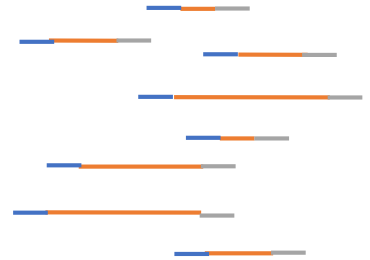
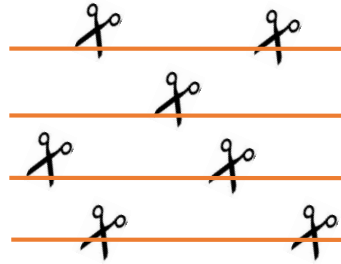


NGS data formats and variant calling

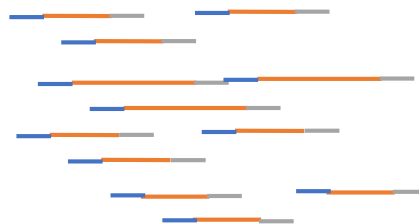
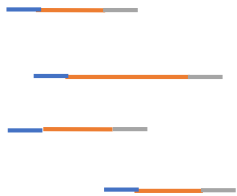
2019 Dragon Star Bioinformatics Course (Day 1)

Sample Preparation



Random shearing of
the DNA

Adding adaptors
and barcodes



Size selection

Amplification

Sequencing

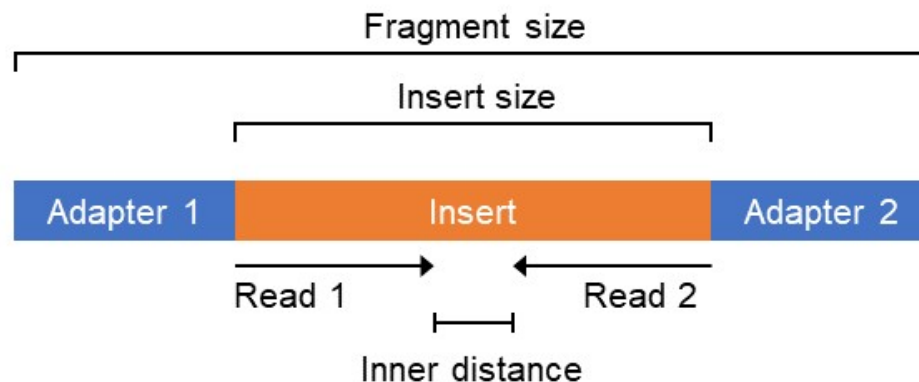
Basic Concepts in NGS

Insert – the DNA portion that is used for sequencing

Read – the part of the insert that is sequenced

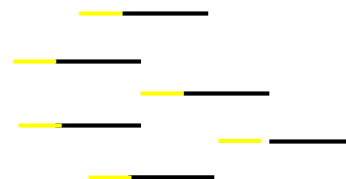
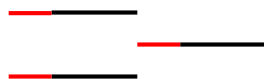
Single End – a sequencing procedure by which the insert is sequenced from one end only

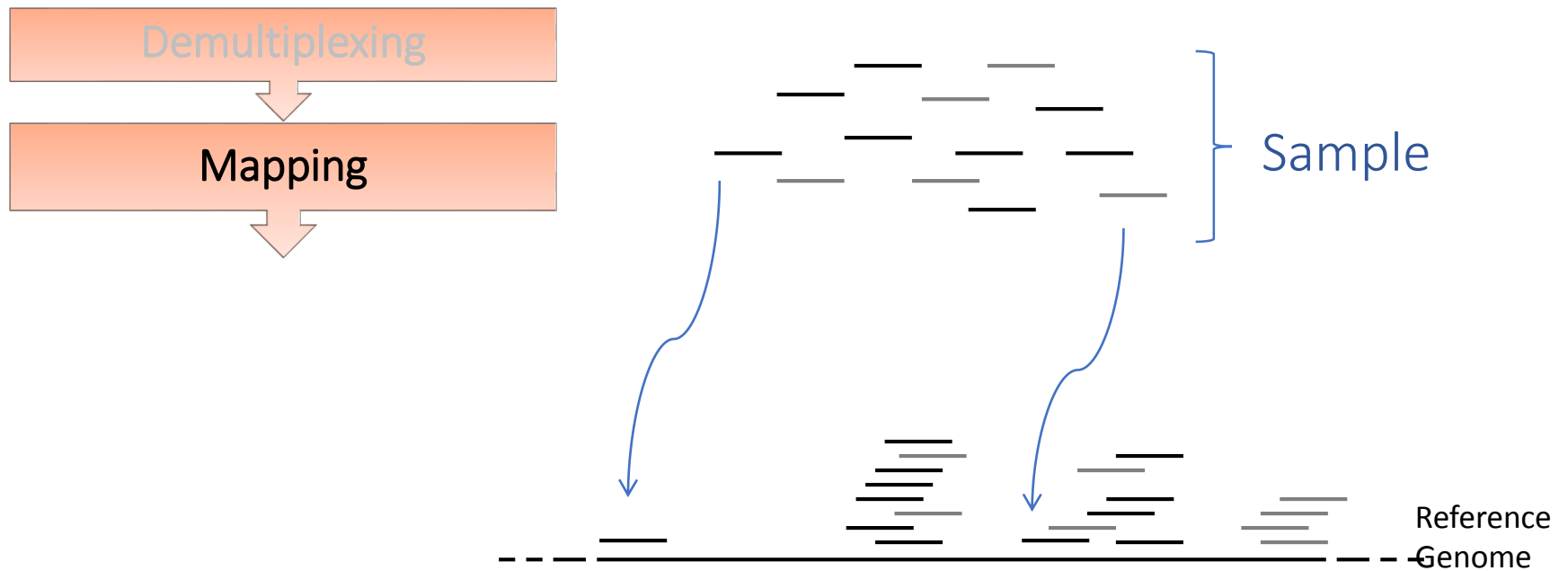
Paired End – a sequencing procedure by which the insert is sequenced from both ends



Demultiplexing

Unknown inserts





Example of mapping parameters:

- Number of mismatches per read
- Scores for mismatch or gaps

Mapping parameters affect the rest of the analysis

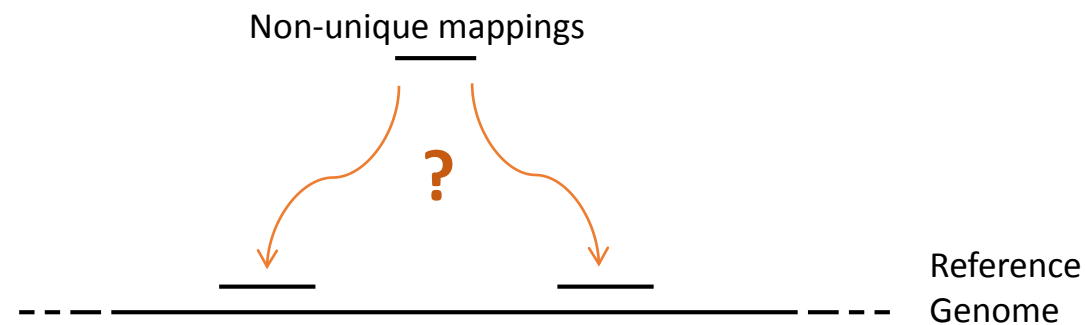
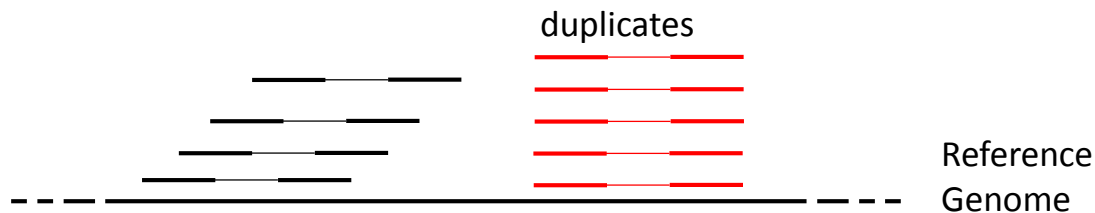
Demultiplexing



Mapping



Removing duplicates and non-unique mappings



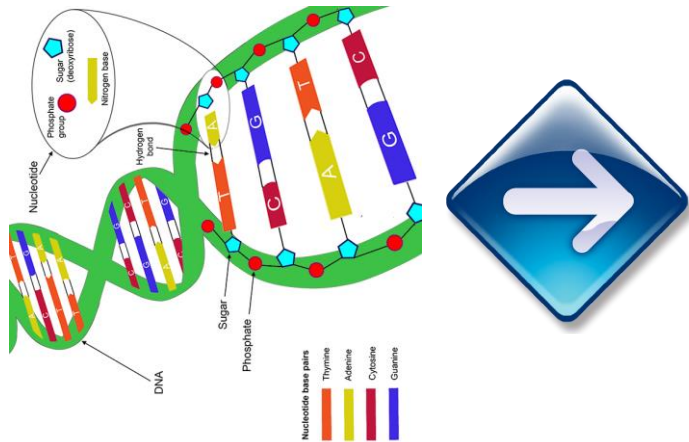
NGS – high-throughput, but

- Shorter reads
 - Sanger sequencing: up to ~1Kb
 - NGS technologies: typically 30-300bp
 - Implication: a lot of computational tasks e.g, assembly, read alignment, haplotyping, detection of SNPs, CNVs, indels etc.
- Higher per-base sequencing error rate
 - Sanger sequencing: < 0.001%
 - NGS: 0.5-1%
 - Implication: Need redundant sequencing of each base to distinguish sequencing errors from true polymorphisms

Now what does NGS data look like?

- What do you want them to look like?

Fantasy Land



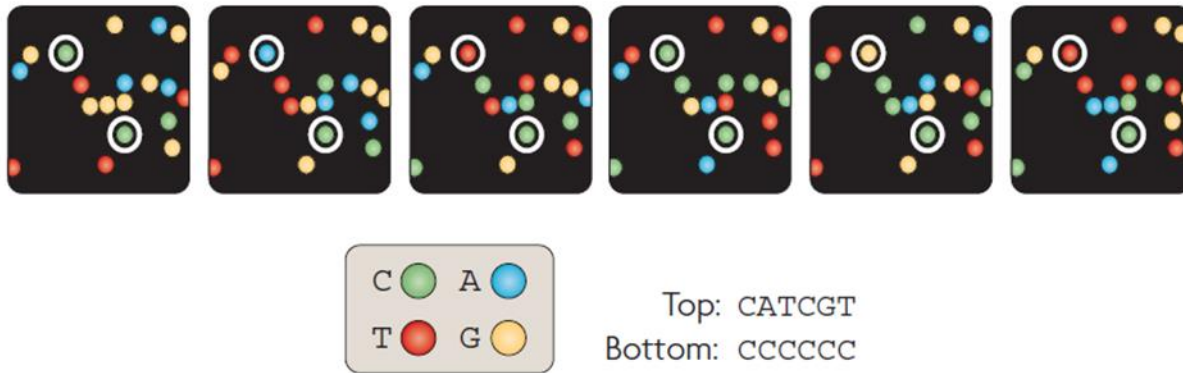
DNA

{ Chr1 haplo1: agttataagat...
{ Chr1 haplo2: agttattagat...
{ Chr2 haplo1: cctagctggat...
{ Chr2 haplo2: ccaagctcgat...
{ Chr3 haplo1: agctctgagcg...
{ Chr3 haplo2: agctctgagcg...
{ Chr4 haplo1: atcgttcgatc...
{ Chr4 haplo2: atcgatcgaac...
etc...

3 billion bases from the
beginning of chromosome 1 to
the end of the last sex
chromosome (2x) in haplotypes

The Rawest of Raw Data

- Typically: images



- The first step is to call nucleotides at each base of each read: **base calling**, which is NOT 100% accurate
 - Typically base calling is done by the sequencer itself, and we start analysis after base calling (for example, fastq format file)

FASTQ: The raw sequence data format

What the data *really* look like: FASTQ

```
@IL27_748:3:286:254:231/1
GTGGAATAATGACCATGACGAAGAGGATGACAGTCC
+
BBBDCDED4DEAECEFEF2DC/>>@&*/C6208'<*
@IL27_748:3:285:138:811/1
AAGTGGATTACTACCTACAGAGAGTCAGTAAGAGAG
+
BB3D2D<D>D7DE0+19242?=57?=4%'6%.2.'(
@IL27_748:3:142:204:780/1
AGAAAAAGAAAGAGAGAGACAGACAGACAGAGAAAG
+
26B3C8<DDD>AAA0FCF7DCA012A?(;2?AC(=/
@IL27_748:3:23:252:759/1
TTTTAGATGAAGTTATTTCTTTACTACCGTAGGCC
+
BB0D;DED>;>CEC:2EFA@69CDC3?@'%=585='
...
```

Millions of short reads from
unknown genetic locations

What the data *really* look like: FASTQ

```
@IL27_748:3:286:254:231/1
GTGGAATAATGACCATGACGAAGAGGATGACAGTCC
+
BBBDCDED4DEAECEFEF2DC/>>@&*/C6208'<*
```

Read 1

```
@IL27_748:3:285:138:811/1
AAGTGGATTACTACCTACAGAGAGTCAGTAAGAGAG
+
BB3D2D<D>D7DE0+19242?=57?=4%'6%.2.'(
```

Read 2

```
@IL27_748:3:142:204:780/1
AGAAAAAGAAAGAGAGAGACAGACAGACAGAGAAAG
+
26B3C8<DDD>AAA0FCF7DCA012A?(;2?AC(=/
```

Read 3

```
@IL27_748:3:23:252:759/1
TTTTAGATGAAGTTATTTCTTTACTACCGTAGGCC
+
BB0D;DED>;>CEC:2EFA@69CDC3?@'%=585='
```

Read 4

...

Millions of short reads from
unknown genetic locations

What the data *really* look like: FASTQ

unique read identifier → @IL27_748:3:286:254:231/1
Bases/nucleotides read → GTGGAATAATGACCATGACGAAGAGGATGACAGTCC
“+” format line → +
per-base quality scores → BBBDCDED4DEAECEFEF2DC/>>@&*/C6208'<*

Read 1

```
@IL27_748:3:285:138:811/1
AAGTGGATTACTACCTACAGAGAGTCAGTAAGAGAG
+
BB3D2D<D>D7DE0+19242?=57?=4%'6%.2.'(
@IL27_748:3:142:204:780/1
AGAAAAAGAAAGAGAGAGACAGACAGACAGAGAAAG
+
26B3C8<DDD>AAA0FCF7DCA012A?(;2?AC(=/
@IL27_748:3:23:252:759/1
TTTTAGATGAAGTTATTTCTTTACTACCGTAGGCC
+
BB0D;DED>;>CEC:2EFA@69CDC3?@'%=585='
...
```

Millions of short reads from
unknown genetic locations

Base Qualities

Short Read Sequence

GCTAGCTGATAGCTAGCTAGCTGATGAGCCCGA

Short Read Base Qualities

30.30.28.28.29.27.30.29.28.25.24.26.27.24.24.23.20.21.22.10.25.25.20.20.18.17.16.15.14.14.13.12.10

- Each base is typically associated with a quality value
- Measured on a “Phred” scale, which was introduced by Phil Green for his Phred sequence analysis tool

$BQ = -10\log_{10}(\epsilon)$ where ϵ is the probability of an error

Base quality conversion

- Nowadays, we settled on using quality scores on the original Sanger format (Phred+33).

```
SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX.....
.....IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII.....JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ.....
LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [ \ ] ^ _ ` a b c d e f g h i j k l m n o p q r s t u v w x y z { | } ~
|                                     |                                     |                                     |                                     |
33                               59   64   73                               104                                   126
0.....26...31.....40
               -5...0.....9.....40
                   0.....9.....40
                       3.....9.....41
0.2.....26...31.....41
```

S - Sanger Phred+33, raw reads typically (0, 40)
X - Solexa Solexa+64, raw reads typically (-5, 40)
I - Illumina 1.3+ Phred+64, raw reads typically (0, 40)
J - Illumina 1.5+ Phred+64, raw reads typically (3, 41)
with 0=unused, 1=unused, 2=Read Segment Quality Control Indicator (bold)
(Note: See discussion above).
L - Illumina 1.8+ Phred+33, raw reads typically (0, 41)

BED: Genomic region format

BED format

BED format

[Index](#)

BED format provides a flexible way to define the data lines that are displayed in an annotation track. BED lines have three required fields and nine additional optional fields. The number of fields per line must be consistent throughout any single set of data in an annotation track. The order of the optional fields is binding: lower-numbered fields must always be populated if higher-numbered fields are used.









If your data set is BED-like, but it is very large (over 50MB) and you would like to keep it on your own server, you should use the [bigBed](#) data format.

The first three required BED fields are:

1. **chrom** - The name of the chromosome (e.g. chr3, chrY, chr2_random) or scaffold (e.g. scaffold10671).
2. **chromStart** - The starting position of the feature in the chromosome or scaffold. The first base in a chromosome is numbered 0.
3. **chromEnd** - The ending position of the feature in the chromosome or scaffold. The *chromEnd* base is not included in the display of the feature. For example, the first 100 bases of a chromosome are defined as *chromStart=0*, *chromEnd=100*, and span the bases numbered 0-99.

The 9 additional optional BED fields are:

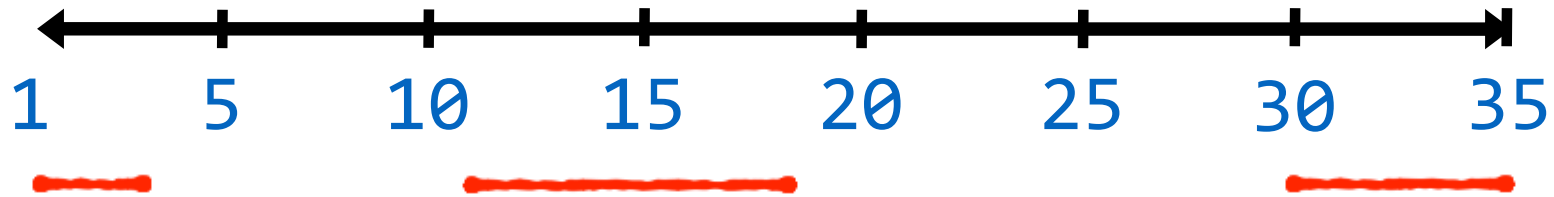
4. **name** - Defines the name of the BED line. This label is displayed to the left of the BED line in the Genome Browser window when the track is open to full display mode or directly to the left of the item in pack mode.
5. **score** - A score between 0 and 1000. If the track line *useScore* attribute is set to 1 for this annotation data set, the *score* value will determine the level of gray in which this feature is displayed (higher numbers = darker gray). This table shows the Genome Browser's translation of BED score values into shades of gray:

shade								
score in range	≤ 166	167-277	278-388	389-499	500-611	612-722	723-833	834-944 ≥ 945

6. **strand** - Defines the strand - either '+' or '-'.
7. **thickStart** - The starting position at which the feature is drawn thickly (for example, the start codon in gene displays). When there is no thick part, *thickStart* and *thickEnd* are usually set to the *chromStart* position.
8. **thickEnd** - The ending position at which the feature is drawn thickly (for example, the stop codon in gene displays).
9. **itemRgb** - An RGB value of the form R,G,B (e.g. 255,0,0). If the track line *itemRgb* attribute is set to "On", this RGB value will determine the display color of the data contained in this BED line. NOTE: It is recommended that a simple color scheme (eight colors or less) be used with this attribute to avoid overwhelming the color resources of the Genome Browser and your Internet browser.
10. **blockCount** - The number of blocks (exons) in the BED line.
11. **blockSizes** - A comma-separated list of the block sizes. The number of items in this list should correspond to *blockCount*.
12. **blockStarts** - A comma-separated list of block starts. All of the *blockStart* positions should be calculated relative to *chromStart*. The number of items in this list should correspond to *blockCount*.

Minimal BED format. So-called BED3 format.

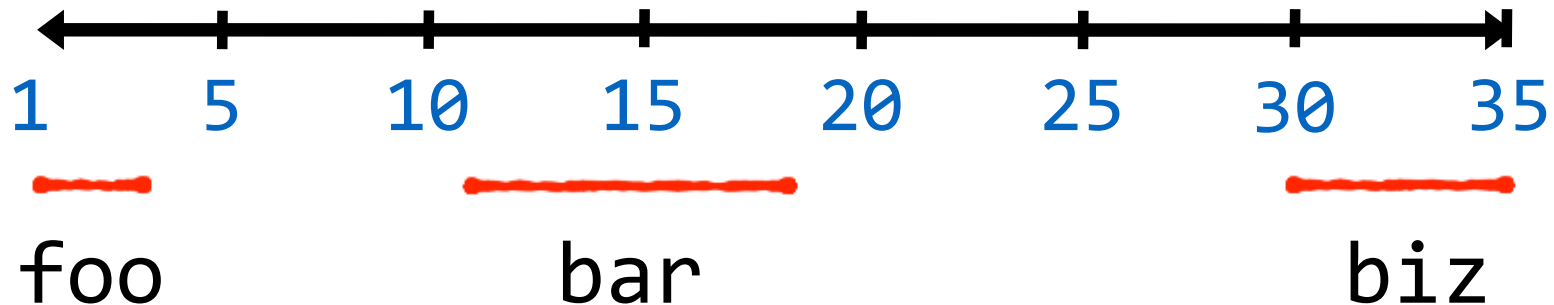
CAGTCGACATAGACTGATATGACACCACACTGAGC...



chr1	0	3
chr1	11	18
chr1	29	35

BED format supports “labels”

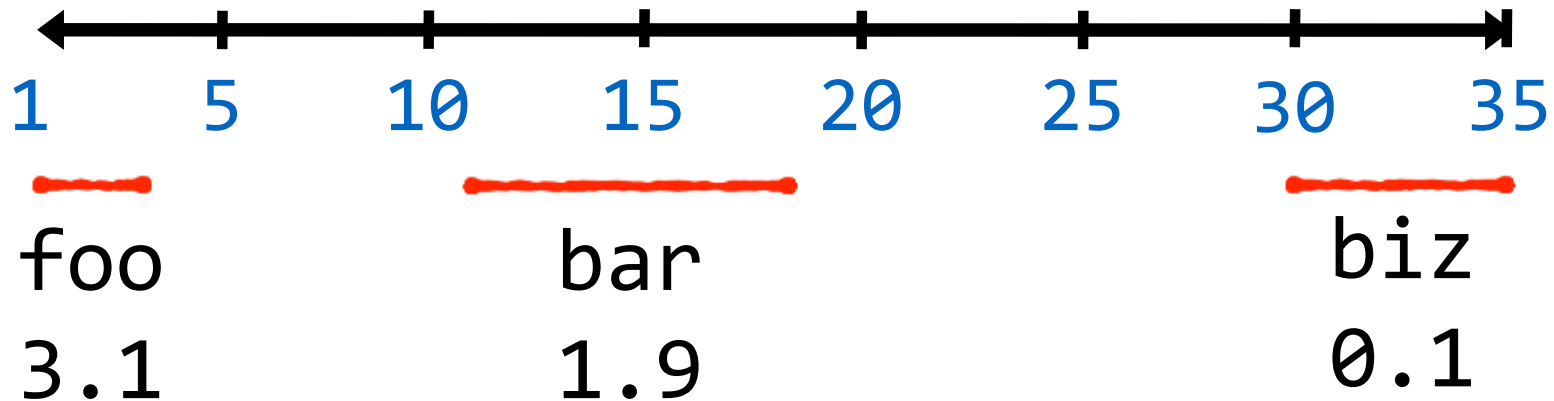
CAGTCGACATAGACTGATATGACACCACACTGAGC...



chr1	0	3	foo
chr1	11	18	bar
chr1	29	35	biz

And scores

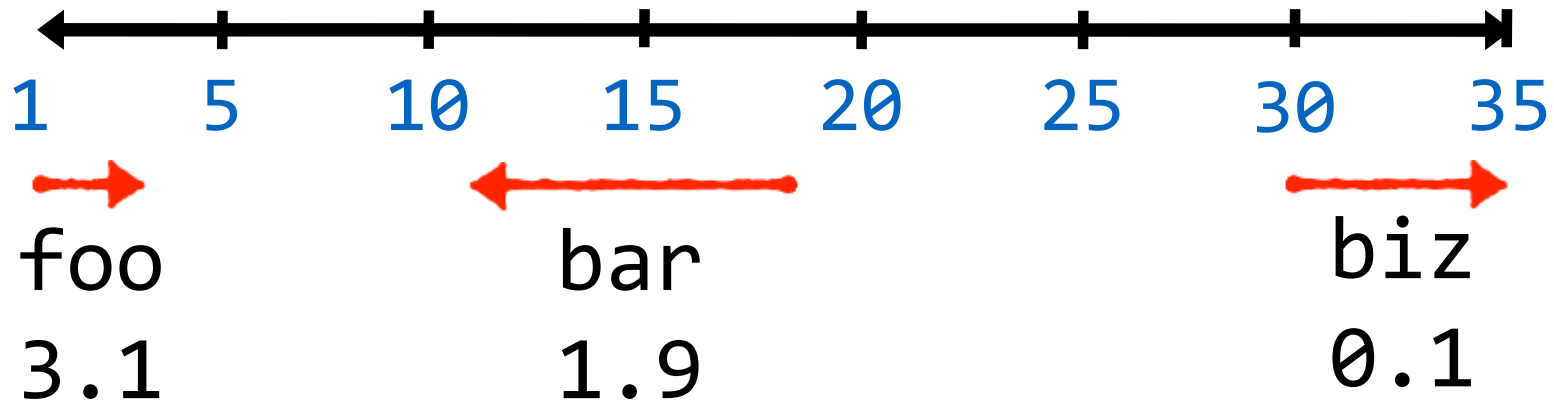
CAGTCGACATAGACTGATATGACACCACACTGAGC...



chr1	0	3	foo	3.1
chr1	11	18	bar	1.9
chr1	29	35	biz	0.1

And strands. This is so-called BED6 format.

CAGTCGACATAGACTGATATGACACCACACTGAGC...



chr1	0	3	foo	3.1	+
chr1	11	18	bar	1.9	-
chr1	29	35	biz	0.1	+

And more! BED12 format

BED format

[Index](#) ▸

BED format provides a flexible way to define the data lines that are displayed in an annotation track. BED lines have three required fields and nine additional optional fields. The number of fields per line must be consistent throughout any single set of data in an annotation track. The order of the optional fields is binding: lower-numbered fields must always be populated if higher-numbered fields are used.

If your data set is BED-like, but it is very large (over 50MB) and you would like to keep it on your own server, you should use the [bigBed](#) data format.

The first three required BED fields are:

1. **chrom** - The name of the chromosome (e.g. chr3, chrY, chr2_random) or scaffold (e.g. scaffold10671).
2. **chromStart** - The starting position of the feature in the chromosome or scaffold. The first base in a chromosome is numbered 0.
3. **chromEnd** - The ending position of the feature in the chromosome or scaffold. The *chromEnd* base is not included in the display of the feature. For example, the first 100 bases of a chromosome are defined as *chromStart*=0, *chromEnd*=100, and span the bases numbered 0-99.

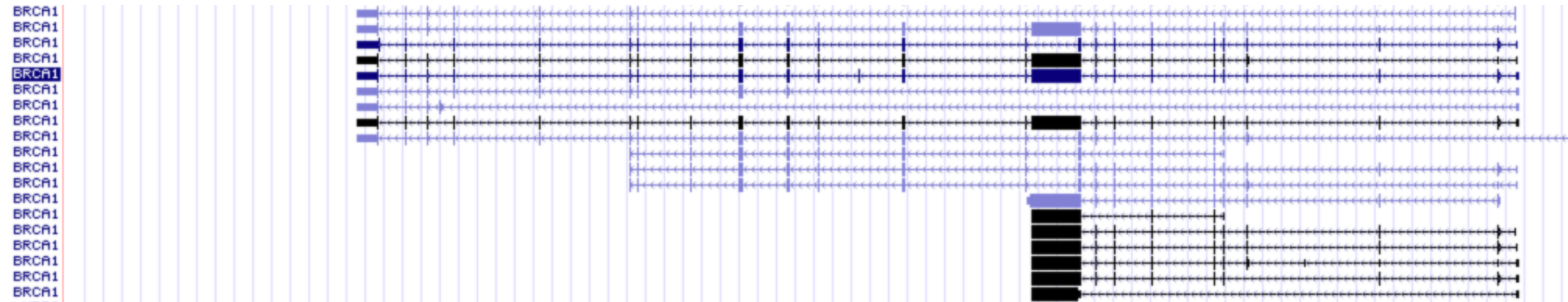
The 9 additional optional BED fields are:

4. **name** - Defines the name of the BED line. This label is displayed to the left of the BED line in the Genome Browser window when the track is open to full display mode or directly to the left of the item in pack mode.
5. **score** - A score between 0 and 1000. If the track line *useScore* attribute is set to 1 for this annotation data set, the *score* value will determine the level of gray in which this feature is displayed (higher numbers = darker gray). This table shows the Genome Browser's translation of BED score values into shades of gray:

shade								
score in range	≤ 166	167-277	278-388	389-499	500-611	612-722	723-833	834-944 ≥ 945

6. **strand** - Defines the strand - either '+' or '-'.
7. **thickStart** - The starting position at which the feature is drawn thickly (for example, the start codon in gene displays). When there is no thick part, thickStart and thickEnd are usually set to the chromStart position.
8. **thickEnd** - The ending position at which the feature is drawn thickly (for example, the stop codon in gene displays).
9. **itemRgb** - An RGB value of the form R,G,B (e.g. 255,0,0). If the track line *itemRgb* attribute is set to "On", this RGB value will determine the display color of the data contained in this BED line. NOTE: It is recommended that a simple color scheme (eight colors or less) be used with this attribute to avoid overwhelming the color resources of the Genome Browser and your Internet browser.
10. **blockCount** - The number of blocks (exons) in the BED line.
11. **blockSizes** - A comma-separated list of the block sizes. The number of items in this list should correspond to *blockCount*.
12. **blockStarts** - A comma-separated list of block starts. All of the *blockStart* positions should be calculated relative to *chromStart*. The number of items in this list should correspond to *blockCount*.

BED12 example



chr17	41196311	41277340	uc010whm.2	0	-	41197694	41277202	0	8	1508,61,74,55,84,41,78,142,	0,3348,4826,6768,12757,19038,19579,80887,
chr17	41196311	41277340	uc002icp.4	0	-	41197694	41258496	0	23	1508,61,74,55,84,41,78,88,311,191,127,172,89,3426,77,46,103,140,89,56,54,99,142,	
0,3348,4826,6768,12757,19038,19579,23313,26633,30036,32193,38109,46649,47140,51551,52949,55480,59827,60573,62183,71431,79722,80887,											
chr17	41196311	41277468	uc002icu.3	0	-	41197800	41276113	0	22	1508,61,55,84,41,78,88,311,191,124,172,89,117,77,46,106,140,89,78,54,99,175,	
0,3348,6768,12757,19038,19579,23313,26633,30036,32193,38109,46649,50449,51551,52949,55480,59827,60573,62161,71431,79722,80982,											
chr17	41196311	41277468	uc010cyx.3	0	-	41197694	41258543	0	22	1508,61,74,55,84,41,78,88,311,191,127,172,89,3426,77,46,106,140,89,78,99,175,	
0,3348,4826,6768,12757,19038,19579,23313,26633,30036,32193,38109,46649,47140,51551,52949,55480,59827,60573,62161,79722,80982,											
chr17	41196311	41277500	uc002ict.3	0	-	41197694	41276113	0	24	1508,61,74,55,84,41,78,88,311,191,124,66,172,89,3426,77,46,106,140,89,78,54,99,213,	
0,3348,4826,6768,12757,19038,19579,23313,26633,30036,32193,35039,38109,46649,47140,51551,52949,55480,59827,60573,62161,71431,79722,80976,											
chr17	41196311	41277500	uc010whn.2	0	-	41197694	41226495	0	11	1508,61,74,55,84,41,78,88,311,191,213,	0,3348,4826,6768,12757,19038,19579,23313,26633,30036,80976,
chr17	41196311	41277500	uc010who.3	0	-	41197694	41202109	0	5	1508,61,74,129,213,	0,3348,4826,5767,80976,
chr17	41196311	41277500	uc002icq.3	0	-	41197694	41276113	0	23	1508,61,74,55,84,41,78,88,311,191,127,172,89,3426,77,46,106,140,89,78,54,99,213,	
0,3348,4826,6768,12757,19038,19579,23313,26633,30036,32193,38109,46649,47140,51551,52949,55480,59827,60573,62161,71431,79722,80976,											
chr17	41196311	41322420	uc010whp.2	0	-	41197694	41258543	0	22	1508,61,74,55,84,41,78,88,311,191,124,172,89,117,77,46,106,140,89,78,54,278,	
0,3348,4826,6768,12757,19038,19579,23313,26633,30036,32193,38109,46649,50449,51551,52949,55480,59827,60573,62161,71431,125831,											
chr17	41215349	41256973	uc010whq.1	0	-	41215349	41256198	0	12	41,78,88,311,191,127,172,89,117,106,140,89,	
0,541,4275,7595,10998,13155,19071,27611,31411,36442,40789,41535,											
chr17	41215349	41277468	uc002ide.1	0	-	41215349	41276113	0	18	41,78,88,311,191,127,172,89,117,77,46,103,140,89,78,54,99,175,	
0,541,4275,7595,10998,13155,19071,27611,31411,32513,33911,36442,40789,41535,43123,52393,60684,61944,											
chr17	41215349	41277468	uc010whr.1	0	-	41215349	41258543	0	17	41,78,88,311,191,127,172,89,117,77,46,106,140,89,78,99,175,	
0,541,4275,7595,10998,13155,19071,27611,31411,32513,33911,36442,40789,41535,43123,60684,61944,											
chr17	41243116	41276132	uc002idd.3	0	-	41243347	41276113	0	9	3761,77,46,106,140,89,78,54,99,	0,4746,6144,8675,13022,13768,15356,24626,32917,
chr17	41243451	41256973	uc002ide.1	0	-	41243452	41256198	0	4	3426,103,140,89,	0,8340,12687,13433,
chr17	41243451	41277340	uc010cyp.1	0	-	41243452	41276113	0	10	3426,77,46,106,140,89,78,54,99,142,	0,4411,5809,8340,12687,13433,15021,24291,32582,33747,
chr17	41243451	41277468	uc010whs.1	0	-	41243452	41276113	0	10	3426,77,46,106,140,89,78,54,99,175,	0,4411,5809,8340,12687,13433,15021,24291,32582,33842,
chr17	41243451	41277500	uc010cyz.2	0	-	41243452	41258543	0	11	3426,77,46,106,140,89,78,116,54,99,213,	0,4411,5809,8340,12687,13433,15021,19030,24291,32582,33836,
chr17	41243451	41277500	uc010cca.2	0	-	41243452	41276113	0	9	3426,77,46,106,140,89,54,99,213,	0,4411,5809,8340,12687,13433,24291,32582,33836,
chr17	41243451	41277500	uc010wht.1	0	-	41243452	41246659	0	2	3426,213,	0,33836,
chr17	41277599	41292342	uc002idf.3	0	+	41277599	41277599	0	4	188,63,182,1669,	0,5625,7373,13074,
chr17	41277599	41292342	uc010ceb.2	0	+	41277599	41277599	0	2	188,1669,	0,13074,
chr17	41277599	41297125	uc002idg.3	0	+	41277599	41277599	0	5	188,63,266,468,381,	0,5625,13074,14233,19145,
chr17	41277599	41305688	uc002idh.3	0	+	41277599	41277599	0	8	188,63,125,182,205,266,120,70,	0,5625,7016,7373,12573,13074,16874,28019,

Slide in this section are from Jim Havrilla and Aaron Quinlan

GFF: annotates one line per feature

GFF format

GFF format

[Index](#) ▸

GFF (General Feature Format) lines are based on the Sanger [GFF2 specification](#). GFF lines have nine required fields that *must* be tab-separated. If the fields are separated by spaces instead of tabs, the track will not display correctly. For more information on GFF format, refer to Sanger's [GFF page](#).

Note that there is also a GFF3 specification that is not currently supported by the Browser. All GFF tracks must be formatted according to Sanger's GFF2 specification.

If you would like to obtain browser data in GFF (GTF) format, please refer to [Genes in gff or gtf format](#) on the Wiki.

Here is a brief description of the GFF fields:

1. **seqname** - The name of the sequence. Must be a chromosome or scaffold.
2. **source** - The program that generated this feature.
3. **feature** - The name of this type of feature. Some examples of standard feature types are "CDS", "start_codon", "stop_codon", and "exon".
4. **start** - The starting position of the feature in the sequence. The first base is numbered 1.
5. **end** - The ending position of the feature (inclusive).
6. **score** - A score between 0 and 1000. If the track line *useScore* attribute is set to 1 for this annotation data set, the *score* value will determine the level of gray in which this feature is displayed (higher numbers = darker gray). If there is no score value, enter ".".
7. **strand** - Valid entries include '+', '-', or '.' (for don't know/don't care).
8. **frame** - If the feature is a coding exon, *frame* should be a number between 0-2 that represents the reading frame of the first base. If the feature is not a coding exon, the value should be '.'.
9. **group** - All lines with the same group are linked together into a single item.

Example:

Here's an example of a GFF-based track. This [example](#) can be pasted into the browser without editing. NOTE: Paste operations on some operating systems will replace tabs with spaces, which will result in an error when the GFF track is uploaded. You can circumvent this problem by pasting the URL of the above example (<http://genome.ucsc.edu/goldenPath/help/regulatory.txt>) instead of the text itself into the custom annotation track text box. If you encounter an error when loading a GFF track, check that the data lines contain tabs rather than spaces.

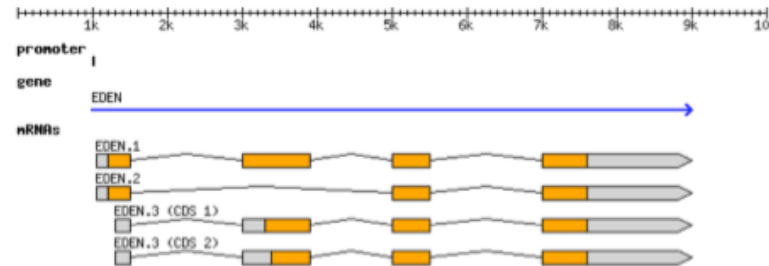
```
browser position chr22:10000000-10025000
browser hide all
track name=regulatory description="TeleGene(tm) Regulatory Regions"
visibility=2
chr22 TeleGene enhancer 10000000 10001000 500 + . touch1
chr22 TeleGene promoter 10010000 10010100 900 + . touch1
chr22 TeleGene promoter 10020000 10025000 800 - . touch2
```

Click [here](#) to display this track in the Genome Browser.

chr22	TeleGene	enhancer	10000000	10001000	500	+	.	touch1
chr22	TeleGene	promoter	10010000	10010100	900	+	.	touch1
chr22	TeleGene	promoter	10020000	10025000	800	-	.	touch2

GFF example

ly
spliced transcripts, isoform 3 has two
alternative translation start sites



```
##gff-version 3
##sequence-region ctgl123 1 1497228
ctgl123 . gene 1000 9000 . + . ID=gene00001;Name=EDEN

ctgl123 . TF_binding_site 1000 1012 . + . ID=tfbs00001;Parent=gene00001

ctgl123 . mRNA 1050 9000 . + . ID=mRNA00001;Parent=gene00001;Name=EDEN.1
ctgl123 . mRNA 1050 9000 . + . ID=mRNA00002;Parent=gene00001;Name=EDEN.2
ctgl123 . mRNA 1300 9000 . + . ID=mRNA00003;Parent=gene00001;Name=EDEN.3

ctgl123 . exon 1300 1500 . + . ID=exon00001;Parent=mRNA00003
ctgl123 . exon 1050 1500 . + . ID=exon00002;Parent=mRNA00001,mRNA00002
ctgl123 . exon 3000 3902 . + . ID=exon00003;Parent=mRNA00001,mRNA00003
ctgl123 . exon 5000 5500 . + . ID=exon00004;Parent=mRNA00001,mRNA00002,mRNA00003
ctgl123 . exon 7000 9000 . + . ID=exon00005;Parent=mRNA00001,mRNA00002,mRNA00003

ctgl123 . CDS 1201 1500 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
ctgl123 . CDS 3000 3902 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
ctgl123 . CDS 5000 5500 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
ctgl123 . CDS 7000 7600 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1

ctgl123 . CDS 1201 1500 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
ctgl123 . CDS 5000 5500 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
ctgl123 . CDS 7000 7600 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2

ctgl123 . CDS 3301 3902 . + 0 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
ctgl123 . CDS 5000 5500 . + 1 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
ctgl123 . CDS 7000 7600 . + 1 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3

ctgl123 . CDS 3391 3902 . + 0 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
ctgl123 . CDS 5000 5500 . + 1 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
ctgl123 . CDS 7000 7600 . + 1 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
```

BAM/SAM: Genome alignment format

BAM/SAM format

- SAM: Sequence Alignment/Map format (tab-delimited text file).
- BAM: The binary equivalent of a SAM file, which stores the same data in a compressed binary representation

Col	Field	Type	Brief description
1	QNAME	String	Query template NAME
2	FLAG	Int	bitwise FLAG
3	RNAME	String	References sequence NAME
4	POS	Int	1- based leftmost mapping POSition
5	MAPQ	Int	MAPping Quality
6	CIGAR	String	CIGAR String
7	RNEXT	String	Ref. name of the mate/next read
8	PNEXT	Int	Position of the mate/next read
9	TLEN	Int	observed Template LENgth
10	SEQ	String	segment SEquence
11	QUAL	String	ASCII of Phred-scaled base QUALity+33

Example of a SAM file

```
@HD VN:1.5 SO:coordinate
@SQ SN:ref LN:45
r001 99 ref 7 30 8M2I4M1D3M = 37 39 TTAGATAAAGGATACTG *
r002 0 ref 9 30 3S6M1P1I4M * 0 0 AAAAGATAAGGATA *
r003 0 ref 9 30 5S6M * 0 0 GCCTAAGCTAA * SA:Z:ref,29,-,6H5M,17,0;
r004 0 ref 16 30 6M14N5M * 0 0 ATAGCTTCAGC *
r003 2064 ref 29 17 6H5M * 0 0 TAGGC * SA:Z:ref,9,+,5S6M,30,1;
r001 147 ref 37 30 9M = 7 -39 CAGCGGCAT * NM:i:1
```

Header
section

Alignment
section

Optional fields in the format of TAG:TYPE:VALUE

QUAL: read quality; * meaning such information is not available

SEQ: read sequence

TLEN: the number of bases covered by the reads from the same fragment. Plus/minus means the current read is the leftmost/rightmost read. E.g. compare first and last lines.

PNEXT: Position of the primary alignment of the NEXT read in the template. Set as 0 when the information is unavailable. It corresponds to POS column.

RNEXT: reference sequence name of the primary alignment of the NEXT read. For paired-end sequencing, NEXT read is the paired read, corresponding to the RNAME column.

CIGAR: summary of alignment, e.g. insertion, deletion

MAPQ: mapping quality

POS: 1-based position

RNAME: reference sequence name, e.g. chromosome/transcript id

FLAG: indicates alignment information about the read, e.g. paired, aligned, etc.

QNAME: query template name, aka. read ID

CRAM

- CRAM was designed to be an efficient reference-based alternative to SAM/BAM file formats
- Better lossless compression than BAM, but also allow for controlled loss of BAM data
- Typically used for large-scale population-based genome/exome sequencing project (for example, CRAMs has ~50TB for 50K exomes in UK Biobank).

VCF: Genetic variant file format

VCF file format

- Variant Call Format, established > 10 years ago and is now a gold standard for describing variants.
- One locus per line, and it may contain more than one mutations, but most lines contain one variant only.
- Additional header lines starts with “#” to explain the meaning of the various tags in the file

Example of a VCF file

```
##fileformat=VCFv4.3
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",taxonomy=x>
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA00001 NA00002 NA00003
20 14370 rs6054257 G A 29 PASS NS=3;DP=14;AF=0.5;DB;H2 GT:GQ:DP:HQ 0|0:48:1:51,51 1|0:48:8:51,51 1/1:43:5:.,.
20 17330 . T A 3 q10 NS=3;DP=11;AF=0.017 GT:GQ:DP:HQ 0|0:49:3:58,50 0|1:3:5:65,3 0/0:41:3
20 1110696 rs6040355 A G,T 67 PASS NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:6:23,27 2|1:2:0:18,2 2/2:35:4
20 1230237 . T . 47 PASS NS=3;DP=13;AA=T GT:GQ:DP:HQ 0|0:54:7:56,60 0|0:48:4:51,51 0/0:61:2
20 1234567 microsat1 GTC G,GTCT 50 PASS NS=3;DP=9;AA=G GT:GQ:DP 0/1:35:4 0/2:17:2 1/1:40:3
```

As of June 2019, the latest version is 4.3

The header line of a VCF file

- The header line names the 8 fixed, mandatory columns. These columns are as follows:
 - #CHROM POS ID REF ALT QUAL FILTER INFO
- If genotype data is present in the file, these are followed by a FORMAT column header, then an arbitrary number of sample IDs

The INFO line of a VCF file

- INFO fields are encoded as a semicolon-separated series of short keys with optional values in the format: key[=data[,data]].
- Some keys are reserved:

Key	Number	Type	Description
AA	1	String	Ancestral allele
AC	A	Integer	Allele count in genotypes, for each ALT allele, in the same order as listed
AD	R	Integer	Total read depth for each allele
ADF	R	Integer	Read depth for each allele on the forward strand
ADR	R	Integer	Read depth for each allele on the reverse strand
AF	A	Float	Allele frequency for each ALT allele in the same order as listed (estimated from primary data, not called genotypes)
AN	1	Integer	Total number of alleles in called genotypes
BQ	1	Float	RMS base quality
CIGAR	A	String	Cigar string describing how to align an alternate allele to the reference allele
DB	0	Flag	dbSNP membership
DP	1	Integer	Combined depth across samples

The genotypes in a VCF file

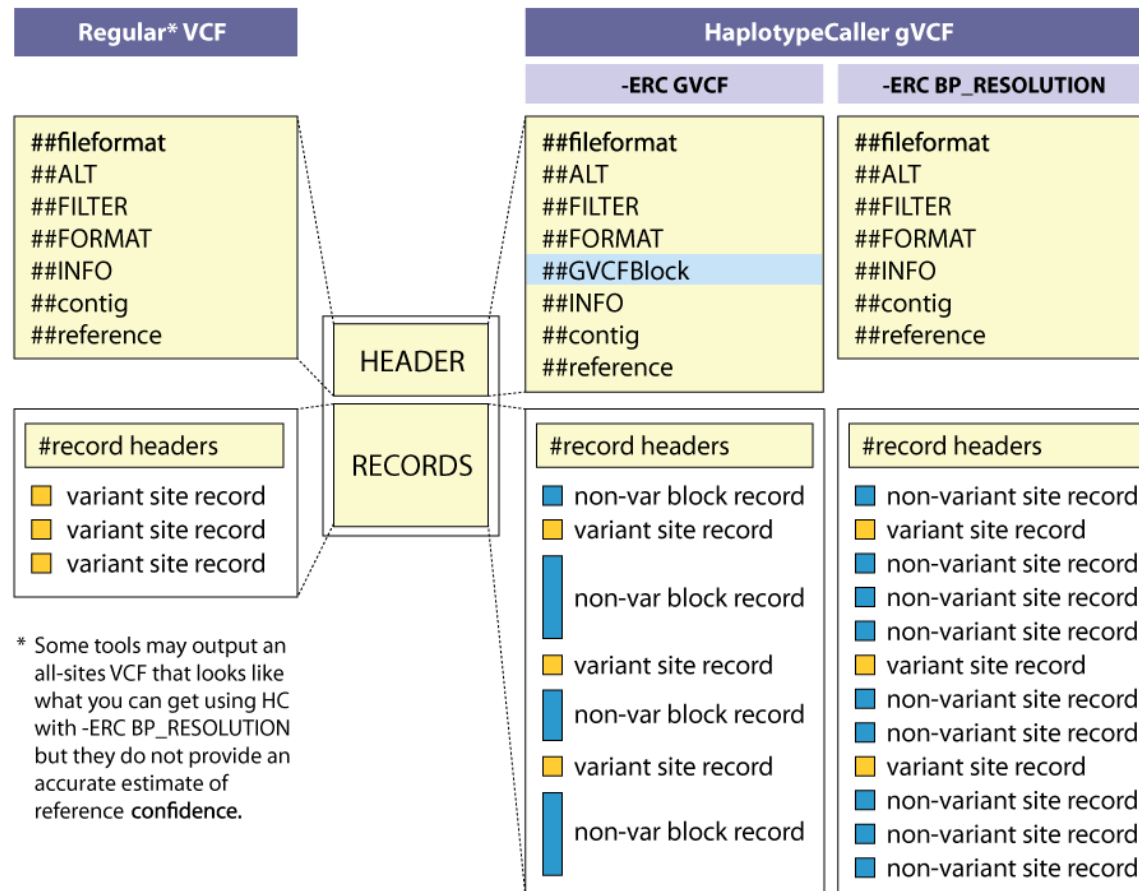
- A FORMAT field is given specifying the data types and order
- This is followed by one data block per sample, with the colon-separated data corresponding to the types specified in the format.

FORMAT	NA000001	NA000002	NA000003
GT:GQ:DP:HQ	0 0:48:1:51,51	1 0:48:8:51,51	1/1:43:5:.,.
GT:GQ:DP:HQ	0 0:49:3:58,50	0 1:3:5:65,3	0/0:41:3
GT:GQ:DP:HQ	1 2:21:6:23,27	2 1:2:0:18,2	2/2:35:4
GT:GQ:DP:HQ	0 0:54:7:56,60	0 0:48:4:51,51	0/0:61:2
GT:GQ:DP	0/1:35:4	0/2:17:2	1/1:40:3

gVCF format

- gVCF (Genomic VCF): the basic format specification is the same as for a regular VCF, but gVCF contains extra information.
- gVCF was developed to store sequencing information for both variant and non-variant positions, which is required for human clinical applications.

VCF versus gVCF



Formats use different coordinate systems, which adds confusion

BED: 0-based, half-open

GFF: 1-based, fully closed

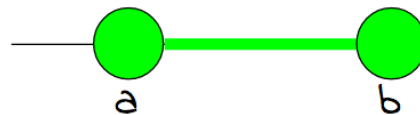
SAM: 1-based, fully closed

BAM: 0-based, half-open

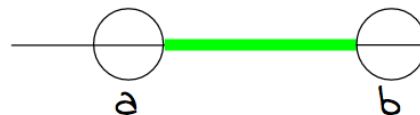
VCF: 1-based, fully closed

...

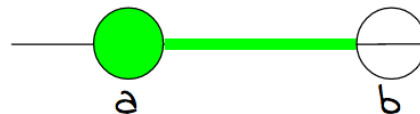
Given an interval ---a-----b---



FULLY CLOSED (HINT: THINK EN-CLOSED!)
CLOSED-START, CLOSED-END
BOTH ENDPOINTS "A" AND "B" ARE INCLUDED



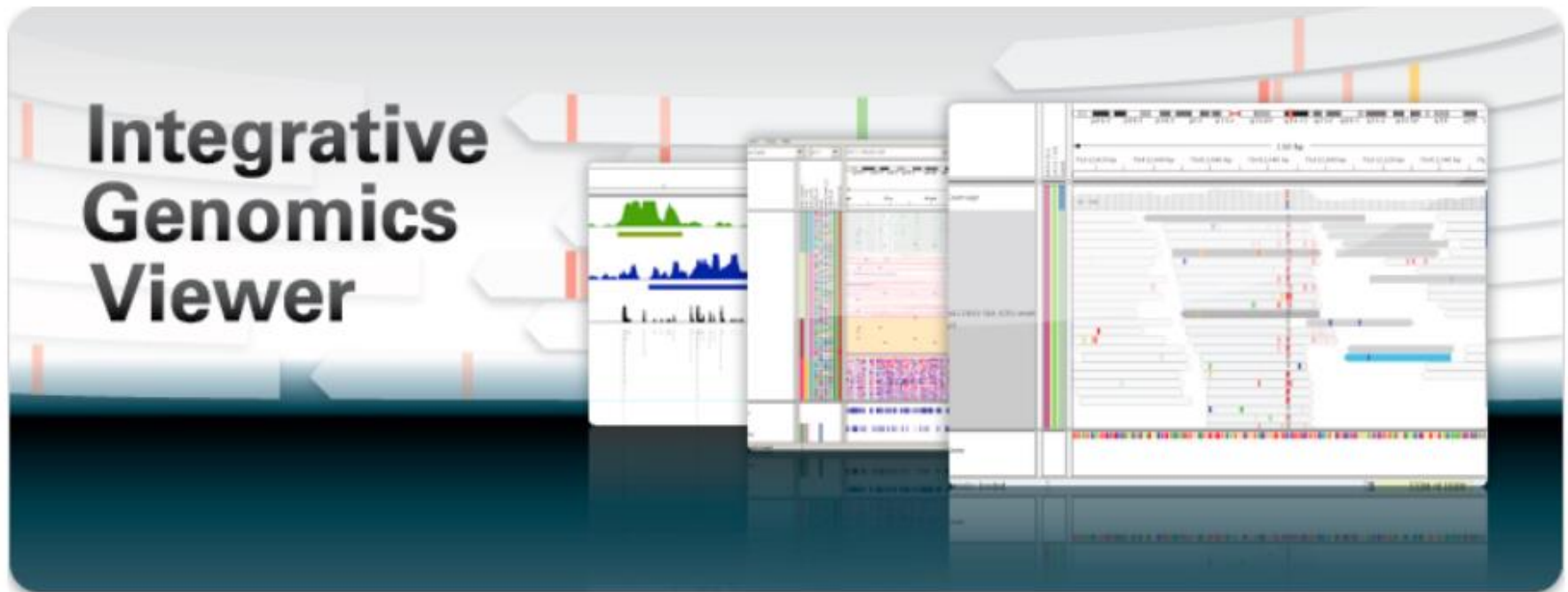
FULLY OPEN
OPEN-START, OPEN-END
BOTH ENDPOINTS "A" AND "B" ARE EXCLUDED



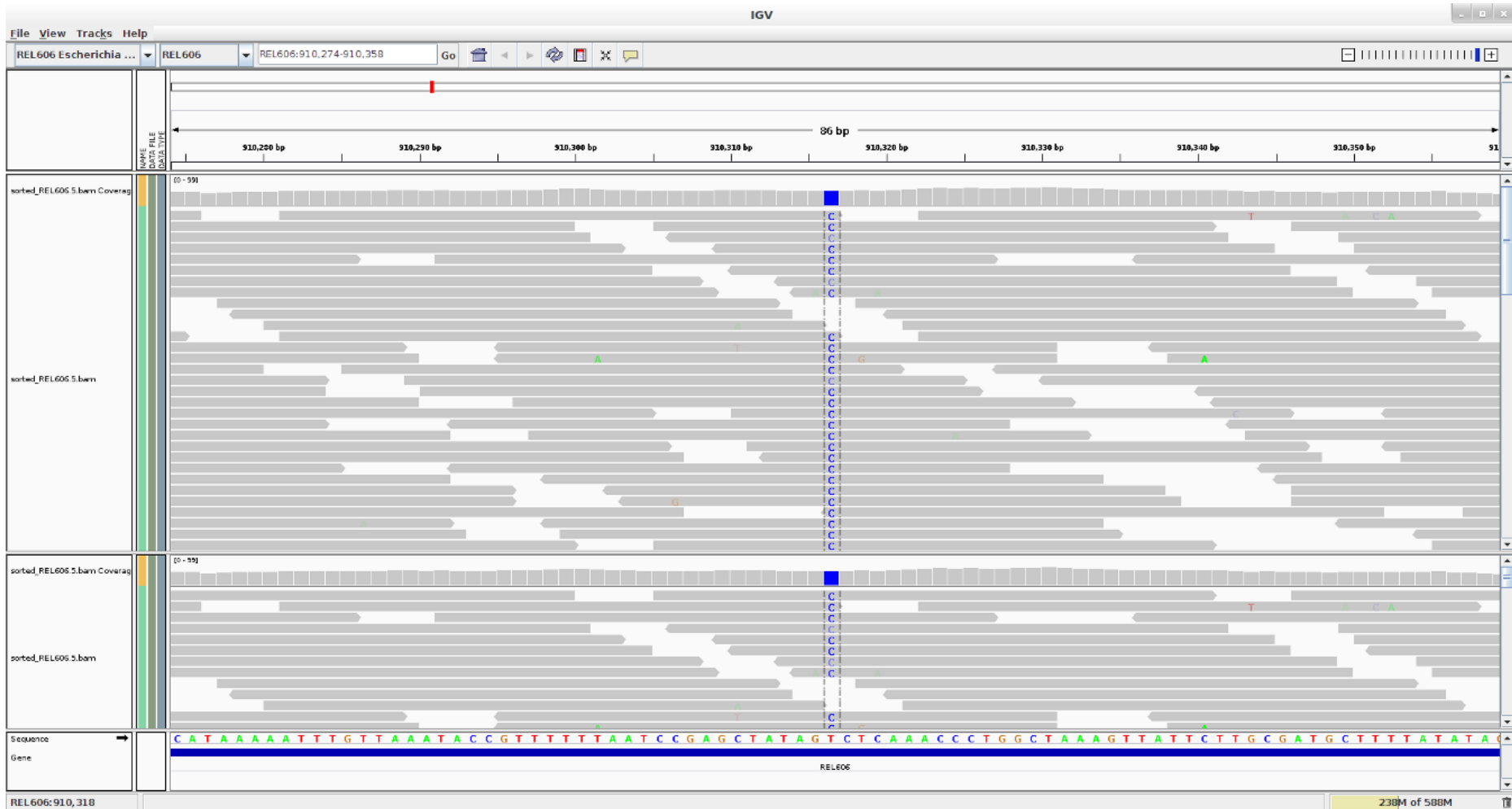
HALF-OPEN
CLOSED-START, OPEN-END
ENDPOINT "A" IS INCLUDED, "B" IS EXCLUDED

Visualization of genomic data

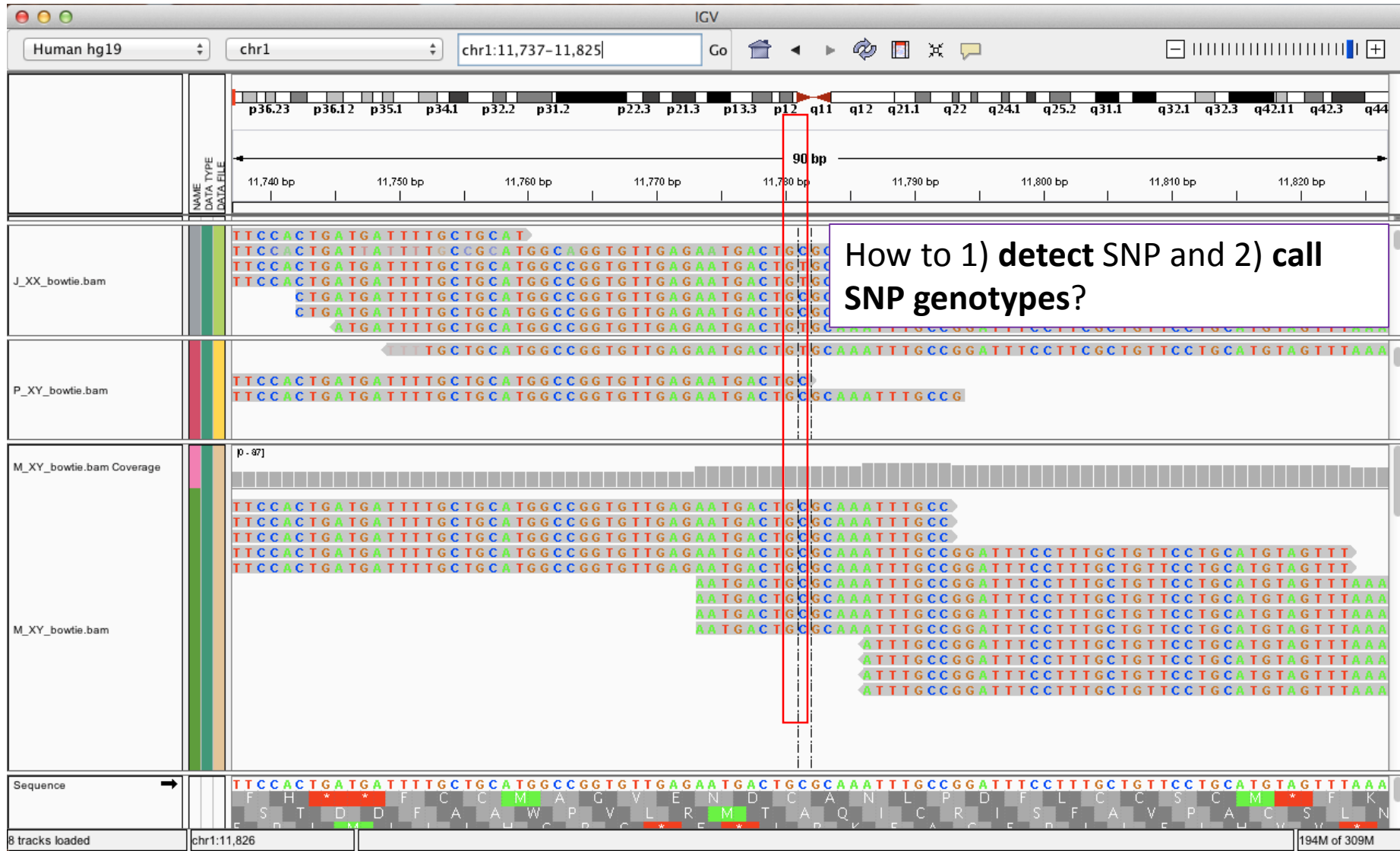
- Integrative Genomics Viewer (IGV) is a high-performance visualization tool for interactive exploration of large, integrated genomic datasets.



Visualization: IGV Viewer



Visualization: IGV Viewer



What is coverage?

- depth of sequencing coverage can be defined theoretically as LN/G , where L is the read length, N is the number of reads and G is the haploid genome length.
- The breadth of coverage is the percentage of target bases that have been sequenced for a given number of times.
- The accuracy of variant calling is affected by sequence quality, uniformity of coverage and the threshold of false-discovery rate that is used.

Read 1:	CGGATTACGTGGACCATG (read length of 18)
Read 2:	ATTACGTGGACCATGAATTGCTGACA
Read 3:	ACCATGAATTGCTGACATTCGTCA
Read 4:	TGAATTGCTGACATTCGTCAT
Depth:	11122222222233334433333333332222221

Coverage: how many reads we need to cover the genome?

Depth of coverage model

In a shotgun sequencing, genome size G , read size S , N reads

Probability of a read starts at an interval L is $p=L/G$

D : Number of reads start in an interval of length L :

$$D \sim \text{Binomial}(N, L/G)$$

Let $L = S$, then D is also the number of reads that cover the last position of the interval \rightarrow
 D is depth of coverage.

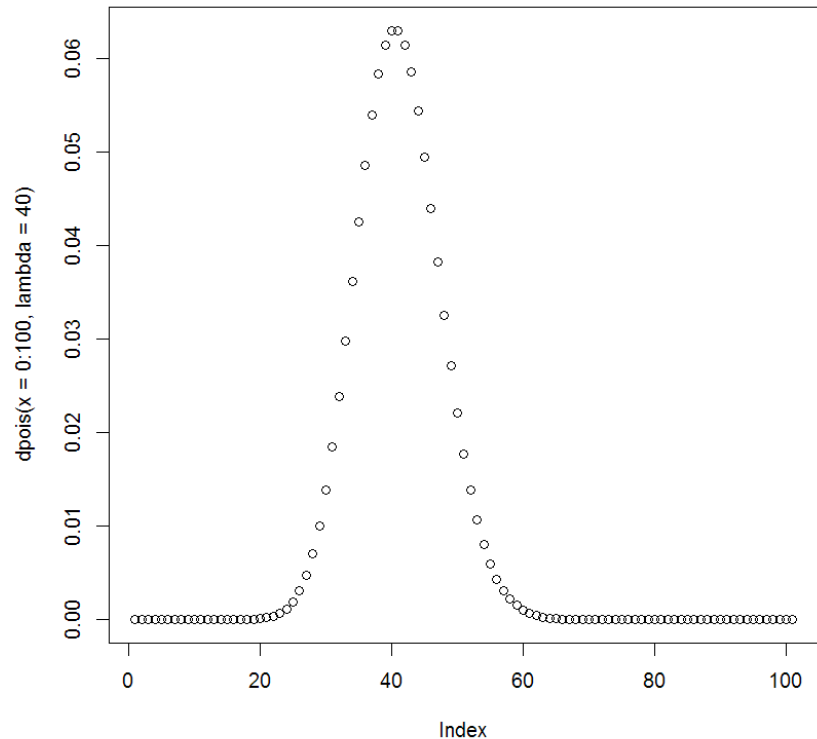
$S \ll G$, N very large, therefore depth of coverage is approximated by a Poisson distribution:

$$D \sim \text{Poisson}(\lambda)$$

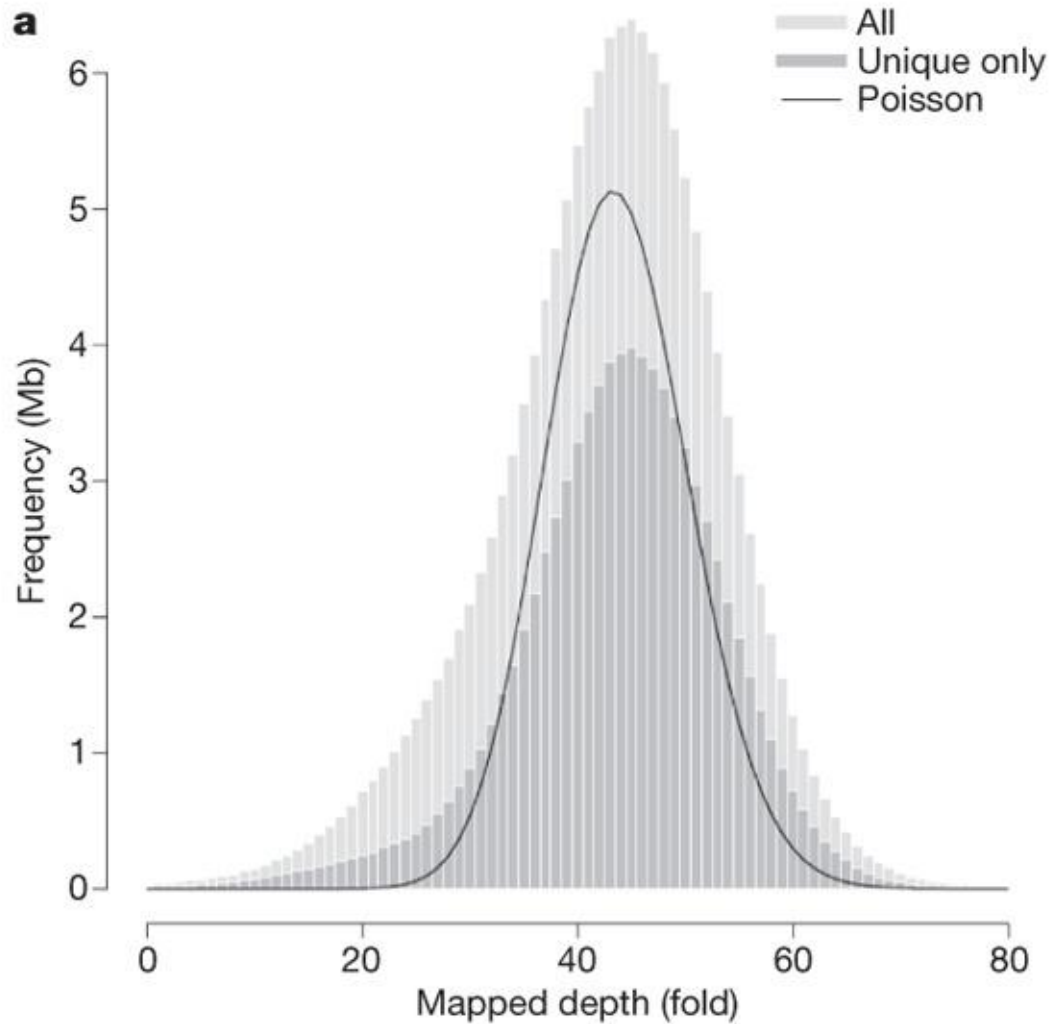
$$\lambda = SN/G \text{ (average depth of coverage)}$$

Fraction of genome that are covered

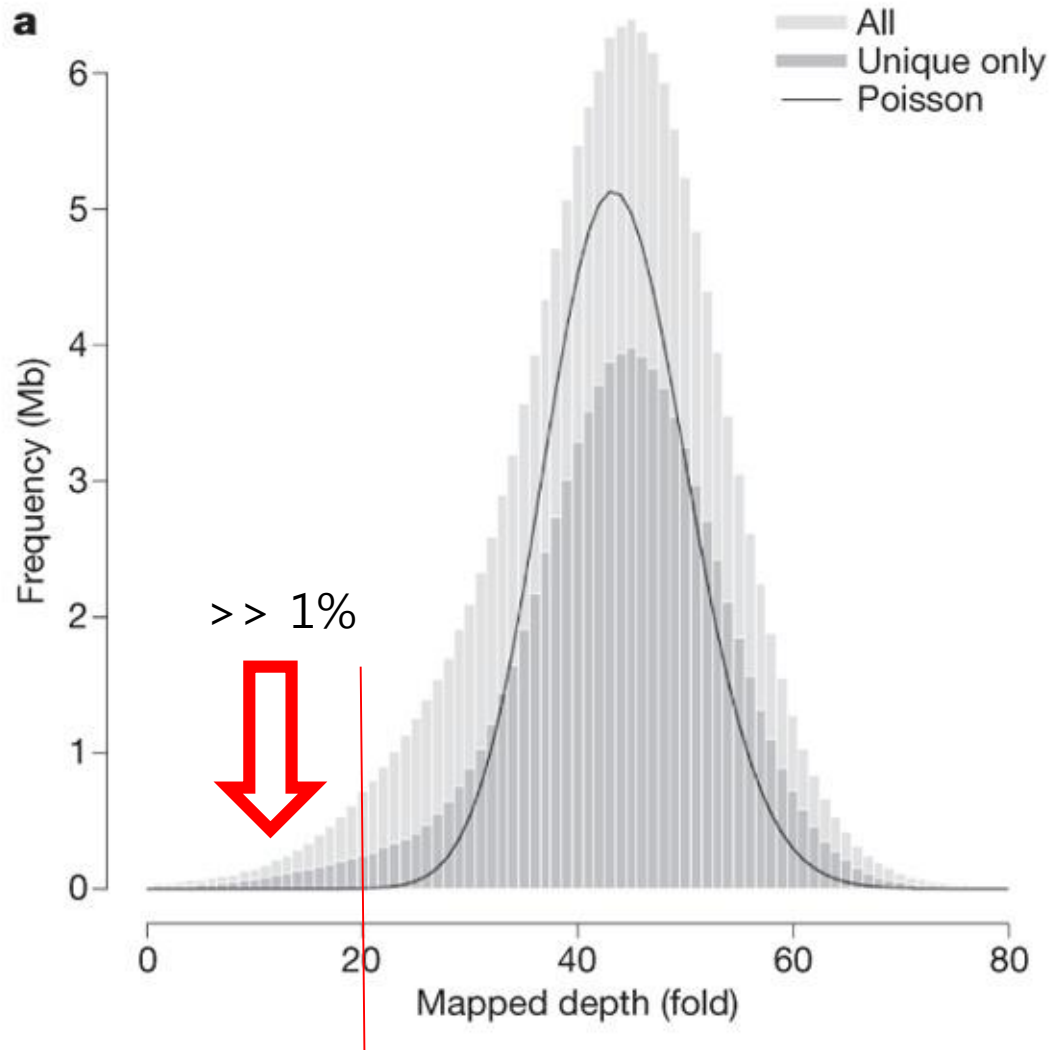
- Given $\lambda=40$, the fraction of genome that are covered more than 30x ($D>30$) is: 0.938
- Given $\lambda=40$, the fraction of genome that are covered more than 20x ($D>20$) is: 0.9996



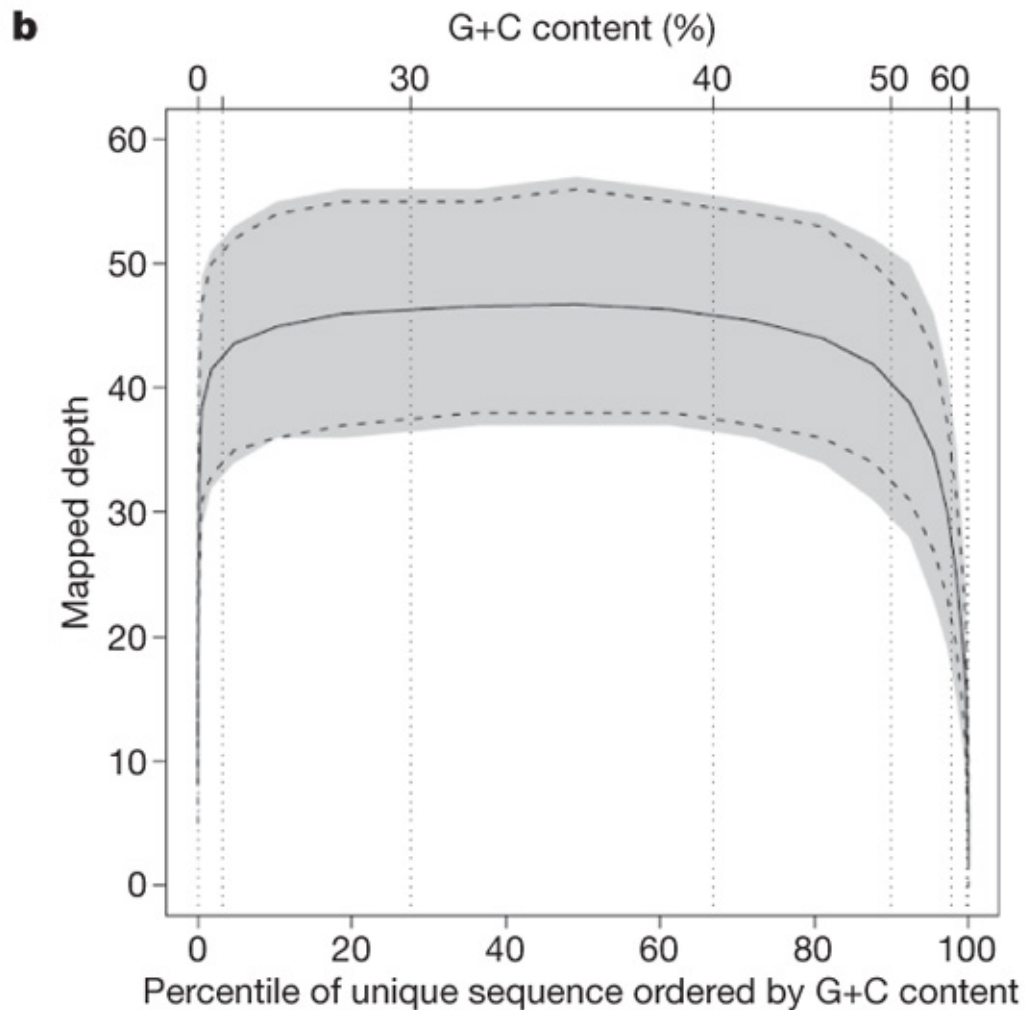
Empirical results



Empirical results



Main cause of overdispersion



How to model overdispersion

GC bias and other technical factors lead to systematic bias in coverage, resulting in overdispersion

Ideal situation (Poisson distribution):

$$\text{Var}(D) = \mu$$

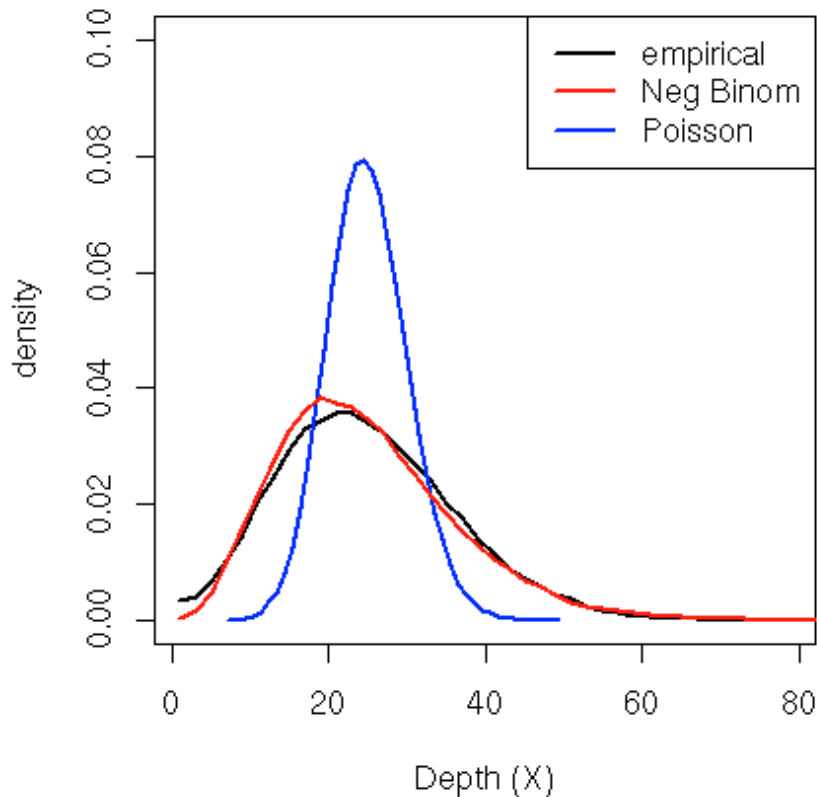
Gamma-Poisson is equivalent to Negative Binomial, which is a commonly used model for dealing with overdispersion in count data:

$$\text{Var}(D) = \mu + \mu^2/k$$

It is generalizable to model overdispersion caused by other factors, e.g. biological noise.

Larger $k \rightarrow$ smaller overdispersion

How to model overdispersion



- Over-dispersion: **var** > μ
- Due to sequencing bias:
 - **GC content**
 - Other context dependent factors

Question on coverage

- Why do we need average 30-50x in a typical WGS experiment, and 100x in WES?

General strategy for variant calling

- Reads piled up at each base of interest
- With per-base qualities and mapping quality


TAGCTGATAGCTAG**A**TAGCTGATGAGCCCGAT
ATAGCTAG**A**TAGCTGATGAGCCCGATCGCTGCTAGCTC
ATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCC
AGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTG
GCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

A/C

Predicted Genotype

NGS Data

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAGCTAGCTAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

$P(\text{reads} \mid A/A, \text{read mapped}) = 1.0$

$P(\text{reads} \mid A/C, \text{read mapped}) = 1.0$

$P(\text{reads} \mid C/C, \text{read mapped}) = 1.0$

Possible Genotypes

NGS Data

GCTAGCTGATAGCTAGCTAGCTAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAGCTAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

P(reads | A/A, read mapped) = P(C observed | A/A, read mapped)

P(reads | A/C, read mapped) = P(C observed | A/C, read mapped)

P(reads | C/C, read mapped) = P(C observed | C/C, read mapped)

Possible Genotypes

NGS Data

Assuming that sequencing error rate is 0.01

GCTAGCTGATAGCTAGCTAGCTAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAGCTAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

$$P(\text{reads} | A/A, \text{read mapped}) = 0.01$$

$$P(\text{reads} | A/C, \text{read mapped}) = 0.50$$

$$P(\text{reads} | C/C, \text{read mapped}) = 0.99$$

Possible Genotypes

NGS Data


AGCTGATAGCTAGCTAGCTGATGAGCCCGATCGCTG
GCTAGCTGATAGCTAGCTAGCTAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAGCTAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

$P(\text{reads} | A/A, \text{read mapped}) = 0.0001$

$P(\text{reads} | A/C, \text{read mapped}) = 0.25$

$P(\text{reads} | C/C, \text{read mapped}) = 0.98$

Possible Genotypes

NGS Data

ATGCTAGCTGATAGCTAGCTAGCTGATGAGCC
AGCTGATAGCTAGCTAGCTGATGAGCCCGATCGCTG
GCTAGCTGATAGCTAGCTAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAGCTAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

$$P(\text{reads} | A/A, \text{read mapped}) = 0.000001$$

$$P(\text{reads} | A/C, \text{read mapped}) = 0.125$$

$$P(\text{reads} | C/C, \text{read mapped}) = 0.97$$

Possible Genotypes

NGS Data


ATAGCTAG**A**TAGCTGATGAGCCCGATCGCTGCTAGCTC
ATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCC
AGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTG
GCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

$P(\text{reads} | A/A, \text{read mapped}) = 0.00000099$

$P(\text{reads} | A/C, \text{read mapped}) = 0.0625$

$P(\text{reads} | C/C, \text{read mapped}) = 0.0097$

Possible Genotypes

NGS Data



TAGCTGATAGCTAG**A**TAGCTGATGAGCCCGAT
ATAGCTAG**A**TAGCTGATGAGCCCGATCGCTGCTAGCTC
ATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCC
AGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTG
GCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

$P(\text{reads} | A/A, \text{read mapped}) = 0.00000098$

$P(\text{reads} | A/C, \text{read mapped}) = 0.03125$

$P(\text{reads} | C/C, \text{read mapped}) = 0.000097$

Possible Genotypes

NGS Data



TAGCTGATAGCTAG**A**TAGCTGATGAGCCCGAT

ATAGCTAG**A**TAGCTGATGAGCCCGATCGCTGCTAGCTC

ATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCC

AGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTG

GCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

$$P(\text{reads} | A/A, \text{read mapped}) = 0.00000098$$

$$P(\text{reads} | A/C, \text{read mapped}) = 0.03125$$

$$P(\text{reads} | C/C, \text{read mapped}) = 0.000097$$

Combine these likelihoods with a prior incorporating information from other individuals and flanking sites to assign a genotype.

NGS Data



TAGCTGATAGCTAG**A**TAGCTGATGAGCCCGAT

ATAGCTAG**A**TAGCTGATGAGCCCGATCGCTGCTAGCTC

ATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCC

AGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTG

GCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

$$P(\text{Genotype}|\text{reads}) = \frac{P(\text{reads}|\text{Genotype})\text{Prior}(\text{Genotype})}{\sum_G P(\text{reads}|G)\text{Prior}(G)}$$

Combine these likelihoods with a prior incorporating information from other individuals and flanking sites to assign a genotype.

From Sequence to Genotype: Individual Based Prior



TAGCTGATAGCTAG**A**TAGCTGATGAGCCCGAT

ATAGCTAG**A**TAGCTGATGAGCCCGATCGCTGCTAGCTC

ATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCC

AGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTG

GCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

$P(\text{reads} | A/A) = 0.00000098$ $\text{Prior}(A/A) = 0.00034$ $\text{Posterior}(A/A) = <.001$

$P(\text{reads} | A/C) = 0.03125$ $\text{Prior}(A/C) = 0.00066$ $\text{Posterior}(A/C) = 0.175$

$P(\text{reads} | C/C) = 0.000097$ $\text{Prior}(C/C) = 0.99900$ $\text{Posterior}(C/C) = 0.825$

Individual Based Prior: Every site has 1/1000 probability of varying.

From Sequence To Genotype: Population Based Prior



TAGCTGATAGCTAG**A**TAGCTGATGAGCCCGAT
 ATAGCTAG**A**TAGCTGATGAGCCCGATCGCTGCTAGCTC
 ATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCC
 AGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTG
 GCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGA

Sequence Reads

5'-ACTGGTCGATGCTAGCTGATAGCTAG**C**TAGCTGATGAGCCCGATCGCTGCTAGCTCGACG-3'

Reference Genome

$P(\text{reads} | A/A) = 0.00000098$ $\text{Prior}(A/A) = 0.04$

$\text{Posterior}(A/A) = <.001$

$P(\text{reads} | A/C) = 0.03125$ $\text{Prior}(A/C) = 0.32$

$\text{Posterior}(A/C) = 0.999$

$P(\text{reads} | C/C) = 0.000097$ $\text{Prior}(C/C) = 0.64$

$\text{Posterior}(C/C) = <.001$

Population Based Prior: Use frequency information from examining others at the same site.

In the example above, we estimated $P(A) = 0.20$

More on Prior

- **Population Based Prior**

- Uses frequency information obtained from examining other individuals
- Calling very rare polymorphisms still requires 20-30x coverage of the genome
- Calling common polymorphisms requires much less data