Computation of melting temperature of nucleic acid duplexes with `rmelting`

Aravind, J.[1] and Krishna, G. K.[2]

2019-01-04

1. Division of Germplasm Conservation, ICAR-National Bureau of Plant Genetic Resources, New Delhi.
2. Division of Crop Physiology, ICAR-Indian Agricultural Research Institute, New Delhi.

Contents

1 Introduction

The R package rmelting is an interface to the MELTING 5 program (Le Novère, 2001; Dumousseau et al., 2012) to compute melting temperatures of nucleic acid duplexes (DNA/DNA, DNA/RNA, RNA/RNA or 2'-O-MeRNA/RNA) along with other thermodynamic parameters such as hybridisation enthalpy and entropy.

Melting temperatures are computed by Nearest-neighbour methods for short sequences or approximative estimation formulae for long sequences. Apart from these, several corrections are available to take into account the presence of Cations (Na, Tris, K and Mg) or denaturing agents (DMSO and formamide).

2 Installation

The package can be installed using the following functions:

```
# Install development version from Github
devtools::install_github("aravind-j/rmelting")
```

Then the package can be loaded using the function

```
library(rmelting)
```

3 Basic usage

Melting temperatures are computed in rmelting through the core function melting which takes a number of arguments (see ?melting). The following are the essential arguments which are mandatory for computation.

- sequence
  - 5' to 3' sequence of one strand of the nucleic acid duplex as a character string. Recognises A, C, G, T, U, I, X_C, X_T, A*, AL, TL, GL and CL (**Table 1**). U and T are not considered identical.

Table 1: Recognized sequences

Code	Type
A	Adenine
C	Cytosine
G	Guanine
T	Thymine
U	Uracil
I	Inosine
X_C	Trans azobenzenes
X_T	Cis azobenzenes
A*	Hydroxyadenine
AL	Locked nucleic acid

Code	Type
TL	”
GL	”
CL	”

- `Comp.sequence`
  - Mandatory if there are mismatches, inosine(s) or hydroxyadenine(s) between the two strands. If not specified, it is computed as the complement of `sequence`. Self-complementarity in `sequence` is detected even though there may be (are) dangling end(s) and `comp.sequence` is computed.
- `nucleic.acid.conc`
  - In molar concentration (M or mol L^{-1}).
- `Na.conc, Mg.conc, Tris.conc, K.conc`
  - At least one cation (Na, Mg, Tris, K) concentration is mandatory, the other agents(dNTP, DMSO, formamide) are optional.
- `hybridisation.type`
  - The possible options for hybridisation type are as follows (**Table 2**).

Table 2: Hybridisation type options

Option	Sequence	Complementary sequence
`dnadna`	DNA	DNA
`rnarna`	RNA	RNA
`dnarna`	DNA	RNA
`rnadna`	RNA	DNA
`mrnarna`	2-o-methyl RNA	RNA
`rnamrna`	RNA	2-o-methyl RNA

With these arguments, the melting temperature can be computed as follows.

```
melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1)
```

```
## [1] 73.35168
```

Only the melting temperature is given as a console output. However, the output can be assigned to an object which contains the details of the environment, options and the thermodynamics results as a list.

```
# Get output as list
out <- melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
               hybridisation.type = "dnadna", Na.conc = 1)
# Environment output
out$Environment
```

```
## $Sequence
## [1] "CAGTGAGACAGCAATGGTCG"
##
## $`Complementary sequence`
## [1] "GTCACTCTGTCGTTACCAGC"
##
## $`Nucleic acid concentration (M)`
## [1] 2e-06
##
## $`Hybridization type`
## [1] "dnadna"
```

```
##
## $`Na concentration (M)`
## [1] 1
##
## $`Mg concentration (M)`
## [1] 0
##
## $`Tris concentration (M)`
## [1] 0
##
## $`K concentration (M)`
## [1] 0
##
## $`dNTP concentration (M)`
## [1] 0
##
## $`DMSO concentration (%)`
## [1] 0
##
## $`Formamide concentration (M or %)`
## [1] 0
##
## $`Self complementarity`
## [1] FALSE
##
## $`Correction factor`
## [1] 4
```

```r
# Options used
out$Options
```

```
## $`Approximative formula`
## [1] NA
##
## $`Nearest neighbour model`
## [1] "all97"
##
## $`GU model`
## [1] NA
##
## $`Single mismatch model`
## [1] "allsanpey"
##
## $`Tandem mismatch model`
## [1] "allsanpey"
##
## $`Single dangling end model`
## [1] "bom00"
##
## $`Double dangling end model`
## [1] "sugdna02"
##
## $`Long dangling end model`
## [1] "sugdna02"
##
```

```
## $`Internal loop model`
## [1] "san04"
##
## $`Single bulge loop model`
## [1] "tan04"
##
## $`Long bulge loop model`
## [1] "san04"
##
## $`CNG repeats model`
## [1] NA
##
## $`Inosine bases model`
## [1] "san05"
##
## $`Hydroxyadenine bases model`
## [1] "sug01"
##
## $`Azobenzenes model`
## [1] "asa05"
##
## $`Locked nucleic acids model`
## [1] "mct04"
##
## $`Ion correction method`
## [1] NA
##
## $`Na equivalence correction method`
## [1] "ahs01"
##
## $`DMSO correction method`
## [1] "ahs01"
##
## $`Formamide correction method`
## [1] "bla96"
##
## $Mode
## [1] "def"
```

```
# Thermodynamics results
out$Results
```

```
## $`Enthalpy (cal)`
## [1] -159000
##
## $`Entropy (cal)`
## [1] -430
##
## $`Enthalpy (J)`
## [1] -664620
##
## $`Entropy (J)`
## [1] -1797.4
##
## $`Melting temperature (C)`
```

```
## [1] 73.35168
```

The command for the MELTING 5 java version is saved as an attribute in the list `out` and can be retrieved as follows.

```
# Command for MELTING 5
attributes(out)$command
```

```
## [1] "-S CAGTGAGACAGCAATGGTCG -H dnadna -P 2e-06 -E Na=1 -T 60"
```

4 Melting temperature computation

Melting temperature is computed by either approximative or nearest neighbour methods according to the length of the oligonucleotide sequences. For longer sequences (longer than the threshold value, the threshold value set by `size.threshold` with the default value 60) approximative method is used, while for others, nearest neighbour method is used.

4.1 Approximative methods

The approximative method for computation can be specified by the argument `method.approx`. The available methods are given in **Table 3**.

Table 3: Details of approximative methods

Formula	Type	Limits/Remarks	Reference
`ahs01`	DNA	No mismatch	Ahsen et al. (2001)
`che93`	DNA	No mismatch; Na=0, Mg=0.0015, Tris=0.01, K=0.05	Marmur and Doty (1962)
`che93corr`	DNA	No mismatch; Na=0, Mg=0.0015, Tris=0.01, K=0.05	Marmur and Doty (1962)
`schdot`	DNA	No mismatch	Wetmur (1991), Marmur and Doty (1962), Chester and Marshak (1993), Schildkraut and Lifson (1965), Wahl et al. (1987), Britten et al. (1974), Hall et al. (1980)
`owe69`	DNA	No mismatch	Owen et al. (1969), Frank-Kamenetskii (1971), Blake (1996), Blake and Delcourt (1998)
`san98`	DNA	No mismatch	SantaLucia (1998), Ahsen et al. (2001)
`wetdna91`*	DNA		Wetmur (1991)
`wetrna91`*	RNA		Wetmur (1991)
`wetdnarna91`*	DNA/RNA		Wetmur (1991)

* Default method for computation.

Examples

```
DNA:TCTAATGTGCTGTTAGATGTATCCAGAGATAGCCGAGCATAAACTTCAACACACGAGACGTTGATTGGATTTAACCATAG
    ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
DNA:AGATTACACGACAATCTACATAGGTCTCTATCGGCTCGTATTTGAAGTTGTGTGCTCTGCAACTAACCTAAATTGGTATC

RNA:UUAAUCUCCGUCAUCUUUAAGCCGUGGAGAGACUGUAGACUUGAACAGGGGUAAGCGGAGGCACGUAGGAUUCACAUCAU
```

```
    ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
RNA:AAUUAGAGGCAGUAGAAAUUCGGCACCUCUCUGACAUCUGAACUUGUCCCCAUUCGCCUCCGUGCAUCCUAAGUGUAGUA

DNA:TCTAATGTGCTGTTAGATGTATCCAGAGATAGCCGAGCATAAACTTCAACACACGAGACGTTGATTGGATTTAACCATAG
    ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
RNA:AGAUUACACGACAAUCUACAUAGGUCUCUAUCGGCUCGUAUUUGAAGUUGUGUGCUCUGCAACUAACCUAAAUUGGUAUC
```

```
# Long Nucleotide sequence
DNAseq <- c("TCTAATGTGCTGTTAGATGTATCCAGAGATAGCCGAGCATAAACTTCAACACACGAGACGTTGATTGGATTTAACCATAG")
RNAseq <- c("UUAAUCUCCGUCAUCUUUAAGCCGUGGAGAGACUGUAGACUUGAACAGGGGUAAGCGGAGGCACGUAGGAUUCACAUCAU")

# Approximative method - default (DNA/DNA)
melting(sequence = DNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1)
```

```
## [1] 87.82455
```

```
# Approximative method - wetdna91 (DNA/DNA)
melting(sequence = DNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1,
        method.approx = "wetdna91")
```

```
## [1] 87.82455
```

```
# Approximative method - ahs01 (DNA/DNA)
melting(sequence = DNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1,
        method.approx = "ahs01")
```

```
## [1] 87.325
```

```
# Approximative method - che93 (DNA/DNA)
melting(sequence = DNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1,
        method.approx = "che93")
```

```
## [1] 77.575
```

```
# Approximative method - che93corr (DNA/DNA)
melting(sequence = DNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1,
        method.approx = "che93corr")
```

```
## [1] 79.0125
```

```
# Approximative method - schdot (DNA/DNA)
melting(sequence = DNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1,
        method.approx = "schdot")
```

```
## [1] 89.4625
```

```
# Approximative method - owe69 (DNA/DNA)
melting(sequence = DNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1,
        method.approx = "owe69")
```

```
## [1] 100.96
```

```
# Approximative method - san98 (DNA/DNA)
melting(sequence = DNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1,
        method.approx = "san98")
```

```
## [1] 86.9
```

```
# Approximative method - default (RNA/RNA)
melting(sequence = RNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "rnarna", Na.conc = 1)
```

```
## [1] 101.1745
```

```
# Approximative method - wetrna91 (RNA/RNA)
melting(sequence = RNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "rnarna", Na.conc = 1,
        method.approx = "wetrna91")
```

```
## [1] 101.1745
```

```
# Approximative method - wetdnarna91 (DNA/RNA)
melting(sequence = DNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnarna", Na.conc = 1)
```

```
## [1] 88.92455
```

```
# Approximative method - wetdnarna91 (DNA/RNA)
melting(sequence = DNAseq, nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnarna", Na.conc = 1,
        method.approx = "wetdnarna91")
```

```
## [1] 88.92455
```

4.2 Nearest neighbour methods

4.2.1 Perfectly matching sequences

The nearest neighbour model for computation in case of perfectly matching sequences can be specified by the argument `method.nn`. The available methods are given in **Table 4**.

Table 4: Details of nearest neighbour methods for perfectly matching sequences

Model	Type	Limits/Remarks	Reference
`all97*`	DNA		Allawi and SantaLucia (1997)
`bre86`	DNA		Breslauer et al. (1986)
`san04`	DNA		SantaLucia and Hicks (2004)
`san96`	DNA		SantaLucia et al. (1996)
`sug96`	DNA		Sugimoto et al. (1996)
`tan04`	DNA		Tanaka et al. (2004)
`fre86`	RNA		Freier et al. (1986)
`xia98*`	RNA		Xia et al. (1998)
`sug95*`	DNA/ RNA		SantaLucia et al. (1996)
`tur06*`	2'-O-MeRNA/ RNA	A sodium correction (`san04`) is automatically applied to convert the entropy (Na = 0.1M) into the entropy (Na = 1M)	Kierzek et al. (2006)

* Default method for computation.

Examples

```
DNA:CAGTGAGACAGCAATGGTCG
    ||||||||||||||||||||
DNA:GTCACTCTGTCGTTACCAGC

RNA:CAGUGAGACAGCAAUGGUCG
    ||||||||||||||||||||
RNA:GUCACUCUGUCGUUACCAGC

DNA:CAGTGAGACAGCAATGGTCG
    ||||||||||||||||||||
RNA:GUCACUCUGUCGUUACCAGC
```

```r
# Nearest neighbour method - default (DNA/DNA: No Self-Complimentarity)
melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1)
```

```
## [1] 73.35168
```

```r
# Nearest neighbour method - all97 (DNA/DNA: No Self-Complimentarity)
melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "all97")
```

```
## [1] 73.35168
```

```r
# Nearest neighbour method - bre86 (DNA/DNA: No Self-Complimentarity)
melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "bre86")
```

```
## [1] 83.2203
```

```r
# Nearest neighbour method - san04 (DNA/DNA: No Self-Complimentarity)
melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "san04")
```

```
## [1] 73.30191
```

```r
# Nearest neighbour method - san96 (DNA/DNA: No Self-Complimentarity)
melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "san96")
```

```
## [1] 75.7102
```

```r
# Nearest neighbour method - sug96 (DNA/DNA: No Self-Complimentarity)
melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "sug96")
```

```
## [1] 78.17556
```

```r
# Nearest neighbour method - tan04 (DNA/DNA: No Self-Complimentarity)
melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "tan04")
```

```
## [1] 71.31413
```

```
# Nearest neighbour method - default (RNA/RNA: No Self-Complimentarity)
melting(sequence = "CAGUGAGACAGCAAUGGUCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "rnarna", Na.conc = 1)
```

```
## [1] 86.77685
```

```
# Nearest neighbour method - xia98 (RNA/RNA: No Self-Complimentarity)
melting(sequence = "CAGUGAGACAGCAAUGGUCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "rnarna", Na.conc = 1, method.nn = "xia98")
```

```
## [1] 86.77685
```

```
# Nearest neighbour method - fre86 (RNA/RNA: No Self-Complimentarity)
melting(sequence = "CAGUGAGACAGCAAUGGUCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "rnarna", Na.conc = 1, method.nn = "fre86")
```

```
## [1] 83.81257
```

```
# Nearest neighbour method - default (mRNA/RNA: No Self-Complimentarity)
melting(sequence = "CAGUGAGACAGCAAUGGUCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "mrnarna", Na.conc = 1)
```

```
## [1] 99.01986
```

```
# Nearest neighbour method - tur06 (mRNA/RNA: No Self-Complimentarity)
melting(sequence = "CAGUGAGACAGCAAUGGUCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "mrnarna", Na.conc = 1, method.nn = "tur06")
```

```
## [1] 99.01986
```

```
# Nearest neighbour method - default (DNA/RNA: No Self-Complimentarity)
melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnarna", Na.conc = 1)
```

```
## [1] 66.77049
```

```
# Nearest neighbour method - sug95 (DNA/RNA: No Self-Complimentarity)
melting(sequence = "CAGTGAGACAGCAATGGTCG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnarna", Na.conc = 1, method.nn = "sug95")
```

```
## [1] 66.77049
```

Self complementarity for perfect matching sequences or sequences with dangling ends is detected automatically. However it can be enforced by the argument `force.self = TRUE`.

Examples

```
DNA:CATATGGCCATATG
    ||||||||||||||
DNA:GTATACCGGTATAC

RNA:AUGUACAU
    ||||||||
RNA:UACAUGUA
```

```
# Nearest neighbour method - default (DNA/DNA: Self-Complimentarity)
melting(sequence = "CATATGGCCATATG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1)
```

```
## [1] 56.00644
```

```r
# Nearest neighbour method - all97 (DNA/DNA: Self-Complimentarity)
melting(sequence = "CATATGGCCATATG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "all97")
```

```
## [1] 56.00644
```

```r
# Nearest neighbour method - bre86 (DNA/DNA: Self-Complimentarity)
melting(sequence = "CATATGGCCATATG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "bre86")
```

```
## [1] 63.44605
```

```r
# Nearest neighbour method - san04 (DNA/DNA: Self-Complimentarity)
melting(sequence = "CATATGGCCATATG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "san04")
```

```
## [1] 57.80792
```

```r
# Nearest neighbour method - san96 (DNA/DNA: Self-Complimentarity)
melting(sequence = "CATATGGCCATATG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "san96")
```

```
## [1] 55.0921
```

```r
# Nearest neighbour method - sug96 (DNA/DNA: Self-Complimentarity)
melting(sequence = "CATATGGCCATATG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "sug96")
```

```
## [1] 59.06213
```

```r
# Nearest neighbour method - tan04 (DNA/DNA: Self-Complimentarity)
melting(sequence = "CATATGGCCATATG", nucleic.acid.conc = 2e-06,
        hybridisation.type = "dnadna", Na.conc = 1, method.nn = "tan04")
```

```
## [1] 55.65824
```

```r
# Nearest neighbour method - default (RNA/RNA: Self-Complimentarity)
melting(sequence = "AUGUACAU", nucleic.acid.conc = 2e-06,
        hybridisation.type = "rnarna", Na.conc = 1)
```

```
## [1] 30.27015
```

```r
# Nearest neighbour method - xia98 (RNA/RNA: Self-Complimentarity)
melting(sequence = "AUGUACAU", nucleic.acid.conc = 2e-06,
        hybridisation.type = "rnarna", Na.conc = 1, method.nn = "xia98")
```

```
## [1] 30.27015
```

```r
# Nearest neighbour method - fre86 (RNA/RNA: Self-Complimentarity)
melting(sequence = "AUGUACAU", nucleic.acid.conc = 2e-06,
        hybridisation.type = "rnarna", Na.conc = 1, method.nn = "fre86")
```

```
## [1] 31.48175
```

4.2.2 GU wobble base pairs effect

The nearest neighbour model for computation in case of sequences with GU wobble base pairs can be specified by the argument `method.GU`. The available methods are given in **Table 5**.

Table 5: Details of methods for sequences with GU wobble base pairs

Model	Type	Limits/Remarks	Reference
tur99	RNA		Mathews et al. (1999)
ser12*	RNA		Chen et al. (2012)

* Default method for computation.

Examples

```
RNA:CCAGCGUCCU
    ||||||||||
RNA:GGTCGCAGGA
```

```r
# GU wobble base pairs effect - default (RNA/RNA)
melting(sequence = "CCAGCGUCCU", nucleic.acid.conc = 0.0001,
        hybridisation.type = "rnarna", Na.conc = 1)
```

```
## [1] 79.46955
```

```r
# GU wobble base pairs effect - ser12 (RNA/RNA)
melting(sequence = "CCAGCGUCCU", nucleic.acid.conc = 0.0001,
        hybridisation.type = "rnarna", Na.conc = 1, method.GU = "ser12")
```

```
## [1] 79.46955
```

```r
# GU wobble base pairs effect - tur99 (RNA/RNA)
melting(sequence = "CCAGCGUCCU", nucleic.acid.conc = 0.0001,
        hybridisation.type = "rnarna", Na.conc = 1, method.GU = "tur99")
```

```
## [1] 79.46955
```

4.2.3 Single mismatch effect

The nearest neighbour model for computation in case of sequences with a single mismatch can be specified by the argument `method.singleMM`. The available methods are given in **Table 6**.

Table 6: Details of methods for sequences with single mismatch

Model	Type	Limits/Remarks	Reference
allsanpey*	DNA		Allawi and SantaLucia (1997), Allawi and SantaLucia (1998a), Allawi and SantaLucia (1998b), Allawi and SantaLucia (1998c), Peyret et al. (1999)
wat10*	DNA/RNA		Watkins et al. (2011)
tur06	RNA		Lu et al. (2006)
zno07*	RNA		Davis and Znosko (2007)
zno08	RNA	At least one adjacent GU base pair.	Davis and Znosko (2008)

* Default method for computation.

Examples

```
DNA:CAACTTGATATTAATA
```

```
    |||||||| |||||||
DNA:GTTGAACTCTAATTAT

RNA:GACAGGCUG
    |||| ||||
RNA:CUGUGCGAC

DNA:CCATAACTACC
    |||| ||||||
RNA:GGUAAUGAUGG
```

```r
# Single mismatch effect - default (DNA/DNA)
melting(sequence = "CAACTTGATATTAATA", comp.sequence = "GTTGAACTCTAATTAT",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna", Na.conc = 1)
```

```
## [1] 51.97499
```

```r
# Single mismatch effect - allsanpey (DNA/DNA)
melting(sequence = "CAACTTGATATTAATA", comp.sequence = "GTTGAACTCTAATTAT",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna",
        Na.conc = 1, method.singleMM = "allsanpey")
```

```
## [1] 51.97499
```

```r
# Single mismatch effect - default (RNA/RNA)
melting(sequence = "GACAGGCUG", comp.sequence = "CUGUGCGAC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna", Na.conc = 1)
```

```
## [1] 54.40363
```

```r
# Single mismatch effect - zno07 (RNA/RNA)
melting(sequence = "GACAGGCUG", comp.sequence = "CUGUGCGAC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.singleMM = "zno07")
```

```
## [1] 54.40363
```

```r
# Single mismatch effect - zno08 (RNA/RNA)
melting(sequence = "CAGUACGUC", comp.sequence = "GUCGGGCAG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.singleMM = "zno08")
```

```
## [1] 38.26298
```

```r
# Single mismatch effect - tur06 (RNA/RNA)
melting(sequence = "GACAGGCUG", comp.sequence = "CUGUGCGAC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.singleMM = "tur06")
```

```
## [1] 58.27825
```

```r
# Single mismatch effect - default (DNA/RNA)
melting(sequence = "CCATAACTACC", comp.sequence = "GGUAAUGAUGG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnarna", Na.conc = 1)
```

```
## [1] 40.32976
```

```r
# Single mismatch effect - wat11 (DNA/RNA)
melting(sequence = "CCATAACTACC", comp.sequence = "GGUAAUGAUGG",
```

```r
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnarna",
        Na.conc = 1, method.singleMM = "wat11")
```

```
## [1] 40.32976
```

4.2.4 Tandem mismatches effect

The nearest neighbour model for computation in case of sequences with tandem mismatches can be specified by the argument `method.tandemMM`. The available methods are given in **Table 7**.

Table 7: Details of methods for sequences with tandem mismatches

Model	Type	Limits/Remarks	Reference
`allsanpey`*	DNA	Only GT mismatches and TA/TG mismatches.	Allawi and SantaLucia (1997), Allawi and SantaLucia (1998a), Allawi and SantaLucia (1998b), Allawi and SantaLucia (1998c), Peyret et al. (1999)
`tur99`*	RNA	No adjacent GU or UG base pairs.	Mathews et al. (1999), Lu et al. (2006)

* Default method for computation.

Examples

```
DNA:GACGTTGGAC
    ||||  ||||
DNA:CTGCGGCCTG

RNA:GAGCGGAG
    |||  |||
RNA:CUCCACUC
```

```r
# Tandem mismatches effect - default (DNA/DNA)
melting(sequence = "GACGTTGGAC", comp.sequence = "CTGCGGCCTG",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna", Na.conc = 1)
```

```
## [1] 50.20175
```

```r
# Tandem mismatches effect - allsanpey (DNA/DNA)
melting(sequence = "GACGTTGGAC", comp.sequence = "CTGCGGCCTG",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna",
        Na.conc = 1, method.tandemMM = "allsanpey")
```

```
## [1] 50.20175
```

```r
# Tandem mismatches effect - default (RNA/RNA)
melting(sequence = "GAGCGGAG", comp.sequence = "CUCCACUC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna", Na.conc = 1)
```

```
## [1] 21.07224
```

```r
# Tandem mismatches effect -  tur06 (RNA/RNA)
melting(sequence = "GAGCGGAG", comp.sequence = "CUCCACUC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.tandemMM = "tur99")
```

```
## [1] 21.07224
```

4.2.5 Single dangling end effect

The nearest neighbour model for computation in case of sequences with a single dangling end can be specified by the argument `method.single.dangle`. The available methods are given in **Table 8**.

Table 8: Details of methods for sequences with single dangling end

Model	Type	Limits/Remarks	Reference
`bom00`*	DNA		Bommarito et al. (2000)
`sugdna02`	DNA	Only terminal poly A self complementary sequences.	Ohmichi et al. (2002)
`sugrna02`	RNA	Only terminal poly A self complementary sequences.	Ohmichi et al. (2002)
`ser08`*	RNA	Only 3' UA, GU and UG terminal base pairs only 5' UG and GU terminal base pairs.	O'Toole et al. (2006), Miller et al. (2008)

* Default method for computation.

Examples

```
DNA:-GTAGCTACA
     ||||||||
DNA:ACATCGATG-

RNA:-GGCGCUG
     |||||||
RNA: CCGCGAC

DNA:-GGCGCUG
     |||||||
RNA: CCGCGAC
```

```r
# Single dangling end effect - default (DNA/DNA)
melting(sequence = "-GTAGCTACA",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna",
        Na.conc = 1)
```

```
## [1] 52.58935
```

```r
# Single dangling end effect - bom00 (DNA/DNA)
melting(sequence = "-GTAGCTACA",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna",
        Na.conc = 1, method.single.dangle = "bom00")
```

```
## [1] 52.58935
```

```r
# Single dangling end effect - sugdna02 (DNA/DNA)
melting(sequence = "-GTAGCTACA",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna",
        Na.conc = 1, method.single.dangle = "sugdna02")
```

```
## [1] 50.78548
```

```r
# Single dangling end effect - default (RNA/RNA)
melting(sequence = "-GGCGCUG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1)
```

```
## [1] 65.7647
```

```r
# Single dangling end effect -  ser08 (RNA/RNA)
melting(sequence = "-GGCGCUG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.single.dangle = "ser08")
```

```
## [1] 65.7647
```

```r
# Single dangling end effect -  sugrna02 (RNA/RNA)
melting(sequence = "-GGCGCUG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.single.dangle = "sugrna02")
```

```
## [1] 65.7647
```

4.2.6 Double dangling end effect

The nearest neighbour model for computation in case of sequences with a double or secondary dangling ends can be specified by the argument `method.double.dangle`. The available methods are given in **Table 9**.

Table 9: Details of methods for sequences with double dangling ends

Model	Type	Limits/Remarks	Reference
`sugdna02`*	DNA	Only terminal poly A self complementary sequences.	Ohmichi et al. (2002)
`sugrna02`	RNA	Only terminal poly A self complementary sequences.	Ohmichi et al. (2002)
`ser05`	RNA	Depends on the available thermodynamic parameters for single dangling end.	O'Toole et al. (2005)
`ser06`*	RNA		O'Toole et al. (2006)

* Default method for computation.

Examples

```
DNA:--ATGCATAA
      ||||||
DNA:AATACGTA--

RNA:--AUGCAUAA
      ||||||
RNA:AAUACGUA--
```

```r
# Double dangling end effect - default (DNA/DNA)
melting(sequence = "--ATGCATAA",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna",
        Na.conc = 1)
```

```
## [1] 44.88615
```

```r
# Double dangling end effect - sugdna02 (DNA/DNA)
melting(sequence = "--ATGCATAA",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna",
        Na.conc = 1, method.double.dangle = "sugdna02")
```

```
## [1] 44.88615
```

```r
# Double dangling end effect - default (RNA/RNA)
melting(sequence = "--AUGCAUAA",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1)
```

```
## [1] 42.79724
```

```r
# Double dangling end effect -  ser06 (RNA/RNA)
melting(sequence = "--AUGCAUAA",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.double.dangle = "ser06")
```

```
## [1] 42.79724
```

```r
# Double dangling end effect -  sugrna02 (RNA/RNA)
melting(sequence = "--AUGCAUAA",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.double.dangle = "sugrna02")
```

```
## [1] 41.82788
```

```r
# Double dangling end effect -  ser05 (RNA/RNA)
melting(sequence = "--AUGCAUAA",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.double.dangle = "ser05")
```

```
## [1] 42.78815
```

4.2.7 Long dangling end effect

The nearest neighbour model for computation in case of sequences with a double or secondary dangling ends can be specified by the argument `method.long.dangle`. The available methods are given in **Table 10**.

Table 10: Details of methods for sequences with long dangling ends

Model	Type	Limits/Remarks	Reference
`sugdna02`*	DNA	Only terminal poly A self complementary sequences.	Ohmichi et al. (2002)
`sugrna02`*	RNA	Only terminal poly A self complementary sequences.	Ohmichi et al. (2002)

* Default method for computation.

Examples

```
DNA:----GCATATGCAAAA
        ||||||||
```

```
DNA:AAAACGTATACG----

RNA:AAAAGCAUAUGC----
        ||||||||
RNA:----CGUAUACGAAAA
```

```r
# Long dangling end effect - default (DNA/DNA)
melting(sequence = "----GCATATGCAAAA",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna",
        Na.conc = 1)
```

```
## [1] 55.69854
```

```r
# Long dangling end effect - sugdna02 (DNA/DNA)
melting(sequence = "----GCATATGCAAAA",
        nucleic.acid.conc = 0.0004, hybridisation.type = "dnadna",
        Na.conc = 1, method.long.dangle = "sugdna02")
```

```
## [1] 55.69854
```

```r
# Long dangling end effect - default (RNA/RNA)
melting(sequence = "AAAAGCAUAUGC----",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1)
```

```
## [1] 57.21314
```

```r
# Long dangling end effect -  sugrna02 (RNA/RNA)
melting(sequence = "AAAAGCAUAUGC----",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.long.dangle = "sugrna02")
```

```
## [1] 57.21314
```

4.2.8 Internal loop effect

The nearest neighbour model for computation in case of sequences with an internal loop (more than two adjacent mismatches) can be specified by the argument `method.internal.loop`. The available methods are given in **Table 11**.

Table 11: Details of methods for sequences with internal loops

Model	Type	Limits/Remarks	Reference
`san04`*	DNA	Missing asymmetry penalty. Not tested with experimental results.	SantaLucia and Hicks (2004)
`tur06`	RNA	Not tested with experimental results.	Lu et al. (2006)
`zno07`*	RNA	Only for 1x2 loop.	Badhwar et al. (2007)

* Default method for computation.

Examples

```
DNA:GCGATTGGCACTTTGGTGAAC
        |||||    ||||||||||||
```

```
DNA:CGCTACATATGAAACCACTTG

RNA:GACAC-GCUG
    ||||  ||||
RNA:CUGUAUCGAC
```

```r
# Internal loop effect - default (DNA/DNA)
melting(sequence = "GCGATTGGCACTTTGGTGAAC", comp.sequence = "CGCTACATATGAAACCACTTG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1)
```

```
## [1] 84.09052
```

```r
# Internal loop effect - san04 (DNA/DNA)
melting(sequence = "GCGATTGGCACTTTGGTGAAC", comp.sequence = "CGCTACATATGAAACCACTTG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1, method.internal.loop = "san04")
```

```
## [1] 84.09052
```

```r
# Internal loop effect - default (RNA/RNA)
melting(sequence = "GACAC-GCUG", comp.sequence = "CUGUAUCGAC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1)
```

```
## [1] 45.98713
```

```r
# Internal loop effect -  zno07 (RNA/RNA)
melting(sequence = "GACAC-GCUG", comp.sequence = "CUGUAUCGAC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.internal.loop = "zno07")
```

```
## [1] 40.49012
```

```r
# Internal loop effect -  tur06 (RNA/RNA)
melting(sequence = "GACAC-GCUG", comp.sequence = "CUGUAUCGAC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.internal.loop = "tur06")
```

```
## [1] 45.98713
```

4.2.9 Single bulge loop effect

The nearest neighbour model for computation in case of sequences with a single bulge loop can be specified by the argument `method.single.bulge.loop`. The available methods are given in **Table 12**.

Table 12: Details of methods for sequences with single bulge loop

Model	Type	Limits/Remarks	Reference
`tan04`*	DNA		Tan and Chen (2007)
`san04`	DNA	Missing closing AT penalty.	SantaLucia and Hicks (2004)
`ser07`	RNA	Less reliable results. Some missing parameters.	Blose et al. (2007)
`tur06`*	RNA		Lu et al. (2006)

* Default method for computation.

Examples

```
DNA:TCGATTAGCGACACAGG
    |||||||| ||||||||
DNA:AGCTAATC-CTGTGTCC

RNA:GACUCUGUC
    |||| ||||
RNA:CUGA-ACAG
```

```r
# Single bulge loop effect - default (DNA/DNA)
melting(sequence = "TCGATTAGCGACACAGG", comp.sequence = "AGCTAATC-CTGTGTCC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1)
```

```
## [1] 71.12754
```

```r
# Single bulge loop effect - tan04 (DNA/DNA)
melting(sequence = "TCGATTAGCGACACAGG", comp.sequence = "AGCTAATC-CTGTGTCC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1, method.single.bulge.loop = "tan04")
```

```
## [1] 71.12754
```

```r
# Single bulge loop effect - san04 (DNA/DNA)
melting(sequence = "TCGATTAGCGACACAGG", comp.sequence = "AGCTAATC-CTGTGTCC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1, method.single.bulge.loop = "san04")
```

```
## [1] 62.0496
```

```r
# Single bulge loop effect - default (RNA/RNA)
melting(sequence = "GACUCUGUC", comp.sequence = "CUGA-ACAG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1)
```

```
## [1] 39.47787
```

```r
# Single bulge loop effect -  tur06 (RNA/RNA)
melting(sequence = "GACUCUGUC", comp.sequence = "CUGA-ACAG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.single.bulge.loop = "tur06")
```

```
## [1] 39.47787
```

```r
# Single bulge loop effect -  ser07 (RNA/RNA)
melting(sequence = "GACUCUGUC", comp.sequence = "CUGA-ACAG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.single.bulge.loop = "ser07")
```

```
## [1] 31.42849
```

4.2.10 Long bulge loop effect

The nearest neighbour model for computation in case of sequences with long bulge loop can be specified by the argument `method.long.bulge.loop`. The available methods are given in **Table 13**.

Table 13: Details of methods for sequences with long bulge loop

Model	Type	Limits/Remarks	Reference
san04*	DNA	Missing closing AT penalty.	SantaLucia and Hicks (2004)
tur06*	RNA	Not tested with experimental results.	Mathews et al. (1999), Lu et al. (2006)

* Default method for computation.

Examples

```
DNA:ATATGACGCCACAGCG
    |||||   ||||||||
DNA:TATAC---GGTGTCGC

RNA:AUAUGACGCCACAGCG
    |||||   ||||||||
RNA:UAUAC---GGUGUCGC
```

```r
# Long bulge loop effect - default (DNA/DNA)
melting(sequence = "ATATGACGCCACAGCG", comp.sequence = "TATAC---GGTGTCGC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1)
```

```
## [1] 51.7104
```

```r
# Long bulge loop effect - san04 (DNA/DNA)
melting(sequence = "ATATGACGCCACAGCG", comp.sequence = "TATAC---GGTGTCGC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1, method.long.bulge.loop = "san04")
```

```
## [1] 51.7104
```

```r
# Long bulge loop effect - default (RNA/RNA)
melting(sequence = "AUAUGACGCCACAGCG", comp.sequence = "UAUAC---GGUGUCGC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1)
```

```
## [1] 66.0497
```

```r
# Long bulge loop effect -  tur06 (RNA/RNA)
melting(sequence = "AUAUGACGCCACAGCG", comp.sequence = "UAUAC---GGUGUCGC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.long.bulge.loop = "tur06")
```

```
## [1] 66.0497
```

4.2.11 CNG repeats effect

The nearest neighbour model for computation in case of sequences with CNG repeats can be specified by the argument `method.CNG`. The available methods are given in **Table 14**.

Table 14: Details of methods for sequences with CNG repeats

Model	Type	Limits/Remarks	Reference
bro05*	RNA	Self complementary sequences. 2 to 7 CNG repeats.	Broda et al. (2005)

* Default method for computation.

Examples

```
RNA:GCGGCGGCGGC
    |||||||||||
RNA:CGCCGCCGCCG
```

```r
# CNG repeats effect - default (RNA/RNA)
melting(sequence = "GCGGCGGCGGC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1)
```

```
## [1] 94.25719
```

```r
# CNG repeats effect -  bro05 (RNA/RNA)
melting(sequence = "GCGGCGGCGGC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.CNG = "bro05")
```

```
## [1] 94.25719
```

4.2.12 Inosine bases effect

The nearest neighbour model for computation in case of sequences with inosine bases (I) can be specified by the argument `method.inosine`. The available methods are given in **Table 15**.

Table 15: Details of methods for sequences with inosine bases

Model	Type	Limits/Remarks	Reference
san05*	DNA	Missing parameters for tandem base pairs containing inosine bases.	Watkins and SantaLucia (2005)
zno07*	RNA	Only IU base pairs.	Wright et al. (2007)

* Default method for computation.

Examples

```
DNA:CCGICTGTIGCG
    ||| |||| |||
DNA:GGCCGACACCGC

RNA:GCAICGC
    ||| |||
RNA:CGUUGCG
```

```r
# Inosine bases effect - default (DNA/DNA)
melting(sequence = "CCGICTGTIGCG", comp.sequence = "GGCCGACACCGC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1)
```

```
## [1] 65.36853
```

```r
# Inosine bases effect - san05 (DNA/DNA)
melting(sequence = "CCGICTGTIGCG", comp.sequence = "GGCCGACACCGC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1, method.inosine = "san05")
```

```
## [1] 65.36853
```

```r
# Inosine bases effect - default (RNA/RNA)
melting(sequence = "GCAICGC", comp.sequence = "CGUUGCG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1)
```

```
## [1] 46.75042
```

```r
# Inosine bases effect -  zno07 (RNA/RNA)
melting(sequence = "GCAICGC", comp.sequence = "CGUUGCG",
        nucleic.acid.conc = 0.0001, hybridisation.type = "rnarna",
        Na.conc = 1, method.inosine = "zno07")
```

```
## [1] 46.75042
```

4.2.13 Hydroxyadenine bases effect

The nearest neighbour model for computation in case of sequences with hydroxyadenine bases can be specified by the argument `method.hydroxyadenine`. The available methods are given in **Table 16**.

Table 16: Details of methods for sequences with hydroxyadenine bases

Model	Type	Limits/Remarks	Reference
`sug01`*	DNA	Only 5' GA*C 3'and 5' TA*A 3' contexts.	Kawakami et al. (2001)

* Default method for computation.

Examples

```
               *
    DNA:AGAAATGACACGGTG
        |||||||||||||||
    DNA:TCTTTACCGTGCCAC
```

```r
# Hydroxyadenine bases effect - default (DNA/DNA)
melting(sequence = "AGAAATGA*CACGGTG", comp.sequence = "TCTTTACCGTGCCAC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1)
```

```
## [1] 68.46041
```

```r
# Hydroxyadenine bases effect - sug01 (DNA/DNA)
melting(sequence = "AGAAATGA*CACGGTG", comp.sequence = "TCTTTACCGTGCCAC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1, method.hydroxyadenine = "sug01")
```

```
## [1] 68.46041
```

4.2.14 Azobenzenes effect

The nearest neighbour model for computation in case of sequences with azobenzenes (X_T for trans azobenzenes and X_C for cis azobenzenes) can be specified by the argument `method.azobenzenes`. The available methods are given in **Table 17**.

Table 17: Details of methods for sequences with azobenzenes

Model	Type	Limits/Remarks	Reference
asa05*	DNA	Less reliable results when the number of cis azobenzene increases.	Asanuma et al. (2005)

* Default method for computation.

Examples

```
         C    C    C    C    C
    DNA:CTXTTAAXGAAGXGAGAXTATAXCC
        || |||| |||| |||| |||| ||
    DNA:GA AATT CTTC CTCT ATAT GG
```

```r
# Azobenzenes effect - default (DNA/DNA)
melting(sequence = "CTX_CTTAAX_CGAAGX_CGAGAX_CTATAX_CCC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1)
```

```
## [1] 47.85385
```

```r
# Azobenzenes effect - asa05 (DNA/DNA)
melting(sequence = "CTX_CTTAAX_CGAAGX_CGAGAX_CTATAX_CCC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1, method.azobenzenes = "asa05")
```

```
## [1] 47.85385
```

4.2.15 Locked nucleic acids effect

The nearest neighbour model for computation in case of sequences with locked nucleic acids can be specified by the argument `method.locked`. The available methods are given in **Table 18**.

Table 18: Details of methods for sequences with locked nucleic acids

Model	Type	Limits/Remarks	Reference
mct04*	DNA		McTigue et al. (2004)

* Default method for computation.

Examples

```
            L
    DNA:CCATTGCTACC
        |||||||||||
    DNA:GGTAACGATGG
```

```r
# Locked nucleic acids effect - default (DNA/DNA)
melting(sequence = "CCATTLGCTACC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1)
```

```
## [1] 63.61426
```

```r
# Locked nucleic acids effect - sug01 (DNA/DNA)
melting(sequence = "CCATTLGCTACC",
        nucleic.acid.conc = 0.0001, hybridisation.type = "dnadna",
        Na.conc = 1, method.locked = "mct04")
```

```
## [1] 63.61426
```

5 Corrections

Once the melting temperature is computed, a correction is applied to it according to the concentration of nucleic acids, cations and/or denaturing agents.

5.1 Nucleic acid concentration

For self complementary sequences (auto detected or specified by `force.self`) it is 1. Otherwise it is 4 if the both strands are present in equivalent amount and 1 if one strand is in excess.

5.2 Ion corrections

Melting temperature is computed initially for $[Na^+] = 1$ M, after which a correction for the presence of cations ($[Na^+]$, $[K^+]$, $[Tris^+]$ and $[Mg^+]$) is applied either directly on the computed melting temperature or on the computed entropy.

Th correction methods for cation concentration can be specified by the argument `correction.ion`.

5.2.1 Sodium corrections

The available correction methods for sodium concentration are given in **Table 19**.

Table 19: Details of the corrections for sodium concentration

Correction	Type	Limits/Remarks	Reference
`ahs01`	DNA	Na>0.	Ahsen et al. (2001)
`kam71`	DNA	Na>0; Na>=0.069; Na<=1.02.	Frank-Kamenetskii (1971)
`marschdot`	DNA	Na>=0.069; Na<=1.02.	Marmur and Doty (1962), Blake and Delcourt (1998)
`owc1904`	DNA	Na>0. (equation 19)	Owczarzy et al. (2004)
`owc2004`	DNA	Na>0. (equation 20)	Owczarzy et al. (2004)
`owc2104`	DNA	Na>0. (equation 21)	Owczarzy et al. (2004)

Correction	Type	Limits/Remarks	Reference
`owc2204`*	DNA	Na>0. (equation 22)	Owczarzy et al. (2004)
`san96`	DNA	Na>=0.1.	SantaLucia et al. (1996)
`san04`	DNA	Na>=0.05; Na<=1.1; Oligonucleotides inferior to 16 bases.	SantaLucia and Hicks (2004), SantaLucia (1998)
`schlif`	DNA	Na>=0.07; Na<=0.12.	Schildkraut and Lifson (1965)
`tanna06`	DNA	Na>=0.001; Na<=1.	Tan and Chen (2006)
`tanna07`*	RNA or 2'-O-MeRNA/RNA	Na>=0.003; Na<=1.	Tan and Chen (2007)
`wet91`	RNA, DNA and RNA/DNA	Na>0.	Wetmur (1991)

* Default method for computation.

```
# Na correction - default (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069)
```

```
## [1] 56.70492
```

```
# Na correction - owc2204 (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "owc2204")
```

```
## [1] 56.70492
```

```
# Na correction - ahs01 (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "ahs01")
```

```
## [1] 54.1569
```

```
# Na correction - kam71 (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "kam71")
```

```
## [1] 51.72963
```

```
# Na correction - marschdot (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "marschdot")
```

```
## [1] 49.18075
```

```
# Na correction - owc1904 (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "owc1904")
```

```
## [1] 56.18571
```

```r
# Na correction - owc2004 (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "owc2004")
```

```
## [1] 56.67553
```

```r
# Na correction - owc2104 (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "owc2104")
```

```
## [1] 56.63967
```

```r
# Na correction - san96 (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "san96")
```

```
## [1] 53.01651
```

```r
# Na correction - san04 (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "san04")
```

```
## [1] 54.15157
```

```r
# Na correction - schlif (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "schlif")
```

```
## [1] 48.25579
```

```r
# Na correction - tanna06 (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "tanna06")
```

```
## [1] 55.26711
```

```r
# Na correction - wet91 (DNA/DNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, correction.ion = "wet91")
```

```
## [1] 51.74573
```

```r
# Na correction - default (RNA/RNA)
melting(sequence = "CCAGCCAGUCUCUCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "rnarna",
        Na.conc = 0.069)
```

```
## [1] 75.1552
```

```r
# Na correction -  tanna07 (RNA/RNA)
melting(sequence = "CCAGCCAGUCUCUCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "rnarna",
        Na.conc = 0.069, correction.ion = "tanna07")
```

```
## [1] 75.1552
```

```r
# Na correction -  wet91 (RNA/RNA)
melting(sequence = "CCAGCCAGUCUCUCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "rnarna",
        Na.conc = 0.069, correction.ion = "wet91")
```

```
## [1] 69.55572
```

```r
# Na correction - default (mRNA/RNA)
melting(sequence = "UACGCGUCAAUAACGCUA",
        nucleic.acid.conc = 0.000002, hybridisation.type = "mrnarna",
        Na.conc = 0.069)
```

```
## [1] 81.57763
```

```r
# Na correction -  tanna07 (mRNA/RNA)
melting(sequence = "UACGCGUCAAUAACGCUA",
        nucleic.acid.conc = 0.000002, hybridisation.type = "mrnarna",
        Na.conc = 0.069, correction.ion = "tanna07")
```

```
## [1] 81.57763
```

```r
# Na correction - default (DNA/RNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnarna",
        Na.conc = 0.069)
```

```
## [1] 62.08869
```

```r
# Na correction - wet91 (DNA/RNA)
melting(sequence = "CCAGCCAGTCTCTCC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnarna",
        Na.conc = 0.069, correction.ion = "wet91")
```

```
## [1] 62.08869
```

5.2.2 Magnesium corrections

The available correction methods for magnesium concentration are given in **Table 20**.

Table 20: Details of the corrections for magnesium concentration

Correction	Type	Limits/Remarks	Reference
`owcmg08`*	DNA	Mg>=0.0005; Mg<=0.6.	Owczarzy et al. (2008)
`tanmg06`	DNA	Mg>=0.0001; Mg<=1; Oligomer length superior to 6 base pairs.	Tan and Chen (2006)
`tanmg07`*	RNA or 2'-O-MeRNA/RNA	Mg>=0.1; Mg<=0.3.	Tan and Chen (2007)

* Default method for computation.

```r
# Mg correction - default (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
```

```
        Mg.conc = 0.0015)
```

```
## [1] 65.52043
```

```
# Mg correction - owcmg08 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Mg.conc = 0.0015, correction.ion = "owcmg08")
```

```
## [1] 65.52043
```

```
# Mg correction - tanmg06 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Mg.conc = 0.0015, correction.ion = "tanmg06")
```

```
## [1] 64.88082
```

```
# Mg correction - default (RNA/RNA)
melting(sequence = "CAGCCUCGUCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "rnarna",
        Mg.conc = 0.0015)
```

```
## [1] 82.0796
```

```
# Mg correction -  tanmg07 (RNA/RNA)
melting(sequence = "CAGCCUCGUCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "rnarna",
        Mg.conc = 0.0015, correction.ion = "tanmg07")
```

```
## [1] 82.0796
```

```
# Mg correction - default (mRNA/RNA)
melting(sequence = "UACGCGUCAAUAACGCUA",
        nucleic.acid.conc = 0.000002, hybridisation.type = "mrnarna",
        Mg.conc = 0.0015)
```

```
## [1] 90.06842
```

```
# Mg correction -  tanmg07 (mRNA/RNA)
melting(sequence = "UACGCGUCAAUAACGCUA",
        nucleic.acid.conc = 0.000002, hybridisation.type = "mrnarna",
        Mg.conc = 0.0015, correction.ion = "tanmg07")
```

```
## [1] 90.06842
```

5.2.3 Mixed Sodium and Magnesium corrections

The available correction methods for mixed sodium magnesium concentration are given in **Table 21**.

Table 21: Details of the corrections for mixed sodium and magnesium concentration

Correction	Type	Limits/Remarks	Reference
owcmix08*	DNA	Mg>=0.0005; Mg<=0.6; Na+K+Tris/2>0.	Owczarzy et al. (2008)
tanmix07	DNA, RNA or 2'-O-MeRNA/RNA	Mg>=0.1; Mg<=0.3; Na+K+Tris/2>=0.1; Na+K+Tris/2<=0.3.	Tan and Chen (2007)

* Default method for computation.

```r
# Mixed Na & Mg correction - default (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, Mg.conc = 0.0015)
```

```
## [1] 65.83371
```

```r
# Mixed Na & Mg correction - owcmix08 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, Mg.conc = 0.0015, correction.ion = "owcmix08")
```

```
## [1] 65.83371
```

```r
# Mixed Na & Mg correction - tanmix07 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, Mg.conc = 0.0015, correction.ion = "tanmix07")
```

```
## [1] 63.21723
```

```r
# Mixed Na & Mg correction - default (RNA/RNA)
melting(sequence = "CAGCCUCGUCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "rnarna",
        Na.conc = 0.069, Mg.conc = 0.0015)
```

```
## [1] 79.40119
```

```r
# Mixed Na & Mg correction - tanmix07 (RNA/RNA)
melting(sequence = "CAGCCUCGUCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "rnarna",
        Na.conc = 0.069, Mg.conc = 0.0015, correction.ion = "tanmix07")
```

```
## [1] 79.40119
```

```r
# Mixed Na & Mg correction - default (mRNA/RNA)
melting(sequence = "UACGCGUCAAUAACGCUA",
        nucleic.acid.conc = 0.000002, hybridisation.type = "mrnarna",
        Na.conc = 0.069, Mg.conc = 0.0015)
```

```
## [1] 96.46186
```

```r
# Mixed Na & Mg correction - tanmix07 (mRNA/RNA)
melting(sequence = "UACGCGUCAAUAACGCUA",
        nucleic.acid.conc = 0.000002, hybridisation.type = "mrnarna",
        Na.conc = 0.069, Mg.conc = 0.0015, correction.ion = "tanmix07")
```

```
## [1] 96.46186
```

The ion correction by Owczarzy et al. (2008) is used by default according to the $\frac{[Mg^{2+}]^{0.5}}{[Mon^{+}]}$ ratio, where $[Mon^{+}] = Na^{+}] + [Tris^{+}] + [K^{+}]$.

If,

- $[K^{+}] = 0$, default sodium correction is used;
- Ratio < 0.22, default sodium correction is used;
- $0.22 \leq$ Ratio < 6, default mixed Na and Mg correction is used and

- Ratio ≥ 6,default magnesium correction is used.

Note that [$Tris^+$] is about half of the total tris buffer concentration.

5.2.4 Sodium equivalent concentration methods

The available correction methods for mixed sodium magnesium concentration are given in **Table 22**.

Table 22: Details of the methods for computation of sodium equivalent concentration in the presence of other ions

Correction	Type	Limits/Remarks	Reference
`ahs01`*	DNA		Ahsen et al. (2001)
`mit96`	DNA		Mitsuhashi (1996)
`pey00`	DNA		Peyret (2000)

* Default method for computation.

```
# Na equivalent concentration method - default (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, Mg.conc = 0.0015)
```

```
## [1] 65.83371
```

```
# Na equivalent concentration method - ahs01 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, Mg.conc = 0.0015, method.Naeq = "ahs01")
```

```
## [1] 65.83371
```

```
# Na equivalent concentration method - mit96 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, Mg.conc = 0.0015, method.Naeq = "mit96")
```

```
## [1] 65.83371
```

```
# Na equivalent concentration method - pey00 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 0.069, Mg.conc = 0.0015, method.Naeq = "pey00")
```

```
## [1] 65.83371
```

5.3 Denaturing agent corrections

These include melting temperature corrections for concentration of formamide and DMSO.

5.3.1 DMSO corrections

The available correction methods for DMSO concentration are given in **Table 23**.

Table 23: Details of the corrections for DMSO concentration

Correction	Type	Limits/Remarks	Reference
ahs01*	DNA	Not tested with experimental results.	Ahsen et al. (2001)
cul76	DNA	Not tested with experimental results.	Cullen and Bick (1976)
esc80	DNA	Not tested with experimental results.	Escara and Hutton (1980)
mus81	DNA	Not tested with experimental results.	Musielski et al. (1981)

* Default method for computation.

```
# DMSO correction - default (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 1, DMSO.conc = 10)
```

```
## [1] 65.40154
```

```
# DMSO correction - ahs01 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 1, DMSO.conc = 10, correction.DMSO = "ahs01")
```

```
## [1] 65.40154
```

```
# DMSO correction - cul76 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 1, DMSO.conc = 10, correction.DMSO = "cul76")
```

```
## [1] 67.90154
```

```
# DMSO correction - esc80 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 1, DMSO.conc = 10, correction.DMSO = "esc80")
```

```
## [1] 66.15154
```

```
# DMSO correction - mus80 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 1, DMSO.conc = 10, correction.DMSO = "mus81")
```

```
## [1] 66.90154
```

5.3.2 Formamide corrections

The available correction methods for formamide concentration are given in **Table 24**.

Table 24: Details of the corrections for formamide concentration

Correction	Type	Limits/Remarks	Reference
bla96*	DNA	With formamide concentration in mol/L.	Blake (1996)

Correction	Type	Limits/Remarks	Reference
`lincorr`	DNA	With a % of formamide volume.	McConaughy et al. (1969), Record (1967), Casey and Davidson (1977), Hutton (1977)

* Default method for computation.

```
# Formamide correction - default (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 1, formamide.conc = 0.06)
```

```
## [1] 72.74867
```

```
# Formamide correction - bla96 (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 1, formamide.conc = 0.06, correction.formamide = "bla96")
```

```
## [1] 72.74867
```

```
# Formamide correction - lincorr (DNA/DNA)
melting(sequence = "CAGCCTCGTCGCAGC",
        nucleic.acid.conc = 0.000002, hybridisation.type = "dnadna",
        Na.conc = 1, formamide.conc = 10, correction.formamide = "lincorr")
```

```
## [1] 66.40154
```

6 Equivalent options in MELTING 5

The options in MELTING 5 command line equivalent to the arguments in `rmelting` are given in **Table 25**.

Table 24: Arguments in `rmelting` and their equivalent options in MELTING 5 command line.

`rmelting`	MELTING 5 (command line)
`sequence`	-S
`comp.sequence`	-C
`nucleic.acid.conc`	-P
`hybridisation.type`	-H
`Na.conc`	-E
`Mg.conc`	-E
`Tris.conc`	-E
`K.conc`	-E
`dNTP.conc`	-E
`DMSO.conc`	-E
`formamide.conc`	-E
`size.threshold`	-T
`self`	-self
`correction.factor`	-F
`method.approx`	-am
`method.nn`	-nn
`method.GU`	-GU
`method.singleMM`	-sinMM

`rmelting`	MELTING 5 (command line)
`method.tandemMM`	-tanMM
`method.single.dangle`	-sinDE
`method.double.dangle`	-secDE
`method.long.dangle`	-lonDE
`method.internal.loop`	-intLP
`method.single.bulge.loop`	-sinBU
`method.long.bulge.loop`	-lonBU
`method.CNG`	-CNG
`method.inosine`	-ino
`method.hydroxyadenine`	-ha
`method.azobenzenes`	-azo
`method.locked`	-lck
`correction.ion`	-ion
`method.Naeq`	-naeq
`correction.DMSO`	-DMSO
`correction.formamide`	-for

7 Batch

8 Further reading

Further details about algorithm, formulae and methods are available in the MELTING 5 documentation.

9 Citing `rmelting`

```
##
## To cite the R package 'rmelting' in publications use:
##
##   Aravind, J. and Krishna, G. K. (2019).  rmelting: R Interface to
##   MELTING 5. R package version 0.99.1,
##   https://aravind-j.github.io/rmelting/.
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {rmelting: R Interface to MELTING 5},
##     author = {J. Aravind and G. K. Krishna},
##     year = {2019},
##     note = {R package version 0.99.1},
##     note = {https://aravind-j.github.io/rmelting/},
##   }
##
## This free and open-source software implements academic research by
## the authors and co-workers. If you use it, please support the
## project by citing the package.
```

10 Session Info

```r
sessionInfo()
```

```
## R Under development (unstable) (2018-10-27 r75507)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows >= 8 x64 (build 9200)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_India.1252  LC_CTYPE=English_India.1252
## [3] LC_MONETARY=English_India.1252 LC_NUMERIC=C
## [5] LC_TIME=English_India.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] rmelting_0.99.1 printr_0.1      readxl_1.1.0    testthat_2.0.1
## [5] rJava_0.9-10
##
## loaded via a namespace (and not attached):
##  [1] httr_1.3.1         pkgload_1.0.2      jsonlite_1.5
##  [4] Rdpack_0.10-3      assertthat_0.2.0   xmlparsedata_1.0.2
##  [7] highr_0.7          pander_0.6.3       cellranger_1.1.0
## [10] yaml_2.2.0         remotes_2.0.2      sessioninfo_1.1.1
## [13] pillar_1.3.0       backports_1.1.2    glue_1.3.0
## [16] goodpractice_1.0.2 digest_0.6.18      htmltools_0.3.6
## [19] clisymbols_1.2.0   devtools_2.0.1     bibtex_0.4.2
## [22] rcmdcheck_1.3.2    purrr_0.2.5        processx_3.2.0
## [25] tibble_1.4.2       usethis_1.4.0      withr_2.1.2
## [28] lazyeval_0.2.1     cli_1.0.1          magrittr_1.5
## [31] crayon_1.3.4       memoise_1.1.0      evaluate_0.12
## [34] ps_1.2.1           fs_1.2.6           MASS_7.3-51.1
## [37] xml2_1.2.0         pkgbuild_1.0.2     praise_1.0.0
## [40] tools_3.6.0        hunspell_3.0       prettyunits_1.0.2
## [43] cyclocomp_1.1.0    gbRd_0.4-11        stringr_1.3.1
## [46] xopen_1.0.0        callr_3.0.0        rex_1.1.2
## [49] compiler_3.6.0     pkgdown_1.3.0.9000 covr_3.2.1
## [52] tinytex_0.9        rlang_0.3.0.1      debugme_1.1.0
## [55] rstudioapi_0.8     base64enc_0.1-3    rmarkdown_1.10
## [58] roxygen2_6.1.1     rematch2_2.0.1     R6_2.3.0
## [61] knitr_1.20         commonmark_1.7     rprojroot_1.3-2
## [64] lintr_1.0.3        desc_1.2.0         stringi_1.2.4
## [67] whoami_1.2.0       Rcpp_1.0.0         xfun_0.4
```

References

Ahsen, N. von, Wittwer, C. T., and Schütz, E. (2001). Oligonucleotide melting temperatures under PCR conditions: Nearest-neighbor corrections for Mg2+, deoxynucleotide triphosphate, and dimethyl sulfoxide concentrations with comparison to alternative empirical formulas. *Clinical Chemistry* 47, 1956–1961. Available at: http://clinchem.aaccjnls.org/content/47/11/1956.

Allawi, H. T., and SantaLucia, J. (1997). Thermodynamics and NMR of internal G · T mismatches in dna.

Biochemistry 36, 10581–10594. doi:10.1021/bi962590c.

Allawi, H. T., and SantaLucia, J. (1998a). Nearest neighbor thermodynamic parameters for internal G · A mismatches in DNA. *Biochemistry* 37, 2170–2179. doi:10.1021/bi9724873.

Allawi, H. T., and SantaLucia, J. (1998b). Nearest-neighbor thermodynamics of internal A · C mismatches in dna: Sequence dependence and pH effects. *Biochemistry* 37, 9435–9444. doi:10.1021/bi9803729.

Allawi, H. T., and SantaLucia, J. (1998c). Thermodynamics of internal C · T mismatches in DNA. *Nucleic Acids Research* 26, 2694–2701. doi:10.1093/nar/26.11.2694.

Asanuma, H., Matsunaga, D., and Komiyama, M. (2005). Clear-cut photo-regulation of the formation and dissociation of the DNA duplex by modified oligonucleotide involving multiple azobenzenes. *Nucleic Acids Symposium Series*, 35–36. doi:10.1093/nass/49.1.35.

Badhwar, J., Karri, S., Cass, C. K., Wunderlich, E. L., and Znosko, B. M. (2007). Thermodynamic characterization of RNA duplexes containing naturally occurring 1 × 2 nucleotide internal loops. *Biochemistry* 46, 14715–14724. doi:10.1021/bi701024w.

Blake, R. D. (1996). "Denaturation of DNA," in *Encyclopedia of molecular biology and molecular medicine*, ed. R. A. Meyers (Weinheim, Germany: VCH Verlagsgesellschaft), 1–19.

Blake, R. D., and Delcourt, S. G. (1998). Thermal stability of DNA. *Nucleic Acids Research* 26, 3323–3332. doi:10.1093/nar/26.14.3323.

Blose, J. M., Manni, M. L., Klapec, K. A., Stranger-Jones, Y., Zyra, A. C., Sim, V., et al. (2007). Non-nearest-neighbor dependence of stability for RNA bulge loops based on the complete set of group i single nucleotide bulge loops. *Biochemistry* 46, 15123–15135. doi:10.1021/bi700736f.

Bommarito, S., Peyret, N., and SantaLucia, J. (2000). Thermodynamic parameters for DNA sequences with dangling ends. *Nucleic Acids Research* 28, 1929–1934. doi:10.1093/nar/28.9.1929.

Breslauer, K. J., Frank, R., Blöcker, H., and Marky, L. A. (1986). Predicting DNA duplex stability from the base sequence. *Proceedings of the National Academy of Sciences* 83, 3746. doi:10.1073/pnas.83.11.3746.

Britten, R. J., Graham, D. E., and Neufeld, B. R. (1974). Analysis of repeating DNA sequences by reassociation. *Methods in Enzymology* 29, 363–418. doi:10.1016/0076-6879(74)29033-5.

Broda, M., Kierzek, E., Gdaniec, Z., Kulinski, T., and Kierzek, R. (2005). Thermodynamic stability of RNA structures formed by CNG trinucleotide repeats. Implication for prediction of RNA structure. *Biochemistry* 44, 10873–10882. doi:10.1021/bi0502339.

Casey, J., and Davidson, N. (1977). Rates of formation and thermal stabilities of RNA:DNA and DNA:DNA duplexes at high concentrations of formamide. *Nucleic Acids Research* 4, 1539–1552. doi:10.1093/nar/4.5.1539.

Chen, J. L., Dishler, A. L., Kennedy, S. D., Yildirim, I., Liu, B., Turner, D. H., et al. (2012). Testing the nearest neighbor model for canonical rna base pairs: Revision of GU parameters. *Biochemistry* 51, 3508–3522. doi:10.1021/bi3002709.

Chester, N., and Marshak, D. (1993). Dimethyl sulfoxide-mediated primer Tm reduction: A method for analyzing the role of renaturation temperature in the polymerase chain reaction. *Analytical Biochemistry* 209, 284–290. doi:10.1006/abio.1993.1121.

Cullen, B. R., and Bick, M. D. (1976). Thermal denaturation of DNA from bromodeoxyuridine substituted cells. *Nucleic Acids Research* 3, 49–62. doi:10.1093/nar/3.1.49.

Davis, A. R., and Znosko, B. M. (2007). Thermodynamic characterization of single mismatches found in naturally occurring RNA. *Biochemistry* 46, 13425–13436. doi:10.1021/bi701311c.

Davis, A. R., and Znosko, B. M. (2008). Thermodynamic characterization of naturally occurring RNA single mismatches with G-U nearest neighbors. *Biochemistry* 47, 10178–10187. doi:10.1021/bi800471z.

Dumousseau, M., Rodriguez, N., Juty, N., and Le Novère, N. (2012). MELTING, a flexible platform to predict the melting temperatures of nucleic acids. *BMC Bioinformatics* 13, 101. doi:10.1186/1471-2105-13-101.

Escara, J. F., and Hutton, J. R. (1980). Thermal stability and renaturation of DNA in dimethyl sulfoxide solutions: Acceleration of the renaturation rate. *Biopolymers* 19, 1315–1327. doi:10.1002/bip.1980.360190708.

Frank-Kamenetskii, M. D. (1971). Simplification of the empirical relationship between melting temperature of DNA, its GC content and concentration of sodium ions in solution. *Biopolymers* 10, 2623–2624. doi:10.1002/bip.360101223.

Freier, S. M., Kierzek, R., Jaeger, J. A., Sugimoto, N., Caruthers, M. H., Neilson, T., et al. (1986). Improved free-energy parameters for predictions of RNA duplex stability. *Proceedings of the National Academy of Sciences* 83, 9373. doi:10.1073/pnas.83.24.9373.

Hall, T. J., Grula, J. W., Davidson, E. H., and Britten, R. J. (1980). Evolution of sea urchin non-repetitive DNA. *Journal of Molecular Evolution* 16, 95–110. doi:10.1007/BF01731580.

Hutton, J. R. (1977). Renaturation kinetics and thermal stability of DNA in aqueous solutions of formamide and urea. *Nucleic Acids Research* 4, 3537–3555. doi:10.1093/nar/4.10.3537.

Kawakami, J., Kamiya, H., Yasuda, K., Fujiki, H., Kasai, H., and Sugimoto, N. (2001). Thermodynamic stability of base pairs between 2-hydroxyadenine and incoming nucleotides as a determinant of nucleotide incorporation specificity during replication. *Nucleic Acids Research* 29, 3289–3296. doi:10.1093/nar/29.16.3289.

Kierzek, E., Mathews, D. H., Ciesielska, A., Turner, D. H., and Kierzek, R. (2006). Nearest neighbor parameters for Watson-Crick complementary heteroduplexes formed between 2'-O-methyl RNA and RNA oligonucleotides. *Nucleic Acids Research* 34, 3609–3614. doi:10.1093/nar/gkl232.

Le Novère, N. (2001). MELTING, computing the melting temperature of nucleic acid duplex. *Bioinformatics* 17, 1226–1227. doi:10.1093/bioinformatics/17.12.1226.

Lu, Z. J., Turner, D. H., and Mathews, D. H. (2006). A set of nearest neighbor parameters for predicting the enthalpy change of RNA secondary structure formation. *Nucleic Acids Research* 34, 4912–4924. doi:10.1093/nar/gkl472.

Marmur, J., and Doty, P. (1962). Determination of the base composition of deoxyribonucleic acid from its thermal denaturation temperature. *Journal of Molecular Biology* 5, 109–118. doi:10.1016/S0022-2836(62)80066-7.

Mathews, D. H., Sabina, J., Zuker, M., and Turner, D. H. (1999). Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *Journal of Molecular Biology* 288, 911–940. doi:10.1006/jmbi.1999.2700.

McConaughy, B. L., Laird, C., and McCarthy, B. J. (1969). Nucleic acid reassociation in formamide. *Biochemistry* 8, 3289–3295. doi:10.1021/bi00836a024.

McTigue, P. M., Peterson, R. J., and Kahn, J. D. (2004). Sequence-dependent thermodynamic parameters for locked nucleic acid (LNA)-DNA duplex formation. *Biochemistry* 43, 5388–5405. doi:10.1021/bi035976d.

Miller, S., Jones, L. E., Giovannitti, K., Piper, D., and Serra, M. J. (2008). Thermodynamic analysis of 5' and 3' single- and 3' double-nucleotide overhangs neighboring wobble terminal base pairs. *Nucleic Acids Research* 36, 5652–5659. doi:10.1093/nar/gkn525.

Mitsuhashi, M. (1996). Technical report: Part 1. Basic requirements for designing optimal oligonucleotide probe sequences. *Journal of Clinical Laboratory Analysis* 10, 277–284. doi:10/cw9bn6.

Musielski, H., Mann, W., Laue, R., and Michel, S. (1981). Influence of dimethylsulfoxide on transcription by bacteriophage T3-induced RNA polymerase. *Zeitschrift für allgemeine Mikrobiologie* 21, 447–456. doi:10.1002/jobm.19810210606.

Ohmichi, T., Nakano, S.-i., Miyoshi, D., and Sugimoto, N. (2002). Long RNA dangling end has large energetic contribution to duplex stability. *Journal of the American Chemical Society* 124, 10367–10372. doi:10.1021/ja0255406.

O'Toole, A. S., Miller, S., Haines, N., Zink, M. C., and Serra, M. J. (2006). Comprehensive thermodynamic analysis of 3' double-nucleotide overhangs neighboring Watson-Crick terminal base pairs. *Nucleic Acids Research* 34, 3338–3344. doi:10.1093/nar/gkl428.

O'Toole, A. S., Miller, S., and Serra, M. J. (2005). Stability of 3' double nucleotide overhangs that model the 3' ends of siRNA. *RNA* 11, 512–516. doi:10.1261/rna.7254905.

Owczarzy, R., Moreira, B. G., You, Y., Behlke, M. A., and Walder, J. A. (2008). Predicting stability of DNA duplexes in solutions containing magnesium and monovalent cations. *Biochemistry* 47, 5336–5353. doi:10.1021/bi702363u.

Owczarzy, R., You, Y., Moreira, B. G., Manthey, J. A., Huang, L., Behlke, M. A., et al. (2004). Effects of sodium ions on DNA duplex oligomers: Improved predictions of melting temperatures. *Biochemistry* 43, 3537–3554. doi:10.1021/bi034621r.

Owen, R., Hill, L., and Lapage, S. (1969). Determination of DNA base compositions from melting profiles in dilute buffers. *Biopolymers* 7, 503–516. doi:10.1002/bip.1969.360070408.

Peyret, N. (2000). Prediction of nucleic acid hybridization: Parameters and algorithms. Available at: http://elibrary.wayne.edu/record=2760965.

Peyret, N., Seneviratne, P. A., Allawi, H. T., and SantaLucia, J. (1999). Nearest-Neighbor Thermodynamics and NMR of DNA Sequences with Internal A · A, C · C, G · G, and T · T Mismatches. *Biochemistry* 38, 3468–3477. doi:10.1021/bi9825091.

Record, M. T. (1967). Electrostatic effects on polynucleotide transitions. I. Behavior at neutral pH. *Biopolymers* 5, 975–992. doi:10.1002/bip.1967.360051010.

SantaLucia, J. (1998). A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics. *Proceedings of the National Academy of Sciences* 95, 1460. doi:10.1073/pnas.95.4.1460.

SantaLucia, J., and Hicks, D. (2004). The thermodynamics of DNA structural motifs. *Annual Review of Biophysics and Biomolecular Structure* 33, 415–440. doi:10.1146/annurev.biophys.32.110601.141800.

SantaLucia, John, Allawi, H. T., and Seneviratne, P. A. (1996). Improved nearest-neighbor parameters for predicting DNA duplex stability. *Biochemistry* 35, 3555–3562. doi:10.1021/bi951907q.

Schildkraut, C., and Lifson, S. (1965). Dependence of the melting temperature of DNA on salt concentration. *Biopolymers* 3, 195–208. doi:10.1002/bip.360030207.

Sugimoto, N., Nakano, S., Yoneyama, M., and Honda, K. (1996). Improved thermodynamic parameters and helix initiation factor to predict stability of DNA duplexes. *Nucleic Acids Research* 24, 4501–4505. doi:10.1093/nar/24.22.4501.

Tan, Z.-J., and Chen, S.-J. (2006). Nucleic acid helix stability: Effects of salt concentration, cation valence and size, and chain length. *Biophysical Journal* 90, 1175–1190. doi:10.1529/biophysj.105.070904.

Tan, Z.-J., and Chen, S.-J. (2007). RNA helix stability in mixed Na(+)/Mg(2+) solution. *Biophysical Journal* 92, 3615–3632. doi:10.1529/biophysj.106.100388.

Tanaka, F., Kameda, A., Yamamoto, M., and Ohuchi, A. (2004). Thermodynamic parameters based on a nearest-neighbor model for DNA sequences with a single-bulge loop. *Biochemistry* 43, 7143–7150. doi:10.1021/bi036188r.

Wahl, G. M., Barger, S. L., and Kimmel, A. R. (1987). Molecular hybridization of immobilized nucleic acids: Theoretical concepts and practical considerations. *Methods in Enzymology* 152, 399–407. doi:10.1016/0076-6879(87)52046-8.

Watkins, N. E., Kennelly, W. J., Tsay, M. J., Tuin, A., Swenson, L., Lee, H.-R., et al. (2011). Thermodynamic contributions of single internal rA · dA, rC · dC, rG · dG and rU · dT mismatches in RNA/DNA duplexes. *Nucleic Acids Research* 39, 1894–1902. doi:10/cdm4jh.

Watkins, N. E., and SantaLucia, J. (2005). Nearest-neighbor thermodynamics of deoxyinosine pairs in DNA duplexes. *Nucleic Acids Research* 33, 6258–6267. doi:10.1093/nar/gki918.

Wetmur, J. G. (1991). DNA probes: Applications of the principles of nucleic acid hybridization. *Critical Reviews in Biochemistry and Molecular Biology* 26, 227–259. doi:10.3109/10409239109114069.

Wright, D. J., Rice, J. L., Yanker, D. M., and Znosko, B. M. (2007). Nearest neighbor parameters for inosine · uridine pairs in RNA duplexes. *Biochemistry* 46, 4625–4634. doi:10.1021/bi0616910.

Xia, T., SantaLucia, J., Burkard, M. E., Kierzek, R., Schroeder, S. J., Jiao, X., et al. (1998). Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with Watson-Crick base pairs. *Biochemistry* 37, 14719–14735. doi:10.1021/bi9809425.