# Biology Meets Programming: Bioinformatics for Beginners

University of California San Diego / Coursera

--------------------------------------------------------------------------------------------------------------------------

Week 1 : 1. Where in the Genome Does Replication Begin? (Part 1)

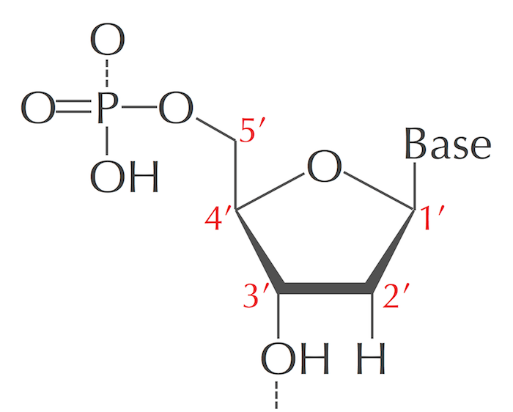
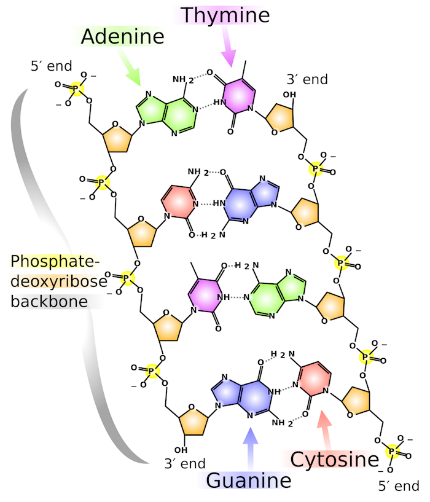
Replication begins in a genomic region called the **replication origin** (denoted ***ori***) and is performed by molecular copy machines called DNA polymerases. Locating ori presents an important task not only for understanding how cells replicate but also for various biomedical problems. For example, some gene therapy methods use genetically engineered mini-genomes, which are called viral vectors because they are able to penetrate cell walls (just like real viruses). Viral vectors carrying artificial genes have been widely used in agriculture, such as to engineer frost-resistant tomatoes and pesticide-resistant corn.

The idea of gene therapy is to intentionally infect a patient who lacks a crucial gene with a viral vector containing an artificial gene that encodes a therapeutic protein. To ensure that the vector actually replicates inside the cell, biologists must know where ori is in the vector’s genome and ensure that the genetic manipulations that they perform do not affect it.

Research has shown that the region of the bacterial genome encoding ori is typically a few hundred nucleotides long. How does the bacterial cell know to begin replication exactly in this short region within the much larger *Vibrio cholerae* chromosome, which consists of over a million nucleotides? There must be some “hidden message” in the ori region ordering the cell to begin replication here. Indeed, we know that the initiation of replication is mediated by **DnaA**, a protein that binds to a short segment within the ori known as a **DnaA box**.

Experiments have revealed that bacterial DnaA boxes are usually 9 nucleotides long. It is very unlikely that a 9-mer would appear three or more times in a randomly generated DNA string of length 500 due to random chance.

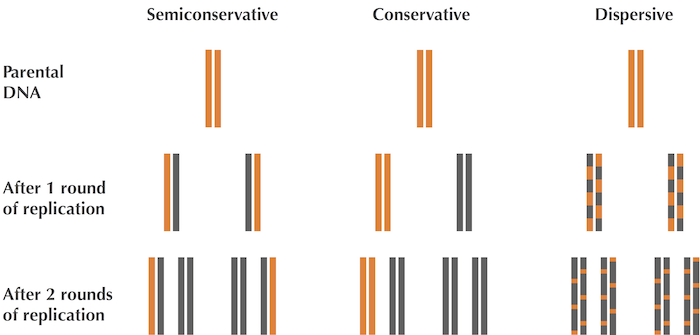
The beginning and end of a DNA strand are denoted 5’ (five prime) and 3’ (three prime), respectively. Each DNA strand is read in the 5' 🡪 3' direction, and the complementary strand runs in the opposite direction to the template strand.

Interestingly, among the four most frequent 9-mers in the ori region of *Vibrio cholerae*, "ATGATCAAG" and "CTTGATCAT" are reverse complements of each other. Finding a 9-mer that appears six or more times (either as itself or as its reverse complement) in a DNA string of length 500 is far more surprising than finding a 9-mer that appears three or more times alone. This statistical evidence leads us to the working hypothesis that "ATGATCAAG" and its reverse complement "CTTGATCAT" indeed represent DnaA boxes in *Vibrio cholerae*.

"ATGATCAAG"와 그 역상보서열은 *Vibrio cholerae*의 전체 genome에서 17번 나타나지만 ori 영역에서만 서로 가까운 위치에 나타난다. 하지만 이는 다른 박테리아에서도 동일한 것은 아니다.

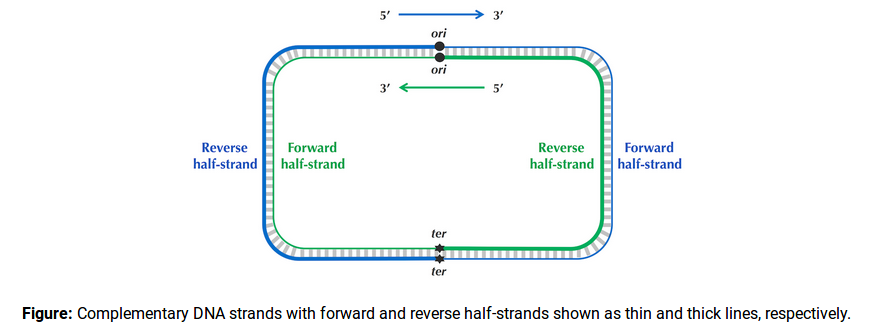
DNA replication – semiconservative model



--------------------------------------------------------------------------------------------------------------------------

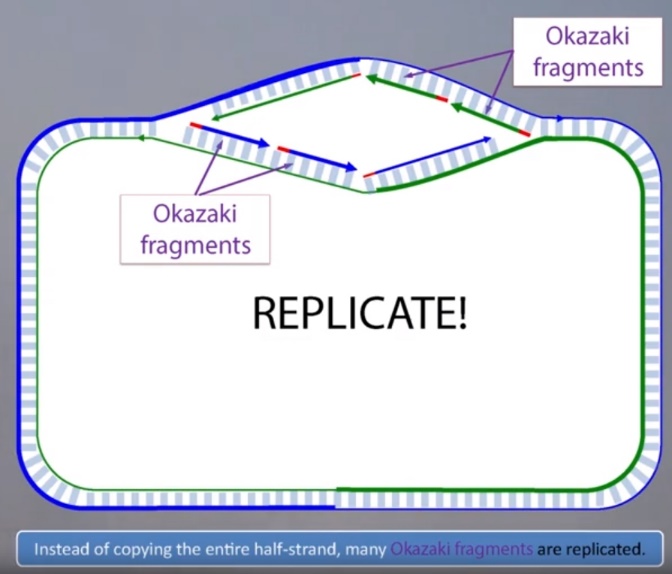
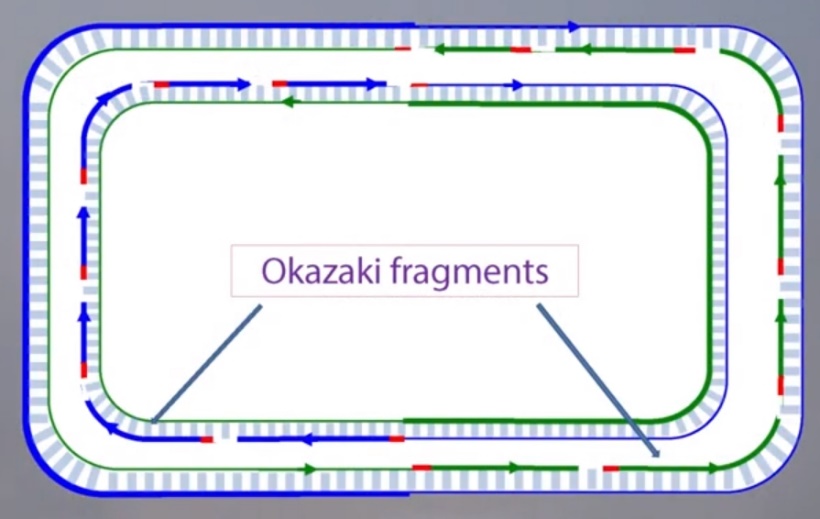
Week 2 : 1. Where in the Genome Does Replication Begin? (Part 2)

(1) Asymmetry of Replication



The two strands run in opposite directions (from 5’ to 3’)

-- blue : clockwise / green : counter-clockwise

The two complementary DNA strands running in opposite directions around a circular chromosome unravel, starting at *ori*. As the strands unwind, they create two **replication forks**, which expand in both directions around the chromosome until the strands completely separate at the **replication terminus** (denoted *ter*). The replication terminus is located roughly opposite to ori in the chromosome.

An important thing to know about replication is that a DNA polymerase does not wait for the two parent strands to completely separate before initiating replication; instead, it starts copying while the strands are unraveling.

Since a DNA polymerase can only move in the reverse (3' 🡪 5') direction, it can copy nucleotides non-stop from ori to ter along reverse half-strands. However, replication on forward half-strands is very different because a DNA polymerase cannot move in the forward (5' 🡪 3') direction; on these half-strands, a DNA polymerase must replicate backwards toward ori.

Finally, consecutive Okazaki fragments must be sewn together by an enzyme called DNA ligase, resulting in two intact daughter chromosomes, each consisting of one parent strand and one newly synthesized daughter strand.

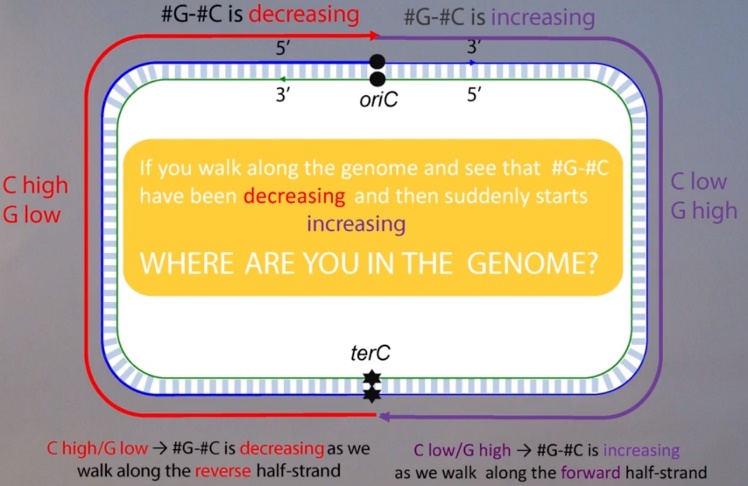
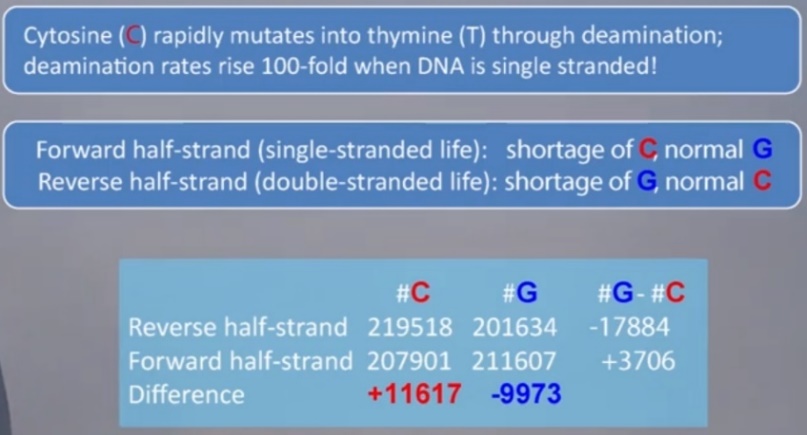
(2) Peculiar Statistics of the Forward and Reverse Half-Strands : Skew Diagram

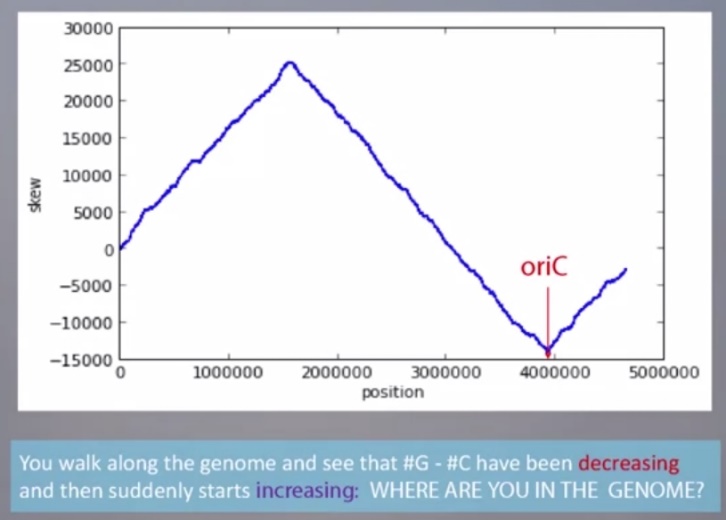
DNA polymerase synthesizes DNA quickly on the reverse half-strand but suffers delays on the forward half-strand.

정방향과 역방향 가닥의 복제 속도가 서로 다름 🡪 ori 영역을 찾는 G-C mutation rate 다름 🡪 ori 위치 탐색.

Cytosine (C) has a tendency to mutate into thymine (T) through a process called deamination. Deamination rates rise 100-fold when DNA is single-stranded, which leads to a decrease in cytosine (C) on the forward half-strand.

\* Thermotoga petrophila genome





skew 최소값 🡪 *ori* 영역의 위치

그림은 *E. coli*에 대한 skew diagram. 최대값은 위치 1600000 주변에서 발생하고 최소값은 위치 4000000 주변에서 발생. 따라서 역방향 절반 가닥은 위치 1600000 주변에서 시작하고 순방향 절반 가닥은 위치 4000000 주변에서 시작한다고 추론 할 수 있다. 즉 E. coli 게놈의 *ori*가 4000000 위치 근처에 있다는 것을 발견.

DnaA can bind not only to “perfect” DnaA boxes but to their slight modifications as well.

- ATGATCAAG and CTTGATCAT

- ATGATCAAC and CATGATCAT (additional occurrences)

두 문자열 p와 q 사이의 총 불일치 수 = Hamming distance.

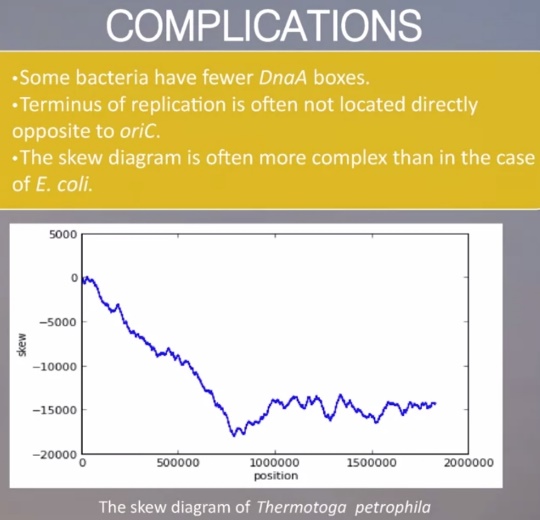
Hamming distance를 이용해 약간 다른 조합 k-mer로 DnaA box를 발견할 수 있다.

\* Compications

- Some bacteria have fewer *DnaA* boxex.

- Terminus of replication is often not located directly opposite to *oriC*.

- The skew diagram is often more complex than in the case of *E. coli*.



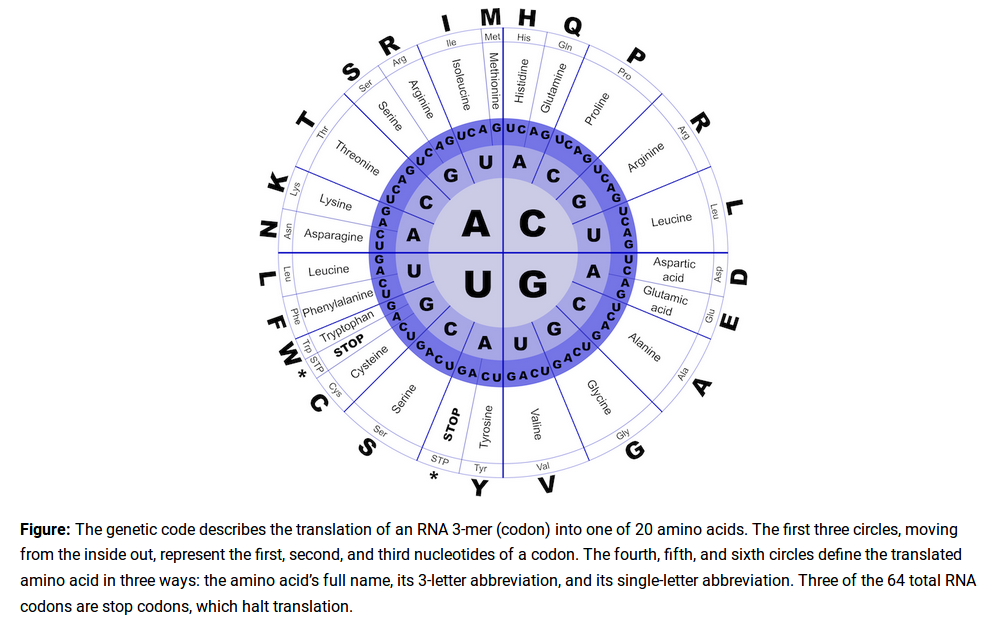
--------------------------------------------------------------------------------------------------------------------------Week 3 : 2. Which DNA Patterns Play The Role of Molecular Clocks? (Part 1)

Gene expression : DNA (transcription) 🡪 RNA (translation) 🡪 Protein

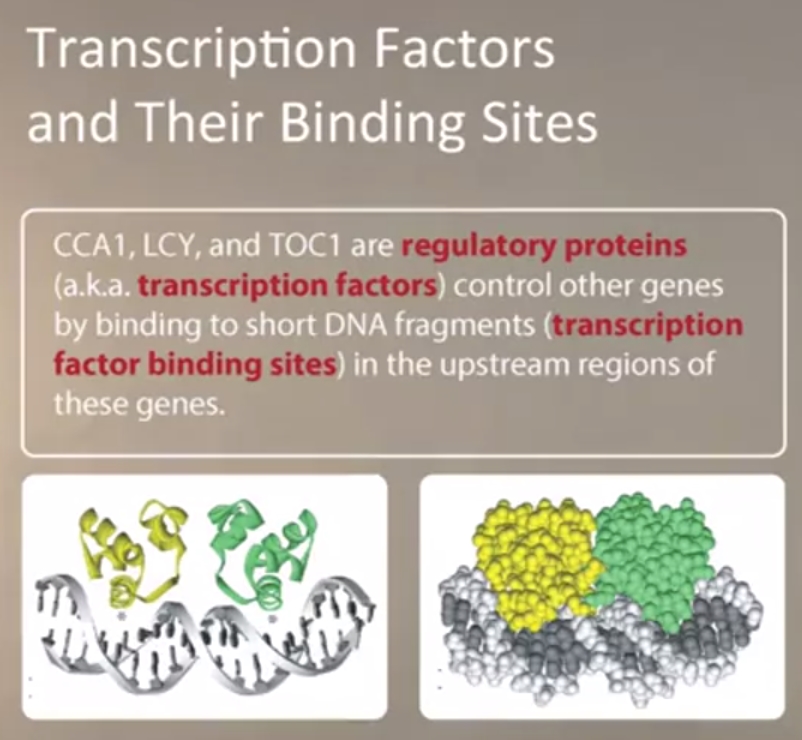
- the RNA strand is partitioned into non-overlapping 3-mers called **codons**

- each codon is converted into one of 20 amino acids

- each of the 64 RNA codons encodes its own amino acid, with the exception of three **stop codons**



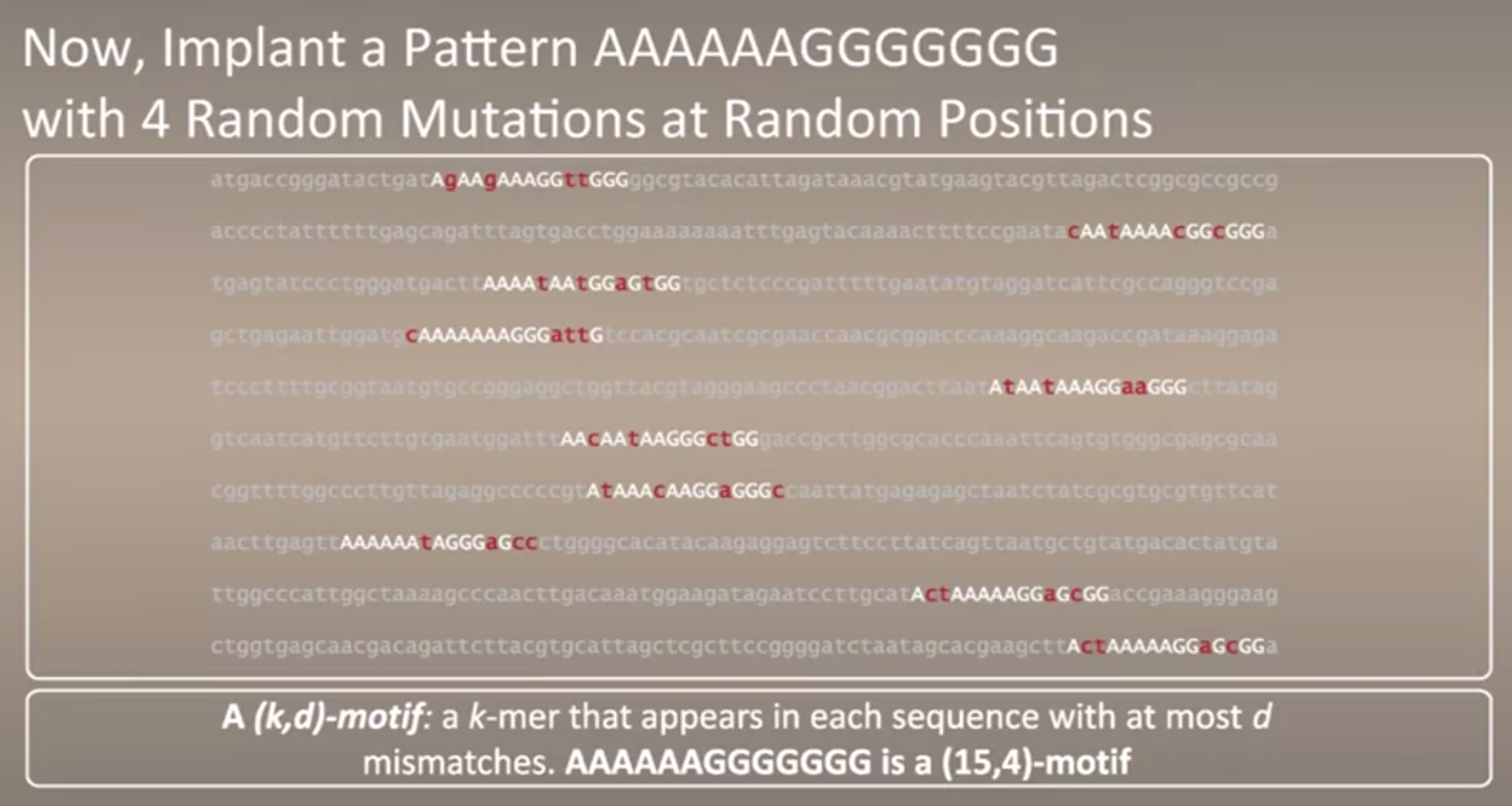
It turns out that every plant cell keeps track of day and night independently of other cells, and that just three plant genes, called LHY, CCA1, and TOC1, are the clock’s master timekeepers. Such genes, and the **regulatory proteins** that they encode, are often controlled by external factors (e.g., nutrient availability or sunlight) in order to allow organisms to adjust their gene expression.



LHY, CCA1, and TOC1 are able to control the transcription of other genes because the regulatory proteins that they encode are **transcription factors**, or master regulatory proteins that turn other genes on and off. A transcription factor regulates a gene by binding to a specific short DNA interval called a **regulatory motif**, or **transcription factor binding site**, in the gene’s upstream region, a 600-1000 nucleotide-long region preceding the start of the gene. For example, CCA1 binds to "AAAAAATCT" in the upstream region of many genes regulated by CCA1. But, the reality is more complex, as regulatory motifs may vary at some positions, e.g., CCA1 may instead bind to "AAGAACTCT".

🡪 Finding regulatory motif : Mismatches Problem으로 인해 찾기도 쉽지 않다.

A *DnaA* box is a pattern that clumps, or appears frequently, within a DNA string. In contrast, a regulatory motif is a pattern that appears at least once in each one of several different regions that are scattered throughout the genome.



Motif finding would score individual instances of motifs depending on how similar they are to an “ideal” motif. However, since the ideal motif is unknown, we attempt to select a k-mer from each string and score these k-mers depending on how similar they are to each other.

Motif Finding Problem: Given a collection of strings, find a set of k-mers, one from each string, that minimizes the score of the resulting motif.

- 길이가 k인 t 개의 DNA string = Motifs matrix

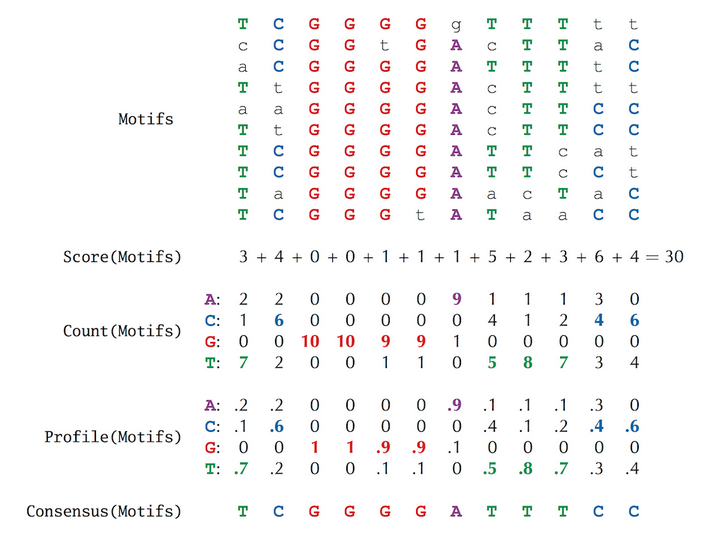
- Count matrix : 예) 길이 6인 DNA string  nucleotide 빈도에 따라 4 \* 6 matrix 생성.

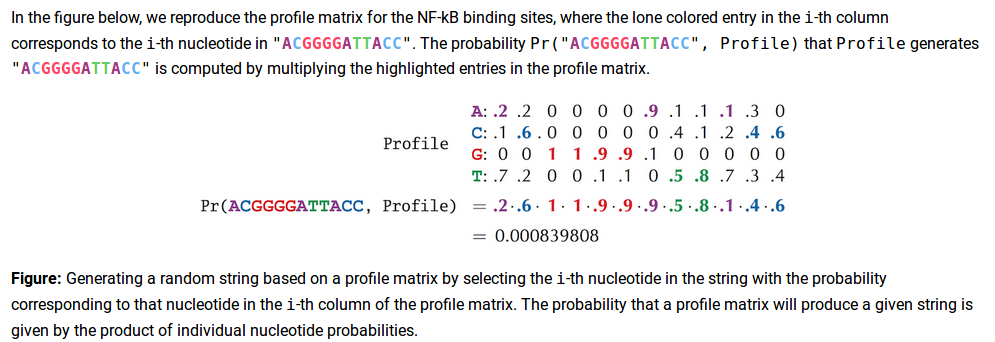
- Profile matrix : Count matrix를 DNA 수인 t 로 나눠 비율로 표시. 각 열의 합은 1.

- Consensus string : the most popular nucleotides in each column of the motif matrix (같은 값이면 임의로).

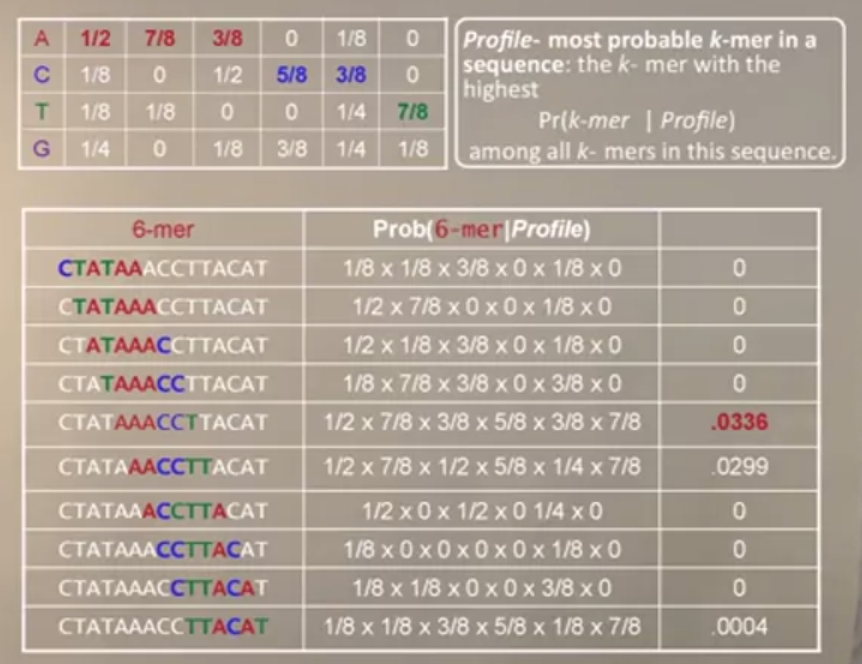
- Motif Score : Consensus 문자열의 j 위치 기호와 일치하지 않는 Motifs의 j 번째 열에있는 기호의 수를 합산

Greedy algorithms select the “most attractive” alternative at each iteration.



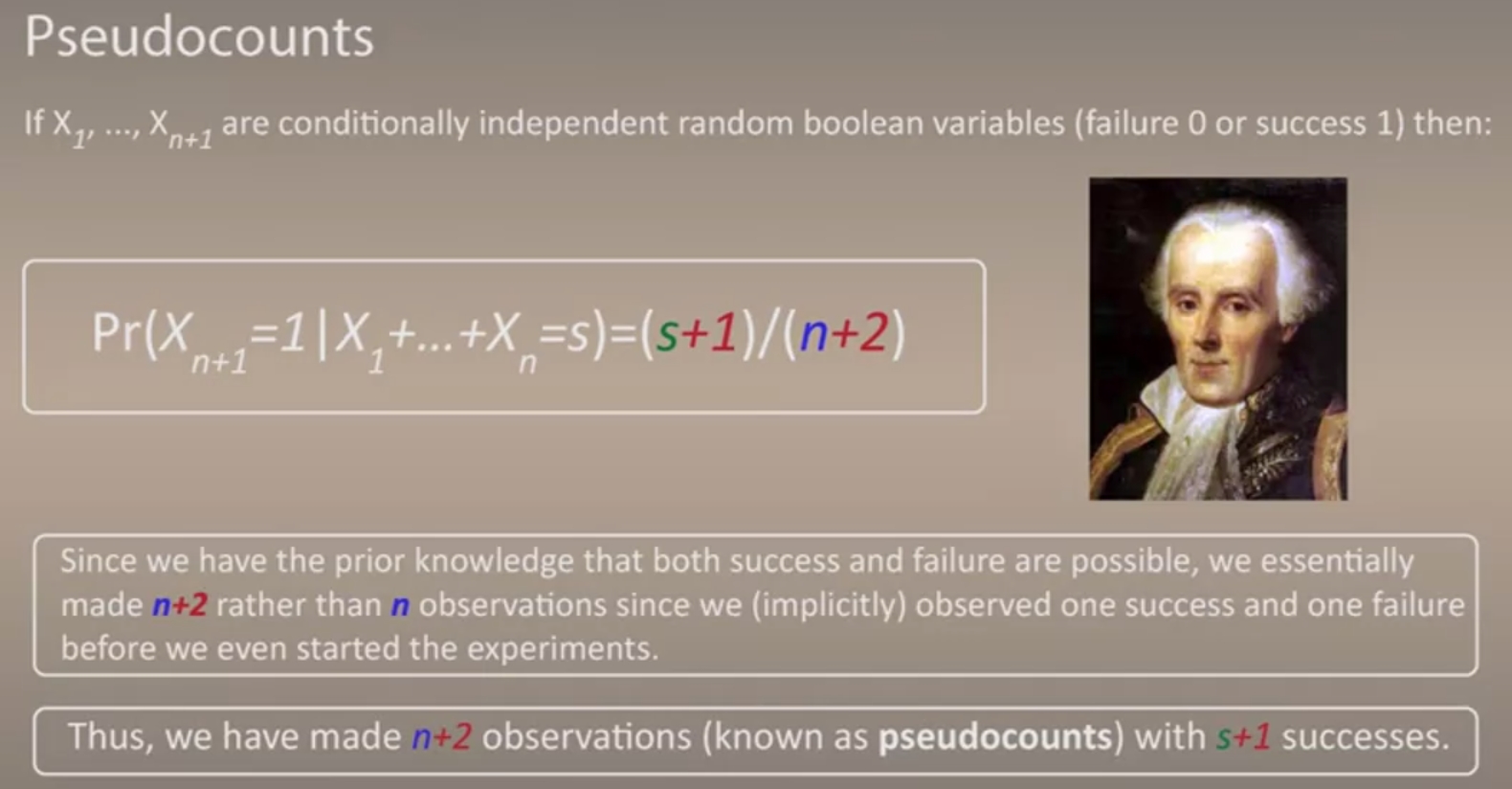


Scoring k-mer with a Profile 🡪 search most probable pattern.



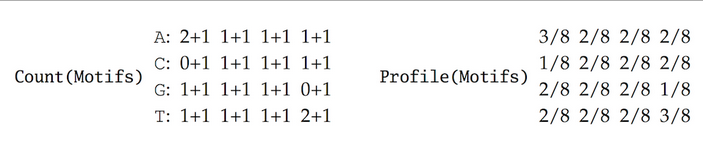
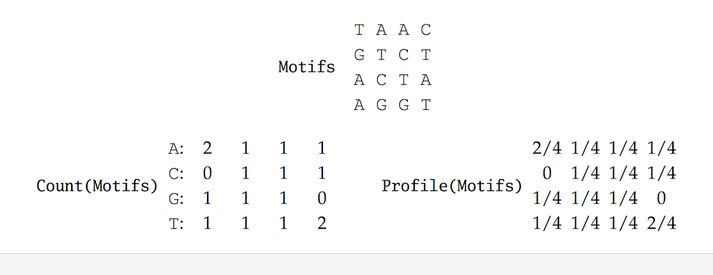
--------------------------------------------------------------------------------------------------------------------------Week 4 : 2. Which DNA Patterns Play The Role of Molecular Clocks? (Part 2)

관찰된 데이터셋에서 어떤 이벤트는 확률이 0은 아니지만 발생하지 않을 가능성이 있다. 관측된 발생빈도는 0 이지만 이 이벤트의 경험적 확률을 0으로 설정하면 문제가 될 수 있다. 이런 경우에는 확률을 인위적으로 조정하여 문제를 완화시킬 수 있다.



**pseudocounts** 라고 불리는 작은 숫자로 0을 대체 (Laplace’s Rule of Succession).

Count(Motifs)의 각 요소에 1 (또는 다른 수)을 더해준다.

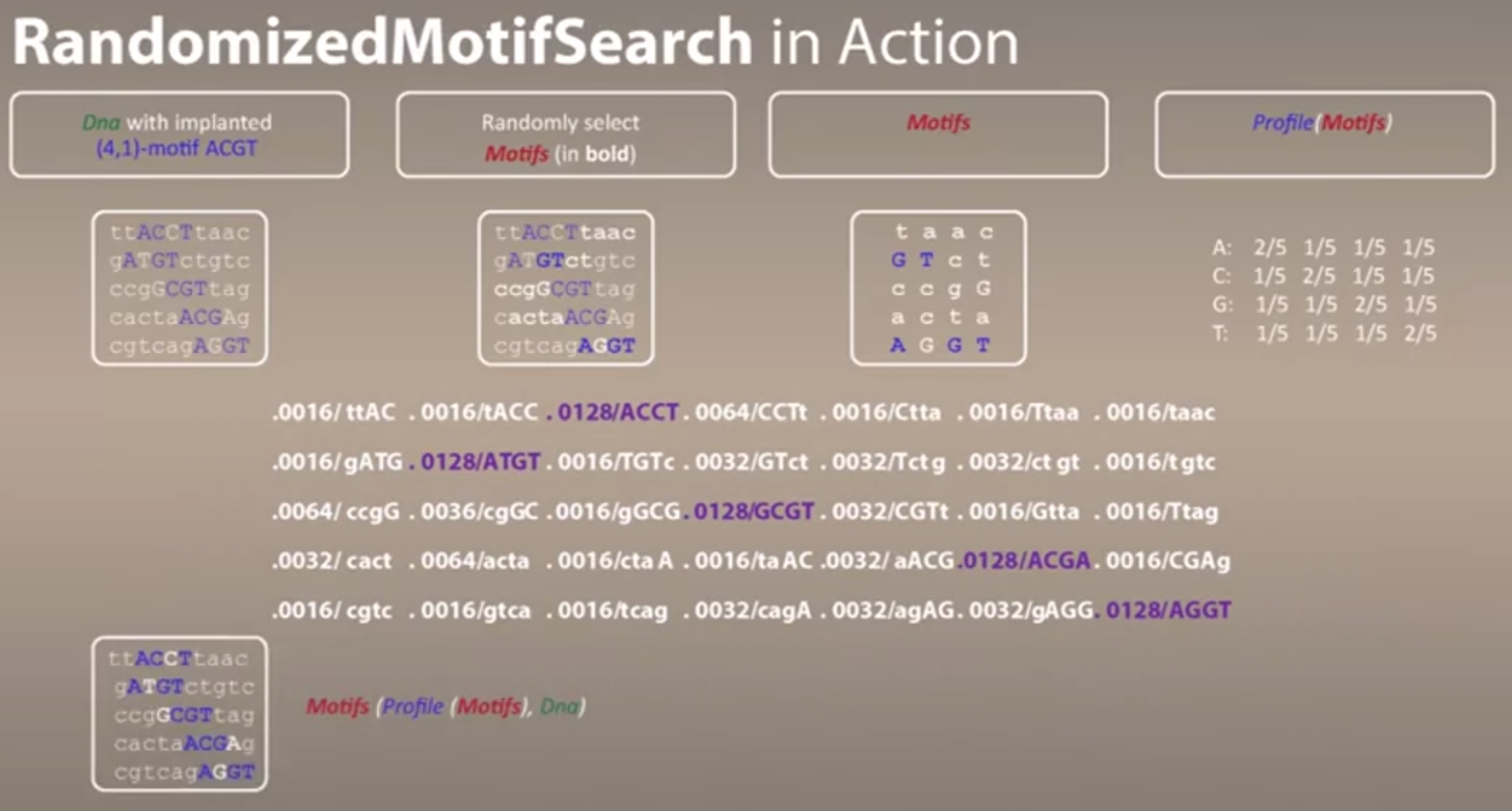


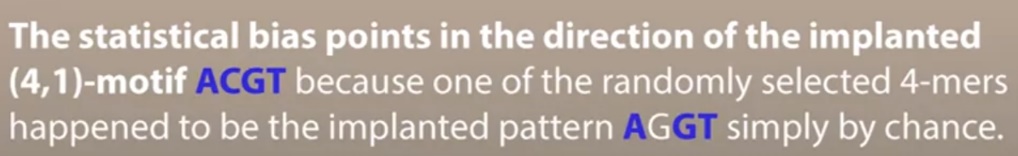
이 count matrix를 바탕으로 profile을 생성하여 GreedyMotifSearch를 실행하면 motifs의 Score가 향상된다.

* Randomized algorithms ~ Monte Carlo algorithms

DNA에서 무작위로 선택한 k-mer 모음, 이를 이용해 Profile 구성하고, 이 Profile을 사용해 새로운 k-mer 모음을 생성. 두번째로 생성한 k-mer의 score가 이전 것보다 더 좋음.

k-mer의 score를 계속 향상되는 한 이 과정을 계속 반복.





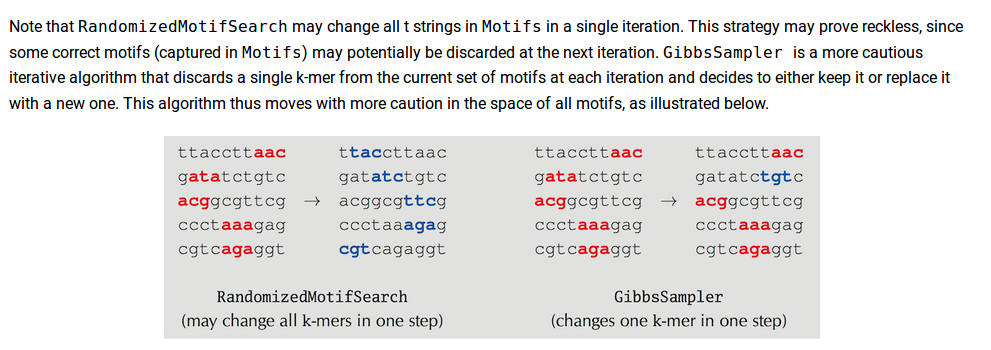
RandomizedMotifSearch는 더 긴 모티프를 찾을 수 있다는 장점이 있다.

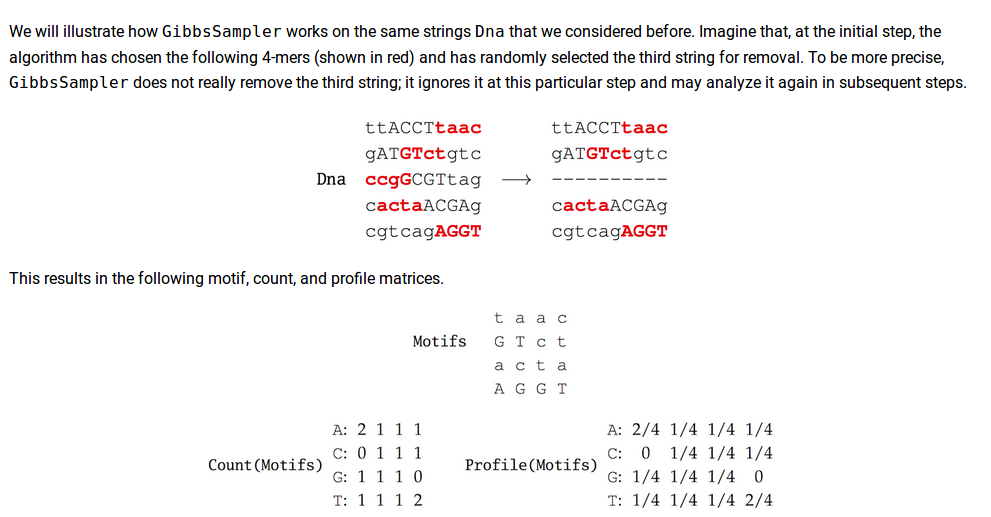
하지만 k-mer의 시작 위치의 수가 많기 때문에 무작위로 motif를 선택하는 전략은 성공적이지 못한 경우가 많다.

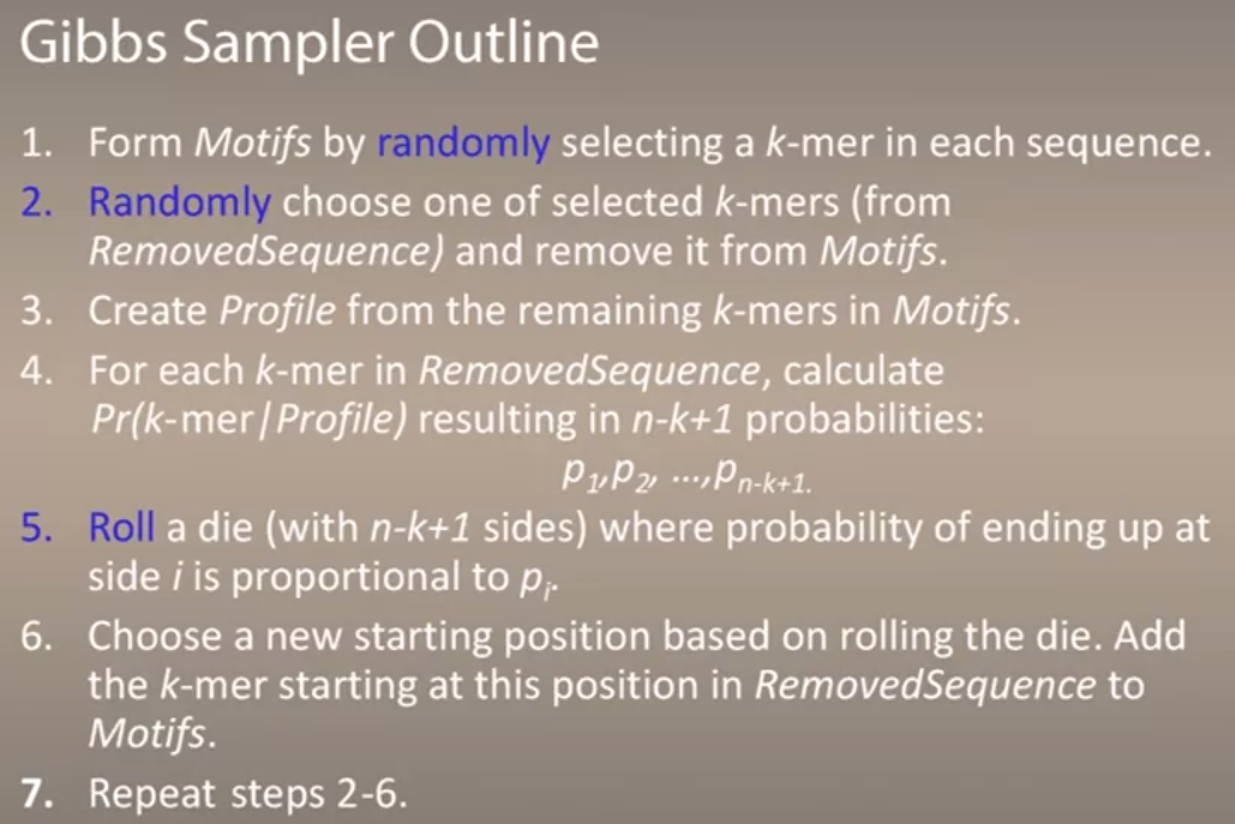
* Gibbs Sampling

결합 분포가 알려져 있지 않으나 각 변수의 조건부 확률분포는 알려져 있을 경우 이에 의존하여 교대로 표본을 채취하는 방법. 특히 MCMC (Markov Chain Monte Carlo) 방법에서 target distribution의 추정에 매우 중요한 역할을 한다.

참조 : blog.naver.com/sw4r/221917843395







--------------------------------------------------------------------------------------------------------------------------

\* reference

https://github.com/vencejo/Curso\_BiologyMeetsProgramming/tree/master/Notebooks