

[별지 제10호서식]

용역연구개발사업최종보고서			
관리 번호	03142약동성537		
사업명	의약품동등성관리		
과제명	국문	SAS를 이용한 생물학적동등성시험결과 통계분석	
	영문	The statistical analysis of bioequivalence studies with SAS	
주관연구기관	기관명	소재지	대표
	이화여자대학교	서울 서대문구 대현동 11-1	신인령
주관연구책임자	성명	소속 및 부서	전공
	강승호	통계학과	생물통계학
연구기간	2003 년 4월 1일 ~ 2003 년 11 월 30일( 8 개월)		
연구비	일금 15,000 천원정 (₩ 15,000,000 )		
참여연구원	2 명 (책임연구원: 1명, 연구원: 0명, 연구보조원: 1명 보조원: 0명)		
<p>2003 년도 용역연구개발사업에 의하여 수행중인 연구과제의 최종보고서를 붙임과 같이 제출합니다.</p> <p>붙임 : 1. 최종보고서 10부 (배포용 25부 별도제출). 2. 최종보고서 수록 디스켓1부 (HWP97).</p> <p style="text-align: right;">2003 년 10 월 30 일</p> <p style="text-align: right;">주관연구책임자 강승호 (인 또는 서명) 주관연구기관장 신인령 (직인)</p> <p>식품의약품안전청장(국립독성연구원장) 귀하</p>			

210mm×297mm

(일반용지60g/㎡(제활용품))

(뒷면)

(측면)

(앞면)

<div data-bbox="159 1171 643 1512"><p>주 의 ( 주 의 내 용 기 재 )  (한글 14 point 고딕체)</p></div>	<p>과 제 명</p> <p>2 0 0 3</p> <p>식 품 의 약 품 안 전 청</p>	<div data-bbox="946 568 1134 645"><p>최종보고서</p></div> <p>국문과제명</p> <p>영문과제명</p> <p>주관연구기관 :</p> <p>식품의약품안전청</p>
--	--	--

※ 주 의 내 용

## 주 의

1. 이 보고서는 식품의약품안전청에서 시행한 용역연구개발사업의 보고서입니다.
2. 이 보고서 내용을 발표할 때에는 반드시 식품의약품안전청에서 시행한 용역연구개발사업의 연구결과임을 밝혀야 합니다.
3. 국가과학기술 기밀유지에 필요한 내용은 대외적으로 발표 또는 공개하여서는 아니 됩니다.

편집순서 2 : 제출문

## 제 출 문

### 식품의약품안전청장 귀하

이 보고서를 “SAS를 이용한 생물학적동등성시험결과 통계분석(이화여자대학교/강승호)”과제의 최종보고서로 제출합니다.

2003. 10. 30

주관연구기관명 : 이화여자대학교

주관연구책임자 : 강승호

편집순서 3 : 요약문

## 연구사업 최종보고서 요약문

연구과제명	SAS를 이용한 생물학적동등성시험결과 통계분석		
중심단어	생물학적 동등성, SAS, 교차실험, 생물통계학		
주관연구기관	이화여자대학교	주관연구책임자	강승호
연구기간	2003. 4.1 - 2003.11.30		
<p>2 x 2 교차시험에 얻어지는 자료에 대한 생물학적 동등성 평가 통계 분석을 위한 SAS 프로그램을 개발하였다. 그 프로그램에는 정규성을 가정하는 모수적 방법뿐만이 아니라, 정규성 가정이 없을 때 사용할 수 있는 비모수적 방법도 포함시켰고, 정규성 가정을 체크할 수 있는 분석 방법도 포함시켰다. 또한 2 x 3 교차시험과 2 x 4 교차시험 자료 분석을 위한 SAS 프로그램을 개발하였다. 개발된 SAS 프로그램을 통계 비전문가도 사용할 수 있도록 사용 방법을 설명한 매뉴얼을 만들었다. SAS 프로그램을 개발한 후 기존의 생물학적 동등성시험 검정 프로그램인 K-BE와 상호비교하였고, 같은 결과를 얻을 수 있음을 확인하였다.</p>			

## Project Summary

Title of Project	The statistical analysis of bioequivalence studies with SAS		
Key Words	bioequivalence, SAS, crossover design		
Institute	Ewha Womans University	Project Leader	Seung-Ho Kang
Project Period	2003. 4.1 - 2003.11.30		
<p>We developed SAS program to analyze the bioequivalence data obtained from 2 x 2 crossover design. The program contains the parametric methods to assume normality, the nonparametric methods not assuming normality as well as the methods to check the normality assumption. We also developed SAS program to analyze the bioequivalence data obtained from 2 x 3 crossover design and 2 x 4 crossover design. We made the manuals so that beginners can use the program easily. We compared the results from SAS programs with those from K-BE program and confirmed that both produced the same results.</p>			

편집순서 4 : 목차 및 본문

목 차

제 1 장 서론 .....	6
제 2 장 국내·외 기술개발 현황 .....	7
제 3 장 연구개발수행 내용 및 결과	
3.1 서론 .....	8
3.2 $2 \times 2$ crossover design	
3.2.1 이론적 배경 .....	11
3.2.2 예제 자료 .....	16
3.2.3 그래프 .....	18
3.2.3.1 각 subject 별로 제제에 따른 AUC값 변화량 그래프	
3.2.3.1-1 프로그램	
3.2.3.2 투여한 약에 따른 각 group 간의 AUC의 평균값 변화량 그래프	
3.2.3.2-1 프로그램	
3.2.3.3 그래프 결과	
3.2.4 PROC MIXED .....	23
3.2.4.1 프로그램	
3.2.4.2 프로그램 설명	
3.2.5 PROC GLM .....	27
3.2.5.1 프로그램	
3.2.5.2 프로그램 설명	
3.2.6 비모수적 방법(Wilcoxon-Mann-Whitney) .....	40
3.2.6.1 Wilcoxon-Mann-Whitney의 two one-sided tests procedure	
3.2.6.2 프로그램	
3.2.6.3 프로그램 설명	
3.2.7 Assumptions and Outliers .....	52
3.2.7.1 정규확률도와 Shpiro-Wilk test	
3.2.7.2 프로그램	
3.2.7.3 프로그램 설명	
3.3 $4 \times 2$ crossover design	
3.3.1 이론적 배경 .....	75
3.3.2 예제 자료 .....	77
3.3.3 carry 변수의 생성 .....	81
3.3.4 잔류효과가 존재하는 경우 .....	83
3.3.4.1 프로그램(PROC MIXED)	
3.3.4.2 프로그램 설명	
3.3.5 잔류효과(carryover effect)가 존재하지 않는 경우 .....	88
3.3.5.1 프로그램(PROC MIXED)	
3.3.5.2 프로그램 설명	
3.4 $2 \times 3$ crossover design	
3.4.1 이론적 배경 .....	93
3.4.2 예제 자료 .....	95
3.4.3 SAS PROGRAM .....	98

3.4.3.1 프로그램(PROC MIXED)	
3.4.3.2 프로그램 설명	
3.5 2 × 4 crossover design	
3.5.1 이론적 배경 .....	103
3.5.2 예제 자료 .....	105
3.5.3 잔류효과(carryover effect)가 존재하는 경우 .....	109
3.5.3.1 프로그램(PROC MIXED)	
3.5.3.2 프로그램 설명	
3.5.4 잔류효과(carryover effect)가 존재하지 않는 경우 .....	114
3.5.4.1 프로그램(PROC MIXED)	
3.5.4.2 프로그램 설명	
3.6 Williams Design with Three Formulations	
3.6.1 이론적 배경 .....	118
3.6.2 예제 자료 .....	119
3.6.3 잔류효과가 존재하는 경우 .....	121
3.6.3.1 프로그램	
3.6.3.2 프로그램 설명	
3.6.4 잔류효과가 존재하지 않는 경우 .....	126
3.6.4.1 프로그램(PROC MIXED)	
3.6.4.2 프로그램 설명	
3.7 Tutorials for 2×2 crossover design	
3.7.1 프로그램 설명 .....	130
3.7.2 프로그램 결과 .....	146
제 4 장 연구개발목표 달성도 및 대외기여도 .....	166
제 5 장 연구개발결과의 활용성과 및 계획 .....	167
제 6 장 참고문헌 .....	168



## 제 1 장 서론

생물학적 동등성을 평가하는데 통계학은 핵심적인 역할을 수행하고 있다. 그러한 통계적 방법을 올바르게 사용하기 위해서는, 사용되는 통계 방법이 가지고 있는 가정들에 대한 이해와 귀무가설 또는 대립가설을 채택하였을 때 발생할 수 있는 오류 확률에 대한 이해가 필수적이다. SAS는 통계 패키지중에서는 그 기능이 가장 방대하여 전 세계적으로 가장 널리 사용되고 있으며, 특히 미국 제약업계에서는 신약의 승인에 필요한 임상실험을 마친 후, 자료를 분석 및 요약하여 미국 식품의약품안전청에 제출할 때, 표준으로 사용되고 있는 통계 패키지이다. 본 연구에서는 한국 식품의약품안전청에서도 그러한 SAS를 도입하여 생물학적 동등성을 평가함에 있어서 통계분석 방법을 향상시킴과 동시에 국내 생물학적 동등성 실험 결과의 국제적 공인을 도모하고자 한다. 이렇게 SAS의 도입으로 인하여 통계분석 방법의 선진화를 이룰 수 있다.

## 제 2 장 국내·외 기술개발 현황

현재 식품의약품안전청장이 생물학적 동등성이 있다고 인정한 품목에 한하여 의사의 사전 동의 없이 대체 조제가 가능해짐에 따라 생물학적 동등성 실험에 대한 수요가 급속히 증가하였다. 또한 의약품의 생물학적 동등성 확보는 의약품 품질에 대한 신뢰를 확보하고, 의약분업의 조기 정착 및 건강보험재정 안정에 기여할 수 있는 있어서, 생물학적 동등성 실험에 대한 중요성은 날로 증대되고 있다. 이러한 생물학적 동등성 실험의 수요 증가 및 중요성 증대에도 불구하고, 생물학적 동등성을 평가하는데 핵심적인 역할을 담당하는 통계학에 대한 적절한 투자가 이루어지지 못하여, 현재 생물학적 동등성 실험 자료 분석에 사용되는 통계 분석 프로그램에는 여러 가지 부족한 점이 많은 실정인으로서, 세계적으로 널리 사용되는 통계 패키지인 SAS의 도입이 절실히 필요한 실정이다.

## 제 3 장 연구개발수행 내용 및 결과

### 3.1 서론

생물학적 동등성 시험이란 동일유효성분을 함유한 동일 투여경로의 두 제제가 생체이용률에 있어서 통계학적으로 동등하다는 것을 입증하기 위해 실시하는 생체 내 시험이다. 생체이용률이란 투여된 의약품 내에 포함된 유효성분량에 대한 전신순환혈에 이행한 약물량의 비율 및 약물의 이행속도를 뜻한다. 생물학적 동등성 시험의 필요성은 다음과 같다.

1. 이미 제조 허가되어 시판 중인 의약품과 동일한 품목의 제조 허가를 신청하고자 할 때
2. 기존 허가품목의 자체 품질 관리를 위해 다른 시판 제제와의 생체이용률이 동일함을 증명하고자 할 때
3. 외국 오리지널 제품과 동일유효성분을 함유한 제품을 개발하여 대체조제 의약품으로 지정받고자 할 때
4. 다국적 제약 기업과의 무한 경쟁 속에 약효와 안전성이 뛰어난 제품만이 생존 가능
5. 생물학적 동등성이 확보된 우수한 제품으로 시장점유율을 확대하는 동시에 기업의 안정적인 성장기반 구축

가장 일반적인 생물학적 동등성 시험방법을 쉽게 설명하면 다음과 같다.

- ① 시험에 참여할 건강한 피험자(被驗者, volunteer)를 모집하여 건강상태 확인  
(시험의 목적 및 시험방법, 이상반응 발생시 피해보상대책 등 시험내용에 대하여 충분히 설명하고 피험자의 자유의사에 의한 시험참가 서면동의서를 받음)
- ② 피험자를 무작위로 두 군으로 나누어 각 군별로 동일 성분의 대조약과 시험약을 각각 투여(제1기 투약)
- ③ 피험자별로 투약후 일정시간마다 혈액 채취  
(예를 들면, 투약후 30분, 1시간, 2시간, 4시간 ... 12시간 등)
- ④ 투여한 의약품이 몸속에 얼마나 흡수되었는지를 확인하기 위하여 채취된 혈액 중 투여한 의약품의 농도를 분석
- ⑤ 이전에 투여한 약이 모두 배설될 정도로 충분한 기간 경과후(일반적으로 1주일) 각 군별로 대조약과 시험약을 바꾸어 각각 투여하고(제2기 투약) 동일하게 혈액 채취 및 혈중농도 분석
- ⑥ 각 피험자별 제1기 및 제2기 투약시 혈중농도를 통계 처리하여 대조약과 시험약의 동등성 평가

여기서는 다음과 같은 통계적 모형을 이용하여 생물학적 동등성 자료를 분석하였다.

시험약과 대조약을 먹은 사람의 피를 각각 뽑아 혈중농도를 비교하는 방법으로 혈중농도 곡선 아래의 면적을 합산하는 AUC를 이용하였다. 이 값이 크다는 것은 약의 흡수력이 좋다는 것이다.

**모형 :**

$Y_{ijk}$  =  $k$  번째 sequence의  $j$  번째 period에서  $k$  번째 sequence의  $i$  번째 subject의 AUC 측정값.

$$Y_{ijk} = \mu + S_{ik} + P_j + F_{j,k} + C_{(j-1,k)} + \varepsilon_{ijk}$$

여기서,

- $\mu$  : 전체 평균
- $S_{ik}$  :  $k$  번째 sequence에서  $i$  번째 subject의 효과(랜덤)
- $P_j$  :  $j$  번째 period의 효과(고정)
- $F_{j,k}$  :  $k$  번째 sequence에서  $j$  번째 period의 제제의 효과(고정)
- $C_{(j-1,k)}$  :  $k$  번째 sequence에서  $(j-1)$  번째 period의 잔류효과(고정)
- $\varepsilon_{ijk}$  : 오차항

이다.

**가정 :**

1.  $S_{ik} \sim N(0, \sigma_s^2)$
2.  $\varepsilon_{ijk} \sim N(0, \sigma_e^2)$
3.  $S_{ik}$  and  $\varepsilon_{ijk}$  : 독립

**가설 ;**

$$H_0: F_R \neq F_T, H_1: F_R = F_T$$

**Classic Confidence Interval :**

만일  $(\mu_T - \mu_R)$ 에 대한  $(1-2\alpha) \times 100\%$  신뢰구간이 FDA에 명시된 채택역(현재 가이드 라인에 의하면  $\ln(0.8)$ ,  $\ln(1.25)$ ) 안에 들어가면 두제제가 생물학적으로 동등하다 결론을 내린다. 여기서  $\mu_T$ 는 시험약에서의 AUC의 모평균값,  $\mu_R$ 은 대조약에서의 AUC의 모평균값이다.

여기서는 2×2를 비롯하여 4×2 , 2×3 , 2×4 그리고 더 나아가서 제제의 종류가 2개 이상인 경우의 Crossover design에 대한 생물학적 동등성 시험을 SAS 통계 프로그램의 PROC MIXED, PROC GLM 등의 절차를 통해서 수행하는 방법을 고찰해보고자 한다.

## 3.2 2 × 2 crossover design

### 3.2.1 이론적 배경

모형:

$$Y_{ijk} = \mu + S_{ik} + P_j + F_{j,k} + C_{(j-1,k)} + \varepsilon_{ijk}$$

여기서,

$\mu$  : 전체 평균

$S_{ik}$  :  $k$  번째 sequence에서  $i$  번째 subject의 효과(랜덤)

$P_j$  :  $j$  번째 period의 효과(고정)

$F_{j,k}$  :  $k$  번째 sequence에서  $j$  번째 period의 제제의 효과(고정)

$C_{(j-1,k)}$  :  $k$  번째 sequence에서  $(j-1)$  번째 period의 잔류효과(고정)

$\varepsilon_{ijk}$  : 오차항

이다.

T : test formulation, 시험약, copy.

R : reference formulation, 대조약, original.

$$\text{Sequence 1 : } Y_{i11} = \mu + S_{i1} + P_1 + F_R + \varepsilon_{i11}$$

$$Y_{i21} = \mu + S_{i1} + P_2 + F_T + C_R + \varepsilon_{i21}$$

$$\text{Sequence 2 : } Y_{i12} = \mu + S_{i2} + P_1 + F_T + \varepsilon_{i12}$$

$$Y_{i22} = \mu + S_{i2} + P_2 + F_R + C_T + \varepsilon_{i22}$$

$$\text{where } P_1 + P_2 = 0, \quad F_R + F_T = 0, \quad C_R + C_T = 0$$

sequence	period 1	washout period	period 2
1(RT)	R( $Y_{i11}$ )		T( $Y_{i21}$ )
2(TR)	T( $Y_{i12}$ )		R( $Y_{i22}$ )

### 3.2.1.1 Two one-sided test procedure

$$H_0: \mu_T - \mu_R \leq \Theta_L \quad \text{or} \quad \mu_T - \mu_R \geq \Theta_U$$

$$\text{vs } H_a: \Theta_L < \mu_T - \mu_R < \Theta_U$$

여기서  $\Theta_L$ ,  $\Theta_U$ 는 각각 하한, 상한 한계값이고,  $\mu_T$ 는 시험약에서의 AUC 값의 모평균값,  $\mu_R$ 은 대조약에서의 AUC 값의 모평균값이다. 귀무가설  $H_0$ 가 기각되면, 두 제제는 생물학적으로 동등하다고 결론을 내린다.

위의 가설은 단측 가설 2가지로 분리할 수 있다.

$$H_{01}: \mu_T - \mu_R \leq \Theta_L \quad H_{a1}: \mu_T - \mu_R > \Theta_L$$

and

$$H_{02}: \mu_T - \mu_R \geq \Theta_U \quad H_{a2}: \mu_T - \mu_R < \Theta_U$$

만일  $H_{01}$  와  $H_{02}$ 를 기각하면,  $\Theta_L < \mu_T - \mu_R < \Theta_U$  라고 결론내릴 수 있다. 즉, 두 제제가 생물학적으로 동등하다.

만일

$$T_L = \frac{(\overline{Y_T} - \overline{Y_R}) - \Theta_L}{\hat{\sigma}_d \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} > t_{(\alpha, n_1 + n_2 - 2)}$$

이고

$$T_U = \frac{(\overline{Y_T} - \overline{Y_R}) - \Theta_U}{\hat{\sigma}_d \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} > -t_{(\alpha, n_1 + n_2 - 2)}$$

이면 생물학적으로 동등하다고 결론내릴 수 있다.

여기서  $\overline{Y_T}$  는 시험약의 AUC의 실제 평균값,  $\overline{Y_R}$ 은 대조약의 AUC의 실제 평균값이고, 식으로 나타내면,

$$\hat{\sigma}_d^2 = \frac{1}{n_1 + n_2 - 2} \sum_{k=1}^2 \sum_{i=1}^{n_k} (d_{ik} - \overline{d_{.k}})^2 \quad \text{이다. } d_{ik} \text{ 는 두 period간의 AUC 관측값의 차이를 구한 값이다.}$$

즉, 식으로 나타내면  $d_{ik} = \frac{1}{2}(Y_{i2k} - Y_{i1k})$ ,  $i = 1, \dots, p, k = 1, 2$  이다.  $t(\alpha, n_1 + n_2 - 2)$  는 자유도가  $(n_1 + n_2 - 2)$  인 t 분포에서 유의수준이  $\alpha$  일 때의 값을 뜻한다.

### 3.2.1.2 Confidence interval approach

**90% Confidence Interval** for  $\mu_T - \mu_R$  은 다음과 같이 주어진다.

$$(\bar{Y}_T - \bar{Y}_R) \pm t(\alpha, n_1 + n_2 - 2) \hat{\sigma}_d \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}$$

위 신뢰구간이  $(\theta_L, \theta_U)$  에 포함되면, 두 제제는 생물학적으로 동등하다고 결론내릴 수 있다. 신뢰구간을 이용하는 방법은 2.1.1 의 two one-sided test procedure 와 동일하다.

### 3.2.1.3 ANOVA

잔류효과와 기간효과에 대한 검정은 ANOVA를 통하여 할 수 있다.

$\bar{Y}_{\dots}$  = 모든 관측치의 총평균

$$\begin{aligned} SS_{total} &= \sum_{k=1}^2 \sum_{j=1}^2 \sum_{i=1}^{n_k} (Y_{ijk} - \bar{Y}_{\dots})^2 \\ &\quad \uparrow \\ &\neq \bar{Y}_{i.k} = \frac{1}{2} (Y_{i1k} + Y_{i2k}) \\ &= \sum_{k=1}^2 \sum_{j=1}^2 \sum_{i=1}^{n_k} (Y_{ijk} - \bar{Y}_{i.k})^2 + 2 \sum_{k=1}^2 \sum_{i=1}^{n_k} (\bar{Y}_{i.k} - \bar{Y}_{\dots})^2 \\ &= SS_{within} + SS_{between} \end{aligned}$$

	df
$SS_{total}$	$2(n_1 + n_2) - 1$
$SS_{within}$	$n_1 + n_2 - 1$
$SS_{between}$	$n_1 + n_2$



$$\circ \quad SS_{between} = SS_{carry} + SS_{inter}$$

$$SS_{carry} = \frac{2n_1n_2}{n_1+n_2} \{ (\overline{Y_{.12}} + \overline{Y_{.22}}) - (\overline{Y_{.11}} + \overline{Y_{.21}}) \}^2$$

$$SS_{inter} = \sum_{k=1}^2 \sum_{i=1}^{n_k} \frac{Y_{i.k}^2}{2} - \sum_{k=1}^2 \frac{Y_{..k}^2}{2n_k}$$

$$E(MS_{carry}) = E\left(\frac{SS_{carry}}{1}\right) = \frac{2n_1n_2}{n_1+n_2} (C_T - C_R)^2 + 2\sigma_s^2 + \sigma_e^2$$

$$E(MS_{inter}) = E(SS_{inter}/(n_1+n_2-2)) = 2\sigma_s^2 + 2\sigma_e^2$$

◦ 잔류 효과 검정

$$H_0: G = C_R \quad \text{vs} \quad H_1: G \neq C_R$$

$F_c > F(\alpha, 1, n_1+n_2-2)$  이면 귀무가설  $H_0$  를 기각한다.

$$\text{여기서 } F_c = MS_{carry} / MS_{inter}$$

$$\circ \quad SS_{within} = SS_{drug} + SS_{period} + SS_{intra}$$

◦ 제제 효과 검정

$$: F = \frac{MS_{drug}}{MS_{intra}}$$

◦ 기간 효과 검정

$$: F = \frac{MS_{period}}{MS_{intra}}$$

### 3.2.2 예제자료

여기서 이용된 자료는 Chow & Liu(2000) p73의 자료이다. 24명을 대상으로 실험약과 대조약의 생체이용률 비교를 위한 실험을 하였다. 투여기간동안 각각의 사람들은 50-mg 의 알약(실험약) 또는 5ml 의 경구약(대조약)을 투여받았다. 그리고 약 투여전과 약 투여후 다양한 시간대에 혈액을 채취하고 그에 따른 AUC값을 trapezoidal 방법을 이용하여 계산하였다.

자료는 엑셀에서 입력한 후 다음과 같이 쉼표로 구분되도록 .csv로 저장하였다.  
자료 입력 순서는 group, subject, formulation, y(AUC), period 의 순서이다.

```
1,1,1,74.675,1
1,4,1,96.4,1
1,5,1,101.95,1
1,6,1,79.05,1
1,11,1,79.05,1
1,12,1,85.95,1
1,15,1,69.725,1
1,16,1,86.275,1
1,19,1,112.675,1
1,20,1,99.525,1
1,23,1,89.425,1
1,24,1,55.175,1
1,1,2,73.675,2
1,4,2,93.25,2
1,5,2,102.125,2
1,6,2,69.45,2
1,11,2,69.025,2
1,12,2,68.7,2
1,15,2,59.425,2
1,16,2,76.125,2
1,19,2,114.875,2
1,20,2,116.25,2
1,23,2,64.175,2
1,24,2,74.575,2
2,2,2,74.825,1
2,3,2,86.875,1
2,7,2,81.675,1
2,8,2,92.7,1
2,9,2,50.45,1
```

2,10,2,66.125,1  
 2,13,2,122.45,1  
 2,14,2,99.075,1  
 2,17,2,86.35,1  
 2,18,2,49.925,1  
 2,21,2,42.7,1  
 2,22,2,91.725,1  
 2,2,1,37.35,2  
 2,3,1,51.925,2  
 2,7,1,72.175,2  
 2,8,1,77.5,2  
 2,9,1,71.875,2  
 2,10,1,94.025,2  
 2,13,1,124.975,2  
 2,14,1,85.225,2  
 2,17,1,95.925,2  
 2,18,1,67.1,2  
 2,21,1,59.425,2  
 2,22,1,114.05,2

**자료(sas) :** 좌측이 sas 변수명이다.

y : 대조약, 시험약에 따른 AUC

group : sequence (1,2)

tmt : formulation ( 1=reference, 2=test )

period (1,2)

subject (n1=n2=12 )

group은 1,2 두 개의 그룹으로 subject 는 각각 한 그룹 당 12명씩 배치, tmt는 대조약을 1로, 시험약을 2로 y값은 AUC 값을, period값은 1,2로 지정한다.

앞으로 이 절에서 사용되는 자료는 group,subject,tmt,y,period(sas 입력 data 명) 순으로 .csv 형태로 c의 work 폴더에 one이라는 이름으로 저장되어 있다하고,(즉, c:/work/one.csv) infile문을 통해서 자료를 불러들일 것이다.

위와 같이 입력된 데이터로 다음과 같은 프로그램들을 통해서 그래프와 신뢰구간 등의 다양한 결과를 얻을 수 있다.

### 3.2.3 그래프

통계적 분석을 하기 이전에 자료를 한눈에 파악하기 위한 가장 좋은 방법은 그래프를 그려보는 것이다. 확인한 차이가 나는 경우는 그래프를 통해 쉽게 확인하고 결과를 예측해볼 수 있다.

#### 3.2.3.1 각 subject 별로 제제에 따른 AUC값 변화량 그래프

##### 3.2.3.1-1 프로그램

각 subject 별로 제제에 따른 AUC값 변화량 그래프를 그리기 위한 프로그램은 다음과 같고, 프로그램의 결과는 3.2.3.3에 제시하였다. 자료는 3.2.2에서 언급한  $2 \times 2$  자료를 이용하였다.

```
data one;
  infile 'c:\work\Wone.csv' delimiter=';';
  input group subject tmt y period;
label y='Area Under Curve' tmt='Formulation';
run;
proc sort data=one;
  by group subject tmt;

footnote 'Formulation 1=R 2=T';
symbol i=join v=x r=50 c=black ;
axis1 label=(angle=90 h=1 f=zapfb);
axis2 label=(angle=90 h=1 f=zapfb) order=(70 to 90 by 5);
axis3 label=(f=zapfb) length=70 offset=(20,20) order=(1 to 2 by 1) ;
axis4 label=(f=zapfb) length=70 offset=(20,20) order=(1 to 2 by 1) ;

proc gplot data=one;
plot y*tmt=subject/description='y' vaxis=axis1 haxis=axis3 nolegend;
by group;run;quit;
```

#### 3.2.3.2 투여한 약에 따른 각 group 간의 AUC의 평균값 변화량 그래프

##### 3.2.3.2-1 프로그램

투여한 약에 따른 각 group 간의 AUC의 평균값 변화량 그래프를 그리기 위한 프로그램은 다음과 같고, 프로그램의 결과는 3.2.3.3에 제시하였다. 여기서 set 명령문으로 불러들인 one이라는 dataset은 바로 앞

3.2.3.1에서 생성된 dataset을 가져온 것이다. 즉, 3.2.3.1 의 프로그램 뒤에 이어서 아래 프로그램을 넣어주면 된다. 3.2.3.1의 프로그램 부분을 넣지않으면 결과를 얻을 수 없다.

```
data one_1;
set one;
if period=1 ;
if tmt=1;
id=1;
proc means data=one_1 noprint;
var y;
output out=out_1 mean=my_1;
data mean1;
set out_1;
id=1;
tmt=1;period=1;
y_m=my_1;
keep y_m tmt period id;
label y_m='Mean Area Under Curve';
data one_2;
set one;
if period=1 ;
if tmt=2;
id=1;
proc means data=one_2 noprint;
var y;
output out=out_2 mean=my_2;
data mean2;
set out_2;
id=1;
tmt=2;period=1;
y_m=my_2;
keep y_m tmt period id;
data one_3;
set one;
if period=2 ;
if tmt=1;
id=1;
proc means data=one_3 noprint;
var y;
output out=out_3 mean=my_3;
data mean3;
set out_3;
id=1;
tmt=1;period=2;
y_m=my_3;
keep y_m tmt period id;
data one_4;
```

```

set one;
if period=2 ;
if tmt=2;
id=1;
proc means data=one_4 noprint;
var y;
output out=out_4 mean=my_4;
data mean4;
set out_4;
id=1;
tmt=2;period=2;
y_m=my_4;
keep y_m tmt period id;
data mean;
  merge mean1 mean2 mean3 mean4;
  by id tmt period;

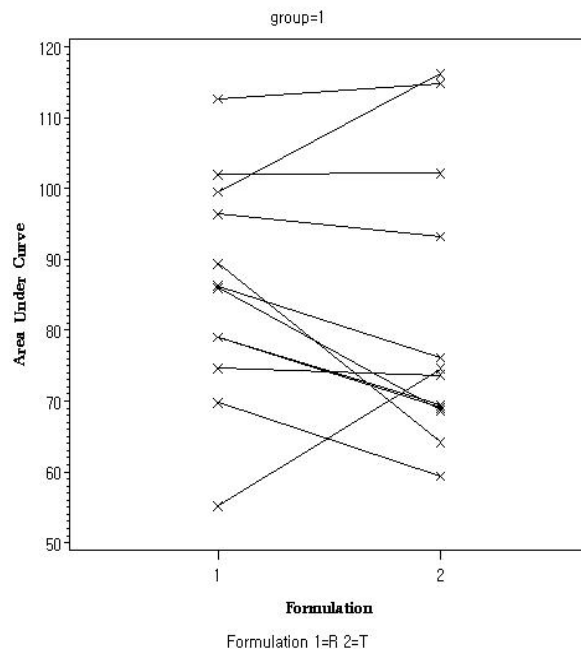
length function text $8;
XSYS = '2'; YSYS = '2';
TEXT = tmt; SIZE =1;
if tmt=1 and period=1 then do;
y = y_m; x = period; function ='MOVE'; output; end;
if tmt=2 and period=1 then do;
y = y_m; x = period; function ='MOVE'; output; end;
y = y_m; x = period; function='DRAW'; output;
y = y_m; x = period; function='LABEL'; output;
proc print; run;

proc gplot data=mean;
plot y_m*period=tmt/annotate=mean vaxis=axis2 haxis=axis4 nolegend;
run;quit;

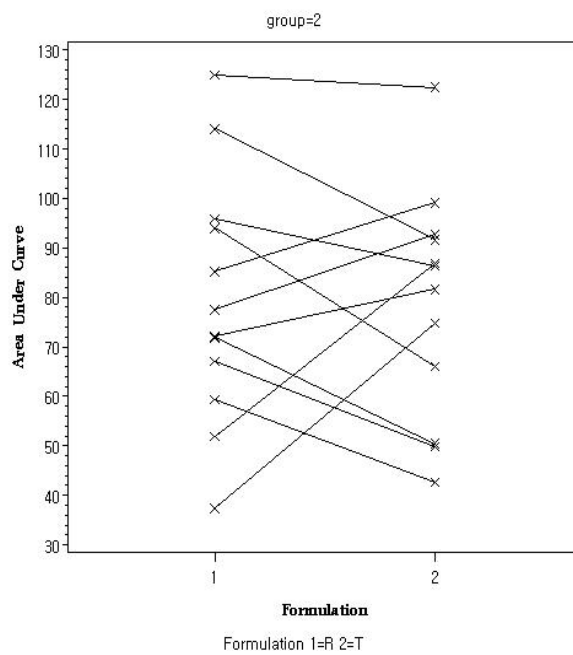
```

### 3.2.3.3 그래프 결과

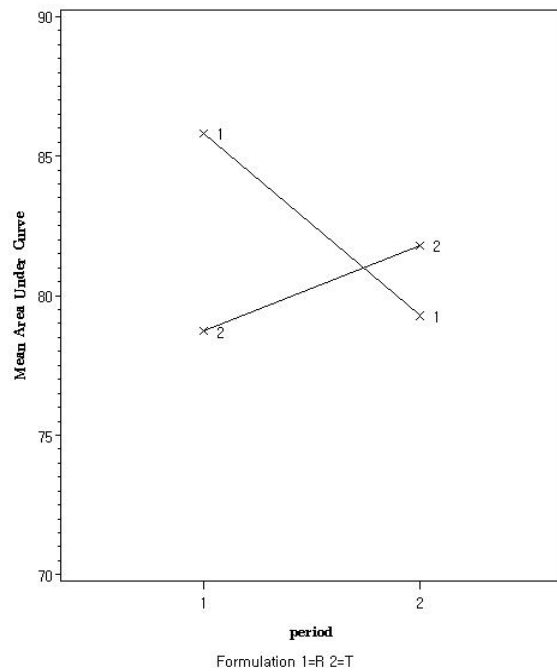
이 그래프는 (Chow & Liu, 2000) p 74의 그래프와 일치한다.



이 그래프는 (Chow & Liu, 2000) p 75의 그래프와 일치한다.



이 그래프는 (Chow & Liu, 2000) p 76의 그래프와 일치한다.



그래프 상에서 숫자 1은 대조약을 숫자 2는 시험약을 뜻한다. 따라서, 투여한 약에 따른 group간의 AUC 평균 값을 비교할 수 있다.



## 3.2.4 PROC MIXED

### 3.2.4.1 프로그램

앞의 3.2.2에서 언급한  $2 \times 2$  자료를 이용하여 PROC MIXED를 이용하여 분석하는 SAS 프로그램은 다음과 같다.

```
Libname out 'c:\work';
data one;
  infile 'c:\work\one.csv' delimiter=';';
  input group subject tmt y period;
run;

proc sort data=one;
  by group subject tmt;

proc mixed data=one;
class tmt period subject group;
model y=group period tmt;
random subject(group);
lsmeans tmt/pdiff cl alpha=0.1;
estimate 'T vs. R' tmt -1 1/cl alpha=0.1;run;
```

### 3.2.4.2 프로그램 설명

```
Libname out 'c:\work';
data one;
  infile 'c:\work\one.csv' delimiter=';';
  input group subject tmt y period;
run;
proc sort data=one;
  by group subject tmt;
```

infile문 아래 자료가 저장되어 있는 경로를 지정해 자료를 읽어들이고, 여기서 csv파일은 ';'로 구분되어 있으므로 구분자로 ';'를 쓴다. 순서대로 group, subject, tmt, y, period로 읽어들이어 이 변수들을 포함하는 one이라는 이름의 dataset이 생성된다. 생성된 dataset one을 group, subject, tmt 순으로 정렬시킨다.

```
proc mixed data=one;
class tmt period subject group;
model y=group period tmt;
random subject(group);
lsmeans tmt/cl alpha=0.1;
```

```
estimate 'T vs. R' tmt -1 1/cl alpha=0.1;run;
```

dataset one에 proc mixed 절차를 적용시키고 tmt,period,subject,group 변수를 class문에 지정해주고, model문에 고정변수 group,period, tmt 변수를 랜덤변수 subject를 random 문에 지정해준다. lsmeans는 이들의 최소제곱평균을 구해주고, 신뢰구간을 유의수준 0.1에서 구해준다.

estimate문을 이용해 tmt 간의 최소제곱평균의 차이와 그에 대한 신뢰구간을 유의수준 0.1에서 구해준다. 위의 프로그램에 따른 출력결과는 다음과 같다.

#### The Mixed Procedure

##### Model Information

Data Set	WORK.ONE
Dependent Variable	y
Covariance Structure	Variance Components
Estimation Method	REML
Residual Variance Method	Profile
Fixed Effects SE Method	Model-Based
Degrees of Freedom Method	Containment

##### Class Level Information

Class	Levels	Values
tmt	2	1 2
period	2	1 2
subject	24	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
group	2	1 2

##### Dimensions

Covariance Parameters	2
Columns in X	7
Columns in Z	24
Subjects	1
Max Obs Per Subject	48
Observations Used	48
Observations Not Used	0
Total Observations	48

##### Iteration History

Iteration	Evaluations	-2 Res Log Like	Criterion
-----------	-------------	-----------------	-----------

0	1	405.20098358	
1	1	394.07438982	0.00000000

Convergence criteria met.

#### Covariance Parameter Estimates

Cov Parm	Estimate
subject(group)	284.82
Residual	167.25

#### Fit Statistics

-2 Res Log Likelihood	394.1
AIC (smaller is better)	398.1
AICC (smaller is better)	398.4
BIC (smaller is better)	400.4

#### The Mixed Procedure

#### Type 3 Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
group	1	22	0.37	0.5468
period	1	22	0.22	0.6474
tmt	1	22	0.38	0.5463

#### Estimates

Label	Estimate	Standard Error	DF	t Value	Pr >  t	Alpha	Lower	Upper
T vs. R	-2.2875	3.7333	22	-0.61	0.5463	0.1	<b>-8.6980</b>	<b>4.1230</b>

Estimates 부분의 Lower 와 Upper 에 제시된 값이 (  $\mu_T$   $\mu_R$  ) 에 대한 90% 신뢰구간의 하한, 상한 값이다. 즉, 위의 결과에서 보면 (  $\mu_T$   $\mu_R$  ) 에 대한 90% 신뢰구간 값은 ( -8.6980 , 4.1230 )이다. 이 신뢰구간 값은 (Chow & Liu, 2000) p 106의 결과와 일치한다.

# Least Squares Means

Effect	tmt	Estimate	Standard Error	DF	t Value	Pr >  t	Alpha	Lower	Upper
tmt	1	82.5594	4.3401	22	19.02	<.0001	0.1	75.1069	90.0119
tmt	2	80.2719	4.3401	22	18.50	<.0001	0.1	72.8194	87.7244

# Differences of Least Squares Means

Effect	tmt	_tmt	Estimate	Standard Error	DF	t Value	Pr >  t	Alpha	Lower	Upper
tmt	1	2	2.2875	3.7333	22	0.61	0.5463	0.1	-4.1230	8.6980

### 3.2.5 PROC GLM

PROC GLM 방법에 비해 PROC MIXED 방법이 간편하고  $2 \times 3$  나  $4 \times 2$  로 확장될 경우에도 적용이 가능해 더 유용하지만, 비모수적 방법에서는 PROC GLM 부분이 사용된다.

#### 3.2.5.1 프로그램

앞의 3.2.2에서 언급한  $2 \times 2$  자료를 이용하여 PROC GLM을 이용하여 분석하는 SAS 프로그램은 다음과 같다.

```
Libname out 'c:\work';  
data one;  
  infile 'c:\work\one.csv' delimiter=',';  
  input group subject tmt y period;  
run;
```

```
proc sort data=one;  
  by group subject tmt;
```

```
proc means n noprint data=one;  
  by group;  
  var y;  
  output out=no n=n;  
run;
```

```
data no;  
  set no;  
  n=n/2;  
run;
```

```
proc sort data=no;  
  by group;  
data no1;  
  set no;  
  if group=1;
```

```

        n1=n;
        id=1;
        keep n1 id;

proc sort data=no1;
    by id;
run;

data no2;
    set no;
    if group=2;
    n2=n;
    id=1;
    keep n2 id;

proc sort data=no2;
    by id;
run;

proc glm data=one outstat=tanova;
    class tmt period subject group;
    model y=group subject(group) period tmt;
        test h=group e=subject(group);
    lsmeans tmt /stderr pdiff out=lsmean1;
run;

data error;
    set tanova;
    if _SOURCE_='ERROR';
    mse=ss/df;
    sse=ss;
    id=1;
    keep id mse sse df;

```

```
proc sort data=error;
    by id;
run;
```

```
data lsmean11;
    set lsmean1;
    if tmt=1;
    my1=lsmean;
    id=1;
    keep id my1;
```

```
proc sort data=lsmean11;
    by id;
```

```
data lsmean12;
    set lsmean1;
    if tmt=2;
    my2=lsmean;
    id=1;
    keep id my2;
```

```
proc sort data=lsmean12;
    by id;run;
```

```
data twoside;
    merge lsmean11 lsmean12 no1 no2 error;
    by id;
    spool=mse*0.5*((1/n1)+(1/n2));
    se=sqrt(spool);
    diff=my2-my1;
    t=diff/se;
    ct=tinv(0.05,df,0);
    absct=abs(ct);
    lower=diff-absct*se;
    upper=diff+absct*se;
    plower=((lower/my1)+1)*100;
```

```
pupper=((upper/my1)+1)*100;
p=2*(1-probt(abs(t),df));
run;
```

```
proc print data=twoside;run;
```

### 3.2.5.2 프로그램 설명

#### ▶ dataset : one

##### ▷ program

```
data one;
infile 'c:\work\Wone.csv' delimiter=';';
input group subject tmt y period;
run;
proc sort data=one;
by group subject tmt;
```

infile문 아래 자료가 저장되어 있는 경로를 지정해 자료를 읽어들이고, 여기서 csv파일은 ‘;’로 구분되어 있으므로 구분자로 ‘;’를 쓴다. 순서대로 group, subject, tmt, y, period로 읽어들이어 이 변수들을 포함하는 one이라는 이름의 dataset이 생성된다. 생성된 data set one을 group, subject, tmt 순으로 정렬시킨다. data set one의 결과는 다음과 같다.

##### ▷ dataset one

Obs	group	subject	tmt	y	period
1	1	1	1	74.675	1
2	1	1	2	73.675	2
3	1	4	1	96.400	1
4	1	4	2	93.250	2
5	1	5	1	101.950	1
6	1	5	2	102.125	2
7	1	6	1	79.050	1
8	1	6	2	69.450	2



9	1	11	1	79.050	1
10	1	11	2	69.025	2
11	1	12	1	85.950	1
12	1	12	2	68.700	2
13	1	15	1	69.725	1
14	1	15	2	59.425	2
15	1	16	1	86.275	1
16	1	16	2	76.125	2
17	1	19	1	112.675	1
18	1	19	2	114.875	2
19	1	20	1	99.525	1
20	1	20	2	116.250	2
21	1	23	1	89.425	1
22	1	23	2	64.175	2
23	1	24	1	55.175	1
24	1	24	2	74.575	2
25	2	2	1	37.350	2
26	2	2	2	74.825	1
27	2	3	1	51.925	2
28	2	3	2	86.875	1
29	2	7	1	72.175	2
30	2	7	2	81.675	1
31	2	8	1	77.500	2
32	2	8	2	92.700	1
33	2	9	1	71.875	2
34	2	9	2	50.450	1
35	2	10	1	94.025	2
36	2	10	2	66.125	1
37	2	13	1	124.975	2
38	2	13	2	122.450	1
39	2	14	1	85.225	2
40	2	14	2	99.075	1
41	2	17	1	95.925	2
42	2	17	2	86.350	1
43	2	18	1	67.100	2
44	2	18	2	49.925	1
45	2	21	1	59.425	2
46	2	21	2	42.700	1
47	2	22	1	114.050	2
48	2	22	2	91.725	1

► dataset : no

▷ program

```
proc means n noprint data=one;
    by group;
    var y;
    output out=no n=n;
run;
```

dataset one의 기초통계량들을 구한다. 여기서 n은 계산에 사용된 관찰치의 개수를 출력하고, noprint는 기초통계량을 output 창에 출력하지 않게한다. 이 옵션을 쓰면 출력dataset이 명시되어야 한다. group 변수의 값을 기준으로 변수 y값을 분석한다. n값이 포함된 no라는 이름의 dataset을 생성한다. dataset no에는 그룹에 따른 n값이 생성된다. dataset no의 결과는 다음과 같다.

▷ dataset no

Obs	group	_TYPE_	_FREQ_	n
1	1	0	24	24
2	2	0	24	24

▷ program

```
data no;
    set no;
    n=n/2;
run;
proc sort data=no;
    by group;
run;
```

각각의 그룹별 관찰치의 개수가 n으로 들어갔는데 이 n의 값은 각 그룹의 피험자의 수의 2배이므로(reference,test 두가지 투여에 따른 y값이 각각 다른 관측치로 들어간다.) 각 그룹별 피험자의 수를 n으로 지정해주기 위해서 2로 나누어 준 값을 n으로 대체한다.  
dataset no를 그룹변수에 의해 정렬한다.dataset no의 결과는 다음과 같다.

▷ dataset no

Obs	group	_TYPE_	_FREQ_	n
1	1	0	24	12
2	2	0	24	12

► dataset : no1, no2

▷ program

```
data no1;
    set no;
    if group=1;
    n1=n;
    id=1;
    keep n1 id;

proc sort data=no1;
    by id;

run;
```

dataset no에서 group이 1인 obs들만으로 새로운 dataset no1을 만든다. 이때, no에서의 n을 n1이라는 새로운 변수로 만들고, 1값을 갖는 id라는 새로운 변수를 만든다. 변수 n1과 id만을 남기고, id에 의해 정렬시킨다.

▷ program

```
data no2;
    set no;
    if group=2;
    n2=n;
    id=1;
```

```
keep n2 id;
```

```
proc sort data=no2;
```

```
by id;
```

```
run;
```

dataset no1과 같은 방법으로 no2를 만든다.

dataset no1과 no2의 결과는 다음과 같다.

▷ dataset no1과 no2

Obs	n1	id
-----	----	----

1	12	1
---	----	---

Obs	n2	id
-----	----	----

1	12	1
---	----	---

## ▶ dataset : tanova, lsmean1

▷ program

```
proc glm data=one outstat=tanova;
```

```
class tmt period subject group;
```

```
model y=group subject(group) period tmt;
```

```
test h=group e=subject(group);
```

```
lsmeans tmt /stderr out=lsmean1;
```

```
run;
```

proc glm을 사용하여 dataset one을 분산분석한다. 이때 결과값을 tanova로 저장한다.

분산분석하고자 하는 인자들을 class문으로 지정한다. model문으로 분석하고자하는 모형을 정하고,

test문으로 carryover effect를 검정할 수 있도록 한다. lsmeans문으로 tmt의 수준들에 대한 최소제곱평균을

계산하여 tmt의 수준간 차이검정을 할 수 있도록 한다. lsmeans의 결과값은 lsmean1으로 저장한다. dataset tanova와 lsmean1의 결과는 다음과 같다.

▷ dataset tanova

Obs	_NAME_	_SOURCE_	_TYPE_	DF	SS	F	PROB
1	y	ERROR	ERROR	22	3679.43	.	.
2	y	group	SS1	1	276.00	1.65026	0.21229
3	y	subject(group)	SS1	22	16211.49	4.40598	0.00048
4	y	period	SS1	1	35.97	0.21505	0.64739
5	y	tmt	SS1	1	62.79	0.37544	0.54633
6	y	group	SS3	1	276.00	1.65026	0.21229
7	y	subject(group)	SS3	22	16211.49	4.40598	0.00048
8	y	period	SS3	1	35.97	0.21505	0.64739
9	y	tmt	SS3	1	62.79	0.37544	0.54633

▷ dataset lsmean1

Obs	_NAME_	tmt	LSMEAN	STDERR
1	y	1	82.5594	2.63981
2	y	2	80.2719	2.63981

► dataset : error

▷ program

```
data error;
    set tanova;
    if _SOURCE_='ERROR';
    mse=ss/df;
```

```

sse=ss;
id=1;
keep id mse sse df;
proc sort data=error;
by id;
run;

```

dataset tanova에서 \_SOURCE\_가 ERROR인 ss값을 sse로 지정해주어 sse라는 새로운 변수로 만들고, sse를 df로 나누어 mse를 만들어 새로운 dataset error를 만든다. 또한 1값을 갖는 id라는 변수를 만든다. 여기서 id, mse, sse와 df 변수만을 남기고 나머지 모든 변수를 drop한 후, dataset error를 id변수에 의해 정렬시킨다. dataset error의 결과는 다음과 같다.

▷ dataset error

Obs	DF	mse	sse	id
1	22	167.247	3679.43	1

## ▶ dataset : lsmean11,lsmean12

▷ program

```

data lsmean11;
set lsmean1;
if tmt=1;
my1=lsmean;
id=1;
keep id my1;
proc sort data=lsmean11;
by id;

```

dataset lsmean1에서 변수 tmt값이 1을 갖는 obs만으로(즉,reference formulation인 경우의 관측치 만으로) 새로운 dataset lsmean11을 만든다. 이때 lsmean값을 my1이라는 새로운 변수에 저장하고, id와 my변수만을

남기고 다른 변수는 모두 drop한 후, 변수 id에 의해 정렬시킨다.

▷ program

```
data lsmean12;
    set lsmean1;
    if tmt=2;
    my2=lsmean;
    id=1;
    keep id my2;
    proc sort data=lsmean12;
    by id;
run;
```

lsmean11과 같은 방법으로 dataset lsmean12을 생성한다.

dataset lsmean11과 dataset lsmean12의 결과는 다음과 같다.

▷ dataset lsmean11과 dataset lsmean12

Obs	my1	id
-----	-----	----

1	82.5594	1
---	---------	---

Obs	my2	id
-----	-----	----

1	80.2719	1
---	---------	---

► dataset : twoside

▷ program

```
data twoside;
    merge lsmean11 lsmean12 no1 no2 error;
```

```

by id;
spool=mse*0.5*((1/n1)+(1/n2));
se=sqrt(spool);
diff=my2-my1;
t=diff/se;
ct=tinv(0.05,df,0);
absct=abs(ct);
lower=diff-absct*se;
upper=diff+absct*se;
plower=((lower/my1)+1)*100;
pupper=((upper/my1)+1)*100;
p=2*(1-probt(abs(t),df));
run;

```

이제 최종적으로 Classic Confidence Interval을 만들기위한 dataset twoside를 만든다.

먼저 만들어놓은 dataset lsmean11, lsmean12, no1, no2 그리고 error을 id에 의해 merge 시킨다. 또한 interval을 구하기위한 새로운 변수들을 생성해낸다.

## ► Classic Confidence Interval

▷ program

```

proc print data=twoside;
run;

```

앞에서 만든 dataset twoside를 출력하면, 아래와 같이 Classic Confidence Interval의 계산과정의 값들과, 상한과 하한 값이 계산되어 나온다.

▷ dataset twoside

Obs	my1	id	my2	n1	n2	DF	mse	sse	spool	se
1	82.5594	1	80.2719	12	12	22	167.247	3679.43	13.9372	3.73326



diff	t	ct	absct	lower	upper	plower	pupper	p
-2.2875	-0.61274	-1.71714	1.71714	<b>-8.69805</b>	<b>4.12305</b>	89.4645	104.994	0.54633

이 신뢰구간 값은 (Chow & Liu, 2000) p 106의 결과와 일치한다. 앞서 3.2.4에서 언급한 PROC MIXED 결과와도 일치한다.

### 3.2.6 비모수적 방법(Wilcoxon-Mann-Whitney)

이전에 살펴본 방법들은 앞서 언급되었던  $S_{ik}$ 이 평균이 0이고 분산이  $\sigma_s$ 인 정규분포를 따르고, 오차항  $\varepsilon_{ijk}$ 이 평균이 0이고 분산이  $\sigma_e$ 인 정규분포를 따르며  $S_{ik}$ 와  $\varepsilon_{ijk}$ 이 서로 독립이라는 가정을 만족하는 경우에 성립한다. 만일 이 정규성의 가정이 위배될 때는 앞의 방법들은 적절하지 않을 것이다. 이런 경우에는 분포와 상관없는 비모수적 방법을 고려해주어야한다. 이 정규성가정에 대한 검정법은 2.7 절에서 논의할 것이다. 여기서는 여러 가지 비모수적 방법 중 Wilcoxon-Mann-Whitney 방법을 이용하겠다.

#### 3.2.6.1 Wilcoxon-Mann-Whitney의 two one-sided tests procedure

$$H_{01} : \mu_T - \mu_R \leq \theta_L \quad H_{a1s} : \mu_T - \mu_R > \theta_L$$

$$\text{이고 } H_{02} : \mu_T - \mu_R \geq \theta_U \quad H_{a2s} : \mu_T - \mu_R < \theta_U$$

위의 식을  $\theta = \mu_T - \mu_R$   $\theta_L^* = \theta - \theta_L$   $\theta_U^* = \theta - \theta_U$  로 두고 다시

표현하면,  $H_{01} : \theta_L^* \leq 0$   $H_{a1} : \theta_L^* > 0$ 와  $H_{02} : \theta_U^* \geq 0$   $H_{a2} : \theta_U^* < 0$ 로 표현할 수 있다.

$$d_{ik} = \frac{1}{2} (Y_{2k} - Y_{1k}) \quad j = 1, \dots, k; k=1, 2$$

$$\text{Let } b_{hik} = \begin{cases} d_{ik} - \theta_h & , \text{ for } h=L, U \text{ for subjects in sequence1} \\ d_{ik} & , \text{ for subjects in sequence2} \end{cases}$$

잔류효과가 없다고 가정하면, 다음과 같이 표현할 수 있다.

$$E[b_{hi1}] = \frac{1}{2} [E(Y_{i21}) - E(Y_{i11})] - \theta_h$$

$$= \frac{1}{2} [(\mu + P_2 F_T) - (\mu + P_1 F_R)] - \theta_h$$

$$= \frac{1}{2} [(P_2 - P_1) + (\theta - 2\theta_h)]$$

$$E[b_{hi2}] = \frac{1}{2} [(P_2 - P_1) - \theta]$$

$$Var(b_{ik}) = Var(d_{ik}) \equiv \sigma_d^2 = \frac{1}{2} \sigma_e^2$$

$$E(b_{h1}) - E(b_{h2}) = \Theta - \Theta_h = \Theta_h^*, h = L, U$$

◦  $H_{01} : \Theta_L^* \leq 0$        $H_{a1} : \Theta_{SL}^* > 0$  에 대해 Wilcoxon-Mann-Whitney 검정을 하기 위해서는 우선,  $\{b_{Lik}\}$ 의 순위를 구해야한다.

$R(b_{Lik})$  = 전체자료에서  $b_{Lik}$ 의 순위  $i=1, \dots, n_k$   $k=1, 2$

$R_L = \sum_{i=1}^{n_1} R(b_{Li1})$  : sequence 1에서의 subject들의 AUC값에 대한 순위매긴 값을 합한 값

$$W_L = R_L - \frac{1}{2} n_1(n_1 + 1)$$

만일,  $W_L > W_{(1-\alpha)}$ 이면  $H_{01}$ 를 기각한다. 여기서  $W_{(1-\alpha)}$ 값은 Chow & Liu (2000) p 551부터 있는 Appedix A.5의 표의 값을 참조한다.

유사하게  $H_{02} : \Theta_U^* \geq 0$        $H_{a2} : \Theta_{SU}^* < 0$ 에 대한 검정은

$W_U = R_U - \frac{1}{2} n_1(n_1 + 1) < w(\alpha)$ 이면  $H_{02}$ 를 기각한다. 이 때  $R_U = \sum_{i=1}^{n_1} R(b_{ui1})$ 이고,  $w(\alpha)$ 값은 역시 Chow & Liu (2000) p 551부터 있는 Appedix A.5의 표의 값을 참조한다.

◦ 만일,  $H_{01}$ 와  $H_{02}$ 가 모두 기각된다면, 생물학적으로 동등하다 결론을 내릴 수 있다.

이와 같은 이론적 배경하에서 다음의 SAS 프로그램을 살펴보겠다.

### 3.2.6.2 프로그램

다음 프로그램에서 part1 부분은 앞의 2.3의 PROC GLM 부분을 이용한 것이다. part 1과 part 2 부분을 합친 것이 하나의 완전한 프로그램이다.

#### <part1>

```
proc means n noprint data=one;
    by group;
    var y;
    output out=no n=n;
```

```
run;
```

```
data no;
```

```
    set no;
```

```
    n=n/2;
```

```
run;
```

```
proc sort data=no;
```

```
    by group;
```

```
data one;
```

```
merge one no;by group;
```

```
proc sort data=one;
```

```
by group subject tmt;
```

```
data no1;
```

```
    set no;
```

```
    if group=1;
```

```
        n1=n;
```

```
        id=1;
```

```
    keep n1 id;
```

```
proc sort data=no1;
```

```
    by id;
```

```
run;
```

```
data no2;
```

```
    set no;
```

```
    if group=2;
```

```
        n2=n;
```

```
        id=1;
```

```
    keep n2 id;
```

```
proc sort data=no2;
```

```
    by id;
```

```
run;
```

```
proc glm data=one outstat=tanova;  
    class tmt period subject group;  
    model y=group subject(group) period tmt;  
        test h=group e=subject(group);  
    lsmeans tmt /stderr pdiff out=lsmean1;  
run;
```

```
data error;  
    set tanova;  
    if _SOURCE_='ERROR';  
    mse=ss/df;  
    sse=ss;  
    id=1;  
    keep id mse sse df;
```

```
proc sort data=error;  
    by id;  
run;
```

```
data lsmean11;  
    set lsmean1;  
    if tmt=1;  
    my1=lsmean;  
    id=1;  
    keep id my1;
```

```
proc sort data=lsmean11;  
    by id;
```

```
data lsmean12;  
    set lsmean1;  
    if tmt=2;
```

```
my2=lsmean;  
id=1;  
keep id my2;
```

```
proc sort data=lsmean12;  
by id;  
run;
```

#### <part2>

```
data one1;  
set one;  
if period=1;  
y1=y;  
drop period y;
```

```
data one2;  
set one;  
if period=2;  
y2=y;  
drop period y;
```

```
proc sort data=one1;  
by group subject;  
proc sort data=one2;  
by group subject;
```

```
data work;  
merge one1 one2;  
by group subject;  
id=1;  
proc sort data=work;  
by id group subject;  
data work;  
merge work lsmean11;by id;  
w1=.;  
w2=.;
```

```

if group=2 then w1=y1;
if group=1 then w1=y2;
if group=2 then w2=y2;
if group=1 then w2=y1;
diffw=w1-w2;
sumw=w1+w2;
d=(y2-y1);
d1=.;
if group=1 then d1=d+(2*(0.2*my1));
if group=2 then d1=d;
d12=d1/2;
d2=.;
if group=1 then d2=d-(2*(0.2*my1));
if group=2 then d2=d;
d22=d2/2;
t=y1+y2;
p=.;
if group=1 then p=d;
if group=2 then p=-d;

proc sort data=work;
by group;

proc npar1way wilcoxon data=work;
class group;
var d d12 d22 p t;
proc print;run;

```

### 3.2.6.3 프로그램 설명

<part1>의 프로그램은 뒷부분의 proc glm 부분을 참고한다.

여기서 처음의 dataset one은 앞에서 만들어진 dataset을 이용한다.

<part2>

```
data one1;
```

```

set one;
if period=1;
y1=y;
drop period y;

```

현재 one이라는 이름의 dataset안에 있는 변수는 group, subject, tmt, y, period이고, group,subject,tmt 순으로 정렬되어 있다. 이 dataset 에서 period가 1인 경우의 y값을 y1이라는 변수에 저장하고, period와 y변수를 제외한 나머지 변수들로 새로운 dataset one1을 생성한다.

```

data one2;
set one;
if period=2;
y2=y;
drop period y;

```

위와 같은 식으로 period가 2인 경우의 y값을 y2로 하여 새로운 변수를 생성하고 period와 y변수를 제외한 나머지 변수들로 새로운 dataset one2를 생성한다.

```

proc sort data=one1;
by group subject;
proc sort data=one2;
by group subject;

```

dataset one1과 one2를 group, subject 변수 순으로 정렬한다.

```

data work;
merge one1 one2;
by group subject;
id=1;
proc sort data=work;
by id group subject;

```

dataset one1과 one2를 group 변수와 subject 변수에 의해 병합하고, id변수를 값 1로 하는 새로운 dataset work를 생성한다. Id,group,subject 변수에 따라서 정렬한다.



이와 같은 일련의 과정을 통해서 dataset one에서 period값을 제거하고 y값을 period에 따라서 y1,y2로 구분한 dataset work가 생성된다. 여기서 tmt 값은 period가 2일 때의 tmt값으로 지정된다.

```
data work;  
merge work lsmean11;by id;
```

lsmean11에는 my1과 id변수가 포함되어 있다. 이 lsmean11과 work dataset을 병합하면 이전의 work dataset에 my1(tmt가 1인 경우의 lsmean 값) 82.5594 값이 포함된 새로운 dataset work가 생성된다.

```
w1=.;  
w2=.;  
if group=2 then w1=y1;  
if group=1 then w1=y2;  
if group=2 then w2=y2;  
if group=1 then w2=y1;
```

새로운 변수 w1과 w2를 생성한다.

만일 group이 2(TR)인 경우에는 y1값이 w1값으로 y2값이 w2으로 지정되고,  
group 1(RT)인 경우에는 y2값이 w1으로 y1값이 w2로 지정된다.

```
diffw=w1-w2;  
sumw=w1+w2;  
d=(y2-y1);
```

diffw는 w1-w2로 formulation이 T일 때의 y값에서 R일 때의 y값을 차감해주는 값으로 설정된다. d의 경우는 y2에서 y1을 즉, period가 2일 때의 y값에서 period가 1 일 때의 y값을 차감해주는 값으로 설정된다.

```
d1=.;  
if group=1 then d1=d+(2*(0.2*my1));  
if group=2 then d1=d;  
d12=d1/2;
```

```
d2=.;
if group=1 then d2=d-(2*(0.2*my1));
if group=2 then d2=d;
d22=d2/2;
```

d1이라는 변수를 새로 생성한다. group이 1인 경우에는  $d+(2*(0.2*my1))$ 이 d1값으로 지정되고, group이 2인 경우는 d값이 d1으로 지정된다.

d12변수는 d1을 2로 나누어준 값으로 지정된다.

d2라는 변수를 새로 생성한다. group이 1인 경우는  $d-(2*(0.2*my1))$ 이 d2값으로 지정되고, group이 2인 경우는 d값이 d2로 지정된다.

d22변수는 d2변수를 2로 나누어준 값으로 지정된다.

```
t=y1+y2;
p=.;
if group=1 then p=d;
if group=2 then p=-d;
```

y1값과 y2값을 더한 것이 t변수로 group1인 경우 d값이 group2인 경우 -d값이 p 변수로 지정된다.

```
proc print;run;
```

```
proc sort data=work;
by group;
```

group별로 정렬한다.

```
proc npar1way wilcoxon data=work;
class group;
var d d12 d22 p t;
```

group변수에 따라서 변수 d,d12,d22,p,t에 대해서 wilcoxon 비모수 검정을 한다.

위의 프로그램에 따른 출력결과는 다음과 같다.

The NPAR1WAY Procedure

**Wilcoxon Scores (Rank Sums) for Variable d12**

Classified by Variable group

group	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	12	207.0	150.0	17.320508	17.250
2	12	93.0	150.0	17.320508	7.750

Wilcoxon Two-Sample Test

Statistic 207.0000

Normal Approximation

Z 3.2620  
 One-Sided Pr > Z 0.0006  
 Two-Sided Pr > |Z| 0.0011

t Approximation

One-Sided Pr > Z 0.0017  
 Two-Sided Pr > |Z| 0.0034

Z includes a continuity correction of 0.5.

Kruskal-Wallis Test

Chi-Square 10.8300  
 DF 1  
**Pr > Chi-Square 0.0010**

The NPAR1WAY Procedure

**Wilcoxon Scores (Rank Sums) for Variable d22**

Classified by Variable group

group	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
-------	---	------------------	----------------------	---------------------	---------------

---

1	12	91.0	150.0	17.320508	7.583333
2	12	209.0	150.0	17.320508	17.416667

#### Wilcoxon Two-Sample Test

Statistic 91.0000

#### Normal Approximation

Z -3.3775

One-Sided Pr < Z 0.0004

Two-Sided Pr > |Z| 0.0007

#### t Approximation

One-Sided Pr < Z 0.0013

Two-Sided Pr > |Z| 0.0026

Z includes a continuity correction of 0.5.

#### Kruskal-Wallis Test

Chi-Square 11.6033

DF 1

**Pr > Chi-Square 0.0007**

이 신뢰구간 값은 (Chow & Liu, 2000) p 113의 결과와 일치한다. 즉,  $R_L$  값은 207,  $R_U$  값은 91이다. 2.6.1에서 언급했듯이, 만일  $W_L > W_{(1-\alpha)}$ 이고,  $W_U < w(\alpha)$ 이면 생물학적으로 동등하다 결론을 내릴 수 있다. 이 때,  $W_{(1-\alpha)}$ 값과  $w(\alpha)$  값은 Chow & Liu (2000) p 551부터 있는 Appedix A.5의 표의 값을 참조하고,  $W_U = R_U - \frac{1}{2} n_1(n_1+1)$ ,  $W_L = R_L - \frac{1}{2} n_1(n_1+1)$  이다.

### 3.2.7 Assumptions and Outliers

#### 3.2.7.1 정규확률도와 Shpiro-Wilk test

앞서 언급했듯이  $2 \times 2$  crossover design 에서는 생물학적 동등성 시험을 할 때 다음의 모형을 가정한다.

$Y_{ijk}$  =  $k$  번째 sequence의  $j$  번째 period에서 AUC 측정값.

$$Y_{ijk} = \mu + S_{ik} + P_j + F_{j,k} + C_{(j-1,k)} + \varepsilon_{ijk}$$

여기서,

$\mu$  : 전체 평균

$S_{ik}$  :  $k$  번째 sequence에서  $i$  번째 subject의 효과(랜덤)

$P_j$  :  $j$  번째 period 의 효과(고정)

$F_{j,k}$  :  $k$  번째 sequence에서  $j$  번째 period의 제제의 효과(고정)

$C_{(j-1,k)}$ :  $k$  번째 sequence에서  $(j-1)$  번째 period의 잔류효과(고정)

$\varepsilon_{ijk}$  : 오차항

이다.

그리고, 그 모형하에서 기본적으로 다음과 같은 정규성 가정을 한다.

1.  $S_{ik} \sim N(0, \sigma_s^2)$
2.  $\varepsilon_{ijk} \sim N(0, \sigma_e^2)$
3.  $S_{ik}$  and  $\varepsilon_{ijk}$  : 독립

대부분의 생물학적 동등성 시험시 위의 가정은 성립하지 않을 것이다. 이 가정의 위배는 생물학적 동등성 시험에 영향을 미치고 결과마저 다르게 할 수 있다. 그리고 가장 일반적으로 발생하는 문제 중 하나가 다른 관측치에 비해서 유난히 크거나 작은 관측치인 outlier가 관측되는 것이다. outlier는 생물학적 동등성 시험의 결과마저 뒤바꿀 수 있다. 따라서 정규성 가정의 위배와 outlier의 존재 여부를 면밀히 검토해야한다.

sequence가  $k$ 인 period  $j$  내에서 subject  $i$  에 대한 intrasubject residual을  $\widehat{e}_{ijk}$  라 표시하고, 이는 관측값

$Y_{ijk}$ 와  $\widehat{Y}_{ijk}$ 의 차이로 정의된다. 여기서  $\widehat{Y}_{ijk}$ 는 위에서 가정한 모델에 적합시킨 경우의 예측값이다.  $\sigma_e^2$ 를 아는 경우,  $\widehat{e}_{ijk}$ 를 표준화시키면

$$e_{ijk}^* = \frac{\widehat{e}_{ijk}}{\left[\frac{n_k-1}{2n_k} \sigma_e^2\right]^{1/2}} \quad \text{이고 이는 평균이 0이고 분산이 1인 정규분포를 따른다. 그러나 대개는 } \sigma_e^2 \text{이 알려져}$$

있지 않으므로,  $MS_{intra}$ 로 대체한  $e_{ijk}' = \frac{\widehat{e}_{ijk}}{\left[\frac{n_k-1}{2n_k} MS_{intra}\right]^{1/2}}$ 를 이용한다. 이  $e_{ijk}$ 의 intrasubject

variability에 대한 정규성 가정에 대한 검사는 이  $e_{ijk}'$ 의 정규확률도를 그려보아 확인할 수 있다. 만일  $e_{ijk}$ 이 실제로 정규성 가정을 만족한다면, 정규확률도는 직선을 그려야한다. 또한,  $e_{ijk}$ 의 정규성 검정은 Shapiro and Wilk test(1965)를 통해 가능한데, 이는 각 관측치들이 독립인 경우에 가능하다. 따라서, 전체 관측치의 수가 적은 경우에는 독립이 아니기 때문에 test시 주의가 요구된다.

intersubject에 대한 검정도 앞의 intrasubject의 검정과 마찬가지로이다.  $S_{ik}$ 의 intersubject variability에 대한 정규성 가정을 검토하기 위해 intrasubject residual을 이용할 수 있다. intrasubject residual  $\widehat{S}_{ik}$ 은  $Y_{i.k} - \overline{Y}_{..k}$ 로 정의되고, 표준화시키면

$$S_{ik}' = \frac{\widehat{S}_{ik}}{\left[\frac{2(n_k-1)}{n_k} MS_{inter}\right]^{1/2}} \quad \text{이다.}$$

따라서,  $S_{ik}$ 의 정규성을 검정하기 위해서는  $S_{ik}'$ 의 정규확률도를 그려보거나 Shapiro-Wilk test를 해보면 된다.

위의 정규성 검정을 위한 프로그램을 다음에 제시하였다.

### 3.2.7.2 프로그램

다음은 intrasubject와 intersubject variability에 대한 정규성 검정을 위한 프로그램이다.

```
Libname out 'c:\Wwork';
data one;
infile 'c:\Wwork\Wone.csv' delimiter=',';
```

```
input group subject tmt y period;  
run;
```

```
proc sort data=one;  
by group subject tmt;
```

```
proc means n noprint data=one;  
by group;  
var y;  
output out=no n=n;  
run;
```

```
proc means n noprint data=one;  
by group subject;  
var y;  
output out=sum sum=ysum;
```

```
data no;  
set no;  
n=n/2;  
run;
```

```
proc sort data=no;  
by group;
```

```
data one; merge one no; by group;  
proc sort data=one; by group subject tmt;
```

```
proc glm data=one;  
class tmt period subject group;  
model y=group subject(group) period tmt;  
test h=group e=subject(group);  
means tmt period group;  
lsmeans tmt period/stderr pdiff;  
lsmeans group/stderr pdiff e=subject(group);  
run;
```

```

proc glm data=one outstat=tanova;
class tmt period subject;
model y=subject period tmt;
output out=pred p=yhat r=resid student=stresid;
means tmt period;
lsmeans tmt period/stderr pdiff out=lsmean1;

proc glm data=sum;
model ysum=group;
output out=predsum p=ysumhat r=residsum student=residsub;

data pred;set pred; if period=1;

proc rank out=rpred normal=blom data=pred;
var stresid;
ranks rankr;
proc sort data=rpred;by rankr;

proc plot data=rpred;
plot stresid*yhat;
plot stresid*rankr;

proc univariate normal plot data=pred;
var stresid;

proc rank out=rpredsum normal=blom data=predsum;
var residsub;
ranks ranksum;
proc sort data=rpredsum; by ranksum;

proc plot data=rpredsum;
plot residsub*ranksum;

proc univariate normal plot data=predsum;
var residsub;

data rpred1; set rpred;

```



```

keep subject stresid;
proc sort data=rpred1; by subject;

data rpredsum; set rpredsum;
keep subject residsub;
proc sort data=rpredsum; by subject;

data indep; merge rpred1 rpredsum; by subject;
proc corr pearson spearman data=indep;
var stresid residsub;quit;

```

### 3.2.7.3 프로그램 설명

```

Libname out 'c:\work';
data one;
  infile 'c:\work\Wone.csv' delimiter=';';
  input group subject tmt y period;
run;

proc sort data=one;
  by group subject tmt;

```

infile문 아래 자료가 저장되어 있는 경로를 지정해 자료를 읽어들이고, 여기서 csv파일은 ‘;’로 구분되어 있으므로 구분자로 ‘;’를 쓴다. 순서대로 group, subject, tmt, y, period로 읽어들이어 이 변수들을 포함하는 one이라는 이름의 dataset이 생성된다. 생성된 data set one을 group, subject, tmt 순으로 정렬시킨다.

```

proc means n noprint data=one;
  by group;
  var y;
  output out=no n=n;
run;

```

PROC MEANS SAS 명령문은 dataset one의 기초통계량들을 구한다. 여기서 n은 계산에 사용된 관찰치의

개수를 출력하고, noprint는 기초통계량을 output 창에 출력하지 않게 한다. 이 옵션을 쓰면 출력dataset이 명시되어야 한다. group 변수의 값을 기준으로 변수 y값을 분석한다. n값이 포함된 no라는 이름의 dataset을 생성한다.

```
proc means n noprint data=one;
by group subject;
var y;
output out=sum sum=ysum;
```

마찬가지로 PROC MEANS를 이용해 기초통계량을 구해서 이번에는 y값의 sum을 값으로 가지는 ysum이라는 변수를 새로 생성해낸다. 이 ysum값이 포함된 sum이라는 이름의 dataset이 생성된다.

```
data no;
    set no;
    n=n/2;
run;
proc sort data=no; by group;
```

각각의 그룹별 관찰치의 개수가 n으로 들어갔는데 이 n의 값은 각 그룹의 피험자의 수의 2배이므로(reference,test 두가지 투여에 따른 y값이 각각 다른 관측치로 들어간다.) 각 그룹별 피험자의 수를 n으로 지정해주기 위해서 2로 나누어 준 값을 n으로 대치한다.

dataset no를 그룹변수에 의해 정렬한다.dataset no의 결과는 다음과 같다.

```
data one; merge one no; by group;
proc sort data=one; by group subject tmt;
```

dataset one과 dataset no를 group 변수에 의해서 병합하여 새로운 dataset one을 생성하고, group,subject,tmt 변수에 의해서 정렬시킨다.

```
proc glm data=one;
    class tmt period subject group;
    model y=group subject(group) period tmt;
        test h=group e=subject(group);
        means tmt period group;
    lsmeans tmt period/stderr pdiff;
```

```
lsmeans group/stderr pdiff e=subject(group);
run;
```

```
proc glm data=one outstat=tanova;
class tmt period subject;
model y=subject period tmt;
output out=pred p=yhat r=resid student=stresid;
means tmt period;
lsmeans tmt period/stderr pdiff out=lsmean1;
```

```
proc glm data=sum;
model ysum=group;
output out=predsum p=ysumhat r=residsum student=residsub;
```

```
data pred;set pred; if period=1;
```

```
proc rank out=rpred normal=blom data=pred;
var stresid;
ranks rankr;
proc sort data=rpred;by rankr;
```

```
proc plot data=rpred;
plot stresid*yhat;
plot stresid*rankr;
```

```
proc univariate normal plot data=pred;
var stresid;
```

intrasubject의 정규성을 검정하기 위한 정규확률도를 그리고 Shapiro-wilk test를 하기 위해서 PROC UNIVARIATE 프로시저를 사용하고 옵션에 normal을 지정해준다.

```
proc rank out=rpredsum normal=blom data=predsum;
var residsub;
ranks ranksum;
proc sort data=rpredsum; by ranksum;
```

```
proc plot data=rpredsum;
plot residsub*ranksum;
```

```
proc univariate normal plot data=predsum;
var residsub;
```

intersubject의 정규성을 검정하기 위한 정규확률도를 그리고 Shapiro-wilk test를 하기 위해서 PROC UNIVARIATE 프로시저를 사용하고 옵션에 normal을 지정해준다.

```
data rpred1; set rpred;
keep subject stresid;
proc sort data=rpred1; by subject;
```

```
data rpredsum; set rpredsum;
keep subject residsub;
proc sort data=rpredsum; by subject;
```

```
data indep; merge rpred1 rpredsum; by subject;
proc corr pearson spearman data=indep;
var stresid residsub;
```

프로그램의 결과는 다음과 같다.

#### The GLM Procedure

##### Class Level Information

Class	Levels	Values
tmt	2	1 2
period	2	1 2
subject	24	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
group	2	1 2

Number of observations 48

The GLM Procedure

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	25	16586.24750	663.44990	3.97	0.0009
Error	22	3679.42953	167.24680		
Corrected Total	47	20265.67703			

R-Square	Coeff Var	Root MSE	y Mean
0.818440	15.88441	12.93239	81.41563

Source	DF	Type I SS	Mean Square	F Value	Pr > F
group	1	276.00021	276.00021	1.65	0.2123
subject(group)	22	16211.48870	736.88585	4.41	0.0005
period	1	35.96672	35.96672	0.22	0.6474
tmt	1	62.79187	62.79187	0.38	0.5463

Source	DF	Type III SS	Mean Square	F Value	Pr > F
group	1	276.00021	276.00021	1.65	0.2123
subject(group)	22	16211.48870	736.88585	4.41	0.0005
period	1	35.96672	35.96672	0.22	0.6474
tmt	1	62.79188	62.79188	0.38	0.5463

Tests of Hypotheses Using the Type III MS for subject(group) as an Error Term

Source	DF	Type III SS	Mean Square	F Value	Pr > F
group	1	276.0002083	276.0002083	0.37	0.5468

The GLM Procedure

Level of -----y-----

tmt	N	Mean	Std Dev
1	24	82.5593750	20.7974547
2	24	80.2718750	21.1152123

Level of period	N	Mean	Std Dev
1	24	82.2812500	19.7083990
2	24	80.5500000	22.1614898

Level of group	N	Mean	Std Dev
1	24	83.8135417	17.5441028
2	24	79.0177083	23.6922106

The GLM Procedure  
Least Squares Means

		H0:LSMean1=		
		Standard	H0:LSMEAN=0	LSMean2
tmt	y LSMEAN	Error	Pr >  t	Pr >  t
1	82.5593750	2.6398137	<.0001	0.5463
2	80.2718750	2.6398137	<.0001	

		H0:LSMean1=		
		Standard	H0:LSMEAN=0	LSMean2
period	y LSMEAN	Error	Pr >  t	Pr >  t
1	82.2812500	2.6398137	<.0001	0.6474
2	80.5500000	2.6398137	<.0001	

The GLM Procedure  
Least Squares Means

Standard Errors and Probabilities Calculated Using the Type III MS for subject(group) as an Error Term

		H0:LSMean1=		
		Standard	H0:LSMEAN=0	LSMean2
group	y LSMEAN	Error	Pr >  t	Pr >  t
1	83.8135417	5.5410809	<.0001	0.5468

2	79.0177083	5.5410809	<.0001
---	------------	-----------	--------

The GLM Procedure

Class Level Information

Class	Levels	Values
tmt	2	1 2
period	2	1 2
subject	24	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

Number of observations 48

The GLM Procedure

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	25	16586.24750	663.44990	3.97	0.0009
Error	22	3679.42953	167.24680		
Corrected Total	47	20265.67703			

R-Square	Coeff Var	Root MSE	y Mean
0.818440	15.88441	12.93239	81.41563

Source	DF	Type I SS	Mean Square	F Value	Pr > F
subject	23	16487.48891	716.84734	4.29	0.0006
period	1	35.96672	35.96672	0.22	0.6474
tmt	1	62.79187	62.79187	0.38	0.5463

Source	DF	Type III SS	Mean Square	F Value	Pr > F
subject	23	16487.48891	716.84734	4.29	0.0006
period	1	35.96672	35.96672	0.22	0.6474

tmt	1	62.79188	62.79188	0.38	0.5463
-----	---	----------	----------	------	--------

The GLM Procedure

Level of	-----y-----		
tmt	N	Mean	Std Dev
1	24	82.5593750	20.7974547
2	24	80.2718750	21.1152123

Level of	-----y-----		
period	N	Mean	Std Dev
1	24	82.2812500	19.7083990
2	24	80.5500000	22.1614898

The GLM Procedure  
Least Squares Means

		H0:LSMean1=		
tmt	y LSMEAN	Standard Error	H0:LSMEAN=0 Pr >  t	LSMean2 Pr >  t
1	82.5593750	2.6398137	<.0001	0.5463
2	80.2718750	2.6398137	<.0001	

		H0:LSMean1=		
period	y LSMEAN	Standard Error	H0:LSMEAN=0 Pr >  t	LSMean2 Pr >  t
1	82.2812500	2.6398137	<.0001	0.6474
2	80.5500000	2.6398137	<.0001	

The GLM Procedure

Number of observations 24

The GLM Procedure

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
--------	----	----------------	-------------	---------	--------



Model	1	552.00042	552.00042	0.37	0.5468
Error	22	32422.97740	1473.77170		
Corrected Total	23	32974.97781			

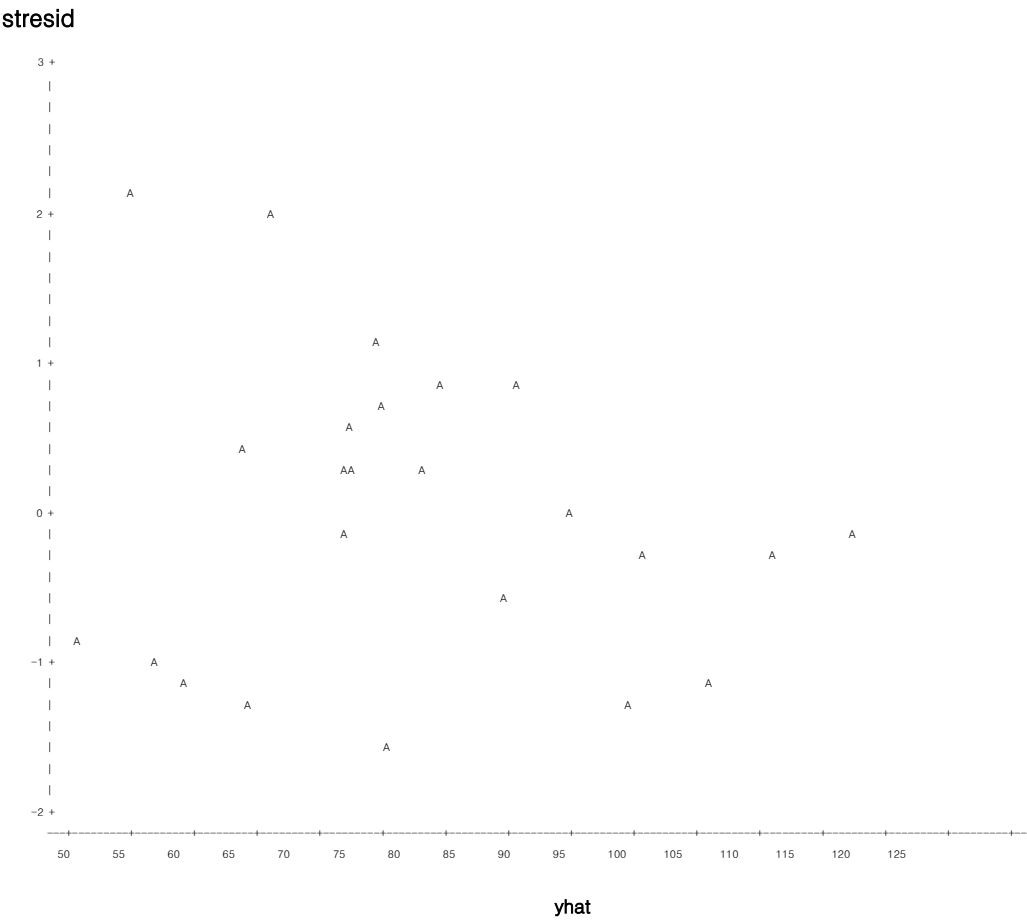
R-Square	Coeff Var	Root MSE	ysum Mean
0.016740	23.57639	38.38973	162.8313

Source	DF	Type I SS	Mean Square	F Value	Pr > F
group	1	552.0004167	552.0004167	0.37	0.5468

Source	DF	Type III SS	Mean Square	F Value	Pr > F
group	1	552.0004167	552.0004167	0.37	0.5468

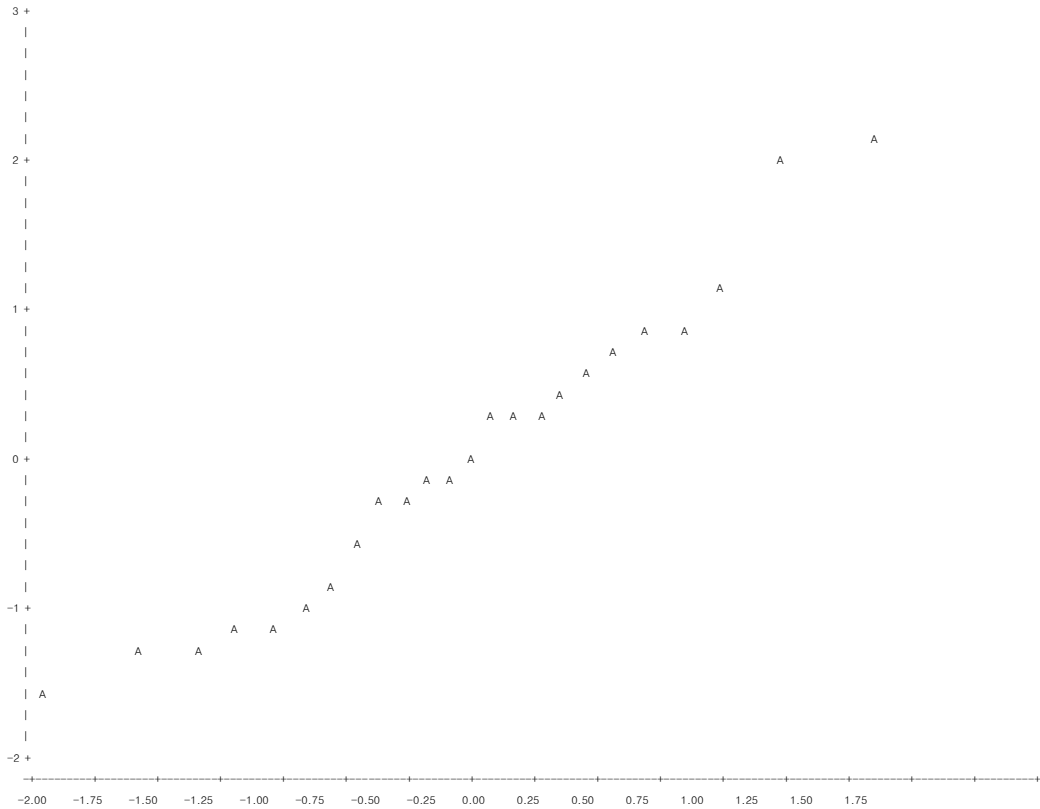
Parameter	Estimate	Standard Error	t Value	Pr >  t
Intercept	177.2187500	24.78046694	7.15	<.0001
group	-9.5916667	15.67254340	-0.61	0.5468

Plot of stresid\*yhat. Legend: A = 1 obs, B = 2 obs, etc.



Plot of stresid\*rankr. Legend: A = 1 obs, B = 2 obs, etc.

stresid



Rank for Variable stresid

## UNIVARIATE 프로시저

변수: stresid

적률

N	24	가중합	24
평균	0	관측치 합	0
표준 편차	1.02150784	분산	1.04347826
왜도	0.40547975	첨도	-0.3198507
제곱합	24	수정 제곱합	24
변동계수	.	표준 오차 평균	0.20851441

기본 통계량의 측정

위치

변동성

평균	0.00000	표준 편차	1.02151
중간값	-0.08102	분산	1.04348
최빈값	.	범위	3.73346
		사분위 범위	1.60118

위치모수 검정:  $\mu_0=0$

검정	통계량	p값
스튜던트의 t	t = 0	Pr >  t  = 1.0000
부호	M = -1	Pr >=  M  = 0.8388
부호 순위	S = -7	Pr >=  S  = 0.8464

### 정규성 검정

검정	통계량	p값
----	-----	----

**Shapiro-Wilk W = 0.957632 Pr < W = 0.3927**

Kolmogorov-Smirnov	D = 0.108651	Pr > D = >0.1500
Cramer-von Mises	W-Sq = 0.035425	Pr > W-Sq = >0.2500
Anderson-Darling	A-Sq = 0.301962	Pr > A-Sq = >0.2500

intrasubject 에 대한 Shapiro-Wilk test의 p-value값이 0.3927로 정규성 가정을 기각하지 않는다. 즉, 정규성이 어긋나지 않는다고 결론 내릴 수 있겠다. 좀 더 살펴보기 위해서 아래의 정규확률도도 함께 확인해본다.

UNIVARIATE 프로시저

변수: stresid

사분위수(정의 5)

사분위수	추정값
100% Max	2.1719058
99%	2.1719058

95%	2.0277069
90%	1.2124838
75% Q3	0.6649565
50% Median	-0.0810226
25% Q1	-0.9362217
10%	-1.2431796
5%	-1.3374085
1%	-1.5615592
0% Min	-1.5615592

### 극값의 관측치

---최소값 부터---

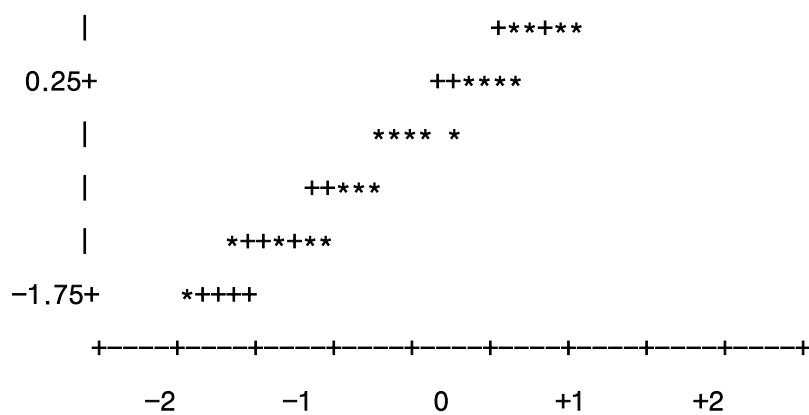
----최대값 부터---

값	관측치	값	관측치
-1.56156	18	0.822719	20
-1.33741	12	0.899815	16
-1.24318	24	1.212484	11
-1.19178	17	2.027707	14
-1.18464	10	2.171906	13

줄기 잎	#	상자그림
2 02	2	
1		
1 2	1	
0 6889	4	+-----+
0 3344	4	+
-0 42210	5	*-----*
-0 995	3	+-----+
-1 3222	4	
-1 6	1	
-----+		

### 정규 확률도

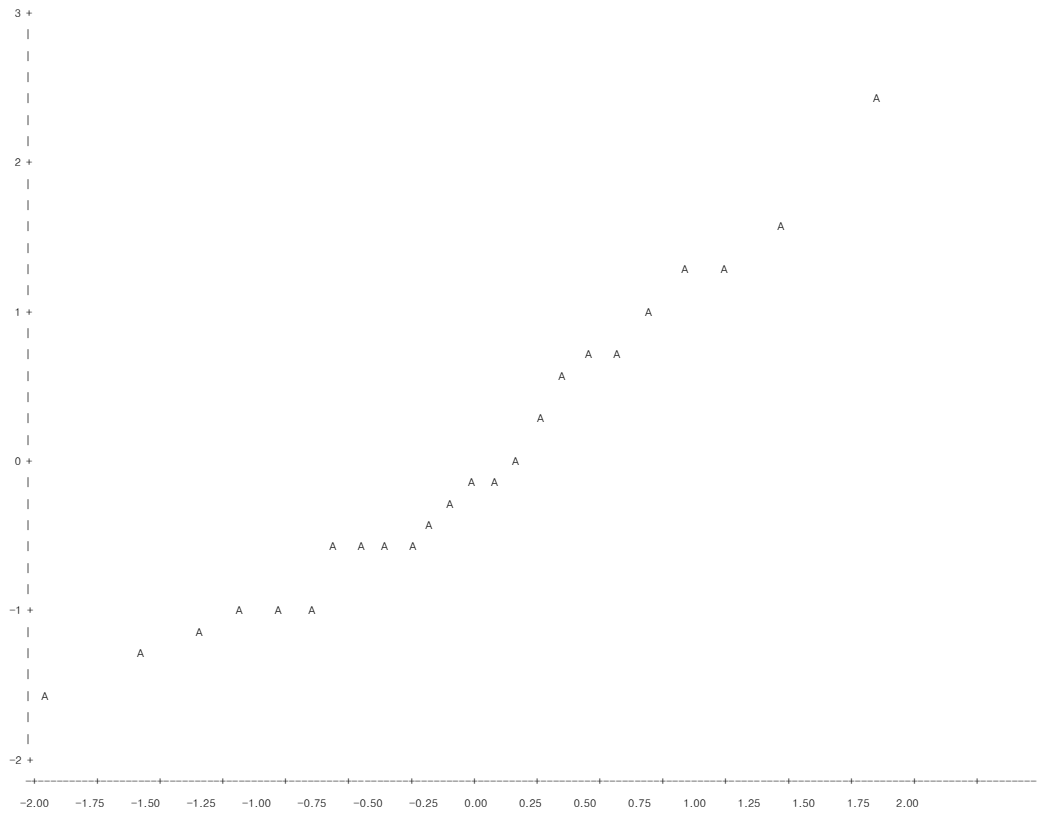
2.25+	*	*+++++
		+++++
		++*++



intersubject에 대한 정규확률도이다. 크게 직선에서 벗어나지 않는 것으로 보아 역시 정규성이 크게 어긋나지 않는다고 결론 내릴 수 있겠다.

Plot of residsub\*ranksum. Legend: A = 1 obs, B = 2 obs, etc.

residsub



Rank for Variable residsub

UNIVARIATE 프로시저

변수: residsub

적률

N	24	가중합	24
평균	0	관측치 합	0
표준 편차	1.02150784	분산	1.04347826
왜도	0.63819595	첨도	-0.2050811
제곱합	24	수정 제곱합	24
변동계수	.	표준 오차 평균	0.20851441

#### 기본 통계량의 측정

위치		변동성	
평균	0.00000	표준 편차	1.02151
중간값	-0.24764	분산	1.04348
최빈값	.	범위	3.95316
		사분위 범위	1.43879

위치모수 검정:  $\mu_0=0$

검정	---통계량---	-----p값-----
스튜던트의 t	t      0      Pr >  t       1.0000	
부호	M      -2      Pr >=  M       0.5413	
부호 순위	S      -6      Pr >=  S       0.8681	

#### 정규성 검정

검정	----통계량----	-----p값-----
----	-------------	--------------

**Shapiro-Wilk    W    0.951602    Pr < W    0.2934**

Kolmogorov-Smirnov	D	0.135191	Pr > D	>0.1500
Cramer-von Mises	W-Sq	0.073612	Pr > W-Sq	0.2433
Anderson-Darling	A-Sq	0.430882	Pr > A-Sq	>0.2500

intersubject 에 대한 Shapiro-Wilk test의 p-value값이 0.2934로 이도 역시 정규성 가정을 기각하지 않는다. 즉, 정규성이 어긋나지 않는다고 결론 내릴 수 있겠다. 좀 더 살펴보기 위해서 아래의 정규확률도도 함께 확인 해본다.

#### UNIVARIATE 프로시저

변수: residsub

사분위수(정의 5)

사분위수	추정값
100% Max	2.432014
99%	2.432014
95%	1.630317



90%	1.309956
75% Q3	0.687031
50% Median	-0.247640
25% Q1	-0.751761
10%	-1.115767
5%	-1.247720
1%	-1.521150
0% Min	-1.521150

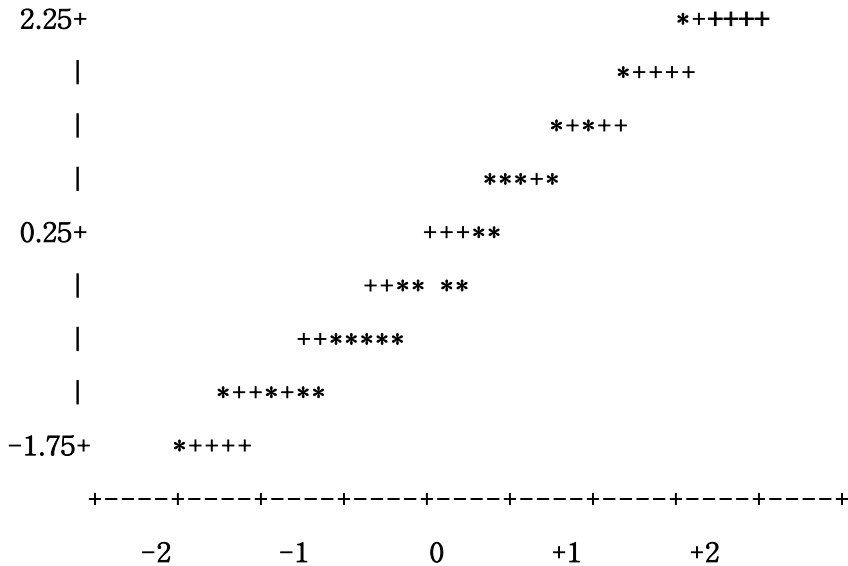
#### 극값의 관측치

---최소값 부터---      ----최대값 부터---

값	관측치	값	관측치
-1.52115	23	0.991635	3
-1.24772	13	1.298846	24
-1.11577	22	1.309956	10
-1.04684	7	1.630317	9
-1.03052	12	2.432014	19

줄기 잎	#	상자그림
2 4	1	
1 6	1	
1 033	3	
0 677	3	+-----+
0 13	2	+
-0 4411	4	*-----*
-0 5555	4	+-----+
-1 21000	5	
-1 5	1	
-----+-----+-----+-----+		

## 정규확률도



정규확률도를 살펴보아도 역시 정규성이 위배되지 않는다.

따라서, 이 장에서 다뤘던 예제들에 대한 가정들은 위배되지 않았다고 할 수 있다. 만일, 정규성이 위배된다면 2.6절에서 제시한 비모수적 방법을 고려해 볼 수 있다.

2×2 standard crossover design은 잔류효과가 존재하는 경우 유용하지 않고, 같은 사람 내에서의 변동 (intrasubject variabilities)을 독립적으로 추정할 수 없는 단점을 가진다. 이와 같은 단점 때문에 period나 sequence 의 수가 제제의 수보다 더 큰 다음과 같은 higher-order crossover design을 고려한다.

### 3.3. 4 × 2 crossover design

#### 3.3.1 이론적 배경

##### ·Balaam's Design : 4 × 2 design

seq	period	
	I	II
1	T	T
2	R	R
3	R	T
4	T	R

앞 절에서의 디자인과 다른 점은 sequence가 2가 아닌 4로 늘어났다는 것이다. 여기서는 sequence 1에서는 각 사람(subject)들에게 period 1,2 에 걸쳐서 시험약만 투여하고, sequence 2에서는 대조약만을 투여한다. sequence 3과 sequence 4에서는 2 × 2 에서와 마찬가지로 시험약과 대조약을 번갈아 투여한다. 따라서, sequence 1,2만 제외한다면 2 × 2 와 동일한 디자인이 된다. 이 디자인의 장점은 잔류효과 존재시 제제효과 F 값 추정이 가능하다는 것과,  $\sigma_T^2$  과  $\sigma_R^2$  을 독립적으로 추정가능하다는 것이다. 여기서 잔류효과란 그 이전 기간의 투여된 약의 효과가 그 다음 기간에 약을 투여할 때까지 약효가 남아있어 영향을 미치는 것을 뜻한다.

。 평균에 대한 기대값

$$Y_{jk}, j=1,2 \quad k=1,2,3,4$$

seq	period	
	I	II
1	$\mu + G_1 + P_1 + F_T$	$\mu + G_1 + P_2 + F_T + C_T$
2	$\mu + G_2 + P_1 + F_R$	$\mu + G_2 + P_2 + F_R + C_R$
3	$\mu + G_3 + P_1 + F_R$	$\mu + G_3 + P_2 + F_T + C_R$
4	$\mu + G_4 + P_1 + F_T$	$\mu + G_4 + P_2 + F_R + C_T$

。 자유도

Source	df
Intersubject	$n_1 + n_2 + n_3 + n_4 - 1$
sequence	3
residual	$n_1 + n_2 + n_3 + n_4 - 4$
Intrasubject	$n_1 + n_2 + n_3 + n_4$
period	1
formulation	1
carryover	1
residual	$n_1 + n_2 + n_3 + n_4 - 3$
Total	$2(n_1 + n_2 + n_3 + n_4) - 1$

모형 :

$$Y_{ijk} = \mu + G_k + S_{ik} + P_j + F_{(j,k)} + C_{(j-1,k)} + e_{ijk}$$

$\mu$  : 전체 평균

$G_k$  : sequence 효과(고정)

$S_{ik}$  :  $k$  번째 sequence에서  $i$  번째 subject의 효과(랜덤)

$P_j$  :  $j$  번째 period 의 효과(고정)

$F_{j,k}$  :  $k$  번째 sequence에서  $j$  번째 period의 제제의 효과(고정)

$C_{(j-1,k)}$  :  $k$  번째 sequence에서  $(j-1)$  번째 period의 잔류효과(고정)

$e_{ijk}$  : 오차항

이 design에서는 모형에  $2 \times 2$  에서는 따로 추정할 수 없었던  $G_k$  가 포함되었다. 이는  $2 \times 2$  에서는 sequence effect가 잔류효과와 교락되어 있었으나 higher-order crossover design에서는  $G_k$ 를 추정할 수 있으므로, 모형에 포함시킨다.

### 3.3.2 예제 자료

여기서 이용된 자료는 Chow & Liu(2000) p265 의 자료이다.

자료는 엑셀에서 입력한 후 다음과 같이 쉼표로 구분되도록 .csv로 저장하였다.

이것은 group, subject, formulation, y(AUC), period 변수가 포함된 2×2와 같은 형태의 원자료이다.

1,1,2,280,1  
1,1,2,482,2  
1,2,2,219,1  
1,2,2,161,2  
1,3,2,230,1  
1,3,2,99,2  
1,4,2,229,1  
1,4,2,260,2  
1,5,2,494,1  
1,5,2,274,2  
1,6,2,112,1  
1,6,2,171,2  
2,7,1,205,1  
2,7,1,221,2  
2,8,1,349,1  
2,8,1,420,2  
2,9,1,285,1  
2,9,1,288,2  
2,10,1,266,1  
2,10,1,247,2  
2,11,1,161,1  
2,11,1,175,2  
2,12,1,240,1  
2,12,1,248,2  
3,13,1,325,1  
3,13,2,225,2  
3,14,1,374,1

3,14,2,439,2  
 3,15,1,416,1  
 3,15,2,372,2  
 3,16,1,243,1  
 3,16,2,119,2  
 3,17,1,248,1  
 3,17,2,269,2  
 3,18,1,345,1  
 3,18,2,334,2  
 4,19,2,177,1  
 4,19,1,290,2  
 4,20,2,174,1  
 4,20,1,224,2  
 4,21,2,235,1  
 4,21,1,271,2  
 4,22,2,380,1  
 4,22,1,340,2  
 4,23,2,308,1  
 4,23,1,270,2  
 4,24,2,269,1  
 4,24,1,249,2

잔류효과의 존재여부를 검정하기 위해서는 group, subject, formulation, y(AUC), period 변수가 포함된 2×2와 같은 형태의 원자료에 새로운 carry라는 잔류효과 변수가 추가된 새로운 자료를 생성해 주어야 한다. 자료 입력 순서는 group, subject, formulation, y(AUC), period, carry 의 순서이다. carry 변수를 생성하는 방법은 3.2.1에서 설명하겠다.

1,1,2,280,1,1  
 1,1,2,482,2,2  
 1,2,2,219,1,1  
 1,2,2,161,2,2  
 1,3,2,230,1,1

1,3,2,99,2,2  
1,4,2,229,1,1  
1,4,2,260,2,2  
1,5,2,494,1,1  
1,5,2,274,2,2  
1,6,2,112,1,1  
1,6,2,171,2,2  
2,7,1,205,1,1  
2,7,1,221,2,1  
2,8,1,349,1,1  
2,8,1,420,2,1  
2,9,1,285,1,1  
2,9,1,288,2,1  
2,10,1,266,1,1  
2,10,1,247,2,1  
2,11,1,161,1,1  
2,11,1,175,2,1  
2,12,1,240,1,1  
2,12,1,248,2,1  
3,13,1,325,1,1  
3,13,2,225,2,1  
3,14,1,374,1,1  
3,14,2,439,2,1  
3,15,1,416,1,1  
3,15,2,372,2,1  
3,16,1,243,1,1  
3,16,2,119,2,1  
3,17,1,248,1,1  
3,17,2,269,2,1  
3,18,1,345,1,1  
3,18,2,334,2,1  
4,19,2,177,1,1  
4,19,1,290,2,2

4,20,2,174,1,1  
4,20,1,224,2,2  
4,21,2,235,1,1  
4,21,1,271,2,2  
4,22,2,380,1,1  
4,22,1,340,2,2  
4,23,2,308,1,1  
4,23,1,270,2,2  
4,24,2,269,1,1  
4,24,1,249,2,2

#### 자료(sas) :

y : 대조약, 시험약에 따른 AUC  
group : sequence(1,2,3,4)  
tmt : formulation ( 1=reference, 2=test )  
period(1,2)  
subject (n1=n2=n3=n4=6 )  
carry

group은 2×2에서보다 2개의 그룹이 늘어난 1,2,3,4 네 개의 그룹이다. 사람(subject)은 각각 한 그룹 당 6명씩 배치하여 총 24명이고, tmt 변수는 역시 referece formulation을 1로, test formulation을 2로 y값은 AUC 값을, period값은 1,2로 지정한다.

앞으로 이 절에서 사용되는 자료는 앞서 제시된 두가지가 사용될 것인데, carry 변수가 포함되지 않은 것, carry변수가 포함된 것 두가지이다. carry 변수가 포함되지 않은 자료는 group,subject,tmt,y,period(sas 입력 data 명) 순으로 .csv 형태로 c 아래 work 폴더에 two 라는 이름으로 저장되어 있다고 하고,(즉, c:/work/two.csv) carry 변수가 포함된 자료는 group,subject,tmt,y,period,carry 순으로 .csv 형태로 c 아래 work 폴더에 twocarry 라는 이름으로 저장되어 있다고 할 것이다.(즉, c:/work/twocarry.csv) 이 자료들은 infile문을 통해서 불러들일 것이다.

우선 carry 변수를 어떻게 생성하는지를 다음 절에서 살펴보겠다.



### 3.3.3 carry 변수의 생성

잔류효과를 검정하기 위해서 그리고 잔류효과가 존재하는 경우의 분석을 위해서는 carry 변수가 필요하다. carry 변수를 생성하는 방법은 Jones & Kenward(2003) p212를 참고하였다.

carry 변수의 값은 전적으로 제제(formulation) 변수에 의해서 생성되어진다. 이는 잔류효과가 무엇인지 정의를 생각해보면 쉽게 이해될 수 있다. 잔류효과란 다음 period에 그 이전 period에 투여되었던 제제의 효과가 영향을 미치는 경우이다. 즉, 그 이전에 투여한 제제가 어느 것인지가 중요하다고 할 수 있겠다. 일단, 3.3.2 에서의 자료를 통해서 살펴보면, group이 1 인 경우의 제제를 나타내는 변수 tmt의 값은 모두 2 이고, subject 값은 다음과 같다. 자료는 subject 별로 정렬된 자료를 이용한다.

subject period tmt

1	1	2
1	2	2
2	1	2
2	2	2
3	1	2
3	2	2
4	1	2
4	2	2
5	1	2
5	2	2
6	1	2
6	2	2

그러면 일단, 생성하려는 carry 변수는 무조건 각 subject 별로 초기값은 1로 둔다. 여기서 각 subject 별로 보는 것이 편한 이유는 각 subject 별로 period가 1,2 로 달라지기 때문이라고 생각해 볼 수 있다. 각 subject 별로 초기값은 1로 둔 결과는 다음과 같다.

subject period tmt carry

1	1	2	1
1	2	2	
2	1	2	1
2	2	2	

3	1	2	1
3	2	2	
4	1	2	1
4	2	2	
5	1	2	1
5	2	2	
6	1	2	1
6	2	2	

그리고 나머지 carry 변수에서 비는 자리는 같은 subject 내에서 바로 전 period의 tmt 값을 넣어준다. 즉 첫 번째 subject의 경우에서 보면, 첫 번째 period의 tmt 값이 2이므로 두 번째 period의 carry 값에 2를 넣어준다. 이런 식으로 각각의 subject 별로 carry 값을 채워넣으면 다음과 같다.

subject	period	tmt	carry
1	1	2	1
1	2	2	2
2	1	2	1
2	2	2	2
3	1	2	1
3	2	2	2
4	1	2	1
4	2	2	2
5	1	2	1
5	2	2	2
6	1	2	1
6	2	2	2

여기서는 gorup1인 경우만을 예로 들어 살펴보았는데, 이와 같은 방법으로 각 group 별로 carry 변수를 생성해 낼 수 있다.

### 3.3.4 잔류효과가 존재하는 경우

#### 3.3.4.1 프로그램(PROC MIXED)

앞의 3.3.2에서 언급한  $4 \times 2$  자료를 이용하여 PROC MIXED를 이용하여 분석하는 SAS 프로그램은 다음과 같다.

```
data two;
  infile 'c:\work\twocarry.csv' delimiter=';';
  input group subject tmt y period carry;
run;
```

```
proc sort data=two;
  by group subject tmt;
```

```
proc mixed data=two;
class tmt period subject group carry;
model y=group period tmt carry;
random subject(group);
lsmeans tmt/pdiff cl alpha=0.1;
estimate 'T vs. R' tmt -1 1/cl alpha=0.1;
run;
```

#### 3.3.4.2 프로그램 설명

앞서  $2 \times 2$  와 달라진 것은 변수에 carry가 포함되었다는 것이다. infile문을 이용해 carry 변수가 포함된 자료 twocarry.csv를 불러들인다. group, subject, tmt, y, period 변수에 carry변수를 포함시켜 input문에서 새롭게 carry 변수를 읽어주고 mixed문장에서 class, model문에 carry 변수를 추가해준다.

위의 프로그램으로 인한 결과는 다음과 같다.

The Mixed Procedure

Model Information

Data Set	WORK.TWO
Dependent Variable	y
Covariance Structure	Variance Components
Estimation Method	REML
Residual Variance Method	Profile
Fixed Effects SE Method	Model-Based
Degrees of Freedom Method	Containment

#### Class Level Information

Class	Levels	Values
tmt	2	1 2
period	2	1 2
subject	24	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
group	4	1 2 3 4
carry	2	1 2

#### Dimensions

Covariance Parameters	2
Columns in X	11
Columns in Z	24
Subjects	1
Max Obs Per Subject	48
Observations Used	48
Observations Not Used	0
Total Observations	48

#### Iteration History

Iteration	Evaluations	-2 Res Log Like	Criterion
0	1	502.81526036	
1	1	494.85062368	0.00000000

#### The Mixed Procedure

Convergence criteria met.

Covariance Parameter  
Estimates

Cov Parm	Estimate
subject(group)	4978.00
Residual	3827.79

Fit Statistics

-2 Res Log Likelihood	494.9
AIC (smaller is better)	498.9
AICC (smaller is better)	499.2
BIC (smaller is better)	501.2

Type 3 Tests of Fixed Effects

Effect	Num	Den	F Value	Pr > F
	DF	DF		
group	3	20	0.65	0.5914
period	1	21	0.17	0.6863
tmt	1	21	1.38	0.2528
carry	1	21	0.48	<b>0.4960</b>

잔류효과가 존재하느냐 존재하지 않는냐에 대한 검정은 mixed의 결과 중 위의 고정변수의 type3 tests에서 확인할 수 있다. 결과를 보면, carry변수에 대한 p-value 값이 0.496이다. p-value값이 작은 값일 수록(보통은 0.1 이하) 그 변수값이 유의미하다는 것이다.따라서, 위의 경우에는 잔류효과가 유의하지 않다고 즉, 잔류효과가 존재하지 않는다고 결론내릴 수 있다.

Estimates

Label	Estimate	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
		Error							
T vs. R	-42.0000	35.7202		21	-1.18	0.2528	0.1	<b>-103.47</b>	<b>19.4652</b>

잔류효과가 존재한다고 가정한 경우의 신뢰구간은 (-103.47, 19.4652) 이다.

이 p-value 값과 신뢰구간은 (Chow & Liu, 2000) p 266의 결과와 일치한다.

#### The Mixed Procedure

##### Least Squares Means

Effect	tmt	Estimate	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
			Error							
tmt	1	283.33	21.1342		21	13.41	<.0001	0.1	246.97	319.70
tmt	2	241.33	32.9336		21	7.33	<.0001	0.1	184.66	298.00

##### Differences of Least Squares Means

Effect	tmt	_tmt	Estimate	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
				Error							
tmt	1	2	42.0000	35.7202		21	1.18	0.2528	0.1	-19.4652	103.47

### 3.3.5 잔류효과(carryover effect)가 존재하지 않는 경우

#### 3.3.5.1 프로그램(PROC MIXED)

```
Libname out 'c:\work';
data two;
  infile 'c:\work\two.csv' delimiter=';';
  input group subject tmt y period;
run;

proc sort data=two;
  by group subject tmt;
run;

proc mixed data=two;
  class tmt period subject group;
  model y=group period tmt;
  random subject(group);
  lsmeans tmt/pdiff cl alpha=0.1;
  estimate 'T vs. R' tmt -1 1/cl alpha=0.1;
run;
```

#### 3.3.5.2 프로그램 설명

자료는 carry 변수가 빠진 two.csv 자료를 infile문을 이용해 불러온다. 앞의 잔류효과가 존재하는 경우의 프로그램에서 carry변수가 빠진 것을 제외하면 일치한다. 즉,  $2 \times 2$ 의 프로그램과 일치한다.

위의 프로그램으로 인한 결과는 다음과 같다.

#### The Mixed Procedure

#### Model Information

Data Set	WORK.TWO
Dependent Variable	y
Covariance Structure	Variance Components
Estimation Method	REML

Residual Variance Method	Profile
Fixed Effects SE Method	Model-Based
Degrees of Freedom Method	Containment

#### Class Level Information

Class	Levels	Values
tmt	2	1 2
period	2	1 2
subject	24	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
group	4	1 2 3 4

#### Dimensions

Covariance Parameters	2
Columns in X	9
Columns in Z	24
Subjects	1
Max Obs Per Subject	48
Observations Used	48
Observations Not Used	0
Total Observations	48

#### Iteration History

Iteration	Evaluations	-2 Res Log Like	Criterion
0	1	513.52104671	
1	1	505.00687800	0.00000000

#### The Mixed Procedure

Convergence criteria met.



Covariance Parameter Estimates

Cov Parm	Estimate
subject(group)	5023.24
Residual	3737.33

Fit Statistics

-2 Res Log Likelihood	505.0
AIC (smaller is better)	509.0
AICC (smaller is better)	509.3
BIC (smaller is better)	511.4

Type 3 Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
group	3	20	0.61	0.6146
period	1	22	0.08	0.7867
tmt	1	22	0.96	0.3369

Estimates

Label	Estimate	Standard Error	DF	t Value	Pr >  t	Alpha	Lower	Upper
T vs. R	-24.5000	24.9577	22	-0.98	0.3369	0.1	<b>-67.3560</b>	<b>18.3560</b>

잔류효과가 존재하지 않는 경우의 신뢰구간은 (-103.47, 19.4652) 이다.

여기서 얻어진 신뢰구간은 (Chow & Liu, 2000) p 266의 결과와 일치한다.

The Mixed Procedure

Least Squares Means

Effect	tmt	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
		Estimate	Error						
tmt	1	283.33	21.0448	22	13.46	<.0001	0.1	247.20	319.47
tmt	2	258.83	21.0448	22	12.30	<.0001	0.1	222.70	294.97

Differences of Least Squares Means

Effect	tmt	_tmt	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
			Estimate	Error						
tmt	1	2	24.5000	24.9577	22	0.98	0.3369	0.1	-18.3560	67.3560

### 3.4. 2 × 3 crossover design

#### 3.4.1 이론적 배경

period가 3이고 sequence가 2인 경우이다. 즉, 그룹 1에 있는 사람들은 처음에 시험약을 투여받고, 두 번째와 세 번째는 대조약을 그룹 2에 있는 사람은 처음에는 대조약을 두 번째와 세 번째에는 시험약을 각각 투여 받는 경우이다. 각각의 subject 별로 한가지 제제를 두 번 받을 수 있어서 intrasubject variabilities를 예측할 수 있다는 장점이 있다.

sequence	period		
	I	II	III
1	T	R	R
2	R	T	T

。 모형 :

$$Y_{ijk} = \mu + G_k + S_{ik} + P_j + F_{(j,k)} + C_{(j-1,k)} + e_{ijk}$$

$\mu$  : 전체 평균

$G_k$  : sequence 효과(고정)

$S_{ik}$  :  $k$  번째 sequence에서  $i$  번째 subject의 효과(랜덤)

$P_j$  :  $j$  번째 period 의 효과(고정)

$F_{j,k}$  :  $k$  번째 sequence에서  $j$  번째 period의 제제의 효과(고정)

$C_{(j-1,k)}$  :  $k$  번째 sequence에서  $(j-1)$  번째 period의 잔류효과(고정)

$e_{ijk}$  : 오차항

。 기대값

seq	I	II	III
1	$\mu + G_1 + P_1 + F_T$	$\mu + G_1 + P_2 + F_R + C_T$	$\mu + G_1 + P_3 + F_R + C_R$
2	$\mu + G_2 + P_1 + F_R$	$\mu + G_2 + P_2 + F_T + C_R$	$\mu + G_2 + P_3 + F_T + C_T$

。 ANOVA (자유도)

Source	df
Intersubject	$n_1 + n_2 - 1$
sequence	1
residual	$n_1 + n_2 - 2$
Intrasubject	$2(n_1 + n_2)$
period	2
formulation	1
carryover	1
residual	$2(n_1 + n_2 - 2)$
	$3(n_1 + n_2) - 1$
Total	$3(n_1 + n_2) - 1$

### 3.4.2 예제 자료

여기서 이용된 자료는 Chow & Liu(2000) p275 의 자료이다.

자료는 엑셀에서 입력한 후 다음과 같이 쉼표로 구분되도록 .csv로 저장하였다.

이것은 group, subject, formulation, y(AUC), period, carry 변수가 포함된 잔류효과를 고려한 4×2의 경우와 같은 형태의 자료이다. carry 변수의 생성은 3.3.3 절에서 언급한 방법과 동일하다.

2,1,1,26.79,1,1  
2,1,2,27.23,2,1  
2,1,2,31.6,3,2  
1,2,2,32.14,1,1  
1,2,1,42.55,2,2  
1,2,1,36.08,3,1  
2,3,1,26.71,1,1  
2,3,2,30.81,2,1  
2,3,2,26.39,3,2  
2,4,1,28.7,1,1  
2,4,2,40.74,2,1  
2,4,2,30.49,3,2  
1,5,2,51.85,1,1  
1,5,1,47.37,2,2  
1,5,1,56.5,3,1  
1,6,2,34.28,1,1  
1,6,1,30.7,2,2  
1,6,1,33.6,3,1  
2,7,1,35.01,1,1  
2,7,2,45.85,2,1  
2,7,2,43.86,3,2  
1,8,2,27.48,1,1  
1,8,1,27.87,2,2  
1,8,1,33.54,3,1  
2,9,1,30.98,1,1  
2,9,2,34.57,2,1

2,9,2,30.18,3,2  
 1,10,2,17.32,1,1  
 1,10,1,18.93,2,2  
 1,10,1,19.05,3,1  
 1,11,2,31.61,1,1  
 1,11,1,24.93,2,2  
 1,11,1,28.21,3,1  
 2,12,1,57.7,1,1  
 2,12,2,54.39,2,1  
 2,12,2,44.35,3,2  
 1,13,2,39.26,1,1  
 1,13,1,37.36,2,2  
 1,13,1,45.09,3,1  
 2,14,1,27.87,1,1  
 2,14,2,30.02,2,1  
 2,14,2,27,3,2  
 2,15,1,31.03,1,1  
 2,15,2,25.77,2,1  
 2,15,2,29.65,3,2  
 2,16,1,27.67,1,1  
 2,16,2,21.61,2,1  
 2,16,2,25.64,3,2  
 1,17,2,47.55,1,1  
 1,17,1,40,2,2  
 1,17,1,36,3,1  
 1,18,2,36.15,1,1  
 1,18,1,35.2,2,2  
 1,18,1,33.26,3,1

**자료(sas) :**

y : 대조약, 시험약에 따른 AUC  
 group : sequence(1,2)

```
tmt : formulation ( 1=reference, 2=test )  
period(1,2,3)  
subject (n1=n2=9 )  
carry
```

group은 1,2 두 개의 그룹이고, subject는 각각 한 그룹 당 9명씩 배치하여 총 18명이고, tmt 변수는 역시 대조약을 1로, 시험약을 2로 y값은 AUC 값을, period값은 1,2,3로 지정한다.

앞으로 이 절에서 사용되는 자료는 group,subject,tmt,y,period,carry 순으로 .csv 형태로 c 아래 work 폴더에 threecarry 라는 이름으로 저장되어 있다고 가정할 것이다.(즉, c:/work/threecarry.csv) 이 자료는 SAS에서 infile문을 통해서 불러들일 것이다.

### 3.4.3 SAS PROGRAM

#### 3.4.3.1 프로그램(PROC MIXED)

```
Libname out 'c:\work';  
data three;  
  infile 'c:\work\threecarry.csv' delimiter='';  
  input group subject tmt y period carry;  
run;  
  
proc sort data=three;  
  by group subject tmt;  
  
proc mixed data=three;  
class tmt period subject group carry;  
model y=group period tmt carry;  
random subject(group);  
lsmeans tmt/pdiff cl alpha=0.1;  
estimate 'T vs. R' tmt -1 1/cl alpha=0.1;  
run;
```

#### 3.4.3.2 프로그램 설명

자료는 c 의 work아래 저장된 threecarry.csv를 infile문을 이용해 불러들인다. 그 외에는 4×2 의 잔류효과가 존재하는 경우, 즉 carry 변수가 포함된 경우와 일치한다.

위의 프로그램으로 인한 결과는 다음과 같다.

The Mixed Procedure

Model Information

Data Set	WORK.THREE
Dependent Variable	y
Covariance Structure	Variance Components



Estimation Method	REML
Residual Variance Method	Profile
Fixed Effects SE Method	Model-Based
Degrees of Freedom Method	Containment

#### Class Level Information

Class	Levels	Values
tmt	2	1 2
period	3	1 2 3
subject	18	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
group	2	1 2
carry	2	1 2

#### Dimensions

Covariance Parameters	2
Columns in X	10
Columns in Z	18
Subjects	1
Max Obs Per Subject	54
Observations Used	54
Observations Not Used	0
Total Observations	54

#### Iteration History

Iteration	Evaluations	-2 Res Log Like	Criterion
0	1	369.49912568	
1	1	330.17467139	0.00000000

#### The Mixed Procedure

Convergence criteria met.

Covariance Parameter Estimates

Cov Parm	Estimate
subject(group)	75.8709
Residual	16.6659

Fit Statistics

-2 Res Log Likelihood	330.2
AIC (smaller is better)	334.2
AICC (smaller is better)	334.4
BIC (smaller is better)	336.0

Type 3 Tests of Fixed Effects

Effect	Num	Den	F Value	Pr > F
	DF	DF		
group	1	16	0.25	0.6259
period	2	32	0.44	0.6505
tmt	1	32	0.33	0.5713
carry	1	32	2.44	<b>0.1282</b>

Estimates

Label	Estimate	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
		Error							
T vs. R	0.6742	1.1785		32	0.57	0.5713	0.1	<b>-1.3221</b>	<b>2.6704</b>

결과에서 보여주는 신뢰구간은 (-1.3221, 2.6704) 이다.

이 신뢰구간은 (Chow & Liu, 2000) p 276의 결과와 일치한다.

The Mixed Procedure

Least Squares Means

Effect	tmt	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
		Estimate	Error						
tmt	1	33.3178	2.2186	32	15.02	<.0001	0.1	29.5597	37.0759
tmt	2	33.9920	2.2186	32	15.32	<.0001	0.1	30.2339	37.7501

Differences of Least Squares Means

Effect	tmt	_tmt	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
			Estimate	Error						
tmt	1	2	-0.6742	1.1785	32	-0.57	0.5713	0.1	-2.6704	1.3221

## 3.5 2 × 4 crossover design

### 3.5.1 이론적 배경

group이 2 그룹으로 나뉘고 period가 4인 경우이다. 표와 같이 첫 번째 sequence에 배정된 사람들은 시험약, 대조약, 대조약, 시험약의 순서로 약을 투여받고, 두 번째 sequence에 배정된 사람들은 대조약, 시험약, 시험약의 순서로 약을 투여받는다. 이는 마지막 period가 없다면, 앞 절의 2 × 3 디자인과 동일하다.

seq	period			
	I	II	III	IV
1	T	R	R	T
2	R	T	T	R

。 모형 :

$$Y_{ijk} = \mu + G_k + S_{ik} + P_j + F_{(j,k)} + C_{(j-1,k)} + \varepsilon_{ijk}$$

$\mu$  : 전체 평균

$G_k$  : sequence 효과(고정)

$S_{ik}$  :  $k$  번째 sequence에서  $i$  번째 subject의 효과(랜덤)

$P_j$  :  $j$  번째 period 의 효과(고정)

$F_{j,k}$  :  $k$  번째 sequence에서  $j$  번째 period의 제제의 효과(고정)

$C_{(j-1,k)}$  :  $k$  번째 sequence에서  $(j-1)$  번째 period의 잔류효과(고정)

$\varepsilon_{ijk}$  : 오차항

。 기대값

seq	period			
	I	II	III	IV
1	$\mu + G_1 + P_1 + F_T$	$\mu + G_1 + P_2 + F_R + C_T$	$\mu + G_1 + P_3 + F_R + C_R$	$\mu + G_1 + P_4 + F_T + C_R$
2	$\mu + G_2 + P_1 + F_R$	$\mu + G_2 + P_2 + F_T + C_R$	$\mu + G_2 + P_3 + F_T + C_T$	$\mu + G_2 + P_4 + F_R + C_T$

。 ANOVA (자유도)

Source	df
Intersubject	$n_1 + n_2 - 1$
sequence	1
residual	$n_1 + n_2 - 2$
Intrasubject	$3(n_1 + n_2)$
period	3
formulation	1
carryover	1
residual	$3(n_1 + n_2) - 5$
Total	$4(n_1 + n_2) - 1$

### 3.5.2 예제 자료

여기서 사용된 자료는 Chow & Liu (2000) p 284의 자료를 이용하였다. 여기서 이용된 자료의 연구는 2×2 디자인 두가지로 이루어진 것으로 볼 수 있다. 즉, 일단 2×2의 디자인으로 10명의 지원자를 통해 자료를 얻어낸 후, 6개월 후에 제제의 순서만 뒤바뀌어서 같은 지원자들을 상대로 다시 자료를 얻어냈다. 이 때, 두 번째 단계에서 다섯 번째 subject의 자료가 누락되었다.

자료는 엑셀에서 입력한 후 다음과 같이 쉼표로 구분되도록 .csv로 저장하였다.

자료 입력 순서는 group, subject, formulation, y(AUC), period 의 순서이다.

원자료에서 부분적으로 누락된 다섯 번째 subject의 자료를 아예 제외시켰다.

1,1,2,106.3,1

1,1,1,36.4,2

1,1,1,94.7,3

1,1,2,58.9,4

1,2,2,149.2,1

1,2,1,107.1,2

1,2,1,104.6,3

1,2,2,119.4,4

1,3,2,134.8,1

1,3,1,155.1,2

1,3,1,132.5,3

1,3,2,122,4

1,4,2,108.1,1

1,4,1,84.9,2

1,4,1,33.2,3

1,4,2,24.8,4

2,6,1,85,1

2,6,2,92.8,2

2,6,2,81.9,3

2,6,1,59.5,4

2,7,1,64.1,1

2,7,2,112.8,2

2,7,2,70.4,3  
2,7,1,55.2,4  
2,8,1,15.3,1  
2,8,2,30.1,2  
2,8,2,22.3,3  
2,8,1,17.5,4  
2,9,1,77.4,1  
2,9,2,67.6,2  
2,9,2,72.9,3  
2,9,1,48.9,4  
2,10,1,102,1  
2,10,2,106.1,2  
2,10,2,67.9,3  
2,10,1,70.4,4

위의 자료는 잔류효과가 고려되지 않은 경우, 즉, carry 변수가 포함되지 않은 경우이다.

잔류효과의 존재여부를 검정하기 위해서는 group, subject, formulation, y(AUC), period 변수가 포함된 위의 자료에 새로운 carry라는 잔류효과 변수를 추가해야 한다. 자료 입력 순서는 group, subject, formulation, y(AUC), period, carry 의 순서이다. carry 변수의 생성방법은 3.3 절의 방법을 이용하였다.

1,1,2,106.3,1,1  
1,1,1,36.4,2,2  
1,1,1,94.7,3,1  
1,1,2,58.9,4,1  
1,2,2,149.2,1,1  
1,2,1,107.1,2,2  
1,2,1,104.6,3,1  
1,2,2,119.4,4,1  
1,3,2,134.8,1,1  
1,3,1,155.1,2,2  
1,3,1,132.5,3,1  
1,3,2,122.4,1  
1,4,2,108.1,1,1

1,4,1,84.9,2,2  
 1,4,1,33.2,3,1  
 1,4,2,24.8,4,1  
 2,6,1,85,1,1  
 2,6,2,92.8,2,1  
 2,6,2,81.9,3,2  
 2,6,1,59.5,4,2  
 2,7,1,64.1,1,1  
 2,7,2,112.8,2,1  
 2,7,2,70.4,3,2  
 2,7,1,55.2,4,2  
 2,8,1,15.3,1,1  
 2,8,2,30.1,2,1  
 2,8,2,22.3,3,2  
 2,8,1,17.5,4,2  
 2,9,1,77.4,1,1  
 2,9,2,67.6,2,1  
 2,9,2,72.9,3,2  
 2,9,1,48.9,4,2  
 2,10,1,102,1,1  
 2,10,2,106.1,2,1  
 2,10,2,67.9,3,2  
 2,10,1,70.4,4,2

**자료(sas) :**

y : 대조약, 시험약에 따른 AUC  
 group : sequence(1,2)  
 tmt : formulation ( 1=reference, 2=test )  
 period(1,2,3,4)  
 subject (n1=n2=5)  
 carry



group은 두 개의 그룹이고, 사람(subject)은 각각 한 그룹 당 5명씩 배치하여 총 10명이고, tmt 변수는 역시 referece formulation을 1로, test formulation을 2로 y값은 AUC 값을, period값은 1,2,3,4로 지정한다.

앞으로 이 절에서 사용되는 자료는 앞서 제시된 두가지가 사용될 것인데, carry 변수가 포함되지 않은 것, carry변수가 포함된 것 두가지이다. carry 변수가 포함되지 않은 자료는 group,subject,tmt,y,period(sas 입력 data 명) 순으로 .csv 형태로 c 아래 work 폴더에 five 라는 이름으로 저장되어 있다고 하고,(즉, c:/work/five.csv) carry 변수가 포함된 자료는 group,subject,tmt,y,period,carry 순으로 .csv 형태로 c 아래 work 폴더에 fivecarry 라는 이름으로 저장되어 있다고 할 것이다.(즉, c:/work/fivecarry.csv) 이 자료들은 infile문을 통해서 불러들일 것이다.

위와 같이 입력된 데이터로 다음과 같은 프로그램들을 통해서 그래프와 신뢰구간 등의 다양한 결과를 얻을 수 있다.

### 3.5.3 잔류효과(carryover effect)가 존재하는 경우

#### 3.5.3.1 프로그램(PROC MIXED)

```
Libname out 'c:\work';
data five;
  infile 'c:\work\fivecarry.csv' delimiter=';';
  input group subject tmt y period carry;
run;

proc sort data=five;
  by group subject tmt;

proc mixed data=five;
class tmt period subject group carry;
model y=group period tmt carry;
random subject(group);
lsmeans tmt/pdiff cl alpha=0.1;
estimate 'T vs. R' tmt -1 1/cl alpha=0.1;
run;
```

#### 3.5.3.2 프로그램 설명

자료는 잔류효과 변수 carry를 포함시킨 fivecarry.csv 자료를 infile문을 이용해 불러온다. input 문에서 순서대로 group, subject, tmt, y, period, carry 의 순서로 변수를 읽어들이고 proc sort문을 이용해 group, subject, tmt 의 순으로 정렬한 후 앞의 다른 디자인들과 동일하게 proc mixed 프로시저를 이용해 분석을 한다.

위의 프로그램으로 인한 결과는 다음과 같다.

The Mixed Procedure

Model Information

Data Set

WORK.THREE

Dependent Variable	y
Covariance Structure	Variance Components
Estimation Method	REML
Residual Variance Method	Profile
Fixed Effects SE Method	Model-Based
Degrees of Freedom Method	Containment

#### Class Level Information

Class	Levels	Values
tmt	2	1 2
period	4	1 2 3 4
subject	9	1 2 3 4 6 7 8 9 10
group	2	1 2
carry	2	1 2

#### Dimensions

Covariance Parameters	2
Columns in X	11
Columns in Z	9
Subjects	1
Max Obs Per Subject	36
Observations Used	36
Observations Not Used	0
Total Observations	36

#### Iteration History

Iteration	Evaluations	-2 Res Log Like	Criterion
0	1	302.30473864	
1	1	285.43303805	0.00000000

#### The Mixed Procedure

Convergence criteria met.

Covariance Parameter Estimates

Cov Parm	Estimate
subject(group)	827.46
Residual	381.41

Fit Statistics

-2 Res Log Likelihood	285.4
AIC (smaller is better)	289.4
AICC (smaller is better)	289.9
BIC (smaller is better)	289.8

Type 3 Tests of Fixed Effects

Effect	Num	Den	F Value	Pr > F
	DF	DF		
group	1	7	2.45	0.1614
period	3	22	3.33	0.0383
tmt	1	22	2.56	0.1240
carry	1	22	0.01	<b>0.9337</b>

결과를 보면, carry변수에 대한 p-value 값이 0.937 이다.

이는 (Chow & Liu, 2000) p 283의 값과 일치한다.

Estimates

Label	Estimate	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
		Error							
T vs. R	10.9883	6.8702		22	1.60	0.1240	0.1	<b>-0.8089</b>	<b>22.7854</b>

The Mixed Procedure

Least Squares Means

Effect	tmt	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
		Estimate	Error						
tmt	1	76.5463	10.7025	22	7.15	<.0001	0.1	58.1685	94.9240
tmt	2	87.5345	10.9011	22	8.03	<.0001	0.1	68.8157	106.25

Differences of Least Squares Means

Effect	tmt	_tmt	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
			Estimate	Error						
tmt	1	2	-10.9883	6.8702	22	-1.60	0.1240	0.1	-22.7854	0.8089

### 3.5.4 잔류효과(carryover effect)가 존재하지 않는 경우

#### 3.5.4.1 프로그램(PROC MIXED)

```
Libname out 'c:\work';  
data five;  
  infile 'c:\work\five.csv' delimiter=',';  
  input group subject tmt y period;  
run;  
  
proc sort data=five;  
  by group subject tmt;  
  
proc mixed data=five;  
class tmt period subject group;  
model y=group period tmt;  
random subject(group);  
lsmeans tmt/pdiff cl alpha=0.1;  
estimate 'T vs. R' tmt -1 1/cl alpha=0.1;  
run;
```

#### 3.5.4.2 프로그램 설명

자료는 잔류효과 변수 carry를 제외한 five.csv 자료를 infile문을 이용해 불러온다. input 문에서 순서대로 group, subject, tmt, y, period 의 순서로 변수를 읽어들이고 proc sort문을 이용해 group, subject, tmt 의 순으로 정렬한 후 앞의 다른 디자인들과 동일하게 proc mixed 프로시저를 이용해 분석을 한다.

위의 프로그램으로 인한 결과는 다음과 같다.

#### Model Information

Data Set	WORK.FIVE
Dependent Variable	y
Covariance Structure	Variance Components
Estimation Method	REML
Residual Variance Method	Profile
Fixed Effects SE Method	Model-Based
Degrees of Freedom Method	Containment

#### Class Level Information

Class	Levels	Values
tmt	2	1 2
period	4	1 2 3 4
subject	9	1 2 3 4 6 7 8 9 10
group	2	1 2

#### Dimensions

Covariance Parameters	2
Columns in X	9
Columns in Z	9
Subjects	1
Max Obs Per Subject	36
Observations Used	36
Observations Not Used	0
Total Observations	36

#### Iteration History

Iteration	Evaluations	-2 Res Log Like	Criterion
0	1	309.48668790	
1	1	291.48499986	0.00000000

#### The Mixed Procedure

Convergence criteria met.

Covariance Parameter  
Estimates

Cov Parm	Estimate
subject(group)	831.57
Residual	364.95

Fit Statistics

-2 Res Log Likelihood	291.5
AIC (smaller is better)	295.5
AICC (smaller is better)	295.9
BIC (smaller is better)	295.9

Type 3 Tests of Fixed Effects

Effect	Num	Den	F Value	Pr > F
	DF	DF		
group	1	7	2.50	0.1576
period	3	23	4.25	0.0158
tmt	1	23	3.03	0.0949

Estimates

Label	Estimate	Standard	DF	t Value	Pr >  t	Alpha	Lower	Upper
		Error						
T vs. R	11.1625	6.4075	23	1.74	0.0949	0.1	<b>0.1808</b>	<b>22.1442</b>

The Mixed Procedure



# Least Squares Means

Effect	tmt	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
		Estimate	Error						
tmt	1	76.5463	10.6808	23	7.17	<.0001	0.1	58.2407	94.8518
tmt	2	87.7088	10.6808	23	8.21	<.0001	0.1	69.4032	106.01

# Differences of Least Squares Means

Effect	tmt	_tmt	Standard		DF	t Value	Pr >  t	Alpha	Lower	Upper
			Estimate	Error						
tmt	1	2	-11.1625	6.4075	23	-1.74	0.0949	0.1	-22.1442	-0.1808

## 3.6 Williams Design with Three Formulations

### 3.6.1 이론적 배경

앞 절에서 살펴본 여러 가지 디자인에서는 시험약 한가지, 대조약 한가지로 실험을 한다. 그렇지만, 시험약을 한가지 이상 비교하고 싶은 경우도 종종 있으므로, 여기서는 시험약을  $T_1$ ,  $T_2$  두가지로 하는 경우를 살펴보겠다.

Seq	Period		
	I	II	III
1	R	$T_2$	$T_1$
2	$T_1$	R	$T_2$
3	$T_2$	$T_1$	R
4	$T_1$	$T_2$	R
5	$T_2$	R	$T_1$
6	R	$T_1$	$T_2$

。 모형 :

$$Y_{ijk} = \mu + G_k + S_{ik} + P_j + F_{(j,k)} + C_{(j-1,k)} + e_{ijk}$$

$\mu$  : 전체 평균

$G_k$  : sequence 효과(고정)

$S_{ik}$  :  $k$  번째 sequence에서  $i$  번째 subject의 효과(랜덤)

$P_j$  :  $j$  번째 period 의 효과(고정)

$F_{j,k}$  :  $k$  번째 sequence에서  $j$  번째 period의 제제의 효과(고정)

$C_{(j-1,k)}$  :  $k$  번째 sequence에서  $(j-1)$  번째 period의 잔류효과(고정)

$e_{ijk}$  : 오차항

### 3.6.2 예제 자료

여기서 사용된 자료는 Chow & Liu (2000) p 319의 자료를 이용하였다.

자료는 엑셀에서 입력한 후 다음과 같이 쉼표로 구분되도록 .csv로 저장하였다. 자료 입력 순서는 group, subject, formulation, y(AUC), period, carry 의 순서이다. 여기서 carry 변수의 생성은 3.3.3 절의 방법을 이용하였다.

1,1,1,5.68,1,1

1,1,3,4.21,2,1

1,1,2,6.83,3,3

1,2,1,3.6,1,1

1,2,3,5.01,2,1

1,2,2,5.78,3,3

2,3,2,3.55,1,1

2,3,1,5.07,2,2

2,3,3,4.49,3,1

2,4,2,7.31,1,1

2,4,1,7.42,2,2

2,4,3,7.86,3,1

3,5,3,6.59,1,1

3,5,2,7.72,2,3

3,5,1,7.26,3,2

3,6,3,9.68,1,1

3,6,2,8.91,2,3

3,6,1,9.04,3,2

4,7,2,4.63,1,1

4,7,3,7.23,2,2

4,7,1,5.06,3,3

4,8,2,8.75,1,1

4,8,3,7.59,2,2

4,8,1,4.82,3,3

5,9,3,7.25,1,1

5,9,1,7.88,2,3

5,9,2,9.02,3,1  
 5,10,3,5,1,1  
 5,10,1,7.84,2,3  
 5,10,2,7.79,3,1  
 6,11,1,4.63,1,1  
 6,11,2,6.77,2,1  
 6,11,3,5.72,3,2  
 6,12,1,3.87,1,1  
 6,12,2,7.62,2,1  
 6,12,3,6.74,3,2

。 자료(sas) :

y : 대조약, 시험약에 따른 AUC  
 group : sequence(1,2,3,4,5,6)  
 tmt : formulation ( 1=reference, 2=test 1, 3= test 2 )  
 period(1,2,3)  
 subject (n1=n2=n3=n4=n5=n6=2 )  
 carry

group은 여섯 개의 그룹으로 subjecct 는 각각 한 그룹 당 2명씩 배치, y값은 AUC 값을, period값은 1,2,3 으  
 로 지정한다. 앞 절에서의 디자인과 다른 점은 tmt가 referece formulation을 1로, 첫 번째 test formulation을  
 2로, 두 번째 test formulation을 3으로 세가지가 되었다는 것이다.

앞으로 이 절에서 사용되는 자료는 group,subject,tmt,y,period, carry(sas 입력 data 명) 순으로 .csv 형태로 c  
 의 work 폴더에 purich 라는 이름으로 저장되어 있다하고,(즉, c:/work/purich.csv) infile문을 통해서 자료를  
 불러들일 것이다.

위와 같이 입력된 데이터로 다음과 같은 프로그램들을 통해서 그래프와 신뢰구간 등의 다양한 결과를 얻을 수  
 있다.

### 3.6.3 잔류효과가 존재하는 경우

#### 3.6.3.1 프로그램

```
Libname out 'c:\work';  
data two;  
  infile 'c:\work\purich.csv' delimiter=',';  
  input group subject tmt y period carry;  
run;  
  
proc sort data=two;  
  by group subject tmt;  
  
proc mixed data=two;  
class tmt period subject group carry;  
model y=group period tmt carry;  
random subject(group);  
lsmeans tmt/pdiff cl alpha=0.1;  
estimate 'T vs. R' tmt -1 1/cl alpha=0.1;  
run;
```

#### 3.6.3.2 프로그램 설명

앞 절에서 나왔던 시험약이 한가지였던 경우와 프로그램은 동일하다.  
위의 프로그램으로 인한 결과는 다음과 같다.

#### The Mixed Procedure

#### Model Information

Data Set	WORK.TWO
Dependent Variable	y
Covariance Structure	Variance Components
Estimation Method	REML
Residual Variance Method	Profile
Fixed Effects SE Method	Model-Based

Degrees of Freedom Method Containment

#### Class Level Information

Class	Levels	Values
tmt	3	1 2 3
period	3	1 2 3
subject	12	1 2 3 4 5 6 7 8 9 10 11 12
group	6	1 2 3 4 5 6
carry	3	1 2 3

#### Dimensions

Covariance Parameters	2
Columns in X	16
Columns in Z	12
Subjects	1
Max Obs Per Subject	36
Observations Used	36
Observations Not Used	0
Total Observations	36

#### Iteration History

Iteration	Evaluations	-2 Res Log Like	Criterion
0	1	105.30914987	
1	1	100.16679881	0.00000000

#### The Mixed Procedure

Convergence criteria met.

Covariance Parameter Estimates	
Cov Parm	Estimate

subject(group)	1.1377
Residual	1.1320

#### Fit Statistics

-2 Res Log Likelihood	100.2
AIC (smaller is better)	104.2
AICC (smaller is better)	104.7
BIC (smaller is better)	105.1

#### Type 3 Tests of Fixed Effects

Effect	Num	Den	F Value	Pr > F
	DF	DF		
group	5	6	1.66	0.2756
period	2	18	1.32	0.2926
tmt	2	18	3.69	0.0454
carry	2	18	1.21	<b>0.3204</b>

잔류효과의 존재여부를 검정하는 유의확률(p-value)값은 0.3204이다.

이 결과는 (Chow & Liu, 2000) p315 에 제시된 값과 일치한다.

#### Estimates

Label	Estimate	Standard	DF	t Value	Pr >  t	Alpha	Lower	Upper
		Error						
T vs. R	1.2721	0.4856	18	2.62	0.0174	0.1	0.4300	2.1142

#### The Mixed Procedure

##### Least Squares Means

Effect	tmt	Estimate	Standard	DF	t Value	Pr >  t	Alpha	Lower	Upper
			Error						

tmt	1	6.0142	0.4349	18	13.83	<.0001	0.1	5.2600	6.7683
tmt	2	7.2863	0.4861	18	14.99	<.0001	0.1	6.4433	8.1292
tmt	3	6.3471	0.4861	18	13.06	<.0001	0.1	5.5041	7.1900

#### Differences of Least Squares Means

Effect	tmt	_tmt	Estimate	Standard Error	DF	t Value	Pr >  t	Alpha	Lower	Upper
tmt	1	2	-1.2721	0.4856	18	-2.62	0.0174	0.1	<b>-2.1142</b>	<b>-0.4300</b>
tmt	1	3	-0.3329	0.4856	18	-0.69	0.5017	0.1	<b>-1.1750</b>	<b>0.5092</b>
tmt	2	3	0.9392	0.4856	18	1.93	0.0690	0.1	<b>0.09707</b>	<b>1.7813</b>

여기서 얻어진 신뢰구간은 (Chow & Liu, 2000) p 320의 결과와 일치한다. 한가지 유의해야 할 것은, 부호가 뒤바뀐 결과가 있다는 것이다. 이는 위의 결과표에서 tmt 부분을 유의해서 보면 된다. 첫 번째 줄의 결과는 tmt 1(즉,reference. 대조약)- tmt 2(즉, test 1) 에 대한 신뢰구간이고, 두 번째 줄의 결과는 tmt1 - tmt3(test2) 의 ,마지막은 tmt2 - tmt3 에 대한 신뢰구간을 보여주고 있다. 따라서 세 번째는 (Chow & Liu, 2000)과 동일하고 첫 번째와 두 번째의 경우는 역으로 생각해보면 될 것이다. 즉, tmt2 - tmt1 에 대한 신뢰구간을 구하고 싶다면 (-2.1142, -0.4300)을 부호를 바꿔주면 (0.43, 2.1142) 를 얻을 수 있다.



### 3.6.4 잔류효과가 존재하지 않는 경우

#### 3.6.4.1 프로그램(PROC MIXED)

```
data two;  
  infile 'c:\work\purich.csv' delimiter=',';  
  input group subject tmt y period;  
run;
```

```
proc sort data=two;  
  by group subject tmt;
```

```
proc mixed data=two;  
class tmt period subject group;  
model y=group period tmt;  
random subject(group);  
lsmeans tmt/pdiff cl alpha=0.1;  
estimate 'T vs. R' tmt -1 1/cl alpha=0.1;  
run;
```

#### 3.6.4.2 프로그램 설명

##### The Mixed Procedure

##### Model Information

Data Set	WORK.TWO
Dependent Variable	y
Covariance Structure	Variance Components
Estimation Method	REML
Residual Variance Method	Profile
Fixed Effects SE Method	Model-Based
Degrees of Freedom Method	Containment

##### Class Level Information

Class	Levels	Values
tmt	3	1 2 3
period	3	1 2 3
subject	12	1 2 3 4 5 6 7 8 9 10 11 12
group	6	1 2 3 4 5 6

#### Dimensions

Covariance Parameters	2
Columns in X	13
Columns in Z	12
Subjects	1
Max Obs Per Subject	36
Observations Used	36
Observations Not Used	0
Total Observations	36

#### Iteration History

Iteration	Evaluations	-2 Res Log Like	Criterion
0	1	109.48337164	
1	1	104.26344958	0.00000000

#### The Mixed Procedure

Convergence criteria met.

#### Covariance Parameter Estimates

Cov Parm	Estimate
subject(group)	1.1297
Residual	1.1561

#### Fit Statistics

-2 Res Log Likelihood	104.3
AIC (smaller is better)	108.3
AICC (smaller is better)	108.8
BIC (smaller is better)	109.2

#### Type 3 Tests of Fixed Effects

Effect	Num	Den	F Value	Pr > F
	DF	DF		
group	5	6	1.65	0.2791
period	2	20	3.22	0.0615
tmt	2	20	2.85	0.0817

#### Estimates

Label	Estimate	Standard	DF	t Value	Pr >  t	Alpha	Lower	Upper
		Error						
T vs. R	1.0425	0.4390	20	2.37	0.0277	0.1	0.2854	1.7996

#### The Mixed Procedure

##### Least Squares Means

Effect	tmt	Estimate	Standard	DF	t Value	Pr >  t	Alpha	Lower	Upper
			Error						
tmt	1	6.0142	0.4364	20	13.78	<.0001	0.1	5.2614	6.7669
tmt	2	7.0567	0.4364	20	16.17	<.0001	0.1	6.3039	7.8094
tmt	3	6.4475	0.4364	20	14.77	<.0001	0.1	5.6948	7.2002

##### Differences of Least Squares Means

Effect	tmt	_tmt	Estimate	Standard	DF	t Value	Pr >  t	Alpha	Lower	Upper
				Error						

tmt	1	2	-1.0425	0.4390	20	-2.37	0.0277	0.1	<b>-1.7996</b>	<b>-0.2854</b>
tmt	1	3	-0.4333	0.4390	20	-0.99	0.3353	0.1	<b>-1.1904</b>	<b>0.3238</b>
tmt	2	3	0.6092	0.4390	20	1.39	0.1805	0.1	<b>-0.1479</b>	<b>1.3663</b>

여기서 얻어진 신뢰구간은 (Chow & Liu, 2000) p 320의 결과와 일치한다. 이 경우도 잔류효과가 존재하는 경우와 같이 신뢰구간을 원하는 대로 뒤집어서 생각해 볼 수 있다.

## 3.7 Tutorials for $2 \times 2$ crossover design

### 3.7.1 PROGRAM 설명

```
Libname out 'c:\work';  
data one;  
  infile 'c:\work\Wone.csv' delimiter=',' firstobs=2;  
  input group subject tmt y period;  
label y='Area Under Curve' tmt='Formulation';  
run;
```

infile문 아래 자료가 저장되어 있는 경로를 지정해 자료를 읽어들이고, 여기서 csv파일은 ‘,’로 구분되어 있으므로 구분자로 ‘,’를 쓴다. firstobs=2는 자료의 첫 번째 줄에는 변수명이 있고 두 번째 줄부터 실제 자료가 시작되는 경우 두 번째 줄부터 읽으라는 명령어이다. infile문에 명시된 순서대로 group, subject, tmt, y, period 순으로 읽어들이 이 변수들을 포함하는 one이라는 이름의 dataset을 생성한다.

그래프 출력시 y값 축에 y 대신 'Area Under Curve'가 나타나도록, tmt값 축에 tmt 대신 'Formulation' 이 나타나도록 label을 지정해준다.

```
data one;  
set one;  
y=log(y);
```

dataset one 의 y 변수들을 log변환 시켜준다. y변수가 log(y)값으로 대체된다. log변환을 원하지 않는 경우는 이 부분을 생략하거나 프로그램의 앞 뒤에 /\*와 \*/를 붙여 주석처리한다. 주석 처리 하는 경우 그 부분은 SAS 프로그램 상에서 초록색으로 변환된다.

```
proc sort data=one;  
  by group subject tmt;
```

생성된 dataset one을 group, subject, tmt 순으로 정렬시킨다.

```
footnote 'Formulation 1=R 2=T';
```

```

symbol i=join v=x r=50 c=black ;
axis1 label=(angle=90 h=1 f=zapfb);
axis2 label=(angle=90 h=1 f=zapfb) order=(70 to 90 by 5);
axis3 label=(f=zapfb) length=70 offset=(20,20) order=(1 to 2 by 1) ;
axis4 label=(f=zapfb) length=70 offset=(20,20) order=(1 to 2 by 1) ;

```

그래프 출력 모양을 지정해준다. order=(70 to 90 by 5)는 그래프 결과가 좀 더 보기 좋도록 그래프 축의 값을 지정해주는 옵션이다. 즉, 여기서는 axis2를 지정해준 축의 값은 70부터 90까지 보여주고 간격은 5로 준다는 것이다. 이 값은 일단 order 부분을 생각하고 그래프를 출력해본 후 보기 좋은 값으로 범위를 지정해주는 것이 좋다. 만일 70에서 90사이의 값이 아닌데 위의 범위를 그대로 줄 경우는 그래프에 아무 것도 나타나지 않는 경우가 있으니 order문을 줄 경우에는 값의 범위에 잘 맞추도록 주의해야 한다.

```

proc gplot data=one;
plot y*tmt=subject/description='y' vaxis=axis1 haxis=axis3 nolegend;
by group;run;quit;

```

PROC GLOT 프로시저를 이용해 y축을 y(AUC)값으로 x축을 tmt(Formulation) 값으로 하는 그래프를 출력한다. vaxis는 y축값을 haxis는 x값을 정의해준다. 즉 y축은 vaxis에서 지정해준 axis1의 형태로 x축은 haxis에서 지정해준 axis3의 형태로 출력된다.

```

data one_1;
set one;
if period=1 ;
if tmt=1;
id=1;

```

dataset one에서 period가 1이고 tmt가 1인 자료만을 포함하는 dataset one\_1을 생성하고, id 변수를 생성하여 1이라는 값을 부여한다.

```

proc means data=one_1 noprint;
var y;
output out=out_1 mean=my_1;

```

PROC MEANS 프로시저를 이용해 dataset one\_1의 기초통계량들을 구한다. noprint는 기초통계량을 output 창에 출력하지 않게하며, 이 옵션을 쓰면 출력 dataset이 명시되어야 한다. var 에 y를 지정해 y값을 분석하고, mean값을 갖는 my\_1이라는 변수를 생성한다. 이 변수를 포함하는 새로운 dataset out\_1이 생성된다.

```
data mean1;
set out_1;
id=1;
tmt=1;period=1;
y_m=my_1;
keep y_m tmt period id;
label y_m='Mean Area Under Curve';
```

my\_1 변수만을 포함하고 있는 dataset out\_1에 id=1, tmt=1, period=1의 값을 가지는 변수들을 생성하고 my\_1변수 값을 y\_m라는 변수에 지정하여 이 변수들을 포함하는 dataset mean1을 생성한다. 그래프 출력시 y값 축에 y\_m라는 변수명 대신 'Mean Area Under Curve'가 나타나도록 label을 지정해준다.

```
data one_2;
set one;
if period=1 ;
if tmt=2;
id=1;
```

dataset one에서 period가 1이고 tmt가 2인 자료만을 포함하는 dataset one\_2를 생성하고, id 변수를 생성하여 1이라는 값을 부여한다.

```
proc means data=one_2 noprint;
var y;
output out=out_2 mean=my_2;
```

PROC MEANS 프로시저를 이용해 dataset one\_2의 기초통계량들을 구한다. noprint는 기초통계량을 output 창에 출력하지 않게하며, 이 옵션을 쓰면 출력 dataset이 명시되어야 한다. var 에 y를 지정해 y값을

분석하고, mean값을 갖는 my\_2라는 변수를 생성한다. 이 변수를 포함하는 새로운 dataset out\_2이 생성된다.

```
data mean2;
set out_2;
id=1;
tmt=2;period=1;
y_m=my_2;
keep y_m tmt period id;
```

my\_2 변수만을 포함하고 있는 dataset out\_2에 id=1, tmt=2, period=1의 값을 가지는 변수들을 생성하고 my\_2변수 값을 y\_m라는 변수에 지정하여 이 변수들을 포함하는 dataset mean2를 생성한다.

```
data one_3;
set one;
if period=2 ;
if tmt=1;
id=1;
```

```
proc means data=one_3 noprint;
var y;
output out=out_3 mean=my_3;
```

```
data mean3;
set out_3;
id=1;
tmt=1;period=2;
y_m=my_3;
keep y_m tmt period id;
```

```
data one_4;
set one;
if period=2 ;
if tmt=2;
```



```
id=1;
```

```
proc means data=one_4 noprint;
```

```
var y;
```

```
output out=out_4 mean=my_4;
```

```
data mean4;
```

```
set out_4;
```

```
id=1;
```

```
tmt=2;period=2;
```

```
y_m=my_4;
```

```
keep y_m tmt period id;
```

위의 mean1,2의 경우와 마찬가지로 dataset mean3,4를 생성한다.

```
data mean;
```

```
merge mean1 mean2 mean3 mean4;
```

```
by id tmt period;
```

dataset mean1, mean2, mean3, mean4를 merge를 이용하여 변수 id, tmt, period에 의해 병합하여 새로운 dataset mean을 생성해낸다.

```
length function text $8;
```

```
XSYS = '2'; YSYS = '2';
```

```
TEXT = tmt; SIZE =1;
```

```
if tmt=1 and period=1 then do;
```

```
y = y_m; x = period; function ='MOVE'; output; end;
```

```
if tmt=2 and period=1 then do;
```

```
y = y_m; x = period; function ='MOVE'; output; end;
```

```
y = y_m; x = period; function='DRAW'; output;
```

```
y = y_m; x = period; function='LABEL'; output;
```

그래프 선 위에 tmt가 1인지 2인지 값을 보여주기 위한 옵션이다.

```
proc gplot data=mean;
plot y_m*period=tmt/annotate=mean vaxis=axis2 haxis=axis4 nolegend;
run;quit;
```

PROC GLOT 프로시저를 이용해 y축을 y\_m(tmt, period에 따른 AUC의 평균)값으로 x축을 period 값으로 하는 그래프를 출력한다.

```
proc mixed data=one;
class tmt period subject group;
model y=group period tmt;
random subject(group);
lsmeans tmt/pdiff cl alpha=0.1;
estimate 'T vs. R' tmt -1 1/cl alpha=0.1;run;
```

신뢰구간을 구하기 위한 방법 중 PROC MIXED 프로시저를 이용한 것이다.

dataset one에 PROC MIXED 절차를 적용시키고 tmt,period,subject,group 변수를 class문에 지정해주고, model문에 고정변수 group,period, tmt 변수를 랜덤변수 subject를 random 문에 지정해준다. lsmeans는 이들의 최소제곱평균을 구해주고, 신뢰구간을 유의수준 0.1에서 즉, 90% 신뢰구간을 구해준다. estimate문을 이용해 tmt 간의 최소제곱평균의 차이와 그에 대한 신뢰구간을 유의수준 0.1에서 구해주고 그 값을 출력한다.

다음부터는 PROC GLM 프로시저를 적용하기 위한 단계이다.

```
proc means n noprint data=one;
by group;
var y;
output out=no n=n;
run;
```

PROC MEANS 절차를 이용해 dataset one의 기초통계량들을 구한다. 여기서 n은 계산에 사용된 관찰치의 개수를 출력하고, noprint는 기초통계량을 output 창에 출력하지 않게한다. 이 옵션을 쓰면 출력dataset이 명시되어야 한다. group 변수의 값을 기준으로 변수 y값을 분석한다. n값이 포함된 no라는 이름의 dataset을

생성한다. dataset no에는 그룹에 따른 n값이 생성된다.

```
data no;
    set no;
    n=n/2;
run;

proc sort data=no;
    by group;
```

각각의 그룹별 관찰치의 개수가 n으로 들어갔는데 이 n의 값은 각 그룹의 피험자의 수의 2배이므로(reference,test 두가지 투여에 따른 y값이 각각 다른 관측치로 들어간다.) 각 그룹별 피험자의 수를 n으로 지정해주기 위해서 2로 나누어 준 값을 n으로 대체한다.

dataset no를 그룹변수에 의해 정렬한다.

```
data no1;
    set no;
    if group=1;
    n1=n;
    id=1;
    keep n1 id;
```

```
proc sort data=no1;
    by id;
run;
```

dataset no에서 group이 1인 obs들만으로 새로운 dataset no1을 만든다. 이때, no에서의 n을 n1이라는 새로운 변수로 만들고, 1값을 갖는 id라는 새로운 변수를 만든다. 변수 n1과 id만을 남기고, id에 의해 정렬시킨다.

```
data no2;
    set no;
    if group=2;
```

```

n2=n;
id=1;
keep n2 id;

```

```

proc sort data=no2;
    by id;
run;

```

dataset no1과 같은 방법으로 no2를 만든다.

```

proc glm data=one outstat=tanova;
    class tmt period subject group;
    model y=group subject(group) period tmt;
        test h=group e=subject(group);
    lsmeans tmt /stderr pdiff out=lsmean1;
run;

```

PROC GLM을 사용하여 dataset one을 분산분석한다. 이때 결과값을 tanova로 저장한다.

분산분석하고자 하는 인자들을 class문으로 지정한다. model문으로 분석하고자하는 모형을 정하고, test문으로 carryover effect를 검정할 수 있도록 한다. lsmeans문으로 tmt의 수준들에 대한 최소제곱평균을 계산하여 tmt의 수준간 차이검정을 할 수 있도록 한다. lsmeans의 결과값은 lsmean1으로 저장한다.

```

data error;
    set tanova;
    if _SOURCE_='ERROR';
    mse=ss/df;
    sse=ss;
    id=1;
    keep id mse sse df;

```

```

proc sort data=error;
    by id;
run;

```

dataset tanova에서 \_SOURCE\_가 ERROR인 ss값을 sse로 지정해주어 sse라는 새로운 변수로 만들고, sse를 df로 나누어 mse를 만들어 새로운 dataset error를 만든다. 또한 1값을 갖는 id라는 변수를 만든다. 여기서 id, mse, sse와 df 변수만을 남기고 나머지 모든 변수를 drop한 후, dataset error를 id변수에 의해 정렬시킨다.

```
data lsmean11;
    set lsmean1;
    if tmt=1;
    my1=lsmean;
    id=1;
    keep id my1;
```

```
proc sort data=lsmean11;
    by id;
```

dataset lsmean1에서 변수 tmt값이 1을 갖는 obs만으로(즉,reference formulation인 경우의 관측치 만으로) 새로운 dataset lsmean11을 만든다. 이때 lsmean값을 my1이라는 새로운 변수에 저장하고, id와 my변수만을 남기고 다른 변수는 모두 drop한 후, 변수 id에 의해 정렬시킨다.

```
data lsmean12;
    set lsmean1;
    if tmt=2;
    my2=lsmean;
    id=1;
    keep id my2;
```

```
proc sort data=lsmean12;
    by id;run;
```

lsmean11과 같은 방법으로 dataset lsmean12을 생성한다.

```
data twoside;
    merge lsmean11 lsmean12 no1 no2 error;
    by id;
```

```

spool=mse*0.5*((1/n1)+(1/n2));
se=sqrt(spool);
diff=my2-my1;
t=diff/se;
ct=tinv(0.05,df,0);
absct=abs(ct);
lower=diff-absct*se;
upper=diff+absct*se;
plower=((lower/my1)+1)*100;
pupper=((upper/my1)+1)*100;
p=2*(1-probt(abs(t),df));

run;

```

```
proc print data=twoside;run;
```

이제 최종적으로 Classic Confidence Interval을 만들기위한 dataset twoside를 만든다.

먼저 만들어놓은 dataset lsmean11, lsmean12, no1, no2 그리고 error을 id에 의해 merge 시킨다. 또한 interval을 구하기위한 새로운 변수들을 생성해낸다.

PROC PRINT 절차를 통해서 결과값을 출력하면, 아래와 같이 Classic Confidence Interval의 계산과정의 값들과, 상한과 하한 값이 계산되어 나온다. 앞의 PROC MIXED의 결과와 일치한다.

다음부터는 비모수적 방법을 적용하기 위한 절차이다.

```

data one1;
set one;
if period=1;
y1=y;
drop period y;

```

현재 one이라는 이름의 dataset안에 있는 변수는 group, subject, tmt, y, period이고, group,subject,tmt 순으로 정렬되어 있다. 이 dataset 에서 period가 1인 경우의 y값을 y1이라는 변수에 저장하고, period와 y변수를 제외한 나머지 변수들로 새로운 dataset one1을 생성한다.

```

data one2;
set one;
if period=2;
y2=y;
drop period y;

```

위와 같은 식으로 period가 2인 경우의 y값을 y2로 하여 새로운 변수를 생성하고 period와 y변수를 제외한 나머지 변수들로 새로운 dataset one2를 생성한다.

```

proc sort data=one1;
by group subject;

```

```

proc sort data=one2;
by group subject;

```

dataset one1과 one2를 group, subject 변수 순으로 정렬한다.

```

data work;
merge one1 one2;
by group subject;
id=1;

```

```

proc sort data=work;
by id group subject;

```

dataset one1과 one2를 group 변수와 subject 변수에 의해 병합하고, id변수를 값 1로 하는 새로운 dataset work를 생성한다. Id,group,subject 변수에 따라서 정렬한다.

이와 같은 일련의 과정을 통해서 dataset one에서 period값을 제거하고 y값을 period에 따라서 y1,y2로 구분한 dataset work가 생성된다. 여기서 tmt 값은 period가 2일 때의 tmt값으로 지정된다

```

data work;

```

```
merge work lsmean11;by id;
```

lsmean11에는 my1과 id변수가 포함되어 있다. 이 lsmean11과 work dataset을 병합하면 이전의 work dataset에 my1(tmt가 1인 경우의 lsmean 값) 82.5594 값이 포함된 새로운 dataset work가 생성된다.

```
w1=.;  
w2=.;  
if group=2 then w1=y1;  
if group=1 then w1=y2;  
if group=2 then w2=y2;  
if group=1 then w2=y1;
```

새로운 변수 w1과 w2를 생성한다.

만일 group이 2(TR)인 경우에는 y1값이 w1값으로 y2값이 w2로 지정되고,  
group 1(RT)인 경우에는 y2값이 w1으로 y1값이 w2로 지정된다.

```
diffw=w1-w2;  
sumw=w1+w2;  
d=(y2-y1);
```

diffw는 w1-w2로 formulation이 T일 때의 y값에서 R일 때의 y값을 차감해주는 값으로 설정된다. d의 경우는 y2에서 y1을 즉, period가 2일 때의 y값에서 period가 1 일 때의 y값을 차감해주는 값으로 설정된다.

```
d1=.;  
if group=1 then d1=d+(2*(0.2*my1));  
if group=2 then d1=d;  
d12=d1/2;  
d2=.;  
if group=1 then d2=d-(2*(0.2*my1));  
if group=2 then d2=d;  
d22=d2/2;
```

d1이라는 변수를 새로 생성한다. group이 1인 경우에는 d+(2\*(0.2\*my1))이 d1값으로 지정되고, group이 2인



경우는 d값이 d1으로 지정된다.

d12변수는 d1을 2로 나누어준 값으로 지정된다.

d2라는 변수를 새로 생성한다. group이 1인 경우는  $d - (2 * (0.2 * my1))$ 이 d2값으로 지정되고, group이 2인 경우는 d값이 d2로 지정된다.

d22변수는 d2변수를 2로 나누어준 값으로 지정된다.

```
t=y1+y2;
```

```
p=.
```

```
if group=1 then p=d;
```

```
if group=2 then p=-d;
```

y1값과 y2값을 더한 것이 t변수로 group1인 경우 d값이 group2인 경우 -d값이 p 변수로 지정된다.

```
proc sort data=work;
```

```
by group;
```

group별로 정렬한다.

```
proc npar1way wilcoxon data=work;
```

```
class group;
```

```
var d d12 d22 p t;
```

```
proc print;run;
```

group변수에 따라서 변수 d,d12,d22,p,t에 대해서 wilcoxon 비모수 검정을 한다.

다음부터는 정규성 검정을 위한 절차이다.

```
data one;
```

```
merge one no;
```

```
by group;
```

```
proc sort data=one;
```

```
by group subject tmt;
```

dataset one과 dataset no를 group 변수에 의해서 병합하여 새로운 dataset one을 생성하고, group,subject,tmt 변수에 의해서 정렬시킨다.

```
proc glm data=one;
    class tmt period subject group;
    model y=group subject(group) period tmt;
        test h=group e=subject(group);
        means tmt period group;
    lsmeans tmt period/stderr pdiff;
    lsmeans group/stderr pdiff e=subject(group);
run;
```

```
proc glm data=one outstat=tanova;
class tmt period subject;
model y=subject period tmt;
output out=pred p=yhat r=resid student=stresid;
means tmt period;
lsmeans tmt period/stderr pdiff out=lsmean1;
```

```
proc means n noprint data=one;
by group subject;
var y;
output out=sum sum=ysum;
```

```
proc glm data=sum;
model ysum=group;
output out=predsum p=ysumhat r=residsum student=residsub;
```

```
data pred;set pred; if period=1;
```

```
proc rank out=rpred normal=blom data=pred;
var stresid;
```

```
ranks rankr;
```

```
proc sort data=rpred;by rankr;
```

```
proc plot data=rpred;
```

```
plot stresid*yhat;
```

```
plot stresid*rankr;
```

```
proc univariate normal plot data=pred;
```

```
var stresid;
```

intrasubject의 정규성을 검정하기 위한 정규확률도를 그리고 Shapiro-wilk test를 하기 위해서 PROC UNIVARIATE 프로시저를 사용하고 옵션에 normal을 지정해준다.

```
proc rank out=rpredsum normal=blom data=predsum;
```

```
var residsub;
```

```
ranks ranksum;
```

```
proc sort data=rpredsum; by ranksum;
```

```
proc plot data=rpredsum;
```

```
plot residsub*ranksum;
```

```
proc univariate normal plot data=predsum;
```

```
var residsub;
```

intersubject의 정규성을 검정하기 위한 정규확률도를 그리고 Shapiro-wilk test를 하기 위해서 PROC UNIVARIATE 프로시저를 사용하고 옵션에 normal을 지정해준다.

```
data rpred1; set rpred;
```

```
keep subject stresid;
```

```
proc sort data=rpred1; by subject;
```

```
data rpredsum; set rpredsum;
```

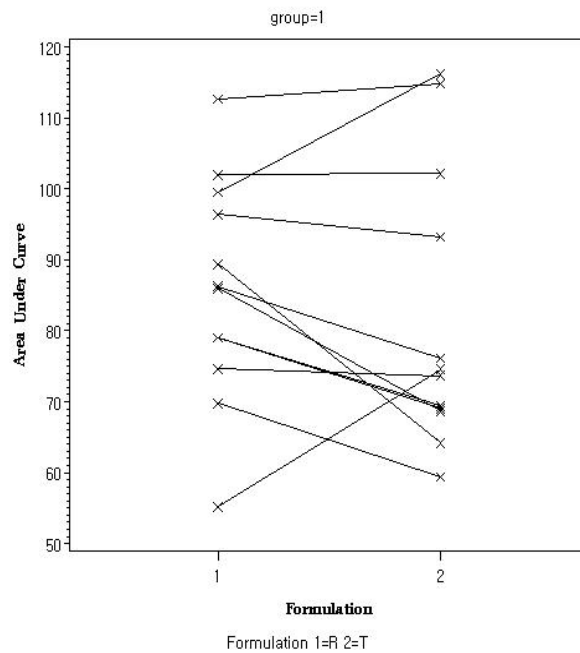
```
keep subject residsub;  
proc sort data=rpredsum; by subject;  
  
data indep; merge rpred1 rpredsum; by subject;  
  
proc corr pearson spearman data=indep;  
var stresid residsub;  
quit;
```

### 3.7.2 프로그램 결과

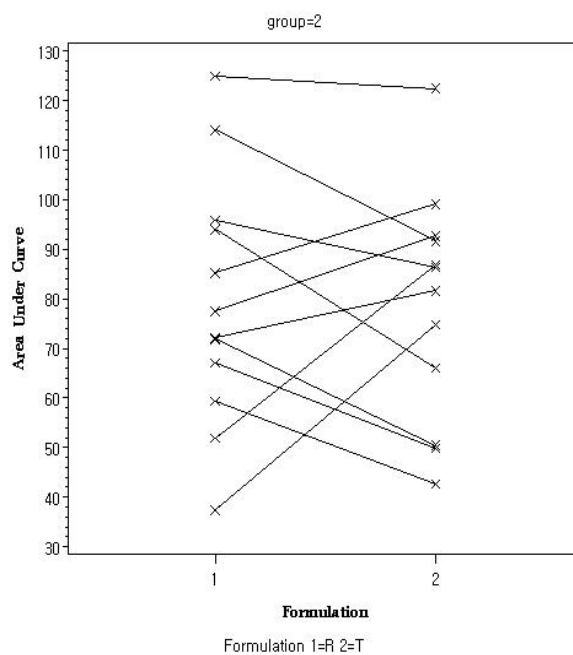
Chow & Liu(2000) p73의 자료에 대한 위의 프로그램의 결과를 다음에 제시하였다.

#### ▶ PROC GPLOT에 대한 결과

이 그래프는 (Chow & Liu, 2000) p 74의 그래프와 일치한다.



이 그래프는 (Chow & Liu, 2000) p 75의 그래프와 일치한다.



이 그래프는 (Chow & Liu, 2000) p 76의 그래프와 일치한다.



그래프 상에서 숫자 1은 대조약을 숫자 2는 시험약을 뜻한다. 따라서, 투여한 약에 따른 group간의 AUC 평균값을 비교할 수 있다.

## ▶ PROC MIXED에 대한 결과

### The Mixed Procedure

#### Model Information

Data Set	WORK.ONE
Dependent Variable	y
Covariance Structure	Variance Components
Estimation Method	REML
Residual Variance Method	Profile
Fixed Effects SE Method	Model-Based
Degrees of Freedom Method	Containment

#### Class Level Information

Class	Levels	Values
tmt	2	1 2
period	2	1 2
subject	24	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
group	2	1 2

#### Dimensions

Covariance Parameters	2
Columns in X	7
Columns in Z	24
Subjects	1
Max Obs Per Subject	48
Observations Used	48
Observations Not Used	0
Total Observations	48

#### Iteration History

Iteration	Evaluations	-2 Res Log Like	Criterion
0	1	405.20098358	
1	1	394.07438982	0.00000000

Convergence criteria met.

#### Covariance Parameter Estimates

Cov Parm	Estimate
subject(group)	284.82

Residual 167.25

#### Fit Statistics

-2 Res Log Likelihood	394.1
AIC (smaller is better)	398.1
AICC (smaller is better)	398.4
BIC (smaller is better)	400.4

#### The Mixed Procedure

#### Type 3 Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
group	1	22	0.37	0.5468
period	1	22	0.22	0.6474
tmt	1	22	0.38	0.5463

#### Estimates

Label	Estimate	Standard Error	DF	t Value	Pr >  t	Alpha	Lower	Upper
T vs. R	-2.2875	3.7333	22	-0.61	0.5463	0.1	<b>-8.6980</b>	<b>4.1230</b>

#### Least Squares Means

Effect	tmt	Estimate	Standard Error	DF	t Value	Pr >  t	Alpha	Lower	Upper
tmt	1	82.5594	4.3401	22	19.02	<.0001	0.1	75.1069	90.0119
tmt	2	80.2719	4.3401	22	18.50	<.0001	0.1	72.8194	87.7244

#### Differences of Least Squares Means

Effect	tmt	_tmt	Estimate	Standard Error	DF	t Value	Pr >  t	Alpha	Lower	Upper
tmt	1	2	2.2875	3.7333	22	0.61	0.5463	0.1	-4.1230	8.6980

▶ PROC MIXED의 SAS 결과에서 Estimates 부분의 Lower 와 Upper 에 제시된 값이 (  $\mu_R$  ) 에 대



한 90% 신뢰구간의 하한, 상한 값이다. 즉, 위의 결과에서 보면 (  $\mu - 1.645 \sigma$  ) 에 대한 90% 신뢰구간 값은 ( -8.6980 , 4.1230 )이다.

이 데이터는 로그변환을 해주지 않았으나 만일 로그변환을 해주었다면 위에 제시된 신뢰구간 값 상한, 하한 값에 exponential을 취해준다.

이 신뢰구간 값은 (Chow & Liu, 2000) p 106의 결과와 일치한다.

#### ▶ PROC GLM에 대한 결과

The GLM Procedure																																											
Class Level Information																																											
Class	Levels	Values																																									
tmt	2	1	2																																								
period	2	1	2																																								
subject	24	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24																		
group	2	1	2																																								
Number of observations																																											
48																																											
The GLM Procedure																																											
Dependent Variable: y    Area Under Curve																																											
Source	DF	Sum of Squares		Mean Square		F Value		Pr > F																																			
Model	25	16586.24750		663.44990		3.97		0.0009																																			
Error	22	3679.42953		167.24680																																							

Corrected Total 47 20265.67703

R-Square 0.818440  
 Coeff Var 15.88441  
 Root MSE 12.93239  
 y Mean 81.41563

Source	DF	Type I SS	Mean Square	F Value	Pr > F
group	1	276.00021	276.00021	1.65	0.2123
subject(group)	22	16211.48870	736.88585	4.41	0.0005
period	1	35.96672	35.96672	0.22	0.6474
tmt	1	62.79187	62.79187	0.38	0.5463

Source	DF	Type III SS	Mean Square	F Value	Pr > F
group	1	276.00021	276.00021	1.65	0.2123
subject(group)	22	16211.48870	736.88585	4.41	0.0005
period	1	35.96672	35.96672	0.22	0.6474
tmt	1	62.79188	62.79188	0.38	0.5463

Tests of Hypotheses Using the Type III MS for subject(group) as an Error Term

Source	DF	Type III SS	Mean Square	F Value	Pr > F
group	1	276.0002083	276.0002083	0.37	0.5468

The GLM Procedure  
 Least Squares Means

		H0:LSMean1=		LSMean2	
tmt	y LSMEAN	Standard Error	H0:LSMEAN=0 Pr >  t	Pr >  t	
1	82.5593750	2.6398137	<.0001	0.5463	
2	80.2718750	2.6398137	<.0001		

Obs my1 id my2 n1 n2 DF mse sse spool se

```

1 82.5594 1 80.2719 12 12 22 167.247 3679.43 13.9372 3.73326
diff t ct absct lower upper plower pupper p
-2.2875 -0.61274 -1.71714 1.71714 -8.69805 4.12305 89.4645 104.994 0.54633

```

이 신뢰구간 값은 (Chow & Liu, 2000) p 106의 결과와 일치한다. 앞서 언급한 PROC MIXED 결과와도 일치한다.

#### ▶ PROC NPAR1WAY에 대한 결과 (비모수적 방법)

The NPAR1WAY Procedure

**Wilcoxon Scores (Rank Sums) for Variable d12**  
Classified by Variable group

group	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	12	207.0	150.0	17.320508	17.250
2	12	93.0	150.0	17.320508	7.750

#### Wilcoxon Two-Sample Test

Statistic	207.0000
Normal Approximation	
Z	3.2620
One-Sided Pr > Z	0.0006
Two-Sided Pr >  Z	0.0011
t Approximation	
One-Sided Pr > Z	0.0017
Two-Sided Pr >  Z	0.0034

Z includes a continuity correction of 0.5.

#### Kruskal-Wallis Test

Chi-Square	10.8300
DF	1
Pr > Chi-Square	0.0010

The NPAR1WAY Procedure

Wilcoxon Scores (Rank Sums) for Variable d22

Classified by Variable group

group	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	12	91.0	150.0	17.320508	7.583333
2	12	209.0	150.0	17.320508	17.416667

Wilcoxon Two-Sample Test

Statistic 91.0000

Normal Approximation

Z -3.3775

One-Sided Pr < Z 0.0004

Two-Sided Pr > |Z| 0.0007

t Approximation

One-Sided Pr < Z 0.0013

Two-Sided Pr > |Z| 0.0026

Z includes a continuity correction of 0.5.

Kruskal-Wallis Test

Chi-Square 11.6033

DF 1

Pr > Chi-Square 0.0007

이 신뢰구간 값은 (Chow & Liu, 2000) p 113의 결과와 일치한다. 즉,  $R_L$  값은 207,  $R_U$  값은 91이다. 2.6.1에서 언급했듯이, 만일  $W_L > W(1 - \alpha)$ 이고,  $W_U < w(\alpha)$ 이면 생물학적으로 동등하다 결론을 내릴 수 있다. 이 때,  $W(1 - \alpha)$  값과  $w(\alpha)$  값은 Chow & Liu (2000) p 551부터 있는 Appedix A.5의 표의 값을 참조하고,  $W_U = R_U - \frac{1}{2} n_1(n_1 + 1)$ ,  $W_L = R_L - \frac{1}{2} n_1(n_1 + 1)$  이다.

▶ 정규성 검정(나머지 프로그램)

The GLM Procedure

Class Level Information

Class	Levels	Values
tmt	2	1 2
period	2	1 2
subject	24	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
group	2	1 2

Number of observations 48

The GLM Procedure

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	25	16586.24750	663.44990	3.97	0.0009
Error	22	3679.42953	167.24680		
Corrected Total	47	20265.67703			

R-Square	Coeff Var	Root MSE	y Mean
0.818440	15.88441	12.93239	81.41563

Source	DF	Type I SS	Mean Square	F Value	Pr > F
group	1	276.00021	276.00021	1.65	0.2123
subject(group)	22	16211.48870	736.88585	4.41	0.0005
period	1	35.96672	35.96672	0.22	0.6474
tmt	1	62.79187	62.79187	0.38	0.5463

Source	DF	Type III SS	Mean Square	F Value	Pr > F
group	1	276.00021	276.00021	1.65	0.2123
subject(group)	22	16211.48870	736.88585	4.41	0.0005
period	1	35.96672	35.96672	0.22	0.6474
tmt	1	62.79188	62.79188	0.38	0.5463

Tests of Hypotheses Using the Type III MS for subject(group) as an Error Term

Source	DF	Type III SS	Mean Square	F Value	Pr > F
group	1	276.0002083	276.0002083	0.37	0.5468

The GLM Procedure

Level of tmt	N	Mean	Std Dev
1	24	82.5593750	20.7974547
2	24	80.2718750	21.1152123

Level of period	N	Mean	Std Dev
1	24	82.2812500	19.7083990
2	24	80.5500000	22.1614898

Level of group	N	Mean	Std Dev
1	24	83.8135417	17.5441028
2	24	79.0177083	23.6922106

The GLM Procedure  
Least Squares Means

tmt	y LSMEAN	Standard Error	H0:LSMean1=	
			H0:LSMEAN=0 Pr >  t	LSMean2 Pr >  t
1	82.5593750	2.6398137	<.0001	0.5463
2	80.2718750	2.6398137	<.0001	

period	y LSMEAN	Standard Error	H0:LSMean1=	
			H0:LSMEAN=0 Pr >  t	LSMean2 Pr >  t
1	82.2812500	2.6398137	<.0001	0.6474
2	80.5500000	2.6398137	<.0001	

The GLM Procedure  
Least Squares Means

Standard Errors and Probabilities Calculated Using the Type III MS for subject(group) as an Error Term

group	y LSMEAN	Standard Error	H0:LSMean1=	
			H0:LSMEAN=0 Pr >  t	LSMean2 Pr >  t

1	83.8135417	5.5410809	<.0001	0.5468
2	79.0177083	5.5410809	<.0001	

The GLM Procedure

Class Level Information

Class	Levels	Values
tmt	2	1 2
period	2	1 2
subject	24	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

Number of observations 48

The GLM Procedure

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	25	16586.24750	663.44990	3.97	0.0009
Error	22	3679.42953	167.24680		
Corrected Total	47	20265.67703			

R-Square	Coeff Var	Root MSE	y Mean
0.818440	15.88441	12.93239	81.41563

Source	DF	Type I SS	Mean Square	F Value	Pr > F
subject	23	16487.48891	716.84734	4.29	0.0006
period	1	35.96672	35.96672	0.22	0.6474
tmt	1	62.79187	62.79187	0.38	0.5463

Source	DF	Type III SS	Mean Square	F Value	Pr > F
subject	23	16487.48891	716.84734	4.29	0.0006
period	1	35.96672	35.96672	0.22	0.6474
tmt	1	62.79188	62.79188	0.38	0.5463

The GLM Procedure

Level of tmt	N	Mean	Std Dev
1	24	82.5593750	20.7974547
2	24	80.2718750	21.1152123

Level of period	N	Mean	Std Dev
1	24	82.2812500	19.7083990
2	24	80.5500000	22.1614898

The GLM Procedure  
Least Squares Means

tmt	y LSMEAN	Standard Error	H0:LSMean1=	
			H0:LSMEAN=0 Pr >  t	LSMean2 Pr >  t
1	82.5593750	2.6398137	<.0001	0.5463
2	80.2718750	2.6398137	<.0001	

period	y LSMEAN	Standard Error	H0:LSMean1=	
			H0:LSMEAN=0 Pr >  t	LSMean2 Pr >  t
1	82.2812500	2.6398137	<.0001	0.6474
2	80.5500000	2.6398137	<.0001	

The GLM Procedure

Number of observations 24

The GLM Procedure

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	552.00042	552.00042	0.37	0.5468
Error	22	32422.97740	1473.77170		



Corrected Total	23	32974.97781
-----------------	----	-------------

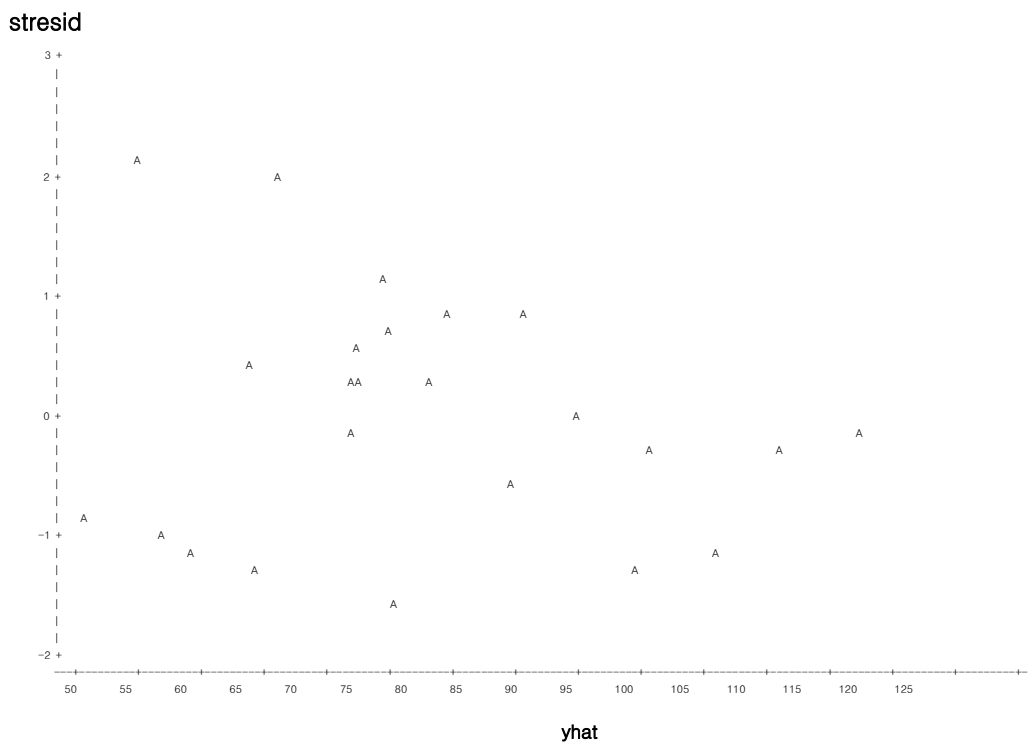
R-Square	Coeff Var	Root MSE	ysum Mean
0.016740	23.57639	38.38973	162.8313

Source	DF	Type I SS	Mean Square	F Value	Pr > F
group	1	552.0004167	552.0004167	0.37	0.5468

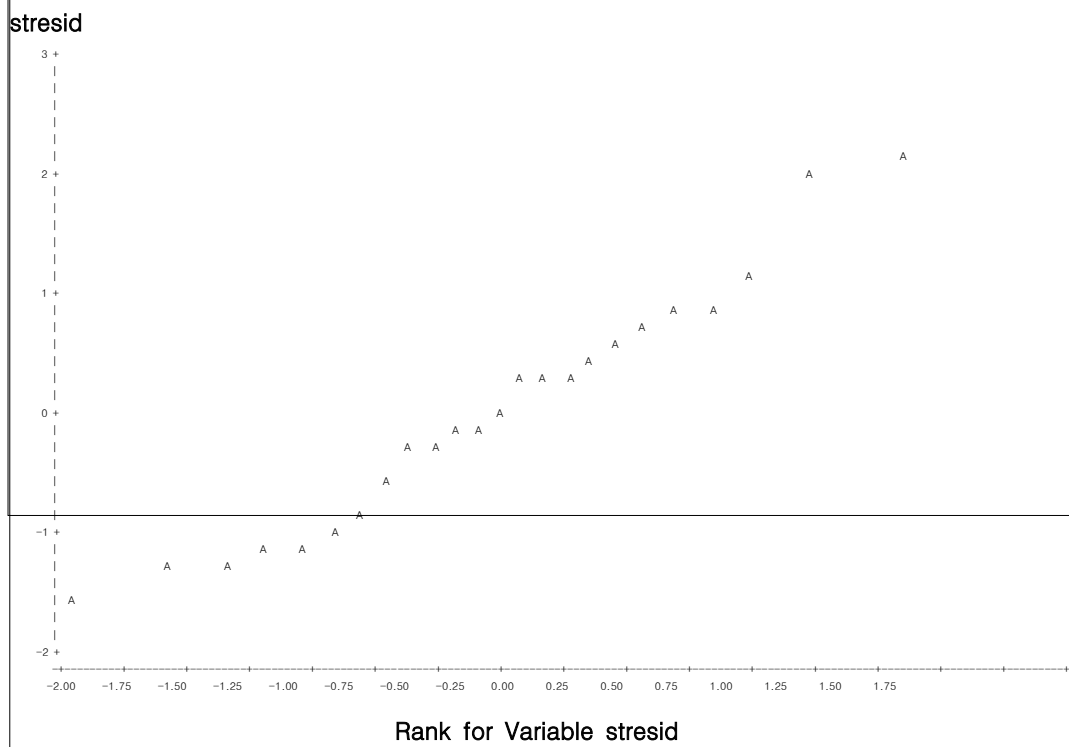
Source	DF	Type III SS	Mean Square	F Value	Pr > F
group	1	552.0004167	552.0004167	0.37	0.5468

Parameter	Estimate	Standard Error	t Value	Pr >  t
Intercept	177.2187500	24.78046694	7.15	<.0001
group	-9.5916667	15.67254340	-0.61	0.5468

Plot of stresid\*yhat. Legend: A = 1 obs, B = 2 obs, etc.



Plot of stresid\*rankr. Legend: A = 1 obs, B = 2 obs, etc.



## UNIVARIATE 프로시저

변수: stresid

### 적률

N	24	가중합	24
평균	0	관측치 합	0
표준 편차	1.02150784	분산	1.04347826
왜도	0.40547975	첨도	-0.3198507
제곱합	24	수정 제곱합	24
변동계수	.	표준 오차 평균	0.20851441

### 기본 통계량의 측정

위치                      변동성

평균	0.00000	표준 편차	1.02151
중간값	-0.08102	분산	1.04348
최빈값	.	범위	3.73346
		사분위 범위	1.60118

위치모수 검정:  $\mu_0=0$

검정	--통계량---		-----p값-----	
스튜던트의 t	t	0	Pr >  t	1.0000
부호	M	-1	Pr >=  M	0.8388
부호 순위	S	-7	Pr >=  S	0.8464

### 정규성 검정

검정                      ----통계량----                      -----p값-----

**Shapiro-Wilk    W   0.957632   Pr < W   0.3927**

Kolmogorov-Smirnov	D	0.108651	Pr > D	>0.1500
Cramer-von Mises	W-Sq	0.035425	Pr > W-Sq	>0.2500
Anderson-Darling	A-Sq	0.301962	Pr > A-Sq	>0.2500

▶ 정규성 검정을 위해 결과에 제시된 Shapiro-Wilk 통계량을 확인한다. intrasubject 에 대한 Shapiro-Wilk test의 p-value값이 0.3927로 정규성 가정을 기각하지 않는다. 즉, 정규성이 어긋나지 않는다고 결론 내릴 수 있겠다. 좀 더 살펴보기 위해서 아래의 정규확률도도 함께 확인해본다.

UNIVARIATE 프로시저

변수: stresid

사분위수(정의 5)

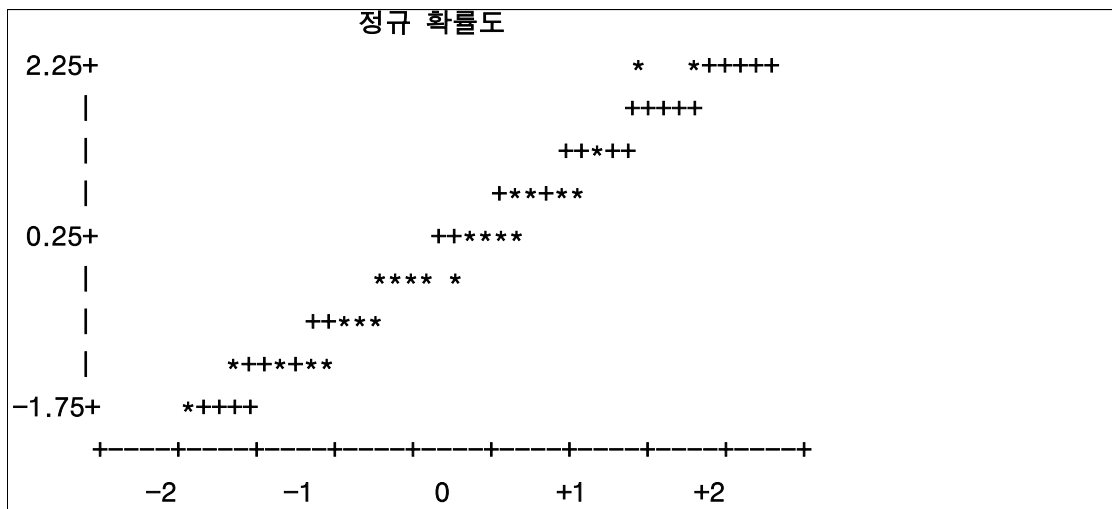
사분위수	추정값
100% Max	2.1719058
99%	2.1719058
95%	2.0277069
90%	1.2124838
75% Q3	0.6649565
50% Median	-0.0810226
25% Q1	-0.9362217
10%	-1.2431796
5%	-1.3374085
1%	-1.5615592
0% Min	-1.5615592

극값의 관측치

---최소값 부터---      ----최대값 부터---

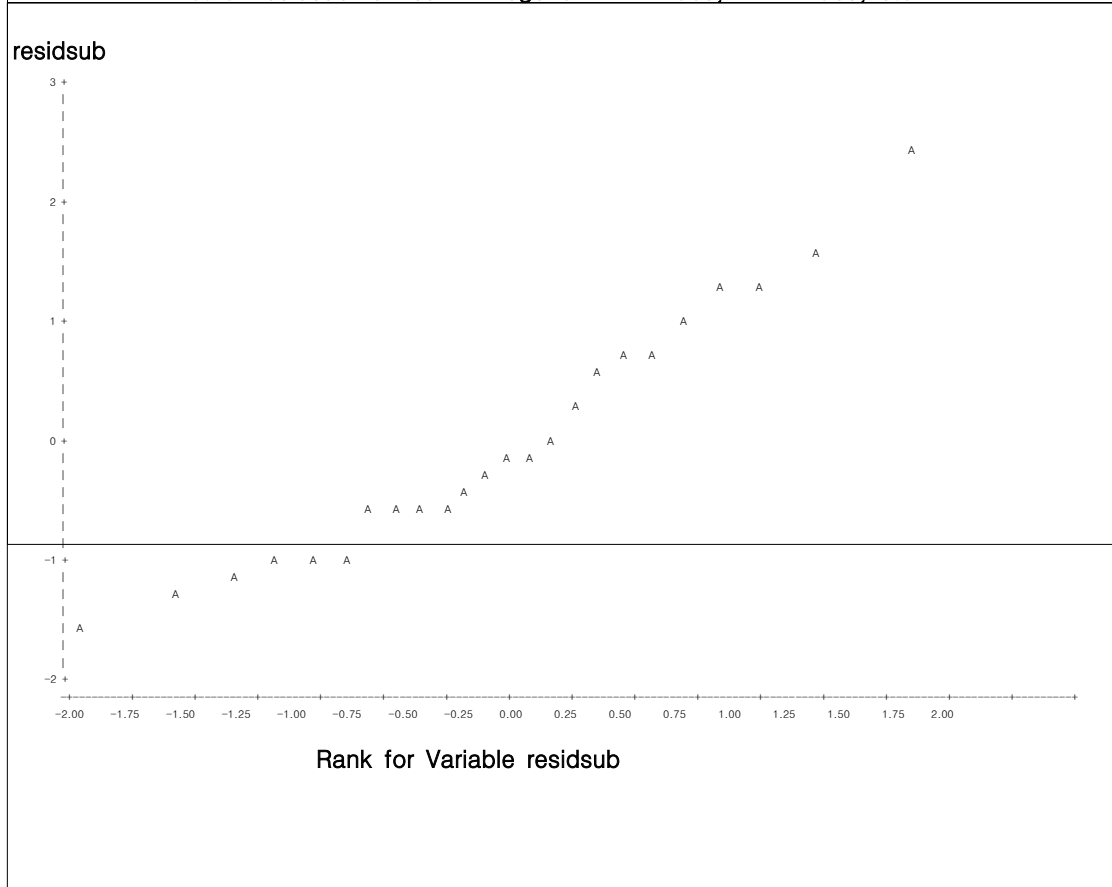
값	관측치	값	관측치
-1.56156	18	0.822719	20
-1.33741	12	0.899815	16
-1.24318	24	1.212484	11
-1.19178	17	2.027707	14
-1.18464	10	2.171906	13

줄기 잎	#	상자그림
2 02	2	
1		
1 2	1	
0 6889	4	+-----+
0 3344	4	+
-0 42210	5	*-----*
-0 995	3	+-----+
-1 3222	4	
-1 6	1	
-----+		



☒ ~intersubject에 대한 정규확률도이다. 크게 직선에서 벗어나지 않는 것으로 보아 역시 정규성이 크게 어긋나지 않는다고 결론 내릴 수 있겠다.

Plot of residsub\*ranksum. Legend: A = 1 obs, B = 2 obs, etc.



# UNIVARIATE 프로시저

변수: residsub

## 적률

N	24	가중합	24
평균	0	관측치 합	0
표준 편차	1.02150784	분산	1.04347826
왜도	0.63819595	첨도	-0.2050811
제곱합	24	수정 제곱합	24
변동계수	.	표준 오차 평균	0.20851441

## 기본 통계량의 측정

위치		변동성	
평균	0.00000	표준 편차	1.02151
중간값	-0.24764	분산	1.04348
최빈값	.	범위	3.95316
		사분위 범위	1.43879

위치모수 검정:  $\mu_0=0$

검정	통계량	p값
스튜던트의 t	t	0 Pr >  t  1.0000
부호	M	-2 Pr >=  M  0.5413
부호 순위	S	-6 Pr >=  S  0.8681

## 정규성 검정

검정      통계량      p값

**Shapiro-Wilk W 0.951602 Pr < W 0.2934**

Kolmogorov-Smirnov D 0.135191 Pr > D >0.1500

Cramer-von Mises W-Sq 0.073612 Pr > W-Sq 0.2433

Anderson-Darling A-Sq 0.430882 Pr > A-Sq >0.2500

▶ Intersubject 에 대한 Shapiro-Wilk test의 p-value값이 0.2934로 이도 역시 정규성 가정을 기각하지 않는다. 즉, 정규성이 어긋나지 않는다고 결론 내릴 수 있겠다. 좀 더 살펴보기 위해서 아래의 정규확률도도 함께 확인해본다.

UNIVARIATE 프로시저

변수: residsub

사분위수(정의 5)

사분위수	추정값
100% Max	2.432014
99%	2.432014
95%	1.630317
90%	1.309956
75% Q3	0.687031
50% Median	-0.247640
25% Q1	-0.751761
10%	-1.115767
5%	-1.247720
1%	-1.521150
0% Min	-1.521150

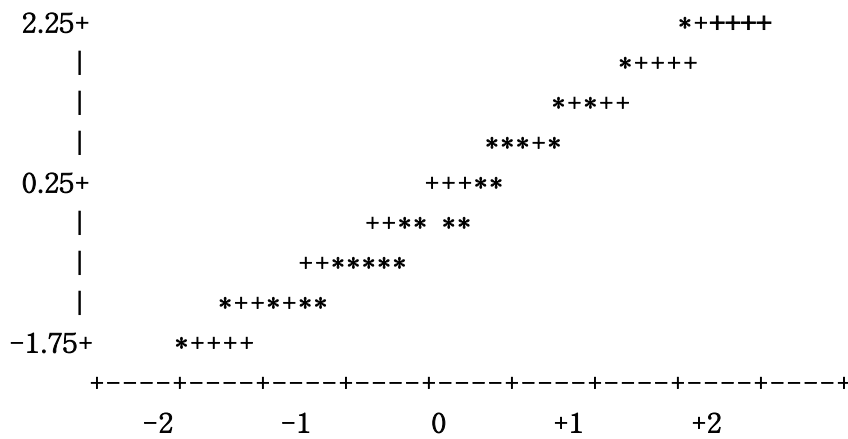
극값의 관측치

---최소값 부터---      ----최대값 부터---

값	관측치	값	관측치
-1.52115	23	0.991635	3
-1.24772	13	1.298846	24
-1.11577	22	1.309956	10
-1.04684	7	1.630317	9
-1.03052	12	2.432014	19

줄기 잎	#	상자그림
2 4	1	
1 6	1	
1 033	3	
0 677	3	+-----+
0 13	2	+
-0 4411	4	*-----*
-0 5555	4	+-----+
-1 21000	5	
-1 5	1	
-----+-----+-----+-----+		

정규확률도



▣ 정규확률도를 살펴보아도 역시 정규성이 위배되지 않는다.

만일, 정규성이 위배된다면 앞서 제시한 비모수적 방법(PROC NPAR1WAY)을 고려해 볼 수 있다.



## 제 4 장 연구개발목표 달성도 및 대외기여도

목표로 한 연구개발 목표는 모두 달성되었다. 개발된 SAS 프로그램을 이용하여 생물학적 동등성 평가에 더욱 고급화되고 정교한 통계 기법의 도입을 도입할 수 있게 되었고, 고급화되고 정교한 통계 기법을 구체적으로 구현해줄 수 있는 체계적인 프로그램을 확보하게 되었다. 이는 생물학적동등성시험 평가의 과학적 기반 보강하고, 생동성시험 결과 통계분석의 국제화를 이루게 되었다.

## 제 5 장 연구개발결과의 활용성과 및 계획

가. 계량적 성과 : 해당사항없음

전문학술지(편수)				학회발표(초록)		특 허(건수)				상품화	기타 (CD-ROM, 저서, 워크샷 개최 등)
국 외		국 내		국 외	국 내	국 외		국 내			
계제	투고중	계제	투고중			출원중	출원등록	출원중	출원등록		

나. 성과내용기술

개발된 SAS 프로그램을 이용하여 생물학적 동등성 평가에 더욱 고급화되고 정교한 통계 기법의 도입을 도입할 수 있게 되었고, 고급화되고 정교한 통계 기법을 구체적으로 구현해줄 수 있는 체계적인 프로그램을 확보하게 되었다.

다. 활용계획

1. 개발된 SAS 프로그램으로 보다 정확하게 생물학적 동등성 평가 가능
2. 생물학적 동등성만이 아니라 신약 개발과정 등에도 SAS 프로그램 사용의 필요성을 인식하게 함

## 제 6 장 참고문헌

[1] Chow & Liu (2000), "Design and Analysis of Bioavailability and Bioequivalence Studies "

[2] Jones & Kenward (2003), " Design and Analysis of Cross-Over Trials "

## 본문작성 요령

1. 본문의 순서는 가급적 장, 절, 1, 가, (1), (가), ①, ㉔ 등으로 유지하되, 각 기관의 형식을 준용하며,
  - 장은 HWP 16 point 고딕체
  - 절은 HWP 14 point 신명조체
  - 본문은 HWP 10 point 신명조체로 한다.단, 본문의 내용 중 중요부분은 고딕체로 사용할 수 있다.
2. 장은 원칙적으로 페이지를 바꾸어 시작한다.
3. 본문은 HWP 10 point에서 낱말간격(자간)은 0, 줄간격은 200% 이내로 한다.
4. 페이지 번호는 하단 중앙 아래에 HWP 10 point로 한다.
5. 각주(註)는 해당 페이지 하단에 HWP 9 point로 표기하며, 본문과 구분토록 한다.
6. 도(圖)와 표(表)는 어떤 프로그램으로 작성하여도 관계없으며, 도(圖)와 표명(票名)은 ( ) 내에 영문으로 표기한다.
7. 페이지 수는 편집순서 2의 제출문부터 시작하여 편집순서 4까지 기재한다.  
단, 삽입물이 있을 때에는 그 삽입물의 크기에 불문하고 1면을 한 페이지로 하여 일련번호를 붙인다.
8. 한글, 한문, 영문을 혼용한다.
9. 참고문헌(Reference) 인용의 경우 본문 중에 사용처를 반드시 표기한다.

※ 각 항목별 여백이 부족할 경우에는 ‘별지 작성’이라고 기재한 후 별도 첨부

## 총괄 연구과제 요약

과제 고유번호	자동부여		공개가능여부	
사업명	의약품동등성관리			
과제명	SAS를 이용한 생물학적동등성시험결과 통계분석			
연구책임자	성 명	강승호	주민등록번호	
	소속 기관명			
	전자우편		전화번호	

### ○ 연구목표 (400 ~600자)

생물학적 동등성을 평가함에 있어서 SAS를 도입하여 통계분석 방법을 향상시킴과 동시에 국내 생물학적 동등성 실험 결과의 국제적 공인을 도모하고자 한다. 이렇게 SAS의 도입으로 인하여 통계분석 방법의 선진화를 이루는데 있다.

### ○ 연구내용 (1000~1200자)

2 x 2 교차시험에 얻어지는 자료에 대한 생물학적 동등성 평가 통계 분석을 위한 SAS 프로그램을 개발하였다. 그 프로그램에는 정규성을 가정하는 모수적 방법뿐만이 아니라, 정규성 가정이 없을 때 사용할 수 있는 비모수적 방법도 포함시켰고, 정규성 가정을 체크할 수 있는 분석 방법도 포함시켰다. 또한 2 x 3 교차시험과 2 x 4 교차시험 자료 분석을 위한 SAS 프로그램을 개발하였다. 개발된 SAS 프로그램을 통계 비전문가도 사용할 수 있도록 사용 방법을 설명한 매뉴얼을 만들었다. SAS 프로그램을 개발한 후 기존의 생물학적 동등성시험 검정 프로그램인 K-BE와 상호비교하였고, 같은 결과를 얻을 수 있음을 확인하였다.

○ 연구성과(응용분야 및 활용범위포함) (400 ~600자)

개발된 SAS 프로그램을 이용하여 생물학적 동등성 평가에 더욱 고급화되고 정교한 통계 기법의 도입을 도입할 수 있게 되었고, 고급화되고 정교한 통계 기법을 구체적으로 구현해줄 수 있는 체계적인 프로그램을 확보하게 되었다. 활용계획으로는 첫째, 개발된 SAS 프로그램으로 보다 정확하게 생물학적 동등성 평가 가능하게 되었고, 둘째, 생물학적 동등성만이 아니라 신약 개발과정 등에도 SAS 프로그램 사용의 필요성을 인식시키게 되었다.

○ 총괄 참여연구원

성명	주민등록번호	성명	주민등록번호
강승호		임다혜	

<b>Keywords</b> (5개 내외)	<b>한글</b>	생물학적 동등성, SAS, 교차실험, 생물통계학
	<b>영문</b>	bioequivalence, SAS, crossover design, biostatistics