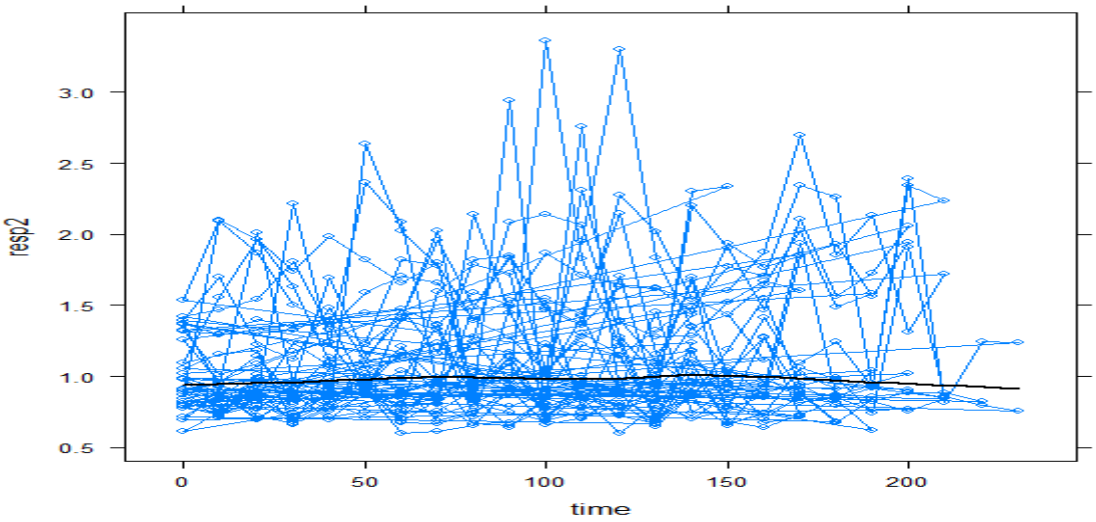

Figure 1: Indiviual Trajectories and LOESS Curve on Entire Dataset

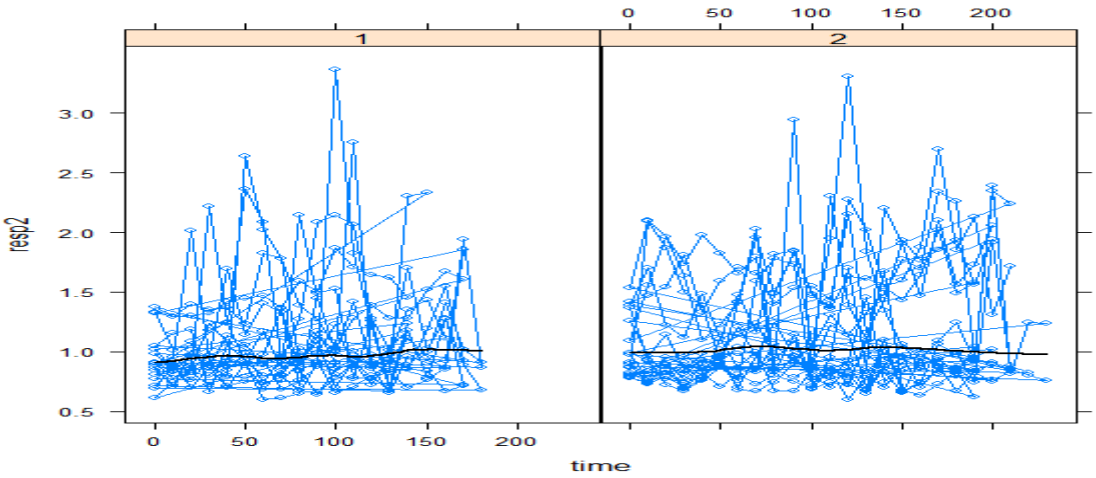

Figure 2: Indiviual Trajectories and LOESS Curve on each of Two Groups

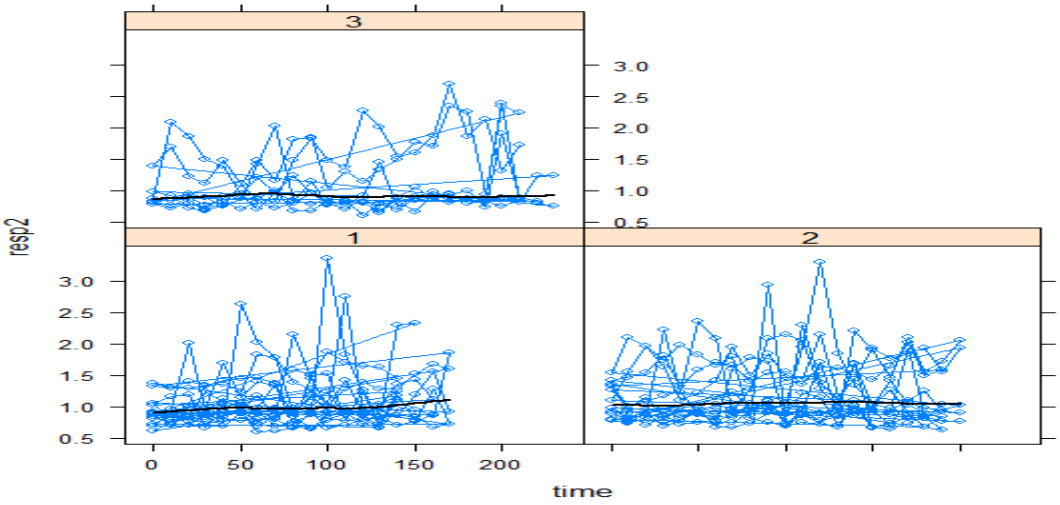

Figure 3: Indiviual Trajectories and LOESS Curve on each of Three Groups

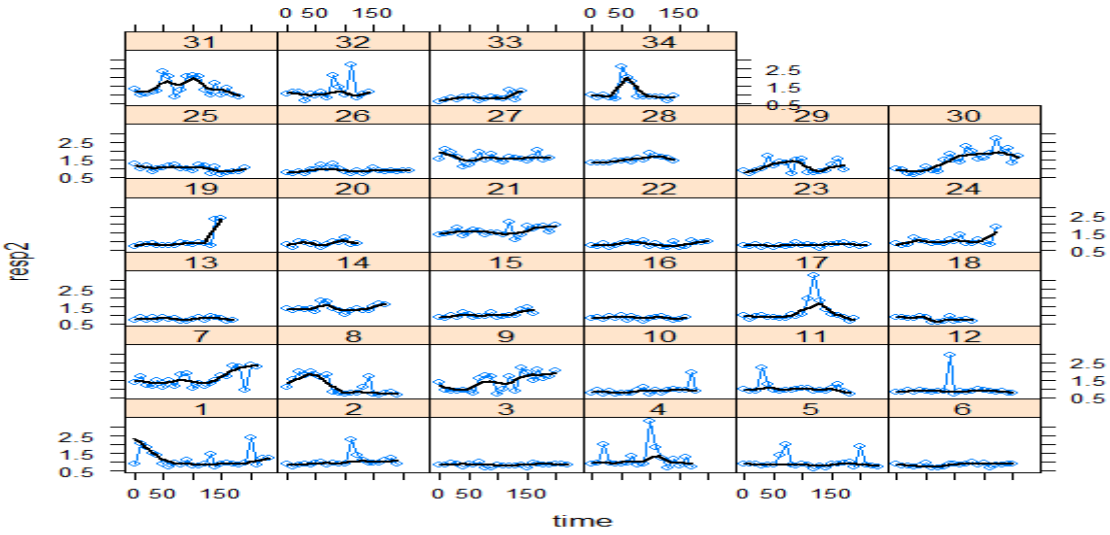

Figure 4: Individual LOESS Curves per Individual

As **Figure 1** to **Figure 4** show, the trajectories are highly irregular but follow similar trends (depicted by the flat LOESS curves), at least until the subject's final examination time period; we do not know what the curves may have looked like had those patients remained. One can see in **Figure 4** that most of the patients who dropped out early (such as 13, 32, 33 and 34), had relatively low values of *resp2*, suggesting that a plausible reason for dropping out is health improvement (recall that *resp2* measures "badness" of health). One may even hypothesize that patients who dropped out early with a high measure of resp2 (subject 19 most notably) could have done so due to severe health problems or even death. However, without further investigation, only speculate, and test these hypotheses on similar future datasets.

After looking at the dataset as a whole, it is necessary to examine the correlations between its different variables, in order to accurately model for them. For instance, in this dataset, since we have three candidate covariates, one should examine the correlations between them, as some models such as LME and regression models assume independence of the covariates (i.e. no multicolinearity). Using R, the covariate correlations are shown to be very low, and hence multicolinearity is not an issue for standard regression and LME models:

	time	**resp1**	**resp3**
time	1.00	0.08	-0.09
resp1	0.08	1.00	0.02
resp3	-0.09	0.02	1.00

Another important issue is measure the correlation of repeated measures which is a "direct consequence of both between-individual heterogeneity and within-individual biological variation in response over time" (Fitzmaurice et al., 42). By looking at these correlations, one would then see if the selected models have successfully accounted for these two types of variability (e.g. one can incorporate certain random effects in the model to account for between-subject variability). Using R, the following lattice shows the correlations between repeated measures for the first 12 time points (chosen on the basis that all subjects have been measured up until Day 120):

	t0	**t10**	**t20**	**t30**	**t40**	**t50**	**t60**	**t70**	**t80**	**t90**	**t100**	**t110**	**t120**
t0	1.0	0.6	0.5	0.5	0.5	0.5	0.6	0.6	0.7	0.5	0.3	0.4	0.4
t10	0.6	1.0	0.8	0.6	0.6	0.2	0.4	0.5	0.4	0.3	0.1	0.1	0.2
t20	0.5	0.8	1.0	0.6	0.6	0.3	0.4	0.6	0.2	0.1	0.5	0.2	0.1
t30	0.5	0.6	0.6	1.0	0.7	0.3	0.3	0.4	0.1	0.1	0.2	0.1	0.2
t40	0.5	0.6	0.6	0.7	1.0	0.4	0.5	0.4	0.2	0.2	0.2	0.1	0.0
t50	0.5	0.2	0.3	0.3	0.4	1.0	0.8	0.5	0.1	0.2	0.3	0.2	0.1
t60	0.6	0.4	0.4	0.3	0.5	0.8	1.0	0.7	0.3	0.2	0.3	0.3	0.1
t70	0.6	0.5	0.6	0.4	0.4	0.5	0.7	1.0	0.3	0.1	0.2	0.0	0.1
t80	0.7	0.4	0.2	0.1	0.2	0.1	0.3	0.3	1.0	0.5	0.2	0.6	0.4
t90	0.5	0.3	0.1	0.1	0.2	0.2	0.2	0.1	0.5	1.0	0.2	0.2	0.2
t100	0.3	0.1	0.5	0.2	0.2	0.3	0.3	0.2	0.2	0.2	1.0	0.5	0.3
t110	0.4	0.1	0.2	0.1	0.1	0.2	0.3	0.0	0.6	0.2	0.5	1.0	0.5
t120	0.4	0.2	0.1	0.2	0.0	0.1	0.1	0.1	0.4	0.2	0.3	0.5	1.0

Although the correlations are positive, which conform to the general norm of longitudinal data, there are other properties that do not. For instance, longitudinal data are known to have decreasing correlations over time, with higher correlation values at closer time points. Here, one sees that this is not necessarily the case; for instance, there is a 0.7 correlation value between **t0** and **t70**, which is greater than the correlation value (=0.6) between **t0** and **t10**. This is repeated throughout the correlation matrix. As will be shown later, this issue will be reflected when modeling correlation structure matrices for GEE models.

Now, we proceed to modeling the dataset.

4. LME Modeling

a) Selecting the LME Model

Using R, we examine several possible combinations of fixed and random effects. On the basis of the AIC value (while also checking other issues such as the significance of the covariates,), we find that the following model yields the lowest (which is desirable):

$$Y_{ij} = \beta_0 + \beta_1 resp1_{ij} + \beta_2 resp3_{ij} + b_0 + b_1 resp1_{ij} + b_2 resp3_{ij} + b_3 time_{ij} + \varepsilon_{ij}$$

The following is the R summary of the model:

Linear mixed-effects model fit by REML
Data: study

AIC	BIC	logLik
416.682	479.2515	-194.341

Random effects:
Formula: ~resp1 + resp3 + time | subject
Structure: General positive-definite, Log-Cholesky parametrization

	StdDev	Corr		
(Intercept)	1.729985846	(Intr)	resp1	resp3
resp1	0.265528111	-0.850		
resp3	0.004595689	-0.798	0.378	
time	0.001657668	-0.384	0.170	0.502
Residual	0.281639256			

Fixed effects: resp2 ~ resp1 + resp3

	Value	Std.Error	DF	t-value	p-value
(Intercept)	3.1174887	0.3504563	612	8.895514	0
resp1	-0.3055783	0.0670798	612	-4.555443	0
resp3	-0.0043672	0.0009626	612	-4.537028	0

Correlation:

	(Intr)	resp1
resp1	-0.839	
resp3	-0.598	0.081

Standardized Within-Group Residuals:

Min	Q1	Med	Q3	Max
-2.83043243	-0.52162118	-0.09192326	0.34140657	5.66783270

Number of Observations: 648
Number of Groups: 34

We also examined this model after incorporating a time covariate in the fixed effect. However, that model both yielded a higher AIC (≈428.86) and using a Wald test, it has been found that the time covariate as a fixed effect was statistically insignificant under a 99% confidence level ($p \approx 0.042$). Furthermore, by using ANOVA tests in R and comparing the results to 50:50 chi-square critical values between our above mentioned model and other candidate models, we found that this model is the most parsimonious one (with the lowest AIC value). Thus, the above mentioned model has been found to be the most suitable candidate for representing the data, using LME modeling.

After choosing the best candidate model, it is essential to make sure that the LME assumptions hold. Otherwise, the model will yield incorrect estimates and/or inaccurate inferences. One should examine issues such as normality of residuals and random effects, mulicolinearity, heteroskedasticity and autocorrelation (known as the departures of regression). Regarding multicolinearity, one sees that the correlation between *resp1* and *resp3* is quite low (also shown in the previous section), and so one can consider that the covariates in this model are independent. Before examining the other two departures, we begin by examining whether the residuals and random effects satisfy normality assumption of LME models.

b) Checking for Normality Assumption in Residuals and Random Effects

By examining the following graphs, we see that the patterns closely conform to normal distributions.

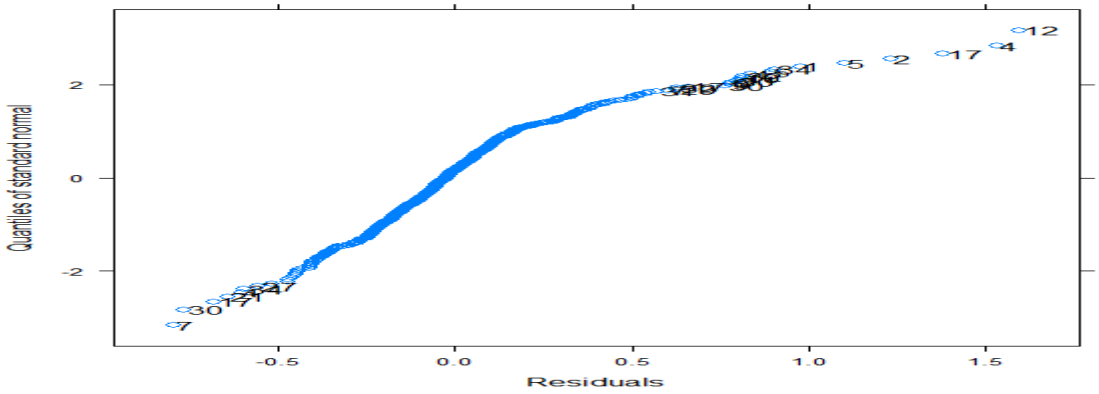

Figure 5: q-q Plot Testing for Normality Assumption of within-group Errors

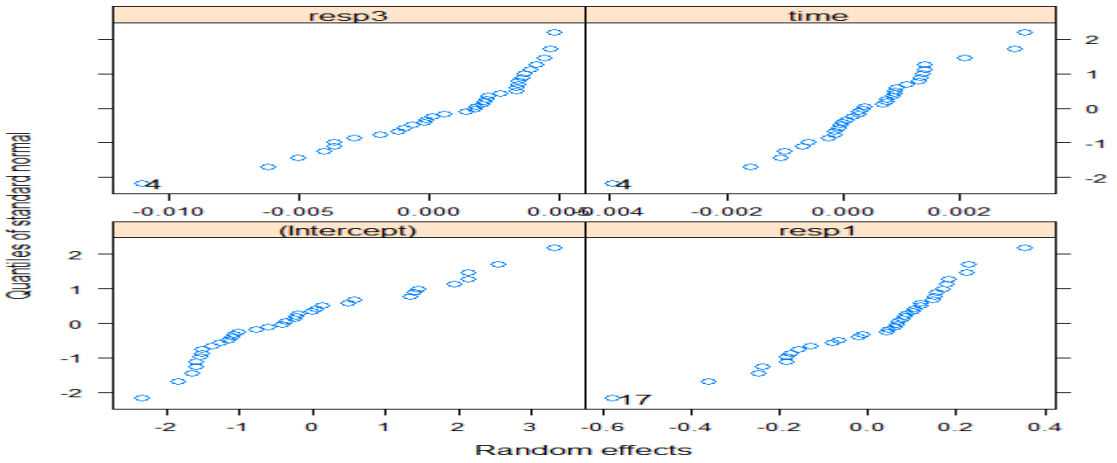

Figure 6: q-q Plot Testing for Normality Assumption of Random Effects

Thus, one can consider that the normality assumption is satisfied in this model. The following two sections examine and attempt to correct for both heteroskedasticity and autocorrelation.

c) Checking for Heteroskedasticity and Attempting to Correct for it if Found

A useful visual aid to examining heteroskedasticity in the model is to plot the fitted values against the standardized residuals:

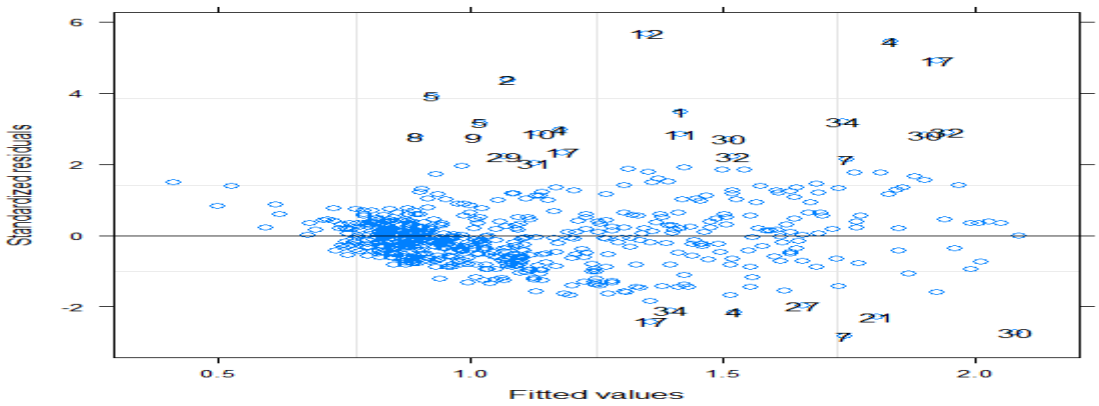

Figure 7: Standardized Residual Plot for Chosen LME Model

For ideal homoskedasticity, the points should form a homogeneous cloud throughout the graph. Although the graph indicates that the variability of the error terms are smaller at the initial fitted value points (with a cluster roughly between 0.75 and 1.2), one may still consider the data to be homoskedastic. For a more precise analysis, one should use tests such as the White-test and the Breusch-Pagan test, to compare to critical values. This however is beyond the scope of the course on which this report is based, and so we simply attempt to find variance class functions in R that could help increase the homoskedasticity of the data.

By using the varIdent function in R –which assumes a constant variance per group-on the two subject groups (there were convergence constraints in R whilst attempting to implement this function on the three subject groups) we have created for the data, we find that the residual plot is nearly identical to the original one:

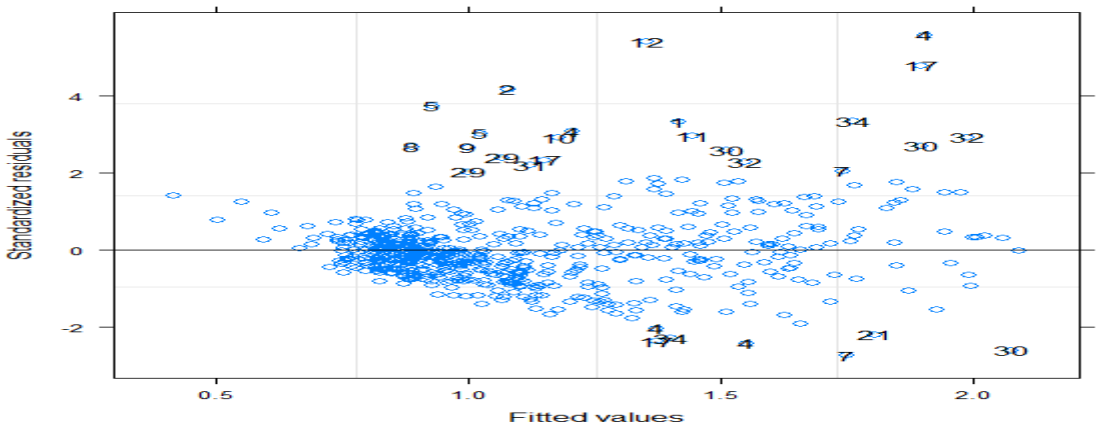

Figure 8: Standardized Residual Plots for each of Three Groups for Chosen LME Model

Although the AIC has slightly improved (decreased to 415.46, after being 416.6) using this lme model (varIdent(form=~1)), we find that the correlation between the covariates have also increased (also slightly). On the basis that we consider covariate independence a higher priority over a very slight improvement in the goodness of fit of the model (reflected by the AIC) we do not consider the new model an improved version of the original one. Hence, based on graphical display, we consider the model both to be "homoskedastic enough" and to have the best fit of the data among its modified versions which have been made as an attempt to increase its homoskedasticity. Now we proceed to check for the final departure: autocorrelation.

d) Checking for Autocorrelation and Attempting to Correct for it if Found

By examining the ACF plot and semi-variogram of the data, one finds that the serial correlation is quite low (the fourth lag lies on the significance line). Note that we use five lags in the ACF plot, since the data is complete until 12 measurement points.

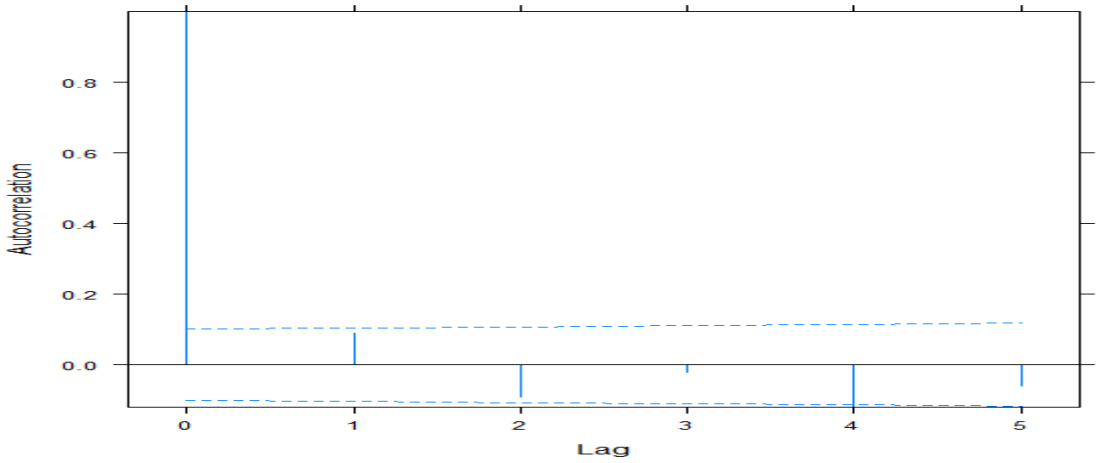

Figure 9: ACF Plot for Exploring Serial Correlation in Chosen LME Model

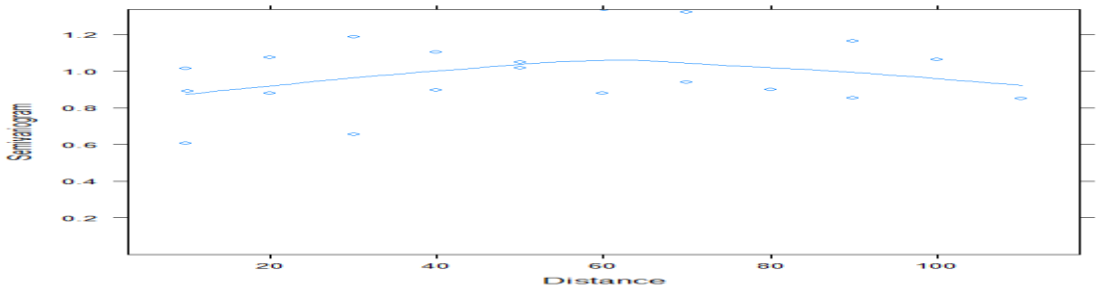

Figure 9: Semivariogram of Chosen Model (form=~time+resp1)

By examining possible correlation structures, such as AR(1) and ARMA(1,1), we find that there is no significant improvement to the original model. Rather, although the correlation between the covariates is unchanged, the AIC values are slightly higher. Thus, as in the previous departure case, we will remain with the original model.

Therefore, regarding the LME fitting, we find that the model we have chosen to be the most suitable among the other LME models we have considered. Graphically, although this model seems to satisfy the LME assumptions, it is suggestible to use numerical tests to examine these assumptions closer (which are beyond the scope of this report). Also, one may consider other programming packages (in R or elsewhere), which use flexible mechanisms that can correct for these departures Although this model seems to Now we turn to more general types of models in the next two sections.

5. GLMM Modeling

Similarly to the LME case, we compared GLMM models with and without time as a fixed covariate, using different possibilities of random effects. One would prefer to model the response variable on a positively skewed, nonnegative distribution (such as gamma, or inverse Gaussian), since the *resp2* most likely conforms to this type of skewness (since the mean is greater than the median). However, due computational limitation in choosing the family type in the current version of R (2.8.1), we are constrained to only choosing the Gaussian distribution to represent *resp2* (with an identity link). Nevertheless, by examining the significance of the covariates and comparing the AIC results of LME models, one obtains a model that is quite appealing. Furthermore, this model has the same fixed and random effects as the optimally chosen LME model shown in the previous section. These are the main numerical statistics obtained using R:

Linear mixed model fit by REML
Formula: resp2 ~ resp1 + resp3 + (time + resp1 + resp3 | subject)
Data: study

AIC	*BIC*	*logLik*	*deviance*	*REMLdev*
416.7	*479.3*	*-194.3*	*368.5*	*388.7*

Random effects:

Groups	*Name*	*Variance*	*Std.Dev.*	*Corr*		
subject	*(Intercept)*	*2.9929e+00*	*1.7299878*			
	time	*2.7479e-06*	*0.0016577*	*-0.384*		
	resp1	*7.0505e-02*	*0.2655287*	*-0.850*	*0.170*	
	resp3	*2.1120e-05*	*0.0045957*	*-0.798*	*0.502*	*0.378*
	Residual	*7.9321e-02*	*0.2816393*			

Number of obs: 648, groups: subject, 34

Fixed effects:

	Estimate	*Std. Error*	*t value*
(Intercept)	*3.1174870*	*0.3504565*	*8.896*
resp1	*-0.3055789*	*0.0670799*	*-4.555*
resp3	*-0.0043672*	*0.0009626*	*-4.537*

Therefore, with the AIC quite close to that of the LME model, and having the covariates statistically significant, this GLMM model is a good candidate for representing the data. Also, by using "approximate" LRT tests, one finds that this model is the most parsimonious one relative to other GLMM models considered here (using different random effects, with and without time as a fixed covariate).

Naturally, there may be subject-specific models that better fit the data than this one; however, due to programming constraints (such as examining a GLMM model having a gamma family) one places the above mentioned one as the optimal model, until proved otherwise. Now we shift from to marginal modeling, specifically GEE models, where the focus is on the overall population, rather than on each specific individual.

6. GEE Modeling

We examine GEE models under three different correlation structures. Similarly to the GLMM, we assume a Gaussian distribution for the response variable (the program does not run if one attempts a gamma or inverse Gaussian distribution). The main criterion of choosing which structure better represents the data over the other is by examining the statistical significance of the covariates, after including the "sandwich term" (i.e. we will examine the robust z values of each covariate).

a) Unstructured Correlation Structure

Examining the GEE estimates with and without time as a fixed covariate, one finds the robust z-values for both cases are statistically insignificant. Hence, in spite of having a flexible structure as such, it does not well represent the dataset. We move on to the next candidate structure.

b) Exchangeable Correlation Structure

Unlike the unstructured type, having compound symmetry yields significant covariate estimates (after including the "sandwich term") under the 95% confidence level. In the case where *resp1* and *resp3* are the only fixed covariates in the model, we have:

	Estimate	*Naive S.E.*	*Naive z*	*Robust S.E.*	*Robust z*
(Intercept)	*3.1685767*	*0.17691806*	*76 17.909854*	*0.3247149545*	*9.758025*
resp1	*-0.2975133*	*0.0431818817*	*-6.889770 0.*	*0715113033*	*-4.160367*
resp3	*-0.0046900*	*0.0005622797*	*-8.341044*	*0.0006794234*	*-6.902912*

Estimated Scale Parameter: 0.1617852
Number of Iterations: 3

As for the case where we also have time as a fixed covariate, we have:

	Estimate	*Naive S.E.*	*Naive z*	*Robust S.E.*	*Robust z*
(Intercept)	*3.1531065292*	*0.1754656071*	*17.969941*	*0.2970098063*	*10.616170*
time	*0.0007733976*	*0.0002267605*	*3.410636*	*0.0003175851*	*2.435245*
resp1	*-0.3334647003*	*0.0440768775*	*-7.565525*	*0.0630082322*	*-5.292399*
resp3	*-0.0041796708*	*0.0005764833*	*-7.250289*	*0.0006981926*	*-5.986415*

Estimated Scale Parameter: 0.1580164
Number of Iterations: 3

In the second model, time is statistically significant at the 95% confidence level, but not at the 99% confidence level. Without the use of model testing comparisons (such as an LRT test for LME models), we speculate that the first model (without the time covariate) is a parsimonious version of the second one. However, for more accurate inferences, one needs the appropriate statistical tests for comparing these two models (which is beyond the academic scope of this report).

c) AR(1) Correlation Structure

Similar to the exchangeable case, the covariate estimates in this case have statistically significant robust z-measures under the 95% confidence level (except for the time covariate, which will be shown below). For the case where *resp1* and *resp3* are the only fixed covariates in the model, we have:

	Estimate	*Naive S.E.*	*Naive z*	*Robust S.E.*	*Robust z*
(Intercept)	*3.092070467*	*0.2000110188*	*15.459501*	*0.3186236367*	*9.704460*
resp1	*-0.269505733*	*0.0453802770*	*-5.938830*	*0.0543008407*	*-4.963196*
resp3	*-0.004908566*	*0.0006168638*	*-7.957293*	*0.0009374749*	*-5.235944*

Estimated Scale Parameter: 0.1631084
Number of Iterations: 4

By adding time as a fixed covariate, we have the following estimates:

	Estimate	*Naive S.E.*	*Naive z*	*Robust S.E.*	*Robust z*
(Intercept)	*3.0274650974*	*0.2027922181*	*14.928902*	*0.3117392646*	*9.711530*
time	*0.0007217857*	*0.0004268494*	*1.690961*	*0.0003911819*	*1.845141*
resp1	*-0.2764704280*	*0.0453154572*	*-6.101018*	*0.0544922049*	*-5.073578*
resp3	*-0.0047621583*	*0.0006151065*	*-7.742006*	*0.0009289768*	*-5.126240*

Estimated Scale Parameter: 0.1607857
Number of Iterations: 4

The time covariate here is statistically insignificant under the 95% confidence level, suggesting that the first model is a parsimonious version of the second model. Still, as mentioned above, one would need further inference analysis to draw more accurate conclusions.

Therefore, comparing the three different structures one can consider GEE models under both the exchangeable and AR(1) structures, with *resp1* and *resp3* as the fixed covariates, as the optimal models. Further testing criteria, other than significance comparisons (which are very close for both GEE models in this case), are required to find which model (out of those two correlation structures) better represents the data.

For instance, by looking at the number of iterations, one finds that the exchangeable structure takes only 3 iterations as opposed to the AR(1) structure which takes 4. However, this is more of an efficiency issue than of having a model better representing the data.

Another factor one could look at that makes the exchangeable structure more appealing is by looking at the correlations between the response variable at different time points, in the **Exploratory Analysis** section. Since the correlations do not decay the wider apart the observations are in time (which is usually the case with longitudinal data). Thus, one finds a fixed ICC a more realistic fit to the data correlations than a decaying one.

Notwithstanding this, there may exist GEE models that better fit the data than those mentioned above. For instance, if there are (or will be) programming packages that can incorporate positively skewed nonnegative distributions (such as gamma and inverse Gaussian)

for the response variable, one may find models that better fit the data (recall that we were restricted only to using Gaussian distributions for the response variable, which may not have been as well representative as other unconsidered distributions). Next to this, one may be able to find more flexible correlation structures that still yield significant estimates for the covariates (the unstructured type here yielded statistically insignificant estimates). Hence, overcoming computational constraints, one may find optimal GEE models for this dataset.

7. Conclusion and Recommendations

Finding optimal models for this dataset was a challenging task, mainly because of the subject dropouts and the time varying covariates. Furthermore, due to having packages such as *glmer* and *gee* in R 2.8.1 that are still "works-in-progress", one is restricted to modeling the response variable according to the available families in these packages (Gaussian in this case). Next to this, comparisons between the different GLMM and GEE models are quite restricted in this version of R, making optimal model selection a challenging issue. However, due to time and resource (academic material) constraints, this report did not seek to address these issues further.

Further possible and more flexible models that can be investigated (given the time and resources), apart from the above mentioned model types (LME, GLMM and GEE), are varying coefficient models (such as additive mixed models) and fully nonparametric models. Also, if one can further examine the data and the reasons behind subject "missingness", one can use models that incorporate MAR (such as weighted GEE models) or even MNAR (such as pattern mixed approaches). However, these are not easy procedures, and a lot of time may be needed to knowing the nature of the studies and the different measurement variables (e.g. unit measurements of *resp1*, *resp2* and *resp3*).

Therefore, this report is aimed to give a general idea of the dataset, providing future researchers with potential models that can well represent the data. Furthermore, we pose the main limitations of our project and possible recommendations for further investigating the models given here, in addition to other more flexible models. In the light of this study, one can hope that the inferences and models established here could be used for estimating and predicting results of future datasets of the same nature. This is especially an important issue for medical data, where health and wellbeing are the focus of the study.

Appendix: R-Code

```
#library(dynCorr)
#data(dynCorrData)
#study=dynCorrData
#study$subject=as.factor(study$subject)
#summary(study)

#Writing into an Excel sheet, for modification
#library(RODBC)
#chan=odbcConnectExcel("Project",readOnly = FALSE)
#sqlSave(chan, study)
#close(chan)

#Importing from an Excel file having an extra group term (group1 for subjects having observations below 185, group2 for above)
library(RODBC)
chan=odbcConnectExcel("Project")
study=sqlFetch(chan,"Data")#Obtain data from Excel file
close(chan)
study$subject=as.factor(study$subject)#Make subject a factor
study$group=as.factor(study$group)#Make group a factor (splitting subjects into two groups)
study$group2=as.factor(study$group2) #Make group 2 a factor (splitting subjects into three groups)
summary(study)
#2) Exploratory Analysis
library(lattice)
xyplot(resp2~time,type='b',data=study, panel=function(x,y){panel.xyplot(x,y,type='b')
panel.loess(x,y,span=0.4,lwd=2,col=1)})

xyplot(resp2~time|group,type='b',data=study, panel=function(x,y){panel.xyplot(x,y,type='b')
panel.loess(x,y,span=0.4,lwd=2,col=1)})
xyplot(resp2~time|group2,type='b',data=study, panel=function(x,y){panel.xyplot(x,y,type='b')
panel.loess(x,y,span=0.4,lwd=2,col=1)})
xyplot(resp2~time|subject,type='b',data=study,panel=function(x,y){panel.xyplot(x,y,type='b')
panel.loess(x,y,span=0.4,lwd=2,col=1)})

summary(study)
attach(study)
study.cor.mtx=data.frame(t0=resp2[time==0],t10=resp2[time==10],t20=resp2[time==20],t30=resp2[time
==30],t40=resp2[time==40],t50=resp2[time==50],t60=resp2[time==60],t70=resp2[time==70],t80=resp2[
```

```
time==80],t90=resp2[time==90],t100=resp2[time==100],t110=resp2[time==110],t120=resp2[time==120
])
round(cor(study.cor.mtx,use="pairwise.complete.obs"),1)
par(lwd=1)
pairs(study.cor.mtx)
mtext('Pairwise Relationships between First 12 Responses at Distinct Time
Points',outer=T,line=0.4,cex=1.0)
detach(study)
covariates=data.frame(study$time,study$resp1,study$resp3)
round(cor(covariates),2)#Examining for multicolinearity
```

#3) LME models

#3a) Finding the best model

```
library(nlme)
study.lme1= lme(fixed=resp2~ resp1+resp3,random=~ resp1+resp3+time|subject,data=study,
control=list(msMaxIter = 100))#The winning model
study.lme2= lme(fixed=resp2~ resp1+resp3,random=~ resp1+resp3|subject,data=study,
control=list(msMaxIter = 100))
study.lme3= lme(fixed=resp2~ resp1+resp3,random=~ resp1 |subject,data=study, control=list(msMaxIter
= 100))
study.lme4= lme(fixed=resp2~ resp1+resp3,random=~ resp3 |subject,data=study, control=list(msMaxIter
= 100))
study.lme5= lme(fixed=resp2~ resp1+resp3,random=~ 1 |subject,data=study, control=list(msMaxIter =
100))
study.lme6= lme(fixed=resp2~time+resp1+resp3,random=~time+resp1+resp3|subject,data=study,
control=list(msMaxIter = 100))
study.lme7= lme(fixed=resp2~time+resp1+resp3,random=~ resp1+resp3|subject,data=study,
control=list(msMaxIter = 100))
study.lme8=lme(fixed=resp2~time+resp1+resp3,random=~time|subject,data=study)
study.lme9= lme(fixed=resp2~time+resp1+resp3,random=~1|subject,data=study)
study.lme10= lme(fixed=resp2~0+resp1+resp3,random=~1| 0+resp1+resp3+time ,data=study)
study.lme11= lme(fixed=resp2~0+resp1+resp3,random=~1| resp1+resp3+time,data=study)
summary(study.lme1)
summary(study.lme2)
summary(study.lme3)
summary(study.lme4)
summary(study.lme5)
summary(study.lme6)
summary(study.lme7)
summary(study.lme8)
summary(study.lme9)
summary(study.lme10)
summary(study.lme11)
anova(study.lme1,study.lme2)
anova(study.lme1,study.lme3)
Wald.test = function(L, lme.object)
  {
   if(is.vector(L)) invisible(L = matrix(L, nrow=1))
   temp.stat = (t(as.matrix(lme.object$coefficients$fixed)) %*% t(L)) %*%
                          solve(L %*% as.matrix(lme.object$varFix) %*% t(L)) %*%
```

```
                    (L %*% as.matrix(lme.object$coefficients$fixed))
          df = min(dim(L)[1], dim(L)[2])
          p.value = 1 - pchisq(temp.stat, df)
          if(p.value >= .0001)
   cat('\n', 'Wald test chi-square statistic is', round(temp.stat,4),
                                'with', df, 'df and p-value =', round(p.value, 4), '.', '\n')
  else
   cat('\n', 'Wald test chi-square statistic is', round(temp.stat,4),
                                'with', df, 'df and p-value < .0001 .', '\n')
 }
Wald.test(L=matrix(c(0,1,0,0),nrow=1,ncol=4,byrow=T),lme.object=study.lme6)
```

 # 3b) Checking for Normality Assumption in Residuals and Random Effects

```
qqnorm(study.lme1,~resid(.),id=.05,adj=-.1)
qqnorm(study.lme1,~ranef(.),id=.05,adj=-.1)
```

 # 3c) Checking for Heteroskedasticity and Attempting to Correct for it if Found

```
plot(study.lme1,id=.05)
plot(study.lme1,resid(.,type='n')~fitted(.)|group,id=.05)
plot(study.lme1,resid(.,type='n')~fitted(.)|group2,id=.05)
#study.lme1b=update(study.lme1,weights=varIdent(form=~1|group2), control=list(msMaxIter = 100))
#plot(study.lme1b,id=.05) Convergence issues when having three groups
summary(study.lme1b)
study.lme1c=update(study.lme1,weights=varIdent(form=~time+resp1+resp3 |group),
control=list(msMaxIter = 100))
study.lme1d=update(study.lme1,weights=varIdent(form=~time+resp1 |group), control=list(msMaxIter =
100))
study.lme1e=update(study.lme1,weights=varIdent(form=~time+resp3 |group), control=list(msMaxIter =
100))
study.lme1f=update(study.lme1,weights=varIdent(form=~time |group), control=list(msMaxIter = 100))
study.lme1g=update(study.lme1,weights=varIdent(form=~1 |group), control=list(msMaxIter = 100))
plot(study.lme1c,id=.05)
plot(study.lme1d,id=.05)
plot(study.lme1e,id=.05)
plot(study.lme1f,id=.05)
plot(study.lme1g,id=.05)
summary(study.lme1c)
summary(study.lme1d)
summary(study.lme1e)
summary(study.lme1f)
summary(study.lme1g)
```

 # 3d) Checking for Autocorrelation and Attempting to Correct for it if Found

```
plot(ACF(study.lme1,maxLag=5,resType= 'n'),alpha=.01)
plot(Variogram(study.lme1,form=~time+resp1,maxDist=120,resType='n')
#plot(Variogram(study.lme1,form=~time,maxDist=120,resType='n'),main="Semivariogram")
#plot(Variogram(study.lme1,form=~1,maxDist=120,resType='n'),main="Semivariogram")
#plot(Variogram(study.lme1,form=~resp1,maxDist=120,resType='n'),main="Semivariogram")
#plot(Variogram(study.lme1,form=~resp3,maxDist=120,resType='n'),main="Semivariogram")
#plot(Variogram(study.lme1,form=~time+resp3,maxDist=120,resType='n'),main="Semivariogram")
#plot(Variogram(study.lme1,form=~resp1+resp3,maxDist=120,resType='n'),main="Semivariogram")
```

```
#plot(Variogram(study.lme1,form=~time+resp1+resp3,maxDist=120,resType='n'),main="Semivariogram
")
study.lme1h=update(study.lme1,correlation=corARMA(form=~time,p=0,q=1))
study.lme1i=update(study.lme1,correlation=corARMA(form=~time,p=1,q=0))
study.lme1j=update(study.lme1,correlation=corARMA(form=~time,p=1,q=1))
plot(ACF(study.lme1h,maxLag=5,resType= 'n'),alpha=.01)
plot(ACF(study.lme1i,maxLag=5,resType= 'n'),alpha=.01)
plot(ACF(study.lme1j,maxLag=5,resType= 'n'),alpha=.01)
summary(study.lme1h)
summary(study.lme1i)
summary(study.lme1j)
```

#4) GLMM Models

```
library(lme4)
study.glmer1=glmer(resp2~ resp1+resp3+ (time+resp1+resp3|subject),data=study,family=
gaussian,nAGQ=5)
study.glmer2=glmer(resp2~ resp1+resp3+ (time+resp1 |subject),data=study,family= gaussian,nAGQ=5)
study.glmer3=glmer(resp2~ resp1+resp3+ (time |subject),data=study,family= gaussian,nAGQ=5)
study.glmer4=glmer(resp2~ resp1+resp3+ (1 |subject),data=study,family= gaussian,nAGQ=5)
study.glmer5=glmer(resp2~ resp1+resp3+ (resp1+resp3 |subject),data=study,family= gaussian,nAGQ=5)
study.glmer6=glmer(resp2~ time+resp1+resp3+ (time+resp1+resp3|subject),data=study,family=
gaussian,nAGQ=5)
study.glmer7=glmer(resp2~ time+resp1+resp3+ (resp1+resp3|subject),data=study,family=
gaussian,nAGQ=5)
study.glmer8=glmer(resp2~ time+resp1+resp3+ (time|subject),data=study,family= gaussian,nAGQ=5)
study.glmer9=glmer(resp2~ time+resp1+resp3+ (1|subject),data=study,family= gaussian,nAGQ=5)
study.glmer10=glmer(resp2~ resp1+resp3+ (time+resp1+resp3|subject),data=study,family=
gaussian(link= "log"),nAGQ=5)#AIC very high!
#study.glmer11=glmer(resp2~ resp1+resp3+ (time+resp1+resp3|subject),data=study,family=
Gamma,nAGQ=5)#Does not work: “General form of glmer_linkinv not yet written”
print(study.glmer1,correlation=F)#Winning model
print(study.glmer2,correlation=F)
print(study.glmer3,correlation=F)
print(study.glmer4,correlation=F)
print(study.glmer5,correlation=F)
print(study.glmer6,correlation=F)
print(study.glmer7,correlation=F)
print(study.glmer8,correlation=F)
print(study.glmer9,correlation=F)
print(study.glmer10,correlation=F)
anova(study.glmer1,study.glmer2)
anova(study.glmer1,study.glmer3)
anova(study.glmer1,study.glmer4)
anova(study.glmer1,study.glmer5)
```

#5) GEE Models

```
library(gee)
```

#5a) Unstructured Correlation Structure

```
#study.gee0=gee(resp2~ resp1+resp3,data=study,family=Gamma,id=subject,corstr=
"unstructured")#does not work, program hangs
```

```
#study.gee02=gee(resp2~ resp1+resp3,data=study,family=inverse.gaussian,id=subject,corstr=
"unstructured")#inverse Gaussian not defined under gee
study.gee1=gee(resp2~ resp1+resp3,data=study,family=gaussian,id=subject,corstr= "unstructured")
study.gee2=gee(resp2~time+ resp1+resp3,data=study,family=gaussian,id=subject,corstr=
"unstructured")
summary(study.gee1)
summary(study.gee2)
```

#5b) Exchangeable Correlation Structure

```
study.gee3=gee(resp2~ resp1+resp3,data=study,family=gaussian,id=subject,corstr= "exchangeable")
study.gee4=gee(resp2~time+ resp1+resp3,data=study,family=gaussian,id=subject,corstr=
"exchangeable")
summary(study.gee3)
summary(study.gee4)
```

#5c) AR(1) Correlation Structure

```
study.gee5=gee(resp2~ resp1+resp3,data=study,family=gaussian,id=subject,corstr= "AR-M",Mv=1)
study.gee6=gee(resp2~time+ resp1+resp3,data=study,family=gaussian,id=subject,corstr= "AR-M",Mv=1)
summary(study.gee5)
summary(study.gee6)
```