On Learning Mixture of Linear Regressions in a Non-Realizable Setting

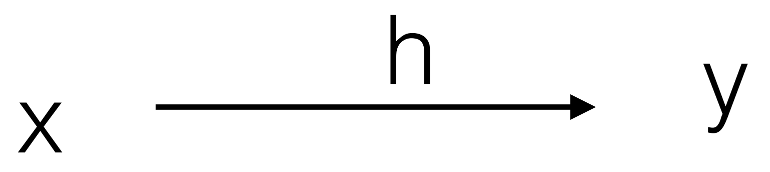
Avishek Ghosh Arya Mazumdar Soumyabrata Pal Rajat Sen

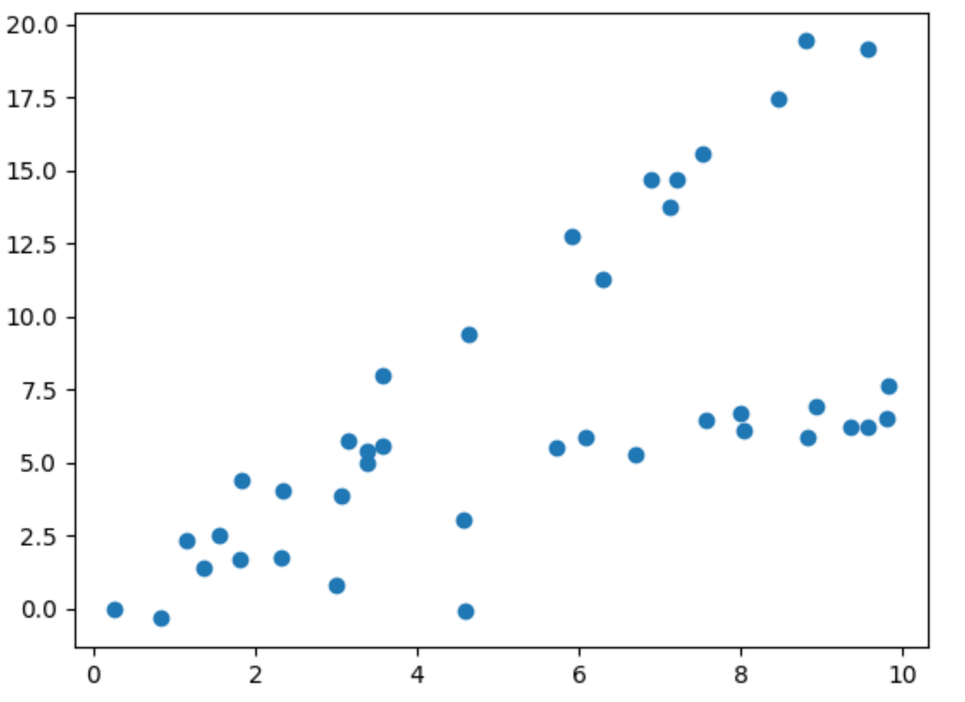
UC San Diego Google Research

ICML 2022

Mixture of Functional Relationships

x: Covariates or Features
y: Label



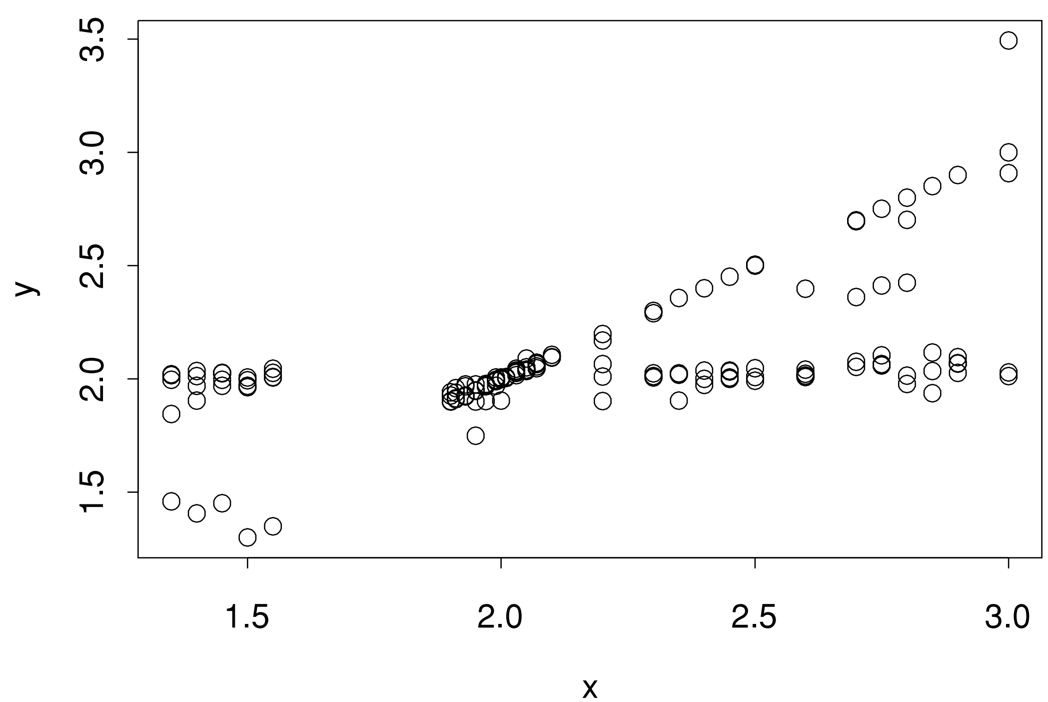


Find a mapping

Parameterize h $y = h_w(x)$

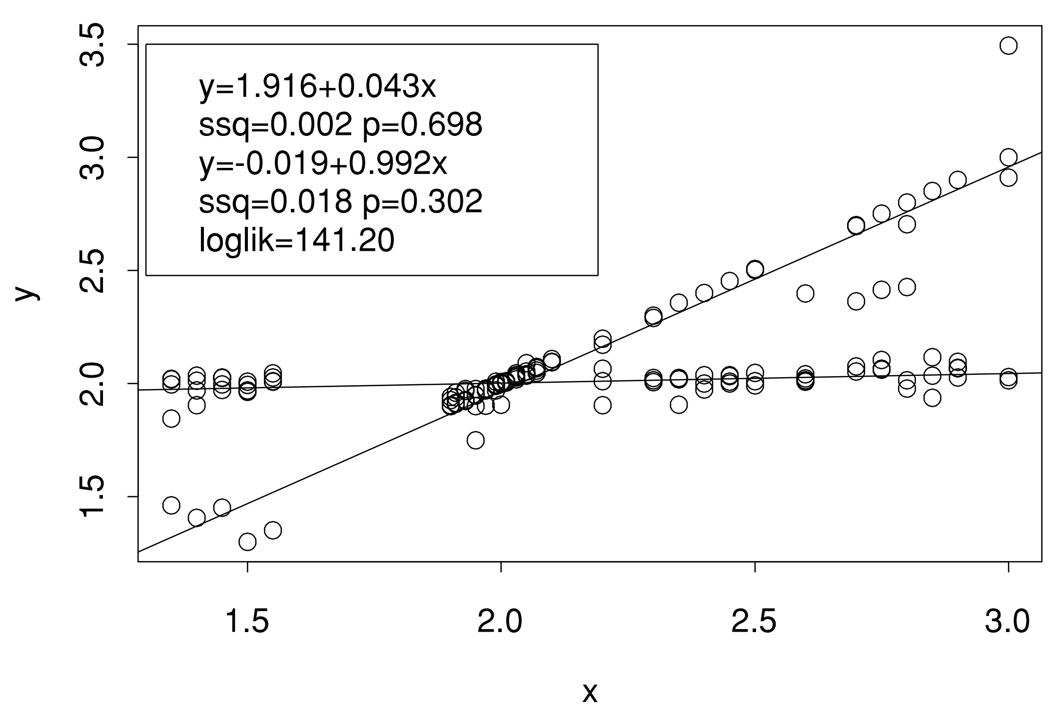
- Linear regression/classification
- Neural Networks

Mixture of Linear Regressions (MLR)



Music Perception

- Cohen 1980
- De Veaux, 1989;
- Viele and Tong, 2002



Biology

- Yi et. al, '2014
- Yin et. al, '2017

Mixture of Linear Regressions

Economics

- Predicting demands

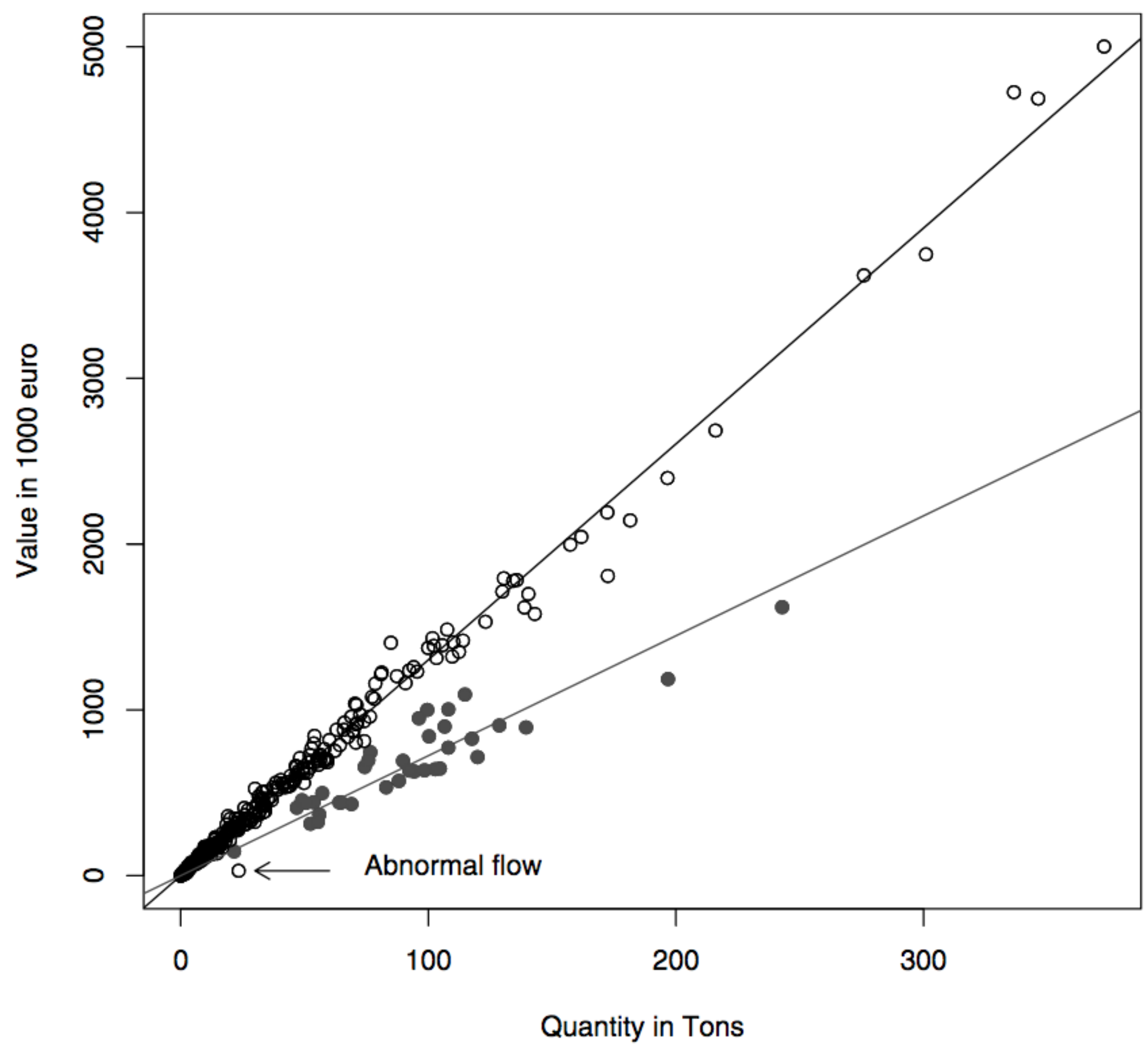


Figure 3. Quantities (in tons) and values (in thousands of euros) of 677 monthly imports of a fishery product from a third country into the EU, over a period of three years. Flows to MS7 (solid dots) and flows to the other Member States (open circles) form distinct groups following different regression lines. On the bottom-left an abnormal single flow to MS11.

The Realizable Model for MLR

Mixture of k Linear Regressions: (x, y)

$$x \sim \mathcal{P}, x \in \mathbb{R}^d$$

Latent variable

$$t \sim_U [k]$$

$$y \mid x, t \sim \mathcal{N}(\langle x, \theta^{(t)} \rangle, \sigma^2)$$

Unknown parameters:

$$\theta^{(1)}, \ldots, \theta^{(k)} \in \mathbb{R}^d$$

Realizable Setting

- Balakrishnan et al., 2017, Klusowski et al., 2019:: EM starting from close enough points; Finite sample
- Yi et al., 2014: Initialization via spectral method; Yi et al., 2016: Extension to k lines
- Kwon, Caramanis, 2018: Random initialization suffice for two lines
- Li, Liang, 2018: Non-Gaussian covariates: Nearly optimal sample and computational complexities
- There are other algorithmic works (Chen et al., Diakonikolas and Kane, 2020)

Non-Realizability: Learning Theory for MLR

Do not assume a generative model on y

Given data points $(x, y) \sim \mathcal{D}$, Let's fit k lines

Non-Realizability: Learning Theory for MLR

Do not assume a generative model on y

Given data points $(x, y) \sim \mathcal{D}$, Let's fit k lines

Now, this is a supervised learning problem

Question? Can you do prediction with mixtures?

Can we use those lines to predict the future labels?

Non-Realizability: Learning Theory for MLR

Do not assume a generative model on y

Given data points $(x, y) \sim \mathcal{D}$, Let's fit k lines

Now, this is a supervised learning problem

Question? Can you do prediction with mixtures?

Can we use those lines to predict the future labels?

Possible!! If we are allowed to predict a list of k labels.

Predicting a list

- As long as the correct label is (or close to) one of the labels in the list it is a success
- In many applications (such as recommendation systems) we already suggest a list
- Even in plain linear regression, list prediction was suggested (Kothari et al., 2018)

Supervised Learning with MLR: What's the Loss?

A vector valued hypothesis class: $\bar{h} = (h_1(\cdot), \cdots, h_k(\cdot))$

$h_1(\,.\,), \ldots, h_k(\,.\,) \in \mathscr{H}$ (base hypothesis class)

Min-loss:

$$\mathcal{L}(y, \bar{h}(x)) := \min_{j \in [k]} \ell(y, \bar{h}(x)_j) = \min_{j \in [k]} \ell(y, h_j(x))$$

$$L(\bar{h}) := \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(y_i, \bar{h}(x_i)).$$

ERM with the Min-Loss

In this paper: Ridge regression— Base class:

$$\mathcal{H} = \{\langle \theta, \cdot \rangle : \forall \theta \in \mathbb{R}^d \text{ s.t } \|\theta\|_2 \leq w\}$$

Empirical Loss:

$$L(\theta_1, \ldots, \theta_k) = \frac{1}{n} \sum_{i=1}^{n} \min_{j \in [k]} \{(y_i - \langle x_i, \theta_j \rangle$$

$$\text{with} (\theta_1^*, \ldots, \theta_k^*) = \underset{\{\theta_j\}_{j=1}^k}{\operatorname{argmin}} L(\theta_1, \ldots, \theta_k$$

The Max. Likelihood loss is close but not exactly
The "min" is replaced by "soft-min"

Generalization Guarantees with MLR

Supervised setup: what can we say about generalization

Recall $\mathrm{Gen} = \sup_{\bar{h}\in\mathcal{H}_k} \mathbb{E}\mathcal{L} - L$

where $\mathscr{H}_k$: vector hypothesis class

We show that the (empirical) Rademacher Complexity of $\mathscr{H}_k$:

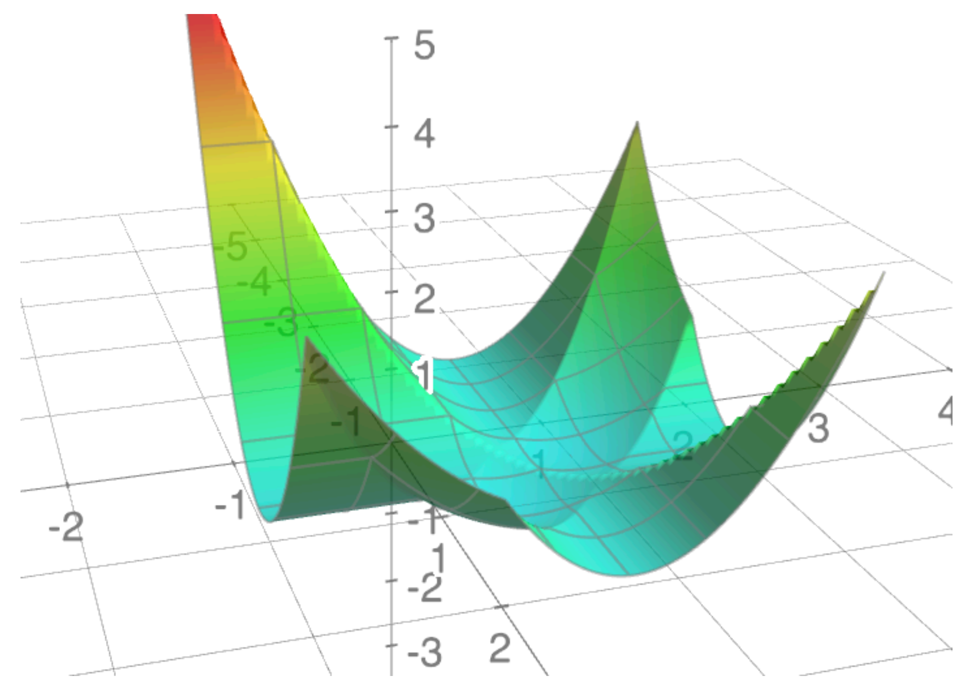
$$\hat{\mathfrak{R}}_S(\bar{\mathcal{H}}_k) \leq k\mu\hat{\mathfrak{R}}_{S_x}(\mathcal{H})$$

Solving the ERM

$$L(\theta_1,\ldots,\theta_k) = \frac{1}{n}\sum_{i=1}^{n} \min_{j\in[k]} \left\{ (y_i - \langle x_i, \theta_j \rangle)^2 \right\}.$$

$$\text{with}(\theta_1^*,\ldots,\theta_k^*) = \underset{\{\theta_j\}_{j=1}^{k}}{\operatorname{argmin}} L(\theta_1,\ldots,\theta_k).$$

1. Non-Convex loss
2. Yi et al.: Intractable (by reduction from subset-sum)



What if we still use EM

$$(x_i, y_i)_{i=1}^n; x_i \in \mathbb{R}^d, y_i \in \mathbb{R}$$

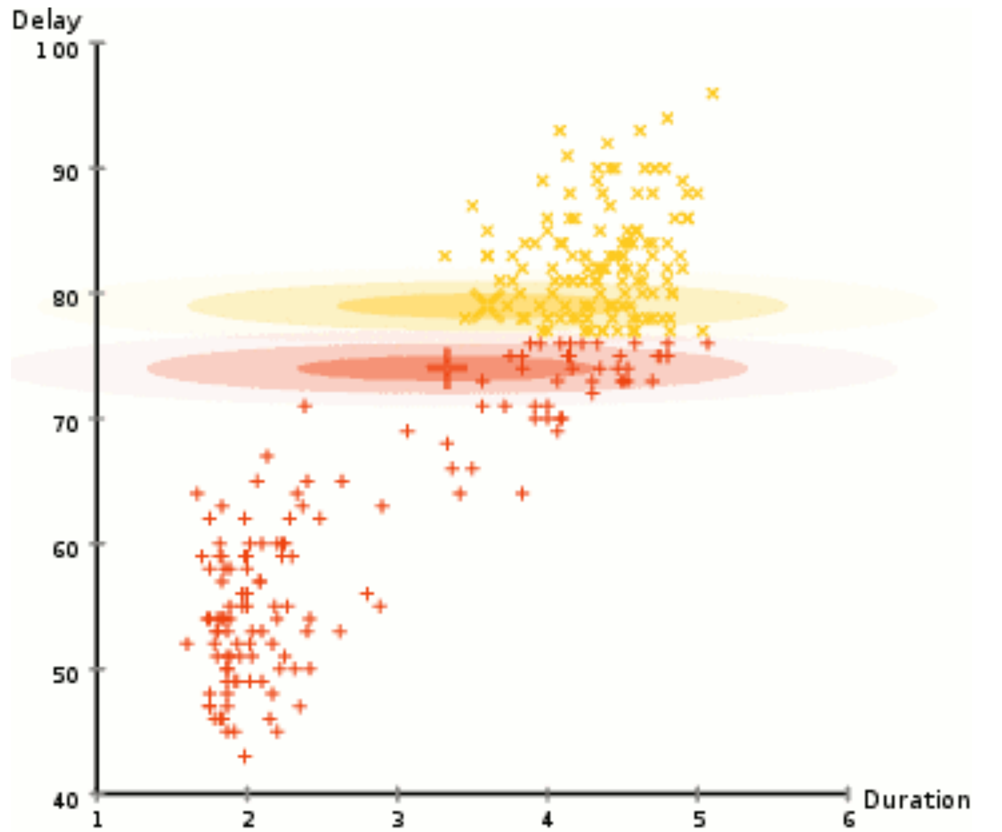
There isn't a probabilistic model anymore
So what's EM?

Let's do AM (Alternating Minimization)

Alternating Minimization—a classical solution

Initialize with k lines.
Repeat:

1. For a fixed set of lines, find the partition
2. For each partition, learn the optimal lines



Gradient AM:
Instead of the optimization in the second step, take a gradient step

Disclaimer: Gradient EM were already used in Balakrishanan, Wainwright, Yu, '17 with the Probabilistic model

Gradient AM

Algorithm 1 Gradient AM for Mixture of Regressions

1: **Input:** $\{x_i, y_i\}_{i=1}^n$, Step size γ
2: **Initialization:** Initial iterate $\{\theta_j^{(0)}\}_{j=1}^k$
3: **for** $t = 0, 1, \ldots, T-1$ **do**
4: Partition:
5: Construct $\{S_j^{(t)}\}_{j=1}^k$ such that

$$S_j^{(t)} = \{i \in [n] : (y_i - \langle x_i, \theta_j^{(t)} \rangle)^2 = \min_{j' \in [k]} (y_i - \langle x_i, \theta_{j'}^{(t)} \rangle)^2\} \forall j \in [k]$$

6: Gradient Step:

$$\theta_j^{(t+1)} = \theta_j^{(t)} - \frac{\gamma}{n} \sum_{i \in S_j^{(t)}} \nabla F_i(\theta_j^{(t)}), \forall j \in [k]$$

7: where $F_i(\theta_j^{(t)}) = (y_i - \langle x_i, \theta_j^{(t)} \rangle)^2$
8: **end for**
9: **Output:** $\{\theta_j^{(T)}\}_{j=1}^k$

Gradient AM convergence

- Under some regularity assumption on data

- And if initial lines are close enough (within 1/sqrt{d})

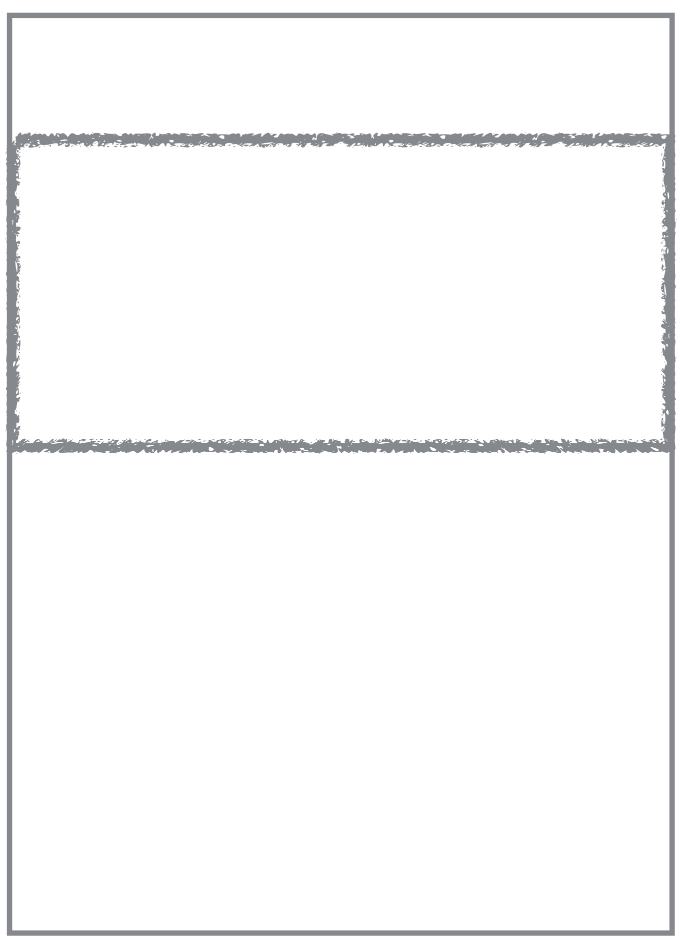
$$\|\theta_{ini,i} - \theta_i^*\| \lesssim \frac{1}{\sqrt{d}} \|\theta_i^*\|$$

Gradient AM converges to the global optimum of the Min-Loss: $\|\theta_{t+1,i} - \theta_i^*\| \leq \frac{1}{2}\|\theta_{t,i} - \theta_i^*\| +$ bias, for all $i \in [k]$ with high probability

- In practice works well with multiple-restarts

Other Algorithms

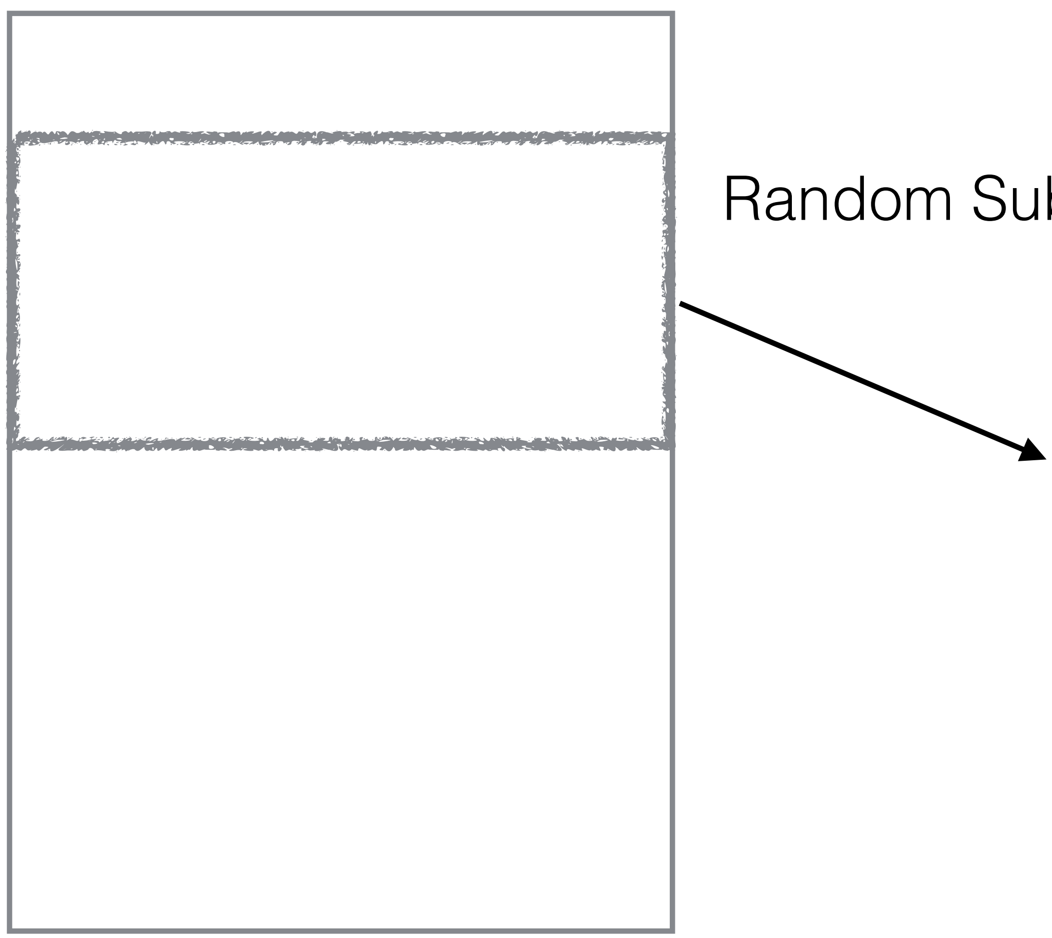
We have another poly-time algorithms with good approximation guarantees



Random Subset $O(k^2 d \log n)$

Other Algorithms

We have another poly-time algorithm with good approximation guarantees

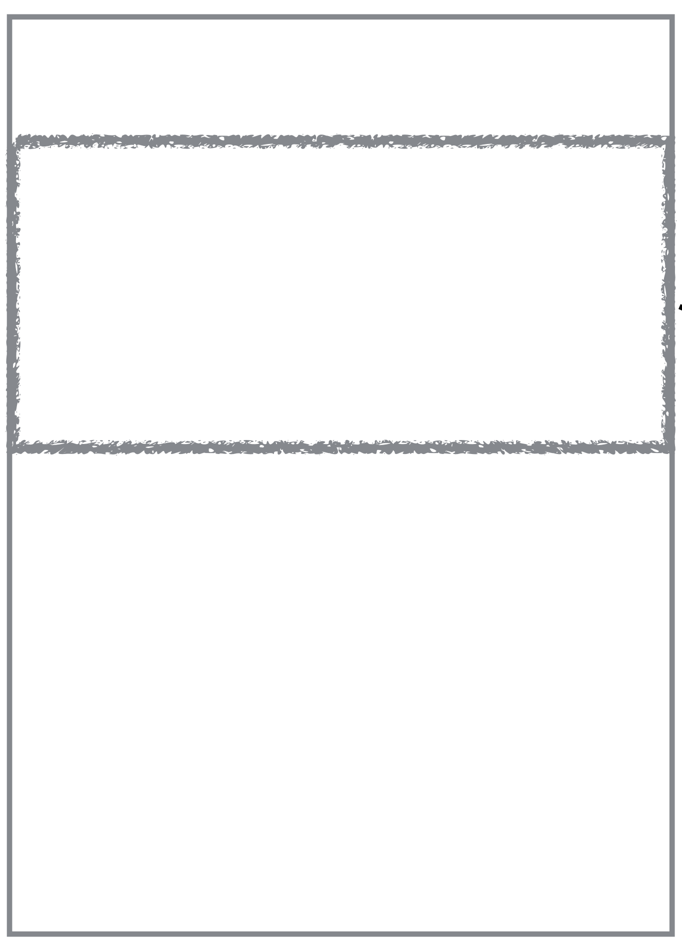


Random Subset $O(k^2 d \log n)$

Go through all k-partitions and calculate least-squares solution in each part and choose the solution with the best loss on the **whole** dataset

Other Algorithms

We have another poly-time algorithm with good approximation guarantees



Random Subset $O(k^2 d \log n)$

Go through all k-partitions and calculate least-squares solution in each part and choose the solution with the best loss on the **whole** dataset

$O(1/\sqrt{\log n})$ approximation

Other Algorithms

We have another poly-time algorithm with good approximation guarantees

In practice this can be used as the initialization for AM.

dataset

$O(1/\sqrt{\log n})$ approximation

On Learning Mixture of Linear Regressions in a Non-Realizable Setting, ICML 2022