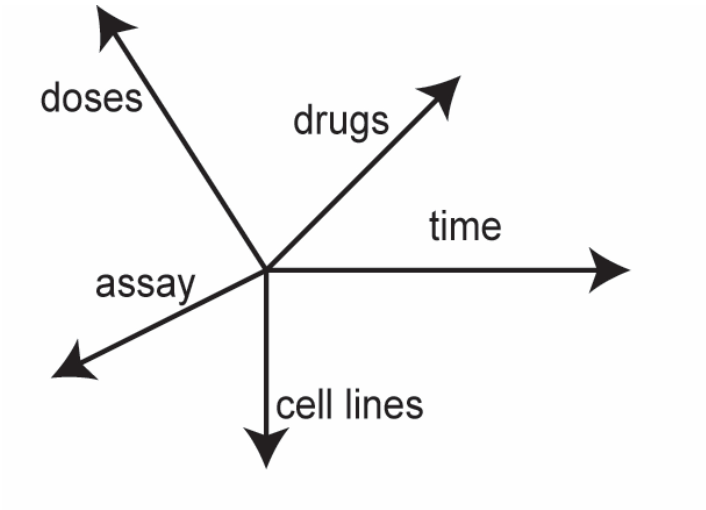
ICSB Workshop: Drug Response Measurement and Analysis

Part 2: Best practices for experimental design, execution, and analysis

Caitlin Mills

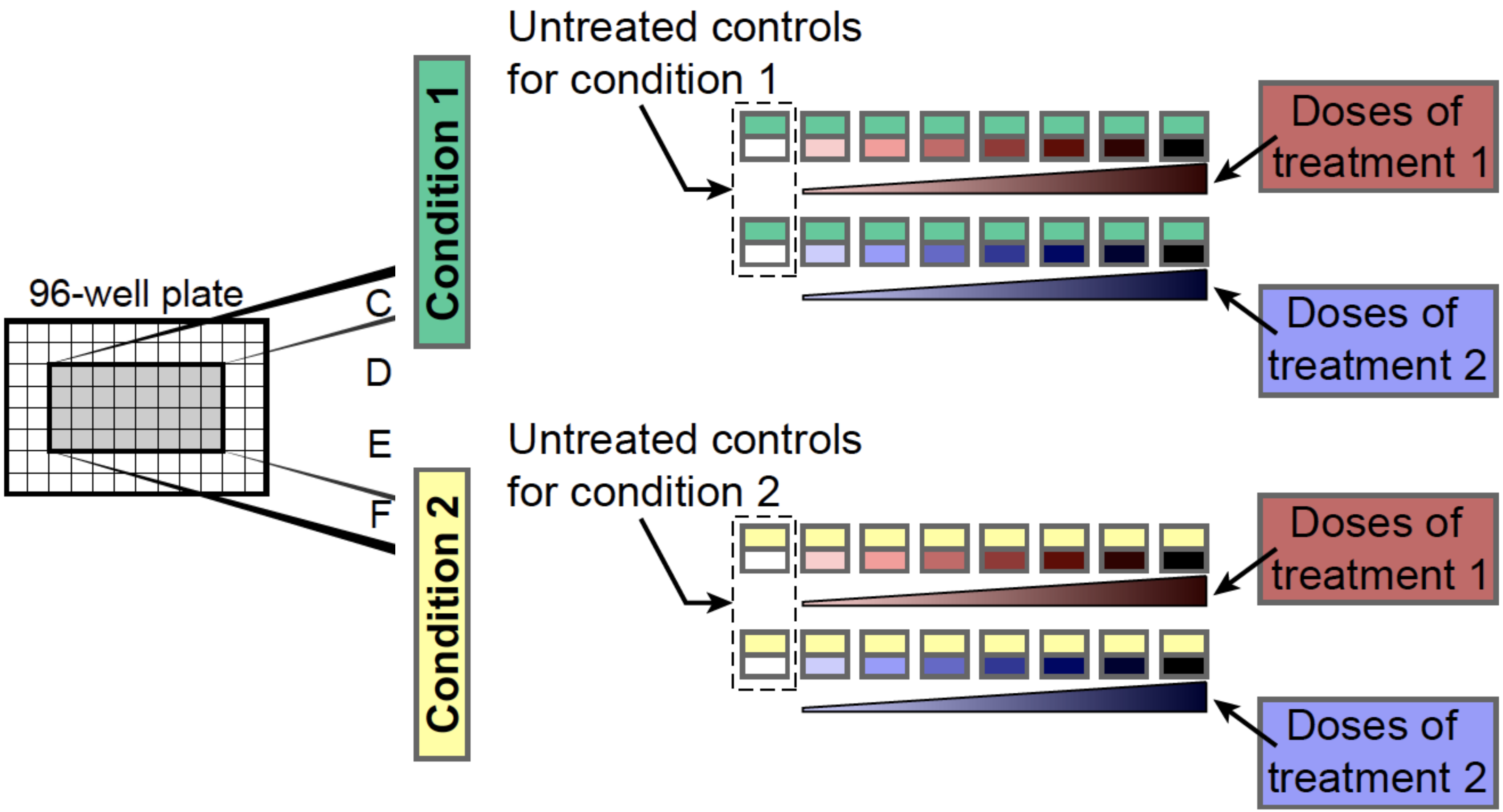
Kartik Subramanian

Drug-response experiments are becoming increasingly high-throughput

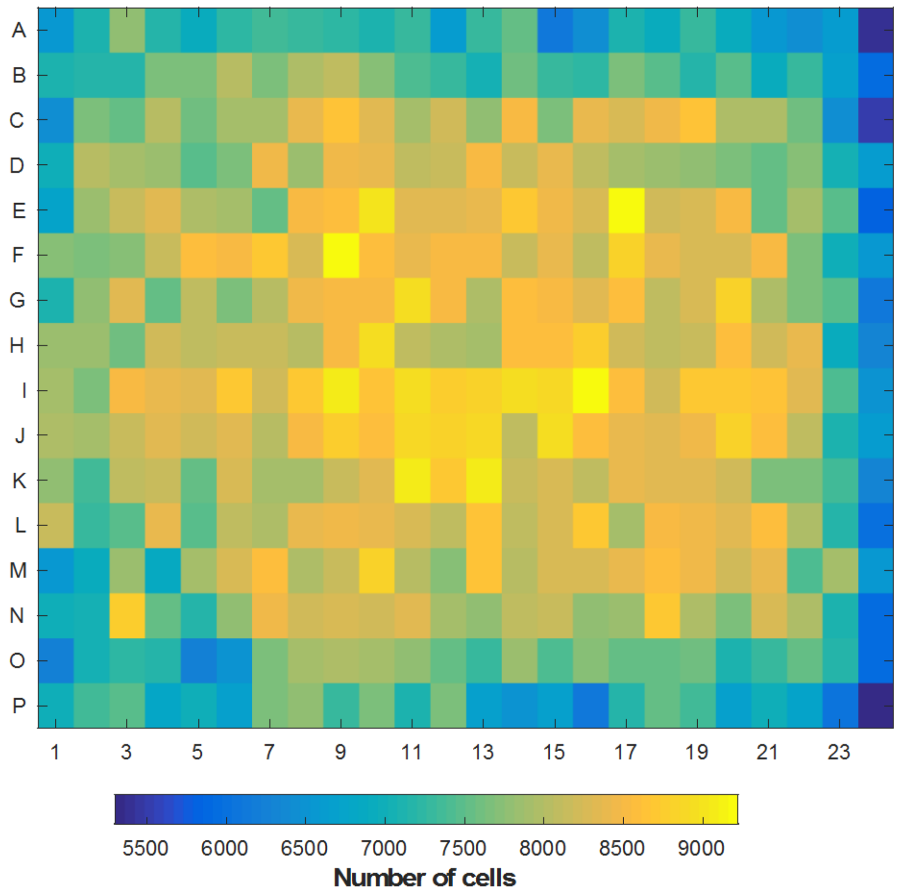


96 or 384 well plates

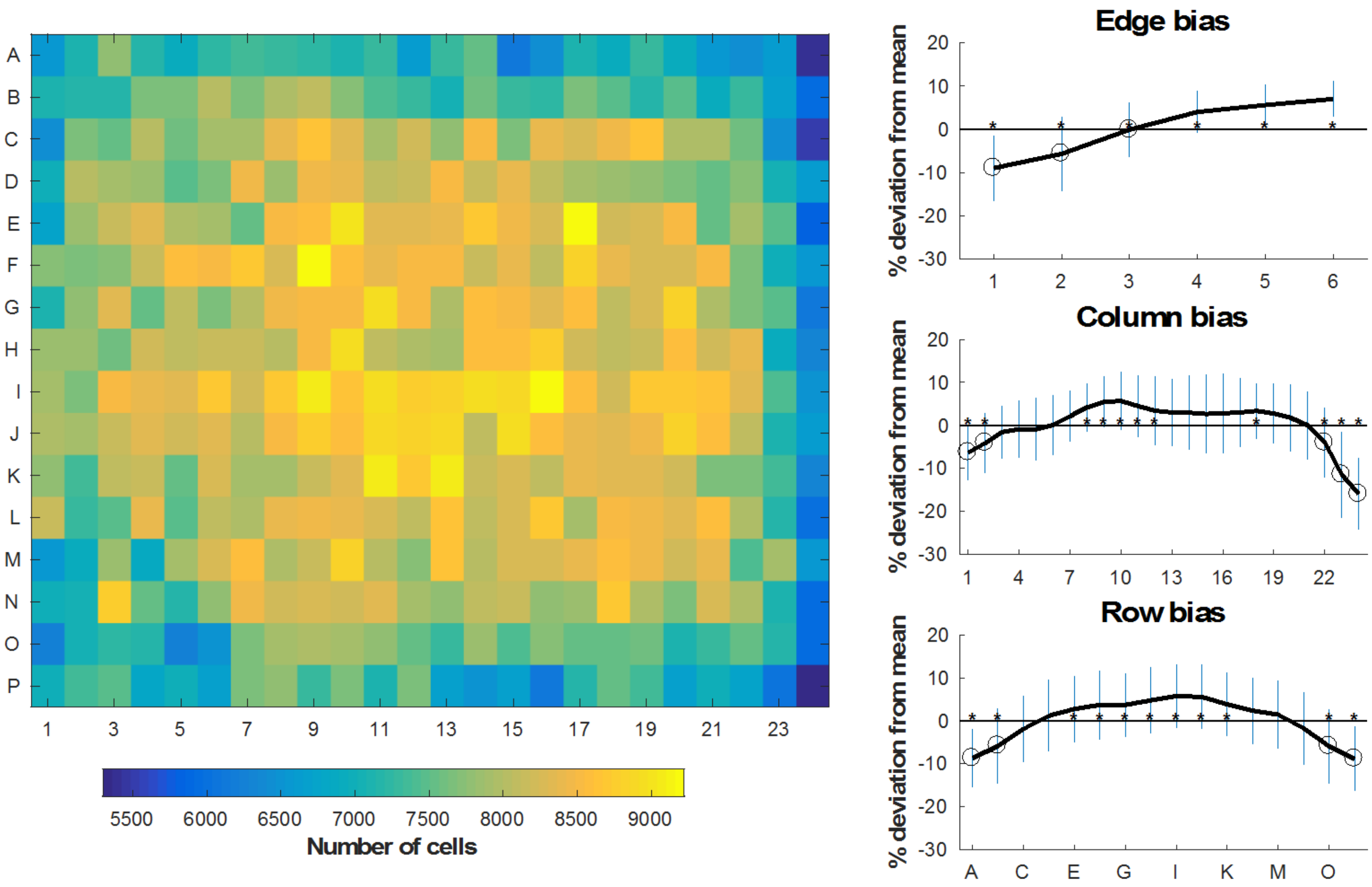
Design example: testing 2 drugs across multiple doses in 2 conditions



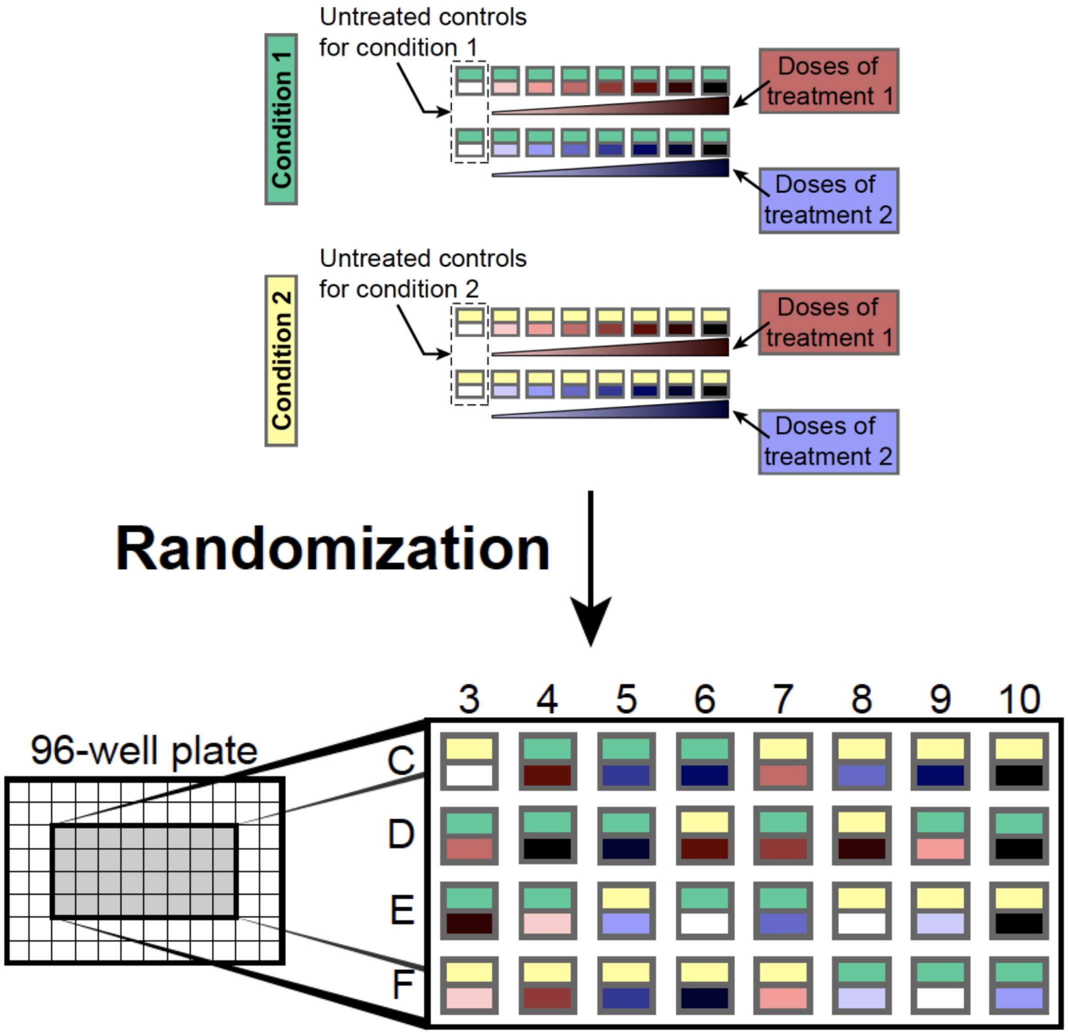
What does the pattern in the output response suggest?



Edge effects warrant randomization



Randomizing the position on the plate avoids biases and artefacts



Spreadsheets are error-prone and disconnected

Drug concentration

	A	B	C
1	0.1	0.31	3
2	0.31	1	0.31
3	0.1	1	1
4	0.31	3	0.1

Drug names

	A	B	C
1	D_1	D_5	D_4
2	D_4	D_1	D_2
3	D-5	D_1	D_2
4	D_4	D_3	D_3

Cell lines

	A	B	C
1	HeLa	HeLa	HeLa
2	MCF7	MCF7	MCF7
3	DU145	DU145	DU145
4	A375	A375	A375

Experimental design long table

Well	Cell Line	Drug
A1	HeLa	D_1
B1	HeLa	D_5
C1	HeLa	D_4
A2	MCF7	D_4
B2	MCF8	D_1
C2	MCF9	D_2

Merging experimental design with measurements

Experimental design long table

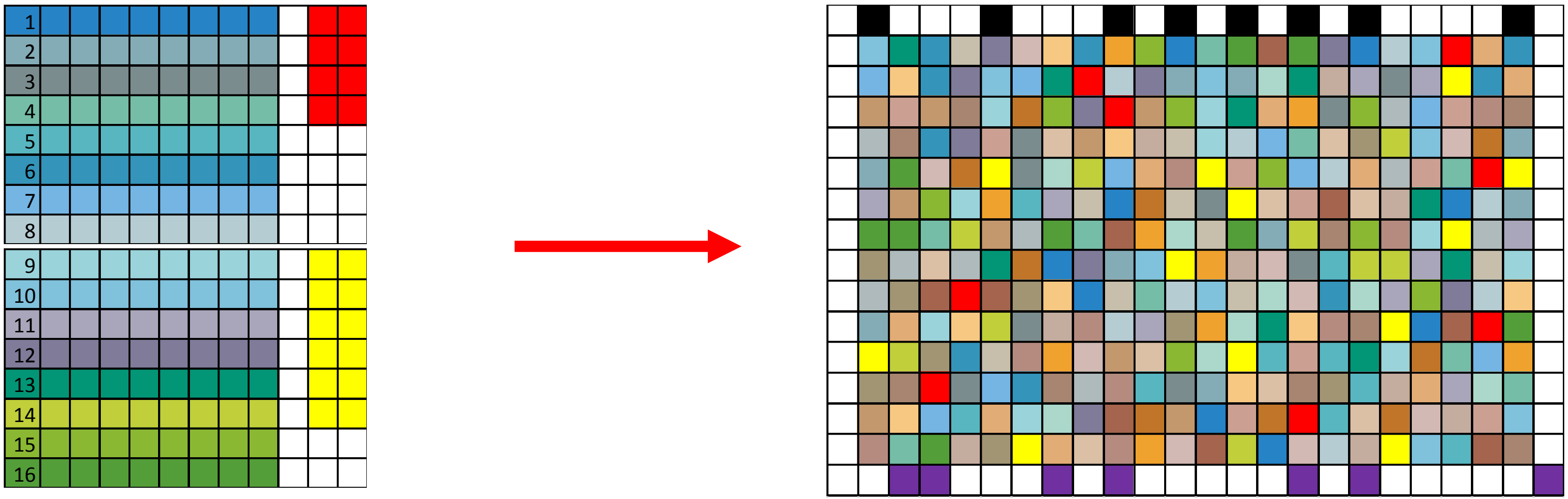
Well	Cell Line	Drug	Concentration
A1	HeLa	D_1	0.1
B1	HeLa	D_5	0.31
C1	HeLa	D_4	3
A2	MCF7	D_4	0.31
B2	MCF7	D_1	1
C2	MCF7	D_2	3

Measurement file

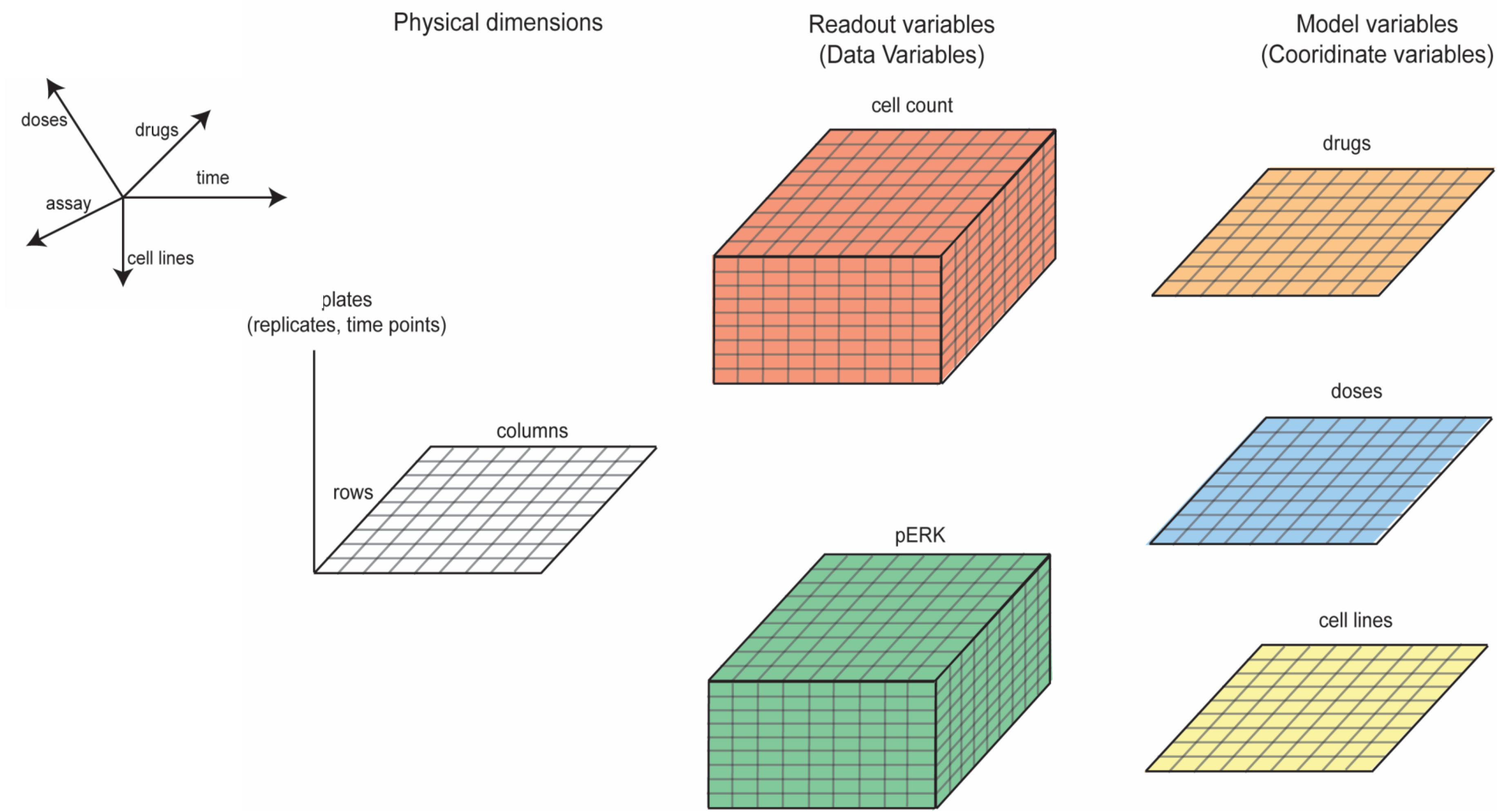
Well	Cell count
A1	2500
B1	3168
C1	2110
A2	5673
B2	4389
C2	1290

Steps to achieve reliable experimental measurements

Manual layout of drugs on a plate | **Randomized Assay Plates**



Using high-dimensional data containers in the design, storage, and analysis of drug-response experiments



Additional notes: types of variables

- Model variables:
  - Treatment variables (drug, concentration, …)
  - Condition variables (growth media, seeding density, …)

- Confounder variables:
  - Plate model
  - Assay date

- Readout variables

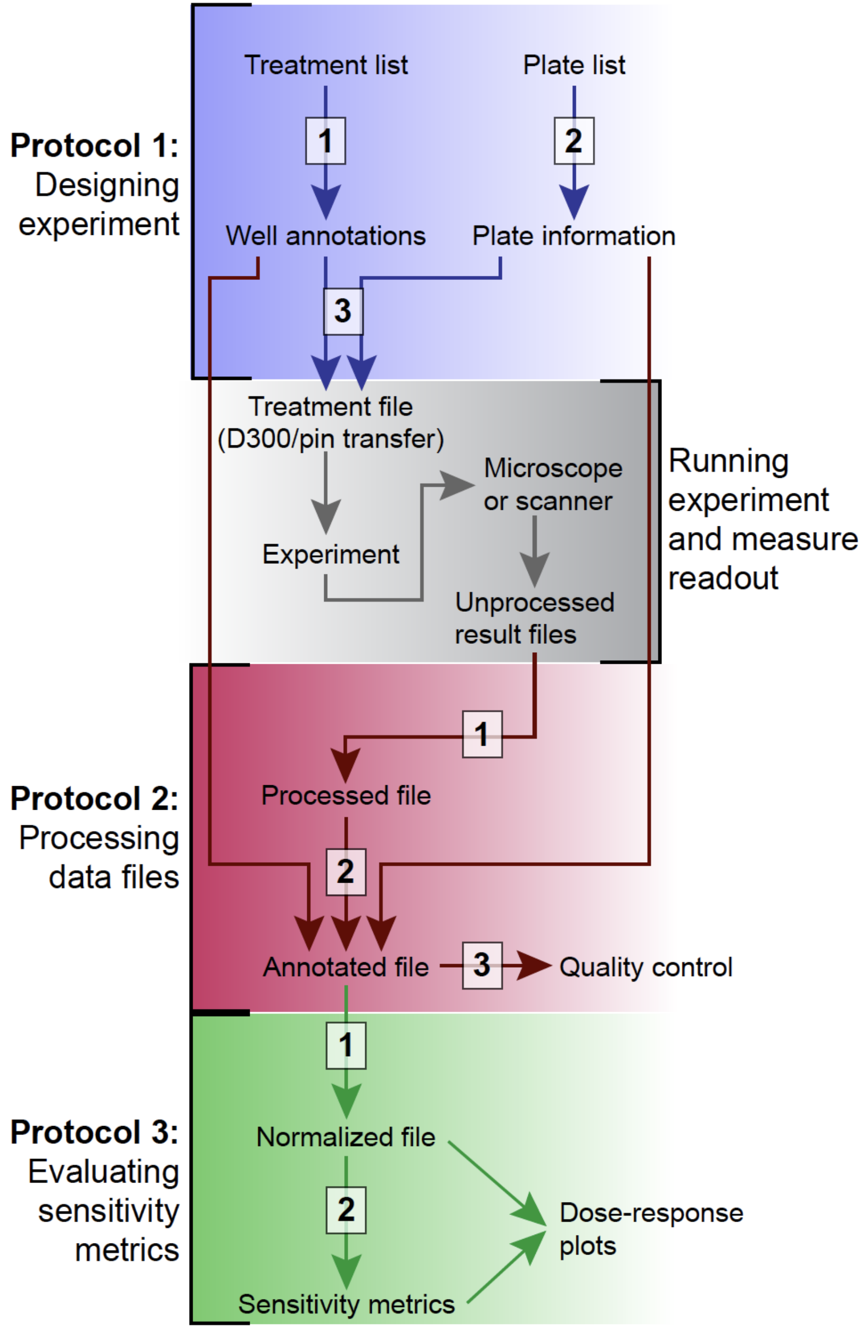
Pipeline

1. Designing experiment

   Running the experiment

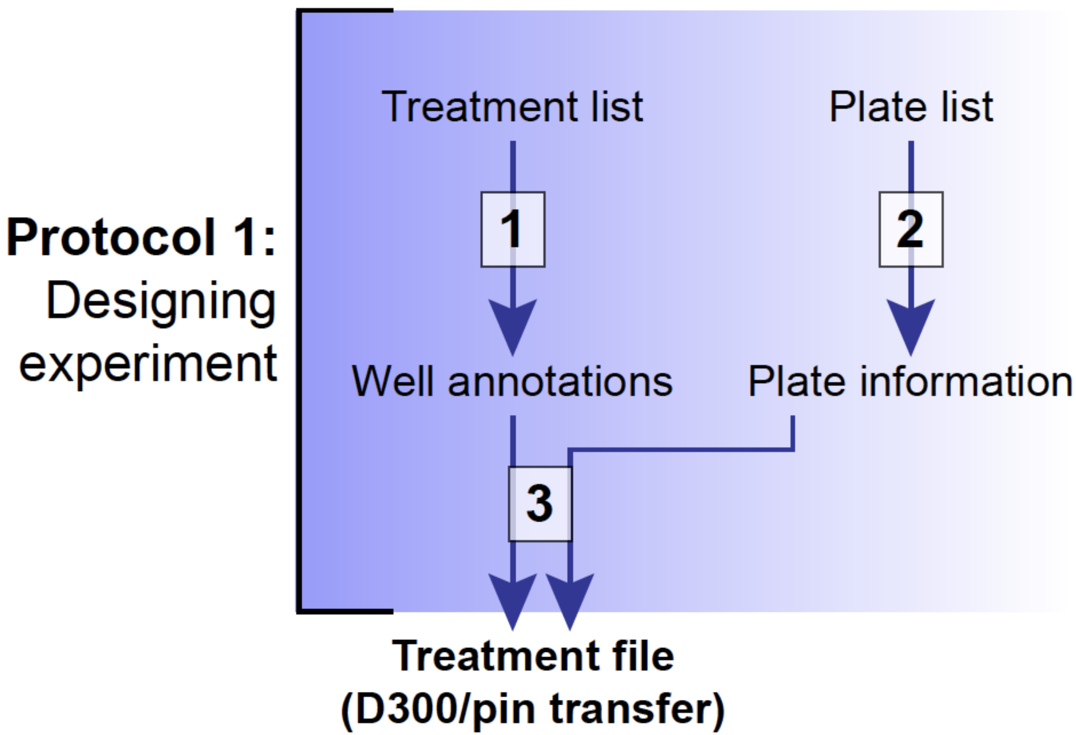
2. Quality control

3. Evaluating sensitivity metrics

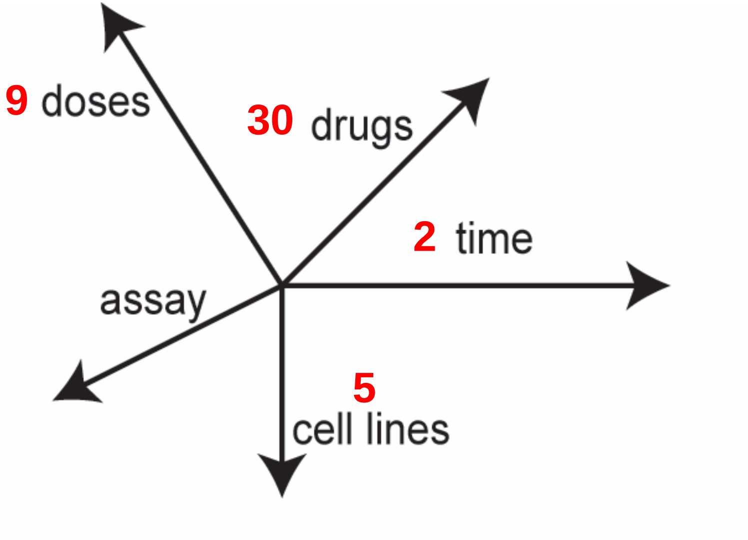


Hafner*, Niepel*, Subramanian*, Sorger
Curr Protoc Chem Biol, (2017)

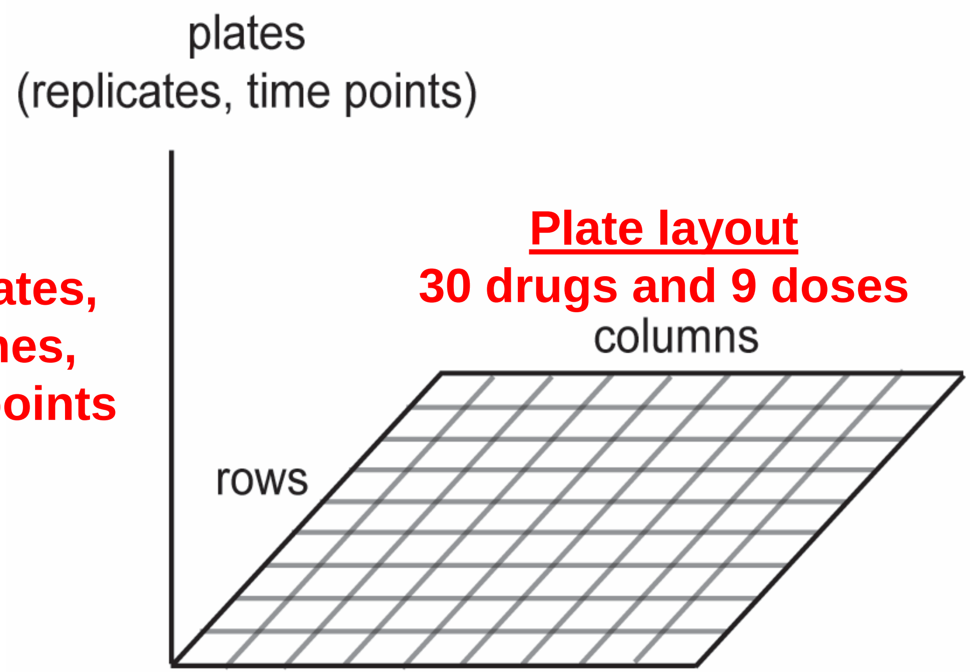
Experimental Design



Details about specific example



3 replicates,
5 cell lines,
2 time points



Use Python and Jupyter notebooks to produce the experimental design

Template for specifiying the experimental design.

The compounds, number of doses and information about the role of each compound (treatment, negative control etc) is defined in the file "compound_list.tsv". The scripts below take this tsv file as input in order to design the layout on the plate.

The size of the plate has to be provided as number of rows and columns. The number of replicates and the plate barcode are also provided in the block of code below.

Explanatory text

Design of the experiment and treatment layout (protocol 1)

```
import datarail.experimental_design.process_assay as process_assay
import datarail.experimental_design.designer as designer
import datarail.experimental_design.plot_panels as design_plot
import matplotlib.pyplot as plt
%matplotlib inline

input_file = 'INPUT/compound_list.tsv'
plate_dims = [16, 24]
fingerprint_prefix = 'DRUG_TRT_'
num_replicates = 3

treatment_dicts = process_assay.read_input(input_file, plate_dims,
                           fingerprint_prefix, encode_plate=True,
                           num_replicates=num_replicates)
```

User inputs

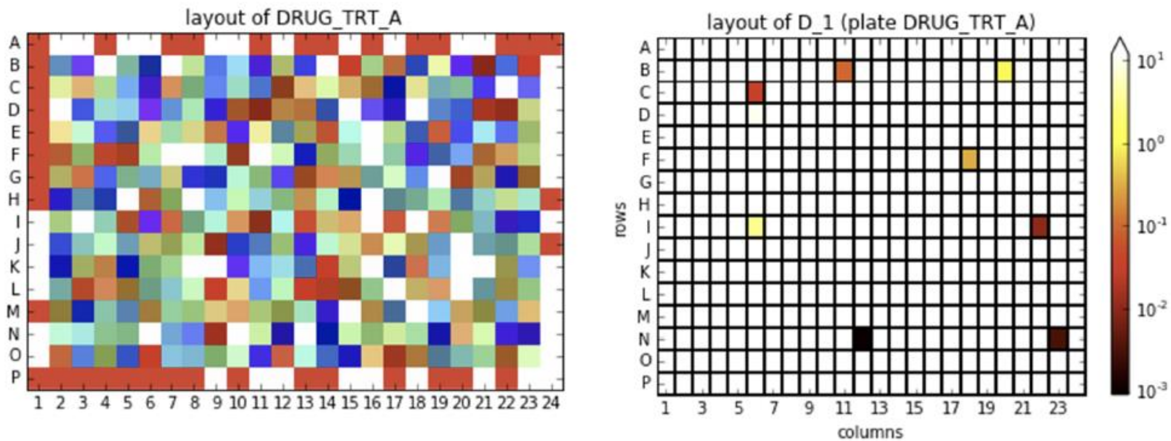
```
There are 20 untreated wells on the inner plate. Consider alloting more wells to negative con
trols
```

Warning messages

Use Jupyter notebooks to keep track of design steps and export drug layout

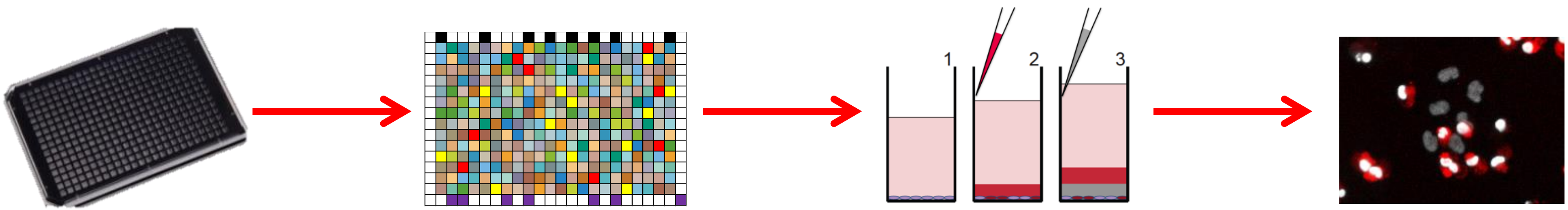
```
Designs = designer.make_layout(treatment_dicts, fingerprint_prefix,
                    encode_fingerprint=True,
                    plate_dims=plate_dims, num_replicates=num_replicates,
                    randomize=True, biased_randomization=True)

design_plot.plot_layout(Designs.sel(plates='DRUG_TRT_A'))
design_plot.plot_drug(Designs.sel(plates='DRUG_TRT_A'), 'D_1')
```

Optional inputs



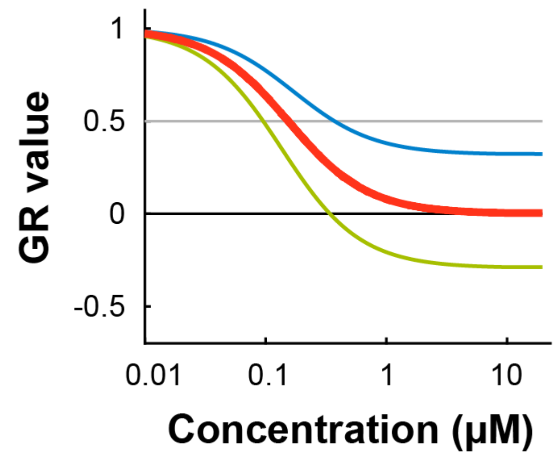
Basic experimental workflow

- Grow (happy) cells
- Seed cells at appropriate densities in multi-well plates
- Deliver drugs to multi-well plates
- Stain and fix cells
- Image cells
- Extract quantitative data from images



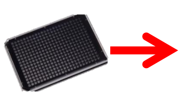
To consider *before* you start

- How many cell lines do I want to test?
  - Are they amenable to imaging?
    - Are they adherent? Do they grow in a monolayer?
  - How densely should they be seeded?
- How many drugs do I want to collect dose response data for?
  - Are they DMSO soluble?
  - What's an appropriate dose range?

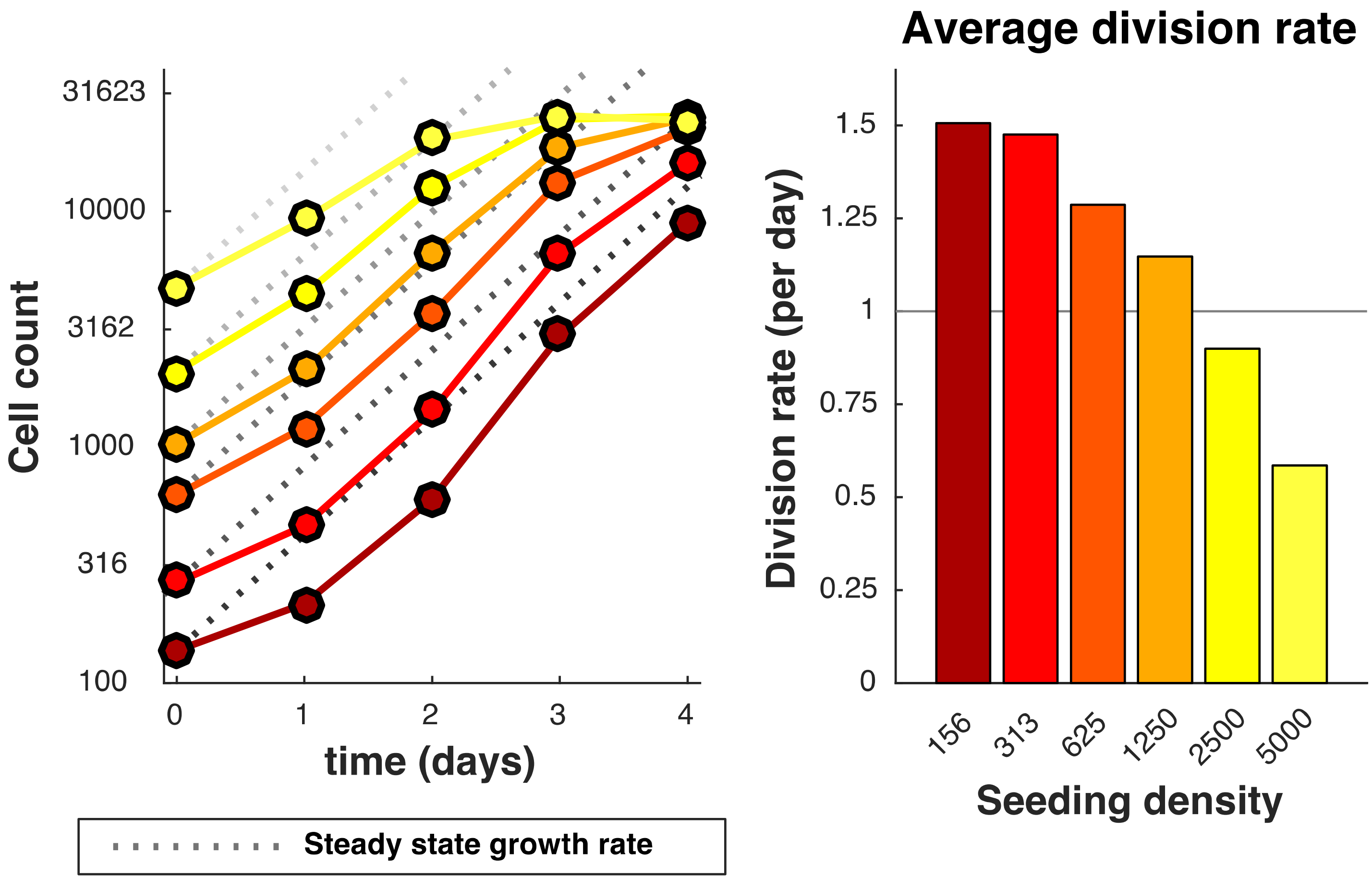


Cell seeding

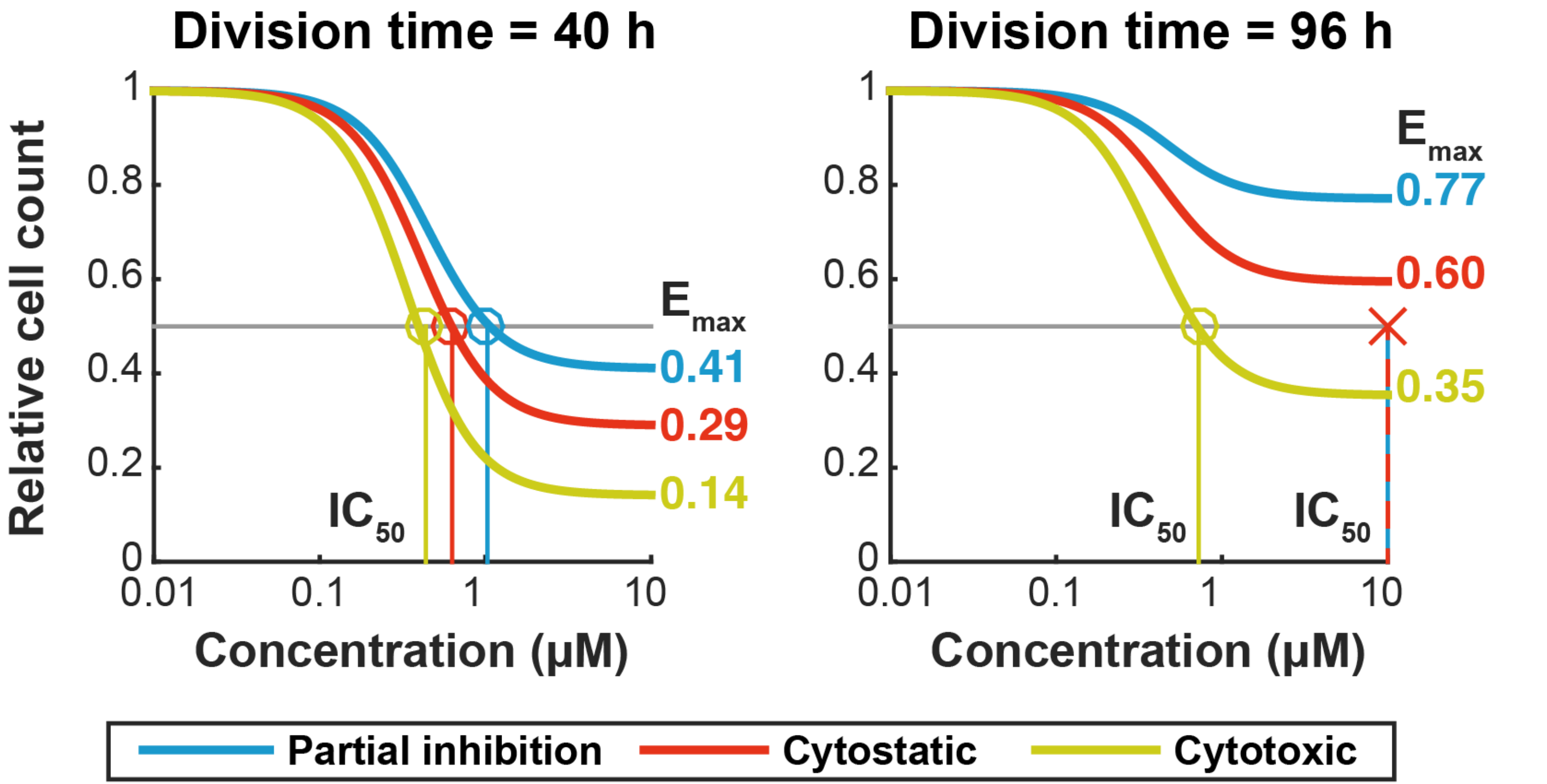
- Seed plates at an appropriate density
- Use automation if possible
- Barcode plates to keep track of them



Cell seeding density influences growth rate...



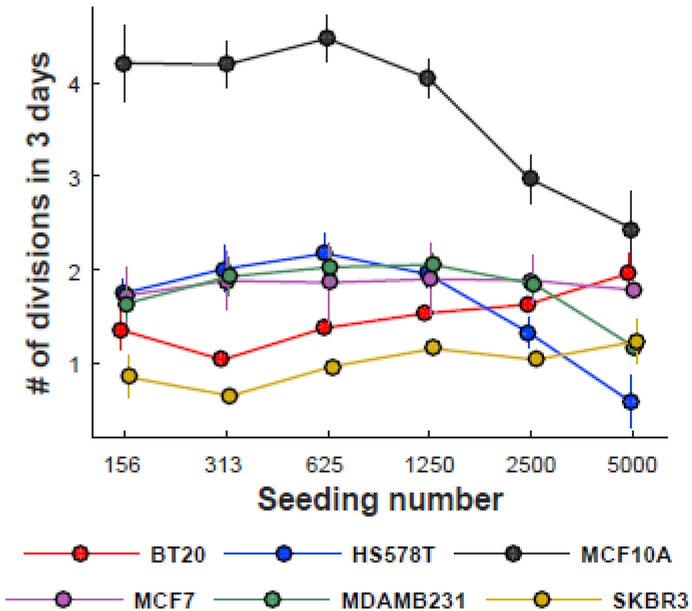
...which influences the dose response

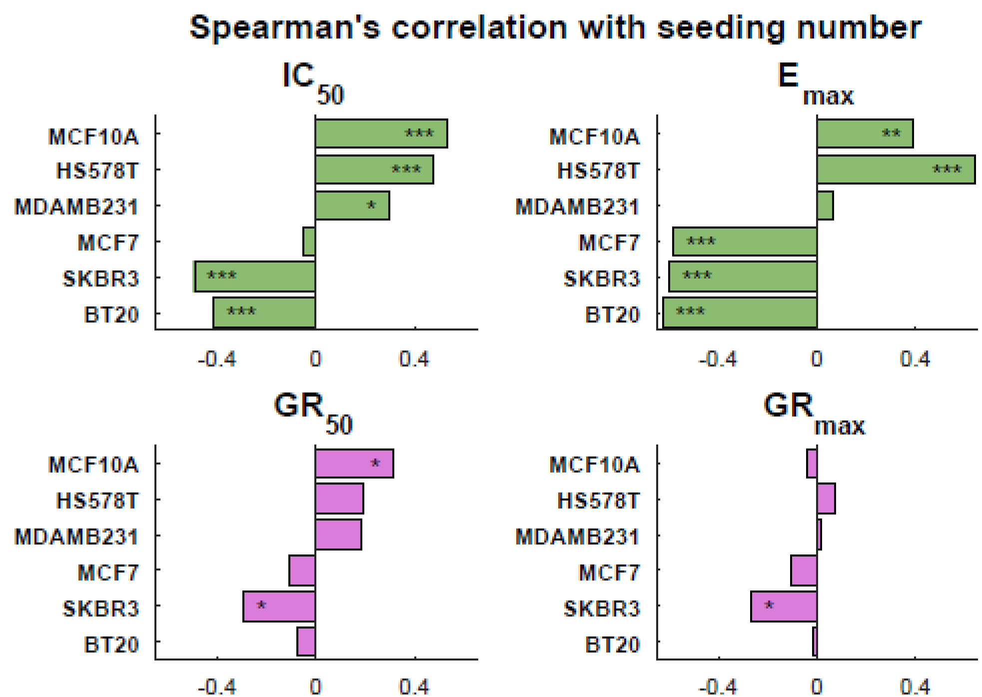


Division rate differs across densities

Seeding density affects the number of divisions.

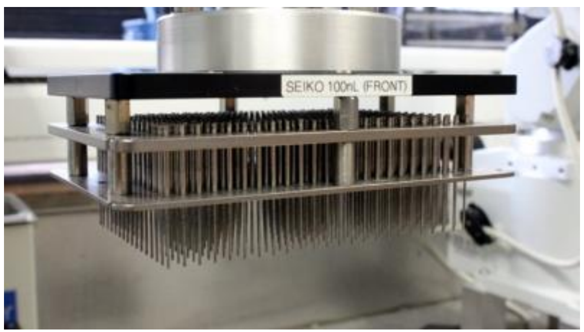
→ IC_{50} and E_{max} are correlated with density.

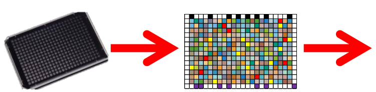




Drug delivery via pin transfer

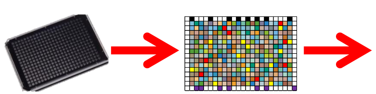
- For simultaneous delivery of many drugs
- For large scale experiments (many cell lines, conditions)
- Facilitates reproducibility



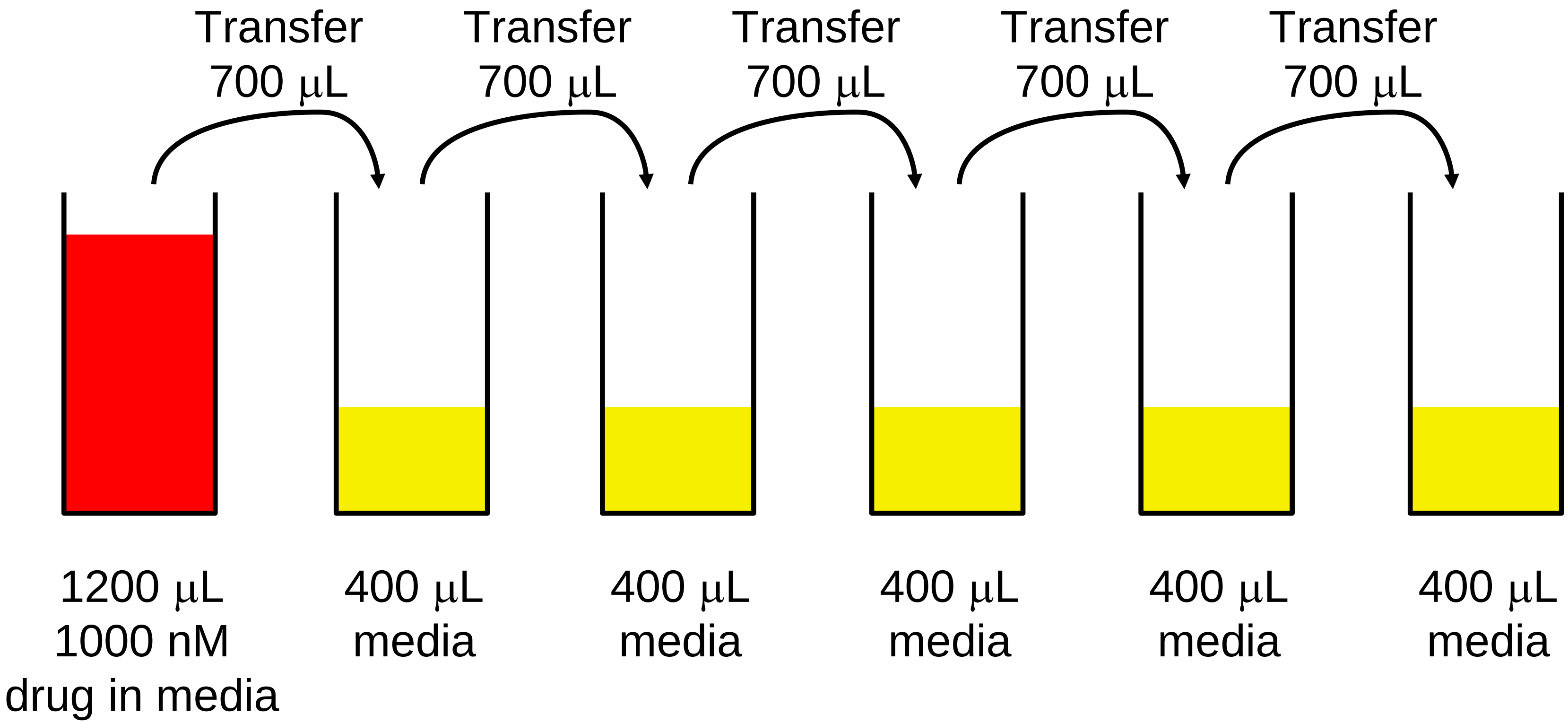


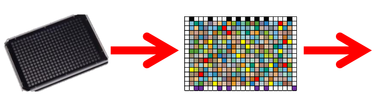
Drug delivery via digital drug dispenser

- For accurate delivery of a few drugs
- Pilot experiments- to identify appropriate doses
- Follow-up experiments, ‘hit’ validation
- Drugs that cannot be prepared in DMSO

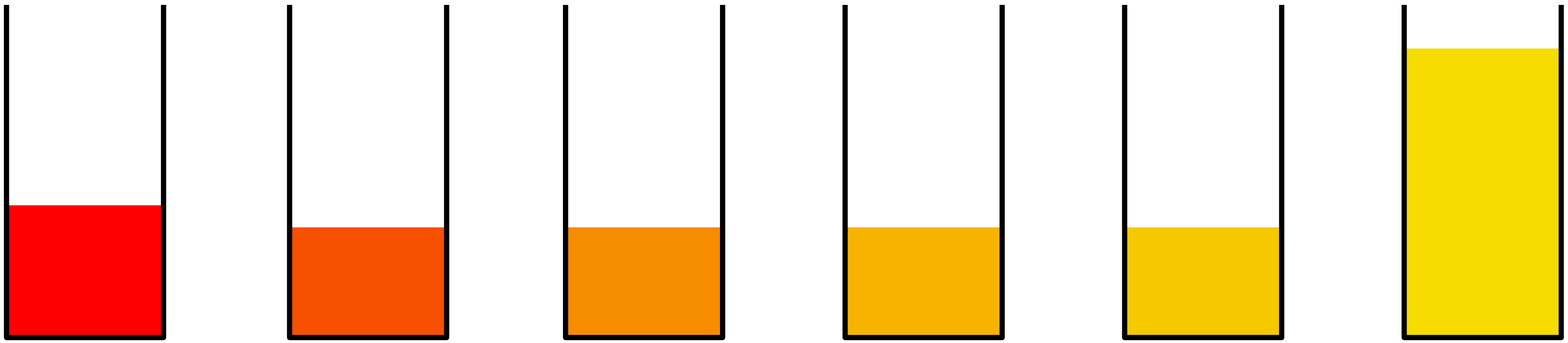


Drug delivery via manual pipetting

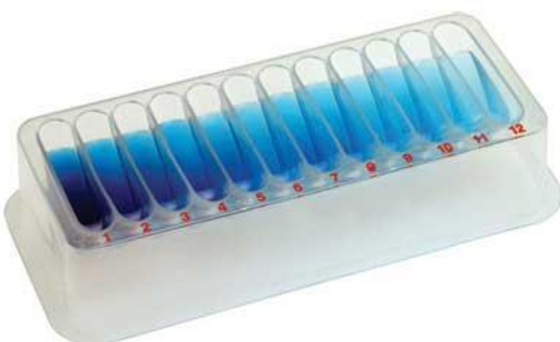




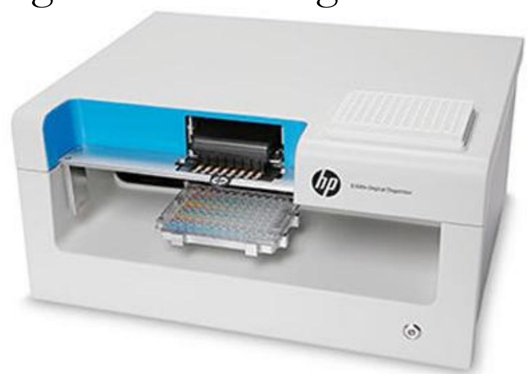
Drug delivery via manual pipetting



1000 nM drug in media | 636 nM drug | 405 nM drug | 258 nM drug | 164 nM drug | 104 nM drug



≈



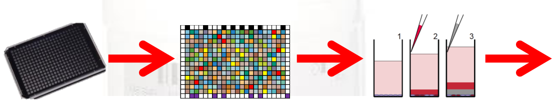
$$\left(\frac{700}{700+400}\right)^5 = 0.104$$

Other considerations

- Artefacts
  - Edge effects
    - Exclude outer wells
    - Use humidified secondary containers
    - Some cell lines are more sensitive than others
    - Depends on the duration of the experiment
  - Systematic bias from automation
- Randomization helps!

Dye-drop assay reagents

- Minimally-disruptive, reagent-sparing cell staining and fixation protocol



Dye-drop assay protocol

- Stain: Hoechst + LDR in 10% optiprep in PBS
- Fix: 4% formaldehyde in 20% optiprep in PBS

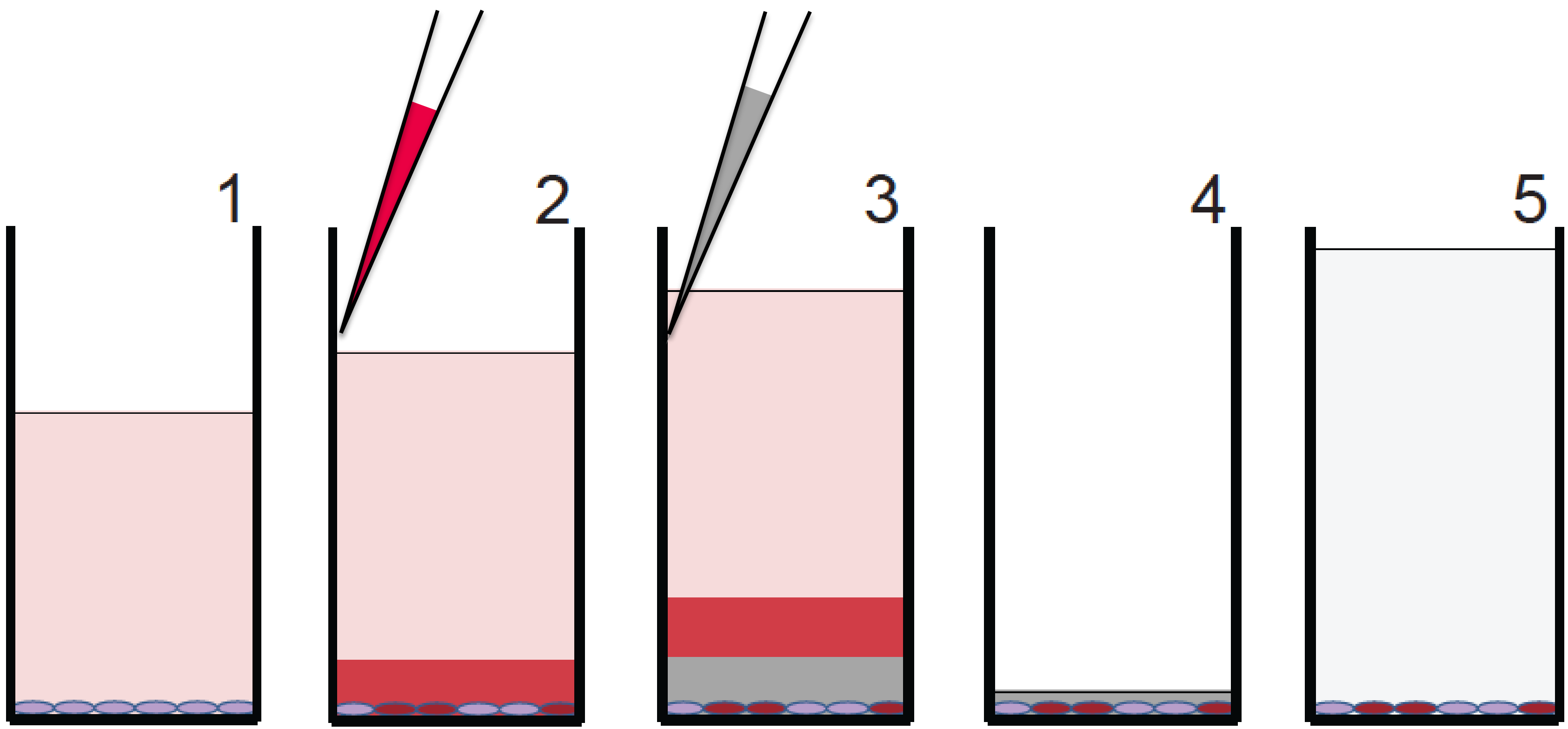


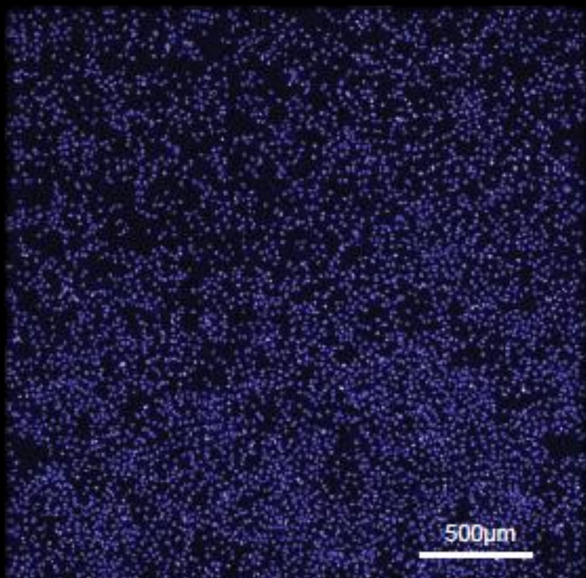
Plate washer

- Uniform and controlled aspiration and liquid dispensing

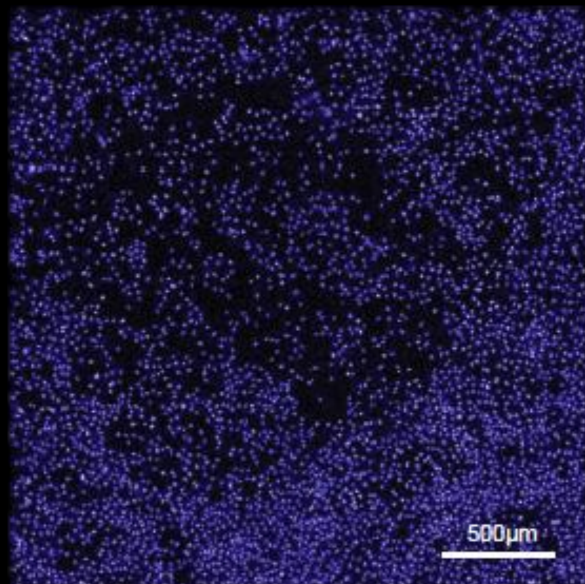
- Is repeat washing really that bad?

Repeat washing can result in cell loss…

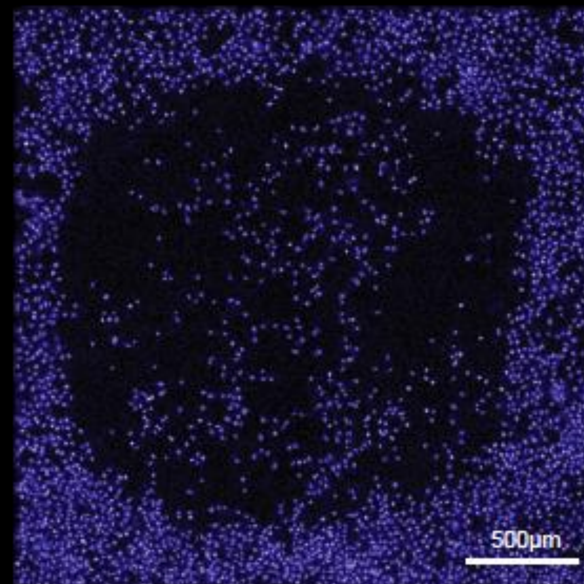
No wash



PBS wash x 1



PBS wash x 2



…that can bias your results

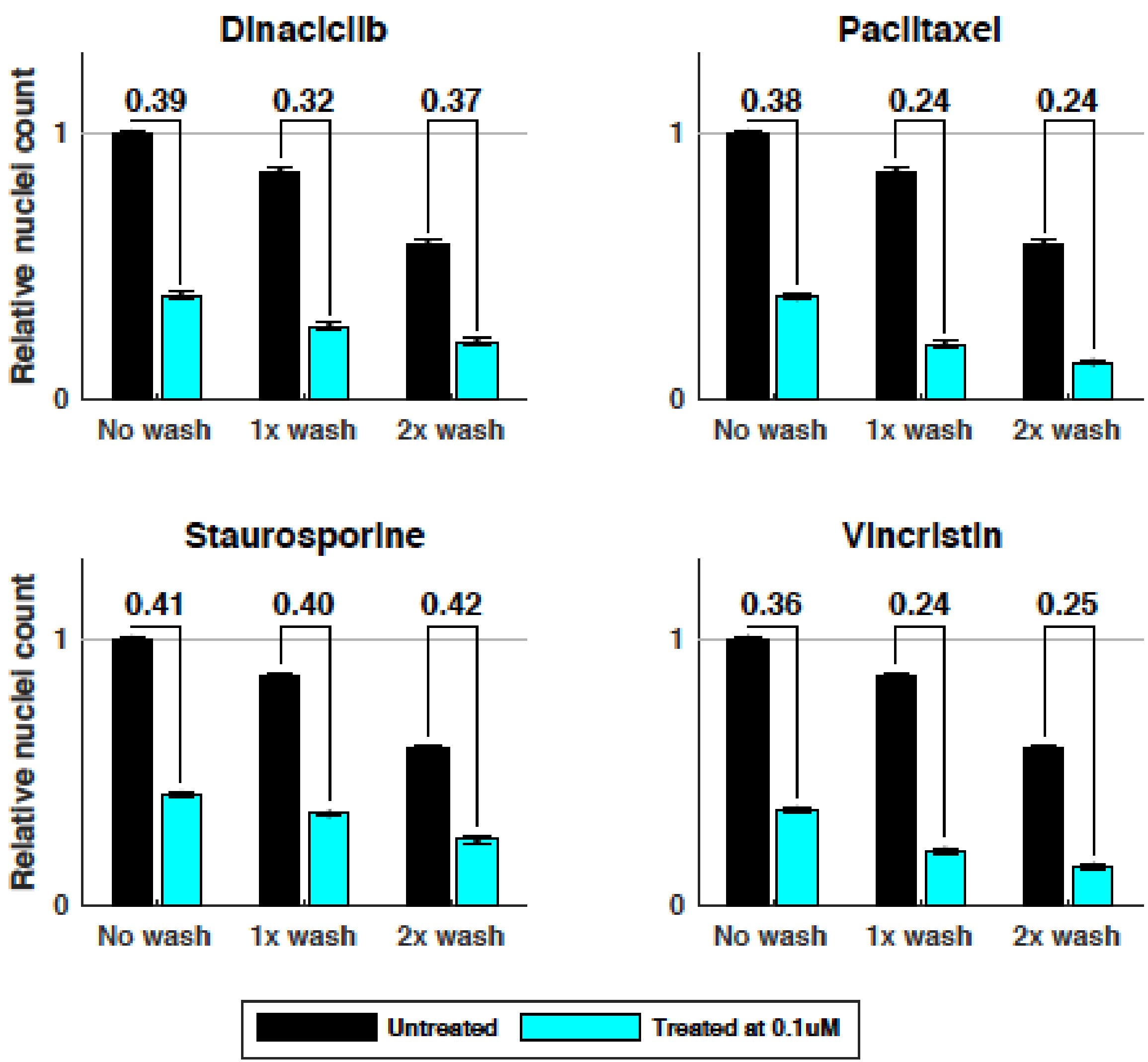


Image acquisition

- Operetta microscope with plate hotel, barcode reader & robot
  - Automated data collection for 40+ plates

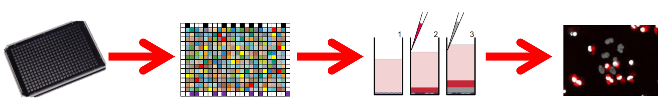
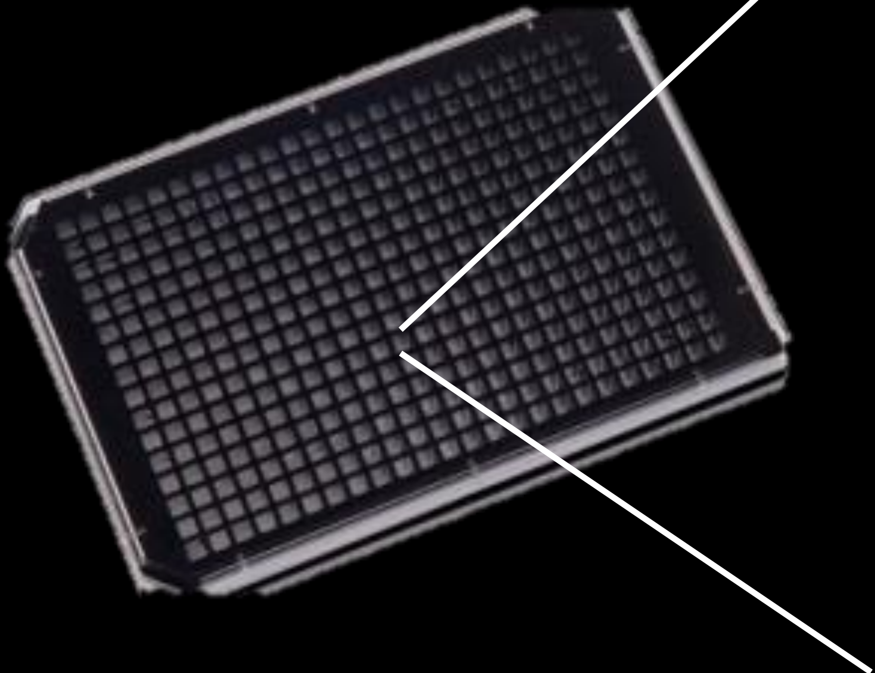
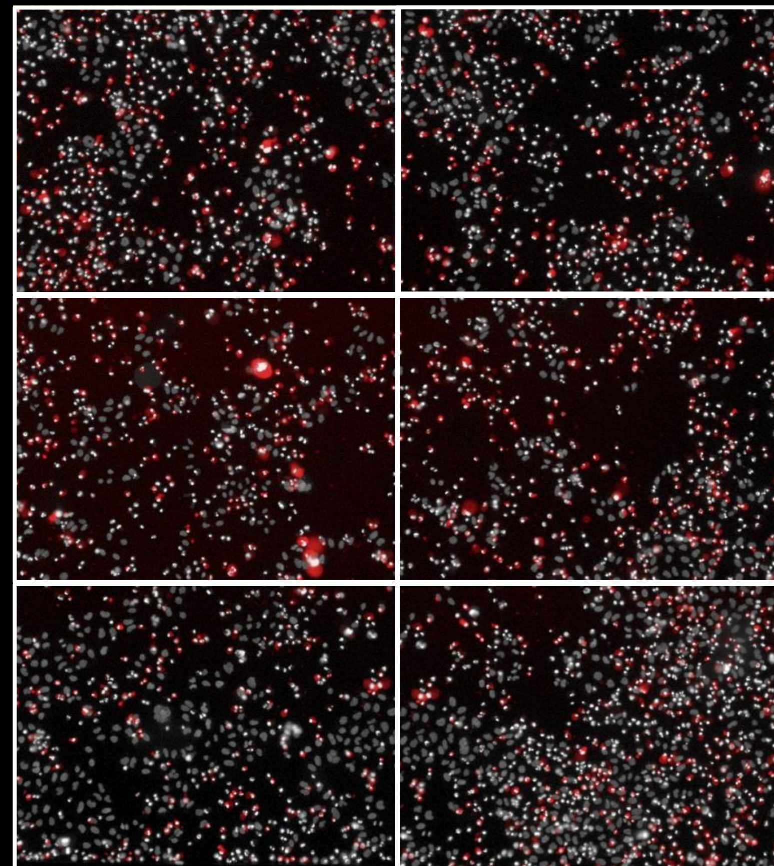


Image acquisition





Imaging 6 fields of view @ 10x captures *almost* the entire well

Image acquisition

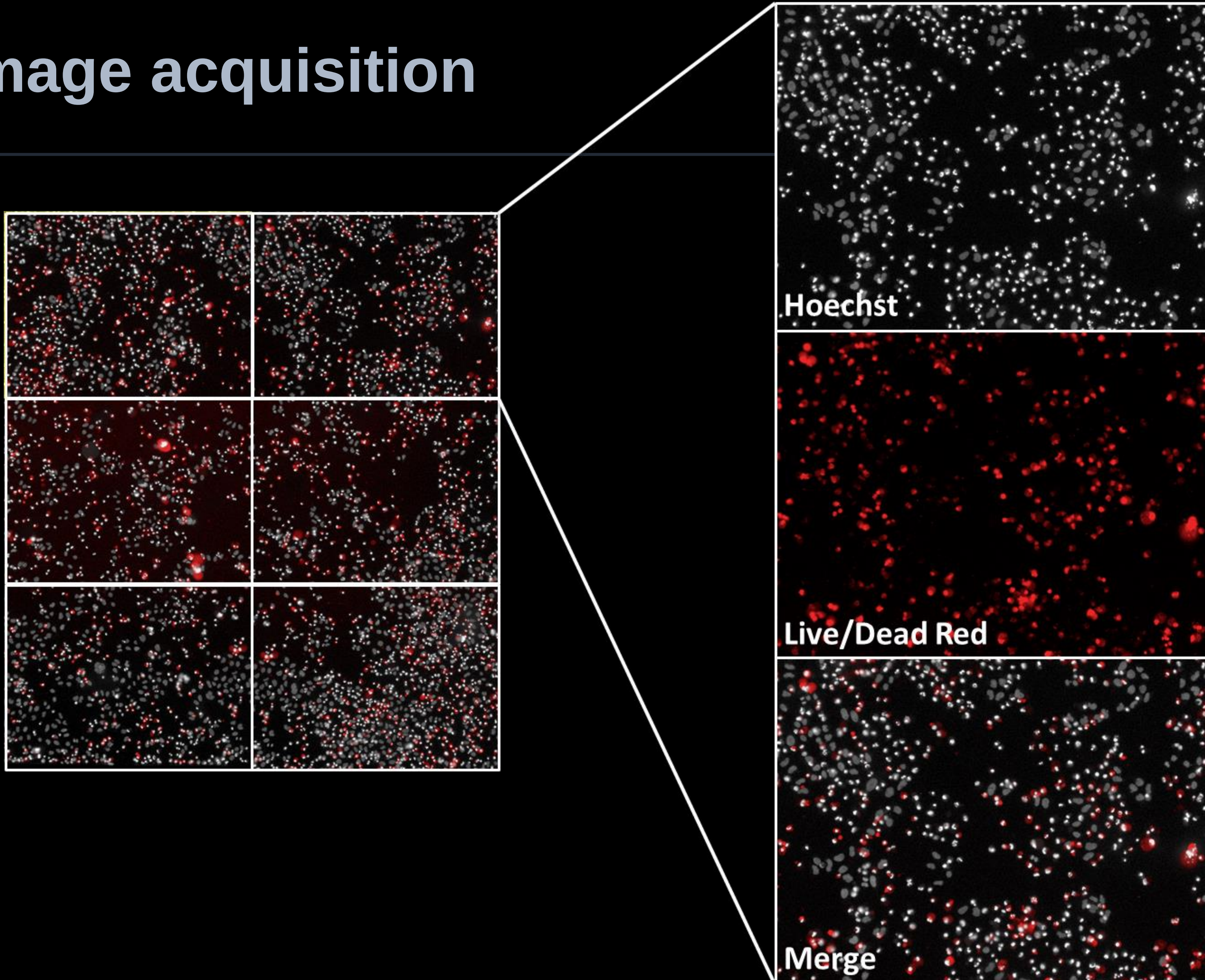
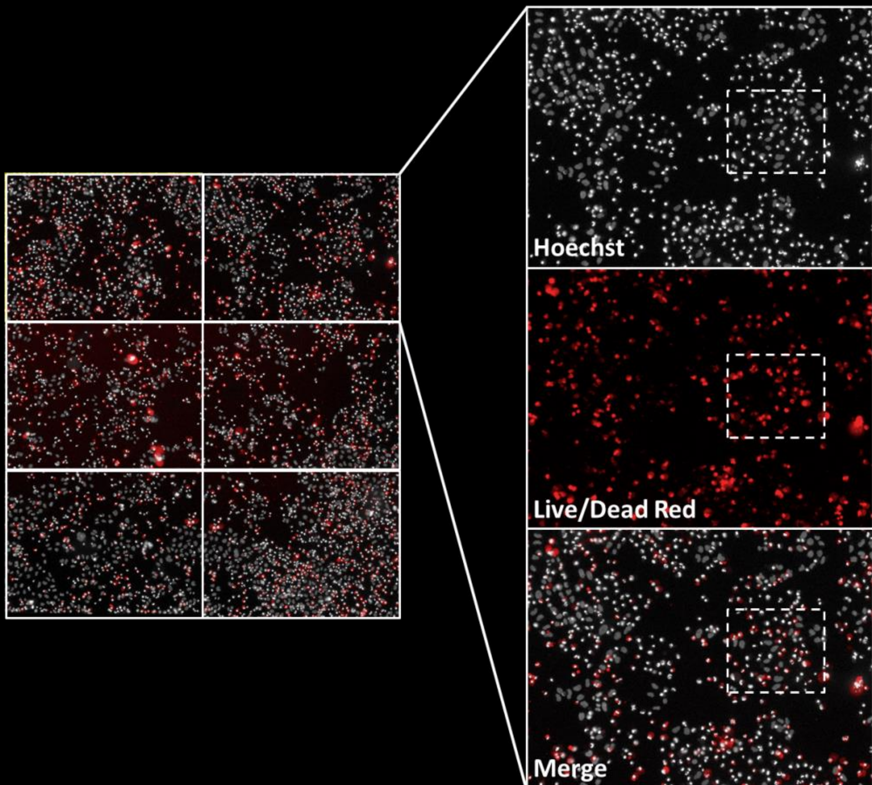
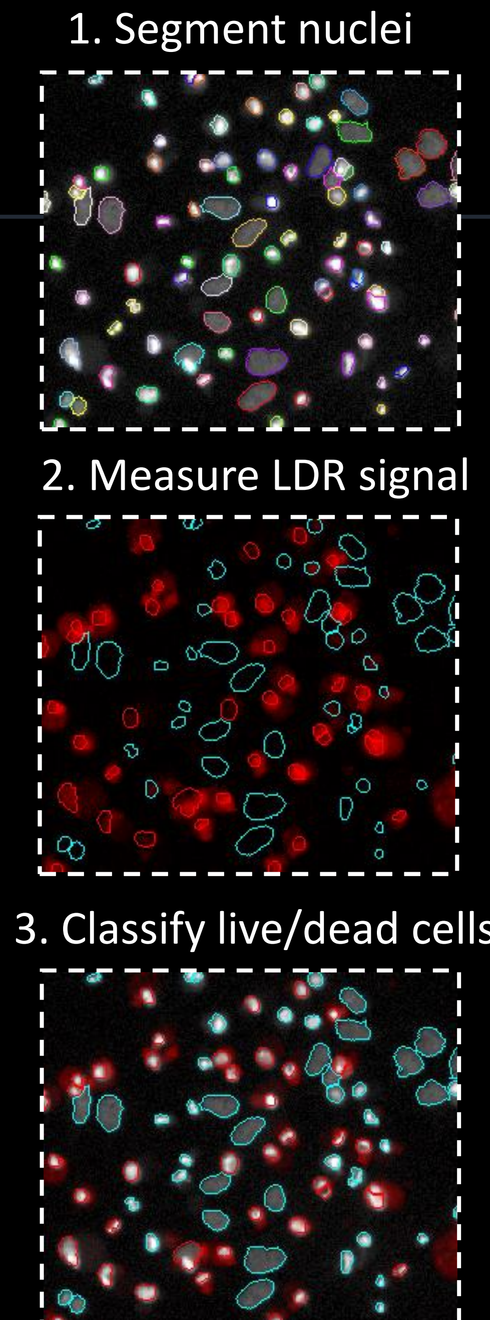


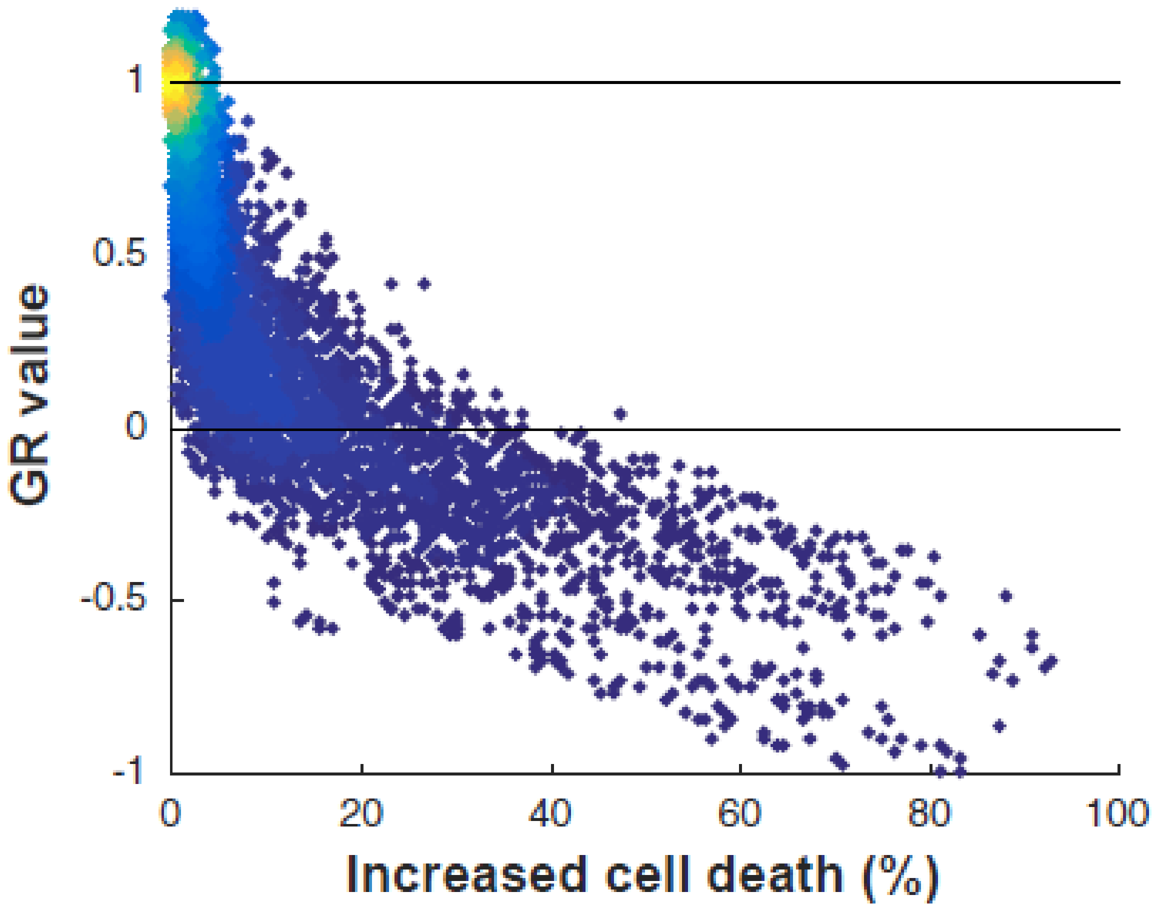
Image analysis





Well	Row	Column	Cell Line	Time point	Treatment	Dose (uM)	Cell count	Dead cell count	Cell count t=0
C2	3	2	MCF10A	72	Staurosporine	1	5091	1833	1954
C3	3	3	MCF10A	72	Staurosporine	1	5929	2137	1954
C4	3	4	MCF10A	72	Staurosporine	1	5663	2021	1954
C5	3	5	MCF10A	72	DMSO	0	8000	297	1954
C6	3	6	MCF10A	72	Staurosporine	0.316	6613	1142	1954
C7	3	7	MCF10A	72	DMSO	0	7732	329	1954
C8	3	8	MCF10A	72	Staurosporine	1	5463	2473	1954
D2	4	2	MCF10A	72	DMSO	0	8746	88	1954
D3	4	3	MCF10A	72	Staurosporine	0.316	6168	1496	1954
D4	4	4	MCF10A	72	Staurosporine	0.1	7941	636	1954
D5	4	5	MCF10A	72	DMSO	0	8529	360	1954
D6	4	6	MCF10A	72	Staurosporine	0.316	6994	1157	1954
D7	4	7	MCF10A	72	DMSO	0	8872	160	1954
D8	4	8	MCF10A	72	DMSO	0	9166	73	1954
C2	3	2	MCF10A	72	Staurosporine	1	5091	1833	1954
C3	3	3	MCF10A	72	Staurosporine	1	5929	2137	1954
C4	3	4	MCF10A	72	Staurosporine	1	5663	2021	1954
C5	3	5	MCF10A	72	DMSO	0	8000	297	1954
C6	3	6	MCF10A	72	Staurosporine	0.316	6613	1142	1954
C7	3	7	MCF10A	72	DMSO	0	7732	329	1954
C8	3	8	MCF10A	72	Staurosporine	1	5463	2473	1954
D2	4	2	MCF10A	72	DMSO	0	8746	88	1954
D3	4	3	MCF10A	72	Staurosporine	0.316	6168	1496	1954
D4	4	4	MCF10A	72	Staurosporine	0.1	7941	636	1954
D5	4	5	MCF10A	72	DMSO	0	8529	360	1954
D6	4	6	MCF10A	72	Staurosporine	0.316	6994	1157	1954
D7	4	7	MCF10A	72	DMSO	0	8872	160	1954
D8	4	8	MCF10A	72	DMSO	0	9166	73	1954
C2	3	2	MCF10A	72	Staurosporine	1	5091	1833	1954
C3	3	3	MCF10A	72	Staurosporine	1	5929	2137	1954
C4	3	4	MCF10A	72	Staurosporine	1	5663	2021	1954
C5	3	5	MCF10A	72	DMSO	0	8000	297	1954
C6	3	6	MCF10A	72	Staurosporine	0.316	6613	1142	1954
C7	3	7	MCF10A	72	DMSO	0	7732	329	1954
C8	3	8	MCF10A	72	Staurosporine	1	5463	2473	1954
D2	4	2	MCF10A	72	DMSO	0	8746	88	1954
D3	4	3	MCF10A	72	Staurosporine	0.316	6168	1496	1954
D4	4	4	MCF10A	72	Staurosporine	0.1	7941	636	1954
D5	4	5	MCF10A	72	DMSO	0	8529	360	1954
D6	4	6	MCF10A	72	Staurosporine	0.316	6994	1157	1954
D7	4	7	MCF10A	72	DMSO	0	8872	160	1954
D8	4	8	MCF10A	72	DMSO	0	9166	73	1954
C2	3	2	MCF10A	72	Staurosporine	1	5091	1833	1954
C3	3	3	MCF10A	72	Staurosporine	1	5929	2137	1954
C4	3	4	MCF10A	72	Staurosporine	1	5663	2021	1954
C5	3	5	MCF10A	72	DMSO	0	8000	297	1954
C6	3	6	MCF10A	72	Staurosporine	0.316	6613	1142	1954
C7	3	7	MCF10A	72	DMSO	0	7732	329	1954
C8	3	8	MCF10A	72	Staurosporine	1	5463	2473	1954
D2	4	2	MCF10A	72	DMSO	0	8746	88	1954
D3	4	3	MCF10A	72	Staurosporine	0.316	6168	1496	1954
D4	4	4	MCF10A	72	Staurosporine	0.1	7941	636	1954
D5	4	5	MCF10A	72	DMSO	0	8529	360	1954
D6	4	6	MCF10A	72	Staurosporine	0.316	6994	1157	1954
D7	4	7	MCF10A	72	DMSO	0	8872	160	1954
D8	4	8	MCF10A	72	DMSO	0	9166	73	1954
C2	3	2	MCF10A	72	Staurosporine	1	5091	1833	1954
C3	3	3	MCF10A	72	Staurosporine	1	5929	2137	1954
C4	3	4	MCF10A	72	Staurosporine	1	5663	2021	1954
C5	3	5	MCF10A	72	DMSO	0	8000	297	1954
C6	3	6	MCF10A	72	Staurosporine	0.316	6613	1142	1954
C7	3	7	MCF10A	72	DMSO	0	7732	329	1954
C8	3	8	MCF10A	72	Staurosporine	1	5463	2473	1954
D2	4	2	MCF10A	72	DMSO	0	8746	88	1954
D3	4	3	MCF10A	72	Staurosporine	0.316	6168	1496	1954
D4	4	4	MCF10A	72	Staurosporine	0.1	7941	636	1954
D5	4	5	MCF10A	72	DMSO	0	8529	360	1954
D6	4	6	MCF10A	72	Staurosporine	0.316	6994	1157	1954
D7	4	7	MCF10A	72	DMSO	0	8872	160	1954
D8	4	8	MCF10A	72	DMSO	0	9166	73	1954

Can I just count cells?

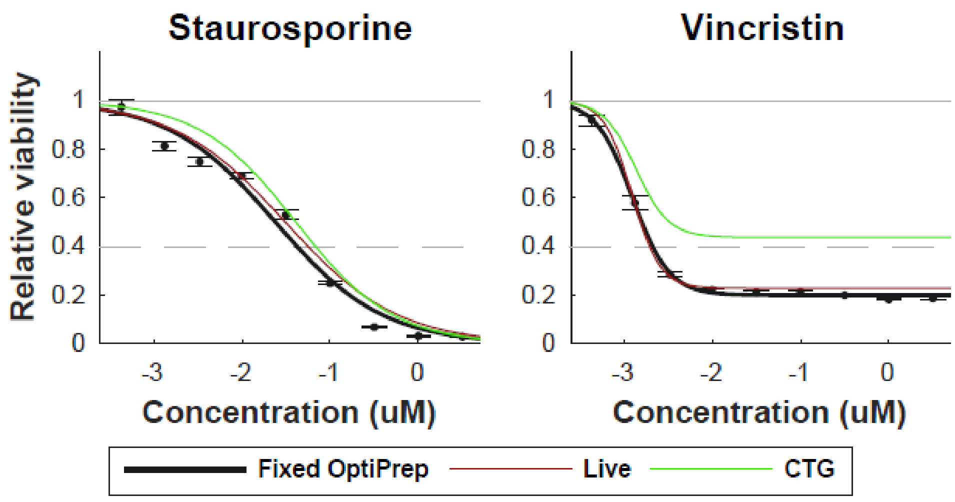


Strengths and limitations

- Imaging based
  - Best suited for adherent cells that grow in monolayer culture
- Image analysis can be time consuming
- Can go back and visually inspect imaging data
- Potential for multiplexing, immunofluorescence
- Fate of live cells unknown
- Reagent sparing
- Distinction between cytotoxic and cytostatic effects

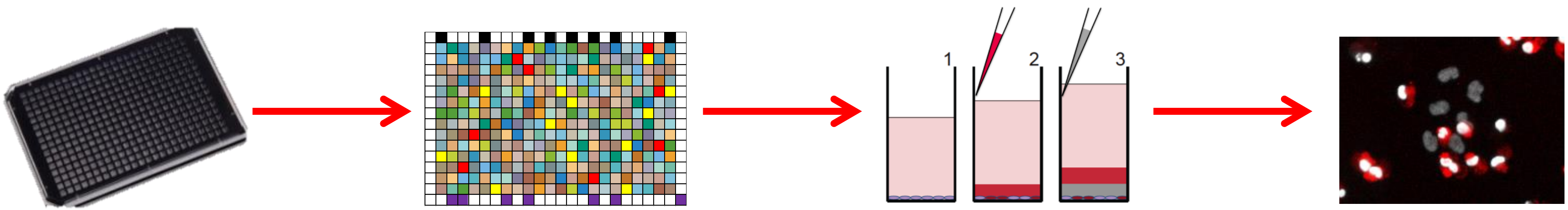
Other assays

- CellTiter-Glo etc.
  - Simple, no wash protocol
  - Luminescence read-out, simple analysis, rapid results
  - Treatment-induced changes in metabolic activity of cells can skew results
- Measurement of confluency
  - Inaccurate
  - Treatment-induced changes in morphology can skew results

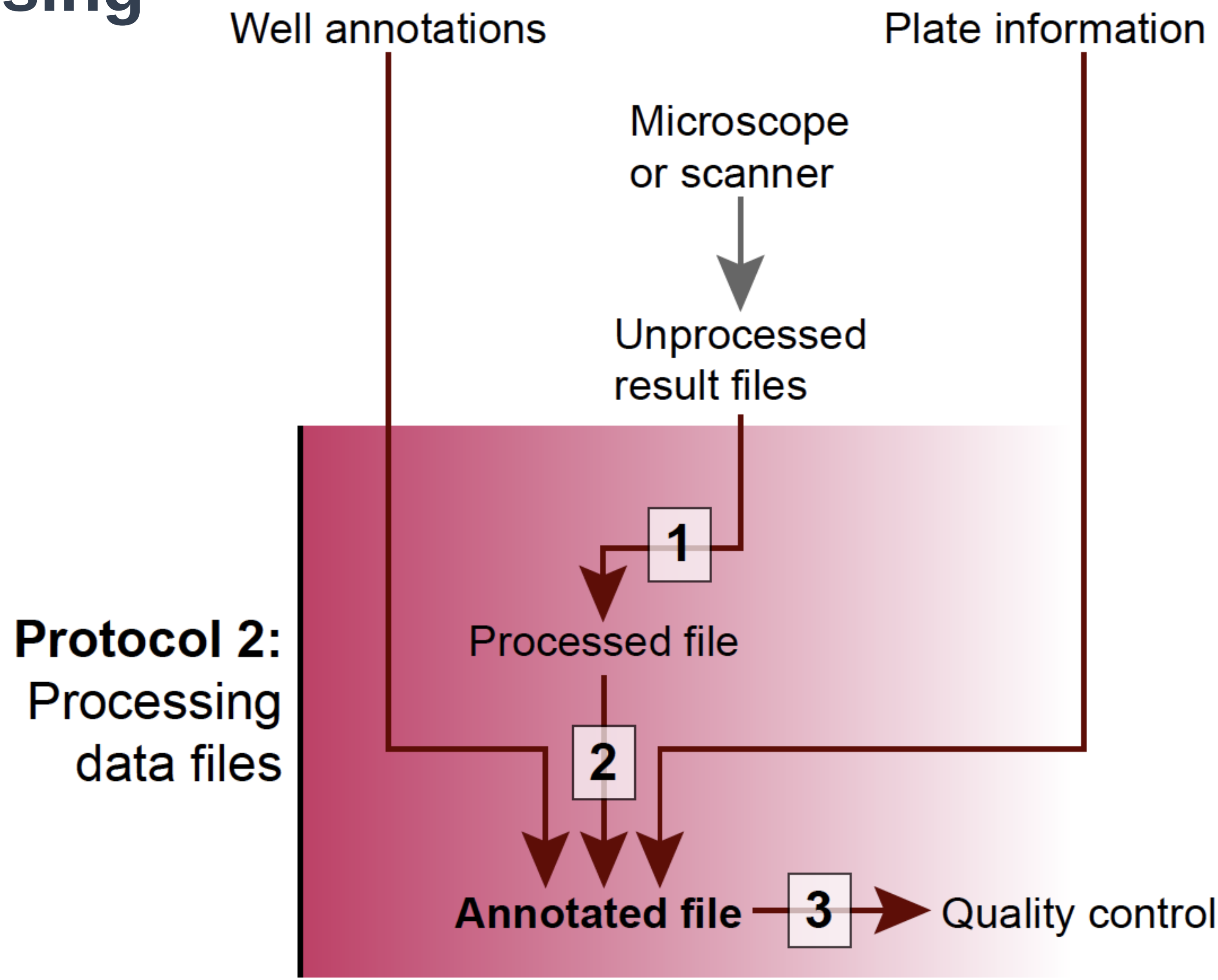


Take away messages

- Include a t=0 plate
- Optimize conditions
  - Seeding density per cell line
  - Dose range per drug
  - Duration of assay
- Automate as much as possible



Processing


Well annotations
Plate information
Microscope
or scanner
Unprocessed
result files
1
Processed file
2
Annotated file
3
Quality control
Protocol 2:
Processing
data files

Use Jupyter notebooks to import and annotate results from experiments

Data import and annotation (protocol 2, steps 1 and 2)

```
# load the synthetic data (Columbus output)
df=CoImp.Columbus_processing('../tests/drug_response_data/OUTPUT/Example1_Columbus_output.tsv',

                              (('Hoechst_pos', 'cell_count__total'),
                               ('LDR_pos_Hoechst_neg', 'corpse_count'),
                               ('Hoechst_LDR_pos', 'cell_count__dead')),
                              'cell_count__total - cell_count__dead')

# annotate the data
df_annotated = TrtAnnot.add_treatments(
    TrtAnnot.add_plate_info(df,
        pd.read_csv('../tests/drug_response_data/OUTPUT/Example1_plate_info.tsv', sep='\t')),
    '../tests/drug_response_data/OUTPUT/')

df_annotated.to_csv('OUTPUT/AnnotatedData_Example1.tsv', sep='\t', index=False)
```

```
Default number of fields: 6 ; 505 wells with missing field(s)
Concentrations rounded in the log domain
```

Data file

Readout variables

Folder containing well annotation files

Annotated file

Check for unwanted biases using embedded functions

Quality control (protocol 2, step 3)

```
import datarail.data_processing.drug_response.qc_plate as qcfct

# use the raw data to perform the plate QC
qcfct.Plate_bias(pltfct.dfplate2xr(df), filename='OUTPUT/QC_report_Example1.pdf')
# use the annotated data to perform QC based on the negative controls
qcfct.Negative_control_bias(df_annotated)
```

QC report file

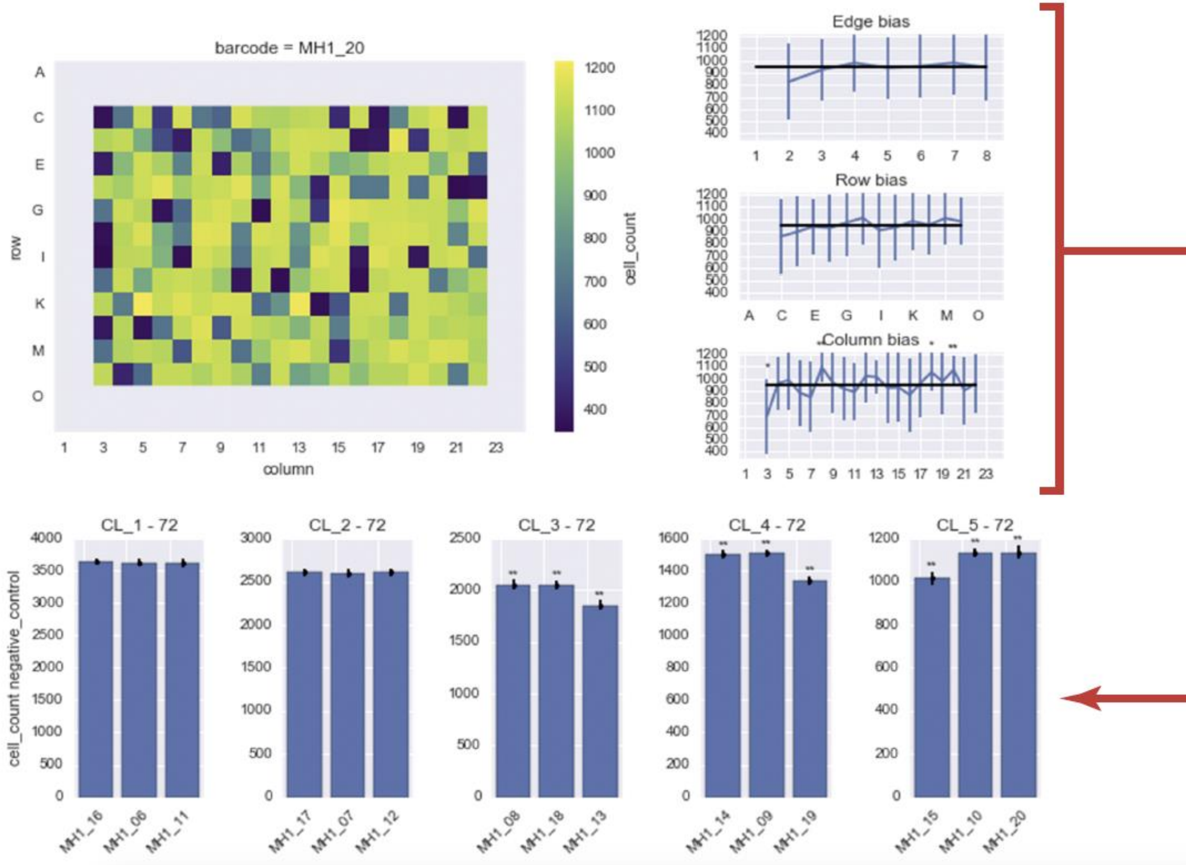
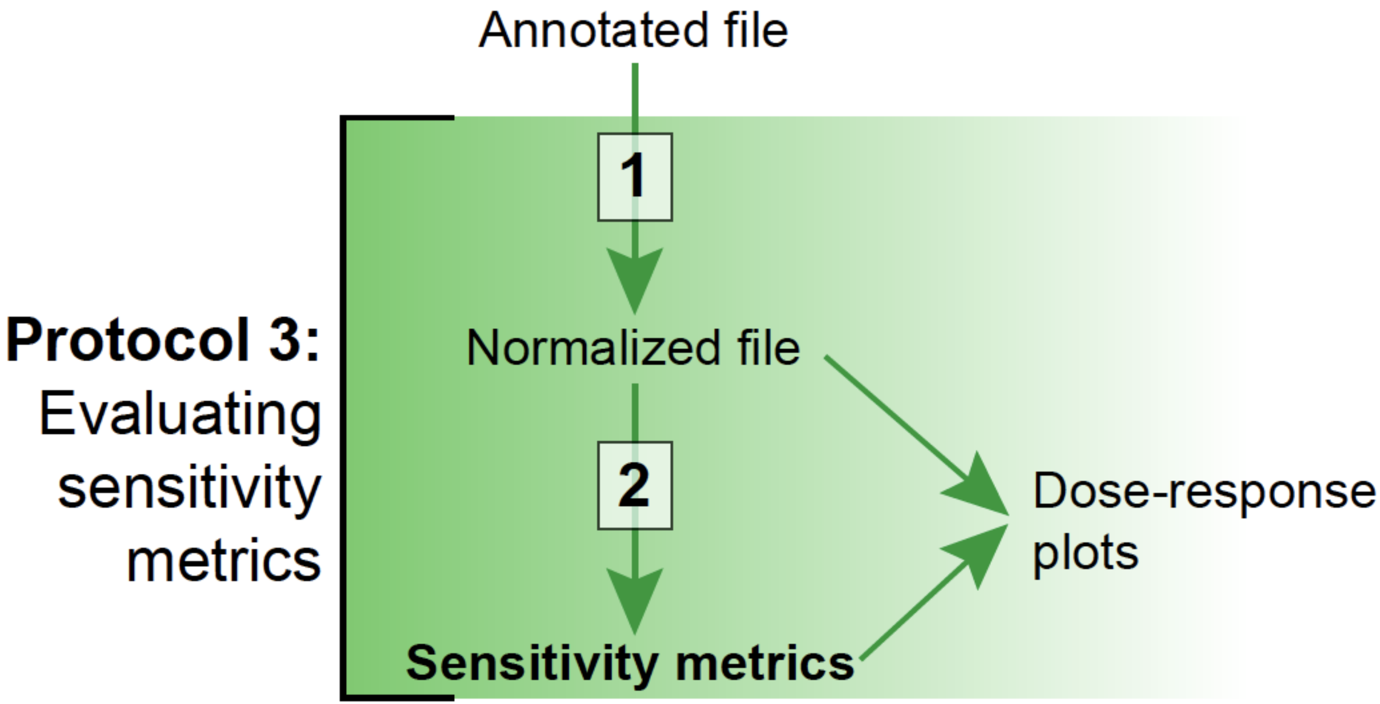


Plate bias QC

Negative control QC

Analysis: data normalization and dose-response curve parametrization



Normalize the data to obtain the GR values

Calculate the GR values (protocol 3, step 1)

```
# first calculate the GR values for each replicate then merge them
df_mean = TrtAnnot.average_replicates(
    gr50.compute_gr(
        gr50.assign_ctrls(df_annotated, ['cell_line'])))

df_gr = df_mean.drop(['cell_count__dead','corpse_count','role','cell_count__total'],axis=1)
df_gr.to_csv('OUTPUT/GRvalues_Example1.tsv', sep='\t', index=False)
df_gr.head()
```

Calculate GR values

Normalized file

```
Columns to average: "corpse_count" "cell_count__total" "cell_count" "cell_count__dead" "cell_
count__ctrl" "GRvalue" "cell_count__time0"
Columns added as annotations: "date"

-->Following columns are discarded:
 "treatment_file" "well" "barcode"
        (set as key if necessary)
```

	cell_line	treatment_duration	concentration	agent	date	cell_count	cell_count__ctrl	GRvalue	cell_count__t
0	CL_1	72.0	0.001	D_1	2016-06-06 12:34:56	3583.444444	3627.85	0.991393	491.525
1	CL_1	72.0	0.001	D_2	2016-06-06 12:34:56	3612.000000	3627.85	0.996945	491.525

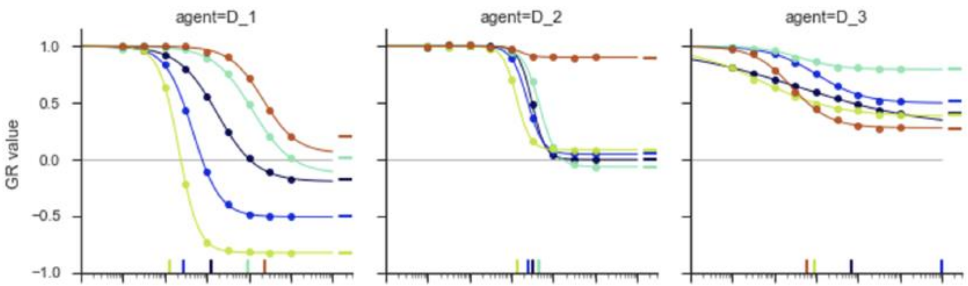
Fit a dose-response curve to obtain sensitivity metrics

Evaluate the GR metrics and plot the dose-response curves (protocol 3, step 2)

```
df_grmetrics = gr50.gr_metrics(df_gr)
df_grmetrics.to_csv('OUTPUT/GRmetrics_Example1.tsv', sep='\t', index=False)
print df_grmetrics.head()
fig = gr50.plot.plot_curves(df_grmetrics, df_gr, colorvar='cell_line', colvar='agent')
fig.savefig('OUTPUT/GRcurves_Example1.pdf')
```

Evaluate and plot GR metrics

```
                  date cell_line agent  treatment_duration      GR50     GRmax  \
0 2016-06-06 12:34:56       CL_1   D_1                72.0  0.126677 -0.173488
1 2016-06-06 12:34:56       CL_1   D_2                72.0  0.311289 -0.004219
2 2016-06-06 12:34:56       CL_1   D_3                72.0  0.690251  0.401416
3 2016-06-06 12:34:56       CL_1   D_4                72.0  3.703113 -0.096859
4 2016-06-06 12:34:56       CL_1   D_5                72.0  0.668006 -0.094478

     GR_AOC     GEC50     GRinf      h_GR        r2          pval
0  0.521884  0.180590 -0.195387  0.930270  0.999900  9.992007e-15
1  0.375490  0.311573 -0.001281  2.805116  0.999949  8.881784e-16
2  0.403562  0.039185  0.276279  0.280334  0.999646  8.330003e-13
3  0.117477  4.170068 -0.194446  2.766167  0.999968  2.220446e-16
4  0.312724  0.739848 -0.107022  1.898776  0.999925  3.552714e-15
```



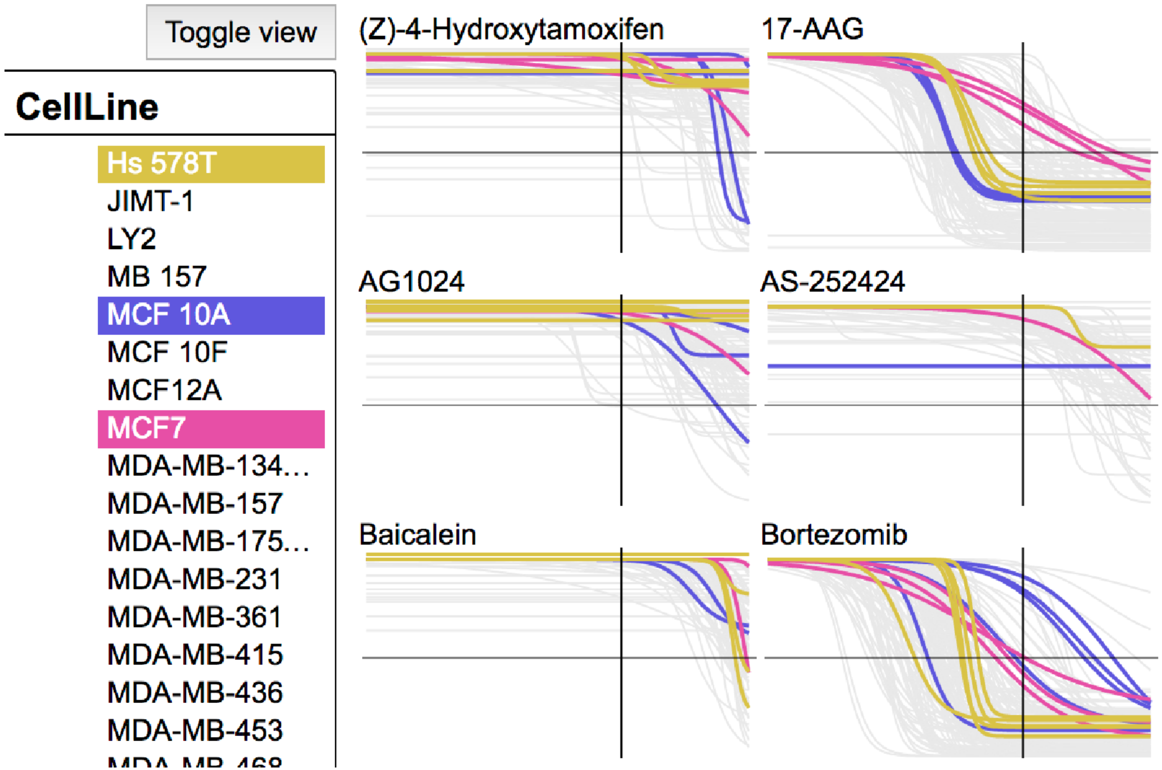
Dose-response plots

GRcalculator.org can replace the last part of the protocol

GRcalculator.org

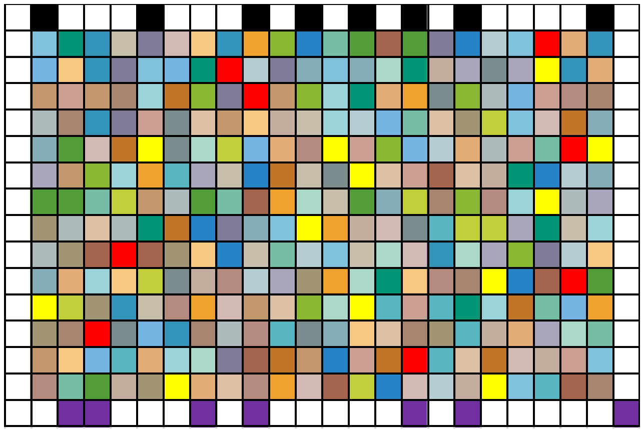
Clark*, Hafner* et al.,
BMC Cancer, in review

Hafner*, Heiser* et al.,
Sci Data, in review

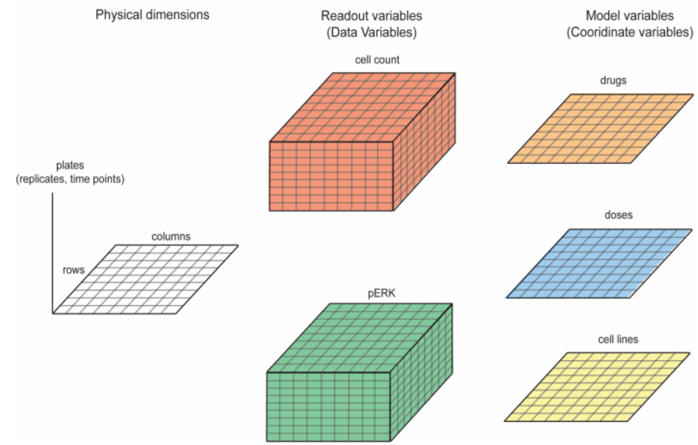


Advantages of an automated pipeline

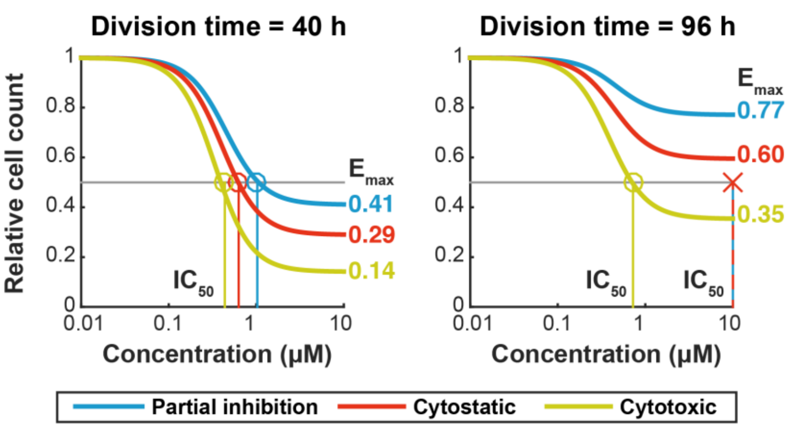
1. Complex plate layouts can be designed



2. A single data container for data and metadata
3. Extensions and modifications can be recorded



4. Integration with analysis tools



4. Jupyter notebooks enable ease of documentation end executions