Datamodels

Predicting Predictions with Training Data

Andrew Ilyas*, Sung Min (Sam) Park*, Logan Engstrom*,
Guillaume Leclerc, and Aleksander Mądry

@andrew_ilyas

gradientscience.org

Anatomy of an ML prediction

Input x

Output y

"dog" (85%)

Anatomy of an ML prediction

Training set S

Input x

Output y

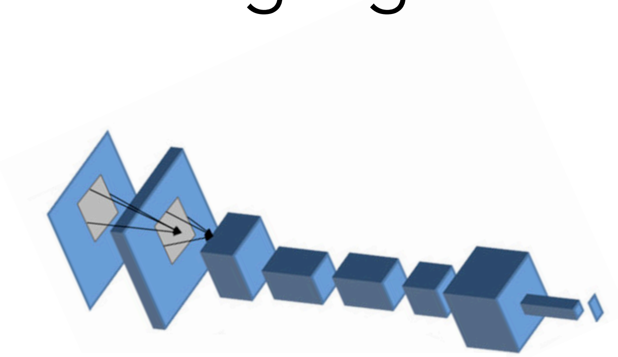
"dog" (85%)

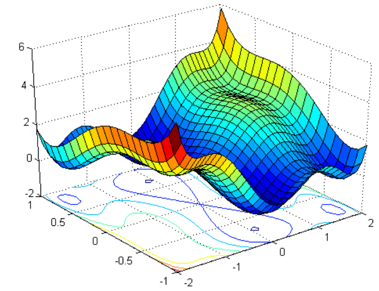
Anatomy of an ML prediction

Training set S

+

Learning algorithm





Input x

Output y

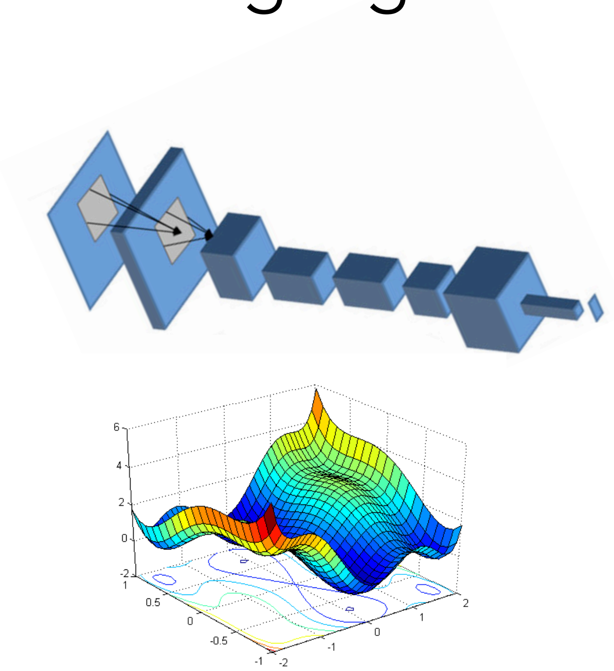
"dog" (85%)

Anatomy of an ML prediction

Training set S

+

Learning algorithm



Input x

⟶

Output y

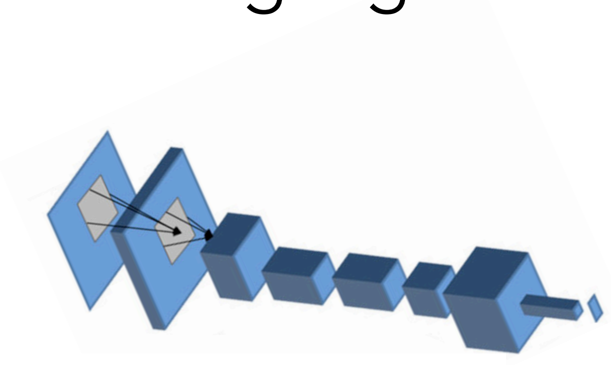
"dog" (85%)

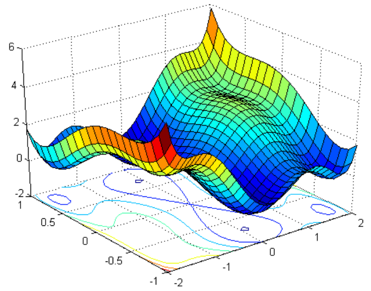
Anatomy of an ML prediction

Training set S

+

Learning algorithm





Input x

→

Output y

"dog" (85%)

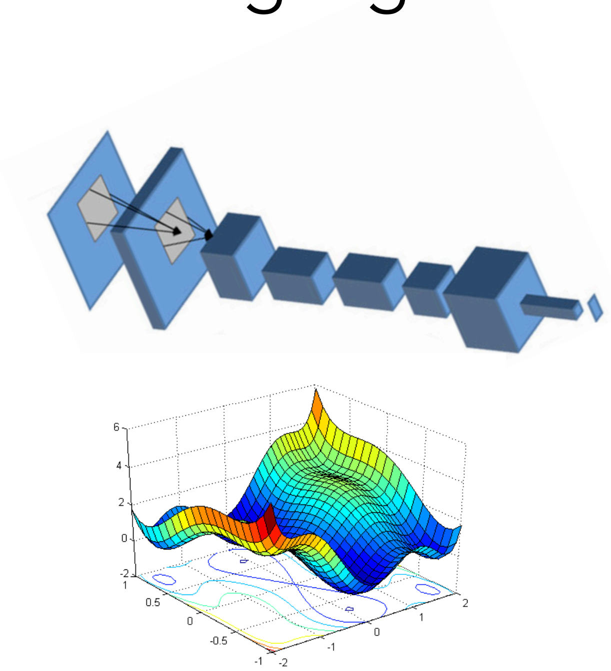
Question: How do training data and learning algorithms combine to yield model outputs?

Anatomy of an ML prediction

Training set S

+

Learning algorithm



Input x

→

Output y

"dog" (85%)

Question: How do training data and learning algorithms combine to yield model outputs?

We introduce **datamodels** to study this problem

What is a datamodel?

What is a datamodel?

Model output

$$f(x, S')$$

What is a datamodel?

Model output

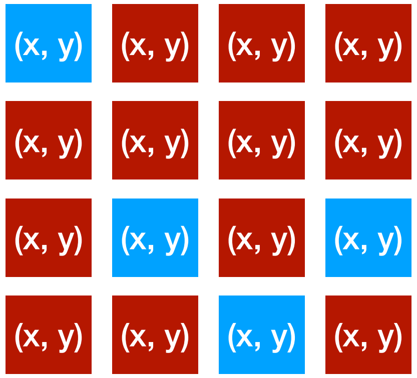
$$f(x, S')$$

Specific example x

What is a datamodel?

Model output

$$f(x, S')$$



🟦 (x, y)	🟥 (x, y)	🟥 (x, y)	🟥 (x, y)
🟥 (x, y)	🟥 (x, y)	🟥 (x, y)	🟥 (x, y)
🟥 (x, y)	🟦 (x, y)	🟥 (x, y)	🟦 (x, y)
🟥 (x, y)	🟥 (x, y)	🟦 (x, y)	🟥 (x, y)

Subset S' of the training set S

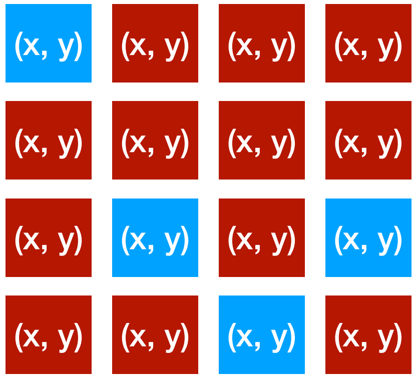
Specific example x

What is a datamodel?

Loss of interest on x
(think: margin of correct class)
after training on S'

Model output

$$f(x, S')$$


(x, y) (x, y) (x, y) (x, y)
(x, y) (x, y) (x, y) (x, y)
(x, y) (x, y) (x, y) (x, y)
(x, y) (x, y) (x, y) (x, y)

Subset S' of the training set S

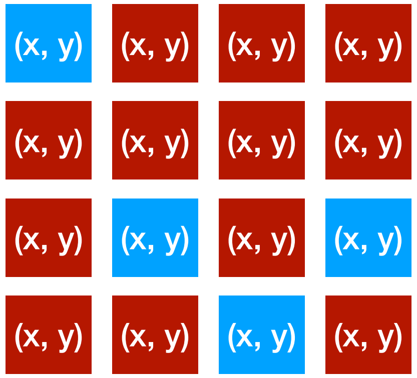
Specific example x

What is a datamodel?

Loss of interest on x
(think: margin of correct class)
after training on S'

Model output

$$f(x, S')$$



(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)

Problem: Function f is complex and hard to analyze

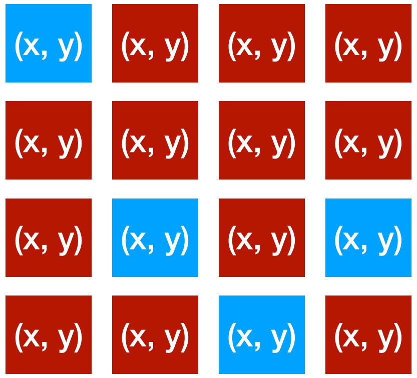
Specific example x

What is a datamodel?

Loss of interest on x
(think: margin of correct class)
after training on S'

Model output

$$f(x, S') \approx \hat{f}(x, S')$$



(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)

Problem: Function f is complex and hard to analyze

S

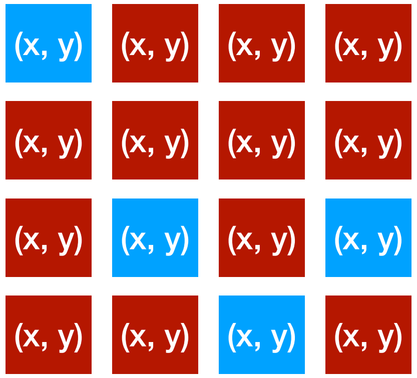
Solution: Study a simple approximation to f

What is a datamodel?

Loss of interest on x
(think: margin of correct class)
after training on S'

Model output

$$f(x, S') \approx \hat{f}(x, S') = \mathbf{1}_{S'} \cdot \theta_x$$



Problem: Function f is complex and hard to analyze

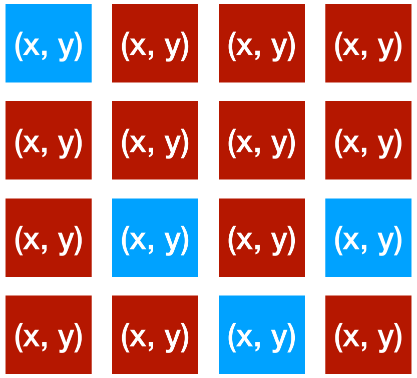
Solution: Study a simple approximation to f

What is a datamodel?

Loss of interest on x
(think: margin of correct class)
after training on S'

Model output

$$f(x, S') \approx \hat{f}(x, S') = \mathbf{1}_{S'} \cdot \theta_x$$



(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)

Indicator vector of S'
[1 0 0 0 0 0 1 0 0 1 0 1 0 0 1 0]

Problem: Function f is complex and hard to analyze

Solution: Study a simple approximation to f

What is a datamodel?

Loss of interest on x
(think: margin of correct class)
after training on S'

Learned vector (one weight
per training example in S)

Model output

$$f(x, S') \approx \hat{f}(x, S') = \mathbf{1}_{S'} \cdot \theta_x$$

(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)

Indicator vector of S'

[1 0 0 0 0 0 1 0 0 1 0 1 0 0 1 0]

Problem: Function f is complex and hard to analyze

Solution: Study a simple approximation to f

What is a datamodel?

Datamodels successfully predict the outcome of training on subsets of the training set!

after training on S'

Model output

$$f(x, S') \approx \hat{f}(x, S') = \mathbf{1}_{S'} \cdot \theta_x$$

🟦 (x, y)	🟥 (x, y)	🟥 (x, y)	🟥 (x, y)
🟥 (x, y)	🟥 (x, y)	🟥 (x, y)	🟥 (x, y)
🟥 (x, y)	🟦 (x, y)	🟥 (x, y)	🟦 (x, y)
🟥 (x, y)	🟥 (x, y)	🟦 (x, y)	🟥 (x, y)

Indicator vector of S'

[1 0 0 0 0 0 1 0 0 1 0 1 0 0 1 0]

Problem: Function f is complex and hard to analyze

Solution: Study a simple approximation to f

What is a datamodel?

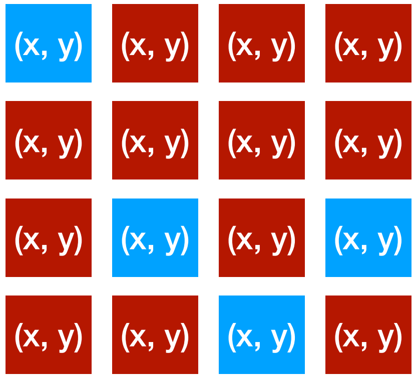
Datamodels successfully predict the outcome of training on subsets of the training set!

(Open Q: why?)

after training on S'

Model output

$$f(x, S') \approx \hat{f}(x, S') = \mathbf{1}_{S'} \cdot \theta_x$$



(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)
(x, y)	(x, y)	(x, y)	(x, y)

Indicator vector of S'

[1 0 0 0 0 0 1 0 0 1 0 1 0 0 1 0]

Problem: Function f is complex and hard to analyze

Solution: Study a simple approximation to f

Applying datamodels

$$f(x, S') \approx \theta_x^\top \mathbf{1}_{\mathbf{S}'}$$

Applying datamodels

$$f(x, S') \approx \theta_x^\top \mathbf{1}_{\mathbf{S}'}$$

Datamodels provide a versatile framework for analyzing model predictions and data

Applying datamodels

$$f(x, S') \approx \theta_x^\top \mathbf{1}_{\mathbf{S}'}$$

Datamodels provide a versatile framework for analyzing model predictions and data

We can use datamodels:

Applying datamodels

$$f(x, S') \approx \theta_x^\top \mathbf{1}_{\mathbf{S}'}$$

Datamodels provide a versatile framework
for analyzing model predictions and data

We can use datamodels:
→ To analyze **model brittleness**

Applying datamodels

$$f(x, S') \approx \theta_x^\top \mathbf{1}_{\mathbf{S}'}$$

Datamodels provide a versatile framework
for analyzing model predictions and data

We can use datamodels:

→ To analyze **model brittleness**

→ To predict **data counterfactuals**

Applying datamodels

$$f(x, S') \approx \theta_x^\top \mathbf{1}_{\mathbf{S}'}$$

Datamodels provide a versatile framework for analyzing model predictions and data

We can use datamodels:

→ To analyze **model brittleness**

→ To predict **data counterfactuals**

→ To find **similar train images** to a given target

Applying datamodels

$$f(x, S') \approx \theta_x^\top \mathbf{1}_{\mathbf{S}'}$$

Datamodels provide a versatile framework
for analyzing model predictions and data

We can use datamodels:

→ To analyze **model brittleness**

→ To predict **data counterfactuals**

→ To find **similar train images** to a given target

→ To find **train-test leakage**

Applying datamodels

$$f(x, S') \approx \theta_x^\top \mathbf{1}_{\mathbf{S}'}$$

Datamodels provide a versatile framework
for analyzing model predictions and data

We can use datamodels:

→ To analyze **model brittleness**

→ To predict **data counterfactuals**

→ To find **similar train images** to a given target

→ To find **train-test leakage**

→ To identify **data subpopulations**

Applying datamodels

$$f(x, S') \approx \theta_x^\top \mathbf{1}_{\mathbf{S}'}$$

Datamodels provide a versatile framework for analyzing model predictions and data

We can use datamodels:

→ To analyze **model brittleness**

→ To predict **data counterfactuals**

→ To find **similar train images** to a given target

→ To find **train-test leakage**

→ To identify **data subpopulations**

→ As a rich **data embedding**

Applying datamodels

$$f(x, S') \approx \theta_x^\top \mathbf{1}_{\mathbf{S}'}$$

Datamodels provide a versatile framework
for analyzing model predictions and data

We can use datamodels:

→ To analyze **model brittleness**

→ To predict **data counterfactuals**

→ To find **similar train images** to a given target

→ To find **train-test leakage**

→ To identify **data subpopulations**

→ As a rich **data embedding**

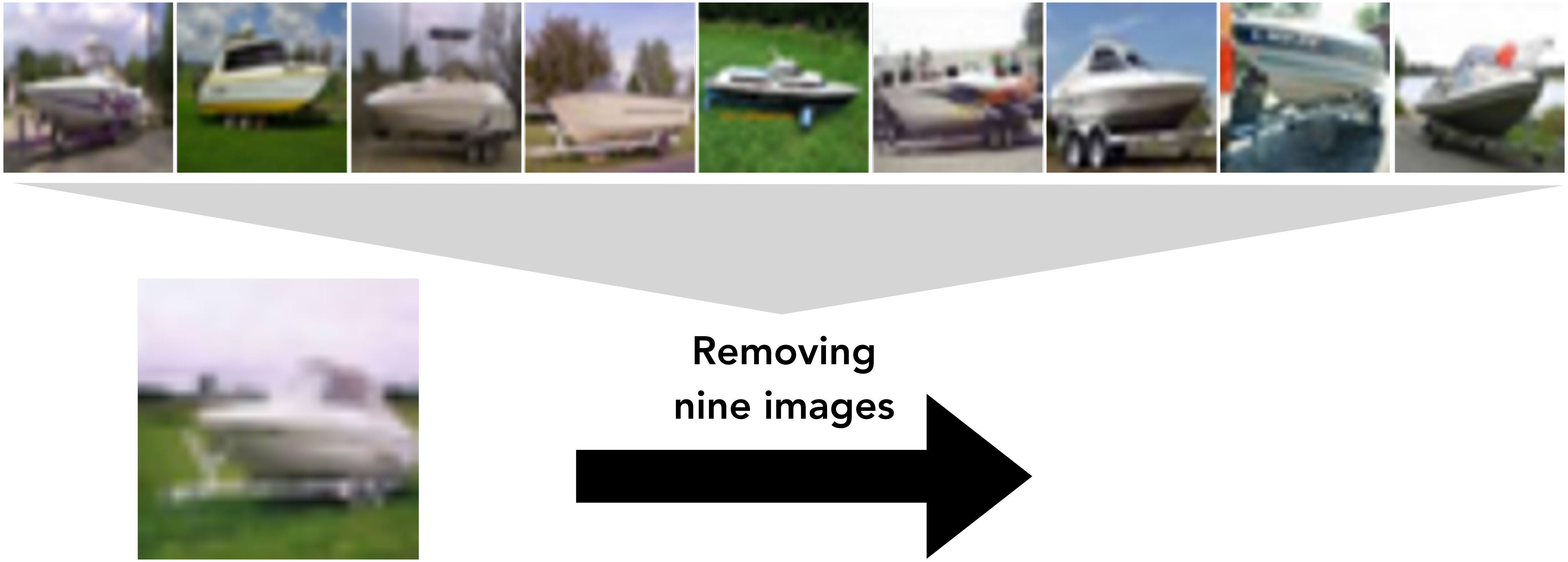
Datamodels: Analyzing model brittleness

Datamodels: Analyzing model brittleness

“boat”

(71% confidence)

Datamodels: Analyzing model brittleness

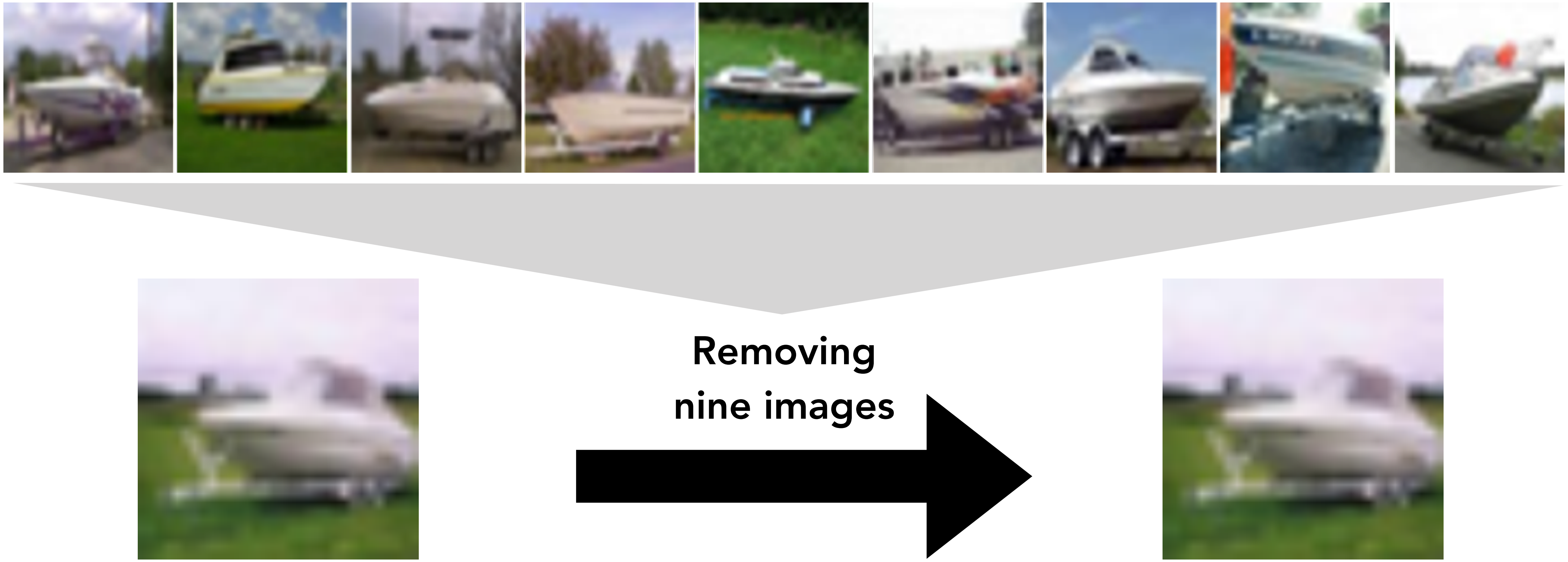


Removing nine images

“boat”

(71% confidence)

Datamodels: Analyzing model brittleness



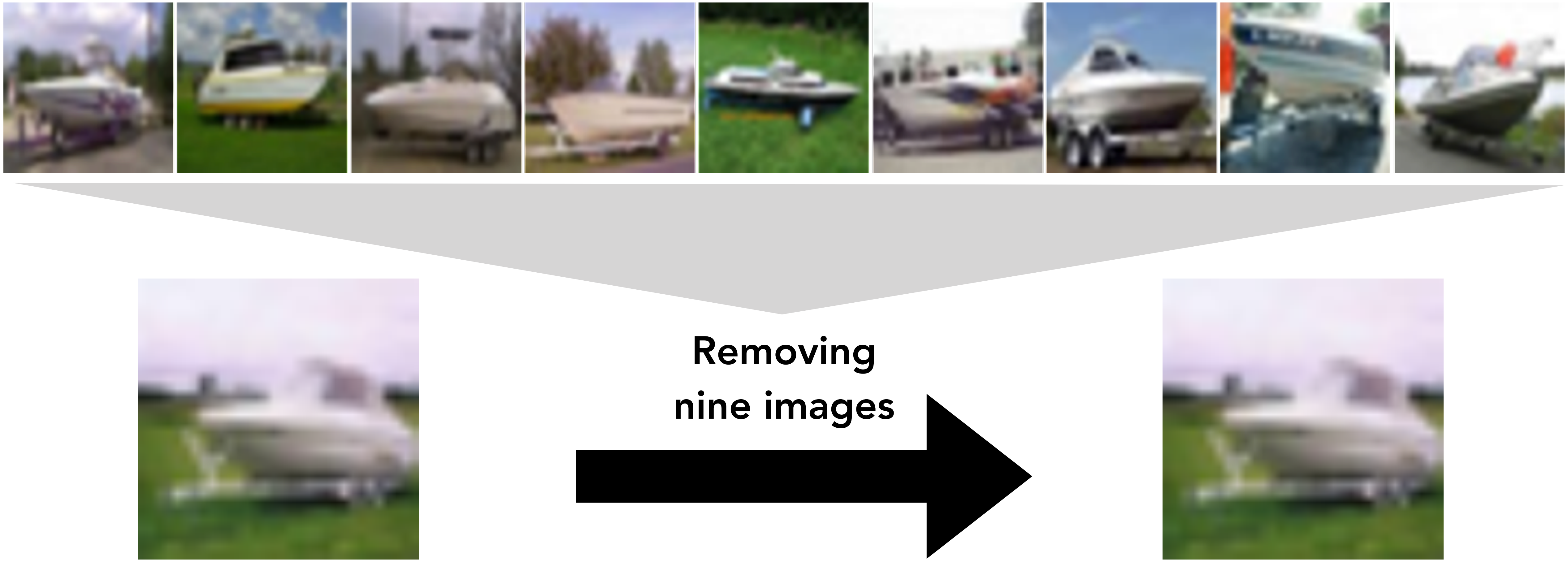
Removing nine images

"boat"

(71% confidence)

"airplane"

Datamodels: Analyzing model brittleness



Removing nine images

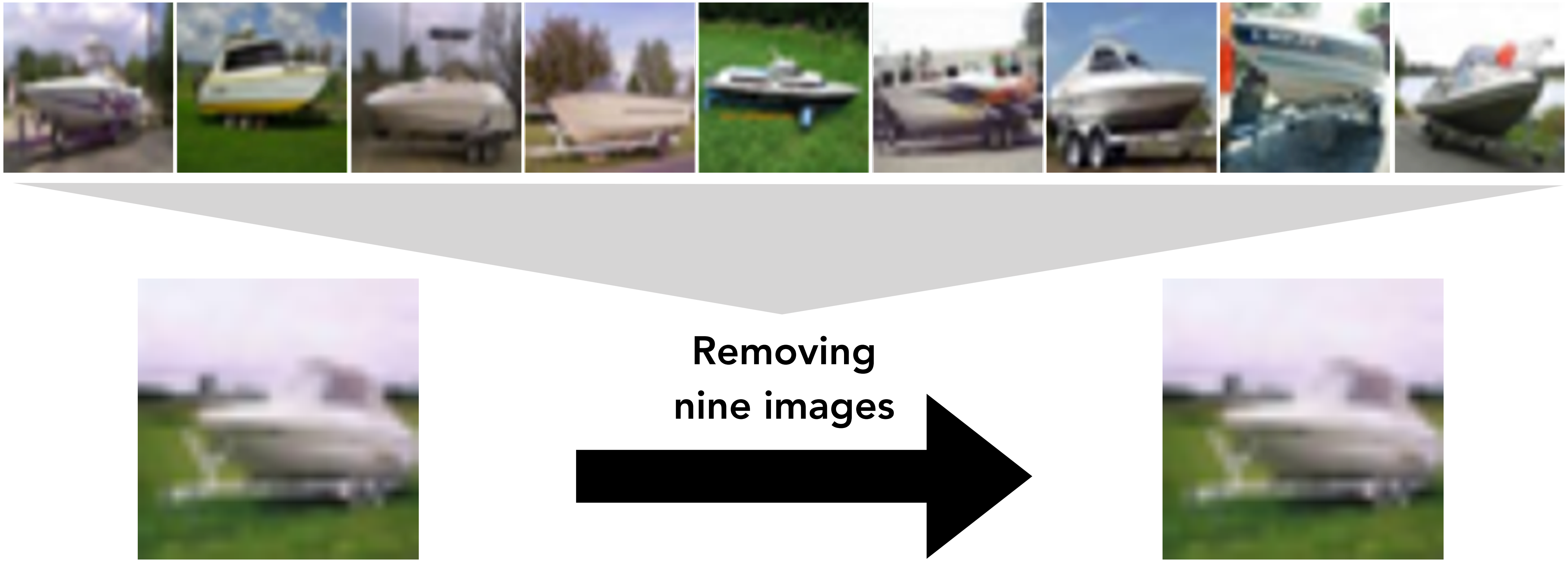
"boat"

(71% confidence)

"airplane"

Can use datamodels to **efficiently** find brittle predictions

Datamodels: Analyzing model brittleness



Removing nine images

"boat"

(71% confidence)

"airplane"

Can use datamodels to **efficiently** find brittle predictions

Turns out: ~25% of test examples can be misclassified by removing **< 0.2%** of training examples

Takeaways

Datamodels:

A framework for understanding both data and predictions

→ Learn simple data-to-output mapping

→ A versatile tool for model-data understanding

  → Analyzing model brittleness

  → (Many) more applications

See paper for (much) more! https://arxiv.org/abs/2202.00622

Blog posts at:

@andrew_ilyas

gradientscience.org