Machine Learning for Data Science (CS4786)
Lecture 24

Differential Privacy and Re-useable Holdout

Defining Privacy

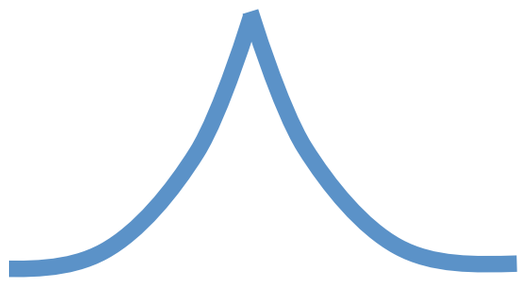
Defining Privacy

Dataset +

Defining Privacy

Dataset +

Learning Algorithm

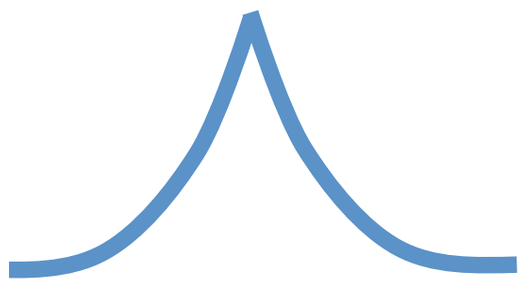


Distribution of outcome

Defining Privacy

Dataset +

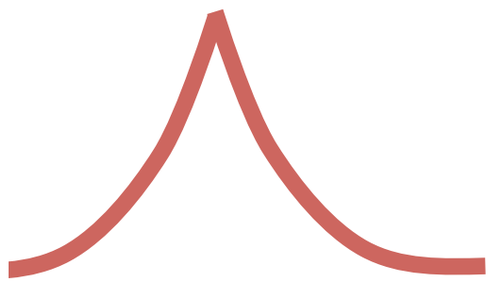
Learning Algorithm



Distribution of outcome

Dataset +

Learning Algorithm

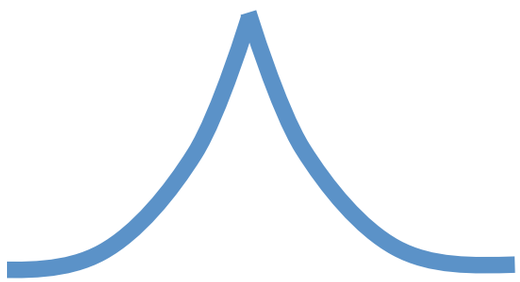


Distribution of outcome

Defining Privacy

Dataset +

Learning Algorithm

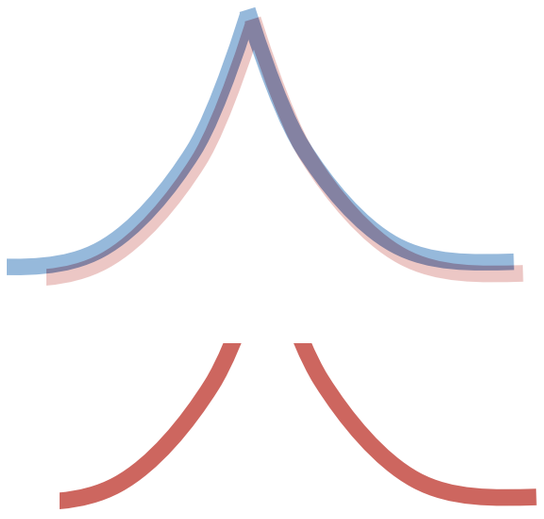


Distribution of outcome

Dataset +

Similar

Learning Algorithm



Distribution of outcome

Differential Privacy

Differential Privacy

- A deterministic algorithm cannot preserve privacy

Differential Privacy

- A deterministic algorithm cannot preserve privacy
- Say S = ($Data_1$,…,$Data_n$) is the data provided to learning algorithm (be it clustering, supervised learning etc).

Differential Privacy

- A deterministic algorithm cannot preserve privacy
- Say S = ($Data_1$,…,$Data_n$) is the data provided to learning algorithm (be it clustering, supervised learning etc).
- Say (randomized) learning algorithm A takes this training data and returns solution as A(S)

Differential Privacy

- A deterministic algorithm cannot preserve privacy

- Say S = ($Data_1$,…,$Data_n$) is the data provided to learning algorithm (be it clustering, supervised learning etc).

- Say (randomized) learning algorithm A takes this training data and returns solution as A(S)

- Algorithm A is (ϵ,δ)- differentially private if for all samples S and S' that only differ by one data point and any set C

$$P(A(S) \in C) \leq e^{\epsilon} P(A(S') \in C) + \delta$$

Differential Privacy

- A deterministic algorithm cannot preserve privacy

- Say S = ($Data_1$,…,$Data_n$) is the data provided to learning algorithm (be it clustering, supervised learning etc).

- Say (randomized) learning algorithm A takes this training data and returns solution as A(S)

- Algorithm A is (ϵ,δ)- differentially private if for all samples S and S' that only differ by one data point and any set C

$$P(A(S) \in C) \leq e^{\epsilon} P(A(S') \in C) + \delta$$

- δ=0 is called pure differential privacy

Differential Privacy

Differential Privacy

Dataset +

Dataset +

Differential Privacy

Dataset +

Dataset +

$$P(A(S) \in C) \leq e^{\epsilon} P(A(S') \in C)$$

Differential Privacy

Dataset +

Dataset +

$$P(A(S) \in C) \leq e^{\epsilon} P(A(S') \in C)$$

≀≀

1 (for small ϵ)

Obtaining Differential Privacy

Obtaining Differential Privacy

- Typical mechanism: Add noise to outcome or inside algorithm

Obtaining Differential Privacy

- Typical mechanism: Add noise to outcome or inside algorithm

- More privacy we want the more noise we add

Why it works?

Why it works?

- Take any arbitrary (possibly deterministic) function f(S).

Why it works?

- Take any arbitrary (possibly deterministic) function f(S).
- Say $B = \max_{S,S'} |f(S) - f(S')|$ where S and S' differ on one data point

Why it works?

- Take any arbitrary (possibly deterministic) function f(S).
- Say $B = \max_{S,S'} |f(S) - f(S')|$ where S and S' differ on one data point
- A(S) = f(S) + B Laplace(0,1)/ϵ

Why it works?

- Take any arbitrary (possibly deterministic) function f(S).
- Say $B = \max_{S,S'} |f(S) - f(S')|$ where S and S' differ on one data point
- A(S) = f(S) + B Laplace(0,1)/ϵ
- A is (ϵ,0)-differentially private

Why it works?

$$\frac{p(x; f(S))}{p(x; f(S'))}$$

Why it works?

$$\frac{p(x; f(S))}{p(x; f(S'))} = \frac{e^{-\epsilon|f(S)-x|/B}}{e^{-\epsilon|f(S')-x|/B}}$$

Why it works?

$$\frac{p(x; f(S))}{p(x; f(S'))} = \frac{e^{-\epsilon|f(S)-x|/B}}{e^{-\epsilon|f(S')-x|/B}}$$

$$= e^{\epsilon \frac{|f(S')-x|-|f(S)-x|}{B}}$$

Why it works?

$$\frac{p(x; f(S))}{p(x; f(S'))} = \frac{e^{-\epsilon|f(S)-x|/B}}{e^{-\epsilon|f(S')-x|/B}}$$

$$= e^{\epsilon \frac{|f(S')-x|-|f(S)-x|}{B}}$$

$$\leq e^{\epsilon \frac{|f(S')-f(S)|}{B}}$$

Why it works?

$$\frac{p(x; f(S))}{p(x; f(S'))} = \frac{e^{-\epsilon|f(S)-x|/B}}{e^{-\epsilon|f(S')-x|/B}}$$

$$= e^{\epsilon \frac{|f(S')-x|-|f(S)-x|}{B}}$$

$$\leq e^{\epsilon \frac{|f(S')-f(S)|}{B}}$$

$$\leq e^{\epsilon}$$

The Magic Broker Scam

Based on Aaron Roth's slide

The Magic Broker Scam

Based on Aaron Roth's slide

Day 1

Tomorrow Stock X

The Magic Broker Scam

Based on Aaron Roth's slide

Day 1

Tomorrow Stock X

Day 2

Tomorrow Stock X

The Magic Broker Scam

Based on Aaron Roth's slide

Day 1

Tomorrow Stock X

Day 2

Tomorrow Stock X

Day 3

Tomorrow Stock X

The Magic Broker Scam

Based on Aaron Roth's slide

Day 1

Tomorrow Stock X

Day 2

Tomorrow Stock X

Day 3

Tomorrow Stock X

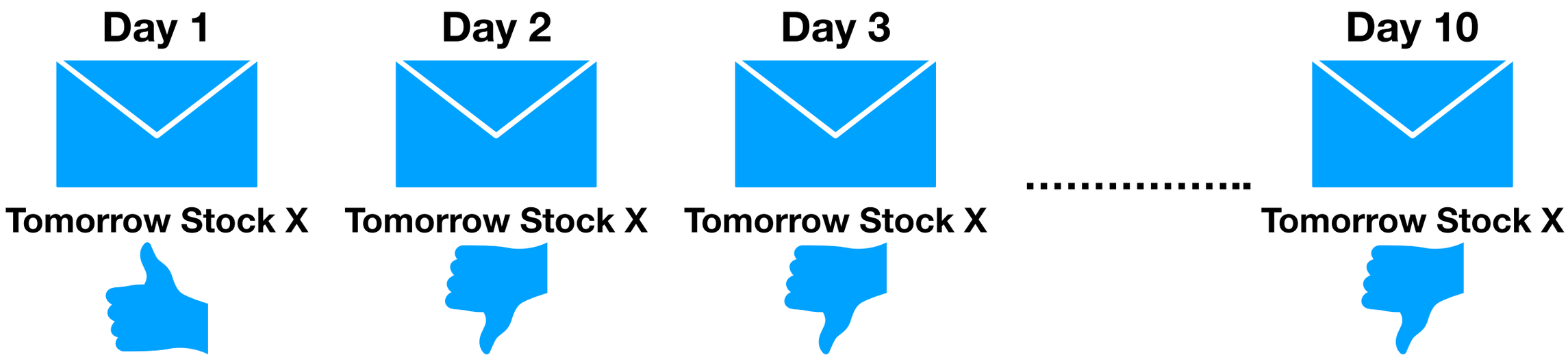
.....

Day 10

Tomorrow Stock X

The Magic Broker Scam

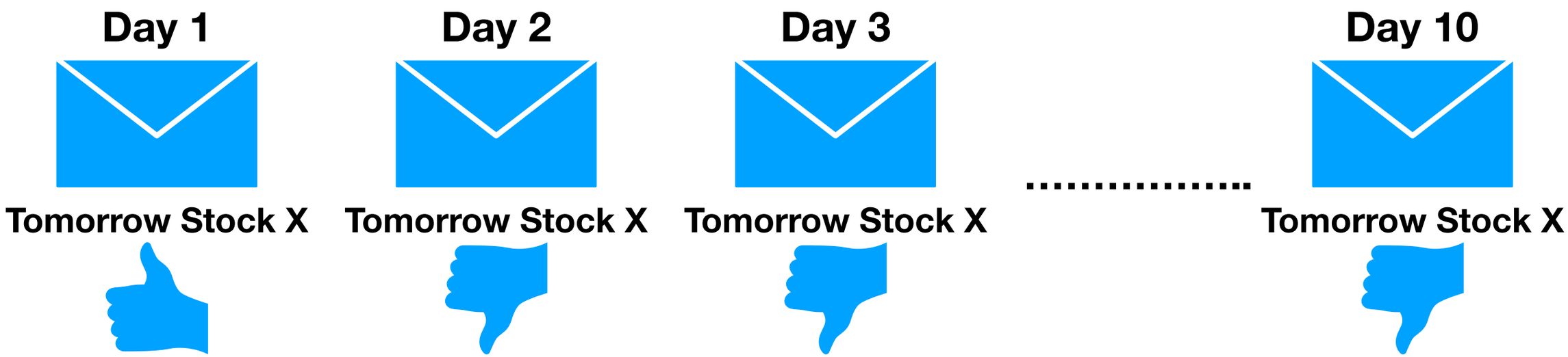
Based on Aaron Roth's slide



- Day 11: Pay me I will help you invest

The Magic Broker Scam

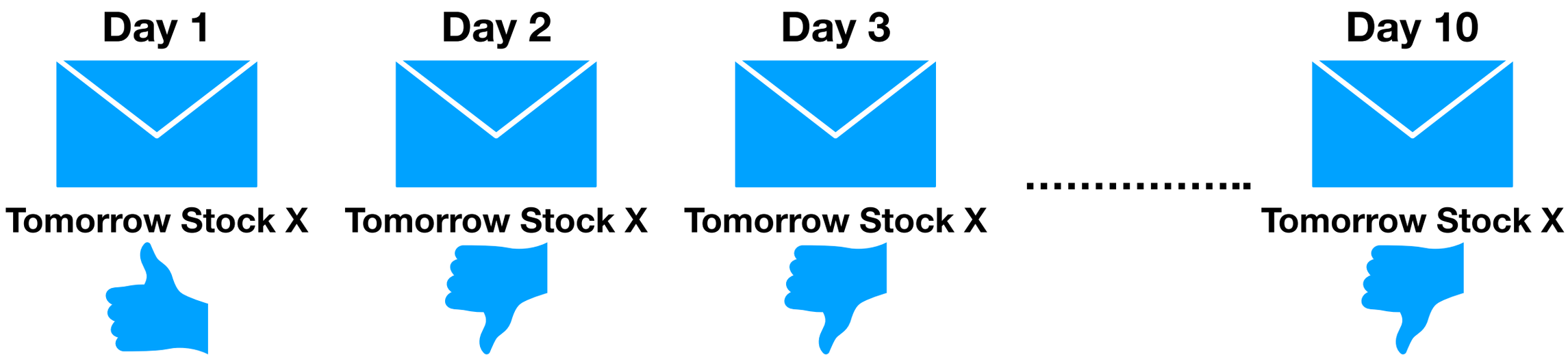
Based on Aaron Roth's slide



- Day 11: Pay me I will help you invest
- What are the chances that some one guesses randomly and gets correct?

The Magic Broker Scam

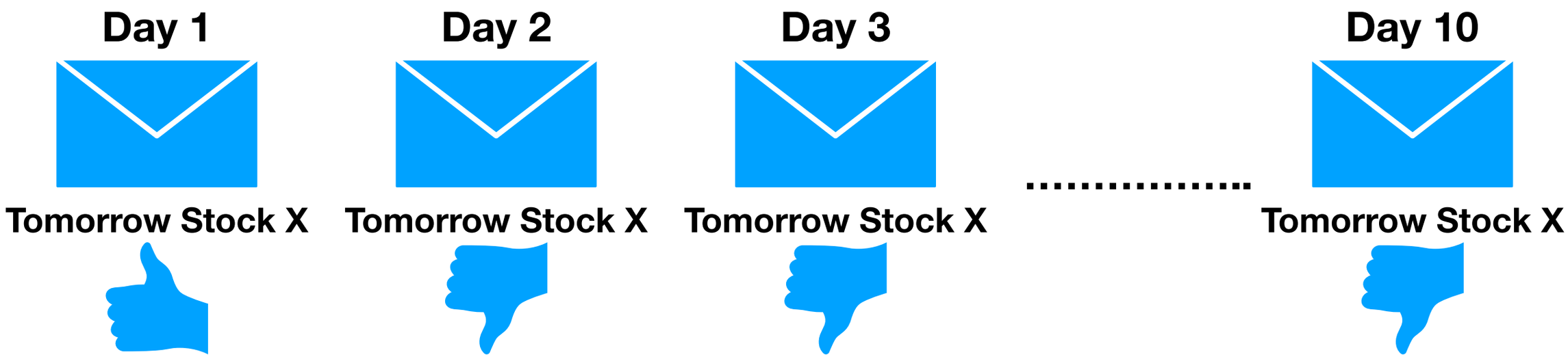
Based on Aaron Roth's slide



- Day 11: Pay me I will help you invest
- What are the chances that some one guesses randomly and gets correct?
  - 1 in 1024

The Magic Broker Scam

Based on Aaron Roth's slide



- Day 11: Pay me I will help you invest
- What are the chances that some one guesses randomly and gets correct?
  - 1 in 1024
- But the broker will always scam people, why?

The Magic Broker Scam

Based on Aaron Roth's slide

Day 1

Tomorrow Stock X

Day 2

Tomorrow Stock X

Day 3

Tomorrow Stock X

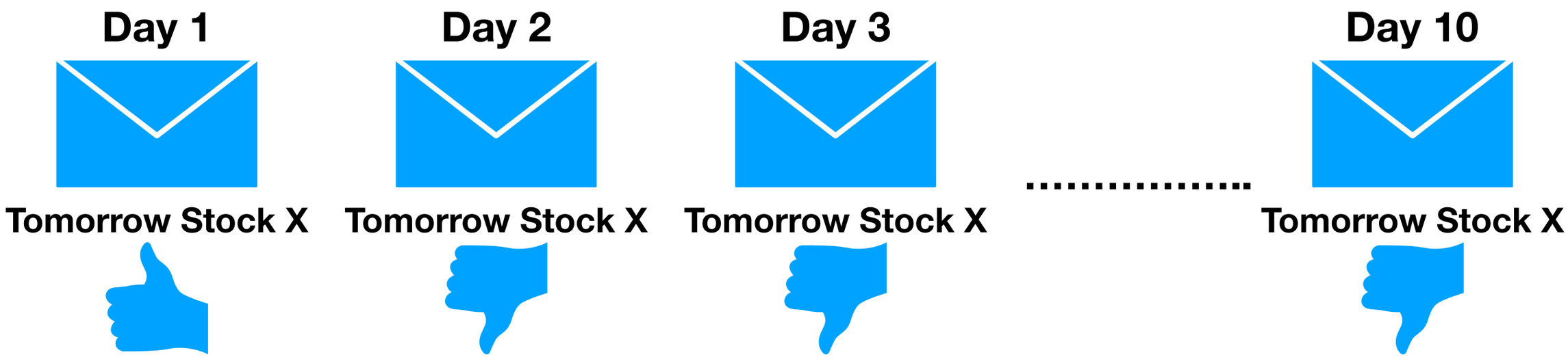
.....

Day 10

Tomorrow Stock X

The Magic Broker Scam

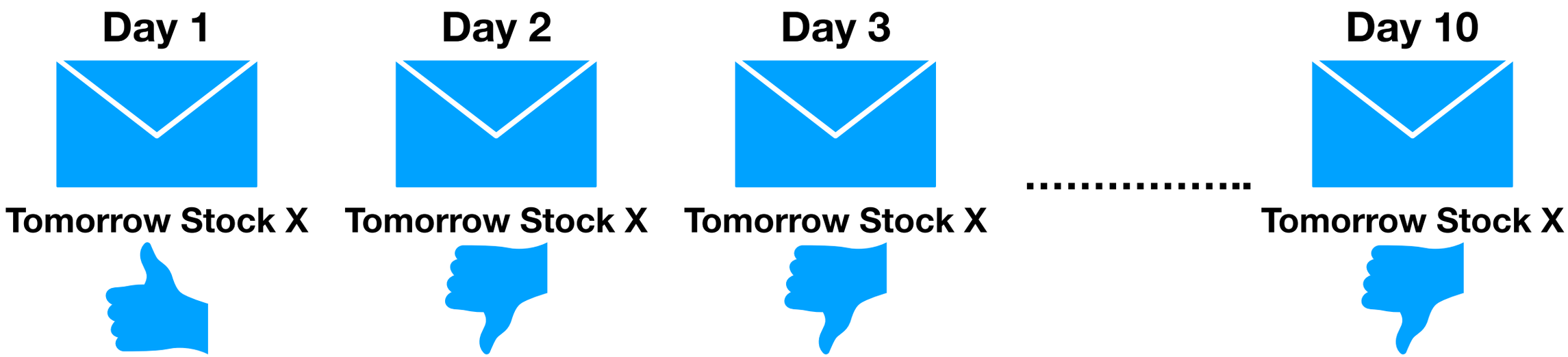
Based on Aaron Roth's slide



- Day 1: send email to a million people (1/2 + and 1/2 -)

The Magic Broker Scam

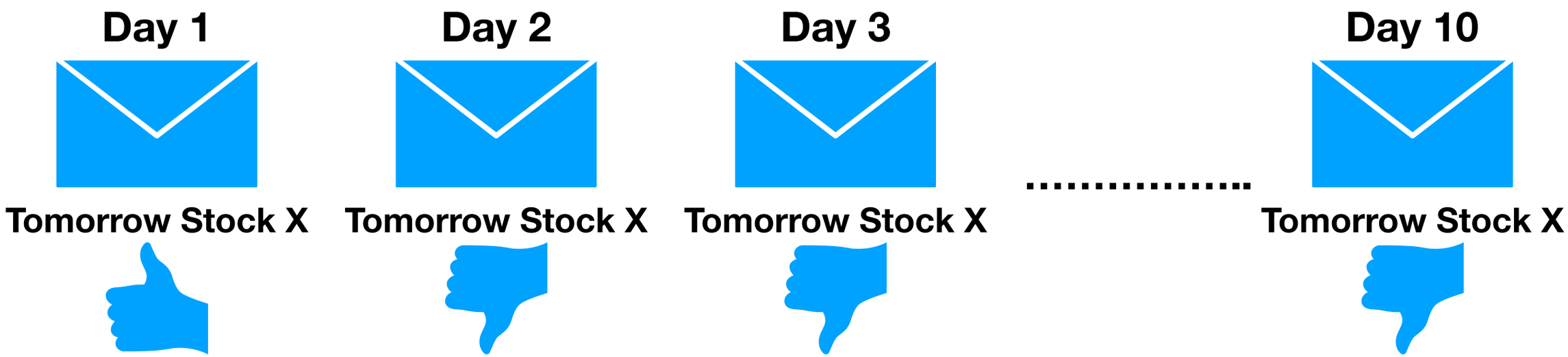
Based on Aaron Roth's slide



- Day 1: send email to a million people (1/2 + and 1/2 -)
- Day 2: only send email to 500,000 people we got right (1/2 + and 1/2 -)

The Magic Broker Scam

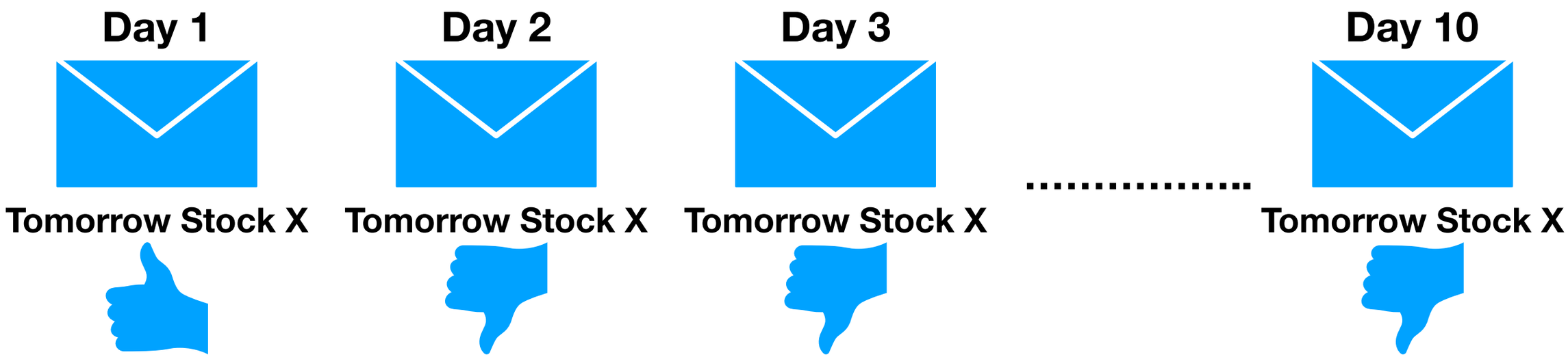
Based on Aaron Roth's slide



- Day 1: send email to a million people (1/2 + and 1/2 -)
- Day 2: only send email to 500,000 people we got right (1/2 + and 1/2 -)
- …..

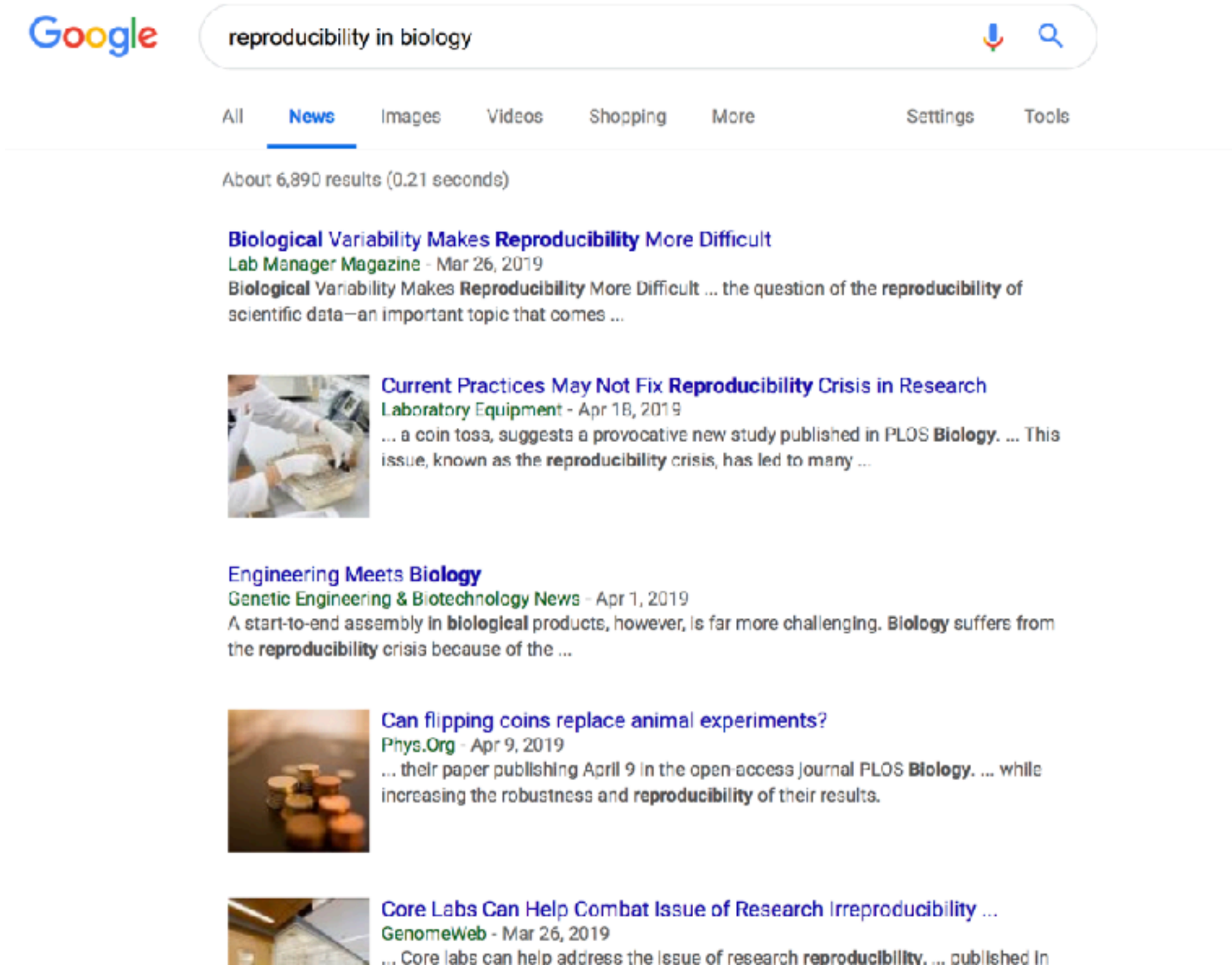
The Magic Broker Scam

Based on Aaron Roth's slide



- Day 1: send email to a million people (1/2 + and 1/2 -)
- Day 2: only send email to 500,000 people we got right (1/2 + and 1/2 -)
- …..
- Day 11: We are left with 1000000/1024 ~ 1000 people we can scam

Reproducibility problem

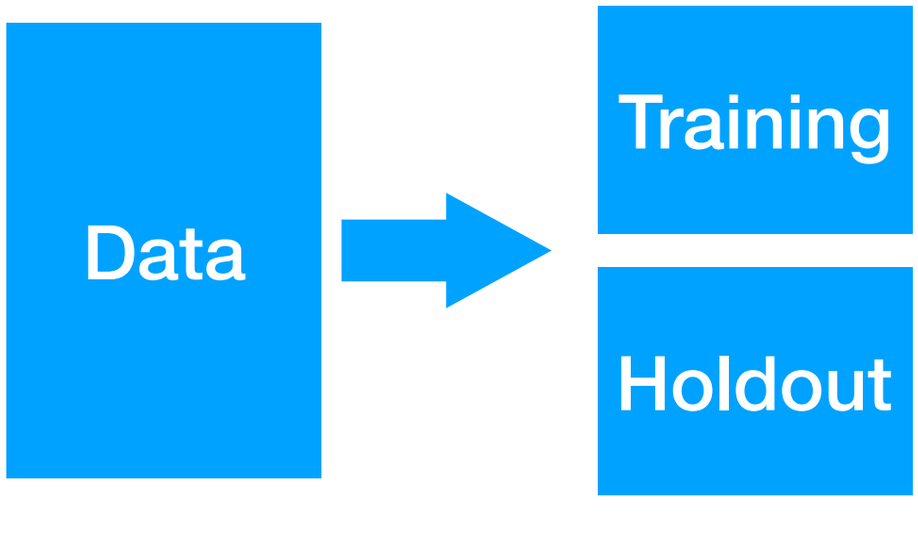


Ideal ML Experiment

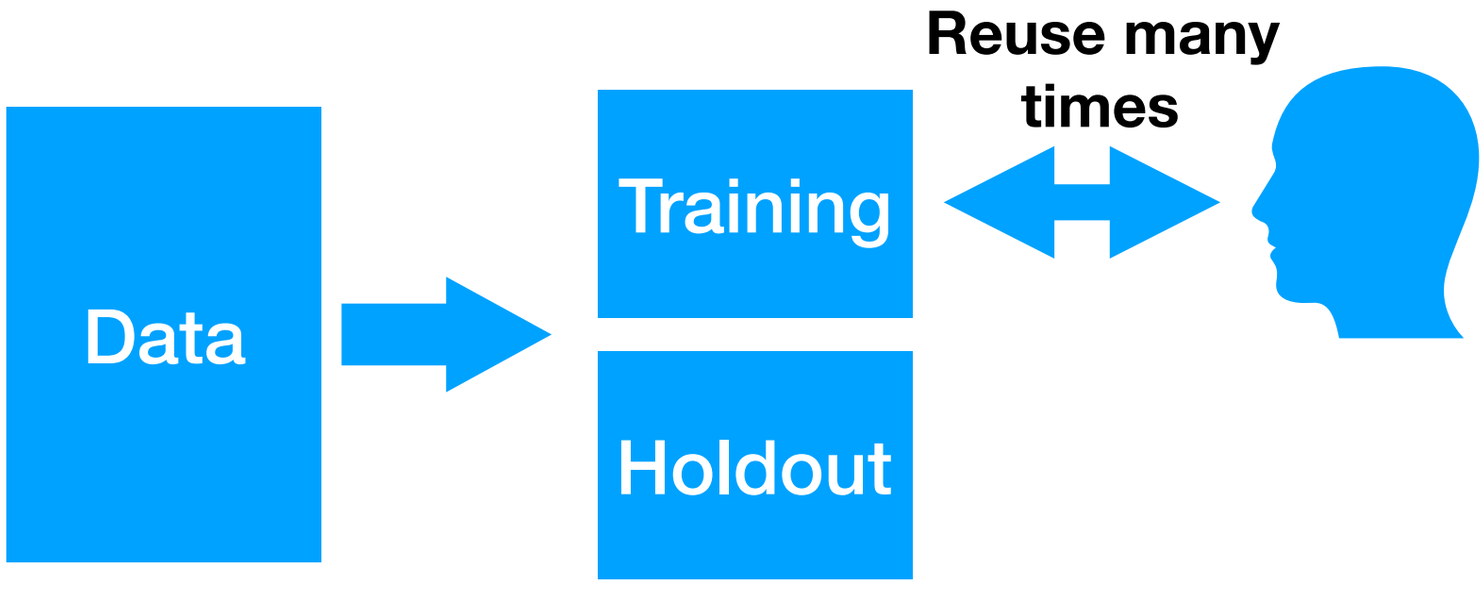
Ideal ML Experiment

Data

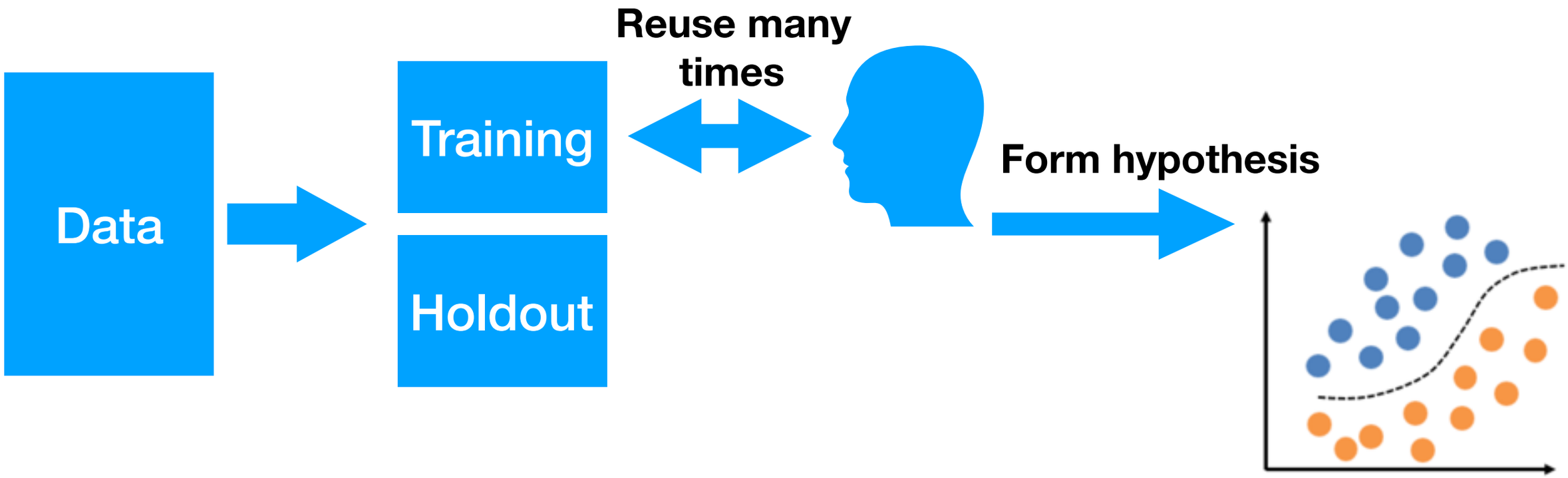
Ideal ML Experiment



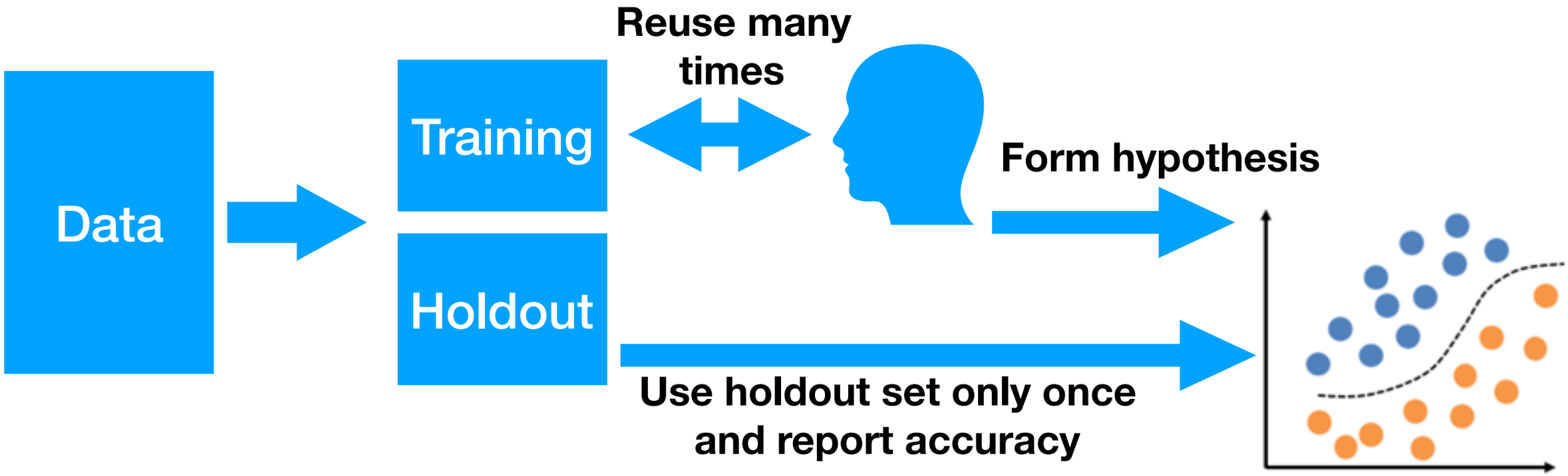
Ideal ML Experiment



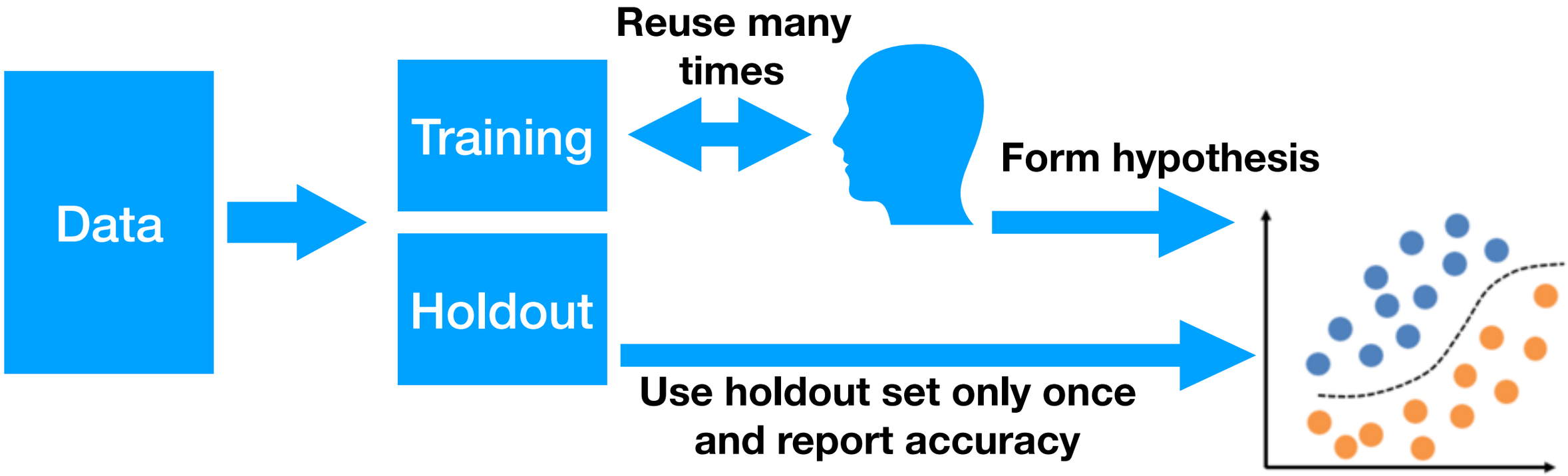
Ideal ML Experiment



Ideal ML Experiment

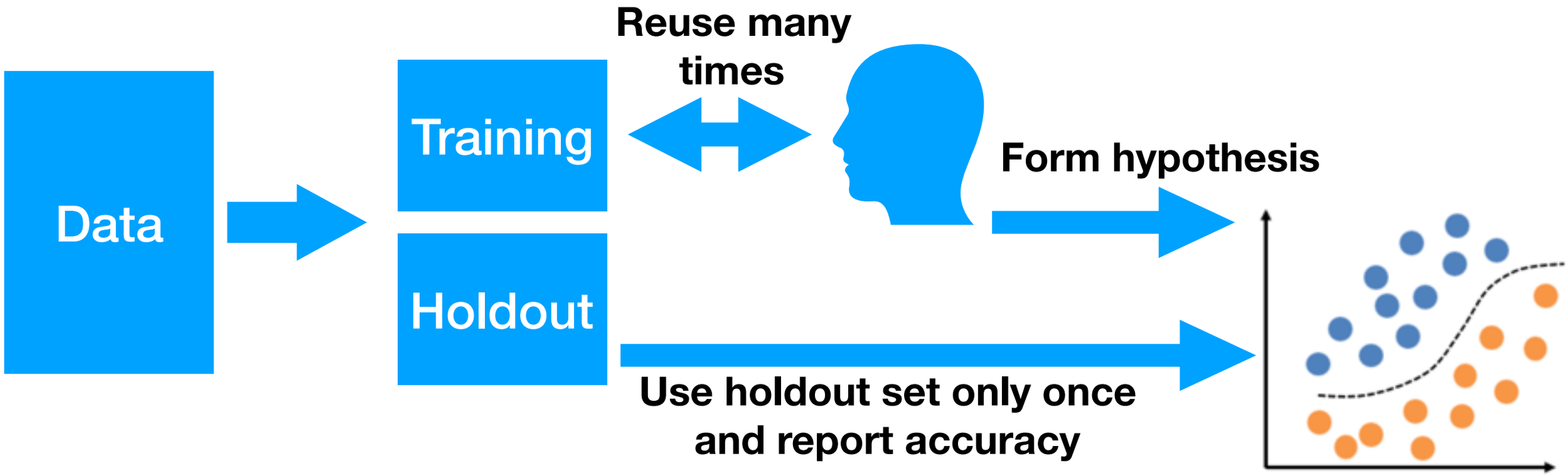


Ideal ML Experiment



- We are not allowed to form hypothesis based on data we used to test: age old statistics

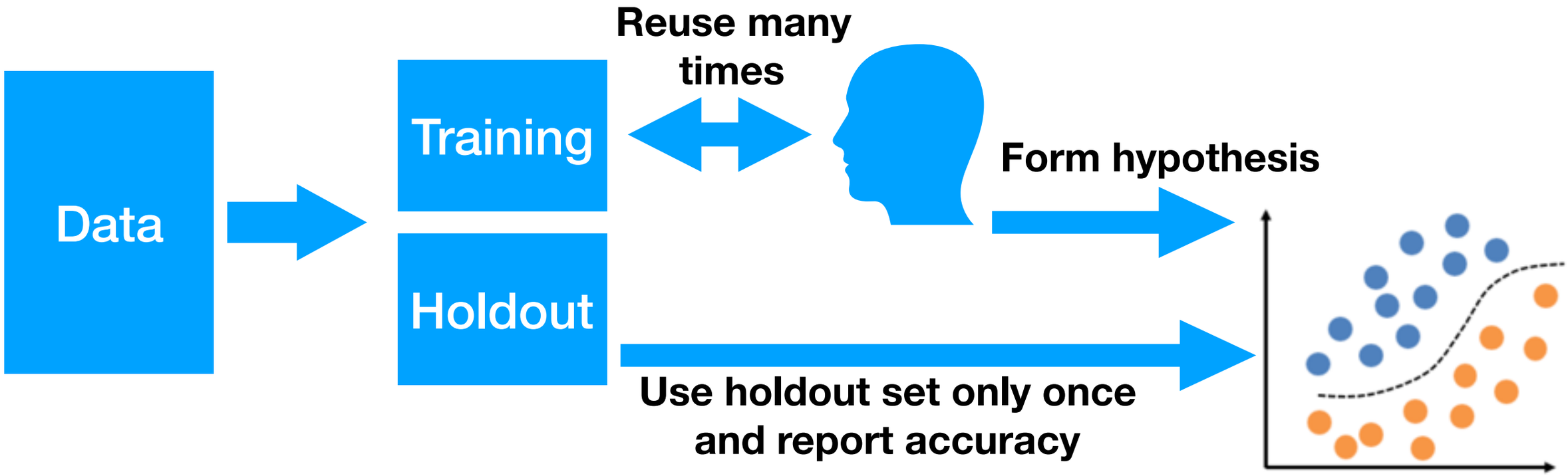
Ideal ML Experiment



- We are not allowed to form hypothesis based on data we used to test: age old statistics

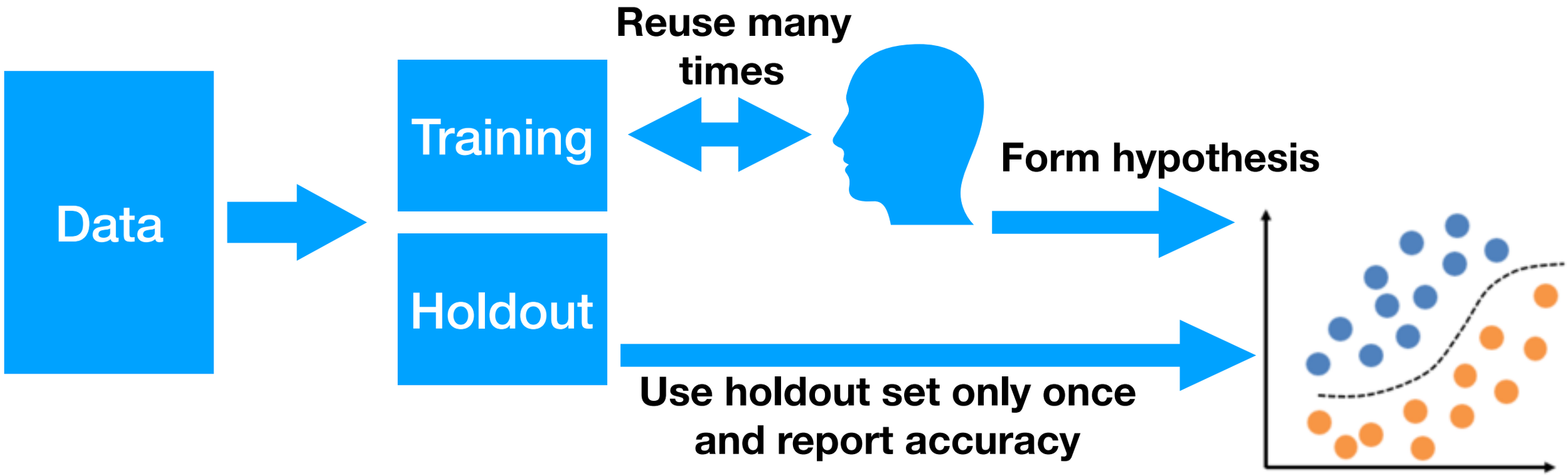
- But too tempting to form more informed opinion

Ideal ML Experiment



- We are not allowed to form hypothesis based on data we used to test: age old statistics
- But too tempting to form more informed opinion
- We do a train/validation/test set split

Ideal ML Experiment



- We are not allowed to form hypothesis based on data we used to test: age old statistics
- But too tempting to form more informed opinion
- We do a train/validation/test set split
- But this means we can’y reuse datasets over time

ML Today

ML Today

- Benchmark dataset from MNIST to IMAGENET

ML Today

- Benchmark dataset from MNIST to IMAGENET
- Competitions run with feedback from test set to competitors ;)

ML Today

- Benchmark dataset from MNIST to IMAGENET
- Competitions run with feedback from test set to competitors ;)
- Effort to collect public dataset to share….

ML Today

- Benchmark dataset from MNIST to IMAGENET
- Competitions run with feedback from test set to competitors ;)
- Effort to collect public dataset to share….
- All great but we need to be careful!

ML Today

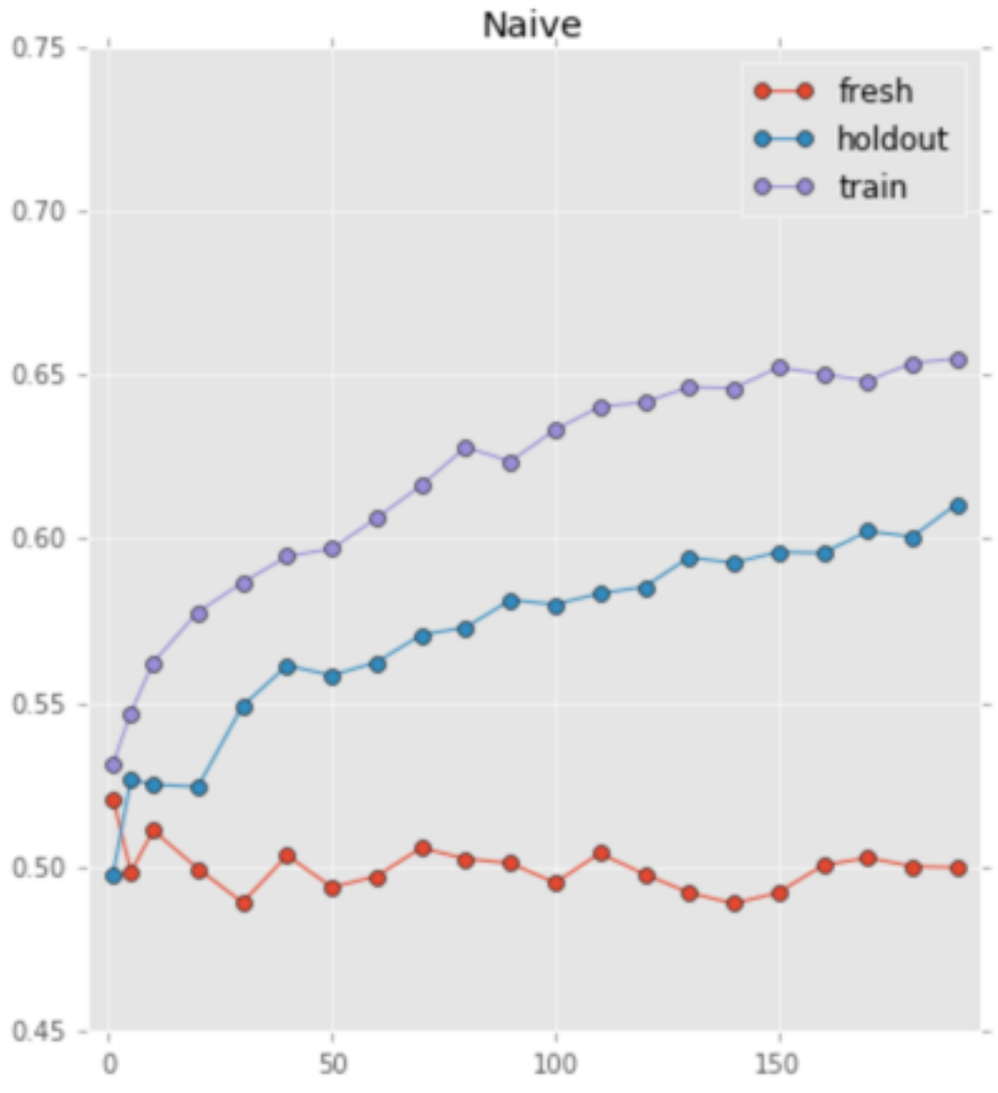
- Benchmark dataset from MNIST to IMAGENET
- Competitions run with feedback from test set to competitors ;)
- Effort to collect public dataset to share….
- All great but we need to be careful!
    - Old fix: pre-register experiment, many many examples of people fudging this….

An Example

Code at:
https://github.com/isofer/thresholdout-experiments/blob/master/Thresholdout%20experiments.ipynb

- Data: 20,000 data points in 10000 dimensions drawn randomly
- Labels: random +1 or -1 (no correlation with input)
- Data-scientist runs:
  - Split data into train and test of equal size
  - Select best k features on training data
  - Only use variables also good on holdout
  - Build linear predictor out of these k variables
  - Find best k = 10,20,30,40,….

An Example


Naive
fresh
holdout
train
0.75
0.70
0.65
0.60
0.55
0.50
0.45
0
50
100
150
Number of features

Can we reuse holdout set?

Maybe if we do it right…

Maybe if we do it right…

- Idea: when we report back accuracies from the dataset, we add noise so as to not to leak too much information

Maybe if we do it right…

- Idea: when we report back accuracies from the dataset, we add noise so as to not to leak too much information

- Eg. Report back accuracies on holdout set only when training and test accuracy are significantly different

Threshold-out Algorithm

Input:

Data S, holdout H, threshold $T > 0$, tolerance $\sigma > 0$

Given function q:

Sample η, η' from $N(0,\sigma^2)$

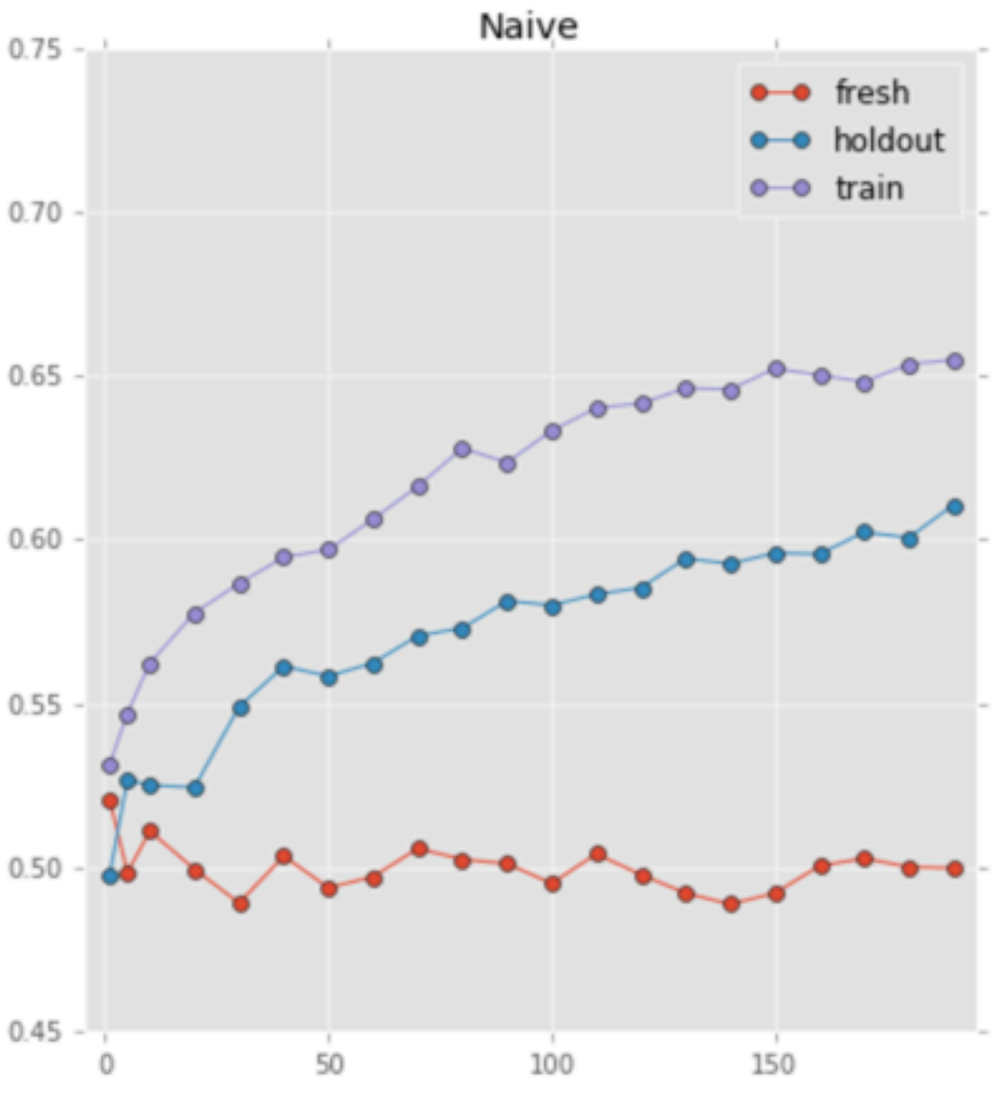
If $|avg_H[q] - avg_S[q]| > T + \eta$:

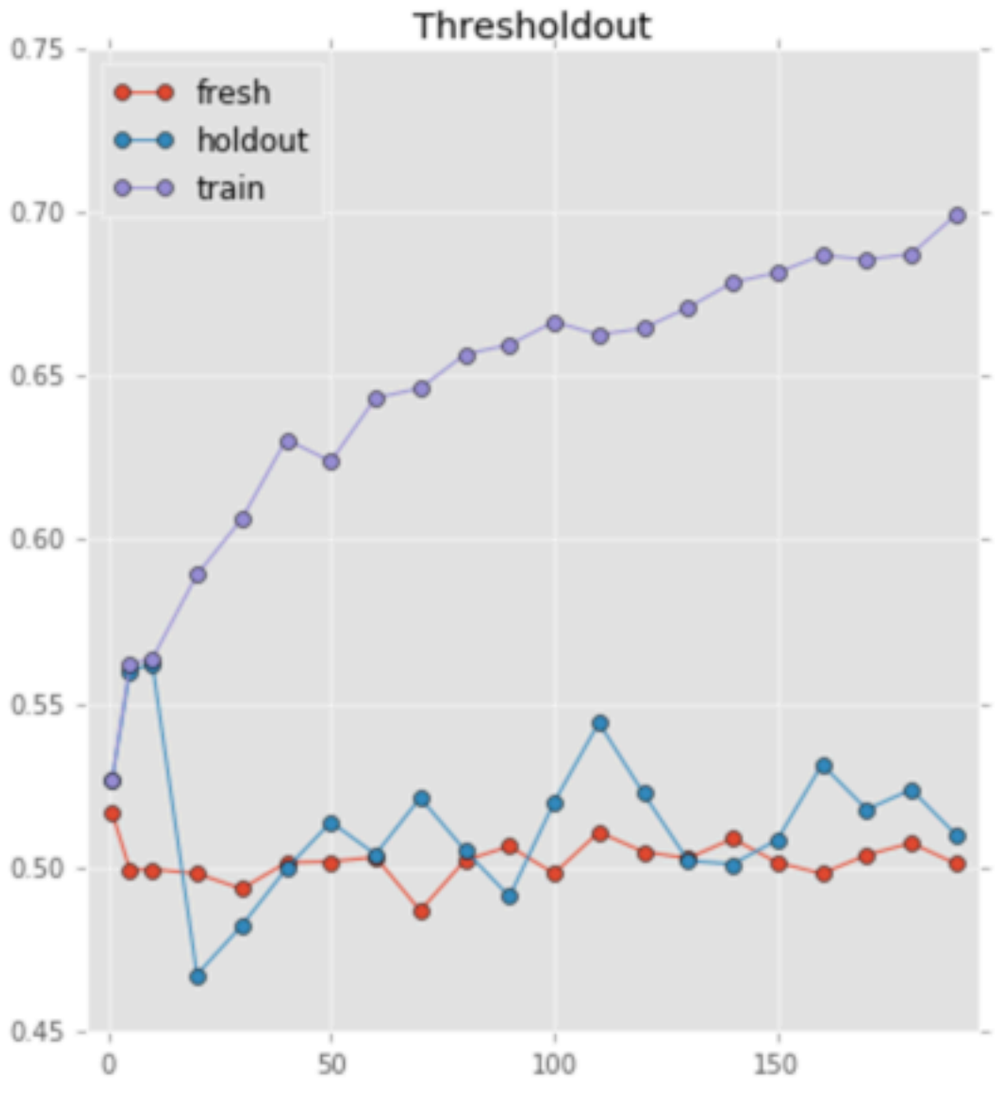
output $avg_H[q] + \eta'$

Otherwise:

output $avg_S[q]$

The Example


Naive
fresh
holdout
train
0.75
0.70
0.65
0.60
0.55
0.50
0.45
0
50
100
150


Thresholdout
fresh
holdout
train
0.75
0.70
0.65
0.60
0.55
0.50
0.45
0
50
100
150

The Guarantee

- Roughly: Thresholdout guarantees high accuracy, and can handle up to *asked is order n^2 over-fitting queries*

Differential Privacy

Differential Privacy

- A deterministic algorithm cannot preserve privacy

Differential Privacy

- A deterministic algorithm cannot preserve privacy
- Say S = ($Data_1$,…,$Data_n$) is the data provided to learning algorithm (be it clustering, supervised learning etc).

Differential Privacy

- A deterministic algorithm cannot preserve privacy
- Say S = ($Data_1$,…,$Data_n$) is the data provided to learning algorithm (be it clustering, supervised learning etc).
- Say (randomized) learning algorithm A takes this training data and returns solution as A(S)

Differential Privacy

- A deterministic algorithm cannot preserve privacy

- Say S = ($Data_1$,…,$Data_n$) is the data provided to learning algorithm (be it clustering, supervised learning etc).

- Say (randomized) learning algorithm A takes this training data and returns solution as A(S)

- Algorithm A is (ϵ,δ)- differentially private if for all samples S and S' that only differ by one data point and any set C

$$P(A(S) \in C) \leq e^{\epsilon} P(A(S') \in C) + \delta$$

Differential Privacy

- A deterministic algorithm cannot preserve privacy

- Say S = ($Data_1$,…,$Data_n$) is the data provided to learning algorithm (be it clustering, supervised learning etc).

- Say (randomized) learning algorithm A takes this training data and returns solution as A(S)

- Algorithm A is (ϵ,δ)- differentially private if for all samples S and S' that only differ by one data point and any set C

$$P(A(S) \in C) \leq e^{\epsilon} P(A(S') \in C) + \delta$$

- δ=0 is called pure differential privacy

Differential Privacy

- Post-processing: If A is a differentially private algorithm, and f is any mapping on the outcome space of A, then the algorithm that maps from sample S to f(A(S)) is also differentially private

- Composability: Algo(S) is given by following procedure
    - For i in 1 to k
        - If we choose any epsilon differentially private algorithm A_i based on outcomes O_1,…,O_{i-1}
        - Set O_i = A_i(S)
    - Return O_1,…,O_n
- The above algorithm is O(epsilon sqrt(k)) differentially private