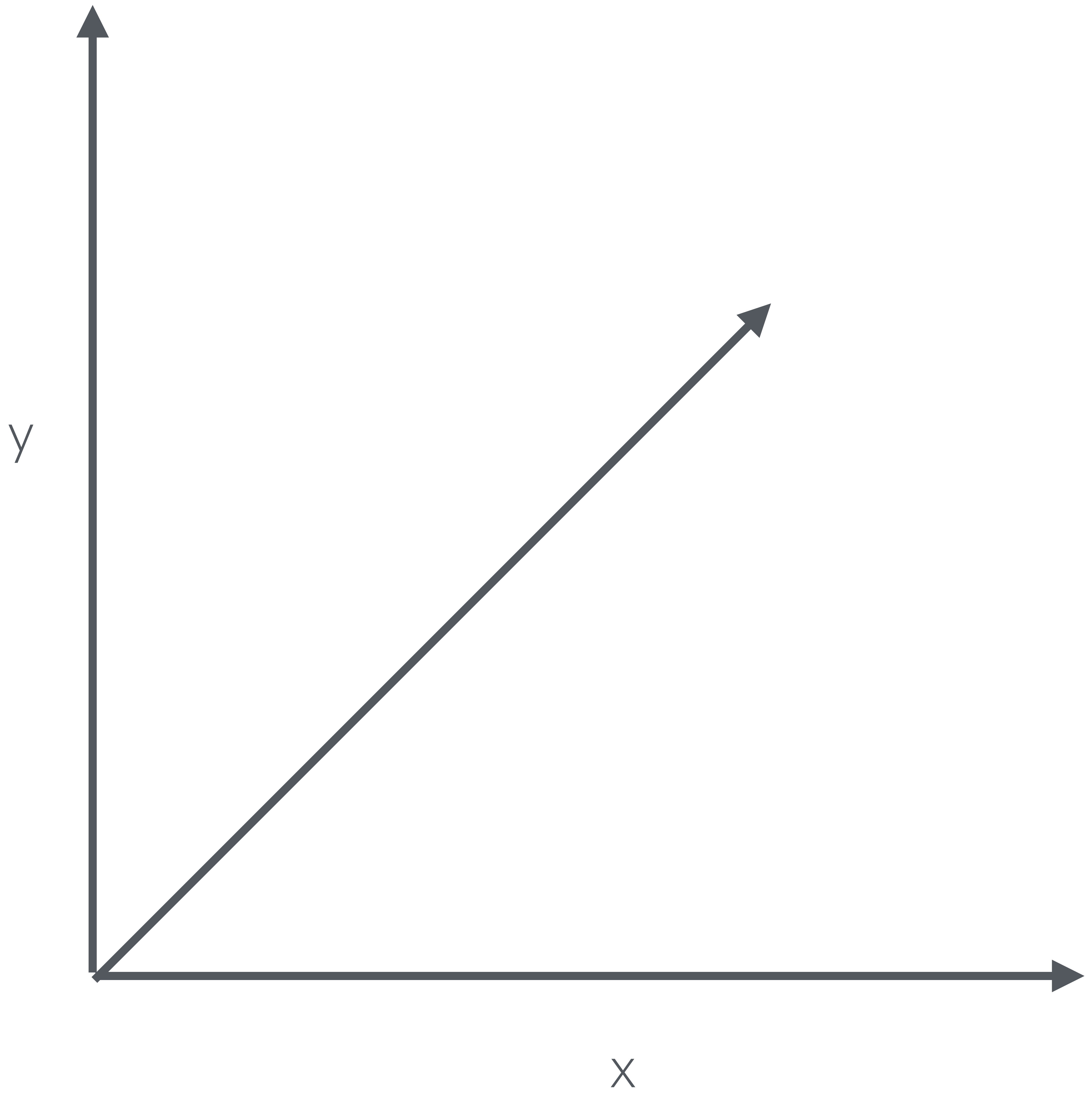
Module 9: Contextual Bandits

DAV-6300-1: Experimental Optimization

David Sweet // 20241107

Notation: "Proportional to"

- $y = kx$
- $y \propto x$



Review: Thompson sampling

- Allocate observations to arms in proportion to the probability each arm is best
    - $p_{\text{arm}} \propto p_{\text{best}}$
- Stop when $\max\{p_{\text{best}}\} > 0.95$

Review: Predictor-in-controller

- Predictor: Estimates a target, ex., P{click} on an ad
- Controller: Uses predictions to make a decision / choose an action
  - Ex., “Of the 1000 ads available, show the one with the highest P{click}”

Industrial engineered systems

Predictors in controllers

Controller	Prediction	Action	Reward
Ad server	P{click}	Show ad with highest P{click}	CPC revenue
Fraud detector	P{fraudulent}	Hold charges with high P{fraudulent} until customer gives OK	Avoid losing money to fraud
Trading strategy	E[return]	Buy when E[return] > 0, sell when E[return] < 0	Revenue (“PnL”)
Social media feed	P{like}	Show posts with highest P{like}	Users spend more time on feed

Production logs

- Every time you show an ad, log
  - features of the ad
  - features of the user
  - whether the user clicked
- data = $\{(x_i, y_i)\}$, where
  - x_i = all of the features
  - y_i = 1 if clicked, else 0; “click indicator”

Typical design

Estimate P{click}

- Fit an SL model to the data, like
  - Logistic regression
  - Neural network
- Model may have many parameters
- Model estimates P{click} = click-through-rate = CTR
- More precisely: P{click | ad & user features} = CTR for ad

Typical design

Periodic refitting

- Fit model every day
- Use data from trailing month's logs
- Production uses latest, refit model
- Refitting tracks changes in system over time

Problem: Variance

- Day 1: Show Ads A & B 100 times each
  - 10/100 clicks on A from NYC
  - 5/100 clicks on B from NYC
- Over night: Fit a regression
  - $P\{\text{click} \,|\, NYC, A\} = .10$
  - $P\{\text{click} \,|\, NYC, B\} = .05$
- Day 2: Always show ad A to NYC users.

NYC is context

A,B are arms or “actions”

Problem: Variance

- Better
  - $P\{\text{click} \mid NYC, A\} = .10 \pm .07$
  - $P\{\text{click} \mid NYC, B\} = .05 \pm .03$
- Is $E[\text{click} \mid NYC, A] > E[\text{click} \mid NYC, B]$?

Problem: Missing counterfactuals

- Just by chance, no measurements in fourth box
- *Counterfactual*: What would have happened if we had taken another action?

	NYC	**LA**
A	10/100	20/100
B	5/100	

Problem: Missing counterfactuals

- Regression predicts the last box
- No actual data for fourth box
- Day 2: Show only Ad A to LA users

	NYC	**LA**
A	10/100	20/100
B	5/100	*10/100 ?*

Problem: Missing counterfactuals

- Oops. Model was wrong.
- Missed opportunity

	NYC	**LA**
A	10/100	20/100
B	5/100	**30/100**

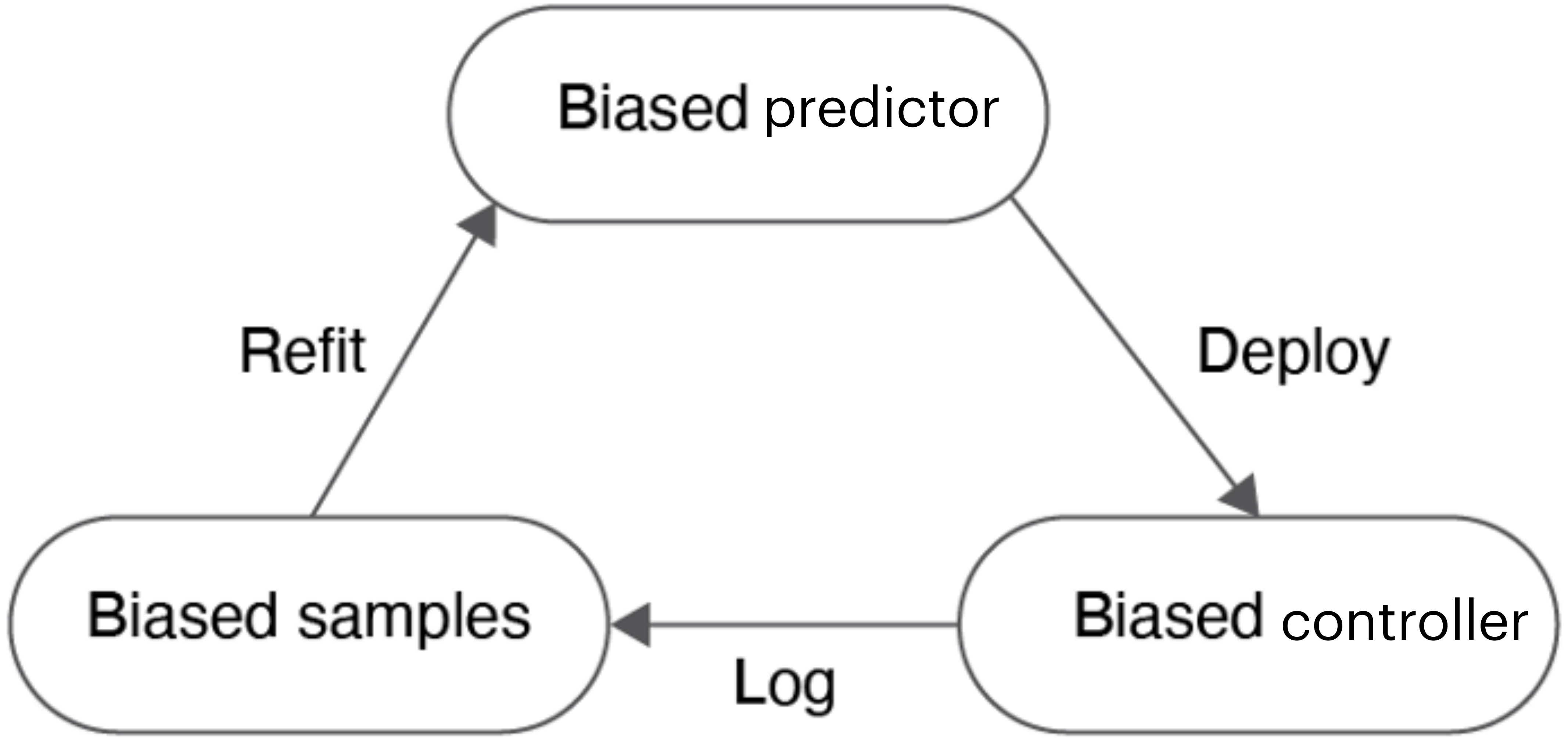
Small-Sample Bias

- Smaller samples have larger variance
- Smaller sampler have larger biases
- Predictor (e.g., regression) inherits/learns bias
- Only more data can fix it.

More *independent, identically distributed* data, that is

Small-Sample Bias

- Biased predictor ==> Biased controller
- Endless feedback loop



Solution: Explore Arms

- Epsilon-greedy ad selection:
  - Exploit: w/probability 0.90, use the predictor
  - Explore: w/probability 0.10, show an ad at random
- Exploration collects counterfactuals
- Exploration collects everything, actually

Biased

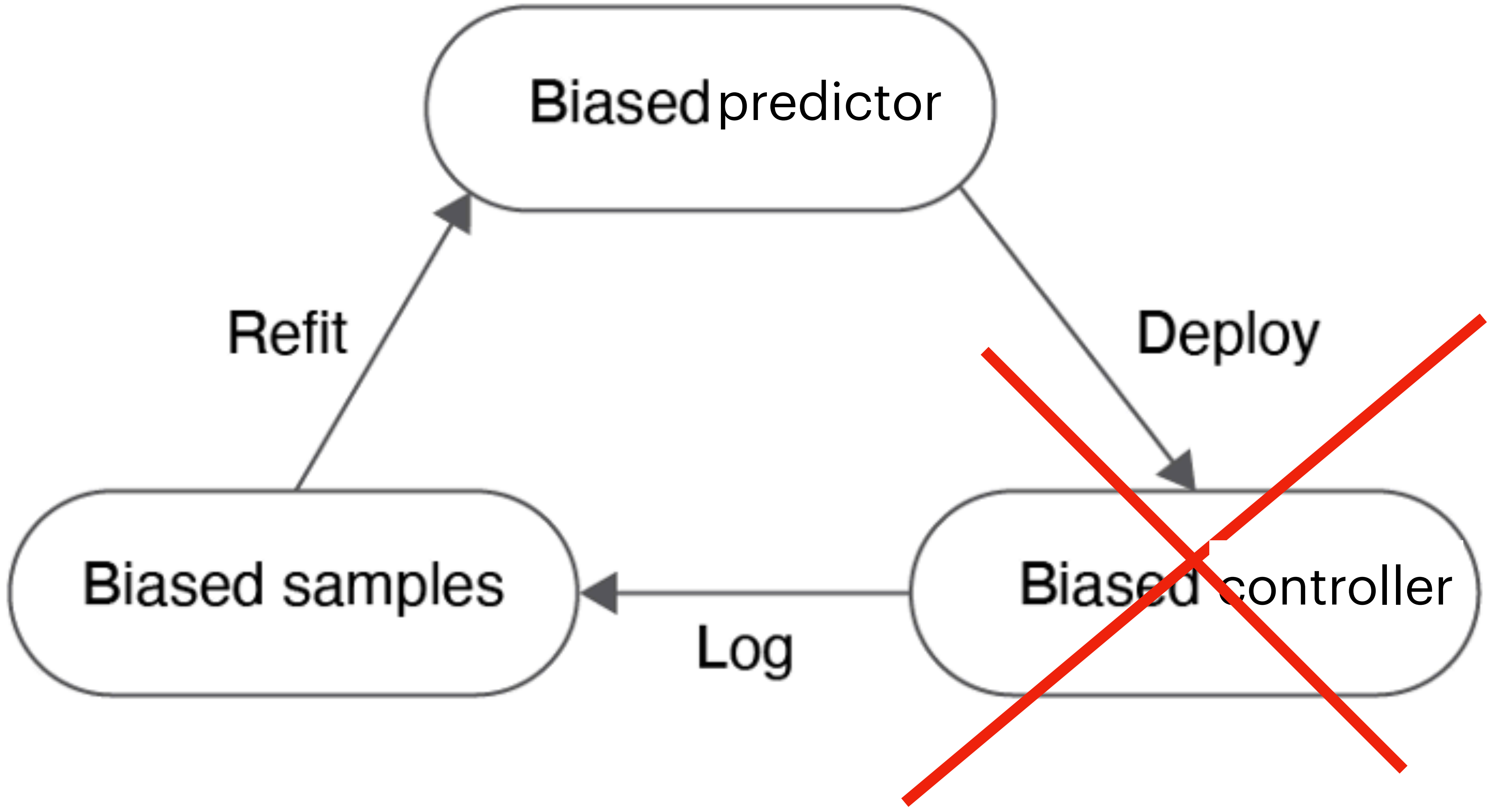
Unbiased

Reduces bias

Reduces variance

Solution: Explore Arms

- Breaks the feedback loop
- Controller not (totally) biased



Contextual bandits

- Can optimize many — millions — of parameters of predictor
- Only for short-term business metrics, aka *rewards*
  - Ex: CTR, likes, fraudulent transaction
  - But **not**: DAU, daily pnl, purchase following ad, time spent per day

Contextual bandit

- Each ad is an arm
- MAB: Models $\boldsymbol{\mu}$, $\boldsymbol{se}$
- CB: $\boldsymbol{\mu} \mid$ context, $\boldsymbol{se} \mid$ context
- *Context*: Features of ad and user
  - Hence the name *contextual bandit*

Controller classes

	No Exploration	Exploration
No Prediction	Naive Show ad with best CTR so far	MAB Explore to get unbiased data
Prediction	P-in-C Predict CTR from context	CB Explore & predict

Thompson Sampling

- TS works in CB, too:
  - MAB: $m_a \sim \mathcal{N}(\mu_a, se_a^2)$
  - MAB: $m_a = \mu_a + se_a \times \varepsilon, \quad \varepsilon \sim \mathcal{N}(0,1)$
  - CB: $m_a(c) = \mu_a(c) + se_a(c) \times \varepsilon$
- Draw ε once for each arm
- Explores arms

Bootstrap Thompson Sampling

- Generate B bootstrap samples of data
- Fit B models; an *ensemble* of models
- To show an ad:
    - Choose 1 model from ensemble, randomly
    - Use that model to decide which ad to show
- Model more uncertain ==> more exploration ==> more certain model tomorrow

Short-term business metrics only

RHLF is a CB

- Wow: CB can optimize millions of parameters
- Catch: CB only works with short-term business metrics (rewards)
  - Ex: CTR, likes, fraudulent transaction
  - But **not**: DAU, daily pnl, purchase following ad, time spent per day
- CB needs (features, target) pairings and many samples in data set
  - Ex: DAU is 1 number/day, even though many, many ads shown

Summary

- Perspective 1: CBs fix feedback loops that bias predictor-in-controller designs by adding exploration
- Perspective 2: CBs improve MAB decisions by conditioning on context, i.e. adding a prediction model
- Contextual bandits use exploration to collect unbiased data
- Bootstrap Thompson sampling explores models
- Contextual bandits enable optimization of many parameters,
  but only for short-term business metrics