R Notebook

```r
suppressMessages(library("tidyverse"))
library(caret)
library(stringi)
library(ggplot2)
library(plotROC)
```

```r
variants=read_tsv("illumina_variants.tsv")
```

```
## Parsed with column specification:
## cols(
##   Name = col_character(),
##   Minimum = col_integer(),
##   Maximum = col_double(),
##   Length = col_integer(),
##   Change = col_character(),
##   Coverage = col_double(),
##   `Polymorphism Type` = col_character(),
##   `Variant Frequency` = col_character(),
##   replica = col_character(),
##   modality = col_character(),
##   freq = col_double()
## )
```

```r
barcode1v=read_tsv("BC01.variants.freqs.txt")
```

```
## Parsed with column specification:
## cols(
##   Pos = col_integer(),
##   Qual = col_integer(),
##   Freq = col_double(),
##   Ref = col_character(),
##   Base = col_character(),
##   UngappedCoverage = col_integer(),
##   TotalCoverage = col_integer()
## )
```

```r
barcode1v$replica = 'a'
barcode2v=read_tsv("BC02.variants.freqs.txt")
```

```
## Parsed with column specification:
## cols(
##   Pos = col_integer(),
##   Qual = col_integer(),
##   Freq = col_double(),
##   Ref = col_character(),
##   Base = col_character(),
##   UngappedCoverage = col_integer(),
##   TotalCoverage = col_integer()
## )
```

```r
barcode2v$replica = 'b'
barcode3v=read_tsv("BC03.variants.freqs.txt")
```

```
## Parsed with column specification:
## cols(
##   Pos = col_integer(),
##   Qual = col_integer(),
##   Freq = col_double(),
##   Ref = col_character(),
##   Base = col_character(),
##   UngappedCoverage = col_integer(),
##   TotalCoverage = col_integer()
## )
```

```
barcode3v$replica = 'c'
minion_variants=rbind(barcode1v, barcode2v, barcode3v)
```

```
minion_variants %>%
    filter(Qual == 0) %>%
    write_tsv(path="minion_variants.tsv")
```

```
barcode1=read_tsv("BC01.freqs.txt")
```

```
## Parsed with column specification:
## cols(
##   Pos = col_integer(),
##   Qual = col_integer(),
##   Freq = col_double(),
##   Ref = col_character(),
##   Base = col_character(),
##   UngappedCoverage = col_integer(),
##   TotalCoverage = col_integer()
## )
```

```
barcode1$replica = 'a'
barcode2=read_tsv("BC02.freqs.txt")
```

```
## Parsed with column specification:
## cols(
##   Pos = col_integer(),
##   Qual = col_integer(),
##   Freq = col_double(),
##   Ref = col_character(),
##   Base = col_character(),
##   UngappedCoverage = col_integer(),
##   TotalCoverage = col_integer()
## )
```

```
barcode2$replica = 'b'
barcode3=read_tsv("BC03.freqs.txt")
```

```
## Parsed with column specification:
## cols(
##   Pos = col_integer(),
##   Qual = col_integer(),
##   Freq = col_double(),
##   Ref = col_character(),
##   Base = col_character(),
##   UngappedCoverage = col_integer(),
##   TotalCoverage = col_integer()
## )
```

```r
barcode3$replica = 'c'
minion_all=rbind(barcode1, barcode2, barcode3)
```

```r
minion_all %>%
    filter(Qual == 0) %>%
    write_tsv(path="minion_wt_frequencies.tsv")
```

```r
expectedpositions=read_tsv("expectedpositions.txt")
```

```
## Parsed with column specification:
## cols(
##   Position = col_integer(),
##   State = col_character()
## )
```

```r
barcode1snps=read_tsv("BC01.variants.0.03.txt")
```

```
## Parsed with column specification:
## cols(
##   Pos = col_integer(),
##   Qual = col_integer(),
##   Freq = col_double(),
##   Ref = col_character(),
##   Base = col_character(),
##   UngappedCoverage = col_integer(),
##   TotalCoverage = col_integer(),
##   VariantCov = col_integer(),
##   ForwardVariantCov = col_integer(),
##   ReverseVariantCov = col_integer()
## )
```

```r
barcode1snps$replica = 'a'
barcode2snps=read_tsv("BC02.variants.0.03.txt")
```

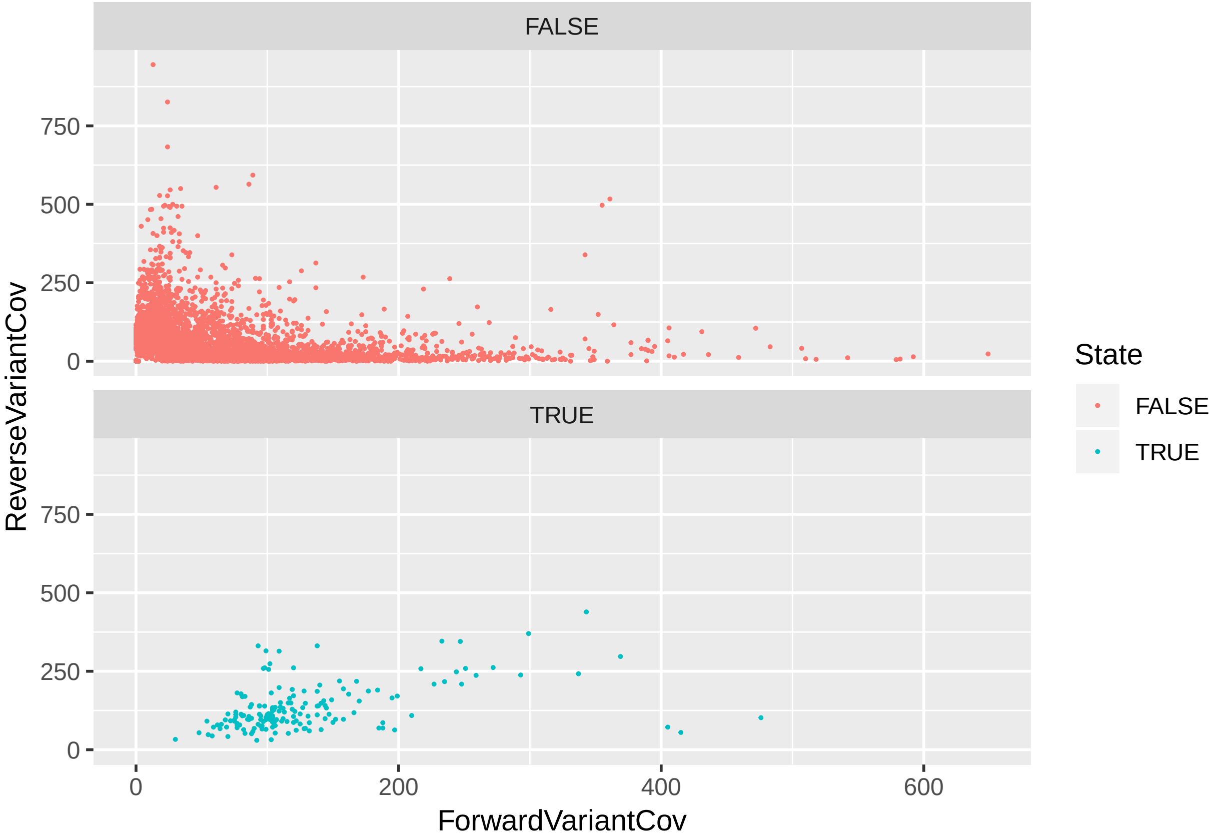
```
## Parsed with column specification:
## cols(
##   Pos = col_integer(),
##   Qual = col_integer(),
##   Freq = col_double(),
##   Ref = col_character(),
##   Base = col_character(),
##   UngappedCoverage = col_integer(),
##   TotalCoverage = col_integer(),
##   VariantCov = col_integer(),
##   ForwardVariantCov = col_integer(),
##   ReverseVariantCov = col_integer()
## )
```

```r
barcode2snps$replica = 'b'
barcode3snps=read_tsv("BC03.variants.0.03.txt")
```

```
## Parsed with column specification:
## cols(
##   Pos = col_integer(),
##   Qual = col_integer(),
##   Freq = col_double(),
##   Ref = col_character(),
##   Base = col_character(),
##   UngappedCoverage = col_integer(),
##   TotalCoverage = col_integer(),
##   VariantCov = col_integer(),
##   ForwardVariantCov = col_integer(),
##   ReverseVariantCov = col_integer()
## )
```

```
barcode3snps$replica = 'c'
minion_all_variants=rbind(barcode1snps, barcode2snps, barcode3snps)
minion_all_variants_positions=minion_all_variants %>%
   left_join(expectedpositions, by=c("Pos" = "Position")) %>%
   filter(State != 'Remove')
```

```
minion_all_variants_positions %>%
  ggplot(aes(x=ForwardVariantCov, y=ReverseVariantCov, color=State)) + geom_point(size=0.2) + facet_wrap
```



```
minion_all_variants_positions %>%
  mutate(StrandAF = pmin(ForwardVariantCov, ReverseVariantCov) / pmax(ForwardVariantCov, ReverseVariant
```

```
## # A tibble: 4,850 x 13
##      Pos  Qual   Freq Ref   Base  UngappedCoverage TotalCoverage
##    <int> <int>  <dbl> <chr> <chr>            <int>         <int>
##  1  1063     0 0.0370 G     T                 2135          2164
##  2  1064     0 0.0426 T     C                 2115          2164
##  3  1066     0 0.0437 G     A                 2104          2164
##  4  1067     0 0.0342 T     A                 2132          2164
##  5  1069     0 0.0402 T     C                  2140          2164
##  6  1070     0 0.0922 C     T                  2083          2164
##  7  1074     0 0.0598 G     A                  2072          2164
##  8  1077     0 0.0456 A     G                  2125          2164
##  9  1078     0 0.0362 T     C                  2126          2164
## 10  1079     0 0.0460 G     A                  2063          2164
## # ... with 4,840 more rows, and 6 more variables: VariantCov <int>,
## #   ForwardVariantCov <int>, ReverseVariantCov <int>, replica <chr>,
## #   State <chr>, StrandAF <dbl>
```

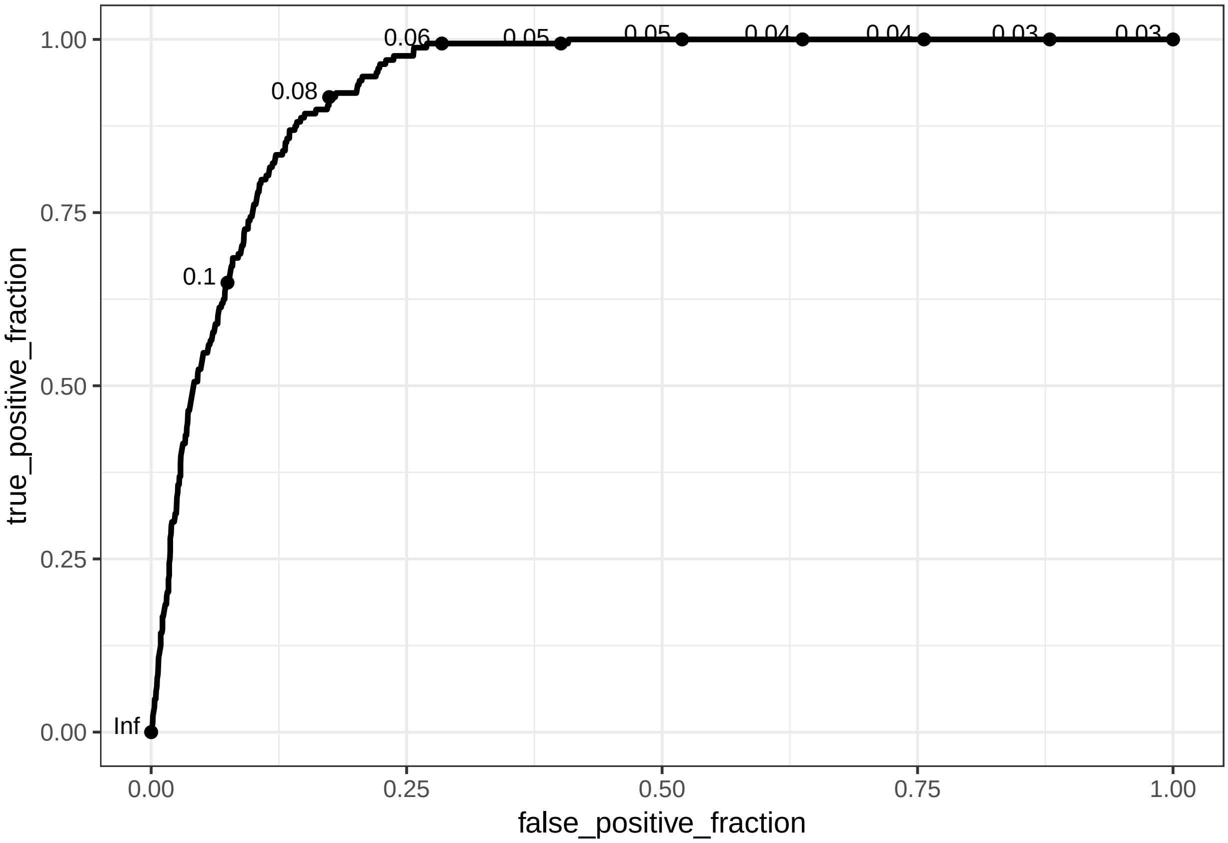
```
minion_all_variants_positions %>%
       group_by(State) %>%
       summarise(n=n())
```

```
## # A tibble: 2 x 2
##   State     n
##   <chr> <int>
## 1 FALSE  4682
## 2 TRUE    168
```

```
minion_all_variants_positions %>%
  mutate(StrandAF = pmin(ForwardVariantCov, ReverseVariantCov) / pmax(ForwardVariantCov, ReverseVariant
  filter(ForwardVariantCov > 10) %>%
  filter(ReverseVariantCov > 10) %>%
  write_tsv("minion_variants_3pc_all.tsv")
```

```
forroc = minion_all_variants_positions %>%
  mutate(StrandAF = pmin(ForwardVariantCov, ReverseVariantCov) / pmax(ForwardVariantCov, ReverseVariant
  filter(ForwardVariantCov > 10) %>%
  filter(ReverseVariantCov > 10) %>%
  mutate(D = ifelse(grepl("TRUE", State), 1, 0))
```

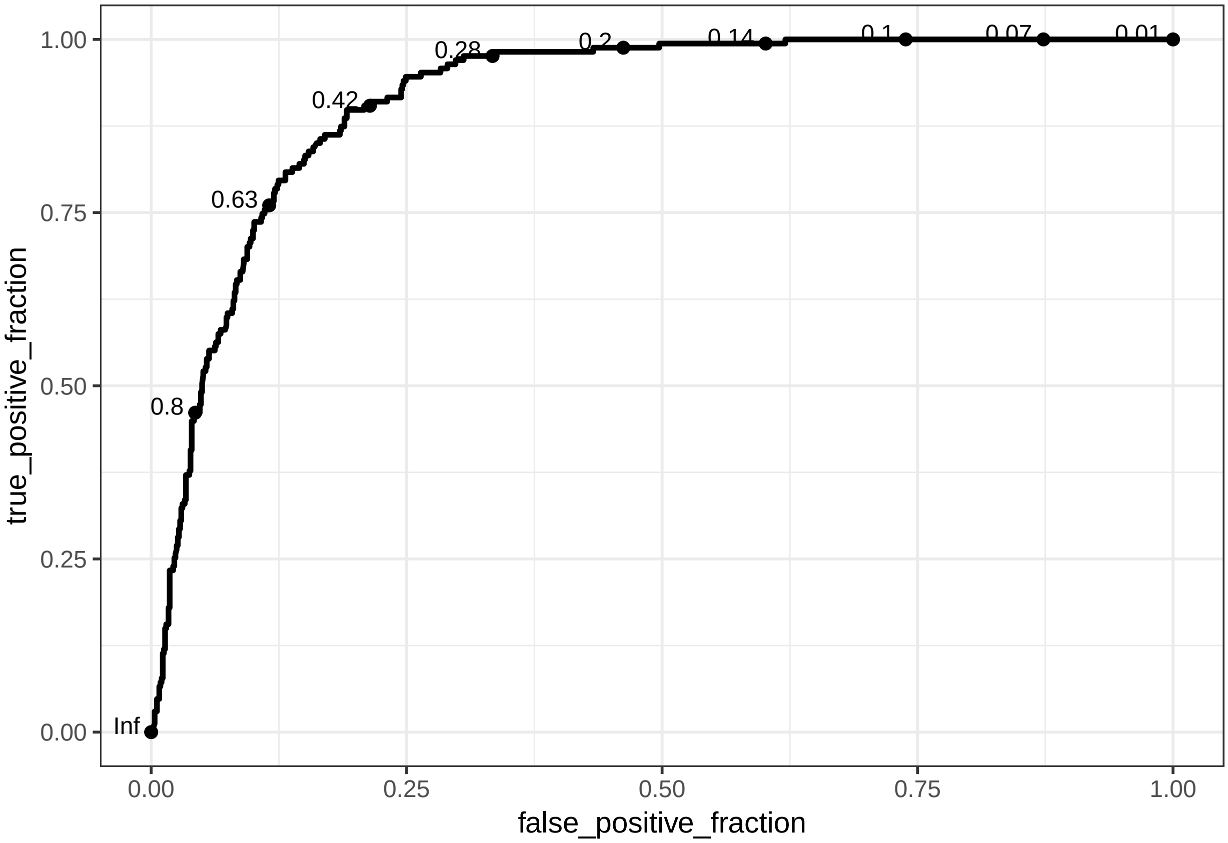
```
ggplot(forroc, aes(d = D, m = Freq)) + geom_roc(labelsize=3, labelround=2) + theme_bw()
```



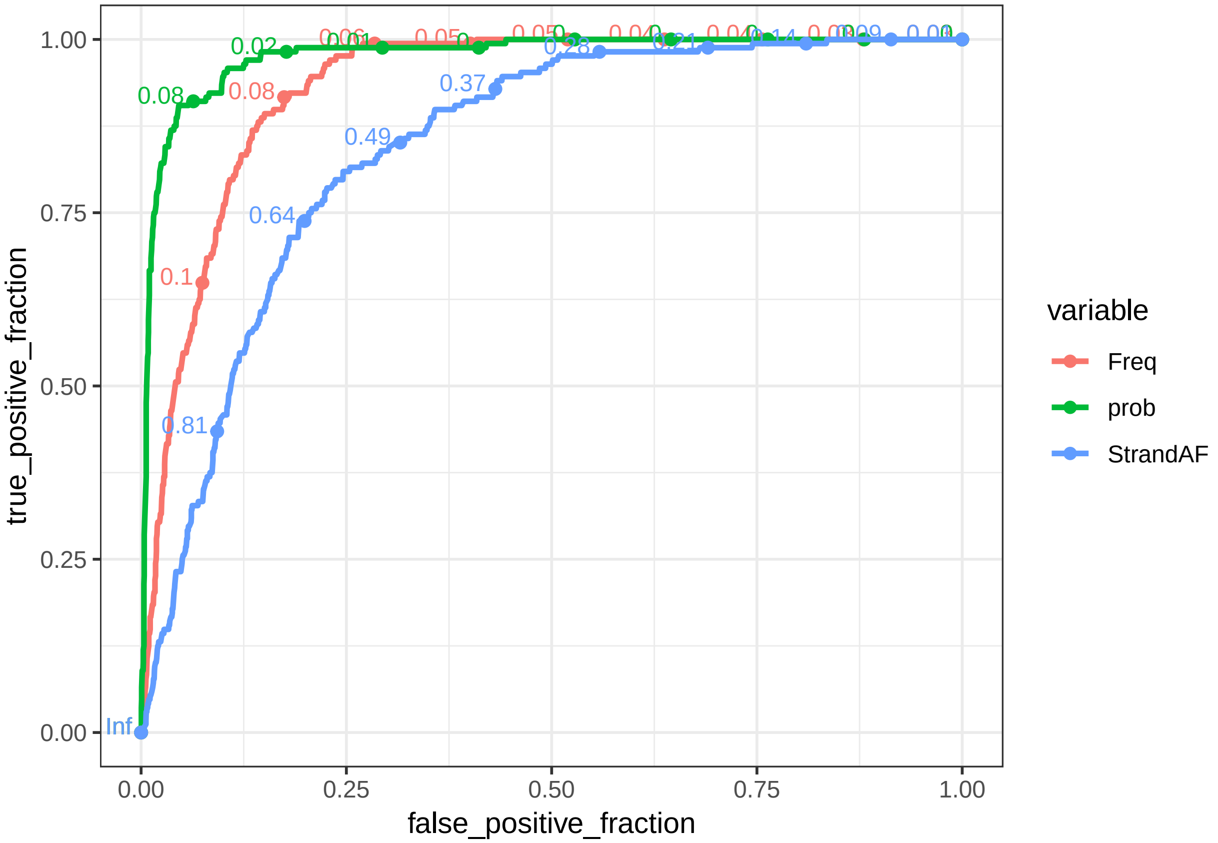
```
nrow(forroc %>% filter(Freq > 0.06))
```

```
## [1] 1050
```

```
ggplot(forroc %>% filter(Freq > 0.06), aes(d = D, m = StrandAF)) + geom_roc(labelsize=3, labelround=2) +
```



```
## mod1<-glm(D ~ Freq + StrandAF, data=forroc, family="gaussian")
## prob=predict(mod1,type=c("response"))
## forroc$prob = prob
library(caret)
forroc.df <- data.frame(forroc)
forroc.df$D <- sapply(forroc.df$State,function(x){ifelse(grepl("TRUE", x), "true", "false")})
set.seed(112358)
fitControl <- trainControl(
    method = "repeatedcv",
    number = 10,
    repeats = 10,
    classProbs = T,
    savePredictions =T)
mod1 <- train(D ~ Freq + StrandAF, data = forroc.df,
                method = "glm",
                family=binomial(),
              trControl = fitControl)
forroc.df$prob <- predict(mod1, data=forroc.df, type=c("prob"))
forroc$prob <- forroc.df$prob[,"true"]
melted=gather(forroc, variable, value, Freq, StrandAF, prob)
ggplot(melted, aes(d = D, color = variable, m = value)) + geom_roc(labelsize=3, labelround=2) + theme_b
```



```
forroc %>%
  filter(prob > 0.2) %>%
  group_by(D) %>%
  write_tsv("minion_roc_classifier_snps.tsv")
```

```
forroc %>%
  filter(prob > 0.2) %>%
  group_by(D) %>%
  summarise(n=n())
```

```
## # A tibble: 2 x 2
##       D     n
##   <dbl> <int>
## 1    0.    71
## 2    1.   137
```

```
#sens=tp/(tp+fn)
sens=160/(160+8)
print(sens)
```

```
## [1] 0.952381
```

```
#spec=tn/(tn+fp)
spec=8/(8+157)
print(spec)
```

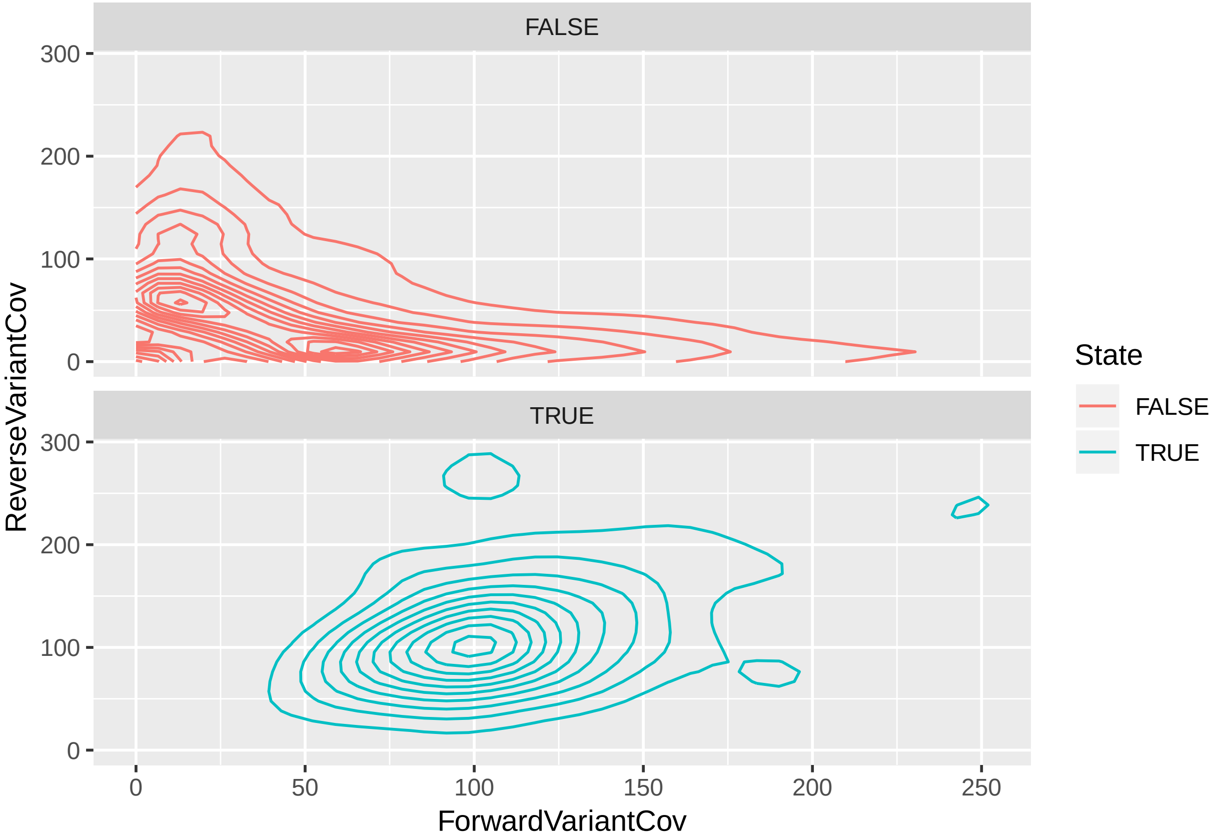
```
## [1] 0.04848485
```

```
fdr=157/(160+157)
print(fdr)
```

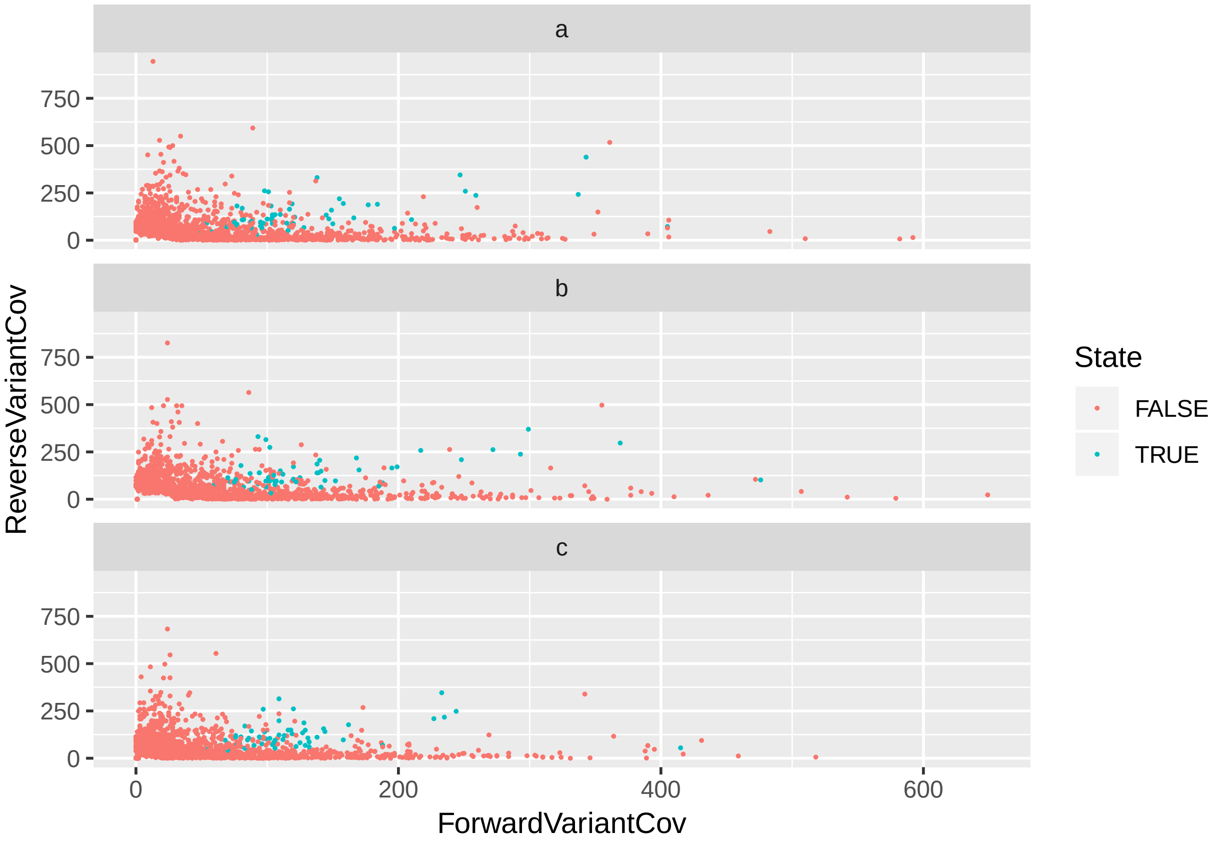
```
## [1] 0.4952681
```

```
minion_all_variants_positions %>%
  mutate(StrandAF = pmin(ForwardVariantCov, ReverseVariantCov) / pmax(ForwardVariantCov, ReverseVariant(
  filter(ForwardVariantCov > 10) %>%
  filter(ReverseVariantCov > 10) %>%
  filter(StrandAF > 0.70) %>%
  write_tsv("minion_variants_3pc_0.7strandaf.tsv")
```

```
minion_all_variants_positions %>%
  ggplot(aes(x=ForwardVariantCov, y=ReverseVariantCov, color=State)) + geom_density2d() + facet_wrap(~St
```



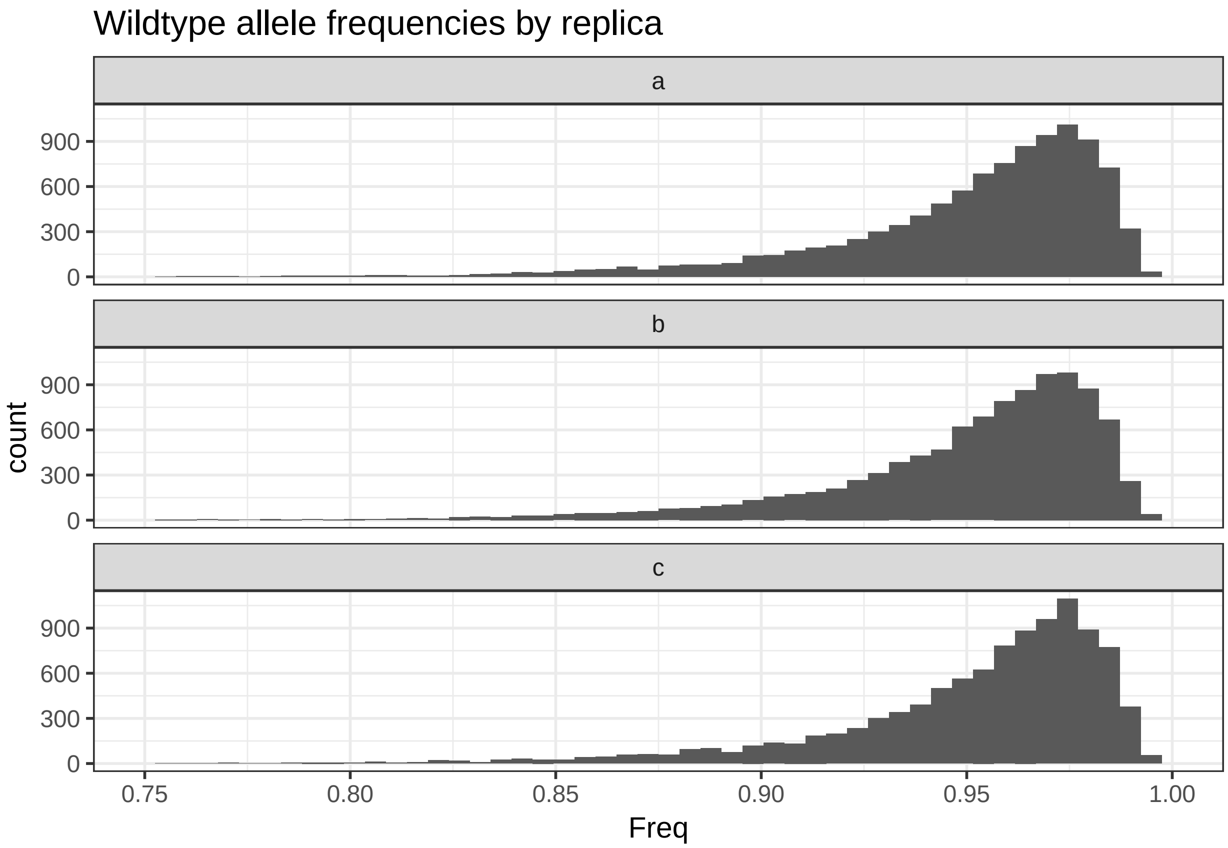
```
minion_all_variants_positions %>%
  ggplot(aes(x=ForwardVariantCov, y=ReverseVariantCov, color=State)) + geom_point(size=0.2) + facet_wrap
```



```
variant_positions=inner_join(minion_all, expectedpositions, by=c("Pos" = "Position"), copy=T)
wildtype_positions=anti_join(minion_all, expectedpositions %>% filter(State == "True"), by=c("Pos" = "Po
```

```
wildtype_positions %>%
  filter(UngappedCoverage >= 50) %>%
  ggplot(aes(x=Freq)) + geom_histogram(bins=50) + facet_wrap(~replica, ncol=1) + theme_bw() + xlim(0.75
```

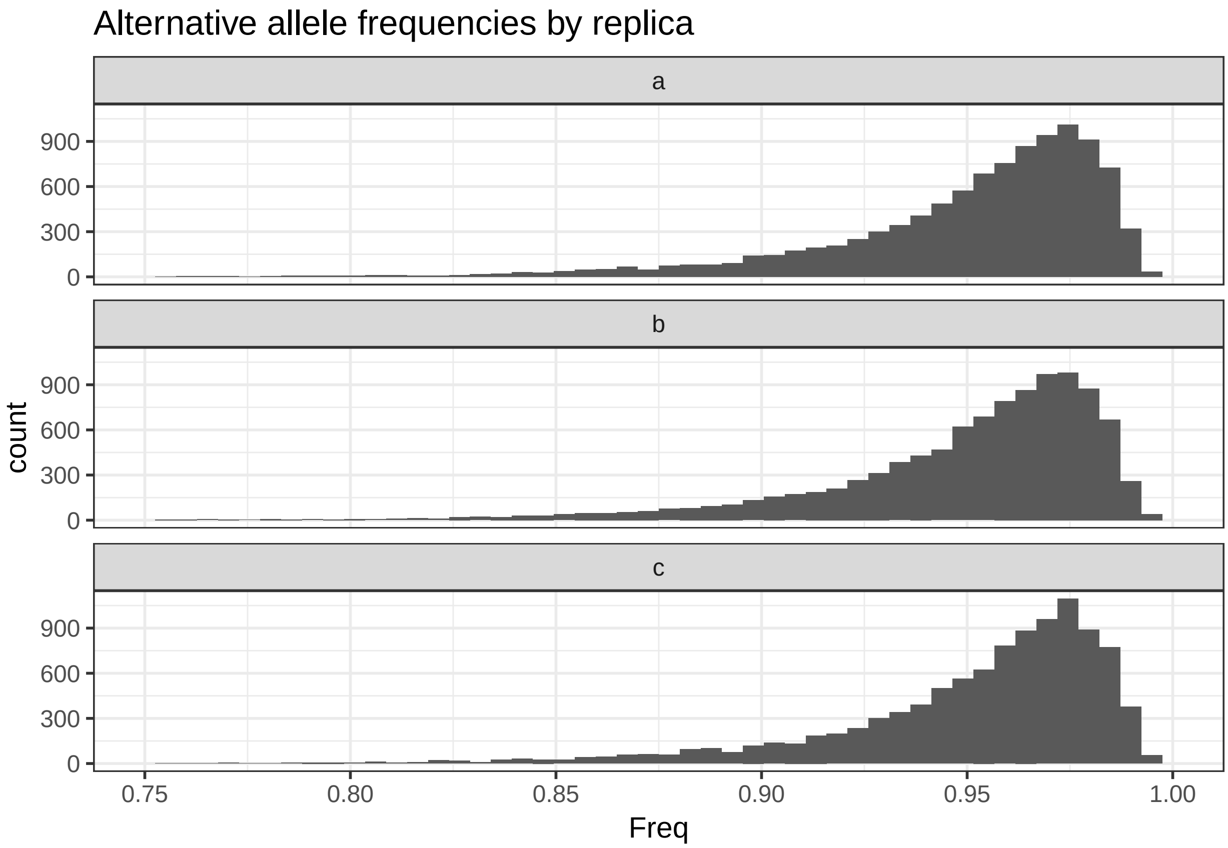
```
## Warning: Removed 101 rows containing non-finite values (stat_bin).

## Warning: Removed 3 rows containing missing values (geom_bar).
```



```
variant_positions %>%
  filter(UngappedCoverage >= 50) %>%
  ggplot(aes(x=Freq)) + geom_histogram(bins=50) + facet_wrap(~replica, ncol=1) + theme_bw() + xlim(0.75
```

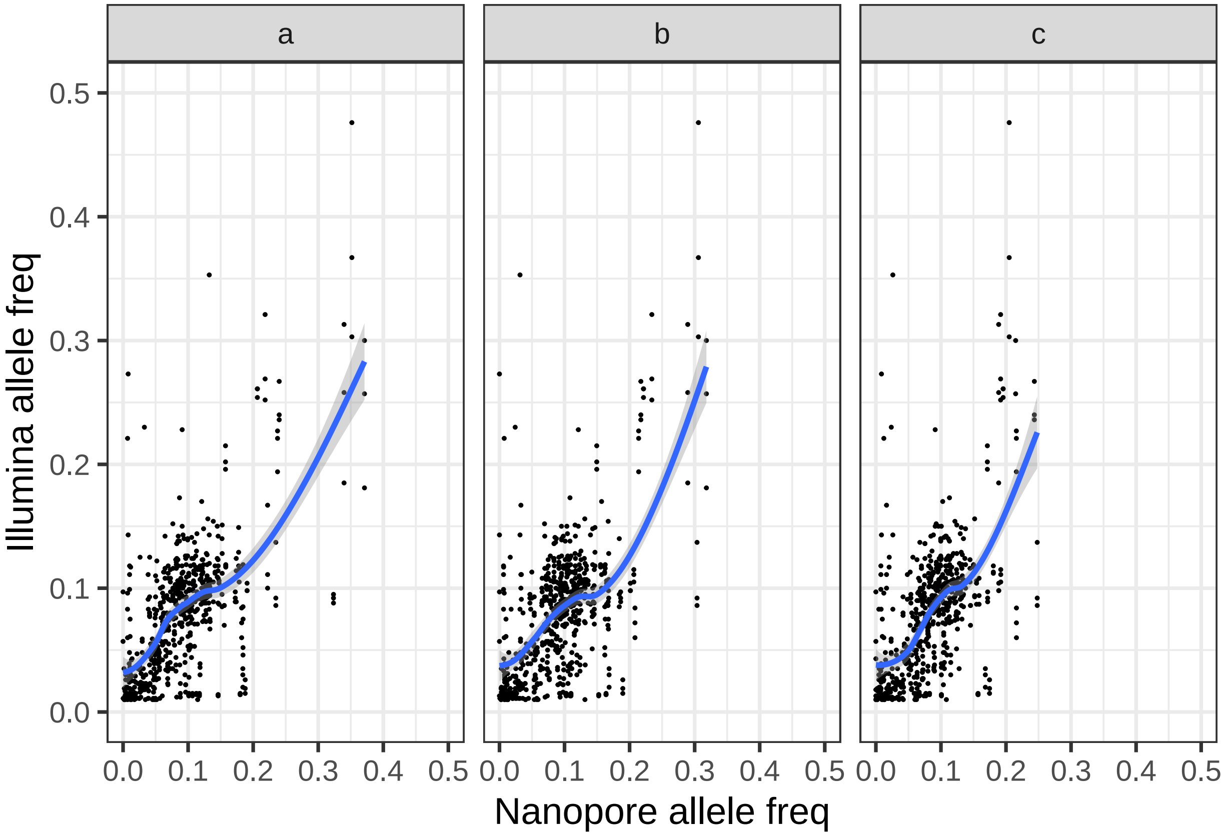
```
## Warning: Removed 101 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 3 rows containing missing values (geom_bar).
```



```
joined=inner_join(minion_variants, variants, by=c("Pos" = "Minimum"), copy=T)
p = joined %>%
  filter(Qual == 0) %>%
  filter(modality == 'amplicon') %>%
  ggplot(aes(x=Freq, y=freq)) + geom_point(size=0.2) + stat_smooth() + xlim(0, 0.5) + ylim(0, 0.5) + xla
p
```

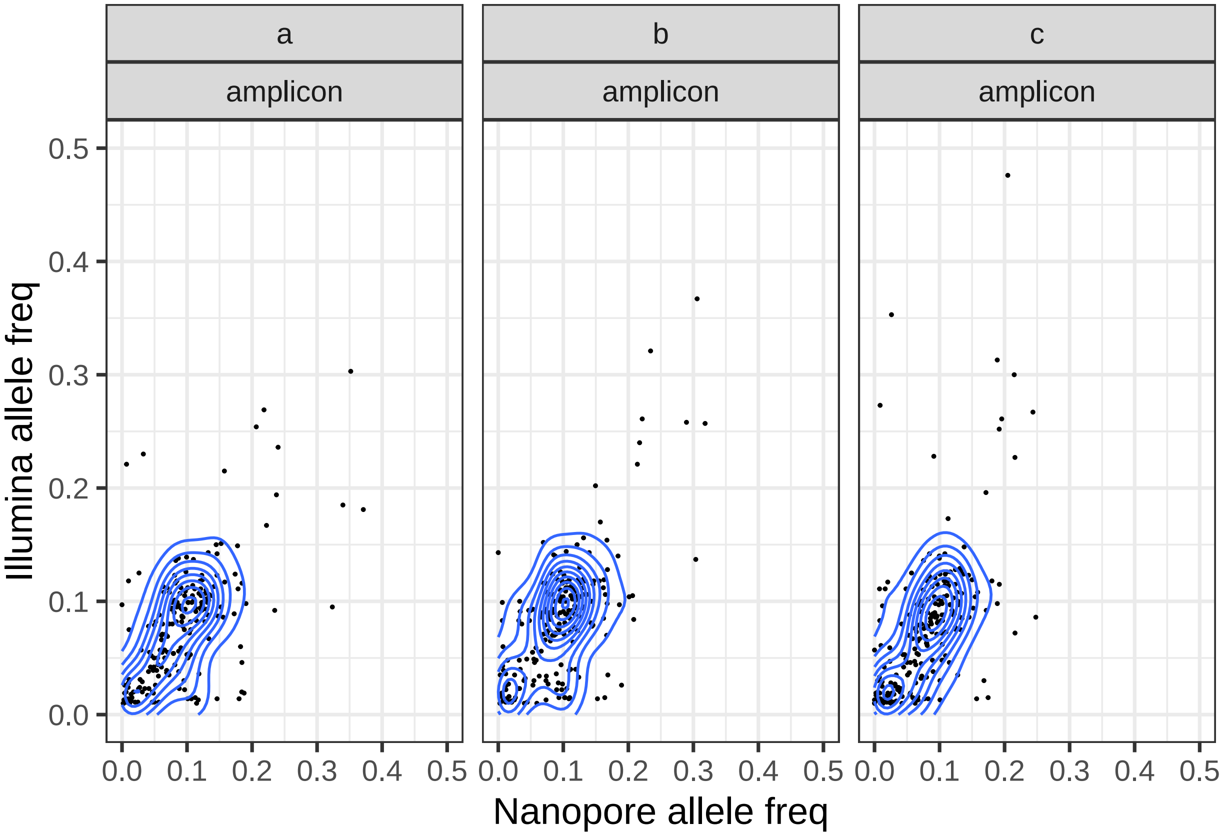
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

## Warning: Removed 12 rows containing non-finite values (stat_smooth).

## Warning: Removed 12 rows containing missing values (geom_point).
```



```
a=joined %>%
  filter(modality == 'amplicon') %>%
  filter(replica.x == replica.y) %>%
  ggplot(aes(x=Freq, y=freq)) + geom_point(size=0.2) + geom_density2d() + xlim(0, 0.5) + ylim(0, 0.5) +
a
```

```
## Warning: Removed 4 rows containing non-finite values (stat_density2d).
```

```
## Warning: Removed 4 rows containing missing values (geom_point).
```


a
amplicon
b
amplicon
c
amplicon
Illumina allele freq
0.5
0.4
0.3
0.2
0.1
0.0
0.0 0.1 0.2 0.3 0.4 0.5
0.0 0.1 0.2 0.3 0.4 0.5
0.0 0.1 0.2 0.3 0.4 0.5
Nanopore allele freq