PADERBORN
UNIVERSITY

PADERBORN UNIVERSITY

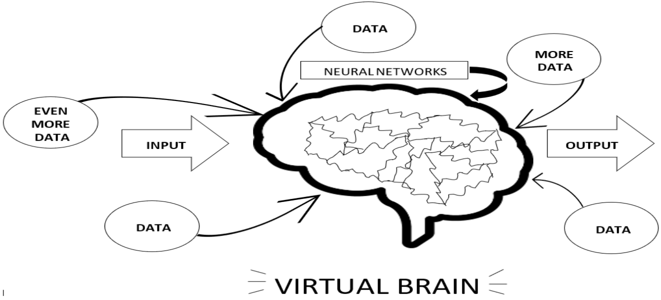
LEARNING TO SORT

MACHINE LEARNING

Prakhar Rathi

Paderborn, Thursday 6th August, 2020

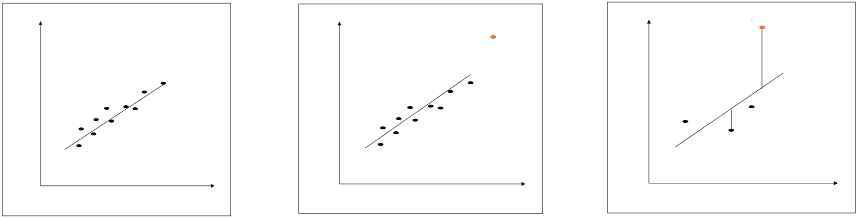
Introduction



- Sorting refers to arranging items in an ordered sequence.

Sorting: Intuition

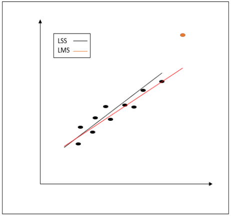
- Originally, Least Sum of Squares Regression: $\frac{\sum_{i=0}^{N}(x_i - \mu)^2}{N}$



- Example: Input data $= 2, 3, 5, 6 \implies$ mean $= 4$
- Input data with an outlier $= 2, 3, 5, 6, 30 \implies$ mean = $= 9.2$

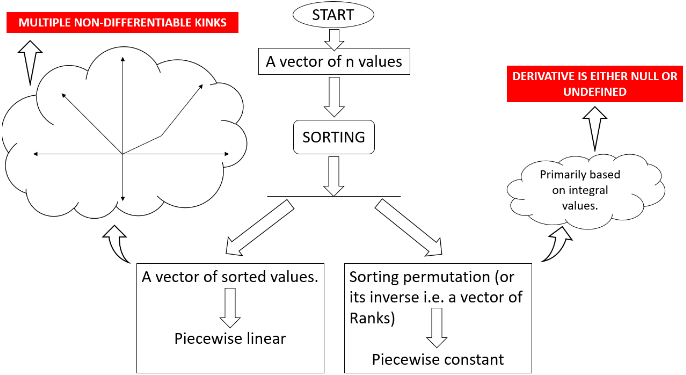
Sorting: Intuition

- Replace by Least Median of Squares Regression: $\underset{i}{\text{median}}\ (x_i - \mu)^2$



- Input data $=$ 2, 3, 5, 6, 30 $\implies$ median = $=$ 5.
- Sorting an array $\theta := (\theta_1, \theta_2, \ldots, \theta_n)$ means finding a permutation σ such that $\theta_\sigma := (\theta_{\sigma 1}, \theta_{\sigma 2}, \ldots, \theta_{\sigma n})$ is in an increasing order [1].

Gradients of Sorting



Permutation Matrix

- The output of any sorting algorithm can be viewed as a permutation matrix.
- A permutation matrix is a square matrix ($n \times n$) with entries in $\{0, 1\}$.
- For example, input vector $= \{5, 3, 6, 2\}$
  $\implies$ Sorting permutation $= \{4, 2, 3, 1\}$
  and the corresponding permutation matrix is:

0	0	0	1
0	1	0	0
0	0	1	0
1	0	0	0

Unimodal Row Stochastic Matrices

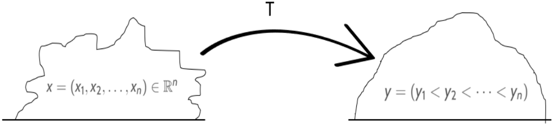
- Replace the permutation matrix with a Unimodal Row Stochastic Matrix [2].

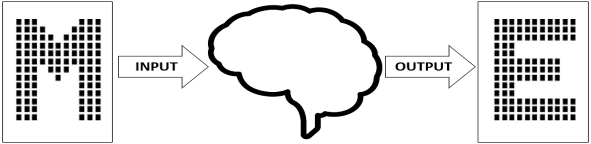
$\frac{3}{8}$	$\frac{1}{8}$	$\frac{1}{2}$
$\frac{3}{4}$	$\frac{1}{4}$	0
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$

- Sum of each row $= 1$.
- Every row has a distinct $\arg\max$.

Optimal Transport

- Idea [1]:



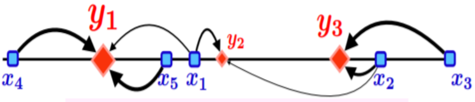


Learning problem:

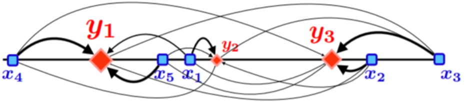
- Minimum number of ways to arrange n-points as the letter E from the input pattern M.

Optimal Transport

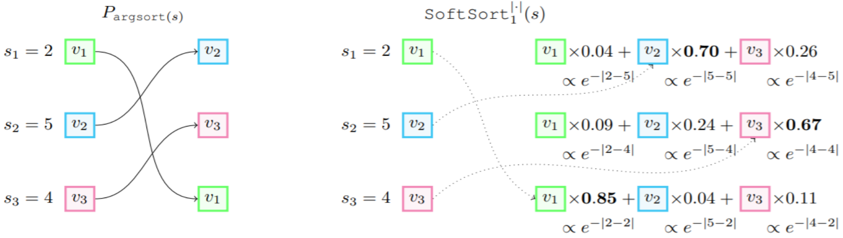
- y is already sorted $\implies$ its corresponding permutation matrix is the identity permutation.
- Derived sorted vectors form continuous ranks for the elements in x [1].



- The optimal regularized transport plan is a dense matrix $\implies$ ensures differentiability everywhere w.r.t x.



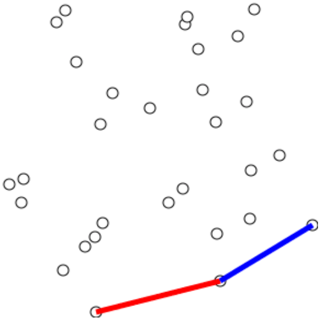
Relaxing the `arg sort` operator



Properties of `SoftSort` [3]:

- It is row-stochastic.
- Converges to $P_{argsort(.)}$.
- Can be projected onto a permutation matrix.

Projections on the Permutahedron

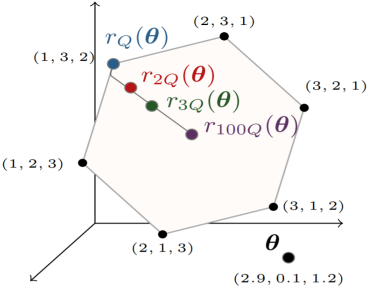


- Permutahedron = convex hull of all permutations.
- Previous approaches do not achieve the desired time complexity.
- Can cast sorting as a linear program over the permutahedron.
- The optimal solution is always a vertex, i.e. a permutation.

Projections on the Permutahedron

- A combination of Quadratic and Euclidean regularization is applied, $\psi \in \{Q, E\}$ [4].
- Soft operators:

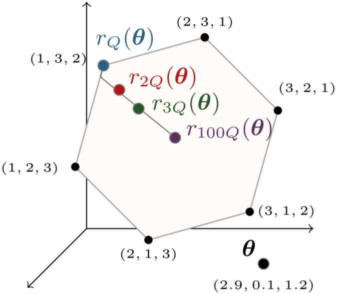
$$s_{\epsilon\psi}(\theta) := P_{\epsilon\psi}(\rho, \theta) \text{ and,}$$
$$r_{\epsilon\psi}(\theta) := P_{\epsilon\psi}(-\theta, \rho) \text{ where}$$
$$\rho = (n, n-1, n-2, \ldots, 1).$$

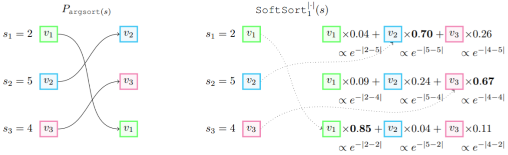


- When $\epsilon \to \infty \implies r_{\epsilon Q}(\theta)$ converges towards the centroid of the permutahedron [4].

Conclusion

- Generally, sorting is not differentiable everywhere.
- Various differentiable proxies have been devised.
- Projections onto the permutahedron achieve the desired $O(n \log n)$ time complexity.





References

1. Cuturi, M., Teboul, O., and Vert, J., 2019. Differentiable Sorting using Optimal Transport: The Sinkhorn CDF and Quantile Operator. arXiv:1905.11885 [cs. LG].
2. Grover, A., Wang, E., Zweig, A., and Ermon, S., 2019. Stochastic optimization of sorting networks via continuous relaxations. arXiv preprint arXiv:1903.08850, 2019.
3. Prillo, S. and Eisenschlos, J., 2020. SoftSort: A Continuous Relaxation for the argsort Operator.
4. Blondel, M., Teboul, O., Berthet, Q., and Djolonga, J., 2020. Fast differentiable sorting and ranking. arXiv preprint arXiv:2002.08871, 2020.

Thank you for your attention!