2013年度 S-PLUS学生研究奨励賞

モンテカルロ法による
次世代シークエンサーの配列データの品質評価

東京農工大学大学院　農学府　生物制御科学専攻
植物病理学研究室所属
佐藤暁

全体の流れ

ゲノム

ゲノムとは　→　生物の設計図。
DNA中の4種類の塩基（A・T・G・C）の並び方で表される。

ATTAGGCAACCCGGGTGCATAG・・・

ヒトの場合には、全部で30億文字に及ぶ

ゲノム解析では、これらの文字の並び順を決定し解析する

次世代シークエンサーによるゲノム解析

GenomeAnalyzerIIx
（イルミナ社）

次世代シークエンサーとは、生物のゲノムを高速で決定できる実験機器である。

数十～数百塩基のDNA断片が
数十万～数億断片産生される

得られたデータはLinuxをベースとしたフリーソフトフェア等を使用し、解析する。

次世代シークエンサーから出力される配列データ

次世代シークエンサーで解読された配列はFastq形式で出力される。

●Fastqファイル

```
@HWI-ST977:153:D0J0KACXX:4:1101:1636:1851 1:N:0:TAGCTT
NGGTCCGGCTTTGAACCCCTGACAGGAAGGTATTATGCTGATCACGATGC
AACATGACAGATCGGCTCATGAAGCTTGGACTTGCTGTTCTCCTCTTTACG
+
#4=BDFFFGHHHHHJJIJJIJHIJJJGGIIFGIIIIJJIIJJJJJJIJJJJJJJJGIJHGHEFF
FDDCCEC@ACACDDDDCCCDDAACCDDCACDCDCC?
@HWI-ST977:153:D0J0KACXX:4:1101:1705:1877 1:N:0:TAGCTT
NTATCTTGACAGATTTTCTAGACTCATCCCAAGTTCTTGACCTAGCGCTGA
CAGAATTTGCTAAAATATGCTTATTCCGGTGCCAACTCCGTGGTATGCCA
+
#1=DFFFFHHHHHJJJJJJJJJJJJJJJJJJIIEGHIHGGIJJJIIJJIJJJIJIIGJJJJIJ
GIJJJJIJJIFHEFHGFFDDEEDDDDDDDDDDDDDDD
```

4行で1組

配列名

DNA配列

記号

クオリティデータ

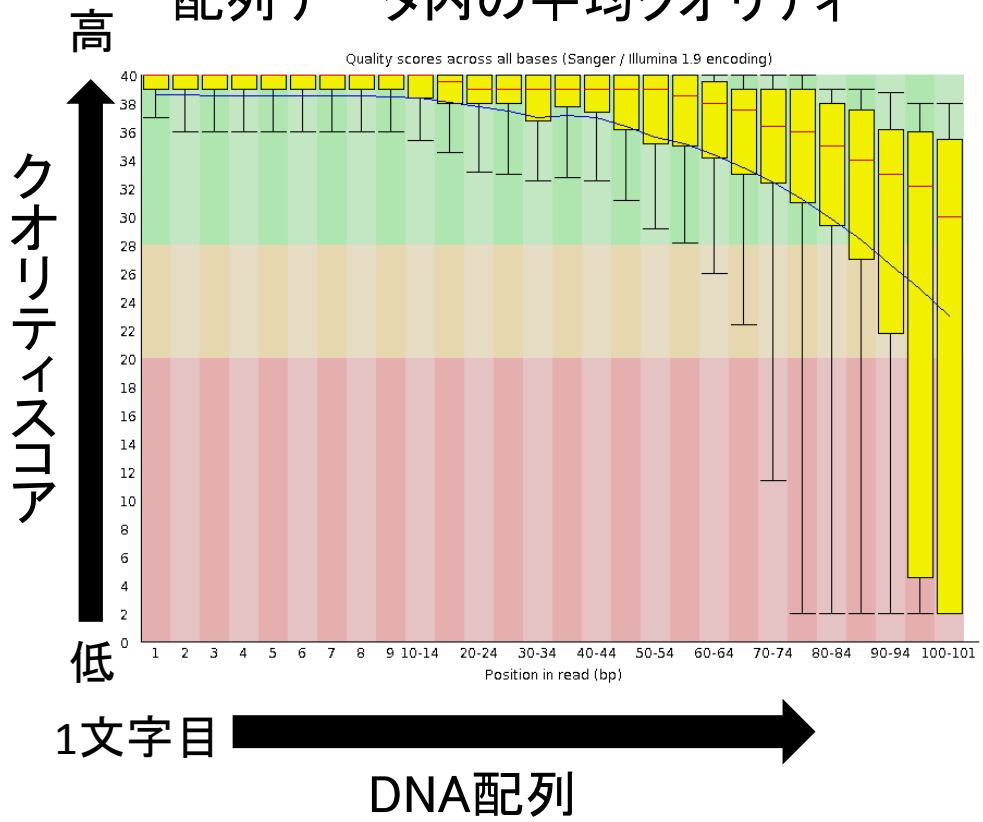
4行1組の配列データが並び、全体で数百万～数億行に及ぶ

配列データのクオリティチェック

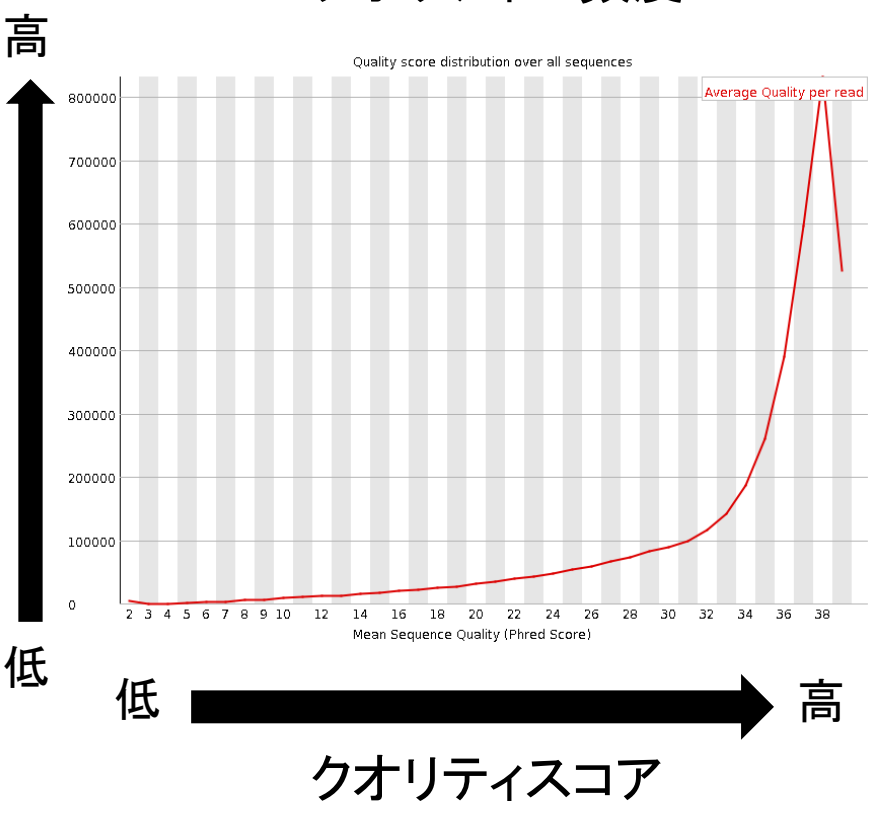
配列の解析を行う前に、解読した配列データの信頼性を確かめる作業（クオリティチェック）が必要となる。クオリティチェックを行うフリーソフトフェアとしてFastQCがある。

)クオリティチェックの例

配列データ内の平均クオリティ

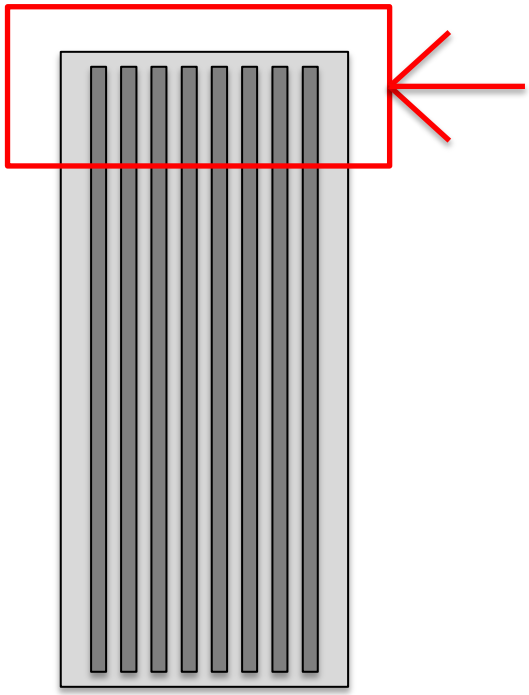


クオリティの頻度



クオリティチェックの問題点

GAⅡxの場合には、フローセルというガラス板上でDNA反応クラスタを作り、反応させることでDNA配列を解読する。



フローセルの図（上から見た図）

FastQCでは、最初の20万個をチェックしている。

しかし、反応が正確に進むとは限らない。

（1）サンプル濃度の間違い

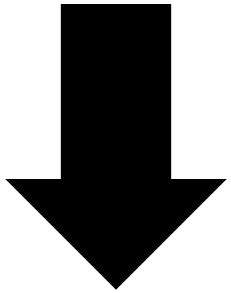
（2）試薬濃度の間違い

（3）操作の荒さ

などの原因によりクオリティが悪化することがある。

研究目的

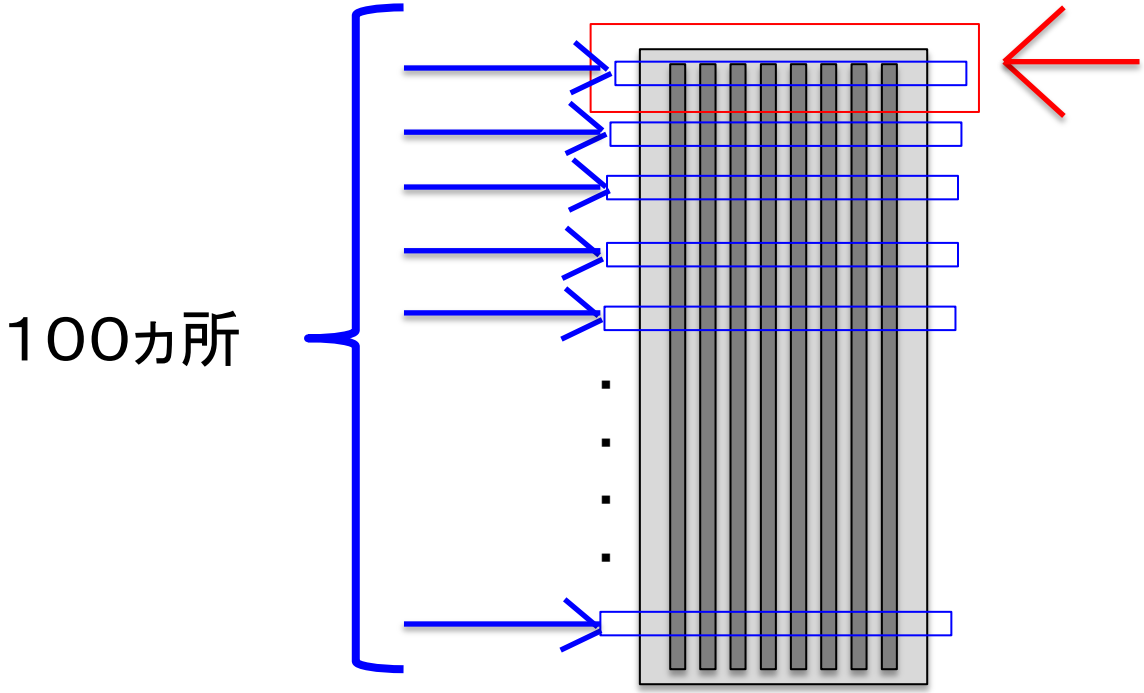
FastQCでは各ファイルの最初の20万個のクオリティデータを使用して計算を行う



配列データが数千万個に及ぶ場合には、クオリティデータ全体を正確に把握することが困難

ファイル全体のクオリティデータを評価する方法を確立する

モンテカルロ法によるクオリティチェック①



FastQC
最初の20万個をチェック

モンテカルロ法の「層化無作為抽出法」を適用

全体を100等分し、100カ所から1000個ずつ配列を抽出

モンテカルロ法によるクオリティチェック②

S-PLUSを使用し一様乱数を発生させる

↓

乱数に基づき、Fastqファイルから無作為にクオリティデータを抽出

↓

クオリティデータを数値データに変換

↓

全体のクオリティを評価

S-PLUSによる乱数の抽出

使用サンプル：L197（Read1：15057415 reads、Read2：15057415 reads)

今回はサンプルを2回読む方法で行い、1回目（Read1）は成功、
2回目（Read2）は失敗した。

15057415read

↓

100区間に等分

1 ～ 150574、150575～311148、・・・・、14906872～15057400

各区間において、関数runif()を使用して、それぞれ1000個の一様乱数を発生させる。
さらに、発生させた100区間分の乱数のデータをcbind()によっての結合し、一つのファイルとしてまとめた。

乱数によるクオリティデータの抽出

●Fastqファイル

```
@HWI-ST977:153:D0J0KACXX:4:1101:1636:1851 1:N:0:TAGCTT
NGGTCCGGCTTTGAACCCCTGACAGGAAGGTATTATGCTGATCACGATGC
AACATGACAGATCGGCTCATGAAGCTTGGACTTGCTGTTCTCCTCTTTACG
+
#4=BDFFFGHHHHHJJIJJIJHIJJJGGIIFGIIIIJJIIJJJJJJIJJJJJJJJGIJHGHEFF
FDDCCEC@ACACDDDDCCCDDAACCDDCACDCDCC?
@HWI-ST977:153:D0J0KACXX:4:1101:1705:1877 1:N:0:TAGCTT
NTATCTTGACAGATTTTCTAGACTCATCCCAAGTTCTTGACCTAGCGCTGA
CAGAATTTGCTAAAATATGCTTATTCCGGTGCCAACTCCGTGGTATGCCA
+
#1=DFFFFHHHHHJJJJJJJJJJJJJJJJJJIIEGHIHGGIJJJIIJJIJJJIJIIGJJJJIJ
GIJJJJIJJIFHEFHGFFDDEEDDDDDDDDDDDDDDD
```

4行で1組

リード名

DNA配列

記号

クオリティデータ

→ Fastqからクオリティデータのみを抽出する

→ 抽出した乱数に従ってクオリティデータを抽出

100区間で1000個のクオリティデータを、無作為に抽出

クオリティデータの数値変換

無作為に抽出したデータをperlのスクリプトを使用して数値データに変換

○クオリティデータ　→　0～40までの数字に対応した文字で表される。

#4=BDFFFGHHHHHJJIJJIJHIJJJGGIIFGIIIIJJIIJJJJJJIJJJJJJJJGIJHGHEFF
FDDCCEC@ACACDDDDCCCDDAACCDDCACDCDCC?

クオリティデータの変換表

Table 1　ASCII Characters Encoding Q-scores 0–40

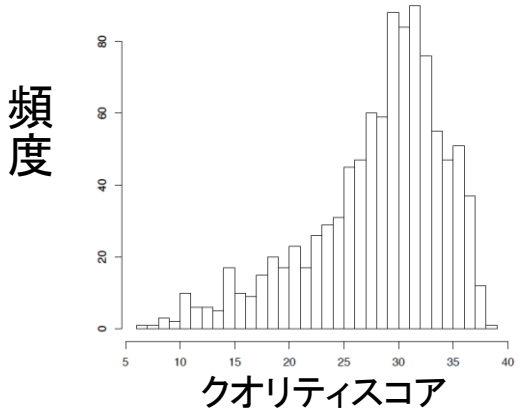
Symbol	ASCII Code	Q-Score	Symbol	ASCII Code	Q-Score	Symbol	ASCII Code	Q-Score
!	33	0	/	47	14	=	61	28
"	34	1	0	48	15	>	62	29
#	35	2	1	49	16	?	63	30
$	36	3	2	50	17	@	64	31
%	37	4	3	51	18	A	65	32
&	38	5	4	52	19	B	66	33
'	39	6	5	53	20	C	67	34
(	40	7	6	54	21	D	68	35
)	41	8	7	55	22	E	69	36
*	42	9	8	56	23	F	70	37
+	43	10	9	57	24	G	71	38
,	44	11	:	58	25	H	72	39
-	45	12	;	59	26	I	73	40
.	46	13	<	60	27			

得られたクオリティの数値データを基にクオリティデータを評価を行った。

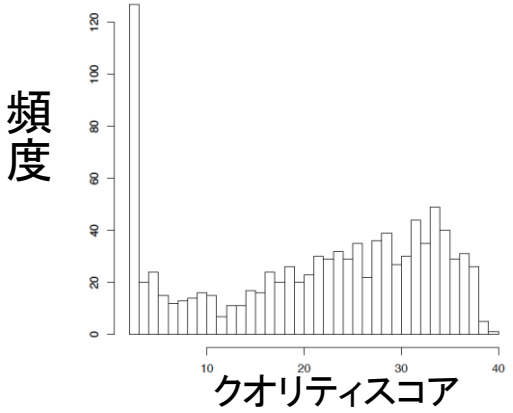
クオリティの頻度の評価①

関数hist()を使用して図を作製した。

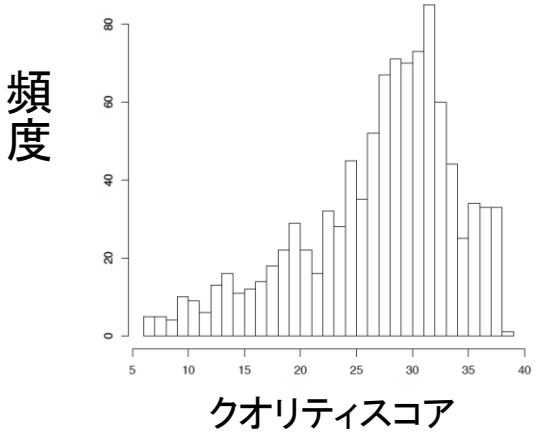
Read1の1番目の区間（1～150574）



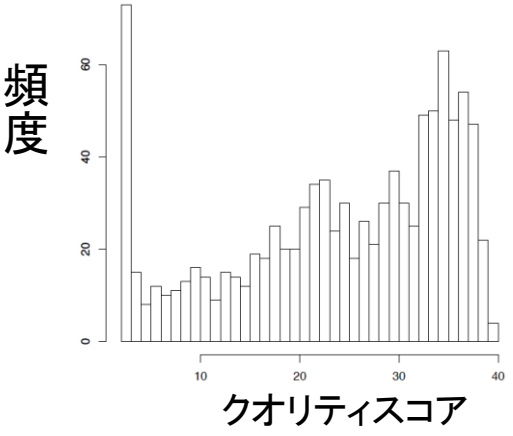
Read2の1番目の区間（1～150574）



ReadR1の41番目の区間（12045921 ～12196494 ）



Read2の41番目の区間（12045921 ～12196494 ）

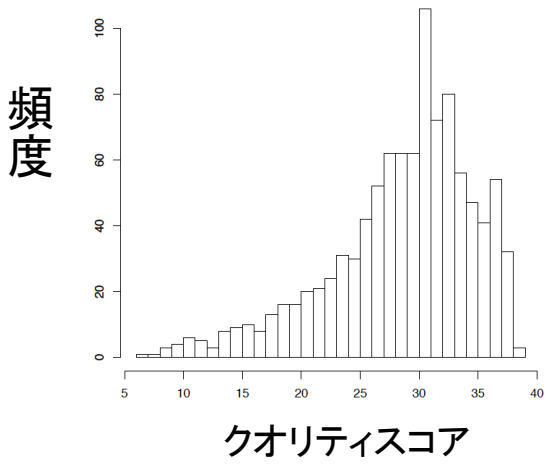


Read2では、区間によってクオリティが異なることが示唆された

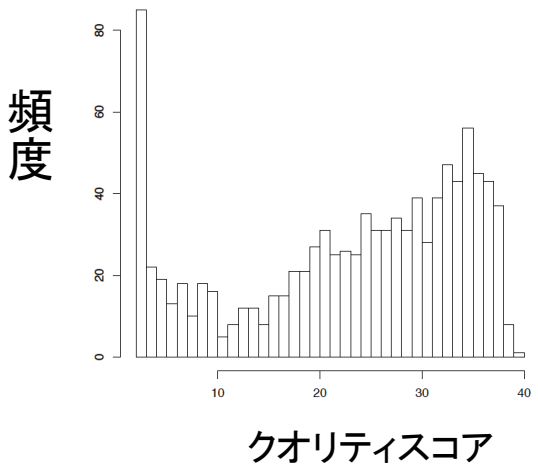
クオリティの頻度の評価②

Rの関数density()を使用して図を作製した。

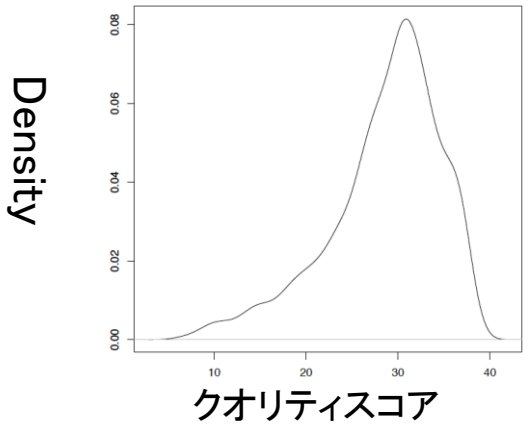
Read1の41番目の区間（12045921 ～12196494 ）



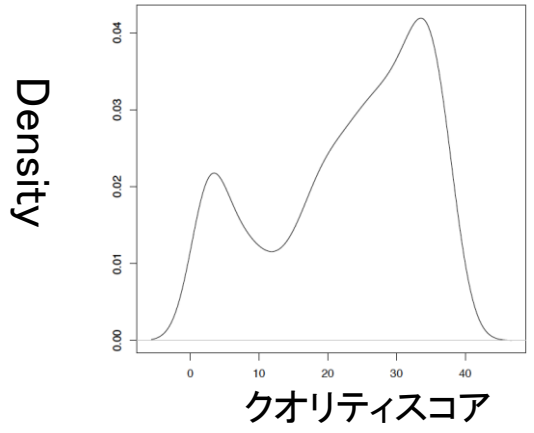
Read2の41番目の区間（12045921 ～12196494 ）



Read1の41番目の区間（12045921 ～12196494 ）



Read2の41番目の区間（12045921 ～12196494 ）



Read2ではクオリティの高い配列と低い配列の２つのピークがあ

ｚ

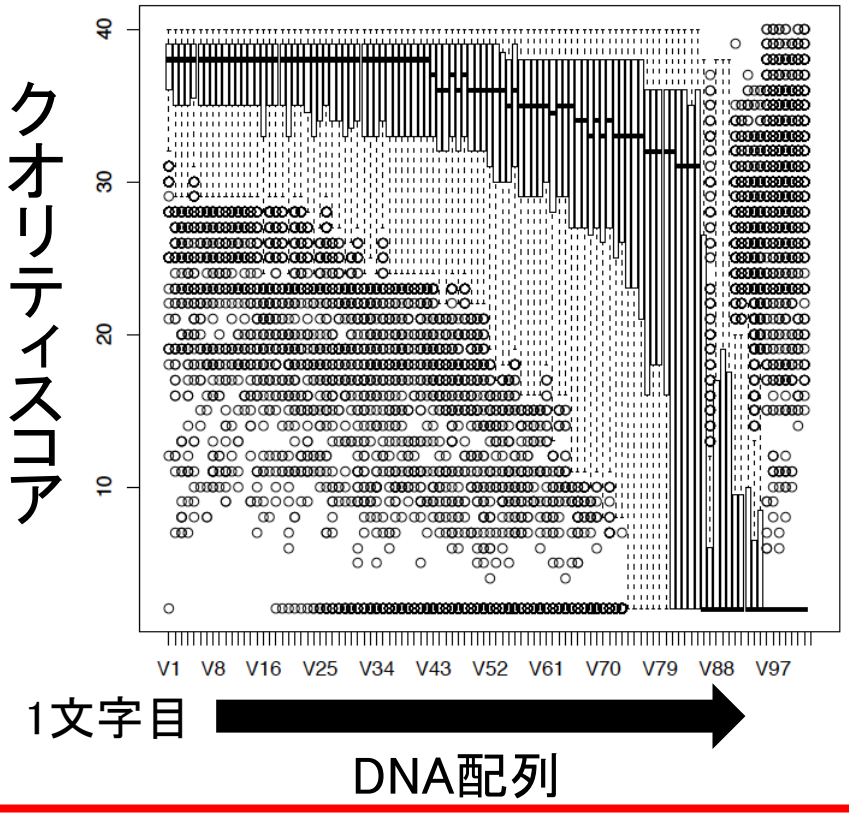
DNA配列のクオリティの評価

Rの関数boxplot()を使用して図を作製した。

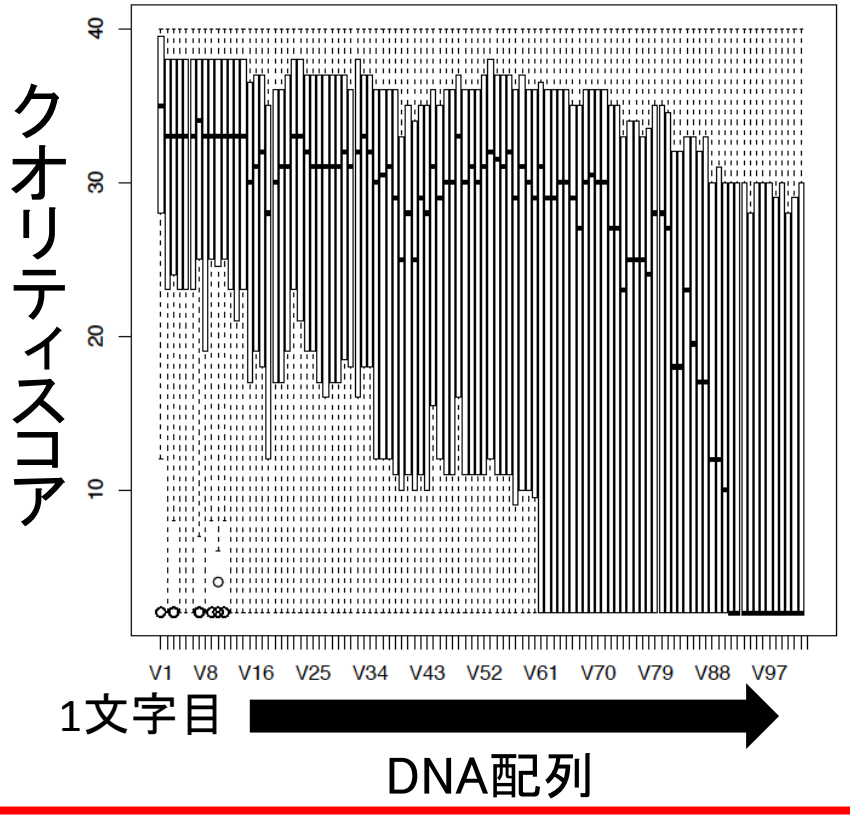
DNA配列

NTATCTTGACAGATTTTCTAGACTCATCCCAAGTTCTTGACCTAGCGCTGACAG
AATTTGCTAAAATATGCTTATTCCGGTGCCAACTCCGTGGTATGCCA

Read1の41番目の区間（12045921 ～12196494 ）

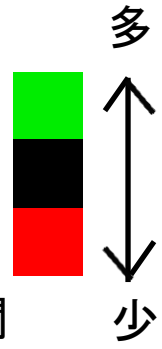


Read2の41番目の区間（12045921 ～12196494 ）



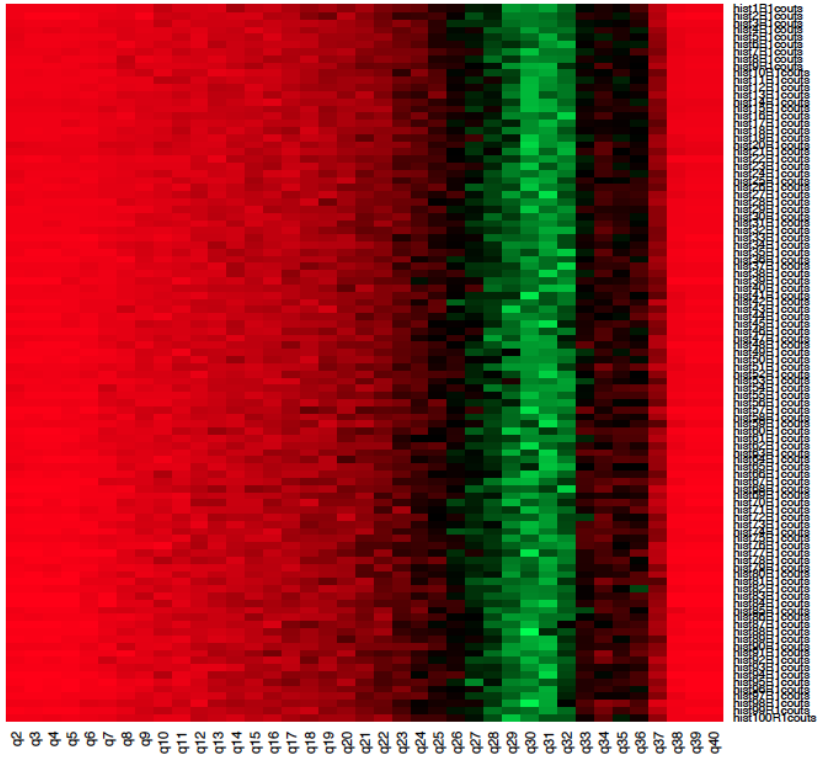
Read2では、Read1に比べてクオリティの低い配列が多く含まれることが把握できた。

データ全体のクオリティの評価

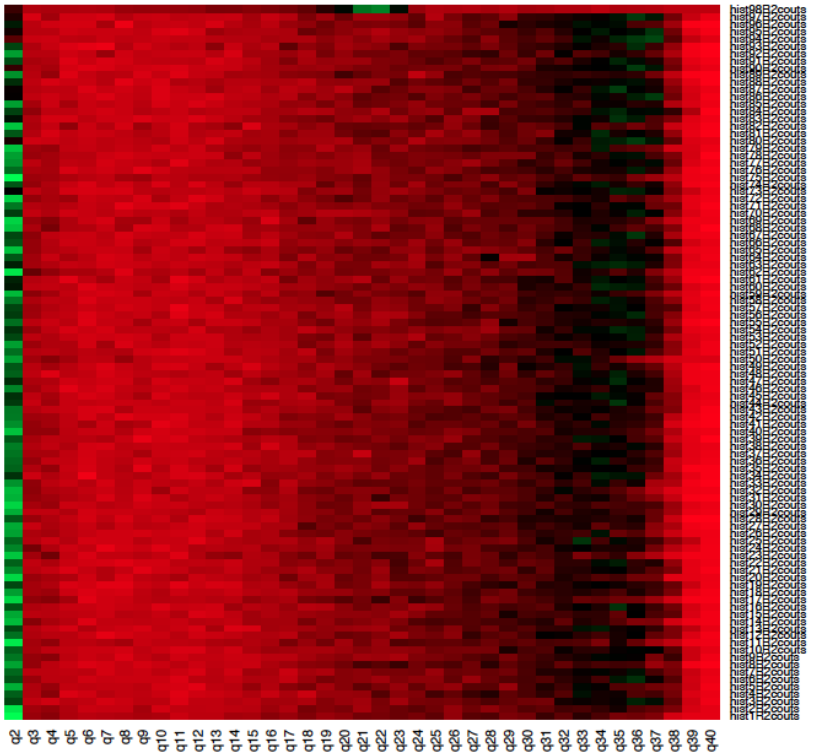


Rの関数heatmap()を使用して図を作製した。

Read1

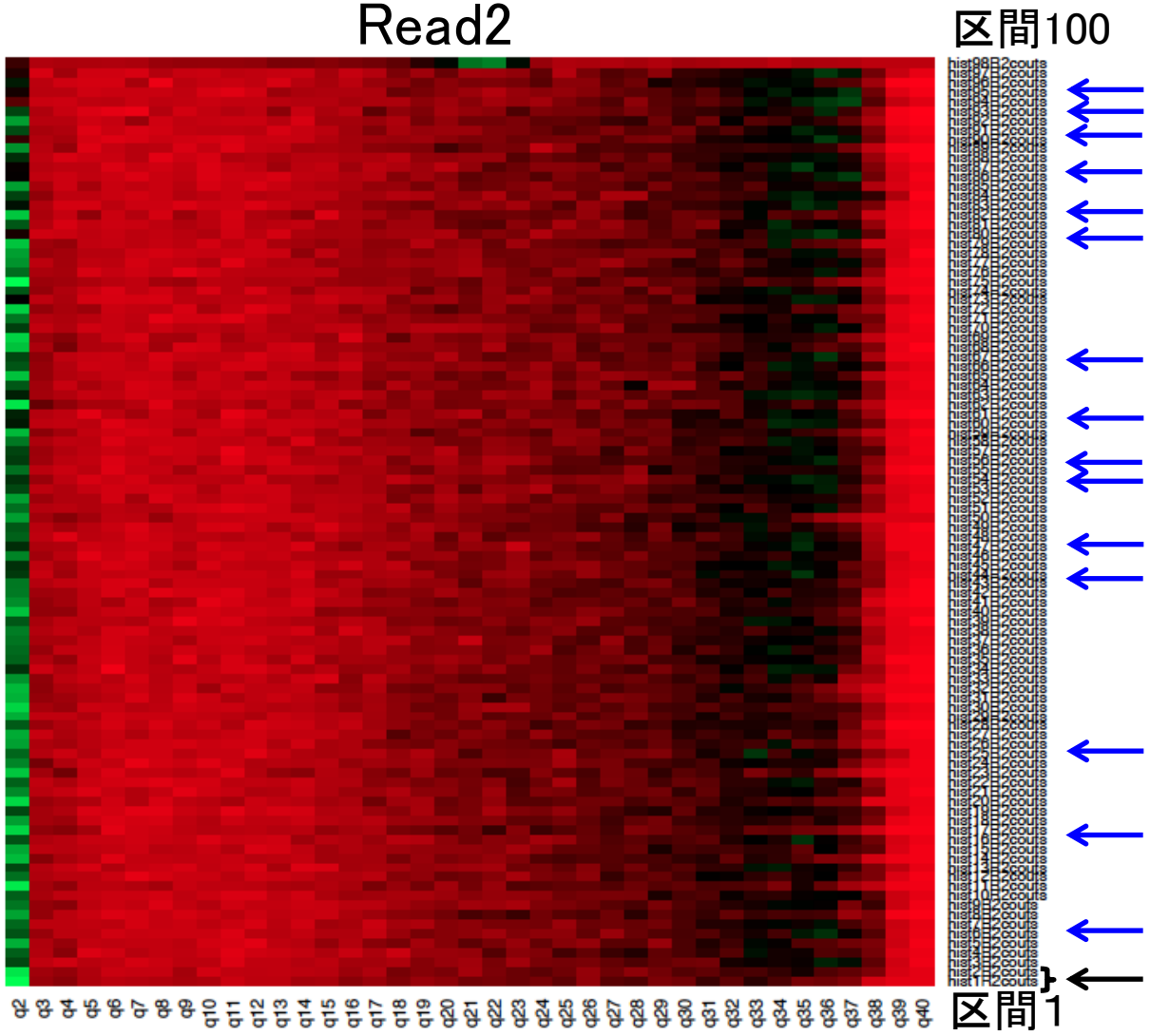


Read2



Read1に比べて、Read2では全体的にクオリティが低いことが把握できた。しかし、Read2の中にも一部、クオリティの高い区間が存在した。

データ全体のクオリティの評価



多

少

←：クオリティの高い区間

← 最初の20万個

矢印で示された区間では、クオリティの高いリードが含まれている

まとめ

- モンテカルロ法による次世代シークエンサーの配列データの品質評価方法を確立した。
- 配列全体からクオリティデータを抽出し、ヒートマップなどで評価することで、データ全体のクオリティを評価することが可能になった。
- クオリティが低い配列データの場合には、データ全体を評価した方が正確にクオリティを把握できる可能性が示唆された。

References

1）石井一夫,佐藤暁,古崎利紀,有江力,寺岡徹(2013).
ゲノム科学におけるビッグデータ・データマイニング,
日本統計学会誌　第43巻　第1号　99頁～111頁

2）Maria L. Rizzo(著),石井一夫(共訳),村田真樹(共訳)(2011).
『R による計算機統計学』(オーム社)