Genomic Data Analysis using Python

>C:\Users\Sebastian\AppData\Local\Programs\Python\Python37\python.exe -m venv .env

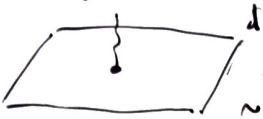
Genome → 23 pairs of chromosomes ~ 3 Gb

approx 1/1000 nucleotide different for any two copies of the human genome
↳ polymorphism

Sequencing Primer (dumy vector) — sequence — 3'
Sanger: detector (hydroxyl group) (ddNTPs) nucleotides + radioactivity
→ gel

Capillary: fluorescent dyes and laser reads

NGS: microarray slide
↳ "pixels"



3,000 and 3 days
1 G spots/slide
~ 300 reads/spot

Overlap detection

Seq1 = $(x_1, x_2, \ldots, x_{300})$
Seq2 = $(y_1, y_2, \ldots, y_{300})$

	x_1	x_2 ..	x_{300}
y_1	1	1	0
y_2	0	0	1
⋮			
y_{300}	1	0	0

match matrix

~ 10^5 ones

10^7 reads → 1-fold coverage
random assembly, needs excess coverage → **100-fold coverage**

$(10^9 \times 10^9$ fragments matrix$)$ $10^9 \times 10^9 \times 10^5 = 10^{23}$ operations ~ 100 z flops
tens of days on fastest machine

byte = 8bits letter = 2bits **4 letters ← 1 byte**
1 Gb for entire genome

$4^{10} = 2^{20} \approx 1$ million possible 10-mers of 4 letters

AAAAAAAAAA
AAAAAAAAAC
⋮

table index
||
hashtable

↳ 290 distinct 10-mers for each 300 b read

If two seqs overlap, there will be multiple seqs in the hashtable that match both seqs. The two seqs do not have to match all of the same seqs in the hash table because there will be regions that do not overlap within the 300 nucleotide read.

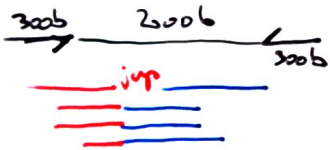
This will reduce the computation 1 million-fold, by aligning only analog seqs. 10-mers are also fast, with 8-mers requiring bigger hash table but leading to better certainty, though, small errors can be minimized using 10-mers.

Transposable elements = repeats ~15% of genome

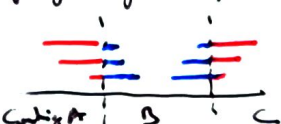
300 b SINES → ~ million copies randomly distributed

7000 b LINES → 60,000
(long interspersed elements)

We can identify repetitive seqs on the hash table, looking at unexpected frequencies, to remove from the overlap analysis → gaps in genome.



= jumping fragments / clones



If the mRNA is transcribed then the region of the gDNA is more likely to be significant to changes in regions that are not transcribed ↳ disease

jumping clone

mRNA → cDNA ← matching genome

Sequence Analysis using Python

hg38.fa.zip : July 2007 NCBI 37/mm9

```
>chr6
xxxxxxxxxx...
>chr11
xxxxxxx...
```

genome-euro.ucsc.edu

mm9_sib_drams_knownGene.txt

name	chrom	strand	txStart	txEnd	cdsStart	cdsEnd	exonCount	exonStarts	exonEnds	proteinID ✓	alignID
uc009aww.1	chr6	+	3638518	3662019	3638878	xxx	3	xxx,xxx,xxx,	xxx,xxx,xxx,	xx	uc009aww.1

?

GenePred table format

table genePred
"A gene prediction."
(
- string name; "Name of gene"
- string chrom;
- char[1] strand;
- uint txStart;
- uint txEnd;
- uint cdsStart;
- uint cdsEnd;
- uint exonCount;
- uint[exonCount] exonStarts;
- uint[exonCount] exonEnds;

)

Schema for knownGene:

genePredExt +
(
- string proteinId; "UniProt display ID, UniProt accession, or RefSeq protein ID"
- string alignId; "unique identifier (GENCODE transcript ID)"

)