Extrapolating Gene Expression with Neural Nets

Summary and discussion of

*Gene Expression Inference with Deep Learning, Yifei et al. 2002**

For 6.874 *Computational Systems Biology* Spring 2017

Dana Gretton

2/27/17

*Yifei Chen, Yi Li, Rajiv Narayan, Aravind Subramanian, Xiaohui Xie; Gene expression inference with deep learning. Bioinformatics 2016; 32 (12):

In Brief

When extrapolating gene expression levels from limited measurements, a new deep learning system beats linear regression

Focus is on comparison with previous methods, but global view of best possible deep learning application is lacking

Capability of a standard gene expression lab assay is pushed forward, a free gain to all who use it

Work implies that, with optimized hardware and NN architecture, inference performance could be extremely good.

Contents

Contents

Example Research Problem

Example Research Problem

Lab Test:
Promote Gene

Assess Gene
Consequences

Refine Public
Databases

High-Throughput Gene
Expression Experiments

Example Research Problem

Lab Test: Promote Gene → Assess Gene Consequences → Refine Public Databases

High-Throughput Gene Expression Experiments

Gene Expression Profiling

Example Research Problem

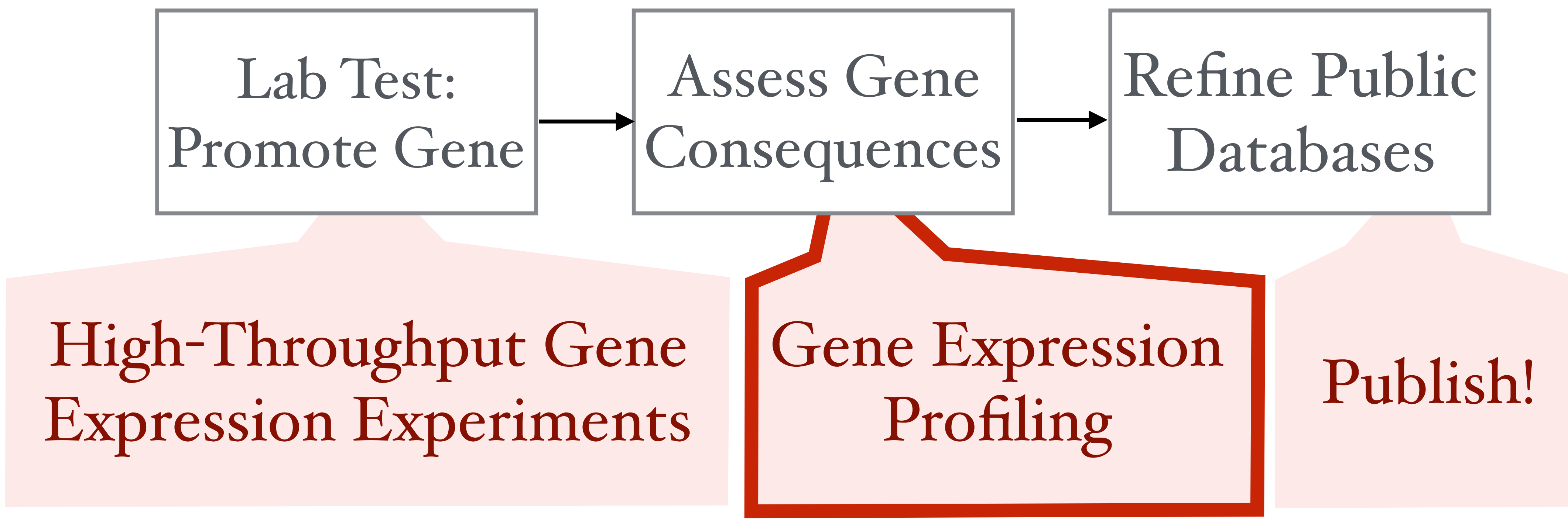
Lab Test: Promote Gene → Assess Gene Consequences → Refine Public Databases

High-Throughput Gene Expression Experiments

Gene Expression Profiling

Publish!

Example Research Problem



Gene Expression Profiling

- Too expensive to profile all gene expression levels
- Opportunity: most genes are highly correlated
  - *A full gene profile is extremely overspecified*
- Complication: gene interplay can be highly irregular

Authors contribute:

...their best effort to extract information about target genes from measurements of a few landmark genes.

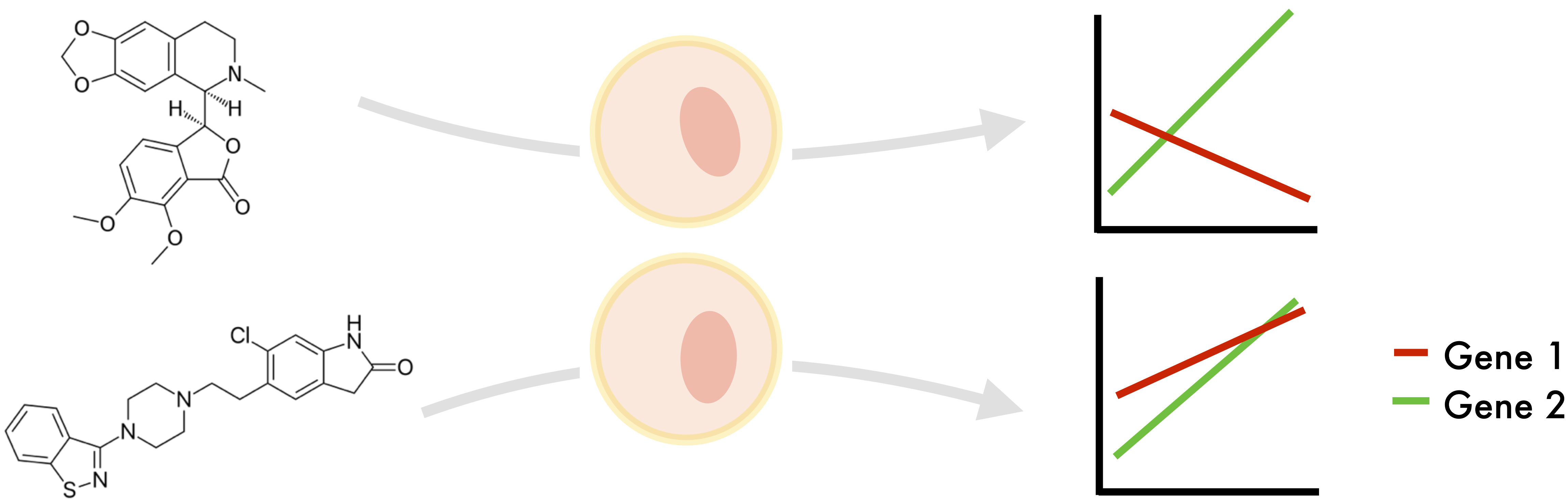
They aim to cut costs and experiment complexity for gene expression research.

They are successful in beating the state of the art in gene profile inference.

Their system is compatible with existing protocols, and its output is available to researchers who use those protocols.

The previous state of the art

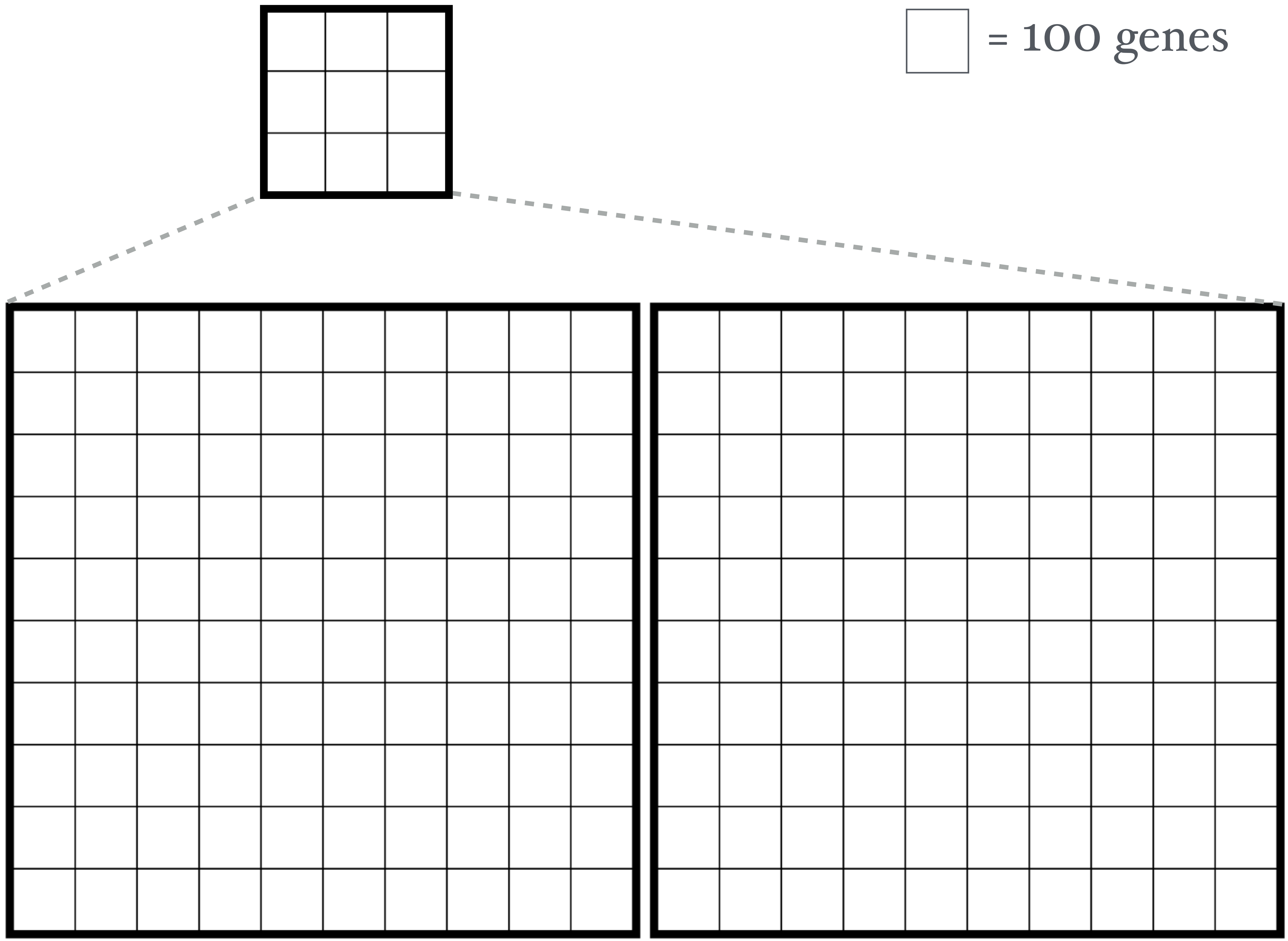
Gene "Connectivity Map" can be inferred by slightly perturbing cells and seeing how genes' expression levels vary *together*

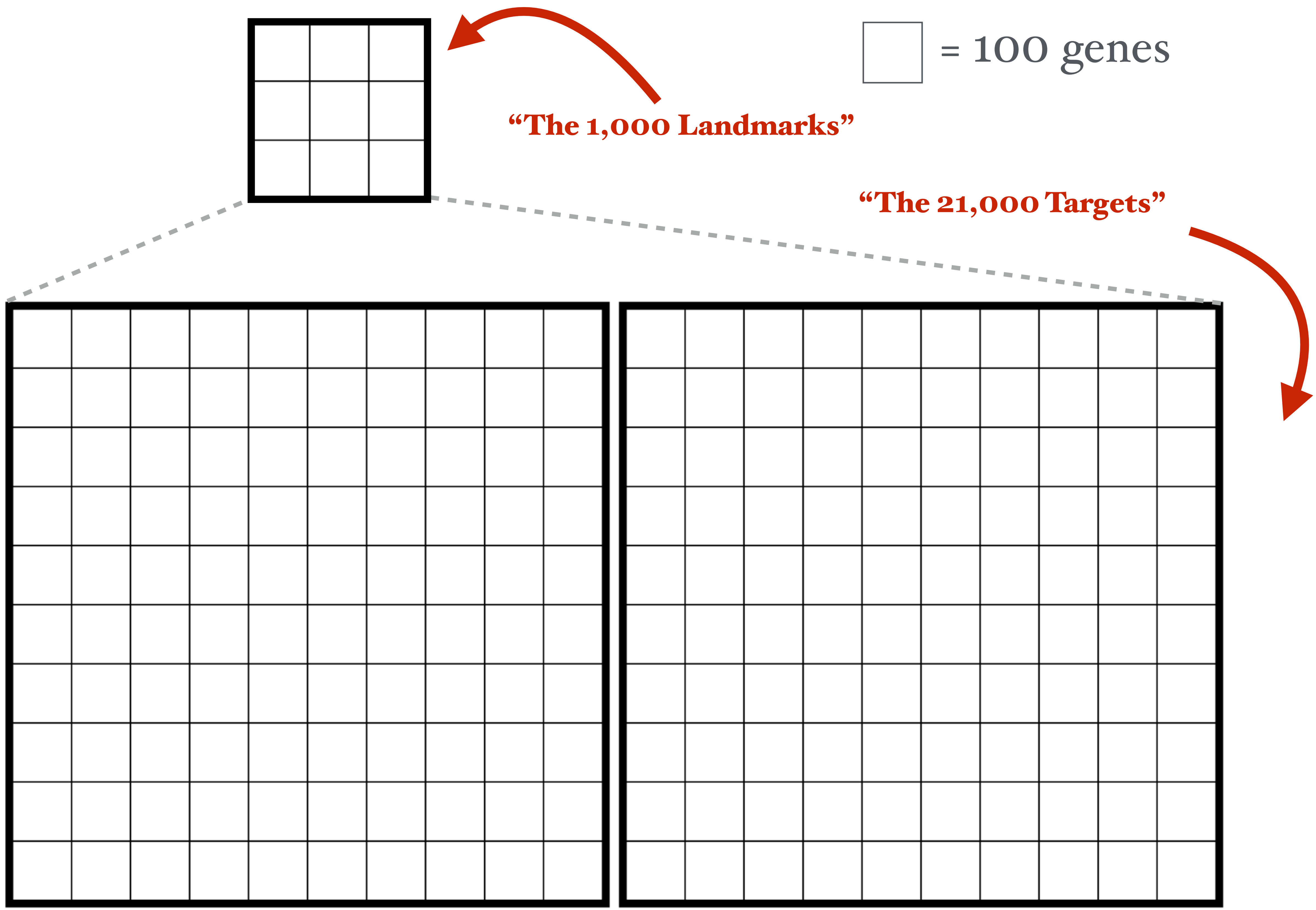


The previous state of the art

HMS LINCS

Linear Regression on 1000 target genes can account for ~80% of the “information” in the (V1.0) perturbative gene Connectivity Map (CMap)


= 100 genes



= 100 genes
"The 1,000 Landmarks"
"The 21,000 Targets"

Two ways to measure 1000 genes

Optical assay

Tag with fluorescent antibodies and measure brightness

RNA-seq

Sequence all RNA, align, and count duplicates

Two ways to measure 1000 genes

Optical assay

Tag with fluorescent antibodies and measure brightness

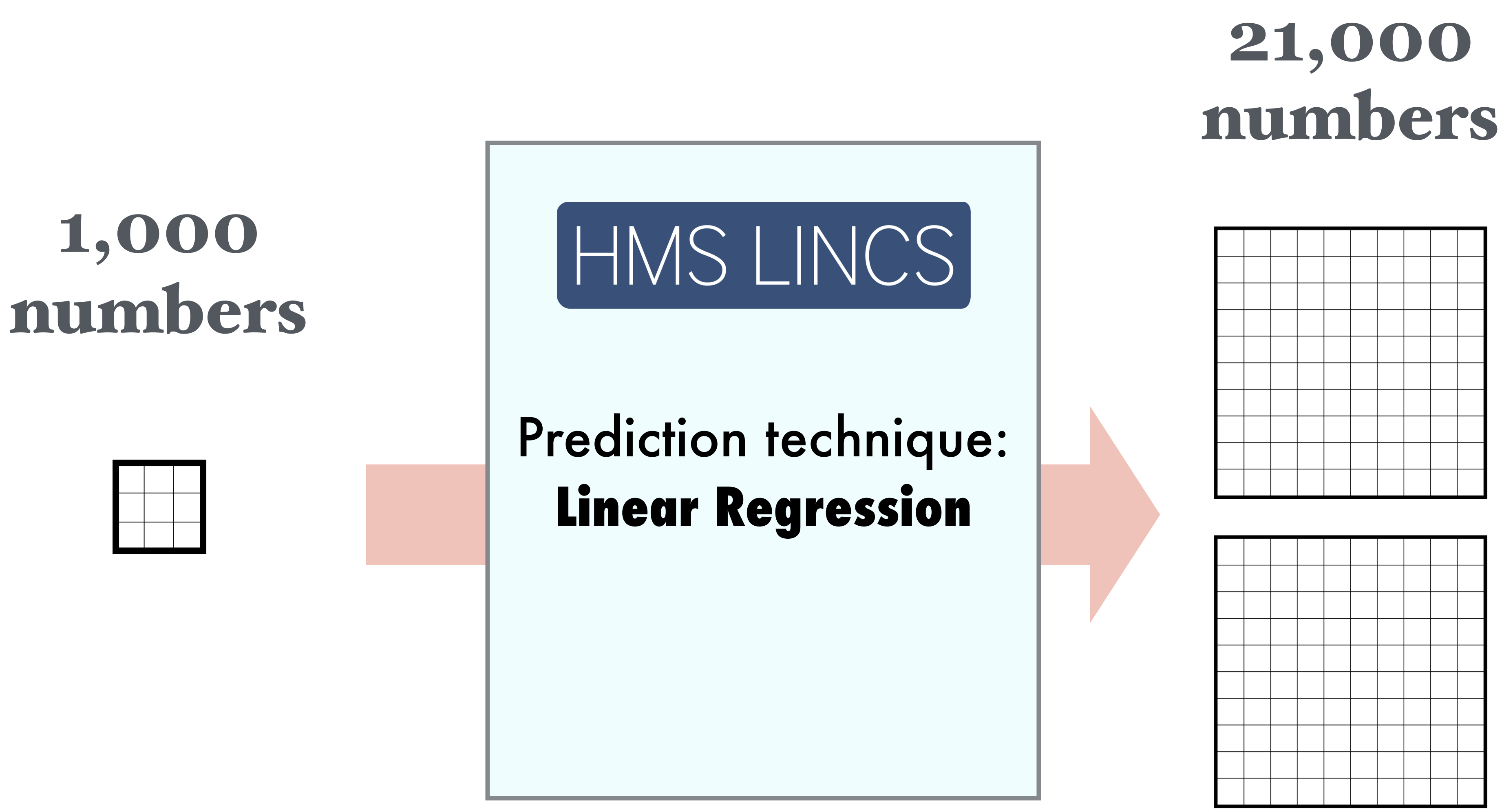
L1000 Standard Luminex Assay ($5)

RNA-seq

Sequence all RNA, align, and count duplicates

Framed as a large-scale machine learning problem

Output dimension is much higher than input dimension



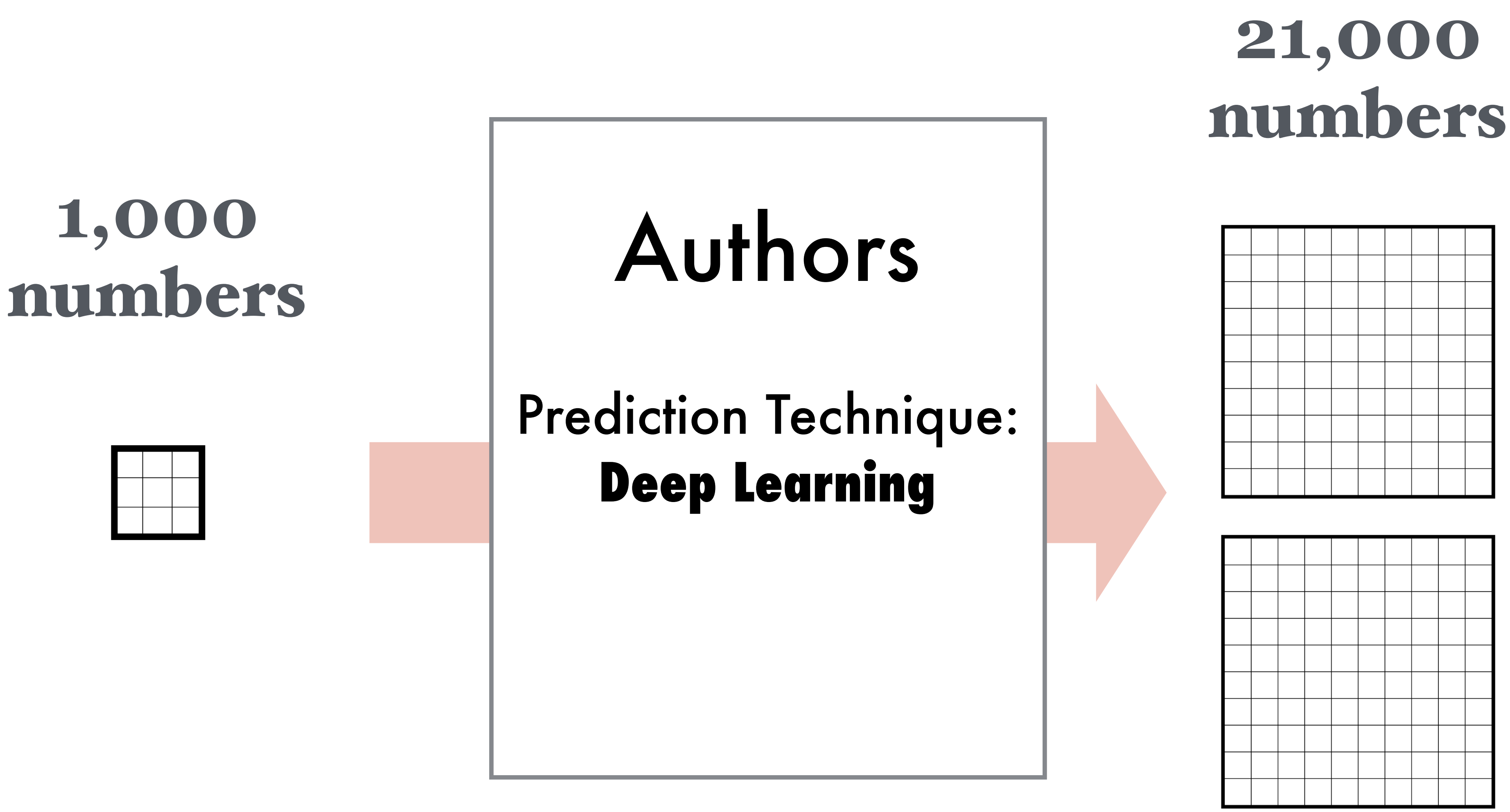
Authors propose:

Since the correlations between gene expressions can be fundamentally nonlinear, a linear regression system is critically limited

An artificial neural net can model high-dimensional nonlinear functions, so it is an attractive option worth exploring

Contents

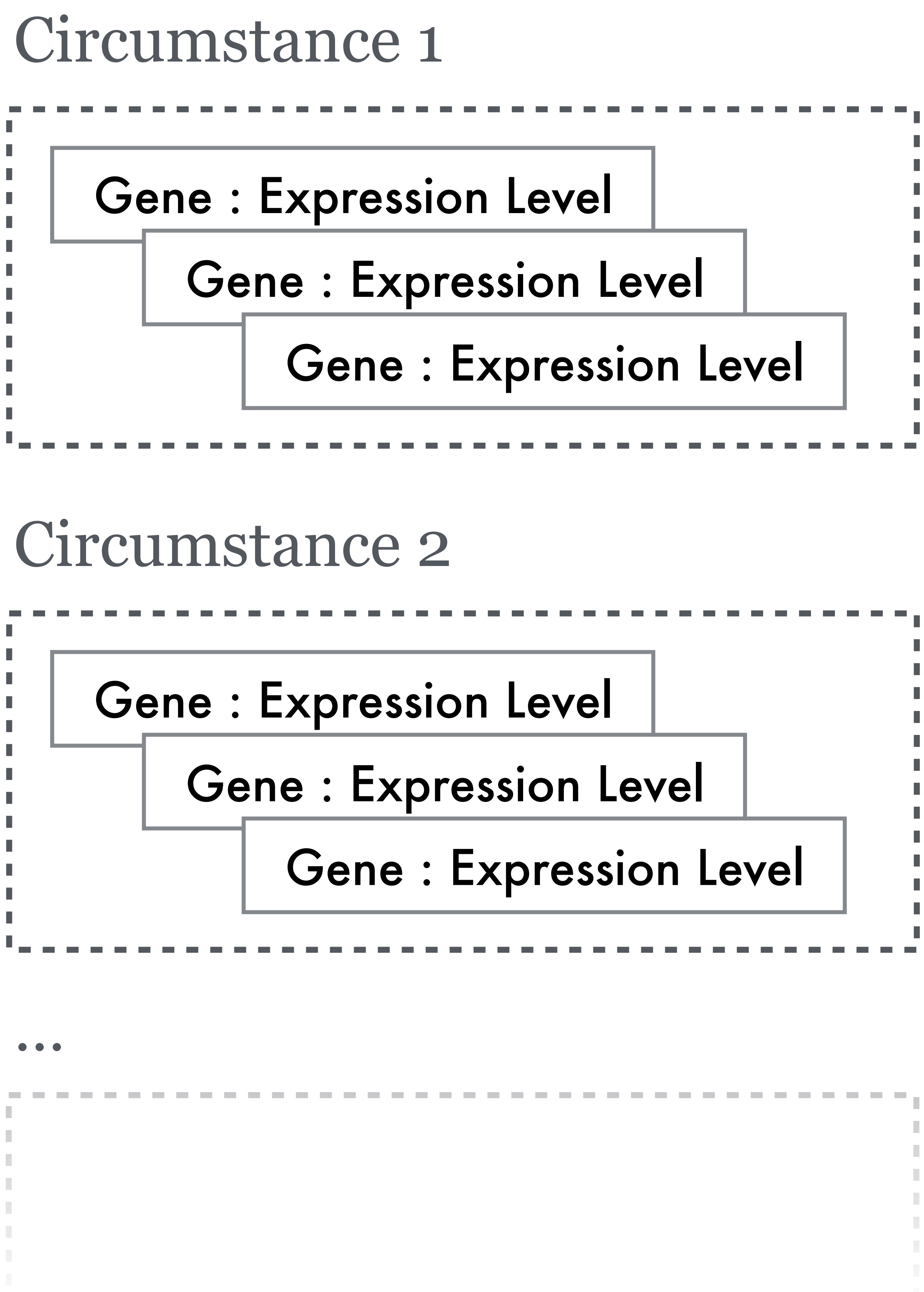
Contents


1,000
numbers
Authors
Prediction Technique:
Deep Learning
21,000
numbers

The Project

- Data preparation
- Architecture
- Performance
- Comparison

Data Preparation


Circumstance 1
Gene : Expression Level
Gene : Expression Level
Gene : Expression Level
Circumstance 2
Gene : Expression Level
Gene : Expression Level
Gene : Expression Level
...

Two ways to measure 1000 genes

Optical assay	RNA-seq
All training data All validation data Test data	Validation data Test data Used to exercise cross-platform transfer

Optical assay

Format: Linear normalized number 4-15

Gene Expression Omnibus (GEO)

111,000 circumstances

RNA-seq

Format: Reads Per Kilobase per Million (RPKM)

GTEx

3000 circumstances

1000 Genomes

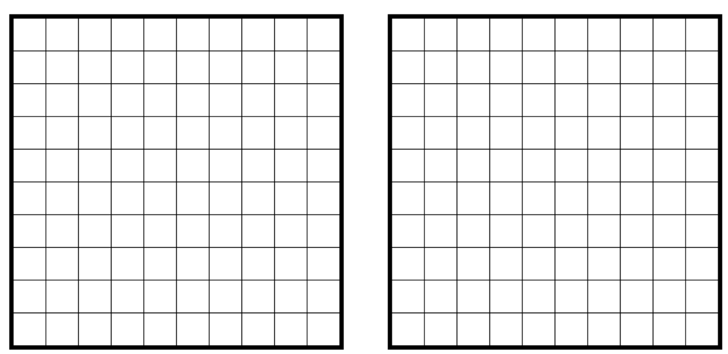
462 circumstances

Optical assay

Format: Linear normalized number 4-15

Gene Expression Omnibus (GEO)

111,000 circumstances



Individual human genes

RNA-seq

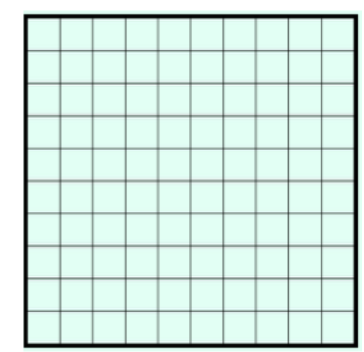
Format: Reads Per Kilobase per Million (RPKM)

GTEx

3000 circumstances

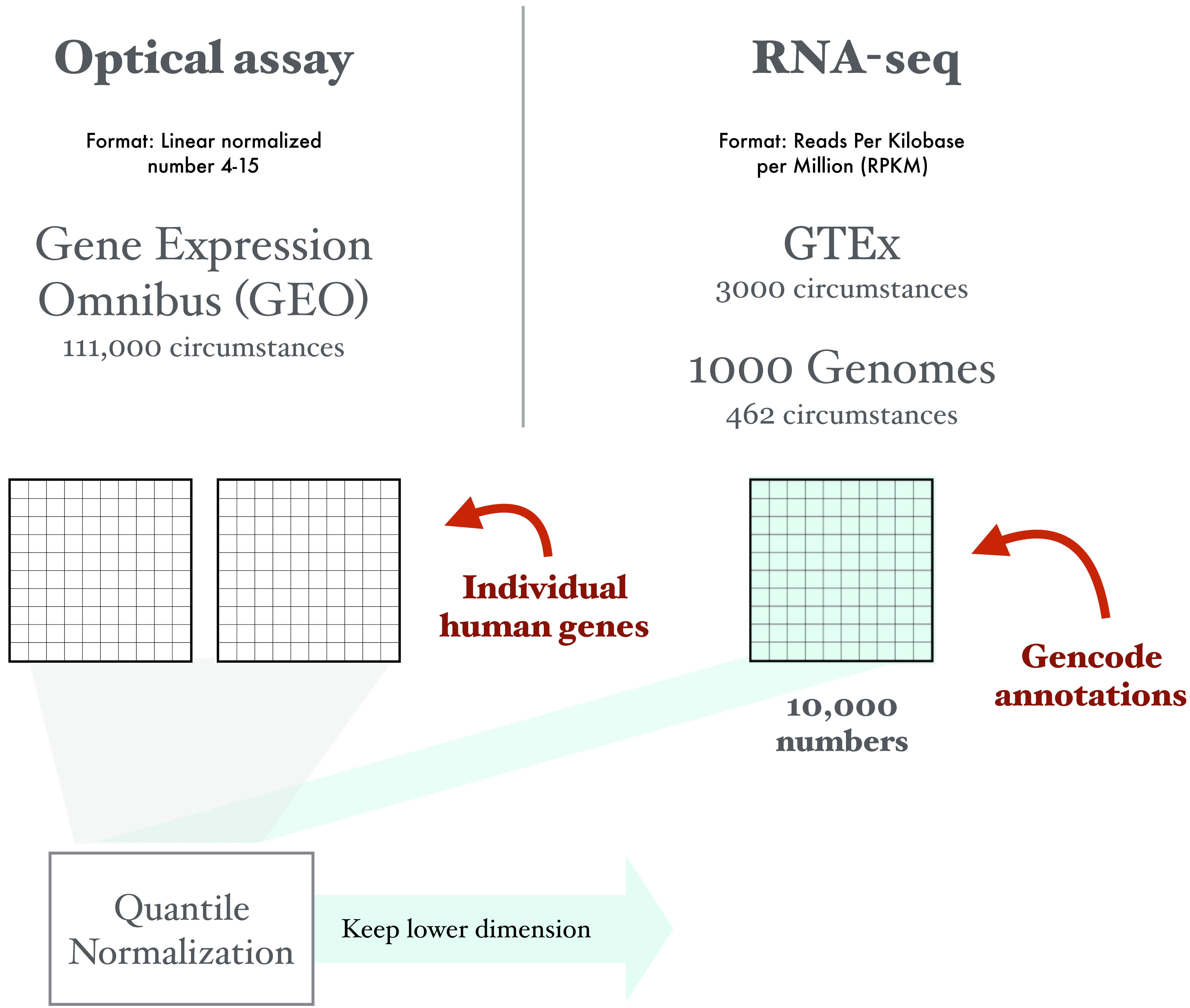
1000 Genomes

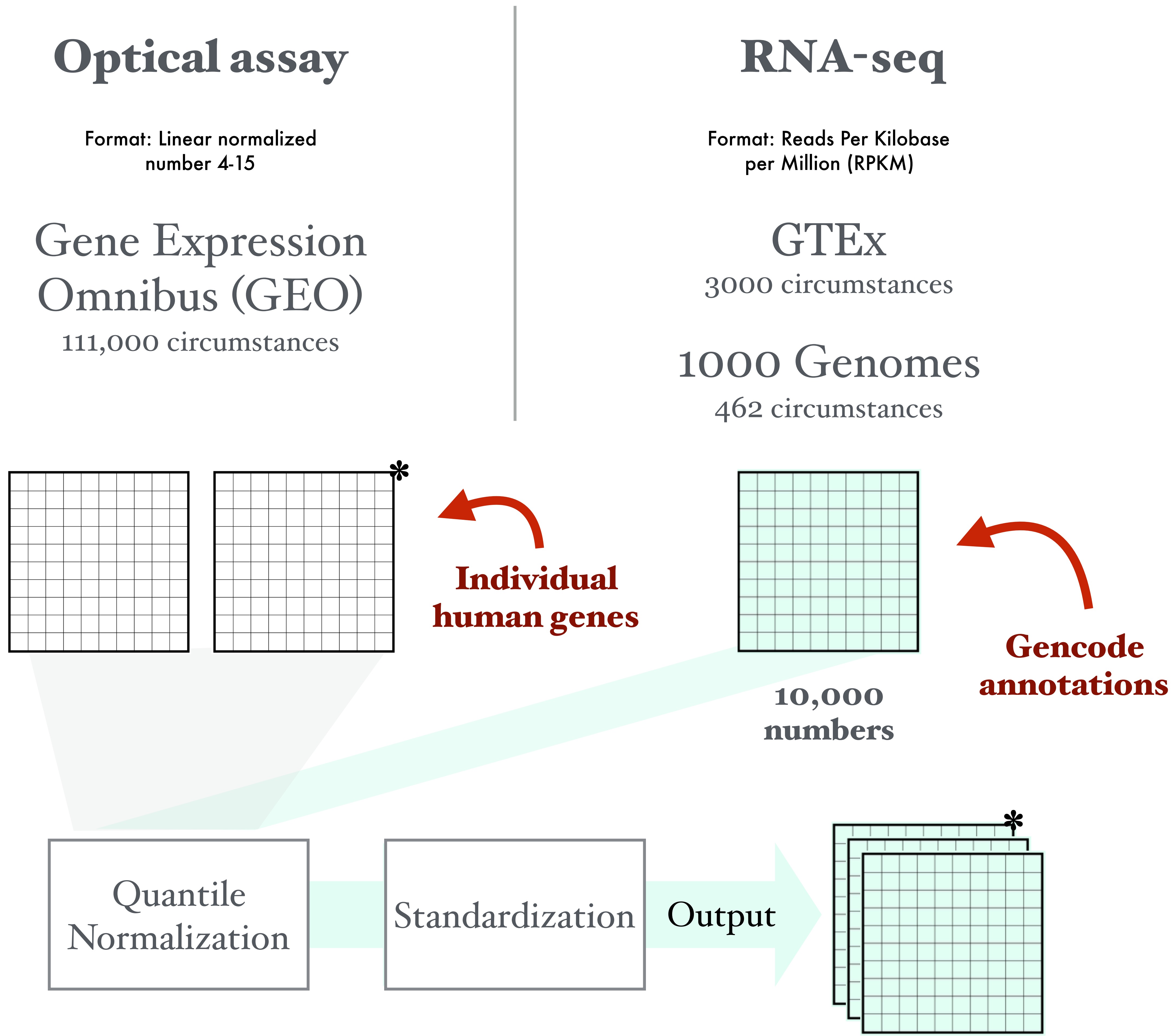
462 circumstances



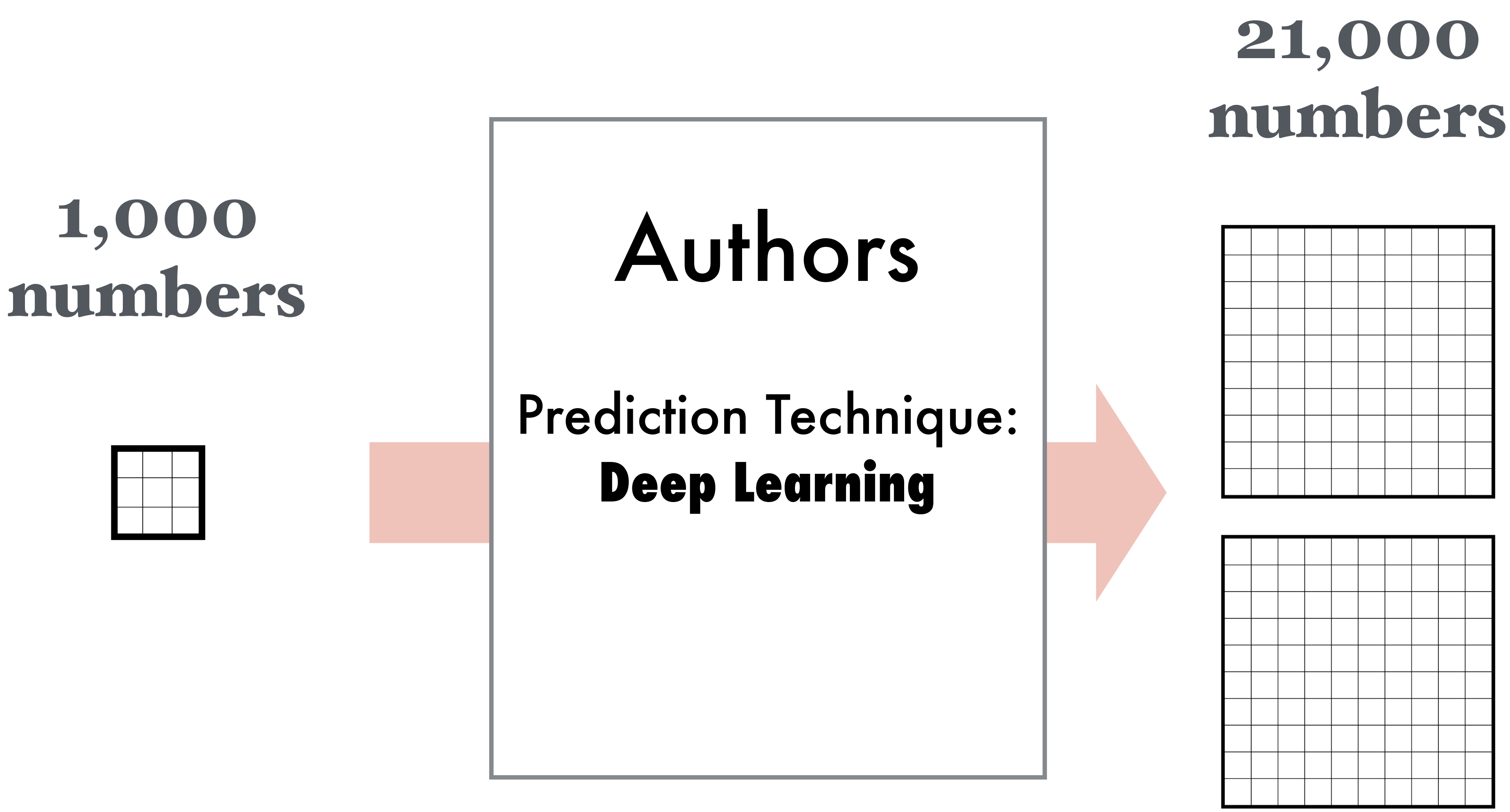
10,000 numbers

Gencode annotations

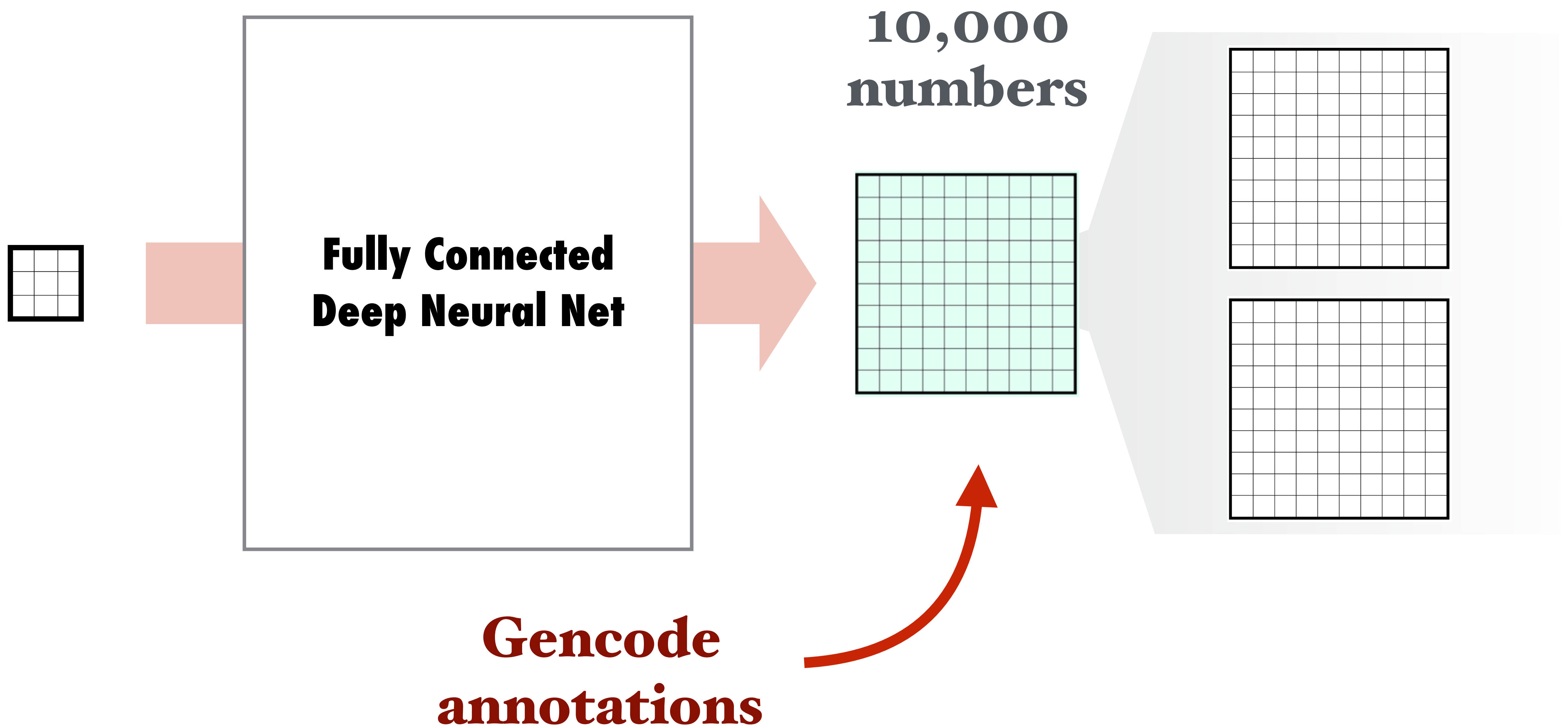

Optical assay
Format: Linear normalized number 4-15
Gene Expression Omnibus (GEO)
111,000 circumstances
RNA-seq
Format: Reads Per Kilobase per Million (RPKM)
GTEx
3000 circumstances
1000 Genomes
462 circumstances
Individual human genes
Gencode annotations
10,000 numbers
Quantile Normalization
Keep lower dimension


Optical assay
Format: Linear normalized number 4-15
Gene Expression Omnibus (GEO)
111,000 circumstances
RNA-seq
Format: Reads Per Kilobase per Million (RPKM)
GTEx
3000 circumstances
1000 Genomes
462 circumstances
*
Individual human genes
10,000 numbers
Gencode annotations
Quantile Normalization
Standardization
Output
*

Architecture

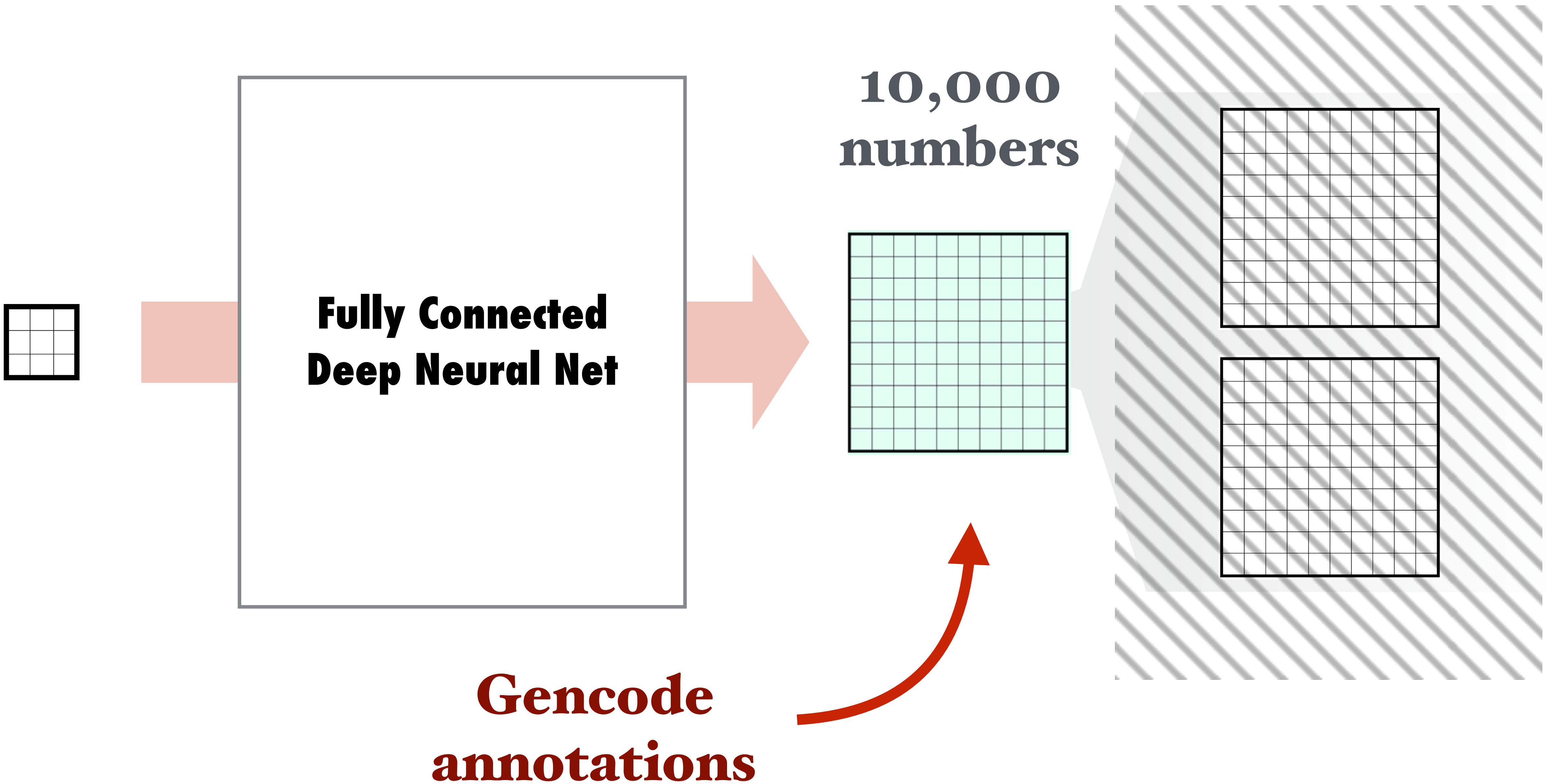

1,000 numbers
Authors
Prediction Technique:
Deep Learning
21,000 numbers

Authors

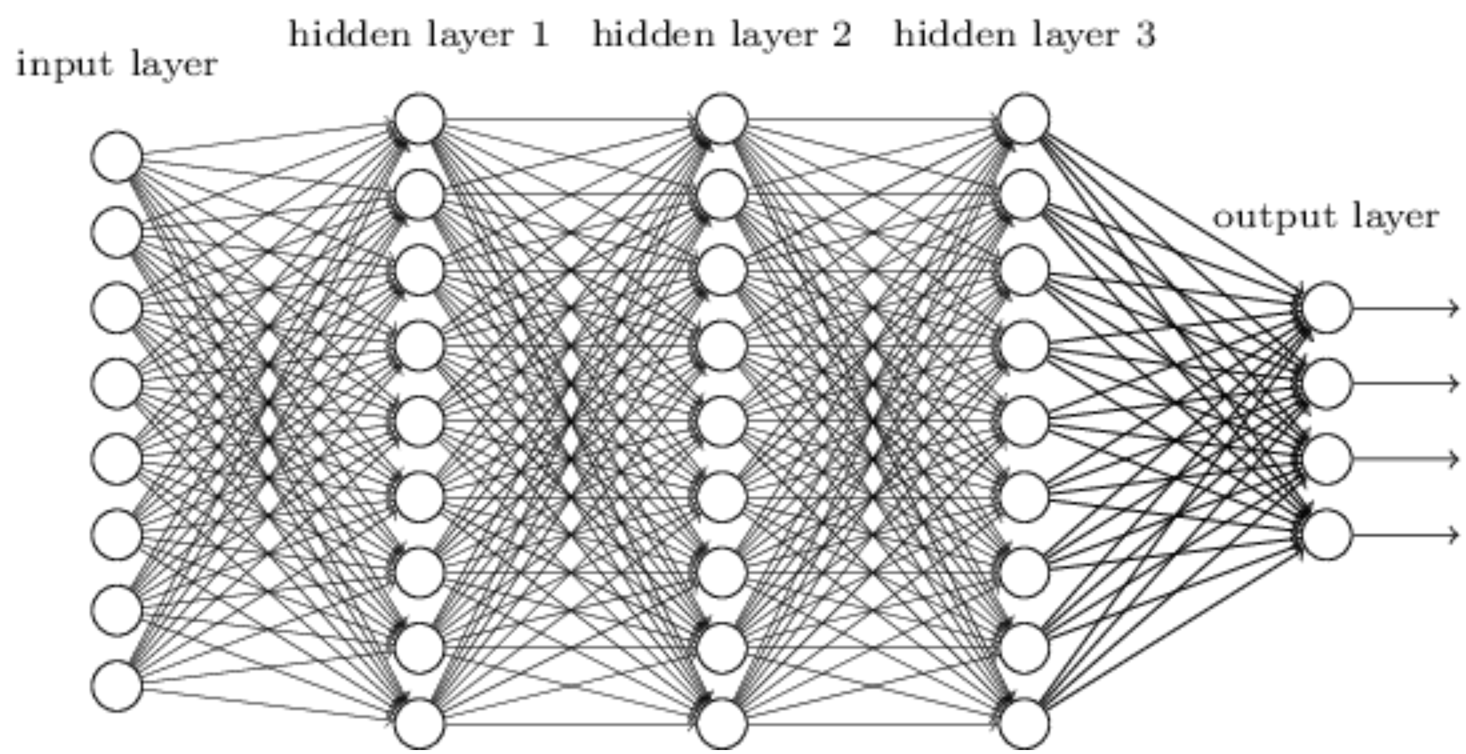


Fully Connected
Deep Neural Net
10,000
numbers
Gencode
annotations

Authors


Fully Connected
Deep Neural Net
10,000
numbers
Gencode
annotations

D-GEX



1,000 outputs

10,000 outputs

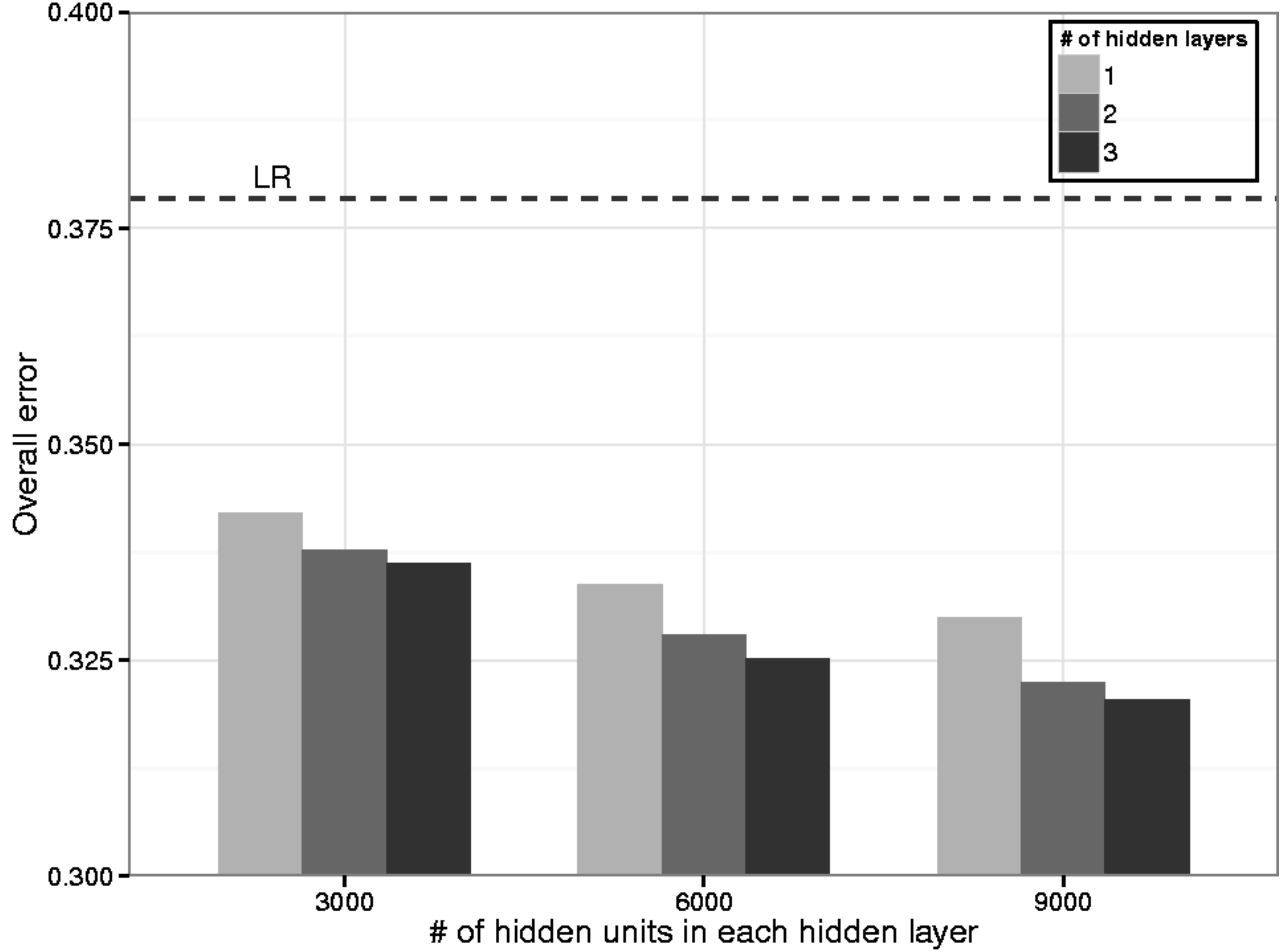
[9000 hidden units] x 2 or 3

D-GEX Implementation Details

- Dropout between all hidden layers present
  - tried at several rates
- Momentum training method
- Adaptive learning rate decay
- Split network. Assumption:

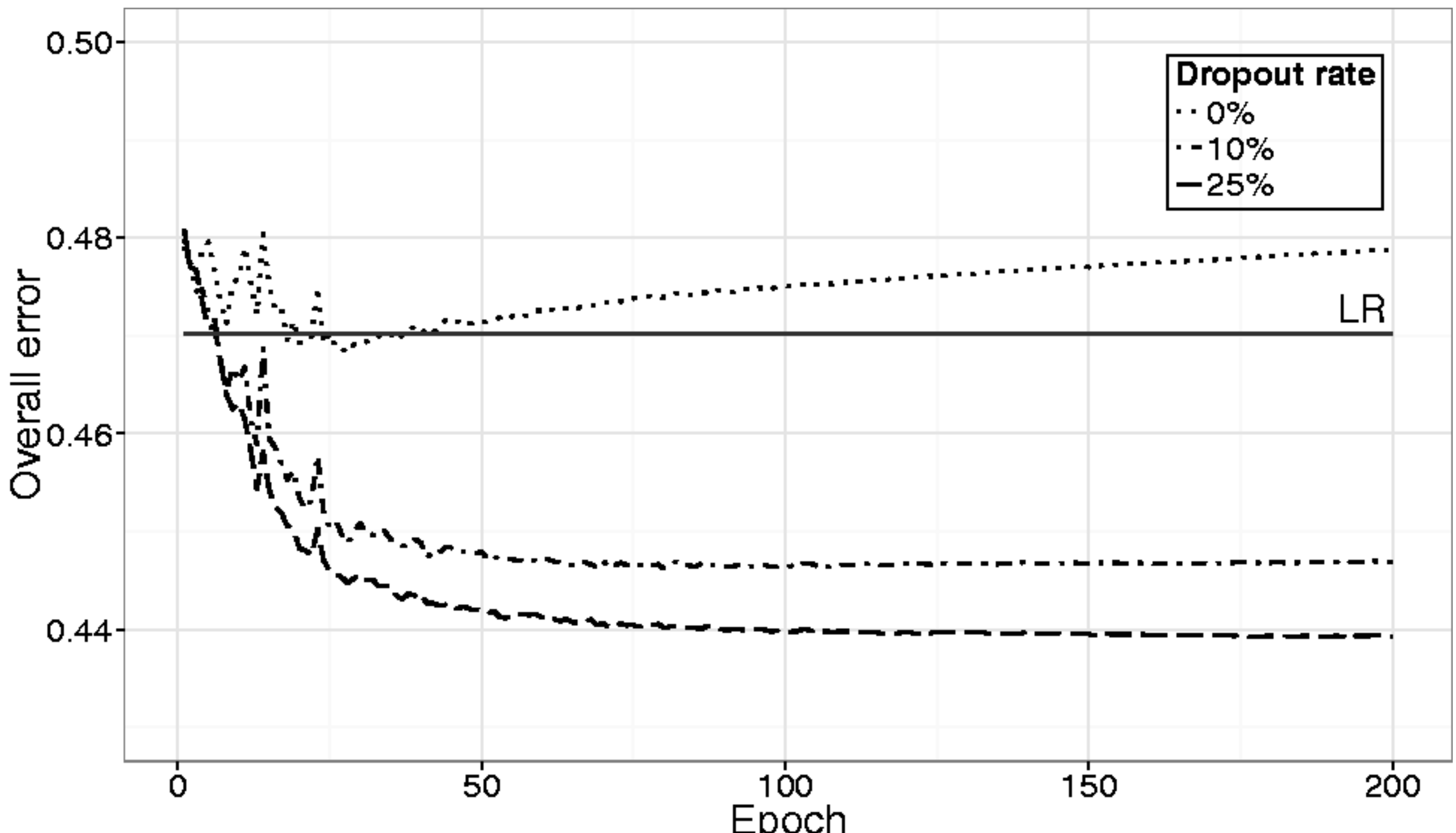
> Partitioning the network neither greatly degraded nor improved performance

- Model selection



0.400
0.375
0.350
0.325
0.300
Overall error
LR
# of hidden layers
1
2
3
3000
6000
9000
# of hidden units in each hidden layer

Training on GEO, error on GTEx



Performance

In what?

- Performance in using the 1000 landmark genes in each data set to predict the rest
- Metric: Summed mean squared output error

$$\mathrm{L} = \sum_{t=1}^{T} \left[\frac{1}{N} \sum_{i=1}^{N} \left(y_{i(t)} - \widehat{y}_{i(t)} \right)^2 \right]$$

- Interpreting success: assumptions come into play

In what?

- After normalization: results are relative
- Assuming:

> If LR does well on "gen-coded" data, and D-GEX beats it, then D-GEX beats LR on the raw GEO data

- Even then, only meaningful result is in comparison

Comparison

Other models to compare

(Regularized) Linear Regression

K-Nearest Neighbors (KNN) clustering algorithm

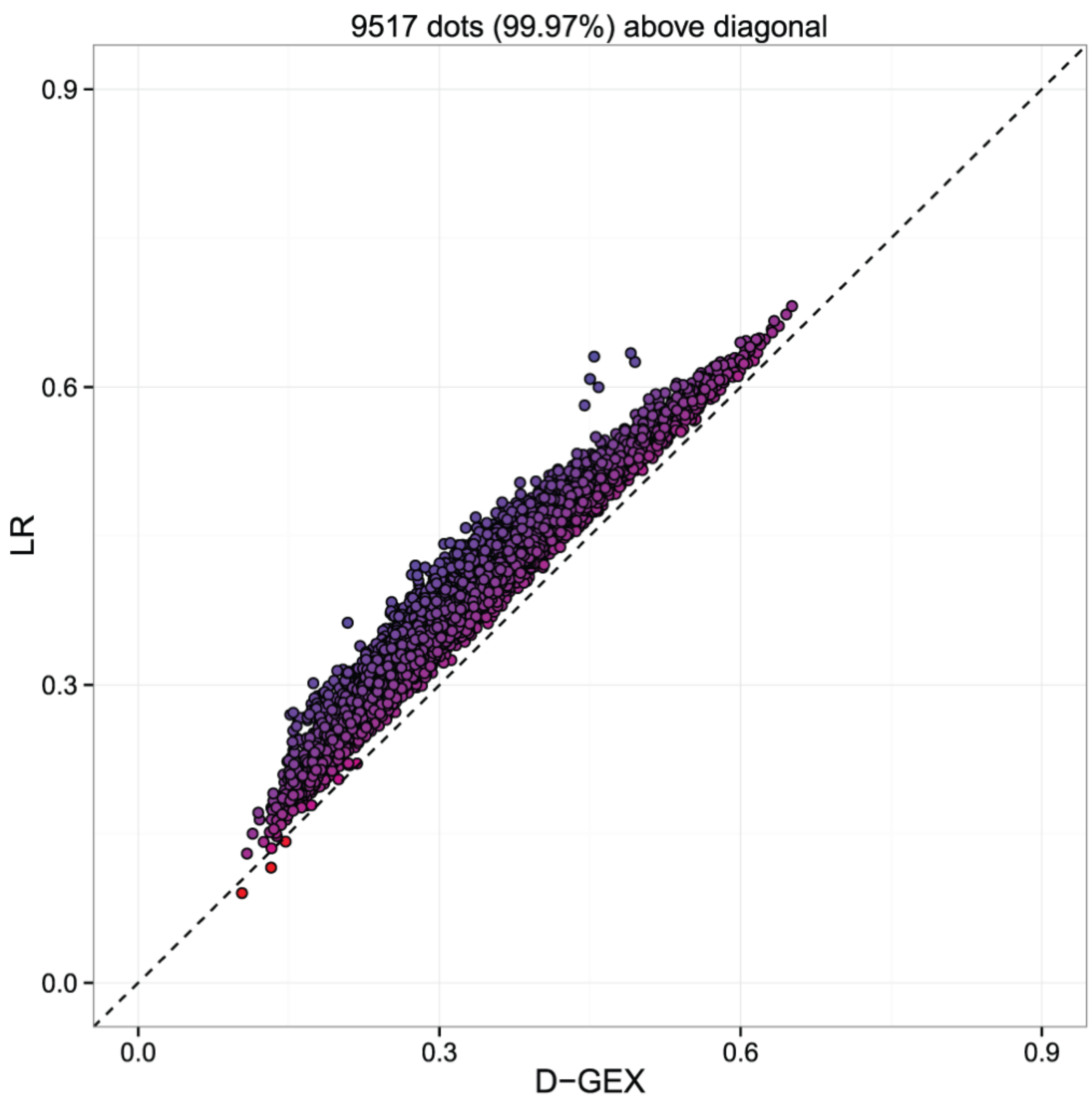
We'll compare: prediction accuracy and transfer accuracy

GEO test set error

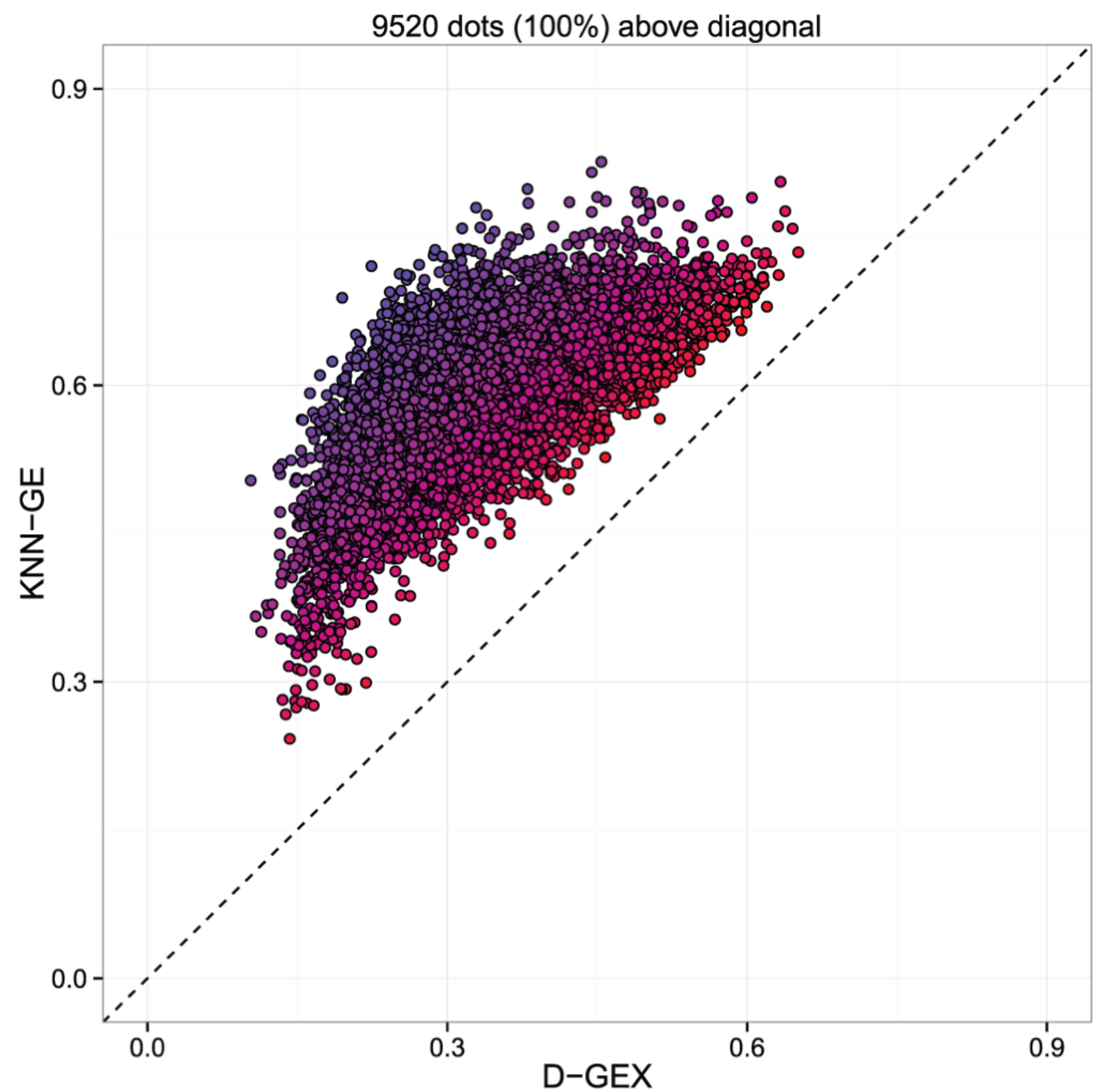
Dropout: 10%

	Number of hidden units		
	3000	6000	9000
Number of hidden layers			
1	0.3421 ± 0.0858	0.3337 ± 0.0869	0.3300 ± 0.0874
2	0.3377 ± 0.0854	0.3280 ± 0.0869	0.3224 ± 0.0879
3	0.3362 ± 0.0850	0.3252 ± 0.0868	**0.3204 ± 0.0879**
LR		0.3784 ± 0.0851	
LR-L1		0.3782 ± 0.0844	
LR-L2		0.3784 ± 0.0851	
KNN-GE		0.5866 ± 0.0698	

GEO test set error



(a)



(b)

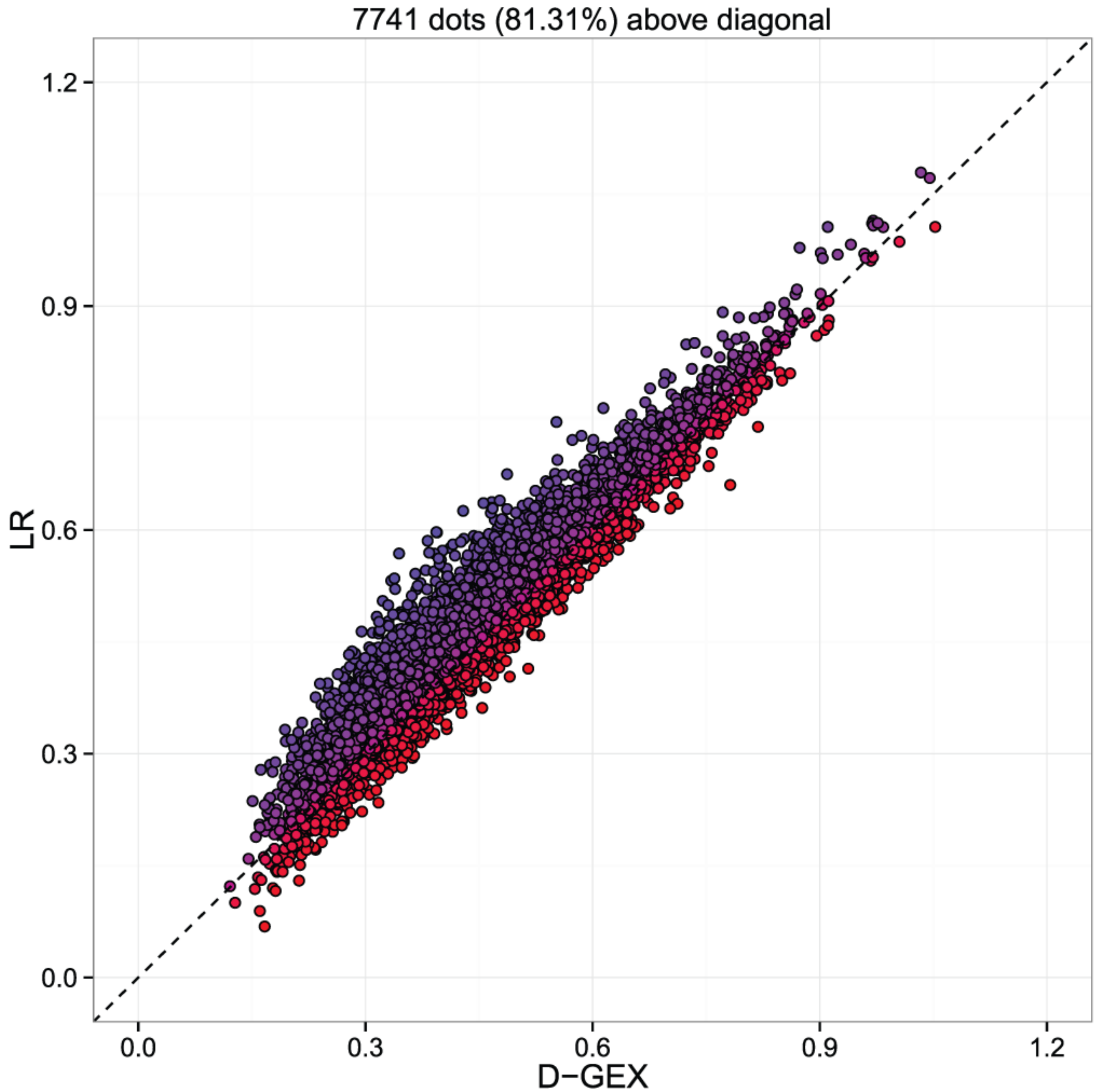
GTEx test set (transfer) error

Dropout: **25%**

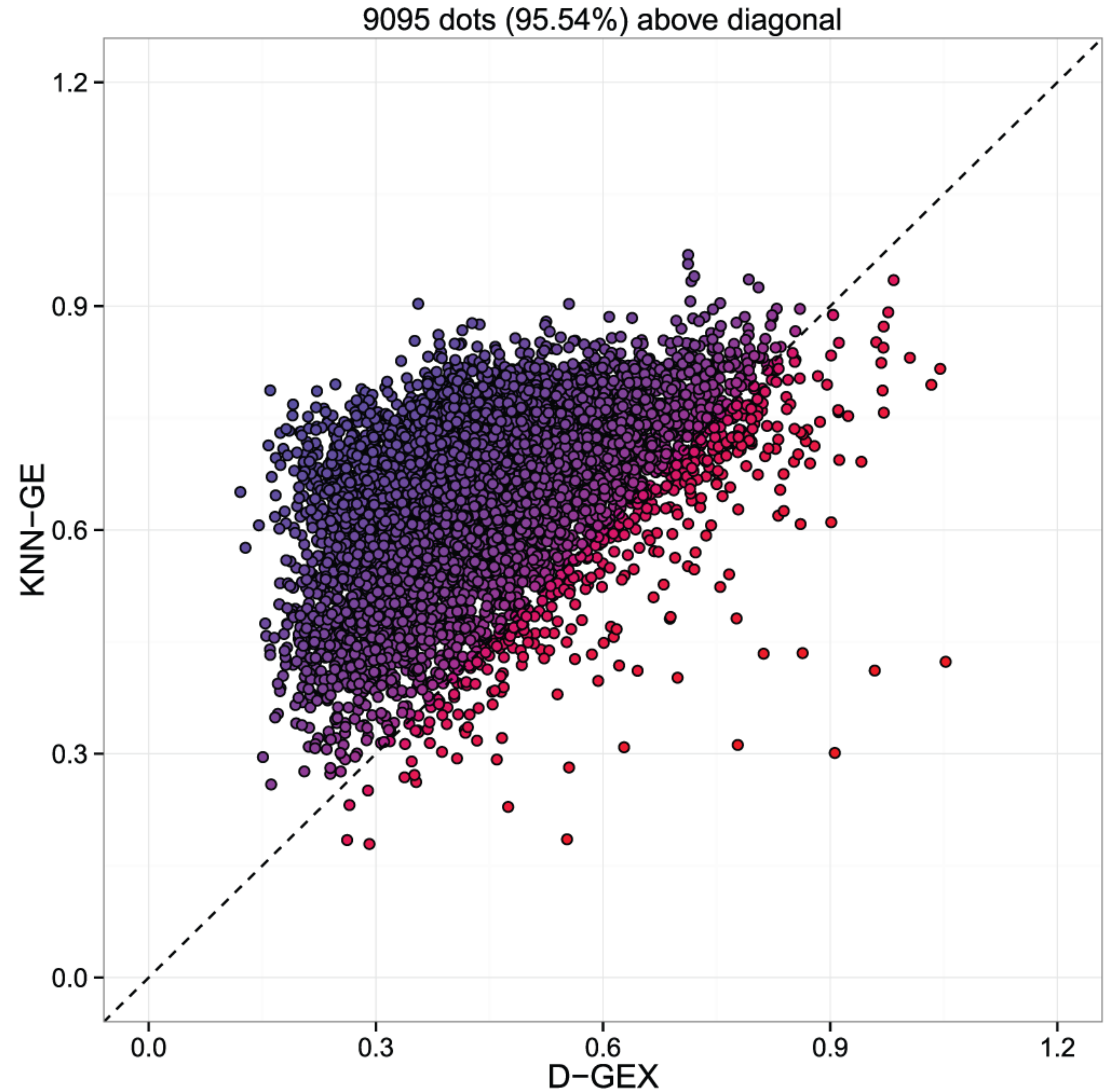
	Number of hidden units		
	3000	6000	9000
Number of hidden layers			
1	0.4507 ± 0.1231	0.4428 ± 0.1246	0.4394 ± 0.1253
2	0.4586 ± 0.1194	0.4446 ± 0.1226	**0.4393** ± **0.1239**
3	0.5160 ± 0.1157	0.4595 ± 0.1186	0.4492 ± 0.1211
LR		0.4702 ± 0.1234	
LR-L1		0.5667 ± 0.1271	
LR-L2		0.4702 ± 0.1234	
KNN-GE		0.6520 ± 0.0982	

Regularization and lower complexity help generalize

GTEx test set (transfer) error



(a)



(b)

Contents

Contents

Claims

- Deep learning system D-GEX makes better predictions on L1000 than other options
- D-GEX is highly performant across gene measurement platforms
- The success of D-GEX is attributable to its ability to capture nonlinearities.

- **Deep learning system D-GEX makes better predictions on L1000 than other options**

  - The L1000 assay was defined in terms of its success in capturing genome-wide gene expression by *linear regression*
  - D-GEX beats out linear regression
  - KNN failed

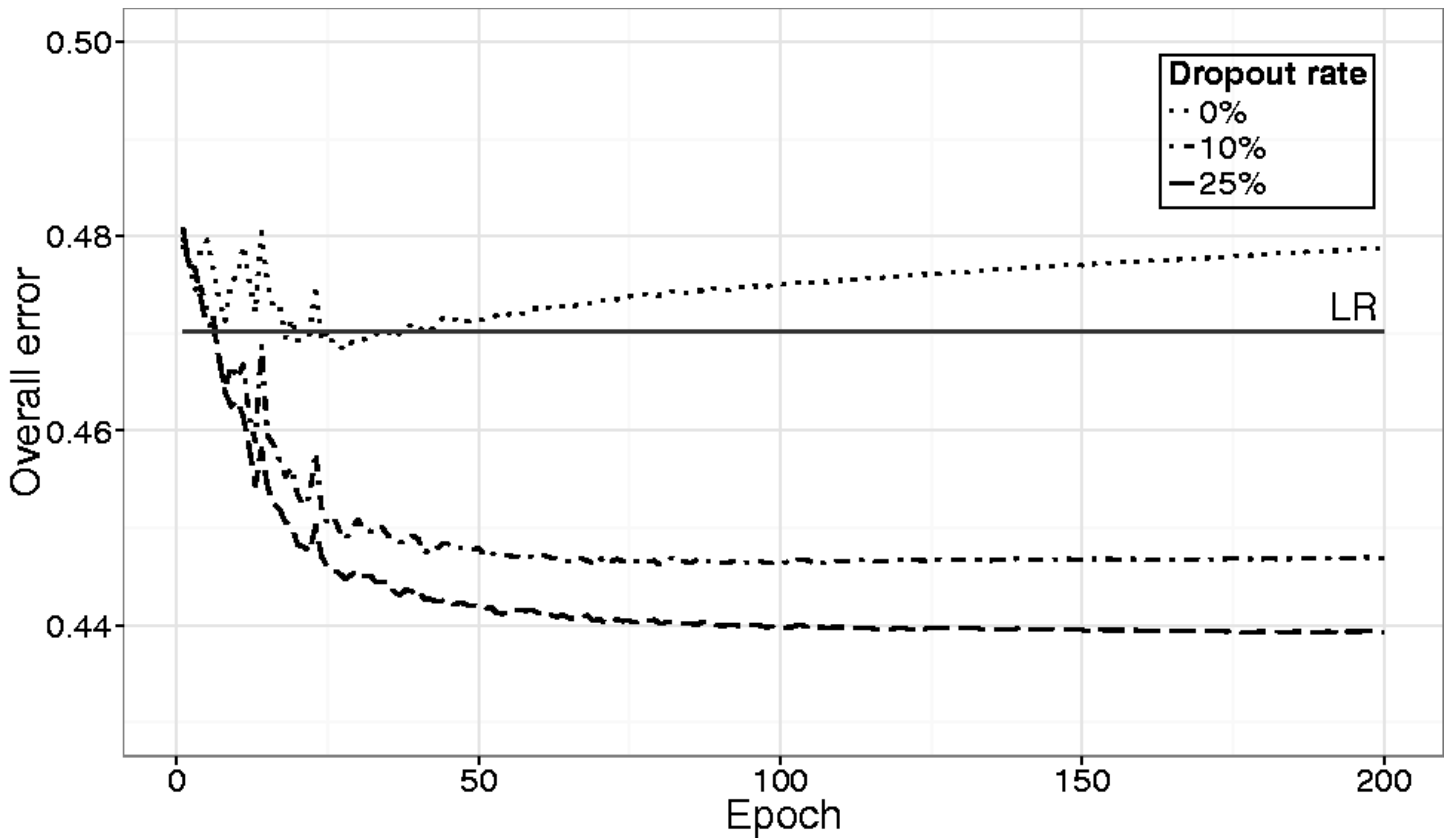
Well supported

- **D-GEX is highly performant across gene measurement platforms**

- Comparable transfer performance to LR
- Limited KNN success: Platforms are different in ways that either aren't statistically compatible (very possible) or aren't captured by D-GEX

Somewhat supported

Training on GEO, error on GTEx



- **The success of D-GEX is attributable to its ability to capture nonlinearities.**
    - Any improvement will need to capture nonlinearities because LR exploits all linear information available
    - Support: regularization on LR made no improvement
    - Linear regression on last hidden layer performs better

Well supported

(vacant?)

Contents

Contents

Then

(Time of writing, June 2016)

Natural extension of available data

- Sensible to exploit ubiquitous L1000 assay data with new technology
- Updated, (probably) more accurate database made available

Now

(February 2017)

Inspiring more deep learning applications

- Cited in lists of exciting applications for "Deep Learning Models," e.g. in this[1] PhD thesis
- Viewed as a stepping stone toward deep learning's inheritance of most of big bio data in a recent bioinformatics survey[2]

[1] Chen, Lujia (2017) Deep learning models for modeling cellular transcription systems. Doctoral Dissertation, University of Pittsburgh.

[2] Seonwoo Min, Byunghan Lee, Sungroh Yoon; Deep learning in bioinformatics. Brief Bioinform 2016 bbw068. doi: 10.1093/bib/bbw068

Future

My prediction: replacing L1000

- L1000 has been an exciting early success in leveraging large bio databases
- LR is simplistic
- The 1000 landmarks are not the 1000 genes that will predict the transcriptome best given a large NN
- As Chen et al. started here, more science will be done on the internals of neural nets to discover new, complex relationships between genes