Introduction to Optimal Transport and its Applications to Computational Biology

Anthony Christidis

Journal Club
Computational Biology Group

Core for Computational Biology
Department of Biomedical Informatics
Harvard Medical School

January 23, 2025

Outline

Relevant Materials

- **Theoretical and Computational Optimal Transport**:
  - "**Optimal transport for single-cell and spatial omics**" by Charlotte Bunne, Geoffrey Schiebinger, Andreas Krause, Aviv Regev and Marco Cuturi. [1]
  - "**Computational Optimal Transport**" by Gabriel Peyré and Marco Cuturi. [5]
  - "**Optimal Transport for Applied Mathematicians**" by Filippo Santambrogio. [6]
- **Software Packages**:
  - `transport` [7]
  - `POT` [3]
  - `scDiagnostics` [2]

Basics of Optimal Transport

- Optimal Transport (OT) focuses on the most efficient way to move mass between distributions.
- Connections to diverse fields: probability, statistics, optimization, functional analysis, and many more.

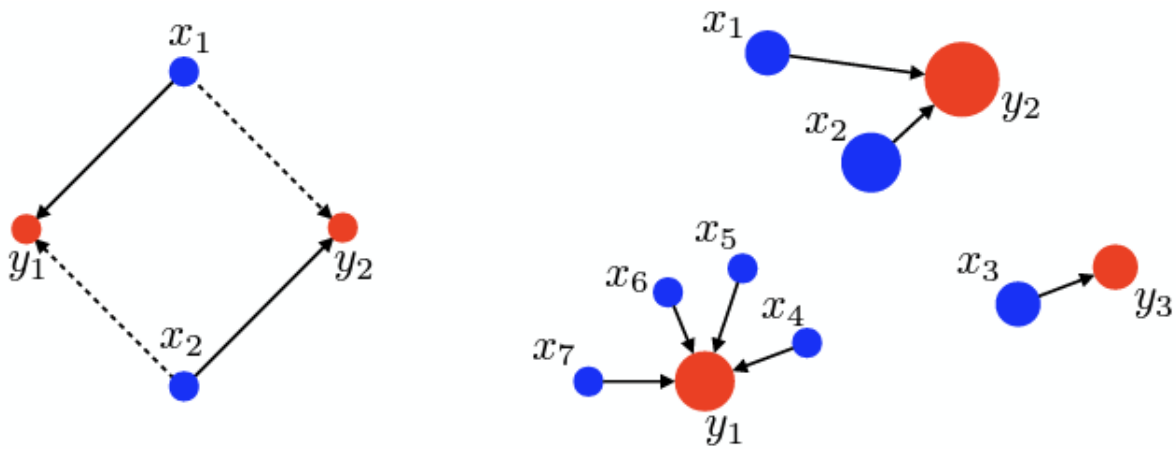
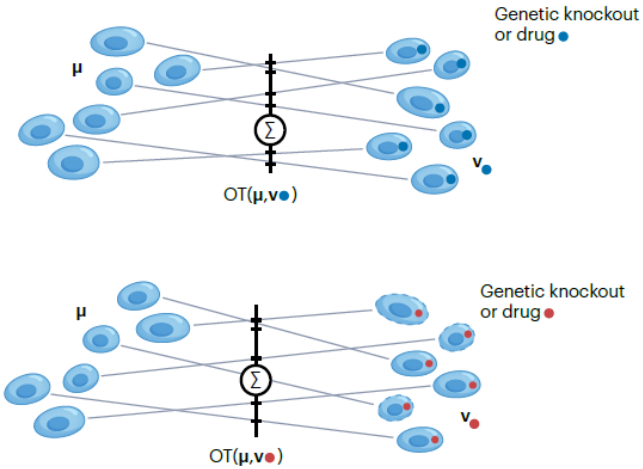


Figure: Example discrete OT problem solutions. [5]

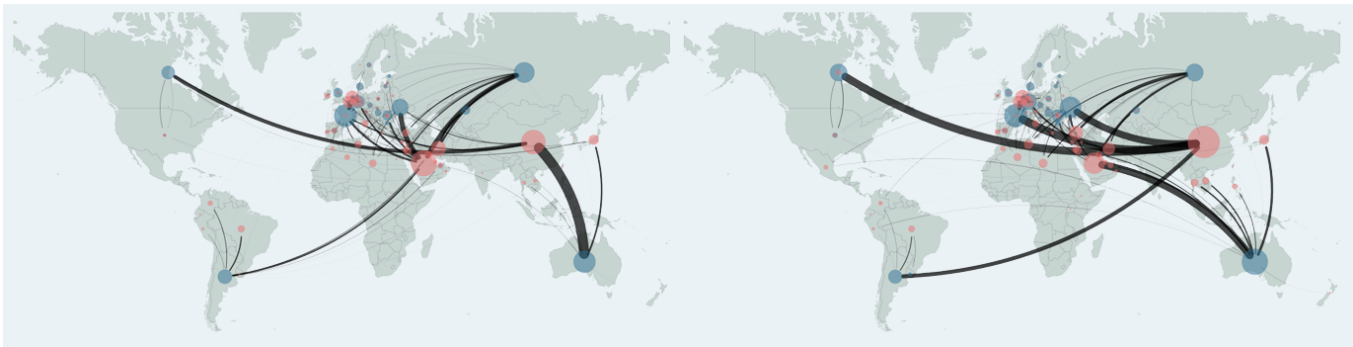
Basics of Optimal Transport

- Areas of applications: **computational biology**, image processing, economics, and many more.



(a) Computational Biology [1]

(b) Image Processing [6]



(c) Economics [4]

Basics of Optimal Transport

- Fundamental problem: Moving mass from one distribution to another at minimal cost.

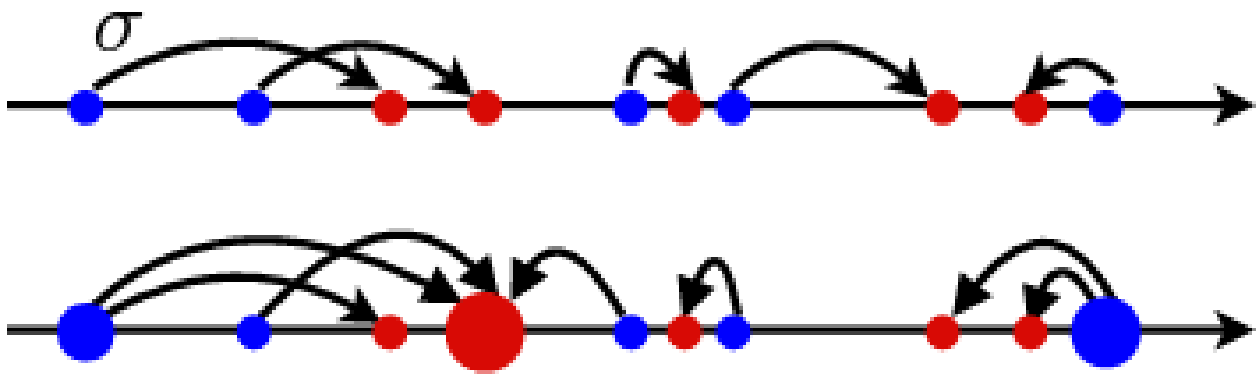


Figure: Fundamental problem in OT. [5]

Discrete OT: Mathematical Background

- **Source Distribution** (μ): $\mu = \sum_{i=1}^{n} a_i \delta_{x_i}$
  - a_i: Mass or probability at source point x_i.
  - δ_{x_i}: Dirac delta function, representing mass concentrated at x_i.
- **Target Distribution** (ν): $\nu = \sum_{j=1}^{m} b_j \delta_{y_j}$
  - b_j: Mass or probability at target point y_j.
  - δ_{y_j}: Dirac delta function, representing mass concentrated at y_j.
- **Dirac Delta Function** (δ_{x_i}):
  - Models an idealized point mass.
- **Measure Theory**:
  - Studies objects like μ and ν, defines how mass is distributed.

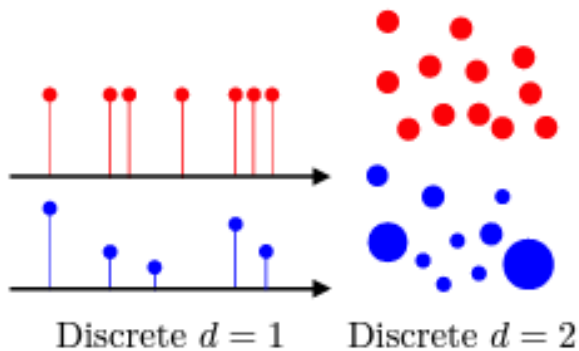


Figure: Example of probability mass functions. [5]

Discrete OT: Transport Plan and Objective

- **Transport Plan** (π): Matrix π_{ij} representing mass flow from x_i to y_j.
- **Objective**:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} \pi_{ij} c(x_i, y_j)$$

- **Constraints**:

$$\sum_{j=1}^{m} \pi_{ij} = a_i, \quad \forall i$$

$$\sum_{i=1}^{n} \pi_{ij} = b_j, \quad \forall j$$

$$\pi_{ij} \geq 0, \quad \forall i, j$$

- Cost function $c(x_i, y_j)$ is often the squared distance $\|x_i - y_j\|^2$.

Entropy Regularization in Discrete OT

- **Objective with Entropy Regularization**: Enhances computational efficiency and stability.

$$\min_{\pi} \left(\sum_{i=1}^{n} \sum_{j=1}^{m} \pi_{ij} c(x_i, y_j) + \epsilon \sum_{i=1}^{n} \sum_{j=1}^{m} \pi_{ij} (\log \pi_{ij} - 1) \right)$$

  - ϵ: Regularization tuning parameter.
- **Benefits of Regularization**:
  - Induces sparsity in π.
  - Improves numerical stability in high-dimensional problems.
  - Faster convergence in optimization problems.
- **Optimization Methods**:
  - Sinkhorn-Knopp Algorithm
  - Other Methods: iterative Bregman projections, stochastic optimization, augmented lagrangian methods

Discrete OT: Dual Formulation

- The dual formulation involves maximizing:

$$\sum_{i=1}^{n} u_i a_i + \sum_{j=1}^{m} v_j b_j$$

- Subject to:

$$u_i + v_j \leq c(x_i, y_j) \quad \forall i, j$$

- Provides a complementary perspective on the transport problem.

Monge vs. Kantorovich in OT

- **Monge Problem**: Seeks a deterministic map T with $T_{\#}\mu = \nu$, i.e. a bijective mapping with no splitting (so it has more constraints than the traditional/Kantorovich OT).
- **Kantorovich Problem**: Uses a transport plan π allowing mass splitting.
- Kantorovich is more flexible and widely applicable, handling broader cases where direct mappings (as in Monge) aren't possible.
- Monge required more computationally intensive methods from combinatorics.

Monge Problem Example

- **Source Points**: $\{x_1, x_2, x_3, x_4, x_5\}$
- **Target Points**: $\{y_1, y_2, y_3, y_4, y_5\}$
- **Complex Cost Function**:

$$\text{Cost}(x_i, y_j) = (i - j)^2 + 2i + 3j$$

- **Objective**: Find a mapping $T : X \to Y$ minimizing the cost.

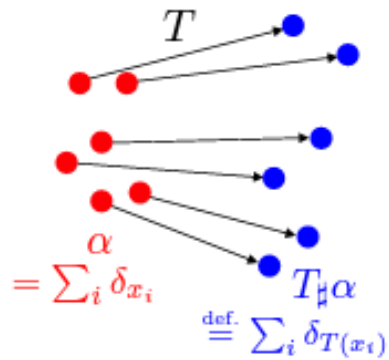


Figure: Bijective mapping example. [5]

Comparison of Solutions

Solution 1:

$$T(x_1) = y_3, T(x_2) = y_4, T(x_3) = y_5, T(x_4) = y_1, T(x_5) = y_2$$

Total Cost for Solution 1:

- Cost = 102

Solution 2:

$$T(x_1) = y_2, T(x_2) = y_3, T(x_3) = y_4, T(x_4) = y_5, T(x_5) = y_1$$

Total Cost for Solution 2:

- Cost = 98

Conclusion:

- Solution 2 has lower cost (98) compared to Solution 1 (102).
- Finding the optimal solution is very computationally expensive.

Kantorovich Problem Example

- **Source Points** with masses:
  - $x_1 : 0.2$, $x_2 : 0.1$, $x_3 : 0.3$, $x_4 : 0.2$, $x_5 : 0.2$
- **Target Points** with masses:
  - $y_1 : 0.15$, $y_2 : 0.25$, $y_3 : 0.25$, $y_4 : 0.15$, $y_5 : 0.2$
- **Cost Function**:

$$\text{Cost}(x_i, y_j) = \begin{cases} 1, & \text{if } i = j \\ 4, & \text{if } i \neq j \end{cases}$$

- **Objective**: Minimize cost with transport plan allowing splits.

Kantorovich Problem Solution

Transport Plan (π):

$$\pi = \begin{bmatrix} 0.15 & 0.05 & 0 & 0 & 0 \\ 0 & 0.05 & 0.05 & 0 & 0 \\ 0 & 0.2 & 0.1 & 0 & 0 \\ 0 & 0 & 0.1 & 0.1 & 0 \\ 0 & 0 & 0 & 0.05 & 0.15 \end{bmatrix}$$

- Reflects optimal mass distribution.
- Allows splitting for more flexibility and lower cost.
- Calculation involves sum of $\pi_{ij} \cdot \text{Cost}(x_i, y_j)$.

Discrete OT: Code Applications

- State-of-the-art solvers in R/CRAN package `transport`.
- See accompanying `R` script.
- Also available in Python via package POT.

Continuous OT: Introduction

- **Extension from Discrete to Continuous**:
  - Generalizes optimal transport to manage continuous probability distributions.
  - Useful for modeling and analyzing scenarios where data is naturally continuous.
- **Source and Target Measures**:
  - μ: Source measure, representing the initial "mass" distribution over space X.
  - ν: Target measure, representing the desired mass distribution over space Y.

Continuous OT: Mathematical Formulation

- **Transport Plan Set**:
  - $\Pi(\mu, \nu)$: Collection of all feasible transport plans π that shift μ into ν while conserving mass.
- **Cost Function**:
  - $c(x, y)$: Represents the cost of moving a unit of mass from $x \in X$ to $y \in Y$.
- **Objective**:
  - Minimize total transport cost:

$$\inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} c(x, y)\, d\pi(x, y)$$

  - inf: Infimum, denotes the greatest lower bound of the total cost.
  - $\int_{X \times Y} c(x, y)\, d\pi(x, y)$: Integral, representing expected transport cost under plan π.

Solving Continuous OT Problems

- **Problem Setup**:
  - Minimize integral-based cost:

$$\inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} c(x,y)\, d\pi(x,y)$$

- **Numerical Methods**:
  - **Discretization**: Transform continuous measures into discrete approximations, linking directly to discrete OT techniques.
  - **Sinkhorn-Knopp Algorithm**:
    - Extends discrete OT regularization for efficiency.
  - **Gradient-Based Approaches**: Leverages derivatives to iteratively approach solutions.
- **Theoretical Tools**:
  - **Kantorovich Duality**: Bridges discrete and continuous OT solutions, using dual variables.
- **Applications**: Extends discrete OT use cases in data science and ML.

Continuous OT Example: Problem Setup

- **Source Distribution (μ)**:
  - Gaussian centered at $(0, 0)$ with variance $\Sigma_1 = I$ (identity matrix).
- **Target Distribution (ν)**:
  - Gaussian centered at $(1, 1)$ with variance $\Sigma_2 = I$.
- **Cost Function**:
  - Squared Euclidean distance: $c(x, y) = \|x - y\|^2$.
- **Objective**:
  - Find transport map $T(x)$ that minimizes:

$$\inf_T \int_{\mathbb{R}^2} \|x - T(x)\|^2 \, d\mu(x)$$

  - Subject to mapping μ to ν.

Continuous OT Example: Solution

- **Optimal Transport Map via Brenier's Theorem**:
  - For Gaussian distributions with quadratic cost, the map is affine:
$$T(x) = Ax + b$$
  - Given $\Sigma_1 = \Sigma_2 = I$, set $A = I$.
- **Translation Vector (b)**:
  - Shift from source to target mean:
$$b = (1, 1) - (0, 0) = (1, 1)$$
- **Result**:
  - Optimal map:
$$T(x) = x + (1, 1)$$
  - Each point is shifted right and up by 1 to align with ν.
- **Explanation**:
  - Map $T(x)$ realigns means without variance adjustment, ideal for equal covariance identity matrices.

Continuous OT: Code Application

- See accompanying `R` script.
- In the continuous code example, we verify our intuition via simulation.
- See plot produced at the end of this example.

Comparing Single-Cell Expressions: Traditional Metrics

- **Limitations of Euclidean Distance**:
  - Sensitive to magnitude differences; lacks gene context scaling.
  - Overlooks functional relationships between genes.
- **Limitations of Correlation Metrics (Pearson/Spearman)**:
  - Emphasizes linear or monotonic relationships.
  - Misses complex interaction and shifts between differing gene profiles.

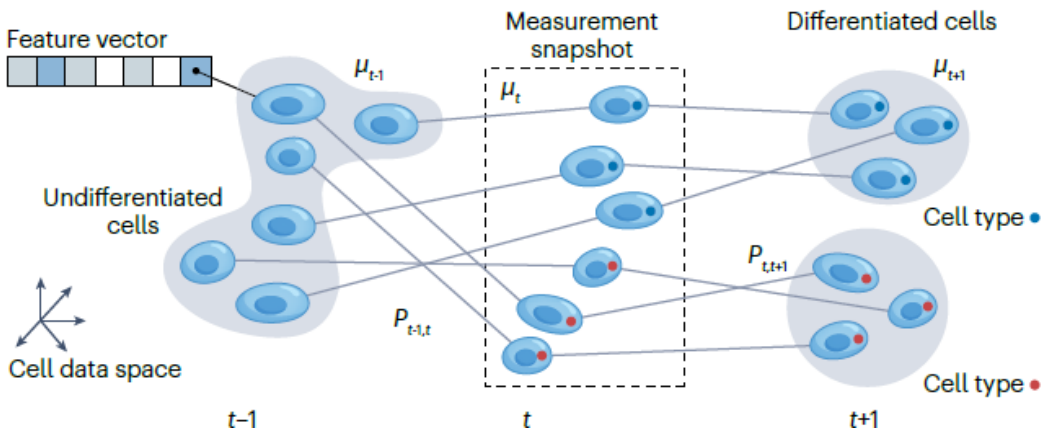


Figure: Cell differentiation relationship. [1]

Optimal Transport: A Flexible Alternative

- **Why Use OT for Single-Cell Comparison?**
  - Flexible cost functions to transition between cell expression profiles.
  - Models biologically relevant transitions in gene pathways and cellular dynamics.
- **Context**:
  - Interested in the evolution of cell populations.
  - OT allows adjustments to the cost matrix, aligning transformation likelihood with gene expression changes.

Motivating Example: Gene Interaction in Immune Response

- **Gene A (Cytokine Receptor)**:
  - Essential for receiving environmental signals—low cost in transitions.
- **Gene B (Transcription Factor)**:
  - Works closely with Gene A to activate immune response genes—low joint transformation cost with Gene A.
- **Gene C (Cell Surface Marker)**:
  - Expression increases in activated cells but costly to adjust in conjunction with B or A independently.
  - Higher transformation cost with both A and B reflects its distinct biological role.
- **Biological Insight**:
  - Shows how OT can prioritize coupled transformations within a pathway (A & B) while maintaining overall functional coherence by recognizing C's selective role.

Interpreting OT in Single-Cell Context

- **OT Advantages**:
  - Enables comparisons considering gene-specific contexts and evolutionary pathways.
  - Cost matrix adjustments reflect biological transformation costs, aligning with cellular evolution likelihood.

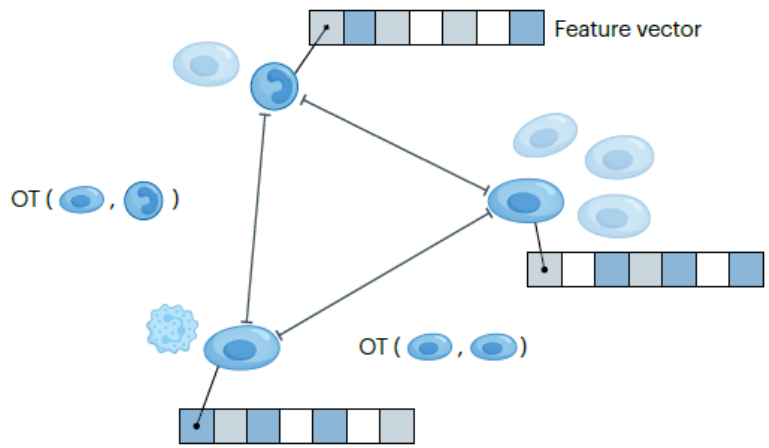


Figure: OT distances for cells. [1]

Spatial Omics: Research Questions

- **Key Biological Questions**:
  - Understanding cell–cell communication and spatial distribution of molecules.
  - Identifying key structural and functional units like microenvironments (MEs) and niches.

Optimal Transport in Spatial Omics

- **Using OT for Analyzing Microenvironments**:
  - OT distance for analysis of multicellular communities.
  - Models the microenvironment (ME) around each cell by aggregating spatial neighbor features into histograms.
- **Computational Approach**:
  - Compute OT distance between MEs: $OT(ME_i, ME_j)$.
  - Forms a pairwise distance matrix for all cellular MEs.

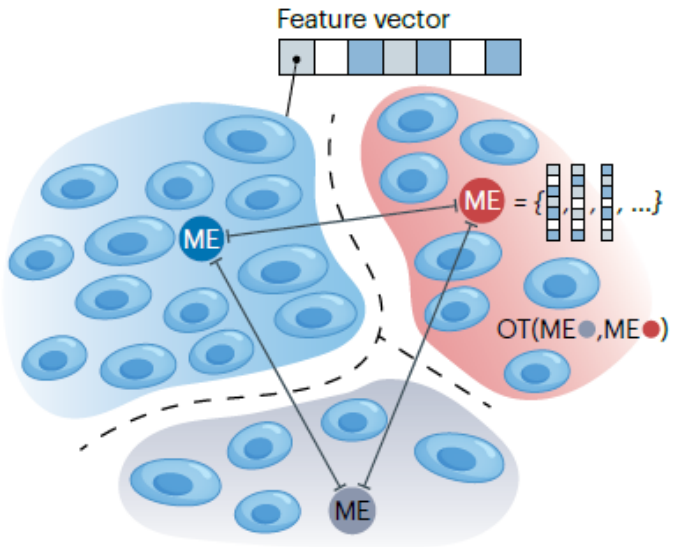


Figure: OT distances of MEs. [1]

Applications and Biological Insights

- **Clustering of Microenvironments**:
  - Apply clustering on the OT-derived distance matrix to identify cellular niches.
- **Real-World Findings**:
  - Studies by Yuan et al. and Mani et al. exemplify the use of OT in spatial omics.
  - OT-based MEs resemble known tissue structures when using multiplex fluorescence data.
- **Visual Confirmation**:
  - Detected microenvironments align with ground-truth tissue sections, corroborating OT's efficacy in characterizing spatial structures.

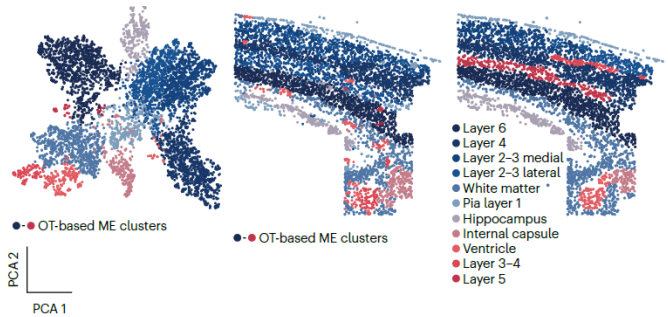


Figure: Clustering visualization of MEs. [1]

Optimal Transport in `scDiagnostics`

- **Purpose**:
  - Calculates Wasserstein distances (another name for Kantorovich distances) between query and reference single-cell datasets.
- **Process Overview**:
  - Projects query data into PCA space defined by the reference dataset.
  - Computes Wasserstein distances within the reference dataset to establish a null distribution.
  - Assesses differences by calculating distances between the query and reference datasets.
- **Biological Context**:
  - Identifies variations in cell populations or expression profiles.

Optimal Transport in `scDiagnostics`

- **Biological Context**:
  - Identifies variations in cell populations or expression profiles.
  - Provides insights into cell type differentiation and evolutionary pathways.
- **Key Outputs**:
  - `null_dist`: A numeric vector of distances from resampled reference dataset pairs.
  - `query_dist`: Mean distance between the query and reference datasets.
  - `cell_type`: Lists unique cell types identified in the reference dataset.

Optimal Transport in `scDiagnostics` Example

- **Data Preparation**:
  - Load `reference_data` and `query_data`.
  - Select CD4 cells from both datasets.
- **Gene Selection**:
  - Extract top 500 highly variable genes and find common genes.
- **Dimensionality Reduction**:
  - Perform PCA on reference data for feature reduction.
- **OT Distance Calculation**:
  - Compute Wasserstein distances and evaluate differences.
- **Visualization**:
  - Plot to compare datasets.
- **Output**:
  - See code example in `R` script.

Conclusion

- **Optimal Transport (OT)**:
  - Provides a versatile framework for comparing complex biological data.
- **Single-Cell Analysis**:
  - Offers a flexible approach to understand cell-type differences and evolutionary dynamics.
- **Spatial Omics**:
  - Enables characterization of microenvironments, revealing key tissue structures.
- **Broader Impact**:
  - OT bridges gaps between computational methods and biological insights, driving advances in multi-omics research.

References

[1] Charlotte Bunne et al. "Optimal transport for single-cell and spatial omics". In: *Nature Reviews Methods Primers* 4.1 (2024), p. 58.

[2] Anthony Christidis et al. *scDiagnostics: Cell type annotation diagnostics*. R package version 1.1.0. 2025. URL: https://github.com/ccb-hms/scDiagnostics.

[3] Rémi Flamary et al. "POT: Python Optimal Transport". In: *Journal of Machine Learning Research* 22.78 (2021), pp. 1–8. URL: http://jmlr.org/papers/v22/20-451.html.

[4] Thomas Gaskin et al. "Modelling Global Trade with Optimal Transport". In: *arXiv preprint arXiv:2409.06554* (2024).

[5] Gabriel Peyré, Marco Cuturi, et al. "Computational optimal transport: With applications to data science". In: *Foundations and Trends® in Machine Learning* 11.5-6 (2019), pp. 355–607.

[6] Filippo Santambrogio. "Optimal transport for applied mathematicians". In: *Birkäuser, NY* 55.58-63 (2015), p. 94.