```
########################################
########################################
####### [KAI] カイ二乗分布 ############
########################################
########################################


library(ggplot2)


#----------------------------------------
# [準備１] データフレームをインポート
#----------------------------------------


# 以下から、奇数偶数の好み実験結果の架空（N=1000）のデータをインポートしてください。
url = "https://lab.kenrikodaka.com/_download/csv/oddeven_1000.csv"
source = read.csv(url)

#誕生日と奇数偶数の好みに関するアンケートの架空のデータ（1000人分）です。
#month（誕生月）・name（誕生日）・preference（奇数が好き：1, 偶数が好き：0）
#gender（男性：1, 女性：0）・$domhand（利き手が左：１, 利き手が右：0）
#age（年齢）

#最初の6行をちょっと出し
head(source)
#  month day preference gender domhand age
#1     8  29          1      1       0  19
#2     1  29          0      0       1  18
#3     3  19          0      1       0  18
#4    10   6          0      0       0  18
#5    10  10          1      0       0  18
#6     1  13          0      1       0  19

# 翻訳すると以下の様になります。
# 08月29日、奇数好き、男性、右利き、19歳
# 01月29日、偶数好き、女性、左利き、18歳
# 03月19日、偶数好き、男性、右利き、18歳
# 10月06日、偶数好き、女性、右利き、18歳
# 10月10日、奇数好き、女性、右利き、18歳
# 01月13日、偶数好き、男性、右利き、19歳




#----------------------------------------
# [準備２] データフレームから集計表をつくる
#----------------------------------------
```

```
# table関数を使うと、指定した変数に関する集計表（対応表）を
# 簡単に作ることができます。

# 「誕生月」と「好み」の関係の集計表（いずれも同じ結果となります。）
table(source$month,source$preference)
table(source[c("month","preference")])
table(source[c(1,3)])
#     preference
#month 0 1
#   1 52 38
#   2 42 18
# ...
#   12 52 30


# 「day」と「好み」の集計表
table(source$day,source$preference)
#    preference
#day 0  1
# 1  19 18
# 2  21  7
# ...
# 31 15 10

# 「性別」と「好み」の集計表
table(source$gender,source$preference)
#         preference
# gender   0   1
#     0 370 203
#     1 230 197


#---------------------------------------
# [準備３] ファクタ
#---------------------------------------

# sourceをexpdatにコピー
expdat = source

# 量的変数（0 1）を、量を持たないカテゴリ変数に変換します。
# 以下の例では、性別の0-1を"FEMALE"-"MALE"に、
# 利き手の0-1を"RIGHT"-"LEFT"に、
# 奇数偶数の好みの0-1を"EVEN"-"ODD"の文字列カテゴリに割り当てます。
# このようなデータのタイプをファクタと呼びます。

expdat$gender =
factor(source$gender,levels=0:1,labels=c("FEMALE","MALE"))
expdat$domhand =
factor(source$domhand,levels=0:1,labels=c("RIGHT","LEFT"))
expdat$preference =
factor(source$preference,levels=0:1,labels=c("EVEN","ODD"))
```

```
# このように変わります。
expdat$preference
#[1] ODD  EVEN EVEN EVEN ODD  EVEN EVEN EVEN EVEN EVEN EVEN EVEN
EVEN EVEN EVEN
# . . .
#[976] EVEN EVEN EVEN ODD  ODD  ODD  EVEN EVEN EVEN EVEN ODD  ODD
EVEN EVEN ODD
#[991] EVEN ODD  EVEN ODD  ODD  EVEN EVEN EVEN ODD  ODD

# クラスはfactorになります。
class(expdat$preference)
#[1] "factor"

# factorの型は常にintegerです。
typeof(expdat$preference)
#[1] "integer"

# str関数を実行すると、ファクタの構造がわかります。
str(expdat$preference)
#Factor w/ 2 levels "EVEN","ODD": 2 1 1 1 2 1 1 1 1 1 ...

# Factorには順序があり、1には"EVEN"、2には"ODD"が対応しています。
# これは、factorが、各カテゴリを特定の整数インデックス（1,2,..）と対応させているためです。
# これを確認するには、ファクタを解除する「as.numeric」または「unclass」を使います。
as.numeric(expdat$preference)
#[1] 2 1 1 1 2 1 1 1 1 1 1 1 1 1 1 2 1 1 1 2 1 2 2 2 2 2 1 1 1 1 1 1
2 1 2 2 1 1 1 1 2
#[42] 1 1 1 1 2 1 2 1 1 1 2 1 1 1 1 1 1 2 2 2 1 1 2 1 1 1 1 1 2 1 1
1 2 1 1 1 2 1 1 2 1
unclass(expdat$preference)
#[1] 2 1 1 1 2 1 1 1 1 1 1 1 1 1 1 2 1 1 1 2 1 2 2 2 2 2 1 1 1 1 1 1
2 1 2 2 1 1 1 1 2
#[42] 1 1 1 1 2 1 2 1 1 1 2 1 1 1 1 1 1 2 2 2 1 1 2 1 1 1 1 1 2 1 1
1 2 1 1 1 2 1 1 2 1

# このようにファクタ化されたpreferenceは
# 1がEVEN、2がODDに対応していることに注意してください。

# ファクタの順序は、凡例などの順序に反映されます。
# もし、ODD（1）、EVEN（2）の順に変数を管理したければ以下のようにします。
expdat$preference2 =
factor(source$preference,levels=1:0,labels=c("ODD","EVEN"))
str(expdat$preference2)
#Factor w/ 2 levels "ODD","EVEN": 1 2 2 2 1 2 2 2 2 2 ...

## preference2を消去します。
expdat = expdat[,1:6]
```

```
# Factorにすると、各カテゴリが何を意味するか一目瞭然となります。
table(expdat$gender,expdat$preference)
#        preference
# gender  EVEN ODD
# FEMALE  370 203
# MALE    230 197


# ついでに月もファクタの変えてみましょう。
# あらかじめ用意されている、便利な月の文字列ベクトルを使いましょう。
# month.abbはmonthのabbreviation（略語）の意味です。
month.abb
#[1] "Jan" "Feb" "Mar" "Apr" "May" "Jun" "Jul" "Aug" "Sep" "Oct"
"Nov" "Dec"

expdat$month = factor(source$month,levels=1:12,labels=month.abb)
expdat$month
#[1] Aug Jan Mar Oct Oct Jan Jul Aug Nov Apr Dec Jul Oct May Apr Mar
Jan Jul
#[19] Apr Apr Jan Jan Jul Apr Dec Nov Oct Jun May Dec Dec Aug Mar
Dec Sep Nov

table(expdat$month,expdat$preference)
#        preference
# month EVEN  ODD
#   Jan 52 38
#   Feb 42 18
#...
#   Dec 52 30

# もとの数字として扱いたい時は（as.xxxは「xxxとして」と読んでください）
as.numeric(expdat$month)
#[1]  8  1  3 10 10  1  7  8 11  4 12  7 10  5  4  3  1  7  4  4  1
1  7  4 12
#[26] 11 10  6  5 12 12  8  3 12  9 11 10  1 12  3  6  1  6  8 10  9
12  3  1  8

# unclassも同じ効果があります。
unclass(expdat$month)
#[1]  8  1  3 10 10  1  7  8 11  4 12  7 10  5  4  3  1  7  4  4  1
1  7  4 12
#[26] 11 10  6  5 12 12  8  3 12  9 11 10  1 12  3  6  1  6  8 10  9
12  3  1  8




########################################
########################################
```

```
####### [KAI] カイ二乗分布 ############
########################################
########################################

#----------------------------------------
# [KAI.1] 適合度検定：サイコロの例題
#----------------------------------------

# サイコロを60回振った時の各目が以下のようになる。
obs1 = c(5,8,10,20,7,10) #観測事象１
obs2 = c(15,8,14,6,8,9)  #観測事象２

# それぞれの目の出現確率が1/6のとき、期待値は？
expected = c(10,10,10,10,10,10)

# 観測値と期待値が対応するデータフレームを作成します。
dat = data.frame(OBS1 = obs1, OBS2 = obs2, EX = expected)


# 観測事象１と２はそれぞれ、
# 偏りのないサイコロからの出力と言えるか？
# 期待値からのズレを基にして検証する。


# （OBS1の場合）
# 1の出現数の期待値からズレを以下のように標準化します。
# ズレ：5（1の出現数）−10（期待値） = −5
# 二乗して正にする (5−10)∗(5−10) = 25
# 期待値で割ることで正規化：{(5−10)∗(5−10)}/10 = 2.5
# これが1の期待値とのズレを表す統計量

# 同じ操作を1から6まで全て行い加算したものをkai2.OBS1とする
kai2.OBS1 = sum((dat$OBS1 - dat$EX)^2 / dat$EX)
kai2.OBS1
#[1] 13.8

# （OBS2の場合）
# OBS2に対しても同様に計算する
kai2.OBS2 = sum((dat$OBS2 - dat$EX)^2 / dat$EX)
kai2.OBS2
#[1] 6.6

# このようにして計算される「期待値からのズレの二乗和」を
# 「カイ二乗値（χ^2）」と呼びます。
```

```
# 「カイ二乗値」は、Rでは
# chisq.test（観測値ベクトル, p=期待値の確率分布）で求められます。
# 2つ目の引数：期待値の確率分布の総和は1となっている必要があります。
# 今回は, 期待値の確率分布はc(1/6,1/6,1/6,1/6,1/6,1/6)となるので、
result.OBS1 = chisq.test(dat$OBS1,p = rep(1/6,times=6))
result.OBS1$statistic #返り値の名前属性$statisticがχ^2に対応
#X-squared
#     13.8

#F2についても同様に
result.OBS2 = chisq.test(dat$OBS2,p = rep(1/6,times=6))
result.OBS2$statistic
#X-squared
#       6.6


# この統計量（kai2）が大きな値をとるほど、ランダム事象とのズレが大きく、
# 偶然では起こりにくい事象であることがわかります。
# それでは、実際に、サイコロの出力が完全にランダムなときに、
# kai2がどのような分布をとるかを調べましょう。

# ここでは、多数のランダム試行のシミュレーション（モンテカルロ法）により、
# おおよその「当たり」をつけてみます。

# 60回サイコロを振った時のkai.2を算出する関数（getDiceKai2）を作ります。

getDiceKai2 = function(){

  # 60回サイコロを振りベクトルに展開します。
  dice.60 = sample(1:6,60,replace=TRUE);

  # 各出目の出現数のベクトルを作ります。
  dice.6 = vector("integer",6)
  for(i in 1:6){
    dice.6[i] = length(which(dice.60==i))
  }
  # 各出目の期待値は
  dice.ex = c(10,10,10,10,10,10)
  # 期待値からのズレの統計量を各出目毎に加算
  sum((dice.6 - dice.ex)^2  / dice.ex);
}

set.seed(17) #乱数を固定します（同じ出力とするため）。
# getDiceKai2()を100000回実行し、
# その統計量を集めます。
simulation = replicate(100000,getDiceKai2())
simulation
#[1]  1.4  1.6  7.2  2.6  0.8  5.8  2.4  3.0 14.2  8.2  7.6  5.8
6.6  3.4  8.0
```

```
#[16]  2.4  4.8  4.6 16.4  3.4  1.6  3.2  2.6  3.2  3.6  6.4  3.6
6.4  3.0  8.2
#...

#最小値・下位25%・中央値・平均・上位25%・最大値
summary(simulation)
#Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#0.000   2.600   4.400   5.017   6.600  31.200


# ヒストグラムの出力（ggplot2）
# X軸は、0から最大値より大きな値（35）まで,
# 0.2刻みでベクトルを計算
ggplot(NULL,aes(x=simulation)) +

geom_histogram(breaks=seq(0,35,by=0.2),fill="white",colour="black")

# ヒストグラムの計算（リストに出力）
# X軸は、0から最大値より大きな値（35）まで,
# 0.2刻みでベクトルを計算
h = hist(simulation,breaks = seq(0,35,by=0.2))

h$breaks #横軸の刻み 0.0 0.2, ... 34.8 35
h$count #横軸に対応する頻度数（Frequency）222  573  815 ...
h$density #横軸に対応する確率密度 0.01110 0.02865 0.04075 ...

# hist関数はデフォルトでは頻度を縦軸に出力
# 縦軸を確率密度にするには、引数でfreq=FALSEを指定
h = hist(simulation,breaks = seq(0,35,by=0.2),freq=FALSE)

# 以下も同じです（ggplot2を使う場合）
ggplot(NULL,aes(x=simulation)) +
  geom_histogram(aes(y=..density..), #確率密度でY軸を表示

breaks=seq(0,35,by=0.2),fill="white",colour="black")


# getDiceKai2が13.8(kai2.obs1)より大きくなる確率は？
sum(h$density[h$breaks>=13.8]*0.2,na.rm = T)#NAを無視
#0.01711
# → kai2.F1程度の偏りは1.7%程度でしか生じない
# → 有意な偏りが存在する。

# getDiceKai2が18（kai2.obs2）より大きくなる確率は？
sum(h$density[h$breaks>=6.6]*0.2,na.rm = T) #NAを無視
#0.24703
# → kai2.F2以上の偏りはおよそ24.7%の確率で生じる。
# → 有意な偏りが存在しない。
```

```
# このような確率を「p値」と呼ぶ。
# 「p値」もまた、Rの「chisq.test」で正確な値を自動的に算出できる。

result.OBS1 = chisq.test(dat$OBS1,p = rep(1/6,times=6))
result.OBS1$p.value
#[1] 0.01693102
# 0.01711でうまく近似できていたことがわかる

result.OBS2 = chisq.test(dat$OBS2,p = rep(1/6,times=6))
result.OBS2$p.value
#[1] 0.2521282
#0.24703でうまく近似できていたことがわかる


# ちょうど5%の境界を求めるには、例えば以下のようにする

kai2 = max(simulation) #最大のシミュレーション値から始める。

repeat{
  # 統計量がkai2より大きくなる確率は？
  ratio = sum(h$density[h$breaks>kai2]*0.2,na.rm = T)
  # ratioが5%を初めて上回るところで
  # kai2をコンソールにプリントして、繰り返し計算を終了
  if(ratio>=0.05){
    print(kai2)
    break;
  }else{
    # それ以外は、kai2を0.01低くして再実行
    kai2 = kai2 - 0.01
  }
}
#11

# 以上より、観測するカテゴリが「6」のとき、
# カイ二乗の統計量が11付近より大きい場合,
# 偶然以上の偏りを持っていると結論できる。



#----------------------------------------
# [KAI.2] 適合度検定（男女別、奇数と偶数の好み）
#----------------------------------------

# 女性の偶数好き:370、奇数好き:203
preference = c(370,203)

# データフレームから求める場合、例えば以下の方法があります。
```

```
expdat_female = expdat[expdat$gender=="FEMALE",]
preference = table(expdat_female$preference)
preference #名前付きのベクトル
#EVEN  ODD
# 370  203

# 好みがランダムの時の期待値は共に総数の半分
expected = c(0.5*sum(preference),0.5*sum(preference))
expected
#[1] 286.5 286.5

# 定義通り計算すると
kai2 = sum((preference - expected)^2 / (0.5 * sum(preference)))
kai2
#[1] 48.6719

# Rの関数を使うと
result = chisq.test(preference,p = c(0.5,0.5))
result$statistic
#X-squared
# 48.6719


# モンテカルロ法でχ^2の分布をみます。

getPrefKai2 = function(){
  odd.sample = sum(sample(c(0,1),573,replace=T))
  even.sample = 573 - odd.sample
  result = chisq.test(c(even.sample,odd.sample),p = c(0.5,0.5))
  kai2 = result$statistic
  unname(kai2) #名前属性を消して返す
}

set.seed(17) #乱数を固定します。
simulation = replicate(100000,getPrefKai2())
summary(simulation)
#    Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
#0.001745  0.085515  0.504363  0.997944  1.272251 20.734729

# χ^2の分布のピークは0付近にあることに注意。
# カテゴリ数（自由度）によってカイ二乗分布のピークは変わります。
h = hist(simulation,breaks = seq(0,22,by=0.1),probability = T)

## ggplot2による描画
ggplot(NULL,aes(x=simulation)) +
  geom_histogram(breaks = seq(0,22,by=0.1)) +
  scale_x_continuous(limits=c(0,22))


# summary関数より,100000回ランダムサンプリングして、
# 最大のカイ二乗値が20付近にあります。
```

```
# つまり、χ^2>48は、100000回に1回も起きないほどの異常な偏り

# 実際、p値は1/(10^12)付近（1000億回に一回程度）
result = chisq.test(c(370,203),p = c(0.5,0.5))
result$p.value
#[1] 3.025703e-12

# 以上より、女性は有意に偶数を好む傾向にある。


# 男性の偏りも以下のように検証できる。
# 男性の偶数好き:230、奇数好き:197

result = chisq.test(c(230,197),p = c(0.5,0.5))
result$statistic #統計量は2.55
# X-squared
# 2.550351
result$p.value #p値>0.11
# [1] 0.1102697

# すなわち、230/197程度の偏りは、100回に11回程度は生じる。
# 男性が有意に偶数を好む傾向にあるとは言えない。



#----------------------------------------
# [KAI.3] 適合度検定（統計量と自由度）
#----------------------------------------

# カテゴリ数（自由度+1）が異なると
# χ^2分布がどのように変わるかを確認します。

# カテゴリ数2、6、10の分布を比較しましょう。

# 対応するカテゴリ数
## 2が10000個、6が10000個、10が10000個
category = c(rep(2,10000),rep(6,10000),rep(10,10000))

# 10000 x 3のカイ2乗値を収納するベクトル
kai2 = vector("double",30000)

# カテゴリ数をnとして、
# n=2、6、10の3パターンで、
# それぞれ10000のkai^2サンプルを集める
for(n in c(2,6,10)){

  obs = vector("integer",n)   # 観測ベクトル
  expected = rep(1/n,times=n)  # 期待値ベクトル
```

```
  # 各カテゴリ数nで10000回、kai^2値のサンプルを集めます。
  for(i in 1:10000){

    # 1からnのいずれかのカテゴリを500回ランダムに生成します。
    t = sample(1:n,500,replace=T)
    # 1からnの各事象の生起回数を（観測値として）obsベクトルにまとめます。
    for(ii in 1:n){
      obs[ii] = sum(t==ii)
    }
    # 対応する添字のサンプルに、現在のobsでの統計量を登録します。
    # unnameは単に名前属性を消すためです。
    kai2[10000*((n-2)/4)+i] = unname(chisq.test(obs,p = expected)
$statistic)

  }
}

# kai^2とcategoryを列名とするデータフレームを作成
kai2_df = data.frame(KAI2 = kai2, CATEGORY = category)

# カテゴリ数別にヒストグラムを出力する
ggplot(kai2_df,aes(x=KAI2)) +
  geom_histogram(breaks = seq(0,20,by=0.2)) +
  facet_grid(CATEGORY ~., scales="free") + #Y軸を可変とする
  theme(
    axis.title.x = element_text(size = 30),
    axis.title.y = element_text(size = 30),
    axis.text.x = element_text(size = 30),
    axis.text.y = element_text(size =20),
    strip.text = element_text(size=25)
  )

# カテゴリ数が増えるに従って、
# ピークがn付近に寄ることがわかるはずです。

# 一般に統計量は、自由度によって分布の形状が変化します。
# 逆に言えば、自由度が同じであれば、
# サイコロであれ、奇数偶数の好みであれ、
# 同一の尺度で期待値からのズレを計測することが可能となります。




#----------------------------------------
# [KAI.4] 適合度検定（誕生日、奇数と偶数の好み）
#----------------------------------------

# あらためてデータフレームをインポートし直します。

url = "https://lab.kenrikodaka.com/_download/csv/oddeven_1000.csv"
```

```
source = read.csv(url)

#sourceをexpdatにコピー
expdat = source

expdat$gender =
factor(source$gender,levels=0:1,labels=c("FEMALE","MALE"))
expdat$domhand =
factor(source$domhand,levels=0:1,labels=c("RIGHT","LEFT"))
expdat$preference =
factor(source$preference,levels=0:1,labels=c("EVEN","ODD"))
#expdat$month = factor(source$month,levels=1:12,labels=month.abb)
# 以後は、月はファクトせずに、そのまま数字として扱いましょう。

# 月ごとの集計は以下でベクトル化できます。
obs = table(expdat$month);
obs
#1   2   3   4   5   6   7   8   9  10  11  12
#90  60 100  90  79  86  98  87  69  82  77  82

# このサンプルは実際の誕生日の分布と比べて偏りがあると言えるでしょうか。
# 期待値ベクトルは、各月の日数を365日で割ることでつくることができます。
exp = c(31,28,31,30,31,30,31,31,30,31,30,31) / 365

chisq.test(obs,p=exp)
# Chi-squared test for given probabilities
#data:  obs
#X-squared = 12.658, df = 11, p-value = 0.3163

# カイ二乗値は12.658です。
# カテゴリ数が12のため、12.66はそれほど大きな値ではありません。
# 実際、p値は0.3163のため、10回に3回程度は、
# 観測値より大きな偏りがあることになります。
# 以上より、観測値に有意な偏りは存在するとは言えません。

# 次に誕生日の日にちが偶数の人の数は
sum(expdat$day %% 2 == 0)
#[1] 489

# この489人の奇偶の好みは
pref_day.even = expdat$preference[expdat$day %% 2 == 0]
pref_day.even
#[1] EVEN ODD  EVEN EVEN EVEN EVEN EVEN ODD  ODD  ODD  EVEN EVEN
EVEN EVEN EVEN
#[16] EVEN EVEN EVEN EVEN EVEN EVEN EVEN ODD  EVEN EVEN EVEN EVEN
ODD  EVEN EVEN
#...

# このうち、偶数・奇数が好きな人の数は,,
even_sum = sum(pref_day.even == "EVEN")
odd_sum = sum(pref_day.even == "ODD")
```

```
c(even_sum, odd_sum)
#[1] 306 183

# 適合度検定
chisq.test(c(even_sum,odd_sum),p=c(0.5,0.5))
#X-squared = 30.939, df = 1, p-value = 2.663e-08

# 誕生日が偶数日の集団は有意に偶数好きが多い。


# 誕生日が奇数の場合は？
pref_day.odd = expdat$preference[expdat$day %% 2 == 1]
even_sum = sum(pref_day.odd == "EVEN")
odd_sum = sum(pref_day.odd == "ODD")
c(even_sum, odd_sum)
#[1] 294 217

# 適合度検定
chisq.test(c(even_sum,odd_sum),p=c(0.5,0.5))
#X-squared = 11.603, df = 1, p-value = 0.0006585

# 誕生日が奇数日の集団も有意に偶数好きが多い。




# 誕生月も誕生日も奇数の場合は？
pref_monthday.odd = expdat$preference[expdat$day %% 2 == 1 &
expdat$month %% 2 == 1]
even_sum = sum(pref_monthday.odd == "EVEN")
odd_sum = sum(pref_monthday.odd == "ODD")
c(even_sum, odd_sum)
#[1] 146 121

# 適合度検定
chisq.test(c(even_sum,odd_sum),p=c(0.5,0.5))
#X-squared = 2.3408, df = 1, p-value = 0.126

# 誕生月も誕生日も奇数の集団の好みに有意な差は存在しない。




#----------------------------------------
# [KAI.5] 独立性検定：「性別」と「数字の好み」
#----------------------------------------

# 偶数・奇数の好みは、性別によって差があるか？
```

```
# 1000人分の性別のサンプル
expdat$gender
#[1] MALE   FEMALE MALE   FEMALE FEMALE MALE   FEMALE MALE   MALE
FEMALE MALE
#[12] FEMALE FEMALE FEMALE FEMALE FEMALE FEMALE MALE   FEMALE MALE
FEMALE MALE
#...

# （性別に対応する）1000人分の好み
expdat$preference
#[1] ODD  EVEN EVEN EVEN ODD  EVEN EVEN EVEN EVEN EVEN EVEN EVEN
EVEN EVEN EVEN
#[16] ODD  EVEN EVEN EVEN ODD  EVEN ODD  ODD  ODD  ODD  ODD  EVEN
EVEN EVEN EVEN
#...

# 「性別」と「好み」の対応に関する2x2の集計表
obs = table(expdat$gender, expdat$preference)
obs
#        EVEN ODD
#FEMALE  370 203
#MALE    230 197

# ggplot2によるグラフの出力｜geom_barによる100%積み上げグラフ
ggplot(expdat,aes(x=gender,fill=preference)) +
  geom_bar(position =
position_fill(),alpha=0.5,colour="black",size=0.5) +
  scale_x_discrete(limits=c("FEMALE","MALE")) +
  scale_y_continuous(labels = scales::percent, breaks =
c(0,0.2,0.4,0.6,0.8,1)) +

scale_fill_brewer(palette="Set1",name="PREFERENCE",labels=c("EVEN","
ODD")) +
  geom_text(aes(label=..count..),stat="count",
            colour="white",size=20,position =
position_fill(vjust=0.5)) +
  theme(
    title = element_text(size = 30),
    legend.title = element_text(size = 25),
    axis.title.x = element_text(size = 30),
    axis.title.y = element_text(size = 30),
    axis.text.x = element_text(size = 30),
    axis.text.y = element_text(size = 30),
    legend.text = element_text(size=25),
    legend.background =
    element_rect(fill="white",colour="black")
  )



# obsの返り値のクラスは「table」です。
class(obs)
```

```
# [1] "table"

# tableはtableはdata.frame等のデータを
# クロス集計した頻度値を格納するのに使用されます。

# 「table」の各セルの値は以下のような書式で参照できます。
obs[1,1]
obs["FEMALE","EVEN"]
#370

obs[,2]
obs[,"ODD"]
#FEMALE   MALE
#203    197

# addmarginsを使うと自動的に合計欄を作ってくれます。
obs = addmargins(obs)
#       EVEN  ODD  Sum
#FEMALE  370  203  573
#MALE    230  197  427
#Sum     600  400 1000

# 観測度数のテーブルを期待度数のテーブルにコピー
expected = obs

# 合計欄をうまく使って、一行で期待度数を計算
for(i in 1:2){
  for(j in 1:2){
    expected[i,j] = obs[i,"Sum"] * (obs["Sum",j] / obs["Sum","Sum"])
    #expected[i,j] = obs[i,3] * (obs[3,j] / obs[3,3]) #同じです。
  }
}
expected
#         EVEN    ODD    Sum
#FEMALE  343.8  229.2  573.0
#MALE    256.2  170.8  427.0
#Sum     600.0  400.0 1000.0


# 合計欄をカット
obs = obs[1:2,1:2]
expected = expected[1:2,1:2]

# 各セルのχ二乗値の計算
kai2_table = (obs-expected)^2 / expected
#         EVEN      ODD
#FEMALE 1.996626 2.994939
#MALE   2.679313 4.018970

# 全てのセルの誤差項を足したものがχ二乗値
kai2 = sum(kai2_table)
#[1] 11.68985
```

```
# chisq.test関数を使うと、上記の計算を自動的に行ってくれます。
# （correct=Fは、イェーツ補正の有無です。ここではFALSEにします）
result = chisq.test(expdat$gender, expdat$preference, correct=F)
#X-squared = 11.69, df = 1, p-value = 0.0006284

result$statistic
#X-squared
#11.68985

# 既にp値も見えていますが、これを実際に確かめてみましょう。


#----------------------------------------
# [KAI.6] カイ二乗値の分布を調べる
#----------------------------------------

# データと条件を揃えて、
# 427人の男性と573人の女性、
# 偶数対奇数の好みが「60:40」のとき

set.seed(1017)
getKai2 = function(){

  # 1:5を人数分だけランダムに生成
  sample.female = sample(1:5,573,replace=TRUE);
  sample.male = sample(1:5,427,replace=TRUE);

  # 1:3を0に4:5を1に変換（floor関数を使っている）
  sample.female = floor(sample.female / 4)
  sample.male = floor(sample.male / 4)

  # 0を573、1を427並べた性別サンプル
  gender = c(rep(0,times=573),rep(1,times=427))
  # 前半（1:573）に女性の好み、後半（574:1000）に男性の好みを対応させる
  pref = c(sample.female,sample.male)

  # カイ二乗値を計算
  result = chisq.test(gender,pref,correct=F)
  result$statistic
}

# getKai2()を10000回繰り返す
sample = replicate(10000,getKai2())
summary(sample)
#Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
#0.000002  0.094335  0.426451  0.982526  1.278155 18.543110

# ヒストグラムを見る。
```

```
par(cex=2) #フォントサイズを挙げる（デフォルトの2倍）
h = hist(sample,breaks = seq(0,20,by=0.2)) #縦軸はFrequency（頻度）
h = hist(sample,breaks = seq(0,20,by=0.2),probability = T) #縦軸は確率密度

# ggplot2による描画場合
ggplot(NULL,aes(x=sample)) +
  geom_histogram(aes(y=..density..),breaks = seq(0,20,by=0.2))

# 横軸が11.69以上の確率密度を全て加算
sum(h$density[h$breaks>11.69],na.rm = T)
# 0.0025

# これがp値に対応。
# 女性と男性との間で、好みが、観測値以上の偏りを示す確率は、
# 0.0025（5%以下）

# すなわち「性別」と「好み」は、違いに独立ではない。
# より具体的に、女性の方が偶数を有意に好む傾向にある。


# ちなみに、chisq.testを使うと、正確なカイ二乗分布を使った
# （近似ではない）p値がわかる。
result = chisq.test(expdat$gender, expdat$preference, correct=F)
result
#X-squared = 11.69, df = 1, p-value = 0.0006284

result$p.value
#0.0006284205

# 自由度は｜（2-1）＊（2-1）
result$parameter
#df
#1


#-------------------------------------------------------
# [KAI.5] 独立性検定の練習（誕生日の数字の偶数と奇数の影響）
#-------------------------------------------------------

# 誕生月の奇数偶数を示す属性をデータフレームに追加
expdat$mcategory = expdat$month %% 2
expdat$mcategory
#[1] 0 1 1 0 0 1 1 0 1 0 0 1 0 1 0 1 1 1 0 0 1 1 1 0 0 1 0 0 1 0 0 0
1 0 1 1 0 1 0 1 0 1 0 0
#[45] 0 1 0 1 1 0 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 1 0 0 0 0 0 0 1 1 0
1 1 0 1 1 1 0 1 0 1 0 0 0
#...

# ファクタに変換
```

```
expdat$mcategory =
factor(expdat$mcategory,levels=0:1,labels=c("EVEN","ODD"))
expdat$mcategory
#[1] EVEN ODD  ODD  EVEN EVEN ODD  ODD  EVEN ODD  EVEN EVEN ODD
EVEN ODD  EVEN ODD  ODD
#[18] ODD  EVEN EVEN ODD  ODD  ODD  EVEN EVEN ODD  EVEN EVEN ODD
EVEN EVEN EVEN ODD  EVEN


# 誕生日（日にち）の奇数偶数を示す属性をデータフレームに追加
expdat$dcategory = expdat$day %% 2
expdat$dcategory
#[1] 1 1 1 0 0 1 0 1 0 1 1 0 0 1 1 1 1 0 1 0 1 1 1 0 0 1 0 0 0 0 0 0
1 0 1 1 0 0 1 1 1 1 1 0
#[45] 0 1 0 0 0 0 1 1 1 0 1 0 1 1 0 1 1 0 0 0 1 1 1 0 1 0 0 0 0 1 0
1 1 0 0 1 1 1 0 0 1 0 1 1#...
#..

# ファクタに変換
expdat$dcategory =
factor(expdat$dcategory,levels=0:1,labels=c("EVEN","ODD"))
expdat$dcategory
#[1] ODD  ODD  ODD  EVEN EVEN ODD  EVEN ODD  EVEN ODD  ODD  EVEN
EVEN ODD  ODD  ODD  ODD
#[18] EVEN ODD  EVEN ODD  ODD  ODD  EVEN EVEN ODD  EVEN EVEN EVEN
EVEN EVEN EVEN ODD  EVEN
#...

############################
# 男性のケース
############################

# 以下では、男性の行のみを集めた以下のデータフレームを使用する
expdat_male = expdat[expdat$gender=="MALE",]

#--------------------------
# 誕生月と好みの関係（男性）
#--------------------------

table(expdat_male$mcategory, expdat_male$preference)
#     EVEN ODD
#EVEN  121  98
#ODD   109  99

# 棒グラフで可視化
ggplot(expdat_male,aes(x=mcategory,fill=preference)) +
  geom_bar(position =
position_fill(),alpha=0.5,colour="black",size=0.5) +
  scale_x_discrete(name=c("MONTH NUMBER"),limits=c("EVEN","ODD")) +
  scale_y_continuous(labels = scales::percent, breaks =
c(0,0.2,0.4,0.6,0.8,1)) +
  scale_fill_brewer(palette="Set1",name="PREFERENCE",
labels=c("EVEN","ODD")) +
```

```
  geom_text(aes(label=..count..),stat="count",
            colour="white",size=20,position =
position_fill(vjust=0.5)) +
  theme(
    title = element_text(size = 30),
    legend.title = element_text(size = 25),
    axis.title.x = element_text(size = 30),
    axis.title.y = element_text(size = 30),
    axis.text.x = element_text(size = 30),
    axis.text.y = element_text(size = 30),
    legend.text = element_text(size=25),
    legend.background =
      element_rect(fill="white",colour="black")
  )

chisq.test(expdat_male$mcategory, expdat_male$preference, correct=F)
#X-squared = 0.34802, df = 1, p-value = 0.5552

# 男性の「偶数ー奇数の好み」に「誕生月の偶数・奇数」の影響はない。


#-------------------------------------
# 誕生日（日にち）と好みの関係（男性）
#-------------------------------------

# 同様に、、
table(expdat_male$dcategory,expdat_male$preference)
#     EVEN ODD
#EVEN  113  97
#ODD   117 100


# 棒グラフで可視化
ggplot(expdat_male,aes(x=dcategory,fill=preference)) +
  geom_bar(position =
position_fill(),alpha=0.5,colour="black",size=0.5) +
  scale_x_discrete(name=c("DAY NUMBER"),limits=c("EVEN","ODD")) +
  scale_y_continuous(labels = scales::percent, breaks =
c(0,0.2,0.4,0.6,0.8,1)) +
  scale_fill_brewer(palette="Set1",name="PREFERENCE",
labels=c("EVEN","ODD")) +
  geom_text(aes(label=..count..),stat="count",
            colour="white",size=20,position =
position_fill(vjust=0.5)) +
  theme(
    title = element_text(size = 30),
    legend.title = element_text(size = 25),
    axis.title.x = element_text(size = 30),
    axis.title.y = element_text(size = 30),
    axis.text.x = element_text(size = 30),
    axis.text.y = element_text(size = 30),
    legend.text = element_text(size=25),
    legend.background =
```

```
      element_rect(fill="white",colour="black")
  )


chisq.test(expdat_male$dcategory, expdat_male$preference, correct=F)
#X-squared = 0.00049653, df = 1, p-value = 0.9822

# 男性の「偶数ー奇数の好み」に「誕生日の偶数・奇数」の影響はない。


#-----------------------------------------------
# 誕生日の2つの数字の交互作用と好みの関係（男性）
#-----------------------------------------------

# interaction関数を用いると, 2つの属性間の全ての組み合わせを要素化することができる
interaction(expdat$mcategory,expdat$dcategory)
#[1] EVEN.ODD  ODD.ODD   ODD.ODD   EVEN.EVEN EVEN.EVEN ODD.ODD
ODD.EVEN  EVEN.ODD
#[9] ODD.EVEN  EVEN.ODD  EVEN.ODD  ODD.EVEN  EVEN.EVEN ODD.ODD
EVEN.ODD  ODD.ODD

table(interaction(expdat_male$mcategory,expdat_male$dcategory),
      expdat_male$preference)

#          EVEN ODD
#EVEN.EVEN   67  45
#ODD.EVEN    46  52
#EVEN.ODD    54  53
#ODD.ODD     63  47


# 棒グラフで可視化
ggplot(expdat_male,aes(x=interaction(mcategory,dcategory),fill=prefe
rence)) +
  geom_bar(position =
position_fill(),alpha=0.5,colour="black",size=0.5) +
  scale_x_discrete(name=c("MONTH AND DAY NUMBERS"),

limits=c("EVEN.EVEN","ODD.EVEN","EVEN.ODD","ODD.ODD")) +
  scale_y_continuous(labels = scales::percent, breaks =
c(0,0.2,0.4,0.6,0.8,1)) +
  scale_fill_brewer(palette="Set1",name="PREFERENCE",
labels=c("EVEN","ODD")) +
  geom_text(aes(label=..count..),stat="count",
            colour="white",size=20,position =
position_fill(vjust=0.5)) +
  theme(
    title = element_text(size = 30),
    legend.title = element_text(size = 25),
    axis.title.x = element_text(size = 30),
    axis.title.y = element_text(size = 30),
    axis.text.x = element_text(size = 30),
```

```
    axis.text.y = element_text(size = 30),
    legend.text = element_text(size=25),
    legend.background =
      element_rect(fill="white",colour="black")
  )

# 自由度は3になります。
chisq.test(interaction(expdat_male$mcategory,expdat_male$dcategory),
           expdat_male$preference, correct=F)
#X-squared = 4.5019, df = 3, p-value = 0.2121

# 男性の場合、
# 誕生日の月日の交互特性は、奇数と偶数の好みに有意な影響を与えない


############################
# 女性のケース
############################

# 以下では、男性の行のみを集めた以下のデータフレームを使用する
expdat_female = expdat[expdat$gender=="FEMALE",]

#-------------------------
# 誕生月と好みの関係（女性）
#-------------------------

table(expdat_female$mcategory, expdat_female$preference)
#     EVEN ODD
#EVEN  189  79
#ODD   181 124

# 棒グラフで可視化
ggplot(expdat_female,aes(x=mcategory,fill=preference)) +
  geom_bar(position =
position_fill(),alpha=0.5,colour="black",size=0.5) +
  scale_x_discrete(name=c("MONTH NUMBER"),limits=c("EVEN","ODD")) +
  scale_y_continuous(labels = scales::percent, breaks =
c(0,0.2,0.4,0.6,0.8,1)) +
  scale_fill_brewer(palette="Set1",name="PREFERENCE",
labels=c("EVEN","ODD")) +
  geom_text(aes(label=..count..),stat="count",
            colour="white",size=20,position =
position_fill(vjust=0.5)) +
  theme(
    title = element_text(size = 30),
    legend.title = element_text(size = 25),
    axis.title.x = element_text(size = 30),
    axis.title.y = element_text(size = 30),
    axis.text.x = element_text(size = 30),
    axis.text.y = element_text(size = 30),
    legend.text = element_text(size=25),
    legend.background =
```

```
      element_rect(fill="white",colour="black")
  )

chisq.test(expdat_female$mcategory, expdat_female$preference,
correct=F)
#-squared = 7.7917, df = 1, p-value = 0.005249

# p<0.01（誕生月の偶数・奇数が、数字の好みに有意に影響する）

#-----------------------------------
# 誕生日（日にち）と好みの関係（女性）
#-----------------------------------

table(expdat_female$dcategory, expdat_female$preference)
#     EVEN ODD
#EVEN  193  86
#ODD   177 117

# 棒グラフで可視化
ggplot(expdat_female,aes(x=dcategory,fill=preference)) +
  geom_bar(position =
position_fill(),alpha=0.5,colour="black",size=0.5) +
  scale_x_discrete(name=c("MONTH NUMBER"),limits=c("EVEN","ODD")) +
  scale_y_continuous(labels = scales::percent, breaks =
c(0,0.2,0.4,0.6,0.8,1)) +
  scale_fill_brewer(palette="Set1",name="PREFERENCE",
labels=c("EVEN","ODD")) +
  geom_text(aes(label=..count..),stat="count",
            colour="white",size=20,position =
position_fill(vjust=0.5)) +
  theme(
    title = element_text(size = 30),
    legend.title = element_text(size = 25),
    axis.title.x = element_text(size = 30),
    axis.title.y = element_text(size = 30),
    axis.text.x = element_text(size = 30),
    axis.text.y = element_text(size = 30),
    legend.text = element_text(size=25),
    legend.background =
      element_rect(fill="white",colour="black")
  )

chisq.test(expdat_female$dcategory, expdat_female$preference,
correct=F)
#X-squared = 5.0367, df = 1, p-value = 0.02482

# p<0.05（誕生日の日にちの偶数・奇数が、数字の好みに有意に影響する）



#-------------------------------------------
# 誕生日の2つの数字の交互作用と好みの関係（女性）
#-------------------------------------------
```

```
table(interaction(expdat_female$mcategory,expdat_female$dcategory),
      expdat_female$preference)

#          EVEN ODD
#EVEN.EVEN   95  36
#ODD.EVEN    98  50
#EVEN.ODD    94  43
#ODD.ODD     83  74


# 棒グラフで可視化
ggplot(expdat_female,aes(x=interaction(mcategory,dcategory),fill=pre
ference)) +
  geom_bar(position =
position_fill(),alpha=0.5,colour="black",size=0.5) +
  scale_x_discrete(name=c("MONTH AND DAY NUMBERS"),

limits=c("EVEN.EVEN","ODD.EVEN","EVEN.ODD","ODD.ODD")) +
  scale_y_continuous(labels = scales::percent, breaks =
c(0,0.2,0.4,0.6,0.8,1)) +
  scale_fill_brewer(palette="Set1",name="PREFERENCE",
labels=c("EVEN","ODD")) +
  geom_text(aes(label=..count..),stat="count",
            colour="white",size=20,position =
position_fill(vjust=0.5)) +
  theme(
    title = element_text(size = 30),
    legend.title = element_text(size = 25),
    axis.title.x = element_text(size = 30),
    axis.title.y = element_text(size = 30),
    axis.text.x = element_text(size = 30),
    axis.text.y = element_text(size = 30),
    legend.text = element_text(size=25),
    legend.background =
      element_rect(fill="white",colour="black")
  )

# 自由度は3になります。
chisq.test(interaction(expdat_female$mcategory,expdat_female$dcatego
ry),
           expdat_female$preference, correct=F)
#X-squared = 14.173, df = 3, p-value = 0.002678

# p<0.01より、女性の場合、
# 誕生日の月日の交互特性は、奇数と偶数の好みに有意な影響する
```