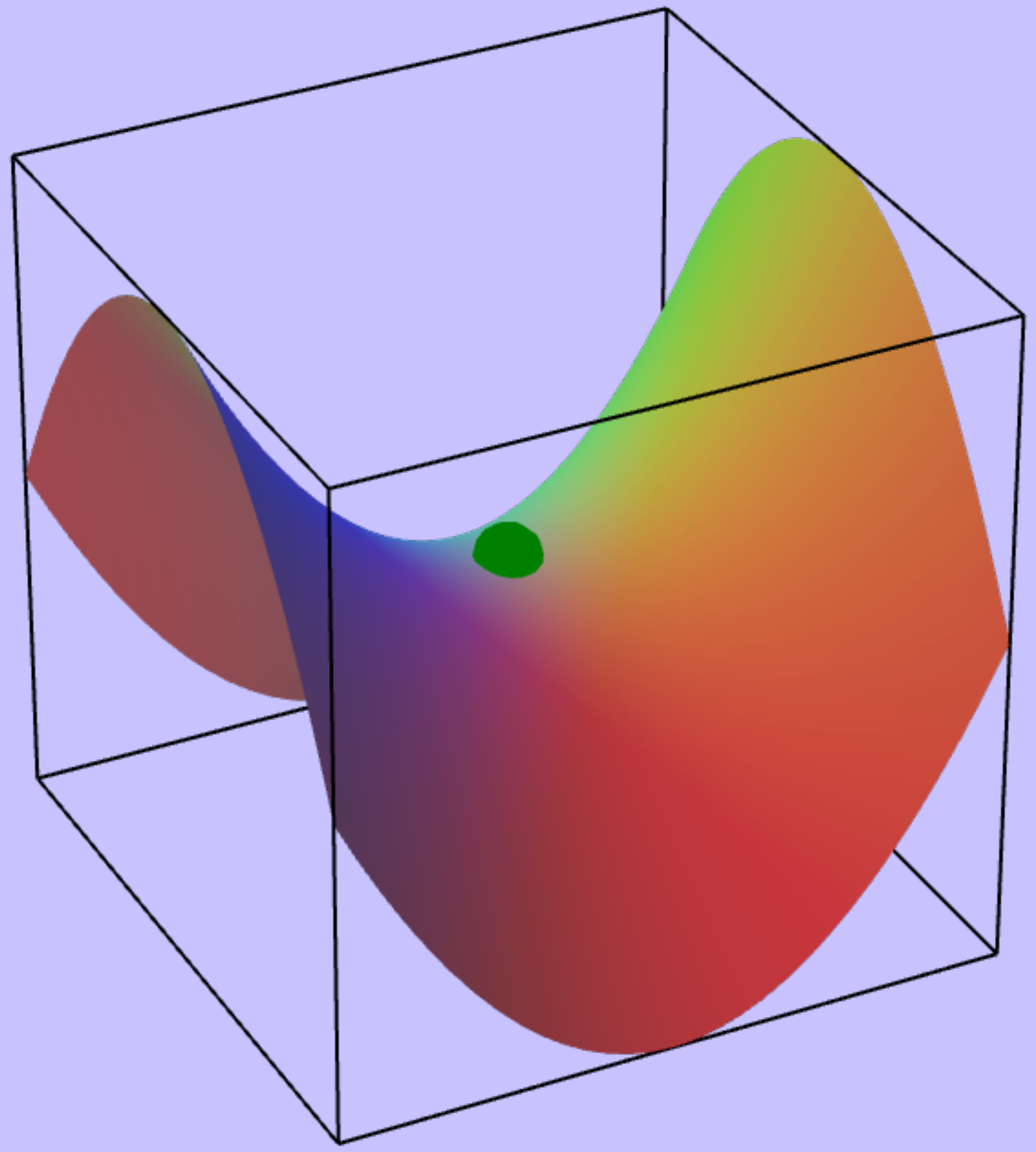
DRAGO

Primal-Dual Coupled Variance Reduction for Faster Distributionally Robust Optimization



NeurIPS 2024

Team

Ronak Mehta
University of Washington

Jelena Diakonikolas
University of Wisconsin-Madison

Zaid Harchaoui
University of Washington

Institute for Foundations of Data Science

UNIVERSITY of WASHINGTON

Problem Setting

Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Problem Setting

Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$

Problem Setting

Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$

Problem Setting

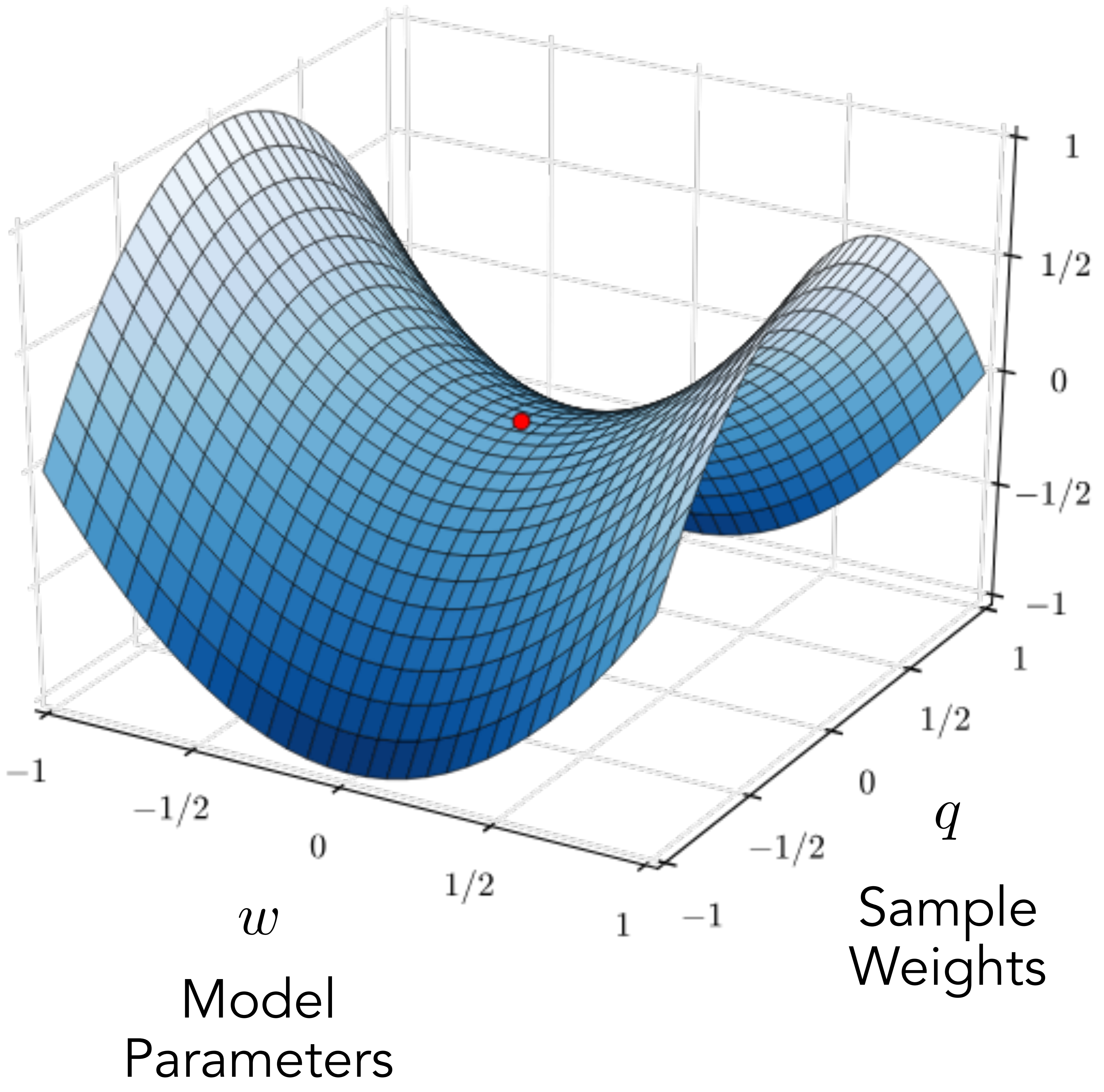
Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$

Nonbilinearly Coupled
Saddle Point Problem



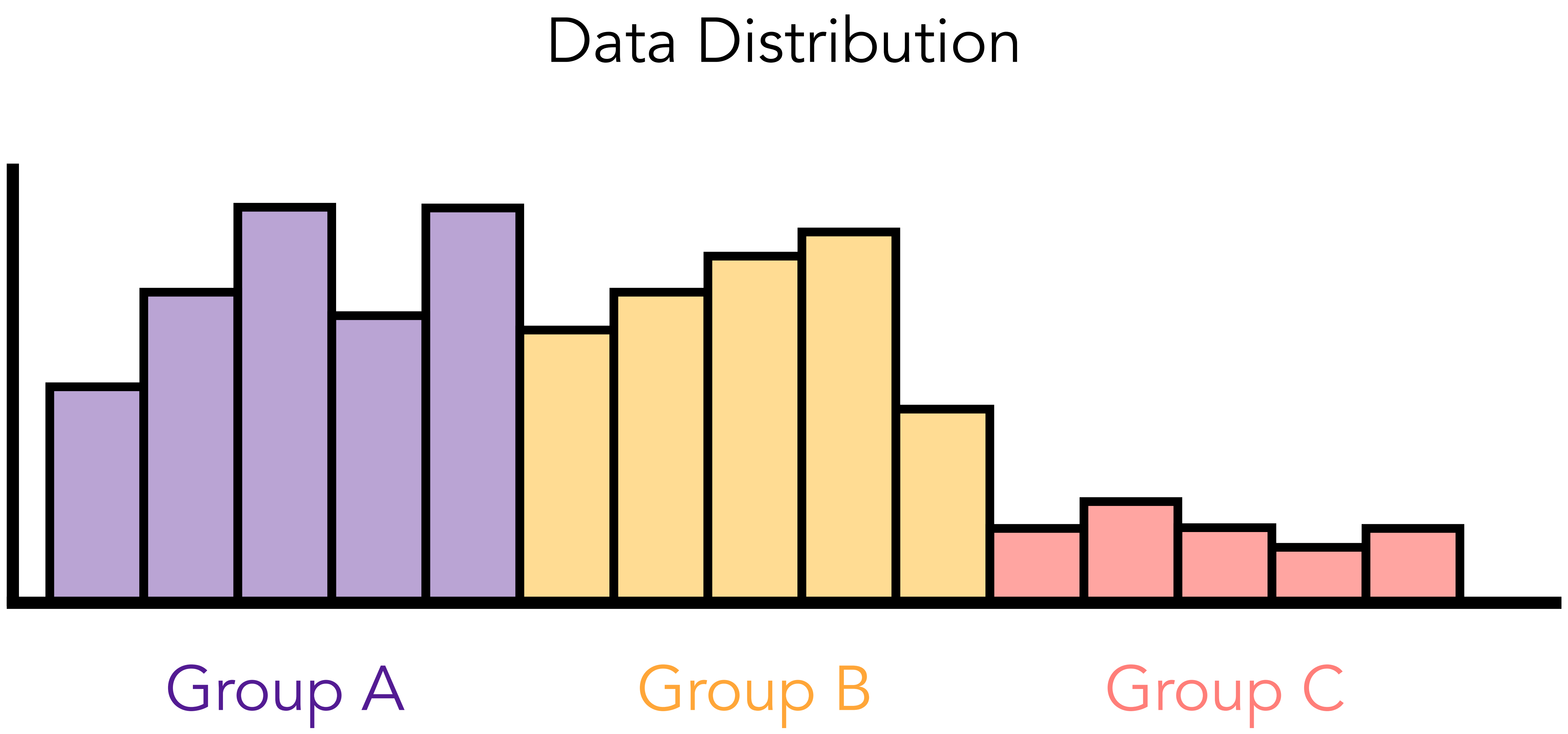
Problem Setting

Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$



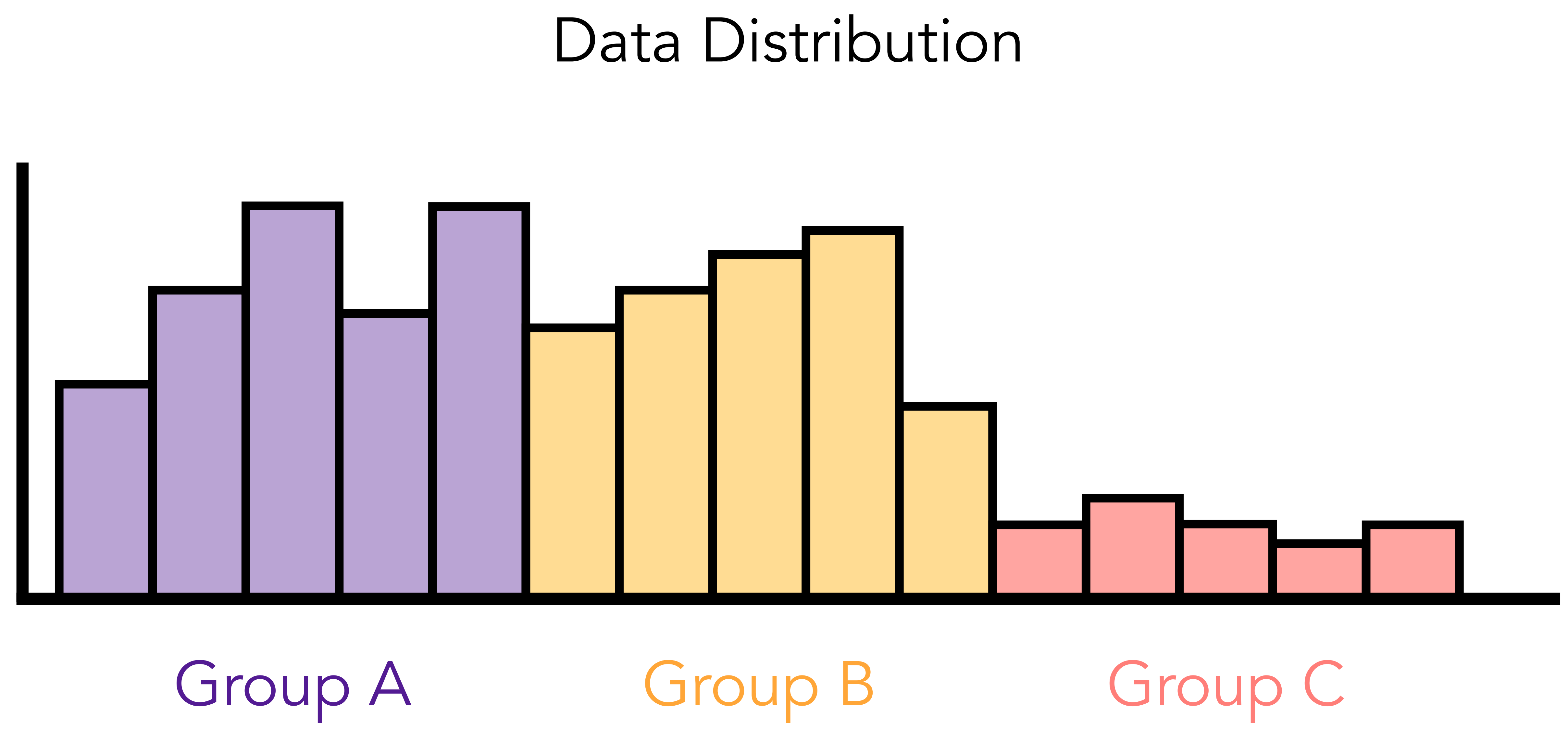
Problem Setting

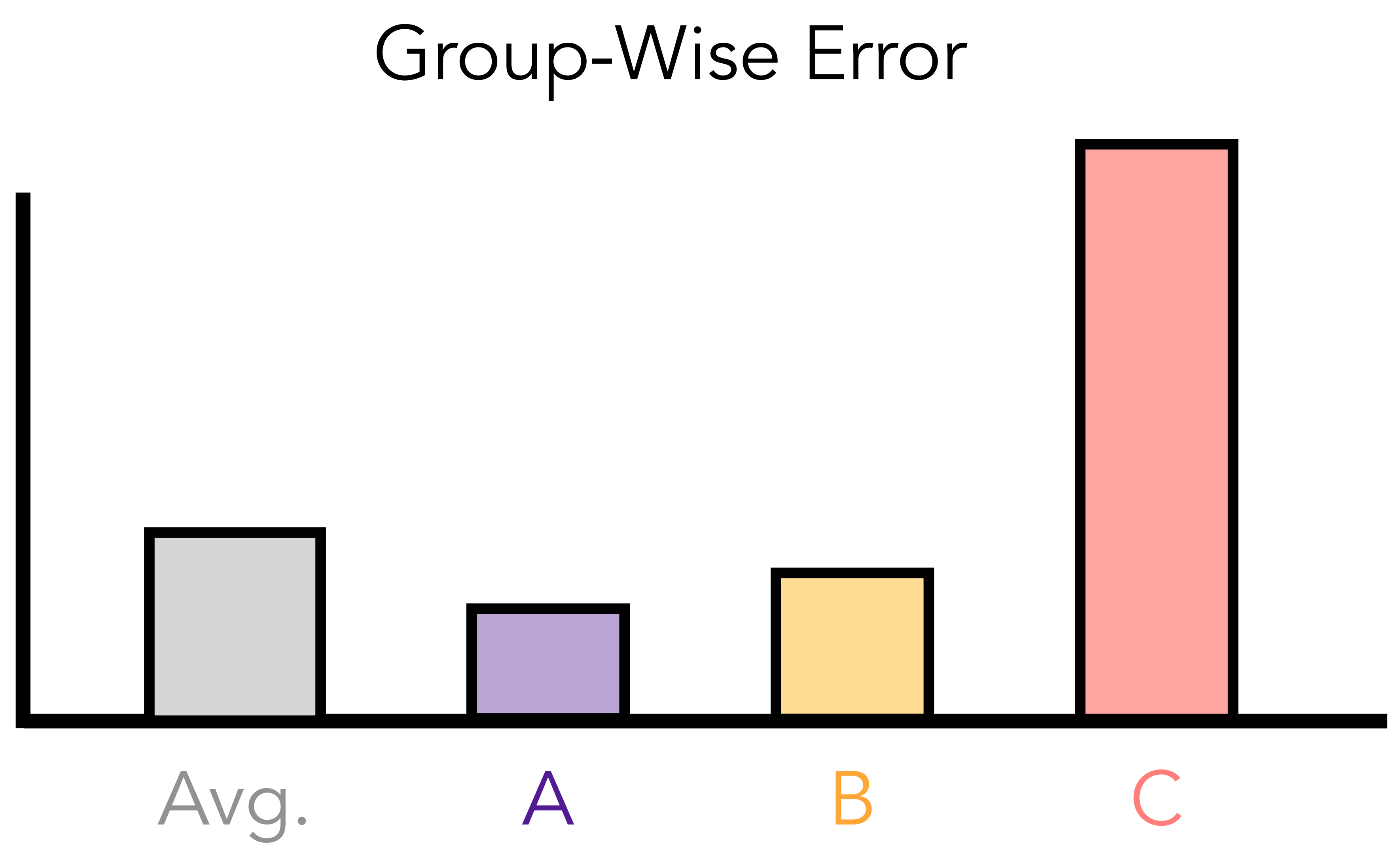
Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$





Problem Setting

Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$

A direct approach: **Stochastic Gradient Descent (SGD):** Estimate gradient of objective with a mini-batch of size m, which is biased unless $m = n$.

$$\Longrightarrow \quad \nabla \left[\max_{q \in \hat{Q}_m} \sum_{j=1}^{m} q_j \ell_{i_j}(w) - \nu D(q \| \mathbf{1}/m) \right] + \mu w$$

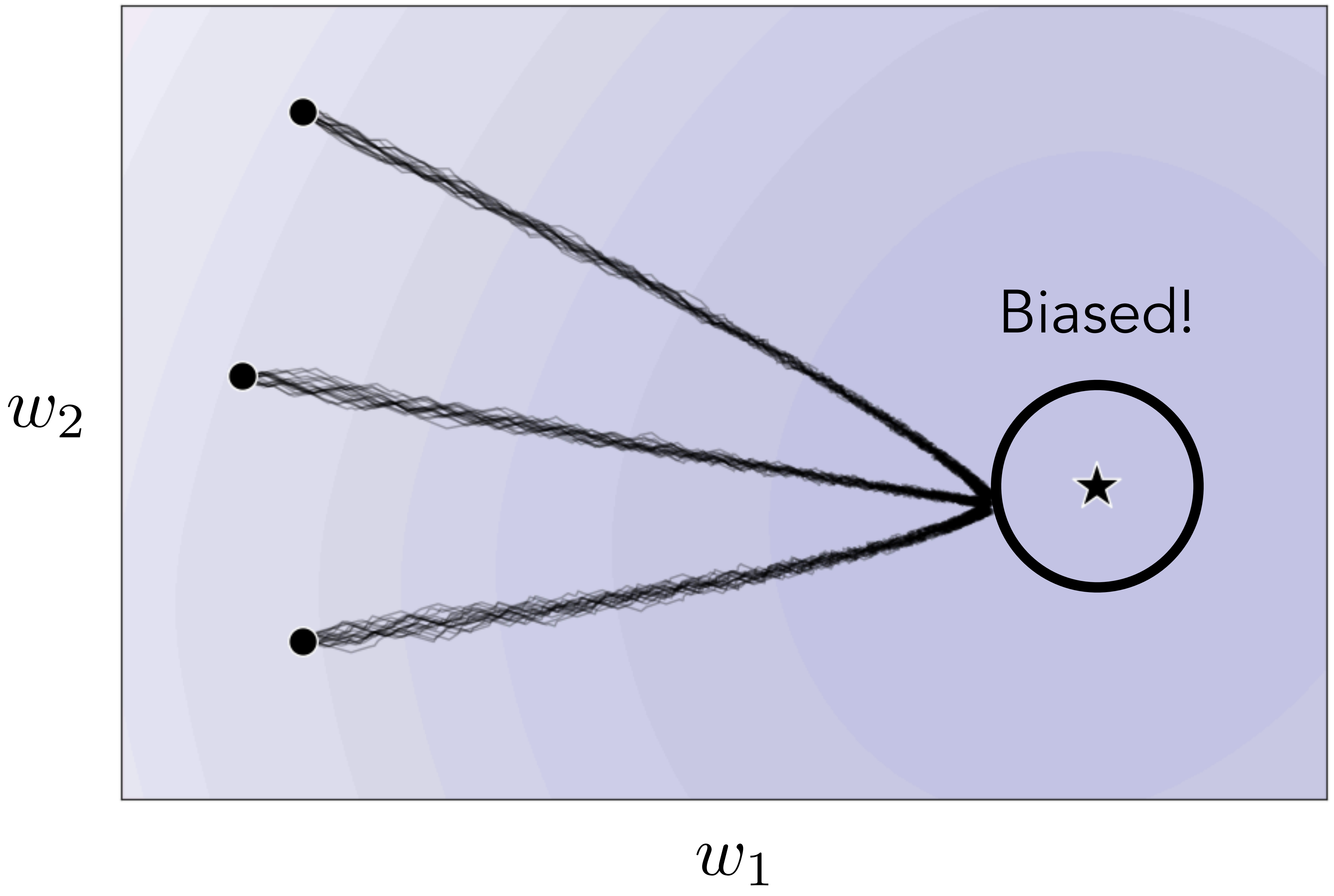
Problem Setting

Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$



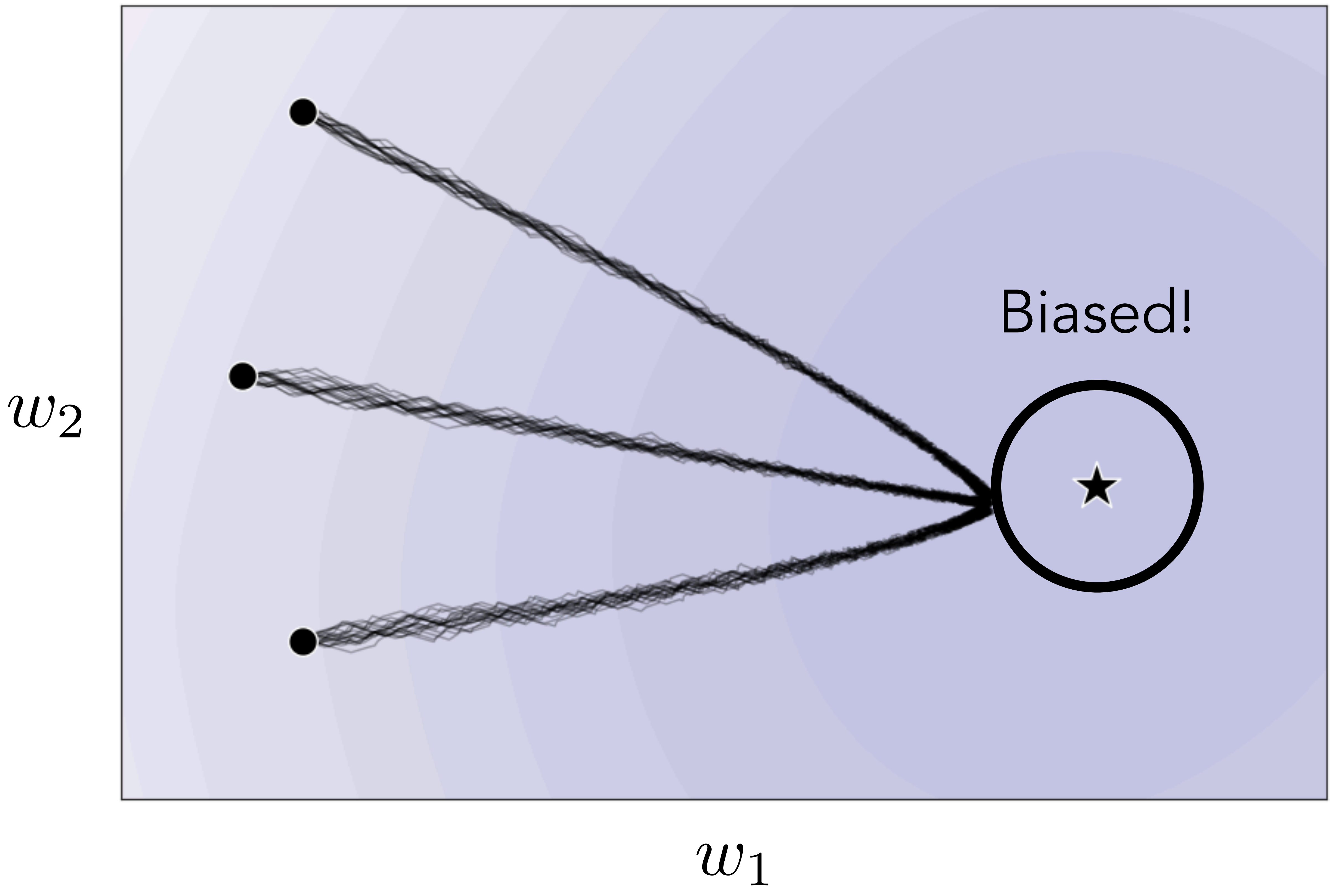
Problem Setting

Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$



In general, current approaches either:

1. have global complexity $O(n^2)$.
2. are biased and do not converge at all.
3. only converge under stringent conditions on the problem parameters.

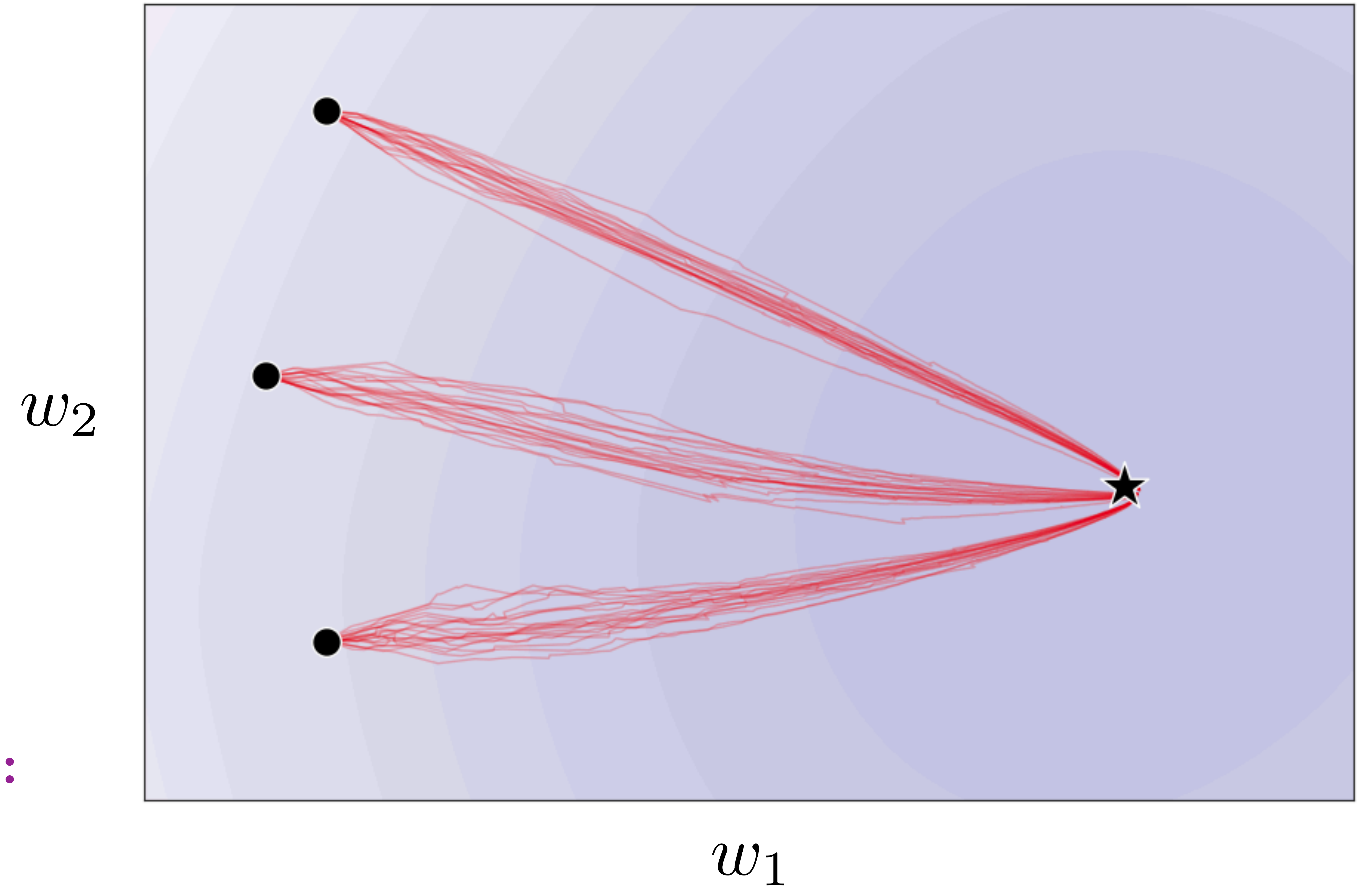
Our Approach: Drago

Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$



We propose Drago, a stochastic DRO algorithm using:

1. a delicate combination of randomized and cyclic coordinate-wise updates for variance reduction.
2. mini-batching to achieve an $O(n^{3/2})$ runtime.
3. a unified, transparent analysis for all parameter regimes.

Theoretical Analysis

Assumptions and Notation

$$|\ell_i(w) - \ell_i(w')| \leq G \|w - w'\|_2 \quad \text{(Lipschitz losses)}$$

$$\|\nabla \ell_i(w) - \nabla \ell_i(w')\|_2 \leq L \|w - w'\|_2 \quad \text{(smooth losses)}$$

$$\kappa_{\mathcal{Q}} = n \max \{q_i : q \in \mathcal{Q}, i \in [n]\} \quad \text{(uncertainty)}$$

Theorem. Drago with block size n/d reaches suboptimality ε with global complexity of the order

$$O\left(nd\left(\frac{\kappa_{\mathcal{Q}} L}{\mu} + \frac{\sqrt{n}G}{\sqrt{\mu\nu}}\right)\log\left(\frac{1}{\varepsilon}\right)\right)$$

Theoretical Analysis

Assumptions and Notation

$$|\ell_i(w) - \ell_i(w')| \leq G \|w - w'\|_2 \quad \text{(Lipschitz losses)}$$

$$\|\nabla \ell_i(w) - \nabla \ell_i(w')\|_2 \leq L \|w - w'\|_2 \quad \text{(smooth losses)}$$

$$\kappa_{\mathcal{Q}} = n \max \{q_i : q \in \mathcal{Q}, i \in [n]\} \quad \text{(uncertainty)}$$

Theorem. Drago with block size n/d reaches suboptimality ε with global complexity of the order

$$O\left(nd\left(\frac{\kappa_{\mathcal{Q}} L}{\mu} + \frac{\sqrt{n}G}{\sqrt{\mu\nu}}\right) \log\left(\frac{1}{\varepsilon}\right)\right)$$

Primal Condition Number   Mixed Condition Number

Theoretical Analysis

Assumptions and Notation

$$|\ell_i(w) - \ell_i(w')| \leq G \|w - w'\|_2 \quad \text{(Lipschitz losses)}$$

$$\|\nabla \ell_i(w) - \nabla \ell_i(w')\|_2 \leq L \|w - w'\|_2 \quad \text{(smooth losses)}$$

$$\kappa_{\mathcal{Q}} = n \max\{q_i : q \in \mathcal{Q}, i \in [n]\} \quad \text{(uncertainty)}$$

Theorem. Drago with block size n/d reaches suboptimality ε with global complexity of the order

$$O\left(nd\left(\frac{\kappa_{\mathcal{Q}} L}{\mu} + \frac{\sqrt{n}G}{\sqrt{\mu\nu}}\right)\log\left(\frac{1}{\varepsilon}\right)\right)$$

Dimension Dependence

Batch size of n/d trades off per-iteration complexity and number of iterations.

Empirical Analysis

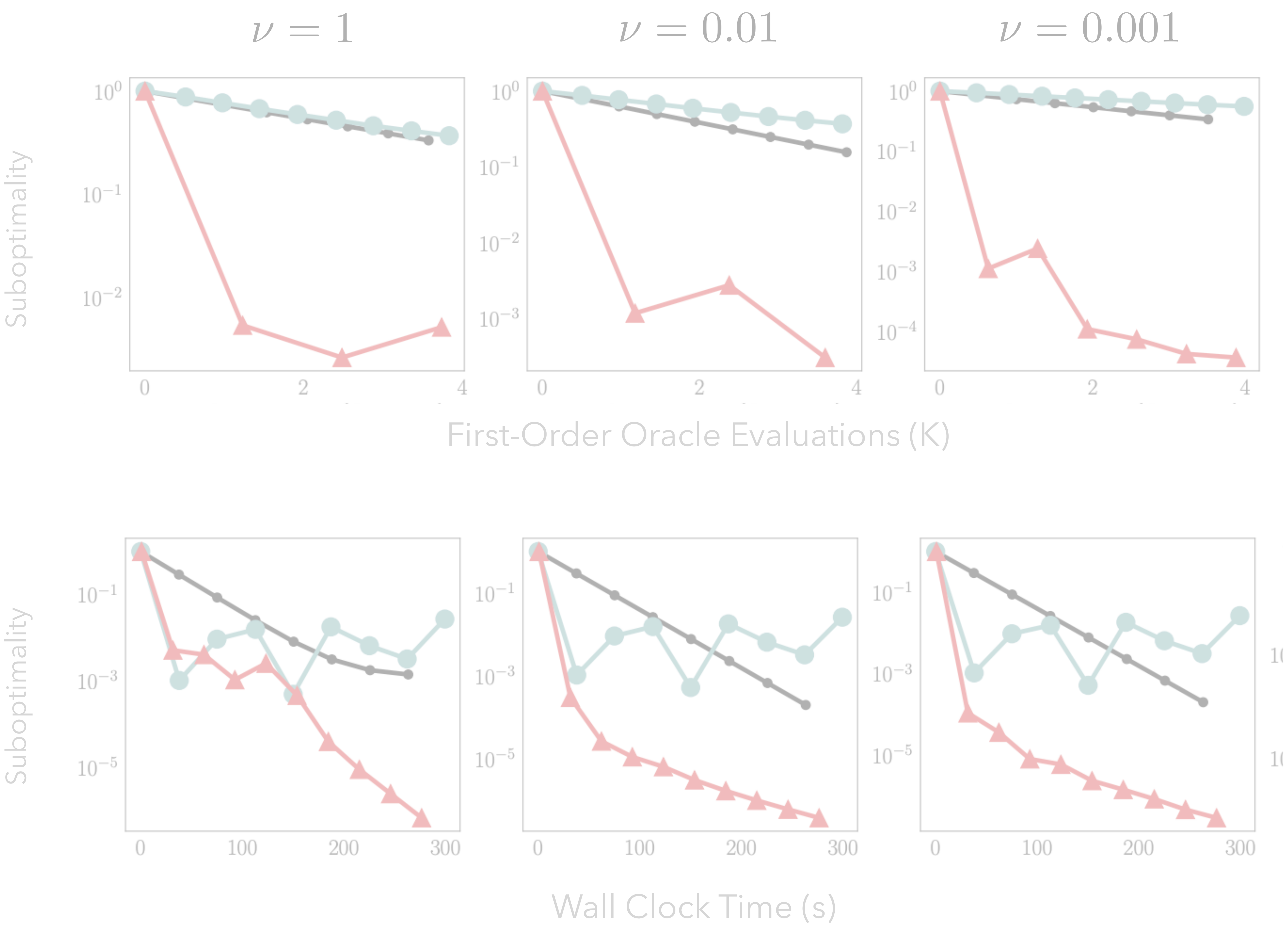
SGD

LSVRG

Drago

Task:
Classify the emotional content (angry, sad, etc.) of sentences based on a neural embedding ($d = 270$) of text passages.

Loss:
Multinomial Logistic Loss.



Empirical Analysis

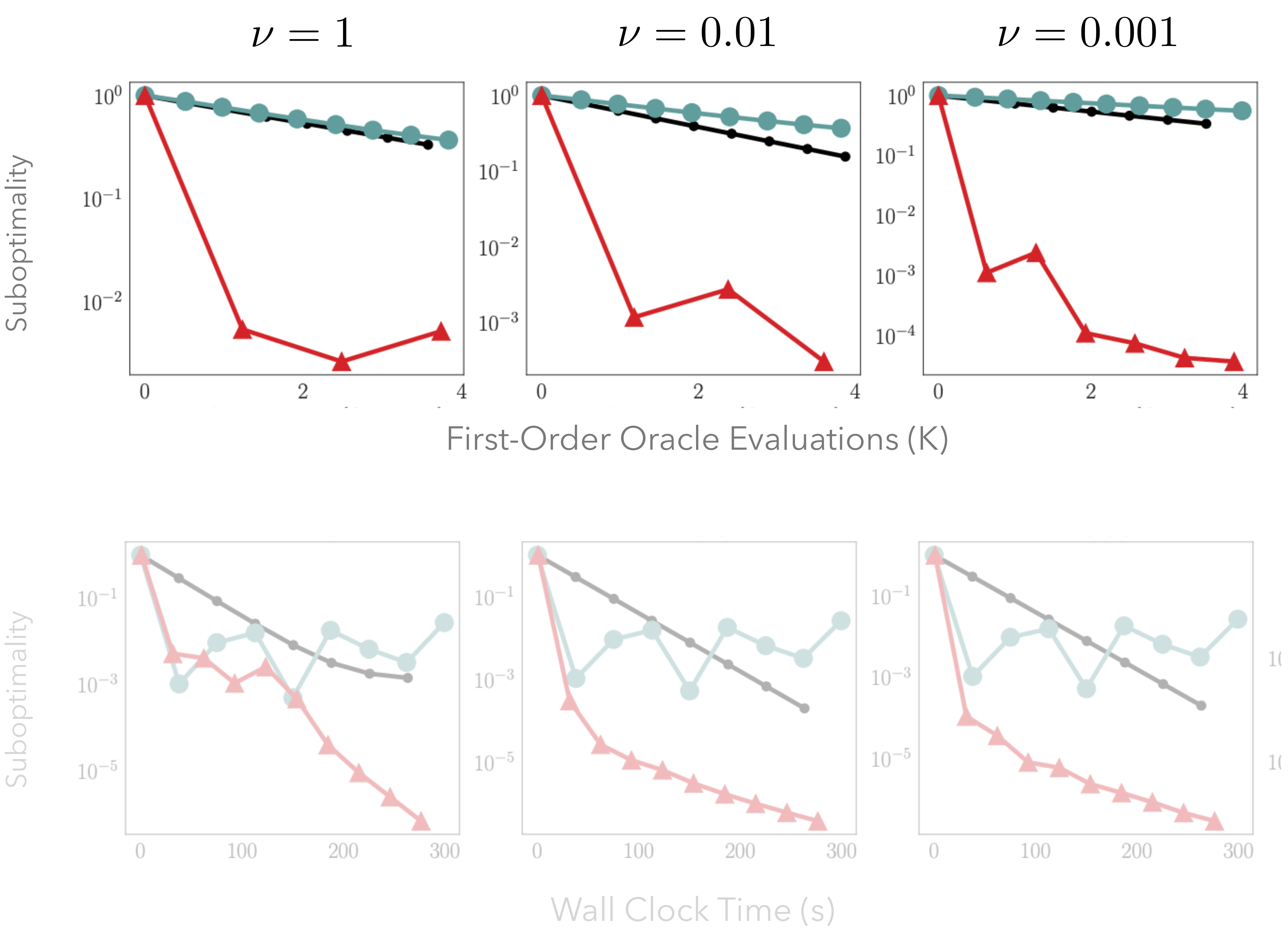
SGD

LSVRG

Drago

Task:
Classify the emotional content (angry, sad, etc.) of sentences based on a neural embedding ($d = 270$) of text passages.

Loss:
Multinomial Logistic Loss.

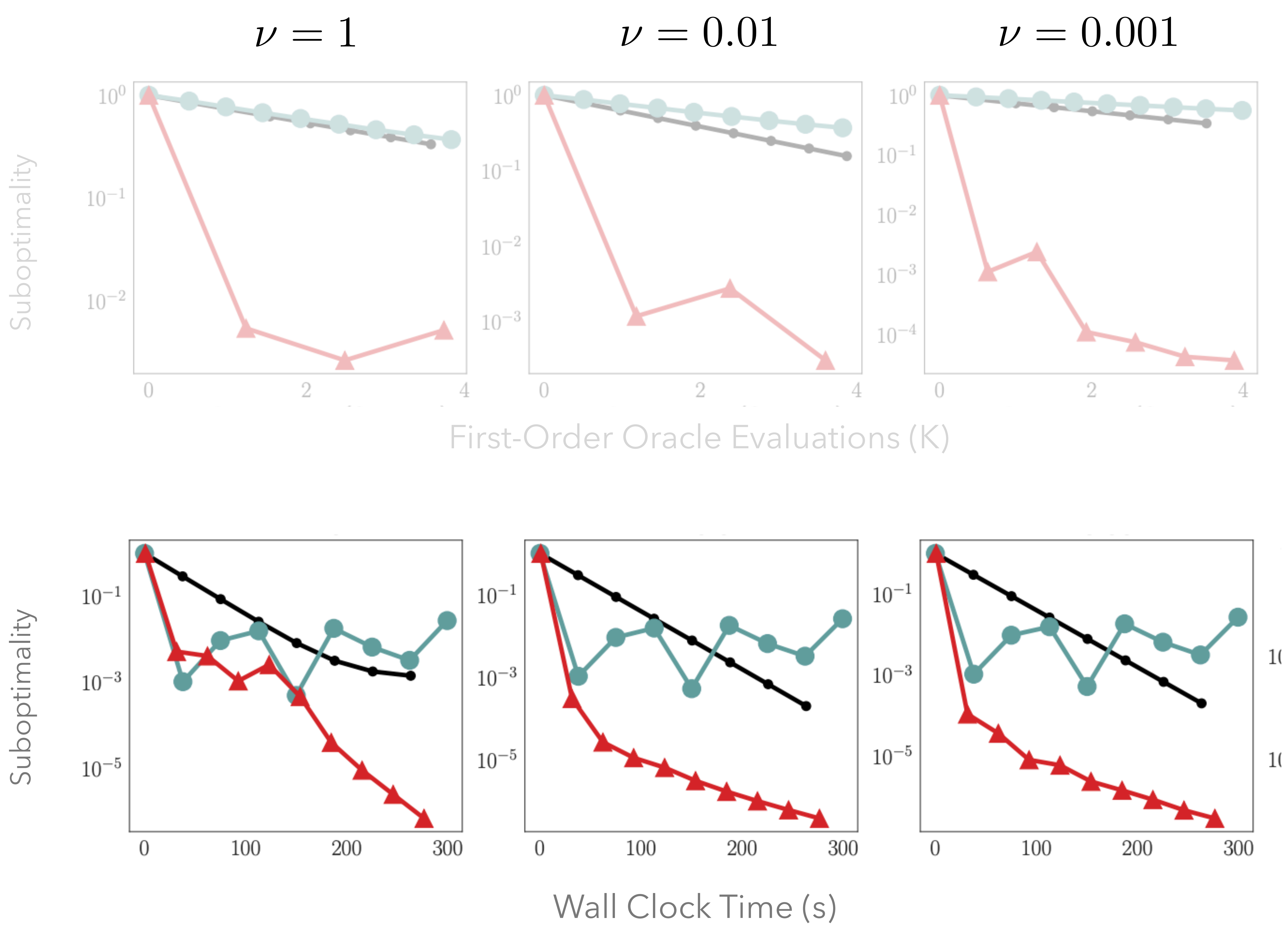


Empirical Analysis

SGD
LSVRG
Drago

Task:
Classify the emotional content (angry, sad, etc.) of sentences based on a neural embedding ($d = 270$) of text passages.

Loss:
Multinomial Logistic Loss.



Our Approach: Drago

Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$

We propose Drago, a stochastic DRO algorithm using:

1. a delicate combination of randomized and cyclic coordinate-wise updates for variance reduction.
2. mini-batching to achieve an $O(n^{3/2})$ runtime.
3. a unified, transparent analysis for all parameter regimes.

Software available in Python on GitHub!

Using Drago for Distributionally Robust Learning

In this Jupyter notebook, we show how to fit models using Drago (and baselines) to reproduce part of Figure 2 from the manuscript. Please see `README.md` for environment setup and other instructions.

In [1]:
```
from src.utils import get_objective, get_optimizer, get_min_loss
from src.data import load_dataset

import matplotlib.pyplot as plt
import numpy as np
import torch
from tqdm import tqdm
```

The two components required are an `Objective` (representing spectral risk measures or Chi-Squared divergence) and an `Optimizer`.

In [2]:
```
# Load a dataset, one of either: 'yacht', 'energy', 'concrete', 'kin8m', 'power', 'acsincome', or 'emotion'.
dataset = "yacht"

X_train, y_train, X_val, y_val = load_dataset(dataset)
```

Specify the primal regularization constant with `l2_reg` and dual regularization constant with `shift_cost`.

In [6]:
```
# Build objective.
model_cfg = {
    "objective": "cvar", # Options: 'cvar', 'chi2',
    "l2_reg": 1.0,
    "loss": "squared_error",  # Options: 'squared_error', 'binary_cross_entropy', 'multinomial_cross_entropy'.
    "n_class": None,
    "shift_cost": 1.0,
}

train_obj = get_objective(model_cfg, X_train, y_train)
val_obj   = get_objective(model_cfg, X_val, y_val)

minimum_loss = get_min_loss(model_cfg, X_train, y_train)
```

Generate the optimizer. The `drago_block` variant uses n/d as the batch/block size.

In [4]:
```
# Build optimizer.
seed = 1
optim_cfg = {
    "optimizer": "drago_block", # Options: 'sgd', 'lsvrg', 'drago', 'drago_auto', 'drago_block
    "lr": 0.0003,
    "epoch_len": 200, # Used as an update interval for LSVRG, and otherwise is simply a logging interval for othe
    "dual_reg": 1.0,
}
optimizer = get_optimizer(optim_cfg, train_obj, seed)
```

Our Approach: Drago

Empirical Risk Minimization

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(w) + \frac{\mu}{2} \|w\|_2^2$$

Distributionally Robust Optimization

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \ell_i(w) - \nu D(q \| \mathbf{1}/n) + \frac{\mu}{2} \|w\|_2^2$$

We propose Drago, a stochastic DRO algorithm using:

1. a delicate combination of randomized and cyclic coordinate-wise updates for variance reduction.
2. mini-batching to achieve an $O(n^{3/2})$ runtime.
3. a unified, transparent analysis for all parameter regimes.

Thank you!