

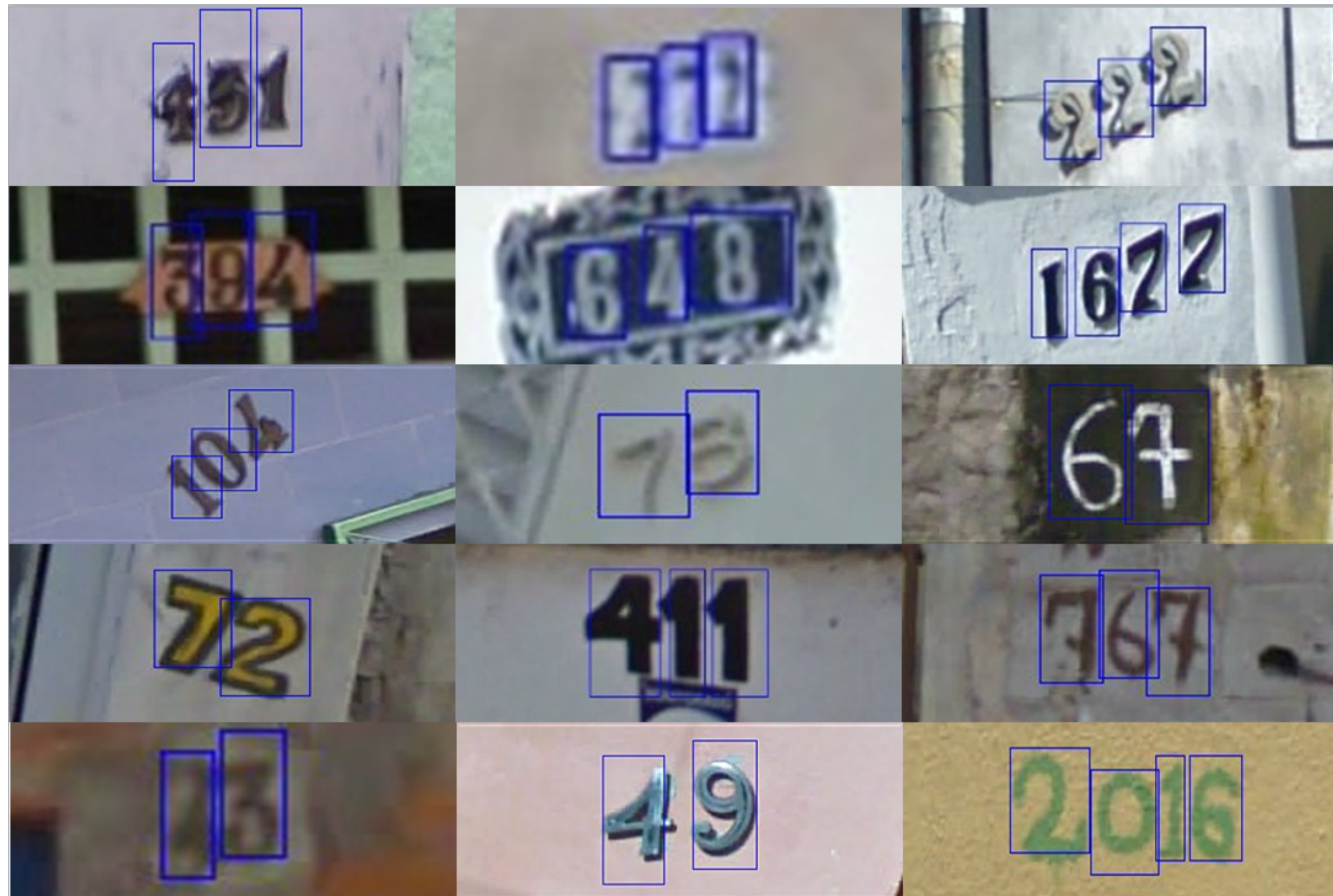
Types of Machine Learning and Model Selection

Machine Learning
CS4824/ECE4424
Bert Huang
Virginia Tech

1st Learning Setting

- Draw data set $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ from distribution \mathbb{D}
- Algorithm A learns hypothesis $h \in H$ from set H of possible hypotheses $A(D) = h$
- We measure the quality of h as the expected **loss**: $E_{(x,y) \in \mathbb{D}} [\ell(y, h(x))]$
- This quantity is known as the **risk**
- E.g., loss could be the Hamming loss $\ell_{\text{Hamming}}(a, b) = \begin{cases} 0 & \text{if } a = b \\ 1 & \text{otherwise} \end{cases}$

Example: Digit Classification




<http://ufldl.stanford.edu/housenumbers/>

Example: Airline Price Prediction

KAYAK

HOTELSFLIGHTSCARSPACKAGES

Login



Advice: **BUY** Confidence: 80%
Prices may rise within 7 days

Create a price alert

Stops

☐ nonstop

☒ 1 stop \$732

☒ 2+ stops \$736

Times

Take-off **Charlotte (CLT)**
Fri 5:00a - 2:30p

Take-off **Honolulu (HNL)**
Fri 2:30p - Sat 12:00a

Show landing times

CLT ↔ HNL

Aug 28 Friday → Aug 28 Friday

Economy cabin

1 traveler

Change

Sort by: price (low to high)

527 of 533 flights

Round-trip | Segment **NEW**

\$367 Honolulu Round Trip


[cheapoair.com/Honolulu-Cheap-Flight](#)

Book Discounted Fares Today & Save! Cheap Fares on Flights to Honolulu.
Search, Select & Save Big · We Make it Easy to Travel · Our Best Price Guarantee · 24/7 Customer Care
Winner - 2014 Customer Focused Innovations Award - CSIA

ads

\$732

US Airways



11:35a CLT → 5:30p HNL 11h 55m 1 stop (PHX)

9:05p HNL → 1:35p CLT 10h 30m 1 stop (PHX)


Select

Show details

Economy

\$732

American Airlines

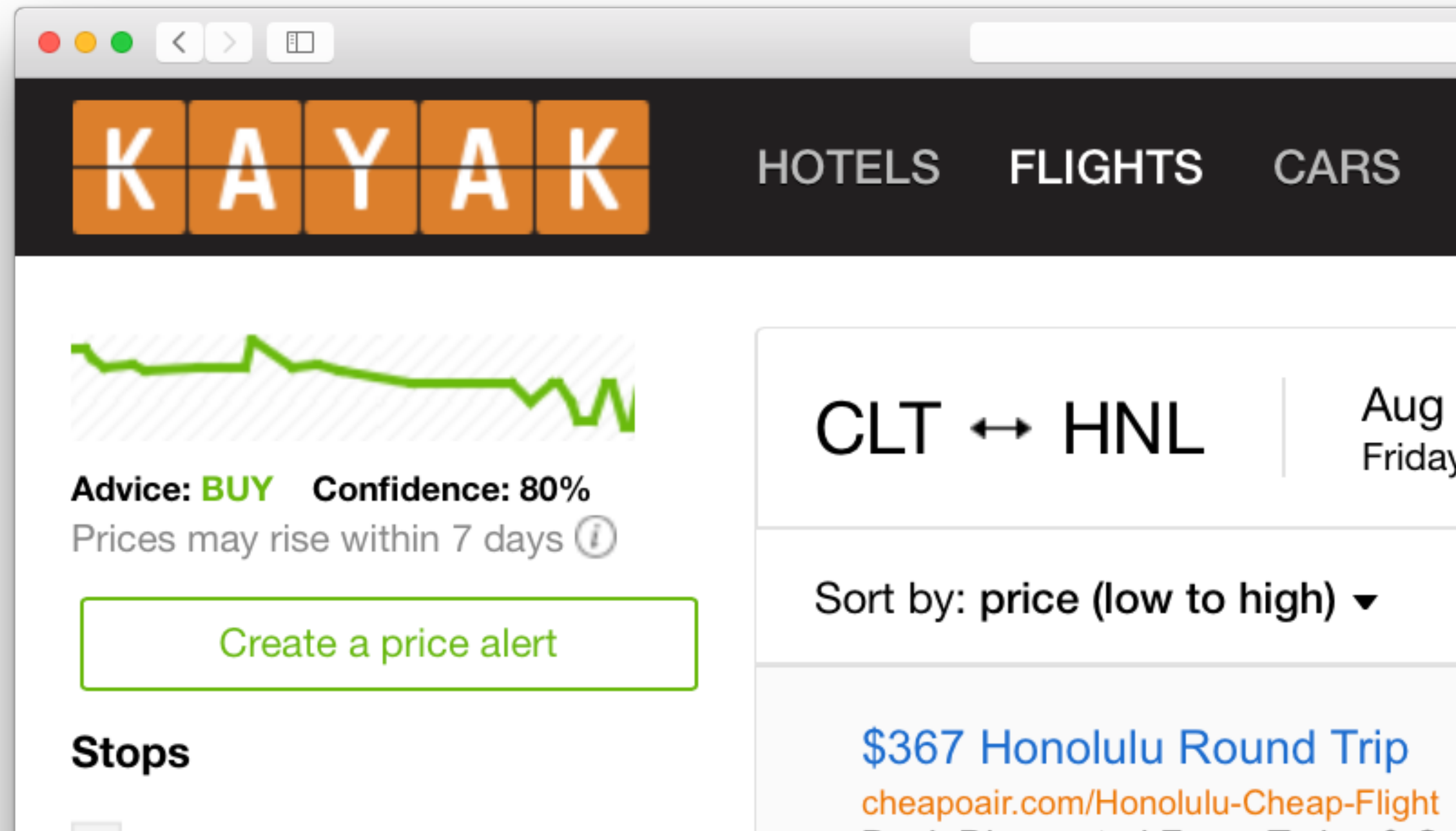


6:10a CLT → 12:22p HNL 12h 12m 1 stop (DFW)

9:05p HNL → 1:35p CLT 10h 30m 1 stop (PHX)


Select

Example: Airline Price Prediction



The screenshot shows the KAYAK website interface. At the top, the KAYAK logo is displayed in orange squares, followed by navigation links for HOTELS, FLIGHTS, and CARS. Below the logo, a green line graph shows price fluctuations. The text "Advice: **BUY** Confidence: 80%" is displayed, along with a warning "Prices may rise within 7 days" and an information icon. A green button labeled "Create a price alert" is visible. On the right side, the flight route "CLT ↔ HNL" is shown, along with the date "Aug Friday". Below this, a dropdown menu indicates "Sort by: price (low to high)". At the bottom right, the price "\$367 Honolulu Round Trip" is displayed, along with a link to "cheapoair.com/Honolulu-Cheap-Flight".

KAYAK HOTELS FLIGHTS CARS



Advice: BUY Confidence: 80%
Prices may rise within 7 days ⓘ

Create a price alert

Stops

CLT ↔ HNL | Aug Friday

Sort by: price (low to high) ▼

\$367 Honolulu Round Trip
cheapoair.com/Honolulu-Cheap-Flight

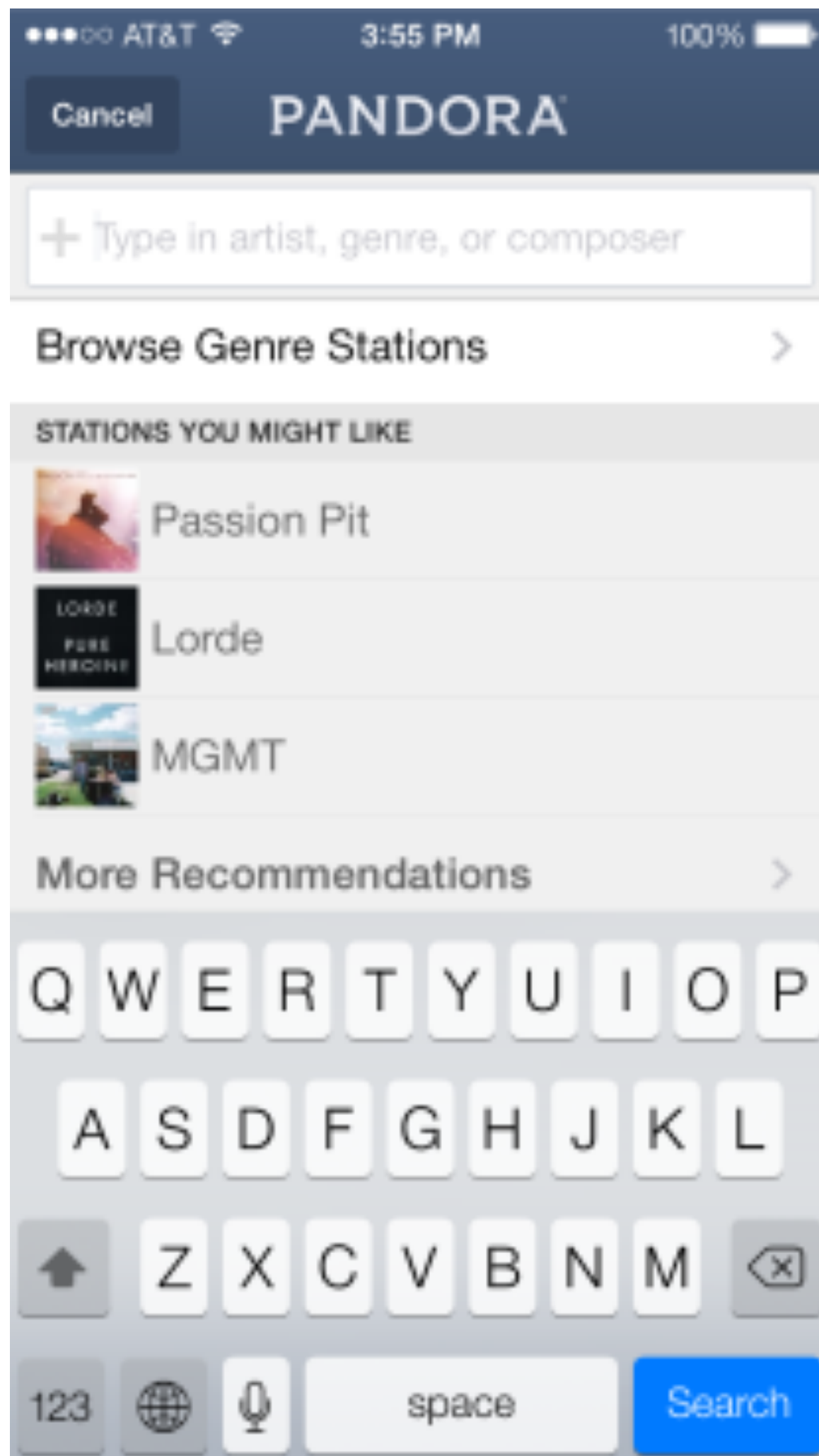
Batch Supervised Learning

- Draw data set $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ from distribution \mathbb{D}
- Algorithm A learns hypothesis $h \in H$ from set H of possible hypotheses $A(D) = h$
- We measure the quality of h as the expected **loss**: $E_{(x,y) \in \mathbb{D}} [\ell(y, h(x))]$
- This quantity is known as the **risk**
- E.g., loss could be the Hamming loss $\ell_{\text{Hamming}}(a, b) = \begin{cases} 0 & \text{if } a = b \\ 1 & \text{otherwise} \end{cases}$
classification

Online Supervised Learning

- In step t , draw data point \mathbf{x} from distribution \mathbb{D}
- Current hypothesis h guesses the label of \mathbf{x}
- Get true label from oracle \mathcal{O}
- Pay penalty if $h(\mathbf{x})$ is wrong (or earn reward if correct)
- Learning algorithm updates to new hypothesis based on this experience
 - Does not store history

Example: Recommendation



Recommended for You

These recommendations are based on items you own and more.

All | [New Releases](#) | [Coming Soon](#)



[Cybertext: Perspectives on Ergodic Literature](#)

by Espen J. Aarseth (Aug 6, 1997)
Average Customer Review: ★★★★★ (3)
In Stock

List Price: \$22.95

Price: **\$19.55**

[29 used & new](#) from **\$10.82**

[Add to cart](#) [Add to](#)

☐ I own it ☐ Not interested ☐ Rate it

Recommended because you added **Hamlet on the Holodeck** to your Shopping Cart and more ([Fix this](#))



[Narrative as Virtual Reality: Immersion and Interactivity in Lit Media \(Parallax: Re-visions of Culture and Society\)](#)

by Mark J. P. O'Connell (Oct 3, 2003)

Learning Settings

- Supervised or unsupervised (or semi-supervised, weakly supervised, transductive...)
- Online or batch (or reinforcement...)
- Classification, regression
 - (or structured output, clustering, dimensionality reduction...)
- Parametric or non-parameteric

Functional Perspective

Input	Learning Setting
Batch of Data Points with Labels	Batch Supervised Learning
Batch of Data Points	Batch Unsupervised Learning
Data Point(s) and Previous Model	Online Supervised Learning

Concepts

- Supervised and unsupervised learning
- Online and batch learning
- Discriminative and generative
- Output of models: classification and regression

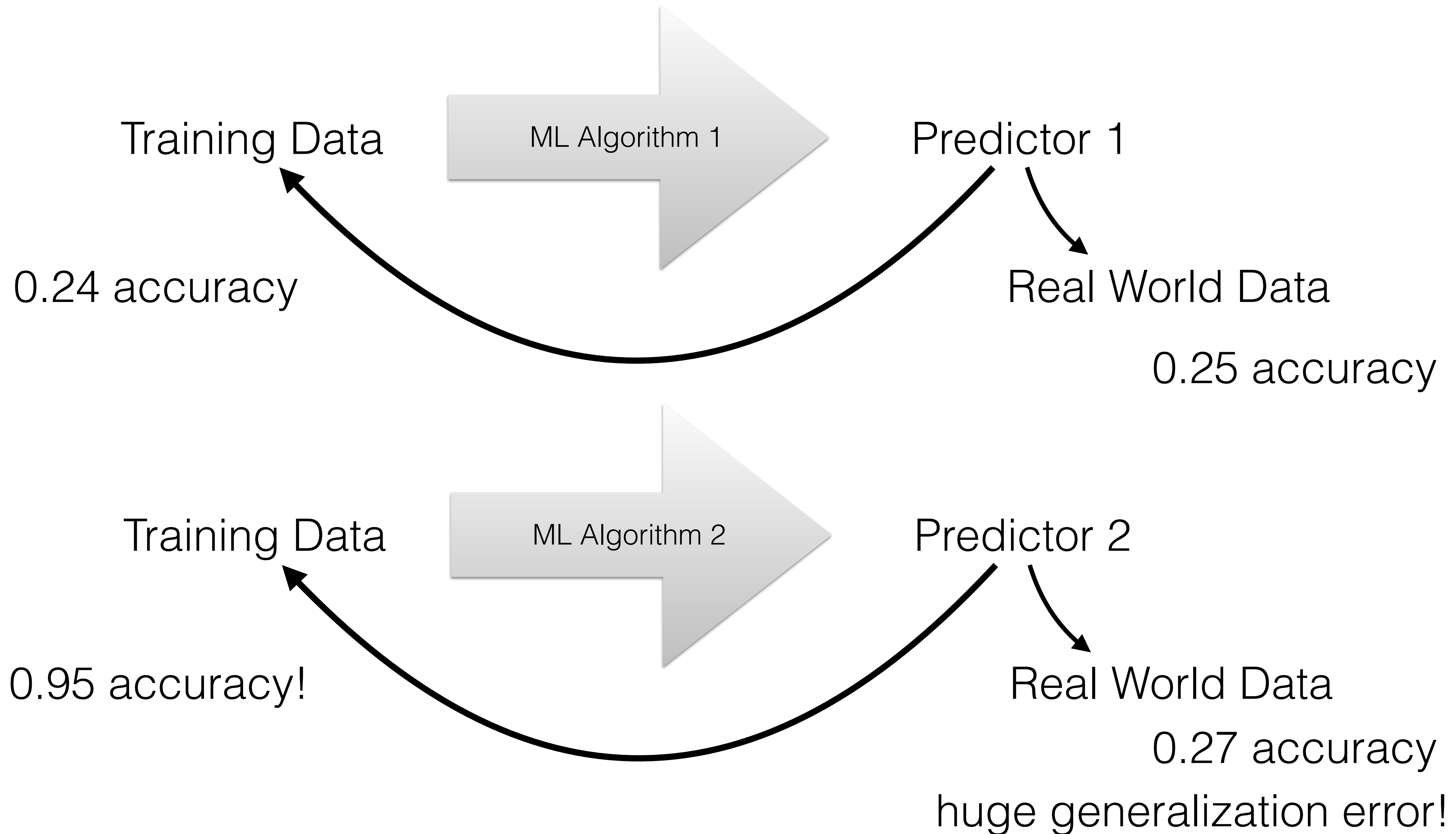
Model Selection

Outline

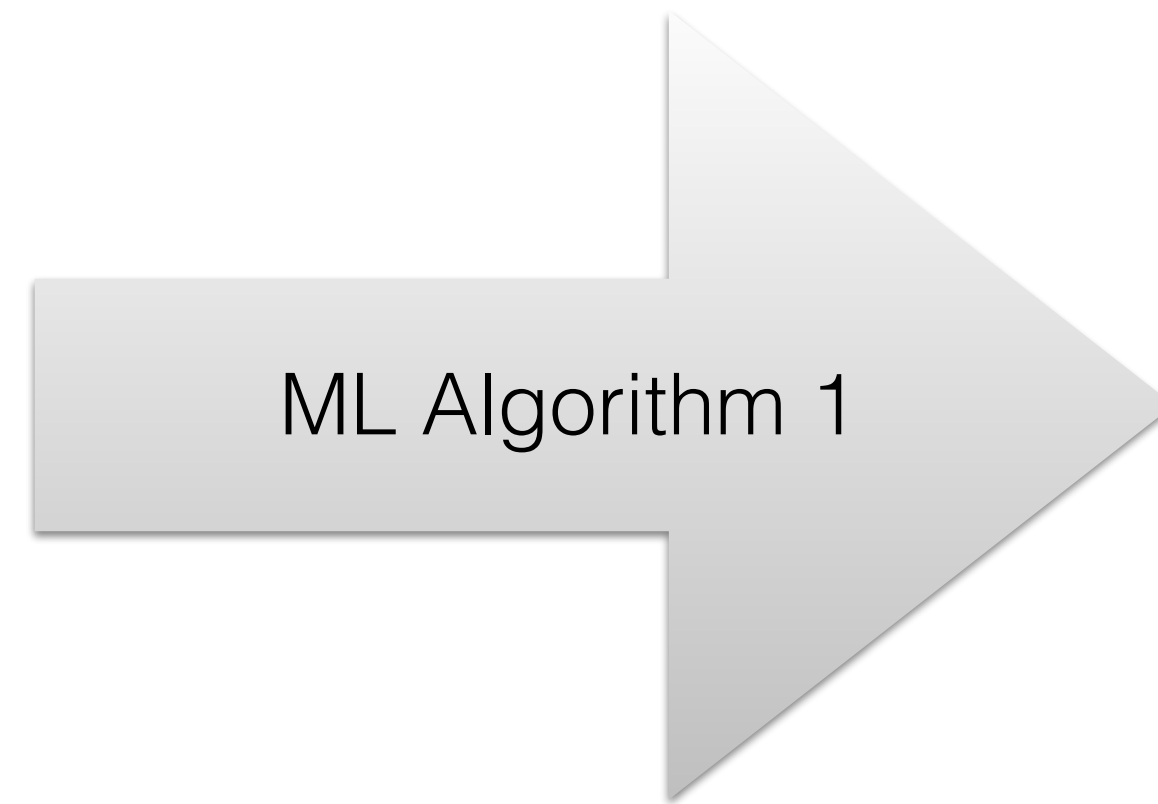
- Overfitting and underfitting
- Bias and variance
- Validation for model selection

Outline

- Overfitting and underfitting
- Bias and variance
- Validation for model selection

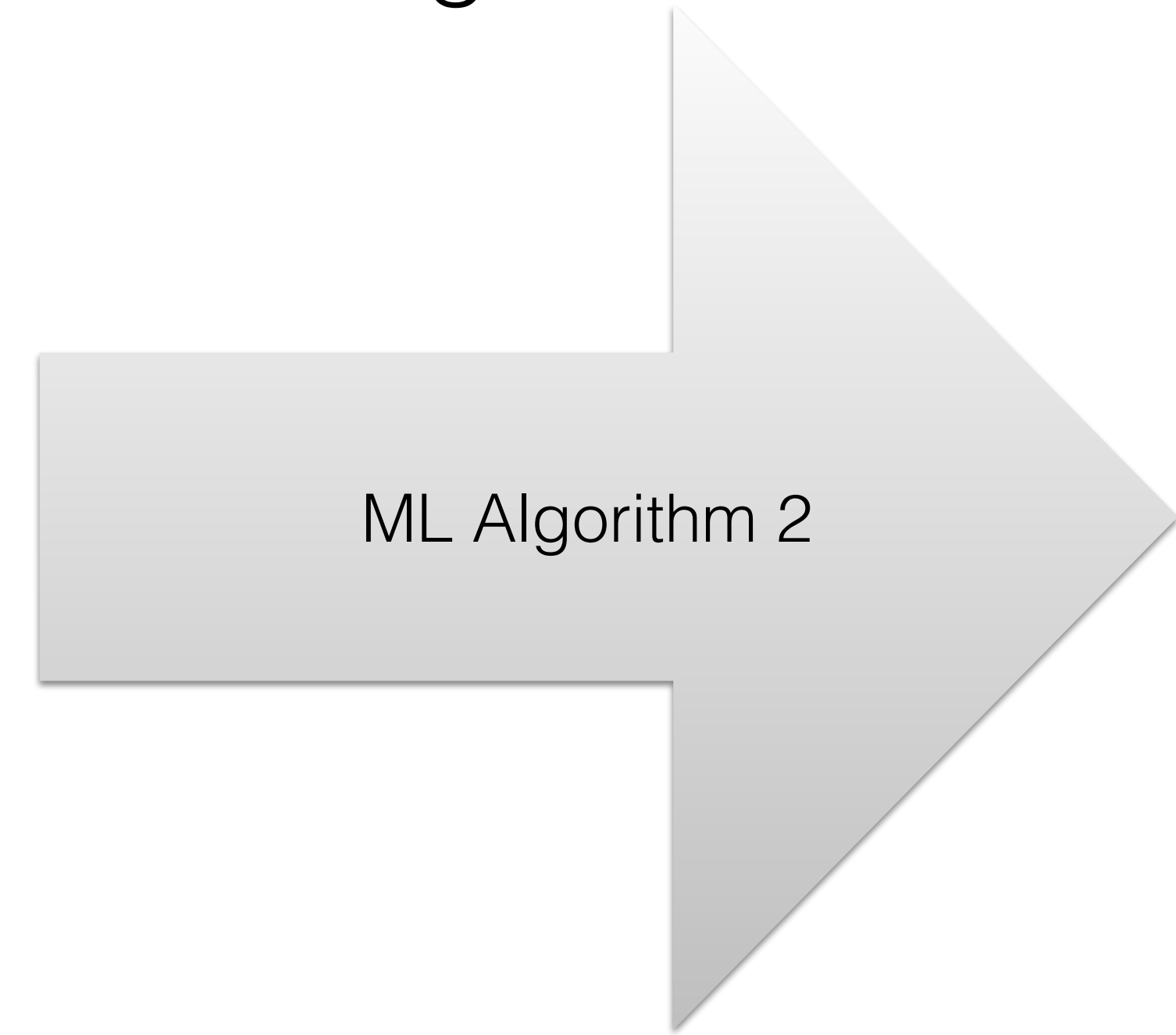


Underfitting

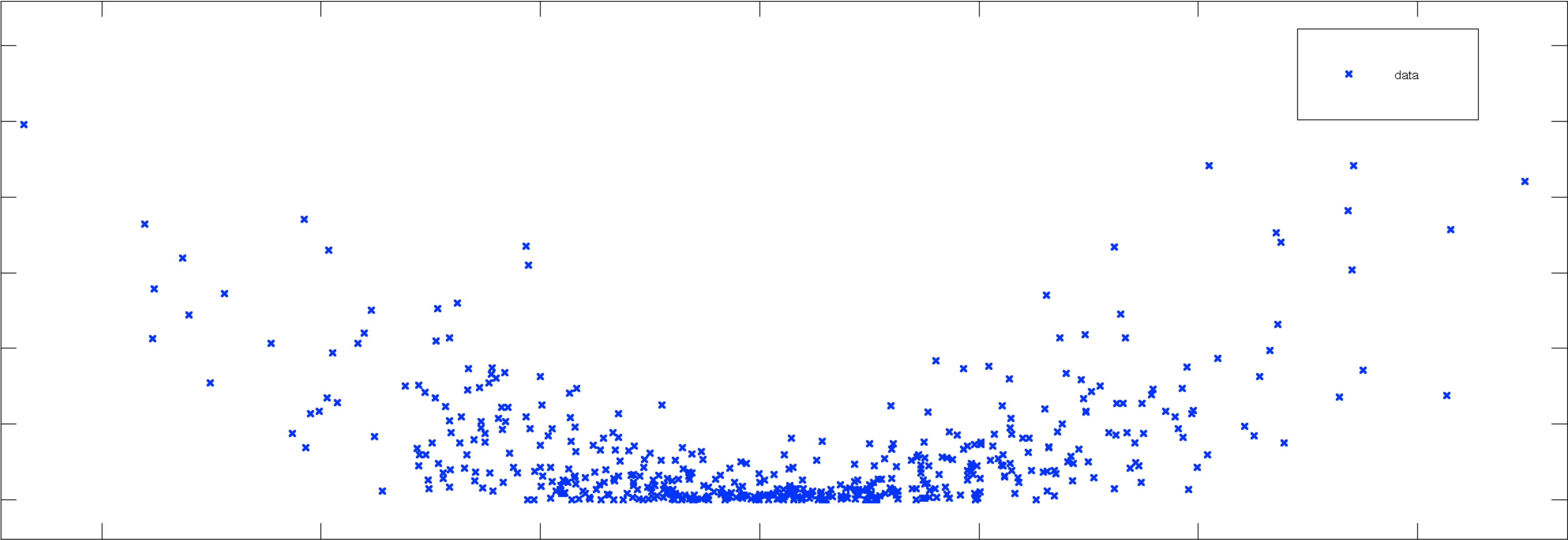


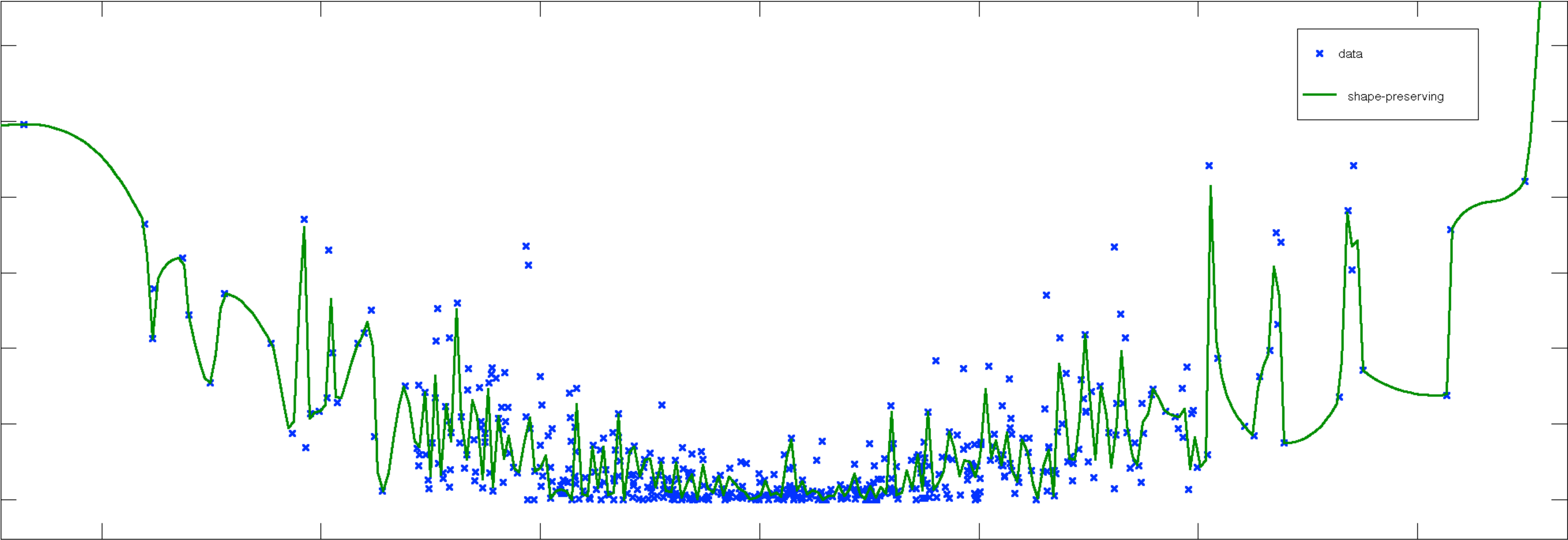
- Low dimensional
- Heavily regularized
- Bad modeling assumptions

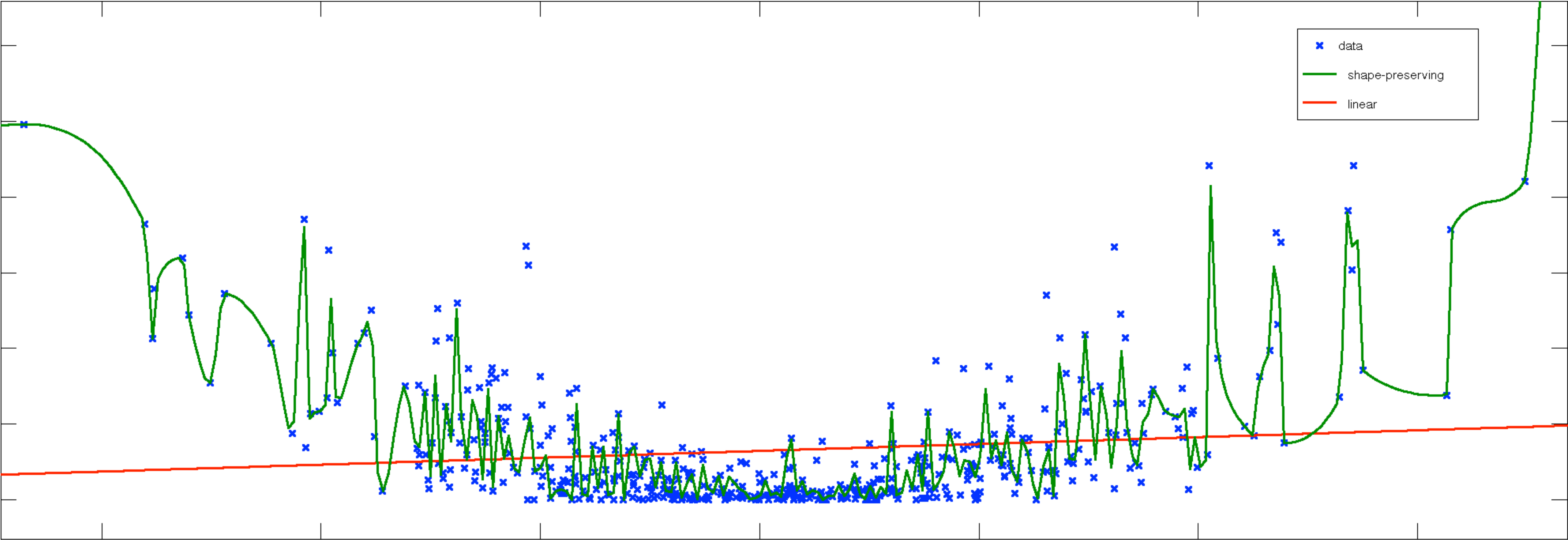
Overfitting

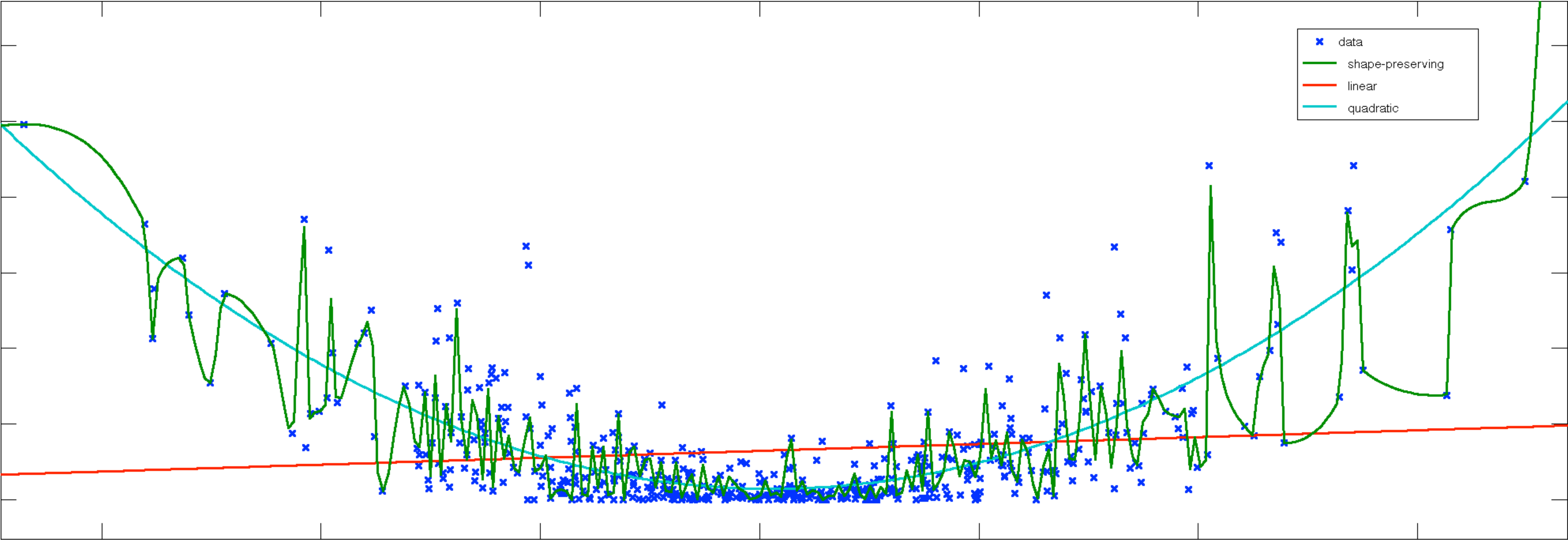


- High dimensional or non-parametric
- Weakly regularized
- Not enough modeling assumptions
- Not enough data









Overfitting and Underfitting

- Training models too complex can cause overfitting
- Training models too simple (or wrong) can cause underfitting

Outline

- Overfitting and underfitting
- Bias and variance
- Validation for model selection

Bias and Variance

- Both contribute to **error**
- Bias: error from incorrect modeling assumptions
- Variance: error from random noise

<http://scott.fortmann-roe.com/docs/BiasVariance.html>

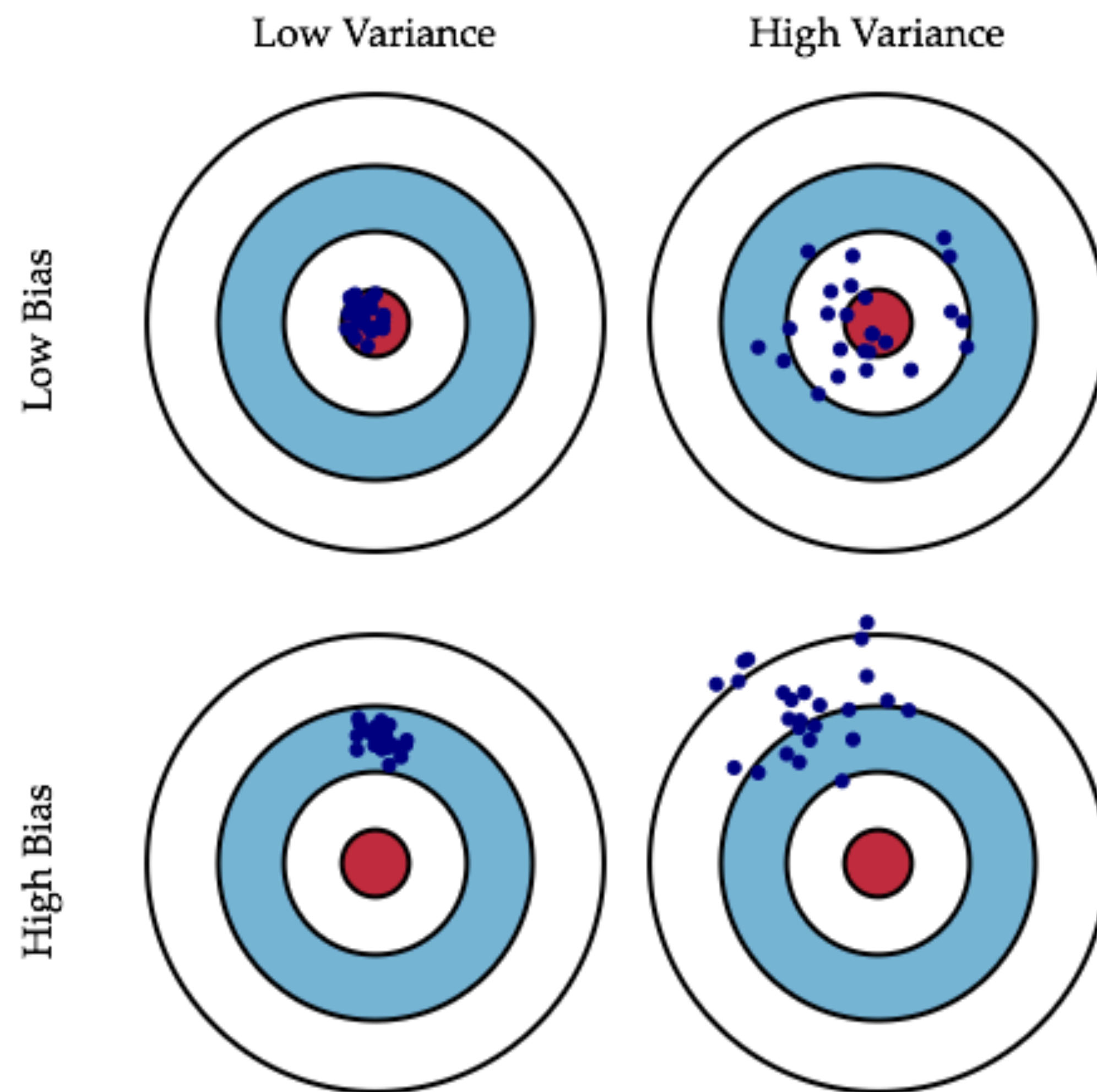


Fig. 1 Graphical illustration of bias and variance.

Mathematical Definition

after Hastie, et al. 2009 ¹

If we denote the variable we are trying to predict as Y and our covariates as X , we may assume that there is a relationship relating one to the other such as $Y = f(X) + \epsilon$ where the error term ϵ is normally distributed with a mean of zero like so $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon)$.

We may estimate a model $f(\hat{X})$ of $f(X)$ using linear regressions or another modeling technique. In this case, the expected squared prediction error at a point x is:

$$Err(x) = E \left[(Y - f(\hat{x}))^2 \right]$$

This error may then be decomposed into bias and variance components:

$$Err(x) = \left(E[f(\hat{x})] - f(x) \right)^2 + E \left[\left(f(\hat{x}) - E[f(\hat{x})] \right)^2 \right] + \sigma_\epsilon^2$$

$$Err(x) = \text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

That third term, irreducible error, is the noise term in the true relationship that cannot fundamentally be reduced by any model. Given the true model and infinite data to calibrate it, we should be able to reduce both the bias and variance terms to 0. However, in a world with imperfect models and finite data, there is a tradeoff between minimizing the bias and minimizing the variance.

$$Err(x) = E \left[(Y - f(\hat{x}))^2 \right]$$

be decomposed into bias and variance components:

$$Err(x) = \underbrace{\left(\underbrace{E[f(\hat{x})]}_{\text{expected learned function}} - \underbrace{f(x)}_{\text{true function}} \right)^2}_{\text{Bias}^2} + \underbrace{E \left[\left(\underbrace{f(\hat{x})}_{\text{learned function}} - \underbrace{E[f(\hat{x})]}_{\text{expected learned function}} \right)^2 \right]}_{\text{Variance}} + \sigma_e^2$$

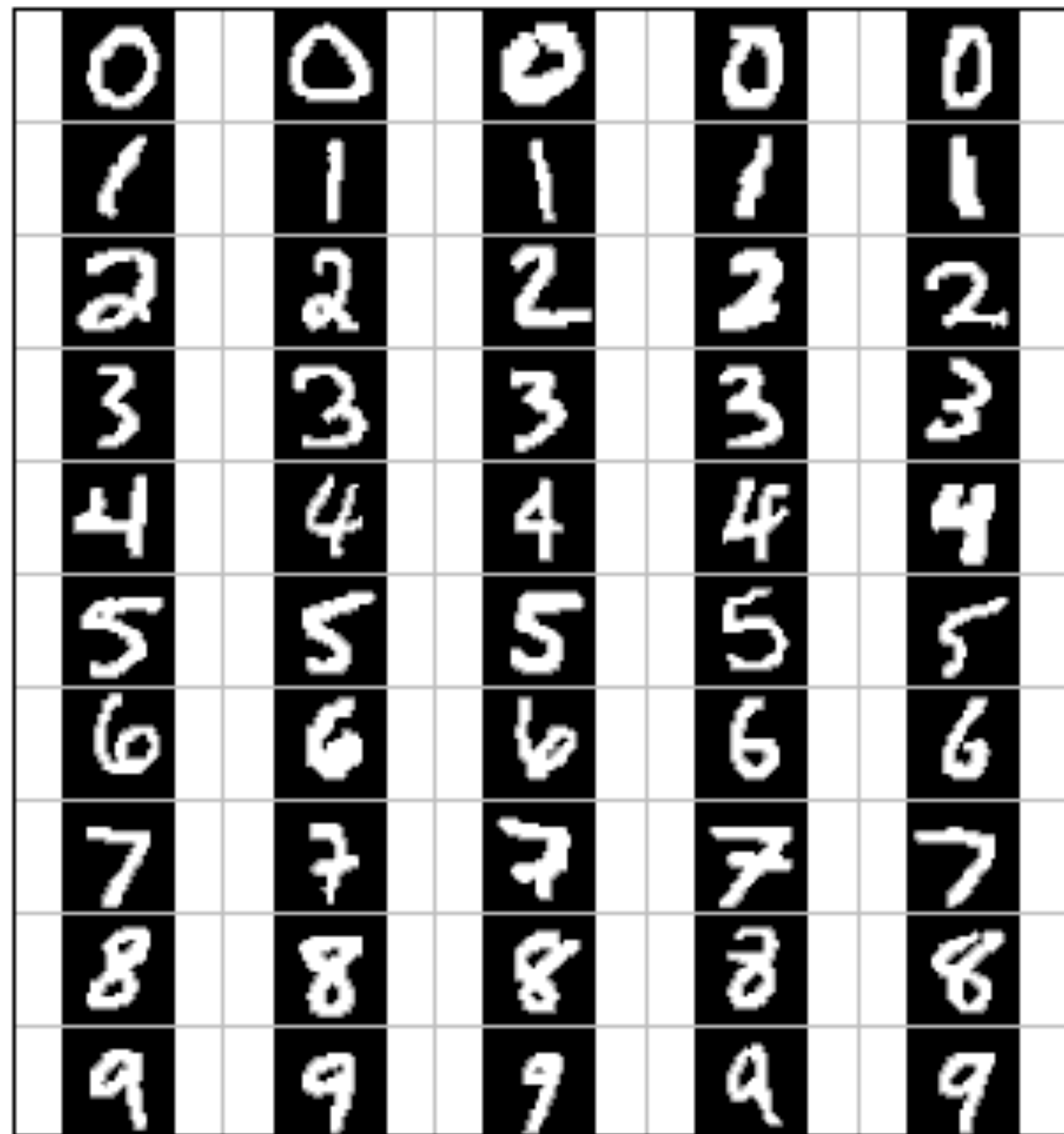
$$Err(x) = \text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

ducible error, is the noise term in the true relationship that cannot be captured by the model. Given the true model and infinite data to calibrate it, we

Outline

- Overfitting and underfitting
- Bias and variance
- Validation for model selection

Nearest-Neighbor Classifiers



classifier = {

 : 0,

 : 0,

 : 0,

 : 0,

 : 0,

 : 1,

 : 1,

...

}

100% training accuracy!



53% testing accuracy...

Held-out Validation

	0			0			0			0			0		
	1			1			1			1			1		
	2			2			2			2			2		
	3			3			3			3			3		
	4			4			4			4			4		
	5			5			5			5			5		
	6			6			6			6			6		
	7			7			7			7			7		
	8			8			8			8			8		
	9			9			9			9			9		

Held-out Validation

0	0	0	0
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9

training data


	Accuracy on training data	Accuracy on validation data
Simple	0.91	0.83
Medium	0.95	0.88
Complex	0.99	0.79
Super Complex	1.0	0.54

0
1
2
3
4
5
6
7
8
9

validation data

Cross Validation

Fold 1



0	0	0	0
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9

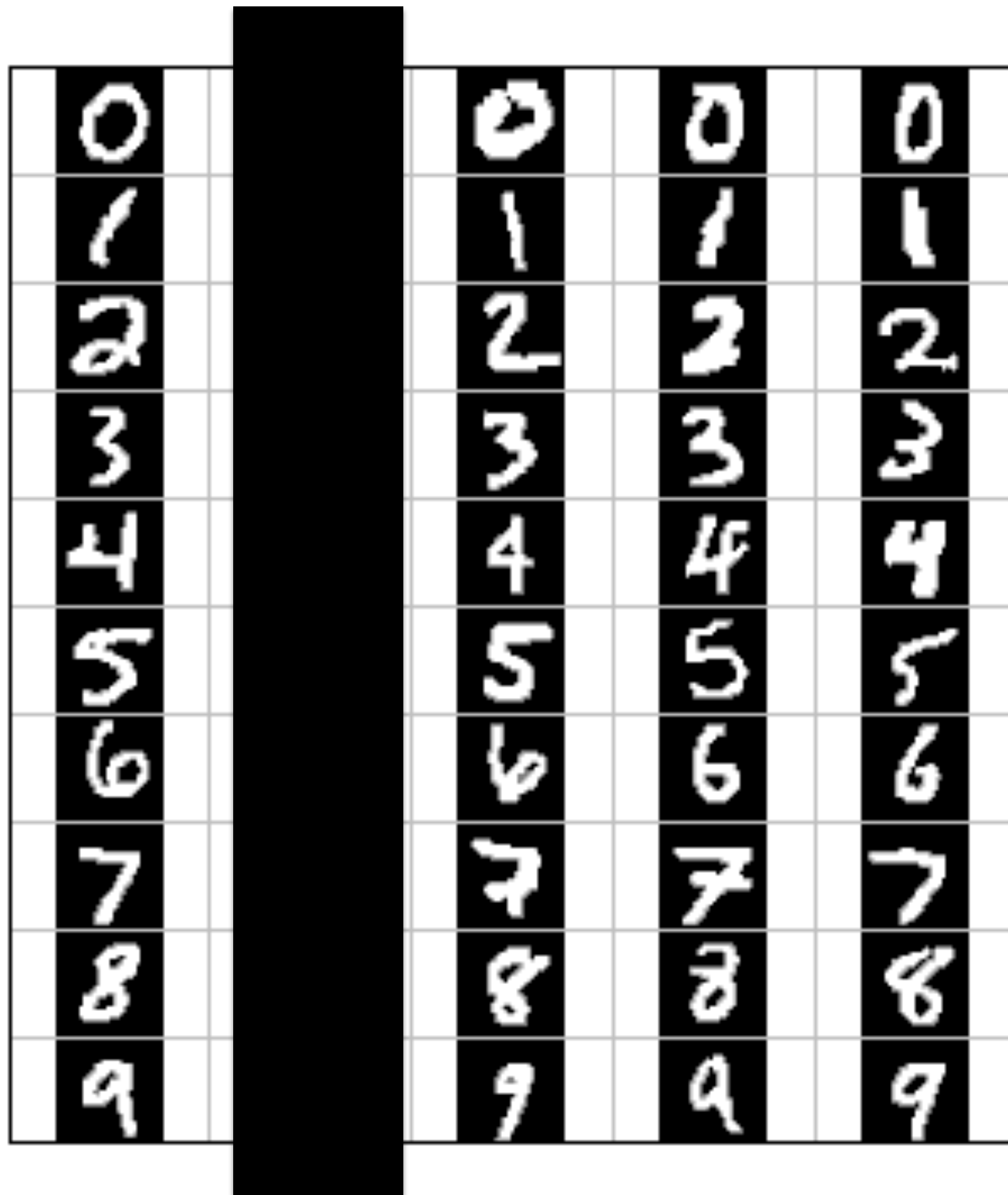
training data

0
1
2
3
4
5
6
7
8
9

validation data

Cross Validation

Fold 2



0				0				0				0				
1				1				1				1				
2				2				2				2				
3				3				3				3				
4				4				4				4				
5				5				5				5				
6				6				6				6				
7				7				7				7				
8				8				8				8				
9				9				9				9				

training data



0
1
2
3
4
5
6
7
8
9

validation data

Cross Validation

Fold 3

0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

training data

0
1
2
3
4
5
6
7
8
9

validation data

Cross Validation

Fold 4

0	0	0
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9

training data

0
1
2
3
4
5
6
7
8
9

validation data

Cross Validation

Fold 5

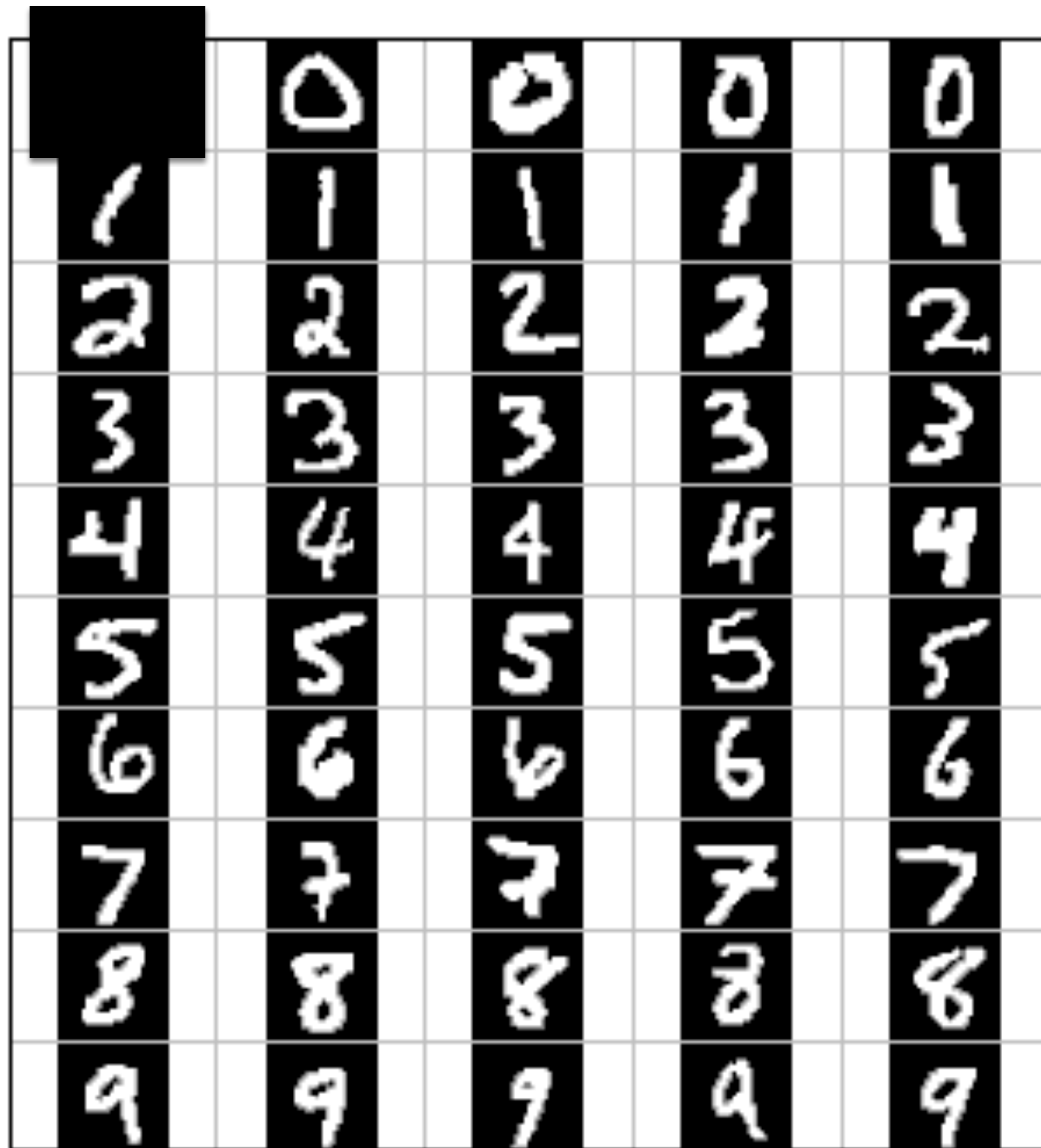
0	0	0	0
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9

training data

0
1
2
3
4
5
6
7
8
9

validation data

Leave-one-out Cross Validation



0	0	0	0	0
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4
5	5	5	5	5
6	6	6	6	6
7	7	7	7	7
8	8	8	8	8
9	9	9	9	9

training data



validation data

Leave-one-out Cross Validation

0				0			0			0
1				1			1			1
2				2			2			2
3				3			3			3
4				4			4			4
5				5			5			5
6				6			6			6
7				7			7			7
8				8			8			8
9				9			9			9

training data



validation data

Leave-one-out Cross Validation

0	0				0	0	0
1	1				1	1	1
2	2				2	2	2
3	3				3	3	3
4	4				4	4	4
5	5				5	5	5
6	6				6	6	6
7	7				7	7	7
8	8				8	8	8
9	9				9	9	9

training data



validation data

Leave-one-out Cross Validation

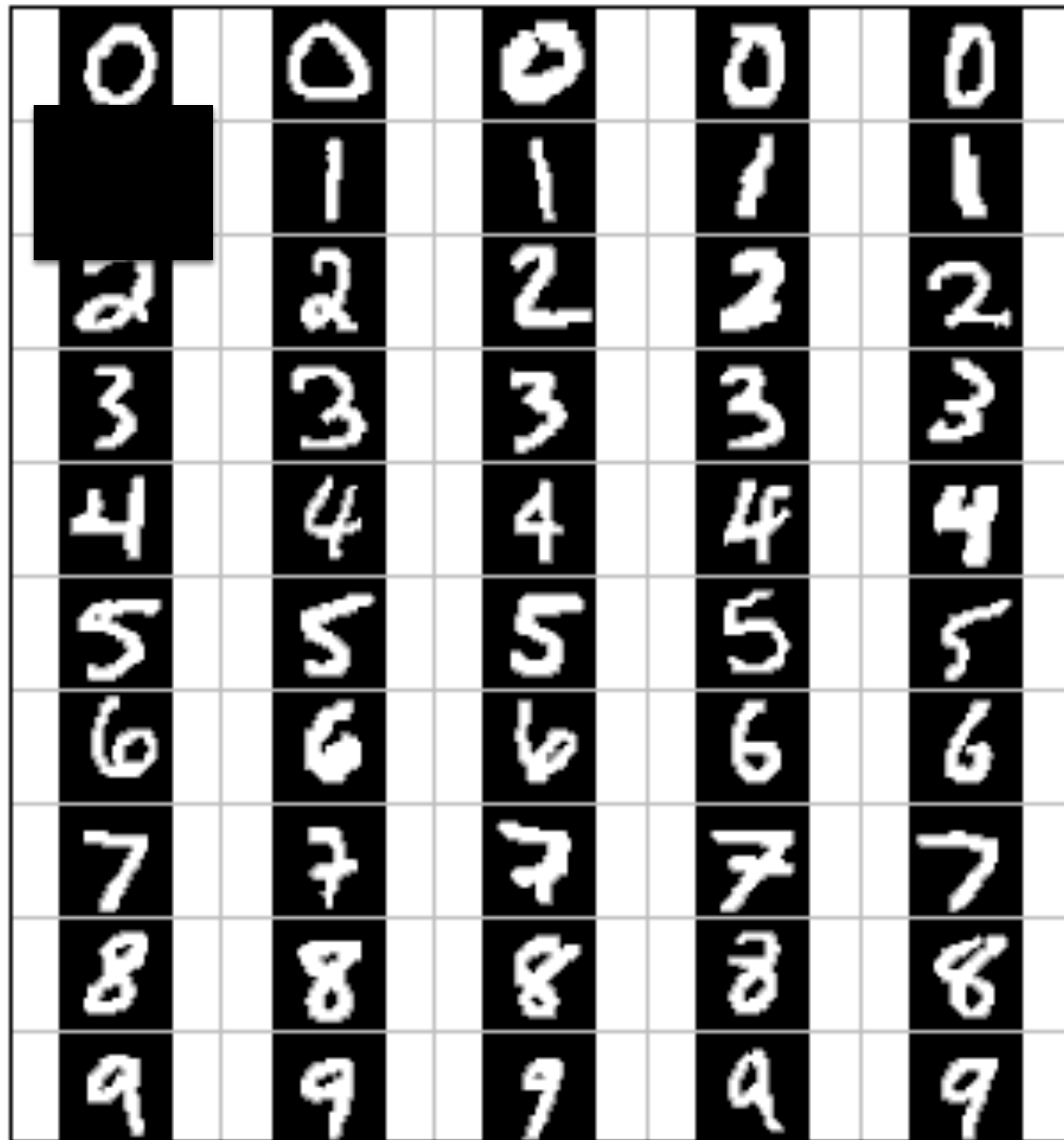
0	0	0	0	
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4
5	5	5	5	5
6	6	6	6	6
7	7	7	7	7
8	8	8	8	8
9	9	9	9	9

training data



validation data

Leave-one-out Cross Validation



training data



validation data

Leave-one-out Cross Validation

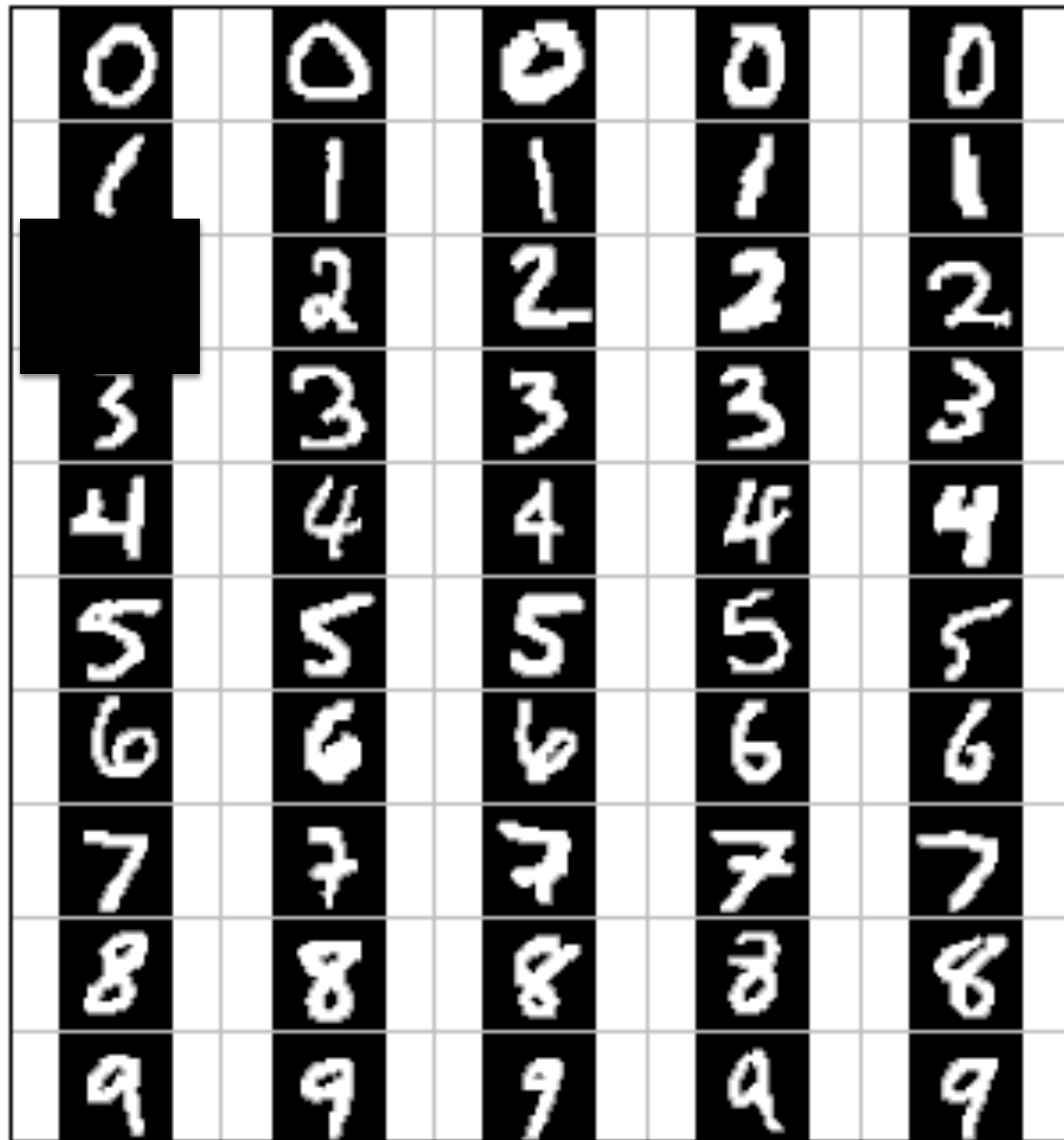
0	0	0	0	0
1	1	1	1	
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4
5	5	5	5	5
6	6	6	6	6
7	7	7	7	7
8	8	8	8	8
9	9	9	9	9

training data



validation data

Leave-one-out Cross Validation



training data



validation data

How Many Folds?

- What are the pros and cons of leave-one-out cross-validation?
- We usually train on N-1 folds and test on 1 fold. What are pros and cons of doing the inverse: train on 1 fold and test on N-1 folds?

0	0	0	0	0
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4
5	5	5	5	5
6	6	6	6	6
7	7	7	7	7
8	8	8	8	8
9	9	9	9	9

Training

0
1
2
3
4
5
6
7
8
9

Testing

Testing versus Validation

- Best practice for experiments:
 - Hold out test set completely hidden from training
 - Use validation on training data for model (or parameter) selection
 - Evaluate on held-out test data

Model Selection via Validation

- Measure performance on **held-out** training data
 - Simulate testing environment
- Rotate **folds** of held-out subsets
- Can even hold out one at a time: **leave-one-out** validation
- Use (cross) validation performance to tune extra parameters

Summary

- Types of machine learning
- Complexity, overfitting, bias
- Validation, cross-validation