Visualization and analysis of high-throughput *in vitro* dose–response datasets with Thunor

Alexander L. R. Lubbock[1], Leonard A. Harris[1], Vito Quaranta[1,2], Darren R. Tyson[1,4,*] and Carlos F. Lopez[1,2,3,*]

[1]Department of Biochemistry, Vanderbilt University School of Medicine, Nashville, TN, USA
[2]Department of Pharmacology, Vanderbilt University School of Medicine, Nashville, TN, USA.
[3]Department of Biomedical Informatics, Vanderbilt University School of Medicine, Nashville, TN, USA
[4]Lead Contact
[*]Correspondence: darren.tyson@vanderbilt.edu (D.R.T.), c.lopez@vanderbilt.edu (C.F.L.)

Revision Date: August 27th, 2018

Abstract

Quantifying the effects of drugs and other environmental factors on cell proliferation in vitro continues to be one of the most prevalent assays in biomedical research. Assessment of the dose-dependent nature of drug effects is typically performed with a variety of commercial software applications or using freely available, but more technically demanding, statistical programming environments such as Python or R. However, with the advent of large, publicly-available drug response databases and continued advancements in high-throughput experimentation, there is a growing need for user-friendly software platforms that can efficiently and reliably facilitate analysis within and across large datasets. Here we introduce Thunor, an open-source software platform for the management, analysis, and visualization of large-scale dose-dependent cell proliferation datasets. Thunor provides a simple, user-friendly interface to upload cell count data and a graphical plate map tool to annotate plate wells with cell lines and drugs. Best-fit dose–response curves are generated based on either cell viability or proliferation rate drug effect metrics. Derived dose–response parameters, such as IC_{50}, E_{max}, and activity area, are automatically calculated by the software back-end. An arrayed plot interface supports multiple plot types, including time course, dose–response curve, box/bar/scatter plots of derived parameters, and quality control analyses, among others. We demonstrate the features of Thunor on large-scale, publicly-available viability data and an in-house, high-throughput proliferation rate dataset. Software, documentation, and an online demo are all available at thunor.net.

Keywords— Pharmacology, Drug screening, Biomarker discovery, Cell proliferation, Oncology

Introduction

The effect of drugs and other perturbagens on cell proliferation *in vitro* is one of the most widely studied phenomena in biomedical research and crucial for biomarker discovery and drug development pipelines, particularly in oncology. The most common approach to quantify drug response is the viability end-point assay, which measures cell count—typically 72 hours post drug addition—as a fraction of control (unperturbed) cells. High-throughput experimentation parallelizes this effort across multiple drugs, concentrations, and cell lines in multi-well plates and has led to the establishment of multiple large-scale, publicly available datasets (Barretina et al., 2012; Garnett et al., 2012; Basu et al., 2013; Iorio et al., 2016; Rees et al., 2016). These datasets are a useful resource for the research community and have successfully established links between molecular features of cell lines and their responses to drugs (Cancer Cell Line Encyclopedia Consortium and Genomics of Drug Sensitivity in Cancer Consortium, 2015; Seashore-Ludlow et al., 2015; Shimada et al., 2016; Rees et al., 2016; Hafner et al., 2017). Recently, we and others introduced novel drug-effect metrics based on cellular proliferation rates, termed the drug-induced proliferation (DIP) rate (Harris et al., 2016) and growth rate (GR) inhibition (Hafner et al., 2016). Both metrics avoid biases inherent in traditional viability assays, which can produce misleading interpretations, e.g., of cell line sensitivity to drug. While estimating proliferation rates has higher resource demands than traditional end-point assays, the increasing prevalence of digital microscopy imaging systems capable of automatically acquiring serial images over time (e.g., Sartorius IncuCyte S3) has greatly expanded the affordability and availability of time course data. Thus, for both end-point and time-series datasets, the challenge has shifted from data generation to data management and interpretation. These tasks generally involve a cumbersome and potentially error-prone workflow of processing multiple instrument-exported file types in a spreadsheet and performing analyses with

various commercial software packages or custom code (see Text S1).

To address these challenges, we introduce Thunor (THOO-nor), an open-source software platform to store, share, analyze, and visualize large-scale drug response datasets. Thunor supports both end-point and time-series drug response data and includes a user-friendly web interface that facilitates data management, access control, data sharing, drag-and-drop upload, and point-and-click analysis and visualization (see Table S1 for feature comparisons to related free and commercial software packages). A paneled interface allows multiple plots to be loaded at once from multiple datasets, including time courses, dose–response curves, box/bar/scatter plots of derived dose–response parameters (e.g., IC_{50}, E_{max}, activity area), and plate map visualizations. Users can also define, upload, share, and categorize *tags*, which are labels applied to groups of cell lines or drugs and can be used for aggregating data in plots or for statistical tests. Thunor users can share datasets with other users on their Thunor installation or keep them private. Cell count data are uploaded and exported in several formats, including CSV, HDF5, and IncuCyte (import only). Converters are also provided for the publicly-available Genomics of Drug Sensitivity in Cancer (GDSC) (Garnett et al., 2012) and Cancer Therapeutics Response Portal (CTRP) (Rees et al., 2016) datasets. In accordance with open-science principles, Thunor is free and open source (General Public License version 3). Software, documentation, and a read-only demonstration are available at www.thunor.net.

Results

Design and security

Thunor is comprised of Thunor Core and Thunor Web (Figure S1). Thunor Core is written in Python and provides core functionality, including data structures, DIP rate calculation, curve fitting, and plotting. Thunor Web adds a web interface with pop-up help, a database that facilitates concurrent

access and efficient retrieval of data subsets, and multi-user capability. Users upload datasets by drag-and-drop that can be designated private (default), shared within a group, or public. If the uploaded data are not pre-annotated with plate maps, an efficient graphical tool is provided (see below). During upload, DIP rate, viability, and derived dose–response curve parameters are automatically calculated and stored. DIP rate calculation accounts for any drug-effect stabilization delays on cell proliferation (Harris et al., 2016). Caching and database indexing are used to improve performance; connections are encrypted and password protected; and *Docker Compose* is utilized to containerize software components for security, reproducibility, and to facilitate deployment to in-house servers or cloud platforms. A list of key software libraries used by Thunor is provided in Table S2.

Interactive plot system

The central feature of Thunor Web is a user-friendly, interactive, multi-paneled plot interface that supports multiple plot types for both single-time-point (viability; Figure 1) and time-course-based (Figure 2) data. Supported visualizations include cell population growth curves (Figure 2A), dose–response curves (Figures 1B and 2B), box/bar/scatter plots of derived dose–response parameters (Figures 1A,C,D and 2C,D), and quality controls (Figure 2E). Multiple datasets can be viewed at once in separate panels and dose–response curves and derived parameters from two datasets can be compared on a single plot (Figure 1D). Dose–response curve and derived parameter plots also support cell line/drug selection by tag (see below). Statistical tests are automatically performed and p-values displayed where appropriate: linear regression for scatter plots (Figure 1D), one-way ANOVA for box plots (Figure 1A), and Mann-Whitney U tests for bar plots with two groups. All plots are scalable, zoomable, and clickable to facilitate analysis and easy comparison across panels. Images (SVG, PNG) and raw data (CSV, JSON) can also be downloaded for offline use.

Tags for cell lines and drugs

Thunor Web provides an interface for creating, editing, viewing, and deleting "tags," labels that are used to group sets of cell lines or drugs in the plot system. Tags can be created manually or uploaded from file and can be private, shared within a group, or public. Typical tags include cell line mutations, cell line tissues of origin, drug molecular targets, and drug classifications. Thunor's tagging feature is rare among similar open-source and commercial software tools (Table S1) and significantly aids in the visualization and analysis of large-scale datasets. For example, we created cell line tags (Dataset S2) based on mutation (`BRAF-mut`, `EGFR-mut`) and primary site (`lung`, `skin`, `aerodig tract`) by processing the metadata files included with the GDSC dataset. We then generated box plots illustrating ranges of drug sensitivity (in terms of activity area, AA) for each tagged set of cell lines over nine small-molecule kinase inhibitors. Cell lines originating from skin (including melanoma) show increased sensitivity to the BRAF inhibitor PLX-4720 and the MEK inhibitor rafemetinib but relative insensitivity to the EGFR inhibitor afatinib (Figure 1A). Conversely, cell lines originating from the aerodigestive tract show little sensitivity to PLX-4720 and rafemetinib but increased sensitivity to afatinib. This result can be further explored in Thunor by plotting and comparing dose–response curves for all aerodigestive tract and skin cell lines for all three drugs (Figure 1B) and generating a color-coded bar plot of cell line drug sensitivity to afatinib (Figure 1C).

Data annotation

High-throughput experiments generally involve multiple plates with differing drugs, concentrations, and cell line layouts. Annotated plate maps are crucial for linking cell counts back to experimental conditions but are often missing from instrument-exported data. Thunor accepts cell count uploads (TSV format; see Table S3) with or without plate map layouts. In the latter case, a user-friendly

graphical interface (Figure S2) is provided that allows users to manually annotate plates, as well as visualize pre-annotated datasets and export plate map annotations to file (TSV and JSON formats). Users can select an individual well on a plate and enter the cell line, drug(s), and concentration(s) used (drug combinations are supported). Cell line and drug name suggestions are provided from the database; the user may also create new ones. The plate mapper also has numerous features that drastically speed up data entry, including an "auto-stepper" that moves the current well selection (e.g., one well to the right), an "auto-dilution" feature for changing concentrations, and a "template" system for annotating multiple plates simultaneously.

Discussion

High-throughput *in vitro* screens of large panels of chemical compounds against hundreds to thousands of cultured cell lines is a powerful and increasingly popular tool for probing complex intracellular networks and identifying druggable targets in hypothesis-driven biomedical research (Macarron et al., 2011). To address the challenges associated with the growing size and complexity of high-throughput cell proliferation data, we have introduced Thunor, a free and open-source computational tool to explore large dose–response datasets. We anticipate that Thunor will stimulate collaboration between researchers, ease the exchange of drug-response data, and aid in reproducibility and transparency across datasets. Thunor is an active project and we encourage input and contributions from the research community (via web chat and/or Github repository, both accessible from www.thunor.net). Extensions to Thunor will be prioritized by user feedback and may include drug combination response modeling, additional statistical analyses, improved quality control checks on data upload, and integration of omics datasets (e.g., RNA-seq) to explore molecular correlates of drug sensitivity.

Author contributions

Conceptualization, A.L.R.L., C.F.L., D.R.T., and L.A.H.; Methodology, A.L.R.L., D.R.T., and L.A.H.; Investigation, A.L.R.L., D.R.T., and L.A.H.; Resources, C.F.L., D.R.T., and V.Q.; Data Curation, A.L.R.L., D.R.T., L.A.H.; Writing – Original Draft, A.L.R.L., D.R.T., and L.A.H.; Writing – Review & Editing, A.L.R.L., C.F.L., D.R.T., and L.A.H.; Visualization, A.L.R.L., D.R.T., and L.A.H.; Supervision, C.F.L., D.R.T., and V.Q.; Funding Acquisition, C.F.L. and V.Q.

Acknowledgements

We gratefully acknowledge technical assistance from J. A. Bauer at the Vanderbilt High Throughput Screening Core; C. M. Lovly for access to IncuCyte instrument and data; and C. E. Hayford, C. Meyer, and D. Westover for useful discussions. Funding was provided by the National Science Foundation (1411482 to C.F.L.), National Cancer Institute (U01CA215845 to V.Q. and C.F.L.), and Defense Advanced Research Projects Agency (Cooperative Agreement no. W911 NF-14-2-0022 to C.F.L.).

Declaration of Interests

The authors declare no competing interests.

References

Barretina, J., Caponigro, G., Stransky, N., Venkatesan, K., Margolin, A. A., Kim, S., Wilson, C. J., Lehár, J., Kryukov, G. V., Sonkin, D., Reddy, A., Liu, M., Murray, L., Berger, M. F., Monahan, J. E., Morais, P., Meltzer, J., Korejwa, A., Jané-Valbuena, J., Mapa, F. A., Thibault, J., Bric-Furlong, E., Raman, P., Shipway, A., Engels, I. H., Cheng, J., Yu, G. K., Yu, J., Aspesi, P., de Silva, M., Jagtap, K., Jones, M. D., Wang, L., Hatton, C., Palescandolo, E., Gupta, S., Mahan, S., Sougnez, C., Onofrio, R. C., Liefeld, T., MacConaill, L., Winckler, W., Reich, M., Li, N., Mesirov, J. P., Gabriel, S. B., Getz, G., Ardlie, K., Chan, V., Myer, V. E., Weber, B. L., Porter, J., Warmuth, M., Finan, P., Harris, J. L., Meyerson, M., Golub, T. R., Morrissey, M. P., Sellers, W. R., Schlegel, R. and Garraway, L. A. (2012). The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity. Nature *483*, 603–607.

Basu, A., Bodycombe, N. E., Cheah, J. H., Price, E. V., Liu, K., Schaefer, G. I., Ebright, R. Y., Stewart, M. L., Ito, D., Wang, S., Bracha, A. L., Liefeld, T., Wawer, M., Gilbert, J. C., Wilson, A. J., Stransky, N., Kryukov, G. V., Dancik, V., Barretina, J., Garraway, L. A., Hon, C. S.-Y., Muñoz, B., Bittker, J. A., Stockwell, B. R., Khabele, D., Stern, A. M., Clemons, P. A., Shamji, A. F. and Schreiber, S. L. (2013). An interactive resource to identify cancer genetic and lineage dependencies targeted by small molecules. Cell *154*, 1151–1161.

Branch, M., Coleman, T. and Li, Y. (1999). A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. SIAM Journal on Scientific Computing *21*, 1–23.

Cancer Cell Line Encyclopedia Consortium and Genomics of Drug Sensitivity in Cancer Consortium (2015). Pharmacogenomic agreement between two cancer cell line data sets. Nature *528*, 84–87.

Fallahi-Sichani, M., Honarnejad, S., Heiser, L. M., Gray, J. W. and Sorger, P. K. (2013). Metrics other than potency reveal systematic variation in responses to cancer drugs. Nature Chemical Biology *9*, 708–714.

Garnett, M. J., Edelman, E. J., Heidorn, S. J., Greenman, C. D., Dastur, A., Lau, K. W., Greninger, P., Thompson, I. R., Luo, X., Soares, J., Liu, Q., Iorio, F., Surdez, D., Chen, L., Milano, R. J., Bignell, G. R., Tam, A. T., Davies, H., Stevenson, J. A., Barthorpe, S., Lutz, S. R., Kogera, F., Lawrence, K., McLaren-Douglas, A., Mitropoulos, X., Mironenko, T., Thi, H., Richardson, L., Zhou, W., Jewitt, F., Zhang, T., O'Brien, P., Boisvert, J. L., Price, S., Hur, W., Yang, W., Deng, X., Butler, A., Choi, H. G., Chang, J. W., Baselga, J., Stamenkovic, I., Engelman, J. A., Sharma, S. V., Delattre, O., Saez-Rodriguez, J., Gray, N. S., Settleman, J., Futreal, P. A., Haber, D. A., Stratton, M. R., Ramaswamy, S., McDermott, U. and Benes, C. H. (2012). Systematic identification of genomic markers of drug sensitivity in cancer cells. Nature *483*, 570–575.

Hafner, M., Niepel, M., Chung, M. and Sorger, P. K. (2016). Growth rate inhibition metrics correct for confounders in measuring sensitivity to cancer drugs. Nature Methods *13*, 521–527.

Hafner, M., Niepel, M. and Sorger, P. K. (2017). Alternative drug sensitivity metrics improve preclinical cancer pharmacogenomics. Nature Biotechnology *35*, 500–502.

Haibe-Kains, B., El-Hachem, N., Birkbak, N. J., Jin, A. C., Beck, A. H., Aerts, H. J. W. L. and Quackenbush, J. (2013). Inconsistency in large pharmacogenomic studies. Nature *504*, 389–393.

Harris, L. A., Frick, P. L., Garbett, S. P., Hardeman, K. N., Paudel, B. B., Lopez, C. F., Quaranta, V. and Tyson, D. R. (2016). An unbiased metric of antiproliferative drug effect *in vitro*. Nature Methods *13*, 497–500.

Iorio, F., Knijnenburg, T. A., Vis, D. J., Bignell, G. R., Menden, M. P., Schubert, M., Aben, N., Gonçalves, E., Barthorpe, S., Lightfoot, H., Cokelaer, T., Greninger, P., van Dyk, E., Chang, H., de Silva, H., Heyn, H., Deng, X., Egan, R. K., Liu, Q., Mironenko, T., Mitropoulos, X., Richardson, L., Wang, J., Zhang, T., Moran, S., Sayols, S., Soleimani, M., Tamborero, D., Lopez-Bigas, N., Ross-Macdonald, P., Esteller, M., Gray, N. S., Haber, D. A., Stratton, M. R., Benes, C. H., Wessels, L. F. A., Saez-Rodriguez, J., McDermott, U. and Garnett, M. J. (2016). A landscape of pharmacogenomic interactions in cancer. Cell *166*, 740–754.

Macarron, R., Banks, M. N., Bojanic, D., Burns, D. J., Cirovic, D. A., Garyantes, T., Green, D. V. S., Hertzberg, R. P., Janzen, W. P., Paslay, J. W., Schopfer, U. and Sittampalam, G. S. (2011). Impact of high-throughput screening in biomedical research. Nature Reviews Drug Discovery *10*, 188–195.

McKinney, W. (2010). Data Structures for Statistical Computing in Python. In Proceedings of the 9th Python in science conference (SCIPY 2010) pp. 51–56,.

Merkel, D. (2014). Docker: lightweight Linux containers for consistent development and deployment. Linux J. *2014*, 2.

Plotly Technologies Inc. (2015). Collaborative data science. https://plot.ly.

Rees, M. G., Seashore-Ludlow, B., Cheah, J. H., Adams, D. J., Price, E. V., Gill, S., Javaid, S., Coletti, M. E., Jones, V. L., Bodycombe, N. E., Soule, C. K., Alexander, B., Li, A., Montgomery, P., Kotz, J. D., Hon, C. S.-Y., Muñoz, B., Liefeld, T., Dancik, V., Haber, D. A., Clish, C. B., Bittker, J. A., Palmer, M., Wagner, B. K., Clemons, P. A., Shamji, A. F. and Schreiber, S. L. (2016). Correlating chemical sensitivity and basal gene expression reveals mechanism of action. Nature Chemical Biology *12*, 109–116.

Ritz, C., Baty, F., Streibig, J. C. and Gerhard, D. (2015). Dose-response analysis using R. PLOS One *10*, e0146021.

Seashore-Ludlow, B., Rees, M. G., Cheah, J. H., Cokol, M., Price, E. V., Coletti, M. E., Jones, V., Bodycombe, N. E., Soule, C. K., Gould, J., Alexander, B., Li, A., Montgomery, P., Wawer,

M. J., Kuru, N., Kotz, J. D., Hon, C. S.-Y., Muñoz, B., Liefeld, T., Dancik, V., Bittker, J. A., Palmer, M., Bradner, J. E., Shamji, A. F., Clemons, P. A. and Schreiber, S. L. (2015). Harnessing connectivity in a large-scale small-molecule sensitivity dataset. Cancer Discovery *5*, 1210–1223.

Shimada, K., Hayano, M., Pagano, N. C. and Stockwell, B. R. (2016). Cell-line selectivity improves the predictive power of pharmacogenomic analyses and helps identify NADPH as biomarker for ferroptosis sensitivity. Cell Chemical Biology *23*, 225–235.

van der Walt, S., Colbert, S. C. and Varoquaux, G. (2011). The NumPy array: a structure for efficient numerical computation. Computing in Science & Engineering *13*, 22–30.

Figures

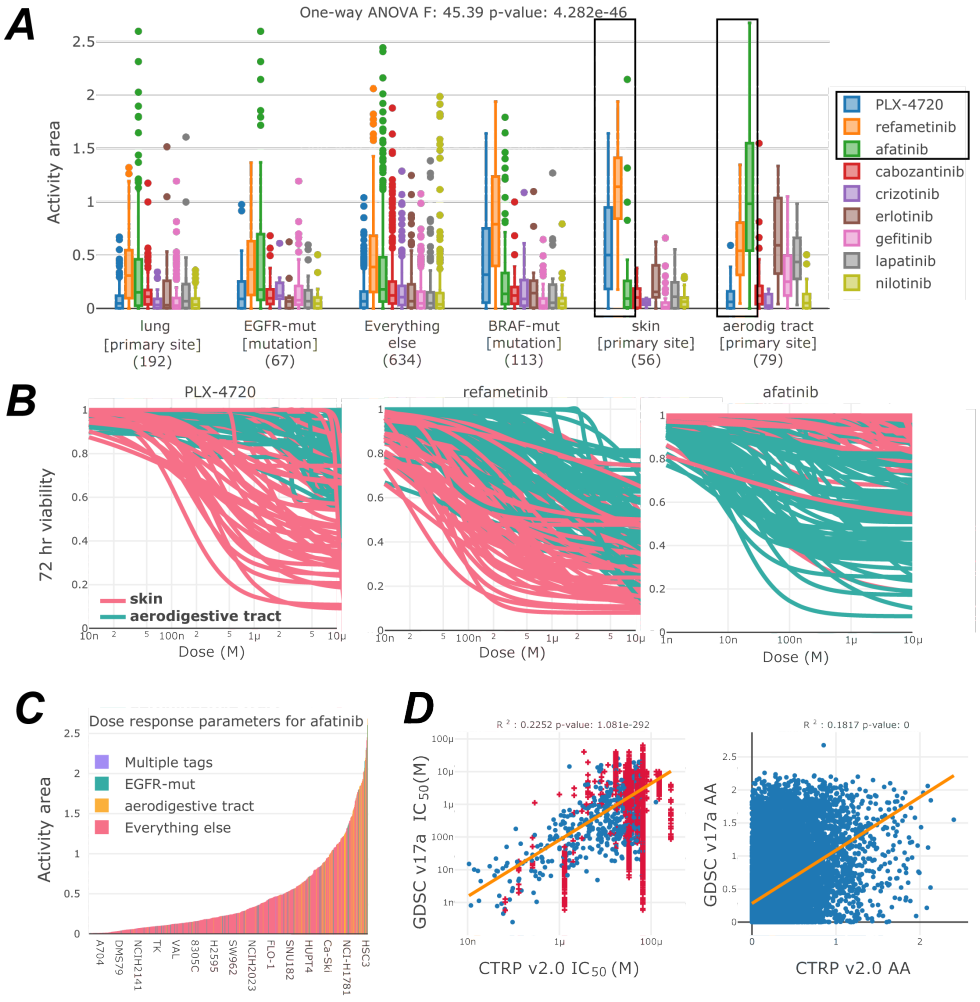

Figure 1: **Thunor-generated plots for GDSC and CTRP viability data.** ***A.*** Custom tags were defined for primary cancer sites and gene mutations and applied to GDSC viability data to identify differential drug sensitivities, quantified in terms of activity area (AA), over a panel of small molecule kinase inhibitors. Number of cell lines within each tag label is shown in parentheses. Boxes denote three drugs with notable differential sensitivities across tags. ***B.*** Dose–response curves for cell lines tagged as "skin" and "aerodigestive tract" for the three drugs with differential sensitivity in (*A*). ***C.*** Bar plot of cell line sensitivity (AA) to afatinib. Colors denote cell line tags. To facilitate visualization, only a subset of cell line labels are shown. ***D.*** Scatter plots for IC_{50} (*left*) and AA (*right*) of cell line/drug pairs common between the GDSC and CTRP datasets. Consistent with previous work (Haibe-Kains et al., 2013), we see general agreement between the datasets for IC_{50} but large disagreement for AA. Red crosses in the left panel indicate truncated values set to the highest concentration tested.

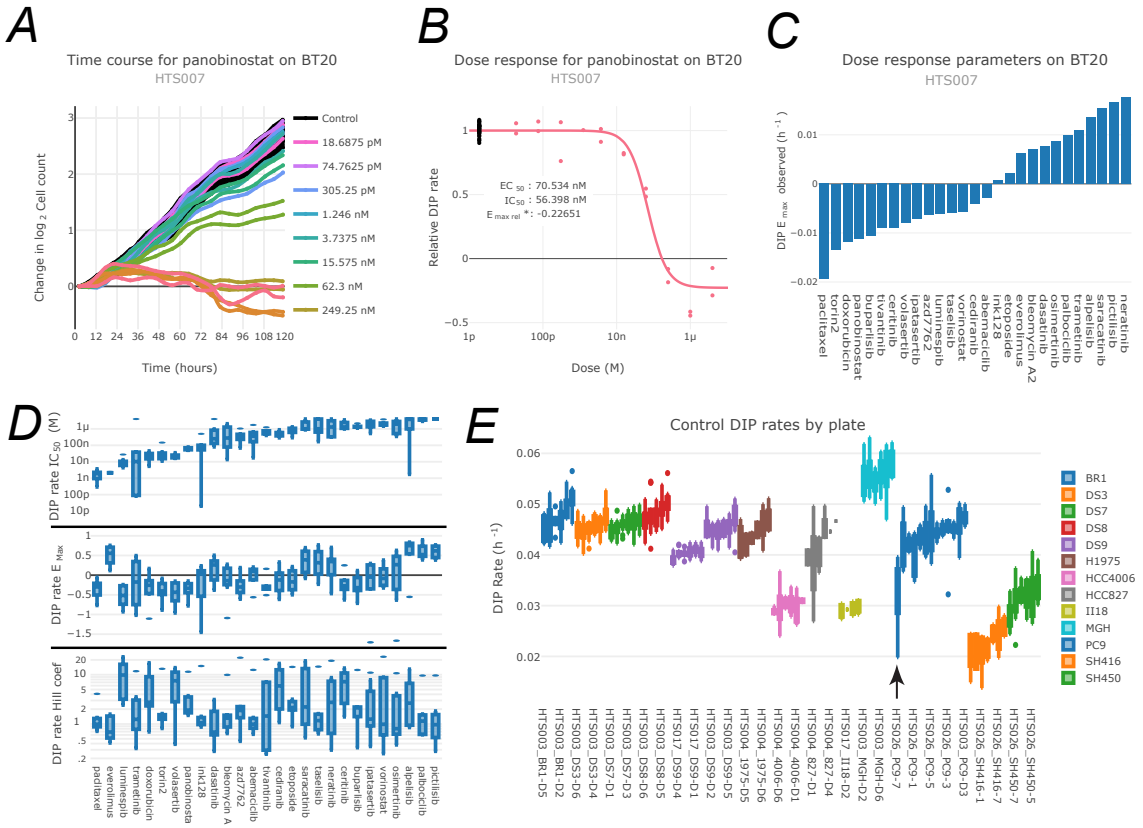

Figure 2: **Thunor-generated plots for high-throughput time-course data.** ***A.*** Cell counts over time (plotted as population doublings) for BT20 cells (triple-negative breast cancer) treated with varying concentrations of panobinostat (an inhibitor of histone deacetylases). ***B.*** Drug-induced proliferation (DIP) rate-based dose–response curve for panobinostat-treated BT20 cells showing the underlying data points (pink circles) and controls (black circles). ***C.*** Bar plot of DIP rate-based E_{max} values for BT20 cells treated with a variety of drugs. ***D.*** Boxplots of extracted dose–response curve parameters (*top row*: IC_{50}, *middle row*: E_{max}, *bottom row*: Hill coefficient) for all cell lines over the full panel of drugs considered. All plots are ordered based on median IC_{50} and demonstrate little correlation between the parameters, similar to previous reports (Fallahi-Sichani et al., 2013). ***E.*** Box plots of control proliferation rates per cell line per plate for four HTS datasets with EGFR-mutant lung adenocarcinoma cell lines. Arrow denotes a discrepant value where the control proliferation rates on a single plate are inconsistent with control rates of the same cell line on other plates. All data used to generate plots in (A)–(D) are provided in Dataset S1.

STAR Methods

Key Resources Table

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Experimental Models: Cell Lines		
BT20	Sorger lab, Harvard Medical School	HTB-19
HCC1143	Sears lab, OHSU	CRL-2321
MCF10A (VU)	ATCC	CRL-10317
MCF10A (HMS)	Sorger lab, Harvard Medical School	CRL-10317
MDAMB231	ATCC	HTB-26
MDAMB453	ATCC	HTB-131
MDADM468	ATCC	HTB-132
Software and Algorithms		
Thunor	This paper	thunor.net
DIP rate algorithm	Harris et al., 2016	DOI: 10.1038/nmeth.3852
Other Software	See Table S2	

Contact for Reagent and Resource Sharing

Further information and requests for reagents or computational resources may be directed to Darren Tyson (darren.tyson@vanderbilt.edu).

Method Details

DIP rate calculation

The DIP rate is defined as the gradient of the log2 cell count over time, after any initial stabilization period. The stabilization period is determined by iteratively fitting all time points, all time points excluding the first, then excluding the first and second, etc. For each time point set, the root mean squared error ($RMSE$) and adjusted R squared ($ARSQ$) is calculated. Each set of time points is fitted using linear regression. The final time point set is then selected,

$$\underset{m \in 1..(N-1)}{\arg\max}\ m \qquad \mathrm{ARSQ}(fit) * (1 - \mathrm{RMSE}(fit))^2 * (n-3)^{0.25} \tag{1}$$

where N is the number of time points available for the well, $n = N - m + 1$ is the number of time points used for the DIP fit on the current iteration, m is the index of first time point used for the fit, and fit is a linear regression fit to the data points $m..N$. A minimum of two time points is required

for a DIP rate fit, but five or more time points is strongly recommended. When exactly two time points are present, the iterative procedure is skipped and the linear regression fit is used.

Viability calculation

For viability calculations, the 72 hour time point is used, unless the dataset contains only a single time point, in which case that time point is used. The closest available time point is used for each well, if an exact time point match is not available. In Thunor Web, the time point used can be verified by hovering the cursor over a viability data point in an individual viability dose response curve. Viability is calculated as the ratio of a well's cell count to the mean of the matched control wells' cell counts at the same time point. Control wells are defined as wells on the same plate, using the same cell line, but with no drugs added to the well (or with drug at concentration of zero).

Dose response curves

Dose response curves are fitted using a log-logistic function with three (viability) or four (DIP rate) parameters. Equation 2 defines the four parameter case, where X is a vector of concentration values and b (Hill slope), c (E_{max}), d (E_0), and e (EC_{50}) are fit parameters. In the three parameter case, d is set to 1, because viability is relative to control, and at the limit of zero concentration is equal to the control viability.

$$f(X, b, c, d, e) = c + \frac{d - c}{1 + e^{b(\ln X - \ln e)}} \tag{2}$$

The curve fitting uses the `curve_fit` function in SciPy (scipy.org). Initial values for the fit parameters are estimated from the data using the same approach as the four-parameter log-logistic (`LL.4`) function in the `drc` R package (Ritz et al., 2015).

In the DIP rate case, the `curve_fit` function selects a least squares fit using the Levenberg-Marquadt algorithm. The fit residuals r are defined in Equation 3, where Y is a vector of response values (here, DIP rate), X is a vector of drug concentrations, P is the set of fit parameters, and f is the log-logistic fit function, as defined previously. The standard error of DIP rate data point is incorporated into the fit by minimizing Equation 4, where σ is the standard error of a response value. Both control and experiment DIP rate values are used in the curve fit. Since the fit takes place in log(concentration) space, a non-zero dose must be assigned to controls. We set the concentration of controls to ten-fold less than the lowest concentration in X. The curve fit is replaced with a "no effect" model (shown as horizontal dashed line in plots) if the dose response curve is not significantly different from that no effect model (F-test, $p<0.05$). The fit is rejected (no dose response curve shown) if any of the following occur: a numerical error occurs in the `curve_fit` function, the fit EC_{50} is less than the minimum concentration observed, or the fit E_0 is greater than the mean plus one standard deviation of the control data points' DIP rate values (where at least five control data points are used in the fit) or greater than 1.2 times the mean of the control data points' DIP rates

(otherwise).

$$R = Y - f(X, P) \tag{3}$$

$$\chi^2 = \sum_{r \in R} \left(\frac{r}{\sigma} \right)^2 \tag{4}$$

In the viability case, the sum of squared residuals R is minimized directly. Dose–response curves are fitted with parameter constraints by the Trust Region Reflective algorithm (Branch et al., 1999). The parameter constraints are that Hill slope b must must positive, and E_{max} must be between 0 and 1, since viability cannot be negative.

This section describes the default methodology, which we have used here. Parameter choices, fit constraints and fit rejection tests are user-configurable in Thunor Core.

Thunor Core

Thunor Core is a Python package, which can be used interactively (Python prompt or within Jupyter Notebooks) or integrated into other data processing pipelines. Documentation is available at core.thunor.net. Thunor Core includes a suite of automated tests to help ensure code quality. Thunor Core is organized into several sub-packages: `thunor.io`, the core data types and file input/output; `thunor.dip`, which provides DIP rate calculations and derivative statistic calculations; `thunor.viability`, which calculates end point/single time point viability; `thunor.curve_fit`, for dose response curve fitting; and `thunor.plots`, for plotting; `thunor.helpers`, for miscellaneous "helper" functions. The core data structures are based around the Pandas DataFrame for high performance (McKinney, 2010). Numerical calculations and curve fitting use NumPy (numpy.org) and SciPy (van der Walt et al., 2011). Interactive plots are built using Plotly (Plotly Technologies Inc., 2015).

Thunor Web

Thunor Web is a web application built on Thunor Core and Django (djangoproject.com), a Python framework for web applications. Data are stored in a PostgreSQL database (postgresql.org) and accessed through the Django object-relational mapper. Redis (redis.io) is used as a cache to improve performance (optional). Thunor Web includes a suite of automated tests to help ensure code quality. Error monitoring, aggregation, and alerts are integrated with Sentry (getsentry.com). Sentry is available as open source software or a hosted service. Thunor Web incorporates user authentication. Account creation can be by open sign-up or by invitation. "Public" datasets can be available to view without authentication, or require a login. These preferences are set by the server administrator.

Containerized Infrastructure

Thunor Web provides a complete configuration set up using Docker (Merkel, 2014) and Docker Compose. After Docker is installed and Thunor Web has been configured, the user can build and start the application, web, and database servers with a single command: `docker compose up -d -build`. A "quick start" configuration is provided. Configuration is entirely managed by setting environment variables, as recommended by the influential software engineering guidelines "The Twelve-Factor App" (12factor.net/config).

The use of Docker containers enhances security (containers are isolated and firewalled), transparency (build specifications are in human- and machine-readable plain text), and reproducibility (specifications are standardized and work the same across different host systems). Docker Compose allows services to scale to incorporate larger installations behind a load balancer and is cross-platform. Thunor Web can be deployed to the local computer (on which the installation is run) or remote servers (including cloud providers) using Docker Machine (docs.docker.com/machine/overview). Installation of Thunor Web using Docker Compose and Docker Machine are described in the documentation (docs.thunor.net).

Static Asset Build System

Static assets are files that do not change depending on who requests them (e.g., JavaScript, cascading stylesheets (CSS), fonts). Complex web applications often require multiple such files for each page load, which can be slow. In addition, assets can get cached by the user's browser, making timely updates more difficult (the cache becomes "stale"). This can lead to errors if, for example, the user loads an updated version of a web page and an older JavaScript file is retrieved from the cache—the two files are not designed to work together. Thunor Web has a static file build tool, which uses Webpack (webpack.js.org) in a Docker container. Webpack combines multiple files into a single bundle, which is "minified" (the file size is reduced by removing whitespace and unused code) and compressed. Each bundle has a unique hash code attached to its filename, which avoids the stale cache issue described above (so-called cache busting).

Automated Setup of TLS Encryption

High throughput data are often proprietary or sensitive. Without encryption, all data transmitted between the web server and the user, including passwords, could potentially be monitored and intercepted. Even with open datasets, the protection of login details and user privacy remains important. These issues can be mitigated by use of transport layer security (TLS), which provides end-to-end encryption between the user's web browser and the server. However, TLS can be challenging to obtain and configure. Most certificate authorities do not issue certificates for free, and the alternative, self-signed certificates, are subject to forgery and generate warnings in the user's web browser.

Thunor Web provides a Docker-based wrapper around the Certbot tool from the Electronic Frontier Foundation certbot.eff.org. Certbot generates TLS encryption certificates for free. The Thunor Web Certbot wrapper generates certificates for the user's domain or subdomain where they host Thunor Web. The wrapper also uses OpenSSL to generate Diffie-Hellman parameters, which are used to create ephemeral keys for "forward security." We also provide an example configuration file for the nginx server used by Thunor Web. The configuration has an A+ rating on Qualys SSL Labs' test (ssllabs.com/ssltest) at the time of writing.

Quantification and Statistical Analysis

Within Thunor Core and Thunor Web's plot interfaces, several statistical tests are available. All statistical tests are calculated using SciPy. On scatter plots, a line of best fit is calculated using linear regression and shown as an orange line. The R^2 and p-value for the fit are shown above the plot. On box plots, one-way ANOVA is used to test for a statistically significant difference in means of a dose response parameter across a set of user-defined groups (using Thunor Web's tag system). Depending on the groups used for the box plots, an ANOVA test may not be statistically appropriate—it is down to the user to consider whether it is so. In particular, one-way ANOVA assumes that (1) response variable residuals are normally distributed; (2) population variances are equal; (3) responses for each group are independent and identically distributed (i.i.d.) and normally distributed random variables. On bar plots, if the criteria described here are met, a two-sided Mann-Whitney U test is calculated, which tests for statistically significant difference in the ranks of two random variables. Here, Thunor will test for the difference in the ranks of a dose response parameter (e.g. IC_{50}) between two groups (cell lines, drugs, or tags). The test will only be run if the plot contains exactly two groups (defined by the number of cell lines/drugs/tags and the "color by" option, which defines the grouping variable), and both groups contain more than 20 data points. The test has continuity correction applied and assumes that data points are independent.

Data and Software Availability

Thunor Core and Thunor Web

Thunor Core and Thunor Web are licensed under the General Public License (GPL) version 3. Thunor Core is available on the Python Package Index (PyPI) and can be installed using `pip`: `pip install thunor`. Documentation and source code for both Core and Web are available at thunor.net.

HTS007 Dataset

The attached HTS007 dataset contains a panel of eight breast cancer cell lines treated with 27 drugs at multiple concentrations (4-fold dilutions). Each cell line was modified to express fluorescent

histone 2B (H2BmRFP) to enable detection of nuclei via fluorescence microscopy. Cells were imaged by automated fluorescence microscopy approximately every four hours over five days in the Vanderbilt University High Throughput Screening Core. Nuclei were quantified by automatic image segmentation.

Additional Resources

- Website: thunor.net
- Online Demo: demo.thunor.net
- Thunor Web Manual: docs.thunor.net
- Thunor Core Documentation: core.thunor.net
- HTS007 Dataset: `hts007.zip`