An Auction Approach to Semi-Supervised Data Classification

Ekaterina Rapinchuk (Merkurjev)

Michigan State University

joint work with Matt Jacobs and Selim Esedoglu

April 26, 2018

DATA CLASSIFICATION (SEGMENTATION)

Problem: Segment a data set into a pre-specified number of clusters, using a small amount of labeled data.

- Applications:
  1. email filtering
  2. medical diagnosis
  3. internet fraud detection
  4. classifying DNA sequences
  5. speech signal segmentation
  6. face recognition
  7. handwritten digits recognition
  8. video tracking
  9. document classification
  10. financial predictions

Goal of the Talk

- Introduce a newly formulated forward and reverse auction method [1] for data classification.

[1] Journal of Computational Physics (2018) (with co-authors M. Jacobs and S. Esedoglu), *Auction Dynamics: A Volume Constrained MBO Scheme.*

Advantages

- Accurate and efficient, compared to state-of-the-art, and requires remarkably little training/labeled data.
- (In)equality volume/class size constraints are incorporated and imposed exactly at every iteration.
- Unconditionally stable; algorithm always terminates with the right properties.

Approach

Input: Data Set

↓

Graphical Representation

↓

Build an Optimization Model on the Graph

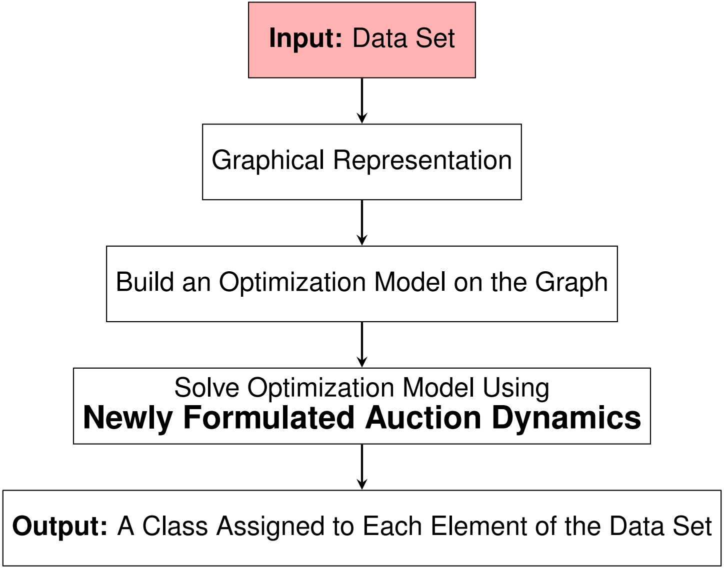
↓

Solve Optimization Model Using
Newly Formulated Auction Dynamics

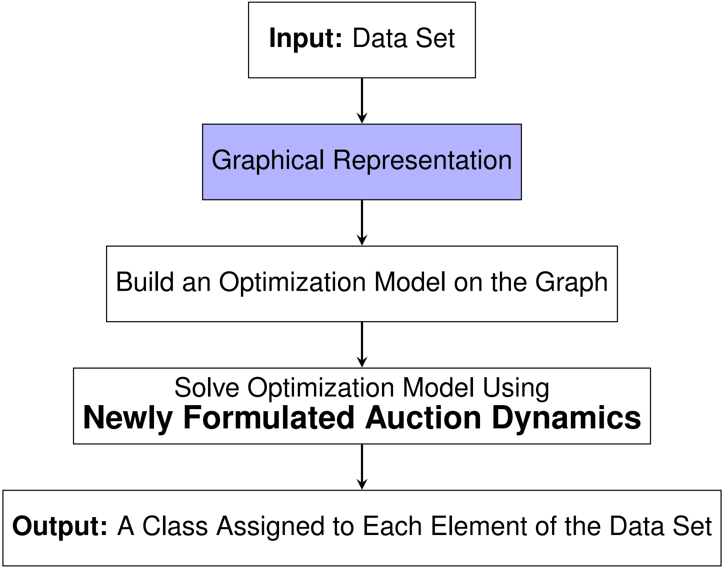
↓

Output: A Class Assigned to Each Element of the Data Set

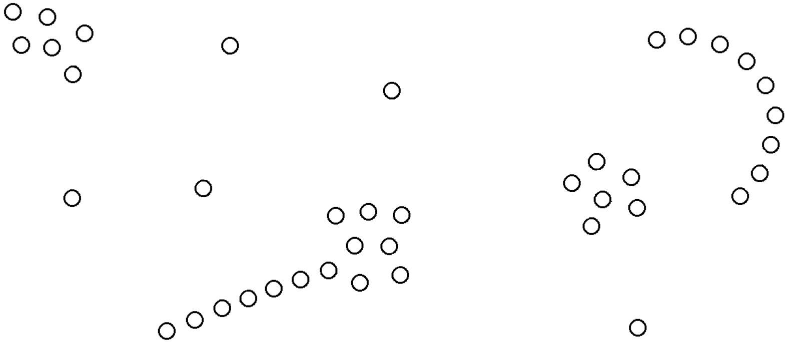
Approach


Input: Data Set
Graphical Representation
Build an Optimization Model on the Graph
Solve Optimization Model Using
Newly Formulated Auction Dynamics
Output: A Class Assigned to Each Element of the Data Set

Approach

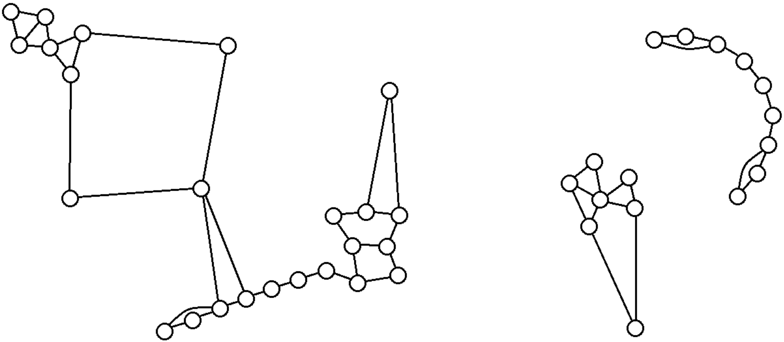

Input: Data Set
Graphical Representation
Build an Optimization Model on the Graph
Solve Optimization Model Using
Newly Formulated Auction Dynamics
Output: A Class Assigned to Each Element of the Data Set

Graphical Framework



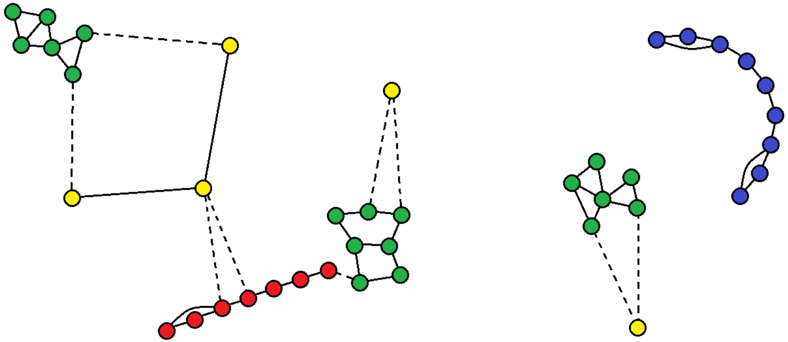
- V: vertices
- E: edges
- n = number of vertices
- N = number of classes

Graphical Framework



- V: set of all points (vertices)
- E: connect certain pairs of vertices (edges)
- Each element of the data set is associated with a vertex.
- A weight function is defined on each edge.

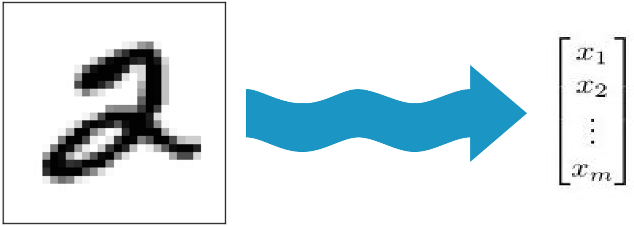
Graphical Framework



- V: set of all points (vertices)
- E: connect certain pairs of vertices (edges)
- Each element of the data set is associated with a vertex.
- A weight function is defined on each edge.

Processing the Data: Step 1

Step 1: Assign to each element of the data set (represented by a vertex on a graph) a vector in $\mathbb{R}^m$, called a FEATURE VECTOR.



Processing the Data: Step 2

Step 2: Use the feature vectors to compute the weight function.

- The weight function w is constructed so that it assigns
  - a big value of $w(x, y)$ for similar elements x and y.
  - a small value of $w(x, y)$ for dissimilar elements x and y.
- An example:

$$w(x, y) = e^{-K(x,y)^2/\sigma},$$

  where
  - $K(x, y)$ = L^2-norm of the difference of the feature vectors of x and y
  - σ is a positive parameter.

Approach

Input: Data Set

↓

Graphical Representation

↓

Build an Optimization Model on the Graph

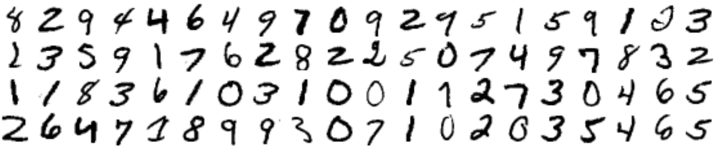
↓

Solve Optimization Model Using
Newly Formulated Auction Dynamics

↓

Output: A Class Assigned to Each Element of the Data Set

Labeled Data and Class Size Information



- Goal: Find an optimal partition $\Sigma = \{\Sigma_1, \Sigma_2, ..., \Sigma_N\}$ of V satisfying two conditions:
- Labeled data F: $F_i \subset F$ is the set of labeled points associated with class i:
$$F_i \subset \Sigma_i \quad \text{for all} \quad 1 \leq i \leq N.$$
- Class size information: we impose the constraints
$$L_i \leq |\Sigma_i| \leq U_i, \tag{1}$$
where L_i and U_i are lower and upper bounds on the class sizes.

Problem

- Direction: find a partition such that vertices belonging to different subsets of the partition are as dissimilar as possible.
- Combining the graph cut with the labeled data and class size constraints, we consider:

$$\arg\min_{\boldsymbol{\Sigma}} \mathrm{Cut}(\boldsymbol{\Sigma}) = \arg\min_{\boldsymbol{\Sigma}} \sum_{i=1}^{N} \sum_{x \in \Sigma_i} \sum_{y \notin \Sigma_i} w(x, y)$$
$$\text{s.t.} \quad F_i \subset \Sigma_i, \quad L_i \leq |\Sigma_i| \leq U_i, \quad (2)$$

where the minimization is over all partitions.

Problem

- First, we rewrite the problem using indicator functions u_i for Σ_i, which indicate the partition. can be described by an indicator functions $u = (u_1, ..., u_n) \ : \ V \mapsto \{0, 1\}^N$:

$$u_i(x) \ := \begin{cases} 1, & x \in \Sigma_i \\ 0, & x \notin \Sigma_i \end{cases} , \quad i = \{1, \ldots, N\} . \tag{3}$$

- The graph cut can be written in u notation as:

$$\mathrm{Cut}(\boldsymbol{u}, w) = \sum_{i=1}^{N} \sum_{x,y \in V} w(x, y) u_i(x)(1 - u_i(y)), \tag{4}$$

- Standard technique: expand the state space of u.

Graph Heat Content (GHC):

$$\mathrm{GHC}(\boldsymbol{u}, w) = \sum_{i=1}^{N} \sum_{x,y \in V} w(x,y) u_i(x)(1 - u_i(y)), \tag{5}$$

where $\boldsymbol{u} : V \to \mathcal{K}_N = \{\boldsymbol{u} : V \to [0,1]^N : \sum_{i=1}^{N} u_i(x) = 1\}$ and $u_i(x)$ represents the probability that x belongs to class i.

- One new approach we have introduced consists of successively minimizing linearizations of the graph heat content under volume/class-size constraints [2]:

$$u^{k+1} = \underset{\{\mathbf{u}:V\setminus F\to[0,1]^N:\sum_{i=1}^N u_i(x)=1\}}{\arg\min} \sum_{i=1}^{N} \sum_{x\in V\setminus F} u_i(x)\Big(\sum_{y\notin\Sigma_i^k} w(x,y)\Big)$$
$$\text{s.t.} \quad L_i - |F_i| \leq \sum_{x\in V\setminus F} u_i(x) \leq U_i - |F_i|, \quad (6)$$

where $\Sigma^k = \{\Sigma_1^k, ...\Sigma_N^k\}$ is a partition obtained using u^k.

- The partition Σ^{k+1} is recovered from u^{k+1}:

$$\Sigma_i^{k+1} = \{x \in V : i = \underset{1\leq j\leq N}{\arg\max}\, u_j^{k+1}(x)\} \quad \text{for all} \quad 1 \leq i \leq N. \quad (7)$$

- Class size constraints make this problem hard!

[2] Journal of Computational Physics (2018) (with co-authors M. Jacobs and S. Esedoglu), *Auction Dynamics: A Volume Constrained MBO Scheme.*

A Standard Technique: Lagrangian Multipliers

- Consider the problem with exact class size constraints, which is less general:

$$u^{k+1} = \underset{\{\boldsymbol{u}:V\setminus F\to[0,1]^N:\sum_{i=1}^N u_i(x)=1\}}{\arg\min} \sum_{i=1}^{N}\sum_{x\in V} u_i(x)\Big(\sum_{y\notin\Sigma_i^k} w(x,y)\Big)$$

$$\text{s.t.} \sum_{x\in V\setminus F} u_i(x) = E_i - |F_i|, \quad (8)$$

- If we incorporate the volume constraints with a Lagrange multiplier $\boldsymbol{\lambda}$, the solution to above is a partition $\boldsymbol{\Sigma}$ given by:

$$\Sigma_i^{k+1} = \{x \in V : i = \underset{1\le i\le N}{\arg\min}\{\Big(\sum_{y\notin\Sigma_i^k} w(x,y)\Big) - \lambda_i^*\}, \quad (9)$$

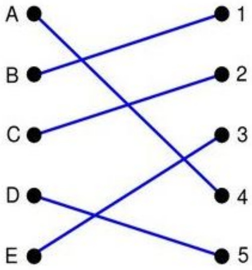
where $\boldsymbol{\lambda}^*$ is the optimal Lagrange multiplier.

A Standard Technique: Lagrangian Multipliers

- While the scheme seems straightforward,
  - computing the optimal Lagrange multiplier $\boldsymbol{\lambda}^*$ is highly challenging for more than 2 classes.
  - We need to consider inequality class size constraints.
- We instead approach our problem by solving its reformulation by novel forward and reverse auction dynamics!

A New Approach

- A new approach: to connect the optimization problem to the ASSIGNMENT PROBLEM, a modified version of which we solve using novel forward and reverse auction dynamics.

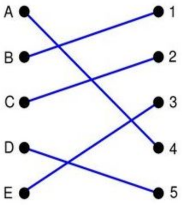


The Assignment Problem

- Given two disjoint sets X and L of equal size r and a value function $a : X \times L \to \mathbb{R}$, the assignment problem seeks to find a one-to-one matching $M = \{(x_1, \ell_1), \ldots, (x_r, \ell_r)\}$ of X and L (i.e. a bijection), such that the total value of the matching

$$\sum_{(x,\ell)\in M} a_\ell(x) \tag{10}$$

is maximized.



The Assignment Problem

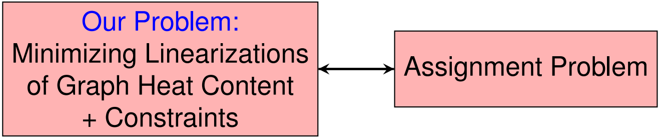
- By using a binary vector $\mathbf{z}$, where $z_\ell(x) = 1$ if (x, ℓ) are matched and $z_\ell(x) = 0$ otherwise, we can rewrite the assignment problem:

$$\operatorname*{arg\,max}_{\mathbf{z}:X\times L\to\{0,1\}} \sum_{x\in X}\sum_{\ell\in L} a_\ell(x) z_\ell(x) \quad \text{s.t.} \sum_{x\in X} z_i(x) = 1, \sum_{i\in L} z_i(x) = 1. \tag{11}$$

- If we relax the binary constraint on $\boldsymbol{z}$ to $\mathbf{0} \leq \boldsymbol{z} \leq \mathbf{1}$, it has been shown that the problem is not changed.

Link

- Let's now link our two problems!!!



Problem

Theorem 1 (JCP(2018), with co-authors M. Jacobs and S. Esedoglu)

The problem of minimizing the linearization of the graph heat content with inequality class size constraints can be expressed as the following modified assignment problem:

$$\underset{\mathbf{z}:V\setminus F\to\mathbb{R},\ \mathbf{0}\leq\mathbf{z}\leq\mathbf{1}}{\arg\max} \sum_{i=1}^{N}\sum_{x\in V} a_i(x)z_i(x)$$

$$s.t. \sum_{i=1}^{N} z_i(x) = 1,$$

$$B_i - |F_i| \leq \sum_{x\in V\setminus F} z_i(x) \leq U_i - |F_i|, \tag{12}$$

where $a_i(x) = 1 - \sum_{y\notin\Sigma_i} w(x,y)$, and z indicates the assignment/classification. [a]

[a] Journal of Computational Physics (2018) (with co-authors M. Jacobs and S. Esedoglu), *Auction Dynamics: A Volume Constrained MBO Scheme*

Intuitive Interpretation of the Modified Assignment Problem

- Each class is an institution that offers a number of memberships.
- Vertices $x \in V$ are people, and each person would like to become a member of some institution.
- The coefficients $a_i(x)$ represent how much person x wants to be a member of class i.
- No person wants to have a membership in more than one class, and each class has a constraint on the number of memberships.

Intuitive Interpretation

- The solution to our modified assignment problem

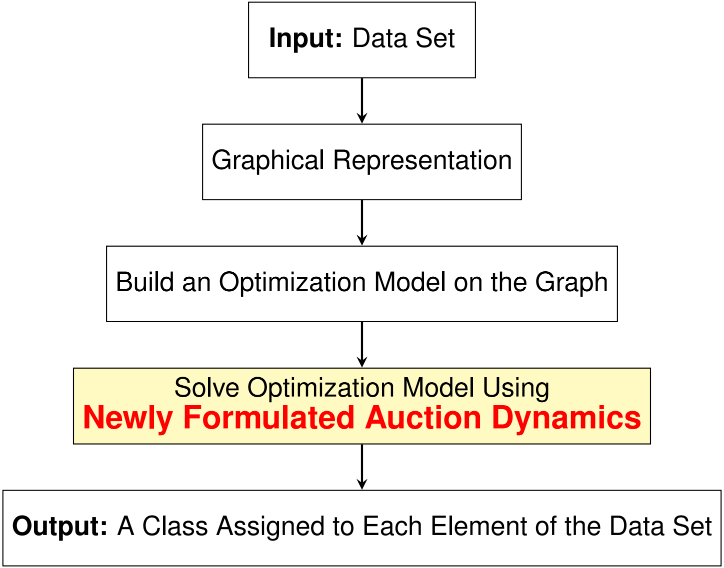
$$\begin{aligned} &\underset{\boldsymbol{z}:V\setminus F\to\mathbb{R},\ \mathbf{0}\le\boldsymbol{z}\le\mathbf{1}}{\arg\max} \quad \sum_{i=1}^{N}\sum_{x\in V} a_i(x)z_i(x) \\ &\text{s.t. } \sum_{i=1}^{N} z_i(x)=1,\ L_i-|F_i|\le \sum_{x\in V\setminus F} z_i(x)\le U_i-|F_i|, \end{aligned} \tag{13}$$

  where $a_i(x) = 1 - \sum_{y\notin\Sigma_i} w(x,y)$, is the matching of people and classes that maximizes the total satisfaction of the population, but still satisfies the constraints.

- Ideally, each person would like to become a member of their favorite class, but this is not possible if more than U_i people want to be members of some class i.

- The main difficulty: how to correctly handle these conflicts.

Approach


Input: Data Set
Graphical Representation
Build an Optimization Model on the Graph
Solve Optimization Model Using
Newly Formulated Auction Dynamics
Output: A Class Assigned to Each Element of the Data Set

Market Mechanism

- Our auction dynamics technique uses a market mechanism:

PRICE

- Each class i has an (evolving) membership price p_i, and if person x is a member of i, then they must pay p_i.
- This can help to resolve conflicts by making the most popular classes more expensive.

INCENTIVE

- Each class can also offer an incentive t_i to attract customers.

STRATEGY

- In general, people will want to buy a membership offering the best value:

$$i^\star \in i_{cs}(x, \boldsymbol{p}) = \underset{1 \le i \le N}{\arg\max}\, a_i(x) - p_i + t_i. \tag{14}$$

Suppose person x is assigned to class offering the best value:
$$i^\star \in i_{cs}(x, \boldsymbol{p}) = \arg\max_{1 \leq i \leq N} a_i(x) - p_i + t_i.$$

↓

Class constraints will probably not be satisfied.

↓

Does there exist an equilibrium price vector $\boldsymbol{p}_*$ and incentive vector $\boldsymbol{t}_*$ that give a feasible matching (i.e satisfies class size constraints)?

↓

Yes! (Theorem)

↓

How do we find the equilibrium prices and incentives of classes?

Duality and Price Vector

- How do we find the equilibrium prices and incentives of classes?
- Answer: DUALITY!!!

Theorem 2 (JCP (2018), with co-authors M. Jacobs and S. Esedoglu)

The equilibrium price vector $\mathbf{p}_*$ and equilibrium incentive vector $\mathbf{t}_*$ that give a feasible matching (with respect to class size constraints) can be obtained from solving the dual of the modified assignment problem [a]*:*

$$\min_{\mathbf{p}\geq \mathbf{0}, \mathbf{t}\geq \mathbf{0}, \boldsymbol{\pi}\in\mathbb{R}^n} \sum_{i=1}^{N} p_i U_i - t_i L_i + \sum_{x\in V} \pi(x) \quad s.t. \quad p_i - t_i + \pi(x) \geq a_i(x), \quad (15)$$

where

- *p = the prices of classes*
- *t = the incentives offered by classes to attract customers.*
- *$\pi(x)$ = auxiliary variable; optimal value is the best deal offered to x by any phase.*

[a] Journal of Computational Physics (2018) (with co-authors M. Jacobs and S. Esedoglu), *Auction Dynamics: A Volume Constrained MBO Scheme*

- Due to the complementary slackness condiction, if a feasible assignment $z^\star$ and dual variables $(p^\star, t^\star, \pi^\star)$ are optimal for their respective problems, then each person x is assigned a membership which satisfies the market strategy; i.e. person x is assigned to class offering the best value:

$$i^\star \in i_{cs}(x, \boldsymbol{p}) = \arg\max_{1 \le i \le N} a_i(x) - p_i^\star + t_i^\star. \tag{16}$$

Forward and Reverse Auction Dynamics

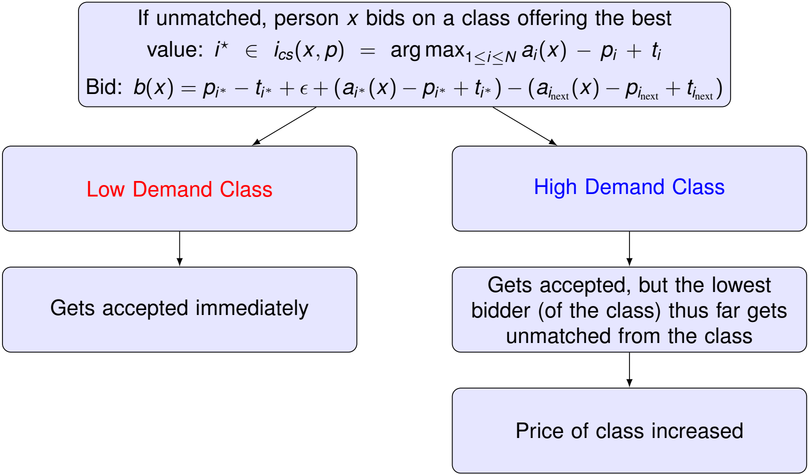
- We propose to solve our modified assignment problem using newly formulated forward and reverse auction dynamics with a market mechanism, inspired by the work of Bertsekas.
- Our algorithm simulates a real-life auction.

Forward and Reverse Auction Dynamics

- Each step of the algorithm either:
  - modifies the prices and incentives of the classes
  - OR increases the number of people matched to a class.

Forward and Reverse Auction Dynamics

- Forward Auction: People bid on the classes.
- Reverse Auction: The classes deficient in members compete for people by providing incentives to attract the necessary number of customers.



- Eventually, the increase of prices of high demand phases will incentivize unmatched people to switch their bid.
- The upper bounds fit nicely into this perspective, but the lower bounds might not be fulfilled and require a reverse auction.

Forward Auction

Algorithm 1: Forward Auction

Input: $\epsilon > 0$, bounds $\boldsymbol{L}$, $\boldsymbol{U}$, coefficients $\boldsymbol{a}$, initial prices $\boldsymbol{p}^0$, initial incentives $\boldsymbol{t}^0$ and people $x \in V$

Result: Prices $\boldsymbol{p}$, admissible incentives $\boldsymbol{t}$, and complete ϵ-CS matching Σ satisfying upper bounds.

Initialization: Mark all x as unassigned, set $\boldsymbol{d} = \boldsymbol{p}^0 - \boldsymbol{t}^0$, set $\Sigma = \varnothing$;

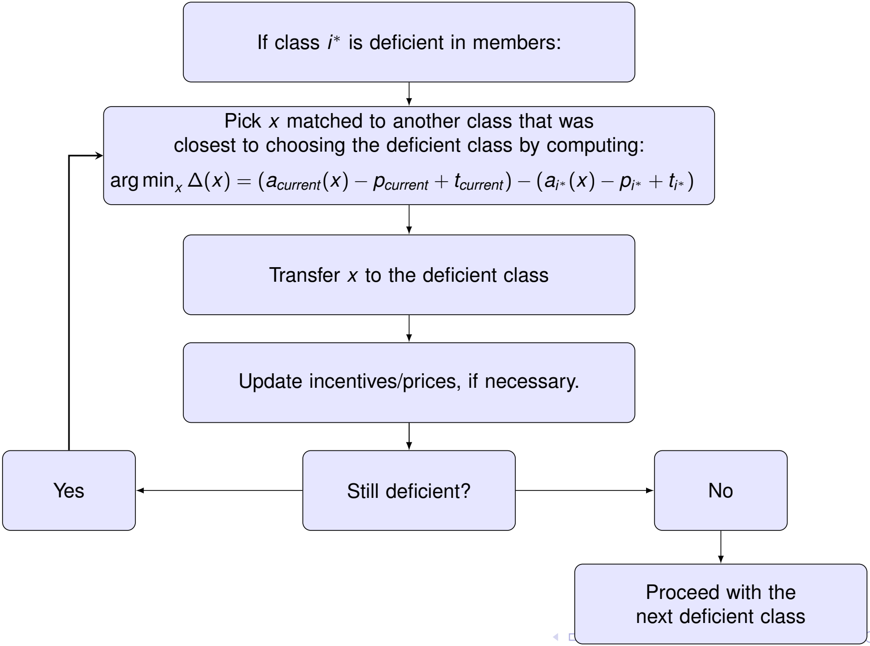
while *some x is marked as unassigned* **do**
- **for** *each unassigned $x \in D_n$* **do**
  - Calculate $i_{cs}(x, \boldsymbol{p})$ and choose some $i^* \in i_{cs}(x, \boldsymbol{d})$;
  - Set $b(x) = d_{i^*} + \epsilon + (a_{i^*}(x) - d_{i^*}) - (a_{i_{\text{next}}}(x) - d_{i_{\text{next}}})$;
  - **if** $|\Sigma_{i^*}| = U_{i^*}$ **then**
    - Find $y = \arg\min_{z \in \Sigma_{i^*}} b(z)$;
    - Remove y from Σ_{i^*} and add x to Σ_{i^*};
    - Mark y as unassigned and mark x as assigned;
    - Set $d_{i^*} = \min_{z \in \Sigma_{i^*}} b(z)$;
  - **else if** $|\Sigma_i| = L_i$ *and* $d_i < 0$ **then**
    - Find $y = \arg\min_{z \in \Sigma_{i^*}} b(z)$;
    - Remove y from Σ_{i^*} and add x to Σ_{i^*};
    - Mark y as unassigned and mark x as assigned;
    - Set $d_{i^*} = \min(\min_{z \in \Sigma_{i^*}} b(z), 0)$;
  - **else**
    - Mark x as assigned and add x to Σ_{i^*};
  - **end**
- **end**

end

Set $\boldsymbol{p} = \max(\boldsymbol{d}, \boldsymbol{0})$, set $\boldsymbol{t} = \max(-\boldsymbol{d}, \boldsymbol{0})$;

return $(\Sigma, \boldsymbol{p}, \boldsymbol{t})$

Part II of the Algorithm: Reverse Auction



Algorithm 2: Reverse Auction

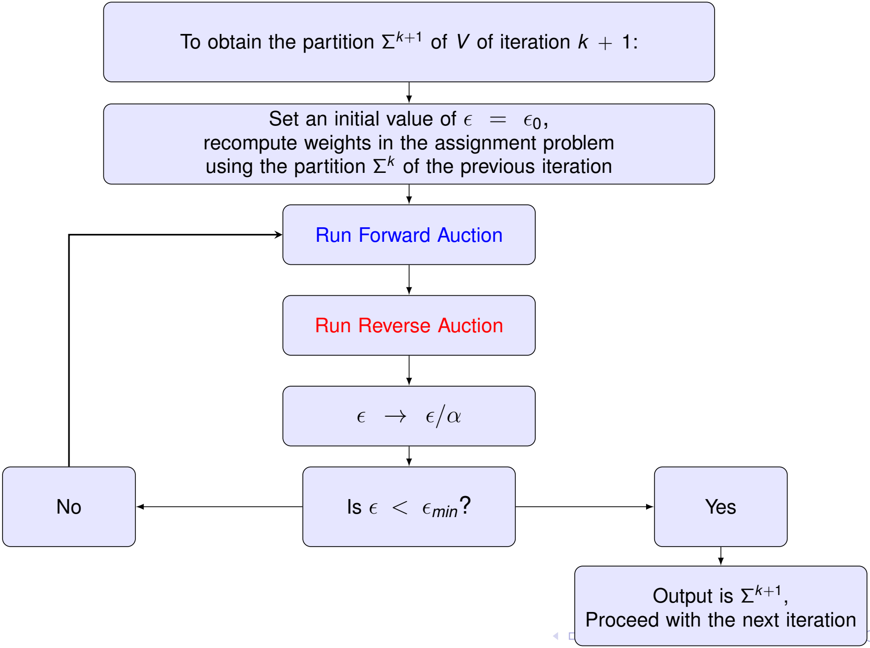
Input: $\epsilon > 0$, bounds $\boldsymbol{L}, \boldsymbol{U}$, coefficients $\boldsymbol{a}$, initial prices $\boldsymbol{p}^0$, initial admissible incentives $\boldsymbol{t}^0$, complete (but possibly lower infeasible) ϵ-CS matching $\boldsymbol{\Sigma}^{\boldsymbol{0}}$

Initialization: Set $\boldsymbol{d} = \boldsymbol{p}^0 - \boldsymbol{t}^0$, set $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}^0$;

Result: complete and feasible ϵ-CS matching and admissible prices and admissible incentives $(\boldsymbol{\Sigma}, \boldsymbol{p}, \boldsymbol{t})$.

```
while there exists some i with (|Σ_i| < U_i and d_i > 0) or (|Σ_i| < L_i) do
    for each i* with (|Σ_i*| < U_i* and d_i* > 0) or (|Σ_i*| < L_i*) do
        for each x ∉ Σ_i* do
            Let j be x's current phase;
            Calculate Δ(x) = (a_j(x) − d_j) − (a_i*(x) − d_i*);
        end
        while (|Σ_i*| < U_i* and d_i* > 0) or (|Σ_i*| < L_i*) do
            Find x ∈ arg min_{y∉Σ_i*} Δ(y);
            if |Σ_i*| < L_i* then
                Remove x from its current phase and add x to Σ_i*;
                if |Σ_i*| = L_i* and Δ(x) ≥ 0 then
                    Subtract Δ(x) + ε from d_i*;
                end
            else
                if Δ(x) + ε ≥ d_i* then
                    Set d_i* = 0;
                else
                    Remove x from its current phase and add x to Σ_i*;
                    if |Σ_i*| = U_i* and Δ(x) ≥ 0 then
                        Subtract Δ(x) + ε from d_i*;
                    end
                end
            end
        end
    end
end
Set p = max(d, 0), set t = max(−d, 0);
return (Σ, p, t)
```

The Algorithm



Algorithm 3: Auction Dynamics with Volume Bounds

Input: Domain V, initial configuration Σ, surface tensions $\boldsymbol{\sigma}$, kernel K, volume bounds $\boldsymbol{L}$, $\boldsymbol{U}$, time step δt, number of steps m, auction error tolerance ϵ_{min}, epsilon scaling factor α, initial epsilon ϵ_0.

Result: Final configuration Σ^m

Initialization: Set $\Sigma^0 := \Sigma$, set $\bar{\epsilon} = \epsilon_{min}/n$;

for *k from* 0 *to* $m-1$ **do**

  Calculate the assignment problem coefficients:

  $\boldsymbol{a}_i^{k+1}(x) = 1 - \sum_{y \notin \Sigma_i^k} w(x, y)$;

  Initialize prices $\boldsymbol{p} = \boldsymbol{0}$, incentives $\boldsymbol{t} = \boldsymbol{0}$, and $\epsilon = \epsilon_0$;

  while $\epsilon \geq \bar{\epsilon}$ **do**

    Run the Forward Auction Algorithm:

    $(\Sigma_{\text{out1}}, \boldsymbol{p}_{\text{out1}}, \boldsymbol{t}_{\text{out1}}) = \text{Forward Auction}(\epsilon, \boldsymbol{L}, \boldsymbol{U}, \boldsymbol{a}^{k+1}, \boldsymbol{p}, \boldsymbol{t}, D_n)$;

    Run the Reverse Auction Algorithm: $(\Sigma_{\text{out2}}, \boldsymbol{p}_{\text{out2}}, \boldsymbol{t}_{\text{out2}}) =$ $\text{Reverse Auction}(\epsilon, \boldsymbol{L}, \boldsymbol{U}, \boldsymbol{a}^{k+1}, \boldsymbol{p}_{\text{out1}}, \boldsymbol{t}_{\text{out1}}, \Sigma_{\text{out1}})$;

    Set $(\boldsymbol{p}, \boldsymbol{t}) = (\boldsymbol{p}_{\text{out2}}, \boldsymbol{t}_{\text{out2}})$;

    Divide ϵ by α;

    if $\epsilon < \bar{\epsilon}$ **then**

      Set $\Sigma^{k+1} = \Sigma_{\text{out2}}$;

    end

  end

end

return Σ^m

Convergence

Theorem 3 (JCP (2018), with co-authors M. Jacobs and S. Esedoglu)

- *The algorithm always terminates with the correct class size properties; it is unconditionally stable with respect to parameters.*

- *The graph-cut energy*

$$\arg\min_{\Sigma} Cut(\Sigma) = \frac{1}{2}\sum_{i=1}^{N}\sum_{x\in\Sigma_i}\sum_{y\notin\Sigma_i} w(x,y)$$

$$s.t. \quad F_i \subset \Sigma_i, \quad L_i \leq |\Sigma_i| \leq U_i. \quad (17)$$

decreases with each iteration of auction dynamics. [a]

[a] Journal of Computational Physics (2018) (with co-authors M. Jacobs and S. Esedoglu), *Auction Dynamics: A Volume Constrained MBO Scheme*

Worst Case Time Complexity

Theorem 4 (JCP (2018), with co-authors M. Jacobs and S. Esedoglu)

- *For a fixed ϵ,*
  - *the worst case time complexity of the forward auction is $O(nN(\log(n) + N)(C + G)/\epsilon)$.*
  - *the worst case time complexity of the reverse auction is $O(n^2N^2(C + G)/\epsilon)$.* [a]
- *Here,*

$$C = \max_{i\in\{1,\ldots,N\}, x\in V} a_i(x)$$

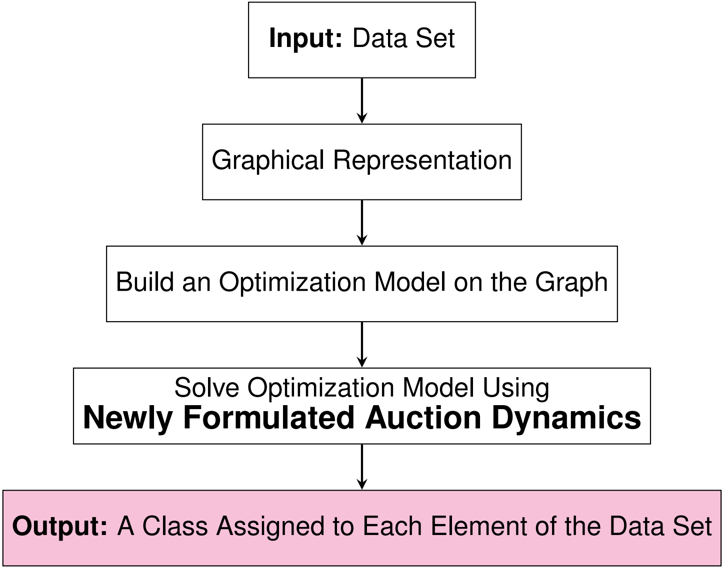
$$G = \max_{i\neq j}(p_j^0 - t_j^0) - (p_i^0 - t_i^0)$$

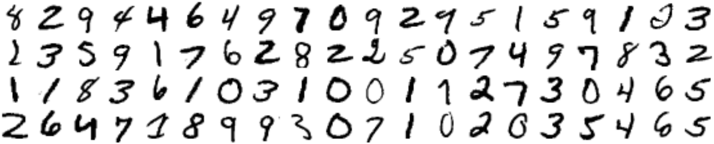
$$n = \text{\# of vertices}$$

$$N = \text{\# of classes}$$

[a] Journal of Computational Physics (2018) (with co-authors M. Jacobs and S. Esedoglu), *Auction Dynamics: A Volume Constrained MBO Scheme*

Approach


Input: Data Set
Graphical Representation
Build an Optimization Model on the Graph
Solve Optimization Model Using
Newly Formulated Auction Dynamics
Output: A Class Assigned to Each Element of the Data Set



- consists of $70,000$ 28×28 images of handwritten digits.

Table: MNIST Results

Labeled Nodes	exact size constraints	small gap between L_i and U_i	no size constraints
0.05%	**94.84%**	**93.17%**	**83.49%**
0.1%	**96.88%**	**95.87%**	**93.16%**
0.5%	**97.38%**	**97.20%**	**97.19%**
1.0%	**97.43%**	**97.31%**	**97.30%**

Results-Optdigits Data Set

- is a database of 5620 handwritten digits.

Table: Optdigits Results.

Labeled Nodes	exact size constraints	small gap between L_i and U_i	no size constraints
0.4%	**93.04%**	**91.70%**	**85.29%**
0.5%	**95.96%**	**94.66%**	**89.76%**
0.75%	**98.07%**	**96.62%**	**94.68%**
1%	**98.39%**	**97.14%**	**96.33%**

Conclusion

- We have derived an auction dynamics technique for data classification.
- Some of the advantages of the method include:
  - requires remarkably little training/labeled data.
  - unconditional stability; the algorithm always terminates with the right properties.
  - ability to incorporate class size constraints and labeled data.
  - high accuracy
- The auction technique is very flexible: applied it to volume-constrained mean curvature motion.

Thank you!