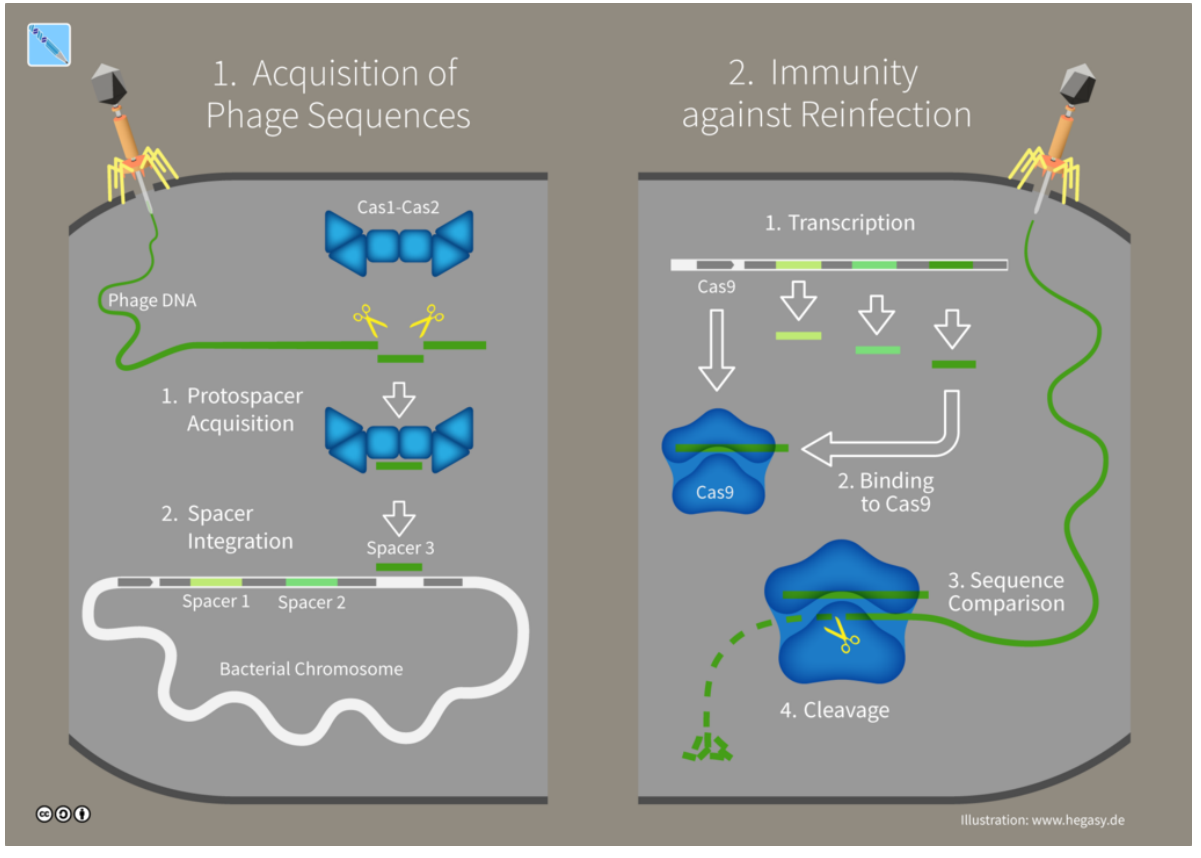
Analysis of CRISPR-Cas9 screens

Pierre Gestraud

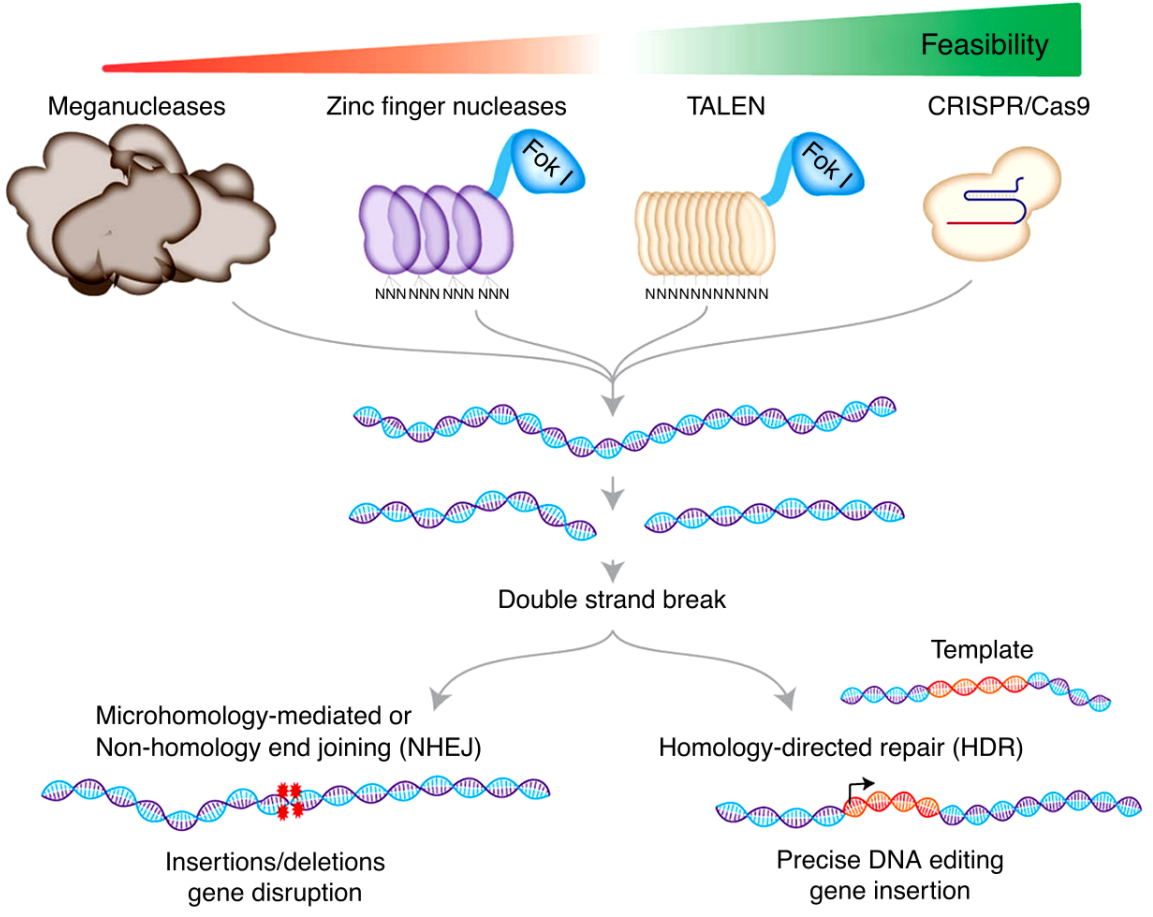
CRISPR-Cas9

- **Cas9** (CRISPR associated protein 9) is a protein of bacterial origin (e.g. *Streptococcus pyogenes*) implicated in anti-vrial response
- **CRISPR** (Clustered Regularly Interspaced Short Palindromic Repeats)



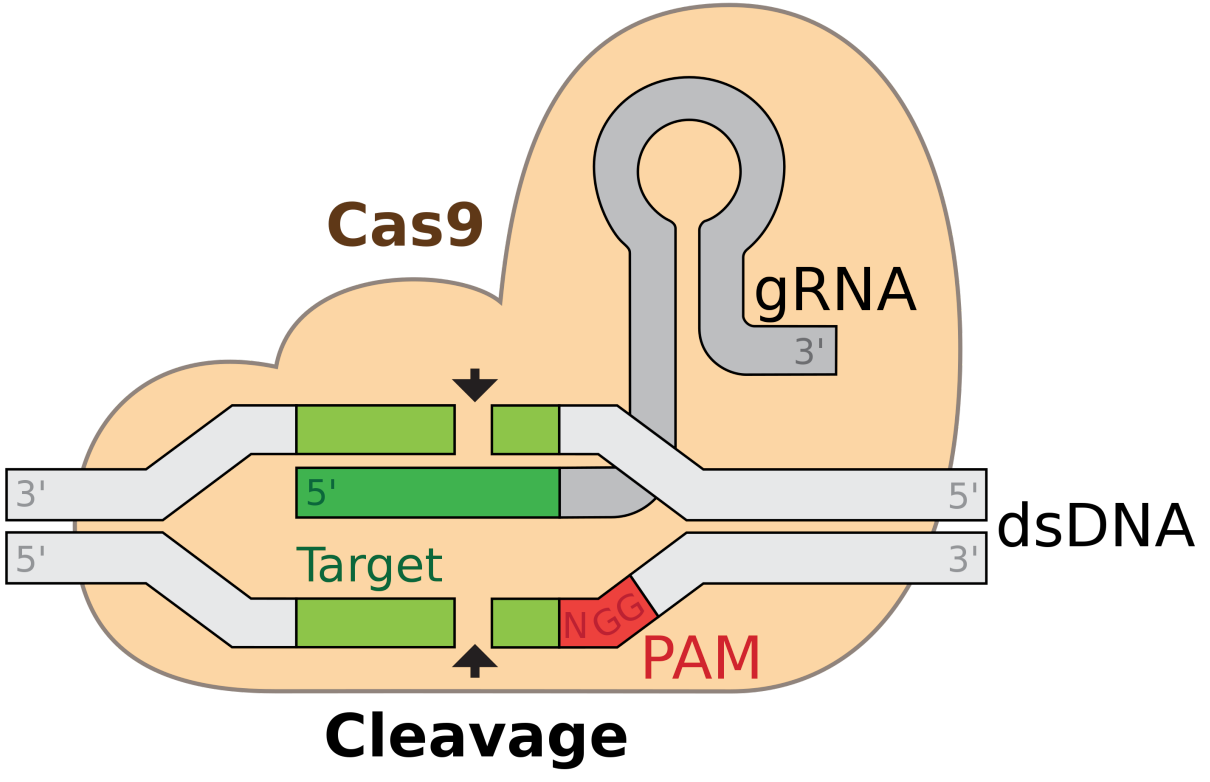
Genome editing

- Create double-strand breaks in DNA which induce gene inactivation or insertion of precise sequence based on DNA repair mecanism



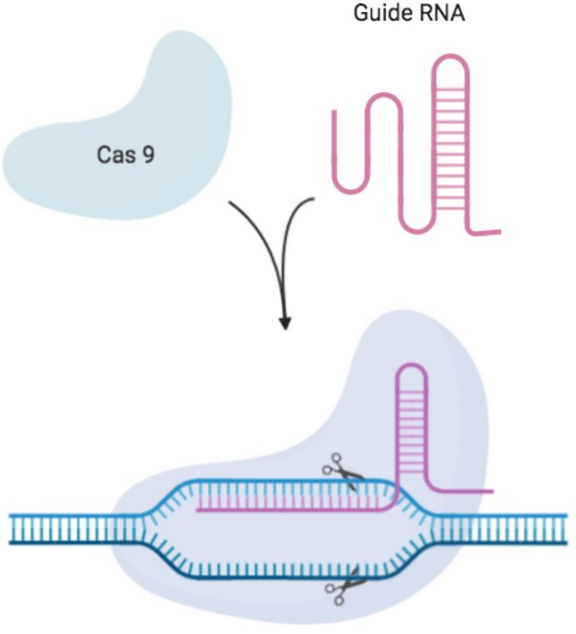
Genome editing with CRISPR-Cas9

- Designed by Emmanuelle Charpentier & Jennifer Doudna
- Efficient and precise technique to edit genome
- gRNA contains a 20bp sequence specific of the target

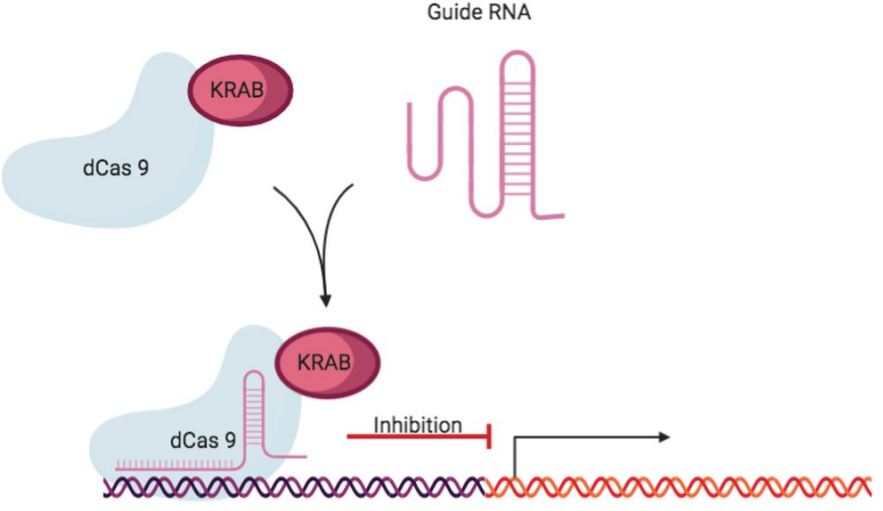


Inactivation vs inhibition

- CRISPR/Cas9
- Gene inactivation
- Knock-out



- CRISPRi/Cas9
- Gene inhibition
- Knock-down

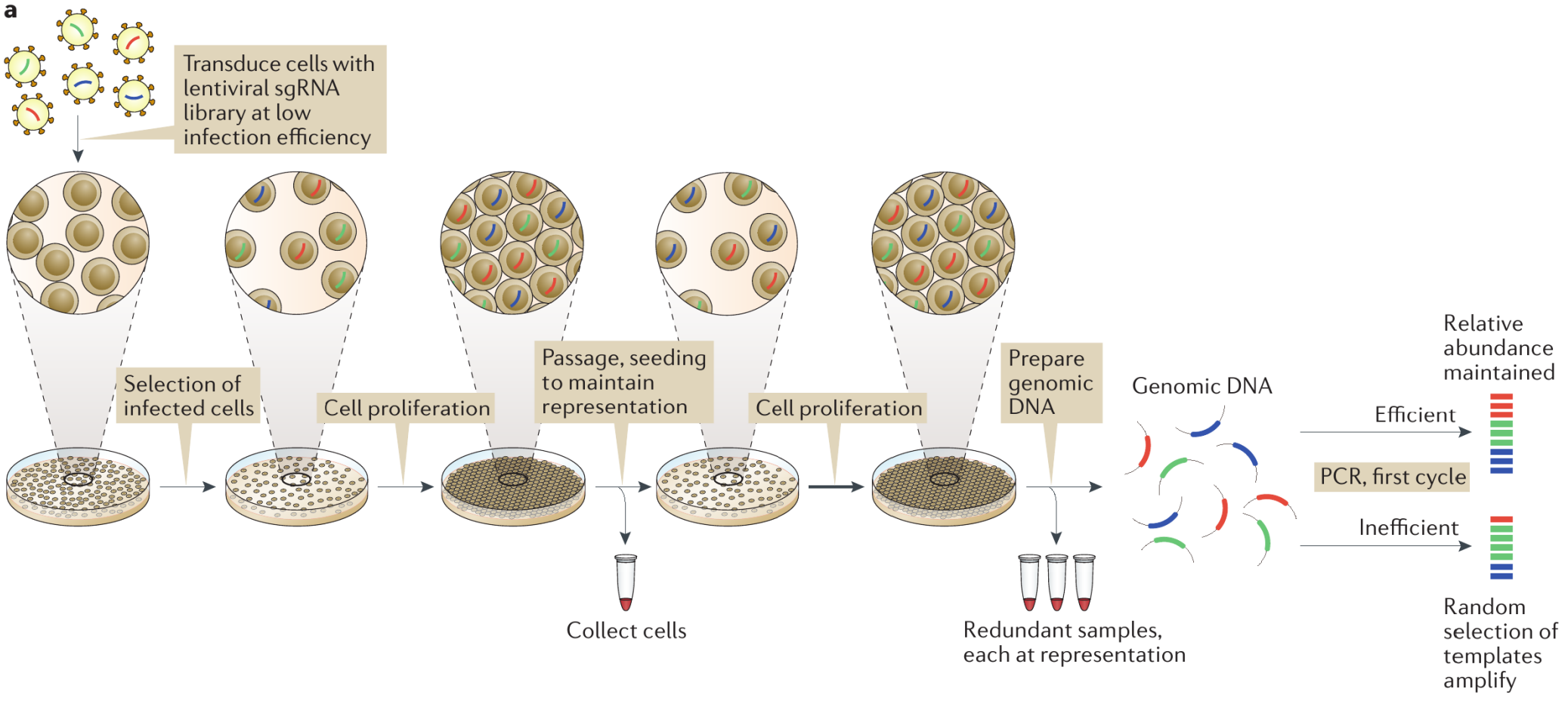


- Also activation with CRISPRa/Cas9

Genome-wide genetic screens

- Editing the genome at a single position in a set of cells if good but…
- What if we can induce one different gene inactivation in every cell?
- Idea: create a library of sgRNA to infect a cell population

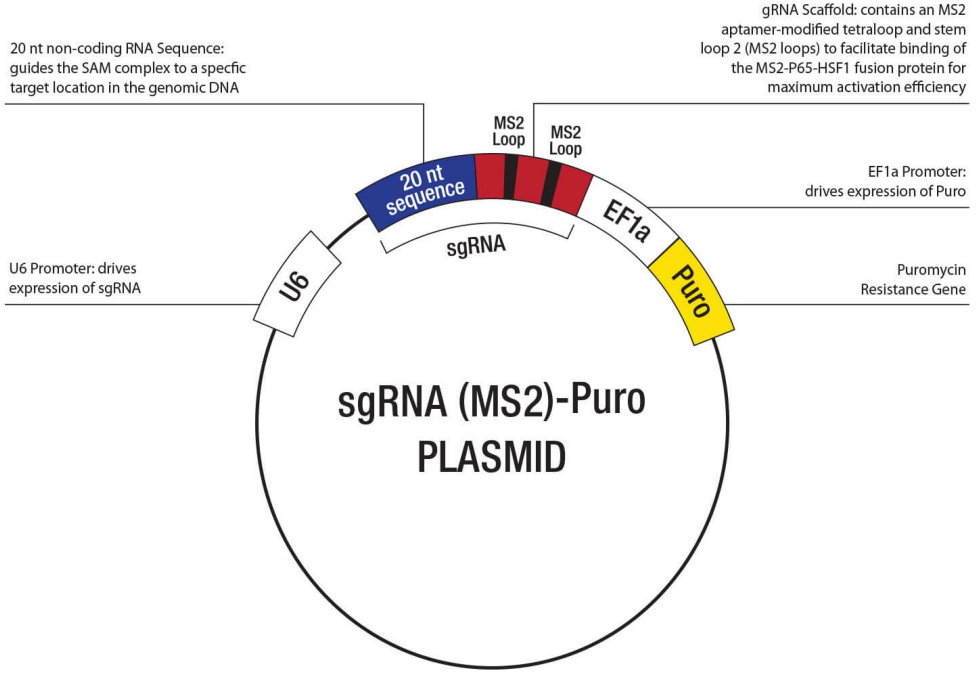
Screen workflow



Doench, Nature review 2017

sgRNA librairies

- Genome-wide librairies of plasmides commercially avalaible (Brunello, Sabatini, Gattinara…) or custom librairies for secondary screens
- Several sgRNAs by gene (4 to 10) -> between 80k and 120k guides for genome-wide screen



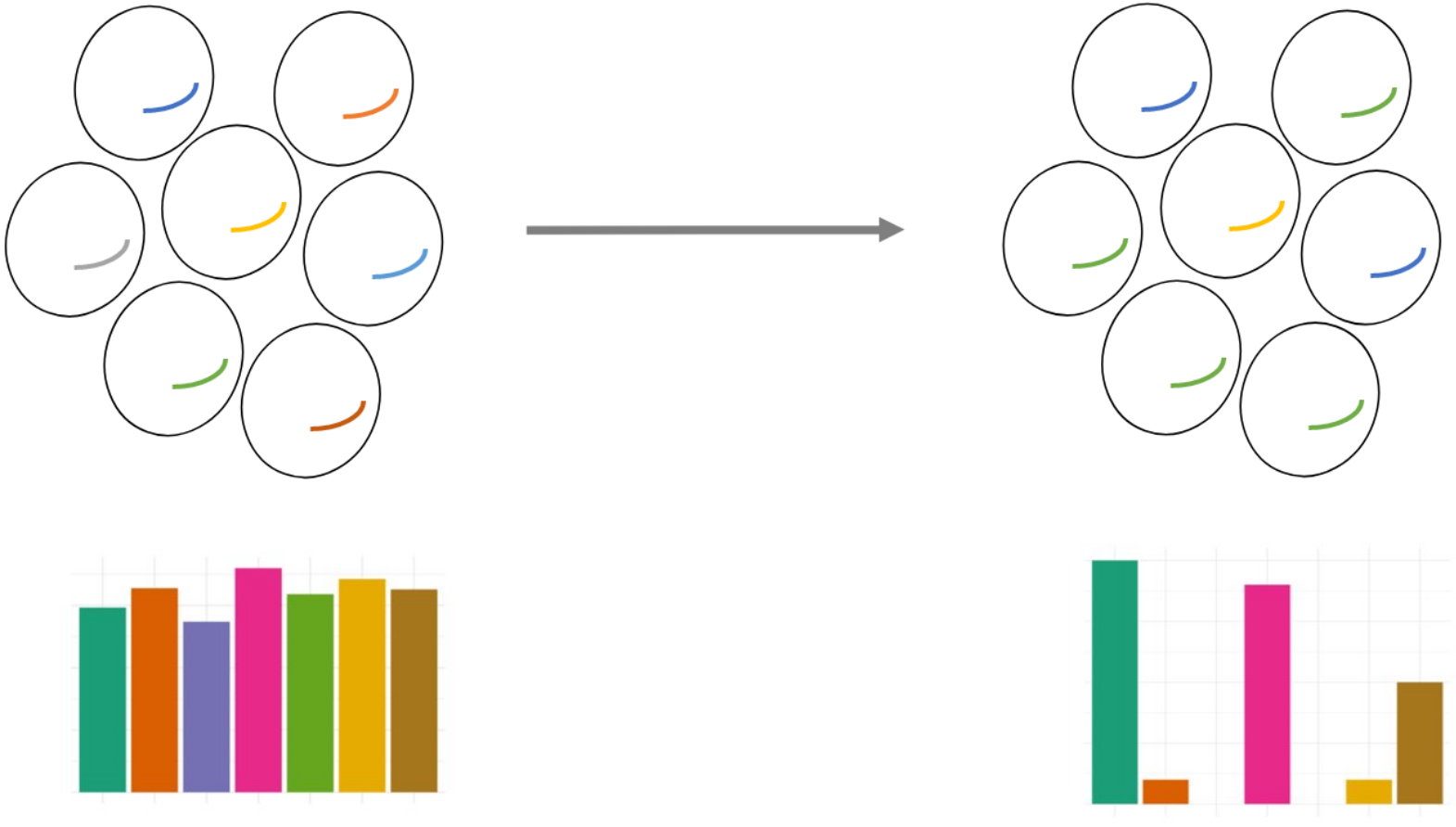
Plasmid anatomy

Cells infection and selection

- Cell lines with constitutive Cas9 expression
- Transduction at low infection efficiency (30%)
  - most cells have at most 1 sgRNA
  - avoid multiple transductions
- Selection of infected cells (Purmomycin resistant, GFP…)
- Growth in challenging environment
- DNA sequencing to identify the guides inserted

What are we looking at?

- We want to compare 2 (or more) cell populations -> differential analysis
- Often one reference population and one selected population

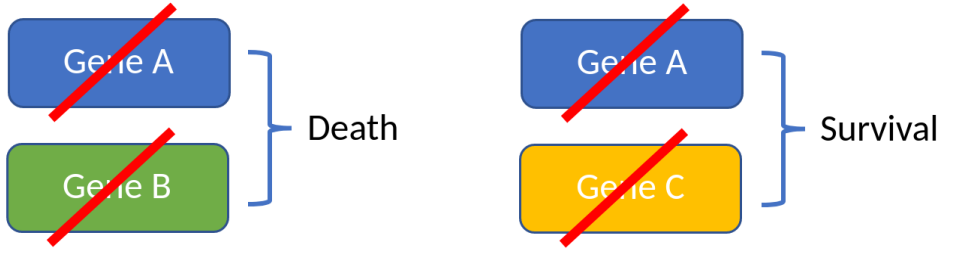


Screen types

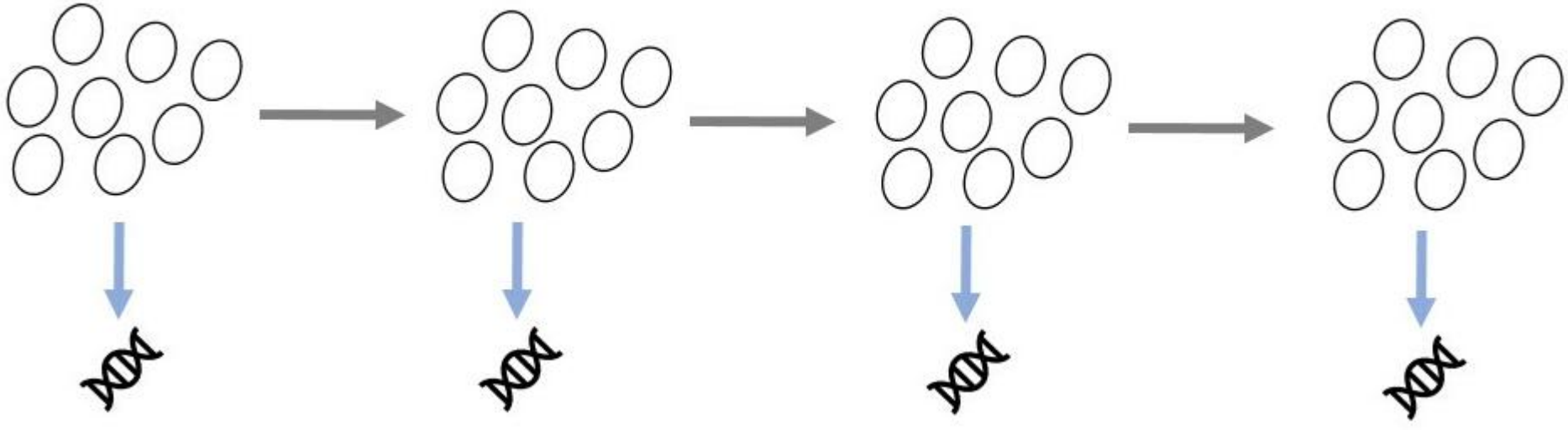
- **Negative**: find **depleted** genes
  - genes that lead to cell death when inactivated
- **Positive**: find **enriched** genes
  - cells are submitted to selection pressure
  - genes allowing escaping selection pressure when inactivated

Synthetic lethality

- Cell line with one deficient gene
- Find which genes conduct to cell death with 2 KO genes

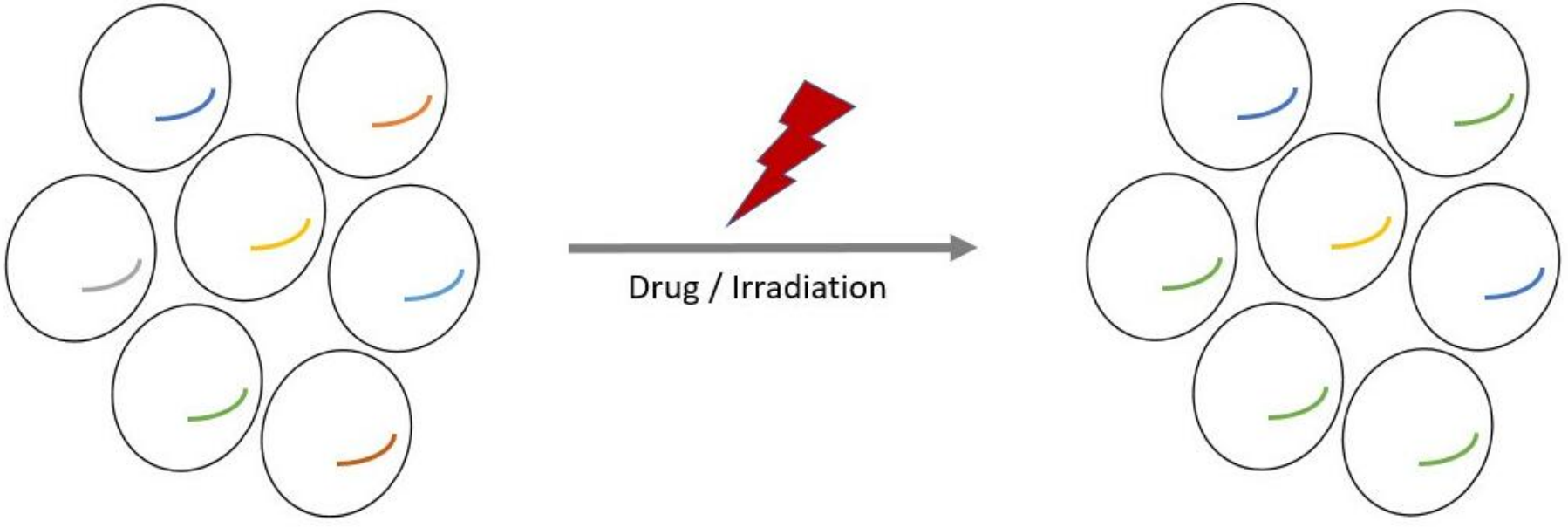


- Several time points and often several cell lines



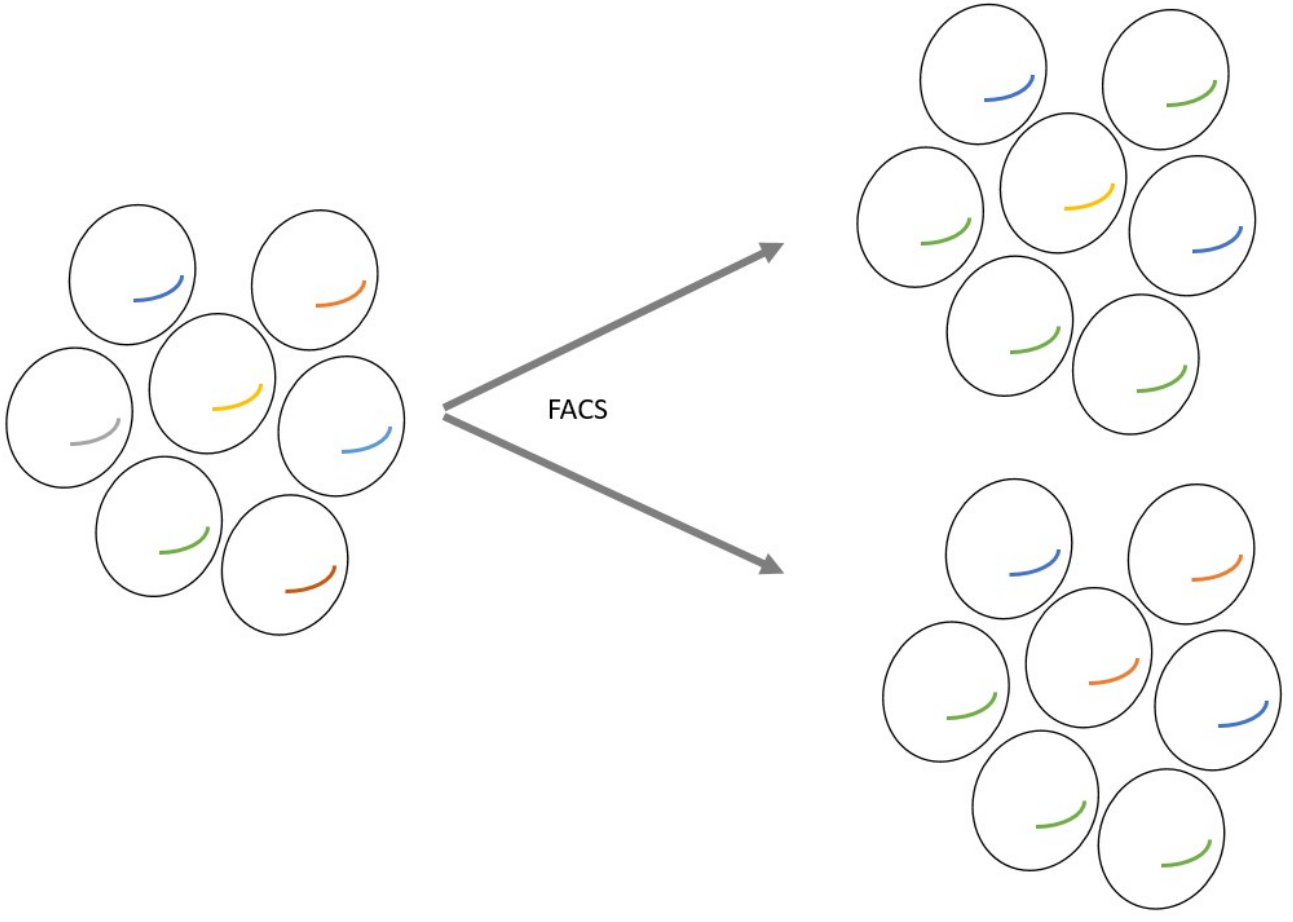
Resistance to treatment

- Identify genes implicated in treatment:
  - resistance
  - sensitivity
- Negative or positive screen



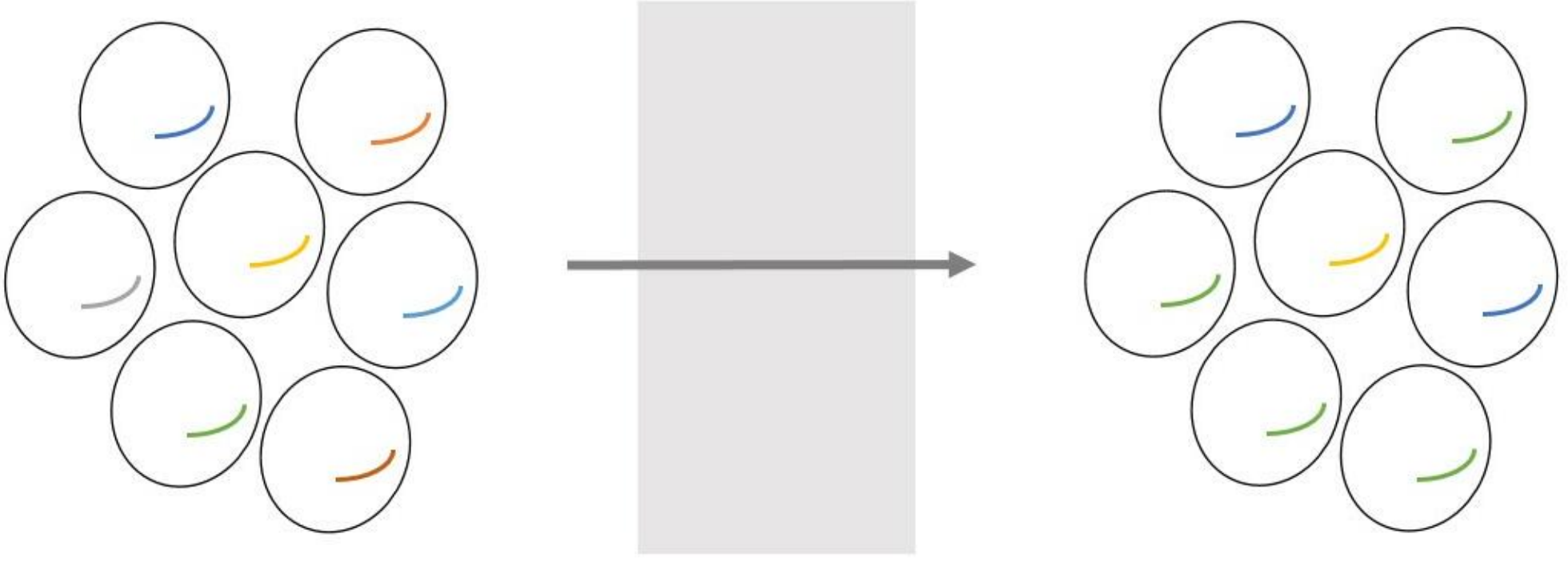
Cell sorting

- Cells are sorted by FACS depending on their phenotypes
- Find genes implicated in the differentiation



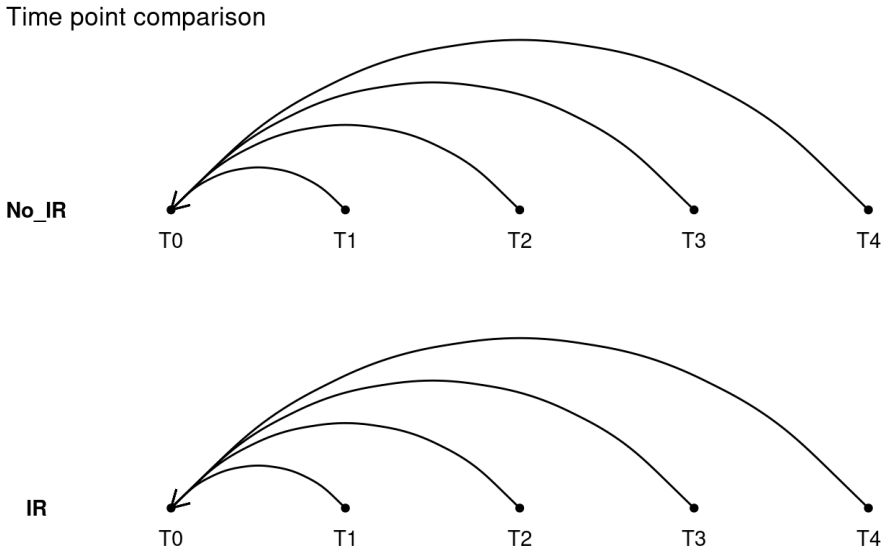
Migration

- Genes implicated in cell migration

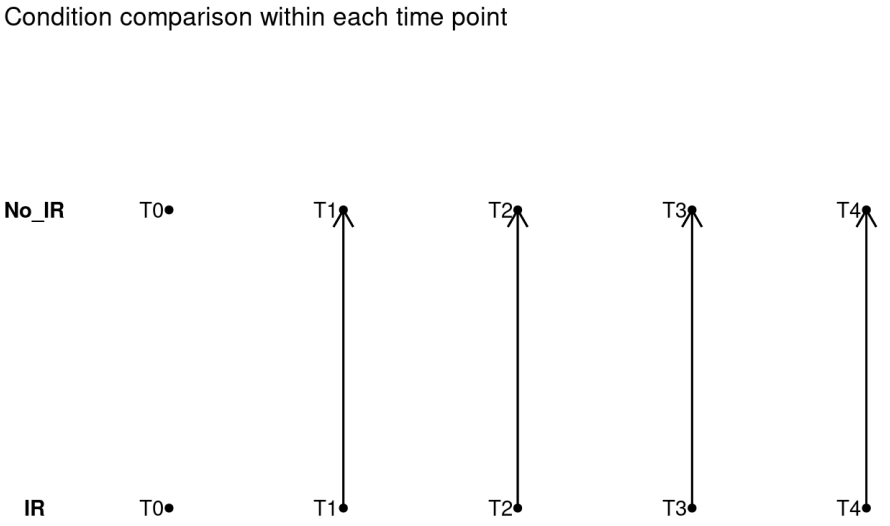


Several cell lines or conditions

- Comparisons to reference



- Direct comparisons



- Direct comparison can be biased if the cell lines have different growth rates

Controls

- Non-targeting guides
  - sgRNA with no target on the genome -> should have no effect
  - e.g. 1000 non-targeting guides
- Essential and non-essential genes
  - Lists of genes established on several cell lines (Wang *et al*, 2015 Science)

Counting

- How to count reads after sequencing?
- No need of traditional mapping
- Dedicated python script to scan each read and find the guide (Marc Deloger) + Nextflow pipeline + MultiQC output

CTTGTGGAAAGGACGAAACACCGCTTCATTTCCCAGCCACCAAGTTTTAG

ACGCAACTTGTGGAAAGGACGAAACACCGCTTCATTTCCCAGCCACCAAG

TGCACCTTGTGGAAAGGACGAAACACCGCTTCATTTCCCAGCCACCAAGT

AGCTTGTGGAAAGGACGAAACACCGCTTCATTTCCCAGCCACCAAGTTTT

ACGCAACTTGTGGAAAGGACGAAACACCGCTTCATTTCCCAGCCACCAAG

TTGTGGAAAGGACGAAACACCGCTTCATTTCCCAGCCACCAAGTTTTAGA

Data

- Counts matrix
- One row by sgRNA
- Counts represent the number of cells with the sgRNA included

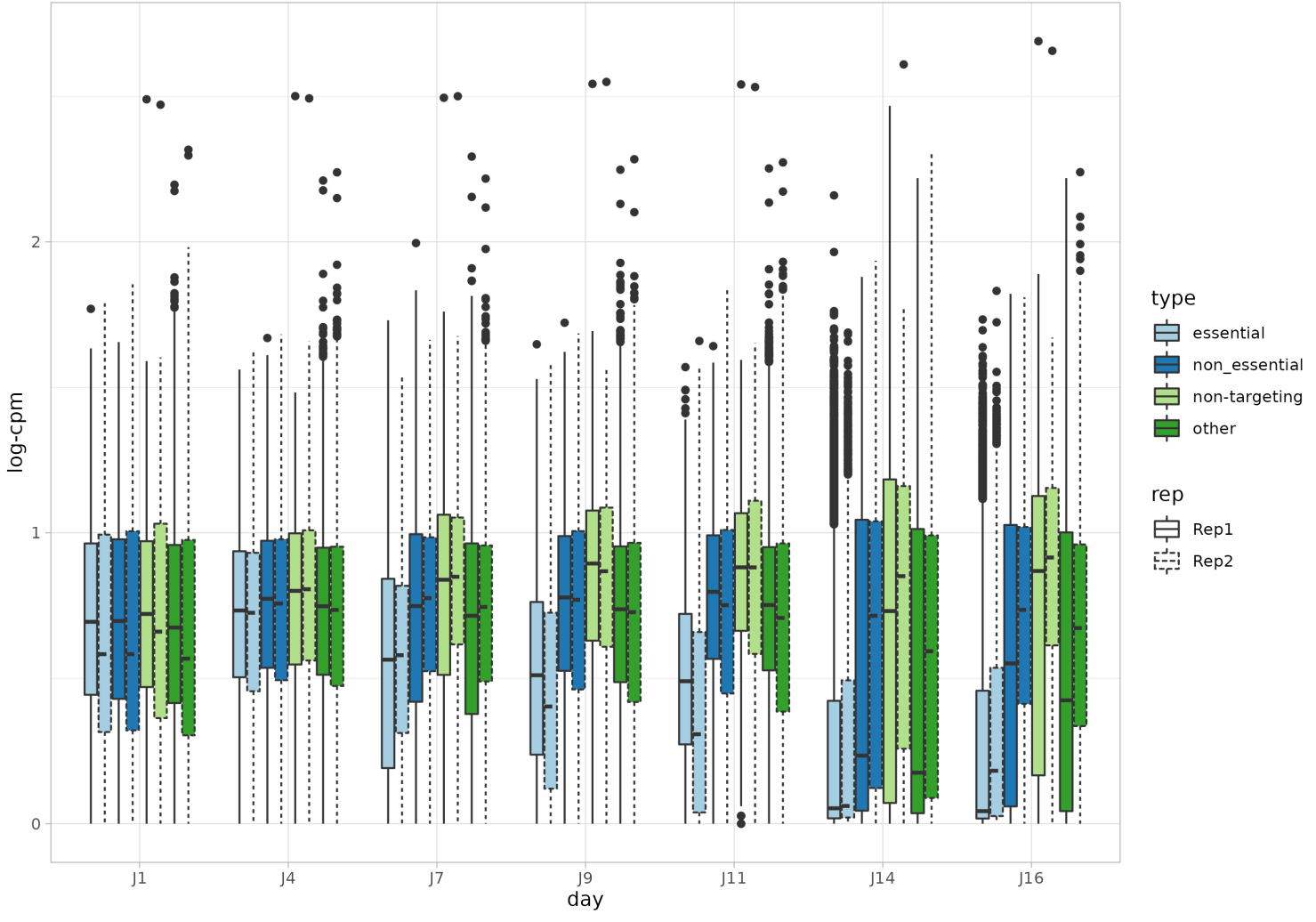
	sgRNA	gene	D115R08	D115R09	D115R10	D115R11	D115R12	D115R13	D115R14
1	sgA1BG_1	A1BG	39	28	229	170	140	497	437
2	sgA1BG_10	A1BG	710	190	120	309	435	454	491
3	sgA1BG_2	A1BG	925	278	73	770	362	466	68
4	sgA1BG_3	A1BG	9	114	1	13	49	12	80
5	sgA1BG_4	A1BG	930	244	116	197	260	14	19
6	sgA1BG_5	A1BG	16	53	285	435	143	778	55
7	sgA1BG_6	A1BG	68	129	75	373	343	4	15
8	sgA1BG_7	A1BG	248	220	147	547	177	395	9
9	sgA1BG_8	A1BG	195	115	38	398	162	621	278
10	sgA1BG_9	A1BG	56	54	2	80	151	2	357
11	sgA1CF_1	A1CF	430	134	382	325	396	13	870
12	sgA1CF_10	A1CF	100	95	4	128	150	5	3
13	sgA1CF_2	A1CF	836	165	201	327	397	494	851
14	sgA1CF_3	A1CF	658	223	109	780	225	171	572
15	sgA1CF_4	A1CF	222	47	13	117	252	2	8
16	sgA1CF_5	A1CF	1255	876	685	1967	1376	1874	3263
17	sgA1CF_6	A1CF	2	0	5	153	2	1	4
18	sgA1CF_7	A1CF	30	76	8	61	178	2	1022

Quality control

- Use controls to estimate screen efficiency
  - We should see a depletion for essential genes (at least)
- Distribution of guides by samples for non-targeting, essential and non-essentials
- ROC curves

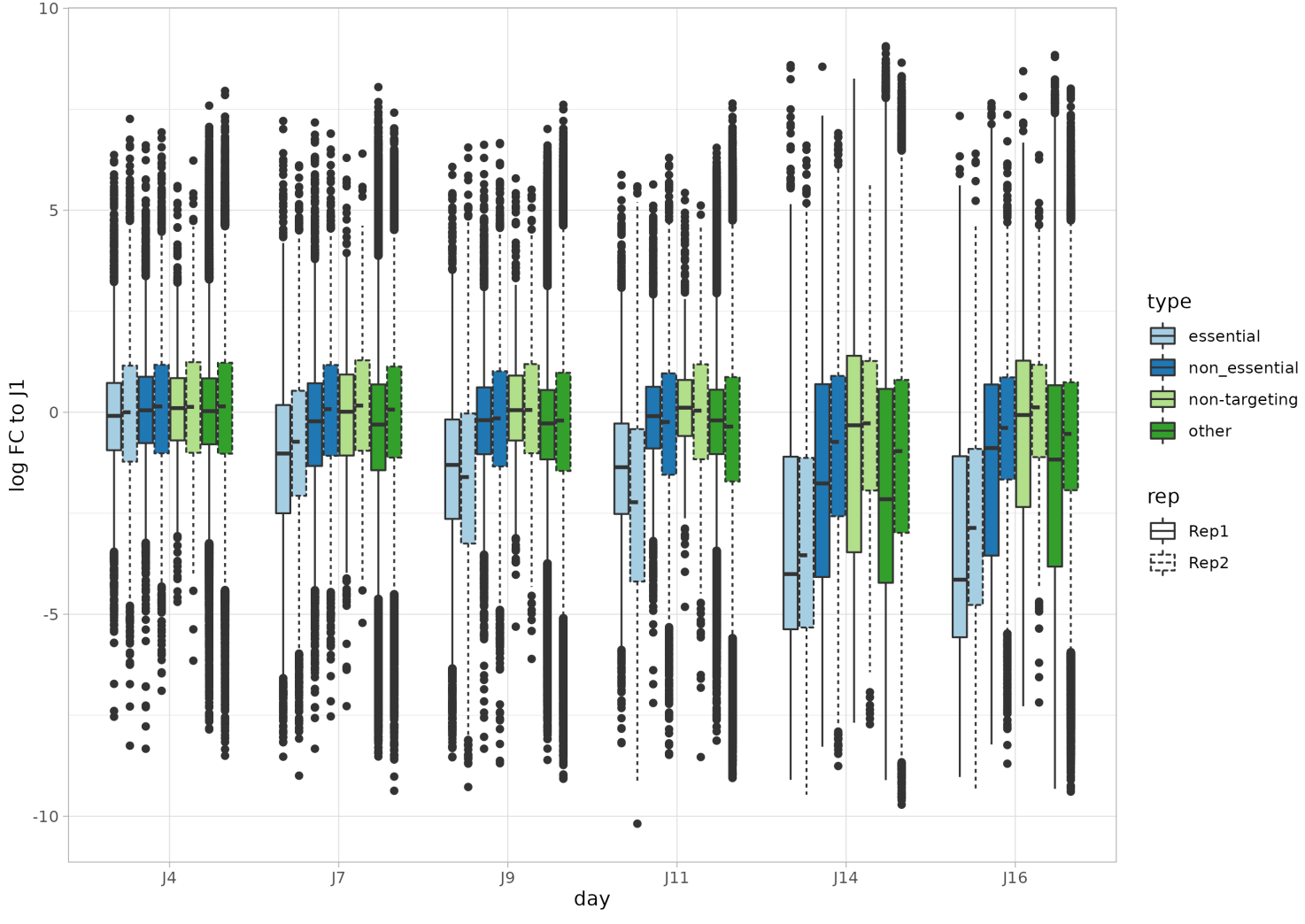
Guides distribution

- Distributions of log-cpm according to sgRNA type



Guides distribution

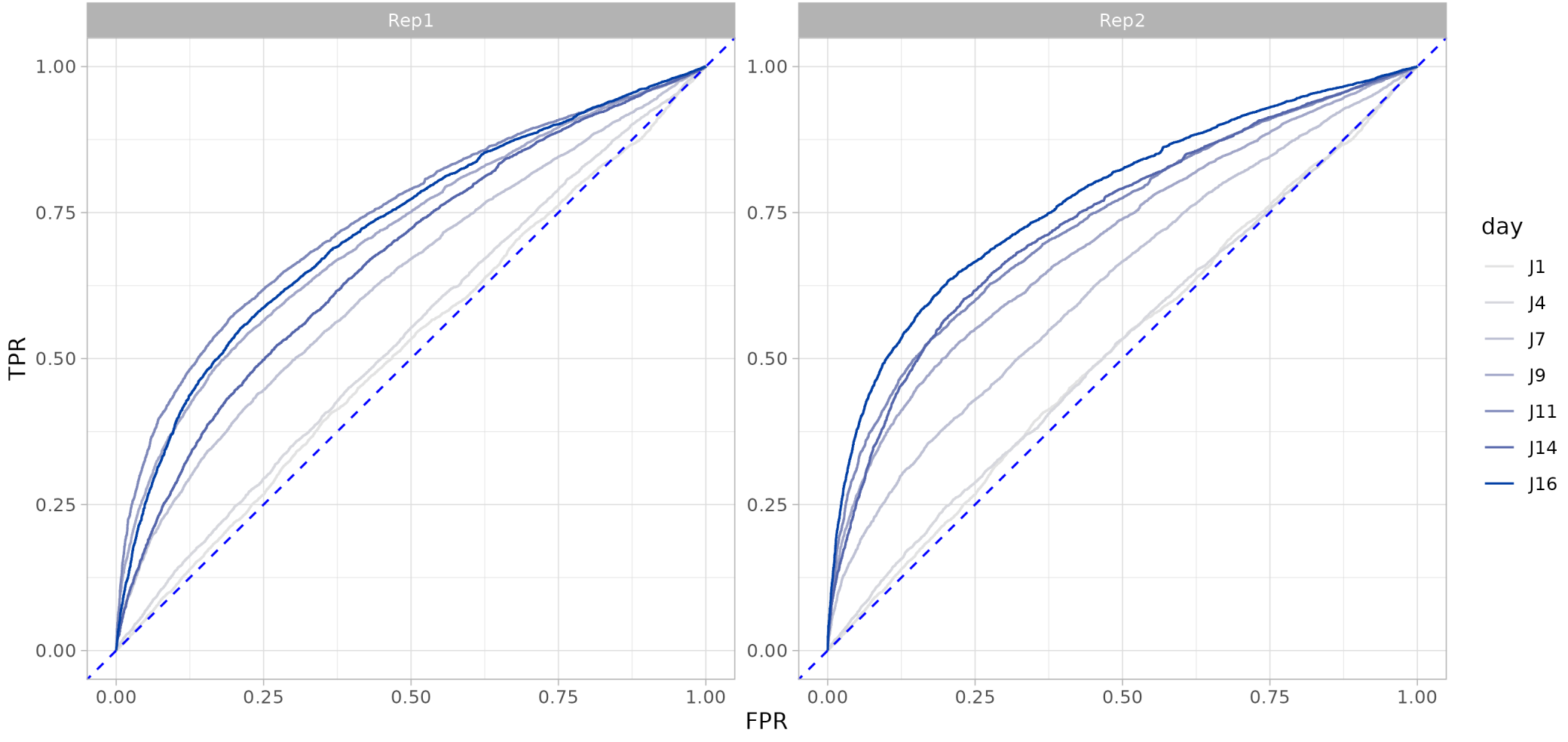
- Distributions of log FC to reference sample (first time point or library)



ROC curves

- Construct ROC curves by setting:
  - essential as “+”
  - non-essential and non-targeting as “-”
- Order guides by cpm
- “+” should be ranked before “-”
- Curve on the diagonal -> no selection
- Too much selection -> no distinction between essential and non-essential

ROC curves


Rep1
Rep2
1.00
0.75
0.50
0.25
0.00
TPR
0.00
0.25
0.50
0.75
1.00
FPR
day
J1
J4
J7
J9
J11
J14
J16

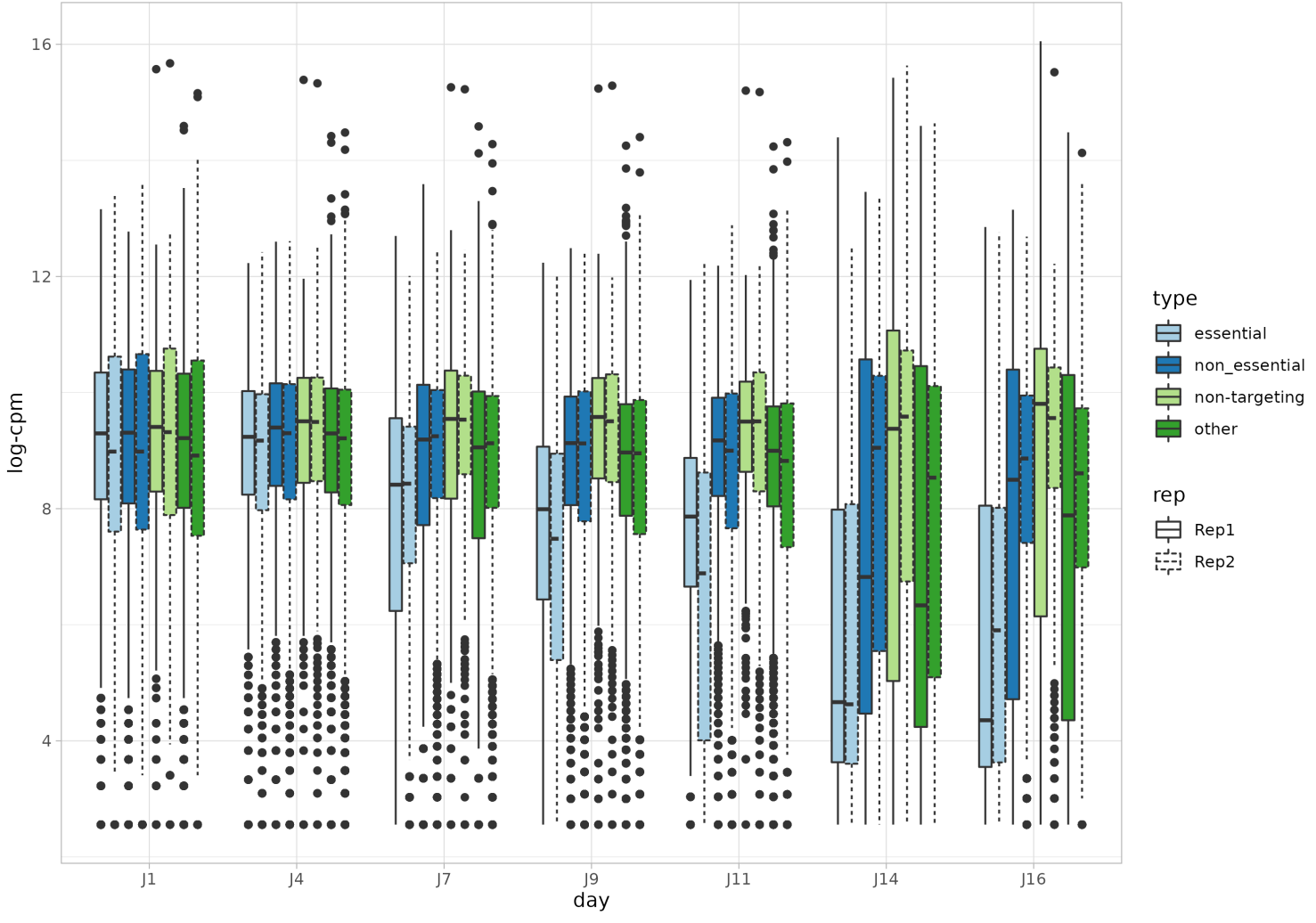
Normalisation

- **How to normalise?**

- TMM with normalisation factors computed on:
  - all the guides
  - the non-targeting guides
- cpm

Guides distribution

- Distributions of normalised cpm



Analysis workflow

- How to analyse such data?
- We can do the analysis for guides but we want results for genes
- Merge all guides from same gene and work as for RNAseq?
  - guides have different efficiency
  - some are without effect

- Need dedicated workflow
- Analysis in 2 steps:
  - **Guide level**: find the enriched/depleted guides
  - **Gene level**: aggregate the guide results by gene

Guide level analysis

- limma/voom framework
- Often good to include replicate effect in the model

```
~ time_point:cell_line + replicate
```

- From the final t statistics, we compute 3 different pvalues:
  - bilateral
  - unilateral for depletion
  - unliateral for enrichment

```
tab <- biobroom::tidy.MArrayLM(object)
tab$p.value_dep <- pt(tab$statistic, df = object$df.total[1], lower.tail = TRUE)
tab$p.value_enrich <- pt(tab$statistic, df = object$df.total[1], lower.tail = FALSE)
```

Gene level analysis

- For each gene, we now have 4 to 10 p-values

```
# A tibble: 10 × 12
# Groups:   Gene [2]
   sgRNA      estimate statistic p.value p.value_dep p.value_enrich adj_p.value
   <chr>         <dbl>     <dbl>   <dbl>       <dbl>          <dbl>       <dbl>
 1 sgACTR1A_1   -1.84     -3.52  0.00148    0.000740          0.999      0.0147
 2 sgACTR1A_2   -1.50     -3.52  0.00147    0.000735          0.999      0.0147
 3 sgACTR1A_3   -1.28     -1.50  0.146      0.0729            0.927      0.312
 4 sgACTR1A_4   -1.22     -2.40  0.0230     0.0115            0.988      0.0877
 5 sgACTR1A_5   -0.705    -0.703 0.488      0.244             0.756      0.681
 6 sgACTR1A_6   -0.772    -2.32  0.0277     0.0139            0.986      0.100
 7 sgALG1_1     -0.463    -0.964 0.343      0.172             0.828      0.552
 8 sgALG1_2     -0.891    -1.81  0.0816     0.0408            0.959      0.211
 9 sgALG1_3     -0.779    -1.48  0.150      0.0751            0.925      0.319
10 sgALG1_4     -1.19     -2.58  0.0151     0.00757           0.992      0.0660
# … with 5 more variables: adj_p.value_dep <dbl>, adj_p.value_enrich <dbl>,
#   Gene <chr>, sequence <chr>, n <int>
```

Gene level analysis

- The guides are not all effective
- How to aggregate p-values?
  - “Pragmatic approach”: keep the genes with at least k significantly depleted/enriched guides
  - Fisher’s method
  - RRA

Robust Rank Aggregation

- Designed for shRNA screen
- Implemented in Mageck
- Use properties of order statistics (the kth order statistic of a statistical sample is equal to its kth-smallest value)
- Order statistics from a uniform distribution between 0 and 1 have marginal distribution following a Beta. The k^{th} value among n uniformly distributed values:

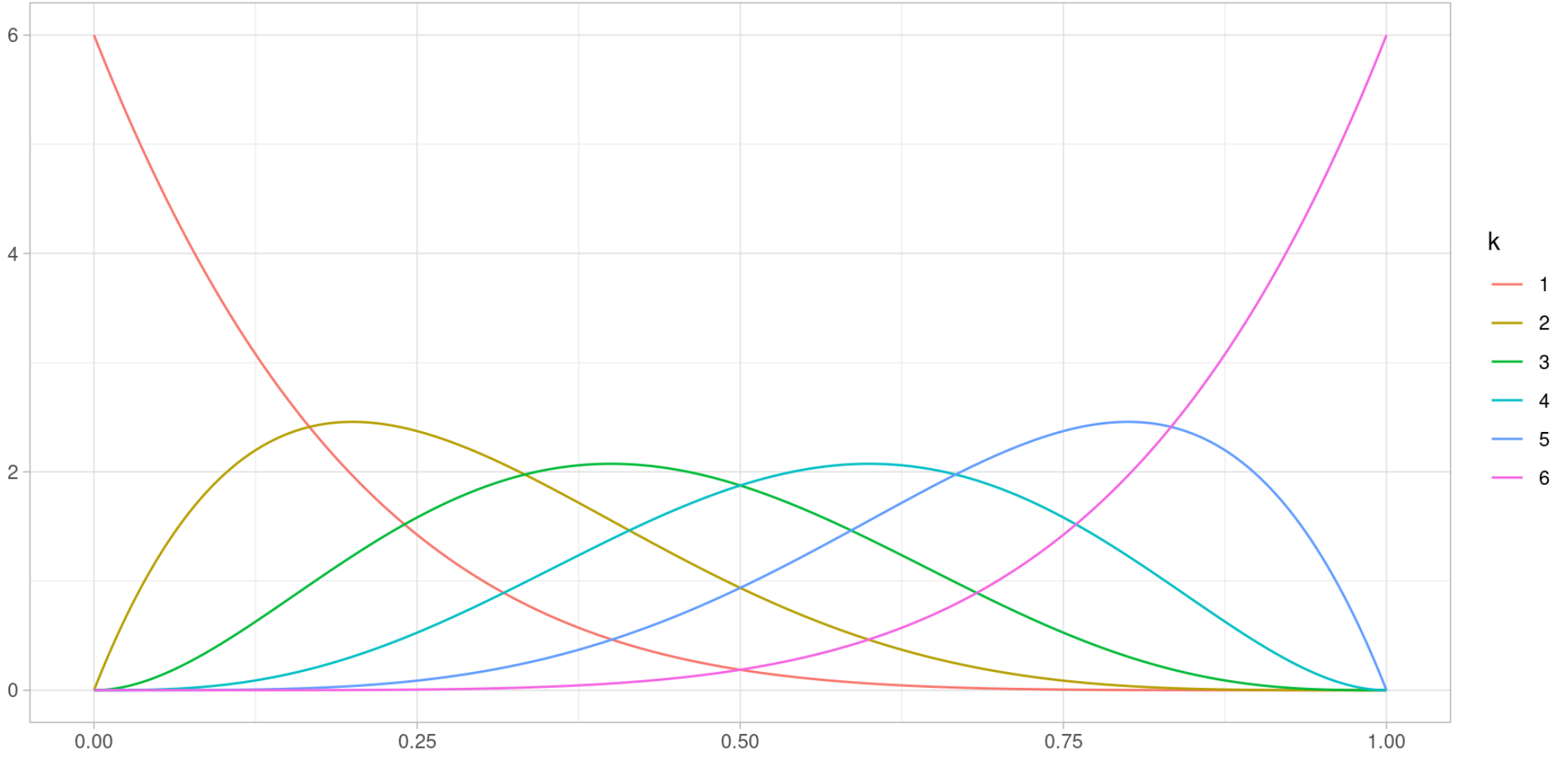
$$U_{(k)} \sim Beta(k, n+1-k)$$

Rationale:

- If a gene has no effect, its p-values follow a uniform distribution

Robust Rank Aggregation

Distributions of order statistics from $U(0, 1)$:



RRA - score for each gene

For each gene:

- Order the n guides by pvalue (p_i)
- Compute the score c_i for each guide i

$$c_i = P(Beta(i, n + 1 - i) < p_i)$$

- Compute a score for the gene:

$$s_g = min(c_i)$$

RRA - score for each gene

- We compute 3 scores for each gene:
  - overall
  - depletion
  - enrichment
- α-RRA modification: We only consider pvalues lower than α (others are set to 1)
  - e.g. $\alpha = 0.2$
- Number of guides supporting the score

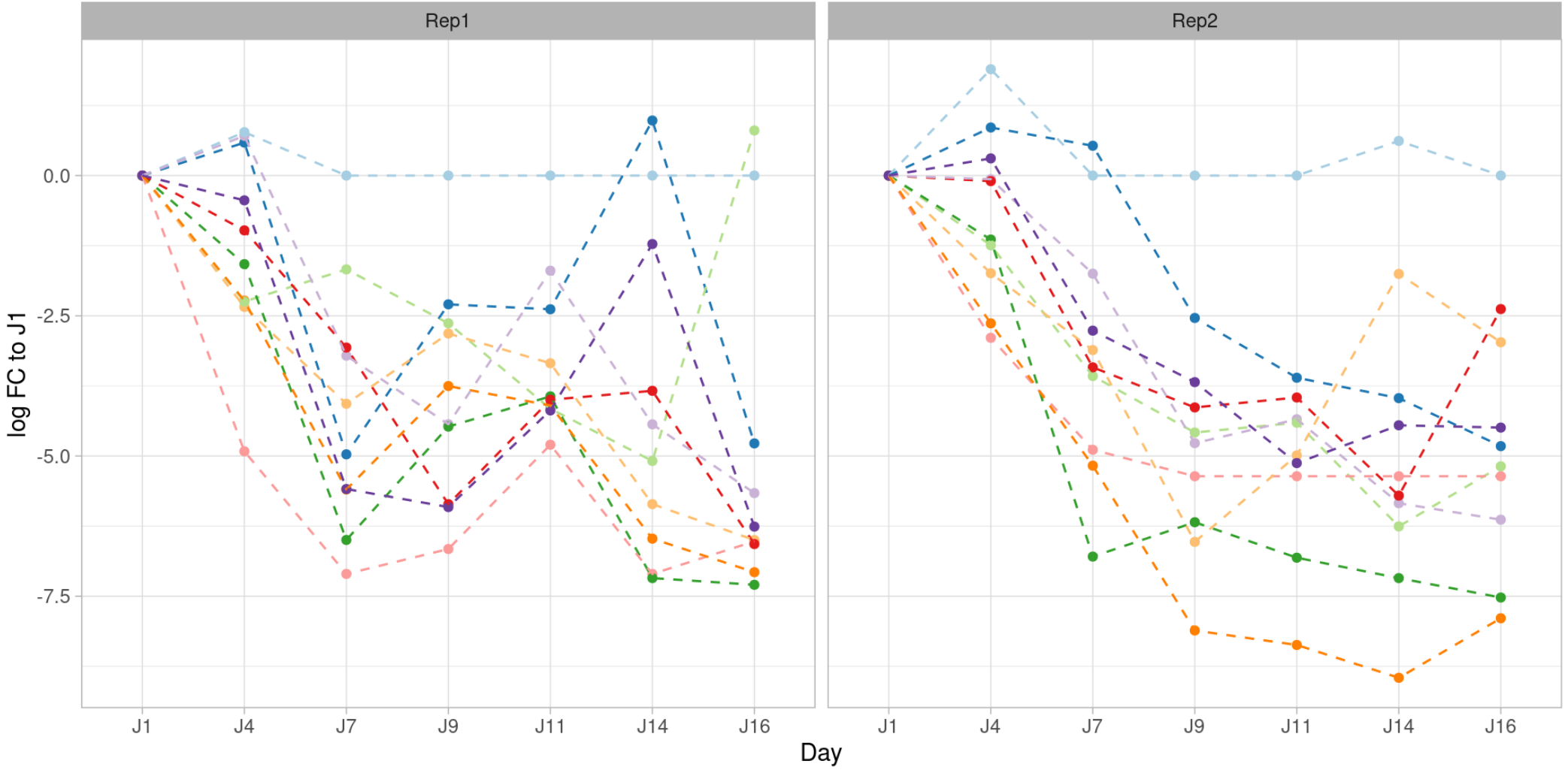
RRA - pvalue for each gene

- Create null distribution of RRA scores from random genes
- Random genes defined as set of guides picked from:
  - all guides
  - non targeting guides

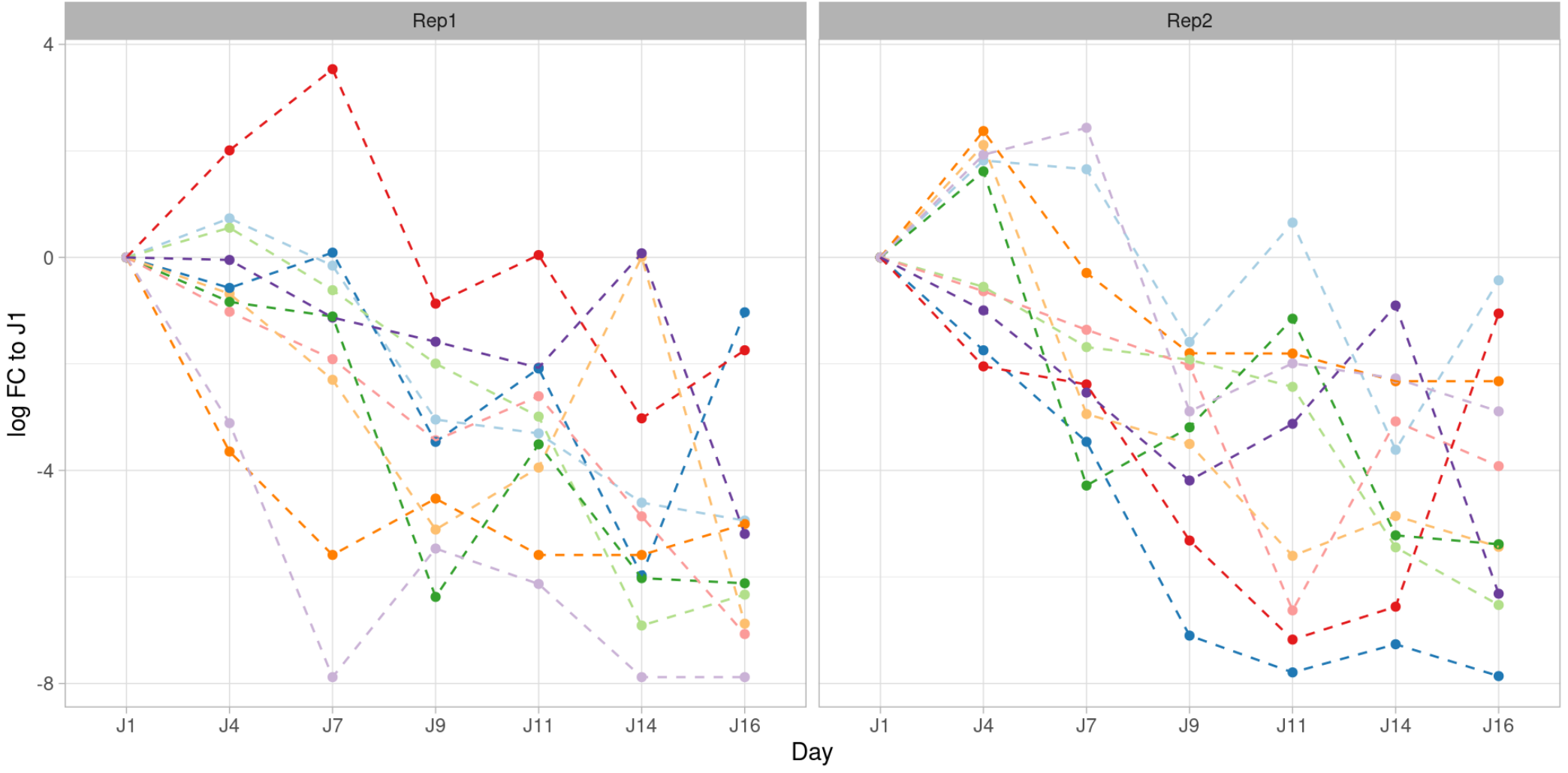
Gene results

```
# A tibble: 10 × 13
   Gene    RRA_dep_score RRA_dep_which RRA_dep_pvalue RRA_dep_adjp RRA_score
   <chr>           <dbl>         <dbl>          <dbl>        <dbl>     <dbl>
 1 DGCR8         4.55e-23             8              0            0  1.16e-20
 2 EFTUD2        9.68e-23             7              0            0  1.24e-20
 3 EEFSEC        4.04e-22            10              0            0  4.14e-19
 4 CCT3          6.82e-22             8              0            0  1.74e-19
 5 DARS          3.53e-21             6              0            0  2.26e-19
 6 LRR1          3.64e-21             9              0            0  1.86e-18
 7 EIF2S1        1.87e-20             6              0            0  1.20e-18
 8 POLR2A        2.03e-20             7              0            0  2.59e-18
 9 INTS9         2.27e-20             7              0            0  2.90e-18
10 DYNC1H1       2.45e-20             7              0            0  3.12e-18
# … with 7 more variables: RRA_which <dbl>, RRA_enrich_score <dbl>,
#   RRA_enrich_which <dbl>, RRA_pvalue <dbl>, RRA_enrich_pvalue <dbl>,
#   RRA_adjp <dbl>, RRA_enrich_adjp <dbl>
```

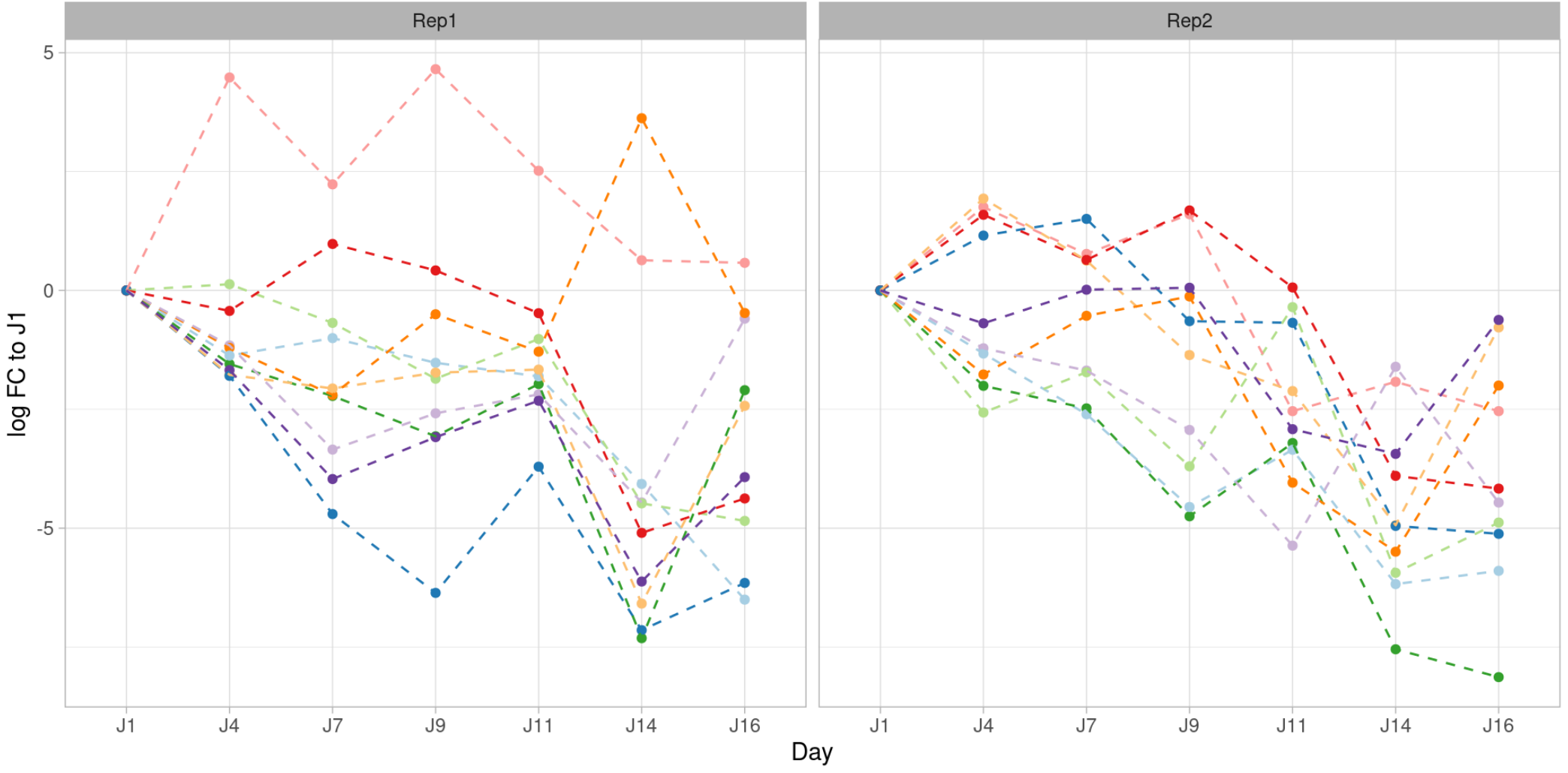
Visualisation



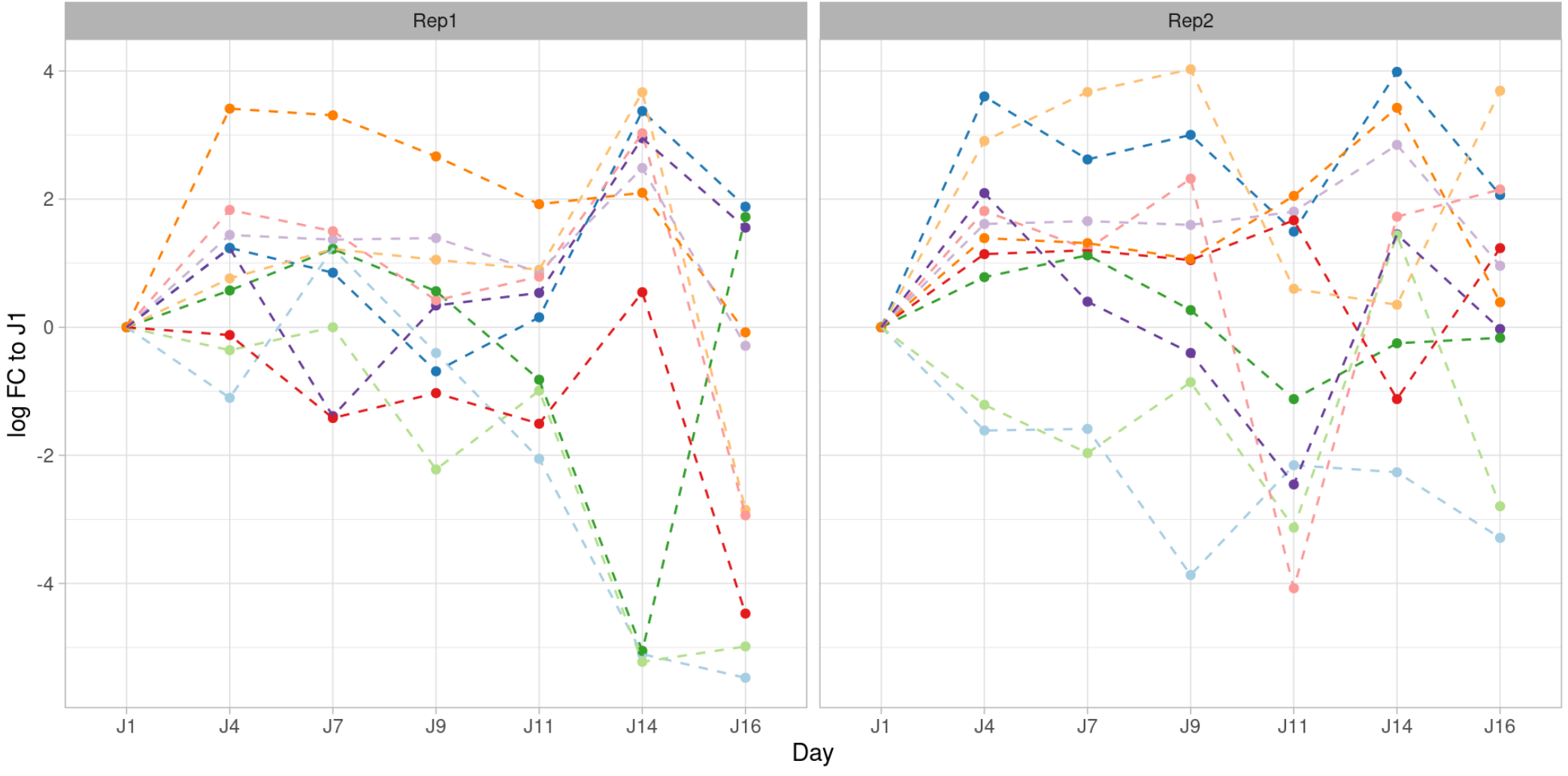
Visualisation



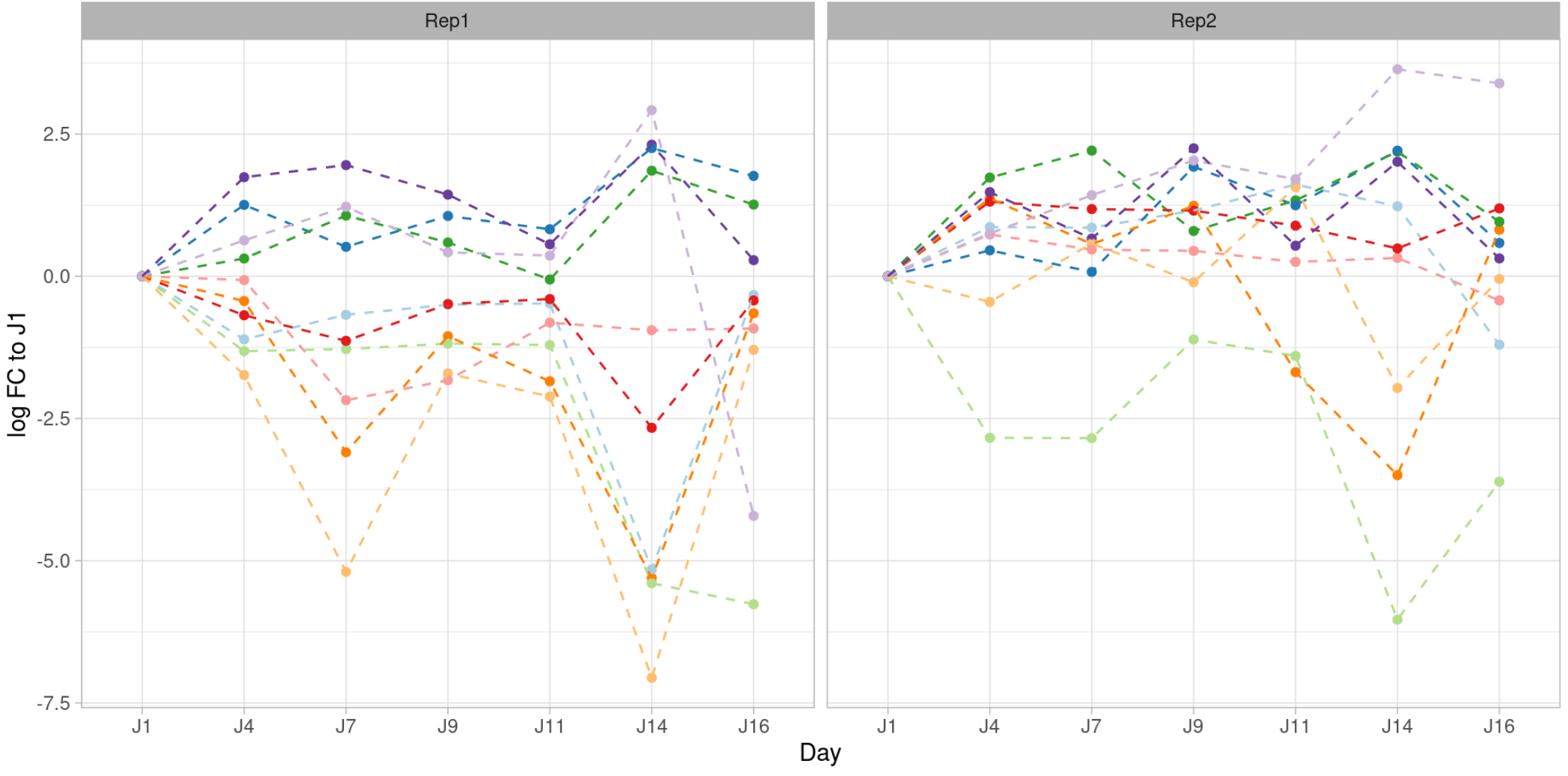
Visualisation



Visualisation



Visualisation



Next steps

- Functional analysis on significant genes
  - GO/KEGG/Reactome enrichment
- Secondary screen on limited set of genes (some 100s)
  - Design custom library
  - Less cells required
  - Include controls (positive, negative, non-targeting)

Current/Future developments

- Double-guide library
  - Each cell will have 2 inactivated genes
  - Pairs fixed or random
- In-vivo screening in PDX
- Secondary library design
  - Select a set of guides among several commercial libraries
- Convince the facility to perform more replicates

Acknowlegments

CRISPR’it core facility (UMR3215/U934)

- Aurélien Bore
- Michel Wassef
- Raphaël Margueron
- *Camille Fouassier*

CUBIC - Bioinformatics core facility (U900)

- Nicolas Servant
- *Marc Deloger*
- *Clément Benoit*