Analysis Project in Japanese

Naoko Ishibashi

2025-01-31

パッケージの読み込み

```
# パッケージ
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(reshape2)
library(ragg) # 日本語フォント
library(tinytex)
library(rmarkdown)
library(bitops)
library(caTools)
library(rprojroot)
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ forcats   1.0.0     ✔ stringr   1.5.1
## ✔ lubridate 1.9.3     ✔ tibble    3.2.1
## ✔ purrr     1.0.2     ✔ tidyr     1.3.1
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(showtext)
```

```
## Loading required package: sysfonts
## Loading required package: showtextdb
```

```
font_add_google("Noto Sans JP", "jp_font")
showtext_auto()


tinytex::reinstall_tinytex(repository = "illinois")
```

```
## If reinstallation fails, try install_tinytex() again. Then install the following packages:
##
## tinytex::tlmgr_install(c("amscls", "amsfonts", "amsmath", "atbegshi", "atveryend", "auxhook", "babel", "bibte
x", "bigintcalc", "bitset", "bookmark", "booktabs", "cm", "ctablestack", "dehyph", "dvipdfmx", "dvips", "ec", "ep
stopdf", "epstopdf-pkg", "etex", "etexcmds", "etoolbox", "euenc", "everyshi", "extractbb", "fancyvrb", "filehoo
k", "firstaid", "float", "fontspec", "framed", "geometry", "gettitlestring", "glyphlist", "graphics", "graphics-c
fg", "graphics-def", "helvetic", "hycolor", "hyperref", "hyph-utf8", "hyphen-base", "iftex", "inconsolata", "infw
arerr", "intcalc", "knuth-lib", "kpathsea", "kvdefinekeys", "kvoptions", "kvsetkeys", "l3backend", "l3kernel", "l
3packages", "latex", "latex-amsmath-dev", "latex-bin", "latex-fonts", "latex-tools-dev", "latexconfig", "latexm
k", "letltxmacro", "lm", "lm-math", "ltxcmds", "lua-alt-getopt", "lua-uni-algos", "luahbtex", "lualatex-math", "l
ualibs", "luaotfload", "luatex", "luatexbase", "mdwtools", "metafont", "mfware", "modes", "natbib", "pdfescape",
"pdftex", "pdftexcmds", "plain", "psnfss", "refcount", "rerunfilecheck", "scheme-infraonly", "selnolig", "stringe
nc", "symbol", "tex", "tex-ini-files", "texlive-scripts", "texlive-scripts-extra", "texlive.infra", "times", "tip
a", "tlgpg", "tools", "unicode-data", "unicode-math", "uniquecounter", "url", "xcolor", "xetex", "xetexconfig",
"xkeyval", "xunicode", "zapfding"))
##
## The directory /Users/naoko/Library/TinyTeX/texmf-local is not empty. It will be backed up to /var/folders/7q/3
1fn2m9d1p9g4dwd3m651x9r0000gn/T//RtmpoSAHfF/file2a1b70e2acef and restored later.
##
## tlmgr install everyshi tlgpg
## tlmgr update --self
## tlmgr install everyshi tlgpg
## tlmgr --repository http://www.preining.info/tlgpg/ install tlgpg
## tlmgr option repository 'https://ctan.math.illinois.edu/systems/texlive/tlnet'
## tlmgr update --list
```

```
tinytex::tlmgr_install("bookmark")
```

```
## tlmgr install bookmark
```

Data cleaning and checking

```
# TSVファイルをデータフレームにインポート
HPAP001 <- read_tsv("/Users/naoko/Desktop/All_Data_Analytics_Classes_UPenn/analysis_project/HPAP001.pooled.tsv")
```

```
## Rows: 26463 Columns: 17
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: "\t"
## chr (7): subject, v_gene, j_gene, cdr3_nt, cdr3_aa, germline, top_copy_seq
## dbl (6): clone_id, cdr3_num_nts, uniques, instances, copies, avg_v_identity
## lgl (4): functional, insertions, deletions, parent_id
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
HPAP003 <- read_tsv("/Users/naoko/Desktop/All_Data_Analytics_Classes_UPenn/analysis_project/HPAP003.pooled.tsv")
```

```
## Rows: 17505 Columns: 17
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: "\t"
## chr (7): subject, v_gene, j_gene, cdr3_nt, cdr3_aa, germline, top_copy_seq
## dbl (6): clone_id, cdr3_num_nts, uniques, instances, copies, avg_v_identity
## lgl (4): functional, insertions, deletions, parent_id
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
HPAP004 <- read_tsv("/Users/naoko/Desktop/All_Data_Analytics_Classes_UPenn/analysis_project/HPAP004.pooled.tsv")
```

```
## Rows: 16222 Columns: 17
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: "\t"
## chr (7): subject, v_gene, j_gene, cdr3_nt, cdr3_aa, germline, top_copy_seq
## dbl (6): clone_id, cdr3_num_nts, uniques, instances, copies, avg_v_identity
## lgl (4): functional, insertions, deletions, parent_id
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
HPAP015 <- read_tsv("/Users/naoko/Desktop/All_Data_Analytics_Classes_UPenn/analysis_project/HPAP015.pooled.tsv")
```

```
## Rows: 47581 Columns: 17
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: "\t"
## chr (7): subject, v_gene, j_gene, cdr3_nt, cdr3_aa, germline, top_copy_seq
## dbl (6): clone_id, cdr3_num_nts, uniques, instances, copies, avg_v_identity
## lgl (4): functional, insertions, deletions, parent_id
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
HPAP030 <- read_tsv("/Users/naoko/Desktop/All_Data_Analytics_Classes_UPenn/analysis_project/HPAP030.pooled.tsv")
```

```
## Rows: 45213 Columns: 17
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: "\t"
## chr (7): subject, v_gene, j_gene, cdr3_nt, cdr3_aa, germline, top_copy_seq
## dbl (6): clone_id, cdr3_num_nts, uniques, instances, copies, avg_v_identity
## lgl (4): functional, insertions, deletions, parent_id
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
HPAP031 <- read_tsv("/Users/naoko/Desktop/All_Data_Analytics_Classes_UPenn/analysis_project/HPAP031.pooled.tsv")
```

```
## Rows: 36831 Columns: 17
## ── Column specification ────────────────────────────────────────────────────
## Delimiter: "\t"
## chr (7): subject, v_gene, j_gene, cdr3_nt, cdr3_aa, germline, top_copy_seq
## dbl (6): clone_id, cdr3_num_nts, uniques, instances, copies, avg_v_identity
## lgl (4): functional, insertions, deletions, parent_id
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# 同じ列を持つデータセットをマージ
all_hpap <- rbind(HPAP001, HPAP003, HPAP004, HPAP015, HPAP030, HPAP031)

# 欠損値の確認
colSums(is.na(all_hpap))
```

```
##       clone_id        subject         v_gene         j_gene     functional
##              0              0              0              0              0
##     insertions      deletions        cdr3_nt   cdr3_num_nts        cdr3_aa
##         189815         189815              0              0              0
##        uniques      instances         copies       germline      parent_id
##              0              0              0              0         189815
## avg_v_identity   top_copy_seq
##              0              0
```

```
# データのクリーニング
# 'insertions', 'deletions', 'parent_id'のNAを削除
hpap_cleaned <- subset(all_hpap, select = -c(insertions, deletions, parent_id))

# クリーンアップしたデータの確認
# colSums(is.na(hpap_cleaned))
```

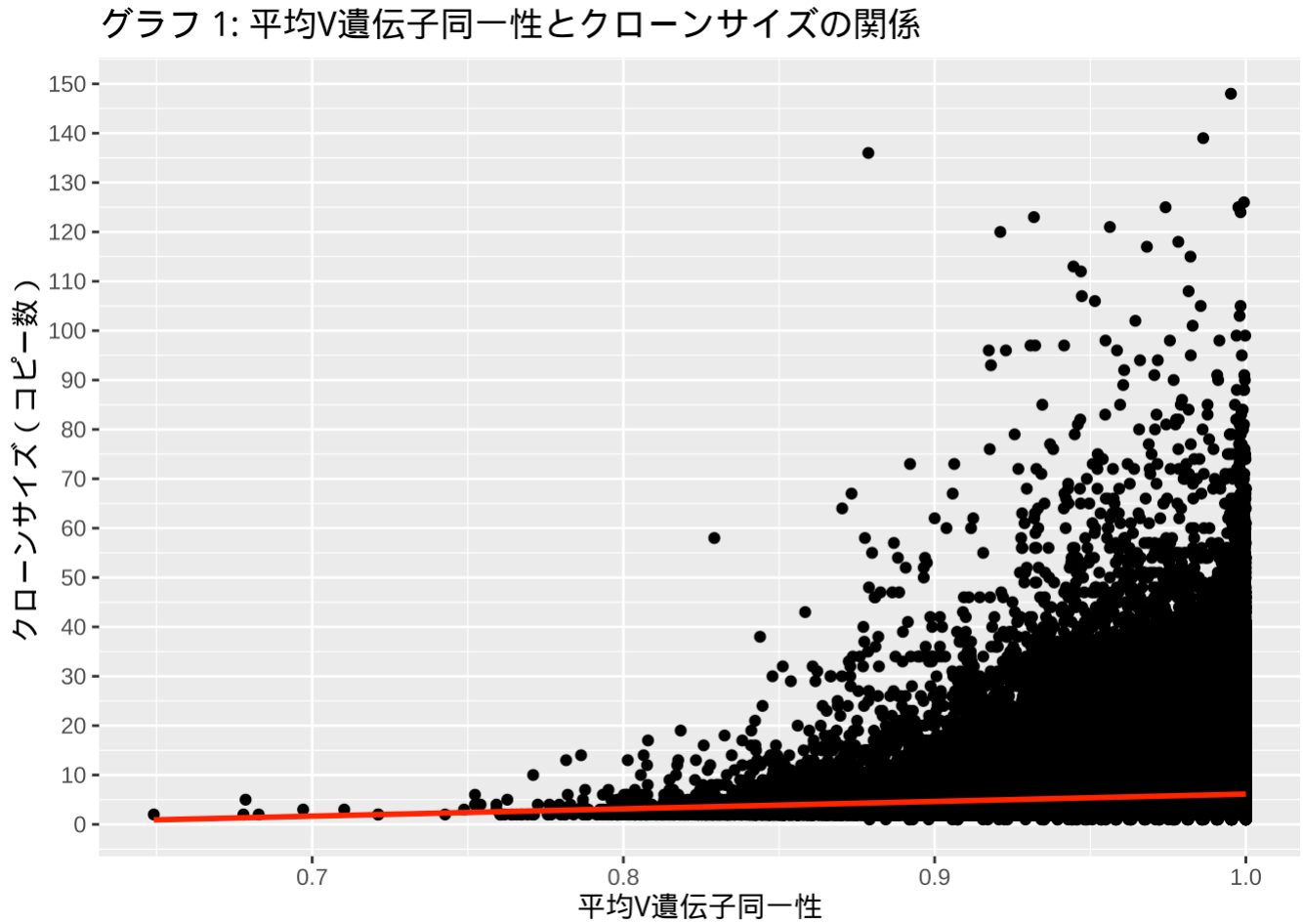
1. クローンサイズと突然変異には相関があるか？ B細胞は抗原刺激があると急速に増殖し、突然変異します。各クローンについて、avg_v_identity と copies という2つのフィールドがあります。前者はクローン内のヌクレオチド塩基のうち、遺伝子本来の配列に一致する割合を示し（例：1 - 突然変異）、後者はそのクローンに関連するリードの数を示し、サイズのおおよその指標となります。これらを踏まえ

て、2つの変数の間に相関があるかを確認するための図を作成してください。その後、これらが関連しているという仮説を統計的に検定し、使用する統計検定の選択とその頑健性についてコメントしてください。

これらの変数の間に相関があるかどうかを確認するための図を作成してください。 次に、2つの変数が関連しているという仮説を統計的に検定し、使用する統計検定の選択とその頑健性についてコメントしてください。

```
# この2つの変数間の相関を調べるためのデータ可視化を作成します。
# dfがデータフレームで、y_varが潜在的な外れ値を持つ場合を想定しています。

# 外れ値を除去
cleaned_data <- hpap_cleaned %>% filter(copies < 150)

# 散布図を作成: 散布図は2つの変数間の関係を効果的に表示できます。
# グラフ 1: 平均V遺伝子同一性 vs. クローンサイズの散布図
ggplot(data = cleaned_data, aes(x = avg_v_identity, y = copies)) +
  geom_point() +
  geom_smooth(method = "lm", color = "red", se = FALSE) +  # 回帰直線を追加
  scale_y_continuous(breaks = seq(0, 150, by = 10)) +  # Y軸の目盛りを追加
  labs(title = "グラフ 1: 平均V遺伝子同一性とクローンサイズの関係",
       x = "平均V遺伝子同一性",
       y = "クローンサイズ（コピー数）")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



分析（グラフ） このグラフは、正の傾向を示しています。x軸上で0.75付近にポイントが集まり始め、多くのポイントは(0.85, 10)と(1, 40)の間に分布しています。傾きは視覚的に分かりやすくするために2に設定されていますが、実際の傾向は穏やかな上向きの傾きです。

```
# クローンサイズと突然変異が関連しているかどうかの仮説を統計的に検定します。
# cor()関数は2つの変数間の相関係数を計算するために使用されます。

# ピアソン
# ピアソン相関を計算: 線形関係を測定します。
correlation_pearson <- cor(hpap_cleaned$avg_v_identity, hpap_cleaned$copies, method = "pearson")
print(correlation_pearson) # 0.0649304
```

```
## [1] 0.0649304
```

```
# スピアマン
# スピアマン相関を計算: 順位に基づく関係を測定します。
correlation_spearman <- cor(hpap_cleaned$avg_v_identity, hpap_cleaned$copies, method = "spearman")
print(correlation_spearman) # -0.02137582
```

```
## [1] -0.02137582
```

```
# ケンドール
# ケンドール相関を計算: 順序関係を測定します。
correlation_kendall <- cor(hpap_cleaned$avg_v_identity, hpap_cleaned$copies, method = "kendall")
print(correlation_kendall) # -0.02137582
```

```
## [1] -0.01497234
```

分析（ピアソン、スピアマン、ケンドール） 私は、この分析にピアソン相関が適切な選択であると考えています。散布図は明確に正の傾向を示しており、ピアソン相関はそのポジティブな関係を反映しています。このため、ピアソン相関が最も信頼できるオプションだと考えています。

```
# 相関検定を実行
correlation_test <- cor.test(hpap_cleaned$copies, hpap_cleaned$avg_v_identity, method = "pearson")
print(correlation_test)
```

```
##
##  Pearson's product-moment correlation
##
## data:  hpap_cleaned$copies and hpap_cleaned$avg_v_identity
## t = 28.348, df = 189813, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.06044939 0.06940878
## sample estimates:
##       cor
## 0.0649304
```

分析（相関検定） t（テスト統計量）= 28.348 2つの変数間の関係の強さ。

df（自由度）= 189813 分析に使用したサンプルのサイズ。

p値 < 2.2e-16 p値が2.2e-16より小さいです。これは従来のアルファレベル0.05よりも大幅に小さく、帰無仮説を棄却できることを示しています。これにより、2つの変数間に相関がある強い証拠が提供されます。

対立仮説： コピー数とavg_v_identityには有意な関係がある。

95%信頼区間：0.06044939 〜 0.06940878 この数値は正の関連性を示しており、1つの変数が増加すると、もう1つも増加する傾向があることを意味します。

サンプル推定値：相関係数 0.0649304 この値は正の関連性を示しています。

```
# 線形回帰
# 関連性の強さ（傾き）を見つけ、この関係の有意性を確認
linear <- summary(lm(avg_v_identity ~ copies, data = hpap_cleaned))

# 表3：線形回帰
print(linear)
```

```
##
## Call:
## lm(formula = avg_v_identity ~ copies, data = hpap_cleaned)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32691 -0.01682  0.01657  0.02370  0.02428
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept) 9.754e-01  9.434e-05 10339.23   <2e-16 ***
## copies      2.873e-04  1.014e-05    28.35   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03211 on 189813 degrees of freedom
## Multiple R-squared:  0.004216,   Adjusted R-squared:  0.004211
## F-statistic: 803.6 on 1 and 189813 DF,  p-value: < 2.2e-16
```

分析（線形回帰） p値が<2.2e-16であることは、コピー数とavg_v_identityとの間に非常に有意な関係があることを示しています。 この極めて低いp値は、帰無仮説を棄却できる強い証拠を提供しており、この関係が偶然によって観察される可能性がほぼゼロであることを示唆しています。 結果は、コピー数とavg_v_identityとの間に統計的に有意な正の相関があることを示しています。 言い換えれば、クローンサイズ（コピー数で測定される）が増加するにつれて、平均V遺伝子同一性（avg_v_identity）も増加する傾向があります。

2. V遺伝子の使用はドナー間で均等かつ一貫しているか？ 発生過程において、各B細胞はV(D)J再構成を行い、V遺伝子、D遺伝子、J遺伝子の1つずつを擬似ランダムに結合して、ヘビーチェーンの一部を作り、その後細胞の抗体の一部をコードします。この質問では、v_gene列によって注釈付けされたV遺伝子に焦点を当てます。

各ドナー内でV遺伝子がクローン間で均等に分布しているように見えますか？この分布をどのように視覚化しますか？V遺伝子の使用をコピー数で重み付けした場合、どうなりますか？

ドナー間でV遺伝子の使用にパターンがあるように見えますか（つまり、ほとんどのドナーが同じV遺伝子を使用しているのか、それとも各ドナーが異なるV遺伝子を使用しているのか）？この分布をどのように視覚化しますか？

```
# 1.
# データの確認
table(hpap_cleaned$subject)
```

```
##
## HPAP001 HPAP003 HPAP004 HPAP015 HPAP030 HPAP031
##   26463   17505   16222   47581   45213   36831
```

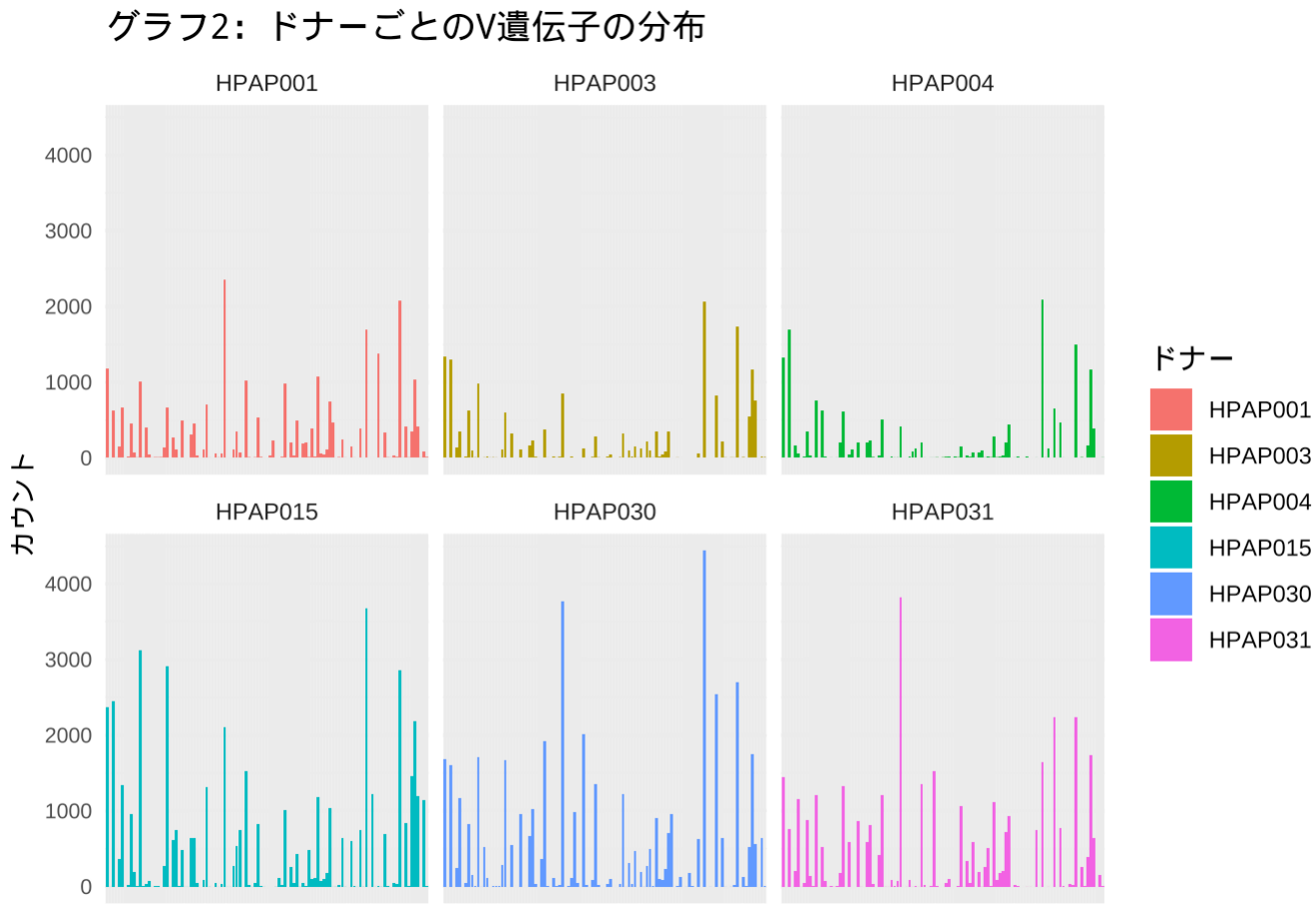
```
str(hpap_cleaned$v_gene)
```

```
##  chr [1:189815] "IGHV1-18" "IGHV1-18" "IGHV1-18" "IGHV1-18" "IGHV1-18" ...
```

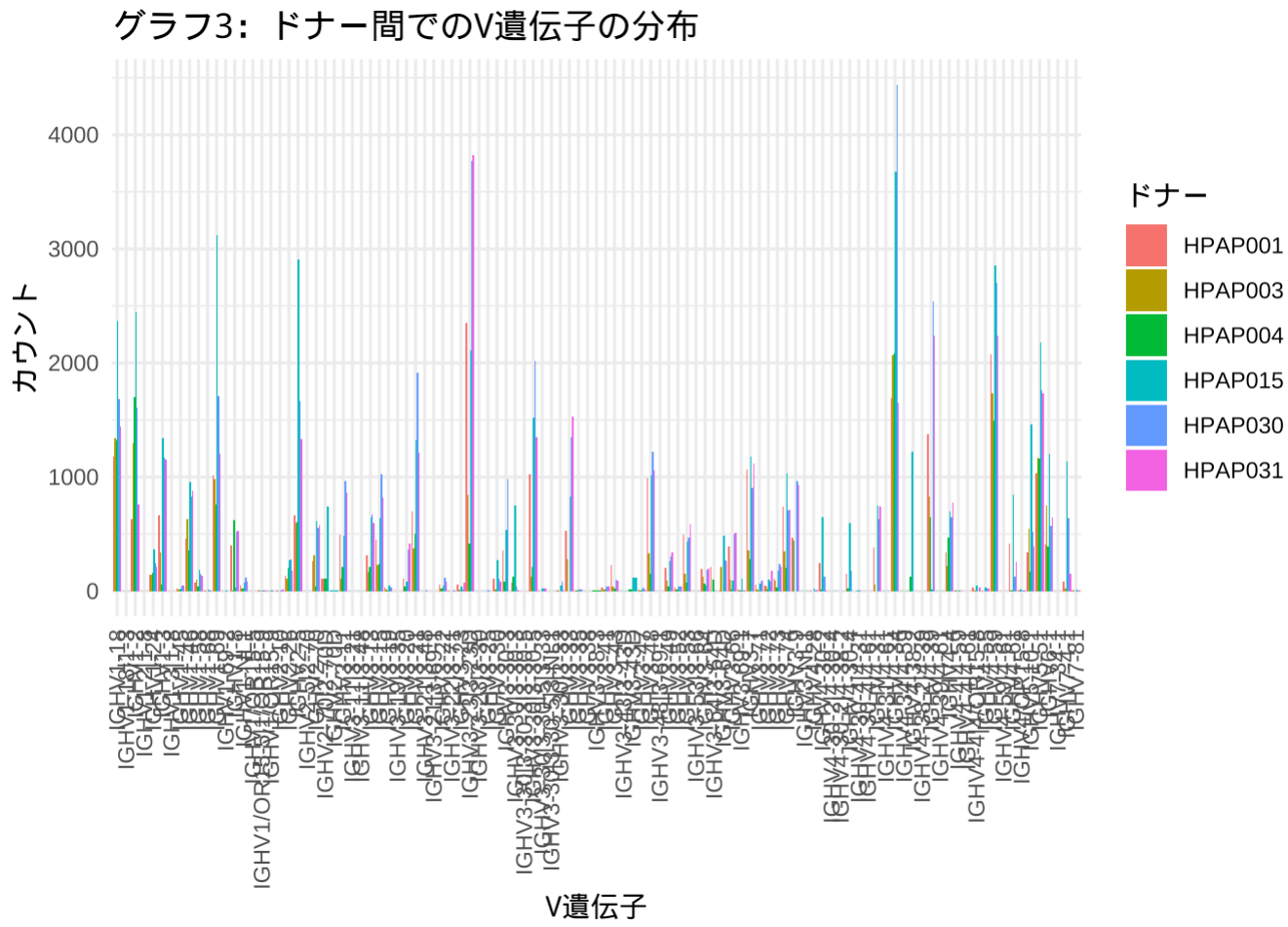
```
# ドナー（subject）ごとにデータをグループ化し、各V遺伝子の出現回数をカウント：
v_gene_summary <- hpap_cleaned %>%
  group_by(subject, v_gene) %>%    # ドナーとV遺伝子でグループ化
  summarise(
    v_gene_count = n(),            # 各V遺伝子がそのドナーに何回出現するかをカウント
    weighted_copies = sum(copies)  # コピー数を合計して加重コピー数を計算
  )
```

```
## `summarise()` has grouped output by 'subject'. You can override using the
## `.groups` argument.
```

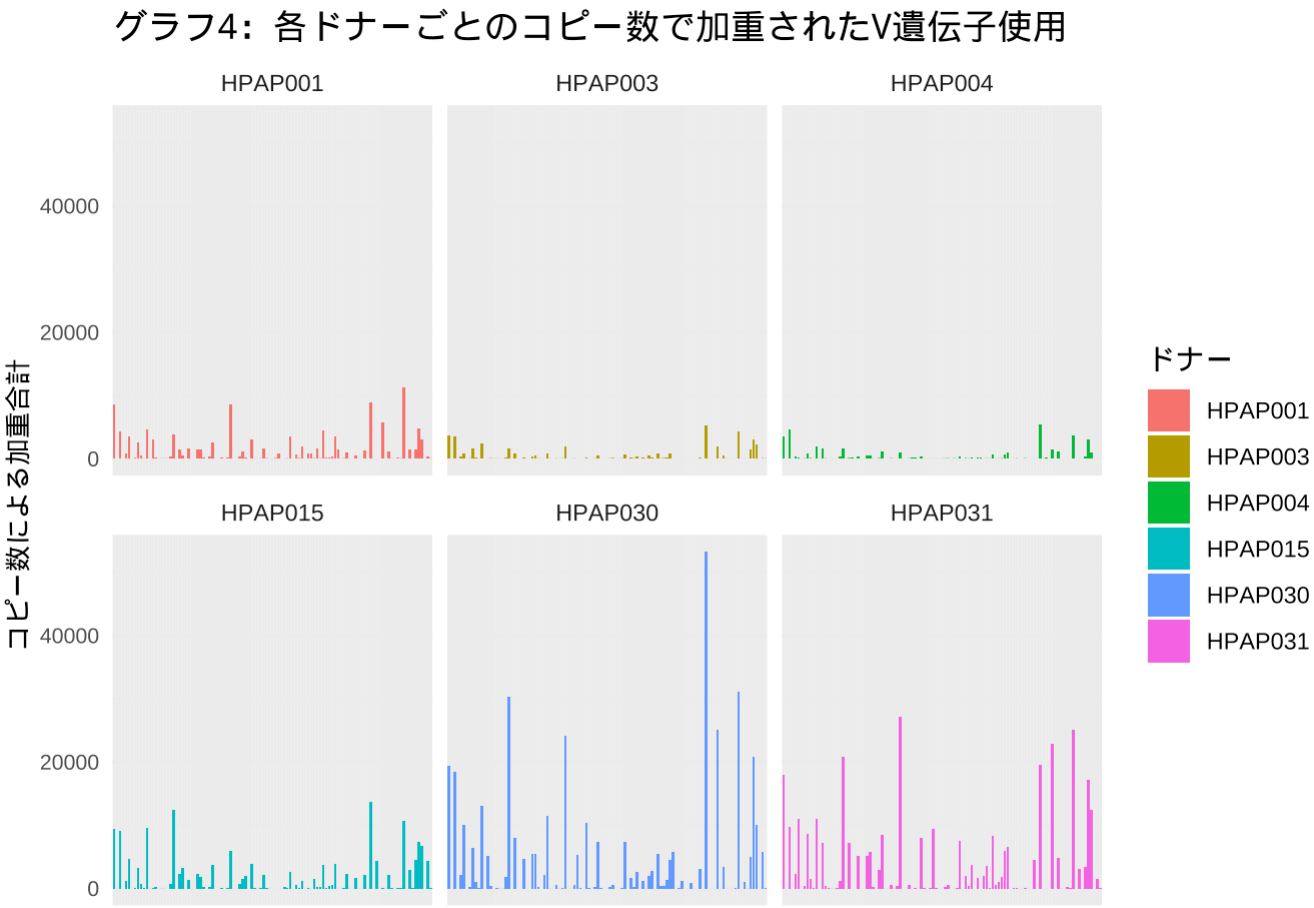
```
# グラフ2：ドナーごとのV遺伝子の分布を示す棒グラフ
ggplot(v_gene_summary, aes(x = v_gene, y = v_gene_count, fill = as.factor(subject))) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "グラフ2：ドナーごとのV遺伝子の分布",
       x = "",  # x軸のラベルを削除
       y = "カウント",
       fill = "ドナー") +
  theme_minimal() +
  theme(axis.text.x = element_blank(),  # x軸のテキストを削除
        axis.ticks.x = element_blank(),  # x軸の目盛りを削除
        axis.text.y = element_text(size = 8),  # y軸のテキストのサイズを変更
        axis.title.y = element_text(size = 10)) +  # y軸のタイトルのサイズを調整（オプション）
  facet_wrap(~ subject)
```



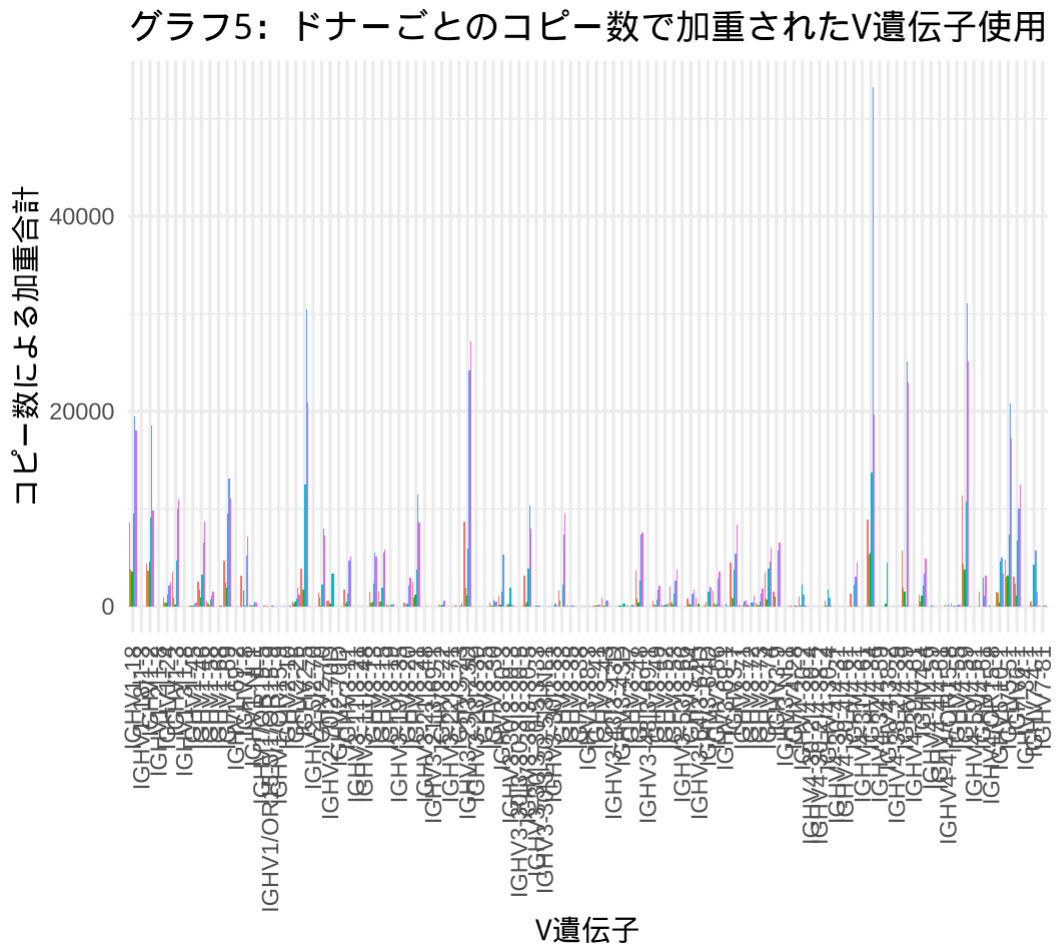

```
# グラフ3：ドナー間でのV遺伝子の分布を示す棒グラフ
ggplot(v_gene_summary, aes(x = v_gene, y = v_gene_count, fill = subject)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "グラフ3：ドナー間でのV遺伝子の分布",
       x = "V遺伝子",
       y = "カウント",
       fill = "ドナー") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```


グラフ3：ドナー間でのV遺伝子の分布
カウント
4000
3000
2000
1000
0
V遺伝子
ドナー
HPAP001
HPAP003
HPAP004
HPAP015
HPAP030
HPAP031

```
# 加重V遺伝子の分布
# グラフ4：各ドナーごとのコピー数で加重されたV遺伝子の使用状況を示すグラフ
ggplot(v_gene_summary, aes(x = v_gene, y = weighted_copies, fill = as.factor(subject))) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "グラフ4：各ドナーごとのコピー数で加重されたV遺伝子使用",
       x = "",  # x軸のラベルを削除
       y = "コピー数による加重合計",
       fill = "ドナー") +
  theme_minimal() +
  theme(axis.text.x = element_blank(),  # x軸のテキストを削除
        axis.ticks.x = element_blank(),  # x軸の目盛りを削除
        axis.text.y = element_text(size = 8),  # y軸のテキストのサイズを変更
        axis.title.y = element_text(size = 10)) +  # y軸のタイトルのサイズを調整（オプション）
  facet_wrap(~ subject)
```




```
# グラフ5：ドナーごとのコピー数で加重されたV遺伝子の使用状況を示すグラフ
ggplot(v_gene_summary, aes(x = v_gene, y = weighted_copies, fill = as.factor(subject))) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "グラフ5：ドナーごとのコピー数で加重されたV遺伝子使用",
       x = "V遺伝子",
       y = "コピー数による加重合計") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```



分析 (グラフ 2-5) コピーによる重み付

けなし： グラフ 2:

HPAP003 と HPAP004 は似たようなパターンを示しています。 HPAP015, HPAP030, HPAP031 の V-遺伝子のカウント値は、HPAP001, HPAP003, HPAP004 よりも高くなっています。 グラフ 3:

V-遺伝子のカウント値はドナーごとに異なりますが、全体的には似たような流れを示しています。 HPAP015 は他のドナーと比較して、特に次の V-遺伝子で有意に高いカウント値を示しています： IGHV1-18 IGHV1-2 IGHV1-69 IGHV2-5 IGHV4-34 HPAP030 と HPAP015 は、特に IGHV4-34 において有意に高い V-遺伝子カウント値を示しています。 HPAP030 と HPAP031 は、特に次の V-遺伝子で有意に高いカウント値を示しています： IGHV2-5 IGHV3-23 | 3-23D コピーによる重み付けあり： グラフ 4:

コピーによる重み付け後、全体的な V-遺伝子のカウント値は、重み付けなしのグラフよりも低くなります。 HPAP003 と HPAP004 は、以前と同様のパターンを維持しています。 対照的に、HPAP030 と HPAP031 は他のドナーよりも高い V-遺伝子のカウント値を示し、違いがより顕著になっています。 HPAP001 と HPAP015 は、似たような波形のパターンを示しています。 グラフ 5:

V-遺伝子のカウント値はドナーごとに異なりますが、全体的なパターンは一貫しています。HPAP030 と HPAP031 は、特に次の V-遺伝子で他のドナーよりも有意に高いコピー値を示しています： IGHV2-5 IGHV3-23 | 3-23D IGHV4-34 IGHV4-39 IGHV4-59 結論： 重み付けありとなしのデータを比較すると、重み付け後の V-遺伝子のカウント値は一般的に低くなります。 特に、HPAP030 と HPAP031 は、他のドナーと比較して IGHV2-5 と IGHV3-23 | 3-23D の値が劇的に高くなっています。 さらに、HPAP030 の IGHV4-34 は、全体的に常に高い値を示しています。 3) 疾患はクローン性レパートリーに影響を与えるか？ この最終課題では、1つの追加情報を導入します。HPAP001、HPAP003、HPAP004 は全て血清学的に健康な個体であり、HPAP015、HPAP030、HPAP031 はすべて1型糖尿病と診断された個体です。これを踏まえて、健康なコントロール群と糖尿病群の間に違いがあるかを調べてみてください。使用する方法やフィールドは自由ですが、少なくとも1〜2つの図を作成して結果を示してください。個々の列を単純に見るだけでは不十分である可能性があるため、フィールドの組み合わせが疾患と健康を区別する方法を考慮してください。

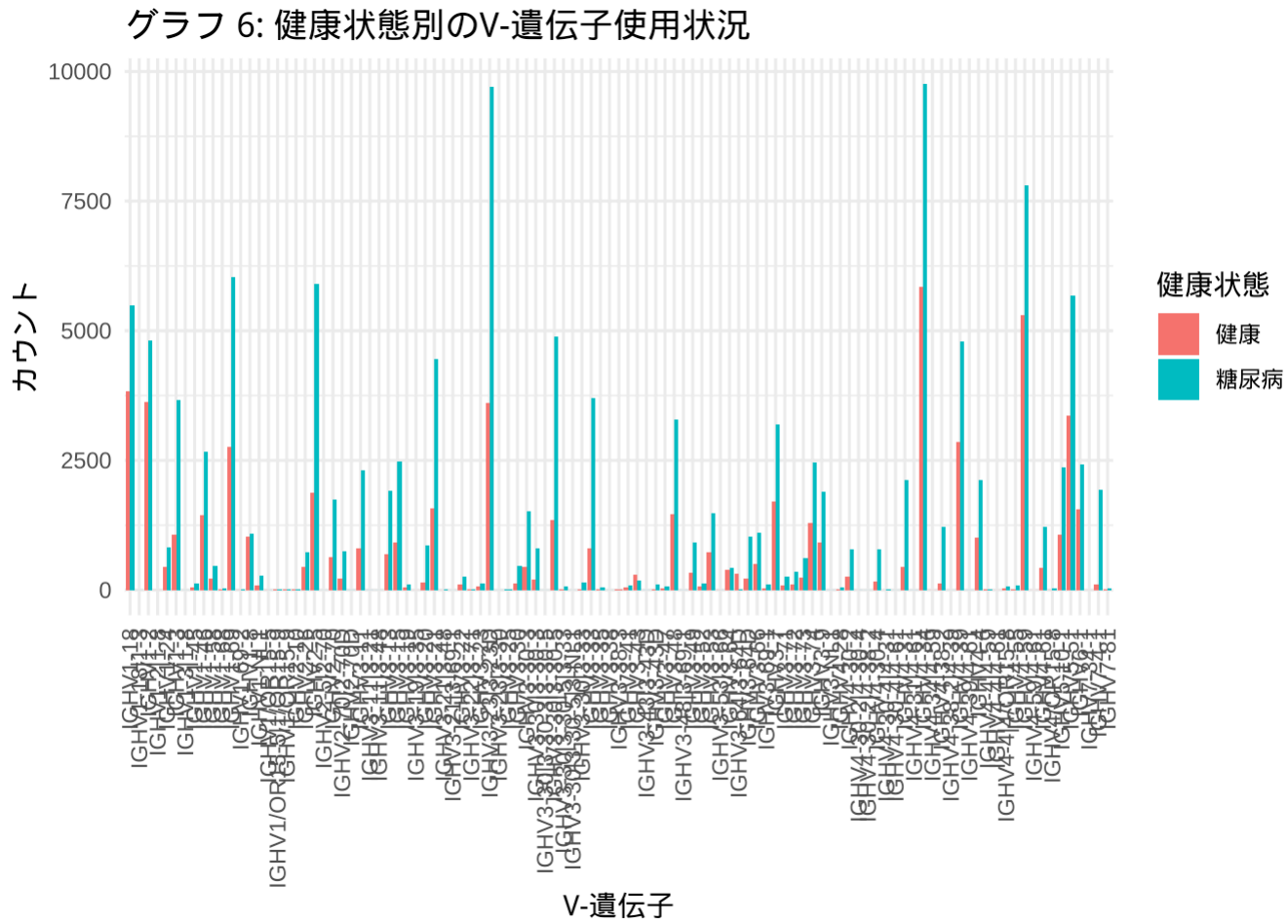
```
# 被験者列に基づいて、健康な個体と糖尿病のある個体を区別する新しい列を作成
hpap_cleaned$health_status[hpap_cleaned$subject == "HPAP001"] <- "健康"
```

```
## Warning: Unknown or uninitialised column: `health_status`.
```

```
hpap_cleaned$health_status[hpap_cleaned$subject == "HPAP003"] <- "健康"
hpap_cleaned$health_status[hpap_cleaned$subject == "HPAP004"] <- "健康"

hpap_cleaned$health_status[hpap_cleaned$subject == "HPAP015"] <- "糖尿病"
hpap_cleaned$health_status[hpap_cleaned$subject == "HPAP030"] <- "糖尿病"
hpap_cleaned$health_status[hpap_cleaned$subject == "HPAP031"] <- "糖尿病"

# 健康な個体と糖尿病のある個体を区別する新しい列を作成
# 列名を確認
colnames(hpap_cleaned) # 確認
```

```
##  [1] "clone_id"       "subject"        "v_gene"         "j_gene"
##  [5] "functional"     "cdr3_nt"        "cdr3_num_nts"   "cdr3_aa"
##  [9] "uniques"        "instances"      "copies"         "germline"
## [13] "avg_v_identity" "top_copy_seq"   "health_status"
```

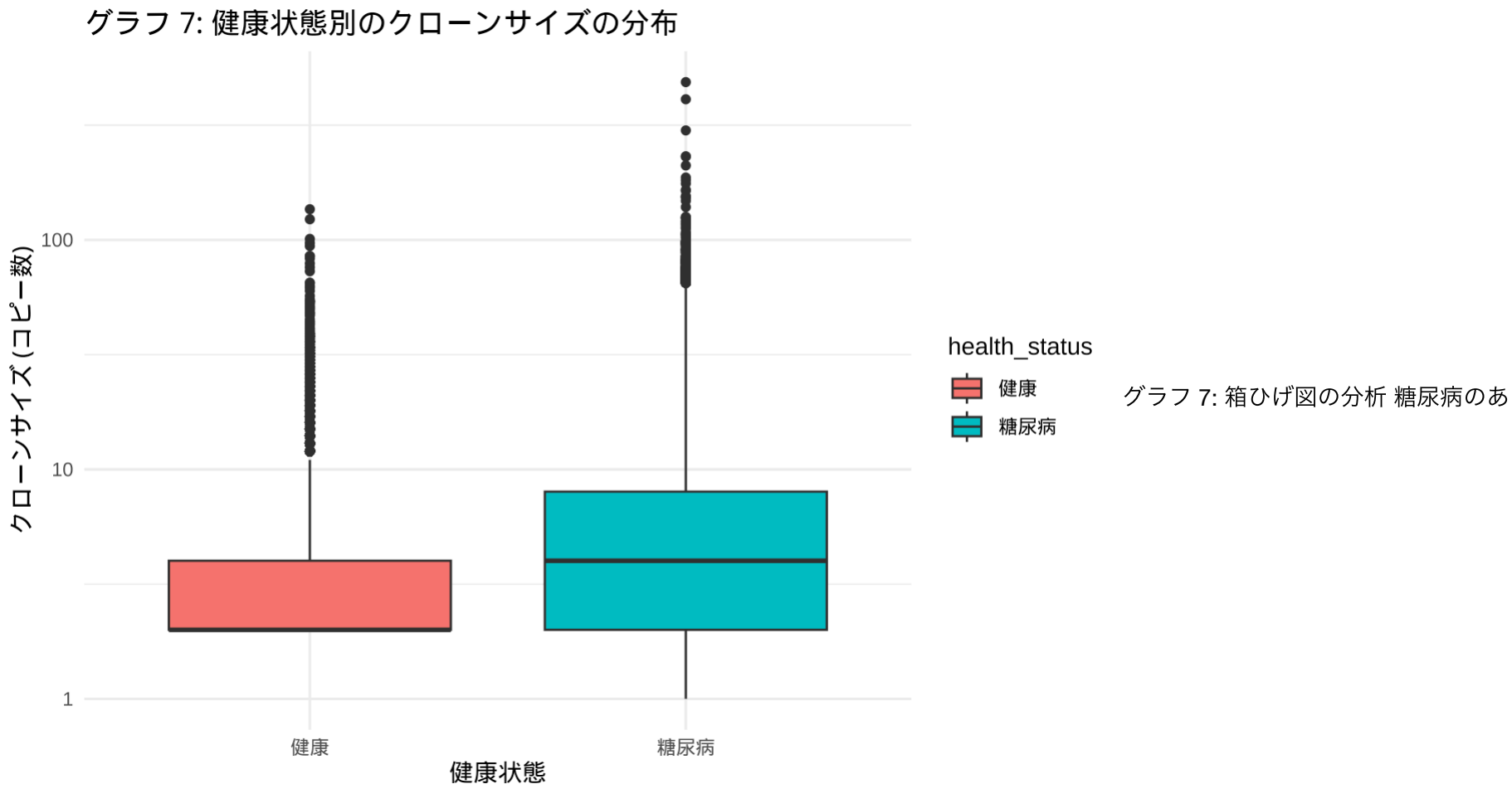
```
# グラフ 6: 健康状態別のV−遺伝子使用状況の可視化
ggplot(data = hpap_cleaned, aes(x = v_gene, fill = health_status)) +
  geom_bar(position = "dodge") +
  labs(title = "グラフ 6: 健康状態別のV−遺伝子使用状況",   # タイトルを追加
       x = "V−遺伝子",
       y = "カウント",
       fill = "健康状態") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```



グラフ 6: 健康状態別のV-遺伝子使用

状況の分析 全体的に、糖尿病のある個体のV-遺伝子のクローンのカウントは、健康な個体よりも高くなっています。

```
# グラフ 7: 箱ひげ図
# 「コピー」の大きな外れ値が原因で分布が理解しにくかったため、
# 見やすくするために対数変換を使用しました。
ggplot(data = hpap_cleaned, aes(x = health_status, y = copies, fill = health_status)) +
  geom_boxplot() +
  scale_y_log10() +  # Y軸のスケールを調整して分布を見やすくしました
  labs(title = "グラフ 7: 健康状態別のクローンサイズの分布",
       x = "健康状態",
       y = "クローンサイズ（コピー数)") +
  theme_minimal()
```



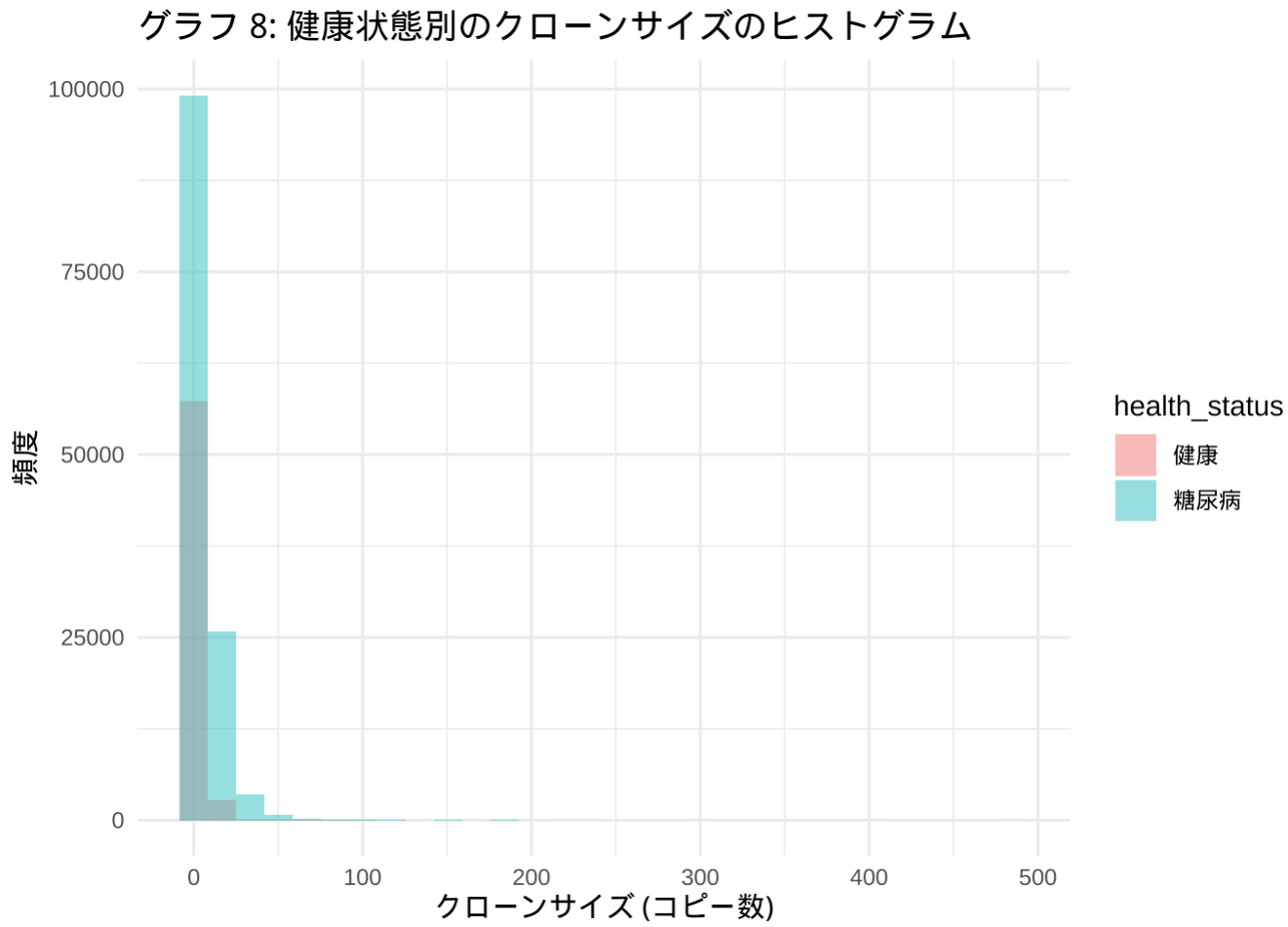
グラフ 7: 箱ひげ図の分析 糖尿病のある個体と健康な個体のクローンサイズを比較するために箱ひげ図を使用しました。

糖尿病:

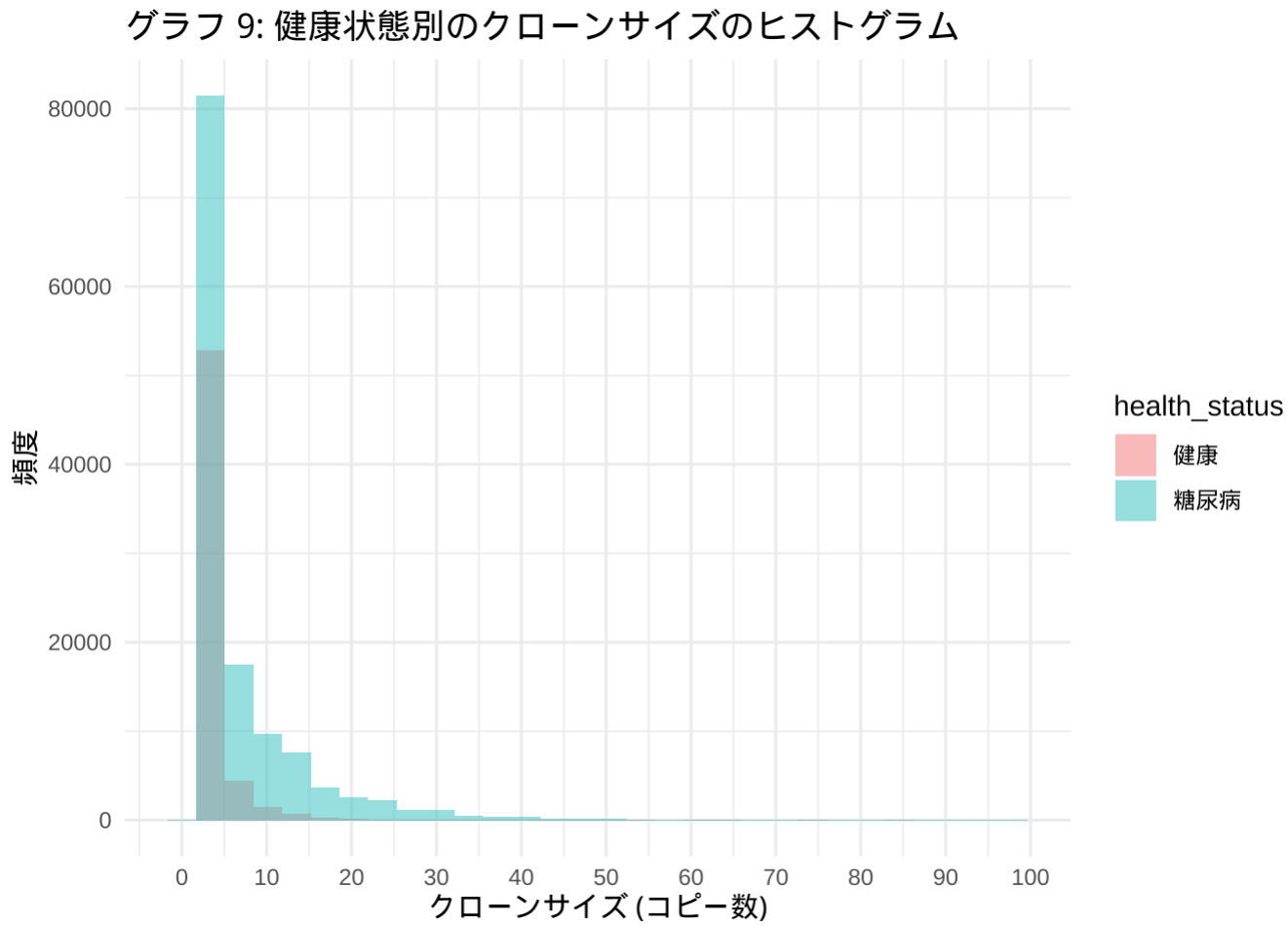
箱の位置は低い。 四分位範囲 (IQR) は約3 (25%) から8 (75%) まで。 中央値は約6。 健康なサンプルに比べてデータが分散している。 健康:

箱の位置は高い。 IQR は約3 (25%) から6 (75%) まで。 中央値は約3。 糖尿病グループよりもデータがより集中している。 結論: 糖尿病のある個体のクローンサイズは健康な個体よりも大きくなる傾向があります。 健康な個体の中央値は約3で、糖尿病のある個体の中央値は約6です。

```
# グラフ 8: ヒストグラム
ggplot(data = hpap_cleaned, aes(x = copies, fill = health_status)) +
  geom_histogram(position = "identity", alpha = 0.5, bins = 30) +
  labs(title = "グラフ 8: 健康状態別のクローンサイズのヒストグラム",
       x = "クローンサイズ (コピー数)",
       y = "頻度") +
  theme_minimal()
```




```
# 外れ値を除去することでグラフが見やすくなります。
cleaned_data2 <- hpap_cleaned %>% filter(copies < 100) # 外れ値を除去

# グラフ 9: ヒストグラム (詳細な検討)
ggplot(data = cleaned_data2, aes(x = copies, fill = health_status)) +
  geom_histogram(position = "identity", alpha = 0.5, bins = 30) +
  labs(title = "グラフ 9: 健康状態別のクローンサイズのヒストグラム",
       x = "クローンサイズ (コピー数)",
       y = "頻度") +
  scale_x_continuous(breaks = seq(0, 100, by = 10)) +  # x軸の目盛りを設定
  theme_minimal()
```




```
# 健康状態別の件数をカウント
health_count <- table(hpap_cleaned$health_status)
# 表 10: 件数を表示
print(health_count)
```

```
##
##    健康 糖尿病
##   60190 129625
```

```
# 健康状態別のクローンサイズの要約統計量
dis_table <- by(hpap_cleaned$copies, hpap_cleaned$health_status, summary)
# 表 11
print(dis_table)
```

```
## hpap_cleaned$health_status: 健康
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.000   2.000   2.000   3.458   4.000 136.000
## ------------------------------------------------------------
## hpap_cleaned$health_status: 糖尿病
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   4.000   6.902   8.000 487.000
```

分析 (Graph 8, 9)

Graph 8 は、健康な個人と1型糖尿病のある個人のクローンサイズの分布を示しています。グラフから、1型糖尿病のある個人の方が健康な個人よりもクローン数が著しく多いことがわかります。

Graph 9 は、クローン分布をさらに詳しく調べたもので、1型糖尿病のクローンは健康なクローンに比べて、数もサイズも大きいことが示されています。

Table 10 は、各グループのクローン数をまとめています。1型糖尿病のある個人は129,625個のクローン、健康な個人は60,190個のクローンがあります。

Table 11 は、健康状態に基づくクローンサイズの要約統計を示しています。糖尿病グループでは、最小クローンサイズは1、平均サイズは6.902、最大サイズは487です。一方、健康なグループでは、最小クローンサイズは2、平均サイズは3.458、最大サイズは136です。

結論 1型糖尿病のある個人は、健康な個人の約2倍のクローン数を持っています。さらに、クローンサイズが2未満または136より大きい個人は、1型糖尿病を持つ可能性が高いことが示唆されています。

4. クローンサイズと突然変異の相関関係は、健康な個人と1型糖尿病のある個人で異なるか？ (自由調査)

```
# 相関テスト
# 全体の相関テスト
correlation_test <- cor.test(hpap_cleaned$copies, hpap_cleaned$avg_v_identity, method = "pearson")
# 表 12: 全体の相関テスト
print(correlation_test)
```

```
##
##  Pearson's product-moment correlation
##
## data:  hpap_cleaned$copies and hpap_cleaned$avg_v_identity
## t = 28.348, df = 189813, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.06044939 0.06940878
## sample estimates:
##       cor
## 0.0649304
```

```
# 健康群の相関テスト
correlation_test_healthy <- cor.test(hpap_cleaned$copies[hpap_cleaned$health_status == "健康"], hpap_cleaned$avg_v
_identity[hpap_cleaned$health_status == "健康"], method = "pearson")
# 表 13: 健康群の相関テスト
print(correlation_test_healthy)
```

```
##
##  Pearson's product-moment correlation
##
## data:  hpap_cleaned$copies[hpap_cleaned$health_status == "健康"] and hpap_cleaned$avg_v_identity[hpap_cleaned$h
ealth_status == "健康"]
## t = -30.386, df = 60188, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1307792 -0.1150428
## sample estimates:
##        cor
## -0.1229187
```

```
# 糖尿病群の相関テスト
correlation_test_diabetes <- cor.test(hpap_cleaned$copies[hpap_cleaned$health_status == "糖尿病"], hpap_cleaned$av
g_v_identity[hpap_cleaned$health_status == "糖尿病"], method = "pearson")
# 表 14: 糖尿病群の相関テスト
print(correlation_test_diabetes)
```

```
##
##  Pearson's product-moment correlation
##
## data:  hpap_cleaned$copies[hpap_cleaned$health_status == "糖尿病"] and hpap_cleaned$avg_v_identity[hpap_cleaned
$health_status == "糖尿病"]
## t = 36.829, df = 129623, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.09637261 0.10714752
## sample estimates:
##      cor
## 0.101763
```

```
# 健康群の回帰分析
healthy_model <- lm(avg_v_identity ~ copies, data = hpap_cleaned[hpap_cleaned$health_status == "健康", ])
# 表 15: 健康群の回帰分析
summary(healthy_model)
```

```
##
## Call:
## lm(formula = avg_v_identity ~ copies, data = hpap_cleaned[hpap_cleaned$health_status ==
##     "健康", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.26626 -0.01846  0.02168  0.02219  0.12945
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.803e-01  1.966e-04 4987.43   <2e-16 ***
## copies      -1.256e-03  4.134e-05  -30.39   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0331 on 60188 degrees of freedom
## Multiple R-squared:  0.01511,    Adjusted R-squared:  0.01509
## F-statistic: 923.3 on 1 and 60188 DF,  p-value: < 2.2e-16
```

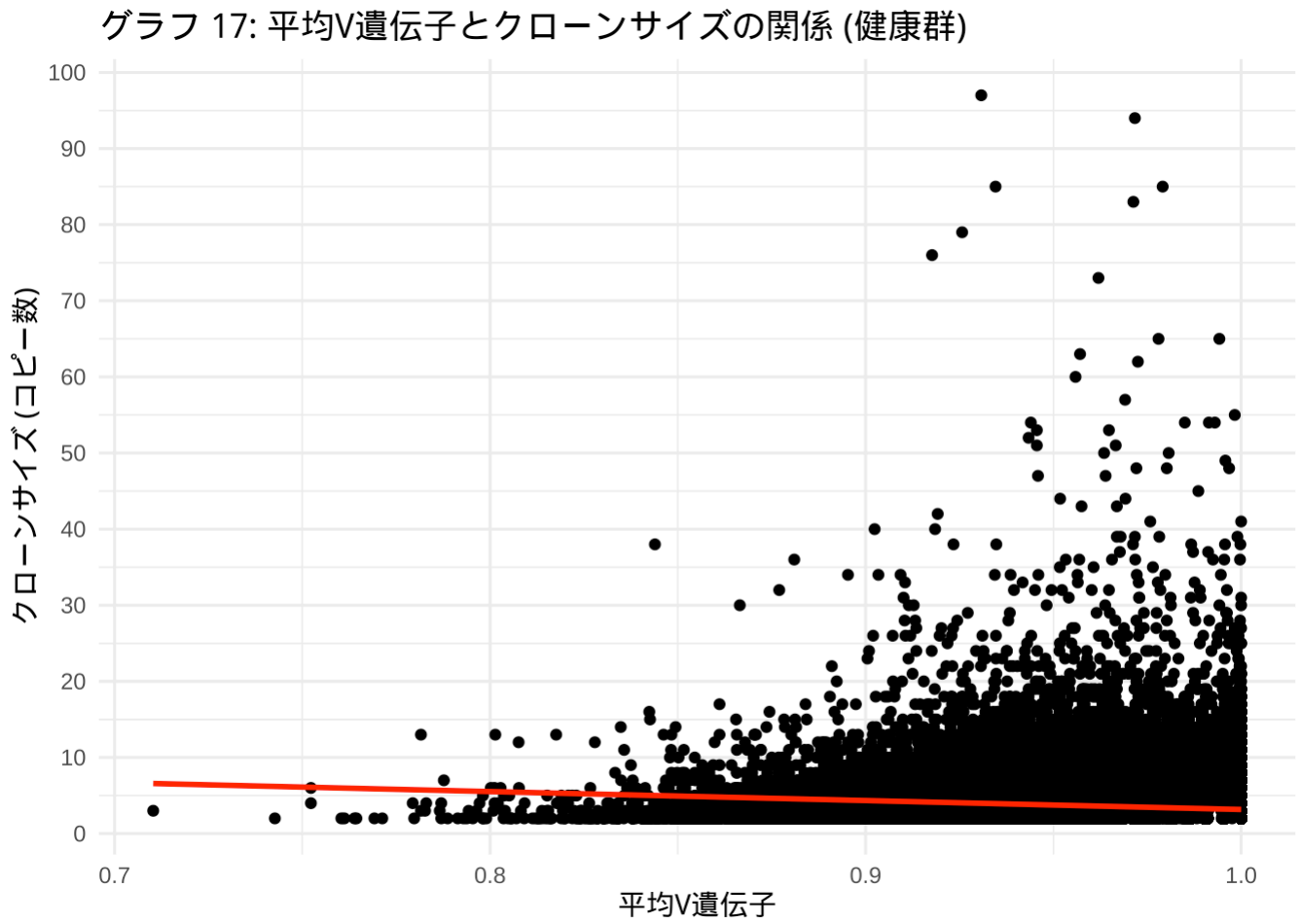
```
# 糖尿病群の回帰分析
diabetes_model <- lm(avg_v_identity ~ copies, data = hpap_cleaned[hpap_cleaned$health_status == "糖尿病", ])
# 表 16: 糖尿病群の回帰分析
summary(diabetes_model)
```

```
##
## Call:
## lm(formula = avg_v_identity ~ copies, data = hpap_cleaned[hpap_cleaned$health_status ==
##     "糖尿病", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32662 -0.01460  0.01495  0.02311  0.02467
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 9.749e-01  1.136e-04 8578.67   <2e-16 ***
## copies      3.880e-04  1.053e-05   36.83   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03145 on 129623 degrees of freedom
## Multiple R-squared:  0.01036,    Adjusted R-squared:  0.01035
## F-statistic:  1356 on 1 and 129623 DF,  p-value: < 2.2e-16
```

相関テストの分析 (表 12 - 16) 健康群 切片 (Intercept): 0.9803 クローンサイズ係数 (Copies Coefficient): -0.001256 (クローンサイズが増加するにつれて、平均V遺伝子のアイデンティティは減少する傾向があります。) R二乗値 (R-squared): 0.01511 有意性 (Significance): p値 < 2e-16 糖尿病群 切片 (Intercept): 0.9749 クローンサイズ係数 (Copies Coefficient): 0.000388 (クローンサイズが増加するにつれて、平均V遺伝子のアイデンティティは増加する傾向があります。) R二乗値 (R-squared): 0.01036 有意性 (Significance): p値 < 2e-16 まとめ 健康群および糖尿病群の両方で、p値が2e-16であり、帰無仮説に対する強い証拠が示されています。これは、クローンサイズが両群において平均V遺伝子のアイデンティティに有意に影響を与えることを示唆しています。健康群では、クローンサイズと平均V遺伝子のアイデンティティの間に負の相関が見られ、一方、糖尿病群では正の相関が見られます。

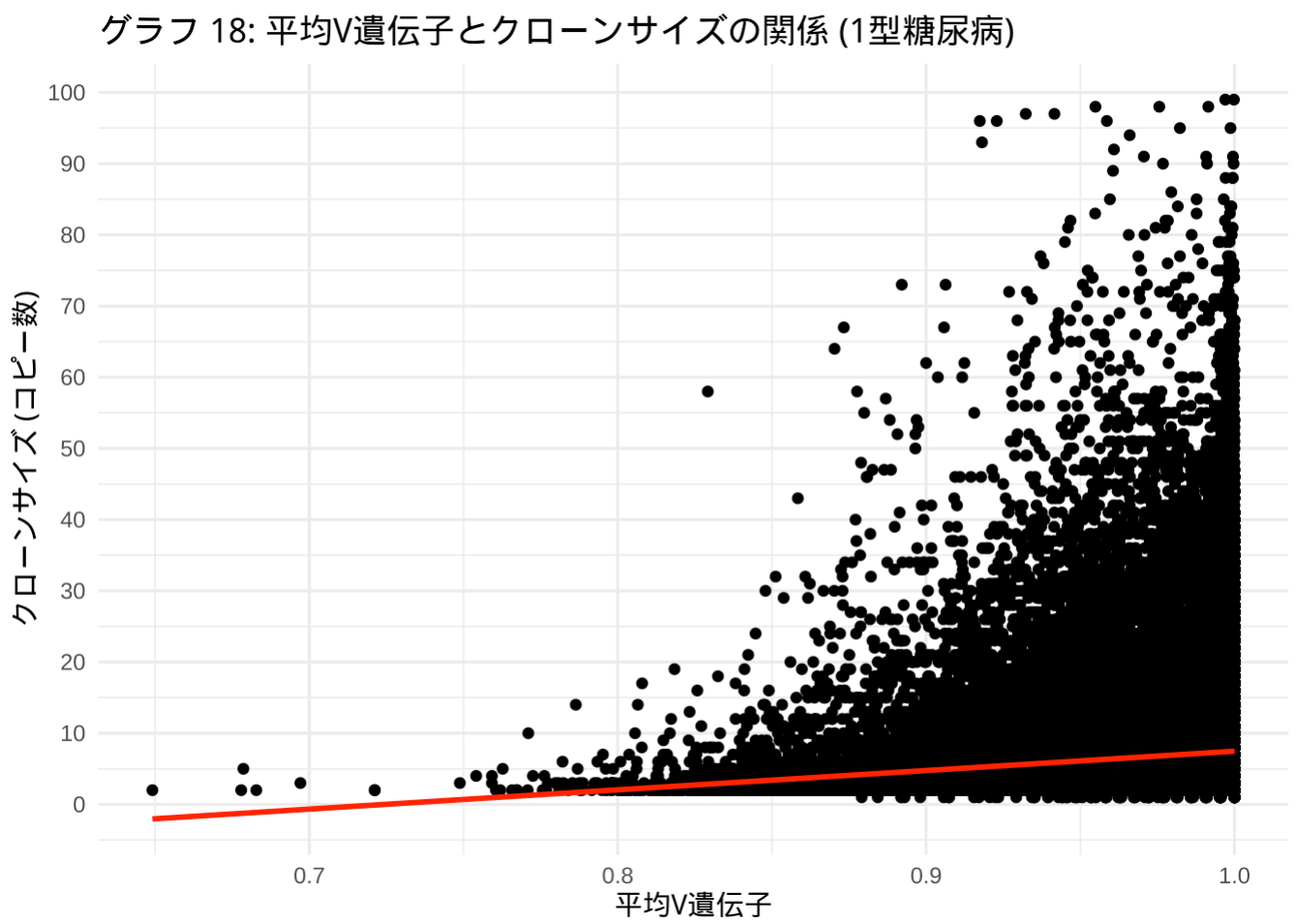
```
# データ可視化
cleaned_data2 <- hpap_cleaned %>% filter(copies < 100) # 外れ値の除去

# グラフ 17: 健康群
ggplot(data = cleaned_data2[cleaned_data2$health_status == "健康", ], aes(x = avg_v_identity, y = copies)) +
  geom_point() +
  geom_smooth(method = "lm", color = "red", se = FALSE) +  # 実際のデータに基づく回帰線を追加
  scale_y_continuous(breaks = seq(0, 150, by = 10)) +  # y軸の目盛りをカスタマイズ
  labs(title = "グラフ 17: 平均V遺伝子とクローンサイズの関係 (健康群)",
       x = "平均V遺伝子",
       y = "クローンサイズ (コピー数)") +
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
# グラフ 18: 糖尿病群
ggplot(data = cleaned_data2[cleaned_data2$health_status == "糖尿病", ], aes(x = avg_v_identity, y = copies)) +
  geom_point() +
  geom_smooth(method = "lm", color = "red", se = FALSE) +  # 実際のデータに基づく回帰線を追加
  scale_y_continuous(breaks = seq(0, 150, by = 10)) +  # y軸の目盛りをカスタマイズ
  labs(title = "グラフ 18: 平均V遺伝子とクローンサイズの関係 (1型糖尿病)",
       x = "平均V遺伝子",
       y = "クローンサイズ (コピー数)") +
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



分析 (表 17 と 18) 表 17: 健康群 -平

均V遺伝子アイデンティティが増加すると、クローンサイズはわずかに増加します。 -多くの点は、平均V遺伝子アイデンティティが8.5付近で、クローンサイズは1から20の範囲に集中しています。

表 18: 糖尿病群 -平均V遺伝子アイデンティティが増加すると、クローンサイズは健康群と比較してより急激に増加します。 -多くの点は、平均V遺伝子アイデンティティが8.5付近で、クローンサイズは1から40の範囲に集中しています。

比較 糖尿病群は、健康群と比較してクローンサイズの増加がより顕著です。