"Deep Learning in Cancer: Example for BDEC"

Rick Stevens
Argonne National Laboratory
The University of Chicago

Crescat scientia; vita excolatur

20 Oncologic Drugs Approved Between 2009 and 2013

Treatment cost $100K per year

Average PFS Improvement < 6 months

Table. Last 20 Oncologic Drugs Approved Between 2009 and 2013 by the US Food and Drug Administration

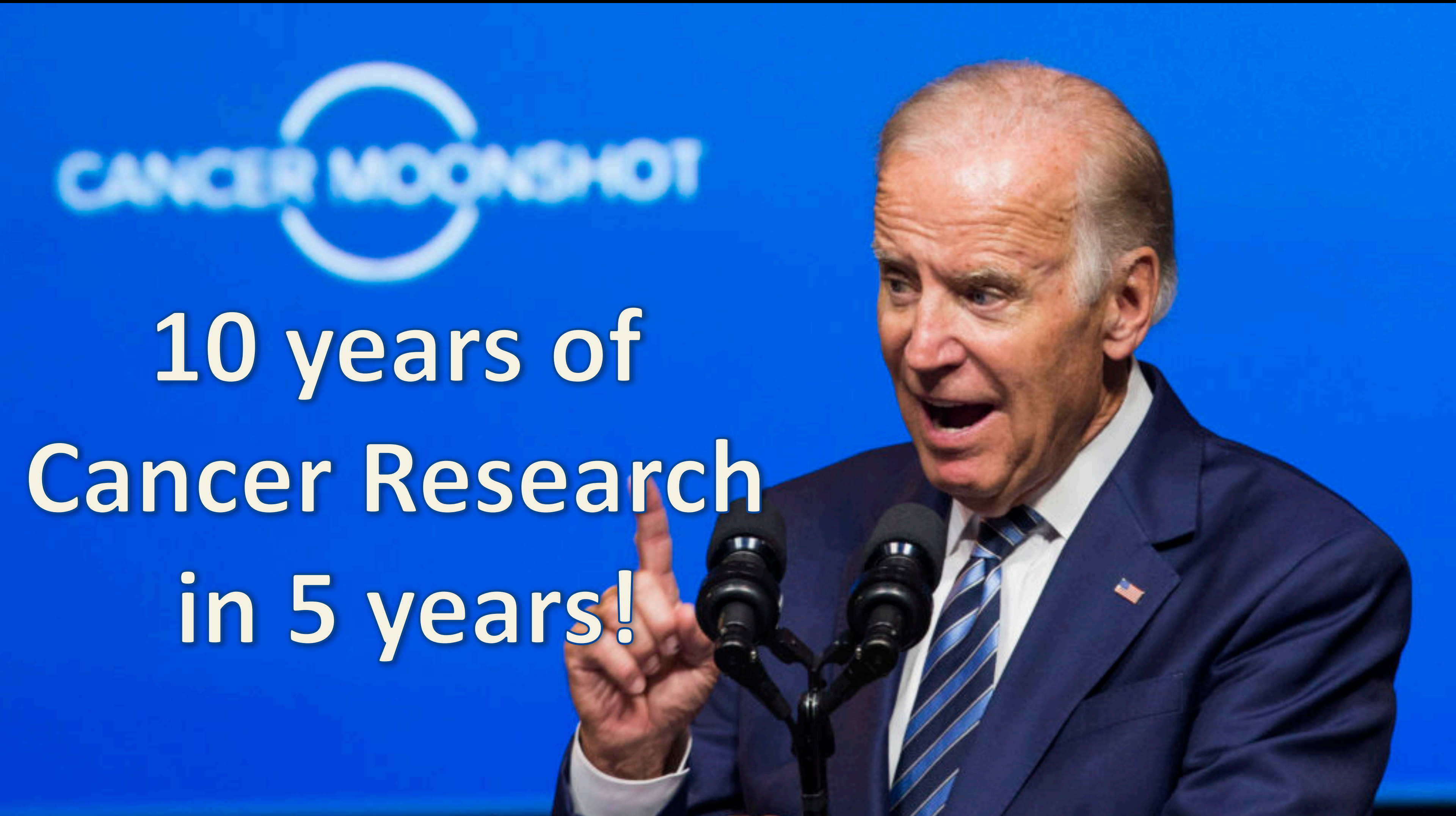
Drug and Indication	Cost per Year of Treatment, $[a]	Parent Drug	Mechanism of Action	Clinical Benefit
Sorafenib for papillary thyroid cancer	140 984	NA	First approved VEGFR and RAS tyrosine kinase inhibitor	Median PFS, 10.8 vs 5.8 mo
Criz[illegible]inib [illegible]	[illegible]	[illegible]	[illegible] kinase [illegible]	[illegible]
Ibru[illegible]	[illegible]	NA	[illegible] inhib[illegible]	[illegible]
Obinutuzumab for chronic lymphocytic leukemia	74 3[illegible]	Rituximab	Anti-CD2[illegible]noclonal antibody	Media[illegible] PFS, 23.0 vs 11.1 mo
Pertuzumab for breast cancer	78 252	Trastuzumab	Anti-her2 monoclonal antibody	Pathologic CR, 39.3% vs 21.5%
[illegible]	[illegible]	Pac[illegible]	[illegible] (micr[illegible])	Media[illegible] 6.7 mo
Afatinib for non-small-cell lung cancer	79 920	Erlotinib	EGFR tyrosine kinase inhibitor	Median PFS, 11.1 vs 6.9 mo; median OS, NS
Lenalidomide for mantle-cell lymphoma	124 870	Thalidomide	Immunomodulatory drug (thalidomide analogue)	RR, 26%; median DOR, 16.6 mo
Trametinib for malignant melanoma	125 280	NA	First approved mek inhibitor	Median PFS, 4.8 vs 1.5 mo
Dabrefenib for malignant melanoma	109 440	Vemurafenib	BRAF inhibitor	Median PFS, 5.1 vs 2.7 mo; median OS, NS
Radium 223 for prostate cancer	82 800	NA	First approved radiotherapeutic drug	Median OS, 14.0 vs 11.2 mo
Erlotinib for non-small-cell lung cancer	82 827	NA	[illegible] inhibit[illegible]	Median PFS, 10.4 vs 5.2 mo; m[illegible]
[illegible] for breast cancer	[illegible]13 161	[illegible]	[illegible] drug conjugate	M[illegible] 9.6 vs [illegible].4 mo; m[illegible] OS, 25.1 vs 20.9 mo
Pomalidomide for multiple myeloma	150 408	Thalidomide	Immunomodulatory drug (thalidomide analogue)	RR, 29%; median DOR, 7.4 mo
Bevacizumab for colorectal cancer	59 422	NA	First anti-VEGF monoclonal antibody	Median PFS, 5.7 vs 4 mo; median OS, 11.2 vs 9.8 mo
Ponatinib for chronic myeloid leukemia and Ph+ acute lymphoblastic leukemia	137 952	Imatinib	Bcr-abl tyrosine kinase inhibitor	Major cytogenetic response, 54%; median DOR, 3.2-9.5 mo
Abiraterone for prostate cancer	92 092	Ketoconazole	Androgen biosynthesis inhibitor	Median OS, 35.3 vs 30.1 mo
Cabozantinib for medullary thyroid cancer	118 800	NA	First multi[illegible] VEGF [illegible]	Median PFS, 11.2 vs 4 mo; median OS, NS
Omacetaxine for chronic myeloid leuke[illegible]	[illegible]	[illegible]	[illegible] inhibitor	Major cytogenetic response, 14.3%; median DOR, 12.5 mo
Nab-paclitaxel[b] for non-small-cell lung cancer	82 231	Paclitaxel	Albumin-bound paclitaxel (microtubule inhibitor)	RR, 33% vs 25%; median OS, NS
Regorafenib for colorectal cancer	141 372	Sorafenib	Multikinase inhib[illegible]	Median PFS, 2 vs 1.7 mo; m[illegible]

[illegible] applicable; NS, n[illegible] significant; OS, overall survival; PFS, progression-free survival; Ph+, Philadelphia chromosome positive; RR, response rate; UA, unavailable; (V)EGF(R), (vascular) endothelial cell growth factor (receptor).

[a] [illegible] were [illegible] ([subscription required] http://www.redbook.com/redbook/online/).

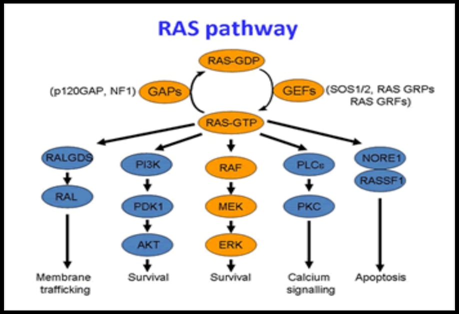
[b] This drug was approved separately for 2 indications.


CANCER MOONSHOT

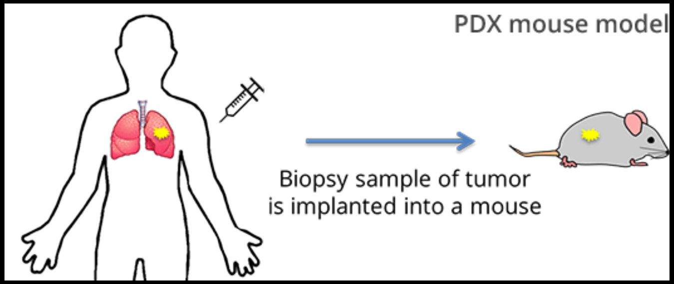

CANCER MOONSHOT
10 years of
Cancer Research
in 5 years!

The NCI-DOE partnership will extend the frontiers of precision oncology (Three Projects)

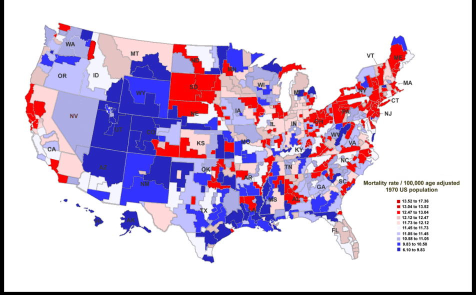
- **Cancer Biology**
  - **Molecular Scale Modeling of RAS Pathways**
  - Unsupervised Learning and Mechanistic models
  - Mechanism understanding and Drug Targets

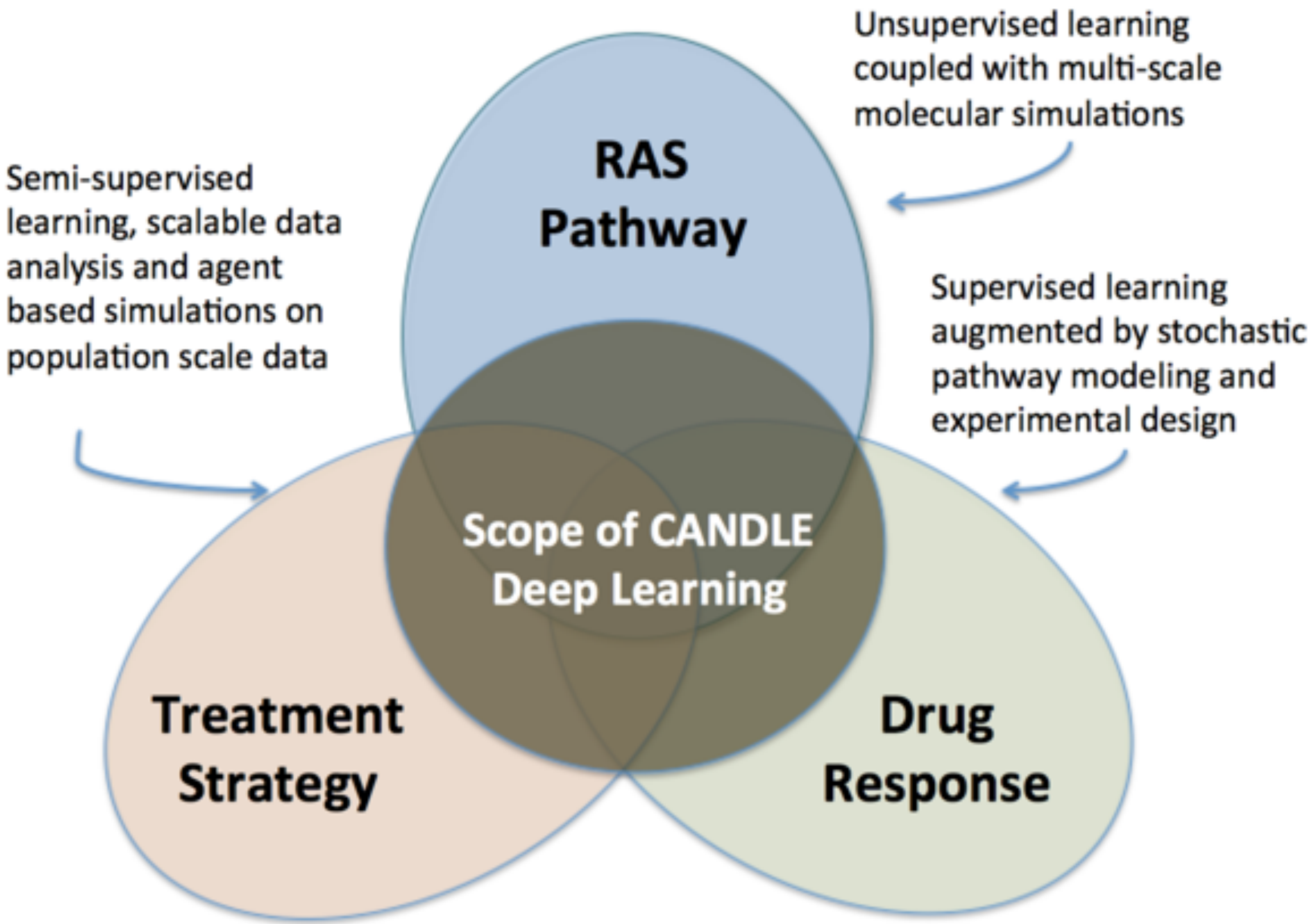


- **Pre-clinical Models**
  - **Cellular Scale PDX and Cell Lines**
  - ML, Experimental Design, Hybrid Models
  - Prediction of Drug Response

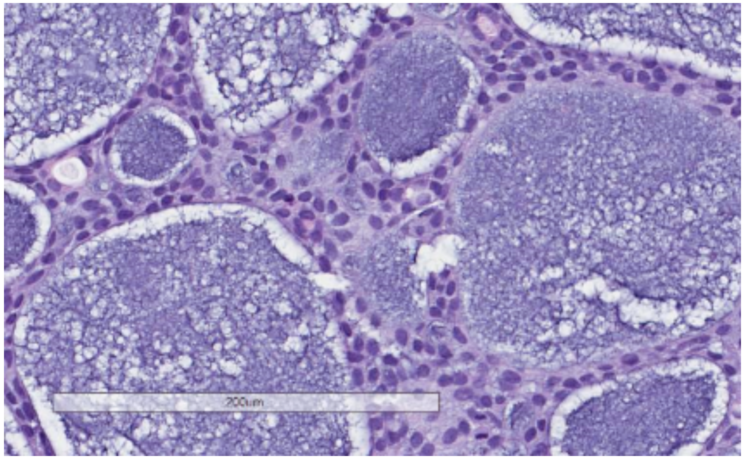


- **Cancer Surveillance**
  - **Population Scale Analysis**
  - Natural Langue and Machine Learning
  - Agent Based Modeling of Cancer PateintTrajectories



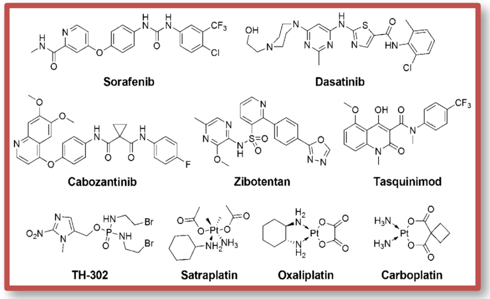

Unsupervised learning coupled with multi-scale molecular simulations
Semi-supervised learning, scalable data analysis and agent based simulations on population scale data
RAS Pathway
Supervised learning augmented by stochastic pathway modeling and experimental design
Scope of CANDLE Deep Learning
Treatment Strategy
Drug Response

Predictive Modeling of Drug Response

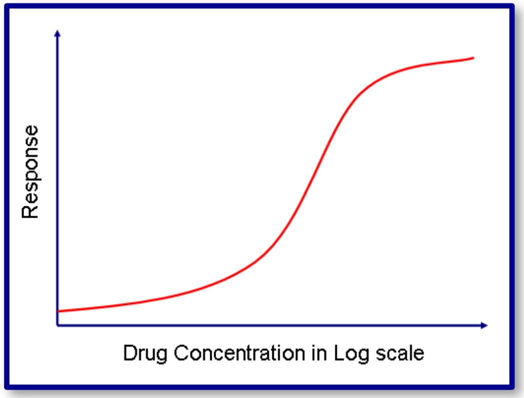


Drug (s)

- descriptors
- Fingerprints
- structures
- dose



$$\mathcal{R} = f(\mathcal{T}, \mathcal{D})$$

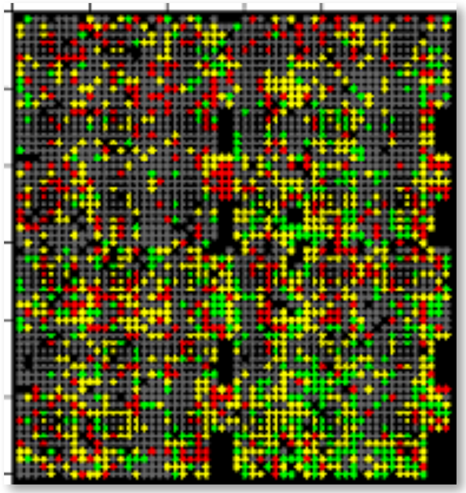


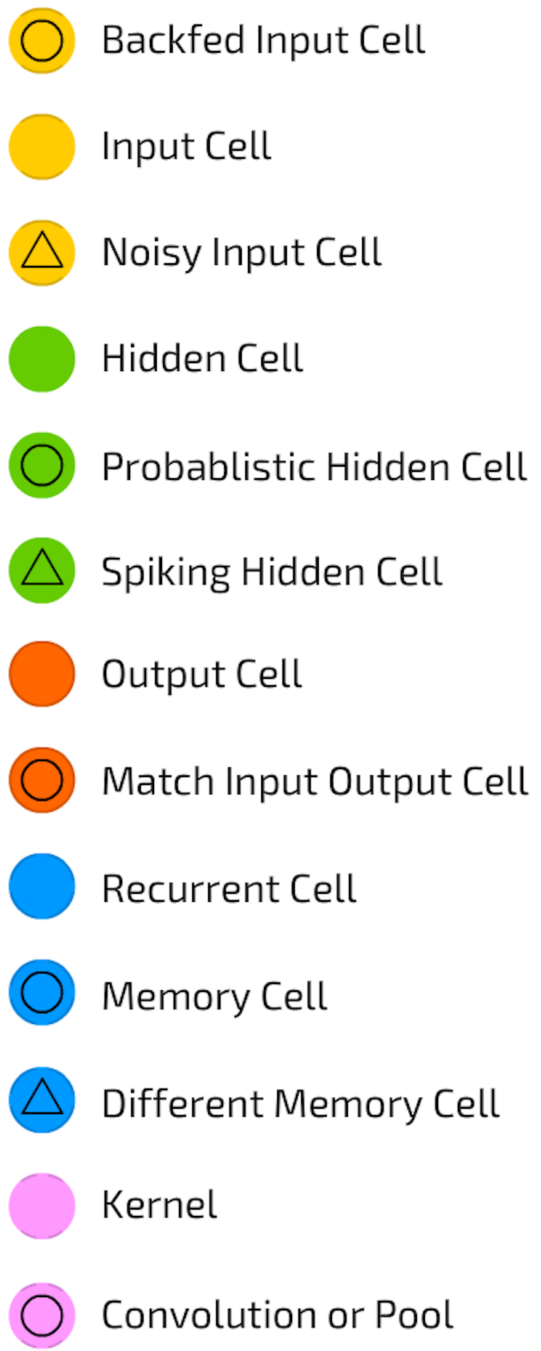
- IC50
- GI50
- % growth
- Z-score

Response

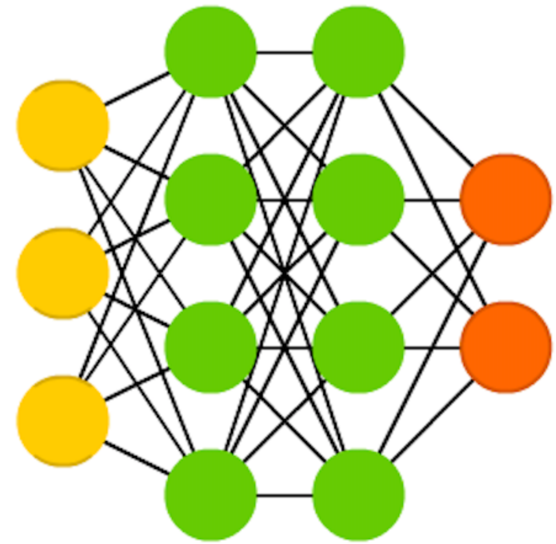
- gene expression levels
- SNPs
- protein abundance
- microRNA
- Methylation

Tumor

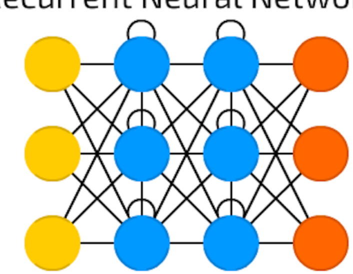




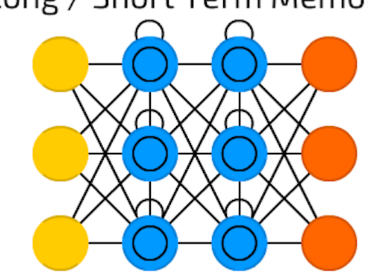
Deep Feed Forward (DFF)



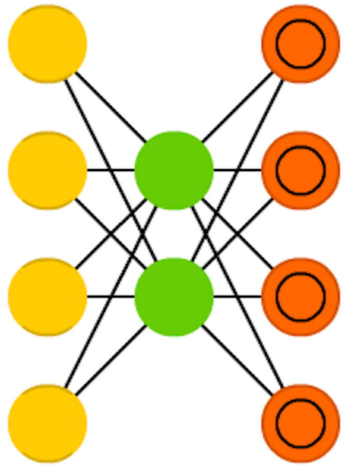
Recurrent Neural Network (RNN)



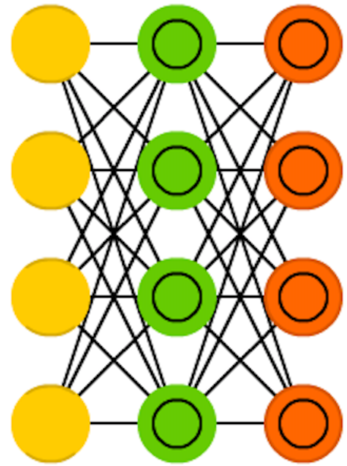
Long / Short Term Memory (LSTM)



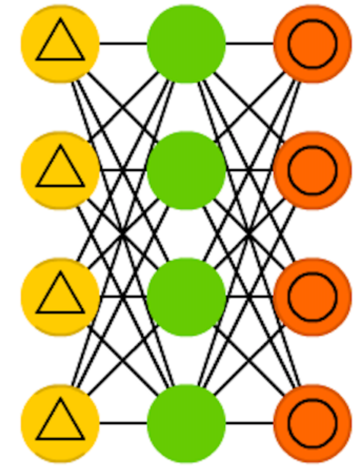
Auto Encoder (AE)



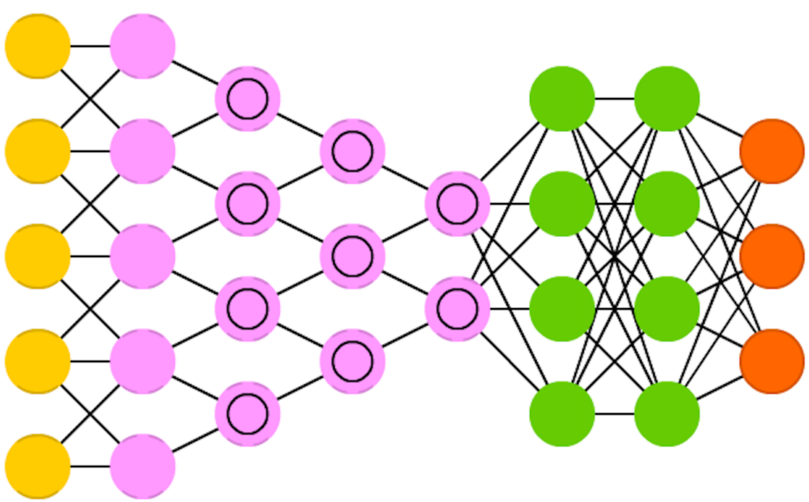
Variational AE (VAE)



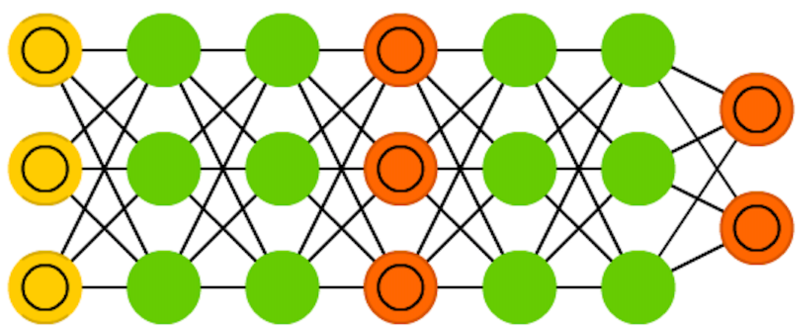
Denoising AE (DAE)

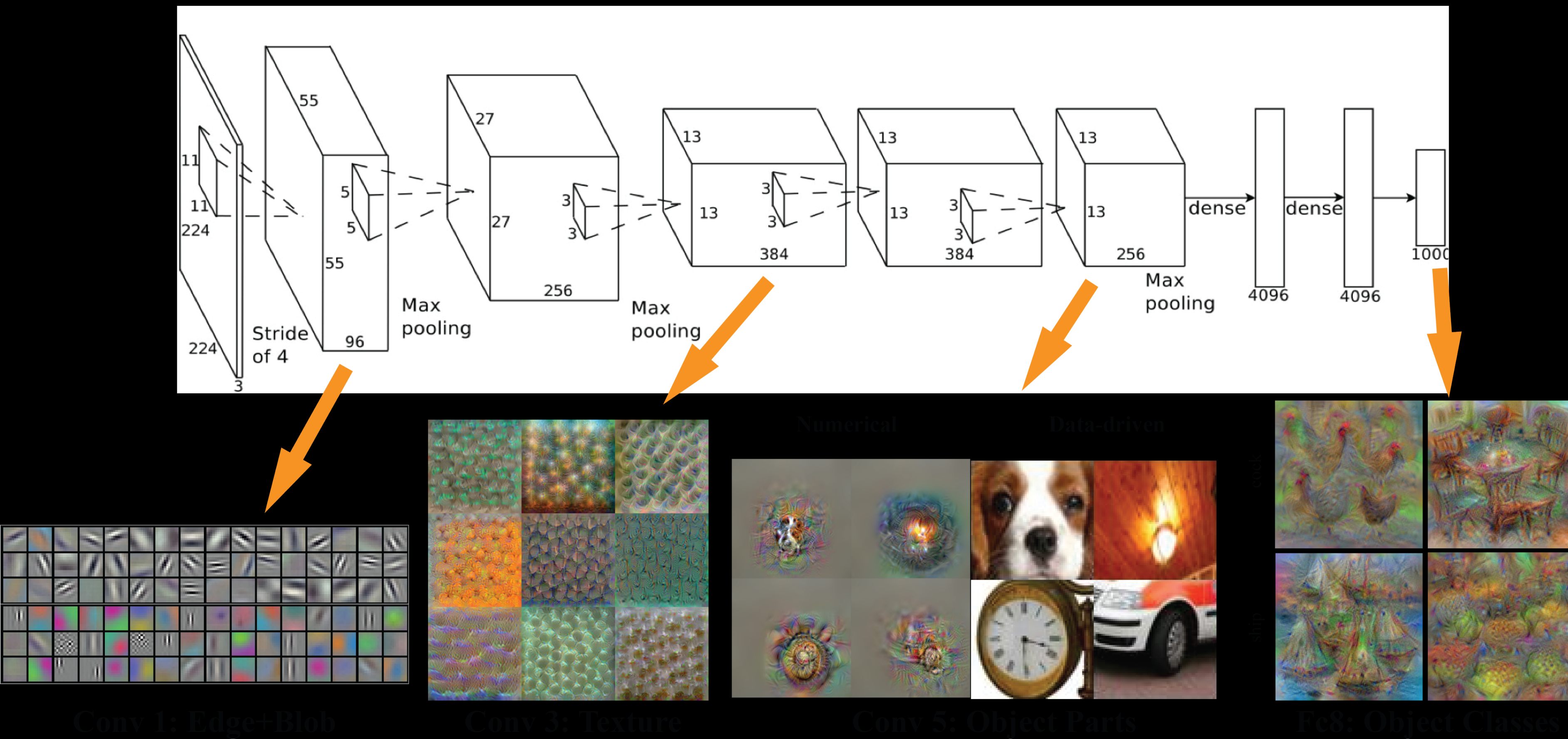


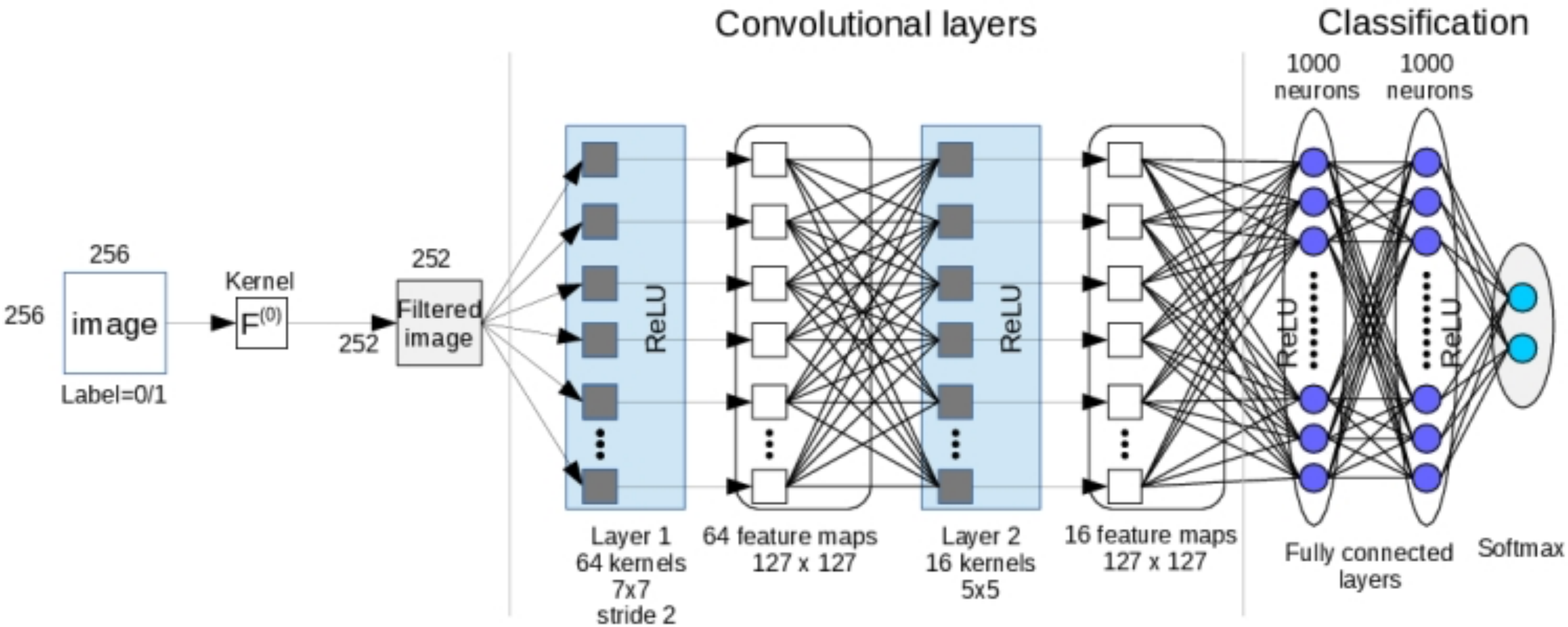
Deep Convolutional Network (DCN)

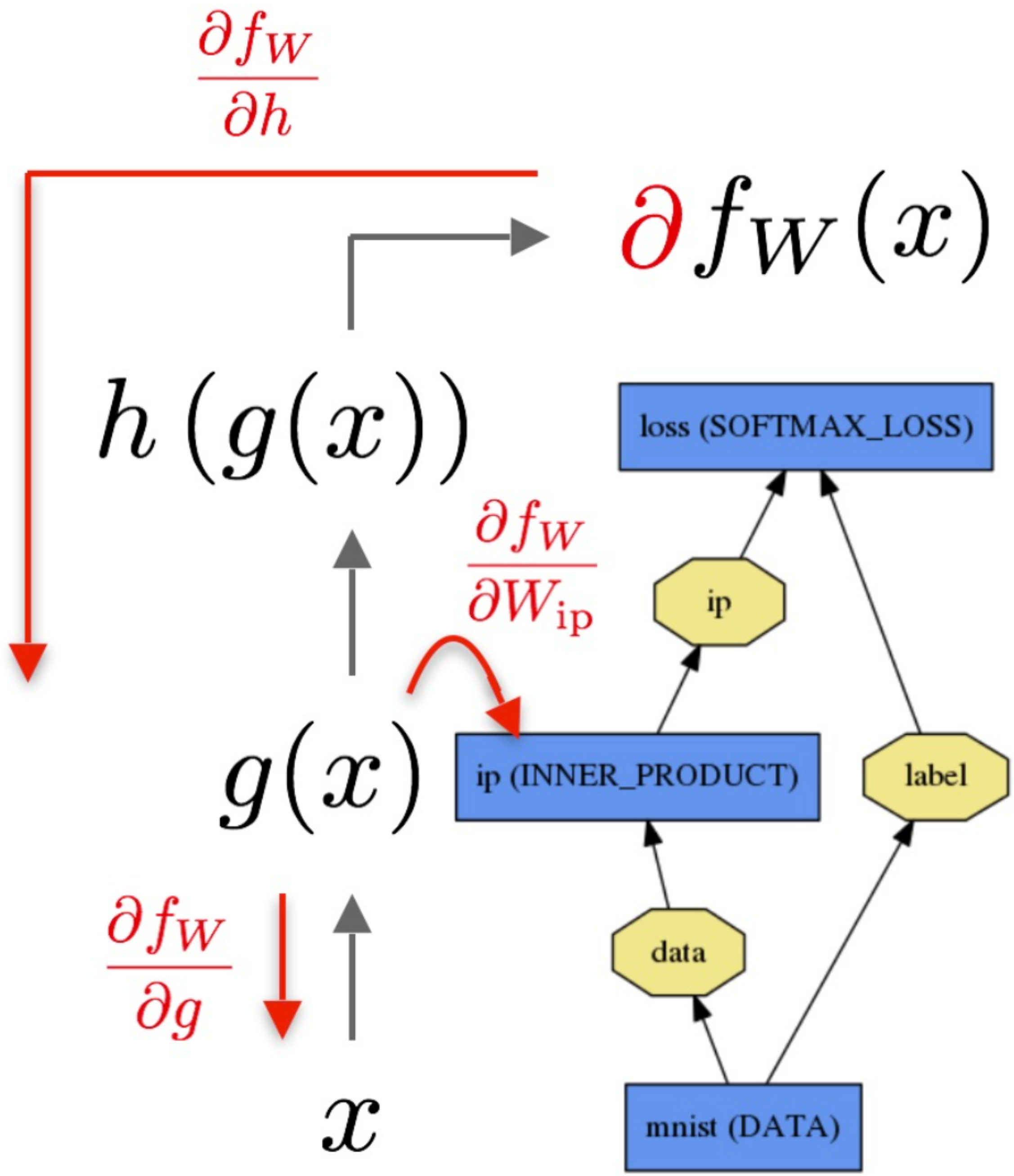


Generative Adversarial Network (GAN)




224
224
3
11
11
Stride of 4
55
55
96
5
5
Max pooling
27
27
256
3
3
Max pooling
13
13
384
13
13
384
13
13
256
Max pooling
dense
dense
4096
4096
1000
Numerical
Data-driven
Conv 1: Edge+Blob
Conv 3: Texture
Conv 5: Object Parts
Fc8: Object Classes

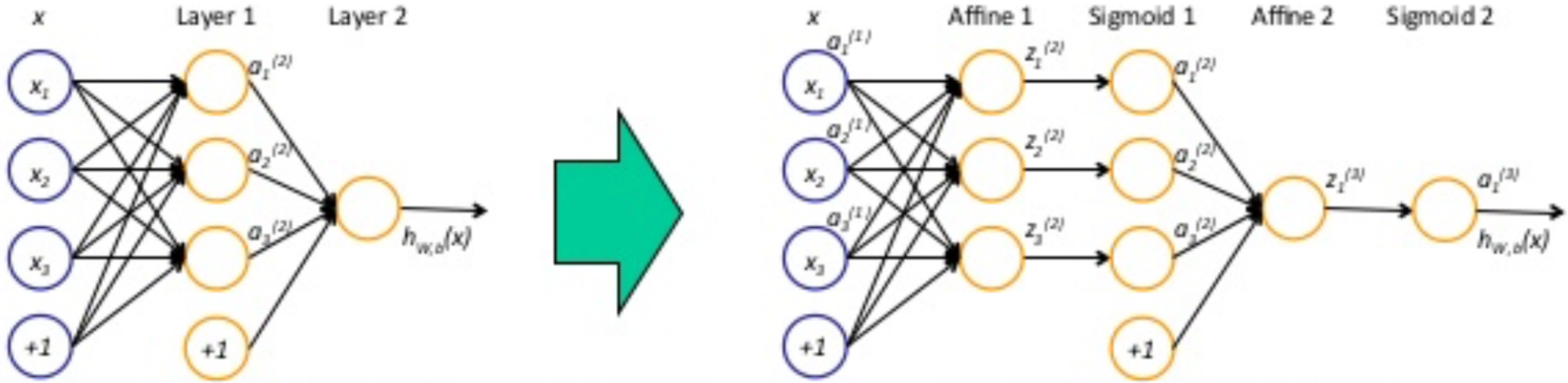

Convolutional layers
Classification
256
256
image
Label=0/1
Kernel
F(0)
252
252
Filtered image
ReLU
ReLU
Layer 1
64 kernels
7x7
stride 2
64 feature maps
127 x 127
Layer 2
16 kernels
5x5
16 feature maps
127 x 127
1000 neurons
1000 neurons
ReLU
ReLU
Fully connected layers
Softmax



$\frac{\partial f_W}{\partial h}$
$\partial f_W(x)$
$h(g(x))$
loss (SOFTMAX_LOSS)
$\frac{\partial f_W}{\partial W_{\mathrm{ip}}}$
ip
ip (INNER_PRODUCT)
$g(x)$
label
$\frac{\partial f_W}{\partial g}$
data
x
mnist (DATA)

Decomposition of Multi-Layer Neural Network

- Composite function representation of a multi-layer neural network



Standard network representation   Decomposition   Composite function representation

$$h_{W,b}(x) = \left(\text{sigmoid} \circ \text{affine}_{W^{(2)},b^{(2)}} \circ \text{sigmoid} \circ \text{affine}_{W^{(1)},b^{(1)}}\right)(x)$$

- Derivatives of function elements w.r.t. inputs and parameters

$$a^{(1)} = x, \; a^{(l_{max})} = h_{w,b}(x)$$

$$\frac{\partial a^{(l+1)}}{\partial z^{(l+1)}} = a^{(l+1)} \bullet \left(1 - a^{(l+1)}\right) \text{ where } a^{(l+1)} = \text{sigmoid}\left(z^{(l+1)}\right) = \frac{1}{1 + \exp\left(-z^{(l+1)}\right)}$$

$$\frac{\partial z^{(l+1)}}{\partial a^{(l)}} = W^{(l)}, \; \frac{\partial z^{(l+1)}}{\partial W^{(l)}} = a^{(l)}, \; \frac{\partial z^{(l+1)}}{\partial b^{(l)}} = I \text{ where } z^{(l+1)} = \left(W^{(l)}\right)^T a^{(l)} + b^{(l)}$$

Backpropagation in Convolution Layer

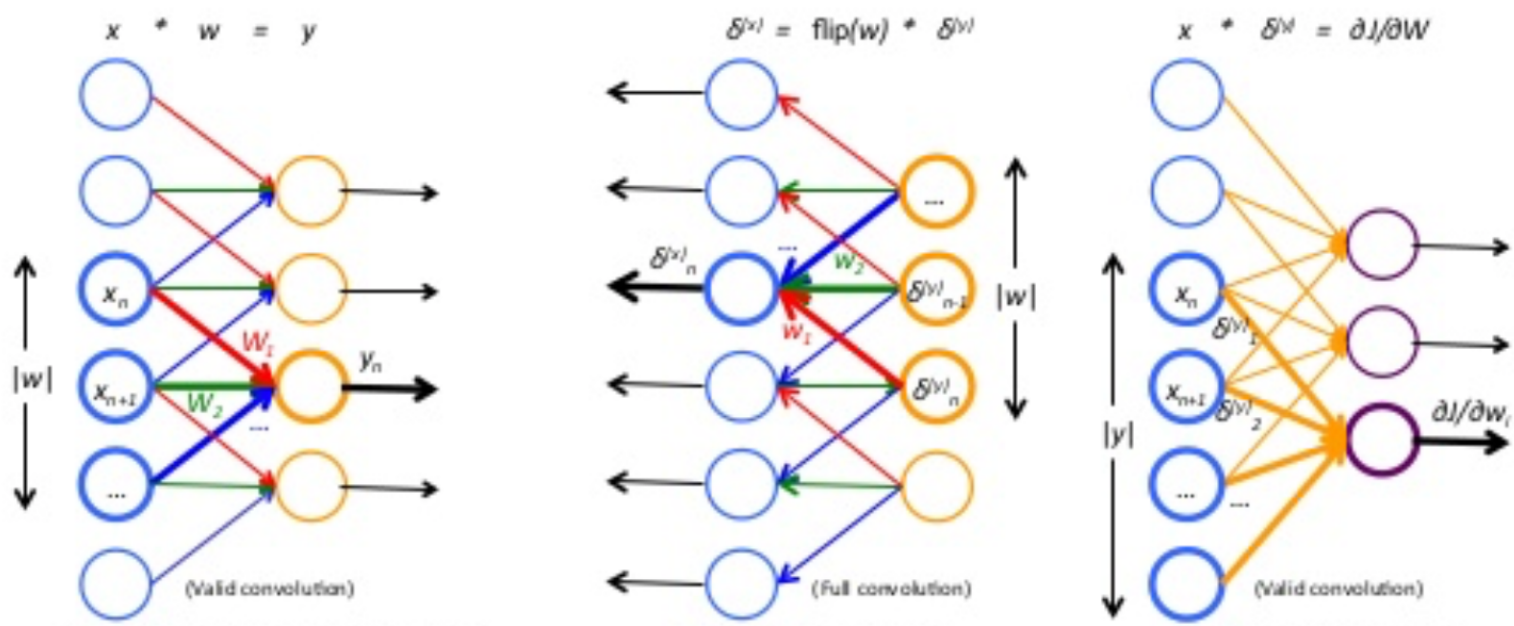
Error signals and gradient for each example are computed by convolution using the commutativity property of convolution and the multivariable chain rule of derivative.

Let's focus on single elements of error signals and a gradient w.r.t. w.

$$\delta_n^{(x)} = \frac{\partial J}{\partial x_n} = \frac{\partial J}{\partial y}\frac{\partial y}{\partial x_n} = \sum_{i=1}^{|w|} \frac{\partial J}{\partial y_{n-i+1}}\frac{\partial y_{n-i+1}}{\partial x_n} = \sum_{i=1}^{|w|} \delta_{n-i+1}^{(y)} w_i = \left(\delta^{(y)} * \mathrm{flip}(w)\right)[n],\ \delta^{(x)} = \left[\delta_n^{(x)}\right] = \delta^{(y)} * \mathrm{flip}(w)$$

↑ Reverse order linear combination

$$\frac{\partial J}{\partial w_i} = \frac{\partial J}{\partial y}\frac{\partial y}{\partial w_i} = \sum_{n=1}^{|x|-|w|+1} \frac{\partial J}{\partial y_n}\frac{\partial y_n}{\partial w_i} = \sum_{n=1}^{|x|-|w|+1} \delta_n^{(y)} x_{n+i-1} = \left(\delta^{(y)} * x\right)[i],\ \frac{\partial J}{\partial w} = \left[\frac{\partial J}{\partial w_i}\right] = \delta^{(y)} * x = x * \delta^{(y)}$$



Forward propagation (convolution) — Backward propagation — Gradient computation

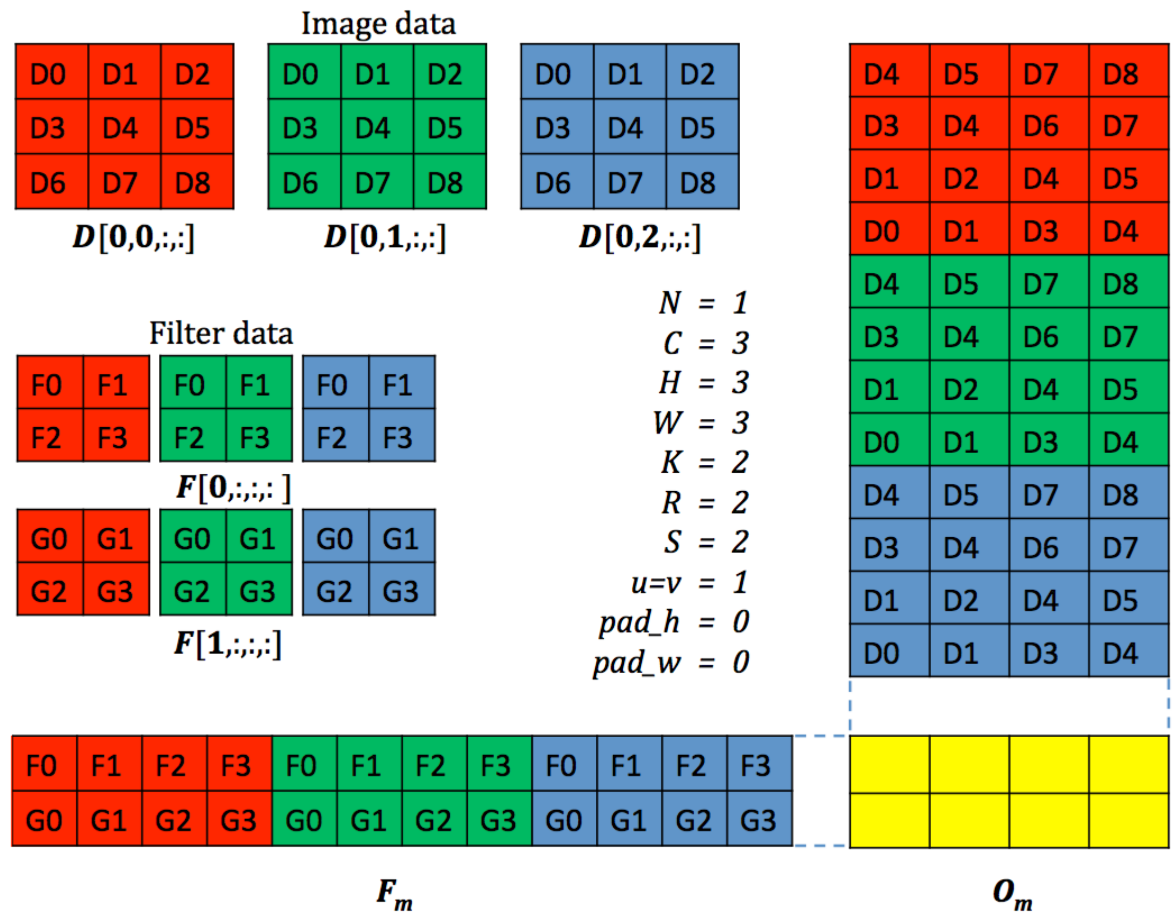
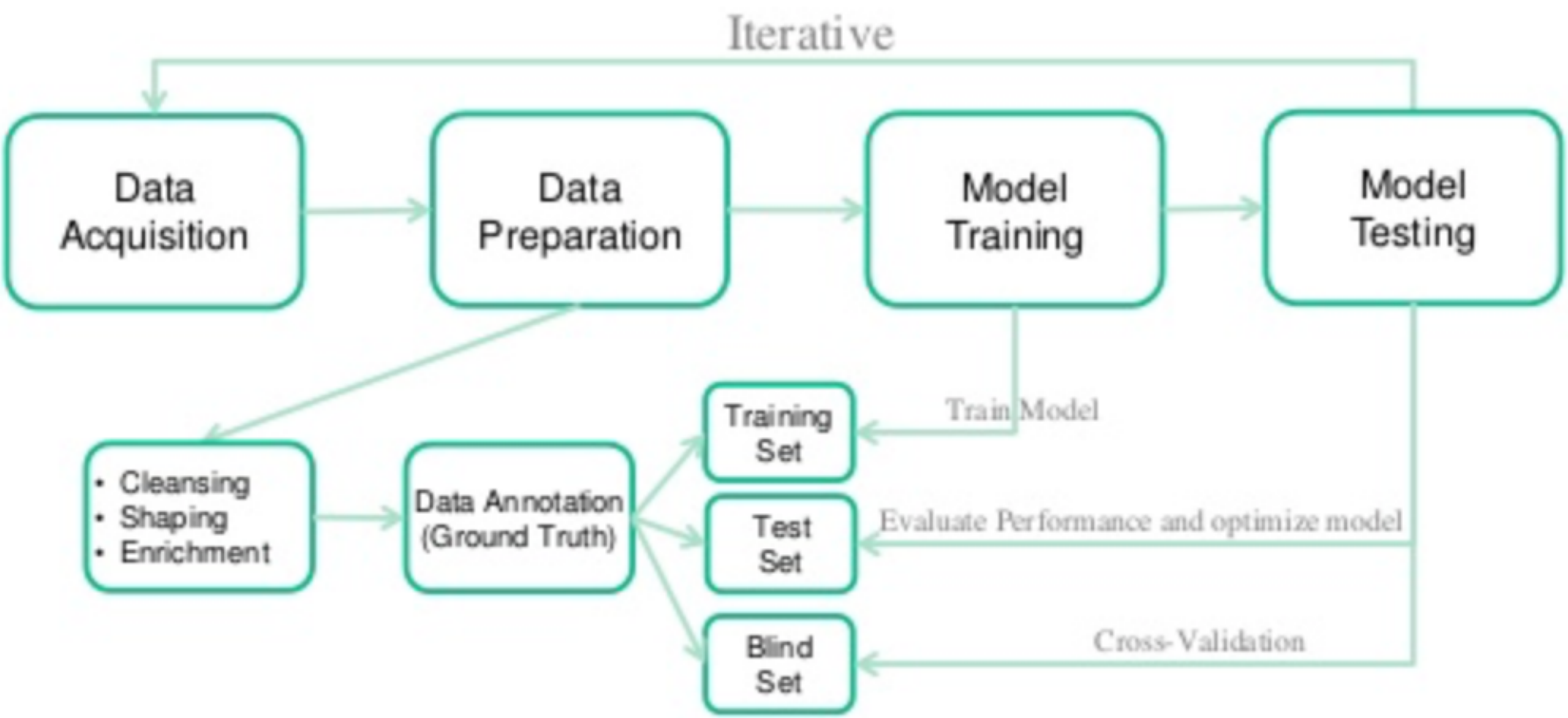


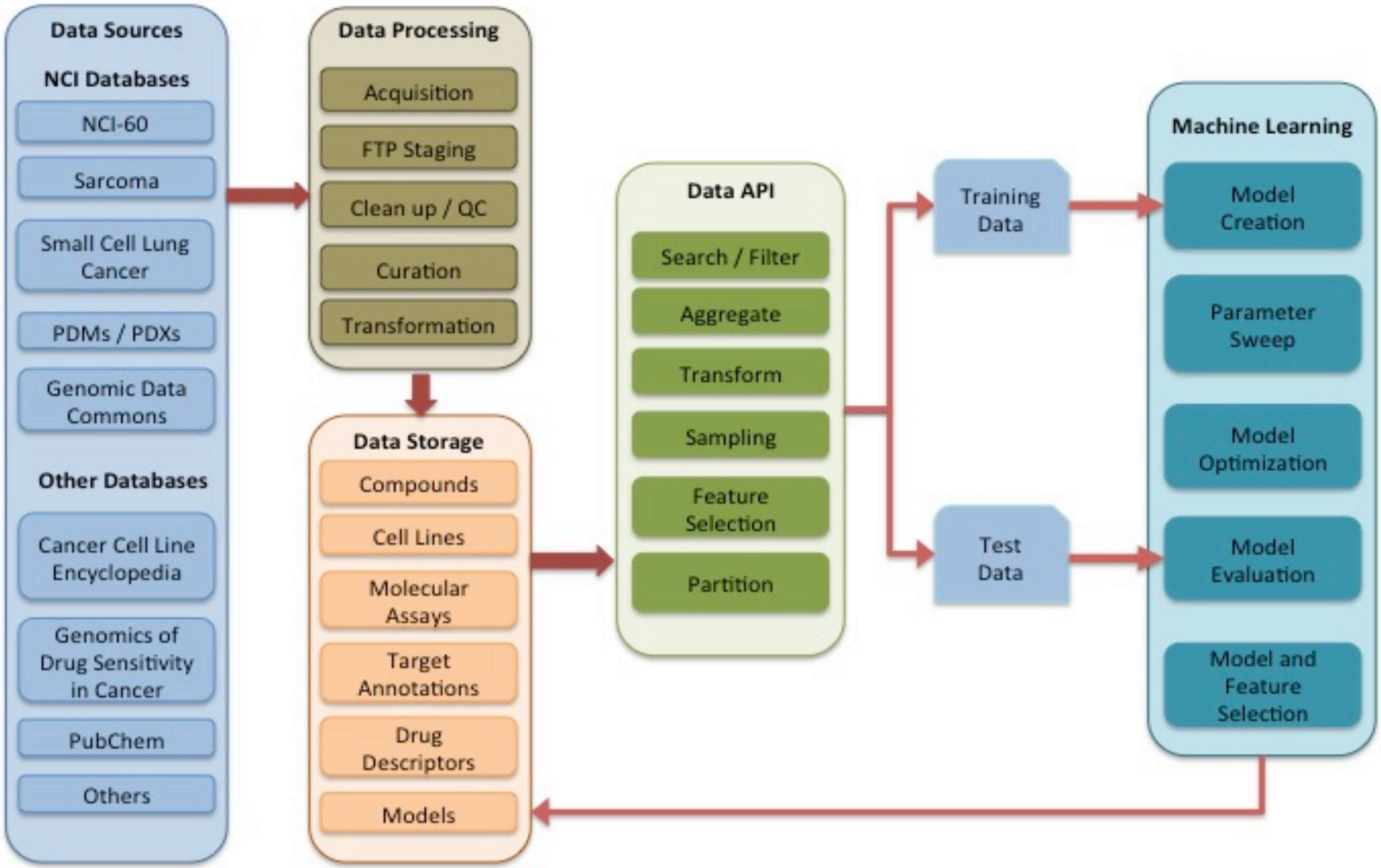
Figure 1: Convolution lowering

Figure 1 illustrates how a simple convolution can be lowered to a matrix multiplication. The colors in this illustration represent the input feature maps, and elements of D and F are uniquely labeled in the illustration so as to show how each participates in forming D_m and F_m. The filter matrix F_m has dimensions $K \times CRS = 2 \times 12$, while the data matrix D_m has dimensions $CRS \times NPQ = 12 \times 4$. Note that each element of D is duplicated up to $RS = 4$ times in D_m. The output matrix O_m has dimensions $K \times NPQ = 2 \times 4$.

Typical Machine Learning Flow diagram



Drug Response CANDLE General Workflow

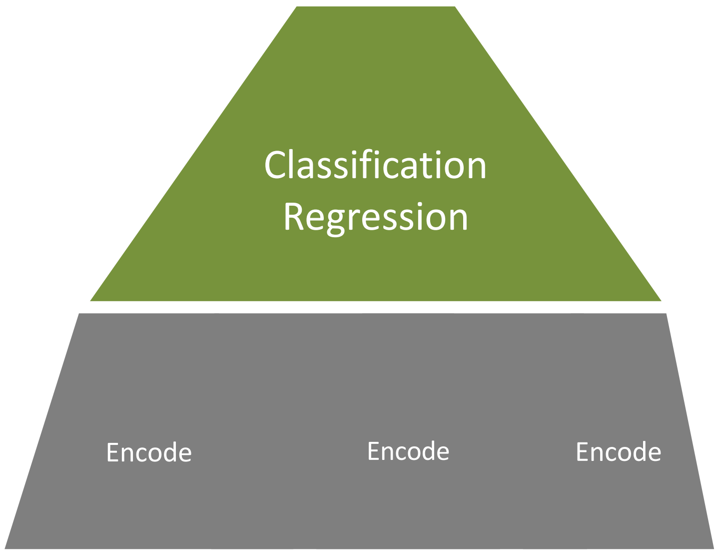


Cancer Data Processing, Storage and Machine Learning Workflow

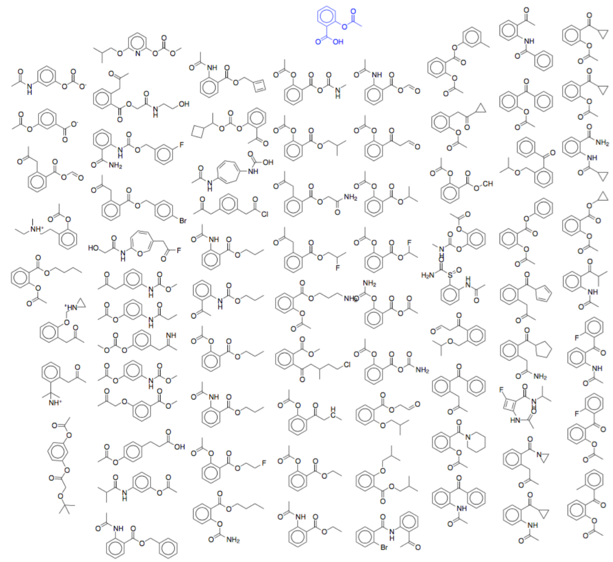
Drug Combination Response Prediction

How to search 1 trillion drug combinations?

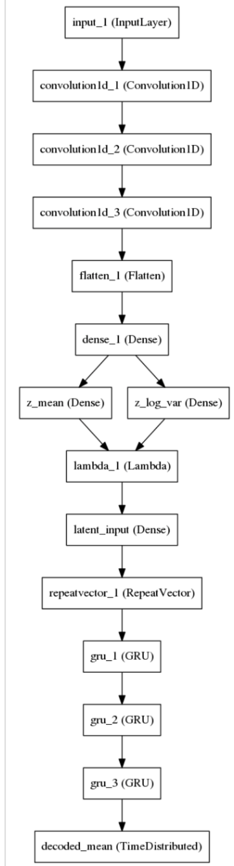
Response Prediction

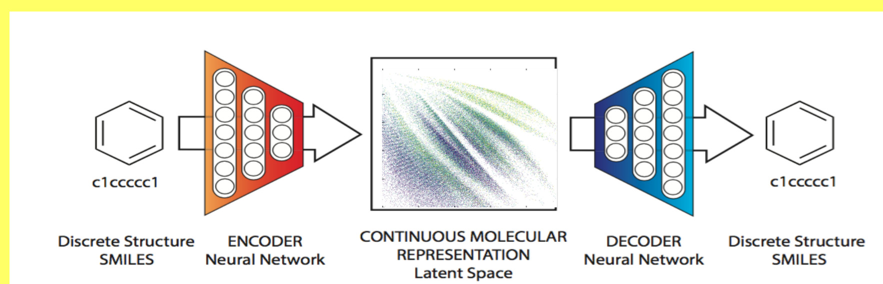


Tumor	Drug1	Drug2
expression SNPs protein microRNA methylation	descriptors fingerprints structures SMILES dose	descriptors fingerprints structures SMILES dose



Generated Molecules





Aspuru-Guzik et.al. Variational Autoencoder

Preliminary Deep Learning on Combination Drug Response

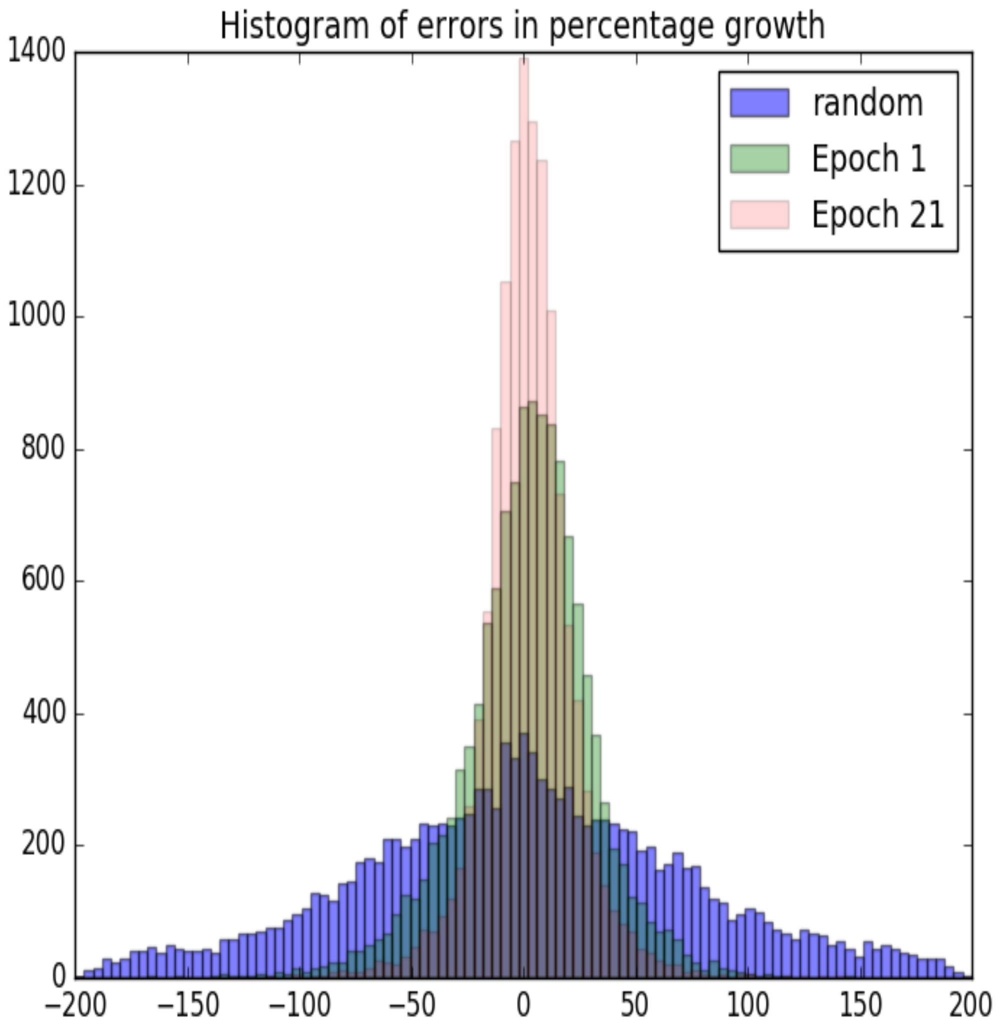
Classification: 87.9% acc

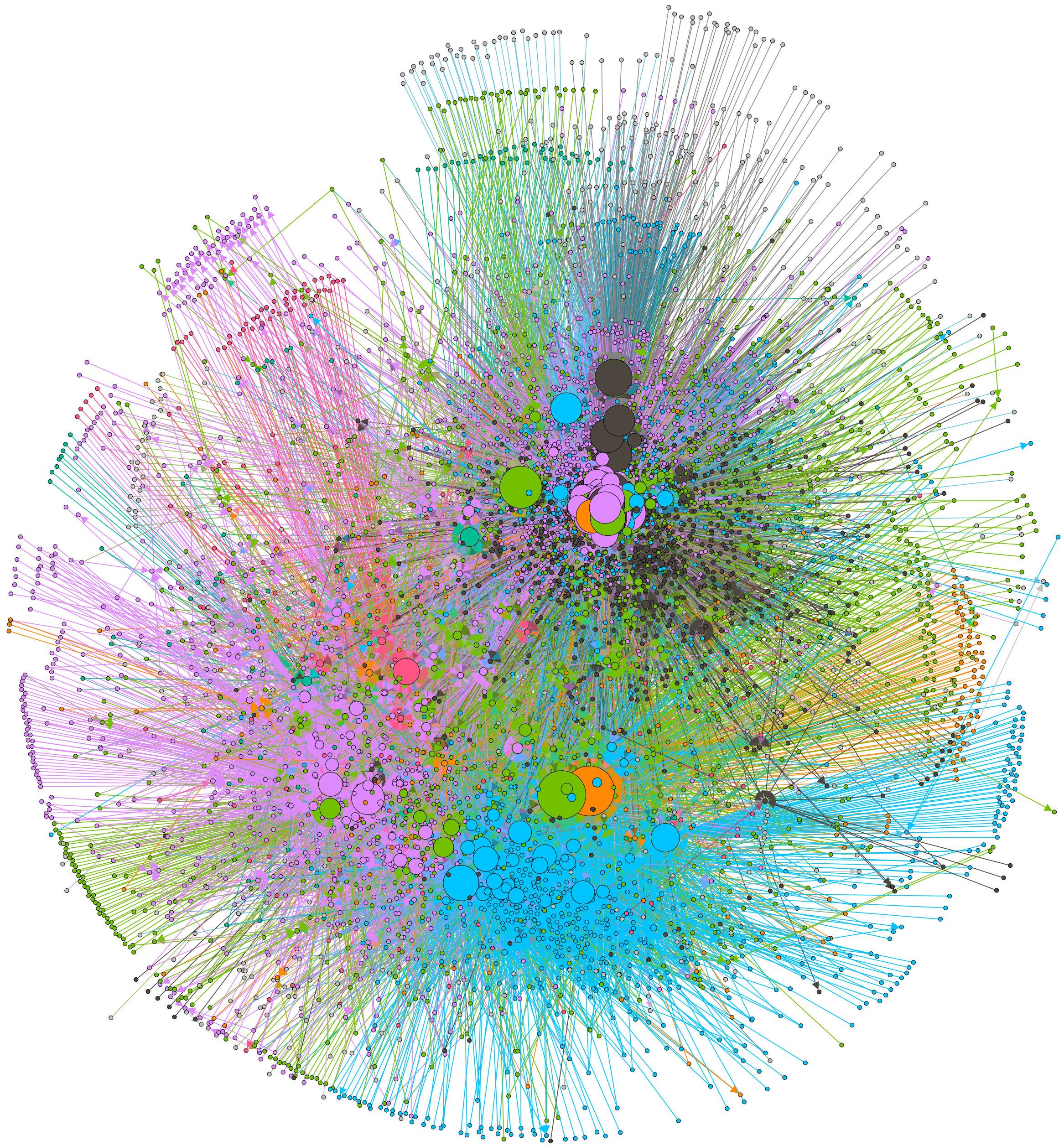
Regression: 0.036 mse loss

- 3,580,891 samples
- 310,898 unique (CL, D1, D2) combinations

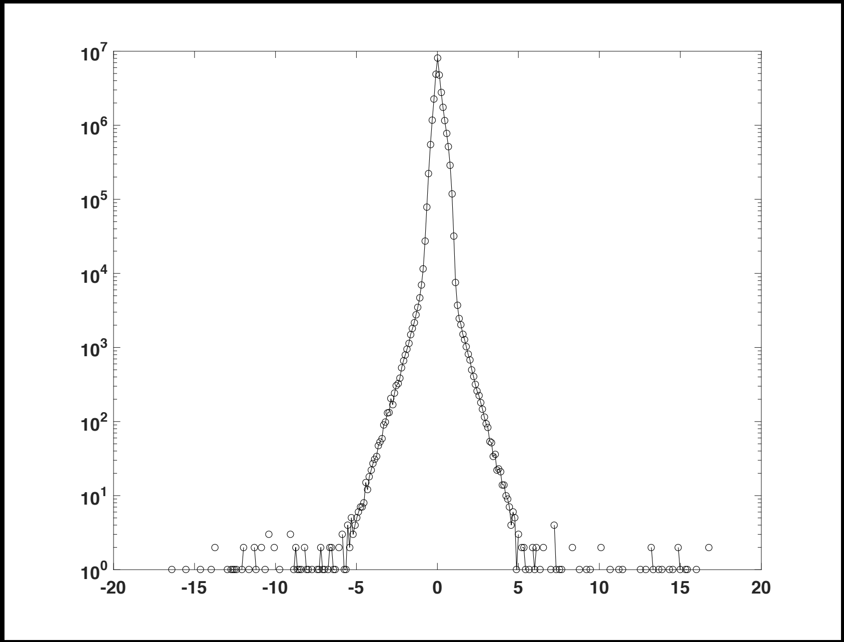
- 33,362 features
- Cell Line: 25,722 RNA
- D1/D2: 3,820 descriptors

HP Sweep is > 10K cases

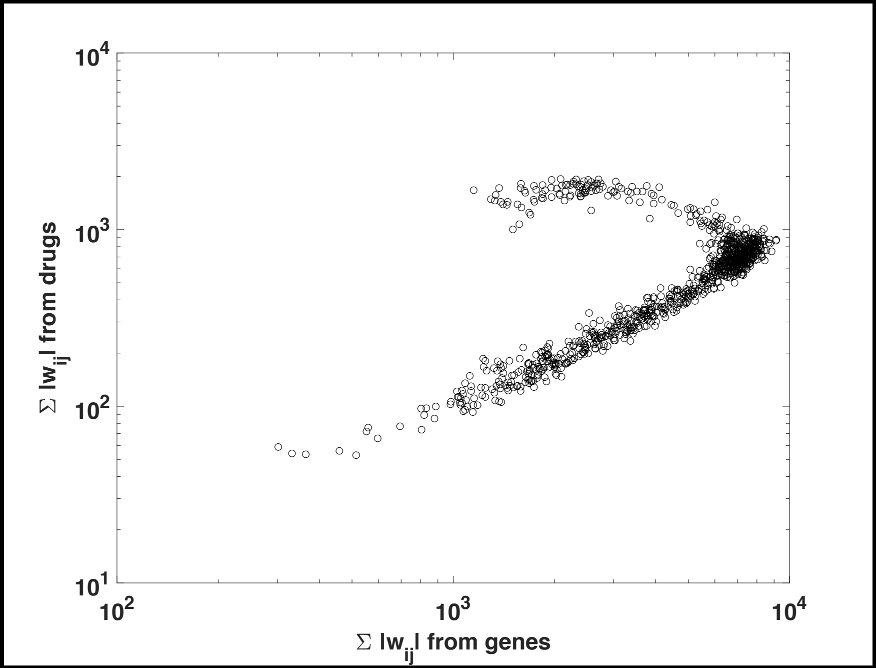




Weights range from -10^7 to $+10^7$ in the first fully connected layer



Limited dyanmic range



Input from drug descriptors and from tumor assay are contributing to upper level features

Hyperparameter Search

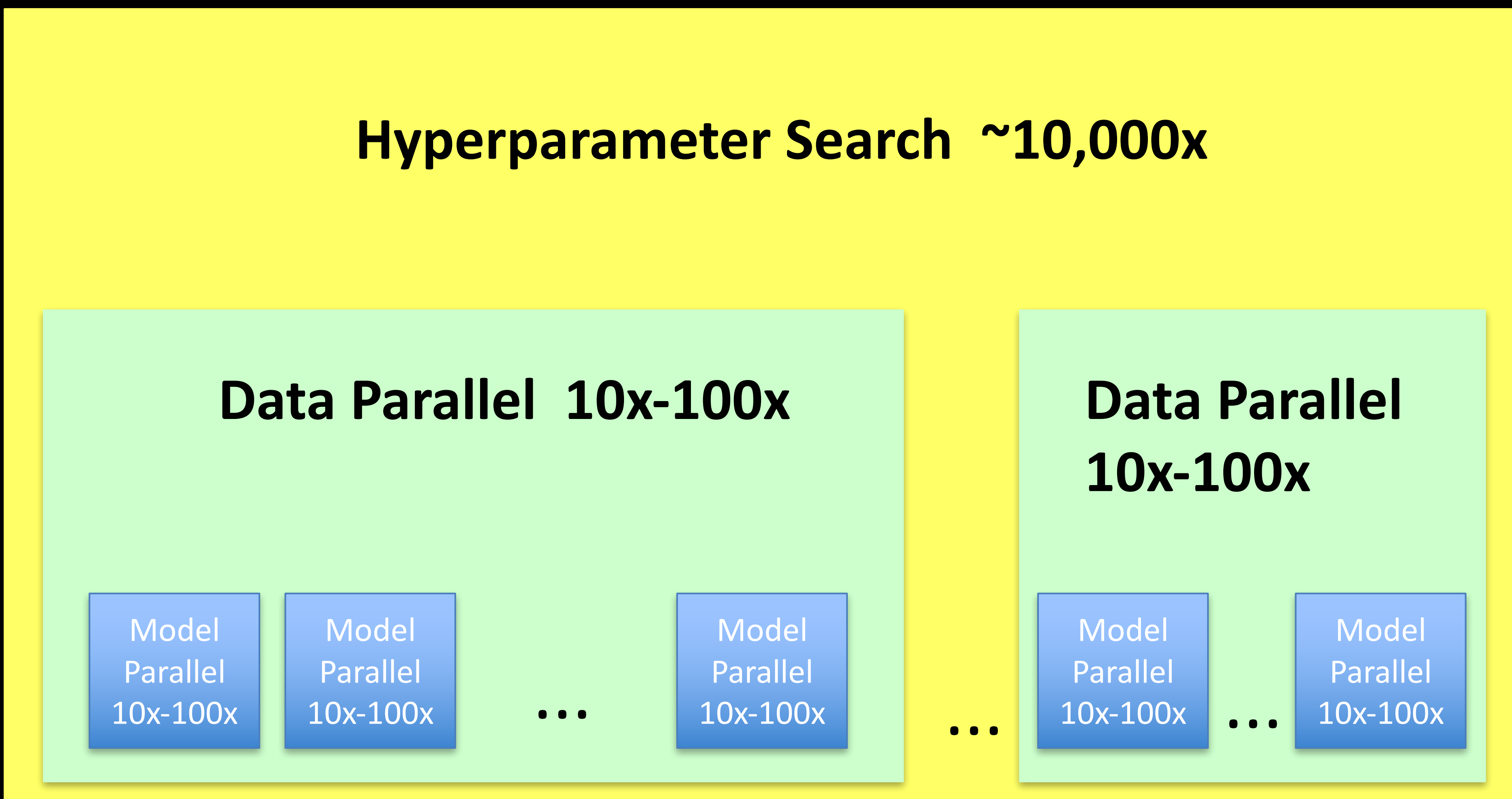
3 x 3 x 3 x 4 x 3 x 3 x 3 x 4 = 11,664 cases

Hyperparameter	Considered values
Normalization	{standard-deviation, tanh, sqrt}
Feature type	{molecular-descriptors, tox-and-scaffold-similarities, ECFP4}
Fingerprint sparseness threshold	{5, 10, 20}
Number of Hidden Units	{1024, 4096, 8192, 16356}
Number of Layers	{1, 2, 3}
Learning Rate	{0.01, 0.05, 0.1}
Dropout	{no, yes (50% Hidden Dropout, 20% Input Dropout)}
L2 Weight Decay	$\{0, 10^{-6}, 10^{-5}, 10^{-4}\}$

Table 1. Hyperparameters considered for the neural networks. **Normalization:** Scaling of the predefined features. **Feature type:** Determines which of the features were used as input features. "molecular-descriptors" were the real-valued descriptors. "tox-and-scaffold-similarities" were the similarity scores to known toxicophores and scaffolds, "ECFP4" were the ECFP4 fingerprint features. We tested all possible combinations of these features. **Fingerprint sparseness threshold:** A feature was not used if it was only present in fewer compounds than the given number. **Number of hidden units:** The number of units in the hidden layer of the neural network. **Number of layers:** The number of layers of the neural network. **Learning rate:** The learning rate for the backpropagation algorithm. **Dropout:** Dropout rates. **L2 Weight Decay:** The weight decay hyperparameter.

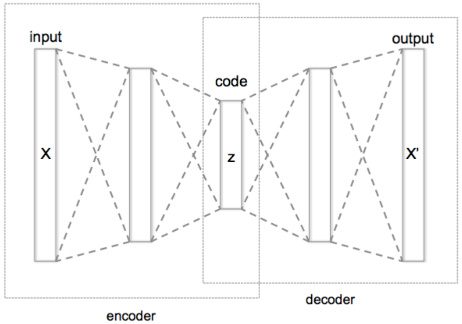
Parallelism Targets in CANDLE

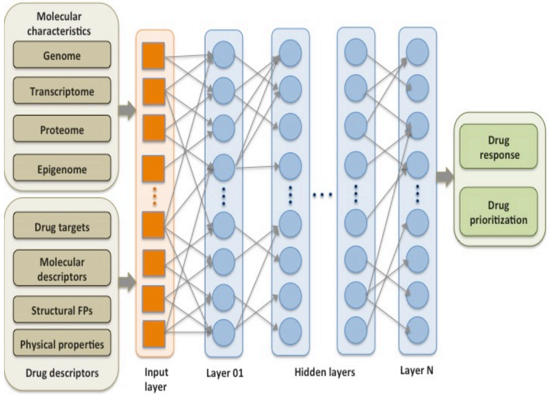
10,000 x 10-100 x 10-100 = 1M – 100M cores

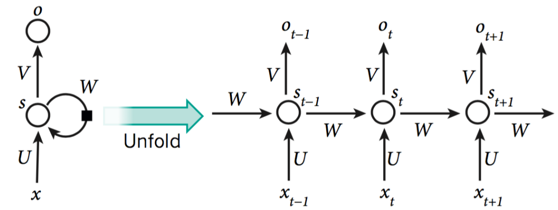


CANDLE Benchmarks.. Representitve problems

- **Variational Autoencoder**
  - Learning (non-linear) features of core data types
- **Autoencoder**
  - Molecuar dynamics trajectory state detection
- **MLP+LCNN Classification**
  - Cancer type from gene expression/SNPs
- **MLP+LCNN Regression**
  - NCI-60 drug response (gene exp, descriptors)
- **CNN**
  - Cancer pathology report term extraction
- **RNN-LSTM**
  - Cancer pathology report text analysis
- **RNN-LSTM**
  - Molecuar dynamics simulation control







7 CANDLE Benchmarks

https://github.com/ECP-CANDLE

Benchmark Owners:
- P1: Fangfang Xia (ANL)
- P2: Brian Van Essen (LLNL)
- P3: Arvind Ramanathan (ORNL)

Benchmark	Type	Data	ID	OD	Sample Size	Size of Network	Additional (activation, layer types, etc.)
1. P1: B1 Autoencoder	MLP	RNA-Seq	10^5	10^5	15K	5 layers	Log2 (x+1) → [0,1] KPRM-UQ
2. P1: B2 Classifier	MLP	SNP → Type	10^6	40	15K	5 layers	Training Set Balance issues
3. P1: B3 Regression	MLP+LCN	expression; drug descs	10^5	1	3M	8 layers	Drug Response [-100, 100]
4. P2: B1 Autoencoder	MLP	MD K-RAS	10^5	10^2	10^6-10^8	5-8 layers	State Compression
5. P2: B2 RNN-LSTM	RNN-LSTM	MD K-RAS	10^5	3	10^6	4 layers	State to Action
6. P3: B1 RNN-LSTM	RNN-LSTM	Path reports	10^3	5	5K	1-2 layers	Dictionary 12K +30K
7. P3: B2 Classification	CNN	Path reports	10^4	10^2	10^5	5 layers	Biomarkers

Innovation: Ensemble Deep Learning

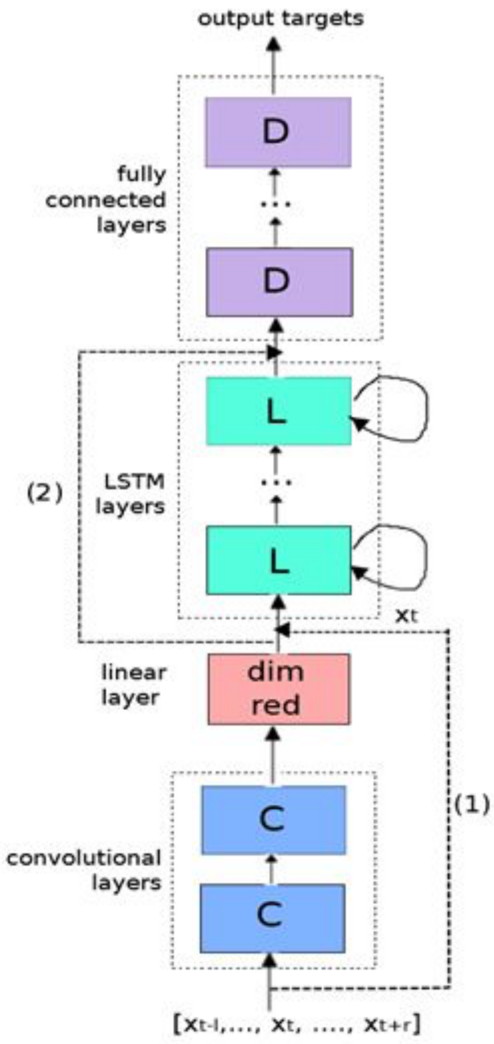
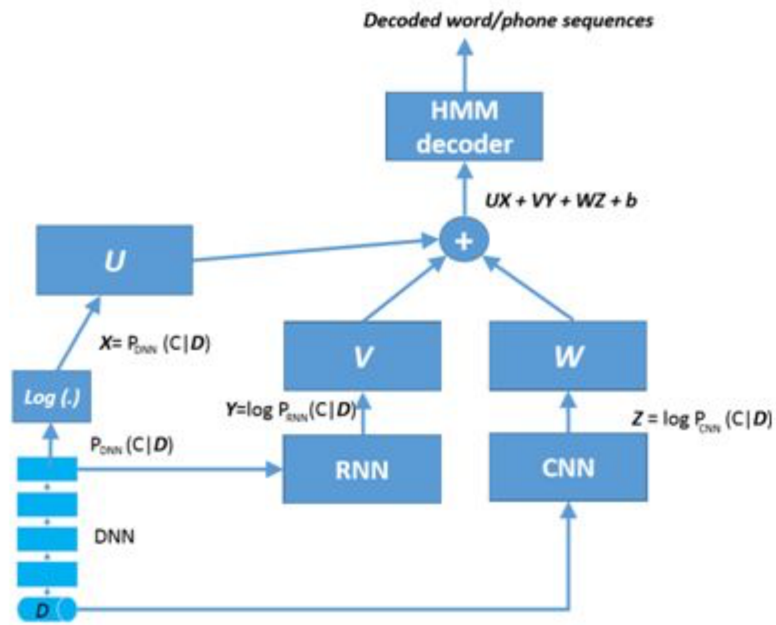
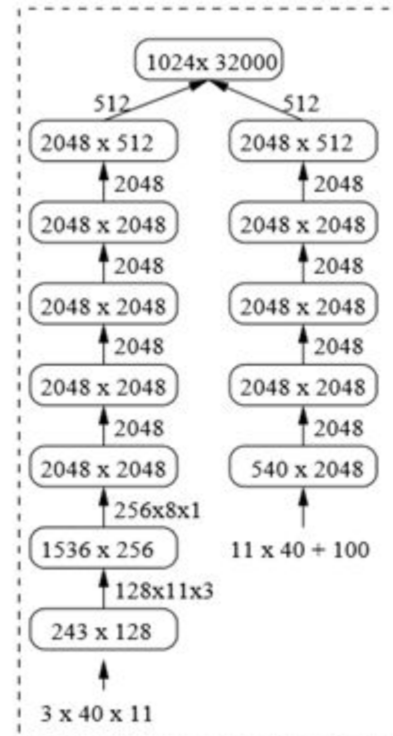


Fig. 1. CLDNN Architecture

- **Ensembles of RNN/LSTM, DNN, & Conv Nets (CNN) give huge gains (state of the art):**
- T. Sainath, O. Vinyals, A. Senior, H. Sak. "Convolutional, Long Short-Term Memory, Fully Connected Deep Neural Networks," ICASSP 2015.
- L. Deng and John Platt, Ensemble Deep Learning for Speech Recognition, Interspeech, 2014.
- G. Saon, H. Kuo, S. Rennie, M. Picheny. "The IBM 2015 English conversational telephone speech recognition system," arXiv, May 2015. (8% WER on SWB-309h)





Joint CNN/DNN

Github and FTP

- **ECP-CANDLE GitHub Organization:**
- https://github.com/ECP-CANDLE

- **ECP-CANDLE FTP Site:**
- The FTP site will be used to host all the public datasets for the benchmarks from three pilots.
- http://ftp.mcs.anl.gov/pub/candle/public/

BDEC Questions for Deep Learning

- What are the key frameworks and workloads for Deep Learning?
- Is Deep Learning becoming a major element of scientific computing applications?
- What hardware and systems architectures are emerging for supporting deep learning?
- Is Deep Learning a distinct class worthy of its own software stack in the BDEC Universse?

What are the key frameworks and workloads for Deep Learning?

Framework Comparison: Basic information*

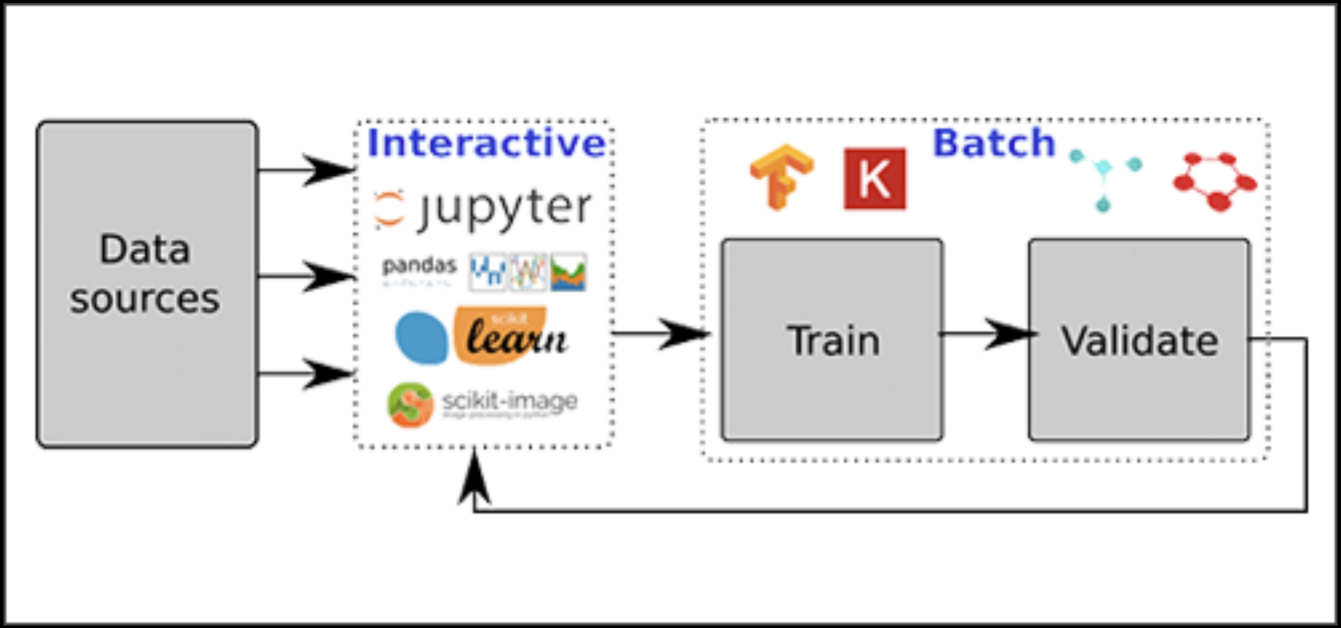
Viewpoint	Torch.nn**	Theano***	Caffe	autograd (NumPy, Torch)	Chainer	MXNet	Tensor-Flow
GitHub stars	4,719	3,457	9,590	N: 654 T: 554	1,295	3,316	20,981
Started from	2002	2008	2013	2015	2015	2015	2015
Open issues/PRs	97/26	525/105	407/204	N: 9/0 T: 3/1	95/25	271/18	330/33
Main developers	Facebook, Twitter, Google, etc.	Université de Montréal	BVLC (U.C. Berkeley)	N: HIPS (Harvard Univ.) T: Twitter	Preferred Networks	DMLC	Google
Core languages	C/Lua	C/Python	C++	Python/Lua	Python	C++	C++/Python
Supported languages	Lua	Python	C++/Python MATLAB	Python/Lua	Python	C++/Python R/Julia/Go etc.	C++/Python

* Data was taken on Apr. 12, 2016
** Includes statistics of Torch7
*** There are many frameworks on top of Theano, though we omit them due to the space constraints

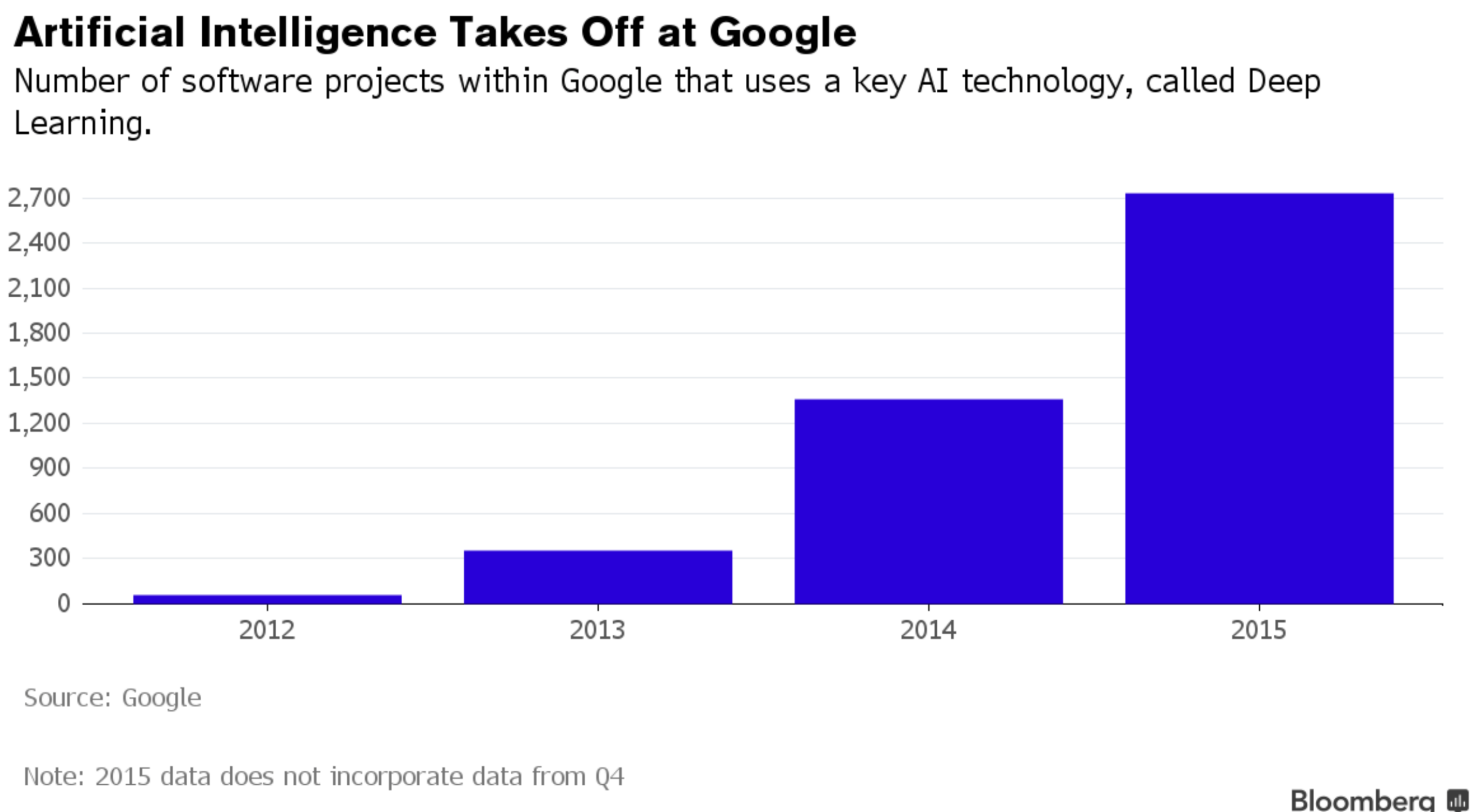
Aggregate popularity (30•contrib + 10•issues + 5•forks)•1e-3

#1:	97.53	tensorflow/tensorflow
#2:	71.11	BVLC/caffe
#3:	43.70	fchollet/keras
#4:	32.07	Theano/Theano
#5:	31.96	dmlc/mxnet
#6:	19.51	deeplearning4j/deeplearning4j
#7:	15.63	Microsoft/CNTK
#8:	13.90	torch/torch7
#9:	9.03	pfnet/chainer
#10:	8.75	Lasagne/Lasagne
#11:	7.84	NVIDIA/DIGITS
#12:	7.83	mila-udem/blocks
#13:	5.95	karpathy/convnetjs
#14:	5.84	NervanaSystems/neon
#15:	4.91	tflearn/tflearn
#16:	3.28	amznlabs/amazon-dsstne
#17:	1.81	IDSIA/brainstorm
#18:	1.38	torchnet/torchnet

Interactive Computing is key in Deep Learning workflows



How Many Projects?



Deep Learning is becoming a major element of scientific computing applications

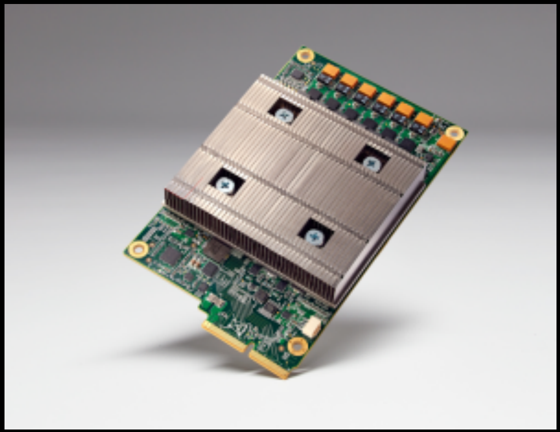
- Across the DOE lab system hundreds of examples are emerging
  - From fusion energy to precision medicine
  - Materials design
  - Fluid dynamics
  - Genomics
  - Structural engineering
  - Intelligent sensing
  - Etc.

DL System Architecture Challenges

- **Node Centric vs Network Centric**
  - Integrated resources on a node
  - Disaggregated resources on a network*
  - Static Ratios or Dynamic Ratios*
- **Name Space/Address Space Across Instances/Stacks**
  - One integrated space across stacks
  - Each stack maintains names and addresses*
  - Are technology components converging?
- **Training vs Inferencing**..
  - Online vs offline training
  - Embeddable in simulation environments*

Hardware and systems architectures are emerging for supporting deep learning?

- CPUs
  - AVX, VNNI, KNM, KNH, …
- GPUs
  - Nvidia P100, AMD Instinct, Baidu GPU, …
- ASICs
  - Nervana, DianNao, Eyeriss, GraphCore, TPU, DLU, …
- FPGA
  - Arria 10, Stratix 10, Falcon Mesa, …
- Neuromorphic
  - True North, Zeroth, N1, …



Is Deep Learning a distinct class worthy of its own software stack in the BDEC Universe?

CANDLE Software Stack

Hyperparameter Sweeps, Data Management (e.g. DIGITS, Swift, etc.)	Workflow
Network description, Execution scripting API (e.g. Keras, Mocha)	Scripting
Tensor/Graph Execution Engine (e.g. Theano, TensorFlow, LBANN-LL, etc.)	Engine
Architecture Specific Optimization Layer (e.g. cuDNN, MKL-DNN, etc.)	Optimization

CANDLE Workflow Layer

- "Convienence and Productivity" layer
- Used to manage large-scale training runs
  - Hyperparameter searches $O(10^4)$ jobs
  - Cross validation (5-fold, 10-fold, etc.)
  - Data encodings (log2, Z-score, percent, etc.)
  - Low-level optimizations (tensor backends)
- Locate and transform input data
- Manage caching on local NV store
  - Internal joins, batching management, epochs
- Each job could be 100's to 1000's of nodes
- Driver scripts manage runs of 1K >10M core/hrs

Model Scripting Interface

- Aimed at the user developing models.. Keras is our canonical example
- **Keras** – python interface
  - Theano and TensorFlow
  - target for LBANN
- **Mocha** – julia interface (similar to Caffe)
  - Pure julia backend
  - cuDNN
- **Lasagne** – python interface
  - Theano
- **Torch7 NN** – Lua Interface
  - Torch (TH Tensor Library)

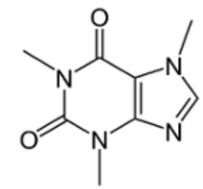
DL Frameworks “Tensor Engines”

- **TensorFlow** (c++, symbolic diff+)
- **Theano** (c++, symbolic diff+)
- **Neon** (integrated) (python + GPU, symbolic diff+)
- **Torch7 TH Tensor** (c layer, symbolic diff-, pgks)
- **Mxnet** (integrated) (c++)
- **Caffe** (integrated) (c++, symbolic diff-)
- **Mocha** backend (julia + GPU)
- **LBANN** (c++, aimed at scalable hardware)
- **CNTK** backend (microsoft) (c++)
- **PaddlePaddle** (Baidu) (python, c++, GPU)

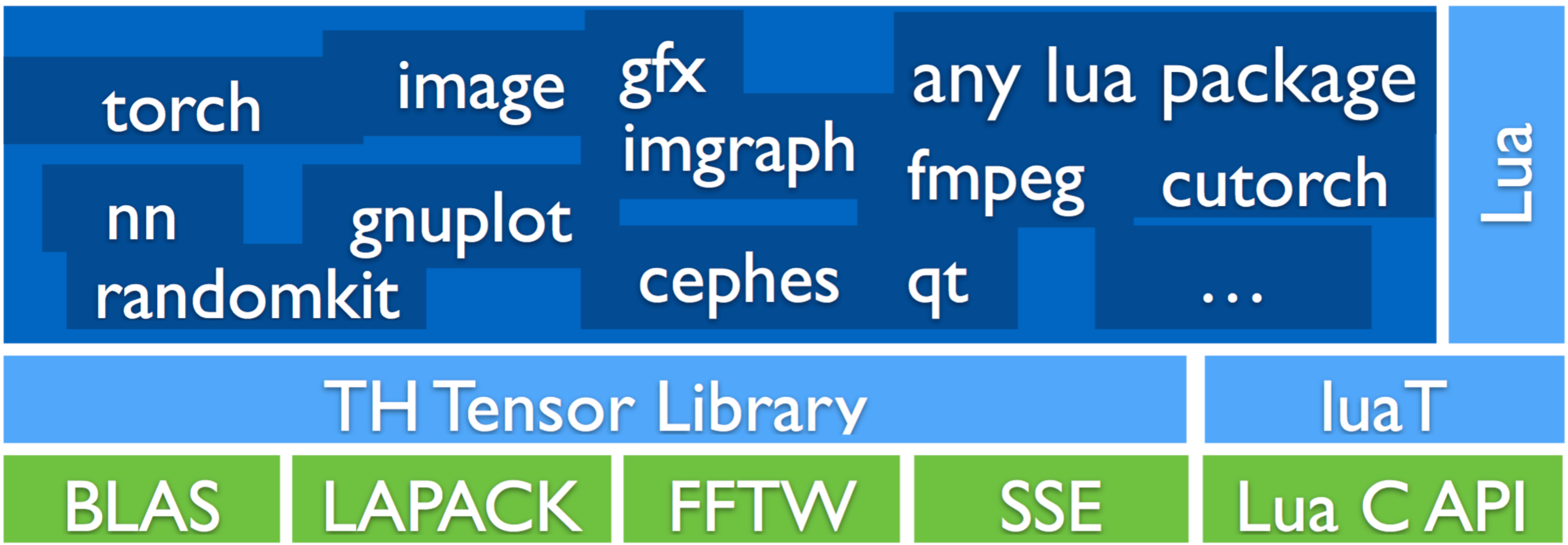
TensorFlow

theano

neon



Torch7 "Stack"



Hardware Optimization Layers

- **cuDNN** – NVIDIA low level library
  - Caffe, TensorFlow, Theano, Torch, CNTK
  - Supports many DL features, forwad and backward layer types for common topologies
  - Forward and backward convolution
- **MKL-DNN** – intel deep learning library
  - Convolution, pooling, ReLU, etc. C API
  - Cifar, *AlexNet*, VGG, *GoogleNet* and ResNet*.

Parallelism Options and I/O

- **Data Parallelism** (distributed training by partitioning training data)
  - Can this be managed at the L2 (L3?) independently of L1?
- **Model parallelism** (parallel training by partitioning network)
  - Can this be managed at the L0 and L1 levels independently of L2?
- **Streaming** training data loaders at what level?
- **Dashboard** reporting at L2?
- **Main IO** at L2?

Hybrid Models in Cancer

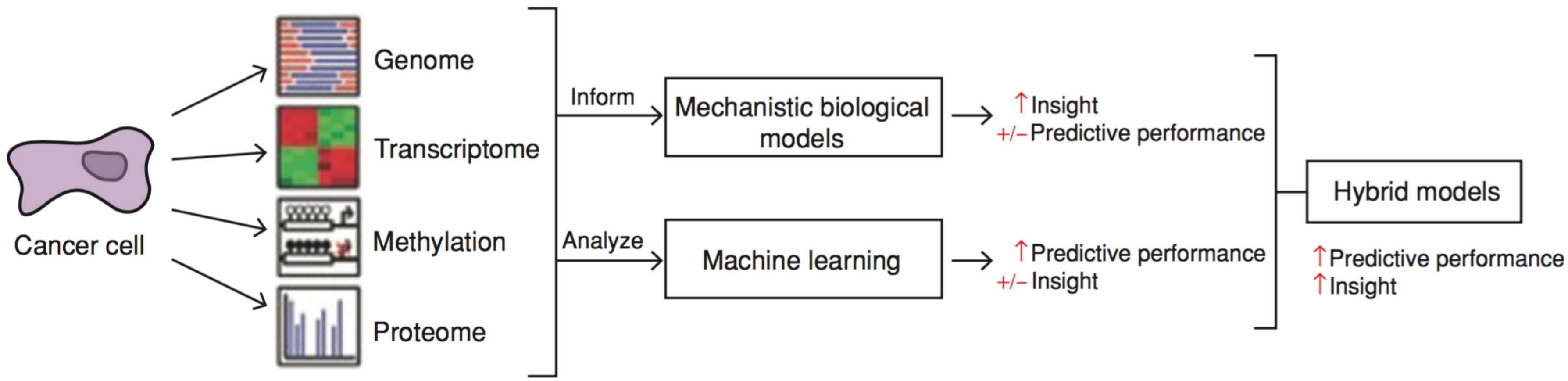


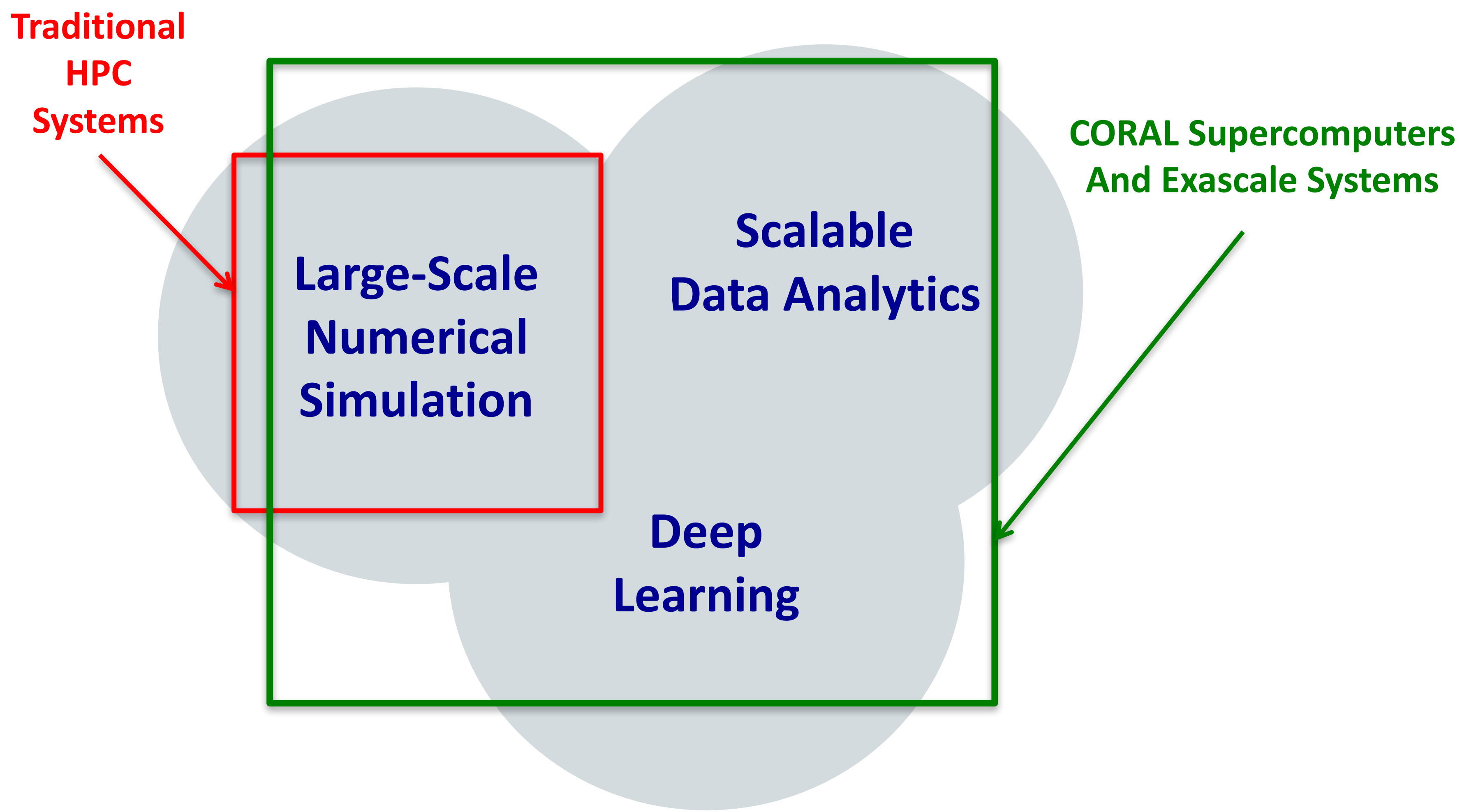
Figure 1. In two DREAM challenges, high throughput data characterizing cancer cells are used to build predictive models. Mechanistic models provide insight into the underlying biology, but do not take full advantage of the information within the data to achieve high performance. Machine learning methods are associative and extract maximum predictive value from the data, but do not always provide insight about mechanism. The future may bring hybrid models that combine the best of both approaches.

Predicting Cancer Drug Response: Advancing the DREAM

Russ B. Altman

Summary: The DREAM challenge is a community effort to assess current capabilities in systems biology. Two

Integration of Simulation, Data Analytics and Machine Learning



END