Genetics and Bioinformatics

Kristel Van Steen, PhD^2

Montefiore Institute - Systems and Modeling

GIGA - Bioinformatics

ULg

kristel.vansteen@uliege.be

Sequence analysis studies

1 Setting the pace

1.a Terminology

1.b Probability distributions

2 Frequency of occurrence of “words”

2.a 1-letter words

3.b 2-letter words

3.c 3-letter words

3 Types of sequence analyses

3.a Summary

3.b Rare variants association studies

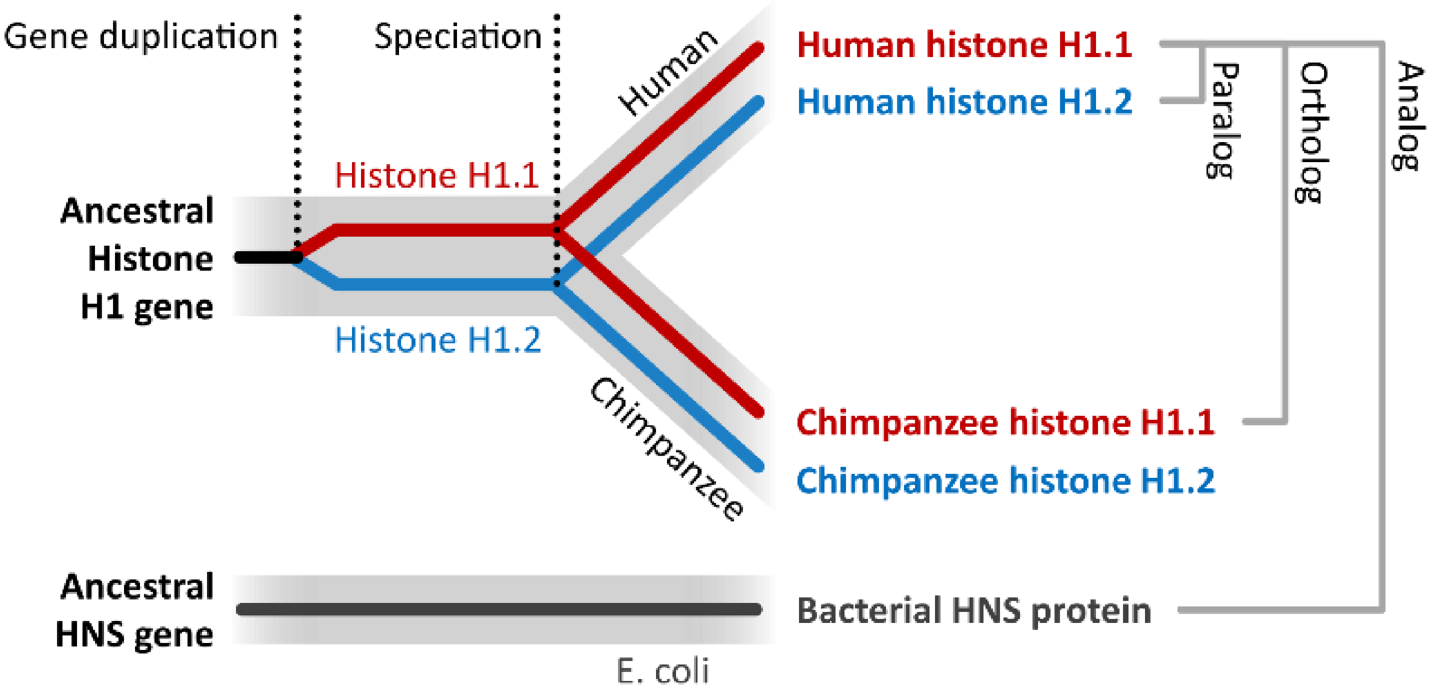
1 Setting the pace

1.a Terminology

Homology

- Homology forms the basis of organization for comparative biology.
- **Sequence homology** is the biological homology between DNA, RNA, or protein sequences, defined in terms of shared ancestry in the evolutionary history of life.
- In genetics, the term “homolog” is used both to refer to a homologous protein and to the gene (DNA sequence) encoding it.

- Two segments of DNA can have shared ancestry because of either a speciation event (orthologs) or a duplication event (paralogs).



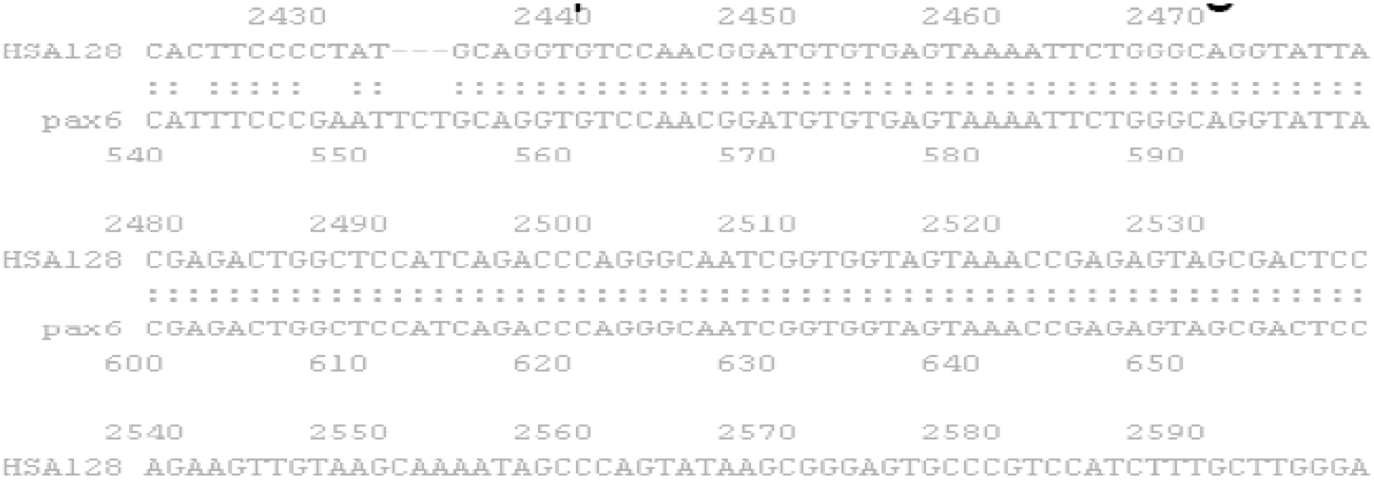
(By Thomas Shafee - Own work, CC BY 4.0, https://commons.wikimedia.org/w/index.php?curid=68505353)

Homology

- Initial characterization of any new DNA or protein sequence starts with a database search aimed at finding out whether homologs of this gene (protein) are already available, and if they are, what is known about them.
- Homology among DNA, RNA, or proteins is typically inferred from their nucleotide or amino acid **sequence similarity**. Homology among proteins or DNA is often incorrectly concluded on the basis of sequence similarity.
- **Significant similarity** is strong evidence that two sequences are related by evolutionary changes from a common ancestral sequence.
- **Alignments** of multiple sequences are used to indicate which regions of each sequence are homologous.

[Stay tuned for RNA + Proteome analyses classes]

Alignment vs frequency of occurrences of “text” (letters, words, …)



```
            2430         2440      2450      2460      2470
HSA128 CACTTCCCCTAT---GCAGGTGTCCAACGGATGTGTGAGTAAAATTCTGGGCAGGTATTA
       :: :::::  ::   :::::::::::::::::::::::::::::::::::::::::::::
  pax6 CATTTCCCGAATTCTGCAGGTGTCCAACGGATGTGTGAGTAAAATTCTGGGCAGGTATTA
     540       550       560       570       580       590

     2480      2490      2500      2510      2520      2530
HSA128 CGAGACTGGCTCCATCAGACCCAGGGCAATCGGTGGTAGTAAACCGAGAGTAGCGACTCC
       ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
  pax6 CGAGACTGGCTCCATCAGACCCAGGGCAATCGGTGGTAGTAAACCGAGAGTAGCGACTCC
     600       610       620       630       640       650

     2540      2550      2560      2570      2580      2590
HSA128 AGAAGTTGTAAGCAAAATAGCCCAGTATAAGCGGGAGTGCCCGTCCATCTTTGCTTGGGA
```

1.b Probability distributions

Our context

- Words are short strings of letters drawn from an alphabet
- In the case of DNA, the set of letters is A, C, T, G
- A word of length k is called a k-word or k-tuple
- Differences in word frequencies help to differentiate between different DNA sequence sources or regions
- Examples: 1-tuple: individual nucleotide; 2-tuple: dinucleotide; 3-tuple: codon
- The **distributions** of the nucleotides over the DNA sequences have been studied for many years → hidden correlations in the sequences (e.g., CpGs)

Probability is the science of uncertainty

1. Rules → data: given the rules, describe the likelihoods of various events occurring
2. Probability is about prediction – looking forwards
3. Probability is mathematics

Statistics is the science of data

1. Rules ← data: given only the data, try to guess what the rules were. That is, some probability model controlled what data came out, and the best we can do is guess – or approximate – what that model was. We might guess wrong, we might refine our guess as we obtain / collect more data
2. Statistics is about looking backward. Once we make our best *statistical guess* about what the probability model is (what the rules are), based on looking backward, we can then use that probability model to predict the future
3. Statistics is an art. It uses mathematical methods but it is much more than mathematics alone
4. The purpose of statistics is to make inference about unknown quantities from samples of data.

Statistics is the science of data

- Probability distributions are a fundamental concept in statistics.
- Before computing an interval or test based on a distributional assumption, we need to verify that the assumption is justified for the given data set.
- For this chapter, the distribution does not always need to be the best-fitting distribution for the data, but an adequate enough model so that the statistical technique yields valid conclusions.
- Simulation studies: one way to obtain empirical evidence for a probability model

Expected values and variances

- Mean and variance are two important properties of real-valued random variables and corresponding probability distributions.
- The "mean" of a discrete random variable *X* taking values $x_1, x_2, \ldots$ (denoted EX (or E(X) or E[X]), where E stands for expectation, which is another term for mean) is defined as:

$$E(X) = \sum_i x_i \, P(X = x_i)$$

  - $E(X_i) = 1 \times p_A + 0 \times (1 - p_A)$ if $x_i = A$ or $\{\text{another letter}\}$
  - If $Y = c\,X$, then $E(Y) = c\,E(X)$
  - $E(X_1 + \ldots + X_n) = E(X_1) + \ldots + E(X_n)$
- Because X_i are assumed to be independent and identically distributed (iid):

$$E(X_1 + \ldots + X_n) = n\,E(X_1) = n\,p_A$$

Expected values and variances

- The idea is to use squared deviations of X from its center (expressed by the mean). Expanding the square and using the linearity properties of the mean, the Var(X) can also be written as:

$$Var(X) = E(X^2) - [E(X)]^2]$$

  - If Y=c X then Var (Y) = c^2 Var (X)
  - The variance of a sum of independent random variables is the sum of the individual variances

- For the random variables X_i taking on values A or sth else:
  Var (X_i) = $[1^2 \; \times p_A + \; 0^2 \; \times' \, (1 - p_A)] - \; p_A^2$ = $p_A(1 - p_A)$
  Var (N) = n Var (X_1) = $np_A(1 - p_A)$

Expected values and variances

- The expected value of a random variable X gives a measure of its location. Variance is another property of a probability distribution dealing with the spread or variability of a random variable around its mean.

$$Var(X) = E\ (\ [X - E(X)]^2\ )$$

  - The positive square root of the variance of X is called its standard deviation sd(X) or σ_X

Independence

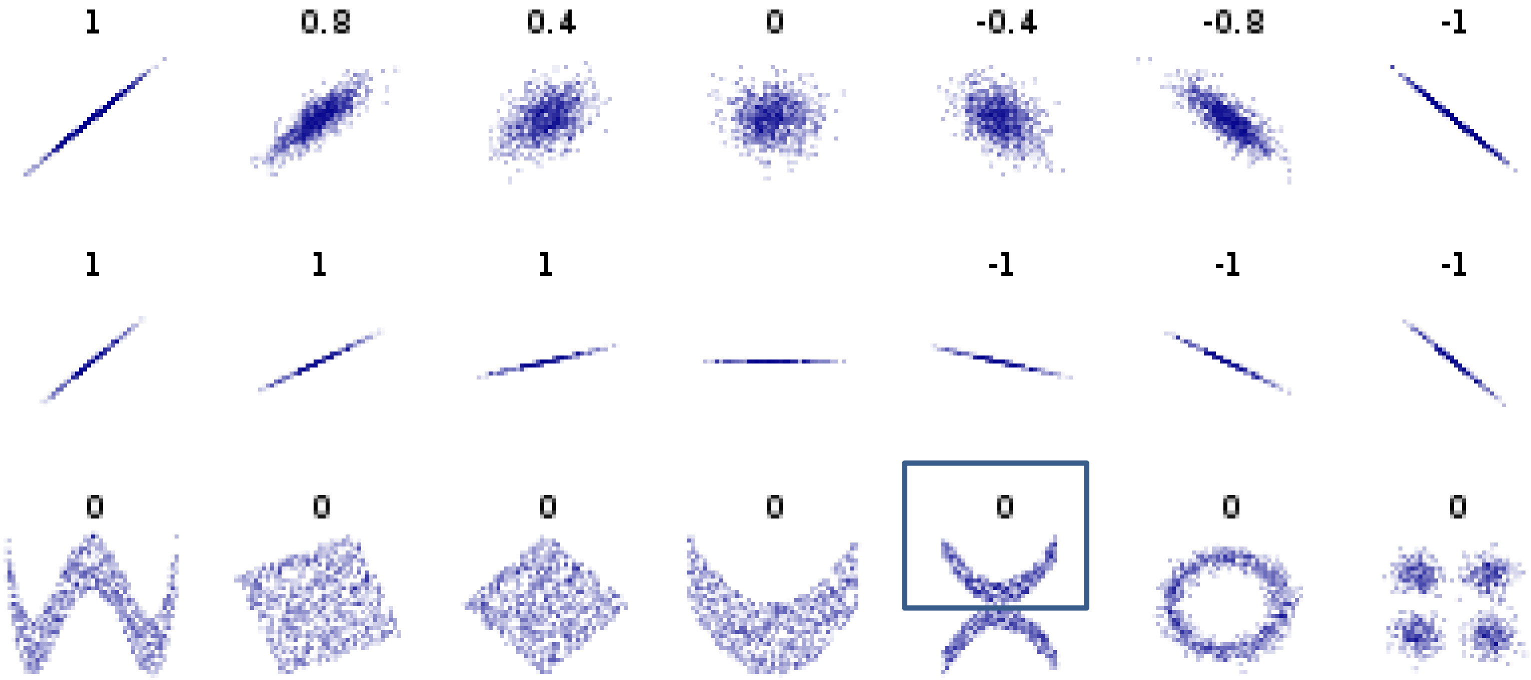
- Discrete random variables X_1, …, X_n are said to **be independent** if for any subset of random variables and actual values, the joint distribution equals the product of the component distributions

Correlation

- Correlation is a measure of association, most often used to reflect how two variables are related/associated
- There are several correlation coefficients, often denoted ρ or r.
- The most common of these is the **Pearson correlation coefficient**, which is sensitive only to a linear relationship between two variables (which may be present even when one variable is a nonlinear function of the other).
- Other correlation coefficients (f.i. Spearman's rank correlation) are more robust and/or sensitive to non-linear relation

$$Corr(X_1, X_2) = \frac{Cov(X_1, X_2)}{\sigma_{X1}\sigma_{X2}}$$

Is independence equivalent to correlation?



(Wikipedia)

2 Frequency of occurrence of “words”

2.a 1-letter words

Assumptions

- Notation for the output of a random string of *n* bases may be: $L_1, L_2, \ldots, L_n$ (L_i = base inserted at position or locus *i* of the sequence)
  - The values l_j for L_j will come from a set χ (with J possibilities)
  - For a DNA sequence, J=4 and $\chi = \{A, \mathrm{C}, \mathrm{T}, \mathrm{G}\}$
- Simple rules specifying a probability model:
  - First base in sequence is either A, C, T or G with prob p_A, p_C, p_T, p_G
  - Suppose the first r bases have been generated, while generating the base at position r+1, no attention is paid to what has been generated before.

- Then we can actually generate A, C, T or G with the probabilities above
- According to our simple model, the L_i are independent and hence

$$P(L_1=l_1,L_2=l_2, \ldots,L_n=l_n)=P(L_1=l_1)\ P(L_2=l_2)\ \ldots P(L_n=l_n)$$

- If p_j is the prob that the value (realization of the random variable *L*) l_j occurs, then
    - $p_1, \ldots, p_J \geq O$ and $p_1 + \ldots + p_J = 1$
- The **probability distribution** (probability mass function) of L is given by the collection $p_1, \ldots, p_J$
    - $P(L=l_j) = p_j$, $j=1, \ldots,$ J
- The probability that an event *S* occurs (subset of χ) is $P(L \in S)$ = $\sum_{j:l_j \in S} (p_j)$

Probability distributions of interest

- What is the probability distribution of the number of times a given pattern occurs in a random DNA sequence $L_1, \ldots, L_n$? Simple pattern = "A"
  - New sequence $X_1, \ldots, X_n$:

$$X_i=1 \text{ if } L_i=A \text{ and } X_i=0 \text{ else}$$

  - The number of times N that A appears is the sum

$$N=X_1+\ldots+X_n$$

  - The prob distr of each of the X_i:

$$P(X_i=1) = P(L_i=A)=p_A$$

$$P(X_i=0) = P(L_i=C \text{ or } G \text{ or } T) = 1 - p_A$$

- What is a "typical" value of N?
  - Depends on how the individual X_i (for different *i*) are interrelated

The binomial distribution

- The binomial distribution is used when there are exactly two mutually exclusive outcomes of a trial. These outcomes are appropriately labeled "success" and "failure". The binomial distribution is used to obtain the probability of observing x successes in a fixed number of trials, with the probability of success on a single trial denoted by p. The binomial distribution assumes that p is fixed for all trials.
- The formula for the binomial probability mass function is :

$$P(N = j) = \binom{n}{j} p^j (1-p)^{n-j}, \text{ j = 0,1, …,n}$$

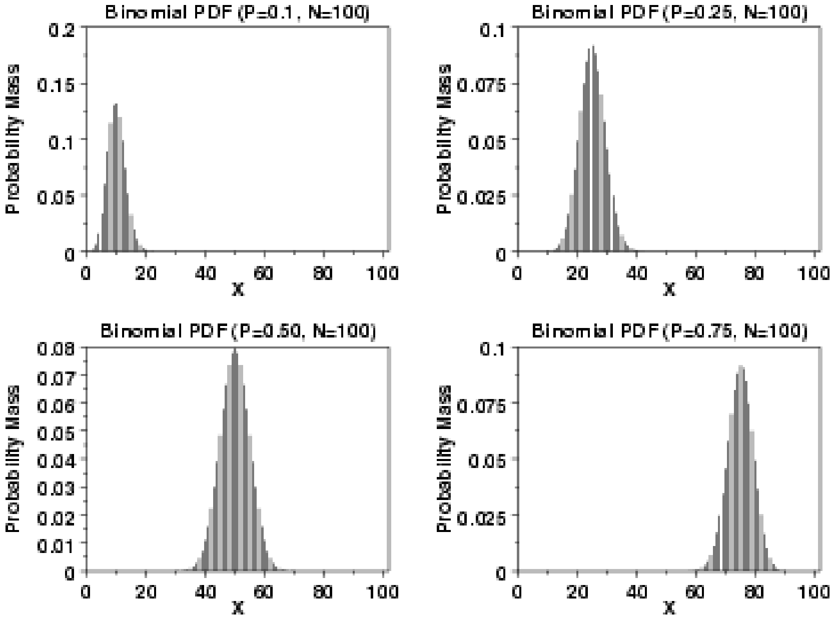
with the binomial coefficient $\binom{n}{j}$ determined by

$$\binom{n}{j} = \frac{n!}{j!\,(n-j)!},$$

and *j!=j(j-1)(j-2)…3.2.1, 0!=1*

The binomial distribution

- The mean is n*p* and the variance is n*p(1-p)*
- The following is the plot of the binomial probability density function for four values of *p* and n = 100.



Simulating from probability distributions

- The idea is that we can study the properties of the distribution of N when we can get our computer to output numbers N_1, …, N_k having the same distribution as N
    - We can use the sample mean to estimate the expected value E(N):

$$\overline{N} = (N_1 + \ldots + N_k)/k$$

    - Similarly, we can use the sample variance to estimate the true variance of N:

$$s^2 = \frac{1}{k-1}\sum_{i=1}^{k}(N_i - \overline{N})^2$$

Why do we use (k-1) and not k in the denominator?

Simulating from probability distributions

- What is needed to produce such a string of observations?
  - Access to **pseudo-random numbers**: random variables that are uniformly distributed on (0,1): any number between 0 and 1 is a possible outcome and each is equally likely
- In practice, simulating an observation with the distribution of X_1:
  - Take a uniform random number u
  - Set X_1=1 if $U \leq p \equiv p_A$ and 0 otherwise.
  - Why does this work? … $P(X_1 = 1) = P(U \leq p_A) = p_A$
  - Repeating this procedure n times results in a sequence X_1, …, X_n from which N can be computed by adding the X’s

Simulating from probability distributions

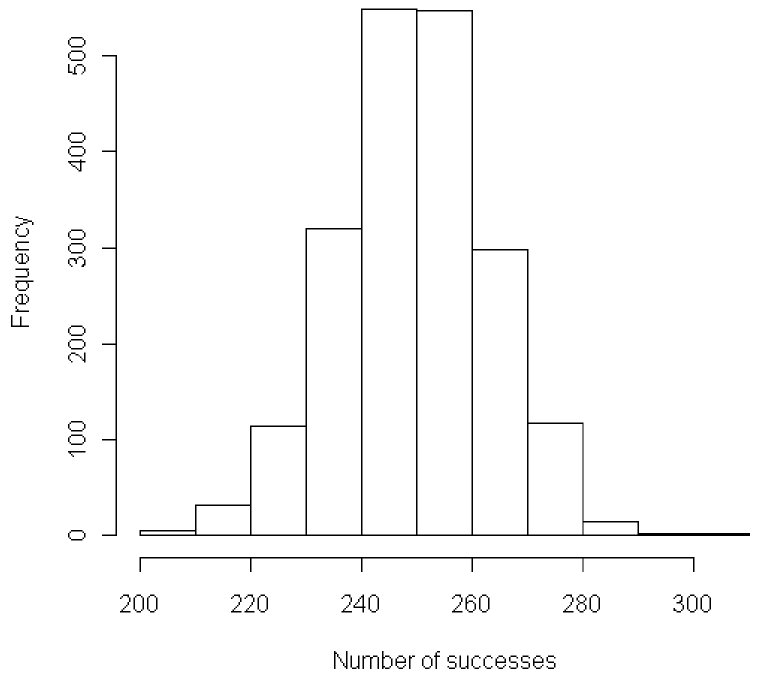
- FYI: Simulate a general DNA sequence of bases A, C, T, G:
  - Divide the interval (0,1) in 4 intervals with endpoints

    $$0, p_A, p_A + p_C, p_A + p_C + p_G, 1$$
  - If the simulated u lies in the leftmost interval, L_1=A
  - If u lies in the second interval, L_1=C; if in the third, L_1=G and otherwise L_1=T
  - Repeating this procedure n times with different values for U results in a sequence L_1, …, L_n
- Use the “sample” function in R:

```
pi <- c(0.25,0.75)
x<-c(1,0)
set.seed(2009)
sample(x,10,replace=TRUE,pi)
```

Simulating from probability distributions

- By looking through a given simulated sequence, we can count the number of times a particular pattern arises (for instance, the base A)
- By repeatedly generating sequences (k times) and analyzing each of them, we can get a feel for whether or not our particular pattern of interest is unusual

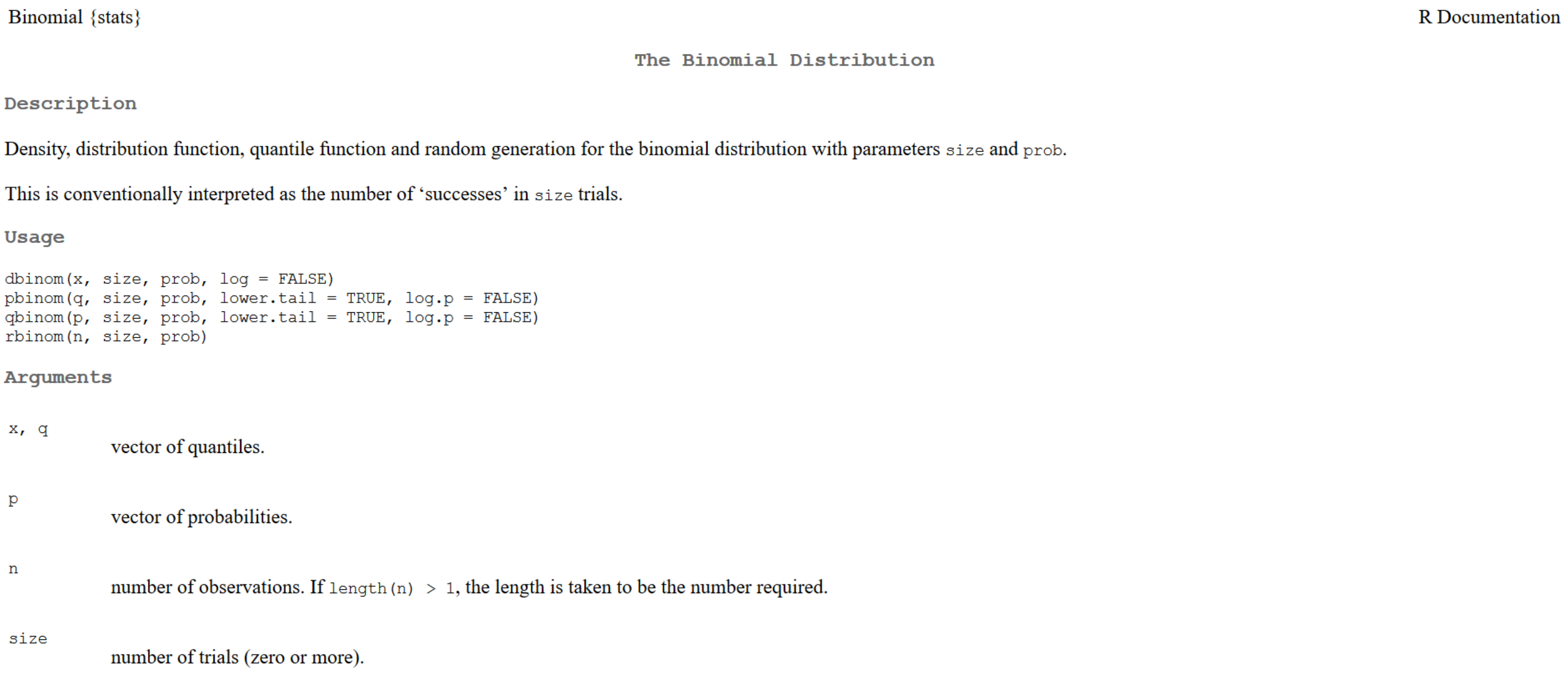


Simulating from a known probability distribution

- Using R code:

```
x<- rbinom(2000,1000,0.25)
mean(x)
sd(x)^2
hist(x,xlab="Number of successes",main="")
```

R documentation



Binomial {stats} R Documentation

The Binomial Distribution

Description

Density, distribution function, quantile function and random generation for the binomial distribution with parameters `size` and `prob`.

This is conventionally interpreted as the number of ‘successes’ in `size` trials.

Usage

```
dbinom(x, size, prob, log = FALSE)
pbinom(q, size, prob, lower.tail = TRUE, log.p = FALSE)
qbinom(p, size, prob, lower.tail = TRUE, log.p = FALSE)
rbinom(n, size, prob)
```

Arguments

`x, q`
vector of quantiles.

`p`
vector of probabilities.

`n`
number of observations. If `length(n) > 1`, the length is taken to be the number required.

`size`
number of trials (zero or more).

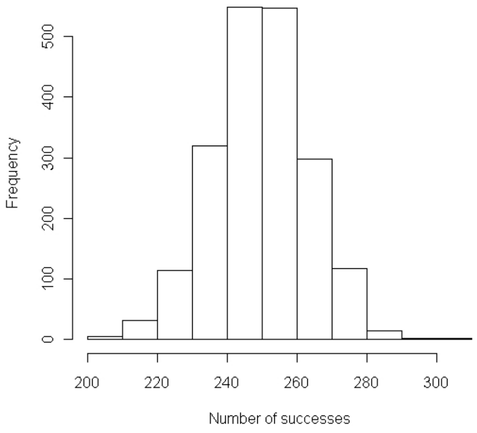
(https://stat.ethz.ch/R-manual/R-devel/library/stats/html/Binomial.html)

Simulating from a known probability distribution

- Using R code:

```
x<- rbinom(2000,1000,0.25)
mean(x)
sd(x)^2
hist(x,xlab="Number of successes",main="")
```

How many entries are taken to compute the mean(x)?



Number of sequences = 2000 = k

Number of trials = 1000 = n

Back to our original question

- Suppose we have a sequence of 1000bp and assume that every base occurs with equal probability. How likely are we to observe at least 300 A's in such a sequence?
  - Exact computation using a closed form of the relevant distribution
  - Approximate via simulation
  - Approximate using the Central Limit Theory

Exact computation via closed form of relevant distribution

- The formula for the binomial probability mass function is :

$$P(N = j) = \binom{n}{j} p^j (1-p)^{n-j}, \text{ j = 0,1, …,n}$$

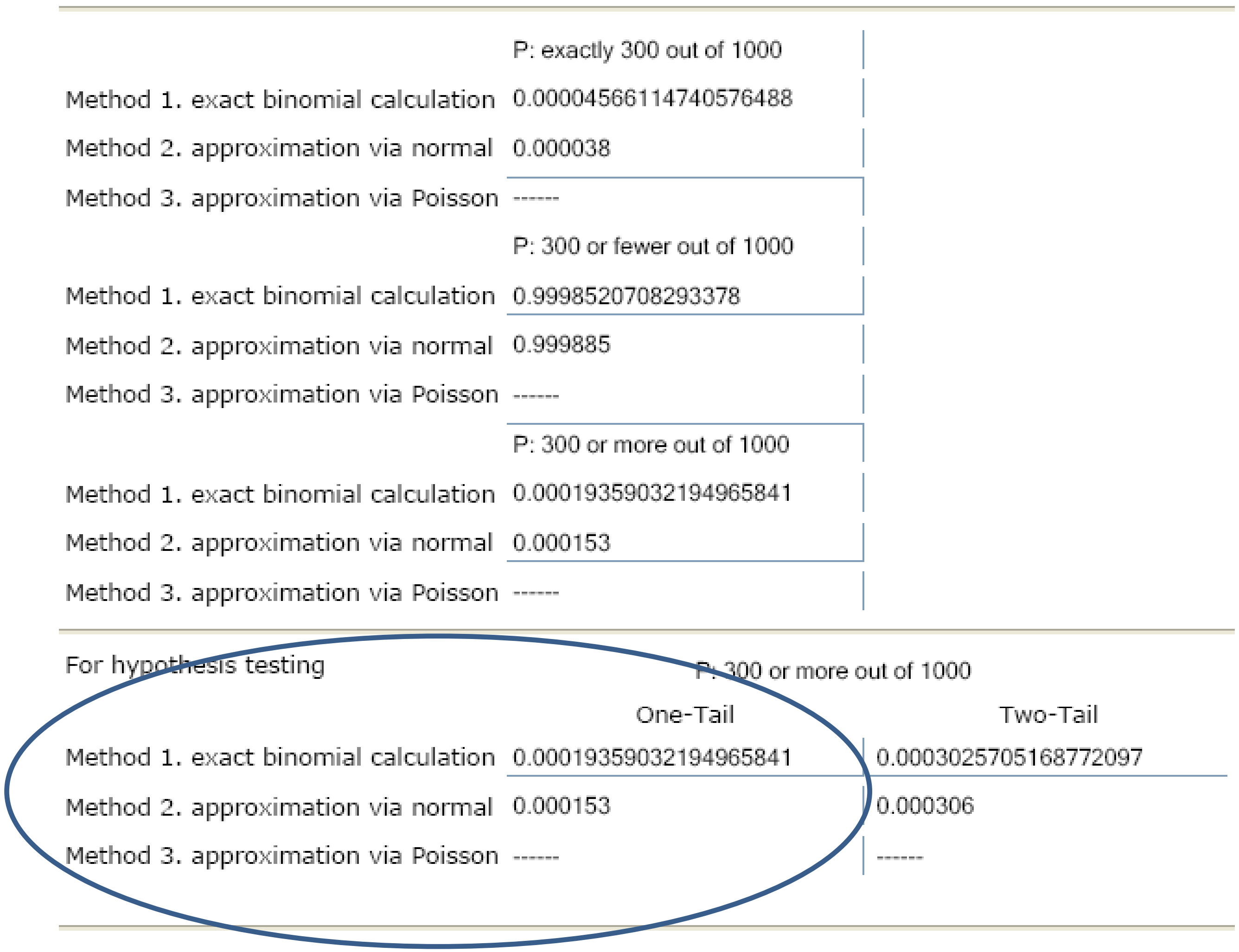
and therefore

$$P(N \geq 300) = \sum_{j=300}^{1000} \binom{1000}{j} (1/4)^j (1-1/4)^{1000-j}$$

= 0.000193590321944965841

- Note that the probability $P(N \geq 300)$ is estimated to be 0.0001479292 via

```
1-pbinom(300,size=1000,prob=0.25)
pbinom(300,size=1000,prob=0.25,lower.tail=FALSE)
```



	P: exactly 300 out of 1000
Method 1. exact binomial calculation	0.00004566114740576488
Method 2. approximation via normal	0.000038
Method 3. approximation via Poisson	------
	P: 300 or fewer out of 1000
Method 1. exact binomial calculation	0.9998520708293378
Method 2. approximation via normal	0.999885
Method 3. approximation via Poisson	------
	P: 300 or more out of 1000
Method 1. exact binomial calculation	0.00019359032194965841
Method 2. approximation via normal	0.000153
Method 3. approximation via Poisson	------

For hypothesis testing

	P: 300 or more out of 1000	
	One-Tail	Two-Tail
Method 1. exact binomial calculation	0.00019359032194965841	0.0003025705168772097
Method 2. approximation via normal	0.000153	0.000306
Method 3. approximation via Poisson	------	------

(http://faculty.vassar.edu/lowry/binomialX.html)

Approximate via simulation

- Using R code and simulations from the theoretical (“known”) distribution, $P(N \geq 300)$ can be estimated as 0.000196 via

```
x<- rbinom(1000000,1000,0.25)
sum(x>=300)/1000000
```

Approximate via Central Limit Theory

- The central limit theorem offers a 3rd way to compute probabilities of a distribution
- It applies to sums or averages of iid random variables
- Assuming that X_1, …, X_n are iid random variables with mean μ and variance σ^2, then we know that for the sample average

$$\bar{X}_n \ = \frac{1}{n}\,(X_1 + \ \ldots + \ X_n),$$

$$\mathrm{E}(\bar{X}_n) = \mu \text{ and Var } \overline{(X_n)} = \frac{\sigma^2}{n}$$

- Hence,

$$E\left(\frac{\bar{X}_n - \ \mu}{\sigma/\sqrt{n}}\right) = 0, Var\left(\frac{\bar{X}_n - \ \mu}{\sigma/\sqrt{n}}\right) = 1$$

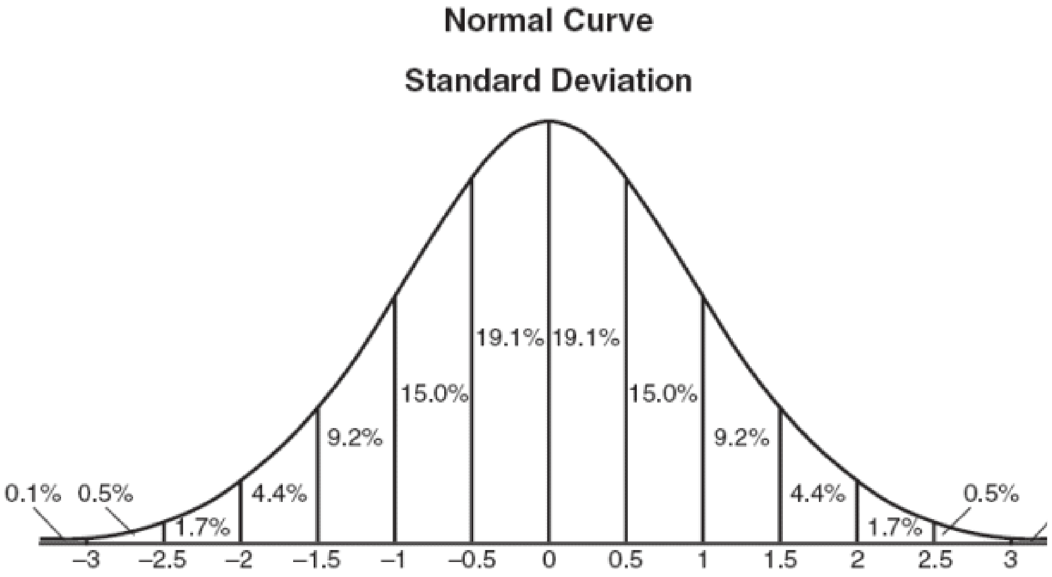
Approximate via Central Limit Theory

- The central limit theorem states that if the sample size n is large enough,

$$P\left(a \leq \frac{\bar{X}_n - \mu}{\frac{\sigma}{\sqrt{n}}} \leq b\right) \approx \phi(b) - \phi(a),$$

with $\phi(.)$ the standard normal distribution defined as

$$\phi(z) = P(Z \leq z) = \int_{-\infty}^{z} \phi(x)dx$$



Approximate via Central Limit Theory

- Estimating the quantity $P(N \geq 300)$ when N has a binomial distribution with parameters n=1000 and p=0.25,

$$E(N) = n\mu = 1000 \ \times 0.25 = 250,$$

$$sd(N) = \ \sqrt{n}\,\sigma = \sqrt{1000 \ \times \frac{1}{4} \times \frac{3}{4}} \approx 13.693$$

$$P(N \geq 300) = P\left(\frac{N - 250}{13.693} > \frac{300 - 250}{13.693}\right)$$

$$\approx P(Z > \ 3.651501) = \ 0.0001303560$$

- R code:

```
pnorm(3.651501,lower.tail=FALSE)
```

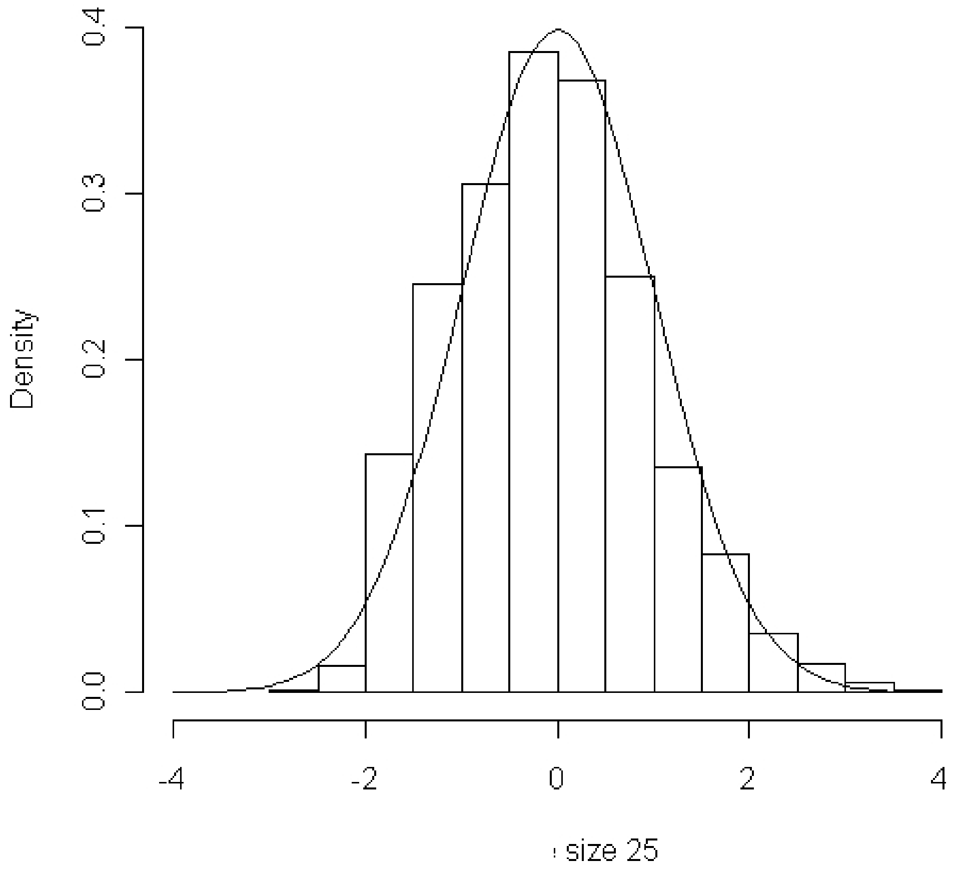
How do the estimates of $P(N \geq 300)$ compare?

Approximate via Central Limit Theory

- The central limit theorem in action using R code:

```
bin25<-rbinom(1000,25,0.25)
av.bin25 <- 25*0.25
stdev.bin25 <- sqrt(25*0.25*0.75)
bin25<-(bin25-av.bin25)/stdev.bin25
hist(bin25,xlim=c(-4,4),ylim=c(0.0,0.4),prob=TRUE,xlab="Sample size 25",main="")
x<-seq(-4,4,0.1)
lines(x,dnorm(x))
```

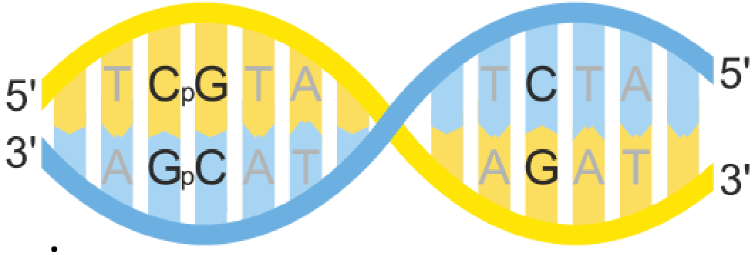
Approximate via Central Limit Theory



2.b 2-letter words

One motivation

- The CpG sites or CG sites are regions of DNA where a cytosine nucleotide is followed by a guanine nucleotide in the linear sequence of bases along its 5' → 3' direction.



- CpG sites occur with high frequency in genomic regions called CpG islands (or CG islands). Cytosines in CpG dinucleotides can be methylated to form 5-methylcytosines. Enzymes that add a methyl group are called DNA methyltransferases.
- In mammals, 70% to 80% of CpG cytosines are methylated [see also Homework 1 assignment: paper style]

Assumptions of independence simplify the problem once again

- Concentrating on abundances, and <u>assuming the iid model</u> for L_1, …, L_n:

$$P(L_i = l_i = C, L_{i+1} = l_{i+1} = G) = p_{l_i}\, p_{l_{i+1}}$$

- Has a given sequence an unusual dinucleotide frequency compared to the iid model?

Which statistic (that we have already seen in class) would apply?

Chi-square statistic

A chi-squared test can be completed by following five simple steps:

- Identify hypotheses (null versus alternative)
- Construct a table of frequencies (observed versus expected)
- Apply the chi-squared formula
- Determine the degree of freedom (df)
- Identify the p value (should be <0.05)

Example

- The trait for smooth peas (R) is dominant over wrinkled peas (r) and yellow pea colour (Y) is dominant to green (y)
- A dihybrid cross between two heterozygous pea plants is performed (RrYy × RrYy)
- The following phenotypic frequencies are observed:
  701 smooth yellow peas ; 204 smooth green peas ; 243 wrinkled yellow peas ; 68 wrinkled green peas

Observed minus expected

Round = R
Wrinkled = r

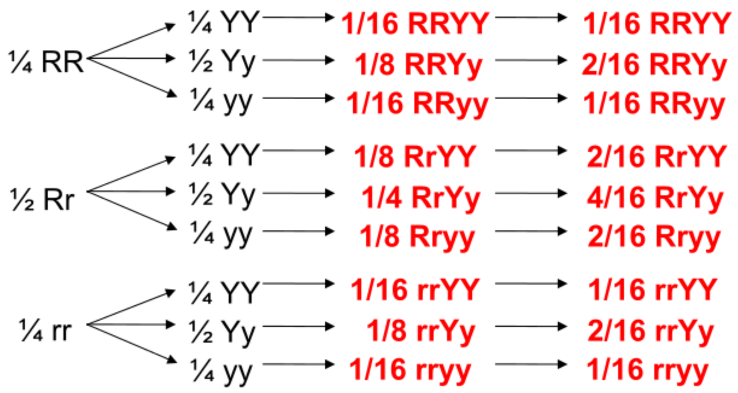
	R	r
R	RR	Rr
r	Rr	rr

¼ RR ½ Rr ¼ rr

Yellow = Y
Green = y

	Y	y
Y	YY	Yy
y	Yy	yy

¼ YY ½ Yy ¼ yy



¼ RR
¼ YY → 1/16 RRYY → 1/16 RRYY
½ Yy → 1/8 RRYy → 2/16 RRYy
¼ yy → 1/16 RRyy → 1/16 RRyy

½ Rr
¼ YY → 1/8 RrYY → 2/16 RrYY
½ Yy → 1/4 RrYy → 4/16 RrYy
¼ yy → 1/8 Rryy → 2/16 Rryy

¼ rr
¼ YY → 1/16 rrYY → 1/16 rrYY
½ Yy → 1/8 rrYy → 2/16 rrYy
¼ yy → 1/16 rryy → 1/16 rryy

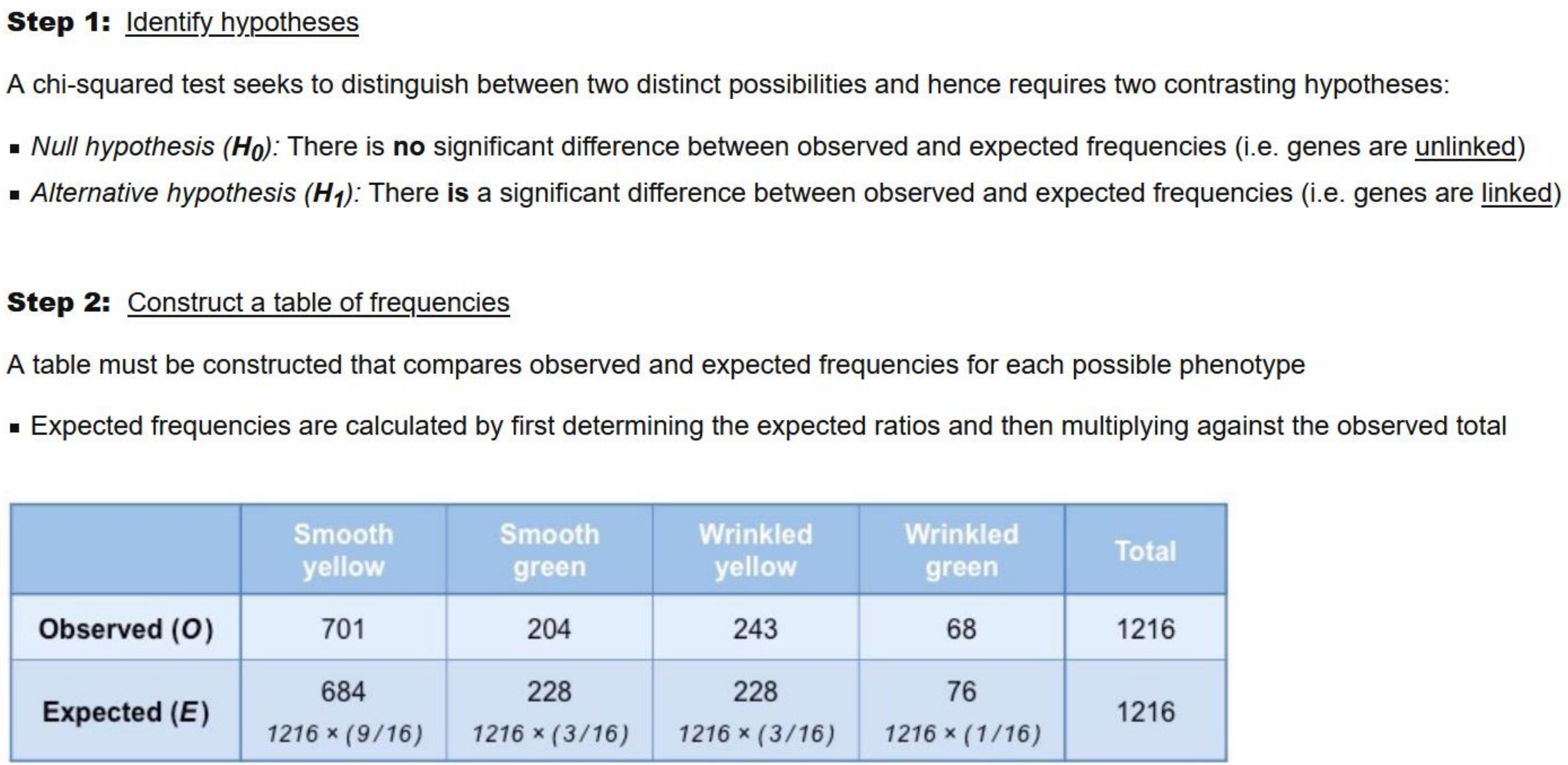
Observed Frequencies

Pea	Phenotype	Frequency
●	Smooth yellow	701
●	Smooth green	204
●	Wrinkled yellow	243
●	Wrinkled green	68
	Total	1216

Expected Ratios for Unlinked Traits

	RY	Ry	rY	ry
RY	RRYY	RRYy	RrYY	RrYy
Ry	RRYy	RRyy	RrYy	Rryy
rY	RrYY	RrYy	rrYY	rrYy
ry	RrYy	Rryy	rrYy	rryy

● = 9 ● = 3 ● = 3 ● = 1



Step 1: Identify hypotheses

A chi-squared test seeks to distinguish between two distinct possibilities and hence requires two contrasting hypotheses:

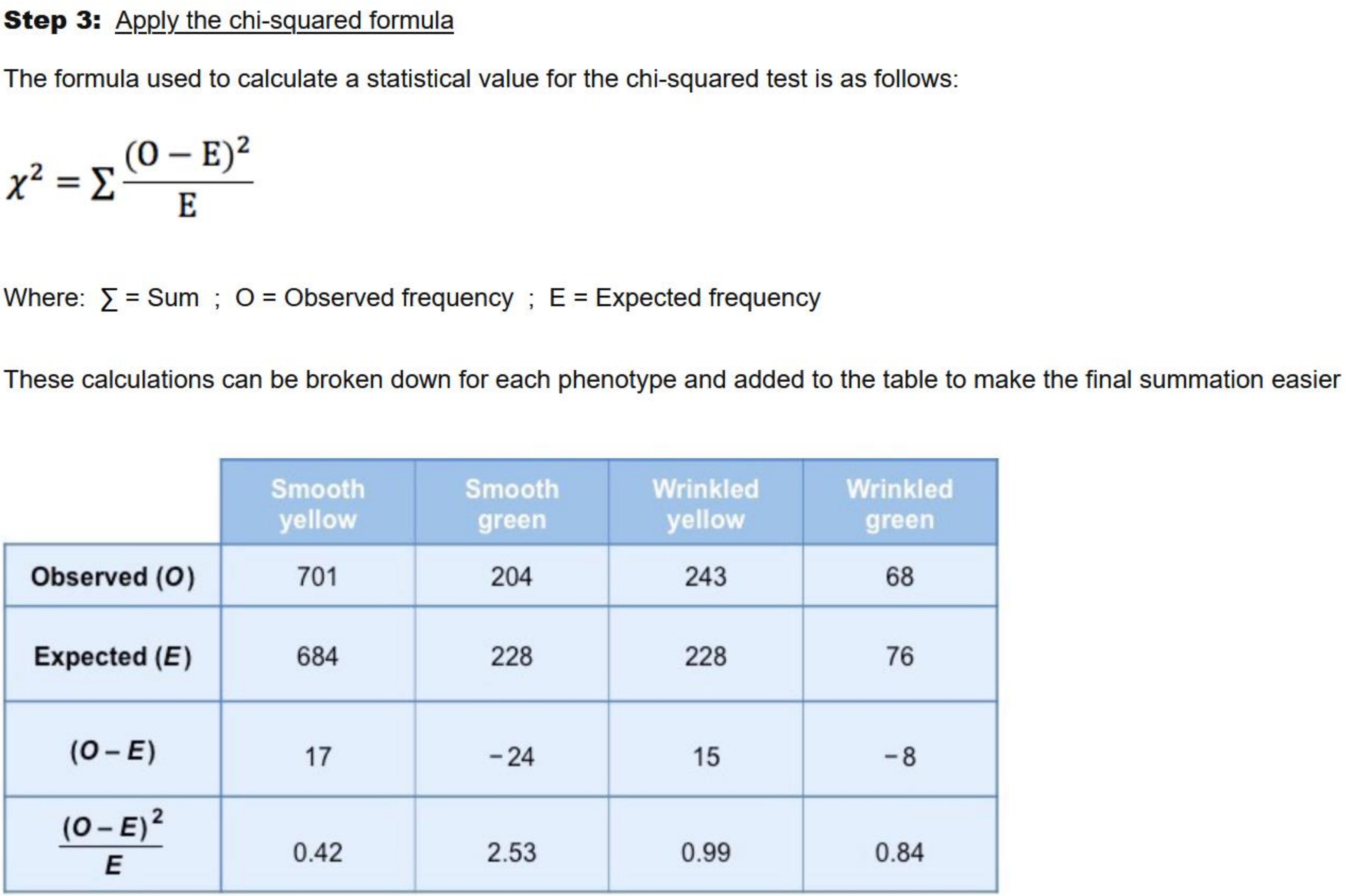
- *Null hypothesis (**H_0**):* There is **no** significant difference between observed and expected frequencies (i.e. genes are unlinked)
- *Alternative hypothesis (**H_1**):* There **is** a significant difference between observed and expected frequencies (i.e. genes are linked)

Step 2: Construct a table of frequencies

A table must be constructed that compares observed and expected frequencies for each possible phenotype

- Expected frequencies are calculated by first determining the expected ratios and then multiplying against the observed total

	Smooth yellow	Smooth green	Wrinkled yellow	Wrinkled green	Total
Observed (*O*)	701	204	243	68	1216
Expected (*E*)	684 *1216 × (9/16)*	228 *1216 × (3/16)*	228 *1216 × (3/16)*	76 *1216 × (1/16)*	1216



Step 3: Apply the chi-squared formula

The formula used to calculate a statistical value for the chi-squared test is as follows:

$$\chi^2 = \sum \frac{(O - E)^2}{E}$$

Where: $\sum$ = Sum ; O = Observed frequency ; E = Expected frequency

These calculations can be broken down for each phenotype and added to the table to make the final summation easier

	Smooth yellow	Smooth green	Wrinkled yellow	Wrinkled green
Observed (*O*)	701	204	243	68
Expected (*E*)	684	228	228	76
$(O - E)$	17	− 24	15	− 8
$\frac{(O-E)^2}{E}$	0.42	2.53	0.99	0.84

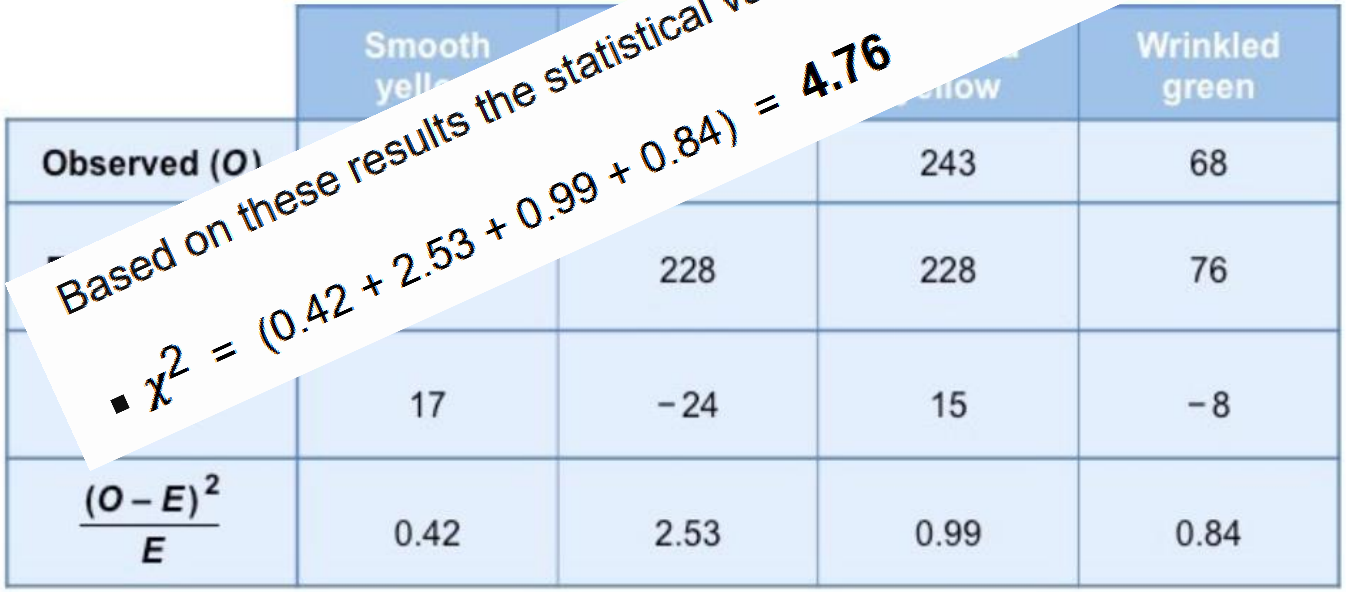
Step 3: Apply the chi-squared formula

The formula used to calculate a statistical value for the chi-squared test is as follows:

$$\chi^2 = \sum \frac{(O - E)^2}{E}$$

Where: ∑ = Sum ; O = Observed frequency ; E = Expected frequency

These calculations can be broken down for each phenot [illegible] ake the final summation easier



	Smooth yellow	[illegible]	[illegible] yellow	Wrinkled green
Observed (*O*)	[illegible]	[illegible]	243	68
[illegible]	[illegible]	228	228	76
[illegible]	17	−24	15	−8
$\frac{(O-E)^2}{E}$	0.42	2.53	0.99	0.84

Based on these results the statistical value calculated by the chi-squared test is as follows:

- χ^2 = (0.42 + 2.53 + 0.99 + 0.84) = **4.76**

Step 4: Determine the degree of freedom (df)

In order to determine if the chi-squared value is statistically significant a degree of freedom must first be identified

- The degree of freedom is a mathematical restriction that designates what range of values fall within each significance level

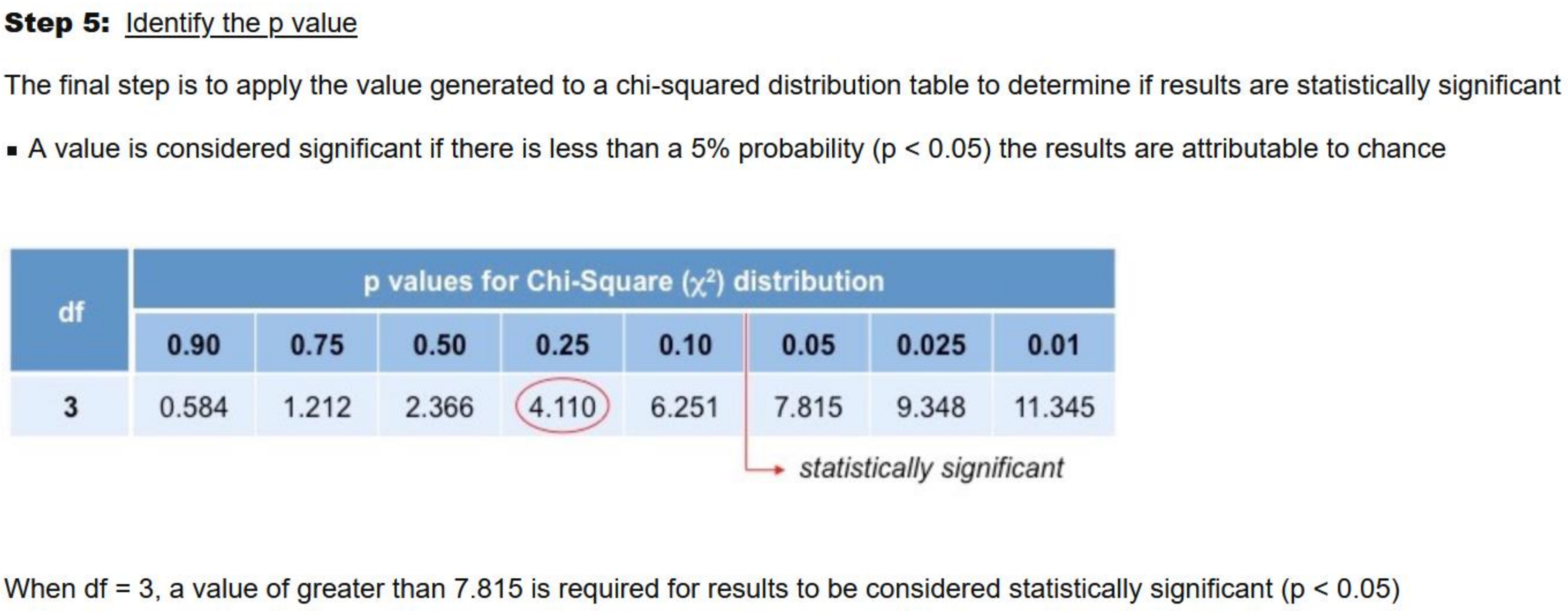
The degree of freedom is calculated from the table of frequencies according to the following formula:

$$df = (m - 1)(n - 1)$$

Where: m = number of rows ; n = number of columns

For all dihybrid crosses, the degree of freedom should be: (number of phenotypes – 1)

- In this particular instance, the degree of freedom is **3**



Step 5: Identify the p value

The final step is to apply the value generated to a chi-squared distribution table to determine if results are statistically significant

- A value is considered significant if there is less than a 5% probability ($p < 0.05$) the results are attributable to chance

df	p values for Chi-Square (χ^2) distribution							
	0.90	0.75	0.50	0.25	0.10	0.05	0.025	0.01
3	0.584	1.212	2.366	4.110	6.251	7.815	9.348	11.345

statistically significant

When df = 3, a value of greater than 7.815 is required for results to be considered statistically significant ($p < 0.05$)

What is your conclusion?

(https://ib.bioninja.com.au/higher-level/)

Returning to CpG counts…

- Compare observed O with expected E dinucleotide numbers

$$\chi^2 = \frac{(\mathrm{O}-\mathrm{E})^2}{\mathrm{E}},$$

with $E = (n-1)p_{l_i}p_{l_{i+1}}$.

Why (n-1) as factor in E above?

- How to determine which values of $\chi^2$are unlikely or extreme?
  - If the observed nr is close to the expected number, then the statistic will be small. Otherwise, the model will be doing a poor job of predicting the dinucleotide frequencies and the statistic will tend to be large…
  - Degrees of freedom?

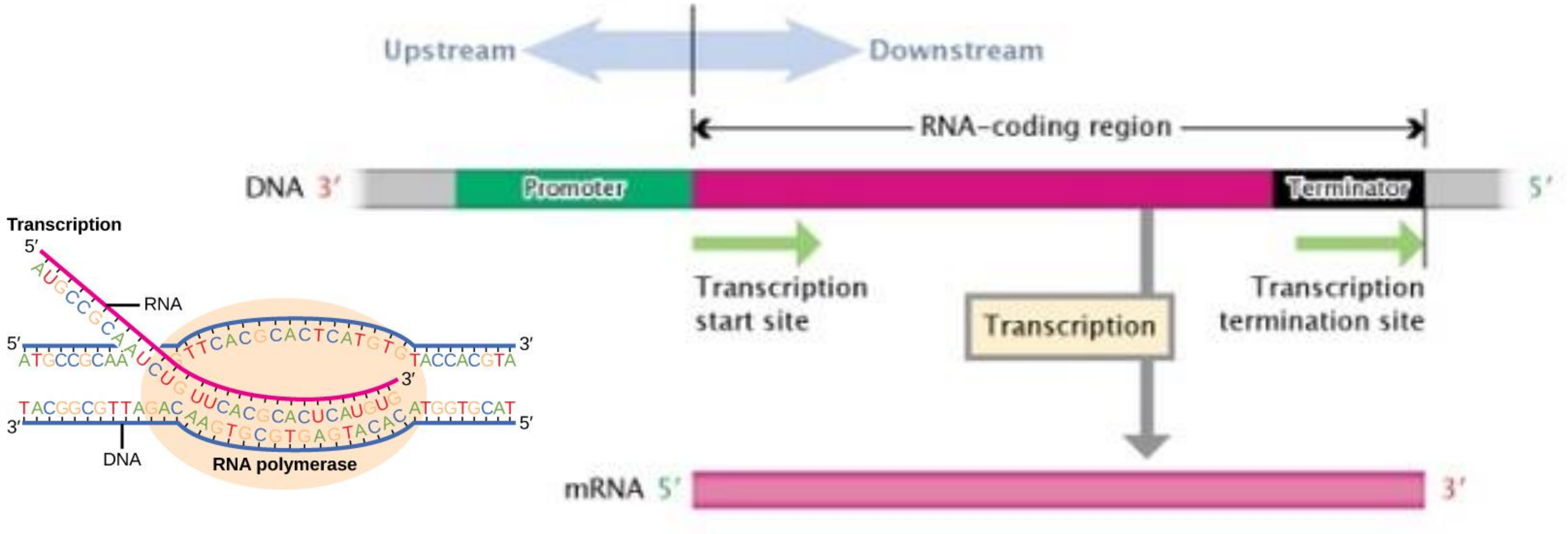
- Recipe:
  - Compute the number c given by

$$c = \begin{cases} 1 + 2p_{l_i} - 3p_{l_i}^2, & \text{if } l_i = l_{i+1} \\ 1 - 3p_{l_i}p_{l_{i+1}}, & \text{if } l_i \neq l_{i+1} \end{cases}$$

  - Calculate the ratio $\frac{\chi^2}{c}$, where $\chi^2$is given as before
  - Compare the ratio to a chi-square distribution: If this ratio is larger than 3.84 then conclude that the iid model is not a good fit.
  - Via R : qchisq(0.95,1) = 3.84

2.b 3-letter words

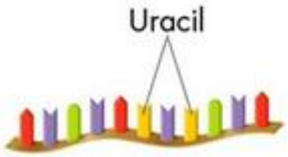
Transcription



(https://www.nature.com/scitable)

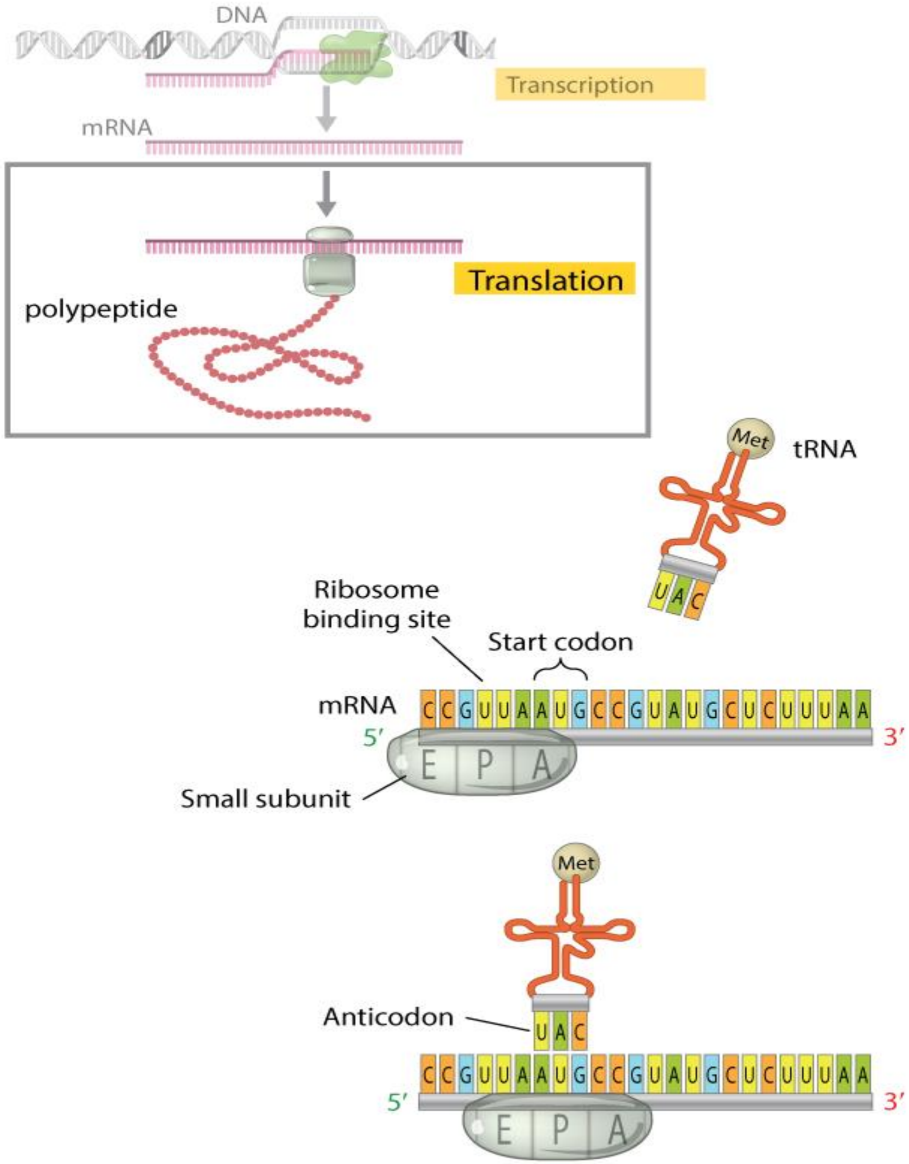
Types of RNA

- Messenger RNA (mRNA)
  - Carry a copy of the instructions from the nucleus to other parts of the cell
- Ribosomal RNA (rRNA)
  - Makes up the structure of ribosomes
- Transfer RNA (tRNA)
  - Transfers amino acids (proteins) to the ribosomes to be assembled



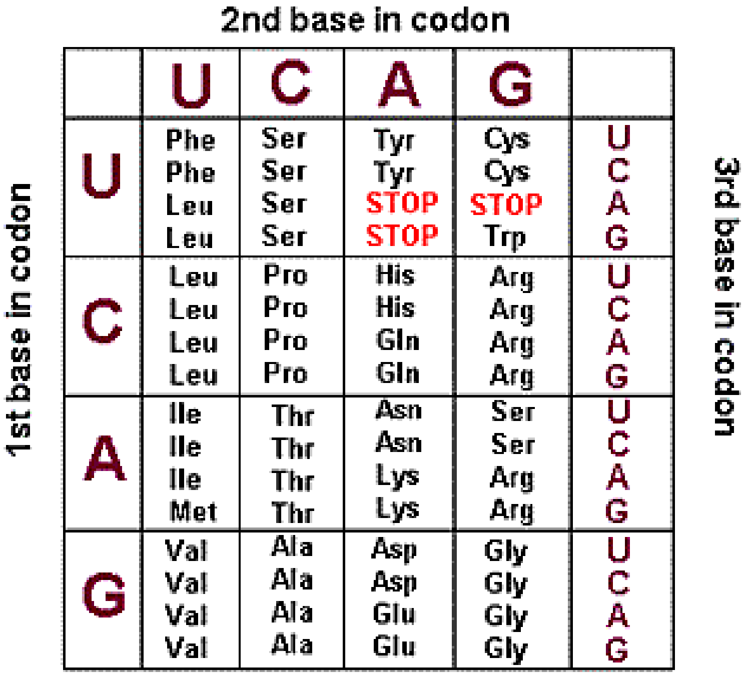
Transcription and

Translation



(https://www.nature.com/scitable)

Amino acids

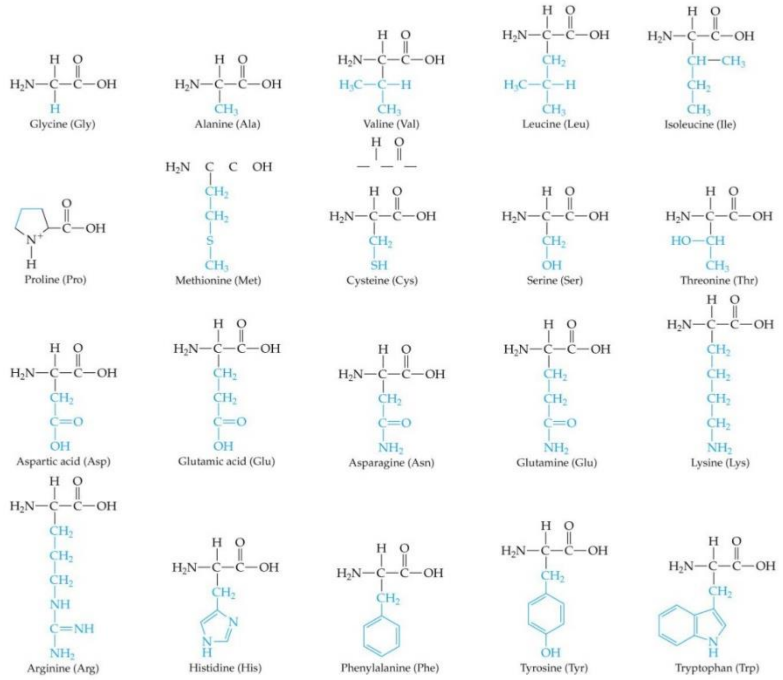


2nd base in codon

1st base in codon

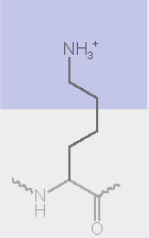
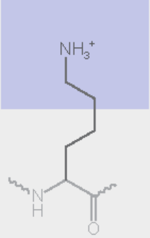
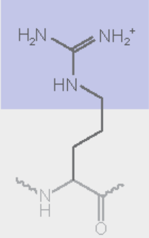
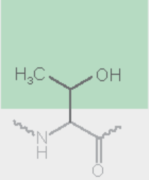
3rd base in codon

	U	C	A	G	
U	Phe	Ser	Tyr	Cys	U
	Phe	Ser	Tyr	Cys	C
	Leu	Ser	STOP	STOP	A
	Leu	Ser	STOP	Trp	G
C	Leu	Pro	His	Arg	U
	Leu	Pro	His	Arg	C
	Leu	Pro	Gln	Arg	A
	Leu	Pro	Gln	Arg	G
A	Ile	Thr	Asn	Ser	U
	Ile	Thr	Asn	Ser	C
	Ile	Thr	Lys	Arg	A
	Met	Thr	Lys	Arg	G
G	Val	Ala	Asp	Gly	U
	Val	Ala	Asp	Gly	C
	Val	Ala	Glu	Gly	A
	Val	Ala	Glu	Gly	G



- There are 61 codons that specify amino acids and three stop codons → 64 meaningful 3-words.
- Since there are 20 common amino acids, this means that most amino acids are specified by more than one codon.

Point mutations

	No mutation	Point mutations			
		Silent	Nonsense	Missense	
				conservative	non-conservative
DNA level	TTC	TTT	ATC	TCC	TGC
mRNA level	AAG	AAA	UAG	AGG	ACG
protein level	**Lys**	**Lys**	**STOP**	**Arg**	**Thr**
					

(adapted from Wikipedia)

Predicted relative frequencies

- In general, an amino acid may be coded in different ways, but **perhaps some codes have a preference**? (higher frequency?)
- For a sequence of independent bases L_1, L_2, ... , L_n the expected 3-tuple relative frequencies can be found by using the logic employed for dinucleotides we derived before
- The probability of a 3-word can be calculated as follows:

$$\mathbb{P}(L_i = r_1, L_{i+1} = r_2, L_{i+2} = r_3) = \\ \mathbb{P}(L_i = r_1)\mathbb{P}(L_{i+1} = r_2)\mathbb{P}(L_{i+2} = r_3).$$

<u>assuming the iid model</u>

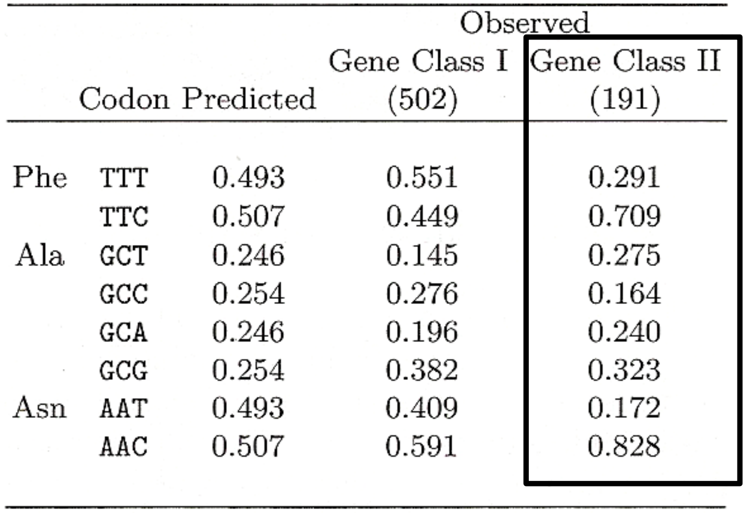
The codon adaptation index

- This provides the expected frequencies of particular codons, using the individual base frequencies. It follows that among those codons making up the amino acid Phe, the expected proportion of TTT is

$$\frac{\mathrm{P(TTT)}}{\mathrm{P(TTT)} + \mathrm{P(TTC)}}$$

- One can then compare predicted and observed triplet frequencies in coding sequences for a subset of genes and codons from f.i. E. coli.
- Médigue et al. (1991) clustered different genes based on codon usage patterns.
- For instance for Phe (TTT → UUU; TTC → UUC), the observed frequency differs considerably from the predicted frequency, when focusing on highly expressed genes (so-called “class II genes” in the work of Médigue et al. (1999)

- Figures in parentheses below each gene class show the number of genes in that class.



			Observed	
			Gene Class I	Gene Class II
	Codon	Predicted	(502)	(191)
Phe	TTT	0.493	0.551	0.291
	TTC	0.507	0.449	0.709
Ala	GCT	0.246	0.145	0.275
	GCC	0.254	0.276	0.164
	GCA	0.246	0.196	0.240
	GCG	0.254	0.382	0.323
Asn	AAT	0.493	0.409	0.172
	AAC	0.507	0.591	0.828

Class II : Highly expressed genes

Class I : Moderately expressed genes

[Gene expression analyses workflow – future class]

(Deonier et al. *Computational Genome Analysis*, 2005, Springer)

3 Types of sequence analyses

3.a Summary

Identification of Genes in a Genomic DNA Sequence

- Examples: prediction of protein coding genes
- In multicellular eukaryotes, most genes are interrupted by introns.
- The mean length of an exon is ~50 codons, but some exons are much shorter; many of the introns are extremely long, resulting in genes occupying up to several megabases of genomic DNA.
- This makes prediction of eukaryotic genes a far more complex problem than prediction of prokaryotic genes.

- A comparison of predictions generated by different programs reveals the cases where a given program performs the best and helps in achieving consistent quality of gene prediction.
- Such a comparison can be performed, for example, using the TIGR Combiner program (http://www.tigr.org/softlab), which employs a voting scheme to combine predictions of different gene-finding programs, such as GeneMark, GlimmerM, GRAIL, GenScan, and Fgenes.

Sequence similarity

- Looking for exactly the same sequence is quite straightforward.
    - One can just take the first letter of the query sequence, search for its first occurrence in the database, and then check if the second letter of the query is the same in the subject.
    - If it is indeed the same, the program could check the third letter, then the fourth, and continue this comparison to the end of the query.
    - If the second letter in the subject is different from the second letter in the query, the program should search for another occurrence of the first letter, and so on.
    - This will identify all the sequences in the database that are identical to the query sequence (or include it).
- Of course, this approach is primitive computation-wise, and there are sophisticated algorithms for text matching that do it much more efficiently

- When **comparing nucleic acid sequences**, there is very little one could do.
  - All the four nucleotides, A, T, C, and G, are found in the database with approximately the same frequencies and have roughly the same probability of mutating one into another.
  - As a result, DNA-DNA comparisons are largely based on straightforward text matching, which makes them fairly slow and not particularly sensitive, although a variety of heuristics have been developed to overcome this
  - Direct nucleotide sequence comparison is indispensable only when non-coding regions are analyzed.

- **Amino acid sequence comparisons** have several distinct advantages over nucleotide sequence comparisons:
  - Firstly, because there are 20 amino acids but only four bases, an amino acid match carries with it >4 bits of information as opposed to only two bits for a nucleotide match. → Statistical significance can be ascertained for much shorter sequences in protein comparisons than in nucleotide comparisons.
  - Secondly, because of the redundancy of the genetic code, nearly one-third of the bases in coding regions are under a weak (if any) selective pressure and represent noise → adversely affects search sensitivity.
  - Thirdly, nucleotide sequence databases are much larger than protein databases because of the vast amounts of non-coding sequences coming out of eukaryotic genome projects, and this further lowers the search sensitivity.

- Fourthly, unlike in nucleotide sequence, the likelihoods of different amino acid substitutions occurring during evolution are substantially different, and taking this into account greatly improves the performance of database search methods.

- Given all these advantages, comparisons of any coding sequences are typically carried out at the level of protein sequences:
  - Substitutions leading to similarities in physio-chemical properties of amino acids should be penalized less than a replacement of an amino acid with one that has dramatically different properties

Connecting DNA sequence and protein sequence comparative analysis:

- Even when the goal is to produce a DNA-DNA alignment (e.g. for analysis of substitutions in silent codon positions), it is usually first done with protein sequences, which are then replaced by the corresponding coding sequences.

Sequence Alignment

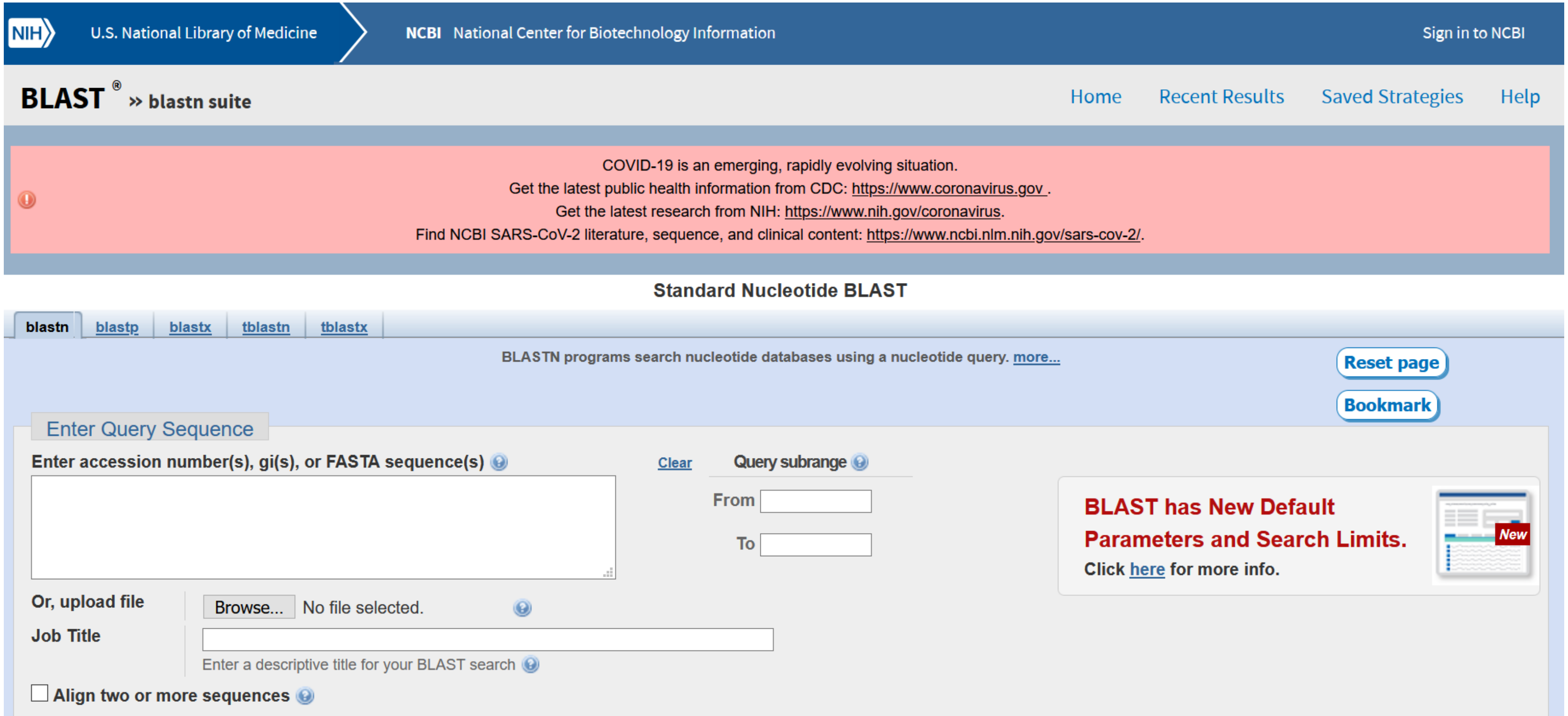
- In principle, the only way to identify homologs is
  - by aligning the query sequence against all the sequences in the database (algorithms exist to skip sequences that are obviously unrelated to the query),
  - sorting these hits based on the degree of similarity, and
  - assessing their statistical significance that is likely to be indicative of homology

- It is important to make a distinction between a **global** (i.e. full-length) **alignment** and a **local alignment**, which includes only parts of the analyzed sequences (sub-sequences).

- Although, in theory, a global alignment is best for describing relationships between sequences, in practice,
  local alignments are of more general use for two reasons:
  - Firstly, it is common that only parts of compared proteins are homologous.
  - Secondly, on many occasions, only a portion of the sequence is conserved enough to carry a detectable signal, whereas the rest have diverged beyond recognition.
- Optimal global alignment of two sequences was first realized in the **Needleman-Wunsch** algorithm, which employs dynamic programming.
- The notion of optimal local alignment (the best possible alignment of two sub-sequences from the compared sequences) and the corresponding dynamic programming algorithm were introduced by **Smith and Waterman**.

- Due to computational burden (the time and memory required to generate an optimal alignment are proportional to the product of the lengths of the compared sequences), multiple sequence alignments usually only produce approximations and do not guarantee the optimal alignment.
- Optimal alignment above:
    - is a purely formal notion, which means that, given a scoring function, the algorithm outputs the alignment with the highest possible score
    - has nothing to with statistical significance of the alignment, which has to be estimated separately
    - has nothing to do with the biological relevance of the alignment
- Examples of popular sequence data base search algorithms:
    - Smith-Waterman
    - FASTA
    - BLAST

(section ref: https://www.ncbi.nlm.nih.gov/books/NBK20261/)



NIH
U.S. National Library of Medicine
NCBI National Center for Biotechnology Information
Sign in to NCBI
BLAST ® » blastn suite
Home
Recent Results
Saved Strategies
Help

COVID-19 is an emerging, rapidly evolving situation.
Get the latest public health information from CDC: https://www.coronavirus.gov .
Get the latest research from NIH: https://www.nih.gov/coronavirus.
Find NCBI SARS-CoV-2 literature, sequence, and clinical content: https://www.ncbi.nlm.nih.gov/sars-cov-2/.

Standard Nucleotide BLAST

BLASTN programs search nucleotide databases using a nucleotide query. more...

Enter accession number(s), gi(s), or FASTA sequence(s)

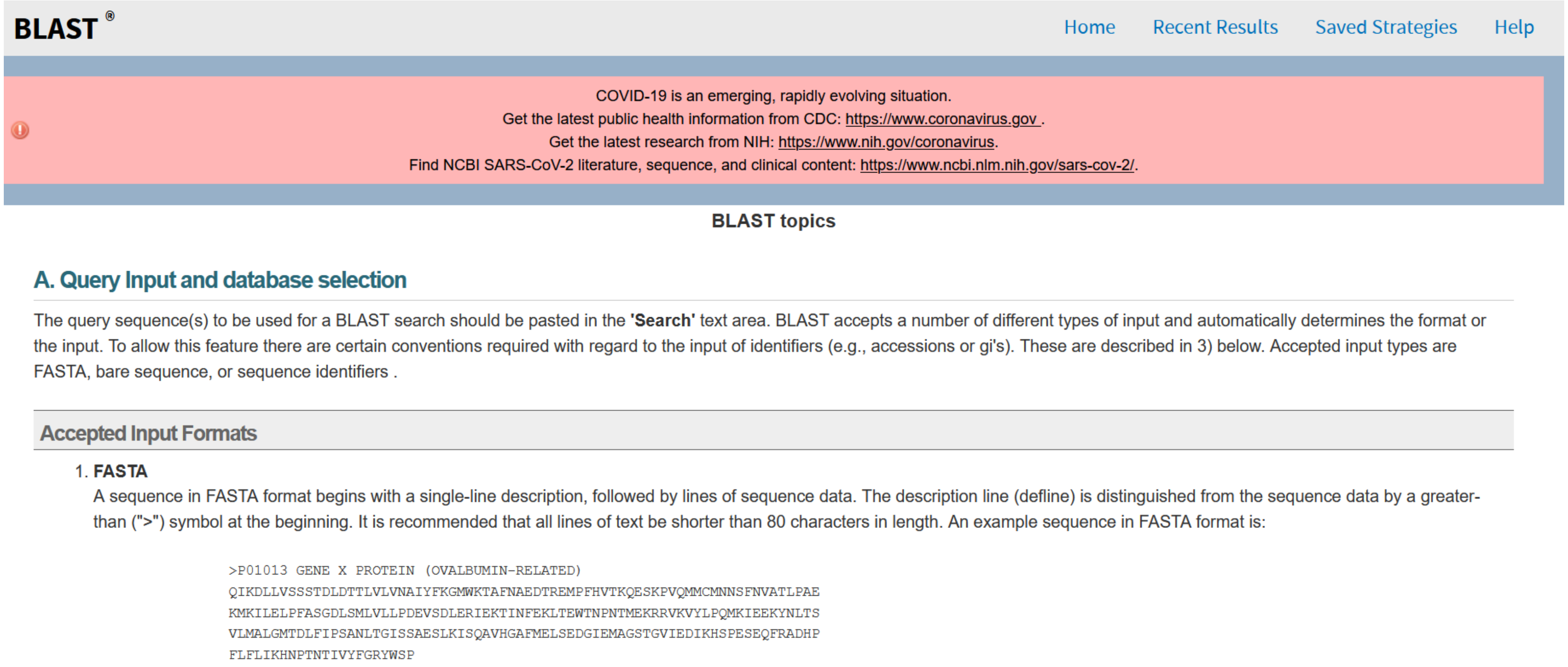
BLAST has New Default Parameters and Search Limits.
Click here for more info.

Or, upload file Browse... No file selected.

Job Title

Enter a descriptive title for your BLAST search

□ Align two or more sequences



COVID-19 is an emerging, rapidly evolving situation.
Get the latest public health information from CDC: https://www.coronavirus.gov .
Get the latest research from NIH: https://www.nih.gov/coronavirus.
Find NCBI SARS-CoV-2 literature, sequence, and clinical content: https://www.ncbi.nlm.nih.gov/sars-cov-2/.

BLAST topics

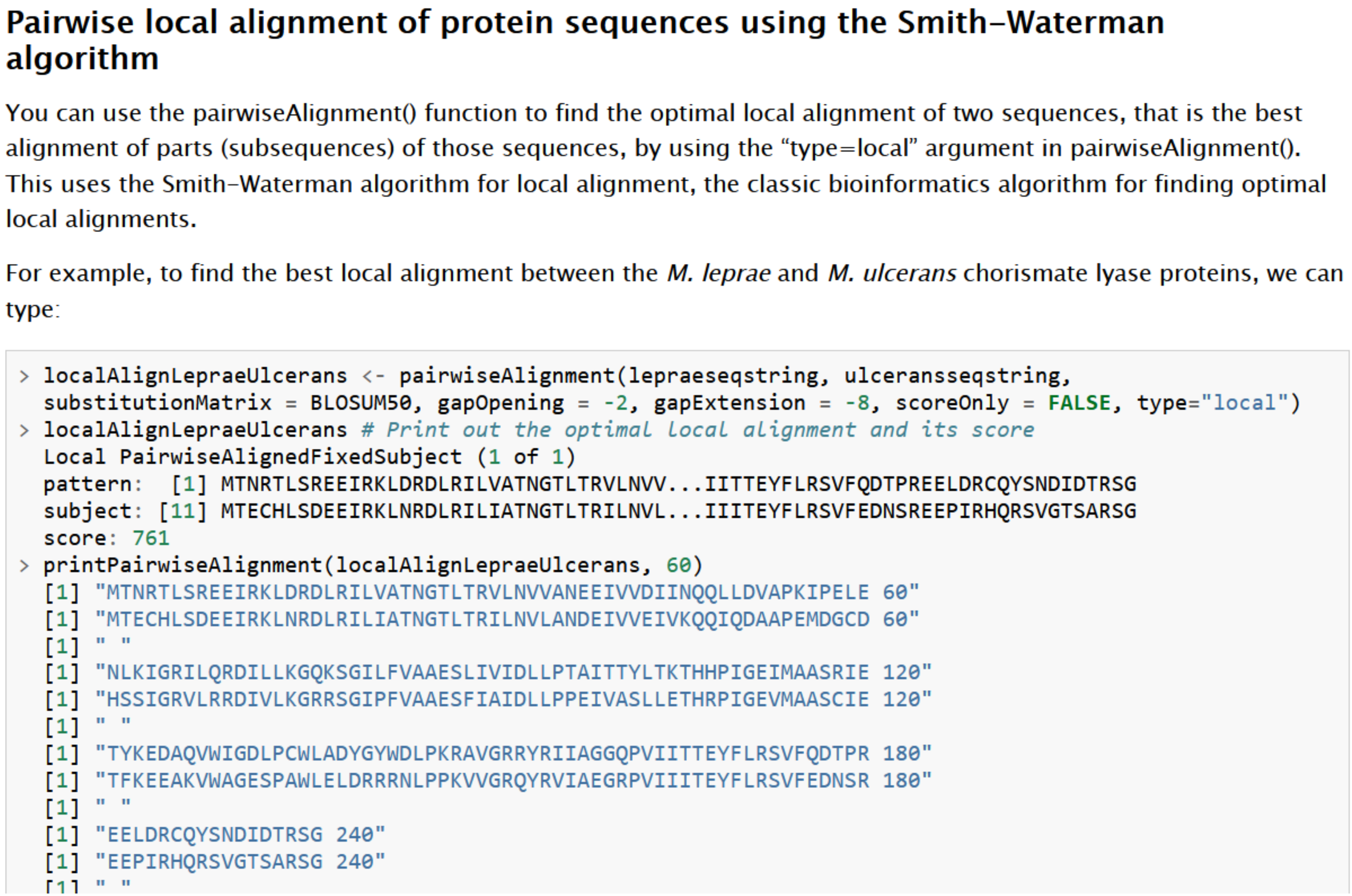
A. Query Input and database selection

The query sequence(s) to be used for a BLAST search should be pasted in the **'Search'** text area. BLAST accepts a number of different types of input and automatically determines the format or the input. To allow this feature there are certain conventions required with regard to the input of identifiers (e.g., accessions or gi's). These are described in 3) below. Accepted input types are FASTA, bare sequence, or sequence identifiers .

Accepted Input Formats

1. **FASTA**
   A sequence in FASTA format begins with a single-line description, followed by lines of sequence data. The description line (defline) is distinguished from the sequence data by a greater-than (">") symbol at the beginning. It is recommended that all lines of text be shorter than 80 characters in length. An example sequence in FASTA format is:

```
>P01013 GENE X PROTEIN (OVALBUMIN-RELATED)
QIKDLLVSSSTDLDTTLVLVNAIYFKGMWKTAFNAEDTREMPFHVTKQESKPVQMMCMNNSFNVATLPAE
KMKILELPFASGDLSMLVLLPDEVSDLERIEKTINFEKLTEWTNPNTMEKRRVKVYLPQMKIEEKYNLTS
VLMALGMTDLFIPSANLTGISSAESLKISQAVHGAFMELSEDGIEMAGSTGVIEDIKHSPESEQFRADHP
FLFLIKHNPTNTIVYFGRYWSP
```



Pairwise local alignment of protein sequences using the Smith–Waterman algorithm

You can use the pairwiseAlignment() function to find the optimal local alignment of two sequences, that is the best alignment of parts (subsequences) of those sequences, by using the “type=local” argument in pairwiseAlignment(). This uses the Smith–Waterman algorithm for local alignment, the classic bioinformatics algorithm for finding optimal local alignments.

For example, to find the best local alignment between the *M. leprae* and *M. ulcerans* chorismate lyase proteins, we can type:

```
> localAlignLepraeUlcerans <- pairwiseAlignment(lepraeseqstring, ulceransseqstring,
  substitutionMatrix = BLOSUM50, gapOpening = -2, gapExtension = -8, scoreOnly = FALSE, type="local")
> localAlignLepraeUlcerans # Print out the optimal local alignment and its score
  Local PairwiseAlignedFixedSubject (1 of 1)
  pattern:  [1] MTNRTLSREEIRKLDRDLRILVATNGTLTRVLNVV...IITTEYFLRSVFQDTPREELDRCQYSNDIDTRSG
  subject: [11] MTECHLSDEEIRKLNRDLRILIATNGTLTRILNVL...IIITEYFLRSVFEDNSREEPIRHQRSVGTSARSG
  score: 761
> printPairwiseAlignment(localAlignLepraeUlcerans, 60)
  [1] "MTNRTLSREEIRKLDRDLRILVATNGTLTRVLNVVANEEIVVDIINQQLLDVAPKIPELE 60"
  [1] "MTECHLSDEEIRKLNRDLRILIATNGTLTRILNVLANDEIVVEIVKQQIQDAAPEMDGCD 60"
  [1] " "
  [1] "NLKIGRILQRDILLKGQKSGILFVAAESLIVIDLLPTAITTYLTKTHHPIGEIMAASRIE 120"
  [1] "HSSIGRVLRRDIVLKGRRSGIPFVAAESFIAIDLLPPEIVASLLETHRPIGEVMAASCIE 120"
  [1] " "
  [1] "TYKEDAQVWIGDLPCWLADYGYWDLPKRAVGRRYRIIAGGQPVIITTEYFLRSVFQDTPR 180"
  [1] "TFKEEAKVWAGESPAWLELDRRRNLPPKVVGRQYRVIAEGRPVIIITEYFLRSVFEDNSR 180"
  [1] " "
  [1] "EELDRCQYSNDIDTRSG 240"
  [1] "EEPIRHQRSVGTSARSG 240"
  [1] " "
```

(https://a-little-book-of-r-for-bioinformatics.readthedocs.io/en/latest/index.html)

Links and further reading

as adapted from

https://a-little-book-of-r-for-bioinformatics.readthedocs.io/en/latest/index.html

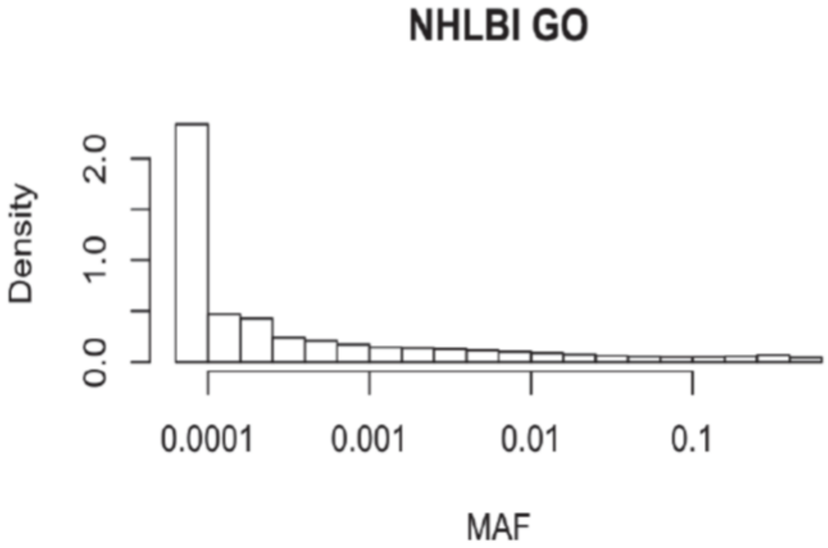
- For background reading on sequence alignment: Chapter 3 of **Introduction to Computational Genomics: a case studies approach** by Cristianini and Hahn (Cambridge University Press; www.computational-genomics.net/book/).
- There is also a very nice chapter on "Analyzing Sequences", which includes examples of using SeqinR and Biostrings for sequence analysis, as well as details on how to implement algorithms such as Needleman-Wunsch and Smith-Waterman in R yourself, in the book **Applied statistics for bioinformatics using R** by Krijnen (available online at cran.r-project.org/doc/contrib/Krijnen-IntroBioInfStatistics.pdf).
- For more information on and examples using the Biostrings package, see the Biostrings documentation at http://www.bioconductor.org/packages/release/bioc/html/Biostrings.html.

3.b Rare variants association studies

(slide Doug Brutlag 2010)

Why studying rare variants?

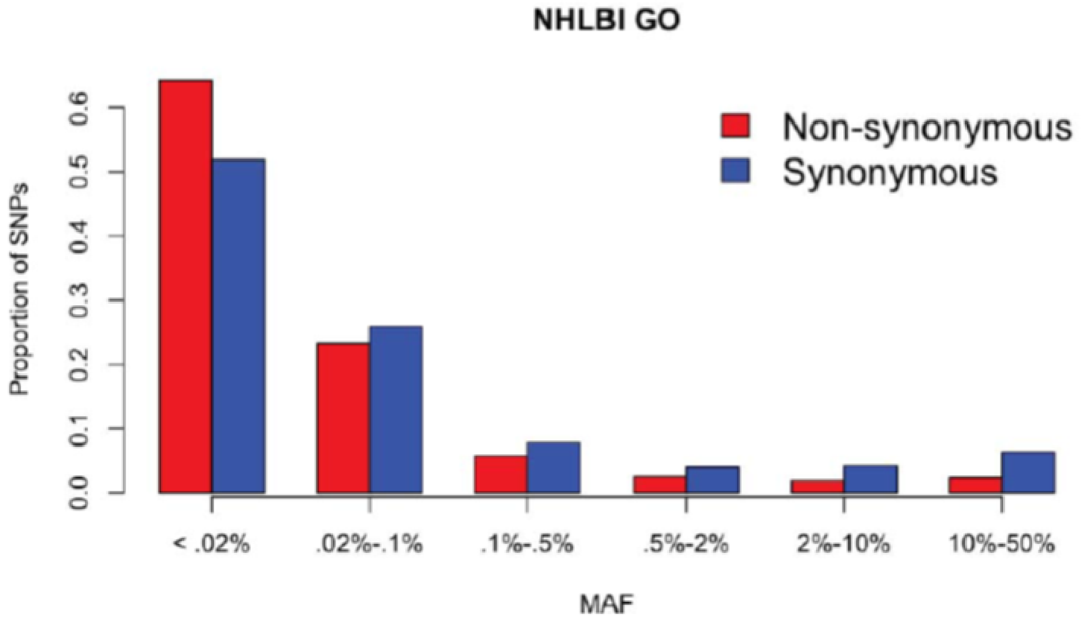
- Most human variants are rare



(refs: slides Fan Li 2016)

Why studying rare variants?

- Functional variants tend to be rare



(refs: slides Fan Li 2016)

Analytic challenges

- A variant – genetic association test implies filling in the table below and performing a chi-squared test for independence between rows and columns

	AA	**Aa**	**aa**
Cases			
Controls			

Sum of entries = cases+controls

How many observations do you expect to have two copies of a rare allele (say MAF = 0.001)?

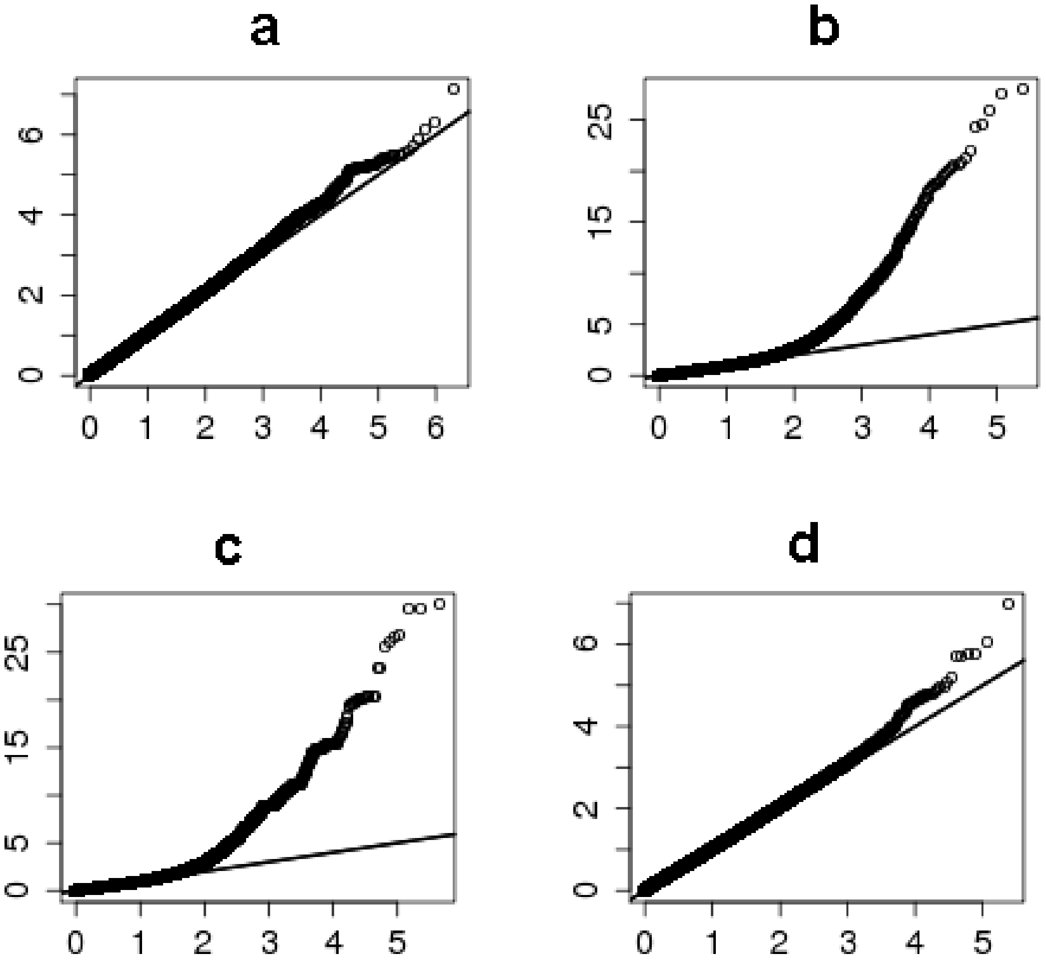
- **In a chi-squared test of independence setting:**
  - When MAF <<< 0.01 then some cells above will be sparse and large-sample statistics (classic chi-squared tests of independence) will no longer be valid.
  - This is the case when there are less than 5 observations in a cell

$$X^2 = \sum_{all\ cells\ i} \frac{(O_i - E_i)^2}{E_!}$$ (contrasting Observed minus Expected)

- **In a regression framework:**
  - The minimum number of observations per independent variable should be 10, using a rough rule of thumb (guideline provided by Hosmer and Lemeshow - Applied Logistic Regression)

Increased false positive rates

Q-Q plots from GWAS data, unpublished



N=~2500

MAF>0.03

N=~2500

MAF<0.03

N=~2500

MAF<0.03

Permuted

N=50000

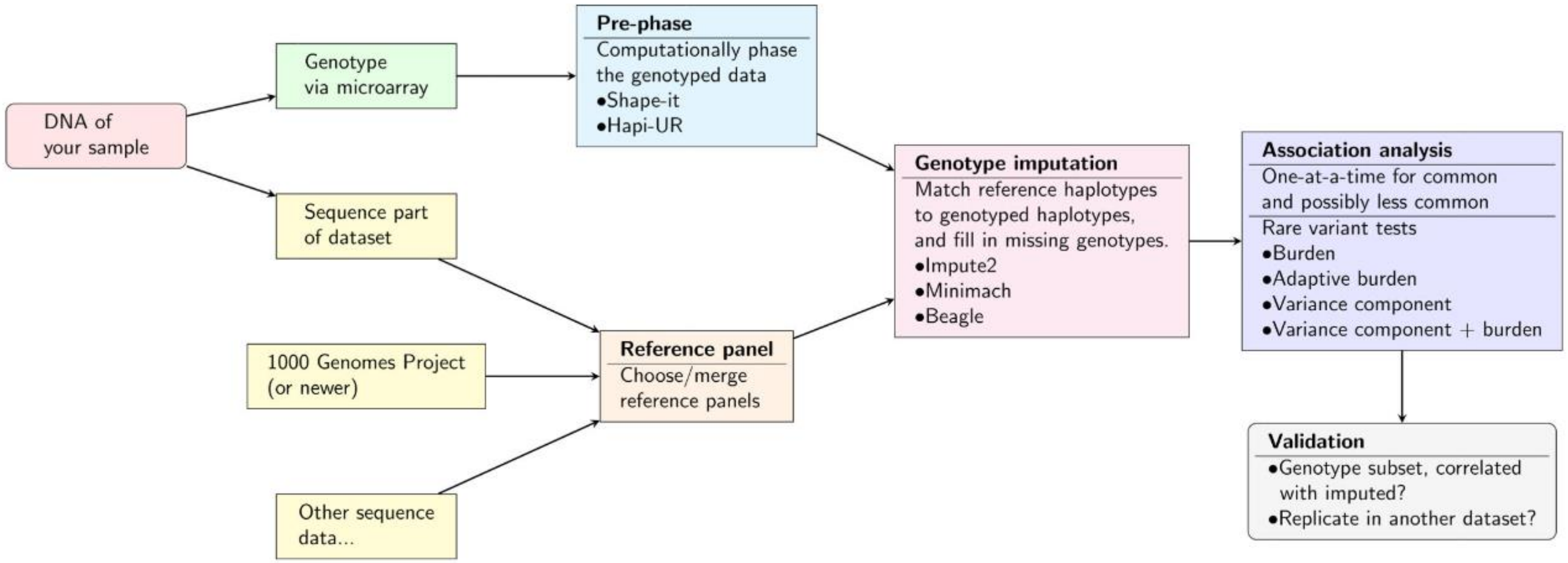
MAF<0.03

Bootstrapped

Remediation: do not look at a single variant at a time, but collapse

- Rationale for aggregation tests
  - At α= 0.05, Bonferroni correction will lead to too stringent thresholds (accounting for all variants included in the study)
  - One needs VERY LARGE samples sizes in order to be able to reach that level, even if you find "the variant".
- Remedy = aggregate / pool variants
  - Requires specification of a so-called "region of interest" (ROI)
  - A ROI can be anything really:
    - Gene
    - Locus
    - Intra-genic area
    - Functional set [see also Homework 1 assignment: paper style]

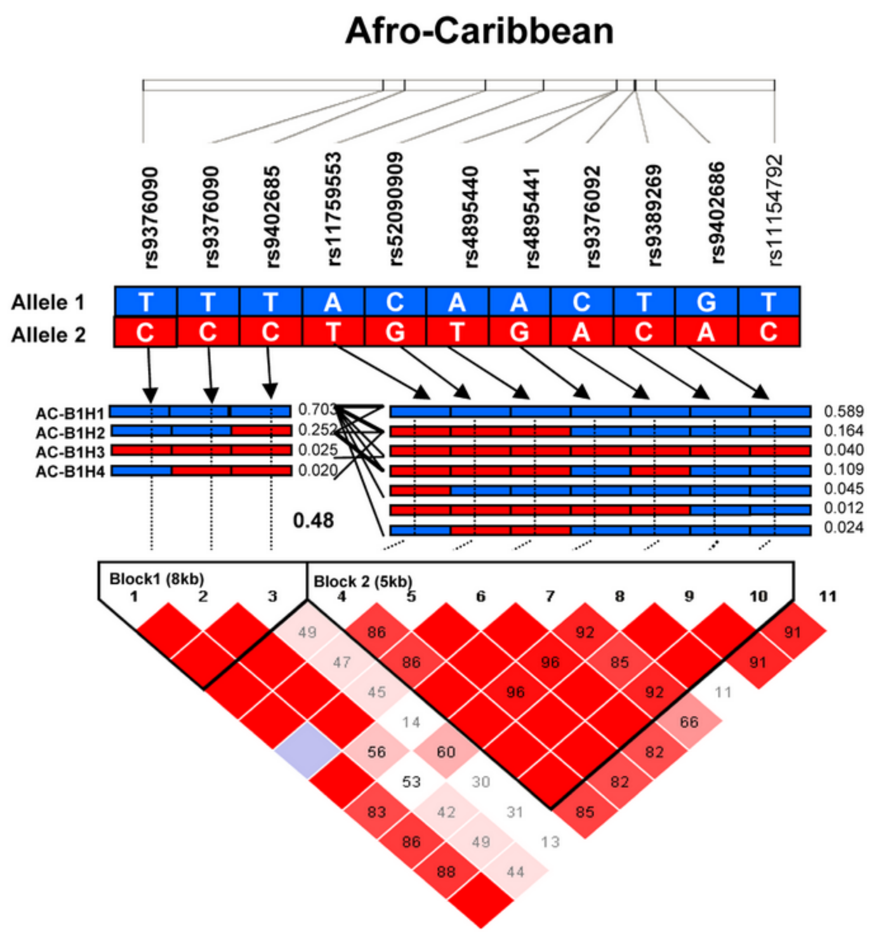
Overall process for imputing and analyzing rare variants


DNA of your sample
Genotype via microarray
Sequence part of dataset
1000 Genomes Project (or newer)
Other sequence data...
Pre-phase
Computationally phase the genotyped data
•Shape-it
•Hapi-UR
Reference panel
Choose/merge reference panels
Genotype imputation
Match reference haplotypes to genotyped haplotypes, and fill in missing genotypes.
•Impute2
•Minimach
•Beagle
Association analysis
One-at-a-time for common and possibly less common
Rare variant tests
•Burden
•Adaptive burden
•Variance component
•Variance component + burden
Validation
•Genotype subset, correlated with imputed?
•Replicate in another dataset?

A flowchart describing the steps in imputing rare variants into genome-wide SNP datasets

(Hoffman and Witte, 2015)

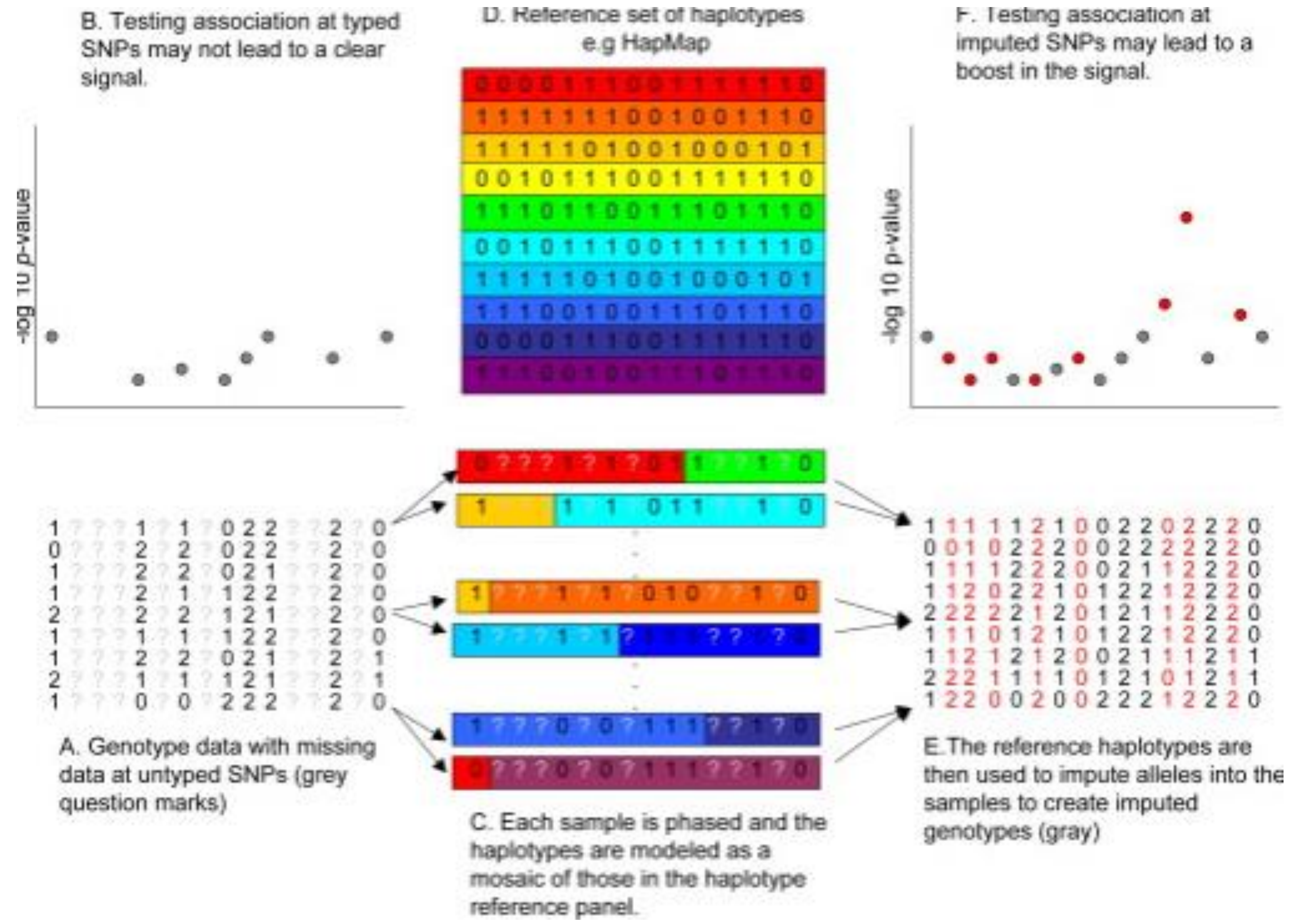
Haplotypes



doi: https://doi.org/10.1371/journal.pone.0004218.g001

- When haplotypes are defined **in relation to SNPs**: a haplotype is a set of SNPs found to be statistically associated on a single chromatid, I.e. adjacent SNPs that are inherited together on the basis of linkage disequilibrium.
- For a more detailed definition is the International HapMap Project

Imputation in common variant GWAS



(book exert: Zeggini and Morris 2011- Analysis of Complex Disease Association Studies)

Imputation in common variant GWAS

- Two easy ways dealing with uncertain genotypes (genotype coding 0, 1, 2):

  - Genotype Calling:

    Choose the most likely genotype and continue as if it is true
    (p11=10%, p12=20% p22=70% => G=2)
  - Mean genotype:

    Use the weighted average genotype
    (p11=10%, p12=20% p22=70% => G=1.6)

(ref: slides E Bouzigon 2020)

- Haplotype imputation services
  - EAGLE - https://data.broadinstitute.org/alkesgroup/Eagle/
  - SHAPEIT - https://jmarchini.org/shapeit3/

Popular imputation programs

- **IMPUTE 5** (Rubinacci S et al 2019, Genotype imputation using the Positional Burrows Wheeler Transform bioRxiv) https://jmarchini.org/impute5/

- **Minimac4** (Das S et al. Nat Genet 2016) https://github.com/statgen/Minimac4

Hidden Markov Model - based

- **HLA allelic imputation programs**
  - HIBAG: https://bioconductor.org/packages/release/bioc/html/HIBAG.html
  - HLA*IMP:02: https://oxfordhla.well.ox.ac.uk/hla/static/tutorial.pdf
  - SNP2HLA: http://software.broadinstitute.org/mpg/snp2hla/

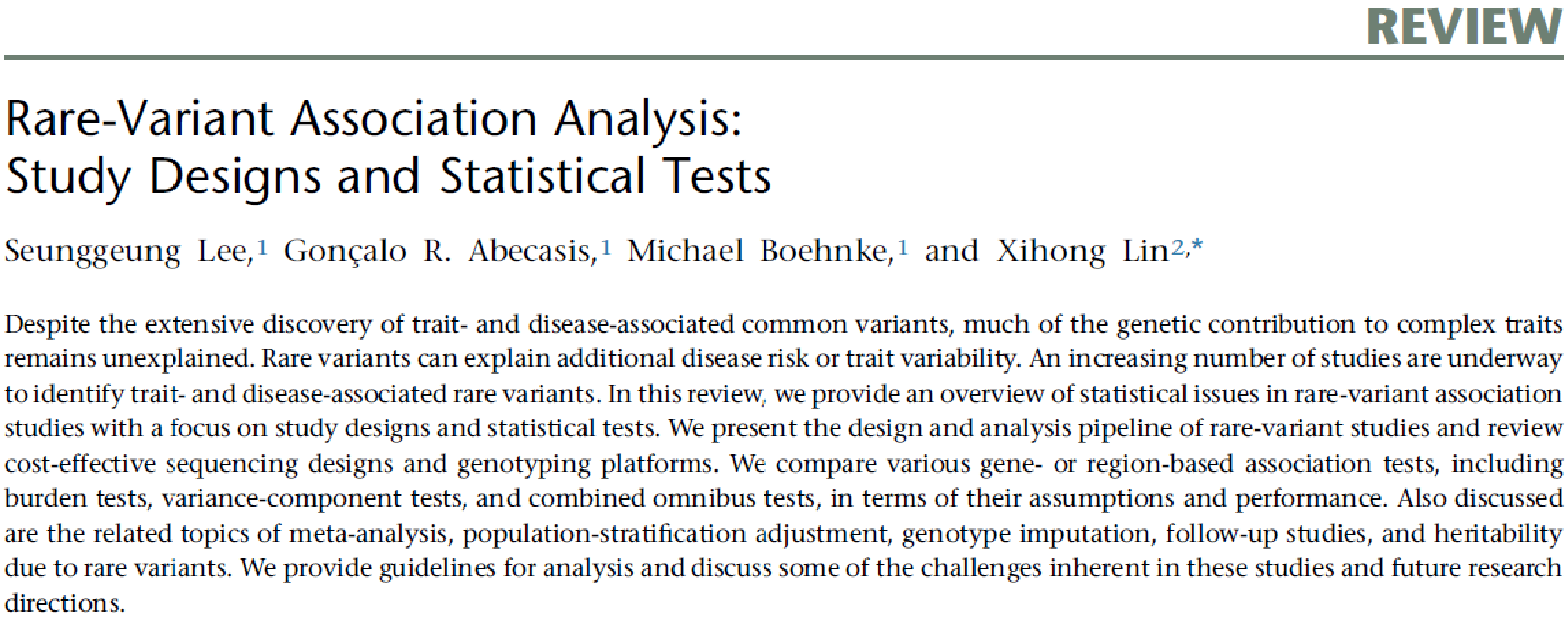
Popular imputation services

- **Michigan Imputation Server** (a free genotype imputation service using Minimac4)
  https://imputationserver.sph.umich.edu

- **Sanger Imputation Server** (using PBWT/IMPUTE 5)
  https://www.sanger.ac.uk/tool/sanger-imputation-service/

Additional analysis considerations

- Variable selection
- Ways to define regions of interest and to test them



REVIEW

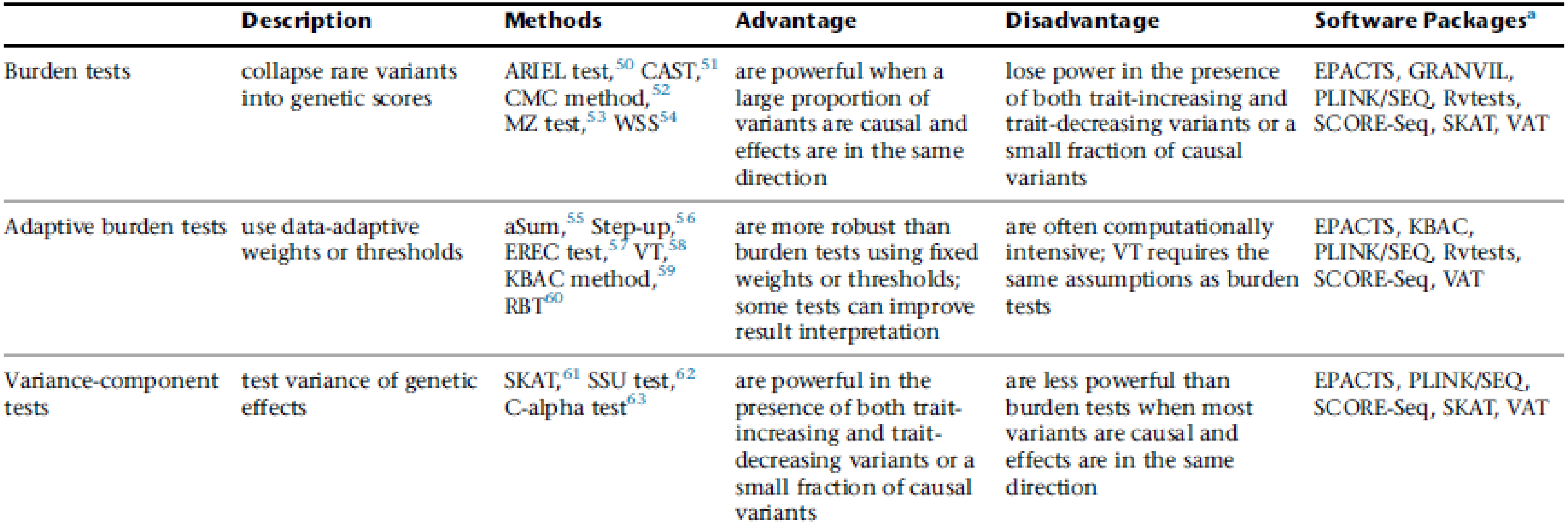
Rare-Variant Association Analysis: Study Designs and Statistical Tests

Seunggeung Lee,[1] Gonçalo R. Abecasis,[1] Michael Boehnke,[1] and Xihong Lin[2,*]

Despite the extensive discovery of trait- and disease-associated common variants, much of the genetic contribution to complex traits remains unexplained. Rare variants can explain additional disease risk or trait variability. An increasing number of studies are underway to identify trait- and disease-associated rare variants. In this review, we provide an overview of statistical issues in rare-variant association studies with a focus on study designs and statistical tests. We present the design and analysis pipeline of rare-variant studies and review cost-effective sequencing designs and genotyping platforms. We compare various gene- or region-based association tests, including burden tests, variance-component tests, and combined omnibus tests, in terms of their assumptions and performance. Also discussed are the related topics of meta-analysis, population-stratification adjustment, genotype imputation, follow-up studies, and heritability due to rare variants. We provide guidelines for analysis and discuss some of the challenges inherent in these studies and future research directions.

(Lee et al. 2014)

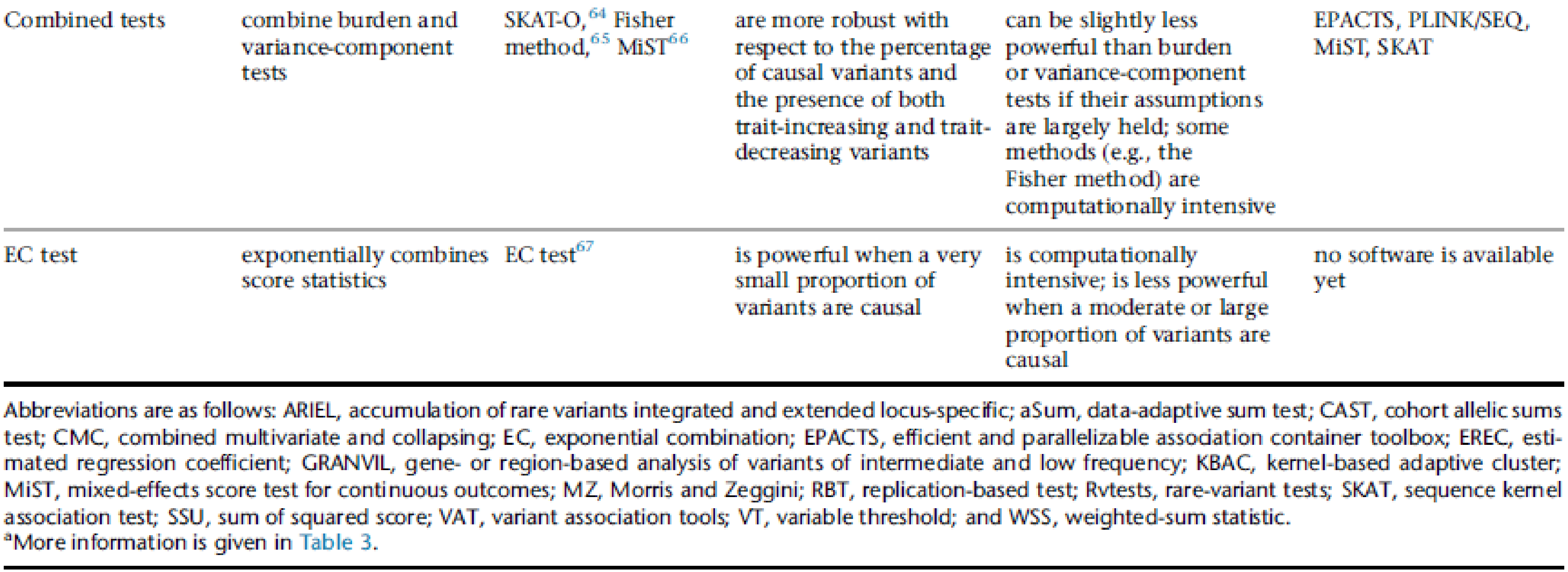
An abundance of tests



	Description	Methods	Advantage	Disadvantage	Software Packages[a]
Burden tests	collapse rare variants into genetic scores	ARIEL test,[50] CAST,[51] CMC method,[52] MZ test,[53] WSS[54]	are powerful when a large proportion of variants are causal and effects are in the same direction	lose power in the presence of both trait-increasing and trait-decreasing variants or a small fraction of causal variants	EPACTS, GRANVIL, PLINK/SEQ, Rvtests, SCORE-Seq, SKAT, VAT
Adaptive burden tests	use data-adaptive weights or thresholds	aSum,[55] Step-up,[56] EREC test,[57] VT,[58] KBAC method,[59] RBT[60]	are more robust than burden tests using fixed weights or thresholds; some tests can improve result interpretation	are often computationally intensive; VT requires the same assumptions as burden tests	EPACTS, KBAC, PLINK/SEQ, Rvtests, SCORE-Seq, VAT
Variance-component tests	test variance of genetic effects	SKAT,[61] SSU test,[62] C-alpha test[63]	are powerful in the presence of both trait-increasing and trait-decreasing variants or a small fraction of causal variants	are less powerful than burden tests when most variants are causal and effects are in the same direction	EPACTS, PLINK/SEQ, SCORE-Seq, SKAT, VAT

(Lee et al. 2014)

An abundance of tests

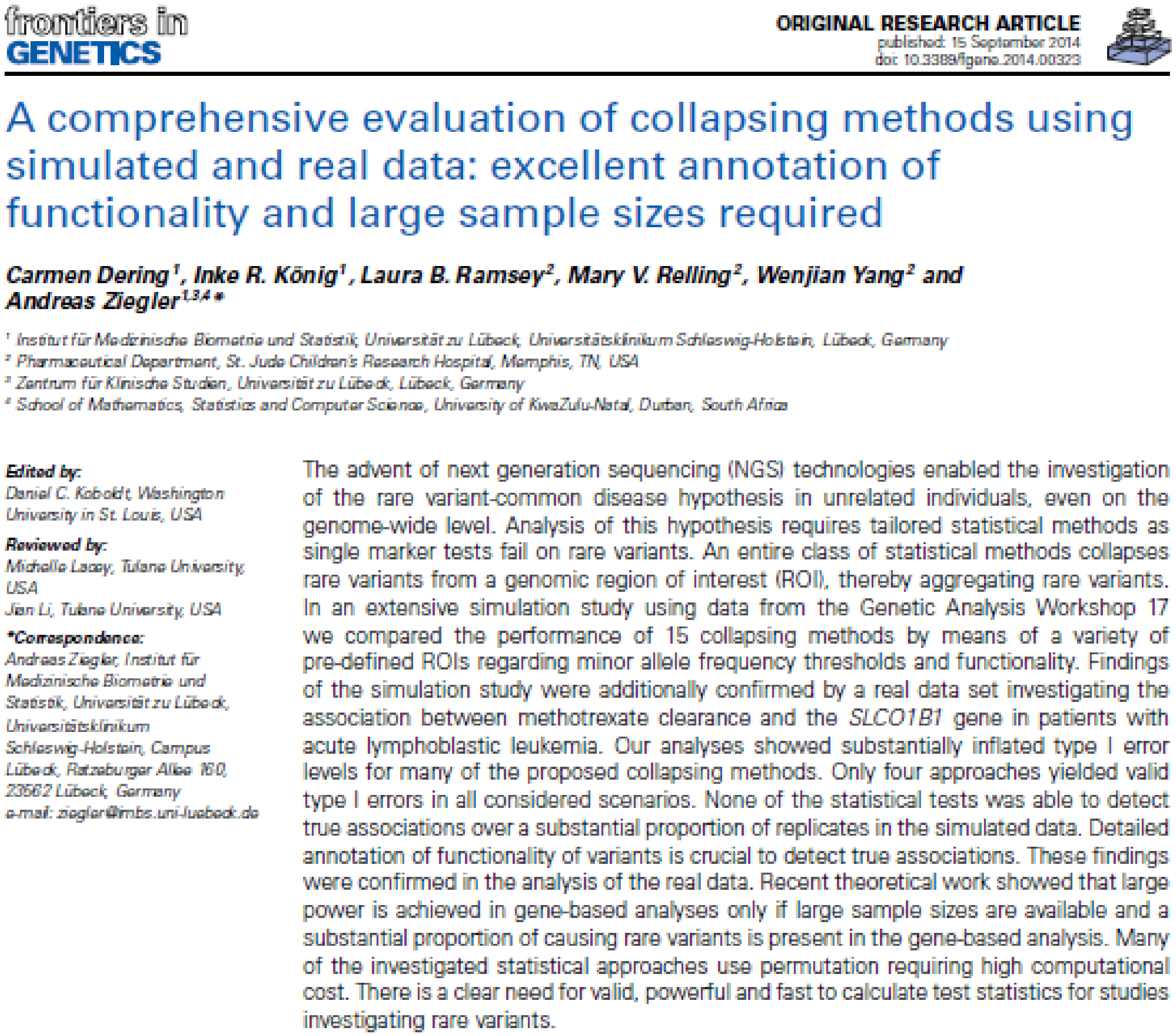


Combined tests	combine burden and variance-component tests	SKAT-O,[64] Fisher method,[65] MiST[66]	are more robust with respect to the percentage of causal variants and the presence of both trait-increasing and trait-decreasing variants	can be slightly less powerful than burden or variance-component tests if their assumptions are largely held; some methods (e.g., the Fisher method) are computationally intensive	EPACTS, PLINK/SEQ, MiST, SKAT
EC test	exponentially combines score statistics	EC test[67]	is powerful when a very small proportion of variants are causal	is computationally intensive; is less powerful when a moderate or large proportion of variants are causal	no software is available yet

Abbreviations are as follows: ARIEL, accumulation of rare variants integrated and extended locus-specific; aSum, data-adaptive sum test; CAST, cohort allelic sums test; CMC, combined multivariate and collapsing; EC, exponential combination; EPACTS, efficient and parallelizable association container toolbox; EREC, estimated regression coefficient; GRANVIL, gene- or region-based analysis of variants of intermediate and low frequency; KBAC, kernel-based adaptive cluster; MiST, mixed-effects score test for continuous outcomes; MZ, Morris and Zeggini; RBT, replication-based test; Rvtests, rare-variant tests; SKAT, sequence kernel association test; SSU, sum of squared score; VAT, variant association tools; VT, variable threshold; and WSS, weighted-sum statistic.

[a]More information is given in Table 3.

(Lee et al. 2014)



frontiers in GENETICS

ORIGINAL RESEARCH ARTICLE
published: 15 September 2014
doi: 10.3389/fgene.2014.00323

A comprehensive evaluation of collapsing methods using simulated and real data: excellent annotation of functionality and large sample sizes required

Carmen Dering[1], Inke R. König[1], Laura B. Ramsey[2], Mary V. Relling[2], Wenjian Yang[2] and Andreas Ziegler[1,3,4]*

[1] *Institut für Medizinische Biometrie und Statistik, Universität zu Lübeck, Universitätsklinikum Schleswig-Holstein, Lübeck, Germany*
[2] *Pharmaceutical Department, St. Jude Children's Research Hospital, Memphis, TN, USA*
[3] *Zentrum für Klinische Studien, Universität zu Lübeck, Lübeck, Germany*
[4] *School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Durban, South Africa*

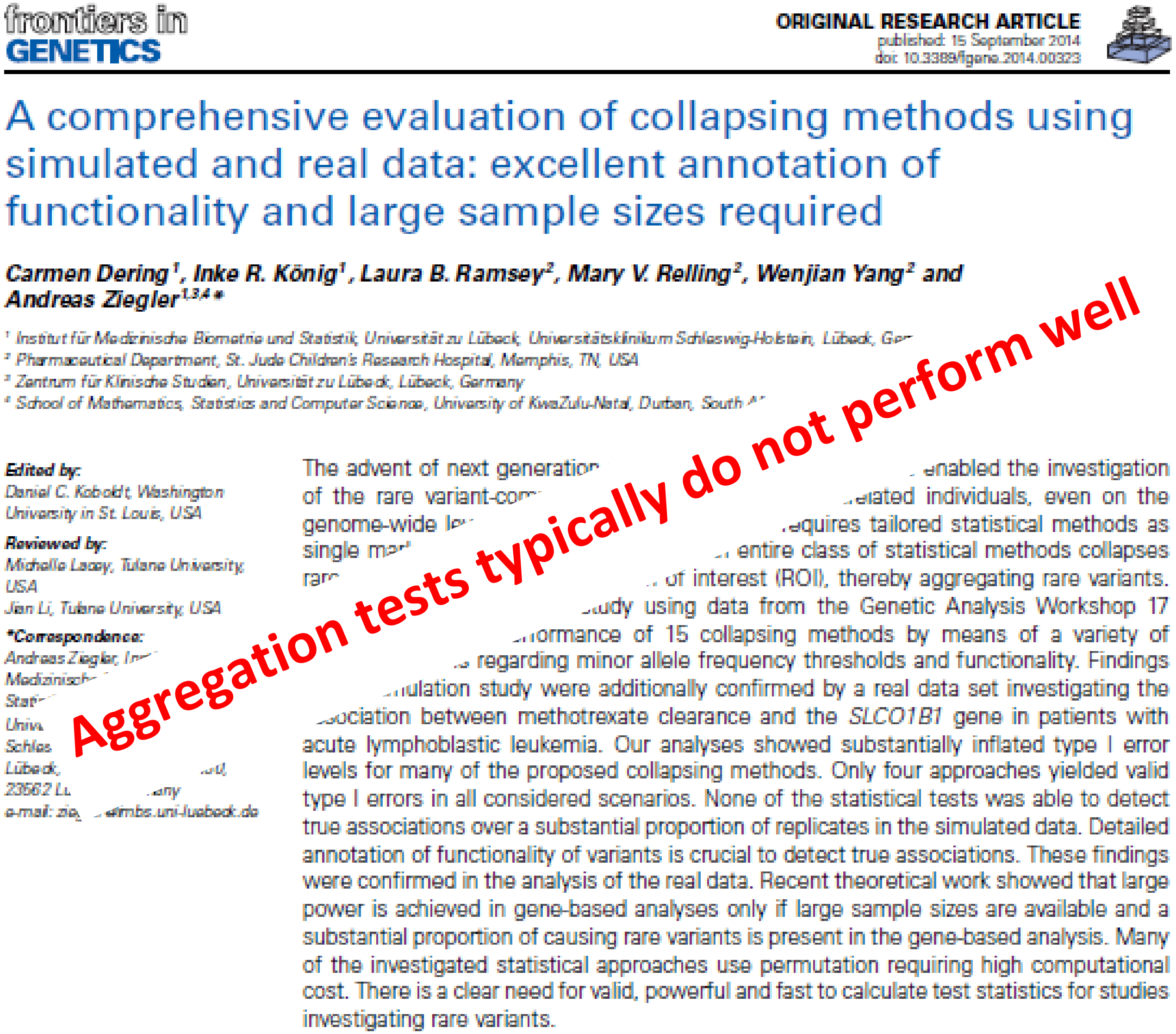
Edited by:
Daniel C. Koboldt, Washington University in St. Louis, USA

Reviewed by:
Michelle Lacey, Tulane University, USA
Jian Li, Tulane University, USA

***Correspondence:**
Andreas Ziegler, Institut für Medizinische Biometrie und Statistik, Universität zu Lübeck, Universitätsklinikum Schleswig-Holstein, Campus Lübeck, Ratzeburger Allee 160, 23562 Lübeck, Germany
e-mail: ziegler@imbs.uni-luebeck.de

The advent of next generation sequencing (NGS) technologies enabled the investigation of the rare variant-common disease hypothesis in unrelated individuals, even on the genome-wide level. Analysis of this hypothesis requires tailored statistical methods as single marker tests fail on rare variants. An entire class of statistical methods collapses rare variants from a genomic region of interest (ROI), thereby aggregating rare variants. In an extensive simulation study using data from the Genetic Analysis Workshop 17 we compared the performance of 15 collapsing methods by means of a variety of pre-defined ROIs regarding minor allele frequency thresholds and functionality. Findings of the simulation study were additionally confirmed by a real data set investigating the association between methotrexate clearance and the *SLCO1B1* gene in patients with acute lymphoblastic leukemia. Our analyses showed substantially inflated type I error levels for many of the proposed collapsing methods. Only four approaches yielded valid type I errors in all considered scenarios. None of the statistical tests was able to detect true associations over a substantial proportion of replicates in the simulated data. Detailed annotation of functionality of variants is crucial to detect true associations. These findings were confirmed in the analysis of the real data. Recent theoretical work showed that large power is achieved in gene-based analyses only if large sample sizes are available and a substantial proportion of causing rare variants is present in the gene-based analysis. Many of the investigated statistical approaches use permutation requiring high computational cost. There is a clear need for valid, powerful and fast to calculate test statistics for studies investigating rare variants.

(Dering et al. 2014)



frontiers in GENETICS

ORIGINAL RESEARCH ARTICLE
published: 15 September 2014
doi: 10.3389/fgene.2014.00323

A comprehensive evaluation of collapsing methods using simulated and real data: excellent annotation of functionality and large sample sizes required

Carmen Dering[1], Inke R. König[1], Laura B. Ramsey[2], Mary V. Relling[2], Wenjian Yang[2] and Andreas Ziegler[1,3,4]*

[1] Institut für Medizinische Biometrie und Statistik, Universität zu Lübeck, Universitätsklinikum Schleswig-Holstein, Lübeck, Ger
[2] Pharmaceutical Department, St. Jude Children's Research Hospital, Memphis, TN, USA
[3] Zentrum für Klinische Studien, Universität zu Lübeck, Lübeck, Germany
[4] School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Durban, South

Edited by:
Daniel C. Koboldt, Washington University in St. Louis, USA

Reviewed by:
Michelle Lacey, Tulane University, USA
Jian Li, Tulane University, USA

***Correspondence:**
Andreas Ziegler, In
Medizinisch
Stat
Univ
Schles
Lübeck,
23562 L
e-mail: zie imbs.uni-luebeck.de

The advent of next generation enabled the investigation of the rare variant-com elated individuals, even on the genome-wide le equires tailored statistical methods as single ma entire class of statistical methods collapses rar of interest (ROI), thereby aggregating rare variants. udy using data from the Genetic Analysis Workshop 17 formance of 15 collapsing methods by means of a variety of regarding minor allele frequency thresholds and functionality. Findings ulation study were additionally confirmed by a real data set investigating the ssociation between methotrexate clearance and the *SLCO1B1* gene in patients with acute lymphoblastic leukemia. Our analyses showed substantially inflated type I error levels for many of the proposed collapsing methods. Only four approaches yielded valid type I errors in all considered scenarios. None of the statistical tests was able to detect true associations over a substantial proportion of replicates in the simulated data. Detailed annotation of functionality of variants is crucial to detect true associations. These findings were confirmed in the analysis of the real data. Recent theoretical work showed that large power is achieved in gene-based analyses only if large sample sizes are available and a substantial proportion of causing rare variants is present in the gene-based analysis. Many of the investigated statistical approaches use permutation requiring high computational cost. There is a clear need for valid, powerful and fast to calculate test statistics for studies investigating rare variants.

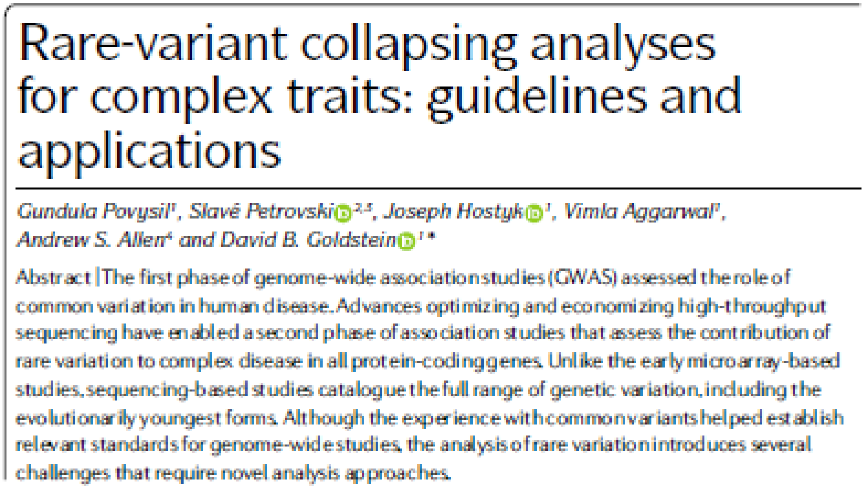
Aggregation tests typically do not perform well

(Dering et al. 2014)

Questions?

Main supporting doc to this class (complementing course slides)

☑



Rare-variant collapsing analyses for complex traits: guidelines and applications

Gundula Povysil[1], Slavé Petrovski[2,3], Joseph Hostyk[1], Vimla Aggarwal[1], Andrew S. Allen[4] and David B. Goldstein[1]*

Abstract | The first phase of genome-wide association studies (GWAS) assessed the role of common variation in human disease. Advances optimizing and economizing high-throughput sequencing have enabled a second phase of association studies that assess the contribution of rare variation to complex disease in all protein-coding genes. Unlike the early microarray-based studies, sequencing-based studies catalogue the full range of genetic variation, including the evolutionarily youngest forms. Although the experience with common variants helped establish relevant standards for genome-wide studies, the analysis of rare variation introduces several challenges that require novel analysis approaches.

Nature reviews Genetics 2019; 20:747