Learn 2 Warm Start

INFORMS 2022

Rajiv Sambharya

PRINCETON UNIVERSITY

ORFE

Joint work with Brandon Amos, Georgina Hall, Bartolomeo Stellato

Real-time Convex Problem Applications

Robotics and Control

Energy grid

Finance

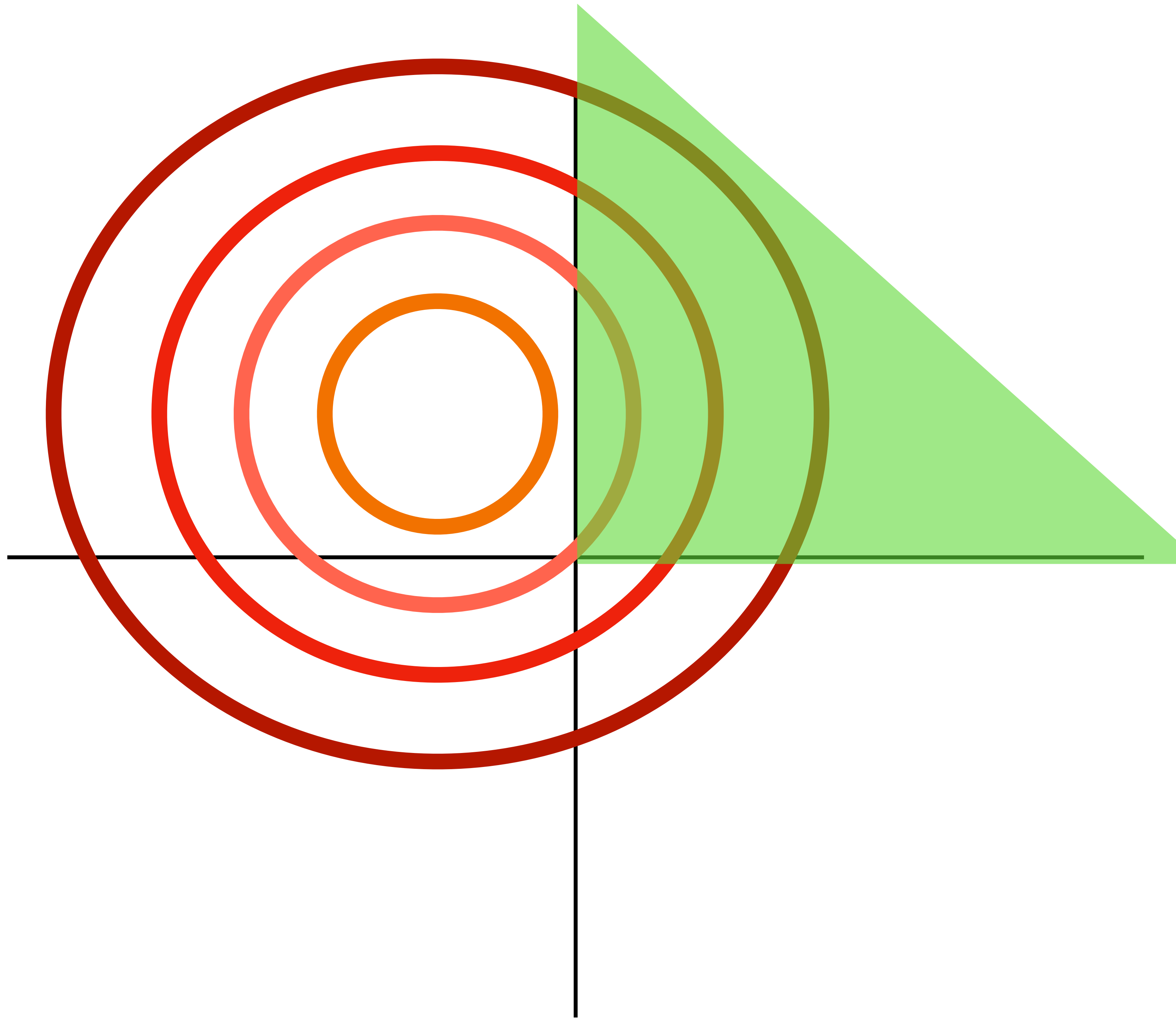
Convex Problems

$$\mathcal{X} = \{x \mid \mathbf{1}^T x \leq 1, x \geq 0\}$$

Example

minimize	$f(x)$	⟵ **Convex function**
subject to	$x \in \mathcal{X}$	⟵ **Convex set**

$$f(x) = \|x - \begin{bmatrix} -1 \\ 1 \end{bmatrix}\|_2^2$$



Primal

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}x^T P x + c^T x \\ \text{subject to} & Ax + s = b \\ & s \in \mathcal{K} \end{array}$$

Conic constraint

Example Cones

$$\{s \mid s \geq 0\}$$

$$\{(s, t) \mid \|s\|_2 \leq t\}$$

$$\{S \mid S \succeq 0\}$$

In many cases, solvers are not fast enough

Parametric Convex programs

Often, we solve **parametric** convex problems from the same family

Goal: Learning mapping efficiently

Parameter $\theta \longrightarrow$

$$\begin{array}{ll} \text{minimize} & f(x,\theta) \\ \text{subject to} & x \in \mathcal{X}(\theta) \end{array}$$

$\longrightarrow \hat{x}$ Optimal solution

$\theta \longrightarrow$ Only Optimization $\longrightarrow \hat{x}$

Accurate
Slow online

$\theta \longrightarrow$ Only Machine Learning $\longrightarrow \hat{x}$

Inaccurate
Fast online

$\theta \longrightarrow$ Optimization + Machine Learning $\longrightarrow \hat{x}$

Goal: accurate
Fast online

Running Example: Markowitz

Risk Returns

$\theta = r \longrightarrow$

$$
\begin{array}{ll}
\text{minimize} & \underbrace{\tfrac{1}{2}x^T\Sigma x}_{\text{Risk}} - \underbrace{\rho r^T x}_{\text{Returns}} \\
\text{subject to} & \mathbf{1}^T x = 1 \\
 & x \geq 0
\end{array}
$$

$\longrightarrow \hat{x}$

Σ: Covariance matrix
r: returns vector
ρ: weighting hyperparameter

Need to Decide

1. Solver choice

Rewrite KKT conditions as a linear complementarity problem

Algorithm: Douglas-Rachford Splitting

Details in next 2 slides

First order method

Fixed point iterations $z^{i+1} = T(z^i)$

2. Learning method with this solver

[SCS: O'Donoghue et. al 2021]

Fixed Point Iterates

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}x^T P x + c^T x \\ \text{subject to} & Ax + s = b \\ & s \in \mathcal{K} \end{array}$$

Fixed point iterate

$$z^i \longrightarrow \boxed{\begin{aligned} u^{i+1} &= (I + M)^{-1}(z^i - q) \\ v^{i+1} &= \Pi(2u^{i+1} - z^i) \\ z^{i+1} &= z^i + v^{i+1} - u^{i+1} \end{aligned}} \longrightarrow z^{i+1}$$

Linear system solve

$$M = \begin{bmatrix} P & A^T \\ -A & 0 \end{bmatrix} \qquad q = (c, b)$$

Projection onto $\mathcal{K}^\star$

Repeat until $\|z^{i+1} - z^i\|_2$ is small

Markowitz Example

$$
\begin{array}{ll}
\text{minimize} & \frac{1}{2}x^T \Sigma x - \rho r^T x \\
\text{subject to} & \mathbf{1}^T x = 1 \\
& x \geq 0
\end{array}
$$

Fixed point iterate

$$
\begin{aligned}
u^{i+1} &= (I + M)^{-1}(z^i - q) \\
v^{i+1} &= \Pi(2u^{i+1} - z^i) \\
z^{i+1} &= z^i + v^{i+1} - u^{i+1}
\end{aligned}
$$

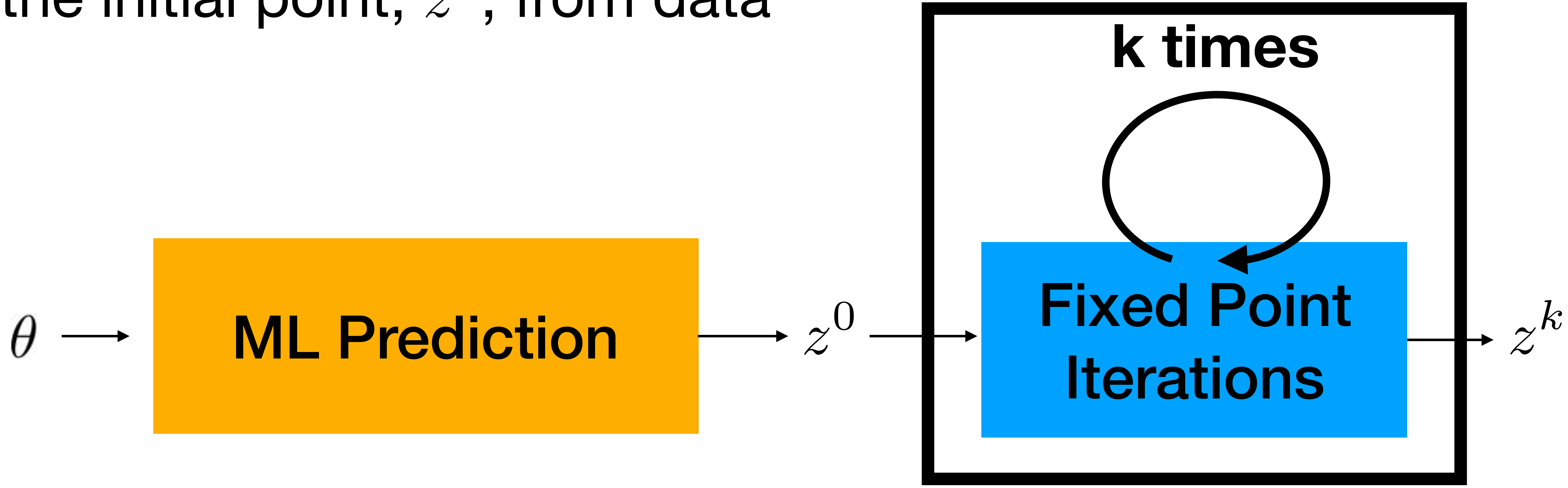
Linear system

$$
M = \begin{bmatrix} \Sigma & \mathbf{1} \\ -\mathbf{1}^T & 0 \end{bmatrix} \qquad q = (\rho r, 1)
$$

Projection: clip negative values

Our Neural Network Architecture

Main idea: Learn the initial point, z^0, from data



$$\text{Loss} = \sum_{i=1}^{N} \|z_i^{k+1} - z_i^k\|_2$$

Number of training problems

i-th problem

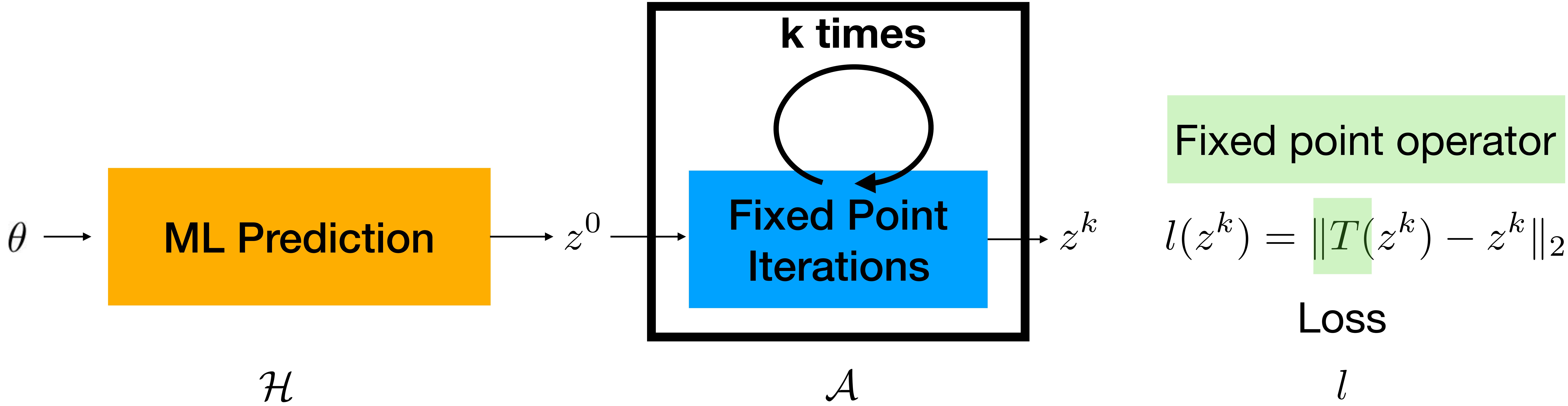
Fixed point residual

Minimize the loss w.r.t. the weights in the ML Prediction block

Apply a gradient-based method

Generalization Bounds

Generalization error can be bounded in terms of the Rademacher complexity



T κ-contractive $\longrightarrow$ $\hat{\mathcal{R}}_N(\mathcal{H} \circ \mathcal{A} \circ l) \leq 2\sqrt{2}\kappa^k \hat{\mathcal{R}}_N(\mathcal{H})$

$O(\frac{1}{\sqrt{k}})$ if T is averaged

Learn 2 warm start: Generalization bounds improve with both N and k

Markowitz Numerical Example

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}x^T\Sigma x - \rho r^T x \\ \text{subject to} & \mathbf{1}^T x = 1 \\ & x \geq 0 \end{array}$$

Sampled from noisy returns

r changes for each problem

10000 training problems

5000 testing problems

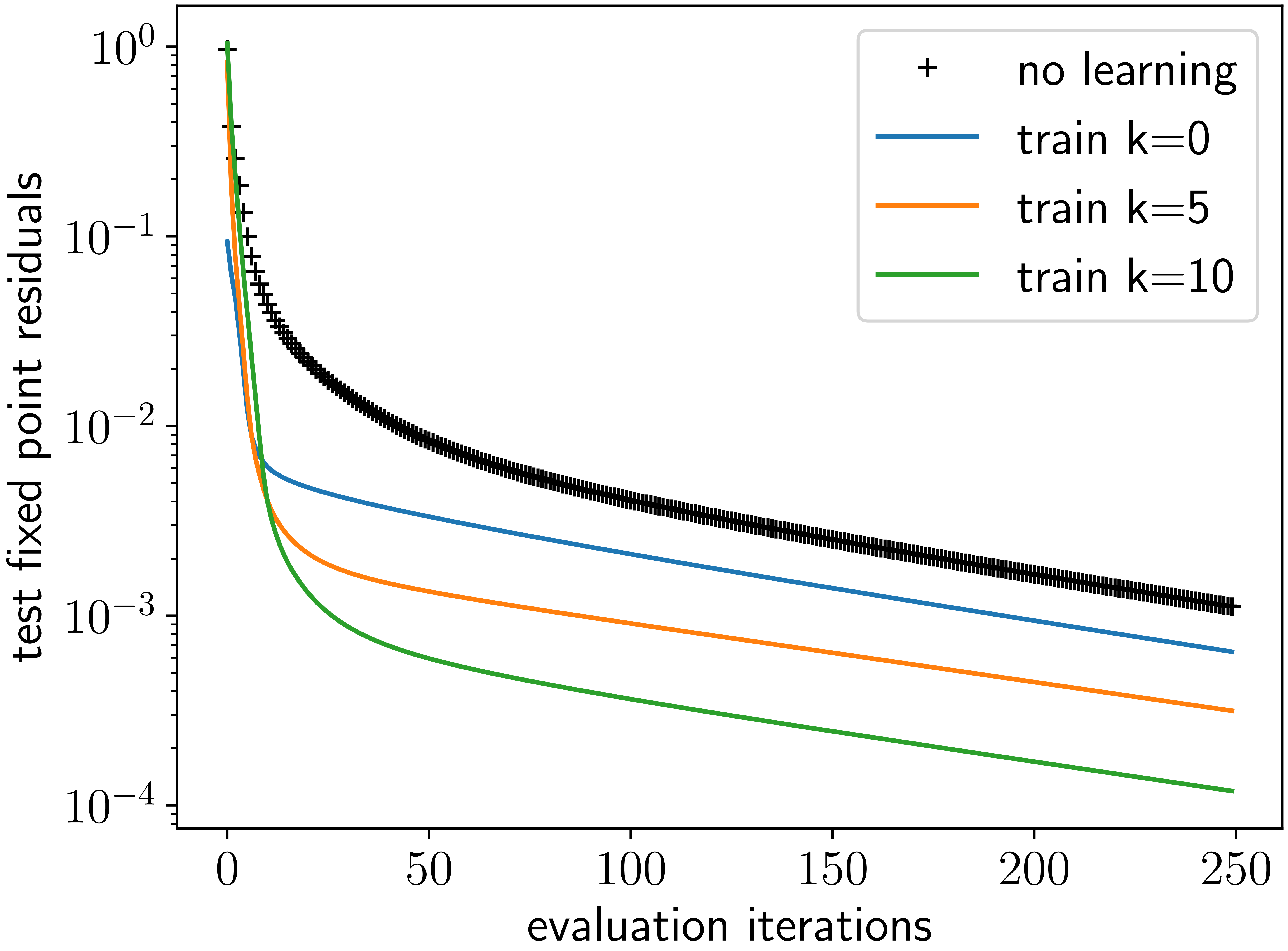
Used Russell index data
3000 assets

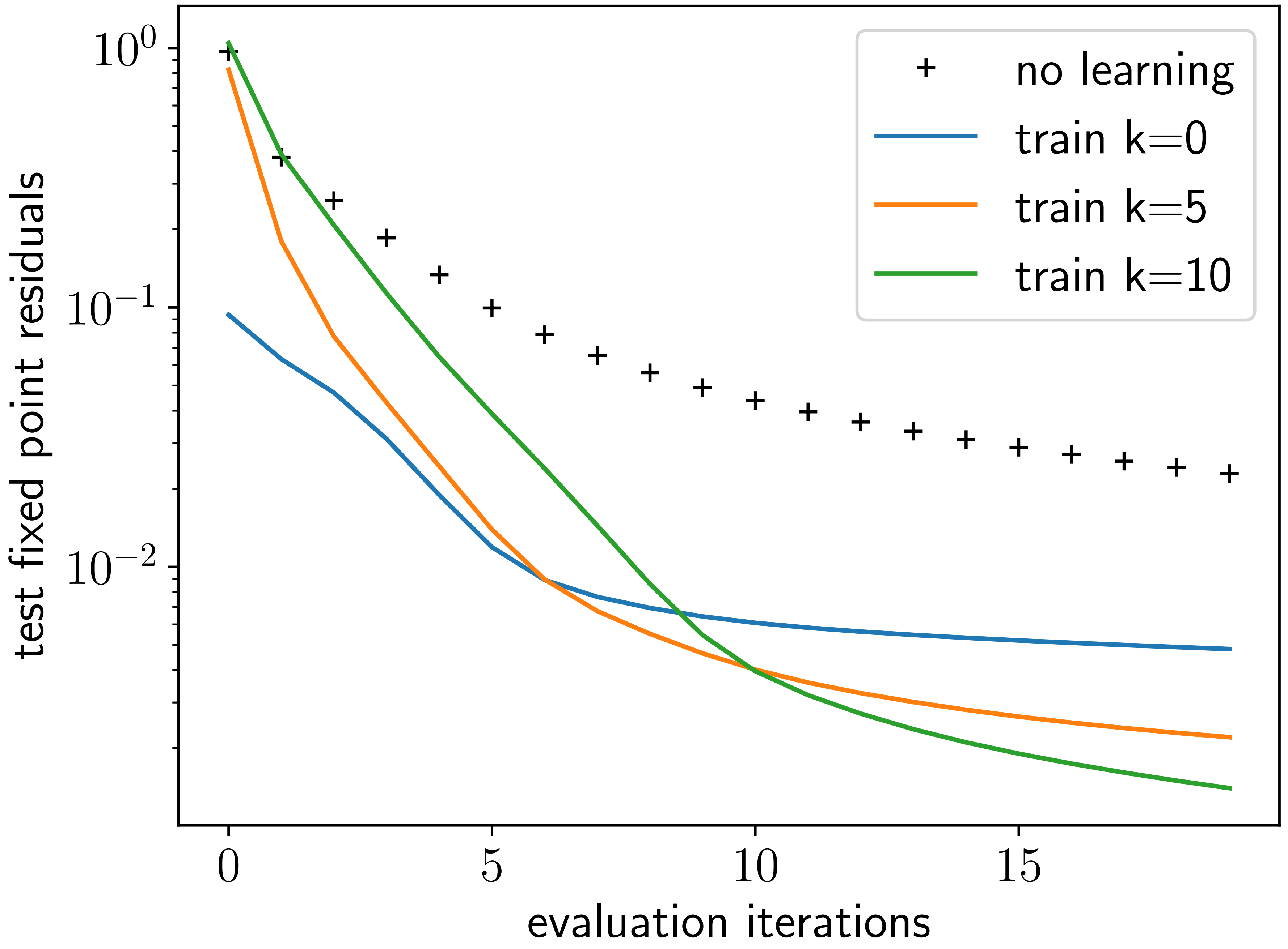
ML Prediction block

2 hidden layer neural network

ReLU activation

Markowitz Results





SCS implementation in C

Warm start	Time (sec)
Learn 2 warm start	**0.62**
None	3.20

Learning significantly decreases the number of iterations

Our contributions

Solve Convex problems in real time

End-to-end learning to accelerate fixed point algorithms

Generalization bounds in terms of N and k

rajivsambharya.github.io

rajivs@princeton.edu