火曜4限 英語学研究 英語教育学専修コース 修士1年 リーブン

Chapter 11: Factorial ANOVA (PDF: pp.205-236)

Key terms of chapter 11

1. 分散分析:

2 つの平均の有意差の検定には t 検定. 3 つ以上の平均の有意差の検定の場合には分散分析.

手順:

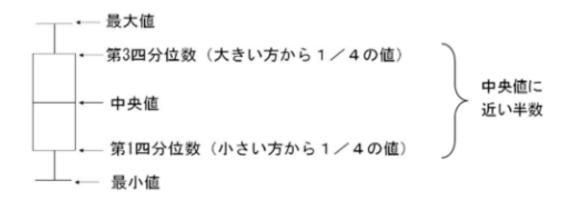
- 1).3つ(以上)の平均値のすべてか等しいかどうかを調べ、
- 2). 等しいときには有意差なしと結論する、
- 3). 等しくない場合(有意差ある)、どの平均対の間に差かあるのか調べる.

*A=B=C 有意差なし

*有意差あり: A と B、A と C, B と C を検定する.

2.Boxplot(箱ひげ図)

計測結果の集計表示手法として「箱ひげ図」を活用する。「箱ひげ図」は、ばらつきのあるデータをわかりやすく表現するための統計学のグラフであり、重要な5種の要約統計量である、最小値、第一四分数、中央値、第三四分数及最大値。



3.Level (水準)/ Factor (因子/要因)

観察される変数は数値で表現される。観察される変数と数値を対応させる基準はデータ水準と呼ばれ、数値の持つ意味には違いがある。データの値を変化させる原因のこと。 \rightarrow p.206のデータを見ると、gender、grade、picturet1と musict1は要因である。Genderという要因で male と female による。

4.Main effect (主効果) /Interaction(交互作用)

どちらも2要因以上の分散分析などで多く用いられます。

主効果は、あるひとつの要因が従属変数に影響を与えている場合、「要因 A の主効果が認められた」というような表現をします。また交互作用は、2 つ以上の要因によって、水準間で従属変数の表れ方に違いが現れるはたらきのことです。たいていの検定では先に交互作用を見て、次に主効果を見ます。

5.Independent variable(独立変数)/response variable(応答変数)

実験の対象となる変数(測定または観測する変数)は、応答変数や従属変数と呼ばれます。一方、応答に影響し、実験で測定されるその他の変数を、予測変数、説明変数、または独立変数と呼びます。

たとえば、ケーキのレシピの推奨される焼き時間を判断したり、新しい交配植物の世話の仕方を教えたりする場合が考えられます。

対象	考えられる予測変数	考えられる応答変数
ケーキのレシピ	焼き時間、オーブンの温度	ケーキの水分、ケーキの厚さ
植物の成長	光量、土壌のpH、水やりの頻度	葉の大きさ、植物の高さ

連続予測変数は共変量、カテゴリ予測変数は因子と呼ばれることがあります。 ケーキの実験では、共変量はさまざまなオーブン温度、因子は別々のオーブン となります。

通常は、予測変数をX軸に、応答変数をY軸にしてプロットを作成します。

6.Regression(回帰分析)

ある変数が他の変数とどのような相関関係にあるのかを推定する統計学的手法の一。原因となる変数 \mathbf{x} (説明変数)と、結果となる変数 \mathbf{y} (目的変数または被説明変数)の間に、回帰式 $\mathbf{y}=\mathbf{a}\mathbf{x}+\mathbf{b}$ と表される関係があるとすると、 \mathbf{x} 、 \mathbf{y} の観測値から最小二乗法を用いて \mathbf{a} 、が求められる。この回帰式をもとに将来予測や要因分析を行う。

この章で扱うデータは: Obarow, S. (2004). The impact of music on the vocabulary acquisition of kindergarten and first grade students. Unpublished Ph.D., Widener University, Chester, PA.

- 11.1 Numerical and Visual Summary of the Data, Including Means Plots
- ▶ Obarow.Story1.sav SPP データを読み込む → obarowStory1 として保存
- Rコンソールでのコマンド:

numSummary(obarowStory1[,c("gnsc1.1", "gnsc1.2")],
groups=obarowStory1\$treatment1, statistics=c("mean", "sd"))

Yes music Yes pics 1.882353 2.027386 17 0

解釈:

- ➤ このスドーリから 20 個単語がテストされたと想定し、gain scores からわかることは、どちらの状況においでも、多くの子供は勉強した単語数が少ない。実際に知った単語数が 1 1。
- ▶ 数値サマリからわかることは標準偏差が大きい、多くのケースの場合に平均より大きい。
- ▶ ビジュアル・サマリの場合は、グールプの差を見るために、ボックスプロットを使ってデータの分散をチェックするのはいい方法である。
- ▶ Figure11.1 は四つ異なるグループのボックスプロを表している。

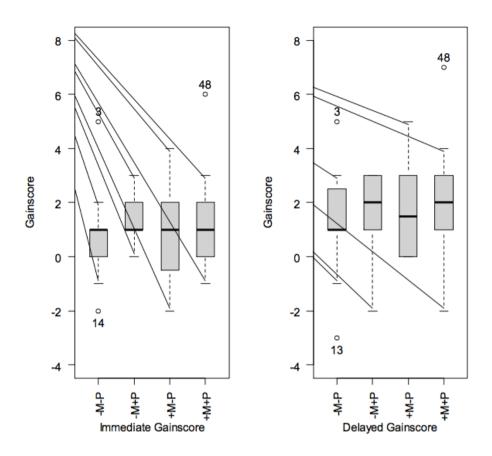


Figure 11.1 Boxplots of the Obarow Story 1 data.

• R コンソール:summary(obarowStory1)

以下の表のように:

```
> summary(obarowStory1)
                                   rdglvl
                                                       picturest1
gender
male :34
                 grade
                                                                       musict1
                                                                                                treatment1
                                                                                                               pretest1
             Min. :1.000
                                      :1.000
                                                no pictures:30
                                                                                                            Min. :11.00
                               Min.
                                                                no music:32
                                                                                  No music No pics :15
female:30
             1st Qu.:1.000
                               1st Qu.:2.000
                                                pictures
                                                           :34
                                                                   music
                                                                           :32
                                                                                  No music Yes pics :17
                                                                                                            1st Qu.:15.00
             Median :2.000
Mean :2.016
                               Median :2.000
                                                                                  Yes music No pics :15
                                                                                                            Median :16.00
                                      :2.094
                                                                                  Yes music Yes pics:17
                                                                                                                   :15.59
                               Mean
                                                                                                            Mean
             3rd Qu.:3.000
Max. :3.000
                               3rd Qu.:3.000
                                                                                                            3rd Qu.:17.00
                                      :3.000
             Max.
                               Max.
                                                                                                            Max.
                                                                                                                    :17.00
   gnsc1.1
                      gnsc1.2
Min. :-2.000
1st Qu.: 0.000
                   Min. :-5.000
1st Qu.: 1.000
Median : 1.000
                   Median : 2.000
Mean
       : 1.125
                   Mean
                          : 1.619
3rd Qu.: 2.000
                   3rd Qu.: 3.000
        : 6.000
                          : 7.000
                         :1
```

解釈:ここで注意するところは、PicturesT1と MusicT1両方とも要因である。 Treatment1は、pictureと musicを組み合わせしている交互作用を表す因子である。もしpictureと music の各自の影響を見たい場合、この二つ因子を独立因子にしなければならない。しかし、両方一緒に見る場合もある、この時、 Treatment 1 という因子が良い。

11.1.1.Means Plots (平均のプロート)

分散分析の研究でよく出る一つタイプのグラフは平均のプロートという。このグラフはデータの趨勢を示しいるですが、各グループの平均しか分からない。 交互作用を図で表す一つの方法をとして、平均のプロートは役に立つですが、 図が簡単すぎで、情報が足りない。そして、二元配置と多元配置の分散分析に 適用しているが、一元配置の分散分析に対してふさわしくない。もし図にすれば、一つの線しかない。

 ■ R コマンダー→グラフ→平均のプロット 「データ」タフで MusicT1 と PicturesT1 を選択 「目的変数」で gnsc1.1 を選択 「オプション」で『信頼区間』を選択、信頼水準は 0.95

Plot of Means

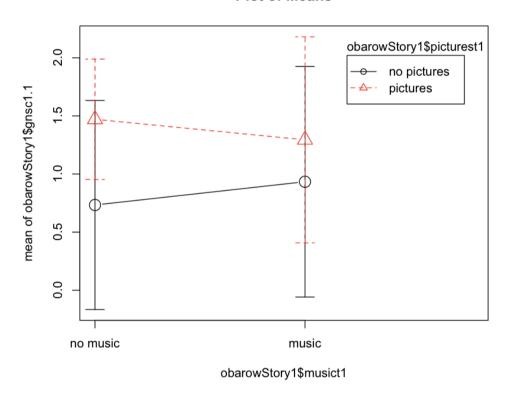


Figure 11.3 の解釈: この図から分かることは、単語を覚える時、pictures ある場合は全体的に効果が良い。(点線はいつでも実線より高い)。しかし、この図は交互作用の趨勢をあらわすはずですが、なぜなら、もし picture がない場合、音楽があったら、単語を覚えるにも有益なことで、しかし、この図で見ると、違う。

<u>ここで注意するところ</u>は picture with no music の平均値(1.5)と picture with music の平均値(1.3)の差が大きくない。このグラフでたぶん差が大きく見えるですが、もしグラフの尺度が小さくなったら、この二つの線は平行線に見えない。一方で、グラフの尺度が大きすぎると、差がなくなる。図で見れば、二つの線

が平行になる場合、交互作用が見えない、というのは、music という要因があるかどうかが picture という要因に影響を及ぼさない。

● R コンソールで:

plotMeans(obarowStory1\$gnsc1.1, obarowStory1\$musict1, obarowStory1\$picturesTt1, error.bars="conf.int")

plotMeans(DepVar, factor 1,factor 2)	一つ従属変数と一つ、二つ独立変数
	が含む平均のプロートを出す
obarow\$gnsc1.1	応答変数、あるいは従属変数を指定
obarow\$musict1,	要因、独立変数を指定
obarow\$picturest1	
error.bars="conf.int"	平均プロートに95%の信頼区間を
	指定、'sd'は標準偏差を指定、'se'
	は標準エーラを指定

<u>ここで注意するところは</u>: 平均のプロートを作る時一つ問題が出るかもしれない。どんな問題というと、要因水準が順番通りに図であらわすわけではない。

● R コンソールで:

with(obarowStory1, plotMeans(gnsc1.1, treatment1, error.bars="conf.int"))

Plot of Means

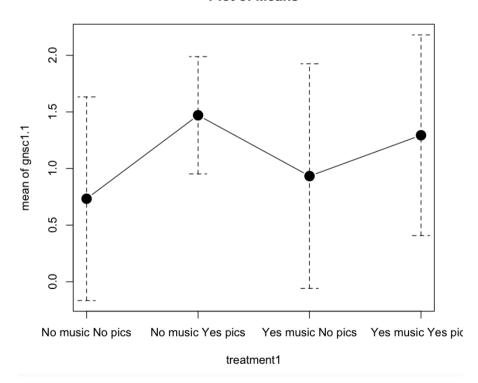


Figure 11.4 の解釈: Figure 11.4 は要因間の交互作用を示していない。そして、筆者はこのように平均のプロートを使うことがおすすめない。しかし、筆者はこの例を使う目的は四つ水準を含む一つ要因を紹介して、どうやって要因の順番を変わると説明する。

そこで、Rコマンダーorder()を使う。

Step1: 要因水準の順番と水準の種類を見るために、以下の変数をリストする。

● R コンソールで: obarowStory1\$treatment1 (まず順番を確認)

```
> obarowStorv1$treatment1
[1] No music No pics
                       No music No pics
                                          No music No pics
                                                            No music No pics
                                                                              No music No pics
 [7] No music No pics
                       No music No pics
                                          No music No pics
                                                            No music No pics
                                                                               No music No pics
[13] No music No pics
                                          No music No pics
                       No music No pics
                                                            No music Yes pics No music Yes pics No music Yes pics
[19] No music Yes pics No music Yes pics No music Yes pics
                                                            No music Yes pics No music Yes pics No music Yes pics
[25] No music Yes pics No music Yes pics No music Yes pics
                                                            No music Yes pics No music Yes pics No music Yes pics
[31] No music Yes pics No music Yes pics Yes music No pics
                                                            Yes music No pics
                                                                              Yes music No pics Yes music No pics
[37] Yes music No pics Yes music No pics Yes music No pics
                                                            Yes music No pics Yes music No pics Yes music No pics
[43] Yes music No pics Yes music Yes pics
[49] Yes music Yes pics Yes music Yes pics
[55] Yes music Yes pics Yes music Yes pics
[61] Yes music Yes pics Yes music Yes pics Yes music Yes pics Yes music Yes pics
Levels: No music No pics No music Yes pics Yes music No pics Yes music Yes pics
```

● R コンソールで: levels(obarowStory1\$treatment1)(水準の種類を提示)

```
> levels(obarowStory1$treatment1)
[1] "No music No pics" "Yes music No pics" "Yes music Yes pics"
```

Step2: 要因の順番を変更する

● R コンソールで:

obarowStory1\$treatment1 <- ordered(obarowStory1\$treatment1, levels= c("No music Yes pics", "Yes music Yes pics", "Yes music No pics", "No music No pics"))

Step3: もう一回平均のプロートを走れる

● R コンソールで

with(obarowStory1, plotMeans(gnsc1.1, treatment1, error.bars="conf.int"))

そして、以下の図が出る:

Plot of Means

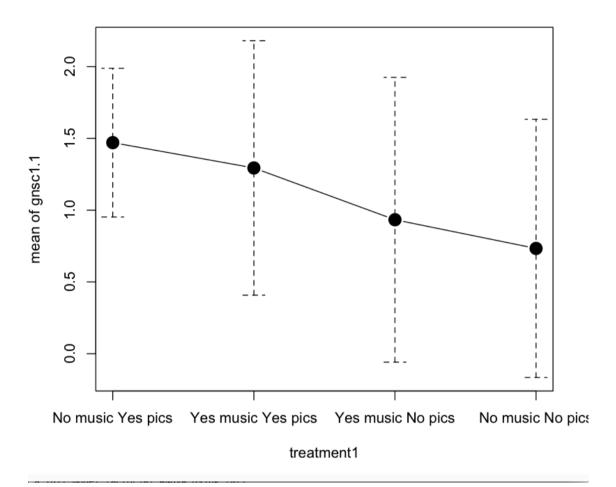


Figure 11.5 の解釈: この図で示している結果は学んだ単語数の最大値から最小値の数列である。

ここで注意するところは、分散分析をやる時、順番要因を避けること。もし順番要因を含んだら、Rで出た結果を解釈できない。

11.2 Putting Data in the Correct Format for a Factorial ANOVA

		nsform Analyze		Utities Add-on	_	Help				
	<u>⊞ 500 }</u>	⊟ mas Bit 1649	*111 till 1	III 475 IIII A	-0.4					
8 : grade	1								,	27 of 27 Variable
	gender	grade	rdglvl	trtmnt1	pretest1	postest1	delpost1	trtmnt2	pretest2	postest2
1	1	1	2	1	18	17	17	2	18	18 4
2	1	1	2	1	17	15	18	2	16	17
3	2	1	1	2	18	19	19	3	12	15
4	2	1	1	2	17	18	18	3	18	19
5	2	1	2	3	15	16	16	4	15	16
6	1	2	1	3	15	16	20	4	18	19
7	2	- 1	1	4	16	16	16	1	14	18

- Figure 11.6 Obarow のデータセートを見で、性別、学年、リーディング.レベルによって $(1 \sim 3)$ 、学生たちがコード化された。そして、pre-test の前に点数を記録する。Treatment グループに対し、一つ特別なスドーリによってコード化された。なぜなら、20 個 target word の中に、学生たち既に知った単語はいくつかを確認するのは大事である。Pre-test の欄を見ると、多くの学生、20 個 target word 大体に知った。
- ▶ Pretest1:事前に記録する点数

Posttest1: 同じ語彙テスト、三回スドーリを聞かせで、点数を記録する Delpost1: 前回より一週間後

テストフォーマット: four-item multiple choice

- ➤ ここで一つ問題がある、データの位置を変える前に、選択した単語はこの年齢の子供にとってよく知らない単語が多いですが、pre-testの段階に既にすごくいい点数を取って、そして、post-testで、知識が増やさない。
- ▶ そこで、考えすべき点は、pre-test の段階で、cut-off ポイントを設定したほうがいい? 子供もしくは 17 個単語を知って、improve スペースが少ない。

Step1: R の一部の行を消すため:

• R コンソールで: new.obarow <- subset(obarow, subset=PRETEST1<18)

Step2:オリジナルデータを並べ換えるため、次は、独立変数をコード化する

● Rコンソールで:

obarow\$musict1 <- recode(obarow\$trtmnt1, '1="no music"; 2="no music"; 3="yes music"; 4="yes music"; ', as.factor.result=TRUE)

Step3:分散分析を行う

🛗 Obaro	w.Story1.sav	- SPSS Data I	Editor					
<u>File</u> <u>E</u> dit	<u>V</u> iew <u>D</u> ata	<u>T</u> ransform <u>A</u>	nalyze <u>G</u> raph	s <u>U</u> tilities A	dd- <u>o</u> ns <u>W</u> indo	w <u>H</u> elp		
26 : Music	T1							
	gender	grade	rdglvl	MusicT1	PicturesT1	Treatment1	pretest1	gnsc1.1
1	1	1	1	1	1	1	14	1.00
2	2	3	3	1	1	1	14	1.00
3	1	1	2	1	2	2	13	3.00
4	2	2	2	1	2	2	13	2.00
5	1	1	3	2	1	3	13	4.00
6	2	1	2	2	1	3	13	2.00
7	1	1	3	2	2	4	11	6.00
8	1	1	3	2	2	4	13	-1.00

11.3Performing a Factorial ANOVA

11.3.1 Performing a Factorial ANOVA Using R Commander

● R コマンダー→統計量→平均→多元配置分散分析 「因子」で gender と musict1 と picturest1 を選択 「目的変数」で gnsc1.2 を選択 Obarow(2004)データセットを使って、三つの独立変数の効果を分析する (gender, music 有り、picture 有り)、応答変数は gnsc.1.1

▶ R コマンダーのアウトプットは full factorial モデルの分散分析表を出る、 この分散分析表は主効果と交互作用効果を含む。

y=Gender + Music + Pictures + Gender*Music + Gender*Pictures + Music*Pictures + Gender*Music*Pictures

ここは*が交互作用を指定する。

> Anova(AnovaModel.1)

Anova Table (Type II tests)

Response: gnsc1.1

```
Sum Sq Df F value Pr(>F)
                           9.990 1 4.3504 0.04157 *
gender
                           0.365 1 0.1588 0.69176
musict1
picturest1
                           2.388 1 1.0399 0.31223
                           5.551 1 2.4174 0.12563
gender:musict1
                           0.270 1 0.1175 0.73308
aender:picturest1
                          1.669 1 0.7266 0.39763
musict1:picturest1
                         1.369 1 0.5959 0.44338
gender:musict1:picturest1
                         128.601 56
Residuals
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

- ▶ 0.05 水準以下の効果は星印の記号がついでいる。つまり、他の要因を見ない、完全に gender によって数値を分けて見ると、male と female は同じタスクで差が出る。
- ▶ 交互作用効果はもし統計的に表している場合、主効果への関心が高くない。
- ▶ 最後の列 Residuals は誤差の列である。

<u>ここで注意するところは</u>もし SPSS という他の統計プログラムを使えば、出る数字が違うかもしれない。

原因: R→Type II 平方和 SPSS→Type III 平方和

- ▶ もし平均平方を計算する場合、R コマンダーのアウトプットで提供された平 方和と自由度を使う。(平均平方:平方和÷自由度)
- ▶ 次のパートの R コマンダーのアウトプットは回帰分析の中のような順番に並べ独立変数の順列の記述統計(平均、標準偏差、回数)を表している。以下の表はこの状態下の平均値を表している。
 - , , PicturesT1 = no pictures

MusicT1

gender no music music male 0.7500000 2.2 female 0.7142857 0.3

, , PicturesT1 = pictures

MusicT1

gender no music music male 1.583333 1.777778 female 1.200000 0.750000 ▶ もしこのデータの組み合わせ方は期待通りではないなら、分散分析モデルの要因の順番を変えれば、変更できる。

主効果と交互作用の偏イータ二乗(偏相関比)効果量は以下の算式を使って計算できる。

$$_{partial}\,\hat{\eta}^{\,2} = rac{SS_{effect}}{SS_{effect} + SS_{error}} \, \cdot$$

重回帰分析:(multiple regression analysis)

多変量解析の一つ。回帰分析において独立変数が2つ以上(2次元以上)のもの。独立変数が1つのものを単回帰分析という。

一般的によく使われている最小二乗法、一般化線形モデルの重回帰は、数学的には線形分析の一種であり、分散分析などと数学的に類似している。適切な変数を複数選択することで、計算しやすく誤差の少ない予測式を作ることができる。

● R コンソールで: summary(AnovaModel.6))

このサマリーはまず他の要因と比べ、default 水準の要因の係数推定量を提供した。最後の三列は:

1."Residual standard error":誤差分散の平方フート

2.二列目は要因回帰分析の全体の効果量を提供した。

3.?

11.3.2 Performing a Factorial ANOVA Using R

- ▶ このセクシーでは、分散分析をどうやって実現するのを説明する。しかし、 注目する点はモデルがデータに適応させるではなく、代わりに、データが モデルに適応させる。
- ➤ 回帰を考えると、データの最小限の適当なモデルを見つけることが理解できる。手順は Regression(回帰)とい章と一致して、復習のために、手順は以下のようになる:
 - 1. Create a full factorial model
 - 2. Examine the output for statistical terms
 - 3. Create a new model that deletes unstatistical entries, beginning with largest terms first and working backwards to simpler terms
 - 4. Compare the two models, and retain the new, simpler model if it does not cause a statistical increase in deviance

ここで、まず手順通り full factorial モデルを作る。

■ R コンソールで:

attach(obarowStory1)
model=aov(gnsc1.1~gender*MusicT1*PicturesT1)
summary(model)

▶ Summary()コマンダーを使って分散分析表を作る。ここは性別の主効果だけ

が統計的に表している。aov()コマンダーを選ぶことは回帰アウトプットが出るわけではなく、分散分析表が出る。

次は、モデルを簡単にする、最下位項を外す。

● R コンソールで:

model2 <- update(model,~.-gender:musict1:picturest1)</pre>

そして、anova介を使って、更新されたモデルとオリジナルモデルを比較する。

● R コンソールで:

anova(model,model2)

次は、model2の summary を見てみる

57 129.97 -1 -1.3685 0.5959 0.4434

● R コンソールで: summary(model2)

```
> summary(model2)
                  Df Sum Sq Mean Sq F value Pr(>F)
gender
                      11.86
                            11.863
                                     5.203 0.0263 *
                   1
                      0.43
                             0.434
musict1
                   1
                                     0.190 0.6643
                      2.44
                             2.438 1.069 0.3055
picturest1
                   1
gender:musict1
                   1
                     4.56
                             4.558
                                     1.999 0.1628
                      0.07
                             0.069
                                     0.030 0.8622
gender:picturest1
                   1
musict1:picturest1 1
                       1.67
                             1.669
                                     0.732 0.3959
Residuals
                  57 129.97
                             2.280
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

次は、gender と picture の交互作用を外し、もう一回モデルを比較する

● R コンソールで:

```
model3 <- update(model2,~.- gender:picturest1) anova(model2,model3)
```

```
Analysis of Variance Table
 Model 1: gnsc1.1 ~ gender + musict1 + picturest1 + gender:musict1 + gender:picturest1 +
    musict1:picturest1
 Model 2: gnsc1.1 ~ gender + musict1 + picturest1 + gender:musict1 + musict1:picturest1
  Res.Df RSS Df Sum of Sq
                          F Pr(>F)
     57 129.97
      58 130.24 -1 -0.26975 0.1183 0.7321
 続いで、必要ではない交互作用と主効果を外す。
● R コンソールで:
   model4 <- update(model3,~.- gender:musict1)
   model5 <- update(model4,~.- musict1:picturest1)
   model6 <- update(model5,~.- picturest1)
そして、要らない部分を全部外した後の null model と比較する
● R コンソールで:
    model8=aov(gnsc1.1\sim1)
Analysis of Variance Table
Model 1: gnsc1.1 ~ gender
Model 2: gnsc1.1 ~ 1
             RSS Df Sum of Sq F Pr(>F)
  Res.Df
       62 139.14
1
2
       63 151.00 -1 -11.863 5.2861 0.02488 *
ここで、model17を選ぶことは null モデルとの間に差がある。
● R コンソールで:
    anova(model7,model8)
> summary(model7)
            Df Sum Sq Mean Sq F value Pr(>F)
             1 11.86 11.863
                               5.286 0.0249 *
aender
Residuals
            62 139.14
                        2.244
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
 ● R コンソールで:
   summary(model7)
Residual standard error: 1.498 on 62 degrees of freedom
Multiple R-squared: 0.07856, Adjusted R-squared: 0.0637
F-statistic: 5.286 on 1 and 62 DF, p-value: 0.02488
```

```
同じモデルの回帰分析はより多い情報を得る
```

● R コンソールで: summary.lm(model7)

```
> summary.lm(model7)
```

Call:

 $aov(formula = gnsc1.1 \sim gender)$

Residuals:

Min 1Q Median 3Q Max -3.5294 -0.8824 0.3333 1.3333 4.4706

Coefficients:

Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.5294 0.2569 5.953 1.34e-07 ***
gender[T.female] -0.8627 0.3752 -2.299 0.0249 *
--Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.498 on 62 degrees of freedom Multiple R-squared: 0.07856, Adjusted R-squared: 0.0637

F-statistic: 5.286 on 1 and 62 DF, p-value: 0.02488

段階的に削除を行う時、boot.stepAICというコマンダーを使う。 ここで、まずパッケージをインストールする

● R コンソールで:

library(bootStepAIC)
boot.stepAIC(model,data=obarowStory1)

以下の表が出る:

```
Initial Model:
```

gnsc1.1 ~ gender * musict1 * picturest1

Final Model:

gnsc1.1 ~ gender + musict1 + gender:musict1

```
        Step Df Deviance Resid. Df Resid. Dev AIC

        1
        56
        128.6008 60.66111

        2 - gender:musict1:picturest1
        1
        1.3685372
        57
        129.9693 59.33859

        3 - gender:picturest1
        1
        0.2697513
        58
        130.2391 57.47128

        4 - musict1:picturest1
        1
        1.4680660
        59
        131.7071 56.18866

        5 - picturest1
        1
        2.3880898
        60
        134.0952 55.33870
```

解釈: このフルなモデルは 60.66 の AIC 点数がある。AIC の点数が低いほど低ければ、より簡略化されたモデルを得る。

赤池情報量規準(元々は An Information Criterion, のちに Akaike Information Criterion と呼ばれるようになる)は、統計モデルの良さを評価するための指標である。AIC は、「モデルの複雑さと、データとの適合度とのバランスを取る」ために使用される。

.....

ここで、まず boot.stepAIC から選んだモデルを作る(model 9),作って model7 と比較する:

● R コンソールで:

model9=aov(gnsc1.1~gender+musict1+gender:musict1) summary.lm(model9)

Residual standard error: 1.495 on 60 degrees of freedom Multiple R-squared: 0.112, Adjusted R-squared: 0.06755 F-statistic: 2.521 on 3 and 60 DF, p-value: 0.06631

分散分析表が示すことはmusict1も交互作用gender:musict1もこの三つの変数のモデルの中に統計的に表していない。

● R コンソールで: summary(model9)

> summary(model9)

やるべきことを確認するために、anovaを使って、model7と model9を比較する。

● R コンソールで:

anova(model7,model9)

```
> anova(model7,model9)
```

Analysis of Variance Table

```
Model 1: gnsc1.1 ~ gender

Model 2: gnsc1.1 ~ gender + musict1 + gender:musict1

Res.Df RSS Df Sum of Sq F Pr(>F)

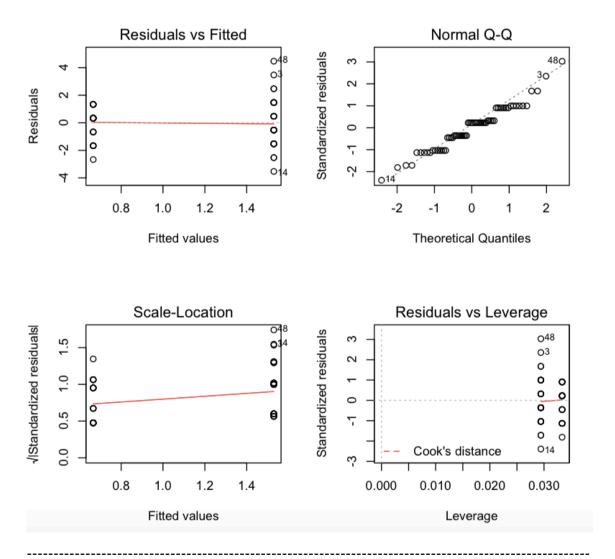
1 62 139.14

2 60 134.09 2 5.042 1.128 0.3304
```

Examine ANOVA (regression) assumptions with diagnostic plots:

R コンソールで: par(mfrow=c(2,2)) plot(model7) detach(obarowStory1)

そして、以下の図が出る:



分位数-分位数 (00) プロット

QQ プロットは、2 つの標本が同じ分布によるものであるかどうかを判定するために使われます。これらは、1 番目と 3 番目の分位数間に描かれる線と、各標本から計算される分位数の散布図です。データが線の近くにある場合、2 つの標本が同じ分布によるものであると仮定することは妥当です。この方法は、どちらの分布の位置とスケールの変化に対してもロバストです。

分位数-分位数プロットを作成するには、関数 gaplot を使用します。

If you want to leave out the influential points (outliers) from the data when you create a full factorial model:

● R コンソールで:

 $model.adj=aov(gnsc1.1[-c(3,14,48)]\sim gender[-c(3,14,48)],data=obarow)$ summary.lm(model.adj)

11.4 Performing Comparison in a Factorial ANOVA

このセクシーはもし変数が2個以上の水準がある場合に、分散分析を行う際に やるべきことを説明する。

ここでまず新しいデータセットをインストールする

DATA: Writing

従属変数: writing 評価の点数 (35~373)

独立変数: L1(アラビア語、日本語、ロシア語、スベイン語)と条件

三つの条件:

correctAll→全てのエーラを訂正する

correctTarget→タゲートのエーラだけを訂正する

noCorrect→訂正しない

Mean scores on writing assessment

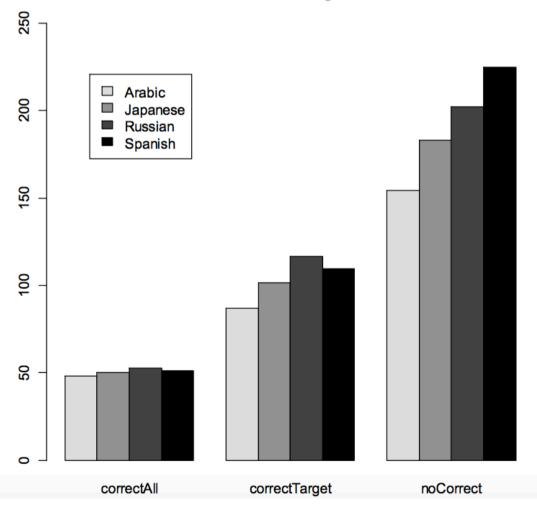


Figure 11.10

解釈: noCorrect の条件では全体的に点数が高い。

CorrectAllの条件では点数が低い。

しかし、correctAll の条件において、L1 が違っても、あまり差が出ない。 noCorrect の条件において、スベイン語の書く手の点数が比較的に高い。 ここで、平均値を得るため、tapply()というコマンダーを使う。

• R コンソールで: attach(Writing)

tapply(score,list(L1,condition),mean)

- > attach(Writing)
- > tapply(score, list(L1, condition), mean)

correctAll correctTarget noCorrect
Arabic 48.23729 87.01316 154.3882
Japanese 49.96667 101.72500 182.9200
Russian 52.43333 116.72500 202.4792
Spanish 51.13333 109.45000 224.8400

• Rコンソールで: write=lm(score~L1*condition) Anova(write)

> write=lm(score~L1*condition)

> Anova(write)

Anova Table (Type II tests)

Response: score

Sum Sq Df F value Pr(>F)
L1 131252 3 27.1899 < 2.2e-16 ***
condition 1777365 2 552.2920 < 2.2e-16 ***
L1:condition 70588 6 7.3114 1.533e-07 ***
Residuals 910740 566

解釈:分散分析表は L1 の主効果、条件の主効果、L1 と条件の交互作用が統計的に表している。しかし、交互作用が統計的に表している時、我々が主効果より交互作用が関心を持っている。そして、点数が L1 と条件の組み合わせに影響されるですが、どうやって L1 と条件が点数に影響を与えることがより詳しく知る必要がある。ここで、比較を行う。

● R コンソールで: summary(write) 以下の表が出る:

Coefficients:

	Estimate S	Std. Error	t value	Pr(>ltl)	
(Intercept)	48.237	5.222	9.237	< 2e-16	***
L1[T.Japanese]	1.729	8.995	0.192	0.84761	
L1[T.Russian]	4.196	8.995	0.466	0.64104	
L1[T.Spanish]	2.896	8.995	0.322	0.74760	
condition[T.correctTarget]	38.776	6.960	5.571	3.92e-08	***
condition[T.noCorrect]	106.151	6.797	15.617	< 2e-16	***
L1[T.Japanese]:condition[T.correctTarget]	12.982	11.929	1.088	0.27693	
L1[T.Russian]:condition[T.correctTarget]	25.516	11.929	2.139	0.03287	*
L1[T.Spanish]:condition[T.correctTarget]	19.541	11.929	1.638	0.10197	
L1[T.Japanese]:condition[T.noCorrect]	26.802	11.490	2.333	0.02002	*
L1[T.Russian]:condition[T.noCorrect]	43.895	11.548	3.801	0.00016	***
L1[T.Spanish]:condition[T.noCorrect]	67.556	11.490	5.880	7.04e-09	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

ここで、L1 という要因はいくつ水準を見るため、以下の levels コマンダーを使う。Arabic グループはベースになる水準。

● R コンソールで: levels(L1)

> levels(L1)

[1] "Arabic" "Japanese" "Russian" "Spanish"

解釈:

L1[T.Japanese]: アラビア語グループと日本語グループを比較する。

Intercept estimate: allCorrect の条件においてアラビア語話者は writing sample で全体の平均値が 48.2 ポイント。日本語話者は同じ条件で、アラビア語話者より 1.7 ポイント高い。

Condition[T.correctTarget]: allCorrect の条件と correctTarget の条件を比較 Condition[T.noCorrect]: allCorrect の条件と noCorrect の条件を比較 下の 6 列は: 異なる L1 グループはいずれか correctTarget の条件と noCorrect の条件の場合とアラビア語話者が allCorrect の条件の場合を比較

最後の6列から分かることは:ロシア語話者は correctTarget の条件とアラビア語話者は allCorrect の条件の間に差がある。

そこで、pairwise.t.test()コマンダーを使って、全ての一対比較を行うことが可能である。

一対比較

一対比較とは、判断の対象となる選択肢を2つ一組として、どちらがよりよいか、どちらを選択するかなどの比較判断を行うことである。一対比較は、最も単純な判断方法のため、他の多肢選択法、評定尺度法(例えば、5 段階評価など)よりも、容易であり、低年齢の被験者に対しても用いることが可能であり、再現性が高く、細かい差に対する識別力が大きいことなどから、非常に広く用いられている。

例えば、L1だけを比較する

• R コンソールで: pairwise.t.test(score,L1, p.adjust.method="fdr")

data: score and L1

Arabic Japanese Russian

Spanish 2.5e-06 0.03519 0.39274

全体的に見ると、アラビア語の母語話者グループは他のグループと異なる。 日本語の母語話者グループとスベイン語の母語話者グループ(ロシア語の母語 話者グループではなく)が異なっている。ロシア語の母語話者グループとスベイン語の母語話者グループが異なっていない。

条件だけを比較することもできる→

• R コンソールで: pairwise.t.test(score,condition, p.adjust.method="fdr")

data: score and L1

Arabic Japanese Russian

Japanese 0.02279 -

Russian 0.00013 0.19195 -

Spanish 2.5e-06 0.03519 0.39274

交互作用を比較すると、以下のコマンダーを入力する

• R コンソールで: pairwise.t.test(score,condition, p.adjust.method="fdr")

data: score and L1:condition

	Arabic:correctAll	Arabic:correctTarget	Arabic:noCorrect	Japanese:correctAll	Japanese:correctTar
Arabic:correctTarget	6.0e-08	-	-	- 1	- 1
Arabic:noCorrect	< 2e-16	< 2e-16	-	-	-
Japanese:correctAll	0.87409	2.9e-05	< 2e-16	-	-
Japanese:correctTarget	2.9e-10	0.07059	3.8e-11	2.0e-07	-
Japanese:noCorrect	< 2e-16	< 2e-16	9.5e-05	< 2e-16	< 2e-16
Russian:correctAll	0.69359	9.4e-05	< 2e-16	0.85050	6.9e-07
Russian:correctTarget	1.3e-15	0.00021	1.7e-06	3.0e-11	0.10812
Russian:noCorrect	< 2e-16	< 2e-16	1.3e-10	< 2e-16	< 2e-16
Spanish:correctAll	0.79583	5.1e-05	< 2e-16	0.91035	3.6e-07
Spanish:correctTarget	7.2e-13	0.00531	1.4e-08	2.6e-09	0.43568
Spanish:noCorrect	< 2e-16	< 2e-16	< 2e-16	< 2e-16	< 2e-16

解釈: この図の全体を見ると、交互作用の本質の概要を分かる: correctAll の条件で、グループ間は差がない。correctTarget の条件で、ロシア語とアラビア語話者、またはスベイン語とアラビア語話者は統計的に互いに異なっている。 noCorrect の条件で、全てのグループは互いに異なっている。

pan wise.i.iesi(score	
	なる
L1:condition	二つ目の変数が比較したい水準を含む変
	数である。ここで、L1 と条件の主効果を
	関心を持っていないですが、注目するとこ
	ろはこの二つの交互作用。そして、二つ目
	の変数は交互作用である。
p.adjust.method="fdr"	多重検定のための調整している p 値の方
	式を指定する

もし"noCorrect"のこの条件だけでどのグループと他のグループが異なるに関心があれば、調整無しの p 値がある post-hoc のコマンダーをいります。

● Rコンソールで:

pairwise.t.test(score,L1:condition, p.adjust.method="none")

	Arabic	Japanese	Russian
Japanese	7.4e-05		
Russian	7.4e-11	.016	
Spanish	<2e-16	2.4e-07	.00599

Put these in the FDR calculation

● Rコンソールで:

pvalue<-c(7.4e-05,7.4e-11,2e-16,.016,2.4e-07,.00599)
sorted.pvalue<-sort(pvalue)
j.alpha<-(1:6)*(.05/6)
dif<-sorted.pvalue-j.alpha
neg.dif<-dif[dif<0]
pos.dif<-neg.dif[length(neg.dif)]
index<-dif==pos.dif
p.cutoff<-sorted.pvalue[index]
p.cutoff</pre>

11.6 Performing a Robust ANOVA

このセクションは Wilcox(2005)の robust methods を使って二重分散分析と多重分散分析を解釈する。

まず、パッケージをインストールする、以下の手順になる:

1) first: install dependent packages

● Rコンソールで:

install.packages(c("MASS", "akima", "robustbase"))

2)second: install suggested packages

● Rコンソールで:

```
install.packages(c("cobs", "robust", "mgcv", "scatterplot3d", "quantreg", "rrcov",
"lars", "pwr", "trimcluster", "mc2d", "psych", "Rfit"))
3)third: install an additional package which provides some C functions
    Rコンソールで:
install.packages("devtools")
library("devtools")
install_github("mrxiaohe/WRScpp")
4) fourth: install WRS
    Rコンソールで:
install_github("nicebread/WRS", subdir="pkg")
grp コマンダーを使ってこの欄がどんな順番で並ぶことを分かる:
    Rコンソールで:
grp < -c(2,3,5,8,4,1,6,7)
Obaroaw のデータを使って music と picture だけの二重分散分析を見て見ましょ
う:
    Rコンソールで:
     levels(obarowStory1$Treatment1) #find out exactly what names to use to
     subset
     02way=list()
     02way[[1]]=subset(obarowStory1, subset=Treatment1=="No music No
     pics", select=c(gnsc1.1))
     O2way[[2]]=subset(obarowStory1, subset=Treatment1=="No music Yes
     pics", select=c(gnsc1.1))
     O2way[[3]]=subset(obarowStory1, subset=Treatment1=="Yes music No
     pics", select=c(gnsc1.1))
     O2way[[4]]=subset(obarowStory1, subset=Treatment1=="Yes music Yes
     pics", select=c(gnsc1.1))
次は、t2way()テストを行う
    Rコンソールで:
     library(WRS)
     t2way(2,2,02way,tr=.2,grp=c(1,2,3,4)
```

```
> t2way(2,2,02way,tr=.2,grp=c(1,2,3,4))
$0a
[1] 0.004277459
$A.p.value
[1] 0.949
$0b
[1] 1.35527
$B.p.value
[1] 0.256
$Qab
[1] 0.4108072
$AB.p.value
[1] 0.528
$means
           [,1]
[1,] 0.6666667 1.363636
[2,] 0.8888889 1.090909
percentile bootstrapping と M-estimators を使って二重分析を行う時、pbad2way()
というコマンダーを使う。
    Rコンソールで:
pbad2way(2,2,02way,est=mom, conall=T, grp=c(1,2,3,4))
three-way 分析のほうはもっと複雑、
                                  t3way()を使う
ここで二つ data set を作る。
One: males
           One: females
    Rコンソールで:
     obarowStory1Male <- subset(obarowStory1, subset=gender=="male")
     obarowStory1Female <- subset(obarowStory1, subset=gender=="female")
     03way=list()
     O3way[[1]]=subset(obarowStory1Male, subset=Treatment1=="No music No
     pics", select=c(gnsc1.1))
     03way[[2]]=subset(obarowStory1Female, subset=Treatment1=="No music
     No pics", select=c(gnsc1.1))
     O3way[[3]]=subset(obarowStory1Male, subset=Treatment1=="No music
     Yes pics", select=c(gnsc1.1))
     O3way[[4]]=subset(obarowStory1Female, subset=Treatment1=="No music
     Yes pics", select=c(gnsc1.1))
```

```
O3way[[5]]=subset(obarowStory1Male, subset=Treatment1=="Yes music No pics", select=c(gnsc1.1))
O3way[[6]]=subset(obarowStory1Female, subset=Treatment1=="Yes music No pics", select=c(gnsc1.1))
O3way[[7]]=subset(obarowStory1Male, subset=Treatment1=="Yes music Yes pics", select=c(gnsc1.1))
O3way[[8]]=subset(obarowStory1Female, subset=Treatment1=="Yes music Yes pics", select=c(gnsc1.1))
```

そして、three-way 分析を行う:

● Rコンソールで:

t3way(2,2,2,03way, tr=.2)

03way[[1]]=c(3,3,1,3,0,1,3,1,0,1,2,1)

O3way[[2]]=c(2,1,1,1,1)

O3way[[3]]=c(6,-1,0,3,2,3,1,0,2)

O3way[[4]]=c(1,0,0,1,2,-2,1,2)

O3way[[5]]=c(4,1,4,2,0)

O3way[[6]]=c(2,2,1,1,-1,-1,0,-2,-1,2)

O3way[[7]]=c(1,5,-1,0,2,0,1,-2)

O3way[[8]]=c(1,2,1,0,1,-1,1)

t3way(2,2,2,03way)

\$Qa

[1] 0.3566423

\$Qa.crit

[1] 5.03417

\$A.p.value

[1] 0.57

\$Qb

[1] 0.7946762

\$Qb.crit

[1] 5.03417

\$B.p.value

[1] 0.4

\$Qc

[1] 2.960803

\$Qc.crit

[1] 2.947086

\$C.p.value

[1] 0.12

\$Qab

[1] 0.4147462

\$Qab.crit

[1] 0.4128055

\$AB.p.value

[1] 0.538

\$Qac

[1] 0.03581812

\$Qac.crit

[1] 0.03560799

\$AC.p.value

[1] 0.855

\$Qbc

[1] 1.509205

\$Qbc.crit

[1] 1.506753

\$BC.p.value

[1] 0.252

\$Qabc

[1] 1.837542