Finding immune receptors in RNA-seq data

Li Song

Department of Biomedical Data Science

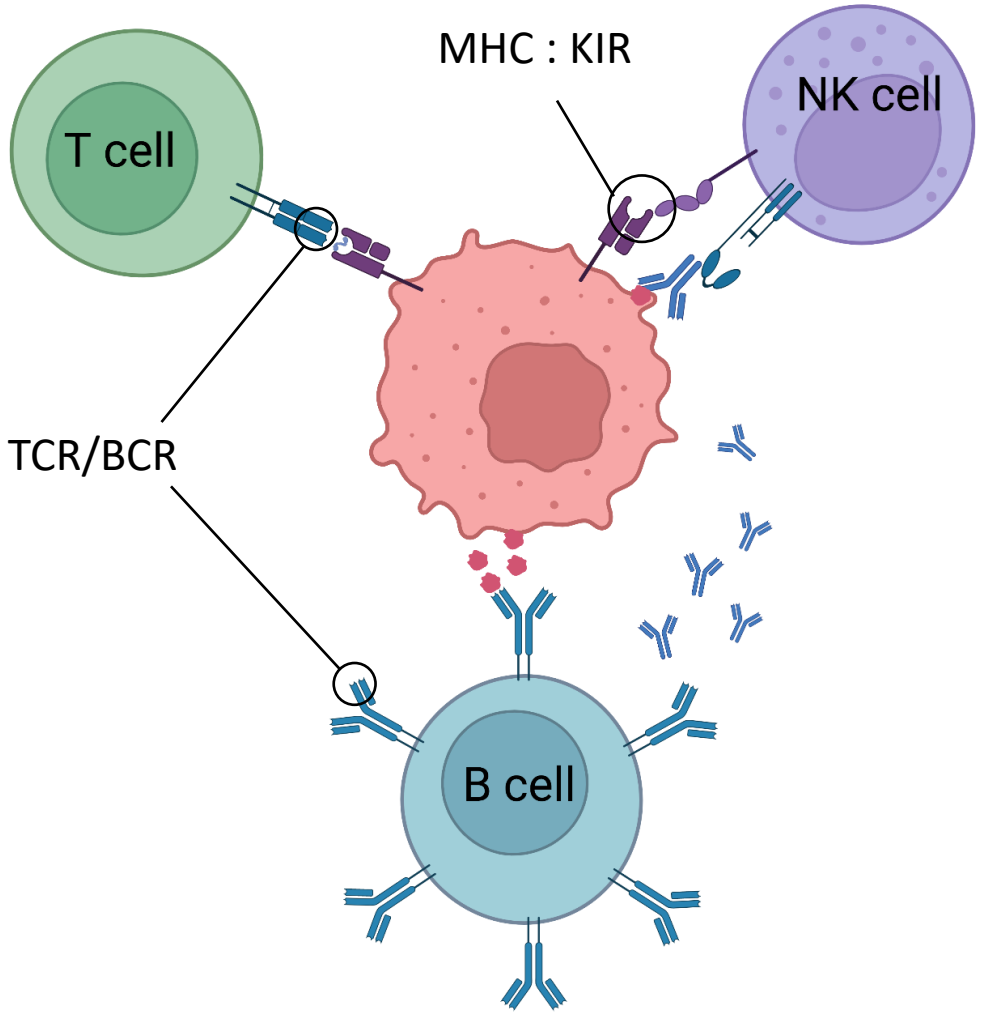
Geisel School of Medicine at Dartmouth

Dartmouth
GEISEL SCHOOL OF
MEDICINE

mourisl.github.io

Twitter @mourisl

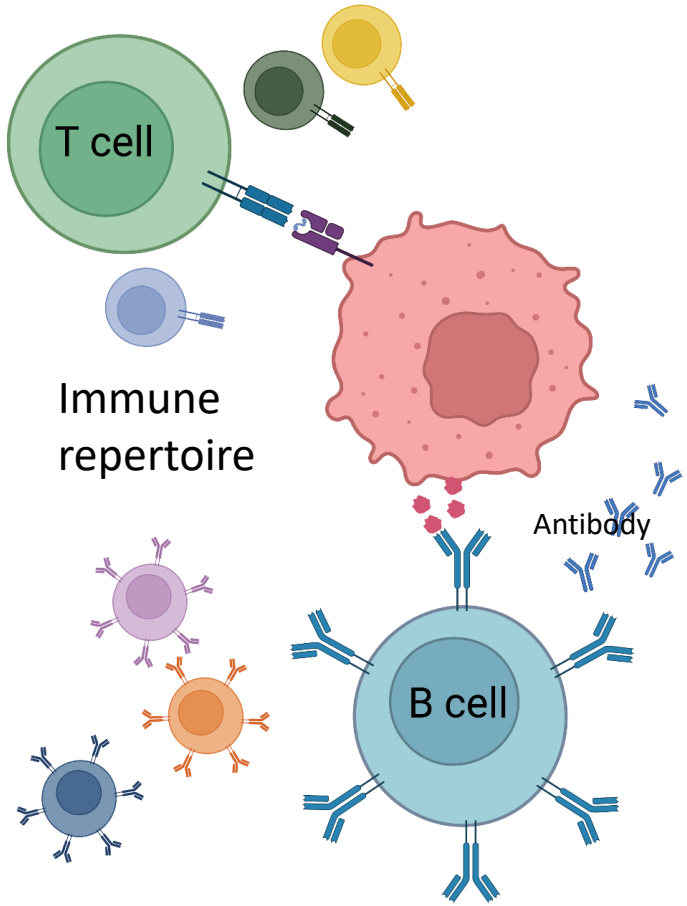
T cell, B cell and NK cell play central roles in immune system

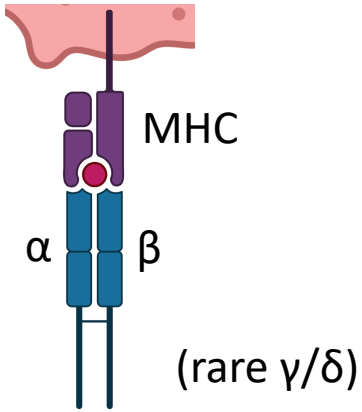


Outline

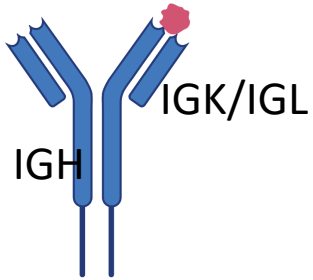
- Immune repertoire assembly: TRUST4
- HLA and KIR genotyping: T1K

T-cell receptors and B-cell receptors have diverse sequences to recognize various antigens



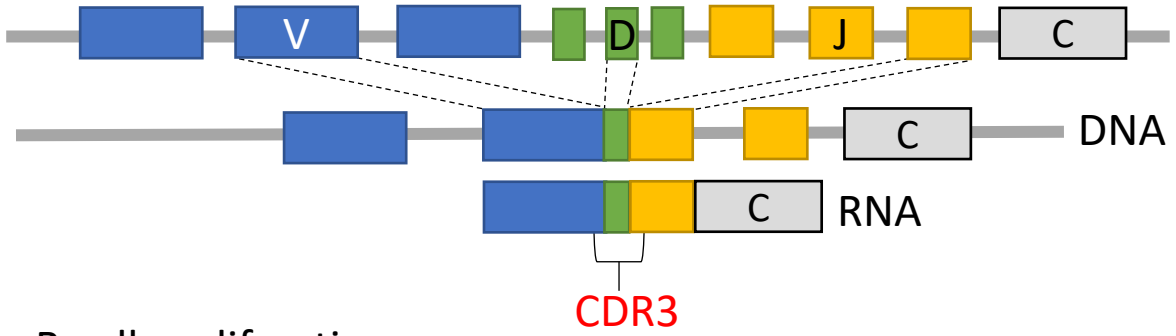


T-cell receptor (TCR)

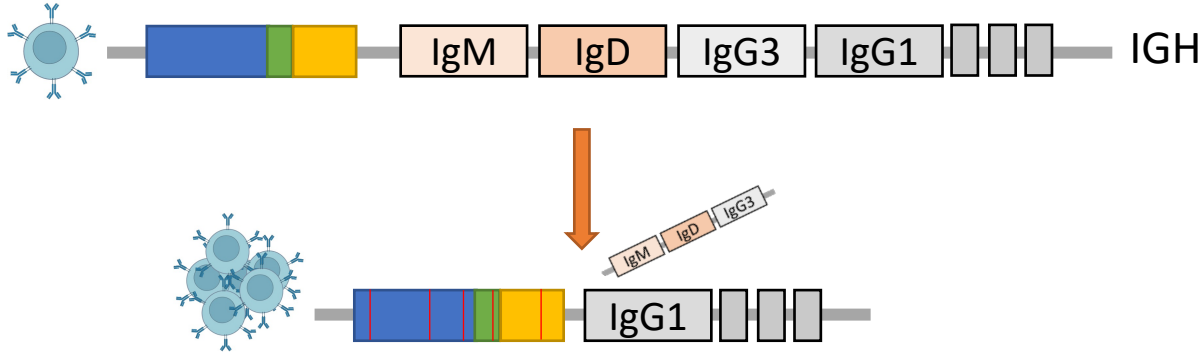


B-cell receptor (BCR)

V(D)J recombination

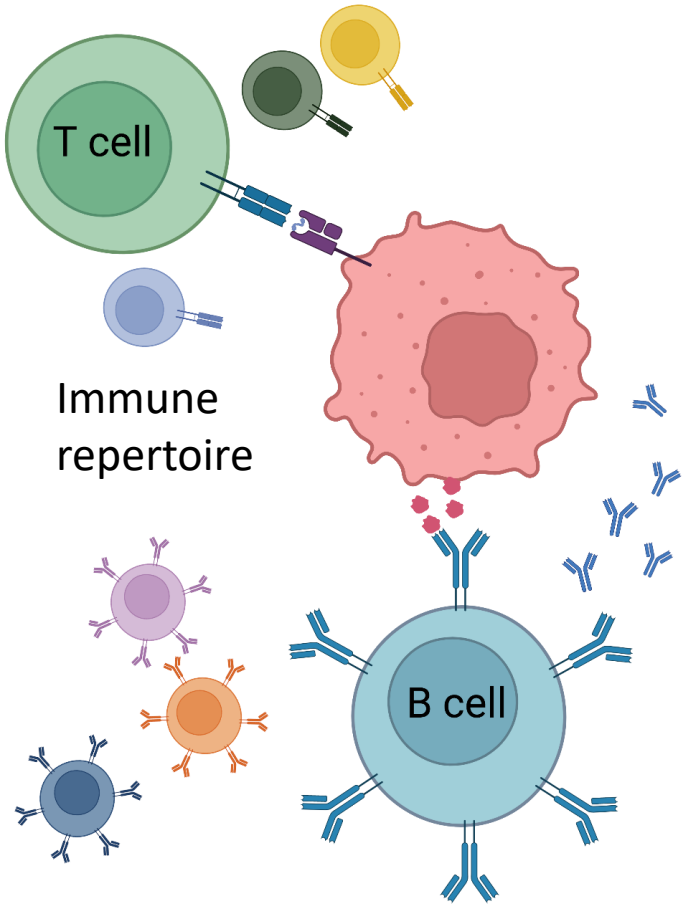


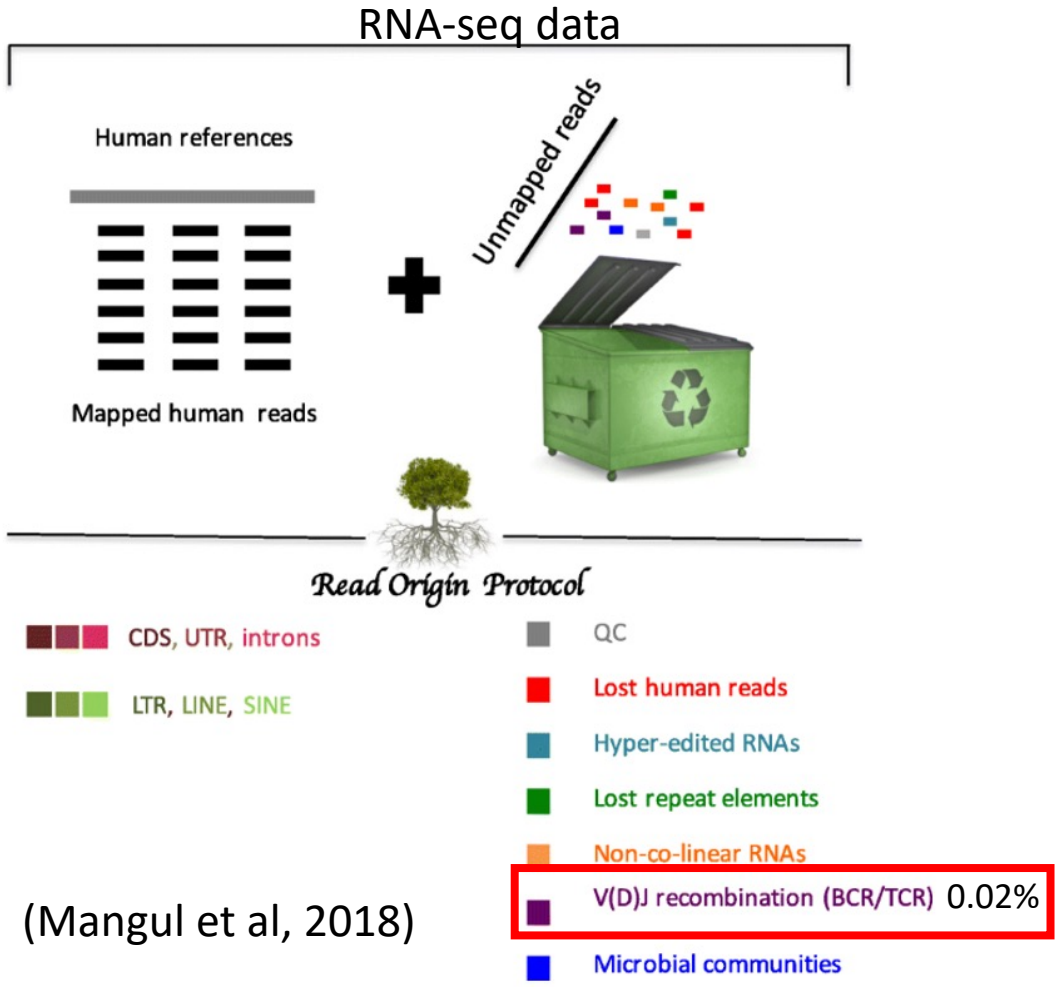
B cell proliferation



Somatic hypermutations (SHM) Class switch recombination (CSR)

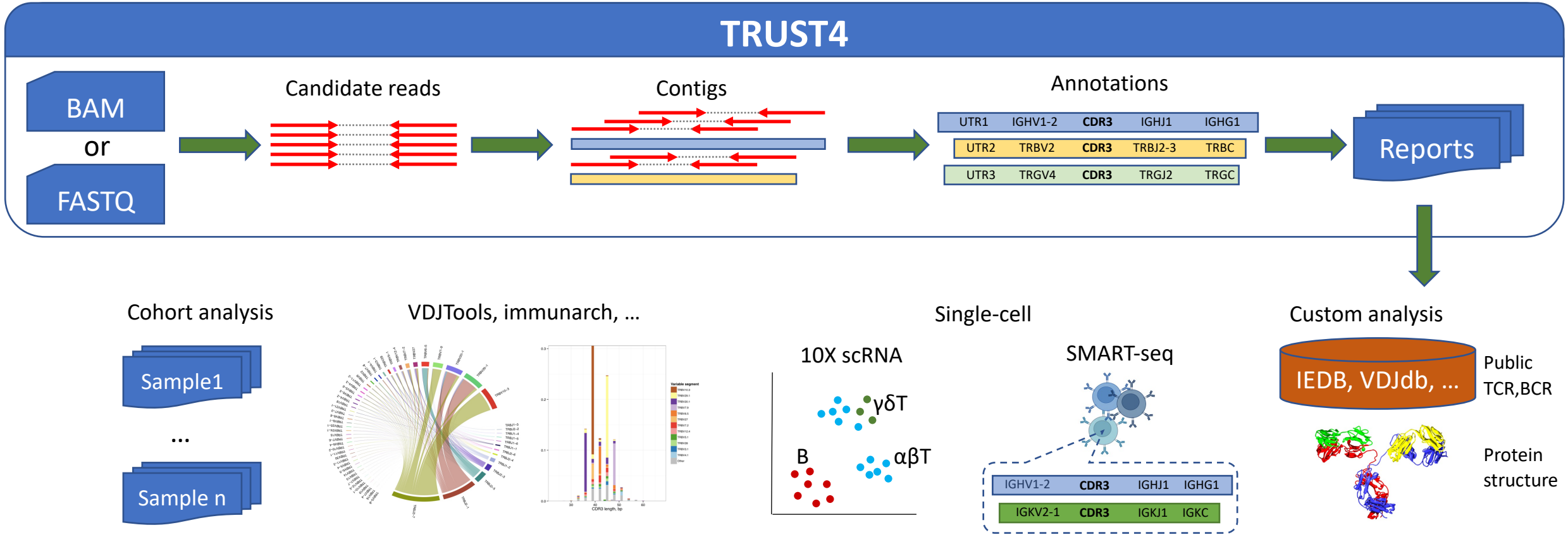
TCR and BCR sequences are presented in RNA-seq data





(Mangul et al, 2018)

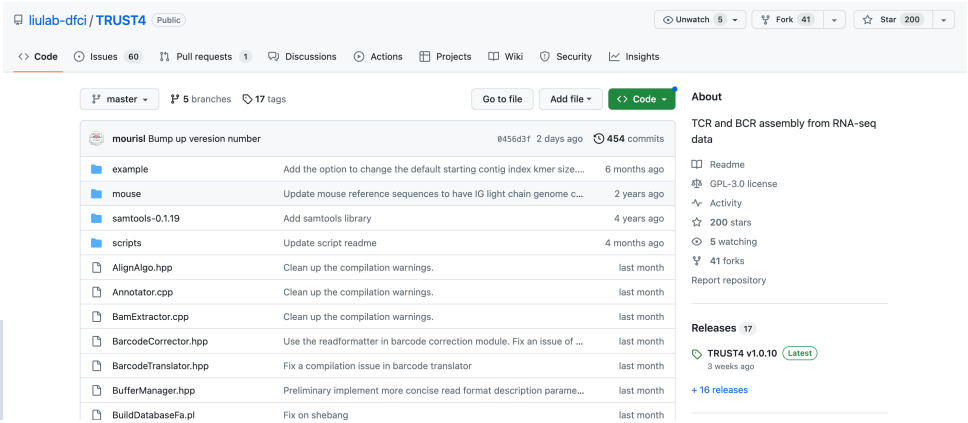
TRUST4: TCR and BCR assembly from RNA-seq data



TRUST4: Song et al, Nature Methods, 2021

Step 1: Download TRUST4

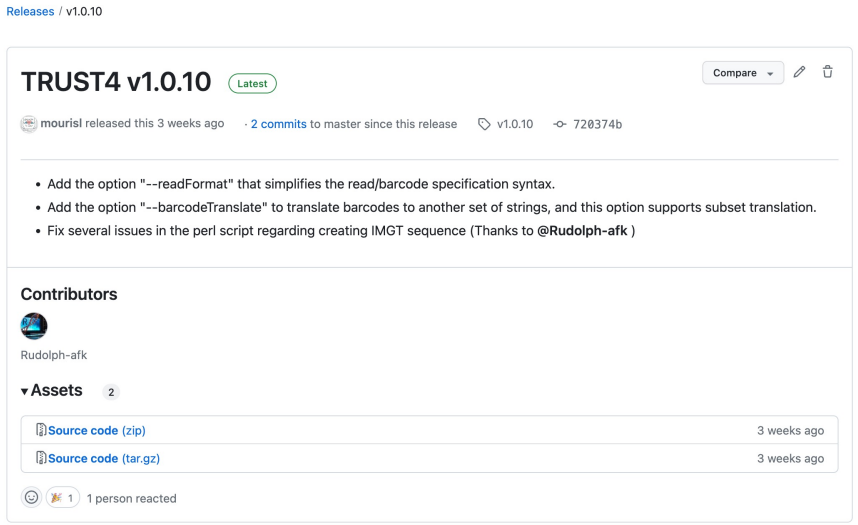
- https://github.com/liulab-dfci/TRUST4



Command line/terminal:

```
git clone https://github.com/liulab-dfci/TRUST4.git
```

Through “Release” page:



Step 2: Compile TRUST4

After cloning github repository or decompressing the package, we need to compile the program

- Make

```
cd trust4_path
make
```

- Conda (downloand+install)

```
conda install -c bioconda trust4
```

Step 3: Run TRUST4

- Suppose TRUST4 is compiled in the folder trust4_path
- Run

```
trust4_path/run-trust4 \
        -1 read1.fq.gz -2 read2.fq.gz  \
        -f hg38_bcrtcr.fa \
        --ref  human_IMGT+C.fa \
        -t 8 \
        -od output_directory \
        -o output_prefix
```

or ”–b alignment.sorted.bam”

Reference sequences

Number of threads

Output path

Checking TRUST4's output

- TRUST4's format (VDJTools format): trust4_report.tsv

#count	frequency	CDR3nt	CDR3aa	V	D	J	C	cid	cid_full_length
34	3.11E-02	GGTCGGCCT	CASSLSPGRPNTGELFF	TRBV28*01	TRBD2*01	TRBJ2-2*01	TRBC	assemble353	1

- AIRR format (https://www.antibodysociety.org/the-airr-community/)
  - prefix_airr.tsv
  - Many data fields, including germline sequence alignment information
  - Ideal for advanced BCR analysis

Application: repertoire diversity analysis

Clonality = 1 - normalized Shannon entropy = 1 – ShannonEntropy/log(N)

$$H(X) = -\sum_i P(x_i) \log P(x_i)$$

Sample I

CDR3 A

CDR3 A

CDR3 A

CDR3 A

CDR3 B

CDR3 C

Entropy=-2/3log(2/3)-1/6log(1/6)-1/6log(1/6) =0.867
Clonality = 1 – 0.867/log3 = 0.210

Sample II

CDR3 A

CDR3 A

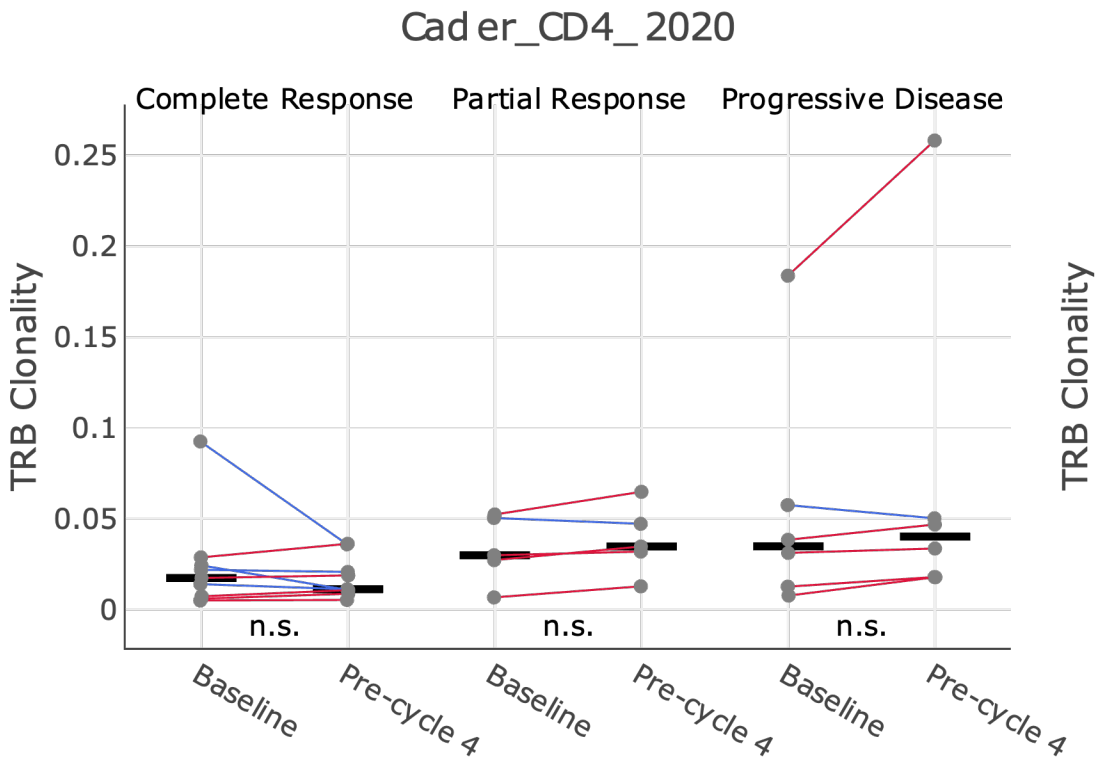
CDR3 B

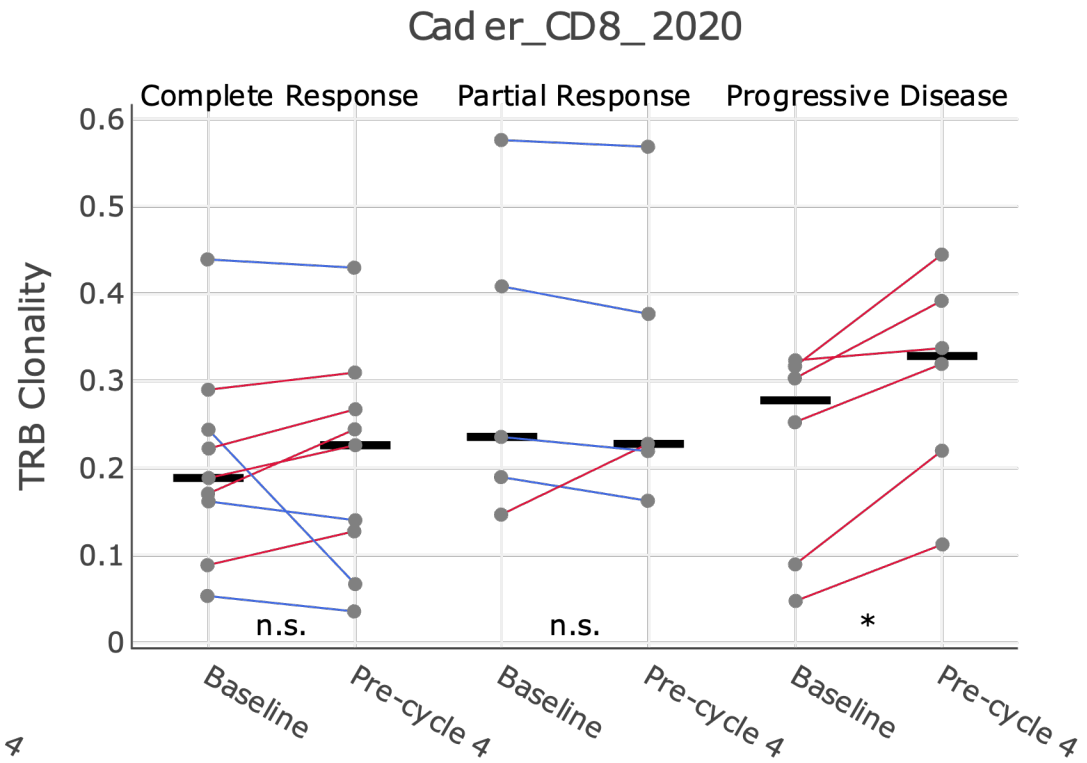
CDR3 B

CDR3 B

Entropy=-2/5log(2/5)-3/5log(3/5)=0.673
Clonality = 1 – 0.673/log2 = 0.029

Application: repertoire diversity analysis on a classic Hodgekin Lymphoma study





Based on Cader et. al. Nat. Med. 2020

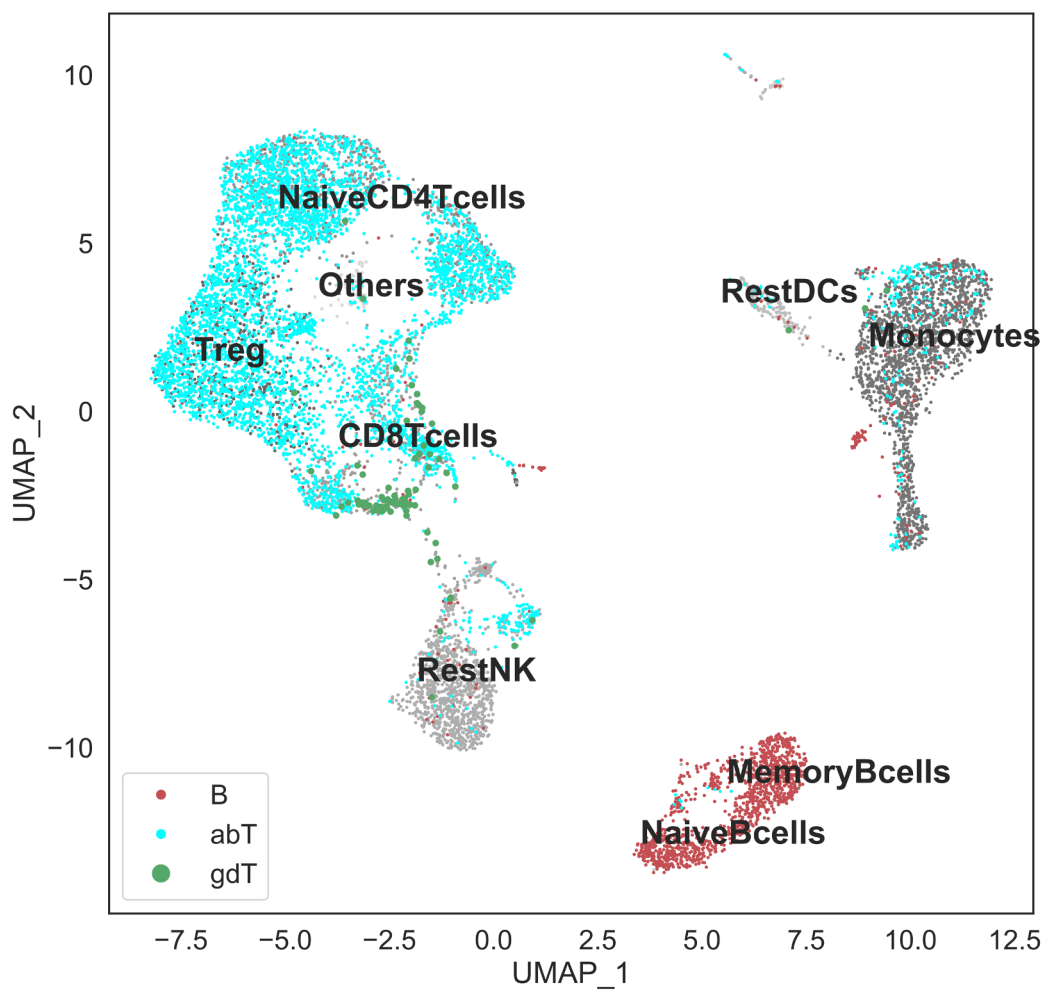
CD8 T cell is not the effector cell in the MHC-I deficient classic Hodgkin's Lymphoma anti-PD1 treatment

Song et al, Cancer Immunology Research, 2022

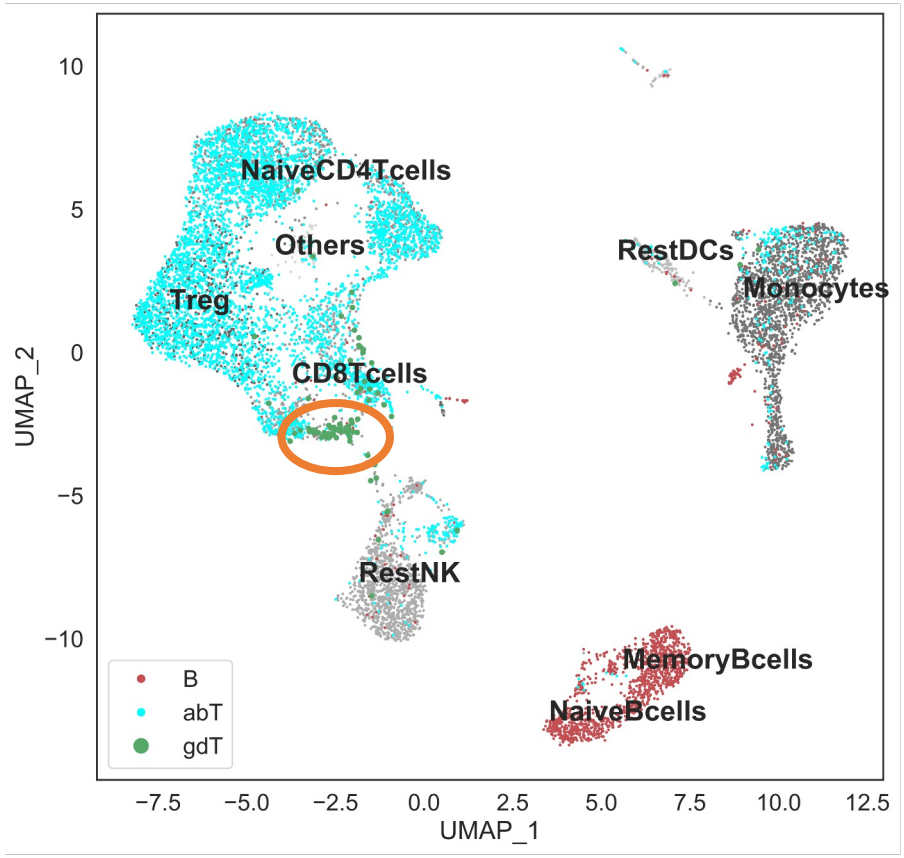
Other downstream analysis related to TCRs

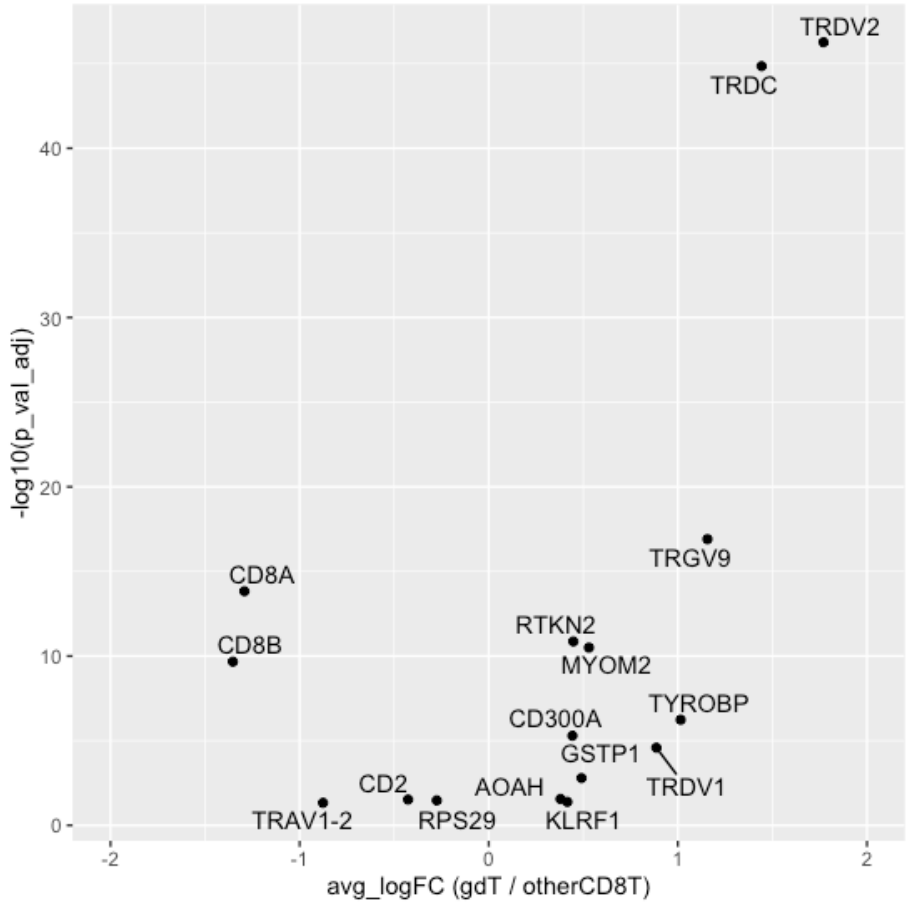
- Antigen-specific TCR clustering
  - GIANA (Zhang et al, 2021)
  - GLIPH2 (Huang et al, 2020)
  - TCRDist (Dash et al, 2017)
- In silico TCR specificity annotation
  - Database: VDJdb, IEDB, McPAS-TCR
  - Method: TcrMatch (Chronister et al, 2021), …
- TCR-epitope binding prediction
  - pMTnet (Lu et al, 2021), …
- TCR function prediction
  - DeepCAT (Beshnova et al, 2020), DeepTCR (Sidhom et al., 2021), …

TRUST4 reconstructs thousands of CDR3s from 5’ 10x single-cell RNA-seq data



TRUST4 identifies the gdT population





10X Genomics V(D)J does not have the kit to amplify gdT cells

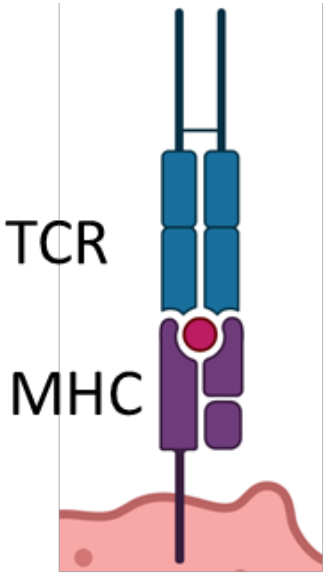
Summary – TRUST4

- An efficient and accurate method to assemble TCRs and BCRs form bulk and single-cell RNA-seq data
- How to run TRUST4?
  - git clone
  - make
  - /path/run-trust4 [-b alignment.sorted.bam] or [-1 r1.fq.z -2 r2.fq.gz] + reference sequences + other options

Outline

- Immune repertoire assembly: TRUST4
- HLA and KIR genotyping: T1K

Introduction: Major Histocompatibility Complex (MHC) class I



HLA-A

HLA-B

HLA-C

(Human Leukocyte Antigen)

IPD-IMGT/HLA

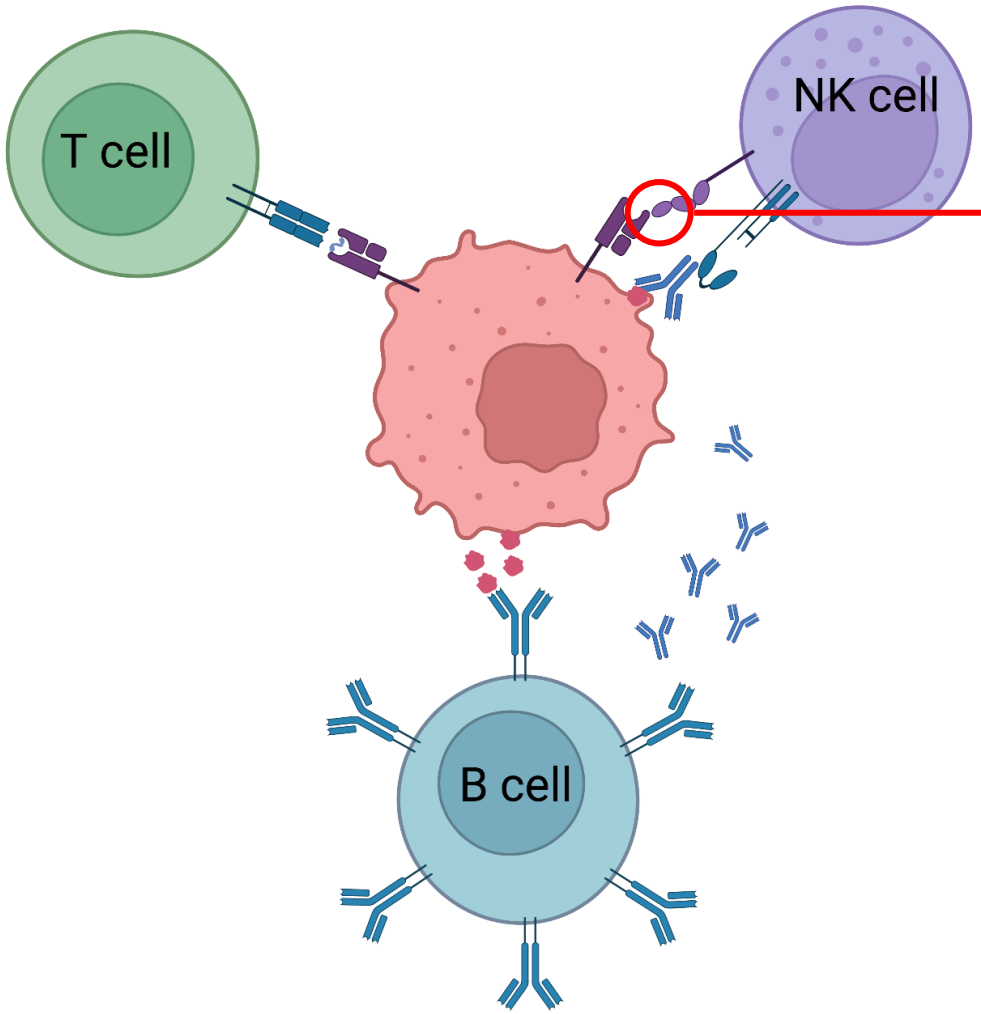
HLA-A:

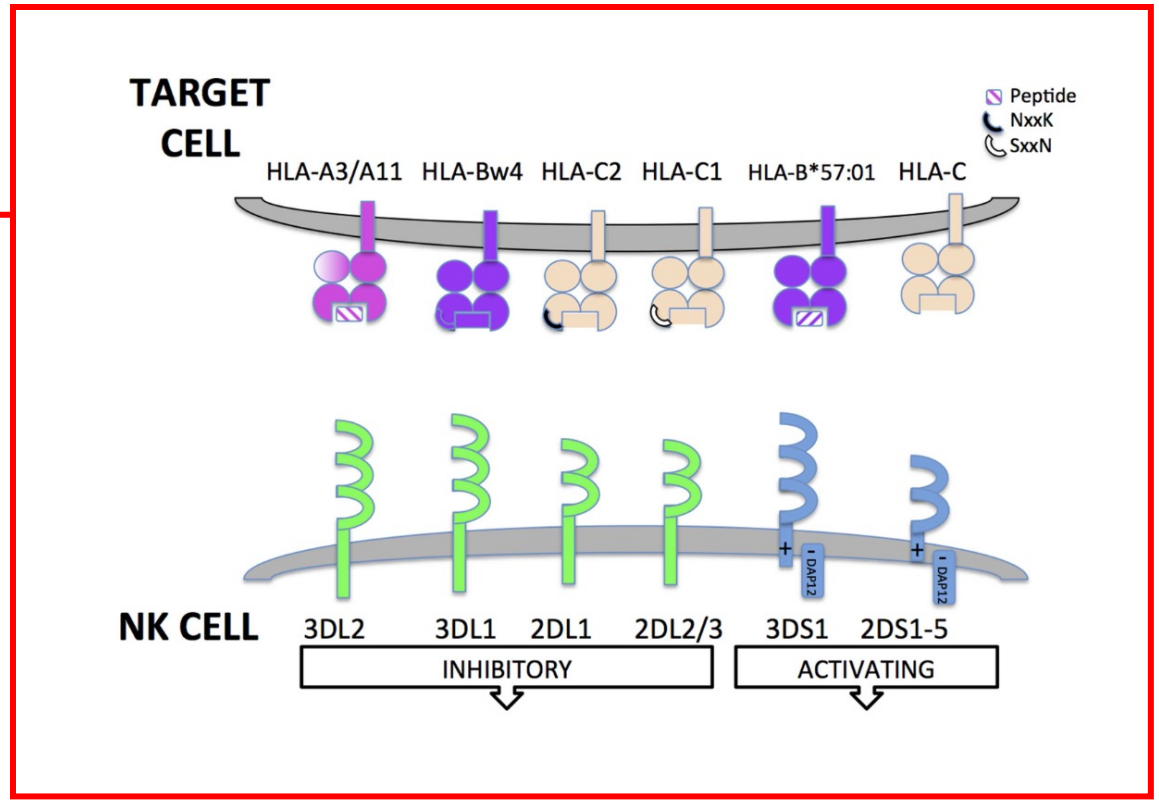
A*01:01:01:01

A*01:01:02:01 — C

A*01:02:01:01 — A — G

KIR is a set of polymorphic genes regulating Natural Killer cell activity

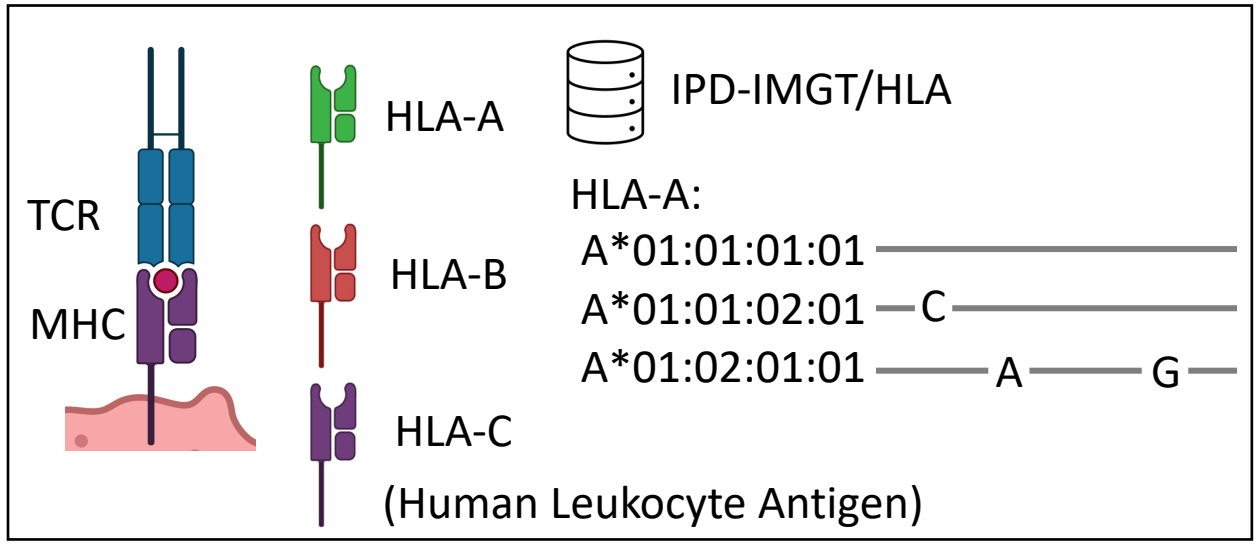




KIR: killer-cell immunoglobulin-like receptor

(Gardiner, 2015)

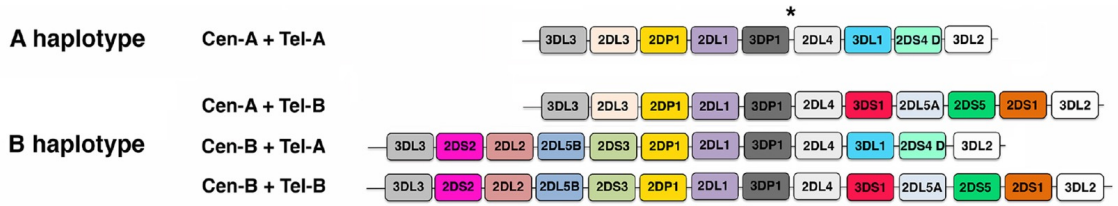
HLA genotypers are not applicable to KIRs

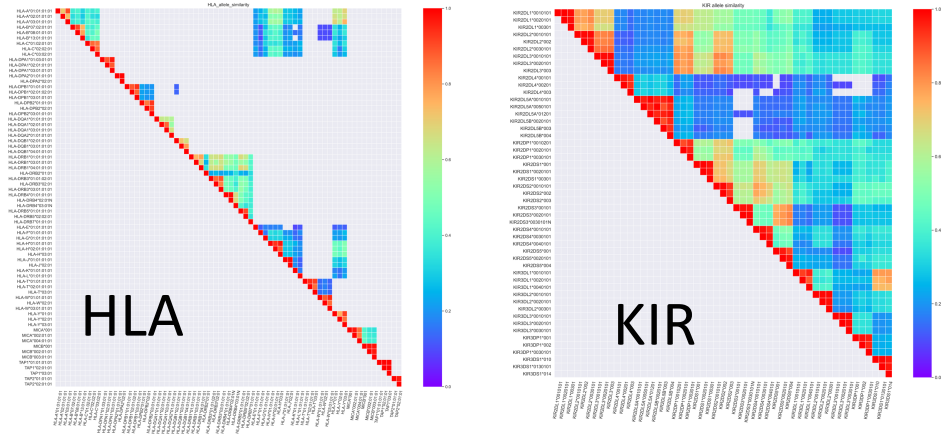


HLA genotypers:

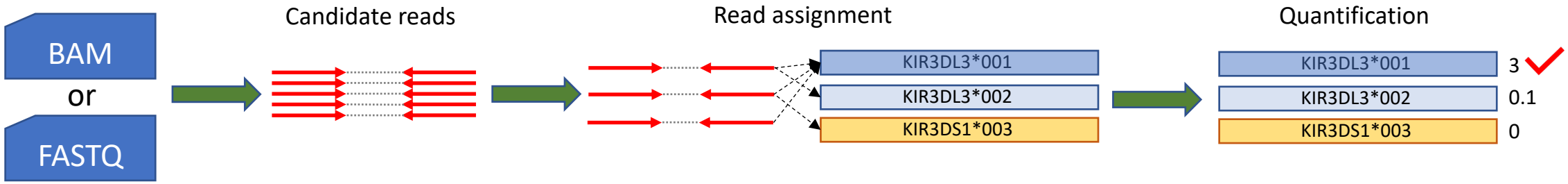
- HISAT-genotype, arcasHLA, OptiType, polysolver, seq2HLA, …

1. Each HLA gene show up in both chromosomes, but a KIR gene can be missing.
2. Many KIR genes are highly similar.





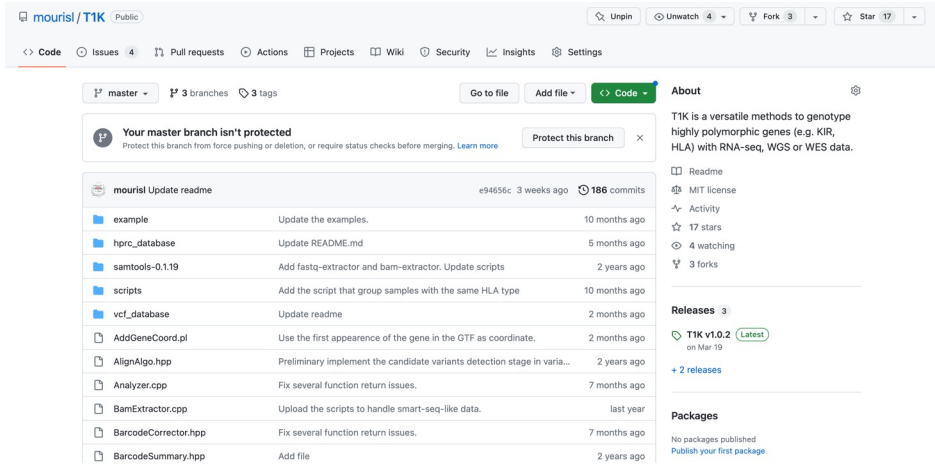
T1K: HLA and KIR genotyping with massive parallel sequencing data



T1K: Song et al, Genome Research, 2023

Step 1: Download T1K

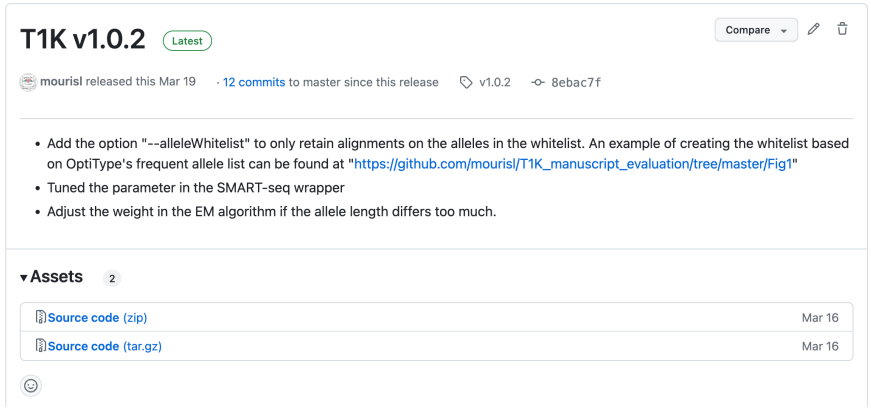
- https://github.com/mourisl/T1K



Command line/terminal:

```
git clone https://github.com/mourisl/T1K.git
```

Through “Release” page:



Step 2: Compile T1K

After cloning github repository or decompressing the package, we need to compile the program

- Make

```
cd t1k_path
make
```

- Conda (downloand+install)

```
conda install -c bioconda t1k
```

Step 3: Build HLA and KIR reference sequences

Assume you are in T1K's source code folder

```
perl t1k-build.pl -o hlaidx --download IPD-IMGT/HLA
perl t1k-build.pl -o kiridx --download IPD-KIR
```

Output:
hlaidx/hlaidx_rna_seq.fa

```
>HLA-A*01:01:01:01 8 50 122 123 392 393 668 669 944 945 1061 1062 1094 1095 1142 1143 1147
TAAAGTCCGCACGCACCCACCGGGACTCAGATTCTCCCCAGACGCCGAGGATGGCCGTCATGGCGCCCCGAACCCTCCTCCTGCTACTCTCGGGG
GCCCTGGCCCTGACCCAGACCTGGGCGGGCTCCCACTCCATGAGGTATTTCTTCACATCCGTGTCCCGGCCCGGCCGCGGGGAGCCCCGCTTCAT
CGCCGTGGGCTACGTGGACGACACGCAGTTCGTGCGGTTCGACAGCGACGCCGCGAGCCAGAAGATGGAGCCGCGGGCGCCGTGGATAGAGCAGG
AGGGGCCGGAGTATTGGGACCAGGAGACACGGAATATGAAGGCCCACTCACAGACTGACCGAGCGAACCTGGGGACCCTGCGCGGCTACTACAAC
CAGAGCGAGGACGGTTCTCACACCATCCAGATAATGTATGGCTGCGACGTGGGGCCGGACGGGCGCTTCCTCCGCGGGTACCGGCAGGACGCCTA
CGACGGCAAGGATTACATCGCCCTGAACGAGGACCTGCGCTCTTGGACCGCGGCGGACATGGCAGCTCAGATCACCAAGCGCAAGTGGGAGGCGG
TCCATGCGGCGGAGCAGCGGAGAGTCTACCTGGAGGGCCGGTGCGTGGACGGGCTCCGCAGATACCTGGAGAACGGGAAGGAGACGCTGCAGCGC
ACGGACCCCCCCAAGACACATATGACCCACCACCCCATCTCTGACCATGAGGCCACCCTGAGGTGCTGGGCCCTGGGCTTCTACCCTGCGGAGAT
CACACTGACCTGGCAGCGGGATGGGGAGGACCAGACCCAGGACACGGAGCTCGTGGAGACCAGGCCTGCAGGGGATGGAACCTTCCAGAAGTGGG
CGGCTGTGGTGGTGCCTTCTGGAGAGGAGCAGAGATACACCTGCCATGTGCAGCATGAGGGTCTGCCCAAGCCCCTCACCCTGAGATGGGAGCTG
TCTTCCCAGCCCACCATCCCCATCGTGGGCATCATTGCTGGCCTGGTTCTCCTTGGAGCTGTGATCACTGGAGCTGTGGTCGCTGCCGTGATGTG
GAGGAGGAAGAGCTCAGATAGAAAAGGAGGGAGTTACACTCAGGCTGCAAGCAGTGACAGTGCCCAGGGCTCTGATGTGTCTCTCACAGCTTGTA
AAGTGTGAGACAGCTGCCTTGTGTGGGACTGAGAGGCAAGAGTTGTTCCTGCCCTTCC
>HLA-A*01:01:01:02N 8 50 122 123 392 393 668 669 944 945 1061 1062 1094 1095 1142 1143 1147
TAAAGTCCGCACGCACCCACCGGGACTCAGATTCTCCCCAGACGCCGAGGATGGCCGTCATGGCGCCCCGAACCCTCCTCCTGCTACTCTCGGGG
GCCCTGGCCCTGACCCAGACCTGGGCGGGCTCCCACTCCATGAGGTATTTCTTCACATCCGTGTCCCGGCCCGGCCGCGGGGAGCCCCGCTTCAT
CGCCGTGGGCTACGTGGACGACACGCAGTTCGTGCGGTTCGACAGCGACGCCGCGAGCCAGAAGATGGAGCCGCGGGCGCCGTGGATAGAGCAGG
AGGGGCCGGAGTATTGGGACCAGGAGACACGGAATATGAAGGCCCACTCACAGACTGACCGAGCGAACCTGGGGACCCTGCGCGGCTACTACAAC
CAGAGCGAGGACGGTTCTCACACCATCCAGATAATGTATGGCTGCGACGTGGGGCCGGACGGGCGCTTCCTCCGCGGGTACCGGCAGGACGCCTA
CGACGGCAAGGATTACATCGCCCTGAACGAGGACCTGCGCTCTTGGACCGCGGCGGACATGGCAGCTCAGATCACCAAGCGCAAGTGGGAGGCGG
TCCATGCGGCGGAGCAGCGGAGAGTCTACCTGGAGGGCCGGTGCGTGGACGGGCTCCGCAGATACCTGGAGAACGGGAAGGAGACGCTGCAGCGC
ACGGACCCCCCCAAGACACATATGACCCACCACCCCATCTCTGACCATGAGGCCACCCTGAGGTGCTGGGCCCTGGGCTTCTACCCTGCGGAGAT
CACACTGACCTGGCAGCGGGATGGGGAGGACCAGACCCAGGACACGGAGCTCGTGGAGACCAGGCCTGCAGGGGATGGAACCTTCCAGAAGTGGG
CGGCTGTGGTGGTGCCTTCTGGAGAGGAGCAGAGATACACCTGCCATGTGCAGCATGAGGGTCTGCCCAAGCCCCTCACCCTGAGATGGGAGCTG
TCTTCCCAGCCCACCATCCCCATCGTGGGCATCATTGCTGGCCTGGTTCTCCTTGGAGCTGTGATCACTGGAGCTGTGGTCGCTGCCGTGATGTG
GAGGAGGAAGAGCTCAGATAGAAAAGGAGGGAGTTACACTCAGGCTGCAAGCAGTGACAGTGCCCAGGGCTCTGATGTGTCTCTCACAGCTTGTA
AAGTGTGAGACAGCTGCCTTGTGTGGGACTGAGAGGCAAGAGTTGTTCCTGCCCTTCC
```

Step 4: Run T1K on RNA-seq data

- Suppose T1k is compiled in the folder t1k_path/
- Run

```
t1k_path/run-t1k \
      -1 read1.fq.gz -2 read2.fq.gz  \
      -f  t1k_path/hlaidx/hlaidx_rna_seq.fa \
      -t 8 \
      --preset hla \
      -od output_directory \
      -o output_prefix
```

Reference sequences

Number of threads

Output path

Step 4’: Run T1K on RNA-seq alignment BAM file

- Create gene coordinate file (in T1K’s folder):

```
perl t1k-build.pl -o hlaidx -d hlaidx/hla.dat -g gencode.gtf
```

- Run

```
t1k_path/run-t1k \
        -b alignment.sorted.bam  \
        -f  t1k_path/hlaidx/hlaidx_rna_seq.fa \
        -c t1k_path/hlaidx/hlaidx_rna_coord.fa \
        -t 8 \
        --preset hla \
        -od output_directory \
        -o output_prefix
```

Reference sequences

Checking T1K's output: HLA

Main output file:

prefix_genotype.tsv

8 Columns

Gene #_of_alleles [allele_id abundance quality]*2

```
HLA-A   2       HLA-A*24:02:01  24258.153833    60      HLA-A*03:01:01  23426.834583    60
HLA-B   1       HLA-B*07:02:01  58600.386825    60      .       0       -1
HLA-C   1       HLA-C*07:02:01  32913.342401    60      .       0       -1
HLA-DMA 1       HLA-DMA*01:01:01        2953.611318     60      .       0       -1
HLA-DMB 1       HLA-DMB*01:01:01        1596.143509     60      .       0       -1
HLA-DOA 2       HLA-DOA*01:01:04        360.755875      60      HLA-DOA*01:01:01        226.582
924     60
HLA-DOB 1       HLA-DOB*01:01:01        535.412144      60      .       0       -1
HLA-DPA1        1       HLA-DPA1*01:03:01       17344.632110    60      .       0       -1
HLA-DPA2        0       .       0       -1      .       0       -1
HLA-DPB1        2       HLA-DPB1*03:01:01       4631.253766     60      HLA-DPB1*02:01:02       3
928.267347      60
HLA-DPB2        1       HLA-DPB2*01:01:01,HLA-DPB2*01:01:02,HLA-DPB2*03:01:01   2.283105      -
1
HLA-DQA1        2       HLA-DQA1*05:05:01       9023.403454     60      HLA-DQA1*01:02:01       6
934.813451      60
HLA-DQA2        1       HLA-DQA2*01:07  2.347872        0       .       0       -1
HLA-DQB1        2       HLA-DQB1*06:02:01       5966.252825     60      HLA-DQB1*03:01:01       5
786.512421      60
HLA-DRA 2       HLA-DRA*01:02:03        21035.105076    60      HLA-DRA*01:02:02        18428.8
26171   60
HLA-DRB1        2       HLA-DRB1*13:03:01       15019.855943    60      HLA-DRB1*15:01:01       8
446.712266      60
HLA-DRB2        0       .       0       -1      .       0       -1
HLA-DRB3        1       HLA-DRB3*01:01:02       6419.185453     60      .       0       -1
HLA-DRB4        2       HLA-DRB4*01:01:01,HLA-DRB4*02:01N       3.219101        0       HLA-DRB
4*03:01N        1.197303        0
HLA-DRB5        1       HLA-DRB5*01:01:01       5752.246530     60      .       0       -1
HLA-DRB7        0       .       0       -1      .       0       -1
HLA-E   2       HLA-E*01:01:01  6934.542655     60      HLA-E*01:03:02  6341.428588     60
HLA-F   2       HLA-F*01:03:01  2712.302035     60      HLA-F*01:01:01  2566.856408     60
HLA-G   2       HLA-G*01:01:08  4.253179        0       HLA-G*01:01:05  4.253179        0
HLA-H   1       HLA-H*02:04:01  28.060851       0       .       0       -1
HLA-HFE 1       HLA-HFE*001:01:01       30.950305       28      .       0       -1
HLA-J   1       HLA-J*01:01:01  10.923090       0       .       0       -1
HLA-K   1       HLA-K*01:01:01  54.194630       0       .       0       -1
HLA-L   1       HLA-L*01:01:01  25.919732       0       .       0       -1
MICA    1       MICA*008:04:01  208.359873      60      .       0       -1
MICB    1       MICB*004:01:01  286.102237      60      .       0       -1
HLA-N   0       .       0       -1      .       0       -1
HLA-S   1       HLA-S*01:03     2.997275        3       .       0       -1
HLA-T   0       .       0       -1      .       0       -1
TAP1    1       TAP1*01:01:01   3747.529168     60      .       0       -1
TAP2    2       TAP2*01:02      664.701344      60      TAP2*02:01:02   651.295623      60
HLA-U   2       HLA-U*01:03     23.648649       18      HLA-U*01:01:01  15.878378       11
HLA-V   0       .       0       -1      .       0       -1
HLA-W   0       .       0       -1      .       0       -1
HLA-Y   1       HLA-Y*02:01     8.629081        0       .       0       -1
```

Checking T1K's output: KIR

Main output file:
prefix_genotype.tsv

8 Columns
Gene #_of_alleles [allele_id abundance quality]*2

RNA-seq

```
KIR2DL1 0       .       0       -1      .       0       -1
KIR2DL2 0       .       0       -1      .       0       -1
KIR2DL3 0       .       0       -1      .       0       -1
KIR2DL4 0       .       0       -1      .       0       -1
KIR2DL5A        0       .       0       -1      .       0       -1
KIR2DL5B        0       .       0       -1      .       0       -1
KIR2DS1 0       .       0       -1      .       0       -1
KIR2DS2 0       .       0       -1      .       0       -1
KIR2DS3 0       .       0       -1      .       0       -1
KIR2DS4 1       KIR2DS4*001,KIR2DS4*003,KIR2DS4*006,KIR2DS4*010,KIR2DS4*011,KIR2DS4*016,KIR2DS4
*017,KIR2DS4*018,KIR2DS4*019,KIR2DS4*020,KIR2DS4*021,KIR2DS4*022       1.007049        1       -
1
KIR2DS5 0       .       0       -1      .       0       -1
KIR3DL1 0       .       0       -1      .       0       -1
KIR3DL2 0       .       0       -1      .       0       -1
KIR3DL3 0       .       0       -1      .       0       -1
KIR3DS1 0       .       0       -1      .       0       -1
KIR2DP1 0       .       0       -1      .       0       -1
KIR3DP1 0       .       0       -1      .       0       -1
```

KIR gene may not have enough expression levels in bulk RNA-seq, but can be found in certain cell types in scRNA-seq data

WES

```
KIR2DL1 2       KIR2DL1*002     95.173816       45      KIR2DL1*008     63.407579       25
KIR2DL2 1       KIR2DL2*003     0.543287        0       .       0       -1
KIR2DL3 1       KIR2DL3*002     141.540822      60      .       0       -1
KIR2DL4 1       KIR2DL4*001     262.739867      60      .       0       -1
KIR2DL5A        2       KIR2DL5A*029    1.344724        0       KIR2DL5A*001    0.231960        0
KIR2DL5B        2       KIR2DL5B*004    1.178725        0       KIR2DL5B*007    0.416216        0
KIR2DS1 0       .       0       -1      .       0       -1
KIR2DS2 0       .       0       -1      .       0       -1
KIR2DS3 1       KIR2DS3*020     0.563666        0       .       0       -1
KIR2DS4 1       KIR2DS4*001     170.897986      60      .       0       -1
KIR2DS5 1       KIR2DS5*037     0.990741        0       .       0       -1
KIR3DL1 2       KIR3DL1*015     99.058498       46      KIR3DL1*002     95.569664       44
KIR3DL2 1       KIR3DL2*002     297.099775      60      .       0       -1
KIR3DL3 2       KIR3DL3*001     92.734991       49      KIR3DL3*019     79.553379       40
KIR3DS1 0       .       0       -1      .       0       -1
KIR2DP1 1       KIR2DP1*003     118.987219      55      .       0       -1
KIR3DP1 2       KIR3DP1*015     87.564099       50      KIR3DP1*006     79.971546       45
```

Other highly polymorphic genes, e.g. pharmacogenes

- https://github.com/mourisl/T1K/tree/master/vcf_database

Using CYP2D6 as the example

step 0: prerequisite files

You will need hg38 human reference genome (hg38.fa) and the gene annotation file such as from gencode (gencode.gtf).

step 1: download and process the VCF files

1.1 Click the "Download Complete Database" button at https://www.pharmvar.org/download.

1.2 Uncompress the pharmvar-XXX.zip file to the {T1K_PATH}/vcf_database/, and make {T1K_PATH}/vcf_database/ your current folder. You shall see the folder ./pharmvar-XXX/CYP2D6/ there. Put the hg38 VCF file names to the file by

```
ls ./pharmvar-XXX/CY2D6/GRCh38/*.vcf > vcflist.out
```

1.3 Generate the combined VCF file

```
perl ./CombinedVcf.pl "CYP2D6*1" vcflist.out > cyp2d6_combined.vcf
```

We need the first parameter "CYP2D6*1" because that the VCF files does not contain the primary allele "CYP2D6.1", and we need a place holder for it in the combined VCF file.

step 2: create the reference files

2.1 Generate the EMBL-ENA format dat file by running:

```
perl ./CombinedVcfToDat.pl genome.fa gencode.gtf cyp2d6_combined.vcf > cyp2d6.dat
```

The dat file should look similar to the dat file from IPD-IMGT/HLA and IPD-KIR.

2.2 Generate the reference files

```
perl {T1K_PATH}/t1k-build.pl -d cyp2d6.dat -g gencode.gtf -o cyp2d6_idx --prefix cyp2d6
```

Overall summary – using RNA-seq to find immune-related information

- TRUST4: TCR and BCR reconstruction from RNA-seq
- T1K: genotyping HLA, KIR (depends) and other highly polymorphic genes from common sequencing data
- /path/run-[t1k|trust4] [-b alignment.sorted.bam | -1 r1.fq.z -2 r2.fq.gz] + reference sequences + other options