Basics of Markov chains

Samuel S. Shepard

Outline

- Discuss basic Markov models.
- Discuss a few applications.

Markov Chains

- Markov models are the basis for many gene prediction programs such as GeneMark.
  - GeneMark uses Hidden Markov models.
  - We developed a sequence prediction algorithm based on Markov chains called BAMM.
- Can apply to any sequence of information: nucleotide, amino acid, etc.

Markov models 101

- Markov models can be used to both *generate* & *classify* sequence data.
- The sequence frequency information must analyzed first, then it can be used.
- Let's get a feel for Markov models with an analogy..

Fill in the blank…

q r t e d u o s f m a y

i. th
ii. gol
iii. fluff
iv. dinosau

Fill in the blank…

q r t e d u o s f m a y

i. the
ii. gol
iii. fluff
iv. dinosau

Fill in the blank…

i. th**e**
ii. gol**d**
iii. fluff
iv. dinosau

q r t e d u o s f m a y

Fill in the blank…

i. th**e**
ii. gol**d**
iii. fluff**y**
iv. dinosau[ ]

q r t e d u o s f m a y

Fill in the blank…

q r t e d u o s f m a y

i. th**e**
ii. gol**d**
iii. fluff**y**
iv. dinosaur

Markov chain fundamentals

- The number of “letters” remembered by the Markov chain are known as its order.

- Markov chains can *generate* the next letter in the sequence based on the model frequencies.

Markov chain fundamentals

- Longer words like “dinosaur” were easier to guess than shorter ones like “gold” (could have been “golf”).

- Larger order Markov chains generally do *better* prediction.

Markov chains for Prediction

- Earlier you became human Markov models to generate words using your knowledge of *English.*

- *What if I only gave you a sequence of characters & wanted to know which language it was???*

Español or English?

tsnottearittheysaidto
oneanothertetsdecid
ebylotwhowillgetitthi
shappenedthatthescr
ipturemightbefulfille
dthatsaidtheydivided
myclothesamongthe
mandcastlotsformyg
armentsothisiswhatt
hesoldi

?
?
?

idamossedijeronunosa
otrosechemossuertesp
araveraquienletocayasi
lohicieronlossoldadose
stosucedioparaquesec
umplieralaescrituraque
diceserepartieronentre
ellosmimantoysobremir
opaecharonsuer

Español or English?

tsnottearit**they**saidto
oneanothertetsdecid
ebylotwhowillgetitthi
shappenedthatthescr
ipturemightbefulfille
dthatsaid**they**divided
myclothesamongthe
mandcastlotsformyg
armentso**this**iswhatt
hesoldi

?
?
?

idamossedijeronunosa
otrosechemossuertesp
araveraquienletocayasi
lohicieron**los**soldadose
stosucedioparaquesec
umplier**la**escrituraque
diceserepartieronentre
ellosmimantoysobre**mi**
ropaecharonsuer

Doing Prediction

- Frequent patterns (words) help you see the *language* or model classification.
- It's difficult to make sense of the sentences without knowing where to start reading.

Help with Reading Frame

tsnottearittheysaidtooneanother**L**e
tsdecidebylotwhowillgetit**T**hishapp
enedthatthescripturemightbefulfill
edthatsaid**T**heydividedmyclothesa
mongthemandcastlotsformygarme
nt**S**othisiswhatthesoldi

Training for the Unknown

- Suppose you **don't know** either language.
- How do you do prediction without learning the meaning of every word in each language?

...beschlossensiediesesuntergewandwollen…

Training a Model

- You'd read lots of books in each language & learn the frequent words!

Example Training

- BAMM project used **6 million** nucleotides of exons and introns each.
- **3 million** bases are used to test prediction.

Markov chain types

- *Inhomogeneous* Markov models can "see" multiple reading frames.
  - Helps detect coding sequences.
  - More accurate.

- *Homogeneous* Markov chains don't care.

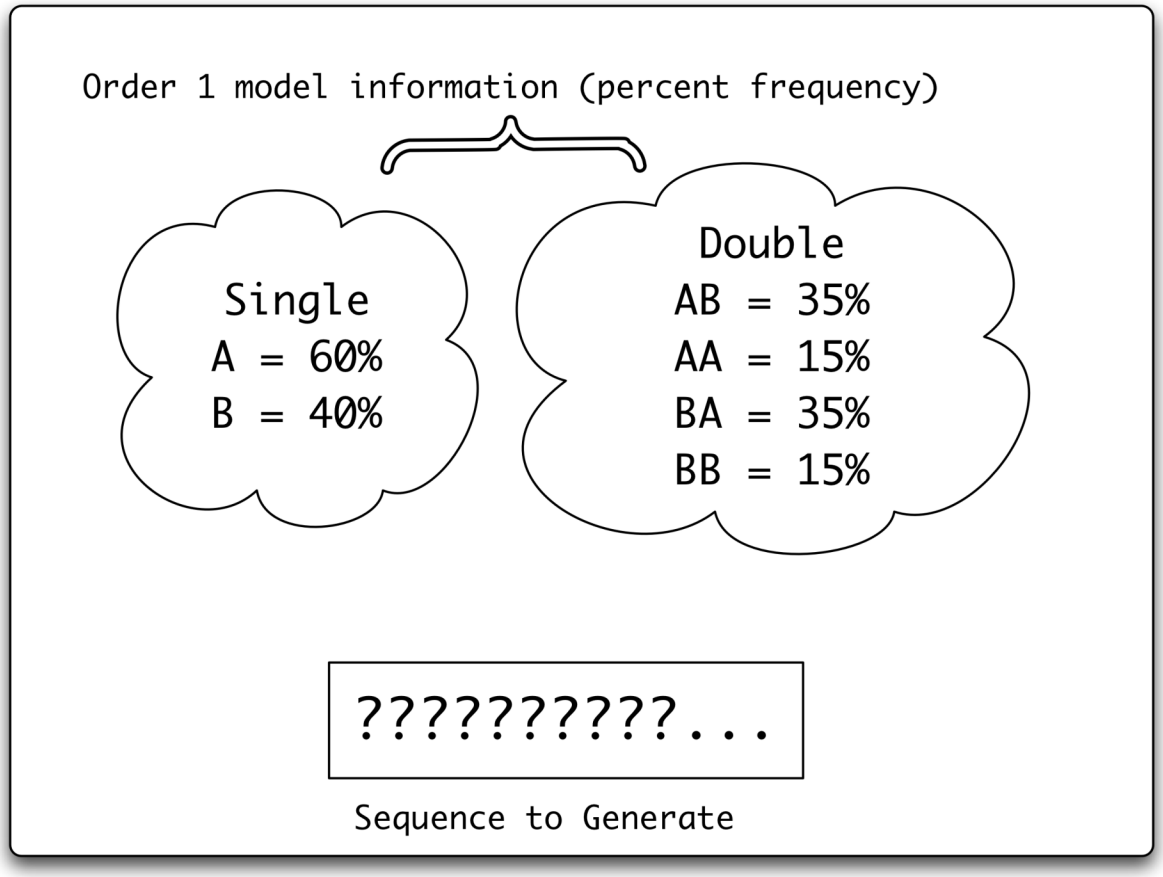
How does it work?

- Suppose we have been reading a lot of naturally occurring sequences represented by the alphabet {A,B} and have come up with some frequencies.

- We can use this information for sequence generation (modeling) and classification (prediction).

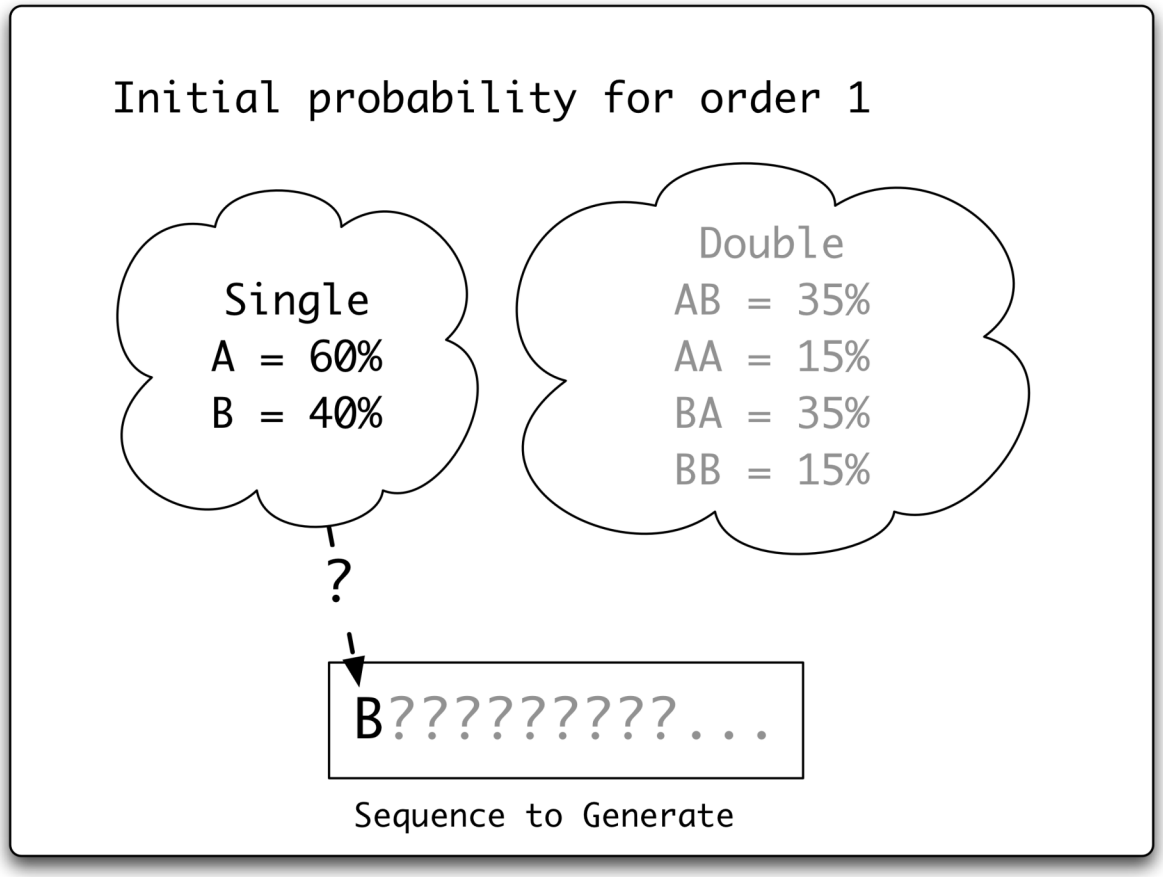
Markov generation

- *Generating* a sequence using a Markov model requires training first.
- Frequency data is for *order* 1 generation.



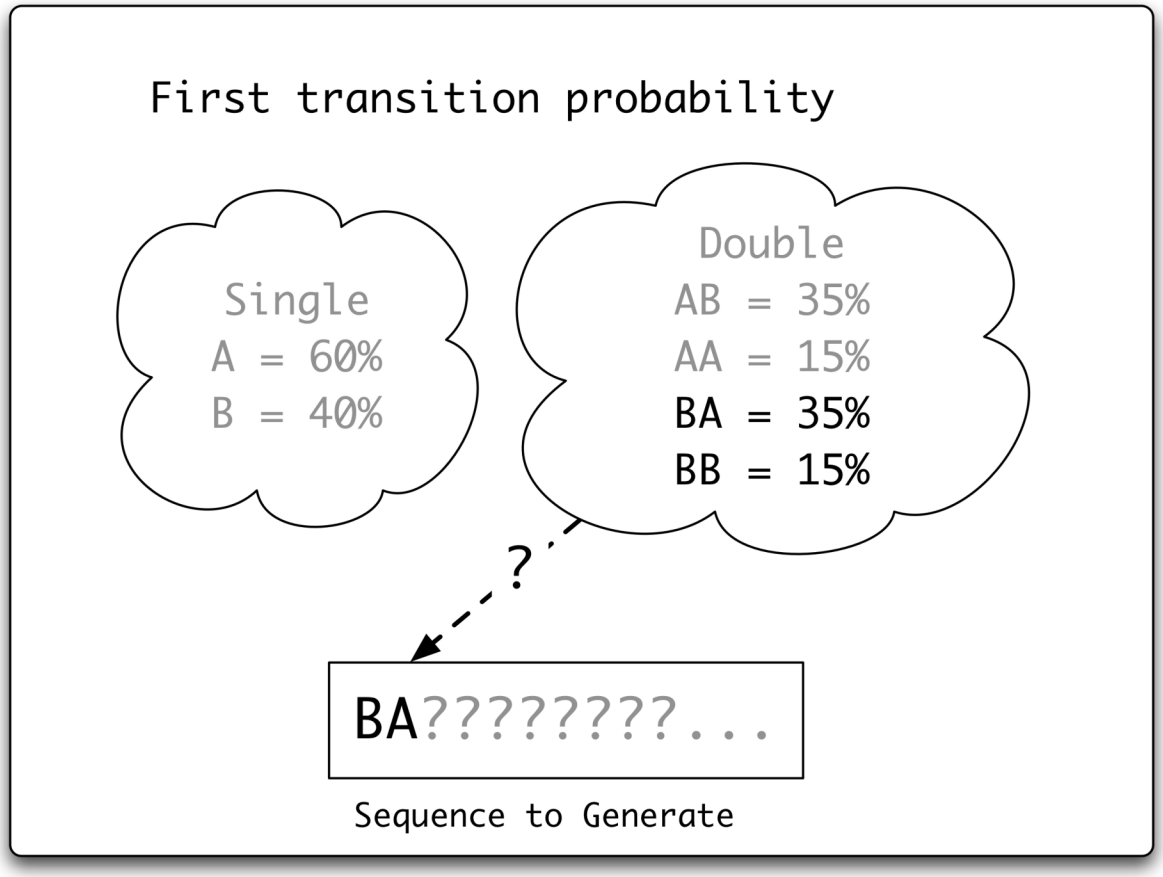
Markov generation

- To start the sequence, we use our initial probabilities.
- Generation is random, so each sequence can be unique.



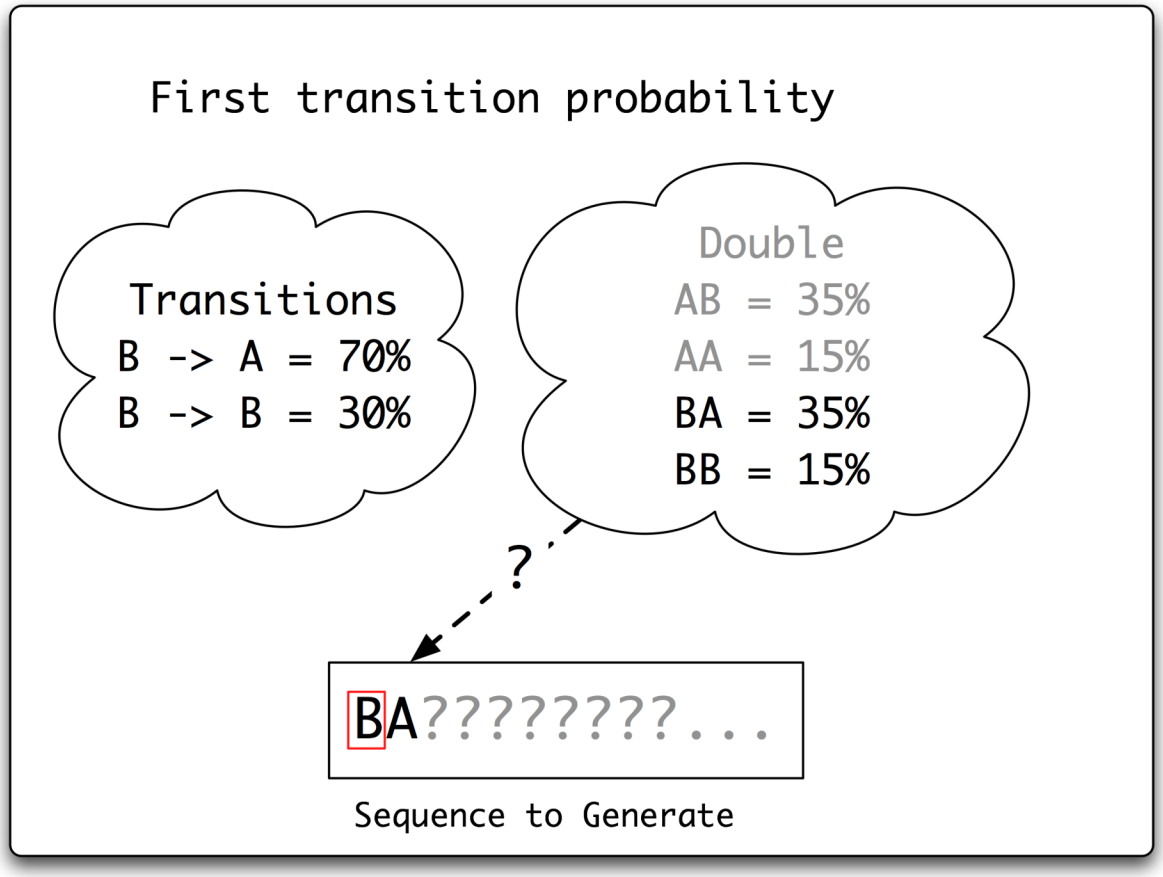
Markov generation

- Order-dependent *transition probabilities* are used to generate the next letter in the sequence.



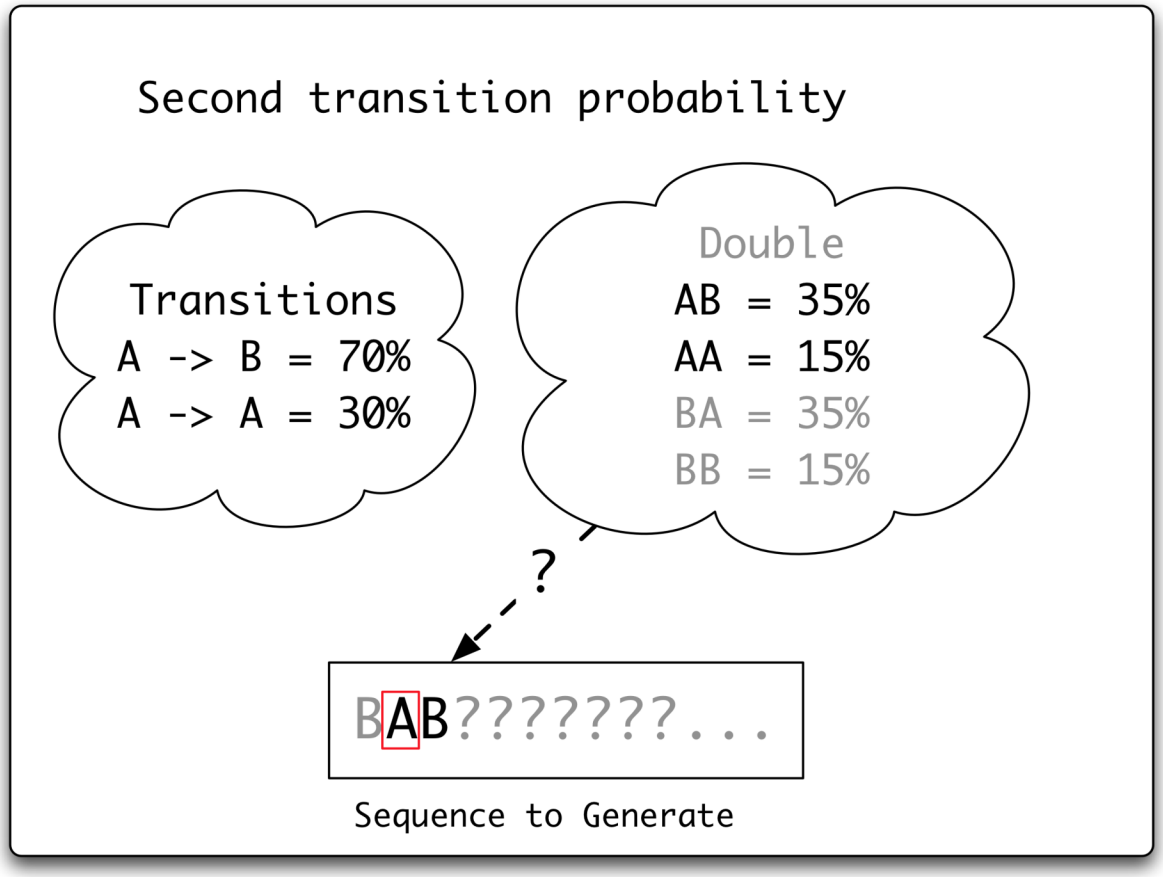
Markov generation

- Relative frequencies are used for the transition probabilities.
- These probabilities depend of the prefix [boxed], whose length is the order.



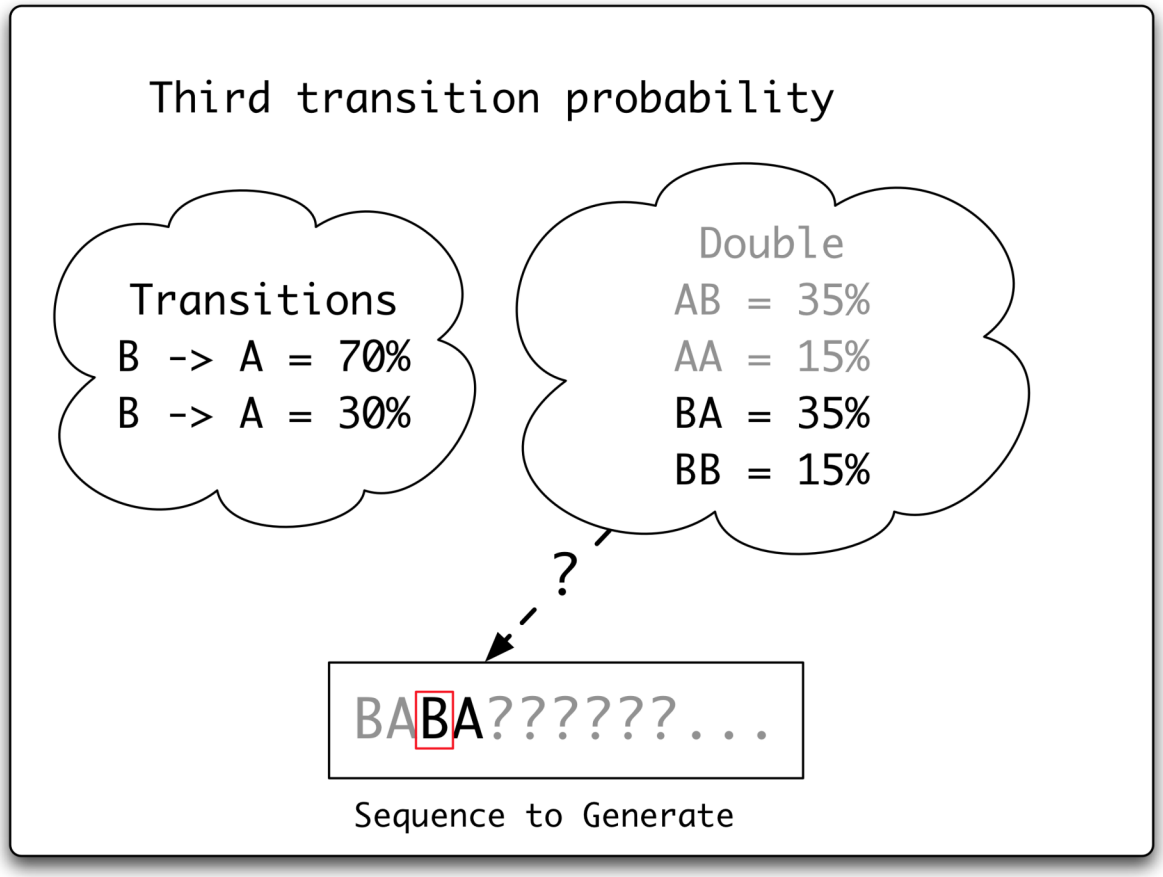
Markov generation

- Each new letter depends on the prefix, usually 0 to 5 bases for nucleotide sequences.



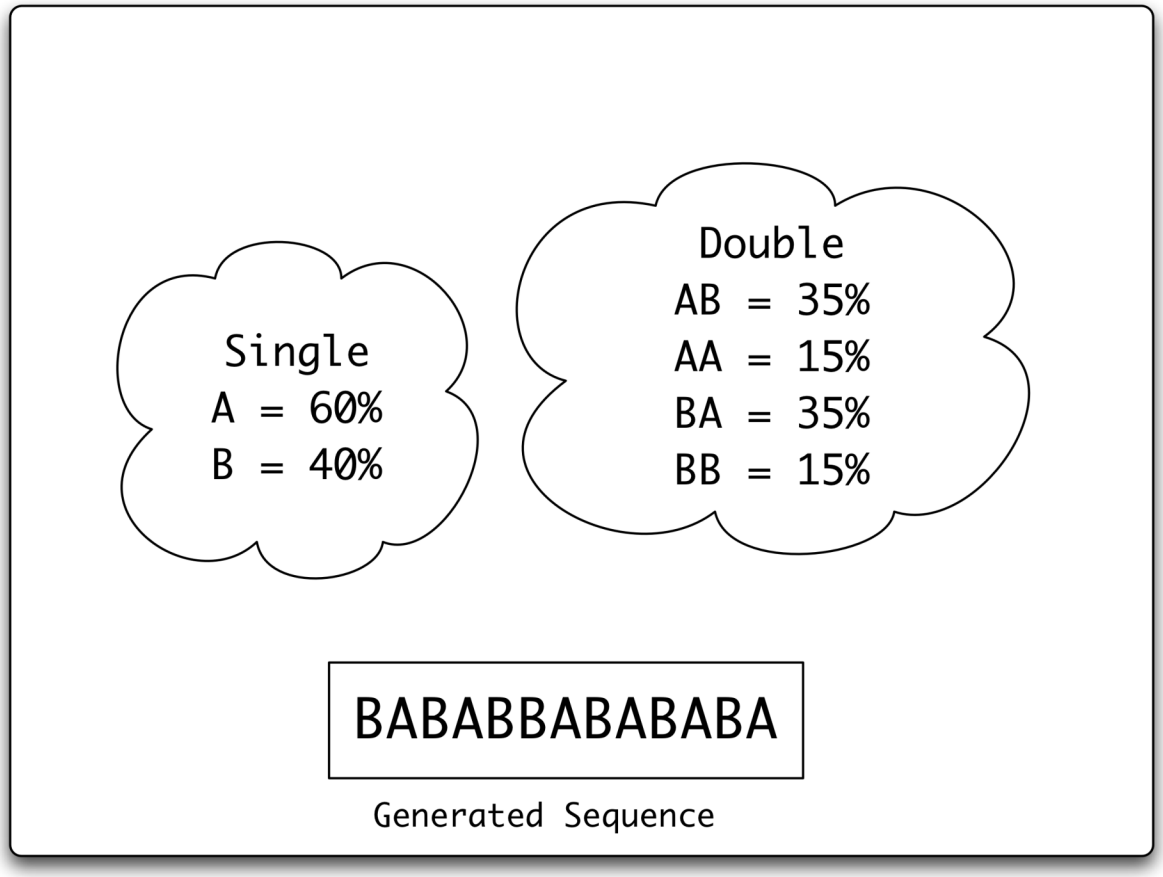
Markov generation

- The Markov chain algorithm continues as before until the desired number of letters is generated.



Markov generation

- While any permutation of the sequence is *possible*, not all sequences will be equally likely...



Testing probability

- Suppose we have two sequences:
  - *BBBB*
  - *ABAB*

How likely is each sequence? Recall:

Single
A = 60%
B = 40%

Double	
AA = 15%	AB = 35%
BB = 15%	BA = 35%

Probability of BBBB

- For the *BBBB* sequence, we get:
  - B (0.40) -> B (0.30) -> B (0.30) -> B (0.30)
  - Total probability = (.4)(.3)(.3)(.3) = 0.0108

What about *ABAB*?

Initial:
B = 40%

Transitions used:
B -> B = 30%

Probability of ABAB

- For the ABAB sequence, we get:
  - A (0.60) -> B (0.70) -> A (0.70) -> B (0.70)
  - Total probability = (.6)(.7)(.7)(.7) = 0.2058

- ABAB is a more probable.
  - Prob(ABAB) = 0.2058 > 0.0108 = Prob(BBBB)

```
Initial:
A = 60%
```

```
Transitions used:
A -> B = 70%
B -> A = 70%
```

Models

- The total probability was determined by the initial & transitions probabilities. These probabilities characterize our model.
  - Let's call our previous example the "Ab model."

- Now consider a *null* model for uniformly random sequences:

```
Single
A = 50%
B = 50%
```

```
Transitions used:
AA = 25% AB = 25%
BB = 25% BA = 25%
```

Now under the *null* model

- Prob(BBBB | null) = $(0.5)^4 = 0.0625$
- Prob(ABAB | null) = $(0.5)^4 = 0.0625$

- Given sequence *ABAB*, what is the probability of the "Ab model" being used to generate it & not the null one?

A little likelihood

- Probability of "Ab model" given *ABAB* is about 77% versus the null model.

$$
\begin{aligned}
Prob(\text{“Ab model”}|ABAB) &= \frac{P(ABAB|\text{“Ab model”})\cdot P(\text{“Ab model”})}{P(ABAB|\text{“Ab model”})\cdot P(\text{“Ab model”})+P(ABAB|\text{null})\cdot P(\text{null})} \\
&= \frac{P(ABAB|\text{“Ab model”})\cdot \frac{1}{2}}{P(ABAB|\text{“Ab model”})\cdot \frac{1}{2}+P(ABAB|\text{null})\cdot \frac{1}{2}} \\
&= \frac{P(ABAB|\text{“Ab model”})}{P(ABAB|\text{“Ab model”})+P(ABAB|\text{null})} \\
&= \frac{0.2058}{0.2058+0.0625} \\
&= 77\%
\end{aligned}
$$

- Probability of "Ab model" given *BBBB* is less than 15% versus the null model.

Feeling the Odds

- Given some sequence *S*, what are the odds of that sequence being the "Ab model" versus null?

$$\text{Odds}(\text{sequence}) = \frac{P(\text{model} = \text{“Ab model”}|\text{sequence})}{P(\text{model} = \text{null}|\text{sequence})}$$

$$= \frac{P(S|\text{“Ab model”})/P(S|\text{“Ab model”})+P(S|\text{null})}{P(S|\text{null})/P(S|\text{“Ab model”})+P(S|\text{null})}$$

$$= \frac{P(S|\text{“Ab model”})}{P(S|\text{null})}$$

Feeling the Odds

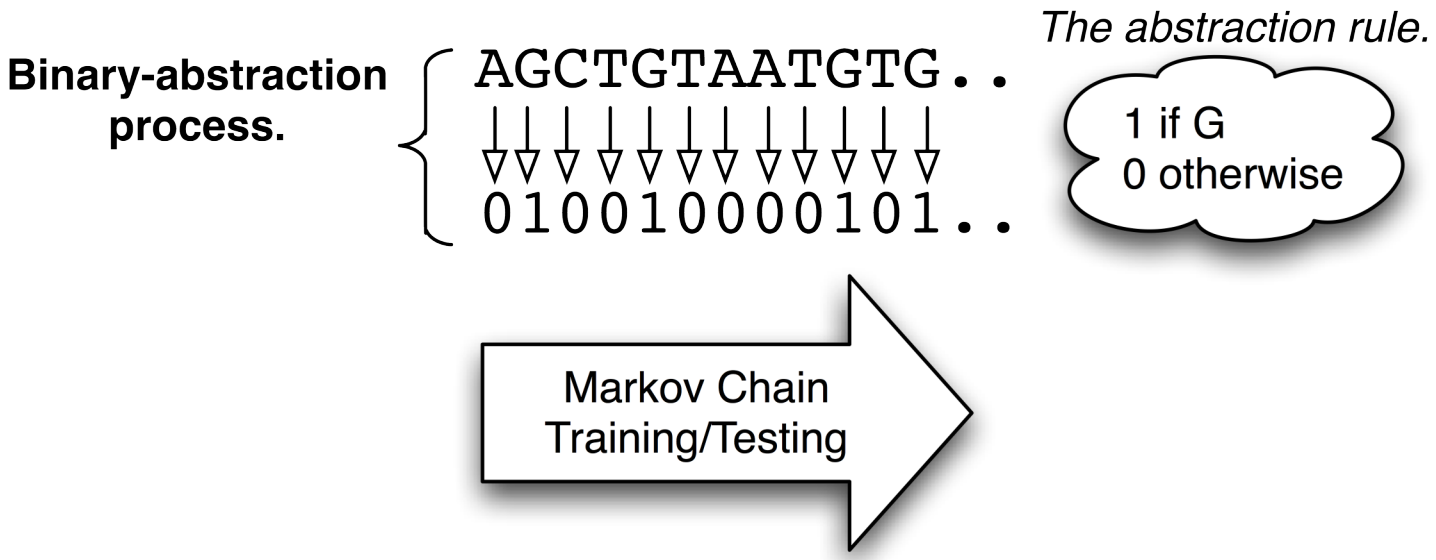
- For the odds of *ABAB* we can see that:
  - 0.767/(1–0.767) = 3.29 = 0.2058/0.0625
- The odds of *BBBB* are: 0.172
- Normally, since Markov chains deal with very small probabilities, the chain is calculated in log-space.
- The score of a sequence being “Ab model” versus *null* is the log odds.
  - Score(*BBBB*) = log(0.172) = –0.762
  - Score(*ABAB*) = log(3.29) = +0.517

Markov chains in demand

- Markov chain log probabilities (or log odds) can be used by themselves or as part of more complicated prediction algorithms.
  - Hidden Markov model
  - Support vector machines (BAMM)

Binary-abstraction Markov model

"G" or not "G", that is the question:



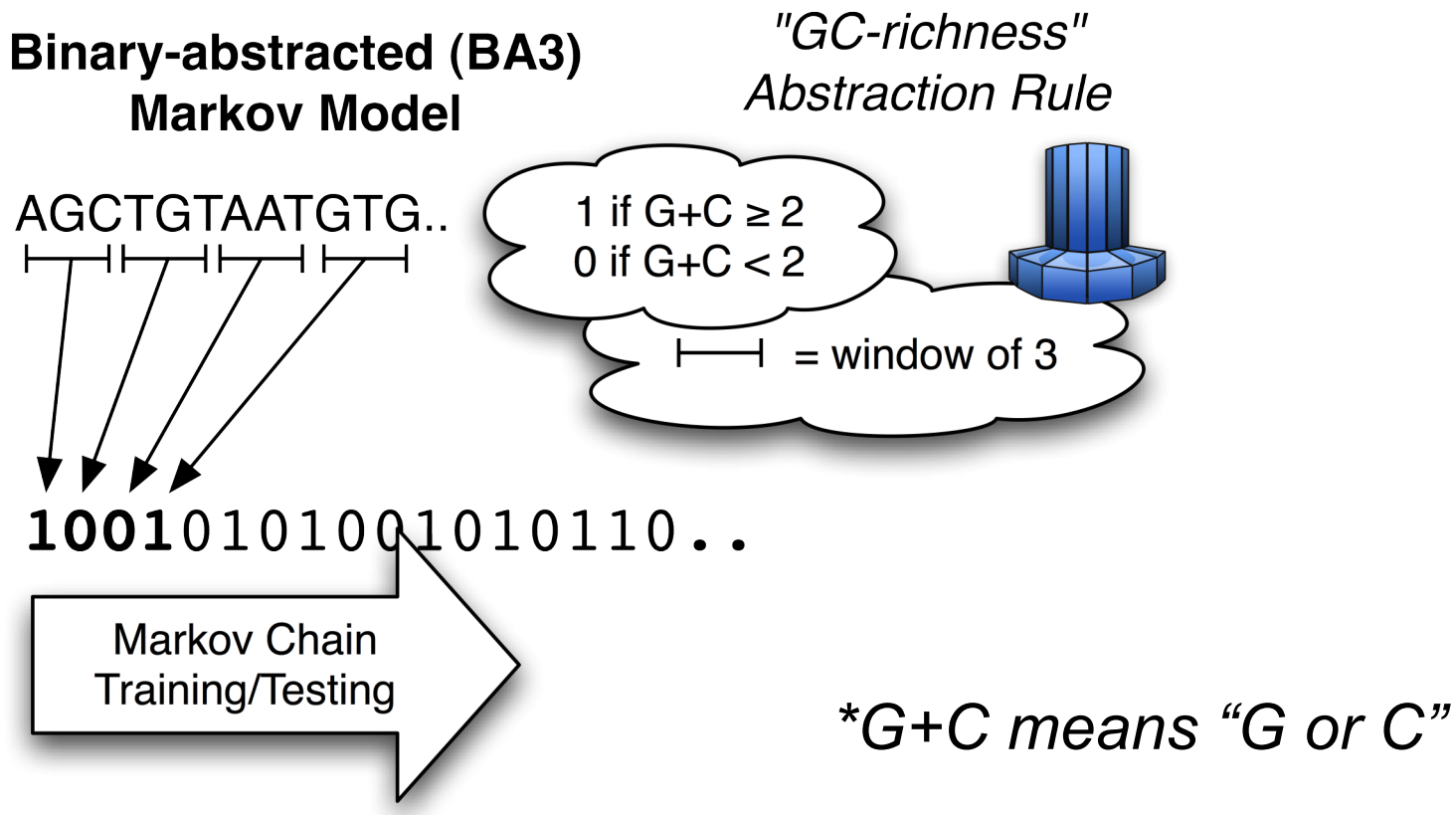
Abstraction Rule

- Abstraction rules indicate how to reduce nucleotide information into a binary code.
- Abstraction rules depend on the nucleotide word length.

How many ways can I reduce nucleotide information?

Word Length	# Words	# Abstraction Rules
1	4	16
2	16	65,536
3	**64**	**1.84×10^{19}**
4	**256**	**1.16×10^{77}**

Nucleotides Words of Length 3



*G+C means “G or C”

Binary-abstraction Markov model

- Binary-abstraction Markov models allow one to analyze longer nucleotide sequence words by reducing the information analyzed.

- Analogous to replacing all articles in a sentence with ‘A’, verbs with ‘V’, and nouns with ‘N’, except in our case one must find what to replace first! Units of meaning are not obvious.

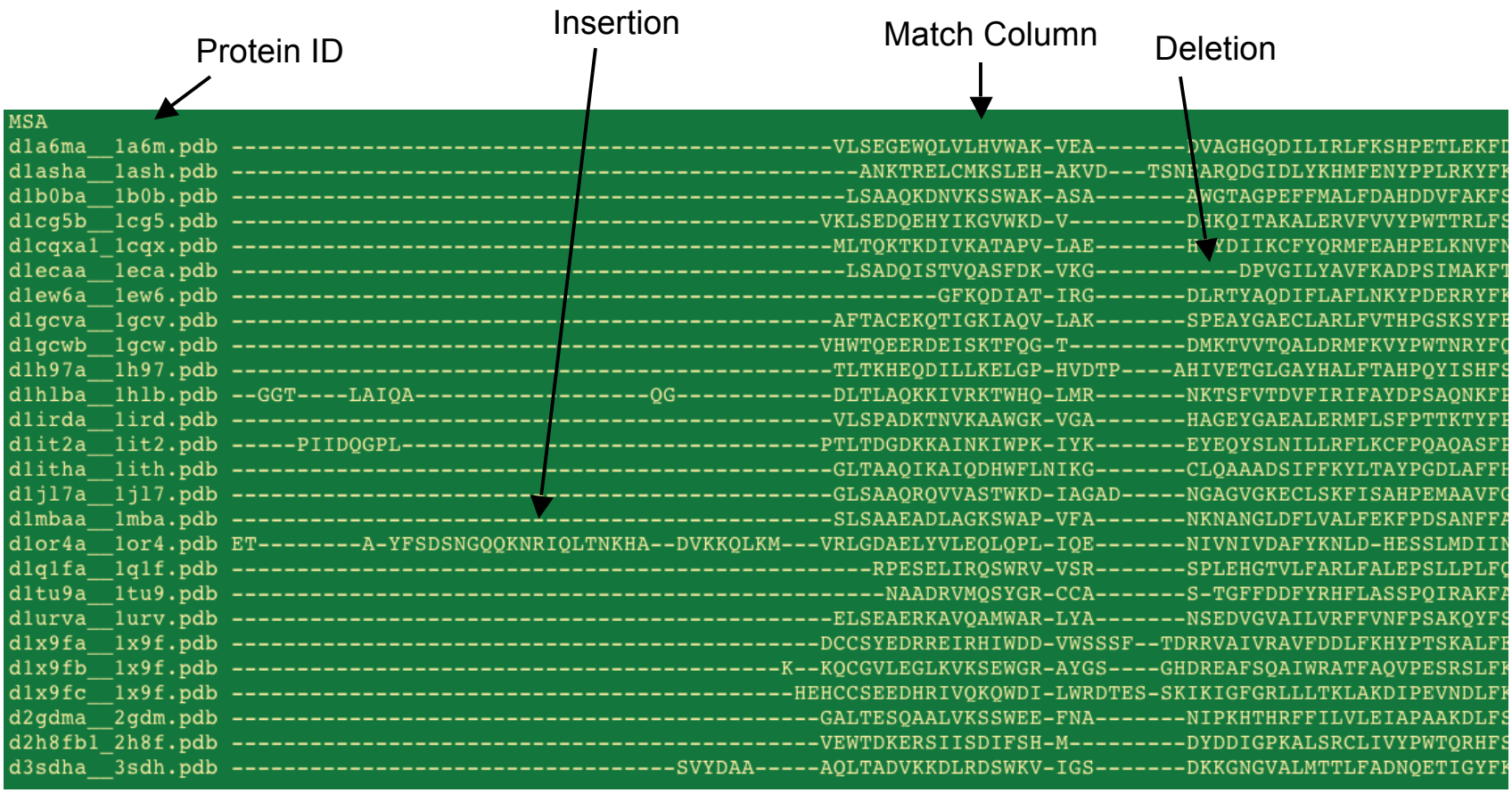
Profile HMMs

- Uses protein multiple sequence alignments to build an HMM profile of related proteins.
- The profile can be used to search for remote protein homologues within databases.

HMM Modeler

- Customizable profile HMM tool for remote homologue identification.
  - Implemented as a Chimera plug-in.
  - Joint effort of the Salzburg University of Applied Sciences with Salzburg University
- Astral Protein Database has protein sequences with less than 40% identity.
- SCOP protein families are grouped by structure.

Sample Alignment

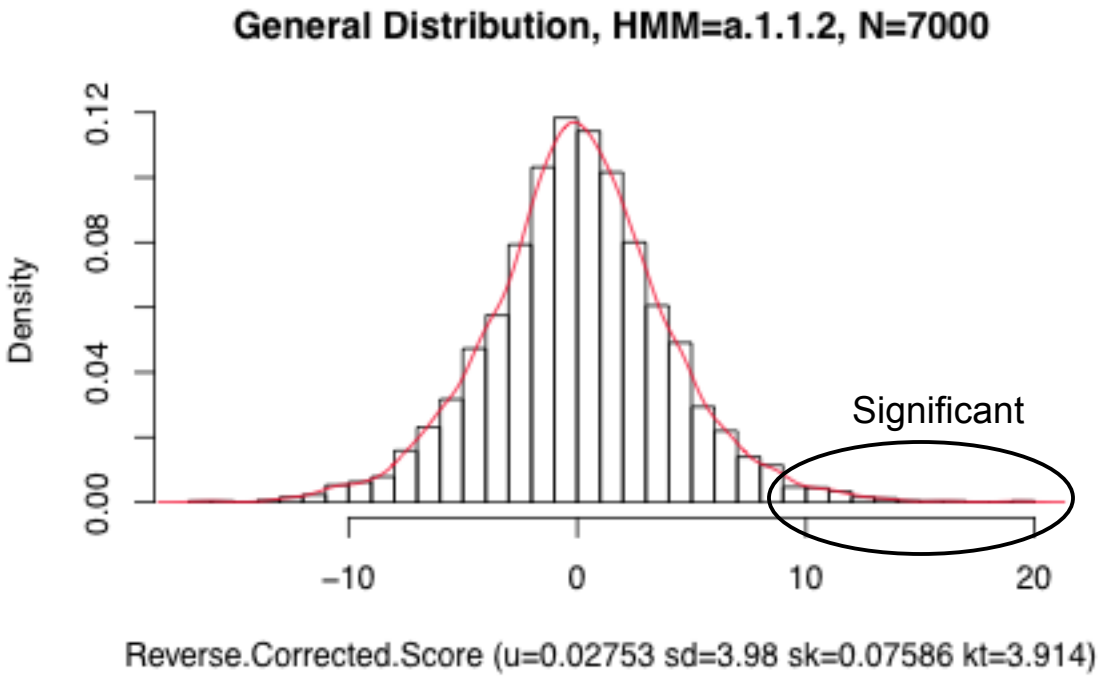


Profile HMMs

- Match columns, deletions, and insertions are used to develop the profile HMM of the protein family.
- One can search protein databases using the profile, and based on the query score, filter for membership.

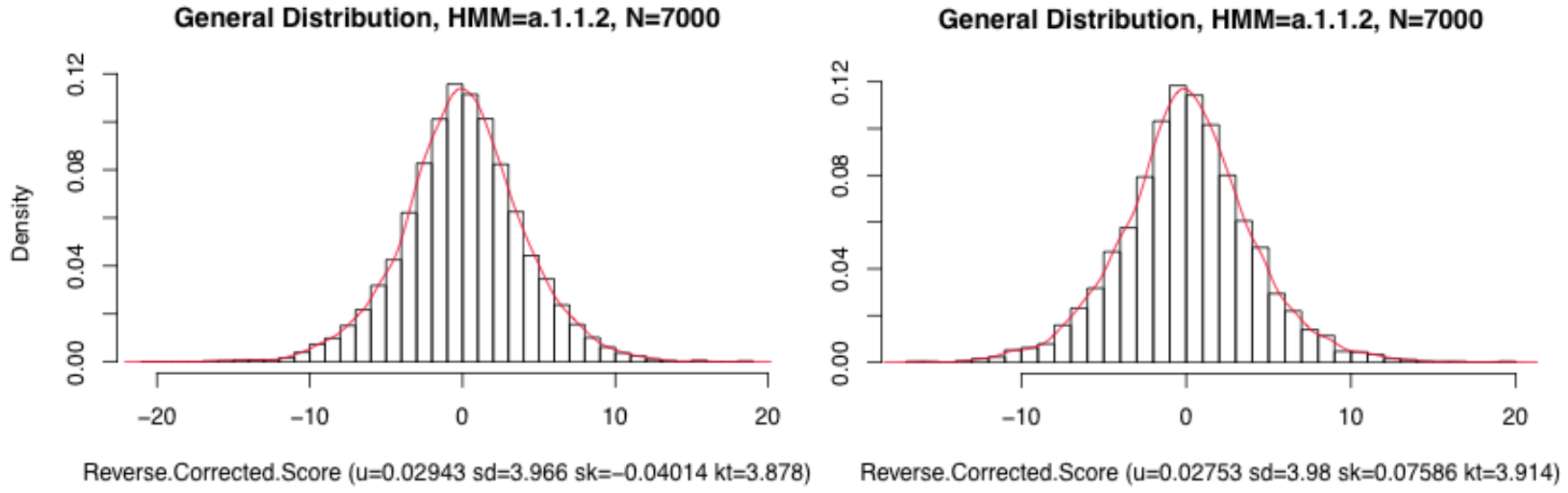
Evaluating remote homologues

- Scores are corrected for length bias.
- A *null* distribution is created of non-protein-family scores.
- Scores that exceed a threshold of significance, say greater than 95% could be counted.



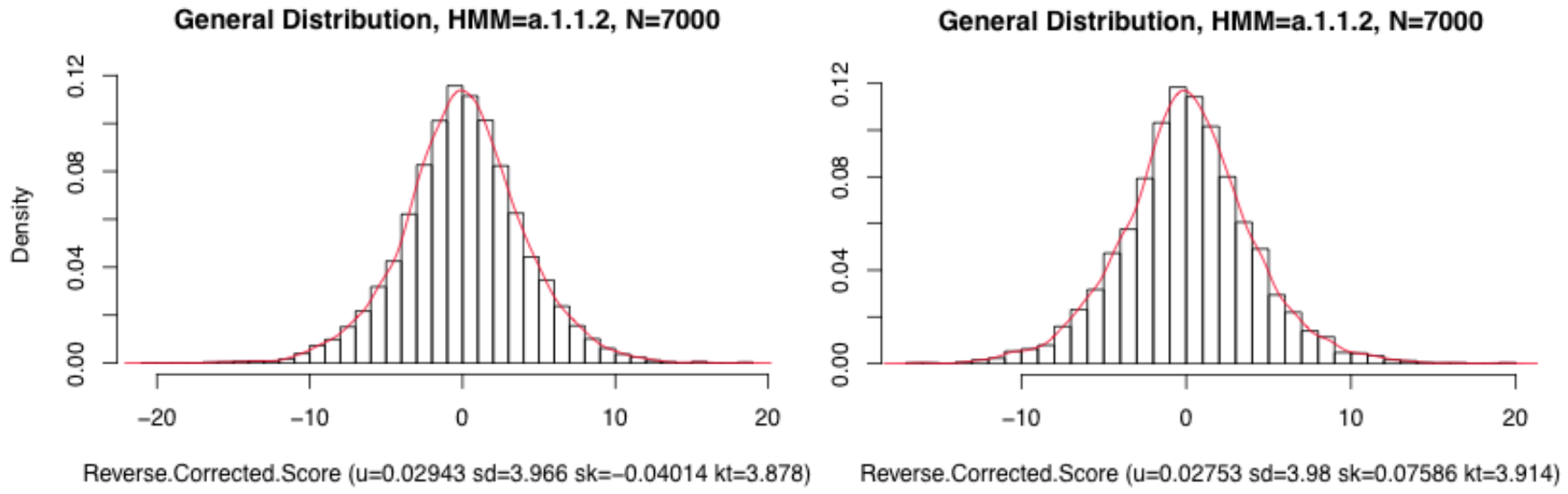
Simulating Proteins with Markov models

- Generated *null* distribution with Markov chain simulated proteins.



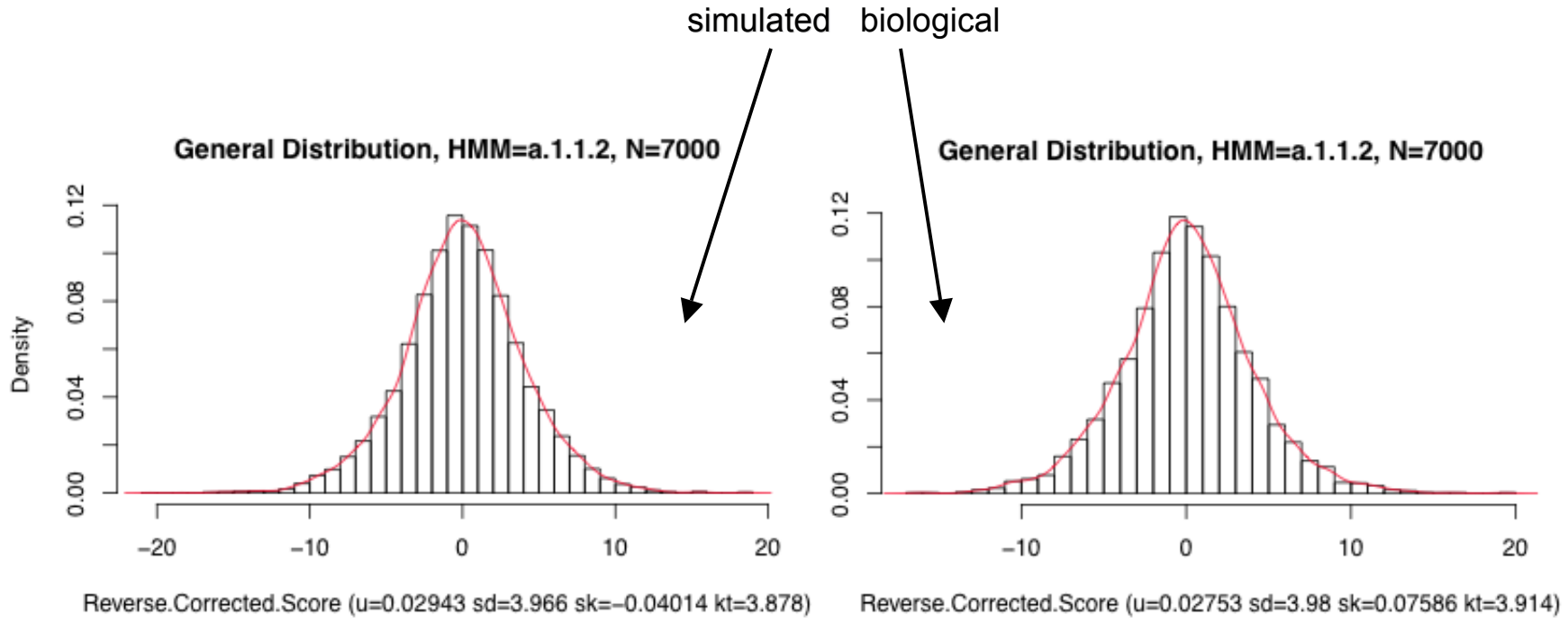
Simulating Proteins with Markov models

- Simulated proteins are generated from Astral database frequency information for Markov order 2.
- Which is the biological distribution?



Simulating Proteins with Markov models

- Simulated proteins can smooth the null distribution or reduce computation time.



In Conclusion

- Markov chains can be used for any sequence data.
- Useful in gene prediction, remote homologue identification, and much more.
- Can be used to generate AND discriminate sequence data.

Thank you for your attention.

Questions? 问题?

¿Preguntas? Fragen?

вопросы? 質問か。