

```
#####  
#####  
##### グラフ2 (実践1) #####  
##### geom_bar / geom_col / geom_line #####  
#####  
#####
```

```
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=2865) のデータをインポートしてください。  
url = "https://lab.kenrikodaka.com/_download/csv/oddeven_2865.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 (2865人分) です。  
#month (誕生月) ・day (誕生日) ・preference (奇数が好き: ODD, 偶数が好き: EVEN)  
#gender (男性: MALE, 女性: FEMALE) ・$domhand (利き手が左: LEFT, 利き手が右:  
RIGHT)  
#age (年齢)
```

```
#最初の6行をちょっと出し  
source[1:6,]  
# month day preference gender age domhand  
#1 10 26 ODD MALE 19 RIGHT  
#2 4 6 ODD MALE NA <NA>  
#3 9 14 EVEN FEMALE 20 RIGHT  
#4 1 21 ODD FEMALE 17 RIGHT  
#5 4 7 EVEN FEMALE 20 RIGHT  
#6 7 27 EVEN FEMALE 18 RIGHT
```

```
##NAは未定です。
```

```
# 翻訳すると以下の様になります。  
# 10月26日、奇数好き、男性、19歳、右利き  
# 04月06日、奇数好き、男性、不明、不明  
# 09月14日、偶数好き、女性、18歳、右利き  
# 01月21日、奇数好き、女性、18歳、右利き  
# 04月07日、偶数好き、女性、18歳、右利き  
# 07月27日、偶数好き、女性、19歳、右利き
```

```
#-----  
# [準備2] mutateの使い方  
#-----
```

```
#mutateを使うと  
#データフレームの属性を簡単に追加/更新できます。
```

```
#mutateを使うには以下のライブラリをインクルードします。  
library(dplyr)
```

```

# mutateの構文は以下の通り
#-----
## 新しいデータフレーム =
## mutate(古いデータフレーム, 新しい属性 = 既存の属性同士の計算)
#-----

# sourceをdatにコピーします。
dat = source
dat[1:3,]
#   month day preference gender age domhand
#1     10  26         ODD  MALE  19   RIGHT
#2      4   6         ODD  MALE  NA    <NA>
#3      9  14         EVEN FEMALE 20   RIGHT

# datに月日の奇数/偶数のラベル(mtype)を新たに追加します。
dat2 = mutate(dat, mtype=month%%2)
dat2[1:4,]
#   month day preference gender age domhand mtype
#1     10  26         ODD  MALE  19   RIGHT     0
#2      4   6         ODD  MALE  NA    <NA>     0
#3      9  14         EVEN FEMALE 20   RIGHT     1
#4      1  21         ODD FEMALE 17   RIGHT     1

# if_else(条件, 条件が真のとき, それ以外)という構文となります。
dat2 = mutate(dat2, mtype = if_else(mtype==0,"EVEN","ODD"))
dat2[1:4,]
#   month day preference gender age domhand mtype
#1     10  26         ODD  MALE  19   RIGHT  EVEN
#2      4   6         ODD  MALE  NA    <NA>  EVEN
#3      9  14         EVEN FEMALE 20   RIGHT  ODD
#4      1  21         ODD FEMALE 17   RIGHT  ODD
#5      4   7         EVEN FEMALE 20   RIGHT  EVEN

# 同様にdtypeも追加
dat2 = mutate(dat2, dtype = if_else(day%%2==0,"EVEN","ODD"))
dat2[1:4,]
#   month day preference gender age domhand mtype dtype
# 1     10  26         ODD  MALE  19   RIGHT  EVEN  EVEN
# 2      4   6         ODD  MALE  NA    <NA>  EVEN  EVEN
# 3      9  14         EVEN FEMALE 20   RIGHT  ODD   EVEN
# 4      1  21         ODD FEMALE 17   RIGHT  ODD   ODD

# 3つ以上の分岐はcase_whenを使います。
dat2 = mutate(dat2, combo =
  case_when(mtype=="EVEN"&dtype=="EVEN"~"EE",
            mtype=="ODD"&dtype=="ODD"~"OO",
            mtype=="EVEN"&dtype=="ODD"~"EO",
            TRUE~"OE"));
dat2[1:4,]
#   month day preference gender age domhand mtype dtype combo
#1     10  26         ODD  MALE  19   RIGHT  EVEN  EVEN  EE
#2      4   6         ODD  MALE  NA    <NA>  EVEN  EVEN  EE
#3      9  14         EVEN FEMALE 20   RIGHT  ODD   EVEN  OE
#4      1  21         ODD FEMALE 17   RIGHT  ODD   ODD   OO

# グラフを作成します
## 4つのcomboごとに、偶数好きと奇数好きの人数を数え上げます。
g = ggplot(dat2, aes(x=combo, fill=preference))
g = g + geom_bar(position = position_dodge()) #fillを横に並べる

```

```

g = g + labs(title = "Graph0_Mutate") #グラフのタイトル
g0 = g
g0 #out Graph0_Mutate

#-----
# [Graph1_Gender]
# 男女別の奇数偶数好き
#-----

# sourceをそのままdatに移します。
dat = source

# [ggplot]
## x軸をgenderに、枠線と塗りつぶしをpreferenceに設定
g = ggplot(dat,aes(x=gender,color=preference, fill=preference))

# [geom_bar] genderごとにpreferenceを数え上げ
## position_dodge: 横並び, width: グラフの幅, alpha: 透明度
g = g + geom_bar(position=position_dodge(width=0.9),alpha=0.9)

# X軸は離散値、Y軸は連続値
g = g + scale_x_discrete(limits = c("FEMALE", "MALE"))
g = g + scale_y_continuous(limits = c(-100,
1100),breaks=c(0,250,500,750,1000))

# 塗りつぶしのパレットをnejmに変更
library(ggsci)
g = g + scale_fill_nejm()

# 90度回転
g = g + coord_flip()

# [geom_text]
## geom_barの数え上げの量を表示
## just=0 (左揃え), 0.5 (中央揃え), 1 (右揃え)
g = g + geom_text(stat = "count",
                  aes(label = paste("(",..count..,")",sep=""),
                      y=..count..*0.95, hjust=1),
                  position = position_dodge(width=0.9),
                  color="white", size=10)

# [labs]
## グラフのタイトル、XY軸のタイトル
g = g + labs(title = "Graph1_Gender", x = "Gender", y = "Population")

# [theme]
g = g + theme(
  # 縦横比を3:4
  aspect.ratio=3/4,
  # ベースを太字、サイズ36に
  text = element_text(face = "bold", size = 36),
  # 軸の目盛りのサイズ
  axis.text.x = element_text(size = 30),
  axis.text.y = element_text(size = 25),
  # 凡例関係の設定
  legend.text = element_text(size=20),
  legend.title = element_text(size=25),
  legend.position=c(0.95,0.95),

```

```

    legend.justification=c(1,1),
    legend.background =
      element_rect(fill = "white", colour = "black"))

g1 = g
g1 #out Graph1_Gender

#-----
# [Graph2_FemaleMonth]
# 女性の月の数字の影響
#-----

# 女性だけを集めたデータフレームをdat_とする
dat_ = source[source$gender=="FEMALE",]

# 女性の数を確認
nrow(dat_)
# [1] 1603

# 奇素数（2を除く素数）を判定する関数の作成
#-----
oddprimeSingle = function(n){
  result = FALSE
  if(n==3|n==5|n==7|n==11|n==13|n==17|n==19|n==23|n==29|n==31){
    result = T
  }
  result
}

oddprime = function(v){
  result = vector("logical", length(v))
  for(i in 1:length(v)){
    result[i] = oddprimeSingle(v[i])
  }
  result
}
#-----

# 月が偶数、奇数、素数の好みをベクトルで取り出す
pref_even = dat_$preference[dat_$month%%2==0]
pref_odd = dat_$preference[dat_$month%%2==1]
pref_prime = dat_$preference[oddprime(dat_$month)]

# それぞれの偶数好きの割合の重み（50%を0とする）
r1 = 100*(sum(pref_even=="EVEN")/length(pref_even)-0.5)
r2 = 100*(sum(pref_odd=="EVEN")/length(pref_odd)-0.5)
r3 = 100*(sum(pref_prime=="EVEN")/length(pref_prime)-0.5)

# 偶数好きの重みをデータフレームとする
x = data.frame(
  ntype = c("EVEN", "ODD", "PRIME"),
  weight = c(r1, r2, r3)
)

# [ggplot]
g = ggplot(x, aes(x = ntype, y = weight, fill = ntype, colour = ntype))

# [geom_bar]

```

```

## 個数 (stat="count") でなく数値 (stat="identity) を描画
g = g + geom_bar(stat = "identity",alpha=4/5)

# yは連続値
g = g + scale_y_continuous(limits=c(-10,30))

# 縦線 (hline) と横線 (vline) の描画
g = g + geom_hline(yintercept = 0, linetype = "solid",
size=1,colour="black")
g = g + geom_vline(xintercept = 1.5, linetype = "dotted",
size=1,colour="black")
g = g + geom_vline(xintercept = 2.5, linetype = "dotted",
size=1,colour="black")

# 色の設定
col1 = rgb(174/255,69/255,50/255) #for even
col2 = rgb(58/255,128/255,77/255) #for odd
col3 = rgb(48/255,112/255,176/255) #for prime

g = g + scale_fill_manual(values = c(col1,col2,col3), name="number type")
g = g + scale_colour_manual(values = c(col1,col2,col3), name = "number
type")

# 90度回転
g = g + coord_flip()

# [labs]
## グラフのタイトル、XY軸のタイトル
g = g + labs(title = "Graph2_FemaleMonth",
x = "Number type", y = "Even number preference (%)")

# [theme]
g = g + theme(
# 縦横比を3:4
aspect.ratio=3/4,
# ベースを太字、サイズ36に
text = element_text(face = "bold", size = 36),
# 軸の目盛りのサイズ
axis.text.x = element_text(size = 30),
axis.text.y = element_text(size = 25),
# 凡例関係の設定
legend.text = element_text(size=20),
legend.title = element_text(size=25),
legend.position=c(0.95,0.95),
legend.justification=c(1,1),
legend.background =
element_rect(fill = "white", colour = "black"))

g2 = g
g2 #out Graph2_FemaleMonth

#-----
# [Graph3_FemaleMonthDay]
# 女性の月x日の数字の影響
#-----

# 月と日にちが偶数、奇数、素数の好みをベクトルで取り出す
pref_even = dat_$preference[dat_$month%%2==0 & dat_$day%%2==0]

```

```

pref_odd = dat_$preference[dat_$month%%2==1 & dat_$day%%2==1]
pref_prime = dat_$preference[oddprime(dat_$month) & oddprime(dat_$day)]

#以下はGraph2とほぼ同様

# それぞれの偶数好きの割合の重み (50%を0とする)
r1 = 100*(sum(pref_even=="EVEN")/length(pref_even)-0.5)
r2 = 100*(sum(pref_odd=="EVEN")/length(pref_odd)-0.5)
r3 = 100*(sum(pref_prime=="EVEN")/length(pref_prime)-0.5)

x = data.frame(
  ntype = c("EVEN", "ODD", "PRIME"),
  weight = c(r1, r2, r3)
)

g = ggplot(x, aes(x = ntype, y = weight, fill = ntype, colour = ntype))
g = g + geom_bar(stat = "identity", alpha=4/5)
g = g + scale_y_continuous(limits=c(-10,30))

# x軸 (離散値) のラベルを変更
g = g + scale_x_discrete(limits=c("EVEN","ODD","PRIME"),
                        label=c("EVENxEVEN","ODDxODD","PRIMExPRIME"))

g = g + geom_hline(yintercept = 0, linetype = "solid",
                  size=1, colour="black")
g = g + geom_vline(xintercept = 1.5, linetype = "dotted",
                  size=1, colour="black")
g = g + geom_vline(xintercept = 2.5, linetype = "dotted",
                  size=1, colour="black")
g = g + scale_fill_manual(values = c(col1,col2,col3), name="number type")
g = g + scale_colour_manual(values = c(col1,col2,col3), name = "number
type")
g = g + coord_flip()
g = g + labs(title = "Graph3_FemaleMonthDay",
            x = "Number type", y = "Even number preference (%)")
g = g + theme(
  aspect.ratio=4/4.5,
  text = element_text(face = "bold", size = 36),
  axis.text.x = element_text(size = 30),
  axis.text.y = element_text(size = 25),
  legend.text = element_text(size=20),
  legend.title = element_text(size=25),
  legend.position=c(0.95,0.95),
  legend.justification=c(1,1),
  legend.background =
    element_rect(fill = "white", colour = "black"))

g3 = g
g3 #Graph3_FemaleMonthDay

#-----
# [Graph4_DayRank]
# 誕生日の日にちランキング (男女分けない)
#-----

dat_ = source;

#-----
# グラフ用のデータフレームの作成
#-----

```

```

#day毎のpreferenceの数を集計
daypref = table(dat_$day, dat_$preference)

#      EVEN ODD
# 1      62  54
# 2      64  27
# 3      55  33

# tableをデータフレームに変換する
df = as.data.frame(daypref)
#   Var1 Var2 Freq
#1     1  EVEN   62
#2     2  EVEN   64
#3     3  EVEN   55

df.e= df[df$Var2=="EVEN",]
df.o= df[df$Var2=="ODD",]

# 列として接続
df = cbind(df.e,df.o)
#   Var1 Var2 Freq Var1 Var2 Freq
#1     1  EVEN   62     1  ODD   54
#2     2  EVEN   64     2  ODD   27
#3     3  EVEN   55     3  ODD   33

# 1,3,6列目だけを取り出す
df = df[,c(1,3,6)]
colnames(df) = c("number","even","odd")
#   number even odd
#1         1   62  54
#2         2   64  27
#3         3   55  33

# 新しい属性erate (偶数を好む確率) をdfに追加
df = mutate(df,erate = even/(even+odd))
#   number even odd   erate
#1         1   62  54 0.5344828
#2         2   64  27 0.7032967
#3         3   55  33 0.6250000

# number属性をfactorからintegerに変更します。
class(df$number)
#[1] "factor"

#df$number = as.integer(df$number)
#同じ結果だがこちらが正しい (映像収録後に編集)
df$number = as.integer(as.character(df$number))
class(df$number)
#[1] "integer"

#mutateを使った分岐
df = mutate(df, type = case_when(number%%2==0~"EVEN",
                                oddprime(number)~"PRIME",
                                TRUE~"ODD"))
#if_elseを使った場合 (上と同じ)
df = mutate(df, type = if_else(number%%2==0,"EVEN",
                                if_else(oddprime(number),"PRIME","ODD")))

```

```

#   number even odd   erate  type
#1     1   62  54 0.5344828  ODD
#2     2   64  27 0.7032967  EVEN
#3     3   55  33 0.6250000  PRIME
#4     4   51  33 0.6071429  EVEN
#5     5   43  43 0.5000000  PRIME
#...

# erateの大きい順に並べる
df = arrange(df,desc(erate)); df
#   number even odd   erate  type
#1     22   57  23 0.7125000  EVEN
#2     2   64  27 0.7032967  EVEN
#3     21   72  39 0.6486486  ODD

# 順位の属性を加える ()
df = cbind(df, data.frame(order=1:31)); df
#   number even odd   erate  type order
#1     22   57  23 0.7125000  EVEN     1
#2     2   64  27 0.7032967  EVEN     2
#3     21   72  39 0.6486486  ODD     3

dat_order = df

#-----
# グラフの作成
#-----

g = ggplot(dat_order,aes(x=order,y=100*erate,fill=type)); g
g = g + geom_bar(stat = "identity", colour="black"); g

#geom_bar(stat="identity")とgeom_colは同じなので、こちらでも同じ結果です。
#g = g + geom_col(colour = "black")

g = g + scale_x_continuous(limits=c(0.5,31.5),breaks=c(1,10,20,30)); g
g = g + scale_y_continuous(limits=c(0,87),breaks=c(0,25,50,75)); g

# 50%のところに線を引く
g = g + geom_hline(yintercept = 50, linetype = "dotted",
size=1,colour="black");
# 各順位に対応する数字をバーの上端より1.5文字分下に描画する
g = g + geom_text(aes(label = number), colour = "white", size = 5, vjust
= 1.5)

# 凡例の色をマニュアルで指定する
g = g + scale_fill_manual(values = c(col1,col2,col3),name="Number type")
g = g + scale_colour_manual(values = c(col1,col2,col3),name="Number type")

g = g + labs(title="Graph4_DayRank",x="ORDER",y="EVEN-PREF RATE")
g = g + theme(
  aspect.ratio=1/4,
  text = element_text(face = "bold", size = 36),
  axis.text.x = element_text(size = 30),
  axis.text.y = element_text(size = 25),
  legend.text = element_text(size=20),
  legend.title = element_text(size=25),
  legend.position=c(1.0,1.03),
  legend.justification=c(1,0),
  legend.background =
  element_rect(fill = "white", colour = "black"))

```



```
g4 = g
g4 #Graph4_FemaleMonthDay
```

```
#! 新しく折れ線グラフ (geom_line) を追加しました。
#! 映像による解説はありません。
```

```
#-----
# [Graph5_AgeEffect]
# 年齢の影響 (男女分けない)
#-----
```

```
dat_ = source;
```

```
# 年齢ごとのpreferenceを集計する
```

```
agepref = table(dat$age,dat$preference); agepref
```

```
#   EVEN ODD
#12    1   1
#13    1   2
#14    1   0
#15   27   6
#16   37  20
#17   30   6
#18  597 367
#19  537 366
#20  141 115
#21   30  33
#22    7  11
#...
```

```
df = as.data.frame(agepref)
df = cbind(df.e= df[df$Var2=="EVEN",],df[df$Var2=="ODD",])
df = df[,c(1,3,6)]
colnames(df) = c("age","even","odd")
# ageがファクタのため、文字列に変換した後、数値に変換する
# 直接数値に変換すると、ファクタのid (1,2,...) が付されてしまう
```

```
df$age = as.numeric(as.character(df$age))
df[df$age>14 & df$age<23,]
#   age even odd
#4   15  27   6
#5   16  37  20
#6   17  30   6
#7   18 597 367
#8   19 537 366
#9   20 141 115
#10  21  30  33
#11  22   7  11
```

```
#mutateを使って、偶数好きの確率erateを属性に追加
```

```
df2 = mutate(df,erate = even/(even+odd))
df2[df2$age>14 & df2$age<23,]
#   age even odd   erate
#4   15  27   6 0.8181818
#5   16  37  20 0.6491228
#6   17  30   6 0.8333333
#7   18 597 367 0.6192946
#8   19 537 366 0.5946844
```

```
#9  20  141 115 0.5507812
#10 21   30  33 0.4761905
#11 22    7  11 0.3888889
```

```
dat_age = df2
```

```
# 15歳から22歳までの偶数好きの割合の遷移を
# 折れ線グラフで視覚化します。
```

```
g = ggplot(dat_age, aes(x=age, y=erate))
g = g + geom_line(colour = "black", size=1.5)
g = g + geom_point(size=10)
g = g + scale_x_continuous(limits=c(14.5,22.5), breaks=15:22); g
g = g +
scale_y_continuous(limits=c(0,1.0), breaks=c(0,0.25,0.50,0.75,1.0)); g
g = g + labs(title="Graph5_AgeEffect", x="AGE", y="EVEN-PREF RATE")
g = g + theme(
  aspect.ratio=3/4,
  text = element_text(face = "bold", size = 36),
  axis.text.x = element_text(size = 30),
  axis.text.y = element_text(size = 25)
)
```

```
g5 = g
g5 #Graph5_AgeEffect
```

```
#-----
# [確認課題]
# ApMedia04_Work
# 男女別の月の偶数好きの数字ランキングのグラフを作成してください。
# 締切は11月中とします。
# ファイル名は「2250xx_work4.R」としてください。
#-----
```

```
# グラフを結合する場合、以下を参照してください。
```

```
library(gridExtra)
```

```
df.e= df[df$Var2=="EVEN",]
df.o= df[df$Var2=="ODD",]
```

```
# 列として接続
```

```
df = cbind(df.e,df.o)
#   Var1 Var2 Freq Var1 Var2 Freq
#1     1  EVEN   62     1  ODD   54
#2     2  EVEN   64     2  ODD   27
#3     3  EVEN   55     3  ODD   33
```

```
## グラフの結合の方法
```

```
grid.arrange(g2,g3,nrow=1) #1行で並べる
grid.arrange(g2,g3,ncol=1) #1列で並べる
```

```
## 3つを配置する例
```

```
g23 = grid.arrange(g2,g3,nrow=1)
grid.arrange(g4,g23,ncol=1)
```