# Analysis of Frank-Wolfe and Pairwise Frank-Wolfe Algorithms on the LASSO Problem

**Optimization for Data Science**

Roya Ghamari - 2071969

July 2024

# Introduction

**Objective:** Study the application of FW and PFW algorithms in minimizing the LASSO regression problem

**Algorithms:** Frank-Wolfe (FW), Pairwise Frank-Wolfe (PFW)

Focus on convergence behavior and optimization efficiency.

**Datasets:** Concrete Compressive Strength, Boston Housing

**LASSO:** Ideal for feature selection in high-dimensional data, explicitly recovering sparse solutions.

**Constraints and Rules:** $\ell_1$- Ball Constraint, Duality Gap, Armijo Rule

# LASSO Problem

1) $\min_{x \in R^n} f(x) := \|Ax - b\|^{\,2},\qquad s.t. \|x\|_1 \leq \tau$

2) $T = \{(r_i, b_i) \in \mathbb{R}^n \times \mathbb{R} : i \,\epsilon [1{:}\, m]\}$

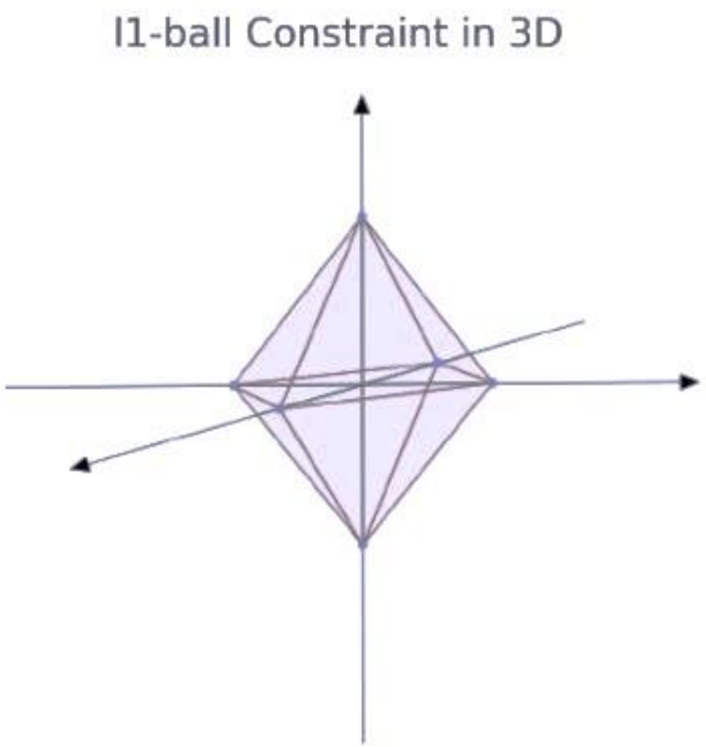
3) $C = \{x \in \mathbb{R}^n\!: \|x\|_1 \leq \tau\} = conv\{\mp \tau e_i : i \in [1{:}\, n]\}$

4) $LMO_C\big(\nabla f(x_k)\big) = sign\big(-\nabla_{i_k} f(x_k)\big) \cdot \tau e_{i_k},\qquad i_k \in \arg\max_i |\nabla_i f(x_k)|$

# $l_1$-ball constraint

1) $B_1(\tau) = \{x \in \mathbb{R}^n : \|x\|_1 \leq \tau\}$

2) $\|x\|_1 = \sum_{i=1}^{n} |x_i|$



# Frank Wolfe Algorithm

**Algorithm 1** Frank-Wolfe method

Choose a point $x_0 \in C$

For k = 0, …

  If $x_k$ satisfies some specific condition (duality gap), then STOP

  Compute $s_k \in LMO_C(\nabla f(x_k))$

  Set $d_k^{FW} = s_k - x_k$

  Set $x_{k+1} = x_k + \alpha_k d_k^{FW}$, with $\alpha_k \in (0,1]$ a suitably chosen stepsize

End for

$$f(x) := \|Ax - b\|^2$$

$$LMO_C(\nabla f(x_k)) = sign(-\nabla_{i_k} f(x_k)) \cdot \tau e_{i_k}$$

# Pairwise Frank Wolfe Algorithm

**Algorithm 2** Pairwise Frank-Wolfe method

Let $x^{(0)} \in \mathcal{A}$ and $\mathcal{S}^{(0)} := \{x^{(0)}\}$

For t = 0 to T do

  Let $s_t := LMO_{\mathcal{A}}(\nabla f(x^t))$ and $d_t^{FW} := s_t - x^{(t)}$

  Let $v_t \,\epsilon\, argmax\langle \nabla f(x^t), v\rangle, v \epsilon \mathcal{S}^{(t)}$ and $d_t^{A} = x^{(t)} - v_t$

  If $g_t^{FW} := \langle -\nabla f(x^t), d_t^{FW}\rangle \leq \epsilon$ then return $x_t$

  Else

    $d_t = d_t^{PFW} = d_t^{FW} + d_t^{A} = s_t - v_t$ , $\gamma_{max} := |\alpha_{v_t}|$

    Determine $\gamma_t \in [0, \gamma_{max}]$ a suitably chosen stepsize

    Update $x^{(t+1)} := x^t + \gamma_t d_t$

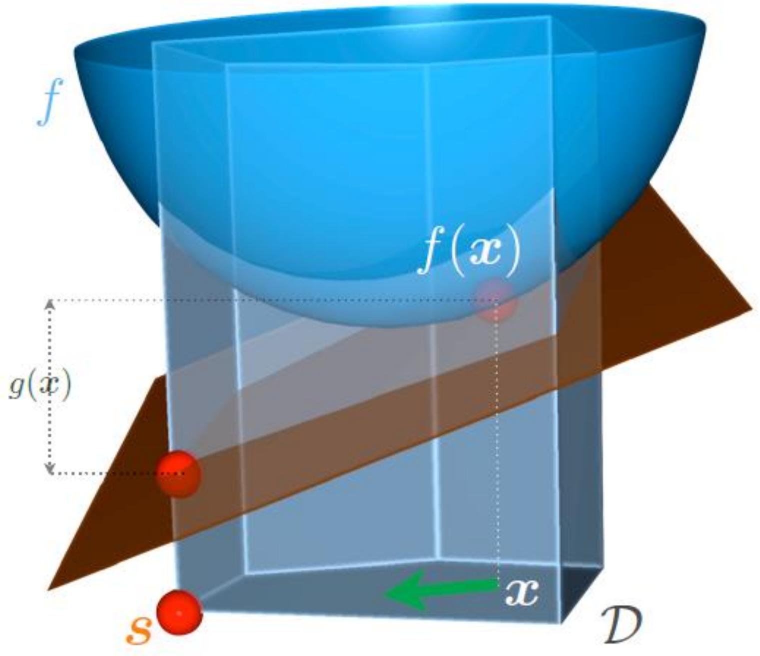
    Update $\mathcal{S}^{(t+1)} := \left\{ v \in \mathcal{A} \middle| |\alpha_v^{(t+1)}| > 0 \right\}$

  End

End

# The Duality Gap

- $g(x) = \max_{s \in \mathcal{D}} \nabla f(x)^T (x - s) = \max_{s \in \mathcal{D}} -\nabla f(x)^T (s - x)$
- $g(x) \geq f(x) - f(x^*)$




Duality gap shown in a step of frank-wolfe algorithm

# Armijo Line Search

- Fix the parameters $\delta \in (0,1)$ and $\gamma \in (0,1/2)$
- A starting step size $\triangle_k$
- The steps $\alpha = \delta^m \triangle_k$ with $m = \{0, 1, 2, \dots\}$ are tried until:

$$f(x_k + \alpha d_k) \leq f(x_k) + \gamma\alpha\nabla f(x_k)^T d_k$$

- Set $\alpha_k = \alpha$

# Datasets

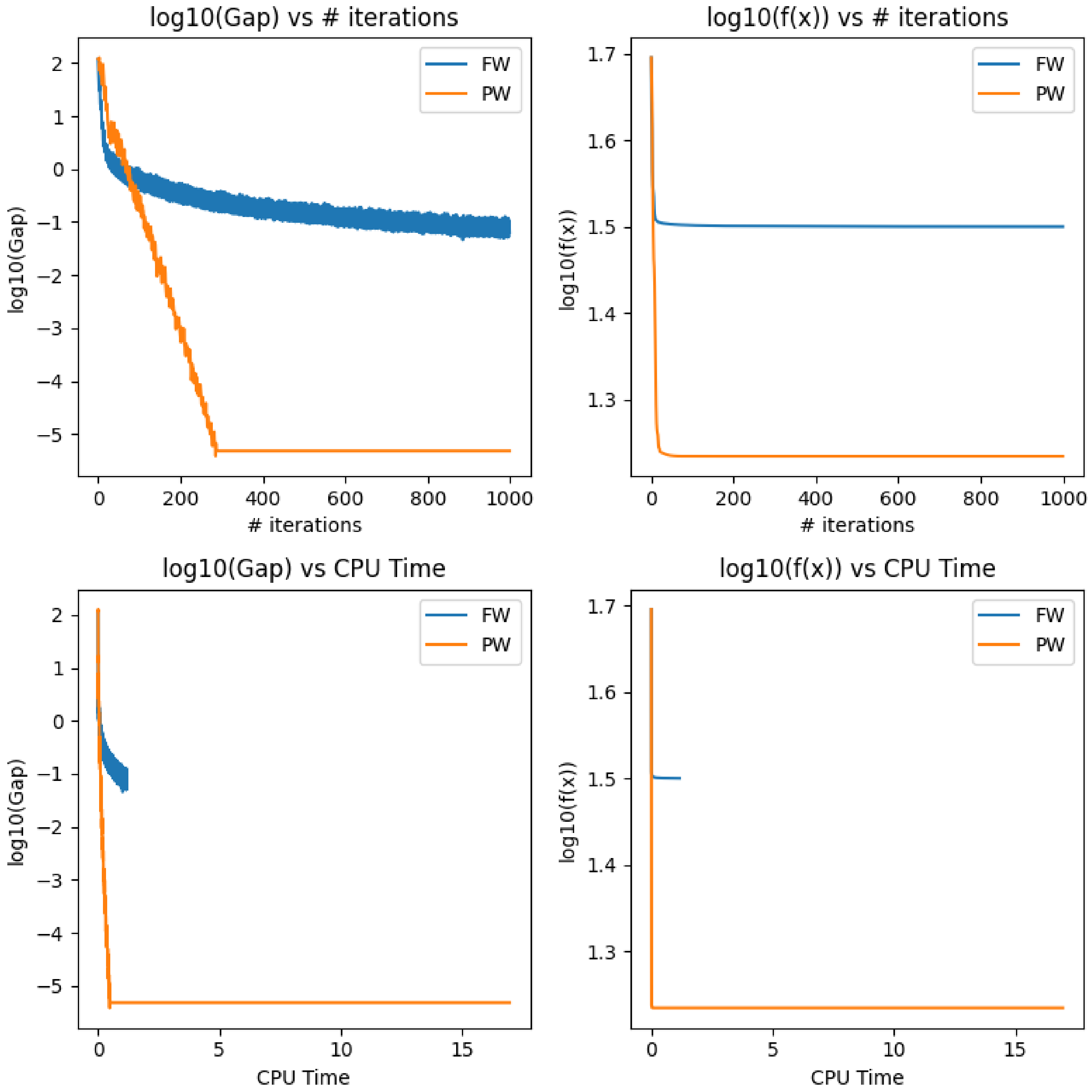
## Concrete Strength Dataset (1030 instances)

- **Cement:** Amount of cement in the mixture
- **Blast Furnace Slag**: Amount of blast furnace slag in the mixture
- **Fly Ash:** Amount of fly ash in the mixture
- **Water:** Amount of water in the mixture
- **Superplasticizer:** Amount of superplasticizer in the mixture
- **Coarse Aggregate:** Amount of coarse aggregate in the mixture
- **Fine Aggregate:** Amount of fine aggregate in the mixture
- **Age:** Age of the concrete
- **Compressive Strength:** Compressive strength of the concrete

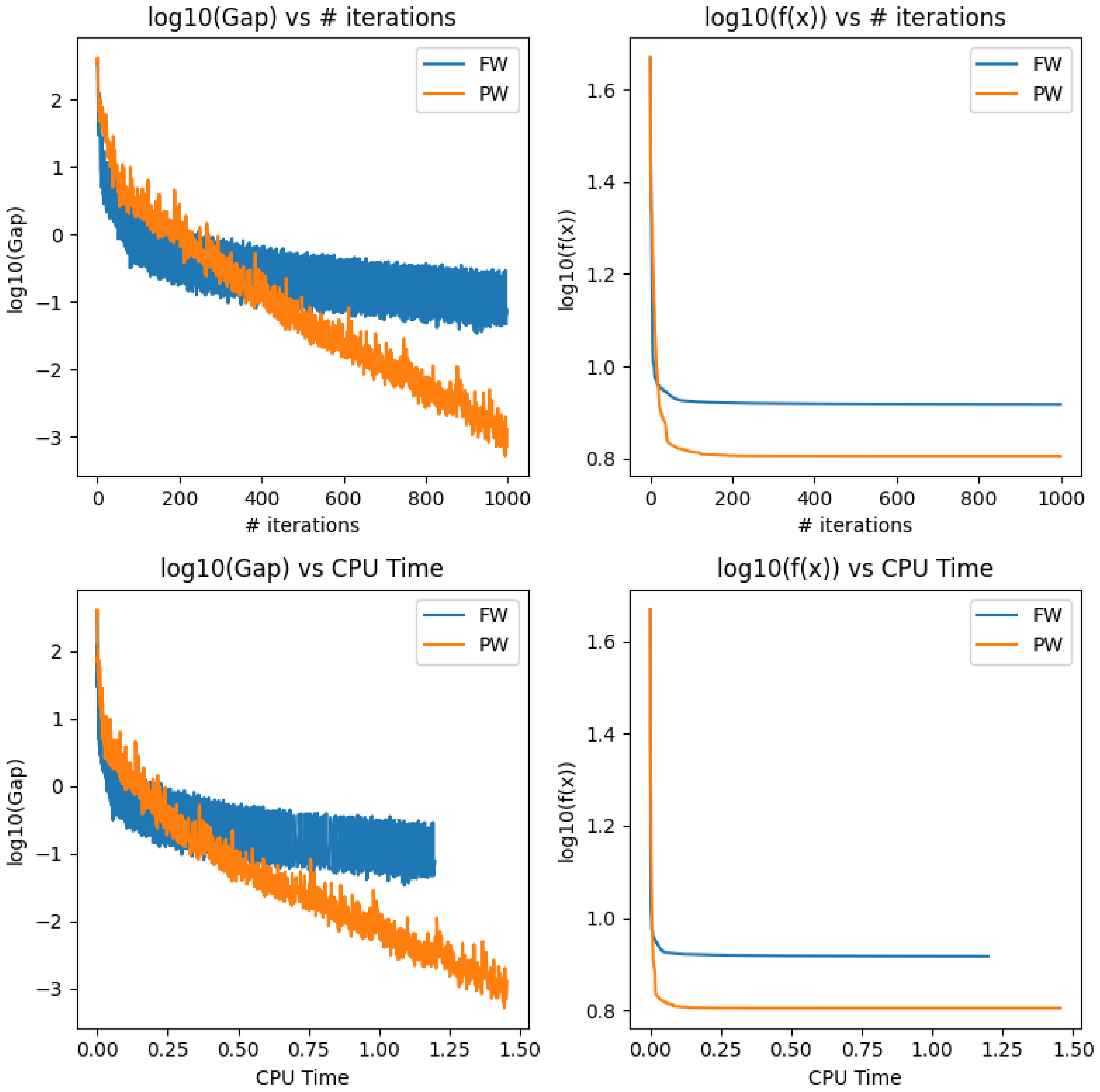
## Boston Housing Dataset (506 instances)

- **CRIM:** Per capita crime rate by town
- **ZN**: Proportion of residential land zoned for lots over 25,000 sq.ft.
- **INDUS:** Proportion of non-retail business acres per town
- **CHAS:** Charles River dummy variable (1 if tract bounds river; 0 otherwise)
- **NOX:** Nitric oxides concentration (parts per 10 million)
- **RM**: Average number of rooms per dwelling
- **AGE:** Proportion of owner-occupied units built prior to 1940
- **DIS:** Weighted distances to five Boston employment centers
- **RAD:** Index of accessibility to radial highways
- **TAX:** Full-value property-tax rate per $10,000
- **PTRATIO:** Pupil-teacher ratio by town
- **B**: 1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town
- **LSTAT:** Percentage of lower status of the population
- **MEDV:** Median value of owner-occupied homes in $1000's

# Results For Concrete Strength Dataset



# Results For Boston Housing Dataset



# Conclusion

### FW Algorithm:

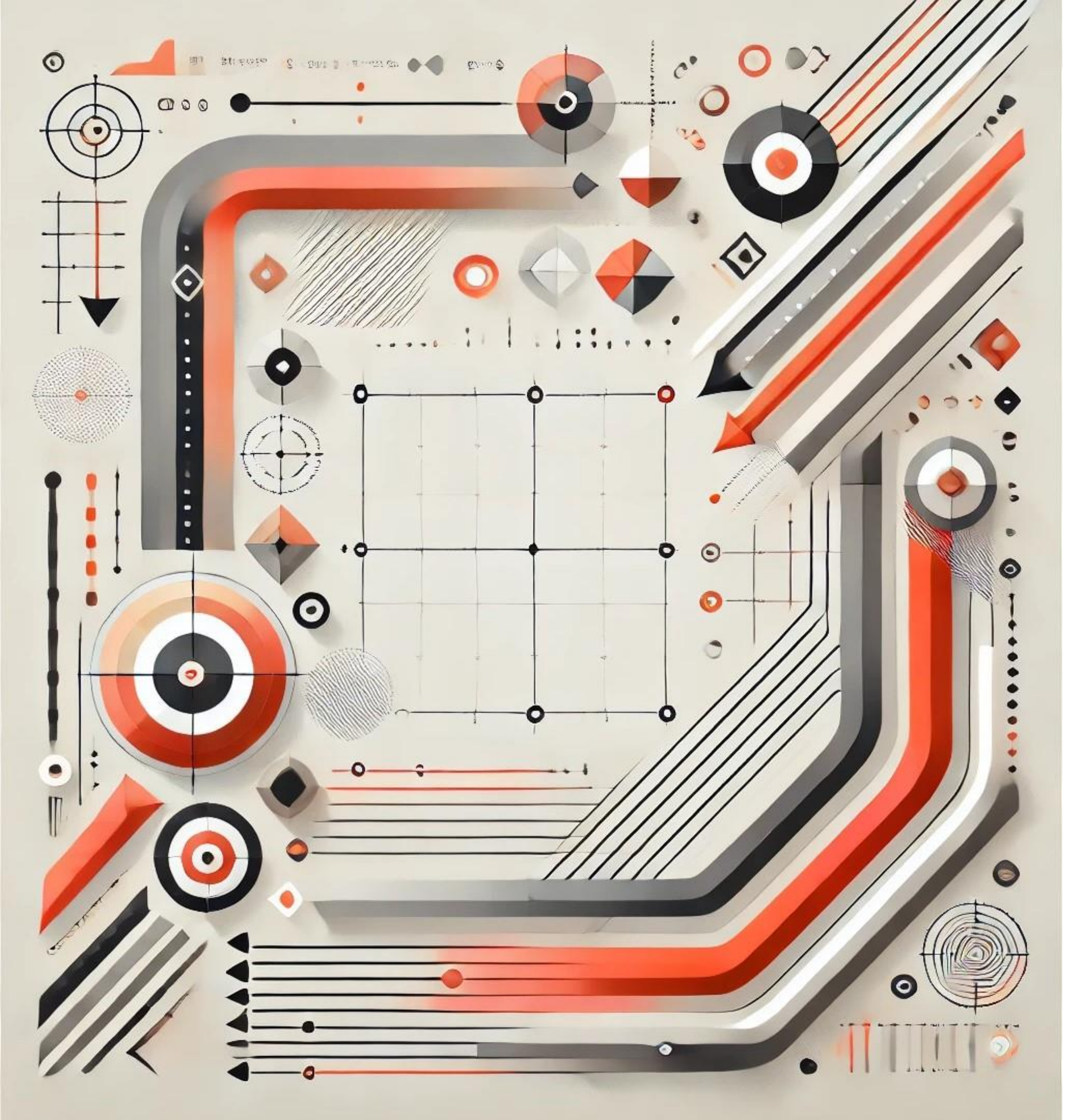
- Efficient in terms of CPU time
- Suitable for quick, significant improvements
- Tends to plateau early, resulting in suboptimal solutions in complex datasets

### PFW Algorithm:

- Better at finding global optimal
- Higher computational cost
- Preferable for applications prioritizing the quality of the final solution

### Datasets:

- Similar trends but less pronounced differences for Boston Housing
- Higher dimensionality and fewer instances made the optimization landscape more complex



# Thank you