# Weighted expected binomial model

Anthony Aylward

2018-07-30

## Prepare the data.

```
library(chenimbalance)
total_reads <- rowSums(accb[, c("cA", "cC", "cG", "cT")])
data <- data.frame(
  total = total_reads,
  allelicRatio = sapply(
    1:nrow(accb),
    function(i) {
      accb[[paste("c", accb[["ref"]][[i]], sep = "")]][[i]] / total_reads[[i]]
    }
  )
)
head(data)
#>   total allelicRatio
#> 1    45    1.0000000
#> 2    59    0.5423729
#> 3   114    0.4736842
#> 4    53    0.5094340
#> 5   119    0.5042017
#> 6    21    0.0952381
```

## Binomial distribution

```
# graded weights for SSE calculation
binSize <- 40
bins <- pretty(0:1, binSize)
w.grad <- graded_weights_for_sse_calculation(r_min = 0, r_max = 1, bins = bins)

# empirical allelic Ratio
minN <- 6
maxN <- min(2500, max(data[["total"]]))
apropor <- length(data[["total"]][data[["total"]] <= 2500]) / nrow(data)
```

```
empirical <- empirical_allelic_ratio(
  data,
  bins,
  maxN = maxN,
  minN = minN,
  plot = TRUE
)
```

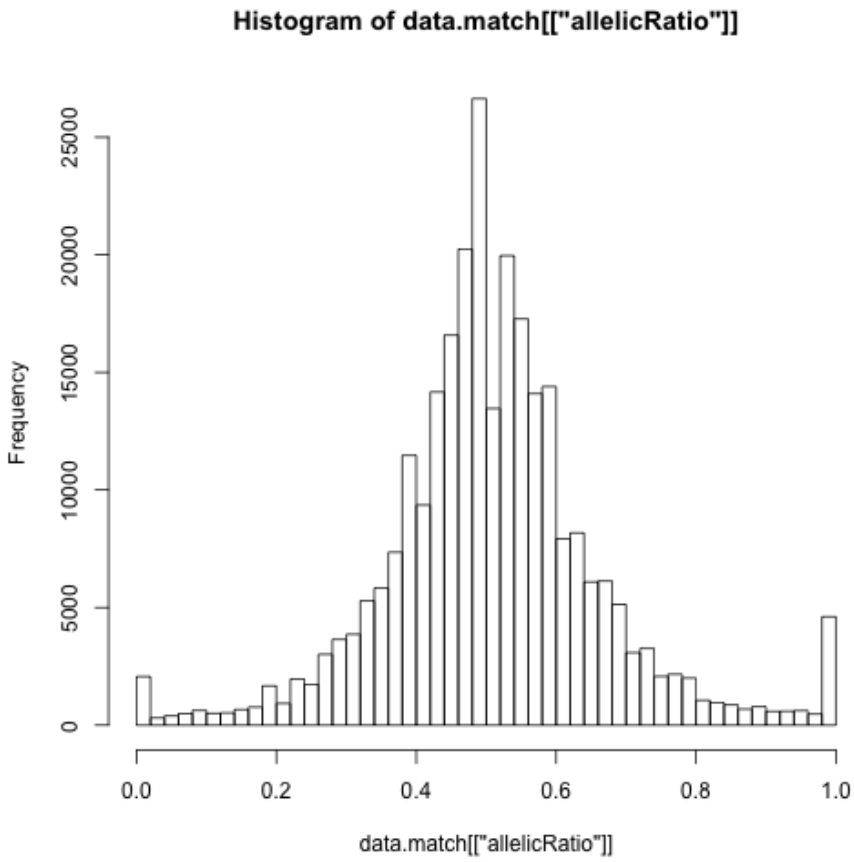


Figure 1: plot of chunk unnamed-chunk-2

Plot the empirical and weighted expected binomial distributions

```
d.combined.sorted.binned <- weighted_expected_binomial(
  data[["total"]],
  minN = minN,
  binSize = binSize
)
```

```
yuplimit <- 0.15
barplot(
  empirical,
  ylim = c(0, yuplimit),
  ylab = "density",
  xlab = "allelicRatio",
  names.arg = bins[2:length(bins)] - bins[[2]] / 2,
  main = paste("n=", minN, "-", maxN)
)
par(new = TRUE)
plot(
  d.combined.sorted.binned,
  ylim = c(0, yuplimit),
  pch=16,
  type='b',
  col='red',
  bty='n',
  ylab='',
  xlab='',
  yaxt='n',
  xaxt='n',
  yaxs="i"
)
```

Compute the sum of squared errors for the binomial distribution.

```
sse = sum((empirical - d.combined.sorted.binned[,2])^2)
sse
#> [1] 0.005153999
```

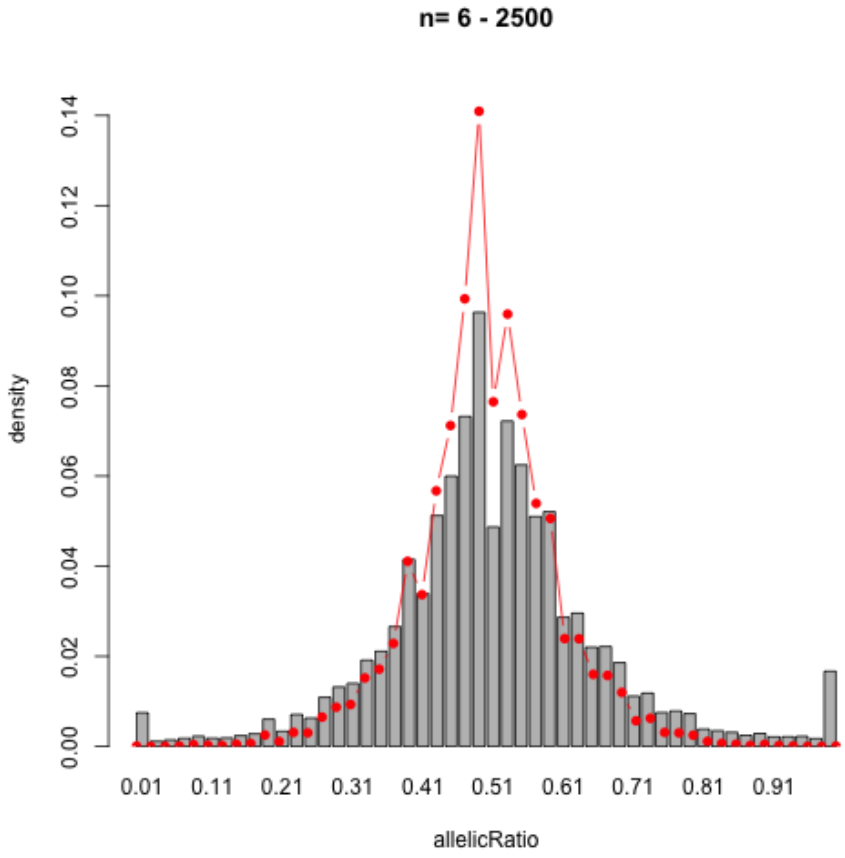


Figure 2: plot of chunk unnamed-chunk-3