

# Immune repertoire forensics

## A RepSeq data analysis tutorial

Mikhail Shugay, PhD

Skoltech, MA03172 course [Term 2, 2017-2018]

December 1, 2017

# Outline

Introduction

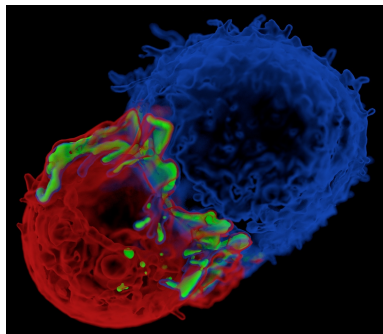
Getting started

Interactive part

The assignment

# T-cell receptor

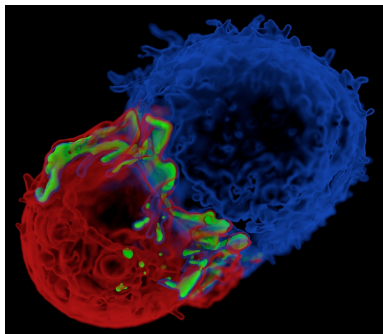
## T-cell:APC contact



From James and Vale, Nature  
2012,  
<https://valelab.ucsf.edu/images/>

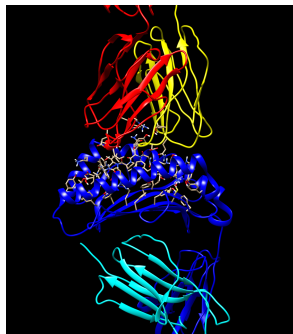
# T-cell receptor

## T-cell:APC contact



From James and Vale, Nature  
2012,  
<https://valelab.ucsf.edu/images/>

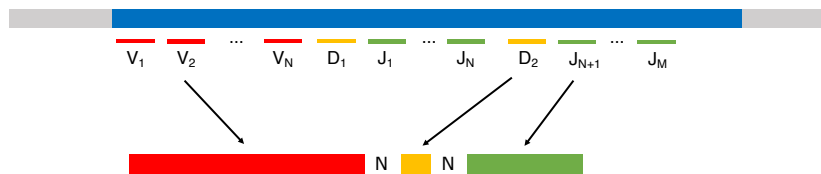
## TCR:pMHC structure



PDB:1ao7, rendered using  
UCSF chimera, colored by  
chain

# VDJ rearrangement

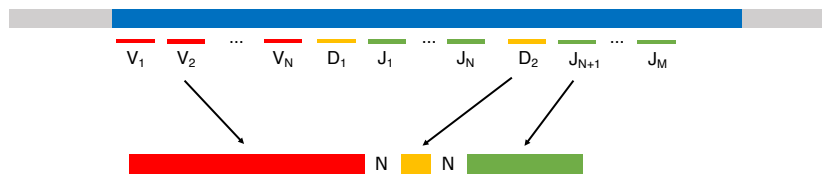
An example schema for TCR $\beta$  locus



Variable, Diversity and Joining are chosen at random, V-D and D-J junctions are filled with non-template N bases.

# VDJ rearrangement

An example schema for TCR $\beta$  locus



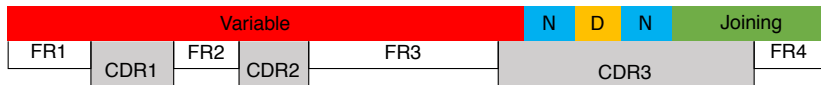
Variable, Diversity and Joining are chosen at random, V-D and D-J junctions are filled with non-template N bases.

VDJ rearrangement mechanism can be efficiently recaptured with a probabilistic model [Murugan et al. PNAS 2012]

$$\begin{aligned} P(\sigma) &= P(V)P(D, J) \\ &\times P(\#del_V|V)P(\#del_J|J)P(\#del_{D5}, \#del_{D3}|D) \\ &\times P(\#ins_{VD})P(\#ins_{DJ}) \prod_{i \in ins_{VD}} P(b_i|b_{i-1}) \prod_{i \in ins_{DJ}} P(b_i|b_{i+1}) \end{aligned}$$

# TCR regions

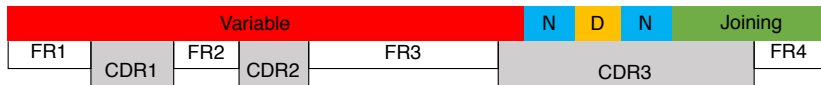
A TCR chains consists of the following regions:



In total there are four framework (FRs) and three complementarity determining regions/loops (CDRs).

# TCR regions

A TCR chains consists of the following regions:



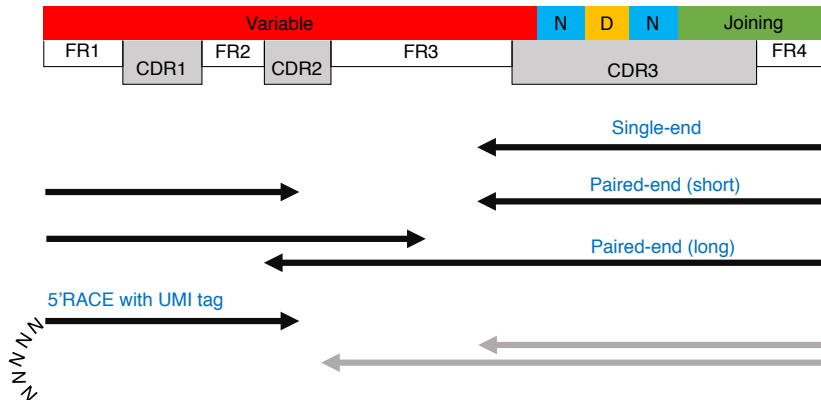
In total there are four framework (FRs) and three complementarity determining regions/loops (CDRs).

The likely functions of these regions are:

- ▶ FR regions maintain TCR secondary structure and (possibly) play role in MHC binding
- ▶ CDR1,2 are germline encoded and play role in antigen recognition, as well as (possibly) MHC binding
- ▶ CDR3 plays a major role in antigen recognition and is extremely variable



# TCR repertoire sequencing



# An example of a RepSeq dataset

After all pre-processing steps:

- ▶ Read grooming (filtering, etc)
- ▶ UMI-based assembly (for molecular barcoded data)
- ▶ V-D-J mapping and clonotype assembly

# An example of a RepSeq dataset

After all pre-processing steps:

- ▶ Read grooming (filtering, etc)
- ▶ UMI-based assembly (for molecular barcoded data)
- ▶ V-D-J mapping and clonotype assembly

We finally get clonotype frequency tables that look like

Index	Frequency	Count	CDR3AA	V	D	J	CDR3NT
1	1.0%	3913	CSA <b>GG</b> L <b>G</b> STDTQYF	TRBV20-1	TRBD1	TRBJ2-3	TGCAGT <b>GCTG</b> <b>G</b> <b>GGGG</b> <b>GC</b> <b>TC</b> GGTAGCACAGATACGCAGTATTTT
2	0.90%	3440	CAS <b>NS</b> GSSYNEQFF	TRBV5-1	TRBD2	TRBJ2-1	TGCGCCAGCA <b>A</b> <b>TAG</b> <b>CGGG</b> <b>AG</b> <b>CT</b> CCTACAATGAGCAGTCTTTC
3	0.79%	3021	CSA <b>R</b> <b>QG</b> N <b>Q</b> PQH <b>F</b>	TRBV20-1	TRBD1	TRBJ1-5	TGCAGT <b>GCGC</b> <b>SAC</b> <b>AGGG</b> <b>GA</b> <b>AAT</b> CAGCCCCAGCATTTT
4	0.65%	2490	CASS <b>Q</b> EP <b>GG</b> EQFF	TRBV4-1	TRBD2	TRBJ2-1	TGCGCCAGCAGCCA <b>AG</b> <b>AG</b> <b>CGGG</b> <b>CGGG</b> <b>GAG</b> CAGTCTTTC
5	0.61%	2336	CASSY <b>G</b> MNTEAFF	TRBV6-6	TRBD2	TRBJ1-1	TGTGCCAGCAGTTAC <b>GGG</b> ATGAACACTGAAGCTTTCTTT
6	0.52%	1992	CASS <b>Q</b> <b>G</b> <b>R</b> <b>A</b> PHTQYF	TRBV4-3	TRBD2	TRBJ2-3	TGCGCCAGCAGCCA <b>AG</b> <b>GG</b> <b>GGG</b> <b>AGGG</b> <b>CCCC</b> ATACGCAGTATTTT
7	0.49%	1871	CASS <b>Q</b> <b>S</b> <b>Q</b> <b>G</b> <b>S</b> YEYQF	TRBV5-1	TRBD1	TRBJ2-7	TGCGCCAGCAGCCA <b>AA</b> <b>AG</b> <b>TCA</b> <b>AGGG</b> <b>GG</b> <b>GT</b> CCTACGAGCAGTACTTC
8	0.48%	1847	CASS <b>R</b> <b>P</b> <b>K</b> <b>S</b> <b>G</b> <b>R</b> <b>S</b> GELFF	TRBV11-2	TRBD2	TRBJ2-2	TGTGCCAGCAG <b>CC</b> <b>G</b> <b>ACCC</b> <b>AA</b> <b>AG</b> <b>AG</b> <b>CGGG</b> <b>AG</b> <b>AA</b> <b>GT</b> GGGAGCTGTTTTTT

# Outline

Introduction

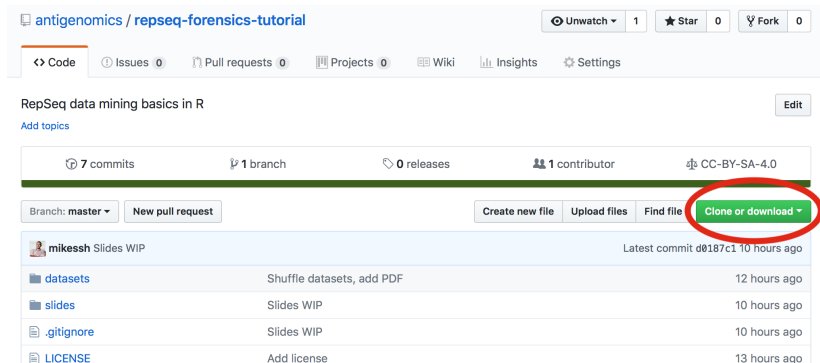
Getting started

Interactive part

The assignment

# Downloading data

Navigate to `https://github.com/antigenomics/repseq-forensics-tutorial` and download the data + code bundle as zip



The screenshot shows the GitHub repository page for `antigenomics / repseq-forensics-tutorial`. The repository is described as "RepSeq data mining basics in R". It has 7 commits, 1 branch, 0 releases, 1 contributor, and is licensed under CC-BY-SA-4.0. The "Clone or download" button is circled in red.

antigenomics / repseq-forensics-tutorial

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

RepSeq data mining basics in R

Add topics

7 commits 1 branch 0 releases 1 contributor CC-BY-SA-4.0

Branch: master New pull request

Create new file Upload files Find file Clone or download

mikessh Slides WIP		Latest commit d0187c1 10 hours ago
datasets	Shuffle datasets, add PDF	12 hours ago
slides	Slides WIP	10 hours ago
.gitignore	Slides WIP	10 hours ago
LICENSE	Add license	13 hours ago

# Executing R code

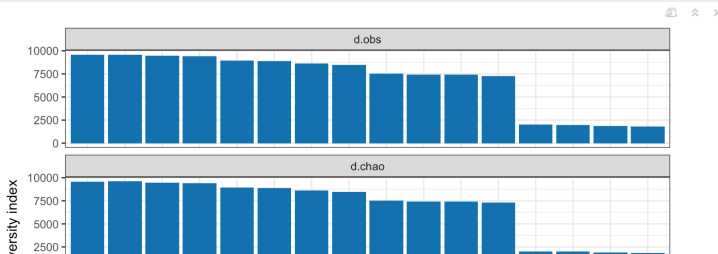
Open the tutorial1.Rmd in RStudio, it can be found in the root folder of the bundle.

```
108 Plot diversity values
109
110 ```{r message=FALSE}
111 diversity %>%
112   melt %>%
113   # set what values we are going to plot
114   # fct_reorder reorders sample id by value
115   ggplot(aes(x=fct_reorder2(sample_id, variable, value), y=value)) +
116   # we'll make a bar plot
117   geom_bar(stat = "identity", fill = "#0570b0") +
118   # show each index on different subplot
119   facet_wrap(~variable, scales = "free_y", ncol = 1) +
120   xlab("") + ylab("Diversity index") +
121   theme_bw()
122 ```
```



# Executing R code

```
108 Plot diversity values
109
110 ```{r message=FALSE}
111 diversity %>%
112   melt %>%
113   # set what values we are going to plot
114   # fct_reorder reorders sample id by value
115   ggplot(aes(x=fct_reorder2(sample_id, variable, value), y=value)) +
116   # we'll make a bar plot
117   geom_bar(stat = "identity", fill = "#0570b0") +
118   # show each index on different subplot
119   facet_wrap(~variable, scales = "free_y", ncol = 1) +
120   xlab("") + ylab("Diversity index") +
121   theme_bw()
122 ```
```



# Outline

Introduction

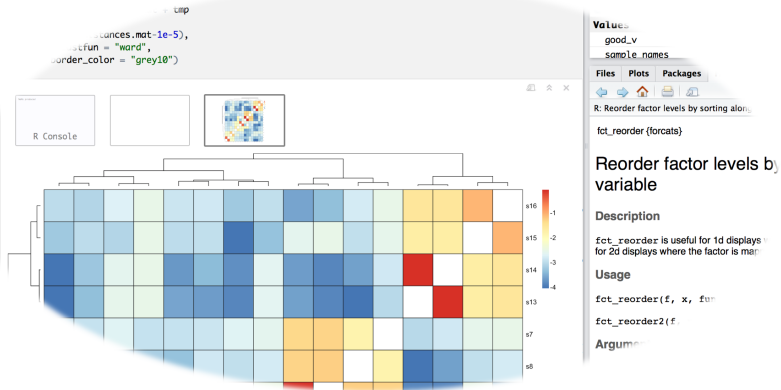
Getting started

Interactive part

The assignment



# Interactive part



# Outline

Introduction

Getting started

Interactive part

The assignment

# The assignment

Using the analysis results we've obtained we need to assign feature labels to each sample. Namely, you need to fill the table with the following structure:

<b>sample</b>	<b>donor</b>	<b>subset</b>	<b>phenotype</b>	<b>CMVstatus</b>
s1	D1	CD4	memory	CMV-
s2	D2		naive	CMV+
s3	D1	CD8	naive	CMV-
...	...	...	...	...

# Details

Table filling rules:

- ▶ Column names should match those on previous slide
- ▶ Sample id should be one of  $s_1..s_{16}$
- ▶ Two distinct donor IDs should be used, naming doesn't matter
- ▶ Subset should be either **CD4** or **CD8**
- ▶ Phenotype should be either **memory** or **naive**
- ▶ CMV status should be either **CMV+** or **CMV-**
- ▶ Unknown/ambiguous fields should be left blank

## A hint

While you can unambiguously assign CD4/8 and memory/naive labels, as well as point out biological replicates of the same sample, assigning donor labels is tricky.

First, it is impossible to link CD4-CD8 cells of the same donor. Same for CMV status, that is unambiguous only for CD8+ memory T-cells. Therefore I expect that you mark donors in the way they will distinguish samples/replicas coming from the same and different donors.

I.e. there is no problem if donor labels are swapped between CD4 and CD8 T-cells as far as they point to distinct donors for CD4 or CD8 T-cells coming from different donor and the same donor for replicas.

# Feedback

Send me filled tables as plain text tab-delimited files, the file name should be in format `your-name.assignment.txt`

# Thanks for your attention!

These slides and a PDF file containing compiled analysis results can be found in `slides/` and `root` folders of the data and code bundle.