

## § 3.2

### 統計的仮説検定の考え方

さらに、統計的仮説検定について単純化して説明をしたいと思います。そのために、これまでの高校生の喫煙の例を使いましょう。

検証したい内容は

喫煙の有無によって学力に差がある。

とします。

この時、次の点を前提とします。

1. 全ての高校生(母集団とする)を集めるわけにはいかない。
2. したがって、高校生から何人かを抽出し(標本集団とする)調査対象とする。
3. 母集団における「学力差」は真の値である。そして、それを「母数」呼ぶ。

私たちの(本当の)興味は

抽出して得られたデータ(すわなち標本)に基づいて推定された値(推定値)が母集団における値(母数)と一致するかどうか

という点にあります\*1。

しかし、現実には母数を知る由もありません。不可能なのです。

もし、仮に、母集団全体からデータを得た場合には統計的仮説検定をする必要はありません。ただし、現実的にそれは不可能なため、データを抽出し、統計的仮説検定によってその結果の正当性を主張します。

そこで、次のようなスキームが考えます。

「母数が0である」と仮定したときにその推定値を得る可能性が十分に低ければ「母数が0であるという仮定が間違えであり、母数は0であるとは言えない」と主張する。

だいぶ、表現が弱くなりました。

内容を順に追って説明します。

\*1統計的仮説検定の解釈をこのようにしないように注意が必要です。

1. 我々が得ることのできる推定値はある1回の標本データから計算されるものである。
2. 標本データは母集団から抽出したものである。
3. 抽出するたびに、手元に得られる標本データは変わる。
4. したがって、推定値はばらつきがある。すなわち、推定値は分布する。
5. その確率分布を理論的に導く(実際は、導かれたものがある。t-分布, F-分布,  $\chi^2$ -分布等)
6. 母数が0であると帰無仮説を立てる。
7. その帰無仮説のもとでの確率分布を考える。
8. 得られた推定値よりも極端な値が、帰無仮説のもとでの確率分布においてどの程度起こりうるかを計算する。
9. その確率があらかじめ設定した確率(有意水準)よりも小さければ「『母数が0である』という仮説は間違えであった」と考える。

統計的仮説検定ではこれだけのことしかしません。すなわち、(多くの)統計的仮説検定では、「母数は0ではない」と言うためにおこなうものなのです。

また、

検定は繰り返してはいけません。

統計的仮説検定は5%の確率で誤りを犯すメカニズムになっています。また、検定はいくつかの前提の上に  $p$  値が計算されています。したがって、5%の可能性で過ちを犯す検定を3回繰り返しておこなった場合、最終的に得られる結果は

$$0.142625 = 1 - 0.95^3$$

もの割合で過ちを犯すこととなります。これはとても大きな数値です。

2回の繰り返しでさえも

$$0.0975 = 1 - 0.95^2$$

と大きな数値になります。

## § 3.3

### 検定をおこなう前に

それでは、喫煙のデータを用いて実際に検定について考えてみます。

検定や統計モデルを構成する前に必ず、データの記述統計量を算出したり、散布図、箱ひげ図などを描いてデータの内容を確認するようにして下さい。

具体的な確認内容は

1. データにおける変数ごとの尺度水準が適切に設定しているか確認する
2. 外れ値がないかを目視する
3. データが切断<sup>\*2</sup>されていないか、床効果<sup>\*3</sup>、天井効果<sup>\*4</sup>が発生していないかを確認する。
4. 変数ごとにデータが正規分布しているかを確認する(検定は正規分布を前提にしているものもあります)。

などです。

それでは、次の作業をしてみてください。

1. Smoke03.csv を、Smoke03 としてデータに読み込みます。

<sup>\*2</sup>ある値を境にデータが確認されなくなること

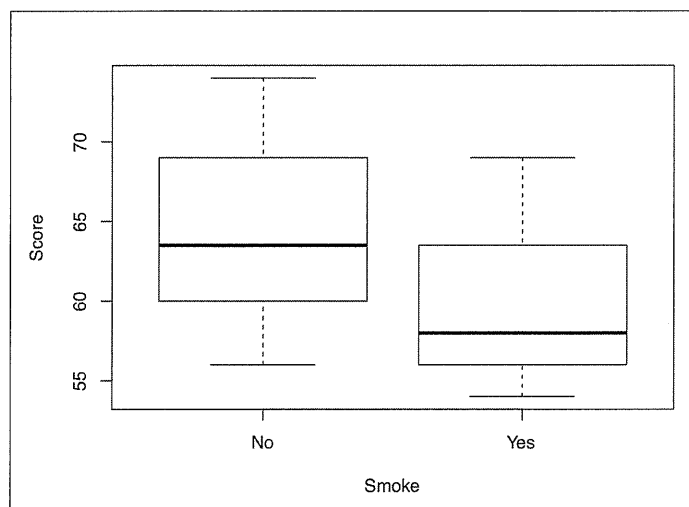
<sup>\*3</sup>データの測定範囲に下限があるためにデータが測定できる下限にたまってしまうこと。

<sup>\*4</sup>データの測定範囲に上限があるためにデータが測定できる上限にたまってしまうこと。

2. Smoke03 から Grade01 のケースのみを抜き出して、それを Smoke03G1 としてオブジェクトに入れてください。
3. Smoke を層化変数として Score に関して箱ひげ図を描いて下さい。
4. Score に関して、Smoke ごとに平均と標準偏差を計算して下さい。
5. 外れ値がないか、またデータが正規分布に従っているか確認してください。

以下のようにスクリプトが生成され、R コンソールの方にグラフが表示されます。

```
> Boxplot(Score~Smoke, data=Smoke03G1, id.method="y")
> numSummary(Smoke03G1[, "Score"], groups=Smoke03G1$Smoke, statistics=c("mean", "sd", "IQR", "#$
+   "quantiles"), quantiles=c(0,.25,.5,.75,1))
      mean      sd IQR 0% 25% 50% 75% 100% data:n
No 64.26316 5.824850 8.75 56 60 63.5 68.75 74 38
Yes 60.33333 7.767453 7.50 54 56 58.0 63.50 69 3
```



見たところ、喫煙の有無によって差はありそうですが、これが統計的に有意な差があるかを確認します。

### § 3.4

#### サンプルサイズと $p$ 値の関係

さて、これまでに使ってきた喫煙データは私が次のようなメカニズムで発生させたものでした。

$$\text{得点} = 50 + 15 \times \text{学年} - 5 \times \text{喫煙の有無} + e, \quad e \sim N(0, 6) \quad (3.1)$$

すなわち、

テスト得点は、学年の数字に 15 をかけたものに 50 を足し、喫煙していれば 5 を引き、していなければ引かない。それに誤差を加えたものになっている。なお、その誤差は平均 0、分散 4 の正規分布にしたがう。

というものです。なお、この誤差は高校生ごとの学力差であったり各高校生の当日の体調によるテスト得点への影響などモデルに含まれないすべての効果と解釈することもできます。

したがって、結論は

喫煙の有無によって学力に差がある。

ということになっています。具体的には「5点の差がある」ということです。

ここでは、そのデータ発生メカニズムに基づいて、データを発生させた場合にサンプルサイズと検定結果がどのように変わるかを確認したいと思います。

それでは、次のような流れで R を用いて疑似データに対する Welch の  $t$  検定をそれぞれの条件で複数回おこなっていきます。

1. サンプルサイズ  $N$  を決める。
  - (a) 非喫煙群のデータを  $N(65, 6)$  から  $N$  個発生させる。
  - (b) 喫煙群のデータを  $N(60, 6)$  から  $N$  個発生させる。
2. 2つのデータをまとめる。
3. Welch の  $t$ -検定をおこなう。
4.  $p$  値を確認する。

具体的には、 $N = 10, 20, 50, 100, 500, 1000$  と変えてみて下さい。

一つの条件で、10回くらい実行し、そのときの差と  $p$  値を記録しながら、サンプルサイズ  $N$  によってどのように値が変化していくかを確認してみてください。

```
### CODE STARTS FROM HERE
# サンプルサイズの設定
N <- 10
# 正規乱数から得点の発生
SmokeScore <- rnorm(N, 60, 6)
NonSmokeScore <- rnorm(N, 65, 6)
Score <- c(SmokeScore, NonSmokeScore)
# 喫煙か非喫煙かのコーディング
Smoke <- c(rep("Yes", N), rep("No", N))
# 二つの変数をデータ化
Data <- data.frame(Smoke = Smoke, Score = Score)
# Welch の t 検定の実行
t.test(Score~Smoke, alternative='two.sided', conf.level=.95, var.equal=FALSE, data=Data)
### END OF CODE
```

```
### CODE STARTS FROM HERE
SimResult <- NULL
for (N in c(10, 20, 50, 100, 500, 1000)) {
  i = 0; P <- NULL; Diff <- NULL
  while (i < 100) {
    i <- i + 1
    SmokeScore <- rnorm(N, 60, 6)
    NonSmokeScore <- rnorm(N, 65, 6)
    Score <- c(SmokeScore, NonSmokeScore)
    Smoke <- c(rep("Yes", N), rep("No", N))
    Data <- data.frame(Smoke = Smoke, Score = Score)
    Result <- t.test(Score~Smoke, alternative='two.sided', conf.level=.95,
                    var.equal=FALSE, data=Data)
    P <- c(P, Result[[3]])
    Diff <- c(Diff, Result[[5]][1] - Result[[5]][2])
  }
  SimResult <- rbind(SimResult, c(mean(Diff), sum(P < 0.05)))
}
SimResult
### END OF CODE
```

---

**§ 3.5**

---

**課題**

---

**課題 1**

1. Smoke03G2 と Smoke03G3 に関しても，母平均の差の検定をそれぞれおこなってください。

**課題 2**

1.  $t$  検定のシミュレーション用のプログラムで喫煙群の平均を 61 点としてデータを発生させて，サンプルサイズと  $p$  値の関係を確認して下さい。

## 4

---

## 変量の検定

---

---

### § 4.1

---

#### 統計的仮説検定の過誤

---

統計的仮説検定においては、2種類の過誤がある。すなわち、

1. 帰無仮説が正しいのに、仮説を棄却してしまう誤りのこと。設定した有意水準の確率で犯す過ち。「あわてものの過ち」
2. 帰無仮説が誤りなのに、これを受容してしまう誤りのこと。「ぼんやりものの過ち」

また、上述の1番目を第一種の過誤と言い、2番目を第二種の過誤と呼ぶ。

なお、第一種の過誤と第二種の過誤の関係は

1. 第一種の過誤を犯す確率を低くする (誤って帰無仮説を棄却しないようにする) と第二種の過誤を犯す確率が高くなる。
2. 第二種の過誤を犯す確率を低くする (=帰無仮説を棄却する確率を上げる) と第一種の過誤を犯す確率が高くなる

とトレードオフの関係性を持つ。

---

### § 4.2

---

#### サンプルサイズの計算

---

$t$  検定

```
> power.t.test(delta=0.5, sd=1, sig.level=0.05, power=0.8, n=NULL)
```

```
Two-sample t test power calculation
```

```
      n = 63.76576
delta = 0.5
      sd = 1
sig.level = 0.05
      power = 0.8
alternative = two.sided
```

```
NOTE: n is number in *each* group
```

### $\chi^2$ 検定

```
> power.prop.test(p1=0.2, p2=0.3, sig.level=0.01, power=0.85, n=NULL)
```

```
Two-sample comparison of proportions power calculation
```

```
      n = 487.4402
      p1 = 0.2
      p2 = 0.3
sig.level = 0.01
      power = 0.85
alternative = two.sided
```

```
NOTE: n is number in *each* group
```

### 分散分析

```
> power.anova.test(groups=3, between.var=1, within.var=3, sig.level=0.05, power=0.90, n=NULL)
```

```
Balanced one-way analysis of variance power calculation
```

```
      groups = 3
      n = 20.01728
between.var = 1
within.var = 3
sig.level = 0.05
      power = 0.9
```

```
NOTE: n is number in each group
```

## § 4.3

### 仮説検定法

#### 4.1 主な仮説検定方法

尺度水準	対象	検定内容	検定名	利用分布
比例 ・間隔尺度	平均	平均の差 (対応のない2標本, 分散既知か $n \geq 100$ )	$z$ 検定	$z$ 分布
		平均の差 (対応のない2標本, 分散未知で $n < 100$ )	Welch の $t$ 検定	$t$ 分布
		平均の差 (対応のない $k$ 標本)	分散分析	$F$ 分布
		平均の差 (対応のある2標本, $n \geq 100$ )	$z$ 検定	$z$ 分布
		平均の差 (対応のある2標本, $n < 100$ )	$t$ 検定	$t$ 分布
		平均の差 (対応のある $k$ 標本)	分散分析	$F$ 分布
	分散	分散の比	$F$ 検定	$F$ 分布
	相関	ピアソンの相関係数	$t$ 検定	$t$ 分布
	回帰	回帰係数	$t$ 検定	$t$ 分布
	順序尺度	代表値	代表値の差 (対応のない2標本)	Mann-Whitney の $U$ 検定
代表値の差 (対応のない $k$ 標本)			Kruskal-Wallis の検定	独自の分布
代表値の差 (対応のある2標本)			Wilcoxon の順位和検定	独自の分布
代表値の差 (対応のある $k$ 標本)			Friedman の検定	独自の分布
相関		相関係数	Spearman の順位相関係数の検定	$t$ 分布
名義尺度	比率	比率の差 (対応のない2標本)	Fisher の正確検定 ( $\chi^2$ 検定)	独自の分布
		比率の差 (対応のない $k$ 標本)	$\chi^2$ 検定	$\chi^2$ 分布
		比率の差 (対応のある2標本)	McNemar の検定	$\chi^2$ 分布
		比率の差 (対応のある $k$ 標本)	Cochran の $Q$ 検定	独自の分布



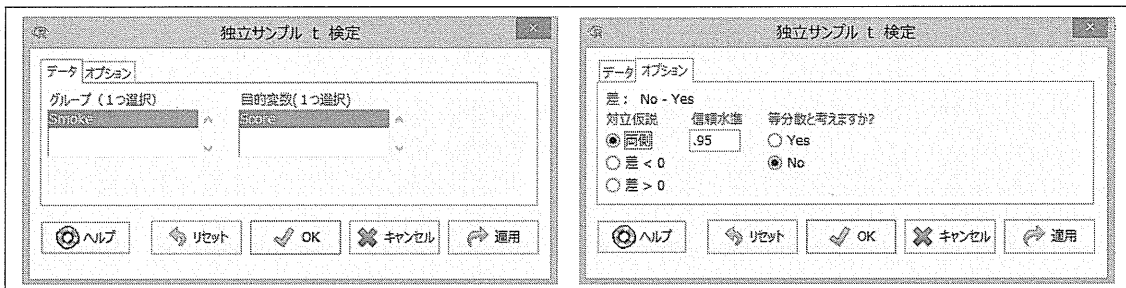
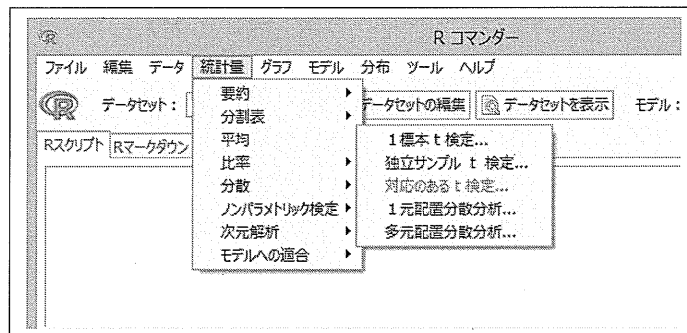
## § 4.4

Welch の  $t$  検定

Welch の  $t$  検定は、対応のない 2 標本で、標本ごとの分散が未知であり、標本の大きさが小さい時に用いる手法であり、平均の差について検定するような方法です。

ここでは、Smoke03(Smoke03.csv) を用いて、実際に検定をおこなってみます。

1. 【データセット】を指定します。
2. 【統計量】 → 【平均】 → 【独立サンプル  $t$  検定】を選びます。
3. 【グループ】を選択し、【目的変数】を選びます。
4. 【オプション】の対立仮説を選び、【等分散と考えますか?】には No を選びます。
5. 【OK】を押します。



次のようにスクリプトが生成され、結果が表示されます。

```
> t.test(Score~Smoke, alternative='two.sided', conf.level=.95, var.equal=FALSE,
+ data=Smoke03)
```

Welch Two Sample t-test

data: Score by Smoke

t = -1.5351, df = 53.513, p-value = 0.1307

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

-8.126192 1.079271

sample estimates:

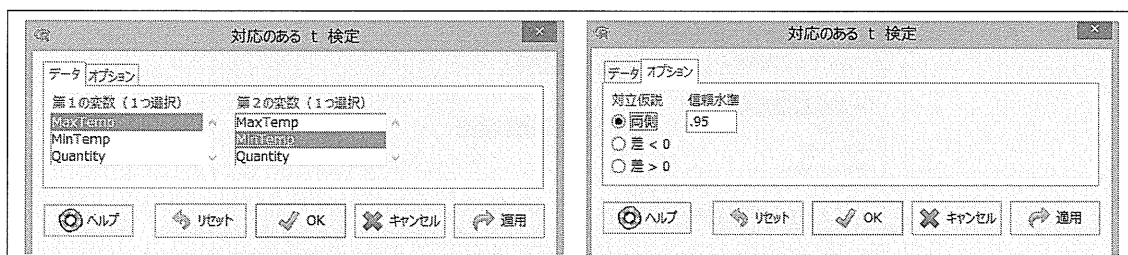
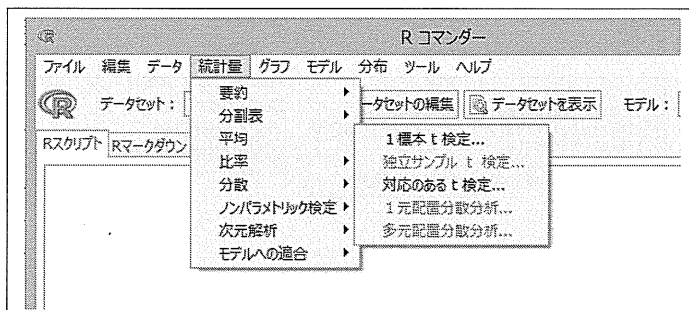
mean in group No	mean in group Yes
79.11290	82.63636

## § 4.5

### 対応のある t 検定

対応のある t 検定は、同一の発生源から得られた 2 つの標本 (すなわち、対応のあるデータ) を比較するような検定手法です。具体的には、「運動前の心拍数と運動後の心拍数」などが対応のあるデータにあたります。

1. 【統計量】 → 【平均】 → 【対応のある t 検定】 を選びます。
2. 【第 1 の変数】 を選択し、【第 2 の変数】 も選びます。
3. 【オプション】 の対立仮説を選び、信頼水準は 0.95 のままにしておきます。
4. 【OK】 を押します。



次のようにスクリプトが生成され、結果が表示されます。

```
> with(Ice01, (t.test(MaxTemp, MinTemp, alternative='two.sided', conf.level=.95, paired=TRUE)))
```

Paired t-test

data: MaxTemp and MinTemp

t = 12.7958, df = 19, p-value = 8.694e-11

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

6.52415 9.07585

sample estimates:

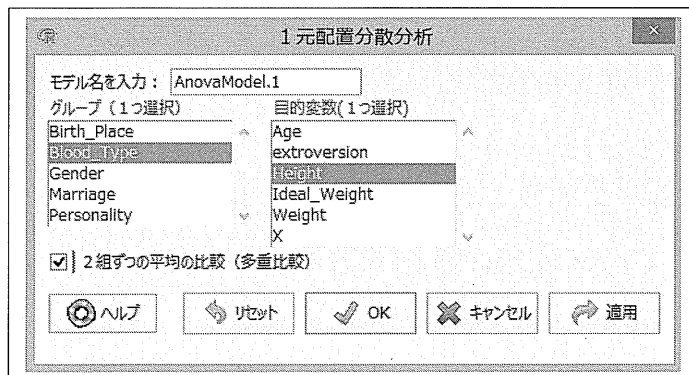
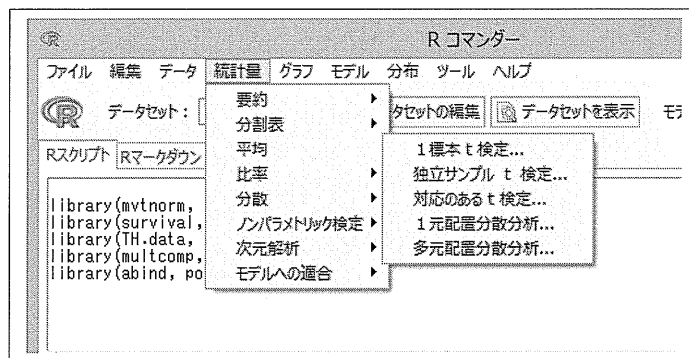
mean of the differences

7.8

## § 4.6

### 分散分析

1. 【統計量】 → 【平均】 → 【1元配置分散分析】 を選びます。
2. 【モデル名】 を入力します。
3. 【グループ】 を選択し, 【目的変数】 も選びます。
4. 【2組ずつの平均の比較 (多重比較)】 にチェックを入れる。
5. 【OK】 を押します。



次のようにスクリプトが生成され, 結果が表示されます。

```

> AnovaModel.1 <- aov(Height ~ Blood_Type, data=Personal)
> summary(AnovaModel.1)

          Df Sum Sq Mean Sq F value Pr(>F)
Blood_Type  3    42   13.89   0.399  0.754
Residuals 187  6503   34.77
2 observations deleted due to missingness

> with(Personal, numSummary(Height, groups=Blood_Type, statistics=c("mean", "sd")))
      mean      sd data:n
A 157.9538 6.096697    65
AB 157.9259 6.256848    27
B  158.0488 6.996253    41
O  156.9655 4.487665    58

> local(
+   .Pairs <- glht(AnovaModel.1, linfct = mcp(Blood_Type = "Tukey"))
+   print(summary(.Pairs)) # pairwise tests
+   print(confint(.Pairs)) # confidence intervals
+   print(cld(.Pairs)) # compact letter display
+   old.oma <- par(oma=c(0,5,0,0))
+   plot(confint(.Pairs))
+   par(old.oma)
+ )

```

```

Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: Tukey Contrasts

Fit: aov(formula = Height ~ Blood_Type, data = Personal)

Linear Hypotheses:
      Estimate Std. Error t value Pr(>|t|)
AB - A == 0 -0.02792    1.35013  -0.021  1.000
B - A == 0  0.09493    1.17605   0.081  1.000
O - A == 0 -0.98833    1.06513  -0.928  0.788
B - AB == 0  0.12285    1.46151   0.084  1.000
O - AB == 0 -0.96041    1.37383  -0.699  0.896
O - B == 0 -1.08326    1.20319  -0.900  0.803
(Adjusted p values reported -- single-step method)

Simultaneous Confidence Intervals

Multiple Comparisons of Means: Tukey Contrasts

Fit: aov(formula = Height ~ Blood_Type, data = Personal)

Quantile = 2.5884
95% family-wise confidence level

Linear Hypotheses:
      Estimate lwr      upr
AB - A == 0 -0.02792 -3.52266  3.46682
B - A == 0  0.09493 -2.94920  3.13907
O - A == 0 -0.98833 -3.74536  1.76870
B - AB == 0  0.12285 -3.66018  3.90589
O - AB == 0 -0.96041 -4.51651  2.59569
O - B == 0 -1.08326 -4.19764  2.03112

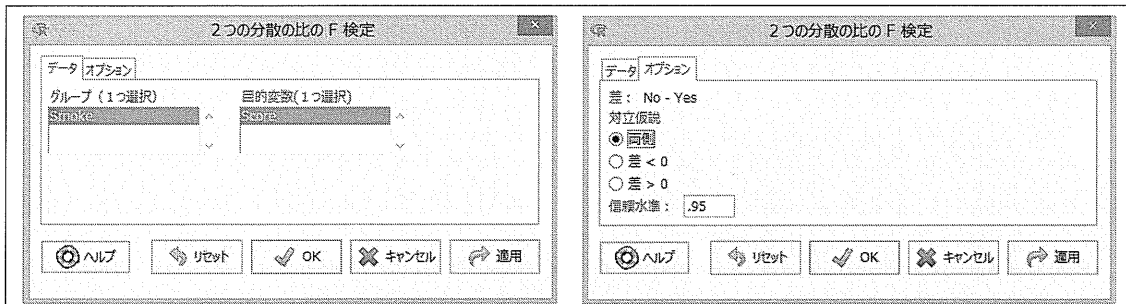
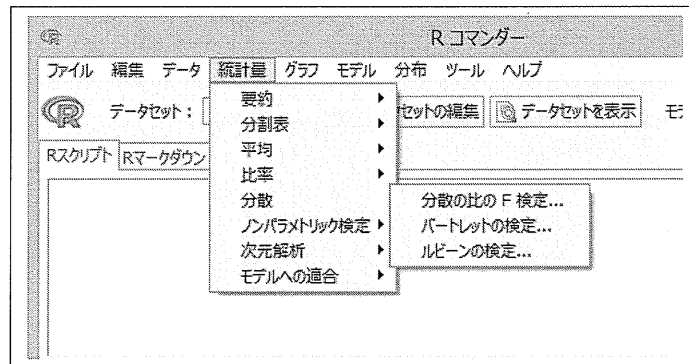
A AB B O
"a" "a" "a" "a"

```

## § 4.7

## F 検定

1. 【統計量】 → 【分散】 → 【分散の比の F 検定】 を選びます。
2. 【データ】 タブにおいて, 【グループ】 を選択し, 【目的変数】 も選びます。
3. 【オプション】 タブにおいて, 【対立仮説】 を選択し, 【信頼水準】 も設定します。
4. 【OK】 を押します。



次のようにスクリプトが生成され, 結果が表示されます。

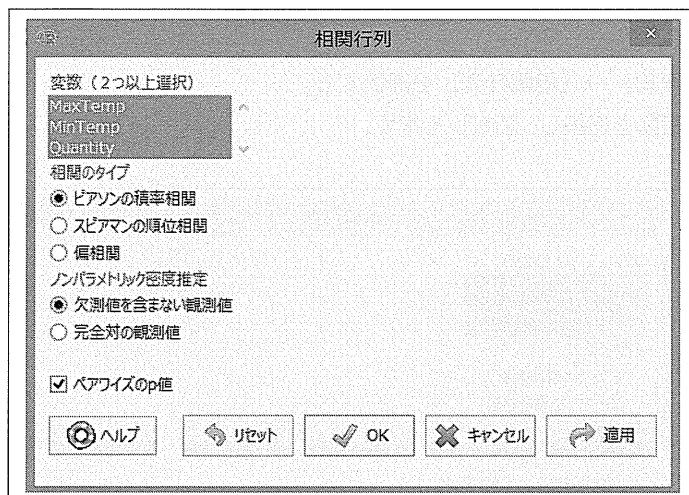
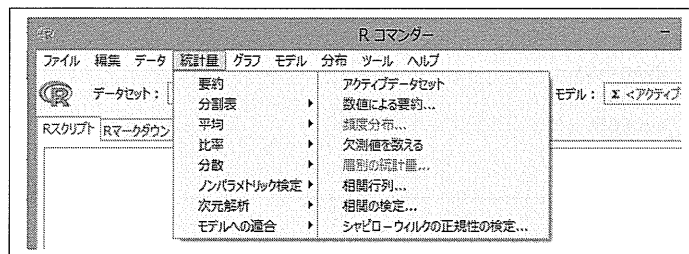
```
> with(Smoke, tapply(Score, Smoke, var, na.rm=TRUE))
      No      Yes
154.2636 132.8011

> var.test(Score ~ Smoke, alternative='two.sided', conf.level=.95, data=Smoke)
F test to compare two variances
data:  Score by Smoke
F = 1.1616, num df = 123, denom df = 32, p-value = 0.6392
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.6351378 1.9382308
sample estimates:
ratio of variances
 1.161613
```

## § 4.8

## 相関の検定

1. データセットをアクティブにします (【データセット:】 からデータを選択)。
2. メニューバーの【統計量】 → 【要約】 → 【相関行列】 を選択します。
3. 【変数】 の中から変数を2つ以上入れます。
4. 【相関のタイプ】 については、連続量の場合にはピアソンの積率相関係数を選び、順序尺度の場合にはスピアマンの順位相関係数を選びます
5. 【ペアワイズの p 値】 にチェックを入れ、【OK】 を押します。



次のようにスクリプトが生成され、結果が表示されます。

```
> rcorr.adjust(IceCream[,c("MaxTemp","MinTemp","Quantity")], type="pearson", use="complete")
```

Pearson correlations:

	MaxTemp	MinTemp	Quantity
MaxTemp	1.000	0.6430	0.7830
MinTemp	0.643	1.0000	0.4178
Quantity	0.783	0.4178	1.0000

Number of observations: 20

Pairwise two-sided p-values:

	MaxTemp	MinTemp	Quantity
MaxTemp		0.0022	<.0001
MinTemp	0.0022		0.0668
Quantity	<.0001	0.0668	

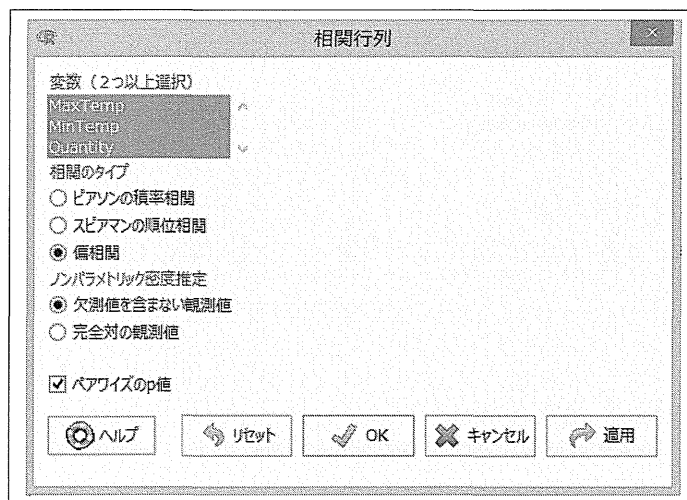
Adjusted p-values (Holm's method)

	MaxTemp	MinTemp	Quantity
MaxTemp		0.0045	0.0001
MinTemp	0.0045		0.0668
Quantity	0.0001	0.0668	

## § 4.9

### 偏相関の検定

1. 相関の検定を参考に、【要約】 → 【相関行列】 を選択する。
2. 【相関のタイプ】 で偏相関を選択し、【OK】 を押す。



次のようにスクリプトが生成され、結果が表示されます。

```
> partial.cor(IceCream[,c("MaxTemp","MinTemp","Quantity")], tests=TRUE, use="complete")
```

```
Partial correlations:
      MaxTemp  MinTemp  Quantity
MaxTemp 0.00000  0.55887  0.73924
MinTemp  0.55887  0.00000 -0.17982
Quantity 0.73924 -0.17982  0.00000
```

```
Number of observations: 20
```

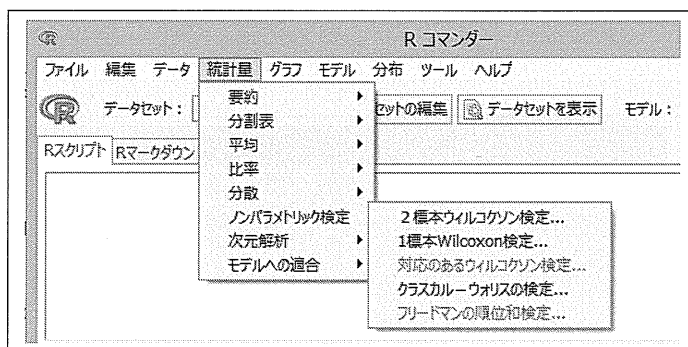
```
Pairwise two-sided p-values:
      MaxTemp  MinTemp  Quantity
MaxTemp      0.0129  0.0003
MinTemp      0.0129  0.4613
Quantity      0.0003  0.4613
```

```
Adjusted p-values (Holm's method)
      MaxTemp  MinTemp  Quantity
MaxTemp      0.0257  0.0009
MinTemp      0.0257  0.4613
Quantity      0.0009  0.4613
```

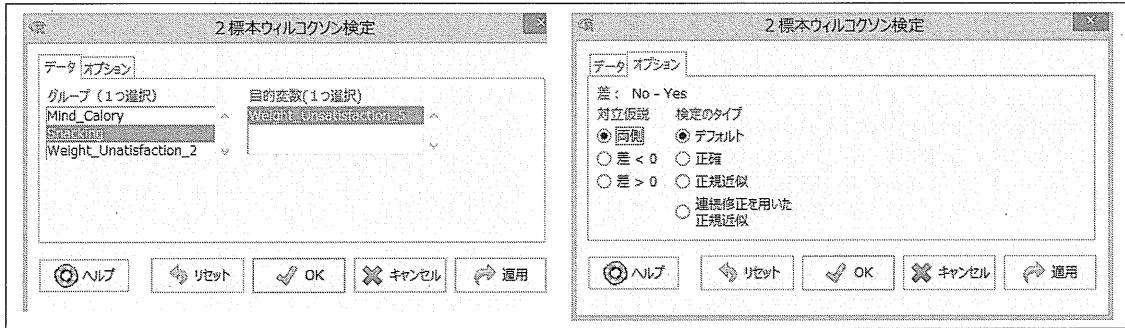
## § 4.10

### Wilcoxon の順位和検定 (Mann-Whitney の $U$ 検定)

1. 【統計量】 → 【ノンパラメトリック検定】 → 【2 標本ウィルコクソン検定】 を選びます。
2. 【データ】 タブにおいて【グループ】 を選択し、【目的変数】 も選びます。
3. 【オプション】 タブにおいて【対立仮説】 を選択し、【検定のタイプ】 では【デフォルト】 のままにします。
4. 【OK】 を押します。







次のようにスクリプトが生成され、結果が表示されます。

```
> with(Calory, tapply(Weight_Unsatisfaction_5, Snacking, median, na.rm=TRUE))
No Yes
 2   4

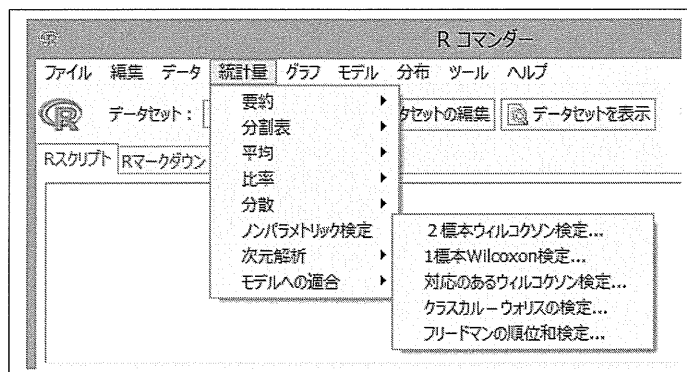
> wilcox.test(Weight_Unsatisfaction_5 ~ Snacking, alternative="two.sided", data=Calory)

Wilcoxon rank sum test with continuity correction
data: Weight_Unsatisfaction_5 by Snacking
W = 99.5, p-value = 0.002094
alternative hypothesis: true location shift is not equal to 0
```

## § 4.11

### Kruskal-Wallis の検定

1. 【統計量】 → 【ノンパラメトリック検定】 → 【クラスカル・ウォリスの検定】 を選びます。
2. 【グループ】 を選択し、【目的変数】 も選びます。
3. 【OK】 を押します。





次のようにスクリプトが生成され、結果が表示されます。

```
> with(Personal, tapply(extroversion, Blood_Type, median, na.rm=TRUE))
  A AB  B  O
  2  2  2  2

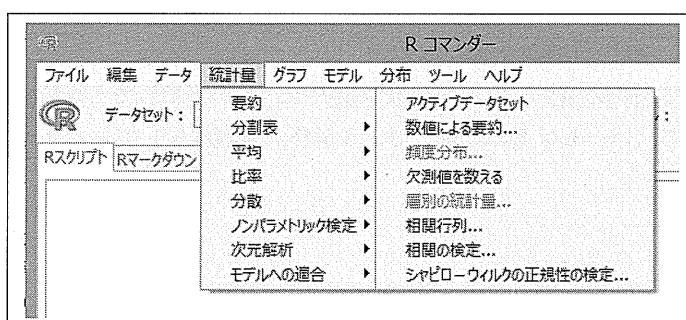
> kruskal.test(extroversion ~ Blood_Type, data=Personal)

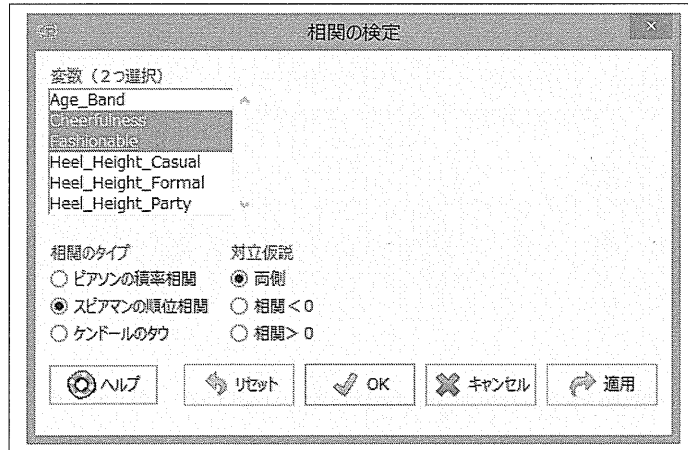
Kruskal-Wallis rank sum test
data:  extroversion by Blood_Type
Kruskal-Wallis chi-squared = 3.3284, df = 3, p-value = 0.3437
```

## § 4.12

### Spearman の順位相関係数の検定

1. 【統計量】 → 【要約】 → 【相関の検定】 を選びます。
2. 【変数】 を 2 つ 選択し, 【相関のタイプ】 はスピアマンの順位相関を選びます. 【対立仮説】 も選びます。
3. 【OK】 を押します。





次のようにスクリプトが生成され、結果が表示されます。

```
> with(Heel, cor.test(Cheerfulness, Fashionable, alternative="two.sided",
+ method="spearman"))
```

Spearman's rank correlation rho

data: Cheerfulness and Fashionable

S = 43224.27, p-value = 0.009177

alternative hypothesis: true rho is not equal to 0

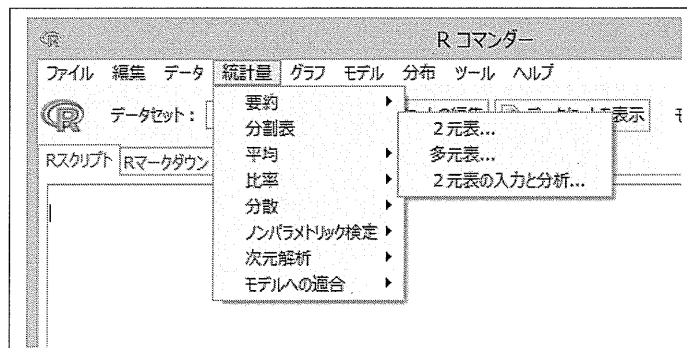
sample estimates:

```
rho
0.3050314
```

## § 4.13

### Fisher の正確検定 ( $\chi^2$ 検定)

1. 【統計量】 → 【分割表】 → 【2元表】 を選びます。
2. 【データ】 タブにおいて、【行の選択】 で1つ変数を選択し、【列の選択】 でも変数を1つ選びます。
3. 【統計量】 タブにおいて、【パーセントの計算】 で1つ選び、【仮説検定】 ではフィッシャーの正確検定にチェックを入れます。
4. 【OK】 を押します。





次のようにスクリプトが生成され、結果が表示されます。

```
> local({
+   .Table <- xtabs(~Blood_Type+Marriage, data=Personal)
+   cat("
nFrequency table:
n")
+   print(.Table)
+   print(fisher.test(.Table))
+ })
```

```
Frequency table:
      Marriage
Blood_Type Married Single
      A      29      35
      AB     12      15
      B      22      18
      O      26      29
```

Fisher's Exact Test for Count Data

```
data: .Table
p-value = 0.7727
alternative hypothesis: two.sided
```

## § 4.14

### $\chi^2$ 検定

1. 【統計量】 → 【分割表】 → 【2元表】 を選びます。
2. 【データ】 タブにおいて、【行の選択】 で1つ変数を選択し、【列の選択】 でも変数を1つ選びます。
3. 【統計量】 タブにおいて、【パーセントの計算】 で1つ選び、【仮説検定】 では独立性のカイ2乗検定にチェックを入れます。
4. 【OK】 を押します。