A Practice of General Form of Luria-Delbrück Distribution

System Dynamics in Biology Workshop: Exercise Day 2

Speaker: Professor Arthur Lander

Course material and exercise are originally in Wolfram Language. This is a practice to get some hands-on experience in R.

> You are working on a line of tumor cells in cell culture. At equilibrium, **95%** of the cells express EGFR and **5%** do not. If the cells are switching back and forth between two stable states, the model could be formulated as:
>
> $$[EGFR^+] \underset{q}{\overset{p}{\rightleftharpoons}} [EGFR^-]$$
>
> Assuming the culture is in equilibrium, what can we say about the ratio of p and q?

Let the culture in equilibrium, from the laws of mass action, we could assume:

$$\frac{d[EGFR^+]}{dt} = p[EGFR^+] = 0 = q[EGFR^-] = \frac{d[EGFR^-]}{dt}$$

and as a result:

$$\frac{p}{q} = \frac{[EGFR^-]}{[EGFR^+]} = \frac{5}{95}$$

> In order to find out the values of p and q, I could have set up a complicated time-lapse experiment. On the other hand, your friend suggest that you could set up a culture starting from single cell for 12 dividing times, repeat for 500 times, and check the variance-mean ratio of $EGFR^+$ cell numbers. The friend suggests that this would also give the value of p and q.

The question could be further boiled down to: **Given that we know the relationship between p and q, whether a specific p corresponds to a specific variance-mean ratio in this experiment?**

To understand that, we set up a simulation of this experiment, and see if there's such a correlation between p and variance-mean ratio.

First, define a function that do:

1. Double the cell number
2. Determine how many cells changed their state of EGFR in the cycle of division
3. Repeat the dividing and transformation for n times (in this case 12)

```
divide <- function(a, b, p){
  # Parameters
  # a: initial EGFR+ number
  # b: initial EGFR- number
  # p: the transformation chance from EGFR+ to EGFR-
```

```
  ## Given the known relation ship between p and q (q = 19p), we need to check whether q > 1
  ## before we start
  if (p * 19 > 1) {
    stop("Your p should not exceed 0.0526, or q would exceed 1\n(Note p and q are probabilities.)")
  }

  # Assign local variables first to make for-loop easier
  this_a <- a
  this_b <- b

  for (division in seq(12)) {
    # First the cells divide
    this_a <- 2 * this_a
    this_b <- 2 * this_b


    # Second the cells decide whether they transform
    # The occurance of transformation should obey the Poisson distribution
    ## Besides that, we need a way to prevent rpois() from drawing more than what we
    ## have (this_a < a_to_b)
    a_to_b <- min(this_a, rpois(n = 1, lambda = this_a * p))
    b_to_a <- min(this_b, rpois(n = 1, lambda = this_b * p * 19))


    # Count the final number of cells in this loop
    this_a <- this_a - a_to_b + b_to_a
    this_b <- this_b + a_to_b - b_to_a
  }
  return(c(this_a, this_b))
}
```

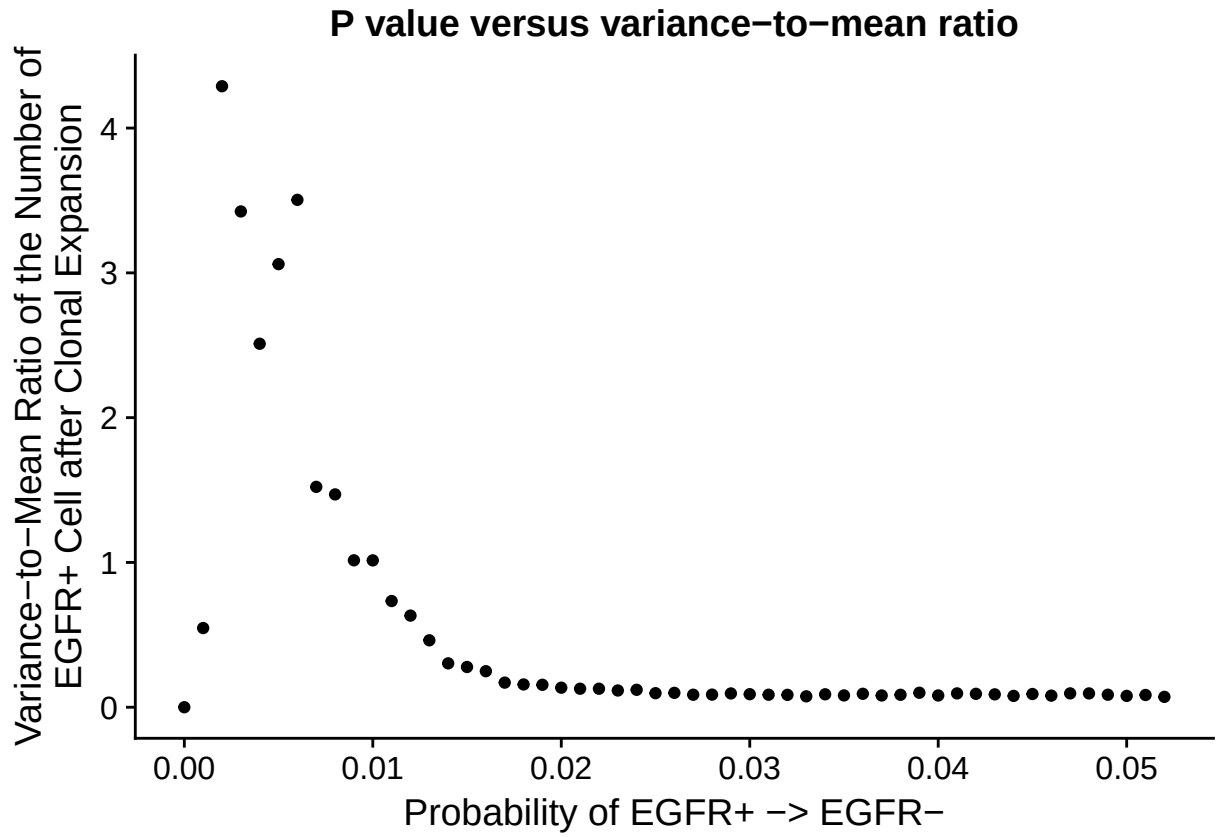
Then, we run this function 500 times with different value of p.

```
# Note that q = 19 * p
# so p must stay within [0,1/19)
## Generate a numerical vector from 0 to 1/19 stepping 0.001
p_range <- seq(from = 0, to = 1/19, by = 0.001)

# Pre-allocate a matrix to store calculated variance-mean ratio for each p
result <- matrix(ncol = 4, nrow = length(p_range))
colnames(result) <- c("p", "mean_cells", "variance_cells", "variance_mean_ratio")

# Run division() 500 times for each p
for (p in p_range) {
  rep_500 <- sapply(seq(500), function(x) divide(1,0,p))
  mean_pos <- mean(rep_500[1,]) # Count the EGFR+ cells
  var_pos <- var(rep_500[1,]) # Count the EGFR+ cells
  # Save the results and leave space for later calculation of VMR
  result[which(p_range == p), ] <- c(p, mean_pos, var_pos, NA)
}

# Calculating variance over mean
result[,4] <- result[,3]/result[,2]
```

p	mean_cells	variance_cells	variance_mean_ratio
0.000	4096.00	0.00	0.000
0.001	4059.13	2219.67	0.547
0.002	4015.86	17224.37	4.289
0.003	3994.15	13676.67	3.424
0.004	3974.21	9977.17	2.510
0.005	3953.55	12098.77	3.060

This would give us some idea about whether a particular p corresponds to a specific variance-mean ratio in this experiment...

```
# Turn the result matrix into data frame for ggplot2
result_df <- as.data.frame(result)
vmr_plot <- ggplot(result_df, aes(x = p, y = variance_mean_ratio)) + geom_point() +
  ggtitle("P value versus variance-to-mean ratio") +
  labs(x = "Probability of EGFR+ -> EGFR-",
       y = "Variance-to-Mean Ratio of the Number of\nEGFR+ Cell after Clonal Expansion")
plot(vmr_plot)
```



With the increase of p, the variance-to-mean ratio first surges up and then gradually weans down. It is intuitive to imagine when $p \approx 0$, very little transformation happens, and thus most of the population stays homogeneous with a low dispersion. On the other hand, when the transformation is sparse, whether transformation happening at the early division would have a much larger impact, and p is not high enough to averge out this effect. Lastly, when p is high, transformation happens offen, and the cells are quickly entering equilibrium where the ratio of the rate constant dominates.

This simulation is actually a general version of Luria-Delbrück experiment. If we let q be zero, it is the spontaneous mutation condition in this classic experiment.

This iconic variability could also be applied to distinguish whether a phenotype is heritable (a state change with moderately low p) or transient (a state change with high p). For example, figure 1a in this article examplified the test.