Soil Moisture Project Final Report

Zhou Li, Yingzi Xu
Advisor: Dr. Ana-Maria Staicu

April 24, 2013

1 Introduction

1.1 Data description

We are provided with the soil moisture data (measured about once a week and different depths under earth) from 2008 to 2012. Take year=2008 and depth=10cm as an example, the variables are:

- M_{ij}: Soil moisture of subject i (averaged over three θ's) at time t_j (t_j is different from year to year and for 2008 t_j ranges from Jun 13 to Dec 19);
- g: Tillage group indicator for each subject i, $g = 1, 2, \ldots, 6$;
- t_j: Time point when soil moisture is measured;
- p_j: Precipitation (cm) at time t_j.

where $i = 1, 2, \ldots, 24, j = 1, 2, \ldots, 25$. Given the data, we are interested in analyzing: 1) how the trend profiles vary over time in each treatment group and 2) whether the trend is different across tillage treatments. We are going to fit generalized additive models on soil moisture over time based on longitudinal analysis. Due to limit of time, in this report we only focus on the data of year=2008 and depth=10cm.

1.2 Generalized additive models

Generalized additive models (GAM) are nonparametric models in which one or more regression variables are present and can make different smooth contributions to the mean function. We assume response variable Y is of some exponential family and has mean μ, through a link function $g(\mu)$, we model it as a smooth function $f(x)$ of at least some (possibly all) covariates:

$$g(\mu) = \beta_0 + f_1(x_1) + \cdots + f_m(x_m) \tag{1}$$

The functions $f_i(x_i)$ in (1) may be fit using parametric or non-parametric means, thus providing the potential for better fits to data than other methods. By allowing nonparametric fits, well designed GAM allow good fits to the training data with relaxed assumptions on the actual relationship, perhaps at the expense of interpretability of results.

2 Experiment Design

2.1 Remove Seasonality

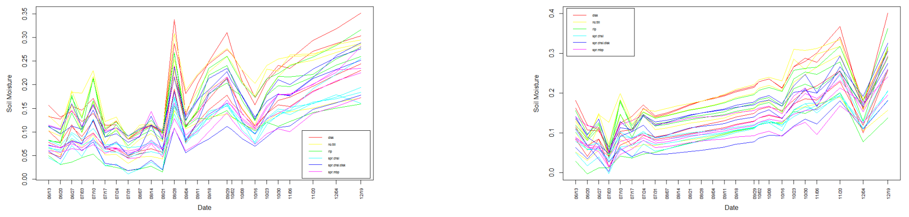
Seasonal components in a time series, theoretically, happen with similar magnitude during the same time period. The seasonal components of a series are often considered to be uninteresting in their own right and to cause the interpretation of a series to be ambiguous. By removing the seasonality, it is easier to focus on other components like trend.

In our scenario, we assume

$$M_{ij} = \mu_{ij} + S_{ij} + \epsilon_{ij} \tag{2}$$

Where μ_{ij} is the trend; the noise ϵ_{ij} fluctuates about zero, i.e.,$E(\epsilon_{ij}) = 0$; the seasonality component S_{ij} is such that $S_{ij} = S_{i,j-d}$,where $d = 12$ denotes the (approximate) number of weeks during the period. We treat fall and winter have the same seasonal component.

We then use function ”decompose” in R to remove the seasonality. Basically, it first uses moving average method to get the trend and then estimates seasonal effects S_{ik} for $k = 1, \ldots, d$.



(a) Original data

(b) Seasonality-detrended data

Figure 1: Soil moisture after removing seasonality for year 2008 and depth=10 cm

2.2 Model specification

2.2.1 Determine covariance structure

One essential part of longitudinal data analysis is to figure out appropriate covariance models and corresponding correlation models between different time points. Let $\boldsymbol{Y}_i^g = (Y_{i1}^g, \ldots, Y_{iJ}^g)^\intercal$ be a vector of soil moisture (after removing seasonality trend) of subject i with tillage g. The model we assume is

$$\boldsymbol{Y}_i^g = \boldsymbol{\beta}_0 + \boldsymbol{\mu}_g + \boldsymbol{\epsilon}_i \tag{3}$$

where $\boldsymbol{\beta}_0$ is the intercept and $\boldsymbol{\mu}_g$ is the mean trend over time of tillage group g. $\boldsymbol{\epsilon}_i$ is an overall random deviation which describes how observations within a data vector vary about the mean and covary among each other. For the purpose of testing the difference between tillage treatments, we alternatively define $\boldsymbol{\mu}_g = \boldsymbol{\mu}_1 + \boldsymbol{\alpha}_g$ for $g = 2, \ldots, 6$. For example, testing $H_0 : \boldsymbol{\mu}_1 = \boldsymbol{\mu}_2$ is equivalent to testing $H_0 : \boldsymbol{\alpha}_2 = \mathbf{0}$. Hence (3) can be written as

Model 1.

$$\boldsymbol{Y}_i^g = \boldsymbol{\beta}_0 + \boldsymbol{\mu}_1 + \sum_{g'=2}^{6} \boldsymbol{\alpha}_{g'} \mathbb{1}\{g = g'\} + \boldsymbol{\epsilon}_i \tag{4}$$

$\boldsymbol{\mu}_1$ and $\boldsymbol{\alpha}_g$ are assumed to be values of some functions of t_j (generalized additive models), i.e.

$$\boldsymbol{\mu}_1 = \Big(\mu_1(t_1), \ldots, \mu_1(t_J)\Big)^\intercal, \boldsymbol{\alpha}_g = \Big(\alpha_g(t_1), \ldots, \alpha_g(t_J)\Big)^\intercal$$

In practice, the GAM model always assume the error term to be independent and identically distributed, which means $\epsilon_{ij} \overset{\text{iid}}{\sim} N(0, \sigma^2)$. Then

$$\boldsymbol{\Sigma}_1 = Var(\boldsymbol{\epsilon}_i) = \sigma^2 \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}$$

Notice that this assumption is too strong and unreasonable for longitudinal data where correlation exists between observations of different time points. But (4) is still helpful for us to determine a better covariance structure, which will be discussed in section 2.2.2.

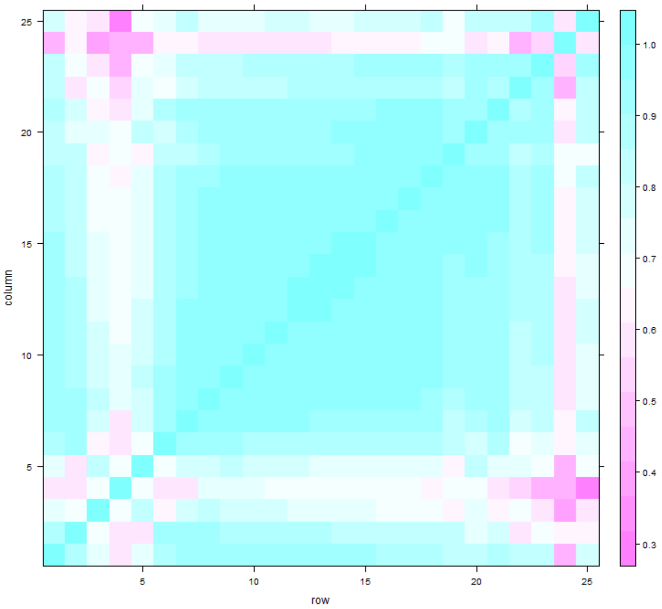


Figure 2: Level plot of correlation matrix for residuals of model 1

2.2.2 GAM based on transformed data

From Figure 2 we can see that the correlation matrix for residuals follows a structure of compound symmetric. Therefore we assume that each element of $\boldsymbol{\epsilon}_i$ is actually the sum of two random terms, i.e.

$$\epsilon_{ij} = b_i + e_{ij}$$

where the random effect b_i has to do with variation among units and e_{ij} has to do with variation within units. Under this assumption the covariance matrix turns out to be

$$\boldsymbol{\Sigma}_2 = Var(\boldsymbol{\epsilon}_i) = \sigma^2 \begin{pmatrix} 1 & \rho & \cdots & \rho \\ \rho & 1 & \cdots & \rho \\ \vdots & \vdots & \ddots & \vdots \\ \rho & \rho & \cdots & 1 \end{pmatrix}$$

For estimating $\boldsymbol{\Sigma}_2$, we use SAS proc mixed on the residuals of the first model with iid assumptions. Once we know the form of $\boldsymbol{\Sigma}_2$, we can left multiply (4) by $\boldsymbol{\Sigma}_2^{-\frac{1}{2}}$ to get

Model 2.

$$\boldsymbol{Y}_i^{g*} = \boldsymbol{\beta}_0^* + \boldsymbol{\mu}_1^* + \sum_{g'=2}^{6} \boldsymbol{\alpha}_{g'}^* \mathbb{1}\{g = g'\} + \boldsymbol{\epsilon}_i^* \tag{5}$$

where $\boldsymbol{Y}_i^{g*} = \boldsymbol{\Sigma}_2^{-\frac{1}{2}}\boldsymbol{Y}_i^g, \boldsymbol{\beta}_0^* = \boldsymbol{\Sigma}_2^{-\frac{1}{2}}\boldsymbol{\beta}_0, \boldsymbol{\mu}_1^* = \boldsymbol{\Sigma}_2^{-\frac{1}{2}}\boldsymbol{\mu}_1, \boldsymbol{\alpha}_{g'}^* = \boldsymbol{\Sigma}_2^{-\frac{1}{2}}\boldsymbol{\alpha}_{g'}, \boldsymbol{\epsilon}_i^* = \boldsymbol{\Sigma}_2^{-\frac{1}{2}}\boldsymbol{\epsilon}_i$. This means if we fit GAM model on the transformed data $\boldsymbol{Y}_i^{g*}$, the elements of $\boldsymbol{\epsilon}_i^*$ would be iid normal.

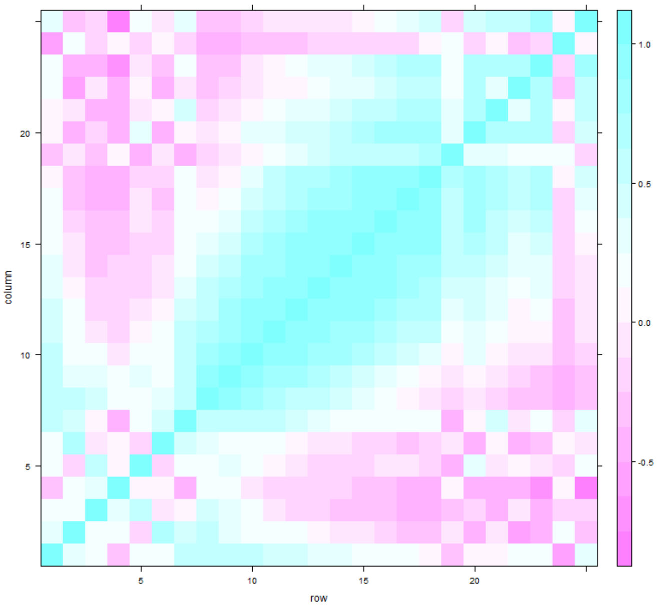


Figure 3: Level plot of correlation matrix for residuals of model 2

Figure 3 shows the correlation matrix for residuals of model 2. We can see that the correlation matrix seems more like diagonal than the first model. Based on the results (see Table 1), we can test whether treatment 1 (disk) is different from the other 5 treatments.

	edf	Ref.df	F	p-value
$\boldsymbol{\mu}_1^*$	7.95	8.51	61.33	0.00
$\boldsymbol{\alpha}_2^*$	7.53	8.60	1.52	0.14
$\boldsymbol{\alpha}_3^*$	7.53	8.60	0.62	0.78
$\boldsymbol{\alpha}_4^*$	7.53	8.60	2.65	0.01
$\boldsymbol{\alpha}_5^*$	7.53	8.60	1.42	0.18
$\boldsymbol{\alpha}_6^*$	7.53	8.60	2.93	0.00

Table 1: Approximate significance of smooth terms

We can also restore the original trend estimate by setting

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_0 &= \hat{\boldsymbol{\Sigma}}^{\frac{1}{2}}\hat{\boldsymbol{\beta}}_0^* \\
\hat{\boldsymbol{\mu}}_1 &= \hat{\boldsymbol{\Sigma}}^{\frac{1}{2}}\hat{\boldsymbol{\mu}}_1^* \\
\hat{\boldsymbol{\alpha}}_{g'} &= \hat{\boldsymbol{\Sigma}}^{\frac{1}{2}}\hat{\boldsymbol{\alpha}}_{g'}^*
\end{aligned}$$

Figure 4 shows the mean trend for all the 6 tillage groups.

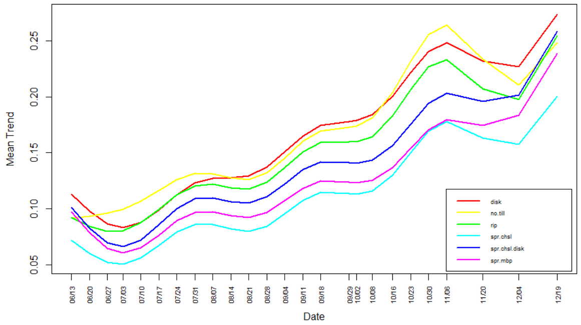


Figure 4: Estimated mean trend for different tillage groups

3 Conclusions

Based on Table 1 and Figure 4, we can conclude that treatment 1 (disk) does not differ significantly on soil moisture from treatment 2 (no.till), treatment 3 (rip) and treatment 5 (spr.chsl.disk); but it is different from treatment 4 (spr.chsl) and treatment 6 (spr.mbp), which lead to lower soil moisture. Similarly we can test the difference between any tillage groups by changing the baseline trend in (4).

Trt	vs	Trt	Trt	vs	Trt	Trt	vs	Trt
2	$\approx$	1	4	$<$	1	6	$<$	1
2	$=$	2	4	$<$	2	6	$<$	2
2	$\approx$	3	4	$\approx$	3	6	$\approx$	3
2	$>$	4	4	$=$	4	6	$\approx$	4
2	$>$	5	4	$\approx$	5	6	$\approx$	5
2	$>$	6	4	$\approx$	6	6	$=$	6

Table 2: Pairwise comparison between different treatments

4 Computing details

4.1 R code

```
# clean up

rm(list = ls())

gc()

library(ggplot2)

library(mgcv)
library(fields)
library(xtable)
setwd(dir="C:/Users/Zhou Li/Desktop/Classes/Consulting/fwdregressionresultsoftheconsultingproject")
load("Data2008.RData")
# load("Data2010.RData")
load("Rain.RData")

####rmatrix.csv is covariance matrix from sas output####
rmatrix<-read.csv(file="rmatrix.csv",header=T)
rmatrix<-as.matrix(rmatrix[,-(1:2)])
tran<-rmatrix+0.000052
sqrtma<-function(x){
  e<-eigen(x)
  V <- e$vectors
  return(V %*% diag(sqrt(e$values)) %*% t(V))
}
sqrtrtran<-sqrtma(tran)
invtran<-solve(tran)
sqrtinvtran<-sqrtma(invtran)

######################################################


# Keep correspoding date point
data2<-data2[data2$date%in%t,]



# CODE WRITTEN ON Tuesday March 26
# [AMS]
# Data 2008 contains:
#
# ydata: 24(rep)*6(depth)*25(time)
#
```

```
# time: 1*25
#
# treat: 1*24
#
# depth: 1*6

##############remove the date of highest###############################
# t<-t[-12]
# ydata<-ydata[,,-12]
# data2<-data2[-12,]
#######################################################################

group=rep(1:6, each=4) # group membership
group=as.factor(group)
data=ydata[,1,] # look at data at depth = 10cm
# data=log(data/0.55/(1-data/0.55))
dim(data)
n.curves=dim(data)[1]
m.tt=dim(data)[2]; length(t)



tt=as.numeric(difftime(t,t[1], units="days"))
tt

c<-rep(rainbow(6),each=4)
ll<-rep(rep(1,24))
matplot(t,t(data),col=c,type="l",lty=ll,
        xlab="Date", ylab="Soil Moisture",
        lwd=1.5,xaxt="n")
axis(1,at=t,labels=format(t,"%m/%d"),las=2,cex.axis=0.7)
legend("bottomright",c("disk","no.till","rip",
                       "spr.chsl","spr.chsl.disk","spr.mbp"),
       lty=rep(1,6),lwd=1.5,col=rainbow(6),inset=0.01,cex=.5)
```

```
# do qq-plot to assess normality
qqnorm(y=(data))
qqline(y=(data))
##############################################################
# remove seasonal trend
data.deseason<-matrix(0,24,25)

for(i in 1:nrow(data.deseason)){
  data.ts<-ts(data[i,],start=c(1,1),end=c(3,1),frequency=12)  #25 time points, 2 cycle
  #data.ts<-ts(data[1,-12],start=c(1,1),end=c(2,12),frequency=12)

  m<-decompose(data.ts)
  deseason<-data[i,]-m$season
  data.deseason[i,]<-deseason

}

# matplot(tt, t(data.deseason), type="l",lty="solid")
matplot(t,t(data.deseason),col=c,type="l",lty=ll,
        xlab="Date", ylab="Soil Moisture",
        lwd=1.5,xaxt="n")
axis(1,at=t,labels=format(t,"%m/%d"),las=2,cex.axis=0.7)
legend("topleft",c("disk","no.till","rip",
                       "spr.chsl","spr.chsl.disk","spr.mbp"),
       lty=rep(1,6),lwd=1.5,col=rainbow(6),inset=0.01,cex=.5)
# qqnorm(y=(data.deseason))
# qqline(y=(data.deseason))
##############################################################
datanew<-data.deseason%*%invcov
# datanew<-data.deseason
```

```
matplot(tt, t(datanew), type="l",lty="solid")

# DO NOT TRANSFORM THE DATA
# tfdata =data; hist((tfdata))
# tfdata =data.deseason; hist((tfdata))
tfdata =datanew; hist((tfdata))

# plot the transformed data
matplot(tt,t(tfdata), type="l")

#play with the transformed data
apply(tfdata, 2, max)
tt.imp=which(data2[,2]==max(data2[,2]))
# high precipitation level this day


# arrange the data IN vector format
tfdata_vec = c(t(tfdata))
tt_vec = rep(tt, n.curves)
group_vec =rep(group, each=m.tt)
gr2=rep(0, length(group_vec))

# indicator of the high precipitation level
tt.imp.vec = rep(0, length(tt));tt.imp.vec[tt.imp]=1
prec_indicat = rep(tt.imp.vec, n.curves)

soil_2008=data.frame(y=tfdata_vec, gr=group_vec,
                     t=tt_vec,prec_indicat=prec_indicat)
# write.table(soil_2008,file="soil2008.csv",sep=",",row.names=F)

###################################################

# wrain<-rep(data2$weight3,n.curves)
# rain<-rep(data2$X2008.cm,n.curves)

###################################################



# fit model with smooth group mean curves + linear effect when the high precipitation occured


# out_rain = gam(y~rain+s(t,id=1)+s(t, by=as.numeric(gr==2),id=1)+
#             s(t, by=as.numeric(gr==3),id=1)+
#             s(t, by=as.numeric(gr==4),id=1)+
#             s(t, by=as.numeric(gr==5),id=1)+
#             s(t, by=as.numeric(gr==6),id=1), data=soil_2008 )
out_ind = gam(y~prec_indicat+s(t,id=1)+s(t, by=as.numeric(gr==2),id=1)+
                 s(t, by=as.numeric(gr==3),id=1)+
                 s(t, by=as.numeric(gr==4),id=1)+
                 s(t, by=as.numeric(gr==5),id=1)+
                 s(t, by=as.numeric(gr==6),id=1), data=soil_2008 )
# out_wrain = gam(y~wrain+s(t,id=1)+s(t, by=as.numeric(gr==2),id=1)+
#                 s(t, by=as.numeric(gr==3),id=1)+
#                 s(t, by=as.numeric(gr==4),id=1)+
#                 s(t, by=as.numeric(gr==5),id=1)+
#                 s(t, by=as.numeric(gr==6),id=1), data=soil_2008 )
out_norain = gam(y~s(t,id=1)+s(t, by=as.numeric(gr==2),id=1)+
                  s(t, by=as.numeric(gr==3),id=1)+
                 s(t, by=as.numeric(gr==4),id=1)+
                  s(t, by=as.numeric(gr==5),id=1)+
                  s(t, by=as.numeric(gr==6),id=1), data=soil_2008 )



#out = gam(y~s(t)+s(t, by=as.numeric(gr==2))+
#            s(t, by=as.numeric(gr==3))+
#            s(t, by=as.numeric(gr==4))+
#            s(t, by=as.numeric(gr==5))+
#            s(t, by=as.numeric(gr==6)), data=soil_2008 )


out<-out_norain

summary(out)

# check the model

AIC(out)

anova(out_norain,out_ind,test="Chisq")

gam.check(out, k.rep = 1000)



plot(out,pages=1,rug=FALSE,shade=T,shade.col=’gray90’,col=’#FF8000’)

# Rain on the x axis, Time on the y

# with values on the response scale given by the contours

# with lighter color indicating higher values

# vis.gam(out, type = "response", plot.type = "contour")

# vis.gam(out, type = "response", plot.type = "persp",

#         phi = 30, theta = 300, n.grid = 500, border = NA)


terms.full <- predict(out, type="terms")

terms.full[1:101,2]

fit.full <- Reduce("+", terms.full)

cm.full <- coef(out)




resid = out$residuals


########output a dataset for SAS analysis###########

id<-rep(1:n.curves,each=m.tt)
```

```
soilresid=data.frame(resid=resid, gr=group_vec,t=tt_vec,id=id,block=block)

write.table(soilresid,file="soil2008.csv",sep=",",row.names=F)

#####################################################


resid_mat=matrix(resid,nrow=m.tt)

matplot(tt,resid_mat, type="l")



cov_est=cov(t(resid_mat))

cor_est=cor(t(resid_mat))



image.plot(cov_est)


plot(diag(cov_est),type="b")


library(lattice)

levelplot(cov_est)

levelplot(cor_est)


##pairwise comparison##

out_B = gam(y~s(t,id=1)+s(t, by=as.numeric(gr==1),id=1)+

                   s(t, by=as.numeric(gr==3),id=1)+

                   s(t, by=as.numeric(gr==4),id=1)+

                   s(t, by=as.numeric(gr==5),id=1)+

                   s(t, by=as.numeric(gr==6),id=1), data=soil_2008 )

summary(out_B)

out_C = gam(y~s(t,id=1)+s(t, by=as.numeric(gr==1),id=1)+

              s(t, by=as.numeric(gr==2),id=1)+

              s(t, by=as.numeric(gr==4),id=1)+

              s(t, by=as.numeric(gr==5),id=1)+

              s(t, by=as.numeric(gr==6),id=1), data=soil_2008 )

summary(out_C)
```

```
out_D = gam(y~s(t,id=1)+s(t, by=as.numeric(gr==1),id=1)+
              s(t, by=as.numeric(gr==2),id=1)+
              s(t, by=as.numeric(gr==3),id=1)+
              s(t, by=as.numeric(gr==5),id=1)+
              s(t, by=as.numeric(gr==6),id=1), data=soil_2008 )
summary(out_D)
out_E = gam(y~s(t,id=1)+s(t, by=as.numeric(gr==1),id=1)+
              s(t, by=as.numeric(gr==2),id=1)+
              s(t, by=as.numeric(gr==3),id=1)+
              s(t, by=as.numeric(gr==4),id=1)+
              s(t, by=as.numeric(gr==6),id=1), data=soil_2008 )
summary(out_E)
out_F = gam(y~s(t,id=1)+s(t, by=as.numeric(gr==1),id=1)+
              s(t, by=as.numeric(gr==2),id=1)+
              s(t, by=as.numeric(gr==3),id=1)+
              s(t, by=as.numeric(gr==4),id=1)+
              s(t, by=as.numeric(gr==5),id=1), data=soil_2008 )
summary(out_F)

##############plot the estimated trend#############
beta0<-sqrcov%*%matrix(out$coefficients[1],25,1)
mu1<-sqrcov%*%terms.full[1:25,1]
mu2<-sqrcov%*%terms.full[1:25,1]+sqrcov%*%terms.full[101:125,2]
mu3<-sqrcov%*%terms.full[1:25,1]+sqrcov%*%terms.full[201:225,3]
mu4<-sqrcov%*%terms.full[1:25,1]+sqrcov%*%terms.full[301:325,4]
mu5<-sqrcov%*%terms.full[1:25,1]+sqrcov%*%terms.full[401:425,5]
mu6<-sqrcov%*%terms.full[1:25,1]+sqrcov%*%terms.full[501:525,6]
trend<-cbind(beta0+mu1,beta0+mu2,beta0+mu3,beta0+mu4,beta0+mu5,beta0+mu6)
matplot(t,trend,col=rainbow(6),lty=1,type="l",lwd=2,
        xlab="Date", ylab="Mean Trend",xaxt="n")
axis(1,at=t,labels=format(t,"%m/%d"),las=2,cex.axis=0.7)
legend("bottomright",legend=c("disk","no.till","rip",
                         "spr.chsl","spr.chsl.disk","spr.mbp"),
       lty=1,lwd=2,col=rainbow(6),inset=0.01,cex=.55)



#######output for Latex#####
# try<-summary(out)
# xtable(try$s.table)
```

4.2 SAS code

```
data soil;
infile 'soil2008.csv' dlm=',' dsd firstobs=2;
input resid gr t id block $;
run;

proc mixed method=ml data=soil;
class id gr block;
model resid=;
repeated/subject=id type=cs r rcorr;
ods output R=rmatrix;
run;
```