

Origen de replicación

Bioinformática 2025-2

Universidad de Sonora

8 de octubre de 2025

¿Cómo encontrar un gato negro en un cuarto oscuro?

53†††305))6.;4826)4†.)4†);806.;48†8^60))85;161;:†.8
†83(88)5.†;46(;88.96.?.8).†(;485);5.†2: .†(;4956.2(5
.-4)8^8.;4069285);)6†8)4††;1(†9;48081;8:8†1;48†85;4
)485†528806.81(†9;48;(88;4(†?34;48)4†;1†(;:188;†?;

¿Cómo encontrar un gato negro en un cuarto oscuro?

53†††305))6·;**48**26)4†.)4†);806·;**48**†8^60))85;161;:†·8
†83(88)5·†;46(;88·96·?;8)·†(;**48**5);5·†2:·†(;4956·2(5
·-4)8^8·;4069285);)6†8)4††;1(†9;**48**081;8:8†1;**48**†85;4
)485†528806·81(†9;**48**; (88;4(†?34;**48**)4†;1†(;:188;†?;

¿Cómo encontrar un gato negro en un cuarto oscuro?

53†††305))6·THE26)H†.)H†)TE06·THE†E^60))E5T161T:†·E
†E3(EE)5·†TH6(TEE·96·?TE)·†(THE5)T5·†2:·†(TH956·2(5
·-H)E^E·TH0692E5)T)6†E)H††T1(†9THE0E1TE:E†1THE†E5TH
)HE5†52EE06·E1(†9THET(EETH(†?3HTHE)H†T1†(T:1EET†?T

Contar k-meros

PatternCount(*Text*, *Pattern*)

count $\leftarrow 0$

for *i* $\leftarrow 0$ to $|Text| - |Pattern|$

if *Text*(*i*, $|Pattern|$) = *Pattern*

count $\leftarrow count + 1$

return *count*

K-meros más frecuentes

FrequentWords(*Text*, *k*)

FrequentPatterns \leftarrow an empty set

Count \leftarrow an array of length $|Text| - k + 1$

for *i* $\leftarrow 0$ to $|Text| - k$

Pattern $\leftarrow Text(i, k)$

Count(*i*) \leftarrow **PatternCount**(*Text*, *Pattern*)

maxCount \leftarrow maximum value in array *Count*

for *i* $\leftarrow 0$ to $|Text| - k$

if *Count*(*i*) = *maxCount*

 add *Text*(*i*, *k*) to *FrequentPatterns*

remove duplicates from *FrequentPatterns*

return *FrequentPatterns*

Algo más eficiente...

ACG	CGT	GTT	TTT	TTC	TCA	CAC	TTA	TAC	CGG
3	2	2	3	1	1	1	1	1	1

FrequencyTable

```
FrequencyTable(Text, k)  
  freqMap  $\leftarrow$  empty map  
  n  $\leftarrow$  |Text|  
  for i  $\leftarrow$  0 to n - k  
    Pattern  $\leftarrow$  Text(i, k)  
    if freqMap[Pattern] doesn't exist  
      freqMap[Pattern]  $\leftarrow$  1  
    else  
      freqMap[pattern]  $\leftarrow$  freqMap[pattern]+1  
  return freqMap
```


BetterFrequentWords

BetterFrequentWords(*Text*, *k*)

FrequentPatterns \leftarrow an array of strings of length 0

freqMap \leftarrow **FrequencyTable**(*Text*, *k*)

max \leftarrow **MaxMap**(*freqMap*)

for all strings *Pattern* in *freqMap*

if *freqMap*[*pattern*] = *max*

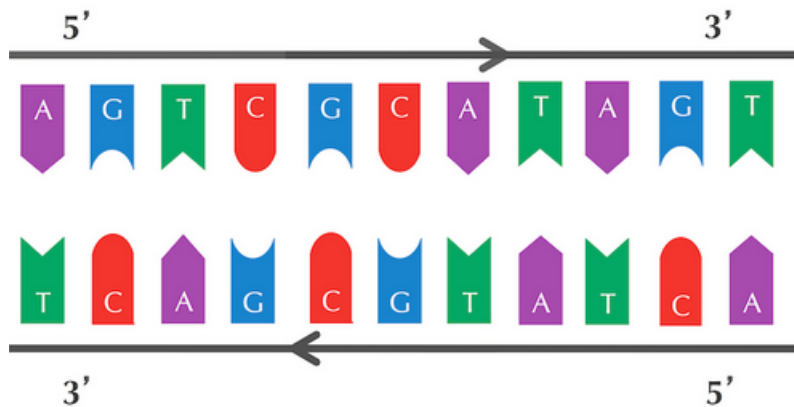
 append *Pattern* to *frequentPatterns*

return *frequentPatterns*

Resultados en el Ori de Vibrio Cholerae

atcaatgatcaacgtaagcttctaagcATGATCAAGgtgctcacacagtttatccacaac
ctgagtggatgacatcaagataggctggtgtatctccttcctctcgtactctcatgacca
cggaaagATGATCAAGagaggatgatttcttggccatatcgcaatgaatacttgtgactt
gtgcttccaattgacatcttcagcgccatattgcgctggccaagggtgacggagcgggatt
acgaaagcatgatcatggctggtgttctgtttatcttgttttgactgagacttgtagga
tagacgggttttcatcactgactagccaaagccttactctgcctgacatcgaccgtaaat
tgataatgaatttacatgcttcgcgacgatttacctcttgatcatcgatccgattgaag
atcttcaattgtaattctcttgccctgactcatagccatgatgagctcttgatcatggt
tccttaaccctctatTTTTTtacggaagaATGATCAAGctgctgctcttgatcatcgtttc

¿Podemos utilizar otro patrón?



Nuevamente, el Ori de Vibrio Cholerae

atcaatgatcaacgtaagcttctaagc**ATGATCAAG**gtgctcacacagtttatccacaac
ctgagtggatgacatcaagataggctggtgtatctccttcctctcgtactctcatgacca
cggaaag**ATGATCAAG**agaggatgatttcttggccatatcgcaatgaatacttgtgactt
gtgcttccaattgacatcttcagcgccatattgcgctggccaagggtgacggagcgggatt
acgaaagcatgatcatggctggtgttctgtttatcttgttttgactgagacttgtagga
tagacgggtttttcatcactgactagccaaagccttactctgcctgacatcgaccgtaaatt
tgataatgaatttacatgcttccgcgacgatttacct**CTTGATCAT**cgatccgattgaag
atcttcaattgttaattctcttgccctcgactcatagccatgatgagct**CTTGATCAT**ggtt
tccttaaccctctattttttacggaaga**ATGATCAAG**ctgctgct**CTTGATCAT**cgtttc

Nuevo problema: localización de los 9-meros.

No solamente nos interesa ahora saber el número de veces que aparece un k-mero, si no donde. Tenemos ahora un nuevo problema computacional:

"Dada una cadena *Genome* y un patrón *Pattern*, encontrar todas las posiciones en las cuales se presenta *Pattern*.

Otro caso: Thermotoga Petrophila

aactctatacctcctttttgtcgaatttgtgtgatttatagagaaaatcttattaactga
aactaaaatggtagggtttGGTGGTAGGttttgtgtacattttgtagtatctgatttttaa
ttacataccgtatattgtatttaaattgacgaacaattgcatggaattgaatatatgcaaa
acaaaCCTACCACCaaactctgtattgaccatttttaggacaacttcagGGTGGTAGGttt
ctgaagctctcatcaatagactatttttagtctttacaaacaatattaccgttcagattca
agattctacaacgctgttttaattgggcgttgagaaaaacttaccacctaataatccagtat
ccaagccgatttcagagaaacctaccacttacctaccacttaCCTACCACCcggttggtga
agttgcagacattattaaaaaacctcatcagaagcttgttcaaaaatttcaatactcgaaa
CCTACCACCTgcgtcccctattattttactactactaataatagcagtataattgatctga

Un nuevo enfoque

Definición. Sean L, k dos números naturales. Decimos que un k -mero *Pattern* forma un (L, t) -grupo dentro de una cadena *Genoma* si existe una subcadena de longitud L dentro de *Genoma* en el cual aparece *Pattern* por lo menos t veces.

Ejemplo

El 4-mero TGCA forma un (25,3)-grupo en la siguiente cadena:

gatcagcataagggtcc**CTGCAATGCAT**GCACAAGCCT**TGCAGT**tgttttac

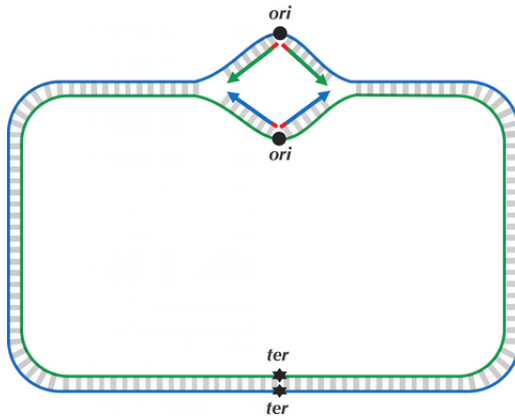
Nuevo problema computacional

Encontrar grupos. Dada una cadena *Genoma* y tres números naturales k, L y t , encontrar todos los distintos k -meros que forme (L, t) -grupos dentro de *Genome*

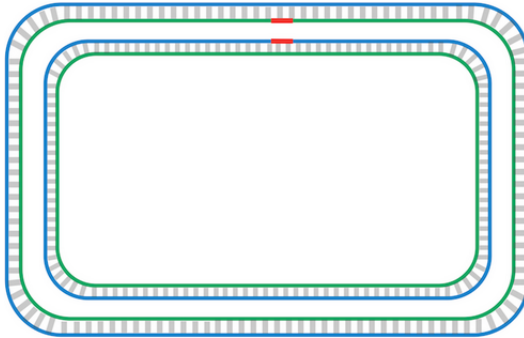
Algoritmo FindClumps

```
FindClumps(Text, k, L, t)  
  Patterns  $\leftarrow$  an array of strings of length 0  
  n  $\leftarrow$  |Text|  
  for every integer i between 0 and n - L  
    Window  $\leftarrow$  Text(i, L)  
    freqMap  $\leftarrow$  FrequencyTable(Window, k)  
    for every key s in freqMap  
      if freqMap[s]  $\geq$  t  
        append s to Patterns  
  remove duplicates from Patterns  
  return Patterns
```

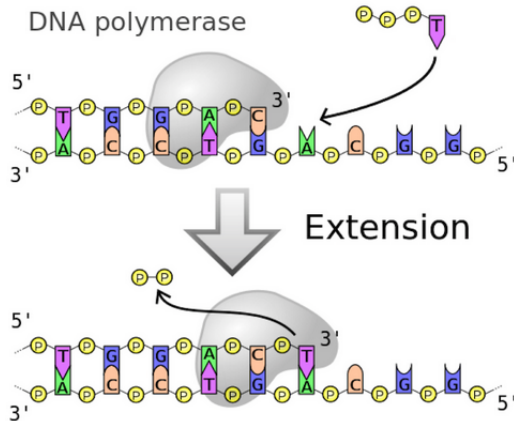
Recordando el proceso de replcación



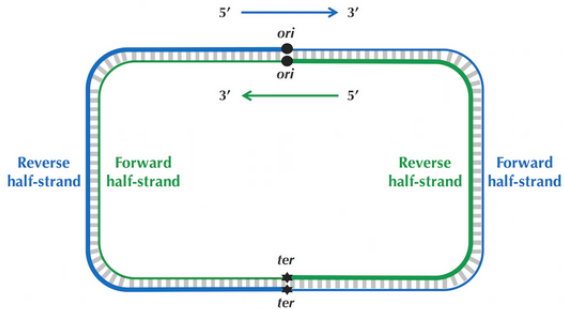
Recordando el proceso de replicación



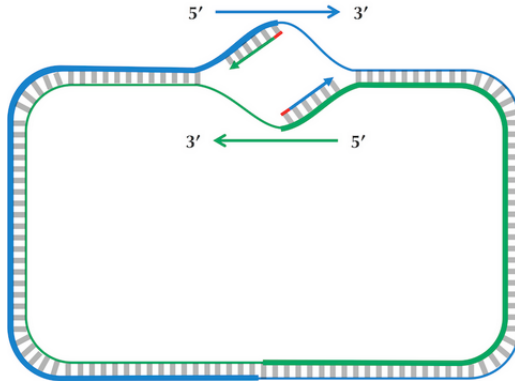
ADN Polimerasa



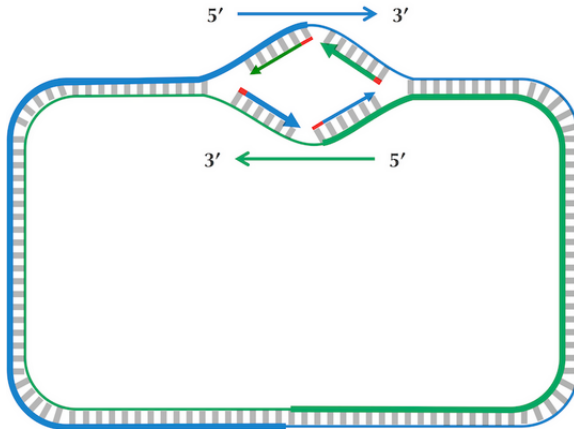
Proceso de replicación



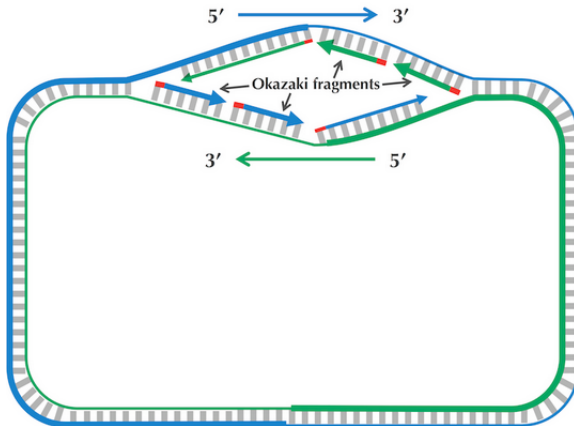
Proceso de replicación



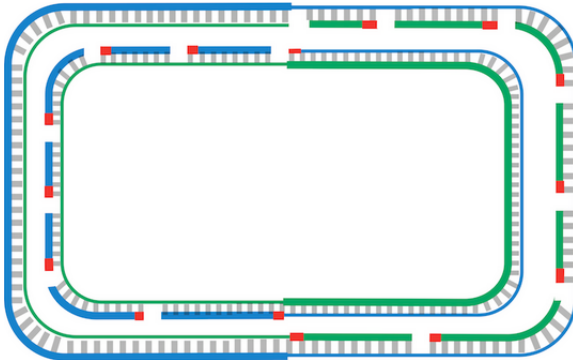
Proceso de replicación



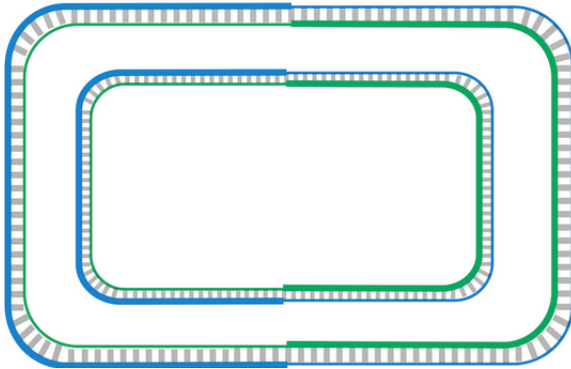
Proceso de replicación



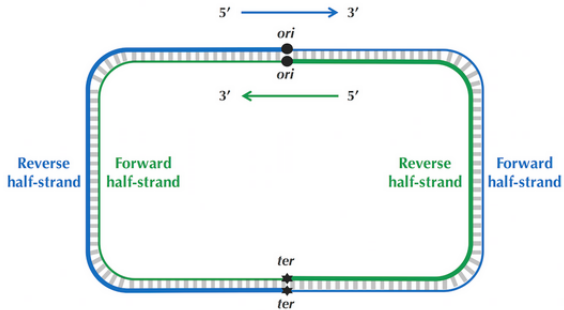
Proceso de replicación



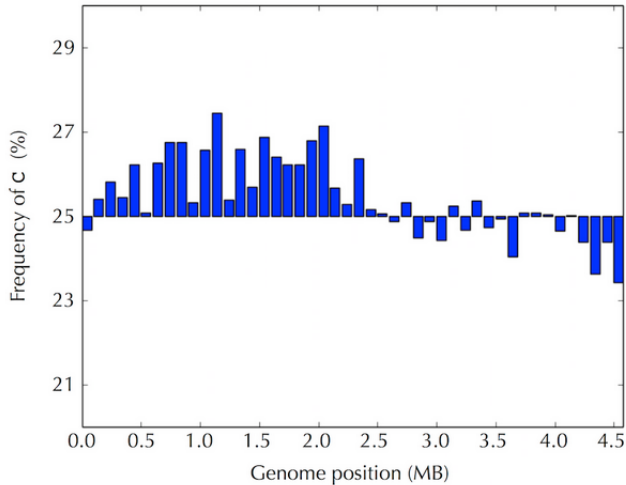
Proceso de replicación



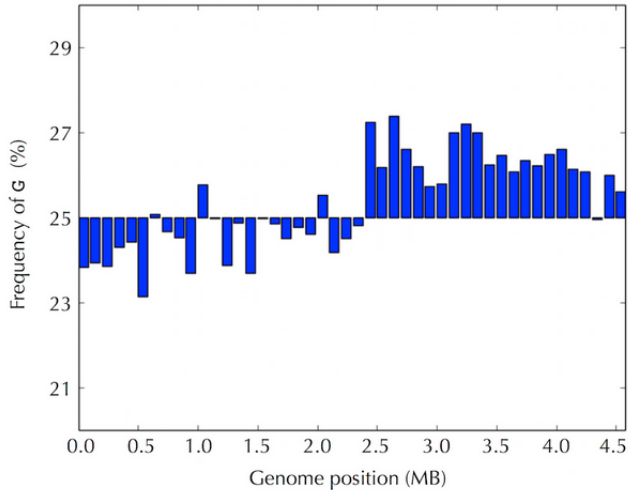
Proceso de replicación



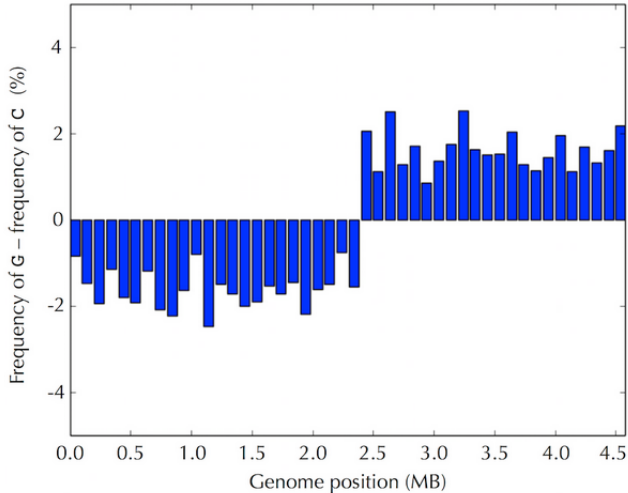
Algunas estadísticas



Algunas estadísticas



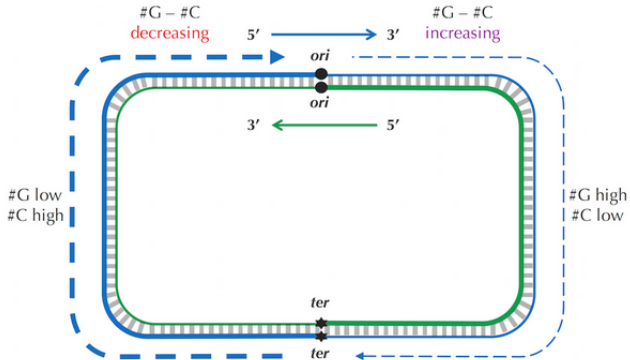
Algunas estadísticas



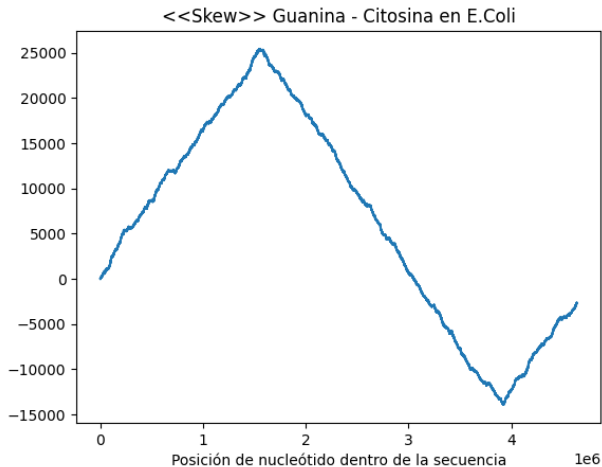
Algunas estadísticas

	#C	#G	#A	#T
Entire strand	427419	413241	491488	491363
Reverse half-strand	219518	201634	243963	246641
Forward half-strand	207901	211607	247525	244722
Difference	+11617	-9973	-3562	+1919

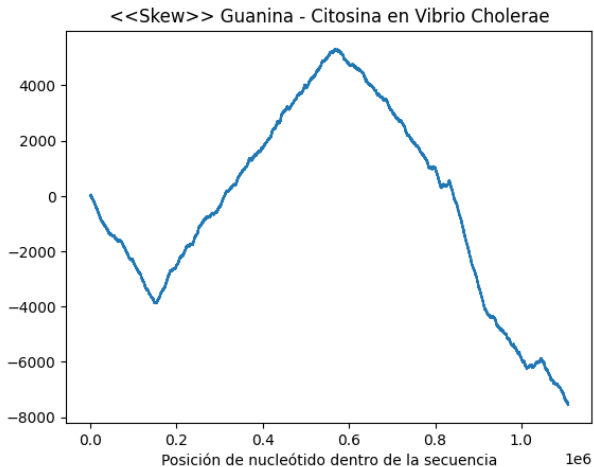
Algunas estadísticas



GC Skew de E. Coli



GC Skew de *Vibrio Cholerae*



Nuevo problema computacional

Dada una cadena de ADN *Genome* encontrar donde el diagrama GC Skew obtiene su valor mínimo (global o local).

¿A que me refiero con mínimo global y local?

Un detalle...

```
aatgatgatgacgtcaaaaggatccggataaaacatggtgattgcctcgc  
ataacgcggtatgaaaatggattgaagcccgggccgtggattctactcaa  
ctttgtcggcttgagaaagacctgggatcctgggtattaaaaagaagatc  
tatttatttagagatctgttctatttgtgatctcttattaggatcgactg  
ccctgtggataacaaggatccggcttttaagatcaacaacctggaaagga  
tcattaactgtgaatgatcggatgatcctggaccgtataagctgggatcag  
aatgaggggttatacacaactcaaaaactgaacaacagttgttctttgga  
taactaccggttgatccaagcttcctgacagagttatccacagtagatcg  
cacgatctgtatacttattttgagtaaattaaccacgatcccagccattc  
ttctgccggatcttcggaatgtcgtgatcaagaatgttgatcttcagtg
```

Un detalle...

atcaATGATCAACgtaagcttctaagcATGATCAAGgtgctcacacagtttatccacaac
ctgagtggatgacatcaagataggtcgttgatatctccttcctctcgtactctcatgacca
cggaaagATGATCAAGagaggatgatttcttgccatatcgcaatgaatacttgtgactt
gtgcttccaattgacatcttcagcgccatattgcgctggccaaggtgacggagcgggatt
acgaaagCATGATCATggctgttgttctgtttatcttgttttgactgagacttgtttagga
tagacgggtttttcatcactgactagccaaagccttactctgcctgacatcgaccgtaa
tgataatgaatttacatgcttcgcgacgatttacctCTTGATCATcgatccgattgaag
atcttcaattgttaattctcttgccctgactcatagccatgatgagctCTTGATCATgtt
tccttaaccctctatTTTTTtacggaagaATGATCAAGctgctgctCTTGATCATcgtttc

Otro problema

Dada una cadena *Genome*, patrón *Pattern* y $d \in \mathbb{N}$ encontrar todas las posiciones de aquellas observaciones que aparezcan como subcadenas en *Genome* tales tengan, a lo más, d diferencias con *Pattern*.