Duke Microbiome Immunology Cancer (MIC) Course

Notes on Differential Expression Analysis using Seurat

Biostatistics and Bioinformatics

Summer 2022

Section 1

Introduction

Differential Expression Analysis in Seurat

- Today's tutorial will cover differential expression analysis using the Seurat `FindMarkers()`
- Its `test.use` argument allows for specification of different types of statistics
- The default is the Wilcoxon statistic (`test.use="wilcox"`
- The observed data from both bulk and single-cell RNA-Seq data are counts (not expression)
- We will provide a general overview of distributions for count data
- *Caveat Emptor*: Most if not all of these approaches suffer from a major flaw in that the clusters, aside from being artificial, were constructed using the same genes used for DA

Section 2

The two-sample Wilcoxon test

Simulating type I error of the two-sample t-test

Consider the function from Week 1 simulating the rejection probability when both samples are drawn from normal distribution assuming that they share the same standard deviation:

```
simttest1 <- function(n1, n2, mean1, mean2, stdev1, stdev2, alpha) {
    ## Simulate n1 observations from group 1 N(mu1,stdev1)
    y1 <- rnorm(n1, mean1, stdev1)
    ## Simulate n2 observations from group 2 N(mu2,stdev2)
    y2 <- rnorm(n2, mean2, stdev2)
    ## Perform two-sample unpaired t-test _assuming_ equal variance
    testresult <- t.test(y1, y2, paired = FALSE, var.equal = TRUE)
    ## Get P-value of test
    pvalue <- testresult$p.value
    ## Apply decision rule: Reject if pvalue < alpha
    reject <- ifelse(pvalue < alpha, TRUE, FALSE)
    ## Return decision
    return(reject)
}
```


Simulation Example: Type I error

Why are the empirical type I error rates above the nominal level in the following examples?

```
set.seed(51621)
B <- 10000L
mean(replicate(B, simttest1(3, 3, 1, 1, 0.5, 1, 0.05)))

## [1] 0.065
```

```
mean(replicate(B, simttest1(3, 3, 1, 1, 0.5, 2, 0.05)))

## [1] 0.0841
```

Answer: The version of the t-test used is not robust against heteroskedasticity (*i.e.*, when the standard deviation differs between the two groups)

Robustifying two-sample testing

- The two-sample t-test used in this example is not robust against heteroskedasticity and deviations from normality
- It is also not robust against outliers
- Question: Why does the undergraduate geography major at UNC Chapel Hill enjoy a historically high mean salary
- A robust alternative of the mean (the average value) is the median (the middle value)
- The two-sample Wilcoxon test uses the ranks (rather than values) of the data
- Ranks are robust against outliers

Mean versus Median

```
set.seed(312569)
x <- rnorm(19)
round(sort(x), 2)

##  [1] -2.54 -1.42 -0.99 -0.53 -0.25 -0.04  0.09  0.09  0.12  0.13  0.16
## [12]  0.21  0.36  0.42  0.75  0.75  1.15  1.80  2.05
```

Replace the second observation by a large number

```
set.seed(123129)
xcorrupt <- x
xcorrupt[2] <- 101.21
round(sort(xcorrupt), 2)

##  [1]  -2.54  -1.42  -0.99  -0.53  -0.25  -0.04   0.09   0.12   0.13   0.16
## [11]   0.21   0.36   0.42   0.75   0.75   1.15   1.80   2.05 101.21
```

Compare mean and median of data without outlier

```
data.frame(mean = mean(x), median = median(x))

##        mean    median
## 1 0.1208582 0.1282803
```

Compare mean and median of corrupted data

```
data.frame(mean = mean(xcorrupt), median = median(xcorrupt))

##       mean    median
## 1 5.443053 0.1570972
```


Wilcoxon Test

```
set.seed(123710)
x <- c(rnorm(5, 0, 1), rnorm(5, 10, 1))
grp <- rep(1:2, each = 5)
rnk <- rank(x)
data.frame(x, rnk, grp) %>%
    kbl()
```

x	rnk	grp
0.0467444	5	1
-1.1233349	1	1
-0.5394873	4	1
-1.0626156	2	1
-0.9470008	3	1
11.6006224	10	2
9.2314776	7	2
9.6009881	8	2
9.2161833	6	2
11.3689849	9	2

- The sum of the ranks of the observations in group 1 is 15
- The sum of the ranks of the observations in group 2 is 40
- Why are more highly ranked observations in group 2?
- How would you expect that these sums would compare under the null hypothesis?
- The Wilcoxon test is based on the sum of the ranks that belong to group 1 (or equivalently to group 2)

Well known fact

- There is a price to be paid for using the Wilcoxon test over the t-test if there no assumptional deviations
- It is well known that the Wilcoxon test is 95% (actually $\frac{3}{\pi}$) “efficient”
- Without getting to technical: there is 5% “loss” for using the Wilcoxon test if the use of the two-sample t-test is fully justified

Section 3

Distributions for Count Data

Two Approaches for Analysis of RNA-Seq

- Two-stage method: Convert counts to ”Expression” and then use statistical methods for microarrays (e.g., t-test, Wilcoxon)
- One-stage method: Relate the counts directly to the phenotype
- This is done through using statistical methods for modeling counts
- DESeq2 is widely used package for modelling count data from bulk RNA-Seq
- Seurat offers DESeq2 (although not recommended) and negative binomial for modelling count data from single-cell RNA-Seq

Some challenges with Count Data

- Counts are not directly comparable
- In RNA-Seq studies the count is among other things dependent on depth
- Count data are over-dispersed
- The actual variance of the data is larger than the one postulated by the model
- For many common count distributions, the mean and variance are entangled (cannot be modelled independently)

Three Distributions for Count Data

- RNA-Seq data are counts (not continuous measurements)
- To properly model RNA-Seq data, we need to consider distributions to model counts
- We will consider three important distributions for counts:
  - Binomial
  - Poisson
  - Negative Binomial
- There are many other distributions for counts (*e.g.*, geometric distribution) that will not be discussed

Flipping the coin

- Throughout this discussing we will consider flipping a coin
- The coin lands a head with probability π (could be biased) or tail with probability $1 - \pi$
- For convenience, we will recode H as 1 and T as 0
- We will flip it n times.
- Notation:
  - n is to denote the number of *trials*
  - On any trial (or flip), if we land an H we will call it an event (or success)
  - or if we land a T we will call it a failure
- RNA-seq connection: You can think of a read mapping to a gene to be an event

Three Variants of the Coin Tossing Experiment

1. Fix the number of trials (n) upfront and then toss the coin n times
   - The number of events (among n trials) is random
2. Toss the coin a large number of times and assume that each one of these many trials has a small probability of being an event
   - Here n is large and π is small (close to 0)
3. Fix the number of desired events upfront, then toss the coin repeatedly to achieve that number
   - Here the number of trials n is random

Binomial Distribution

- The distribution is

$$P[K = k] = \binom{n}{k} \pi^k (1 - \pi)^{n-k},$$

  $k = 0, 1, 2, \ldots, n$
- The average count for this distribution is $n\pi$
- The variance for this distribution is $n\pi(1 - \pi)$
- A famous example: the distribution of number of copies of the variant allele under Hardy-Weinberg Equilibirum

Poisson Distribution

- The Poisson distribution is used to model the count of the occurrence of rare events
- Classical application: Model for earthquakes
- The PMF is

$$P(K = k) = \frac{e^{-\lambda} \lambda^k}{k!},$$

  where $k = 0, 1, 2, \ldots$
- λ is the average number of events for this distribution
- λ is also the variance of this distribution

Negative Binomial Distribution

- How many times do you have to flip a coin to get $r > 0$ events
- Model the number of *random* trials needed to get r events
- This distribution is called the negative binomial distribution
- The probability distribution is

$$P[K = k] = \binom{k+r-1}{r-1}\pi^r(1-\pi)^k,$$

where $k = r, r+1, r+2, \ldots$

Mean and Variance of Negative Binomial

- A negative binomial distribution can be parameterized in terms of
  - r and p
  - or μ and σ^2
  - or μ and a dispersion parameter α (more on this later)
- The relationship between these two parametrizations is given by

$$\mu = r\frac{1-p}{p} \text{ and } \sigma^2 = r\frac{1-p}{p^2},$$

and

$$p = \frac{\mu}{\sigma^2} \text{ and } r = \frac{\mu^2}{\sigma^2 - \mu}$$

- If you provide r and p, you can calculate μ and σ^2
- Or, if you provide μ and σ^2, you can recover r and p.

Negative Binomial PMF in terms of μ and α

- The NB PMF parametrized in terms of p and r (the number of events) is

$$P[K = k] = \binom{k+r-1}{r-1}\pi^r(1-\pi)^k,$$

where $k = r, r+1, r+2, \ldots$
- The NB PMF parametrized in terms of the mean μ and the dispersion parameter α is

$$P[K = k] = \frac{\Gamma[k+\alpha^{-1}]}{\Gamma[\alpha^{-1}]\Gamma[k+1]}\left(\frac{1}{1+\mu\alpha}\right)^{\alpha^{-1}}\left(\frac{\mu}{\alpha^{-1}+\mu}\right)^k,$$

where $k = 0, 1, \ldots$
- The variance is $\mu(1+\alpha\mu)$
- As α shrinks to 0 (no-dispersion), the distribution becomes Poisson

Negative Binomial PMF for RNA-Seq

- We will use the mean/dispersion parameter representation for RNA-Seq

$$P[K=k] = \frac{\Gamma[k+\alpha^{-1}]}{\Gamma[\alpha^{-1}]\Gamma[k+1]}\left(\frac{1}{1+\mu\alpha}\right)^{\alpha^{-1}}\left(\frac{\mu}{\alpha^{-1}+\mu}\right)^{k},$$

  where $k = 0, 1, \ldots$
- The variance is $\mu(1+\alpha\mu)$
- IMPORTANT:
  - If $\alpha > 0$, then the variance is greater than the mean. Why?
  - As α shrinks to 0 (no-dispersion), the distribution becomes Poisson
- More on over-dispersion later

Means and Variances

Distribution	Support	Mean	Variance
Bernoulli(π)	0,1	π	$\pi(1-\pi)$
Binomial(n,π)	$0,1,\ldots,n$	$n\pi$	$n\pi(1-\pi)$
Poisson(λ)	$0,1,2,\ldots,$	λ	λ
NB(p,r)	$r, r+1, r+2, \ldots,$	$r\frac{1-p}{p}$	$r\frac{1-p}{p^2}$
NB(μ,α)	$0,1,\ldots,$	μ	$\mu(1+\alpha\mu)$

Negative Binomial vs Binomial or Poisson

- The Binomial distribution has one parameter π
- The Poisson distribution has one parameter λ
- The Negative Binomial has two parameters μ and α
- Advantage: Having two parameters, gives NB more flexibility
- Disadvantage: The negative binomial distribution poses a more challenging numerical optimization problem