2 GC-skew in bacterial genomes (Projekt 2)

2.1 Implementation

Dependencies To fetch genome sequences I wrote some R functions which use the CGI (HTTP interface) from the "Entrez Programming Utilities" (`http://eutils.ncbi.nlm.nih.gov/`) from NCBI. I implemented their ESearch, ESummary and EFetch functions which return the requested data in XML format. Therefore I'm using the XML library to interpret this data.

If present "Biobase" and "Biostrings" libraries are going to be used (those libraries are recommended but optional, if not present my code will do the calculations using it's own routines). "Biobase" is recommended because it will speed up the GC skew calculations and "Biostrings" because of the **readFASTA** and **writeFASTA** functions, which enable you to load and store sequences to disk easily.

Fetching data A genome sequence can be downloaded using the **retrieve_genome** function, which takes a search term as argument and will return a "FASTA list" (the same format readFASTA returns). As search term you may use any string that can identify a genome (e.g. a string like "e.coli k12 dh10b" or a identifier like the EMBL ID). If the search term matches more than one genome a list of the matches will be shown and you can choose the right one (or cancel). If there are more than 20 matches you are told to be more precise with your search terms, no list will be shown and the function just returns FALSE.

Instead of using the **retrieve_genome** you can of course load a FASTA file on your own, using readFASTA.

GC skew The GC skew of a sequence can be computed with the **GC_skew** function, which takes a sequence as argument and will return a number between -1 and 1. It is using the **skew_characters** function, which counts the "A", "T", "G" and "C" characters in a given string and returns it's frequencies. Those frequencies are used to compute the return value $\frac{\#G-\#C}{\#G+\#C}$.

GC skew (sliding window) To get the GC skew of a sliding window you can use the function **sliding_GC_skew** which takes the arguments *seq* (sequence), *winsize* (window size) and *stepsize* (step size). This function returns a numeric vector of GC skews (one GC per window) which may be used for a plot (see Plots)

GC skew (cumulated) **cumulated_GC_skew** is the function to compute the cumulated GC skew of a given sequence. It needs *seq* (sequence) as argument, the others (n and *step*) are optional:

- n: this is the number of nucleotides of which the cumulated GC skew should be calculated. If not supplied then the length of the sequence is used, which results to the same output as the **GC_skew**(*seq*) would have. n may also be a numeric vector consisting of several nucleotide positions in the sequence, then a vector of the cumulated GC skew of the first n nucleotides for each position of the vector n will be returned. This is useful for a call like **cumulated_GC_skew**(dna, seq(1, nchar(dna), 10000)).
- *step*: it is a shortcut which does the same as n=seq(1, nchar(dna), *step*)

2.2 Plots

2.2.1 sliding and cumulated GC skews for selected genomes

All the following plots have been generated using the **plot_sliding_GC_skew** and the **plot_cumulated_GC_skew** functions.

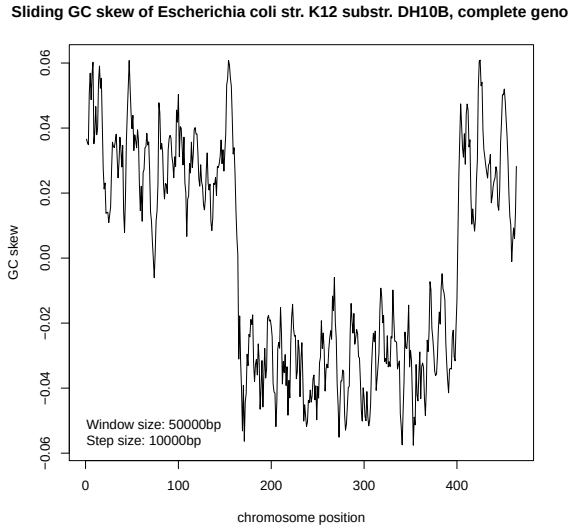

Figure 1: sliding GC skew of the E.coli strain K12, DH10β

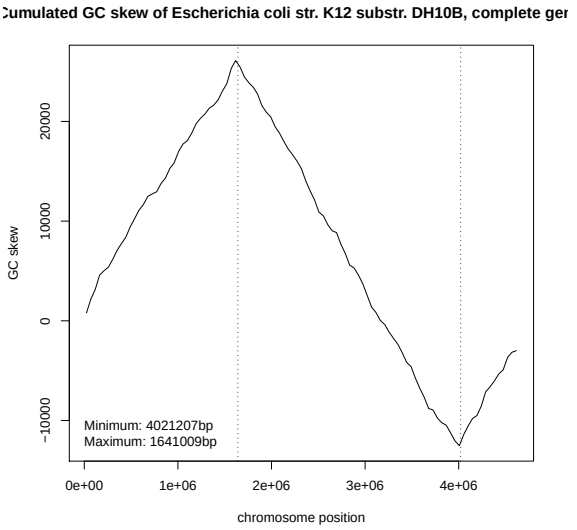

Figure 2: cumulated GC skew of the E.coli strain K12, DH10β

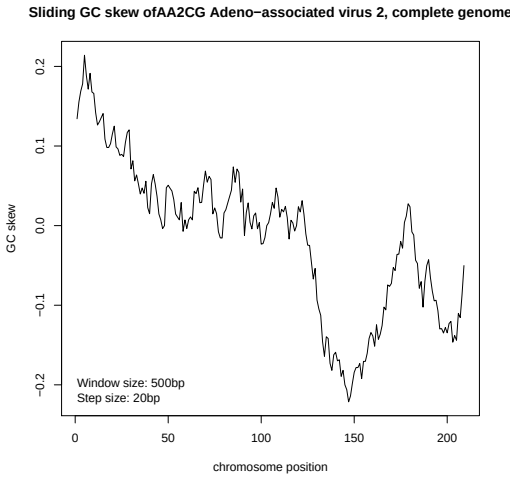

Figure 3: sliding GC skew of the Adeno-associated virus 2 (EMBL: J01901)

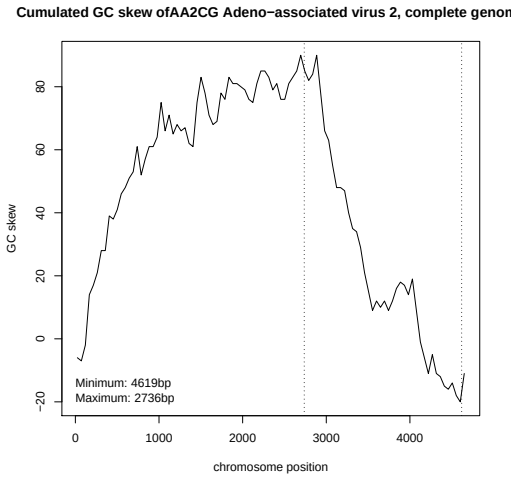

Figure 4: cumulated GC skew of the Adeno-associated virus 2 (EMBL: J01901)

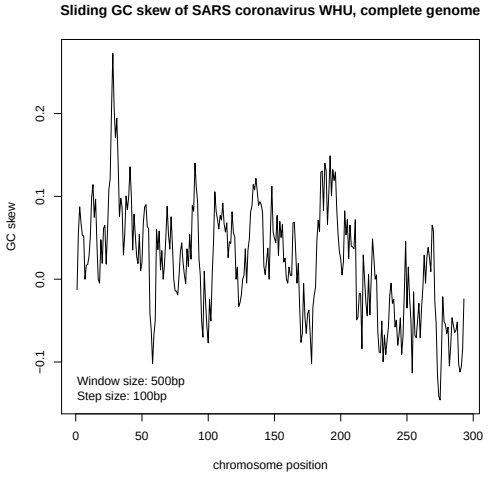

Figure 5: sliding GC skew of the corona virus related to SARS (EMBL: AY394850)

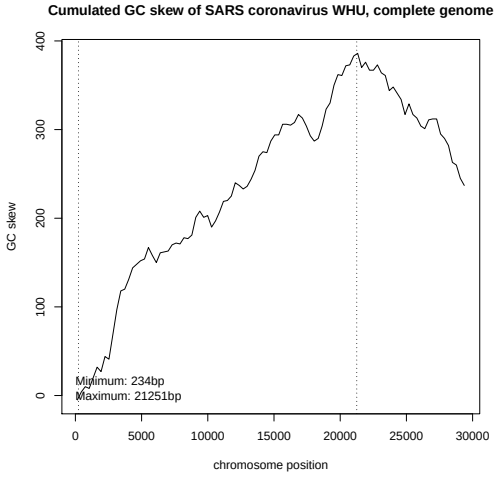

Figure 6: cumulated GC skew of the corona virus related to SARS (EMBL: AY394850)

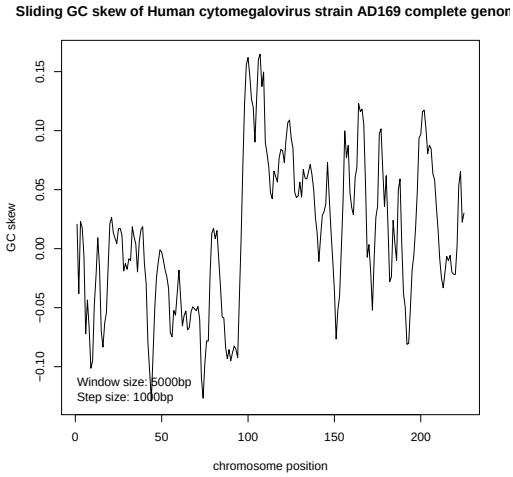

Figure 7: sliding GC skew of Cytomegalovirus strain AD169 (EMBL: X17403)

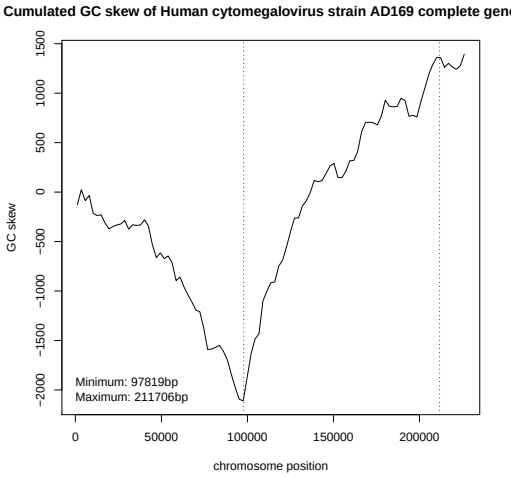

Figure 8: cumulated GC skew of Cytomegalovirus strain AD169 (EMBL: X17403)

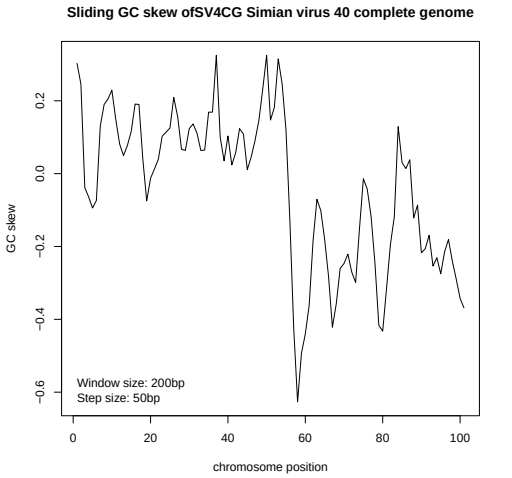

Figure 9: sliding GC skew of Simian (EMBL: J02400)

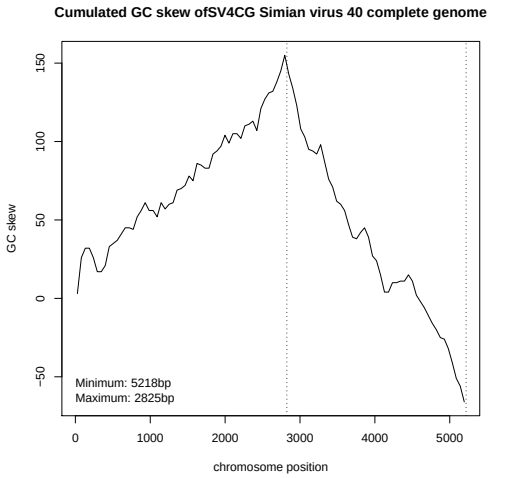

Figure 10: cumulated GC skew of Simian (EMBL: J02400)

2.2.2 effects of changing the window- and stepsize

As you can see in figure 11 on page 6 changing the window- or stepsize has quite important effects on the resulting plot:

- If the window size is too small then the graph is very agitated.
- If the window size is too high then it might be possible that important events are lost in the graph (because it is less detailed).
- If the stepsize is too high the graph will be smoothened too much to be useful.
- If the stepsize is too low it will take very long to calculate the graph.

So you have to choose the right window- and stepsize according to the length of the sequence and your requirements for attention to detail.

2.3 E.coli: Position of the origin of replication (oriC)

As you see on the cumulated GC skew plot of E.coli (figure 2 on page 2) there is a minimum at 4021207bp and a maximum at 1641009bp. Using the **get_annotation** function you can look up these positions in the NCBI annotations database:

```
> ecoli <- retrieve_genome("e.coli k12 dh10b")
> minimum <- plot_cumulated_GC_skew(ecoli, step=10000)$min
> get_annotation(ecoli, minimum, variance=0)
Exact matches:
4021240 4019351 gene
        gene gidA
        locus_tag ECDH10B_3928

4021240 4019351 CDS
        product glucose-inhibited cell-division protein
        protein_id gb|ACB04784.1||gnl|ecoli|ECDH10B_3928
        db_xref ASAP:AEC-0003485
```

This shows that the **gidA** gene is located at the minimum of the cumulated GC skew. So we have identified the position of the replication origin (= "oriC") of Escherichia coli, because the **gidA** gene is located immediately counterclockwise of the oriC on the strand.

At the maximum you will find the replication termination site (= "terC"), which is located on the opposite of the position of the oriC on the genome, because E.coli has a circular genome and the difference between the position of the minimum and maximum is about half of the length of the genome of E.coli.

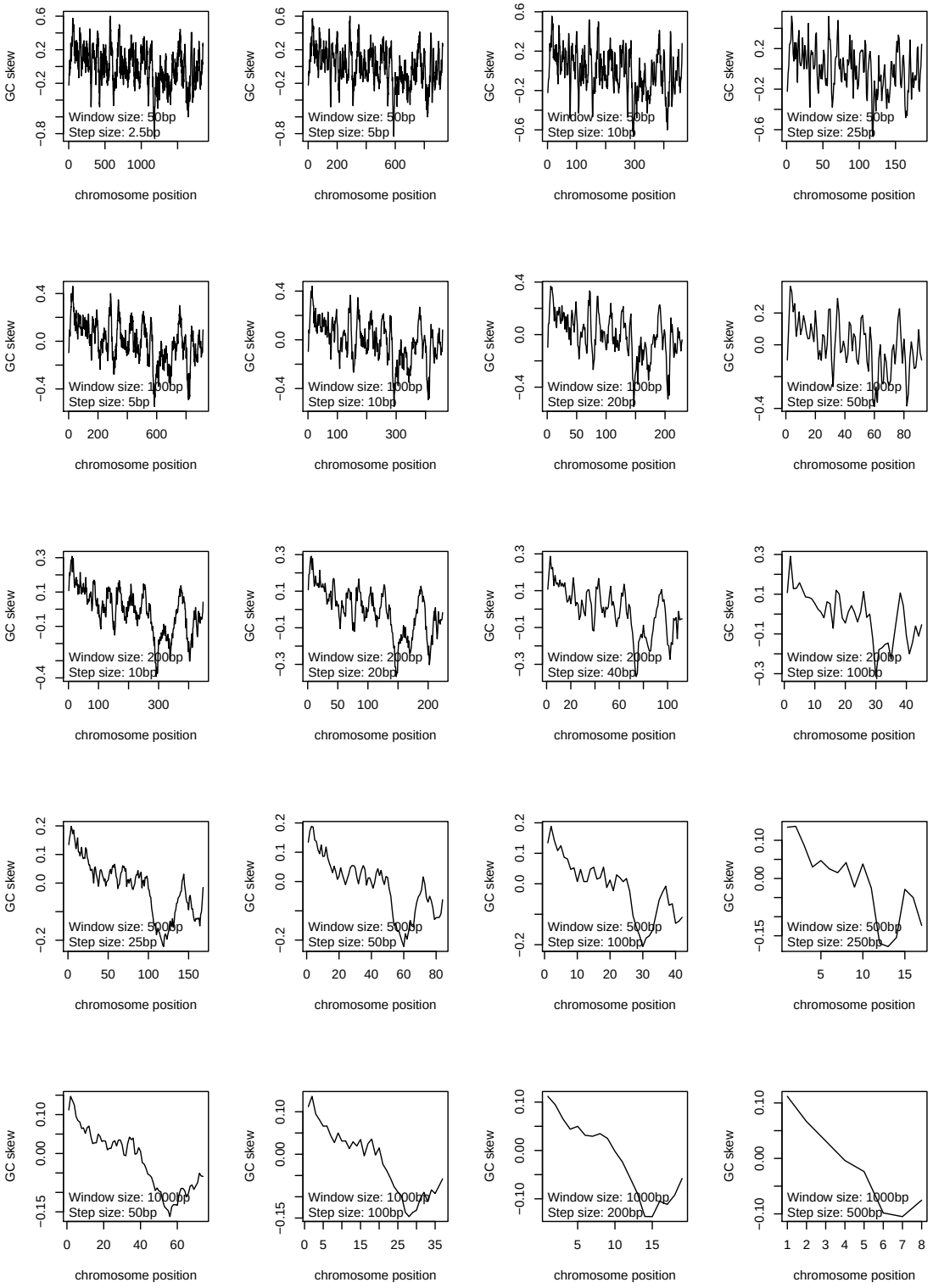

Figure 11: Plots showing the sliding GC skew of the same sequence (Adeno-associated virus 2 (EMBL: J01901)) with different window- and stepsizes