Fundamentals of Big Data Sequence Analysis

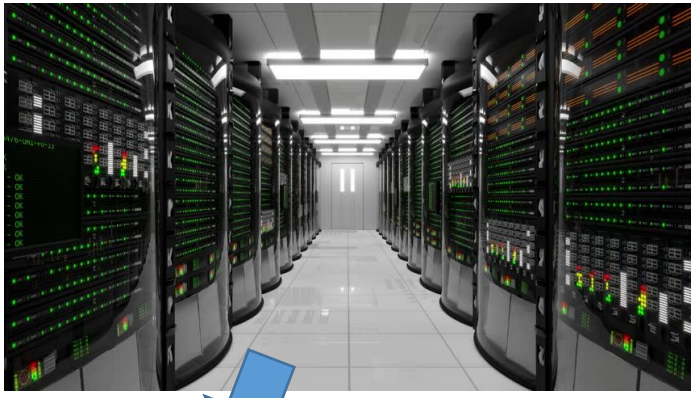
Kirill Kryukov
National Institute of Genetics

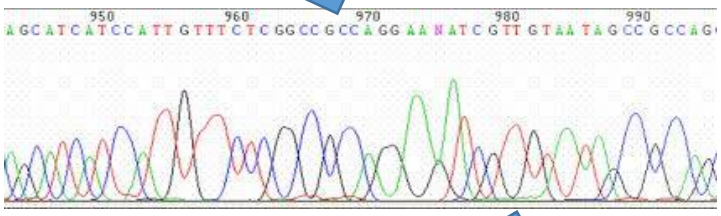
Applications of DNA sequencing

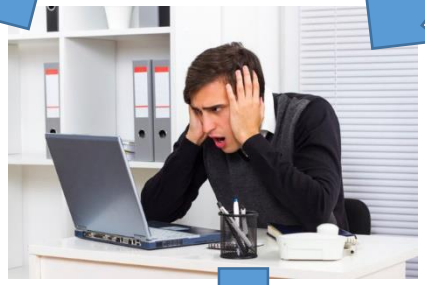
- Agriculture: Engineering varieties of crops with beneficial properties such as resistance to pests.
- Human genetics: detecting genes that are linked to genetic disorders.
- Evolution: Understanding history and relationships of species.
- Forensics: Identifying individual based on traces of tissue.
- Medical metagenomics: making accurate diagnosis of infectious diseases.

Sequencer

Sequence Database





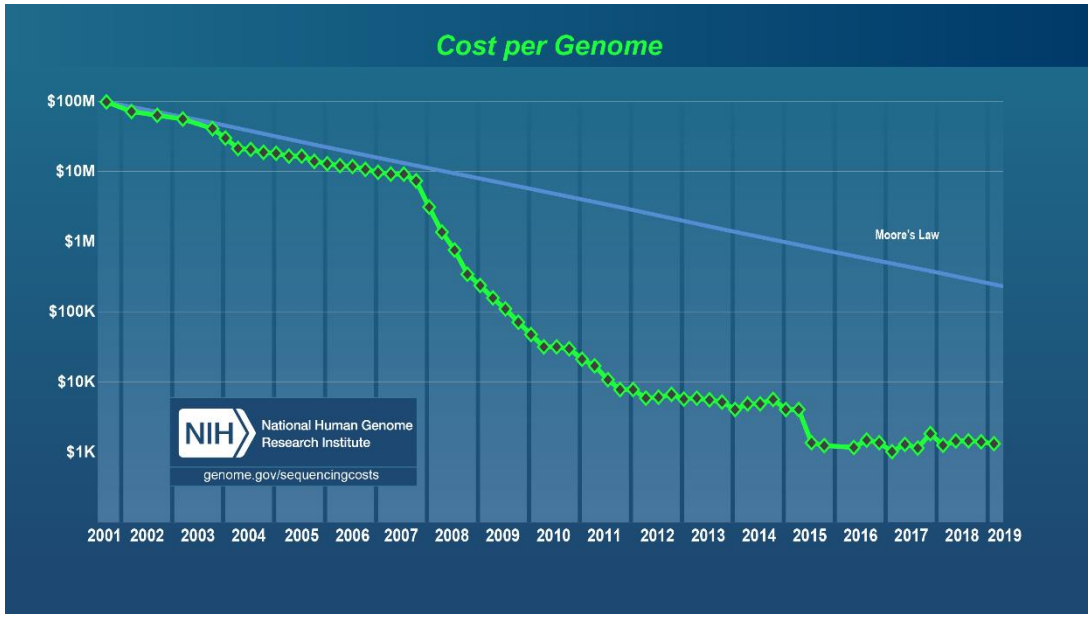


...

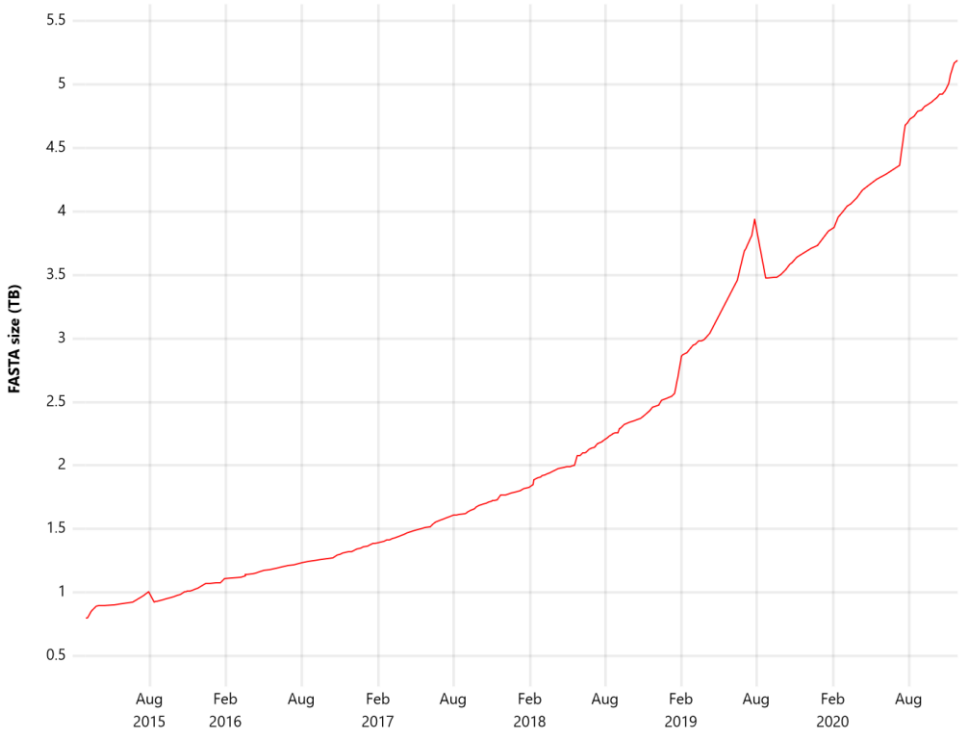
Results

Sequence data size

Sequencing cost is decreasing



Sequence size in the GenomeSync database http://genomesync.org/



Principles of big data analysis

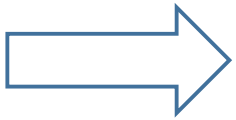
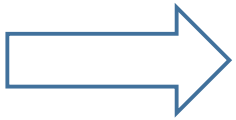
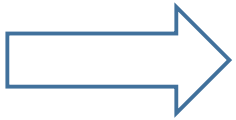
Scalability

Scriptability

Reproducibility

Parallel computing

In practice

Mouse clicks		Keypresses
GUI		Command line
Manual steps		Scripts
Your laptop		Unix machine

Data science, expectations:

Data science, reality

edit seq-split-to-lines.c - Far 3.0.5354 x64

C:\prg\seq-tools\seq-tools\src\seq-split-to-lines.c 1252 Ln 8/82 Col 1 Ch 1

```c
#include "common.c"


static unsigned long long line_length = 0ull;


static void done(void)
{
    free_in_buffer();
}


static void process(void)
{
    unsigned long long line_rem = 0ull;

    for (;;)
    {
        in_begin = 0;
        in_end = fread(in_buffer, 1, in_buffer_size, stdin);
        if (in_end == 0) { break; }

        if (line_rem > 0)
        {
            unsigned long long len1 = in_end - in_begin;
            if (len1 > line_rem) { len1 = line_rem; }
            fwrite(in_buffer, 1, len1, stdout);
            fputc(10, stdout);
            in_begin += len1;
            line_rem -= len1;
        }

        for (size_t i = in_begin + line_length; i <= in_end; i += line_length)
        {
            fwrite(in_buffer + in_begin, 1, line_length, stdout);
            fputc(10, stdout);
            in_begin = i;
        }

        if (in_begin < in_end)
        {
            unsigned long long len1 = in_end - in_begin;
            fwrite(in_buffer + in_begin, 1, len1, stdout);
            line_rem = line_length - len1;
        }
    }

    if (line_rem != 0 && line_rem != line_length) { fputc(10, stdout); }
}


int main(int argc, char **argv)
{
    atexit(done);
```

1Help 2Save 3 4Quit 5 6View 7Search 8OEM 9 10Quit 11Plugin 12Screen

Concepts (keywords)

Terminal (console, shell, command line)

Filesystem, file, directory, relative and absolute path

Commands and arguments, IO redirection

Permissions

File managers and text editors

Shell scripts

Programming languages