UNIVERSITÀ DI PISA

BWT / eBWT similarity

Giovanna Rosone

University of Pisa, Italy

Dagstuhl Seminar 19241
25 Years of the Burrows-Wheeler Transform

Dagstuhl, June 10 - 14, 2019

Measures based on Burrows-Wheeler Transform

Why should we use the (e)BWT?

- The motivation is the clustering effect that the BWT/eBWT produces.
- The (e)BWT places groups symbols with a similar context close together. Such contexts are near in the sorted list!

Intuitive idea

The greater is the number of substrings shared by two strings, the smaller is their "distance"

Measures based on Burrows-Wheeler Transform

Why should we use the (e)BWT?

- The motivation is the clustering effect that the BWT/eBWT produces.
- The (e)BWT places groups symbols with a similar context close together. Such contexts are near in the sorted list!

Intuitive idea

The greater is the number of substrings shared by two strings, the smaller is their “distance”

The Burrows-Wheeler Transform of more strings

We can extend the notion of BWT to a multiset of words in two ways almost equivalents:

- concatenating all strings of the collection separating each string with a distinct end-marker and computing the BWT of the string so obtained
- computing the extended BWT (EBWT) (also known as *multi-string BWT*) of all strings
  - without concatenating the strings belonging to the collection S
  - allowing sets of strings to be removed or added (for instance merge two eBWTs).

The Burrows-Wheeler Transform of more strings

We can extend the notion of BWT to a multiset of words in two ways almost equivalents:

- concatenating all strings of the collection separating each string with a distinct end-marker and computing the BWT of the string so obtained
- computing the extended BWT (EBWT) (also known as *multi-string BWT*) of all strings
  - without concatenating the strings belonging to the collection S
  - allowing sets of strings to be removed or added (for instance merge two eBWTs).

The Burrows-Wheeler Transform of more strings

We can extend the notion of BWT to a multiset of words in two ways almost equivalents:

- concatenating all strings of the collection separating each string with a distinct end-marker and computing the BWT of the string so obtained
- computing the extended BWT (EBWT) (also known as *multi-string BWT*) of all strings
  - without concatenating the strings belonging to the collection S
  - allowing sets of strings to be removed or added (for instance merge two eBWTs).

The Burrows-Wheeler Transform of more strings

We can extend the notion of BWT to a multiset of words in two ways almost equivalents:

- concatenating all strings of the collection separating each string with a distinct end-marker and computing the BWT of the string so obtained
- computing the extended BWT (EBWT) (also known as *multi-string BWT*) of all strings
  - without concatenating the strings belonging to the collection S
  - allowing sets of strings to be removed or added (for instance merge two eBWTs).

The Burrows-Wheeler Transform of more strings

We can extend the notion of BWT to a multiset of words in two ways almost equivalents:

- concatenating all strings of the collection separating each string with a distinct end-marker and computing the BWT of the string so obtained
- computing the extended BWT (EBWT) (also known as *multi-string BWT*) of all strings
  - without concatenating the strings belonging to the collection S
  - allowing sets of strings to be removed or added (for instance merge two eBWTs).

The extended Burrows-Wheeler Transform (eBWT)

Two variants:

1. (circular sorting) eBWT [*Mantaci, Restivo, R. and Sciortino*, 2007]: sorting the conjugates (cyclic rotations) of the input strings by using the lexicographic order on infinite words
   - useful for application where the input strings are circular (for instance mitochondrial dna, ...)
   - the strings in the collection are not ordered.
2. (linear sorting) eBWT [Bauer, Cox and R., 2013]: sorting the suffixes of all words by using the usual lexicographic order (but one needs to append an (implicit) distinct end-marker to each string)
   - useful for application where the input strings are linear (for instance books, NGS libraries, ...)
   - the strings in the collection are ordered.

The extended Burrows-Wheeler Transform (eBWT)

Two variants:

1. (circular sorting) eBWT [*Mantaci, Restivo, R. and Sciortino*, 2007]: sorting the conjugates (cyclic rotations) of the input strings by using the lexicographic order on infinite words
   - useful for application where the input strings are circular (for instance mitochondrial dna, . . .)
   - the strings in the collection are not ordered.
2. (linear sorting) eBWT [Bauer, Cox and R., 2013]: sorting the suffixes of all words by using the usual lexicographic order (but one needs to append an (implicit) distinct end-marker to each string)
   - useful for application where the input strings are linear (for instance books, NGS libraries, . . .)
   - the strings in the collection are ordered.

The extended Burrows-Wheeler Transform (eBWT)

Two variants:

1. (circular sorting) eBWT [*Mantaci, Restivo, R. and Sciortino*, 2007]: sorting the conjugates (cyclic rotations) of the input strings by using the lexicographic order on infinite words
   - useful for application where the input strings are circular (for instance mitochondrial dna, . . .)
   - the strings in the collection are not ordered.
2. (linear sorting) eBWT [Bauer, Cox and R., 2013]: sorting the suffixes of all words by using the usual lexicographic order (but one needs to append an (implicit) distinct end-marker to each string)
   - useful for application where the input strings are linear (for instance books, NGS libraries, . . .)
   - the strings in the collection are ordered.

The extended Burrows-Wheeler Transform (eBWT)

Two variants:

1. (circular sorting) eBWT [*Mantaci, Restivo, R. and Sciortino*, 2007]: sorting the conjugates (cyclic rotations) of the input strings by using the lexicographic order on infinite words
   - useful for application where the input strings are circular (for instance mitochondrial dna, ...)
   - the strings in the collection are not ordered.
2. (linear sorting) eBWT [Bauer, Cox and R., 2013]: sorting the suffixes of all words by using the usual lexicographic order (but one needs to append an (implicit) distinct end-marker to each string)
   - useful for application where the input strings are linear (for instance books, NGS libraries, ...)
   - the strings in the collection are ordered.

eBWT [Bauer, Cox and R., 2013]

Let S be the set of the words that end with the end-markers.
We use implicit distinct end markers, i.e. $\$_1 = \$_2 = \$_3 = \$$: we use the position of the words in the collection in order to establish the order relation between two identical suffixes.

Collection S

	0	1	2	3	4	5	6
S_1	G	C	C	A	A	C	$\$_1$
S_2	G	A	G	C	T	C	$\$_2$
S_3	T	C	G	C	T	T	$\$_3$

$eBWT$	Sorted Suffixes of S
C	$\$_1$
C	$\$_2$
T	$\$_3$
C	$AAC\$_1$
A	$AC\$_1$
G	$AGCTC\$_2$
A	$C\$_1$
T	$C\$_2$
C	$CAAC\$_1$
G	$CCAAC\$_1$
T	$CGCTT\$_3$
G	$CTC\$_2$
G	$CTT\$_3$
$\$_2$	$GAGCTC\$_2$
$\$_1$	$GCCAAC\$_1$
A	$GCTC\$_2$
C	$GCTT\$_3$
T	$T\$_3$
C	$TC\$_2$
$\$_3$	$TCGCTT\$_3$
C	$TT\$_3$

$ebwt(S)$ is a permutation of the symbols in S, obtained as concatenation of the symbols that (circularly) precede the first symbol of the suffix in the list of its lexicographically sorted suffixes of S.

The use of ordered and distinct "end-marker" symbols makes the multiset of sequences an ordered collection.

So the identical or similar sequences could be distant in the collection.

This can make the difference in the clustering effect!!!

eBWT [Bauer, Cox and R., 2013]

Let S be the set of the words that end with the end-markers.
We use implicit distinct end markers, i.e. $\$_1 = \$_2 = \$_3 = \$$: we use the position of the words in the collection in order to establish the order relation between two identical suffixes.

Collection S

	0	1	2	3	4	5	6
S_1	G	C	C	A	A	C	$\$_1$
S_2	G	A	G	C	T	C	$\$_2$
S_3	T	C	G	C	T	T	$\$_3$

$ebwt$(S) is a permutation of the symbols in S, obtained as concatenation of the symbols that (circularly) precede the first symbol of the suffix in the list of its lexicographically sorted suffixes of S.

The use of ordered and distinct "end-marker" symbols makes the multiset of sequences an ordered collection.

So the identical or similar sequences could be distant in the collection.

This can make the difference in the clustering effect!!!

$eBWT$	Sorted Suffixes of S
C	$\$_1$
C	$\$_2$
T	$\$_3$
C	$AAC\$_1$
A	$AC\$_1$
G	$AGCTC\$_2$
A	$C\$_1$
T	$C\$_2$
C	$CAAC\$_1$
G	$CCAAC\$_1$
T	$CGCTT\$_3$
G	$CTC\$_2$
G	$CTT\$_3$
$\$_2$	$GAGCTC\$_2$
$\$_1$	$GCCAAC\$_1$
A	$GCTC\$_2$
C	$GCTT\$_3$
T	$T\$_3$
C	$TC\$_2$
$\$_3$	$TCGCTT\$_3$
C	$TT\$_3$

eBWT [Bauer, Cox and R., 2013]

Let S be the set of the words that end with the end-markers.
We use implicit distinct end markers, i.e. $\$_1 = \$_2 = \$_3 = \$$: we use the position of the words in the collection in order to establish the order relation between two identical suffixes.

Collection S

	0	1	2	3	4	5	6
S_1	G	C	C	A	A	C	$\$_1$
S_2	G	A	G	C	T	C	$\$_2$
S_3	T	C	G	C	T	T	$\$_3$

$ebwt$(S) is a permutation of the symbols in S, obtained as concatenation of the symbols that (circularly) precede the first symbol of the suffix in the list of its lexicographically sorted suffixes of S.

The use of ordered and distinct "end-marker" symbols makes the multiset of sequences an ordered collection.

So the identical or similar sequences could be distant in the collection.

This can make the difference in the clustering effect!!!

$eBWT$	Sorted Suffixes of S
C	$\$_1$
C	$\$_2$
T	$\$_3$
C	$AAC\$_1$
A	$AC\$_1$
G	$AGCTC\$_2$
A	$C\$_1$
T	$C\$_2$
C	$CAAC\$_1$
G	$CCAAC\$_1$
T	$CGCTT\$_3$
G	$CTC\$_2$
G	$CTT\$_3$
$\$_2$	$GAGCTC\$_2$
$\$_1$	$GCCAAC\$_1$
A	$GCTC\$_2$
C	$GCTT\$_3$
T	$T\$_3$
C	$TC\$_2$
$\$_3$	$TCGCTT\$_3$
C	$TT\$_3$

eBWT [Bauer, Cox and R., 2013]

Let S be the set of the words that end with the end-markers.
We use implicit distinct end markers, i.e. $\$_1 = \$_2 = \$_3 = \$$: we use the position of the words in the collection in order to establish the order relation between two identical suffixes.

Collection S

	0	1	2	3	4	5	6
S_1	G	C	C	A	A	C	$\$_1$
S_2	G	A	G	C	T	C	$\$_2$
S_3	T	C	G	C	T	T	$\$_3$

$ebwt$(S) is a permutation of the symbols in S, obtained as concatenation of the symbols that (circularly) precede the first symbol of the suffix in the list of its lexicographically sorted suffixes of S.

The use of ordered and distinct "end-marker" symbols makes the multiset of sequences an ordered collection.

So the identical or similar sequences could be distant in the collection.

This can make the difference in the clustering effect!!!

$eBWT$	Sorted Suffixes of S
C	$\$_1$
C	$\$_2$
T	$\$_3$
C	$AAC\$_1$
A	$AC\$_1$
G	$AGCTC\$_2$
A	$C\$_1$
T	$C\$_2$
C	$CAAC\$_1$
G	$CCAAC\$_1$
T	$CGCTT\$_3$
G	$CTC\$_2$
G	$CTT\$_3$
$\$_2$	$GAGCTC\$_2$
$\$_1$	$GCCAAC\$_1$
A	$GCTC\$_2$
C	$GCTT\$_3$
T	$T\$_3$
C	$TC\$_2$
$\$_3$	$TCGCTT\$_3$
C	$TT\$_3$

Example: swapping sequences

S $= \{TAGA\underline{C}CT, TACC\underline{A}CT, GAGACCT\}$

$EBWT$	Sorted Suffixes
T	$\$$
T	$\$$
T	$\$$
T	$ACCACT\$$
G	$ACCT\$$
G	$ACCT\$$
C	$ACT\$$
T	$AGACCT\$$
G	$AGACCT\$$
C	$CACT\$$
A	$CCACT\$$
A	$CCT\$$
A	$CCT\$$
$\underline{C}$	$\mathbf{CT}\$$
$\underline{A}$	$\mathbf{CT}\$$
C	$\mathbf{CT}\$$
A	$GACCT\$$
A	$GACCT\$$
$\$$	$GAGACCT\$$
C	$T\$$
C	$T\$$
C	$T\$$
$\$$	$TACCACT\$$
$\$$	$TAGACCT\$$

S' $= \{TACC\underline{A}CT, TAGA\underline{C}CT, GAGACCT\}$

$EBWT$	Sorted Suffixes
T	$\$$
T	$\$$
T	$\$$
T	$ACCACT\$$
G	$ACCT\$$
G	$ACCT\$$
C	$ACT\$$
T	$AGACCT\$$
G	$AGACCT\$$
C	$CACT\$$
A	$CCACT\$$
A	$CCT\$$
A	$CCT\$$
$\underline{A}$	$\mathbf{CT}\$$
$\underline{C}$	$\mathbf{CT}\$$
C	$\mathbf{CT}\$$
A	$GACCT\$$
A	$GACCT\$$
$\$$	$GAGACCT\$$
C	$T\$$
C	$T\$$
C	$T\$$
$\$$	$TACCACT\$$
$\$$	$TAGACCT\$$

Example: swapping sequences

$\mathsf{S} = \{TAGA\underline{C}CT, TACC\underline{A}CT, GAGACCT\}$

$EBWT$	Sorted Suffixes
T	$\$$
T	$\$$
T	$\$$
T	$ACCACT\$$
G	$ACCT\$$
G	$ACCT\$$
C	$ACT\$$
T	$AGACCT\$$
G	$AGACCT\$$
C	$CACT\$$
A	$CCACT\$$
A	$CCT\$$
A	$CCT\$$
$\underline{C}$	$\mathbf{CT}\$$
$\underline{A}$	$\mathbf{CT}\$$
C	$\mathbf{CT}\$$
A	$GACCT\$$
A	$GACCT\$$
$\$$	$GAGACCT\$$
C	$T\$$
C	$T\$$
C	$T\$$
$\$$	$TACCACT\$$
$\$$	$TAGACCT\$$

$\mathsf{S}' = \{TACC\underline{A}CT, TAGA\underline{C}CT, GAGACCT\}$

$EBWT$	Sorted Suffixes
T	$\$$
T	$\$$
T	$\$$
T	$ACCACT\$$
G	$ACCT\$$
G	$ACCT\$$
C	$ACT\$$
T	$AGACCT\$$
G	$AGACCT\$$
C	$CACT\$$
A	$CCACT\$$
A	$CCT\$$
A	$CCT\$$
$\underline{A}$	$\mathbf{CT}\$$
$\underline{C}$	$\mathbf{CT}\$$
C	$\mathbf{CT}\$$
A	$GACCT\$$
A	$GACCT\$$
$\$$	$GAGACCT\$$
C	$T\$$
C	$T\$$
C	$T\$$
$\$$	$TACCACT\$$
$\$$	$TAGACCT\$$

Example: swapping sequences

$S = \{TAGA\underline{C}CT, TACC\underline{A}CT, GAGACCT\}$

$EBWT$	Sorted Suffixes
T	\$
T	\$
T	\$
T	$ACCACT\$$
G	$ACCT\$$
G	$ACCT\$$
C	$ACT\$$
T	$AGACCT\$$
G	$AGACCT\$$
C	$CACT\$$
A	$CCACT\$$
A	$CCT\$$
A	$CCT\$$
$\underline{C}$	**CT**\$
$\underline{A}$	**CT**\$
C	**CT**\$
A	$GACCT\$$
A	$GACCT\$$
\$	$GAGACCT\$$
C	$T\$$
C	$T\$$
C	$T\$$
\$	$TACCACT\$$
\$	$TAGACCT\$$

$S' = \{TACC\underline{A}CT, TAGA\underline{C}CT, GAGACCT\}$

$EBWT$	Sorted Suffixes
T	\$
T	\$
T	\$
T	$ACCACT\$$
G	$ACCT\$$
G	$ACCT\$$
C	$ACT\$$
T	$AGACCT\$$
G	$AGACCT\$$
C	$CACT\$$
A	$CCACT\$$
A	$CCT\$$
A	$CCT\$$
$\underline{A}$	**CT**\$
$\underline{C}$	**CT**\$
C	**CT**\$
A	$GACCT\$$
A	$GACCT\$$
\$	$GAGACCT\$$
C	$T\$$
C	$T\$$
C	$T\$$
\$	$TACCACT\$$
\$	$TAGACCT\$$

Reordering [Cox, Bauer, Jakobi and R, 2012]

Ordered collection: S = $\{TAGA\underline{C}CT, TACC\underline{A}CT, GAGACCT\}$

$EBWT$	$Suffixes$
T	\$
T	\$
T	\$
T	$ACCACT\$$
G	$ACCT\$$
G	$ACCT\$$
C	$ACT\$$
T	$AGACCT\$$
G	$AGACCT\$$
C	$CACT\$$
A	$CCACT\$$
A	$CCT\$$
A	$CCT\$$
C	$CT\$$
A	$CT\$$
C	$CT\$$
A	$GACCT\$$
A	$GACCT\$$
\$	$GAGACCT\$$
C	$T\$$
C	$T\$$
C	$T\$$
\$	$TACCACT\$$
\$	$TAGACCT\$$

In these regions, when the non-\$ suffixes are the same, the ordering of the symbols in eBWT depends on the ordering of the sequences in the collection.

- BCR can swap the sequences $TAGACCT$ and $TACCACT$ in the ordered collection
- by swapping the symbols C and A directly in the EBWT during its construction [Cox, Bauer, Jakobi and R, 2012]

Note that the rest of EBWT is unaffected by this change in ordering.

Reordering [Cox, Bauer, Jakobi and R, 2012]

Ordered collection: S $= \{TAGA\underline{C}CT, TACC\underline{A}CT, GAGACCT\}$

$EBWT$	$Suffixes$
T	\$
T	\$
T	\$
T	$ACCACT\$$
G	$ACCT\$$
G	$ACCT\$$
C	$ACT\$$
T	$AGACCT\$$
G	$AGACCT\$$
C	$CACT\$$
A	$CCACT\$$
A	$CCT\$$
A	$CCT\$$
C	$CT\$$
A	$CT\$$
C	$CT\$$
A	$GACCT\$$
A	$GACCT\$$
\$	$GAGACCT\$$
C	$T\$$
C	$T\$$
C	$T\$$
\$	$TACCACT\$$
\$	$TAGACCT\$$

In these regions, when the non-\$ suffixes are the same, the ordering of the symbols in eBWT depends on the ordering of the sequences in the collection.

- BCR can swap the sequences $TAGACCT$ and $TACCACT$ in the ordered collection
- by swapping the symbols C and A directly in the EBWT during its construction [Cox, Bauer, Jakobi and R, 2012]

Note that the rest of EBWT is unaffected by this change in ordering.

Reordering [Cox, Bauer, Jakobi and R, 2012]

Ordered collection: S $= \{TAGACCT, TACCACT, GAGACCT\}$

$EBWT$	$Suffixes$
T	\$
T	\$
T	\$
T	$ACCACT\$$
G	$ACCT\$$
G	$ACCT\$$
C	$ACT\$$
T	$AGACCT\$$
G	$AGACCT\$$
C	$CACT\$$
A	$CCACT\$$
A	$CCT\$$
A	$CCT\$$
C	$CT\$$
A	$CT\$$
C	$CT\$$
A	$GACCT\$$
A	$GACCT\$$
\$	$GAGACCT\$$
C	$T\$$
C	$T\$$
C	$T\$$
\$	$TACCACT\$$
\$	$TAGACCT\$$

In these regions, when the non-\$ suffixes are the same, the ordering of the symbols in eBWT depends on the ordering of the sequences in the collection.

- BCR can swap the sequences $TAGACCT$ and $TACCACT$ in the ordered collection
- by swapping the symbols C and A directly in the EBWT during its construction [Cox, Bauer, Jakobi and R, 2012]

Note that the rest of EBWT is unaffected by this change in ordering.

Reordering [Cox, Bauer, Jakobi and R, 2012]

Ordered collection: $\mathsf{S} = \{TACC\underline{A}CT, TAGA\underline{C}CT, GAGACCT\}$

$EBWT$	$Suffixes$
T	\$
T	\$
T	\$
T	$ACCACT\$$
G	$ACCT\$$
G	$ACCT\$$
C	$ACT\$$
T	$AGACCT\$$
G	$AGACCT\$$
C	$CACT\$$
A	$CCACT\$$
A	$CCT\$$
A	$CCT\$$
$\underline{A}$	$CT\$$
$\underline{C}$	$CT\$$
C	$CT\$$
A	$GACCT\$$
A	$GACCT\$$
\$	$GAGACCT\$$
C	$T\$$
C	$T\$$
C	$T\$$
\$	$TACCACT\$$
\$	$TAGACCT\$$

In these regions, when the non-\$ suffixes are the same, the ordering of the symbols in eBWT depends on the ordering of the sequences in the collection.

- BCR can swap the sequences $TAGACCT$ and $TACCACT$ in the ordered collection
- by swapping the symbols C and A directly in the EBWT during its construction [Cox, Bauer, Jakobi and R, 2012]

Note that the rest of EBWT is unaffected by this change in ordering.

The document array

For simplicity, we use the document array $DA(\mathsf{S})$: a sequence of "colors" that depends on how the suffixes of S are mixed in the sorted list.

Collection S

	0	1	2	3	4	5	6
S_1	G	C	C	A	A	C	$\$_1$
S_2	G	A	G	C	T	C	$\$_2$
S_3	T	C	G	C	T	T	$\$_3$

$DA[i] = h$: if i-th smallest suffix in the sorted collection belongs to the string S_h

Note that the DA is not necessary, one can use the LF-mapping starting from each $\$_i$.

DA	$eBWT$	Sorted Suffixes of S
1	C	$\$_1$
2	C	$\$_2$
3	T	$\$_3$
1	C	$AAC\$_1$
1	A	$AC\$_1$
2	G	$AGCTC\$_2$
1	A	$C\$_1$
2	T	$C\$_2$
1	C	$CAAC\$_1$
1	G	$CCAAC\$_1$
3	T	$CGCTT\$_3$
2	G	$CTC\$_2$
3	G	$CTT\$_3$
2	$\$_2$	$GAGCTC\$_2$
1	$\$_1$	$GCCAAC\$_1$
2	A	$GCTC\$_2$
3	C	$GCTT\$_3$
3	T	$T\$_3$
2	C	$TC\$_2$
2	$\$_3$	$TCGCTT\$_3$
3	C	$TT\$_3$

The document array

For simplicity, we use the document array $DA(\mathsf{S})$: a sequence of "colors" that depends on how the suffixes of S are mixed in the sorted list.

Collection S

	0	1	2	3	4	5	6
S_1	G	C	C	A	A	C	$\$_1$
S_2	G	A	G	C	T	C	$\$_2$
S_3	T	C	G	C	T	T	$\$_3$

$DA[i] = h$: if i-th smallest suffix in the sorted collection belongs to the string S_h

Note that the DA is not necessary, one can use the LF-mapping starting from each $\$_i$.

DA	$eBWT$	Sorted Suffixes of S
1	C	$\$_1$
2	C	$\$_2$
3	T	$\$_3$
1	C	$AAC\$_1$
1	A	$AC\$_1$
2	G	$AGCTC\$_2$
1	A	$C\$_1$
2	T	$C\$_2$
1	C	$CAAC\$_1$
1	G	$CCAAC\$_1$
3	T	$CGCTT\$_3$
2	G	$CTC\$_2$
3	G	$CTT\$_3$
2	$\$_2$	$GAGCTC\$_2$
1	$\$_1$	$GCCAAC\$_1$
2	A	$GCTC\$_2$
3	C	$GCTT\$_3$
3	T	$T\$_3$
2	C	$TC\$_2$
2	$\$_3$	$TCGCTT\$_3$
3	C	$TT\$_3$

The document array

For simplicity, we use the document array $DA(\mathsf{S})$: a sequence of "colors" that depends on how the suffixes of S are mixed in the sorted list.

Collection S

	0	1	2	3	4	5	6
S_1	G	C	C	A	A	C	$\$_1$
S_2	G	A	G	C	T	C	$\$_2$
S_3	T	C	G	C	T	T	$\$_3$

$DA[i] = h$: if i-th smallest suffix in the sorted collection belongs to the string S_h

Note that the DA is not necessary, one can use the LF-mapping starting from each $\$_i$.

DA	$eBWT$	Sorted Suffixes of S
1	C	$\$_1$
2	C	$\$_2$
3	T	$\$_3$
1	C	$AAC\$_1$
1	A	$AC\$_1$
2	G	$AGCTC\$_2$
1	A	$C\$_1$
2	T	$C\$_2$
1	C	$CAAC\$_1$
1	G	$CCAAC\$_1$
3	T	$CGCTT\$_3$
2	G	$CTC\$_2$
3	G	$CTT\$_3$
2	$\$_2$	$GAGCTC\$_2$
1	$\$_1$	$GCCAAC\$_1$
2	A	$GCTC\$_2$
3	C	$GCTT\$_3$
3	T	$T\$_3$
2	C	$TC\$_2$
2	$\$_3$	$TCGCTT\$_3$
3	C	$TT\$_3$

Projections on two strings

One can compute the eBWT of the whole collection and analyze all pairs at the same time or one can get a projection of the two selected strings.

Collection S

	0	1	2	3	4	5	6
S_1	G	C	C	A	A	C	$\$_1$
S_2	G	A	G	C	T	C	$\$_2$
S_3	T	C	G	C	T	T	$\$_3$

In order to remove S_2, we only need to remove the blue symbols (associated with the suffixes of S_2) in $eBWT$.

Actually, we don't need to store the colors, for instance the symbols of S_2 can be found by using the LF-mapping starting from $\$_2$.

$eBWT$	Sorted Suffixes of S
C	$\$_1$
C	$\$_2$
T	$\$_3$
C	$AAC\$_1$
A	$AC\$_1$
G	$AGCTC\$_2$
A	$C\$_1$
T	$C\$_2$
C	$CAAC\$_1$
G	$CCAAC\$_1$
T	$CGCTT\$_3$
G	$CTC\$_2$
G	$CTT\$_3$
$\$_2$	$GAGCTC\$_2$
$\$_1$	$GCCAAC\$_1$
A	$GCTC\$_2$
C	$GCTT\$_3$
T	$T\$_3$
C	$TC\$_2$
$\$_3$	$TCGCTT\$_3$
C	$TT\$_3$

$eBWT$	Sorted Suffixes of S
C	$\$_1$
T	$\$_3$
C	$AAC\$_1$
A	$AC\$_1$
A	$C\$_1$
C	$CAAC\$_1$
G	$CCAAC\$_1$
T	$CGCTT\$_3$
G	$CTT\$_3$
$\$_1$	$GCCAAC\$_1$
C	$GCTT\$_3$
T	$T\$_3$
$\$_3$	$TCGCTT\$_3$
C	$TT\$_3$

Projections on two strings

One can compute the eBWT of the whole collection and analyze all pairs at the same time or one can get a projection of the two selected strings.

Collection S

	0	1	2	3	4	5	6
S_1	G	C	C	A	A	C	$\$_1$
S_2	G	A	G	C	T	C	$\$_2$
S_3	T	C	G	C	T	T	$\$_3$

In order to remove S_2, we only need to remove the blue symbols (associated with the suffixes of S_2) in $eBWT$.

Actually, we don't need to store the colors, for instance the symbols of S_2 can be found by using the LF-mapping starting from $\$_2$.

$eBWT$	Sorted Suffixes of S
C	$\$_1$
C	$\$_2$
T	$\$_3$
C	$AAC\$_1$
A	$AC\$_1$
G	$AGCTC\$_2$
A	$C\$_1$
T	$C\$_2$
C	$CAAC\$_1$
G	$CCAAC\$_1$
T	$CGCTT\$_3$
G	$CTC\$_2$
G	$CTT\$_3$
$\$_2$	$GAGCTC\$_2$
$\$_1$	$GCCAAC\$_1$
A	$GCTC\$_2$
C	$GCTT\$_3$
T	$T\$_3$
C	$TC\$_2$
$\$_3$	$TCGCTT\$_3$
C	$TT\$_3$

$eBWT$	Sorted Suffixes of S
C	$\$_1$
T	$\$_3$
C	$AAC\$_1$
A	$AC\$_1$
A	$C\$_1$
C	$CAAC\$_1$
G	$CCAAC\$_1$
T	$CGCTT\$_3$
G	$CTT\$_3$
$\$_1$	$GCCAAC\$_1$
C	$GCTT\$_3$
T	$T\$_3$
$\$_3$	$TCGCTT\$_3$
C	$TT\$_3$

Projections on two strings

One can compute the eBWT of the whole collection and analyze all pairs at the same time or one can get a projection of the two selected strings.

Collection S

	0	1	2	3	4	5	6
S_1	G	C	C	A	A	C	$\$_1$
S_2	G	A	G	C	T	C	$\$_2$
S_3	T	C	G	C	T	T	$\$_3$

In order to remove S_2, we only need to remove the blue symbols (associated with the suffixes of S_2) in $eBWT$.

Actually, we don't need to store the colors, for instance the symbols of S_2 can be found by using the LF-mapping starting from $\$_2$.

$eBWT$	Sorted Suffixes of S
C	$\$_1$
C	$\$_2$
T	$\$_3$
C	$AAC\$_1$
A	$AC\$_1$
G	$AGCTC\$_2$
A	$C\$_1$
T	$C\$_2$
C	$CAAC\$_1$
G	$CCAAC\$_1$
T	$CGCTT\$_3$
G	$CTC\$_2$
G	$CTT\$_3$
$\$_2$	$GAGCTC\$_2$
$\$_1$	$GCCAAC\$_1$
A	$GCTC\$_2$
C	$GCTT\$_3$
T	$T\$_3$
C	$TC\$_2$
$\$_3$	$TCGCTT\$_3$
C	$TT\$_3$

$eBWT$	Sorted Suffixes of S
C	$\$_1$
T	$\$_3$
C	$AAC\$_1$
A	$AC\$_1$
A	$C\$_1$
C	$CAAC\$_1$
G	$CCAAC\$_1$
T	$CGCTT\$_3$
G	$CTT\$_3$
$\$_1$	$GCCAAC\$_1$
C	$GCTT\$_3$
T	$T\$_3$
$\$_3$	$TCGCTT\$_3$
C	$TT\$_3$

Sequences comparison

Similarity measures based on eBWT can be obtained by using the following property.

Key

Since conjugates/suffixes starting with the same context are close in the sorted list:
The greater is the number of segments shared by u and v, the greater is the mixing of the suffixes of u and v in the sorted list and the greater is the clustering effect in the eBWT.

Sequences comparison

Similarity measures based on eBWT can be obtained by using the following property.

Key

Since conjugates/suffixes starting with the same context are close in the sorted list:
The greater is the number of segments shared by u and v, the greater is the mixing of the suffixes of u and v in the sorted list and the greater is the clustering effect in the eBWT.

First goal

First goal

Similarity measures for comparing DNA genomes

Some measures:

- *Mantaci, Restivo, R. and Sciortino*, 2007
- *Mantaci, Restivo, R. and Sciortino*, 2008
- *Yang, Zhang, and Wang*, 2010

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

DA	$eBWT$	Sorted suffixes
1	C	$\$_1$
2	G	$\$_2$
1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1	C	$A\ C\ G\ G\ C\ \$_1$
2	C	$A\ C\ G\ G\ G\ \$_2$
1	G	$C\ \$_1$
1	A	$C\ A\ C\ G\ G\ C\ \$_1$
2	A	$C\ A\ C\ G\ G\ G\ \$_2$
1	A	$C\ G\ G\ C\ \$_1$
2	A	$C\ G\ G\ G\ \$_2$
2	G	$G\ \$_2$
1	G	$G\ C\ \$_1$
2	G	$G\ G\ \$_2$
1	C	$G\ G\ C\ \$_1$
2	C	$G\ G\ G\ \$_2$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
	2	G	$\$_2$
	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$S = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
0	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
0	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
0	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
0	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
0	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
0	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
0	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
0	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
0	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
0	1	A	$C\ G\ G\ C\ \$_1$
1	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
0	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
0	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
0	1	A	$C\ G\ G\ C\ \$_1$
1	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
0	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
0	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
0	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
0	1	A	$C\ G\ G\ C\ \$_1$
1	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
0	1	G	$G\ C\ \$_1$
0	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
0	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
0	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
0	1	A	$C\ G\ G\ C\ \$_1$
1	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
0	1	G	$G\ C\ \$_1$
0	2	G	$G\ G\ \$_2$
0	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
0	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
0	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
0	1	A	$C\ G\ G\ C\ \$_1$
1	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
0	1	G	$G\ C\ \$_1$
0	2	G	$G\ G\ \$_2$
0	1	C	$G\ G\ C\ \$_1$
0	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

First (historically) similarity measure: based on mixing of colors [*Mantaci, Restivo, R. and Sciortino*, 2007]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Takes into account the alternation (mixing) of the colors, i.e. symbols coming from different sequences in the output of eBWT

$D_{col}(u,v)$	DA	$eBWT$	Sorted suffixes
0	1	C	$\$_1$
0	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
0	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
0	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
0	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
0	1	A	$C\ G\ G\ C\ \$_1$
1	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
0	1	G	$G\ C\ \$_1$
0	2	G	$G\ G\ \$_2$
0	1	C	$G\ G\ C\ \$_1$
0	2	C	$G\ G\ G\ \$_2$

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

In the example:

$$D_{col}(u,v) = 2$$

that we can normalize with the lengths of the sequences, so that

$$D_{col}(u,v) = 2/(8+8)$$

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

DA	$eBWT$	Sorted suffixes
1	C	$\$_1$
2	G	$\$_2$
1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1	C	$A\ C\ G\ G\ C\ \$_1$
2	C	$A\ C\ G\ G\ G\ \$_2$
1	G	$C\ \$_1$
1	A	$C\ A\ C\ G\ G\ C\ \$_1$
2	A	$C\ A\ C\ G\ G\ G\ \$_2$
1	A	$C\ G\ G\ C\ \$_1$
2	A	$C\ G\ G\ G\ \$_2$
2	G	$G\ \$_2$
1	G	$G\ C\ \$_1$
2	G	$G\ G\ \$_2$
1	C	$G\ G\ C\ \$_1$
2	C	$G\ G\ G\ \$_2$

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$S = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
	2	G	$\$_2$
	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form $1^{k_1}2^{k_1}1^{k_3}2^{k_4}\dots 2^{k_m}1^{k_{m+1}}$ where i^{k_j} means i repeats k_j times. Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$S = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1} 2^{k_1} 1^{k_3} 2^{k_4} \dots 2^{k_m} 1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$S = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1}2^{k_1}1^{k_3}2^{k_4}\dots 2^{k_m}1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1}2^{k_1}1^{k_3}2^{k_4}\dots 2^{k_m}1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form $1^{k_1}2^{k_1}1^{k_3}2^{k_4}\dots 2^{k_m}1^{k_{m+1}}$ where i^{k_j} means i repeats k_j times. Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	C	$A\ C\ G\ G\ C\ \$_1$
2^1	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1}2^{k_1}1^{k_3}2^{k_4}\dots 2^{k_m}1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	C	$A\ C\ G\ G\ C\ \$_1$
2^1	2	C	$A\ C\ G\ G\ G\ \$_2$
1^2	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1}2^{k_1}1^{k_3}2^{k_4}\ldots 2^{k_m}1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$,
then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	C	$A\ C\ G\ G\ C\ \$_1$
2^1	2	C	$A\ C\ G\ G\ G\ \$_2$
1^2	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1}2^{k_1}1^{k_3}2^{k_4}\dots 2^{k_m}1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$S = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	C	$A\ C\ G\ G\ C\ \$_1$
2^1	2	C	$A\ C\ G\ G\ G\ \$_2$
1^2	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1} 2^{k_1} 1^{k_3} 2^{k_4} \ldots 2^{k_m} 1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	C	$A\ C\ G\ G\ C\ \$_1$
2^1	2	C	$A\ C\ G\ G\ G\ \$_2$
1^2	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	A	$C\ G\ G\ C\ \$_1$
2^2	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1}2^{k_1}1^{k_3}2^{k_4}\dots 2^{k_m}1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	C	$A\ C\ G\ G\ C\ \$_1$
2^1	2	C	$A\ C\ G\ G\ G\ \$_2$
1^2	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	A	$C\ G\ G\ C\ \$_1$
2^2	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
1^1	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1}2^{k_1}1^{k_3}2^{k_4}\dots 2^{k_m}1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	C	$A\ C\ G\ G\ C\ \$_1$
2^1	2	C	$A\ C\ G\ G\ G\ \$_2$
1^2	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	A	$C\ G\ G\ C\ \$_1$
2^2	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
1^1	1	G	$G\ C\ \$_1$
2^1	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1}2^{k_1}1^{k_3}2^{k_4}\dots 2^{k_m}1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$S = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	C	$A\ C\ G\ G\ C\ \$_1$
2^1	2	C	$A\ C\ G\ G\ G\ \$_2$
1^2	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	A	$C\ G\ G\ C\ \$_1$
2^2	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
1^1	1	G	$G\ C\ \$_1$
2^1	2	G	$G\ G\ \$_2$
1^1	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1}2^{k_1}1^{k_3}2^{k_4}\ldots 2^{k_m}1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$, then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$S = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

Compute the expectation or the entropy of the distribution of the alternations of colors

$r(u,v)$	DA	$eBWT$	Sorted suffixes
1^1	1	C	$\$_1$
2^1	2	G	$\$_2$
1^1	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	C	$A\ C\ G\ G\ C\ \$_1$
2^1	2	C	$A\ C\ G\ G\ G\ \$_2$
1^2	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
2^1	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
1^1	1	A	$C\ G\ G\ C\ \$_1$
2^2	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
1^1	1	G	$G\ C\ \$_1$
2^1	2	G	$G\ G\ \$_2$
1^1	1	C	$G\ G\ C\ \$_1$
2^1	2	C	$G\ G\ G\ \$_2$

Rewrite in the form
$1^{k_1}2^{k_1}1^{k_3}2^{k_4}\dots2^{k_m}1^{k_{m+1}}$
where i^{k_j} means i repeats k_j times.
Note that if $j \neq 1$ and $j \neq m+1$,
then $k_j > 0$.

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$r = 1^{k_1} 2^{k_1} 1^{k_3} 2^{k_4} \dots 2^{k_m} 1^{k_{m+1}}$$

where i^{k_j} means i repeats k_j times.
If t_{k_j} is the number of k_j

$$s = t_1 + t_2 + \dots + t_{k_j} + \dots + t_{\max(|u|,|v|)}$$

In the example:

$$r = 1^1 2^1 1^1 2^1 1^1 2^1 1^2 2^1 1^1 2^2 1^1 2^1 1^1 2^1$$

$t_1 = 12$ (12 times 1^1 or 2^1) and
$t_2 = 2$ (2 times 2^2 or 2^2).

$$s = 12 + 2 = 14$$

The Burrow-Wheeler similarity distribution (BWSD) of u and v is

$$P\{k_j = k\} = \frac{t_k}{s} \text{ for } k = 1, 2, 3, \dots$$

So

$$P\{k_j = 1\} = \frac{11}{14} \text{ and } P\{k_j = 2\} = \frac{2}{14}$$

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$r = 1^{k_1} 2^{k_1} 1^{k_3} 2^{k_4} \ldots 2^{k_m} 1^{k_{m+1}}$$

where i^{k_j} means i repeats k_j times.
If t_{k_j} is the number of k_j

$$s = t_1 + t_2 + \cdots + t_{k_j} + \cdots + t_{\max(|u|,|v|)}$$

In the example:

$$r = 1^1 2^1 1^1 2^1 1^1 2^1 1^2 2^1 1^1 2^2 1^1 2^1 1^1 2^1$$

$t_1 = 12$ (12 times 1^1 or 2^1) and
$t_2 = 2$ (2 times 2^2 or 2^2).

$$s = 12 + 2 = 14$$

The Burrow-Wheeler similarity distribution (BWSD) of u and v is

$$P\{k_j = k\} = \frac{t_k}{s} \text{ for } k = 1, 2, 3, \ldots$$

So

$$P\{k_j = 1\} = \frac{11}{14} \text{ and } P\{k_j = 2\} = \frac{2}{14}$$

A variant: based on distribution of the colors [Yang, Chang and Zhang, 2010]

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

In the example:

$$r(u,v) = 1^1 2^1 1^1 2^1 1^1 2^1 1^2 2^1 1^1 2^2 1^1 2^1 1^1 2^1$$

$t_1 = 12$ (12 times 1^1 or 2^1) and $t_2 = 2$ (2 times 2^2 or 2^2).
The Burrow-Wheeler similarity distribution (BWSD) of u and v is

$$P(k_j = 1) = \frac{12}{14} \text{ and } P(k_j = 2) = \frac{2}{14}$$

Yang et al. defined the following measures between u and v:

- $D_M(u,v) = E(k_j) - 1$, where $E(k_j)$ is the expectation of $BWSD(u,v)$.
- $D_E(u,v) = -\sum_{k \geq 1, t_k \neq 0} (t_k/s) \log_2(t_k/s)$ is the Shannon entropy of $BWSD(u,v)$.

Properties of D_{col} and BWSD (D_M and D_E)

D_{col} and BWSD (D_M and D_E) are symmetric.
The similarity between u and v is equal to the similarity between v and u.

Differences between two eBWT transformations

Now, we consider two conjugates words $u = GAGCTC(\$_1)$ and $v = GCTCGA(\$_2)$

D_{col}	$eBWT$	Conjugates sorted			D_{col}	$eBWT$	Suffixes sorted
					0	C	$\$_1$
					1	A	$\$_2$
0	G	$A\ G\ C\ T\ C$	G			G	$A\ \$_2$
0	G	$A\ G\ C\ T\ C$	G		1	G	$A\ G\ C\ T\ C\ \$_1$
0	T	$C\ G\ A\ G\ C$	T			T	$C\ \$_1$
0	T	$C\ G\ A\ G\ C$	T		0	T	$C\ G\ A\ \$_2$
0	G	$C\ T\ C\ G\ A$	G	$\neq$	0	G	$C\ T\ C\ \$_1$
0	G	$C\ T\ C\ G\ A$	G		1	G	$C\ T\ C\ G\ A\ \$_2$
0	C	$G\ A\ G\ C\ T$	C			C	$G\ A\ \$_2$
0	C	$G\ A\ G\ C\ T$	C		1	$\$_1$	$G\ A\ G\ C\ T\ C\ \$_1$
0	A	$G\ C\ T\ C\ G$	A			A	$G\ C\ T\ C\ \$_1$
0	A	$G\ C\ T\ C\ G$	A		0	$\$_2$	$G\ C\ T\ C\ G\ A\ \$_2$
0	C	$T\ C\ G\ A\ G$	C		0	C	$T\ C\ \$_1$
0	C	$T\ C\ G\ A\ G$	C		0	C	$T\ C\ G\ A\ \$_2$

If the eBWT obtained by sorting the conjugates is used then

- If u is a conjugate of v, then $D_{col}(u,v) = 0$, $D_M(u,v) = 0$ and $D_E(u,v) = 0$
- If u' is a conjugate of u and v' is a conjugate of v, then $D_{col}(u,v) = D_{col}(u',v')$, $D_M(u,v) = D_M(u',v')$ and $D_E(u,v) = D_E(u',v')$.

They are similarity measures for conjugacy classes.

Properties of D_{col} and BWSD (D_M and D_E)

It is not always true that if $D_{col}(u,v)=0$, $D_M(u,v)=0$ and $D_E(u,v)=0$ then $u=v$ or they are conjugates.

Example

Let $u = aabc$ and $v = abbc$. Although the two sequences are not conjugates, $D_{col}(u,v)=0$

$D_{col}(u,v)$	DA	$eBWT$	conjugates sorted			
0	0	c	a	a	b	c
0	1	c	a	b	b	c
0	0	a	a	b	c	a
0	1	a	b	b	c	a
0	0	a	b	c	a	a
0	1	b	b	c	a	b
0	0	b	c	a	a	b
0	1	b	c	a	b	b

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

$$D_{col}(u,v) = 0$$

Idea: use the symbols in the eBWT!

Properties of D_{col} and BWSD (D_M and D_E)

It is not always true that if $D_{col}(u,v) = 0$, $D_M(u,v) = 0$ and $D_E(u,v) = 0$ then $u = v$ or they are conjugates.

Example

Let $u = aabc$ and $v = abbc$. Although the two sequences are not conjugates, $D_{col}(u,v) = 0$

$D_{col}(u,v)$	DA	$eBWT$	conjugates sorted			
0	0	c	a	a	b	c
0	1	c	a	b	b	c
0	0	a	a	b	c	a
0	1	a	b	b	c	a
0	0	a	b	c	a	a
0	1	b	b	c	a	b
0	0	b	c	a	a	b
0	1	b	c	a	b	b

$$D_{col}(u,v) = \sum_{\substack{i=1,\\ n_i \neq 0}}^{k} (n_i - 1)$$

$$D_{col}(u,v) = 0$$

Idea: use the symbols in the eBWT!

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$S = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
	1	C	$\$_1$
	2	G	$\$_2$
	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
	1	C	$\$_1$
	2	G	$\$_2$
	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
1	1	C	$\$_1$
	2	G	$\$_2$
	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

We sum the differences between the number of symbols coming from u and from v in each block.

$$D_{BW}(u,v) = \sum_{i=1}^{k} |c_i(u) - c_i(v)|.$$

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
1	1	C	$\$_1$
1	2	G	$\$_2$
	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

We sum the differences between the number of symbols coming from u and from v in each block.

$$D_{BW}(u,v) = \sum_{i=1}^{k} |c_i(u) - c_i(v)|.$$

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
1	1	C	$\$_1$
1	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

We sum the differences between the number of symbols coming from u and from v in each block.

$$D_{BW}(u,v) = \sum_{i=1}^{k} |c_i(u) - c_i(v)|.$$

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$S = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
1	1	C	$\$_1$
1	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

We sum the differences between the number of symbols coming from u and from v in each block.

$$D_{BW}(u,v) = \sum_{i=1}^{k} |c_i(u) - c_i(v)|.$$

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
1	1	C	$\$_1$
1	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

We sum the differences between the number of symbols coming from u and from v in each block.

$$D_{BW}(u,v) = \sum_{i=1}^{k} |c_i(u) - c_i(v)|.$$

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
1	1	C	$\$_1$
1	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
0	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

We sum the differences between the number of symbols coming from u and from v in each block.

$$D_{BW}(u,v) = \sum_{i=1}^{k} |c_i(u) - c_i(v)|.$$

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
1	1	C	$\$_1$
1	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
0	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
1	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

We sum the differences between the number of symbols coming from u and from v in each block.

$$D_{BW}(u,v) = \sum_{i=1}^{k} |c_i(u) - c_i(v)|.$$

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
1	1	C	$\$_1$
1	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
0	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
1	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
0	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

We sum the differences between the number of symbols coming from u and from v in each block.

$$D_{BW}(u,v) = \sum_{i=1}^{k} |c_i(u) - c_i(v)|.$$

Similarity measure: based on clustering [Mantaci, Restivo, R. and Sciortino, 2008]

Based on differences of the frequencies of the colors in the blocks of the same symbol!

$$\mathsf{S} = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

$D_{BW}(u,v)$	DA	$eBWT$	Sorted suffixes
1	1	C	$\$_1$
1	2	G	$\$_2$
0	1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
	2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
0	1	C	$A\ C\ G\ G\ C\ \$_1$
	2	C	$A\ C\ G\ G\ G\ \$_2$
1	1	G	$C\ \$_1$
0	1	A	$C\ A\ C\ G\ G\ C\ \$_1$
	2	A	$C\ A\ C\ G\ G\ G\ \$_2$
	1	A	$C\ G\ G\ C\ \$_1$
	2	A	$C\ G\ G\ G\ \$_2$
1	2	G	$G\ \$_2$
	1	G	$G\ C\ \$_1$
	2	G	$G\ G\ \$_2$
0	1	C	$G\ G\ C\ \$_1$
	2	C	$G\ G\ G\ \$_2$

We sum the differences between the number of symbols coming from u and from v in each block.

$$D_{BW}(u,v) = \sum_{i=1}^{k} |c_i(u) - c_i(v)|.$$

In the example:

$$D_{BW}(u,v) = \sum_{i=1}^{8} |c_i(u) - c_i(v)| = 4$$

that we can normalize with the lengths of the sequences, so that

$$D_{BW}(u,v) = 4/(8+8)$$

Properties

- $D_{BW}(u,v) = D_{BW}(v,u)$, i.e. the measure D_{BW} is symmetric.
- $D_{BW}(u,v) = 0$ if and only if $u = v$.

Moreover, if the eBWT obtained by sorting the conjugates is used then

- If u is a conjugate of v, then $D_{BW}(u,v) = 0$
- If u' is a conjugate of u and v' is a conjugate of v, then $D_{BW}(u,v) = D_{BW}(u',v')$.

Therefore, D_{BW} is a distance measure for conjugacy classes.

Observation: different measures

$$S = \{u = ACACGGC\$_1, v = ACACGGG\$_2\}$$

DA	$eBWT$	Sorted suffixes
1	C	$\$_1$
2	G	$\$_2$
1	$\$_1$	$A\ C\ A\ C\ G\ G\ C\ \$_1$
2	$\$_2$	$A\ C\ A\ C\ G\ G\ G\ \$_2$
1	C	$A\ C\ G\ G\ C\ \$_1$
2	C	$A\ C\ G\ G\ G\ \$_2$
1	G	$C\ \$_1$
1	A	$C\ A\ C\ G\ G\ C\ \$_1$
2	A	$C\ A\ C\ G\ G\ G\ \$_2$
1	A	$C\ G\ G\ C\ \$_1$
2	A	$C\ G\ G\ G\ \$_2$
2	G	$G\ \$_2$
1	G	$G\ C\ \$_1$
2	G	$G\ G\ \$_2$
1	C	$G\ G\ C\ \$_1$
2	C	$G\ G\ G\ \$_2$

By changing the partition, one can obtain different similarity measures.

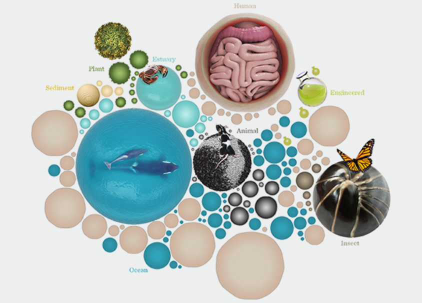
Biological applications

Biological applications based on these eBWT similarity measures:

- for building the phylogenetic tree of mitochondrial dna: [Mantaci et al. 2007, 2008], [Yang, Chang and Zhang, 2010]
- Protein comparison: *Yang, Chang, Zhang and Wang*, 2010: based on measure defined in [Yang, Chang and Zhang, 2010]
- Expressed sequence tags: *Ng, Phon-Amnuaisuk, Ho*: based on measure defined in [Mantaci et al. 2007], a window-based similarity comparison is used.

Second goal: Metagenomics

Metagenomics is the study of genetic material collected from the environment



[Illustration: Spencer Phillips, EMBL-EBI]

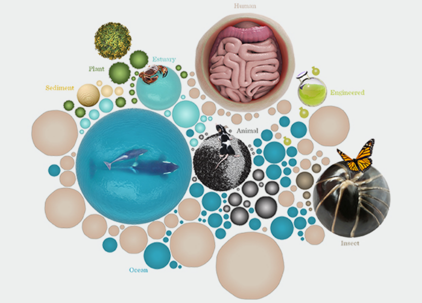
Aim to explore the relations between the microbes and their habitats

Applications. Clinical microbiology, plant-microbe interactions, monitoring pollution, sustainability, ecology, ...

Goal: Identify the taxon of each short read

Second goal: Metagenomics

Metagenomics is the study of genetic material collected from the environment



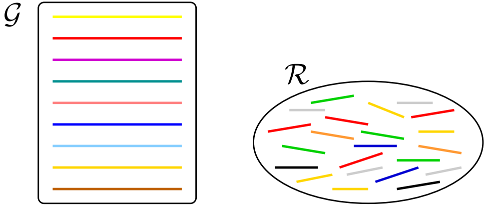
[Illustration: Spencer Phillips, EMBL-EBI]

Aim to explore the relations between the microbes and their habitats

Applications. Clinical microbiology, plant-microbe interactions, monitoring pollution, sustainability, ecology, ...

Goal: Identify the taxon of each short read

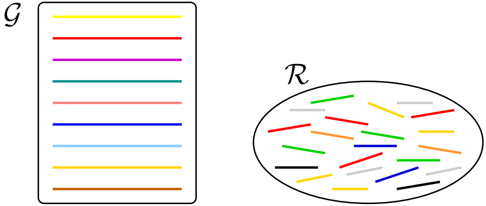
Metagenomic Classification problem



- $\mathcal{R} = \{r_1, \ldots, r_{|R|}\}$ metagenome (collection of short reads)
- $\mathcal{G} = \{g_1, \ldots, g_{|G|}\}$ reference genomes (collection of long sequences)
- $\mathcal{S} = \mathcal{R} \cup \mathcal{G}$ multi-set of biological sequences

Goal: to assign each read r_i in $\mathcal{R}$ to a unique genome g_j in $\mathcal{G}$ by reading $eBWT(\mathcal{S})$, $DA(\mathcal{S})$, $LCP(\mathcal{S})$.

Metagenomic Classification problem



- $\mathcal{R} = \{r_1, \ldots, r_{|R|}\}$ metagenome (collection of short reads)
- $\mathcal{G} = \{g_1, \ldots, g_{|G|}\}$ reference genomes (collection of long sequences)
- $\mathcal{S} = \mathcal{R} \cup \mathcal{G}$ multi-set of biological sequences

Goal: to assign each read r_i in $\mathcal{R}$ to a unique genome g_j in $\mathcal{G}$ by reading $eBWT(\mathcal{S})$, $DA(\mathcal{S})$, $LCP(\mathcal{S})$.

Step 1: Build α-clusters and Similarity Arrays (Part I)

Minimum LCP value $\alpha = 3$

$r_i = NGGCGTACCA\$_i$

$g_j = TTATTTTGGCGGGCGTATGTATTAGTTT\$_j$

i	LCP	$eBWT$	Sorted suffixes
1	0	A	$\$_i$
2	0	T	$\$_j$
3	0	C	$A\$_i$
4	1	T	$ACCA\$_i$
5	1	T	$AGTTT\$_j$
6	1	T	$ATGTATTAGTTT\$_j$
7	2	T	$ATTAGTTT\$_j$
8	0	C	$CA\$_i$
9	1	A	$CCA\$_i$
10	1	G	$CGGGGCGTA\ldots\$_j$
11	2	G	$CGTACCA\$_i$
12	4	G	$CGTATGAT\ldots\$_j$
13	1	T	$CTTTTGGCG\ldots\$_j$
14	0	G	$GCGGGGCGT\ldots\$_j$
15	3	G	$GCGTACCA\$_i$
16	5	G	$GCGTATGTAA\ldots\$_j$
$\vdots$	$\vdots$	$\vdots$	$\vdots$

An α-*cluster* $\mathcal{C}_\alpha$ of $eBWT(\mathcal{S})$ is any pair of indices (pS, pE) such that

- $LCP[pS] < \alpha$ and $LCP[pE+1] < \alpha$,
- $LCP[k] \geq \alpha$, $pS < k \leq pE$,
- $DA[s] \in \mathcal{R}$ and $DA[t] \in \mathcal{G}$, $pS \leq s, t \leq pE$.

$\mathcal{C}_\alpha(r_i, g_j) = \{$

$$Sim_r[g] = \sum_{x \in \mathcal{C}_\alpha} \sum_{a \in \Sigma} \min(n_r, n_g)$$

n_r=number of indices s in x such that $eBWT[s] = a$ and $DA[s] = r$,
n_g=number of indices t in x such that $eBWT[t] = a$ and $DA[t] = g$.

$Sim_{r_i}[g_j] =$

Step 1: Build α-clusters and Similarity Arrays (Part I)

Minimum LCP value $\alpha = 3$

$r_i = NGGCGTACCA\$_i$

$g_j = TTATTTTGGCGGGCGTATGTATTAGTTT\$_j$

i	LCP	$eBWT$	Sorted suffixes
1	0	A	$\$_i$
2	0	T	$\$_j$
3	0	C	$A\$_i$
4	1	T	$ACCA\$_i$
5	1	T	$AGTTT\$_j$
6	1	T	$ATGTATTAGTTT\$_j$
7	2	T	$ATTAGTTT\$_j$
8	0	C	$CA\$_i$
9	1	A	$CCA\$_i$
10	1	G	$CGGGGCGTA\ldots\$_j$
11	2	G	$CGTACCA\$_i$
12	4	G	$CGTATGAT\ldots\$_j$
13	1	T	$CTTTTGGCG\ldots\$_j$
14	0	G	$GCGGGGCGT\ldots\$_j$
15	3	G	$GCGTACCA\$_i$
16	5	G	$GCGTATGTAA\ldots\$_j$
⋮	⋮	⋮	⋮

An α-*cluster* $\mathcal{C}_\alpha$ of $eBWT(\mathcal{S})$ is any pair of indices (pS, pE) such that

- $LCP[pS] < \alpha$ and $LCP[pE+1] < \alpha$,
- $LCP[k] \geq \alpha$, $pS < k \leq pE$,
- $DA[s] \in \mathcal{R}$ and $DA[t] \in \mathcal{G}$, $pS \leq s, t \leq pE$.

$\mathcal{C}_\alpha(r_i, g_j) = \{(11, 12),$

$$Sim_r[g] = \sum_{x \in \mathcal{C}_\alpha} \sum_{a \in \Sigma} \min(n_r, n_g)$$

n_r=number of indices s in x such that $eBWT[s] = a$ and $DA[s] = r$,
n_g=number of indices t in x such that $eBWT[t] = a$ and $DA[t] = g$.

$Sim_{r_i}[g_j] = 1+$

Step 1: Build α-clusters and Similarity Arrays (Part I)

Minimum LCP value $\alpha = 3$

$r_i = NGGCGTACCA\$_i$

$g_j = TTATTTTGGCGGGCGTATGTATTAGTTT\$_j$

i	LCP	$eBWT$	Sorted suffixes
1	0	A	$\$_i$
2	0	T	$\$_j$
3	0	C	$A\$_i$
4	1	T	$ACCA\$_i$
5	1	T	$AGTTT\$_j$
6	1	T	$ATGTATTAGTTT\$_j$
7	2	T	$ATTAGTTT\$_j$
8	0	C	$CA\$_i$
9	1	A	$CCA\$_i$
10	1	G	$CGGGGCGTA\dots\$_j$
11	2	G	$CGTACCA\$_i$
12	4	G	$CGTATGAT\dots\$_j$
13	1	T	$CTTTTGGCG\dots\$_j$
14	0	G	$GCGGGGCGT\dots\$_j$
15	3	G	$GCGTACCA\$_i$
16	5	G	$GCGTATGTAA\dots\$_j$
$\vdots$	$\vdots$	$\vdots$	$\vdots$

An *α-cluster* $\mathcal{C}_\alpha$ of $eBWT(\mathcal{S})$ is any pair of indices (pS, pE) such that

- $LCP[pS] < \alpha$ and $LCP[pE+1] < \alpha$,
- $LCP[k] \geq \alpha$, $pS < k \leq pE$,
- $DA[s] \in \mathcal{R}$ and $DA[t] \in \mathcal{G}$, $pS \leq s, t \leq pE$.

$\mathcal{C}_\alpha(r_i, g_j) = \{(11, 12), (14, 16), \dots$

$$Sim_r[g] = \sum_{x \in \mathcal{C}_\alpha} \sum_{a \in \Sigma} \min(n_r, n_g)$$

n_r=number of indices s in x such that $eBWT[s] = a$ and $DA[s] = r$,
n_g=number of indices t in x such that $eBWT[t] = a$ and $DA[t] = g$.

$Sim_{r_i}[g_j] = 1+1+$

Step 1: Build α-clusters and Similarity Arrays (Part I)

Minimum LCP value $\alpha = 3$

r_i=$N\underline{GGCGTA}CCA\$_i$

g_j=$TTATTTTGCGG\underline{GGCGTA}TGTATTAGTTT\$_j$

i	LCP	$eBWT$	Sorted suffixes
1	0	A	$\$_i$
2	0	T	$\$_j$
3	0	C	$A\$_i$
4	1	T	$ACCA\$_i$
5	1	T	$AGTTT\$_j$
6	1	T	$ATGTATTAGTTT\$_j$
7	2	T	$ATTAGTTT\$_j$
8	0	C	$CA\$_i$
9	1	A	$CCA\$_i$
10	1	G	$CGGGGCGTA\ldots\$_j$
11	2	G	$CGTACCA\$_i$
12	4	G	$CGTATGAT\ldots\$_j$
13	1	T	$CTTTTGGCG\ldots\$_j$
14	0	G	$GCGGGGCGT\ldots\$_j$
15	3	G	$GCGTACCA\$_i$
16	5	G	$GCGTATGTAA\ldots\$_j$
$\vdots$	$\vdots$	$\vdots$	$\vdots$

An *α-cluster* $\mathcal{C}_\alpha$ of $eBWT(\mathcal{S})$ is any pair of indices (pS, pE) such that

- $LCP[pS] < \alpha$ and $LCP[pE+1] < \alpha$,
- $LCP[k] \geq \alpha$, $pS < k \leq pE$,
- $DA[s] \in \mathcal{R}$ and $DA[t] \in \mathcal{G}$, $pS \leq s, t \leq pE$.

$\mathcal{C}_\alpha(r_i, g_j) = \{(11,12),(14,16),\ldots$

$$Sim_r[g] = \sum_{x \in \mathcal{C}_\alpha} \sum_{a \in \Sigma} \min(n_r, n_g)$$

n_r=number of indices s in x such that $eBWT[s] = a$ and $DA[s] = r$,
n_g=number of indices t in x such that $eBWT[t] = a$ and $DA[t] = g$.

$Sim_{r_i}[g_j] = 1+1+\ldots$

Step 1: Build α-clusters and Similarity Arrays (Part II)

Minimum LCP value $\alpha = 3$

$r_i = NGGCGTACCA\$_i$

$g_j = TTATTTTGGCGGGGCGTATGTATTAGTTT\$_j$

i	LCP	$eBWT$	Sorted suffixes
⋮	⋮	⋮	⋮
17	1	T	$GGCGGGGCG\ldots\$_j$
18	4	N	$GGCGTACCA\$_i$
19	6	G	$GGCGTATGTAT\ldots\$_j$
20	2	G	$GGGCGTAT\ldots\$_j$
21	3	C	$GGGGCGTAT\ldots\$_j$
22	1	C	$GTACCA\$_i$
23	3	C	$GTATGTA\ldots\$_j$
24	4	C	$GTATTA\ldots\$_j$
25	2	A	$GTTT\$_j$
26	0	$\$_i$	$NGGCGTACCA\$_i$
27	0	T	$T\$_j$
28	1	G	$TACCA\$_i$
29	2	T	$TAGTTT\$_j$
⋮	⋮	⋮	⋮

An α*-cluster* $\mathcal{C}_\alpha$ of $eBWT(\mathcal{S})$ is any pair of indices (pS, pE) such that

- $LCP[pS] < \alpha$ and $LCP[pE+1] < \alpha$,
- $LCP[k] \geq \alpha$, $pS < k \leq pE$,
- $DA[i] \in \mathcal{R}$ and $DA[j] \in \mathcal{G}$, $pS \leq i, j \leq pE$.

$\mathcal{C}_\alpha = \{(11, 12), (14, 16),$

$$Sim_r[g] = \sum_{x \in \mathcal{C}_\alpha} \sum_{a \in \Sigma} \min(n_r, n_g)$$

n_r=number of indices i in x such that $eBWT[i] = a$ and $DA[i] = r$,
n_g=number of indices i' in x such that $eBWT[i'] = a$ and $DA[i'] = g$.

$Sim_r[g] = 1+1+$

Step 1: Build α-clusters and Similarity Arrays (Part II)

Minimum LCP value $\alpha = 3$

$r_i = NGGCGTACCA\$_i$

$g_j = TTATTTTGGCGGGGCGTATGTATTAGTTT\$_j$

i	LCP	$eBWT$	Sorted suffixes
⋮	⋮	⋮	⋮
17	1	T	$GGCGGGGCG\ldots\$_j$
18	4	N	$GGCGTACCA\$_i$
19	6	G	$GGCGTATGTAT\ldots\$_j$
20	2	G	$GGGCGTAT\ldots\$_j$
21	3	C	$GGGGCGTAT\ldots\$_j$
22	1	C	$GTACCA\$_i$
23	3	C	$GTATGTA\ldots\$_j$
24	4	C	$GTATTA\ldots\$_j$
25	2	A	$GTTT\$_j$
26	0	$\$_i$	$NGGCGTACCA\$_i$
27	0	T	$T\$_j$
28	1	G	$TACCA\$_i$
29	2	T	$TAGTTT\$_j$
⋮	⋮	⋮	⋮

An α-*cluster* $\mathcal{C}_\alpha$ of $eBWT(\mathcal{S})$ is any pair of indices (pS, pE) such that

- $LCP[pS] < \alpha$ and $LCP[pE+1] < \alpha$,
- $LCP[k] \geq \alpha$, $pS < k \leq pE$,
- $DA[i] \in \mathcal{R}$ and $DA[j] \in \mathcal{G}$, $pS \leq i, j \leq pE$.

$\mathcal{C}_\alpha = \{(11,12),(14,16),(17,19),$

$$Sim_r[g] = \sum_{x \in \mathcal{C}_\alpha} \sum_{a \in \Sigma} \min(n_r, n_g)$$

n_r=number of indices i in x such that $eBWT[i] = a$ and $DA[i] = r$,
n_g=number of indices i' in x such that $eBWT[i'] = a$ and $DA[i'] = g$.

$Sim_r[g] = 1+1+1+$

Step 1: Build α-clusters and Similarity Arrays (Part II)

Minimum LCP value $\alpha = 3$

r_i=$NGGCGTACCA\$_i$

g_j=$TTATTTTGGCGGGGCGTATGTATTAGTTT\$_j$

i	LCP	$eBWT$	Sorted suffixes
⋮	⋮	⋮	⋮
17	1	T	$GGCGGGGCG\ldots\$_j$
18	4	N	$GGCGTACCA\$_i$
19	6	G	$GGCGTATGTAT\ldots\$_j$
20	2	G	$GGGCGTAT\ldots\$_j$
21	3	C	$GGGGCGTAT\ldots\$_j$
22	1	C	$GTACCA\$_i$
23	3	C	$GTATGTA\ldots\$_j$
24	4	C	$GTATTA\ldots\$_j$
25	2	A	$GTTT\$_j$
26	0	$\$_i$	$NGGCGTACCA\$_i$
27	0	T	$T\$_j$
28	1	G	$TACCA\$_i$
29	2	T	$TAGTTT\$_j$
⋮	⋮	⋮	⋮

An *α-cluster* $\mathcal{C}_\alpha$ of $eBWT(\mathcal{S})$ is any pair of indices (pS, pE) such that

- $LCP[pS] < \alpha$ and $LCP[pE+1] < \alpha$,
- $LCP[k] \geq \alpha$, $pS < k \leq pE$,
- $DA[i] \in \mathcal{R}$ and $DA[j] \in \mathcal{G}$, $pS \leq i, j \leq pE$.

$\mathcal{C}_\alpha = \{(11,12),(14,16),(17,19),(22,24)\}$

$$Sim_r[g] = \sum_{x\in\mathcal{C}_\alpha} \sum_{a\in\Sigma} \min(n_r, n_g)$$

n_r=number of indices i in x such that $eBWT[i] = a$ and $DA[i] = r$,
n_g=number of indices i' in x such that $eBWT[i'] = a$ and $DA[i'] = g$.

$Sim_r[g] = 1+1+1+1 = 4$

Step 2: Classification

The read r_i is

- assigned to g_j if g_j is the only genome such that $Sim_{r_i}[g_j] \sim \max_g Sim_{r_i}[g]$ and $Sim_{r_i}[g_j] > \beta$.
- not classified if $\max_g Sim_{r_i}[g] \leq \beta$.
- ambiguous if $\max_g Sim_{r_i}[g] > \beta$, but there exist at least two genomes g_p and g_q s.t. $Sim_{r_i}[g_p] \sim Sim_{r_i}[g_q] \sim \max_g Sim_{r_i}[g]$

Example

Let $\alpha = 3$ and $\beta = 0.4$.
Suppose the α-similarity between r_i and g_1, g_2, g_3, g_4, g_5 is
$Sim_{r_i}[g_1] = 0.5$, $Sim_{r_i}[g_2] = 0$,
$Sim_{r_i}[g_3] =$, $Sim_{r_i}[g_4] = 0.2$, $Sim_{r_i}[g_5] = 0$.

Step 2: Classification

The read r_i is

- assigned to g_j if g_j is the only genome such that $Sim_{r_i}[g_j] \sim \max_g Sim_{r_i}[g]$ and $Sim_{r_i}[g_j] > \beta$.
- not classified if $\max_g Sim_{r_i}[g] \leq \beta$.
- ambiguous if $\max_g Sim_{r_i}[g] > \beta$, but there exist at least two genomes g_p and g_q s.t. $Sim_{r_i}[g_p] \sim Sim_{r_i}[g_q] \sim \max_g Sim_{r_i}[g]$

Example

Let $\alpha = 3$ and $\beta = 0.4$.
Suppose the α-similarity between r_i and g_1, g_2, g_3, g_4, g_5 is
$Sim_{r_i}[g_1] = 0.5$, $Sim_{r_i}[g_2] = 0$,
$Sim_{r_i}[g_3] = 0.8$, $Sim_{r_i}[g_4] = 0.2$, $Sim_{r_i}[g_5] = 0$.

Step 2: Classification

The read r_i is

- assigned to g_j if g_j is the only genome such that $Sim_{r_i}[g_j] \sim \max_g Sim_{r_i}[g]$ and $Sim_{r_i}[g_j] > \beta$.
- not classified if $\max_g Sim_{r_i}[g] \leq \beta$.
- ambiguous if $\max_g Sim_{r_i}[g] > \beta$, but there exist at least two genomes g_p and g_q s.t. $Sim_{r_i}[g_p] \sim Sim_{r_i}[g_q] \sim \max_g Sim_{r_i}[g]$

Example

Let $\alpha = 3$ and $\beta = 0.4$.
Suppose the α-similarity between r_i and g_1, g_2, g_3, g_4, g_5 is
$Sim_{r_i}[g_1] = 0.5$, $Sim_{r_i}[g_2] = 0$,
$Sim_{r_i}[g_3] = 0.8$, $Sim_{r_i}[g_4] = 0.2$, $Sim_{r_i}[g_5] = 0$.
$\Rightarrow r_i$ is assigned to g_3.

Step 2: Classification

The read r_i is

- assigned to g_j if g_j is the only genome such that $Sim_{r_i}[g_j] \sim \max_g Sim_{r_i}[g]$ and $Sim_{r_i}[g_j] > \beta$.
- not classified if $\max_g Sim_{r_i}[g] \leq \beta$.
- ambiguous if $\max_g Sim_{r_i}[g] > \beta$, but there exist at least two genomes g_p and g_q s.t. $Sim_{r_i}[g_p] \sim Sim_{r_i}[g_q] \sim \max_g Sim_{r_i}[g]$

Example

Let $\alpha = 3$ and $\beta = 0.4$.
Suppose the α-similarity between r_i and g_1, g_2, g_3, g_4, g_5 is
$Sim_{r_i}[g_1] = 0.4$, $Sim_{r_i}[g_2] = 0$,
$Sim_{r_i}[g_3] = 0.34$, $Sim_{r_i}[g_4] = 0.2$, $Sim_{r_i}[g_5] = 0$.

Step 2: Classification

The read r_i is

- assigned to g_j if g_j is the only genome such that $Sim_{r_i}[g_j] \sim \max_g Sim_{r_i}[g]$ and $Sim_{r_i}[g_j] > \beta$.
- not classified if $\max_g Sim_{r_i}[g] \leq \beta$.
- ambiguous if $\max_g Sim_{r_i}[g] > \beta$, but there exist at least two genomes g_p and g_q s.t. $Sim_{r_i}[g_p] \sim Sim_{r_i}[g_q] \sim \max_g Sim_{r_i}[g]$

Example

Let $\alpha = 3$ and $\beta = 0.4$.
Suppose the α-similarity between r_i and g_1, g_2, g_3, g_4, g_5 is
$Sim_{r_i}[g_1] = 0.5$, $Sim_{r_i}[g_2] = 0$,
$Sim_{r_i}[g_3] = 0.5$, $Sim_{r_i}[g_4] = 0.2$, $Sim_{r_i}[g_5] = 0$.
$\Rightarrow r_i$ is ambiguous.

Preliminary experiments

- Positive and negative control datasets designed in [Lindgreen et al., 2016].
- Reference database $\mathcal{G}$: 930 genomes from 686 species

setA2	CLARK-S –highconf	LightMetaEBWT α 16 β 0.25	LightMetaEBWT α 16 β 0.35	Centrifuge -min-hitlen 16	Centrifuge -min-hitlen 22
SEN (%)	93.03	92.93	92.48	**95.65**	93.01
PREC (%)	99.06	99.81	**99.83**	97.64	99.66
F1 (%)	95.95	**96.24**	96.01	**96.63**	96.22
setB2					
SEN (%)	92.84	93.78	93.25	**95.53**	92.94
PREC (%)	99.11	99.62	99.64	97.68	**99.69**
F1 (%)	95.87	**96.61**	96.34	96.59	96.20
setA2Ran					
TN	5,726,336	5,726,294	5,726,357	150,971	5,712,085
FP	22	64	1	5,575,387	14,273
SPEC (%)	99.99	99.99	**100.00**	**2.64**	99.75
setB2Ran					
TN	5,406,642	5,406,601	5,406,658	141,994	5,393,260
FP	17	58	1	5,264,665	13,399
SPEC (%)	99.99	99.99	**100.00**	**2.63**	99.75

SEN = proportion of the actual positives identified by the method.
PREC = proportion of positives that are correctely identified by the method.
SPEC = proportion of actual negatives that are correctely identified as such.

Open Problems

Open problem

Prove whether some similarity measure based on eBWT is an approximation of Block Edit Distance.

The described "distance" are not a metric because neither it does obeys the triangle inequality.

Open problem

Define a new distance that is a metric.

Open Problems

Open problem

Prove whether some similarity measure based on eBWT is an approximation of Block Edit Distance.

The described “distance” are not a metric because neither it does obeys the triangle inequality.

Open problem

Define a new distance that is a metric.

Thank you!