Interactive Dimensionality Reduction Methods for Visualization

Minh Vu and Benoît Frénay

Dimensionality Reduction (DR) methods in an Interactive Context

- DR method: an unsupervised learning technique to reduce the number of dimensions of a multivariate dataset while preserving some of its important characteristics.
- Can be used for visualizing a high dimensional dataset, but having some issues:
  - Sometimes, it is hard to interpret the visualization results.
  - The algorithms can make errors but we cannot correct them without interacting directly with the system.
- Research questions:
  - How to integrate human knowledge into the DR methods?
  - How are the cognitive feedbacks from users translated to parametric constraints in the DR algorithms?

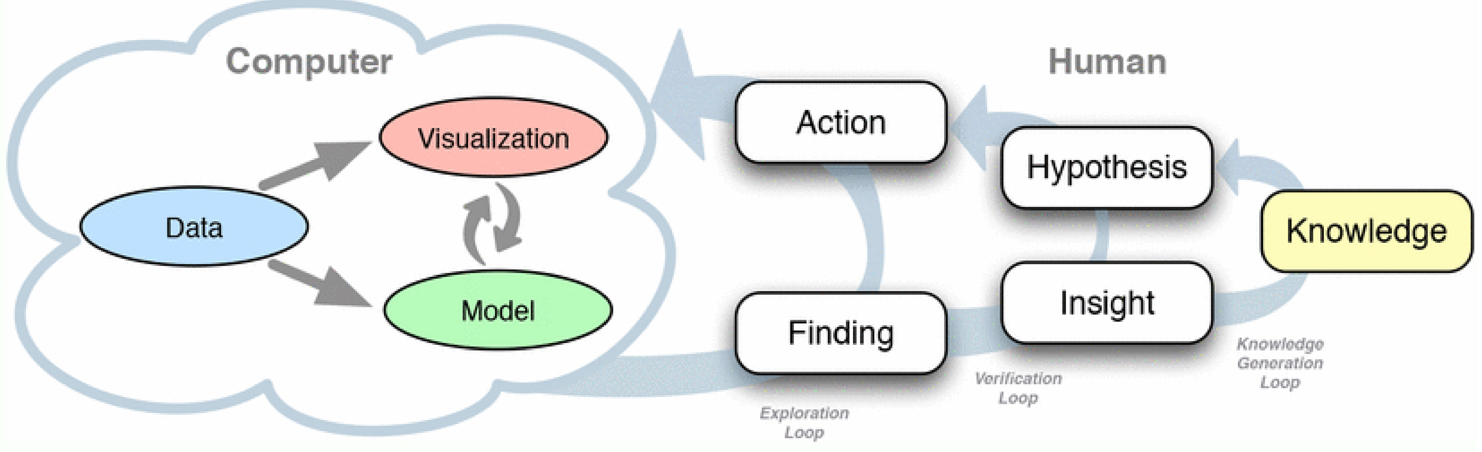


Figure: Visual analytics with Human-in-the-loop [4, 5]

Different approaches for integrating user constraints

- Interactive feedbacks from users or experts can be seen as constraints for the DR methods.
- **Instance-level** [A], **group-level** [B], **feature-level** [C], **dataset-level** [D] constraints.

Feature exploration ([A], [C])

- Moving points to see how the values of their features change.
- Understanding which features determine the position of point in the visualization.

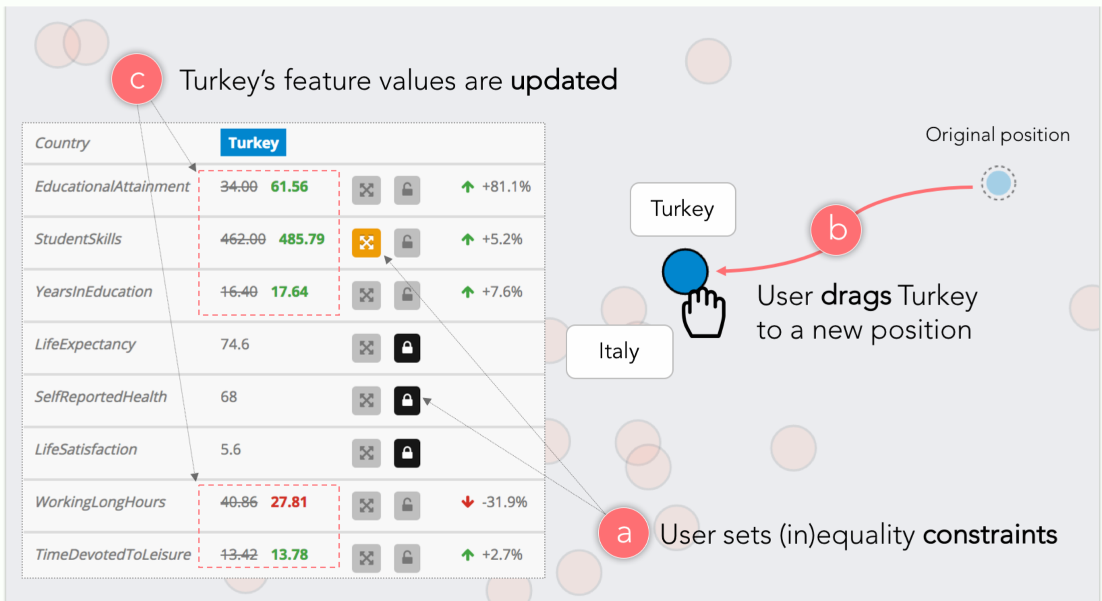


Figure: Forward and Backward Projections [1]

Triplet constraints ([A], [B])

- `Triplet` (i, j, k): object i seems more similar to object j than i does to object k
- More compact than `Must link`, `Cannot link`.
- *Concep embedding* combines t-SNE and Crowd-Kernel Embedding methods, can help experts interactively explore and label the dataset.

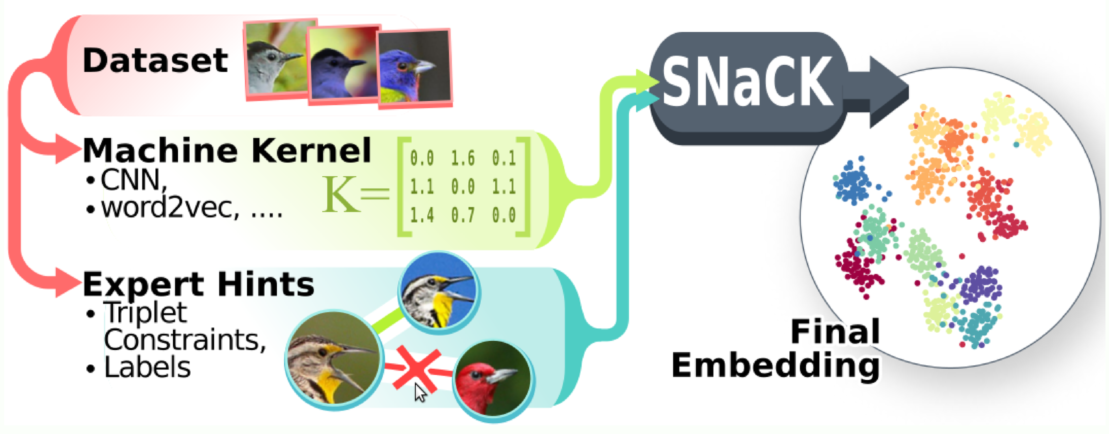


Figure: Stochastic Triplet Embedding [6]

Example-based constraints ([B],[C],[D])

- Using examples to guide the algorithm to construct the understandable axes.

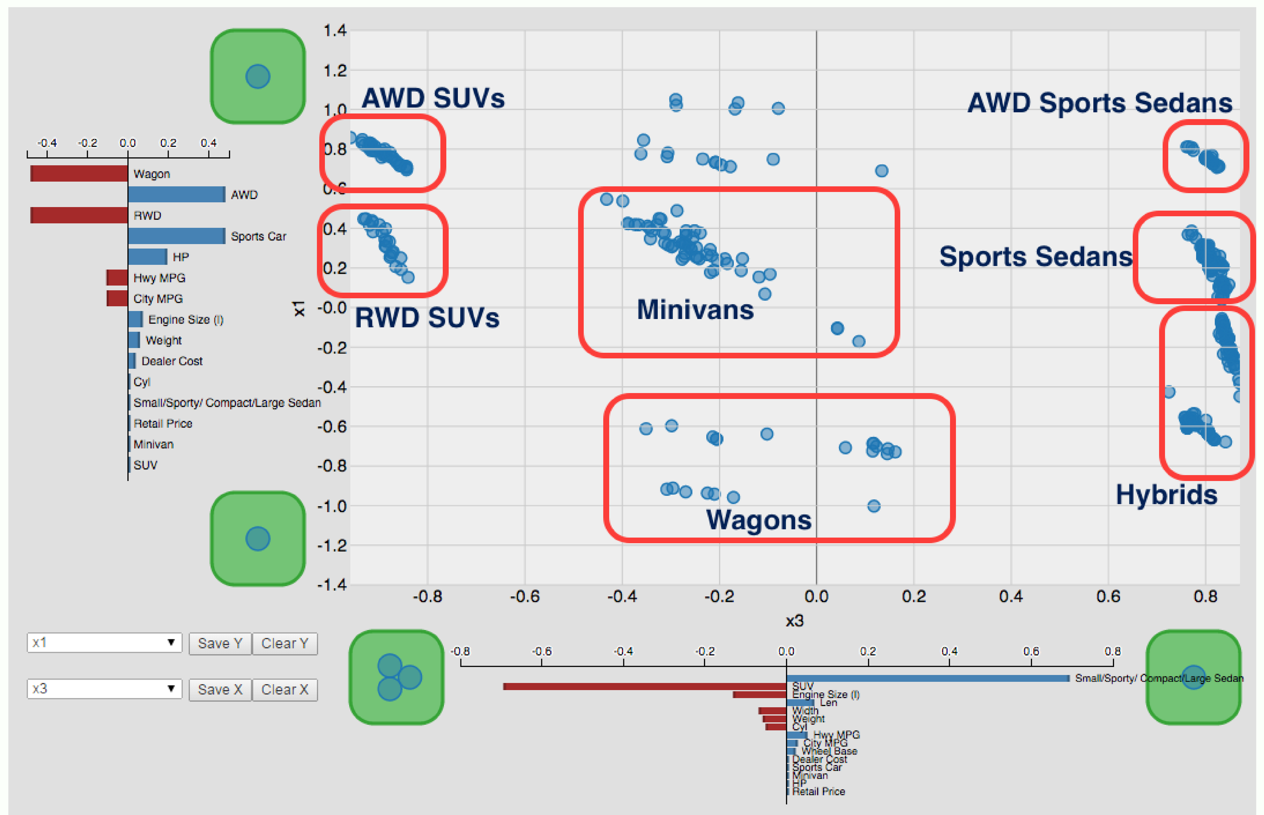
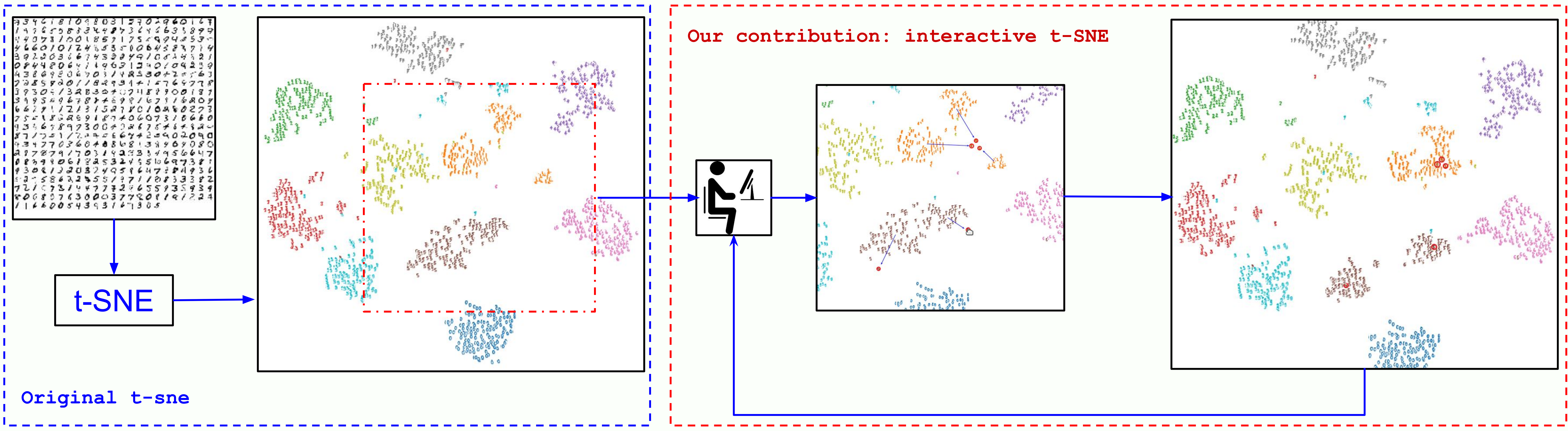


Figure: InterAxis: Steering Scatterplot Axes [2]

Proposed interactive t-SNE method



- Based on t-SNE (t-Distributed Stochastic Neighbor Embedding) [3].
- Goal: Preserve neighborhood information: the points that are neighbors in high dim. space will still be neighbors in low dim. space.
- Point-moving constraints: user can move points to control the groups:
  - Move points far apart to divise a large cluster.
  - Move points close together to merge some small, similar clusters.
- How it works: Add a penalty term to the objective function to force the neighbors of the selected points follow these points when they are moved.
- Work in progress:
  - Choosing the important points to move.
  - Find a parameter-free and interpretable penalty term.

References

[1] Marco Cavallo et al. "Exploring Dimensionality Reductions with Forward and Backward Projections". In: *arXiv preprint arXiv:1707.04281* (2017).

[2] Hannah Kim et al. "InterAxis: Steering Scatterplot Axes via Observation-Level Interaction". In: *IEEE Transactions on Visualization and Computer Graphics* (2016).

[3] Laurens van der Maaten et al. "Visualizing data using t-SNE". In: *Journal of Machine Learning Research* (2008).

[4] Dominik Sacha et al. "Knowledge generation model for visual analytics". In: *IEEE transactions on visualization and computer graphics* (2014).

[5] Dominik Sacha et al. "Visual Interaction with Dimensionality Reduction: A Structured Literature Analysis". In: *IEEE Transactions on Visualization and Computer Graphics* (2017).

[6] Laurens Van Der Maaten et al. "Stochastic triplet embedding". In: IEEE. 2012.

https://github.com/minhvvu/idr
{vuvietminh,benoit.frenay}@unamur.be