

## 14日目：t検定（対応のない場合）

さて、本日は因子分析の結果から構成された下位尺度得点について、t検定を行ってみたいと思います。

SPSSでt検定を行ったことのある人は、等分散性の検定結果を見て、上下どちらの結果を読み取るかを定める…といった手順を覚えていると思います。等分散を仮定できない場合の検定はwelchのt検定などとも呼ばれますが、Rのデフォルトはこのwelchの方法になります。

どうせRでやるなら、すべてwelchの方法でやればいいのかと思いますが…

やろうとしていることを確認しますが、因子分析の結果から構成された下位尺度得点（total\_f1, total\_f2, total\_f3）に対して、性についてのt検定を行おうということです。しかしまずやるべきなのは、対象となる変数の分布や基礎統計量、またその群別の様子です。検定をやってから、平均値を確認するのは順序が逆です！

方法は、7日目にやったdescribeやdescribeByが便利でしょう。

これらは複数の変数に一度にやれるので、以下のようにすれば楽かもしれません。

```
label_f <- c("total_f1", "total_f2", "total_f3")
describe(x[label_f])
describeBy(x[label_f], x$性別)
```

出力結果を整理すると、下表のような状況になります。このような差について検定をするわけです。

		n	平均値	標準偏差	標準誤差
total_f1	機能性	298	18.39	4.11	0.24
	男性	125	17.76	4.25	0.38
	女性	173	18.85	3.96	0.30
total_f2	デザイン性	297	13.32	3.87	0.22
	男性	125	12.62	3.73	0.33
	女性	172	13.82	3.89	0.30
total_f3	グレード感	298	8.08	3.20	0.19
	男性	125	8.49	3.47	0.31
	女性	173	7.79	2.97	0.23

またヒストグラムも作ってみましょう。9 日目に使った `hist` を使って、`hist(x$total_f1)` でまずは様子を見てみます。縦軸の最大値が 60 なので、これを目安に男性、女性のヒストグラムを作成して眺めてみましょう。

7 日目に使った `subset` を使って、性別にデータのファイルを作成し、ひとつの Quartz ウィンドにまとめてヒストグラムを表示させます。また、男女のヒストグラムを重ねて表示させることもしてみました。必須ではありませんが、こういう表示の仕方できます。`"#ff4b0040"` や `"#ff4b00"` は色、透過度の指定です (詳しくは web など調べてみてください)。`ylim=` は縦軸の最小と最大を指定します。`add = TRUE` は一つ前の図に重ね書きをせよという命令です。

```
x.m <- subset(x, 性別==1)
x.f <- subset(x, 性別==2)
```

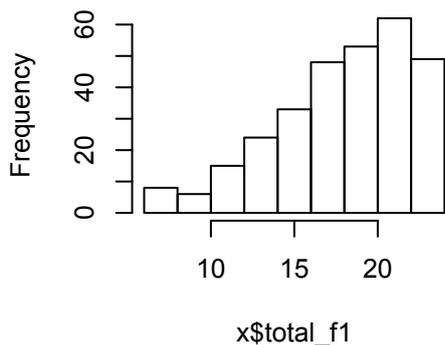
```
par(mfrow=c(2,2))
hist(x$total_f1)
hist(x.m$total_f1, ylim=c(0,60))
hist(x.f$total_f1, ylim=c(0,60))
hist(x.f$total_f1, col = "#ff4b0040", border = "#ff4b00", ylim=c(0,60))
hist(x.m$total_f1, col = "#005aff50", border = "#005aff", ylim=c(0,60),
add = TRUE)
```

結果として、以下のようなグラフを得ることができます。f1 の機能性は、男性の平均値が 17.76、標準偏差は 4.25 でした。女性は平均が 18.85、標準偏差 3.96 でした。女性の平均値の方が高めですが、確かに高得点者が多い傾向がヒストグラムからも確認できます。男性はなだらかな丘のような分布なので、標準偏差が女性より大きめです。

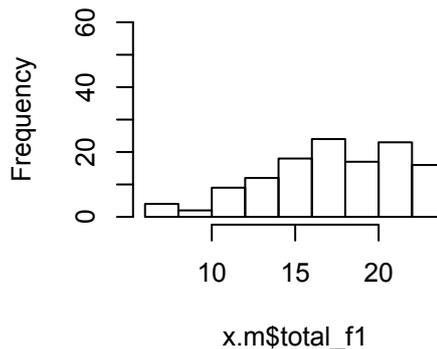
ただし、このような傾向をチェックするにあたっては、男女の人数に 50 人ほどの差があるとところが気になります。縦軸は度数なので、群の人数に多少があると比較がしにくくなります。`hist` では、縦軸を確率密度 (probability densities) にすることもできます。以下は重ね書きの部分ですが、`freq = FALSE` を入れることで確率密度にできます (デフォルトは `TRUE` で頻度)。なお、縦軸の最大値はデータによって大きく変わるので、結果表示を見ながら調整する必要があります。

```
hist(x.f$total_f1, col = "#ff4b0040", border = "#ff4b00", freq = FALSE,
ylim=c(0,0.12))
hist(x.m$total_f1, col = "#005aff50", border = "#005aff", freq = FALSE,
ylim=c(0,0.12), add = TRUE)
```

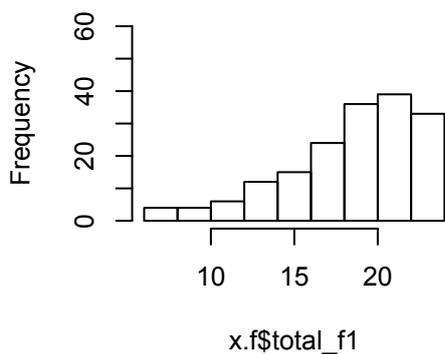
**Histogram of x\$total\_f1**



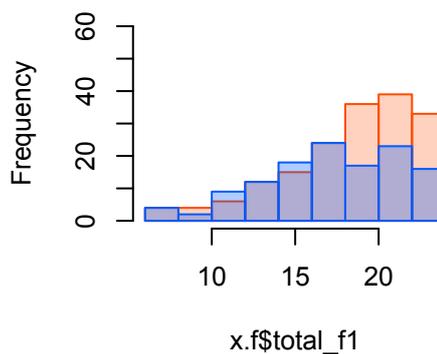
**Histogram of x.m\$total\_f1**



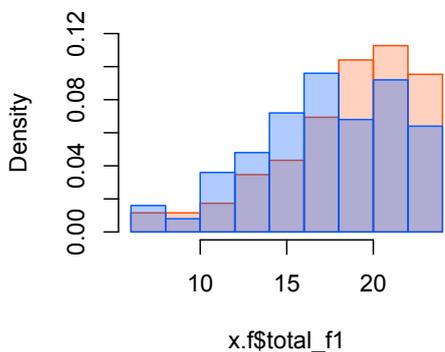
**Histogram of x.f\$total\_f1**



**Histogram of x.f\$total\_f1**



**Histogram of x.f\$total\_f1**



ふたつの重ね書きの図を比較すると、違いがわかると思います。確率の方を見ると、平均値付近を境に、低得点側では男性が多く、高得点側で女性が多い傾向が把握できます。

事前準備の説明が長くなりましたが、このようなチェックをした後に  $t$  検定をやります。等分散を仮定しない `welch` の  $t$  検定を実行する命令は以下のようです。

```
t.test(x$total_f1 ~ x$性別)
```

カッコ内は、前に平均値が検定される変数を、「`~`」で区切って、後ろに群を分ける変数を入れます。

結果は以下のように表示されます。

```
> t.test(x$total_f1 ~ x$性別)

Welch Two Sample t-test

data:  x$total_f1 by x$性別
t = -2.2489, df = 255.75, p-value = 0.02537
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -2.0439134 -0.1355086
sample estimates:
mean in group 1 mean in group 2
    17.76000     18.84971
```

3行目に、 $t$  値、自由度 ( $df$ )、 $p$  値が並んでいます。これだと、5%水準で有意な差があるという結果ですね。その下は対立仮説 (alternative hypothesis)、95%信頼区間 (95%CI) と並び、最後にそれぞれの群の平均値が示されています。

まとめて一気にやりたいなら、以下のように `for` を使って列番号をしてしてやることもできます。でも、変数名が `a` と表示されてしまうので、間違えてしまいそう…

```
for(a in **:*) {
  print(t.test(x[,a] ~ x$性別))
}
```

結果を一覧にしたら、以下ようになります。もちろん、 $t$  値は絶対値表示にして、 $df$  は小数点以下1桁で示しています。論文なら、「 $t$  検定を行った」と書くところを、「`welch` の  $t$  検定を行った」としておけば十分なのではないでしょうか。

			n	平均値	標準偏差	t値 自由度
total_f1	機能性	男性	125	17.76	4.25	2.25 *
		女性	173	18.85	3.96	255.8
total_f2	デザイン性	男性	125	12.62	3.73	2.68 **
		女性	172	13.82	3.89	273.6
total_f3	グレード感	男性	125	8.49	3.47	1.83
		女性	173	7.79	2.97	241.0

\* p<.05; \* p<.01

これで十分だと思っておりますが、等分散の検定をして、等分散ならその場合の  $t$  検定をしたという場合は、以下のような感じです。なお以下の `total_f1` では、等分散の検定の結果、有意ではない（帰無仮説が棄却されない）ので、等分散とみなしてよいという判断になります。

- ・等分散の検定

```
var.test(x$total_f1 ~ x$性別)
```

- ・等分散を仮定する場合の  $t$  検定

```
t.test(x$total_f1 ~ x$性別, var.equal=TRUE)
```

その他、対応のある場合の  $t$  検定とか、1 標本の  $t$  検定とかいろいろありますが、必要な場合には適宜情報をさぐってください。

最後に、近年は情報として記載がすすめられる効果量として、`psych` パッケージにある `cohen.d` を紹介しておきます。その名のまま、`choen's d` を計算してくれます。

```
label_d <- c("性別", "total_f1", "total_f2", "total_f3")
dat_d <- x[label_d]
cohen.d(dat_d, "性別")
```

データの指定がちょっとやっかいで、`cohen.d(x$total_f1, x$性別)` というような設定を認めてくれません。データセット自体をカッコ内のはじめに置かなければならないようです。`cohen.d(x, "性別")` とするとエラーが出たので、必要な変数だけを集めたものを作りました。

結果の一部は以下のようなようです。effect の下の数値が choen' s  $d$  です。2群の平均値の差は絶対値として求めているようなので、負の値にもなります。今回の結果は有意とはいえ、効果量はかなり小さいようです。

```
> cohen.d(dat_d, "性別")
Call: cohen.d(x = dat_d, group = "性別")
Cohen d statistic of difference between two means
      lower effect upper
total_f1 0.035    0.27 0.500
total_f2 0.079    0.31 0.547
total_f3 -0.452  -0.22 0.011
```

本日はここまでにします。