Trajectory Analysis

Pseudotime Inference

Order cells along developmental paths

Branching Processes

Identify cell fate decisions

Monocle Algorithm

Reverse graph embedding

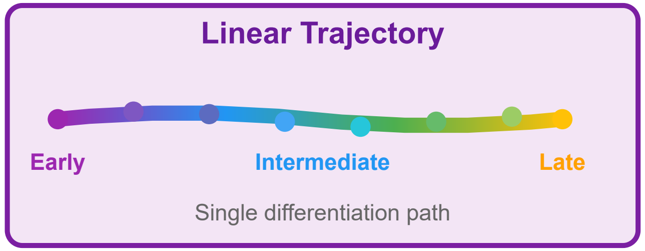
Slingshot Method

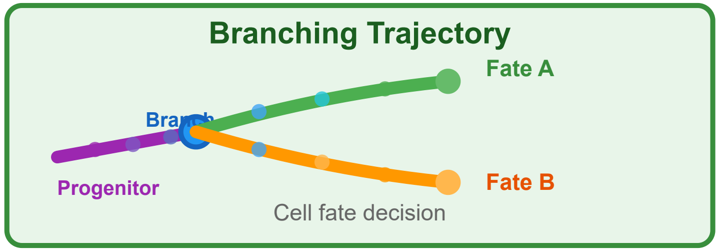
Cluster-based trajectory inference

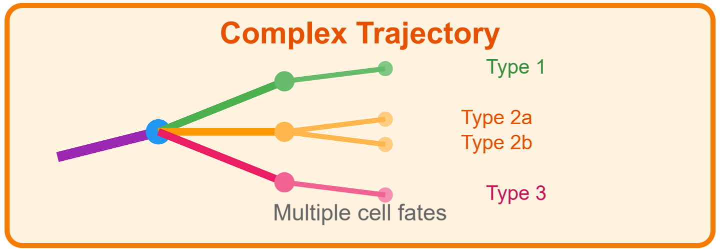
Validation Approaches

Known genes, time-series data

💡 Assumes continuous progression - verify biological relevance







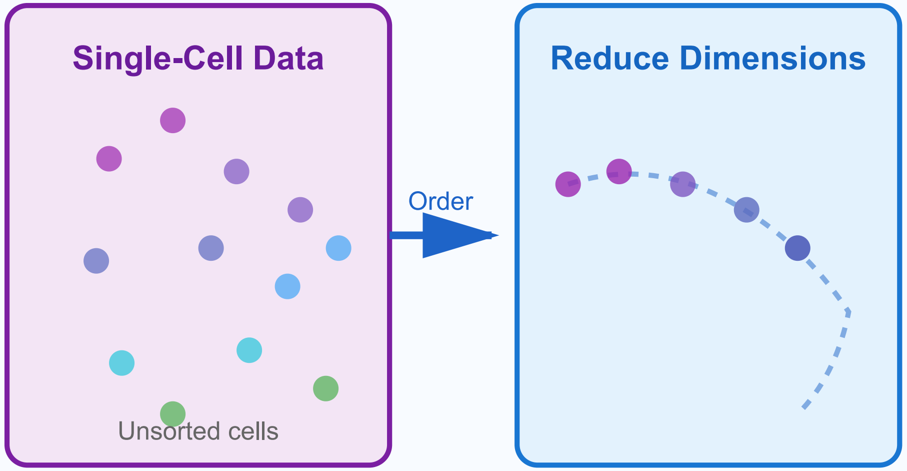
1 Pseudotime Inference

Pseudotime inference is a computational method that orders cells along a continuous trajectory based on their transcriptional similarity, creating a "virtual timeline" of cellular development. Unlike chronological time, pseudotime represents the progression of cells through a biological process such as differentiation, development, or response to stimuli.

Core Principles

- **Dimensionality Reduction:** High-dimensional gene expression data is projected into lower-dimensional space using PCA, t-SNE, or UMAP
- **Trajectory Construction:** Cells are ordered along a path representing progression through a biological process
- **Temporal Ordering:** Each cell receives a pseudotime value indicating its relative position along the trajectory
- **Gene Expression Dynamics:** Identifies genes whose expression changes smoothly along pseudotime

Pseudotime Inference Process



⚠ Key Considerations

- Assumes cells progress along a smooth, continuous trajectory
- Requires sufficient cellular diversity to capture the full process
- May not capture complex, non-linear dynamics or cyclic processes
- Results should be validated with known biological markers

2 Branching Processes

Branching processes identify critical decision points where progenitor cells commit to distinct developmental fates. These bifurcation events represent moments when cellular populations diverge into separate lineages, each characterized by unique gene expression programs.

Key Components

- **Branch Point Detection:** Algorithms identify locations where cell populations split in trajectory space
- **Lineage Assignment:** Cells are assigned to specific branches based on transcriptional profiles
- **Bifurcation Analysis:** Statistical methods determine the significance and timing of branching events
- **Branch-Specific Genes:** Identification of genes that drive or mark distinct cell fates

📊 Real-World Example: T Cell Development

During T cell maturation in the thymus, double-positive (CD4+CD8+) thymocytes reach a critical branch point where they commit to either CD4+ helper T cells or CD8+ cytotoxic T cells. This decision is marked by differential expression of transcription factors like ThPOK (CD4+ fate) and Runx3 (CD8+ fate).

⚠ Key Considerations

- Branch points must be validated with functional assays or known biology
- Technical noise can create false branching patterns
- Requires adequate cell numbers in each branch for reliable inference
- Temporal resolution affects ability to detect transient branch points

3 Monocle Algorithm

Monocle is a pioneering computational tool that uses reverse graph embedding to reconstruct complex cellular trajectories from single-cell RNA-seq data. Developed by the Trapnell lab, Monocle learns a principal graph structure that captures the underlying developmental or temporal progression of cells.

Algorithm Workflow

- **Feature Selection:** Identifies highly variable genes that capture biological variation
- **Dimensionality Reduction:** Uses reversed graph embedding or UMAP to project cells
- **Principal Graph Learning:** Constructs a tree-like structure representing the trajectory
- **Pseudotime Calculation:** Assigns each cell a position along the learned trajectory
- **Differential Expression:** Identifies genes with significant expression changes along pseudotime

Monocle Versions

- **Monocle 1:** Introduced ICA for dimensionality reduction and minimum spanning tree for trajectory construction
- **Monocle 2:** Implemented reversed graph embedding and DDRTree algorithm for complex branching
- **Monocle 3:** Utilizes UMAP and partition-based graph abstraction for scalability on large datasets

📊 Real-World Example: Myoblast Differentiation

Monocle has been successfully applied to study skeletal muscle development, ordering myoblasts through their differentiation into mature myocytes. The algorithm identified key regulatory genes such as MYOD1, MYOG, and MYH3 that are upregulated during differentiation, while proliferation markers like MKI67 decrease.

⚠ **Key Considerations**

- Requires careful parameter tuning for optimal trajectory learning
- Computational complexity increases with dataset size (Monocle 3 addresses this)
- Root cell selection impacts pseudotime ordering—use biological priors when possible
- Works best with datasets containing clear developmental progression

4 Slingshot Method

Slingshot is a cluster-based trajectory inference method that learns smooth, continuous lineage structures from single-cell data. Unlike methods that learn trajectories directly from individual cells, Slingshot first identifies cell clusters and then fits smooth curves through these clusters, making it robust to noise and computationally efficient.

Algorithm Components

- **Cluster Identification:** Uses existing clustering methods like k-means or hierarchical clustering
- **Minimum Spanning Tree:** Constructs a tree connecting cluster centroids to represent global structure
- **Lineage Construction:** Identifies paths from root to terminal clusters as potential lineages

- **Principal Curves:** Fits smooth curves through cells along each lineage for refined representation
- **Cell Weights:** Assigns weights to cells for each lineage based on distance to the curve

Key Advantages

- **Cluster-Based Approach:** Reduces noise by working with cluster summaries
- **Flexible Integration:** Works with any clustering and dimensionality reduction method
- **Multiple Lineages:** Can identify and model multiple diverging trajectories simultaneously
- **Cell Weights:** Provides uncertainty estimates for cell assignment to lineages
- **Computational Efficiency:** Scales well to large datasets

📊 Real-World Example: Pancreatic Development

Slingshot has been applied to study pancreatic endocrine cell differentiation, where pancreatic progenitors diverge into multiple hormone-producing cell types (alpha, beta, delta, and PP cells). The method successfully identified the branching structure and revealed transcriptional programs driving each cell fate decision, including key roles of NKX6.1, ARX, and PAX4.

⚠️ Key Considerations

- Trajectory quality depends heavily on the quality of initial clustering
- Requires specification or inference of root cluster (starting point)
- May oversimplify trajectories if clusters don't capture biological transitions well

- Best suited for datasets with clear cluster structure along developmental paths

5 Validation Approaches

Validating trajectory inference results is critical to ensure that computational predictions reflect true biological processes rather than technical artifacts. Since pseudotime is inferred rather than directly measured, multiple complementary validation strategies should be employed.

Validation Strategies

- **Known Marker Genes:** Check if genes with established temporal expression patterns match pseudotime predictions
- **Time-Series Data:** Compare pseudotime ordering with actual collection time points when available
- **Functional Validation:** Perform perturbation experiments on key predicted regulators
- **Cross-Method Comparison:** Run multiple trajectory inference algorithms and compare results
- **RNA Velocity:** Use splicing information to validate trajectory directionality
- **Lineage Tracing:** Compare with experimental lineage tracking data when available

Marker Gene Validation

- **Early Markers:** Should have high expression in early pseudotime (e.g., stem cell markers like OCT4, NANOG)

- **Late Markers:** Should have high expression in late pseudotime (e.g., differentiation markers)
- **Stage-Specific Markers:** Should show expected temporal patterns at specific developmental stages
- **Correlation Analysis:** Calculate correlation between known marker expression and pseudotime

Time-Series Validation

- **Collection Time Comparison:** Verify that pseudotime ordering correlates with actual sample collection time
- **Temporal Coherence:** Cells collected at similar times should have similar pseudotime values
- **Directionality Check:** Ensure trajectory direction matches biological progression
- **Kendall's Tau:** Statistical measure of concordance between pseudotime and real time

📊 Real-World Example: Neural Differentiation Validation

In a study of neural differentiation from embryonic stem cells, researchers validated pseudotime trajectories by confirming that: (1) pluripotency markers (OCT4, SOX2) decreased along pseudotime, (2) neural progenitor markers (PAX6, SOX1) peaked at intermediate pseudotime, and (3) mature neuronal markers (TUBB3, MAP2) increased in late pseudotime. This pattern matched both time-series experiments and known biology.

Advanced Validation Methods

- **RNA Velocity:** Uses spliced vs unspliced mRNA ratios to predict future cell states and validate trajectory direction
- **CRISPR Screens:** Perturbation of predicted key regulators should alter trajectory structure
- **Genetic Lineage Tracing:** Direct comparison with barcode-based lineage tracking experiments
- **Multi-Omics Integration:** Validate with protein expression, chromatin accessibility, or metabolomics data

⚠ Key Considerations

- Use multiple independent validation approaches for robust conclusions
- Consider biological context—not all predicted patterns may be functionally relevant
- Validation with time-series data is gold standard but not always available
- Negative results (mismatches) can reveal technical issues or unexpected biology
- Document validation methods thoroughly for reproducibility

📚 Best Practice: Always combine multiple validation approaches and document assumptions about biological processes when interpreting trajectory analysis results.