**Supported Interaction**

**Dimension Reduction**
- Change parameters
- Filter data points
- Move data points

**Clustering**
- Change parameters
- Filter data points
- Change membership of data points
- Split/Merge clusters
- Freeze clusters

**Interactive Dimension Reduction with PIVE**

- **Hard replacement interactions** with k-means. At the sixth iteration, we perform cluster merging (green rectangles) and splitting (a purple rectangle) interactions, and the final result due to these interactions is converged in 26th iterations. 1,671 HCI conference papers published between 1999 and 2010 have been used.

**Interactive Clustering with PIVE**

**Motivation**

- **Humans’ perceptual precision**
  - Visual perception does not require highly precise outputs, e.g., \( \pi = 3.14159265358 \).

- **Iterative behavior of machine learning**
  - Many computational methods are iterative.
  - A major improvement typically occurs in early iterations.

- Let’s cut through algorithm iterations to save time for interactive visualization.

**Visual Analytics and Machine Learning**

- However, as ML methods become more advanced, their high costs hinder real-time interactive visualizations with them. Even the state-of-the-art in visual analytics adopts only a few standard techniques and does not properly leverage advanced ML.

**Per-Iteration Visualization Environment (PIVE)**

- **The standard vs. PIVE workflow**
  - Previously...
    - Treat ML as a black box
    - Wait till algorithms finish
  - with PIVE
    - Interact with intermediate result
    - No additional cost
    - Applicable to various iterative methods

**Interaction Methodology**

1. **User replaces intermediate output during algorithm iterations.**
   - 1) Soft replacement
     - Replaced output works as a new initialization.
   - 2) Hard replacement
     - Replaced output remains same.

2. **Algorithm 1 Iterative methods**
   - 1: \( X = \{x_1, \ldots, x_n\} \) and parameter \( \alpha \)
   - 2: \( Y = \{y_1, \ldots, y_m\} \)
   - 3: \( t \leftarrow 0 \)
   - 4: Initialize \( Y'^t = \{y'_1, \ldots, y'_m\} \)
   - 5: repeat
     - 6: \( t \leftarrow t + 1 \)
     - 7: \( ^* \) Per-iteration routine \(*\)
     - 8: for \( i = 1, \ldots, n \) do
       - 9: \( y'^t_i = f(X, Y'^{t-1} \}^*, \alpha) \)
     - 10: \( Y'^t = \{y'_1, \ldots, y'_n\} \)
     - 11: until a stopping criterion is satisfied
     - 12: \( T \leftarrow t \) \# Final iteration index \(*\)
     - 13: \( Y \leftarrow Y'^T \) \# Final output \(*\)

3. **Replace \( y'^t_i \leftarrow y_i \) for this iteration (soft) for all iterations (hard)**

**Stability and Convergence**

- One needs to know when visualization becomes stable.
- PIVE explicitly visualizes intermediate changes in charts and visual encoding.

**Computational Overhead**

- Constant processing of intermediate output
- We use multithreading (two threads for computation and visualization).