MAGERI benchmark using reference standard DNA library

Mikhail Shugay

March 1, 2017

Load metadata

```
df.vmeta <- read.table("hd734_variant_metadata.txt", sep="\t", header=T) # variants observed in HD734 a
df.smeta <- read.table("sample_metadata.txt", sep="\t", header=T) # metadata for amplicon sequencing sa
```

VCF parsing function

```
library(stringr)
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(ggplot2)
library(ggbeeswarm)
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
```

```
read_vcf <- function(file_name) {
  .vcf <- read.table(file_name, header = F, sep = "\t", stringsAsFactors = F)
  colnames(.vcf) <- c("chromosome", "position", "skip1", "from", "to", "qual", "skip2", "info", "skip3"
  .vcf$skip1 <- NULL
  .vcf$skip2 <- NULL
  .vcf$skip3 <- NULL
  .vcf$skip4 <- NULL

  .vcf$qual <- as.integer(.vcf$qual)
  .vcf <- subset(.vcf, nchar(from) == 1 &
                 nchar(to) == 1 & !is.na(qual)) # no indels

  .infosplit <- str_split_fixed(.vcf$info, regex("[=;]"), 15)[,c(2, 4, 12, 14)]

  .vcf$coverage <- as.numeric(.infosplit[,1])
  .vcf$frequency <- as.numeric(.infosplit[,2])
  .vcf$bb.a <- as.numeric(.infosplit[,3])
  .vcf$bb.b <- as.numeric(.infosplit[,4])
  .vcf$info <- NULL

  .vcf$count <- as.integer(round(.vcf$coverage * .vcf$frequency))

  .vcf
}

head(read_vcf("p126.h4_2_ballast_m1.vcf"))
```

```
## Warning in read_vcf("p126.h4_2_ballast_m1.vcf"): NAs introduced by coercion
```

```
##   chromosome  position from to qual coverage    frequency      bb.a
## 1       chr2 212295704    C  A    8     2265 0.0004415011 0.9957058
## 2       chr2 212295705    C  A    8     2265 0.0004415011 0.9957058
## 3       chr2 212295713    A  G    7     2265 0.0004415011 1.0295689
## 4       chr2 212295718    G  A    6     2265 0.0004415011 2.1314828
## 5       chr2 212295725    C  A    8     2264 0.0004416961 0.9957058
## 6       chr2 212295732    C  T   24     2264 0.0013250883 2.1314828
##       bb.b count
## 1 21862.23     1
## 2 21862.23     1
## 3 16394.12     1
## 4 24344.82     1
## 5 21862.23     1
## 6 24344.82     3
```

Read samples with HD734 standard DNA and control human DNA, append metadata

```
df <- data.frame()

read_vcf_with_metadata <- function(file_name, primer_set, replica, ratio, type) {
  .vcf <- read_vcf(file_name)
  .vcf <- merge(.vcf, df.vmeta, all.x = type != "standard", all.y = F)
  .vcf$known.frequency <- .vcf$known.frequency * ratio
  .vcf$known.frequency[is.na(.vcf$known.frequency)] <- 0
  .vcf$primer_set <- primer_set
  .vcf$replica <- primer_set
  .vcf$type <- type

  .vcf <- subset(.vcf, frequency < 0.4 & count > 0) # remove alleles in control

  .vcf
}

for (i in 1:nrow(df.smeta)) {
  df <- with(df.smeta, rbind(df,
                             read_vcf_with_metadata(paste(prefix[i], "vcf", sep="."),
                                                    primer_set[i],
                                                    replica[i],
                                                    ratio[i],
                                                    type[i])))
}
```

```
## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion

## Warning in read_vcf(file_name): NAs introduced by coercion
```

Group observed variants into tiers

```r
df$tier <- cut(df$known.frequency, c(-1, 0, 0.009, 0.02, 1))
levels(df$tier) <- c("error", "0.1%", "1%", "5%")
summary(df$tier)
```

```
## error  0.1%    1%    5%
##  2468    43    46    12
```

Number of detected variants and sampling

```r
df.v.summary <- df.vmeta
df.v.summary$known.frequency <- df.v.summary$known.frequency * 0.1 # 1:9 diluted samples
df.v.summary$ratio <- "1:10"
df.v.summary.2 <- df.vmeta
df.v.summary.2$ratio <- "1:1"
df.v.summary <- rbind(df.v.summary, df.v.summary.2)
df.v.summary$tier  <- cut(df.v.summary$known.frequency, c(-1, 0, 0.009, 0.02, 1))
```

```r
levels(df.v.summary$tier) <- c("error", "0.1%", "1%", "5%")

print(df.v.summary)
```

```
##    chromosome  position             from to
## 1        chr7 140453136                A  T
## 2        chr7 140453136                A  C
## 3        chr7 140453137                C  T
## 4        chr7  55241707                G  A
## 5        chr7  55259515                T  G
## 6        chr7  55259524                T  A
## 7        chr7  55249071                C  T
## 8       chr19   3118942                A  T
## 9        chr2 209113112                C  T
## 10       chr2 209113113                G  A
## 11      chr12  25378562                C  T
## 12      chr12  25398284                C  G
## 13      chr12  25398285                C  A
## 14      chr12  25398284                C  T
## 15      chr12  25398285                C  G
## 16      chr12  25398285                C  T
## 17      chr12  25398284                C  A
## 18      chr12  25398281                C  T
## 19      chr12  25380275                T  G
## 20      chr12  25380276                T  A
## 21       chr1 115256528                T  A
## 22       chr1 115256530                G  T
## 23       chr1 115256529                T  A
## 24       chr1 115256529                T  C
## 25       chr3 178936082                G  A
## 26       chr3 178936091                G  A
## 27       chr3 178952085                A  G
## 28       chr7  55242465 GGGAATTAAGAGAAGC  G
## 29       chr7 140453136                A  T
## 30       chr7 140453136                A  C
## 31       chr7 140453137                C  T
## 32       chr7  55241707                G  A
## 33       chr7  55259515                T  G
## 34       chr7  55259524                T  A
## 35       chr7  55249071                C  T
## 36      chr19   3118942                A  T
## 37       chr2 209113112                C  T
## 38       chr2 209113113                G  A
## 39      chr12  25378562                C  T
## 40      chr12  25398284                C  G
## 41      chr12  25398285                C  A
## 42      chr12  25398284                C  T
## 43      chr12  25398285                C  G
## 44      chr12  25398285                C  T
## 45      chr12  25398284                C  A
## 46      chr12  25398281                C  T
## 47      chr12  25380275                T  G
## 48      chr12  25380276                T  A
## 49       chr1 115256528                T  A
## 50       chr1 115256530                G  T
## 51       chr1 115256529                T  A
## 52       chr1 115256529                T  C
## 53       chr3 178936082                G  A
## 54       chr3 178936091                G  A
## 55       chr3 178952085                A  G
## 56       chr7  55242465 GGGAATTAAGAGAAGC  G
##                                    id known.frequency  reference ratio tier
## 1            BRAF_V600E,BRAF_V600K          0.0080   BRAF_E15  1:10 0.1%
## 2            BRAF_V600G,BRAF_V600R          0.0010   BRAF_E15  1:10 0.1%
## 3  BRAF_V600M,BRAF_V600K,BRAF_V600R          0.0010   BRAF_E15  1:10 0.1%
## 4                        EGFR_G719S          0.0167   EGFR_E18  1:10   1%
## 5                        EGFR_L858R          0.0010   EGFR_E21  1:10 0.1%
## 6                        EGFR_L861Q          0.0010   EGFR_E21  1:10 0.1%
## 7                        EGFR_T790M          0.0010   EGFR_E20  1:10 0.1%
## 8                       GNA11_Q209L          0.0013   GNA11_E5  1:10 0.1%
## 9                        IDH1_R132C          0.0013    IDH1_E4  1:10 0.1%
## 10                       IDH1_R132H          0.0013    IDH1_E4  1:10 0.1%
## 11                       KRAS_A146T          0.0013    KRAS_E4  1:10 0.1%
## 12                        KRAS_G12A          0.0013    KRAS_E2  1:10 0.1%
## 13                        KRAS_G12C          0.0013    KRAS_E2  1:10 0.1%
## 14                        KRAS_G12D          0.0013    KRAS_E2  1:10 0.1%
## 15                        KRAS_G12R          0.0013    KRAS_E2  1:10 0.1%
## 16                        KRAS_G12S          0.0013    KRAS_E2  1:10 0.1%
## 17                        KRAS_G12V          0.0013    KRAS_E2  1:10 0.1%
## 18                        KRAS_G13D          0.0250    KRAS_E2  1:10   5%
## 19                        KRAS_Q61H          0.0013    KRAS_E3  1:10 0.1%
## 20                        KRAS_Q61L          0.0013    KRAS_E3  1:10 0.1%
## 21                        NRAS_Q61H          0.0013    NRAS_E3  1:10 0.1%
## 22                        NRAS_Q61K          0.0013    NRAS_E3  1:10 0.1%
## 23                        NRAS_Q61L          0.0013    NRAS_E3  1:10 0.1%
## 24                        NRAS_Q61R          0.0013    NRAS_E3  1:10 0.1%
## 25                     PIK3CA_E542K          0.0013  PIK3CA_E9  1:10 0.1%
## 26                     PIK3CA_E545K          0.0013  PIK3CA_E9  1:10 0.1%
## 27                    PIK3CA_H1047R          0.0300 PIK3CA_E20  1:10   5%
## 28                  EGFR_dE746-A750          0.0010   EGFR_E19  1:10 0.1%
## 29           BRAF_V600E,BRAF_V600K          0.0800   BRAF_E15   1:1   5%
## 30           BRAF_V600G,BRAF_V600R          0.0100   BRAF_E15   1:1   1%
## 31 BRAF_V600M,BRAF_V600K,BRAF_V600R          0.0100   BRAF_E15   1:1   1%
## 32                       EGFR_G719S          0.1670   EGFR_E18   1:1   5%
## 33                       EGFR_L858R          0.0100   EGFR_E21   1:1   1%
## 34                       EGFR_L861Q          0.0100   EGFR_E21   1:1   1%
## 35                       EGFR_T790M          0.0100   EGFR_E20   1:1   1%
## 36                      GNA11_Q209L          0.0130   GNA11_E5   1:1   1%
## 37                       IDH1_R132C          0.0130    IDH1_E4   1:1   1%
## 38                       IDH1_R132H          0.0130    IDH1_E4   1:1   1%
## 39                       KRAS_A146T          0.0130    KRAS_E4   1:1   1%
## 40                        KRAS_G12A          0.0130    KRAS_E2   1:1   1%
## 41                        KRAS_G12C          0.0130    KRAS_E2   1:1   1%
## 42                        KRAS_G12D          0.0130    KRAS_E2   1:1   1%
## 43                        KRAS_G12R          0.0130    KRAS_E2   1:1   1%
## 44                        KRAS_G12S          0.0130    KRAS_E2   1:1   1%
## 45                        KRAS_G12V          0.0130    KRAS_E2   1:1   1%
## 46                        KRAS_G13D          0.2500    KRAS_E2   1:1   5%
## 47                        KRAS_Q61H          0.0130    KRAS_E3   1:1   1%
## 48                        KRAS_Q61L          0.0130    KRAS_E3   1:1   1%
## 49                        NRAS_Q61H          0.0130    NRAS_E3   1:1   1%
## 50                        NRAS_Q61K          0.0130    NRAS_E3   1:1   1%
## 51                        NRAS_Q61L          0.0130    NRAS_E3   1:1   1%
## 52                        NRAS_Q61R          0.0130    NRAS_E3   1:1   1%
## 53                     PIK3CA_E542K          0.0130  PIK3CA_E9   1:1   1%
## 54                     PIK3CA_E545K          0.0130  PIK3CA_E9   1:1   1%
## 55                    PIK3CA_H1047R          0.3000 PIK3CA_E20   1:1   5%
## 56                  EGFR_dE746-A750          0.0100   EGFR_E19   1:1   1%
```

```
df.v.summary <- df.v.summary %>%
  group_by(tier) %>%
  summarise(total = n())
df.v.summary$total <- 2 * df.v.summary$total # we have 2 independent experiments

df.v.summary <- merge(df.v.summary,
                      df %>% group_by(tier) %>%
                        summarise(found = n(),
                                  detect.prob.m = mean(1-exp(-frequency * coverage)),
                                  detect.prob.sd = sd(1-exp(-frequency * coverage))))

df.v.summary$detection.prob.lb <- with(df.v.summary,
                                       detect.prob.m - 1.96*detect.prob.sd - 5e-3)
df.v.summary$detection.prob.ub <- with(df.v.summary,
                                       detect.prob.m + 1.96*detect.prob.sd)

df.v.summary$detection.prob.ub <- ifelse(df.v.summary$detection.prob.ub > 1,
                                         1,
                                         df.v.summary$detection.prob.ub )

library(scales)

fig3 <- ggplot(df.v.summary, aes(x=tier)) +
  geom_bar(stat="identity", aes(y=detection.prob.ub), fill = "grey70", width=0.5)+
  geom_bar(stat="identity", aes(y=detection.prob.lb), fill = "white")+
  geom_point(aes(y = found/total), size = 5, color="red", shape="+") +
  geom_text(aes(y = found/total - 0.05, label = paste(found, total, sep="/"))) +
  scale_y_continuous("Detection rate", labels=percent) +
  xlab("Variant tier") +
  theme_bw() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_blank())

save(file = "../figures/fig3.Rda", fig3)
```

```
fig3
```

Quality score and error P-values

```
df.e <- df %>%
  filter(type == "blank") %>%
  mutate(true.p.value = -10 * log10(1 - (rank(frequency) - 0.5) / n()))

fig4 <- ggplot(df.e, aes(true.p.value, qual)) +
```

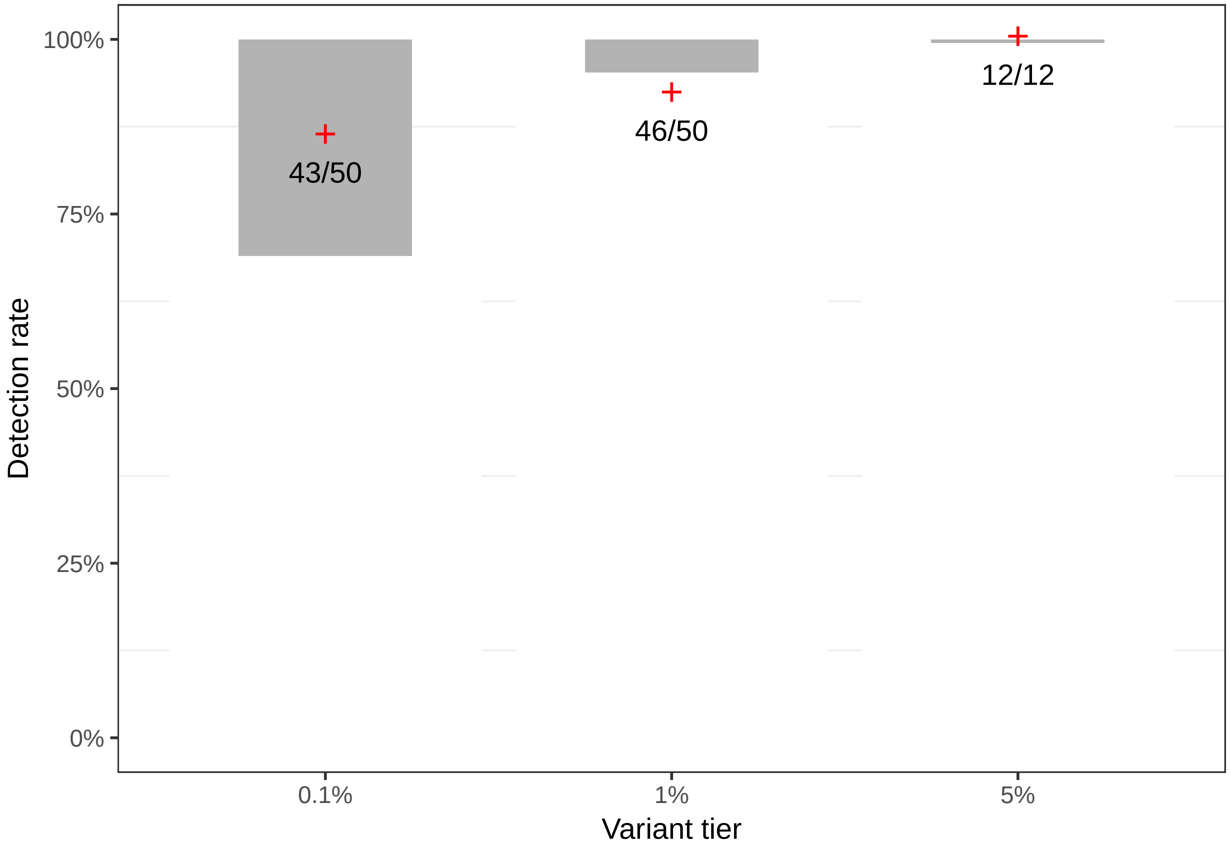

Figure 1: Number of detected variant for each variant frequency tier across two independent experiments with the reference standard. Shaded areas show the 95% confidence intervals for expected fraction of recovered variants, i.e. binomial proportion confidence intervals built using known variant frequency and template coverage.

```
  geom_abline(slope = 1, intercept = 0, linetype = "dashed") +
  geom_smooth(color="red", fill="grey", alpha=0.5) +
  scale_x_continuous("-10 log10 P-value", limits = c(0, 40)) +
  scale_y_continuous("Q score", limits = c(0, 40)) +
  theme_bw()

save(file = "../figures/fig4.Rda", fig4)
```

```
fig4
```

```
## `geom_smooth()` using method = 'gam'
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

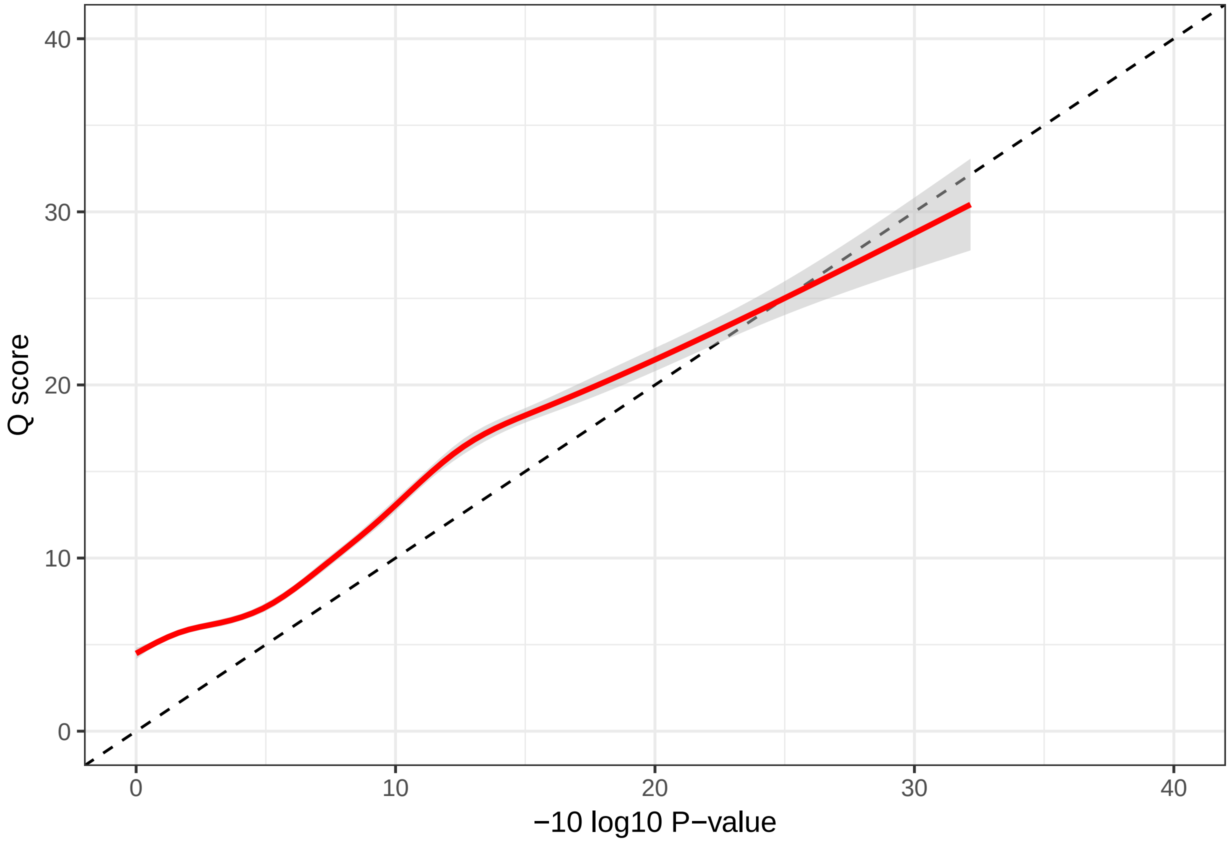

Figure 2: MAGERI Q score and the empirical P-values of erroneous variants detected in control donor DNA.

```
nrow(df.e)
```

```
## [1] 2468
```

```
cor(df.e$qual, df.e$true.p.value, method="pearson")
```

```
## [1] 0.8311097
```

Frequency of variants and errors

```
fig5 <- ggplot(df, aes(x=frequency, color = tier)) +
  geom_density(aes(fill=tier), alpha=0.7) +
  ylab("Density") +
  scale_x_log10("Frequency") +
  scale_color_brewer("Variant tier", palette = "Set1") +
  scale_fill_brewer("Variant tier", palette = "Set1") +
  theme_bw()

save(file = "../figures/fig5.Rda", fig5)
```

```
fig5
```

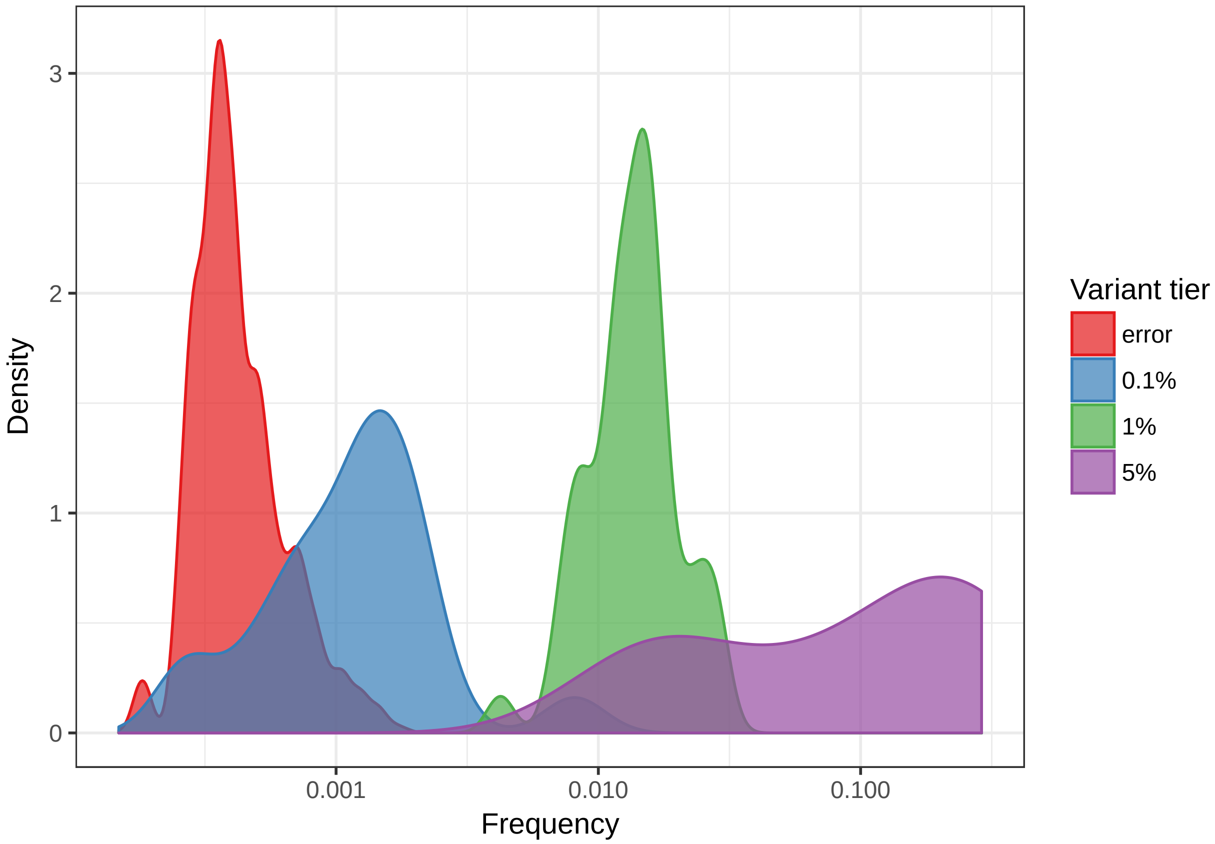

Figure 3: Frequency distribution of known Tru-Q 7 variants coming from each frequency tier and errors in the control donor DNA.

```
df.known <- subset(df, known.frequency > 0)
nrow(df.known)
```

```
## [1] 101
```

```
mean((df.known$frequency - df.known$known.frequency) / df.known$known.frequency)
```

```
## [1] 0.07155274
```

```
cor(df.known$frequency, df.known$known.frequency, method="spearman")
```

```
## [1] 0.830348
```

Variant quality scores

```
fig6 <- ggplot(df,aes(tier, qual, fill=tier)) +
  geom_hline(yintercept = 13.0103, linetype ="dotted") +
  geom_hline(yintercept = 20.0103, linetype ="dashed") +
  geom_quasirandom(varwidth = T, shape=21, color="grey10") +
  annotate("text", label = "p < 0.05", x=4.3,y=13.0103+2) +
  annotate("text", label = "p < 0.01", x=4.3,y=20.0103+4) +
  scale_fill_brewer("", palette = "Set1", guide=F) +
  xlab("Variant tier") + ylab("Q score") +
  theme_bw()

save(file = "../figures/fig6.Rda", fig6)
```

```
fig6
```

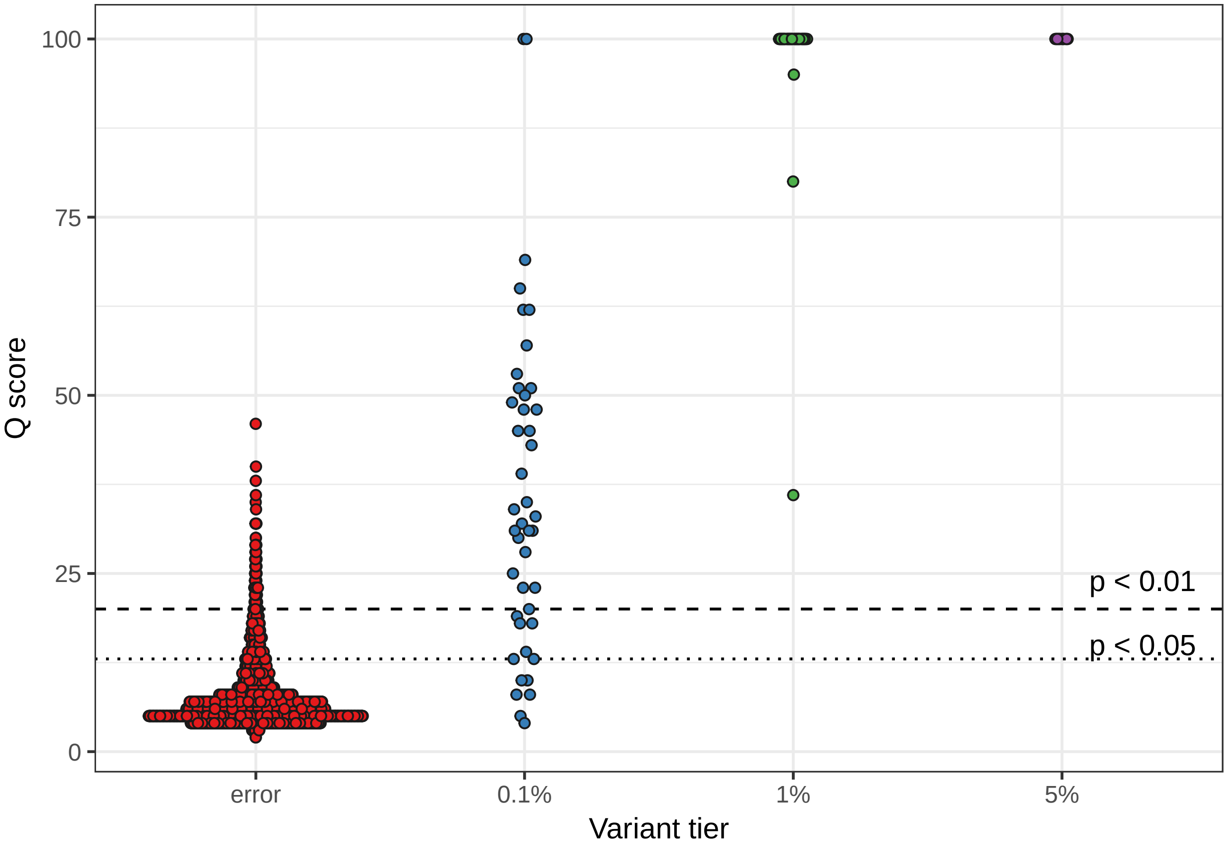

Figure 4: Comparison of Q score distribution of erroneous variants and variants of each frequency tier. Dotted and dashed lines show P < 0.05 and P < 0.01 thresholds respectively.

ROC curve for rare variants

```
df.1 <- subset(df, tier %in% c("error", "0.1%"))
df.1$type <- ifelse(df.1$tier == "error", 0, 1)

make_roc <- function(rocobj, type) {
  .df.roc <- data.frame(spec = seq(0, 1, 0.01))
  sens.ci <- ci.se(rocobj, specificities = .df.roc$spec)

  .df.roc$sens.lo <- sens.ci[,1]
  .df.roc$sens.me <- sens.ci[,2]
  .df.roc$sens.hi <- sens.ci[,3]
  .df.roc$type <- type

  .df.roc
}

rocobj <- roc(type ~ qual, df.1, ci=T)
print(rocobj)
```

```
##
## Call:
## roc.formula(formula = type ~ qual, data = df.1, ci = T)
##
## Data: qual in 2468 controls (type 0) < 43 cases (type 1).
## Area under the curve: 0.9277
## 95% CI: 0.8727-0.9827 (DeLong)
```

```
df.roc <- make_roc(rocobj, "Q score")
rocobj <- roc(type ~ frequency, df.1, ci=T)
print(rocobj)
```

```
##
## Call:
## roc.formula(formula = type ~ frequency, data = df.1, ci = T)
##
## Data: frequency in 2468 controls (type 0) < 43 cases (type 1).
## Area under the curve: 0.8637
## 95% CI: 0.7839-0.9435 (DeLong)
```

```
df.roc <- rbind(df.roc, make_roc(rocobj, "Frequency"))

fig7 <- ggplot(df.roc, aes(x=spec)) +
  geom_ribbon(aes(ymin=sens.lo, ymax=sens.hi, group=type), fill="grey", alpha=0.5) +
  geom_abline(slope = 1, intercept = 1, linetype = "dashed") +
  geom_line(aes(y=sens.me, color = type)) +
  scale_x_reverse("Specificity") +
  scale_y_continuous("Sensitivity", limits=c(0,1)) +
  scale_color_brewer("Threshold", palette = "Set1") +
  theme_bw()

save(file = "../figures/fig7.Rda", fig7)
```

```
fig7
```

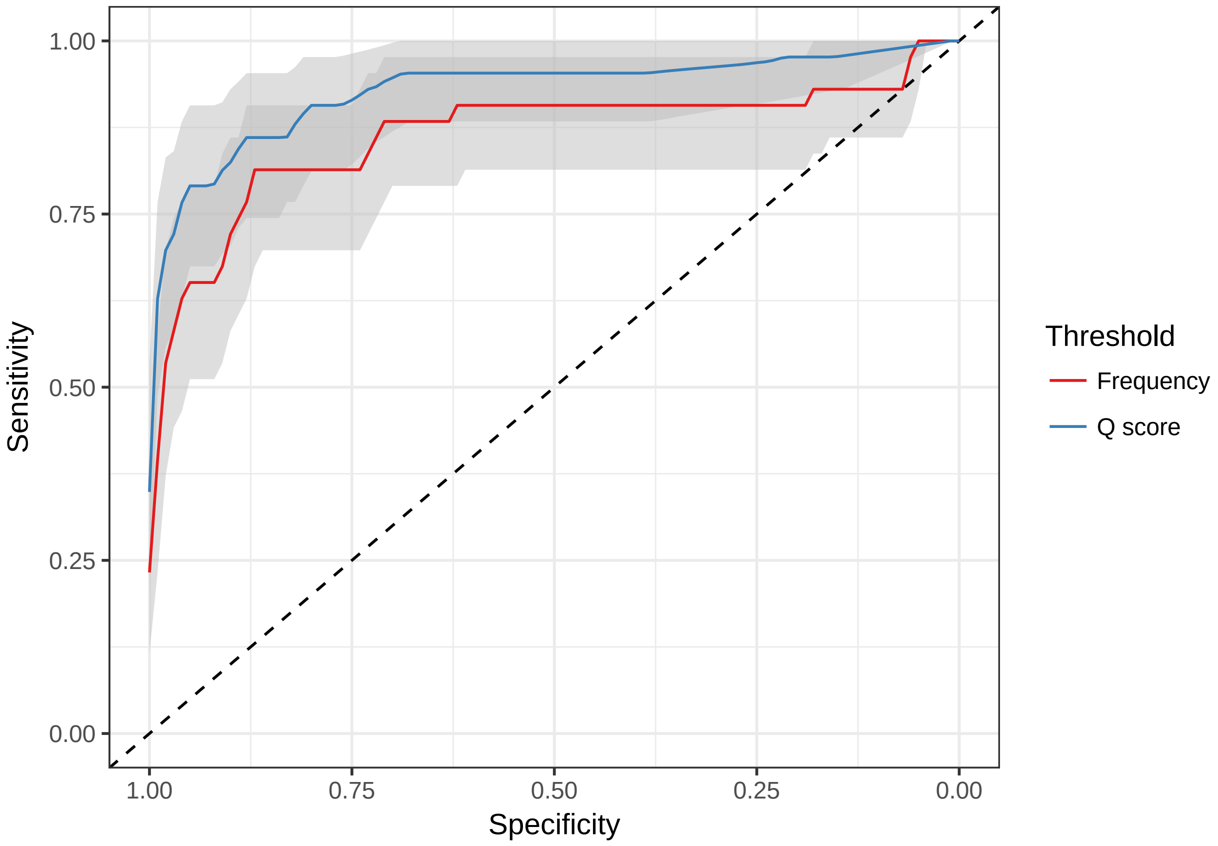

Figure 5: Receiver operation characteristic (ROC) curve comparing the sensitivity and specificity of MAGERI Q scores (blue line) and frequency-based thresholding (red line) in the task of classification of errors and 0.1% tier variants.