Test Report

Wangshu Zhang, March 2016

wangshuz@usc.edu

Problem #1

1. Which biomarkers are changing as a result of treatment?

The biomarker data has 7020 observations, with 39 subjects, 20 markers, and observed during 9 weeks (WEEK 0 – WEEK 8). I exclude all biomarker data that was collected after the occurrence of the Adverse Event (because they are not under treatment any more), and there are 6520 observations left. Then I plot the boxplot of the "MARKER.VALUE" for the 6520 observations grouped by "MARKER.NAME", and each sub-plot demonstrates the fold-change trend during WEEK1 – WEEK8, as shown in Figure 1.

From the figure we can see that the values of Markers M1, M2, M3, and M4 increase more obviously than those of the other markers. However, the four changed biomarkers may not be related to the treatment (for example, may be caused by disease progression).

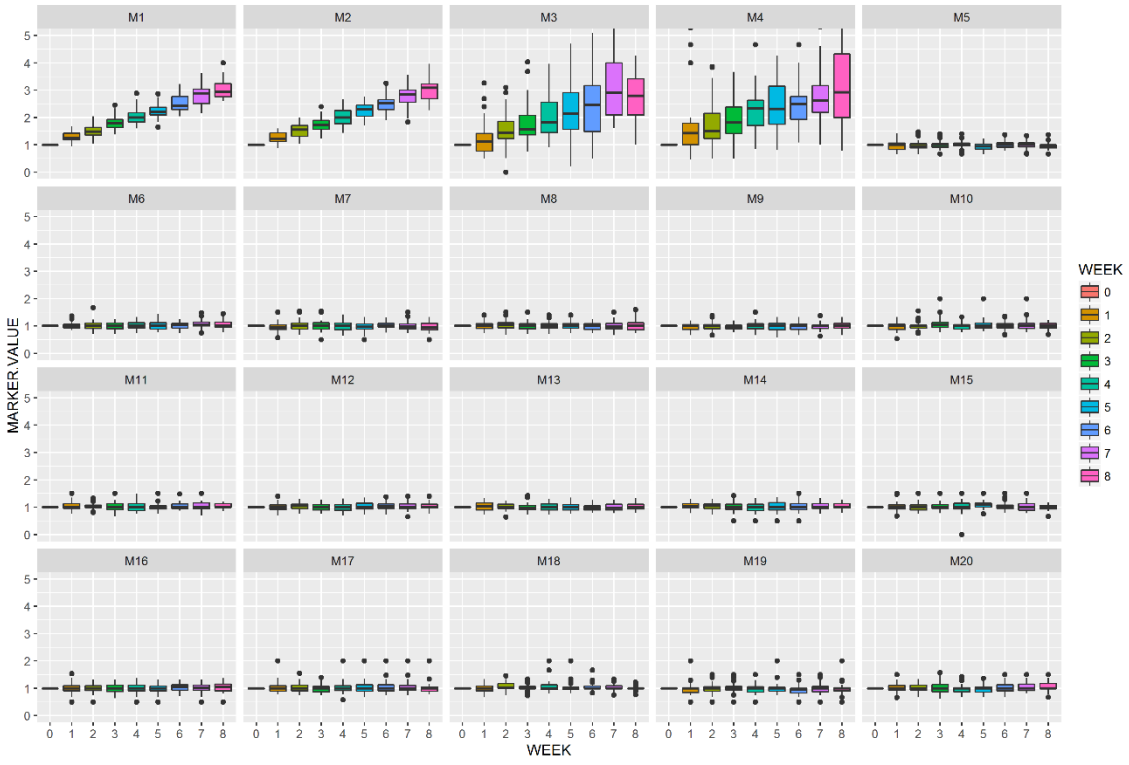

Figure 1: Boxplot of marker values for all 39 subjects under treatment (AE excluded)

To see which biomarkers are changing as a result of treatment, I consider only the 12 subjects that have adverse event (AE) occurred, since they have both treatment and non-treatment observations. I plot the mean of both the treated and the untreated marker values, grouped by "MARKER.NAME", as shown in Figure 2. The error bars are the standard deviations among subjects.

From the figure we can see that for Biomarkers 1 – 4, there are obvious gaps between the treatment group (green lines) and non-treatment (red lines) after WEEK 4, which is the earliest time that AE is observed. Therefore Markers M1, M2, M3, and M4 are all changing as a result of treatment.

To quantify the difference between treatment and non-treatment groups, I implement a paired t-test for them. Values of the various time-course data for the same subject and same marker name are averaged. The result is shown in Table 1. Without correction we can see that only Markers 1 – 4 are significantly different between the two groups, with *p*-value < 0.05.

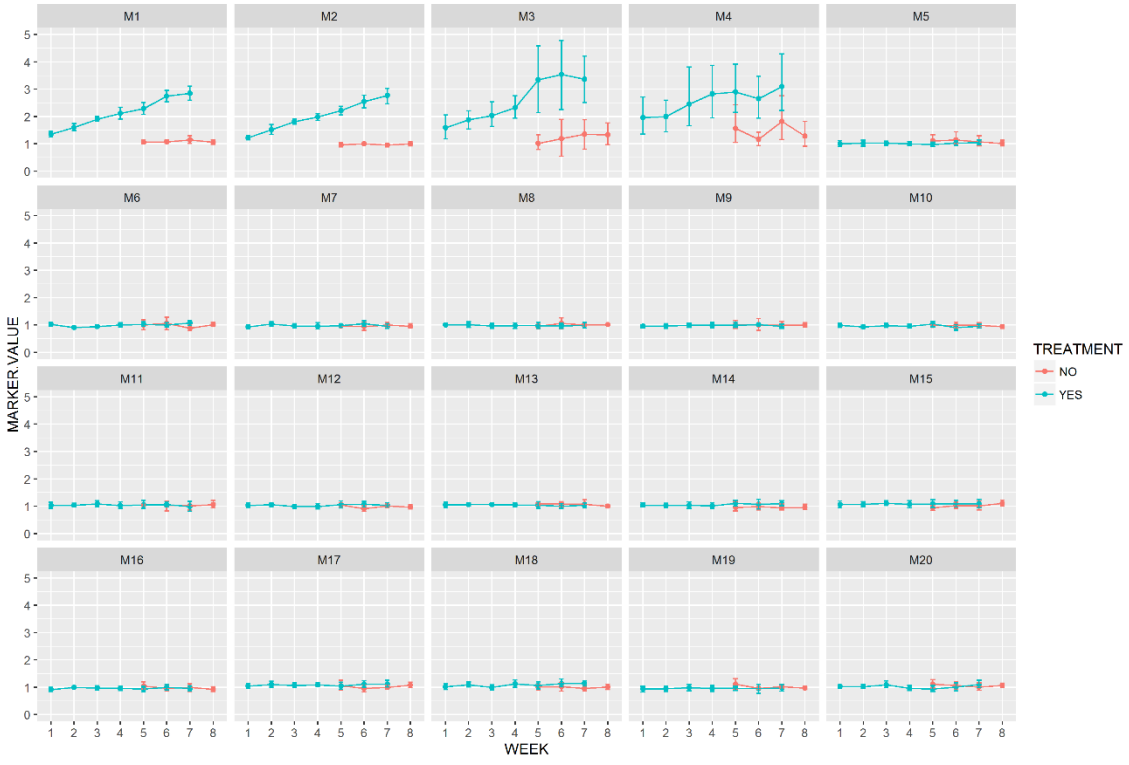

Figure 2: Mean and Standard deviation of marker values for the 12 subjects having AE, comparison between the treated (before AE) vs. the untreated (after AE) groups

Table 1: **Difference in marker values between two groups with or without treatment, for all subjects with adverse events**. Meaning of columns: "**Marker**" – name of Markers; "***p*-value**" – measure of statistical significant between the non-treatment samples and treatment samples; "**Mean**" – mean of marker values in the group; "**S.E**" – standard deviation of marker values in the group.

Marker	*p*-value	Non-treatment		Treatment	
		Mean	*S.E*	*Mean*	*S.E*
M1	4.6930e-10	1.0522	0.1403	2.0069	0.2165
M2	2.0474e-08	0.9994	0.1263	1.8716	0.2934
M3	4.8740e-05	1.2969	0.7525	2.3501	0.9581
M4	6.7454e-04	1.2368	0.6239	2.5382	1.4525
M5	0.6136	1.0225	0.1777	1.0108	0.1335
M6	0.9814	1.0270	0.1402	0.9821	0.1005
M7	0.3847	0.9659	0.1399	0.9754	0.1183
M8	0.8272	1.0170	0.0755	0.9877	0.1231
M9	0.9132	1.0047	0.1253	0.9749	0.1292
M10	0.5840	0.9680	0.0985	0.9630	0.0919
M11	0.7944	1.0650	0.2509	1.0354	0.1647
M12	0.0128	0.9551	0.1250	1.0313	0.1209
M13	0.1727	1.0137	0.0979	1.0476	0.1033
M14	0.0268	0.9816	0.1787	1.0387	0.1521
M15	0.5034	1.0707	0.1957	1.0705	0.1813
M16	0.3024	0.9514	0.1529	0.9672	0.0976
M17	0.0497	1.0599	0.1409	1.0773	0.1491
M18	0.0587	1.0011	0.1846	1.0594	0.1583
M19	0.7155	1.0002	0.0734	0.9740	0.1515
M20	0.9615	1.0690	0.1273	1.0282	0.1461

Finally to further validate our discovery, I plot for each of the 12 subject, for each marker, the marker values before and after the termination of treatment. Subjects with the same AE occurrence time are drawn together. Therefore Figure 3 is for Subjects 1- 4, Figure 4 is for Subject 5, and Figure 5 is for Subjects 6-12. From the figures we can observe obvious climbs-up during the treatment and also obvious falls-down after treatment is terminated for Markers 1- 4. And the sample variance of marker values for Marker 3 and Marker 4 are larger than those of Marker 1 and Marker 2.

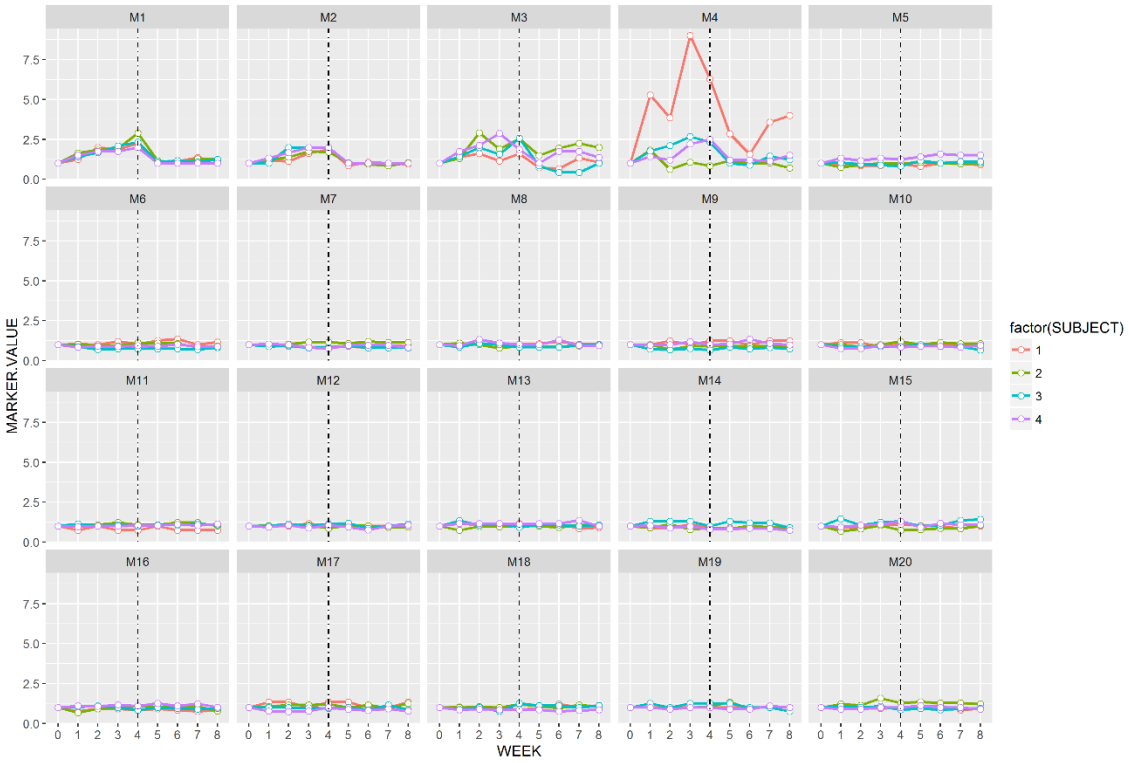

Figure 3: Marker value change for SUBJECT 1-4 (Vertical dash means AE in WEEK 4)

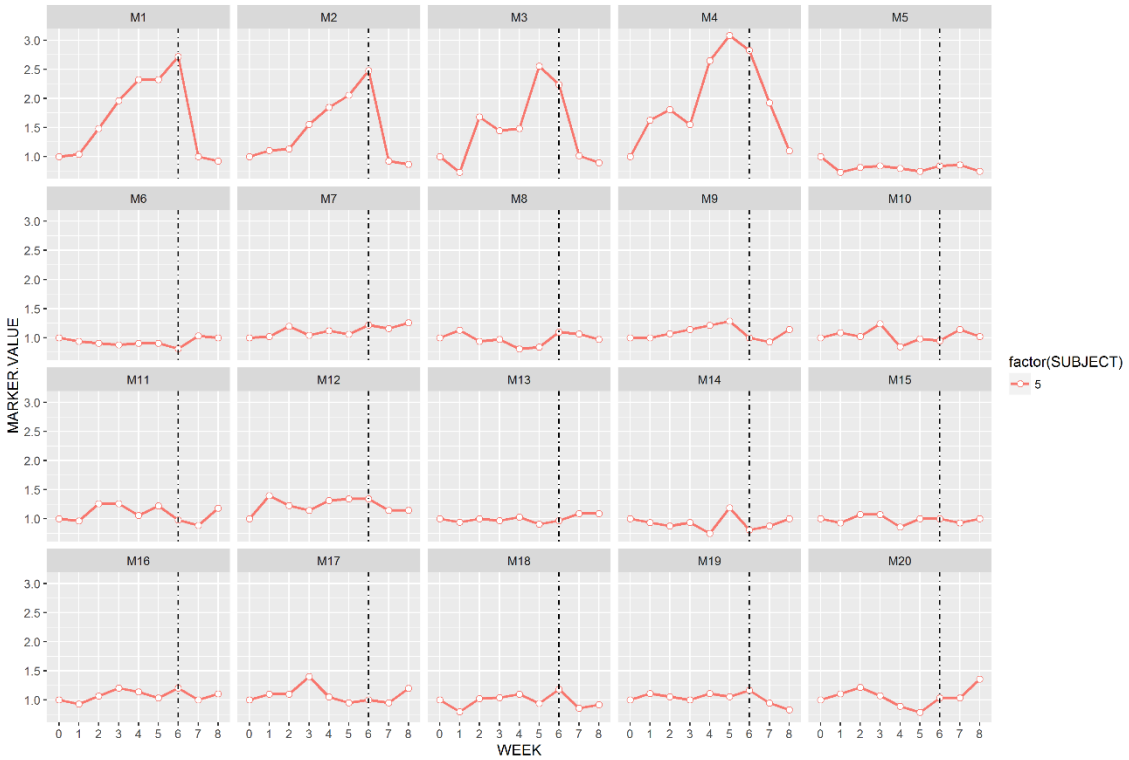

Figure 4: Marker value change for SUBJECT 5 (Vertical dash means AE in WEEK 6)

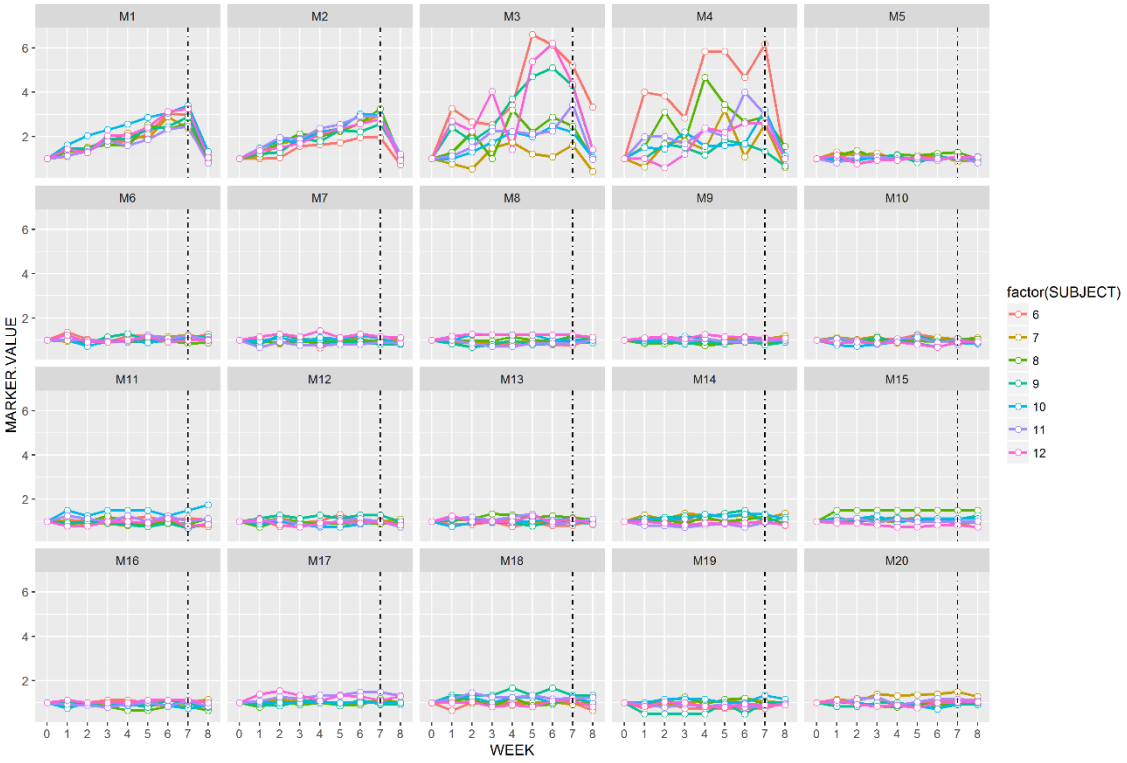

Figure 5: Marker value change for SUBJECT 6-12 (Vertical dash means AE in WEEK 7)

2. The change (fold change) in which markers is related to the event?

Focusing on Markers M1 – M4, I compare the marker values of the 12 subjects having AE (AE group), and the remaining 27 subjects not having AE (NAE group), during their treatment period. I plot the mean of both the AE and the NAE marker values, grouped by "MARKER.NAME", as shown in Figure 6. The error bars are the standard deviations among subjects. From the figure we can see that values of Markers 1, 3, and 4 are obviously different between the AE and the NAE groups, while there's no significant gap between the two groups for Marker 2. In other word, we guess that only Marker 1, 3, and 4 are related to the adverse event.

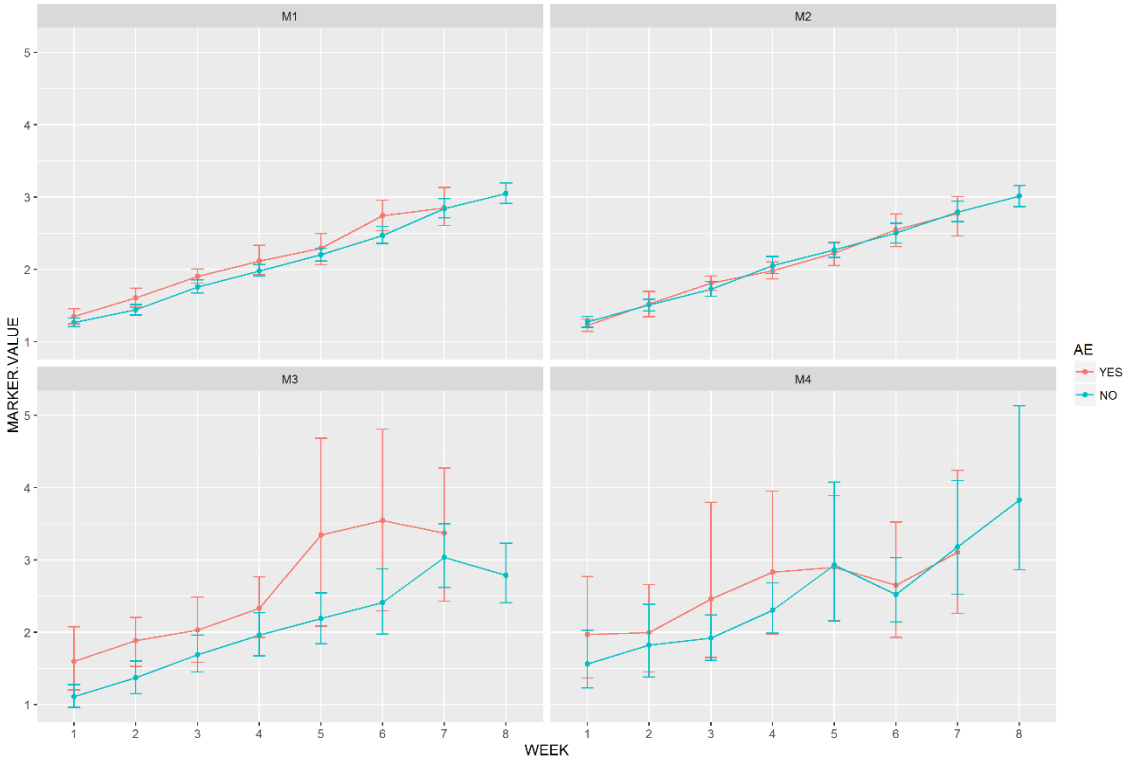

Figure 6: Mean and Standard deviation of marker values for the 12 subjects having AE, and 27 subjects not having AE, during their treatment periods

To quantify this discovery, I calculate the mean and standard deviations of the marker values for the AE subjects and NAE subjects for each week from WEEK 1-WEEK 7, respectively, and also implement a one sample t-test to show if the differences between the two group means is significantly different from 0, with results shown in Table 2. From the result we can see that for Marker 1, 3, and 4, the difference in means of the AE and NAE marker values across eight weeks is significantly not equal to 0. Therefore, we verified that Marker 1, 3, and 4 are related to the adverse event, with possible predictive power M3 > M1 > M4.

Table 2: **Difference in marker values between AE and NAE subjects, during their treatment period**. Meaning of columns: "**Marker**" – name of Markers; "***p*-value**" – statistical significant of difference in AE and NAE sample means; "**Mean**" – mean of marker values in the group; "**S.E**" – standard deviation of marker values in the group.

Marker	WEEK	*p*-value	AE		NAE	
			Mean	*S.E*	*Mean*	*S.E*
M1	1	0.0061	1.351759	0.196117	1.266084	0.158734
	2		1.605836	0.249937	1.441509	0.1913
	3		1.902356	0.179494	1.758997	0.243461
	4		2.116858	0.385678	1.980156	0.22854
	5		2.292096	0.332108	2.203451	0.242635
	6		2.743666	0.334075	2.468083	0.303268
	7		2.851748	0.39368	2.842326	0.390567
M2	1	0.7985	1.226571	0.165424	1.273113	0.210581
	2		1.523266	0.330112	1.507282	0.211605
	3		1.810642	0.18319	1.725955	0.262136
	4		1.983033	0.223678	2.055038	0.328816
	5		2.224983	0.246377	2.271615	0.287219
	6		2.548202	0.344104	2.50383	0.376588
	7		2.7729	0.413105	2.792574	0.403083
M3	1	0.0041	1.593581	0.794252	1.110064	0.43778
	2		1.884304	0.628984	1.369392	0.558592
	3		2.031224	0.850405	1.687596	0.6853
	4		2.330467	0.771247	1.956618	0.808714
	5		3.342498	1.943161	2.189914	0.991724
	6		3.545128	1.972275	2.410601	1.243152
	7		3.369148	1.316816	3.03236	1.183507
M4	1	0.0481	1.969169	1.337165	1.560915	1.102459
	2		1.994339	1.091119	1.823215	1.355003
	3		2.45939	2.128547	1.919116	0.824672
	4		2.83194	1.796227	2.305213	0.988845
	5		2.899249	1.372755	2.926357	2.592658
	6		2.648861	1.211819	2.521261	1.191497
	7		3.103644	1.480555	3.18064	2.177757

3. Can you come up with an algorithm that can predict the event based on biomarker values or fold changes in biomarkers collected at least a week prior to the actual event? I formulate this prediction task as a binary classification problem. That is, given the history biomarker values (at least one week ago, people can choose the number of lagged-time even till the beginning of the experiment), and we give prediction for this week if the adverse event occurs. If the AE occur, the response variable "AE" is labelled as "YES", otherwise it is labelled as "NO". We first choose feature variables then employ the SVM algorithm to do prediction.

Specifically, we build a positive set and a negative set with known label. Since we already know 12 subjects having AE, therefore we choose them as positive samples, and the rest 27 subjects comprise the negative set. For each subjects in both positive and negative set, I set a parameter "lag" to denote data of how many previous weeks are considered. The default value of variable "lag" is 3, because for subject 1-4 who have AE occurred at the fourth week, we only have previous biomarker data for WEEK1, WEEK2 and WEEK3. In this problem I want to predict for the last week all samples having treatment, therefore for different subjects, by fixing the lag variable, data come from different weeks, as shown in Table 3. Here I only use Marker 1, 3, and 4. Hence when lag = 3, there should be altogether 9 feature vectors, for each of marker at each week.

Next we for each of the positive set and the negative set, I randomly select 2/3 subjects composing the train set, and the remaining 1/3 subjects composing the test set.

Table 3: **Illustration of the structure of feature data**. Rows are subject (39 rows), columns are feature variables determined by the Marker-Week interaction (9 columns for lag = 3). For example "M3-W1" means data collected from MARKER 3 at WEEK 1.

	M1-W1	M1-W2	M1-W3	M3-W1	M3-W2	M3-W3	M4-W1	M4-W2	M4-W3
Subject 1-4	Marker values								
5	M1-W3	M1-W4	M1-W5	M3-W3	M3-W4	M3-W5	M4-W3	M4-W4	M4-W5
	Marker values								
6-12	M1-W4	M1-W5	M1-W6	M3-W4	M3-W5	M3-W6	M4-W4	M4-W5	M4-W6
	Marker values								
13-39	M1-W5	M1-W6	M1-W7	M3-W5	M3-W6	M3-W7	M4-W5	M4-W6	M4-W7
	Marker values								

I use the “e1071” R package to implement SVM, with default settings. I first use a grid search to tune parameters, and use the tuned parameter to predict for train set. The contingency table of the prediction result is illustrated in Table 4. From the table we can see that the model performs very well on classifying negative samples (100% accuracy), but don’t very precise, even for the train set, for the positive samples (accuracy = 5/8 = 62.5%).

Table 4: **Contingency table for the train set** (lag = 3)

		Claim	
		YES	NO
Truth	YES	5	3
	NO	0	18

Then I use this model trained to predict for the test set. The result of contingency table is shown in Table 5. From the result we can see that, the classification accuracy for the positive samples is 75% (= 3/4), and that for the negative samples is 77.8% (7/9), proving that the algorithm is effective, and it is harder to predict positive samples correct than that of predicting negative samples.

Table 5: **Contingency table for the test set** (lag = 3)

		Claim	
		YES	NO
Truth	YES	3	1
	NO	2	7

Here I only consider the time by week, but actually the experimental subjects taking pills daily. If time allowed, I will further think about this problem further by using the multivariate time series classification, if possible.

Problem # 3

I use WinBUGS and the "R2BUGS" R package to estimate posterior means using MCMC Gibbs sampling. For each parameter I sample 1 chain, with 1200 interations (including 200 burn-ins). The trace plots for posterior β_1 and β_2 is shown in Figure 7.

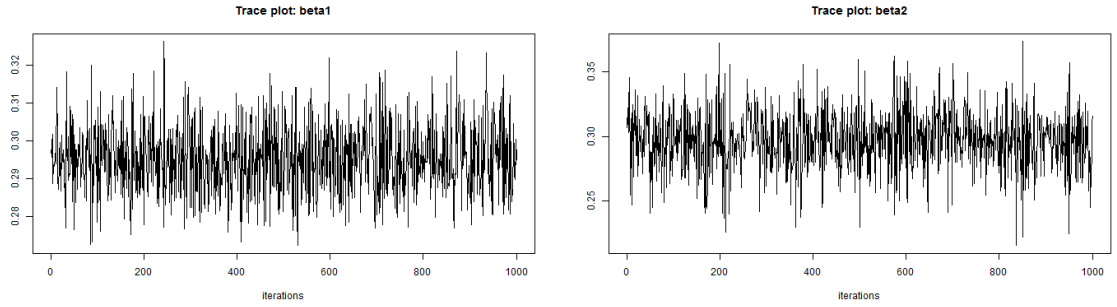

Figure 7: Trace plots for posterior estimates of beta1 and beta2

Also the posterior summaries are:

Inference for Bugs model at "E:/gilead/code_3/model.bug", fit using WinBUGS,
1 chains, each with 1200 iterations (first 200 discarded)
n.sims = 1000 iterations saved

	mean	sd	2.5%	25%	50%	75%	97.5%
beta1	0.3	0	0.3	0.3	0.3	0.3	0.3
beta2	0.3	0	0.2	0.3	0.3	0.3	0.3
deviance	156.3	2	154.4	154.9	155.7	157.1	161.4

DIC info (using the rule, pD = Dbar-Dhat)
pD = 2.0 and DIC = 158.3
DIC is an estimate of expected predictive error (lower deviance is better).
The last plot shows the raw data and the fitted %CV as a function of mu.

More exactly, the posterior mean for β_1 is 0.2953, and that for β_2 is 0.2971, which can be regarded as Bayesian estimates of the two parameters. Then I use these two values to calculate σ, using

$$\sigma(\mu) = \frac{0.2953}{\exp(0.2971\mu)},$$

and the following formula to calculate the fitted CV:

$$CV(\mu) = \sqrt{\exp(\sigma(\mu)^2) - 1} = \sqrt{\exp\left(\left(\frac{0.2953}{\exp(0.2971\mu)}\right)^2\right) - 1}$$

Figure 8 shows the raw data and the fitted %CV as a function of original mu (mean of Y).

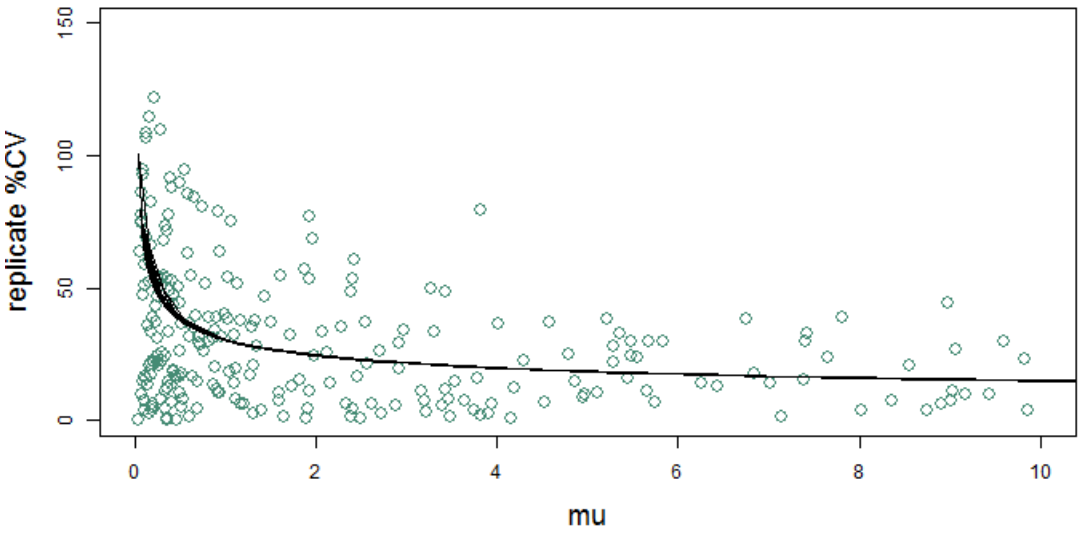

Figure 8: Replicated %CV

Appendix: Derivation of forms of conditional posterior

Let $Z_1 = \log(Y_1)$, $Z_2 = \log(Y_2)$, then given $\sigma = \frac{\beta_1}{e^{\mu\beta_2}}$,

$$p(\beta_1, \beta_2 \mid Z_1, Z_2, \mu) \propto p(Z_1 \mid \beta_1, \beta_2, \mu)\, p(Z_2 \mid \beta_1, \beta_2, \mu)\, p(\beta_1)\, p(\beta_2)$$

$$\propto \frac{1}{\sigma\sqrt{2\pi}} \exp\{-\frac{(z_1-\mu)^2}{2\sigma^2}\} \cdot \frac{1}{\sigma\sqrt{2\pi}} \exp\{-\frac{(z_2-\mu)^2}{2\sigma^2}\} \cdot \frac{0.001^{0.2}}{\Gamma(0.2)} \beta_1^{0.2-1} e^{-0.001\beta_1} \cdot \frac{0.001^{0.2}}{\Gamma(0.2)} \beta_2^{0.2-1} e^{-0.001\beta_2}$$

$$\propto \frac{e^{\mu\beta_2}}{\beta_1} \cdot \frac{e^{\mu\beta_2}}{\beta_1} \cdot \exp\left\{-\frac{e^{2\mu\beta_2}}{2\beta_1^2}[(z_1-\mu)^2 + (z_2-\mu)^2]\right\} \cdot \beta_1^{-0.8} \beta_2^{-0.8} e^{-0.001\beta_1} e^{-0.001\beta_2}$$

Therefore,

$$p(\beta_1, \beta_2 \mid \mathbf{z}_1, \mathbf{z}_2, \mu) \propto \prod_{i=1}^{n} \left(\frac{e^{2\mu_i\beta_2}}{\beta_1^2} \cdot \exp\left\{-\frac{e^{2\mu_i\beta_2}}{2\beta_1^2}[(z_{1i}-\mu_i)^2 + (z_{2i}-\mu_i)^2]\right\}\right) \cdot \beta_1^{-0.8} \beta_2^{-0.8} e^{-0.001\beta_1} e^{-0.001\beta_2}$$

$$\propto \beta_1^{-2n-0.8} e^{-0.001\beta_1} \cdot \beta_2^{-0.8} e^{(2\sum_{i=1}^{n}\mu_i - 0.001)\beta_2} \cdot \exp\{-\frac{1}{2\beta_1^2} \sum_{i=1}^{n} e^{2\beta_2\mu_i}[(z_{1i}-\mu_i)^2 + (z_{2i}-\mu_i)^2]\}$$

Without considering about the constant terms, we obtain the form of conditional posterior for β_1 as:

$$p(\beta_1 \mid \beta_2, \mathbf{z}_1, \mathbf{z}_2) \propto \beta_1^{-2n-0.8} e^{-0.001\beta_1} e^{-\frac{s}{2\beta_1^2}}$$

where $s = \sum_{i=1}^{n} e^{2\beta_2\mu_i}[(z_{1i}-\mu_i)^2 + (z_{2i}-\mu_i)^2]$. Then we further obtain

$$\log p(\beta_1 \mid \beta_2, \mathbf{z}_1, \mathbf{z}_2) \propto (-2n-0.8)\log\beta_1 - 0.001\beta_1 - \frac{s}{2\beta_1^2}$$

$$\frac{\partial \log p(\beta_1 \mid \beta_2, \mathbf{z}_1, \mathbf{z}_2)}{\partial\beta_1} \propto \frac{-2n-0.8}{\beta_1} - 0.001 - \frac{s}{2} \cdot (-2)\beta_1^{-3}$$

$$\propto \frac{-2n-0.8}{\beta_1} - 0.001 + \frac{s}{\beta_1^3}$$

$$\frac{\partial^2 \log p(\beta_1 \mid \beta_2, \mathbf{z}_1, \mathbf{z}_2)}{\partial\beta_1^2} \propto \frac{2n+0.8}{\beta_1^2} - \frac{3s}{\beta_1^3}$$

Also, the form of conditional posterior for β_2 is:

$$p(\beta_2 \mid \beta_1, \mathbf{z}_1, \mathbf{z}_2) \propto \beta_2^{-0.8} e^{(2\sum_{i=1}^{n}\mu_i - 0.001)\beta_2} \cdot e^{-\frac{s}{2\beta_1^2}}$$

and

$$\begin{aligned}
\log p(\beta_2 \mid \beta_1, \mathbf{z}_1, \mathbf{z}_2) &\propto -0.8\log\beta_2 + (2\sum_{i=1}^{n}\mu_i - 0.001)\beta_2 - \frac{s}{2\beta_1^2} \\
&\propto -0.8\log\beta_2 + (2\sum_{i=1}^{n}\mu_i - 0.001)\beta_2 - \frac{1}{2\beta_1^2}\sum_{i=1}^{n} e^{2\mu_i\beta_2}[(z_{1i}-\mu_i)^2 + (z_{2i}-\mu_i)^2]
\end{aligned}$$

$$\frac{\partial \log p(\beta_2 \mid \beta_1, \mathbf{z}_1, \mathbf{z}_2)}{\partial \beta_2} \propto \frac{-0.8}{\beta_2} + (2\sum_{i=1}^{n}\mu_i - 0.001) - \frac{1}{\beta_1^2}\sum_{i=1}^{n} e^{2\mu_i\beta_2} \cdot \mu_i \cdot [(z_{1i}-\mu_i)^2 + (z_{2i}-\mu_i)^2]$$

$$\begin{aligned}
\frac{\partial^2 \log p(\beta_2 \mid \beta_1, \mathbf{z}_1, \mathbf{z}_2)}{\partial {\beta_2}^2} &\propto \frac{0.8}{{\beta_2}^2} - \frac{1}{\beta_1^2}\sum_{i=1}^{n} e^{2\mu_i\beta_2} \cdot 2\mu_i \cdot \mu_i \cdot [(z_{1i}-\mu_i)^2 + (z_{2i}-\mu_i)^2] \\
&\propto \frac{0.8}{{\beta_2}^2} - \frac{2}{\beta_1^2}\sum_{i=1}^{n} e^{2\mu_i\beta_2} \cdot {\mu_i}^2 [(z_{1i}-\mu_i)^2 + (z_{2i}-\mu_i)^2]
\end{aligned}$$

However, it is hard to observe from the conditional posteriors a closed form of any common distributions. If we can prove that the conditional posteriors of β_1 and β_2 are log-concave, then we can apply the adaptive rejection sampling (ARS) algorithm to sample from the posterior means. However, here the log-concave property (second derivative of log (conditional posterior) < 0) only satisfies for specific carefully selected parameters and the data as well. Therefore, here we choose to use WinBUGS, a Bayesian analysis software that can solve very complicated models. Here I use the "R2WinBUGS" library in R to call WinBUGS to obtain the result.