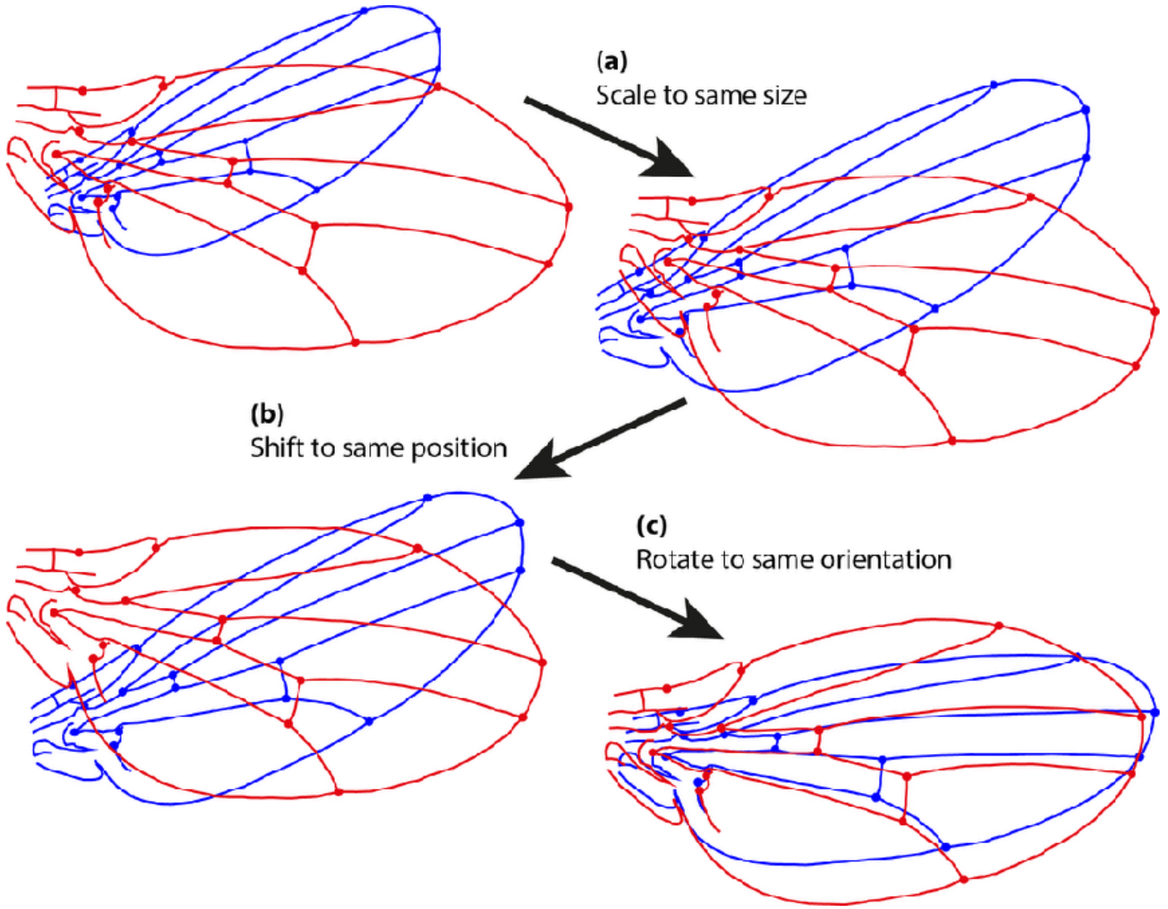
Being less wrong with DR

Procrustes alignment

- Named after ancient Greek bandit who stretched or chopped victims to fit in his bed
- Translate so the means of the two things are the same
- Scale so the standard deviations are the same
- Rotate to minimize the SSD between datapoints

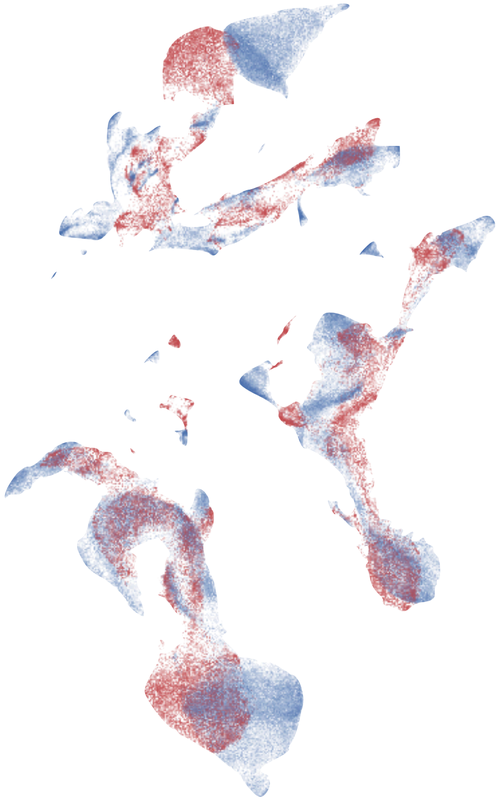


How repeatable are t-SNE/UMAP?

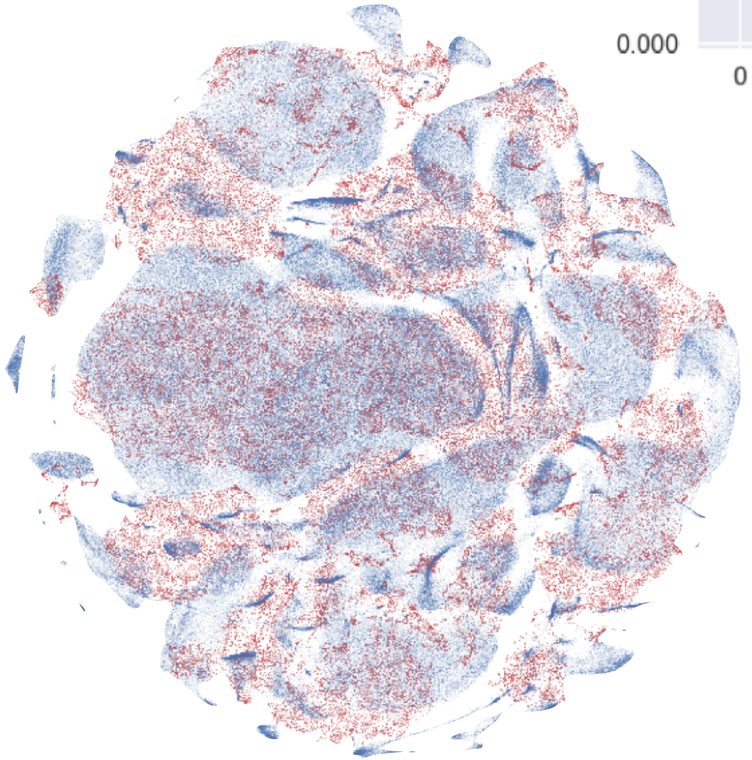
- Considering the normalized Procrustes distance between the embedding of a sub-sample, and the corresponding sub-sample of an embedding of the full dataset

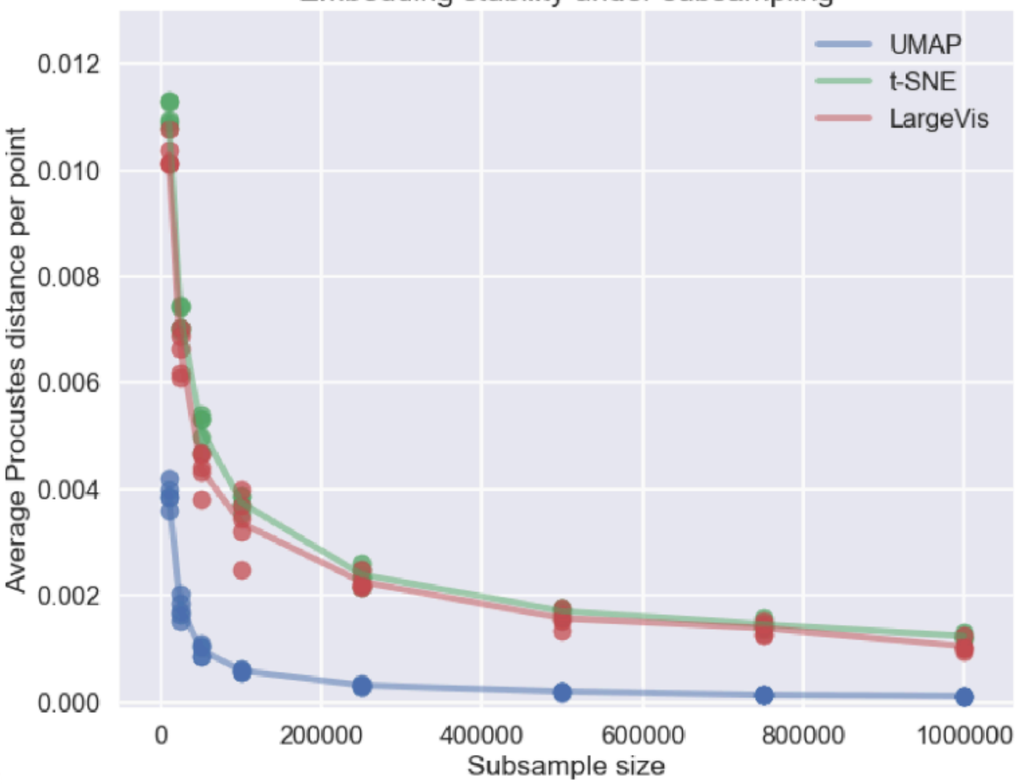
- As the size of the sub-sample increases the average distance per point between the sub-sampled embeddings should decrease, potentially toward some asymptote of maximal agreement under repeated runs. Ideally this asymptotic value would be zero error, but for stochastic embeddings such as UMAP and t-SNE this is not achievable.

UMAP



t-SNE




UMAP
t-SNE
LargeVis
Average Procustes distance per point
0.012
0.010
0.008
0.006
0.004
0.002
0.000
0
200000
400000
600000
800000
1000000
Subsample size

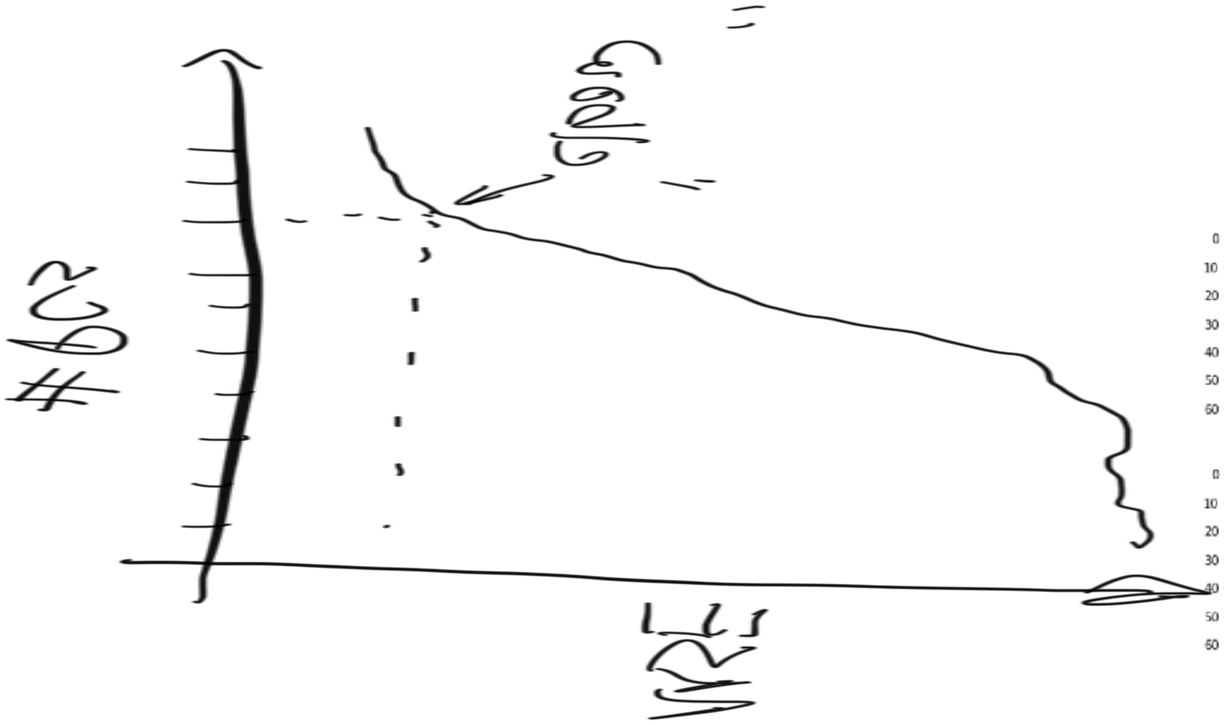
Reconstruction error

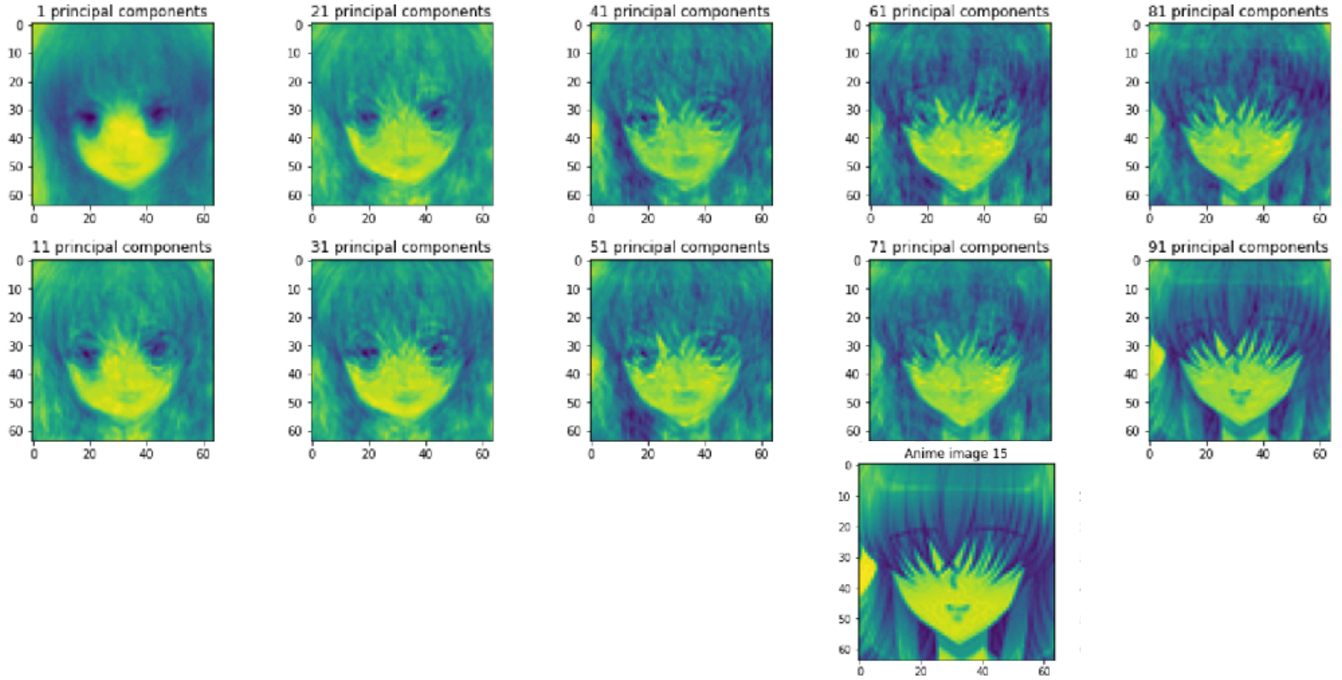
How can we select # of dimensions?

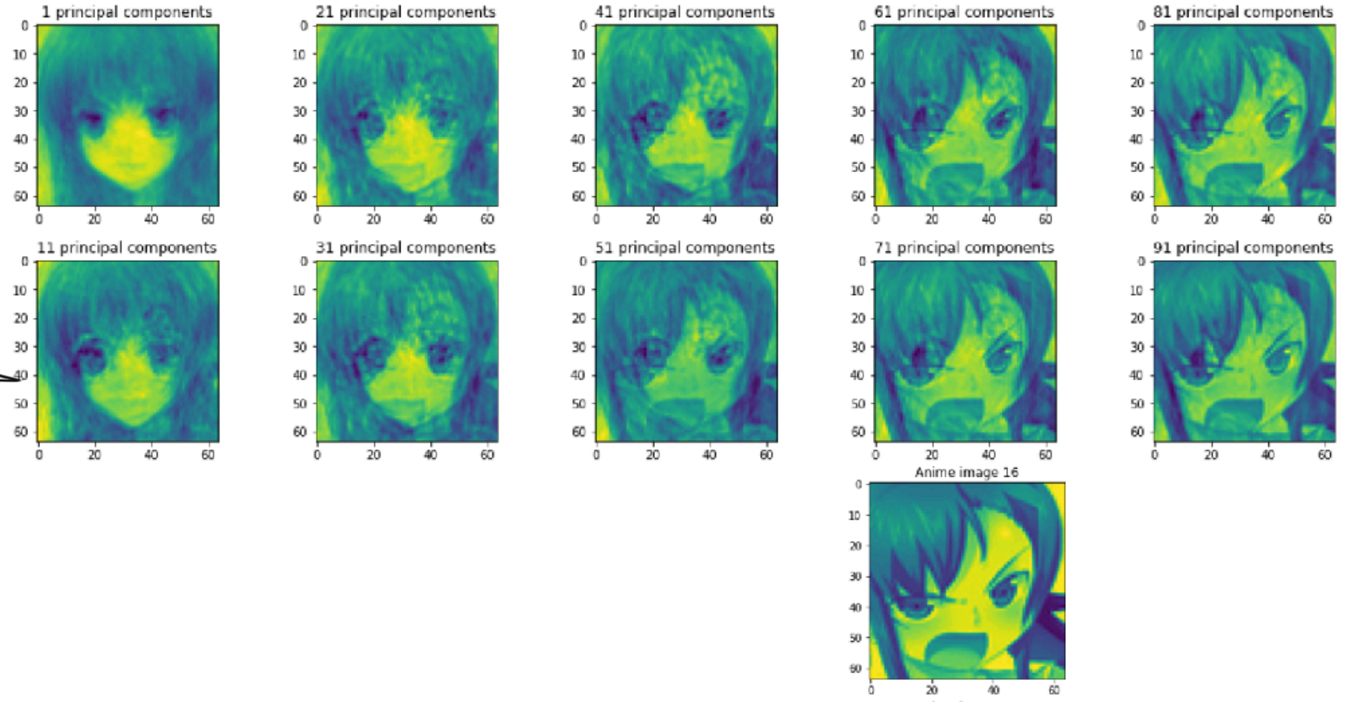
- If a dimension reduction transformation is invertible
  - Find components on entire dataset
  - For each sample in dataset, transform to lower d components
  - Inverse transform back into high d data space
  - Find error between original and reconstruction (usually MSE or RMSE)

Reconstruction error

To determine how many components

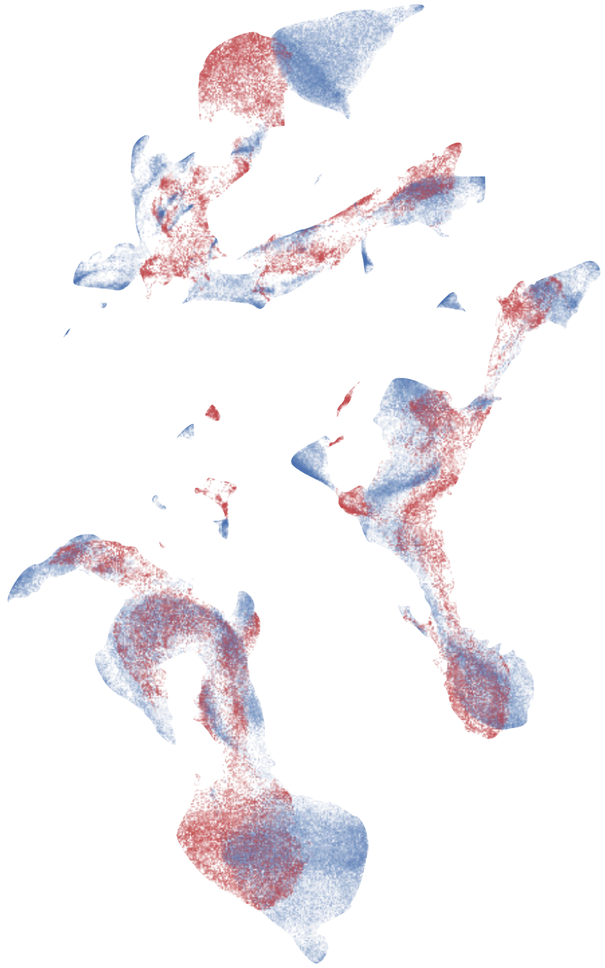






Outlier detection via reconstruction error

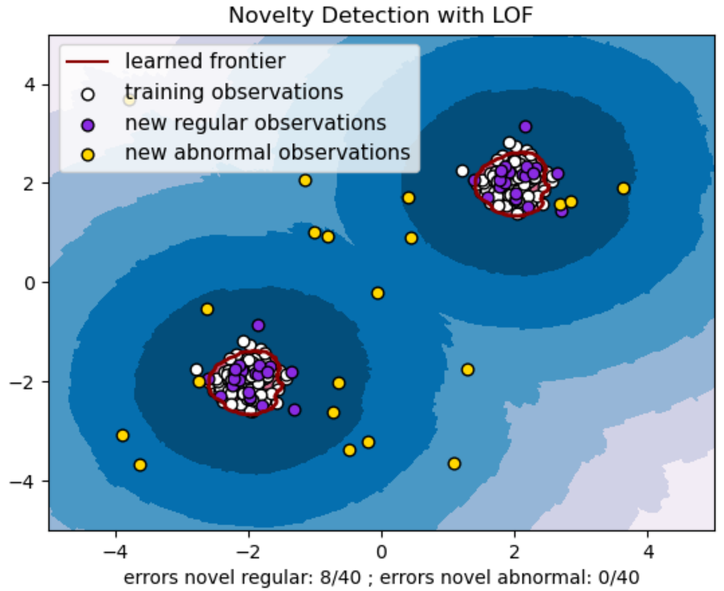
- Train DR on “normals”
- Transform “normal + outliers” to low-d
- Calc reconstruction error for each sample.
- Often outliers will have worse reconstruction error, it can even be so blatant you can just pick a threshold or train a logistic regression on the reconstruction errors

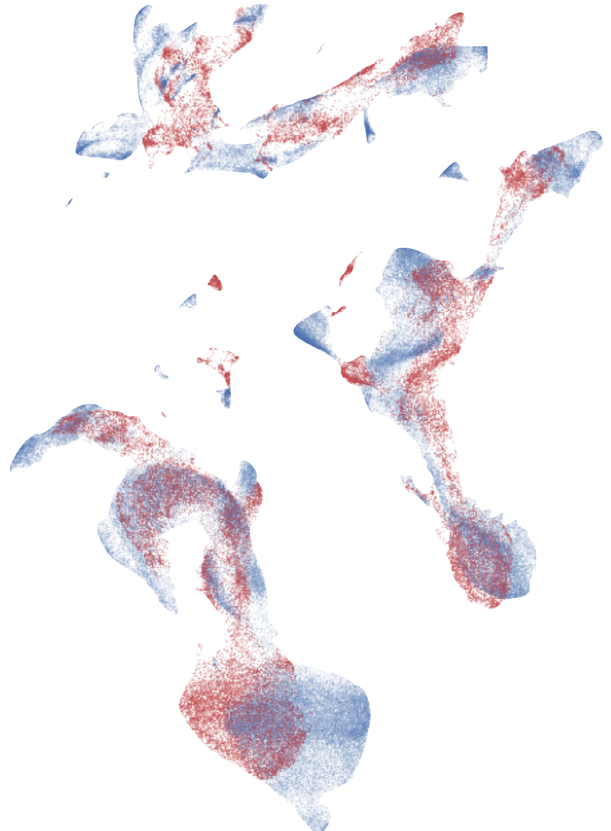


Outlier detection via LOF

Local Outlier Factor

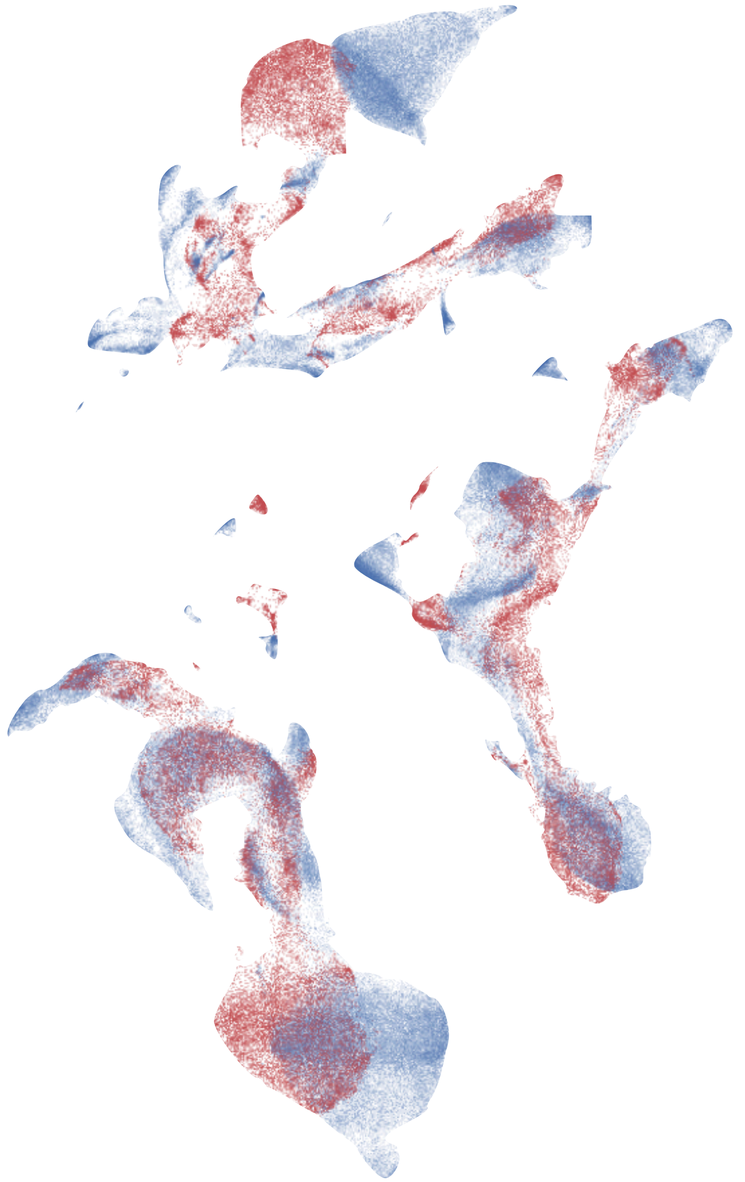
- LOF looks for patches of lower than normal density… any data points in low density regions are outliers
- LOF can operate in any vector space… original data or lower d after DR
- A good DR produces clustering naturally!





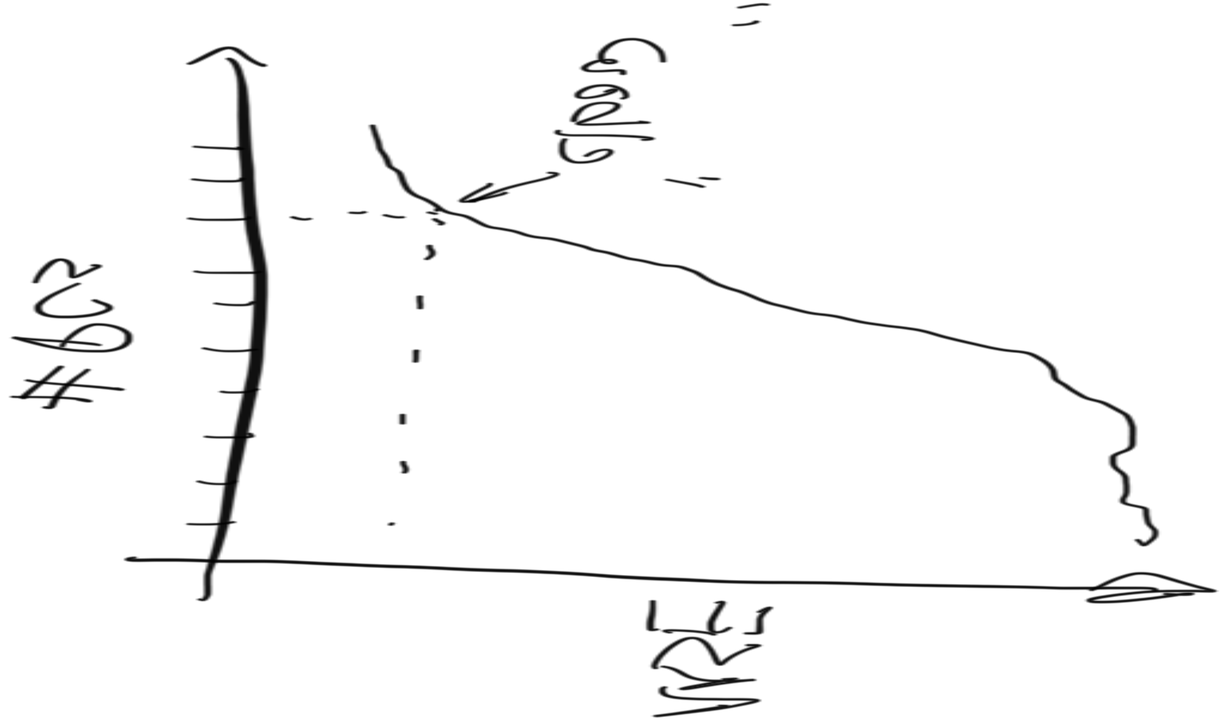
DR as features for supervised ML

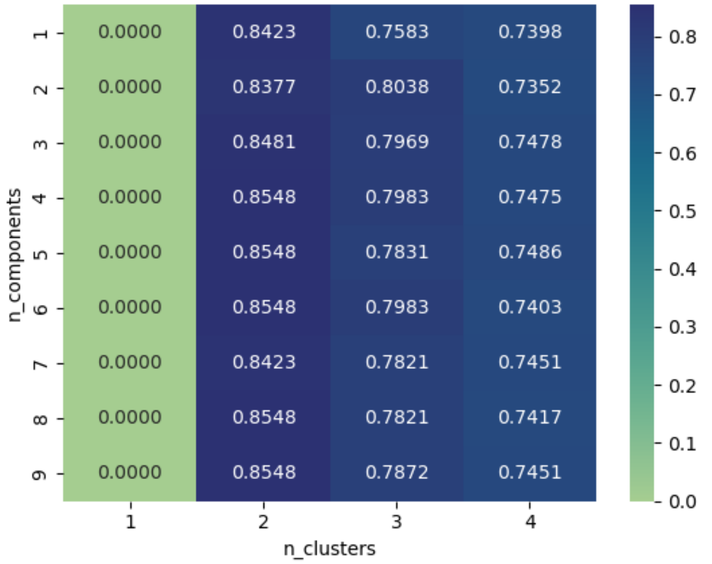
- Recall Procrustes alignment among different subsamples of data
- What does that imply for out of sample performance? (e.g. during cross validation or in deployment)
- Also why you must have DR inside your pipeline during CV or you will leak info and overfit



DR as features for supervised ML

- Use elbow then CV for supervised
- OR use CV for both at once





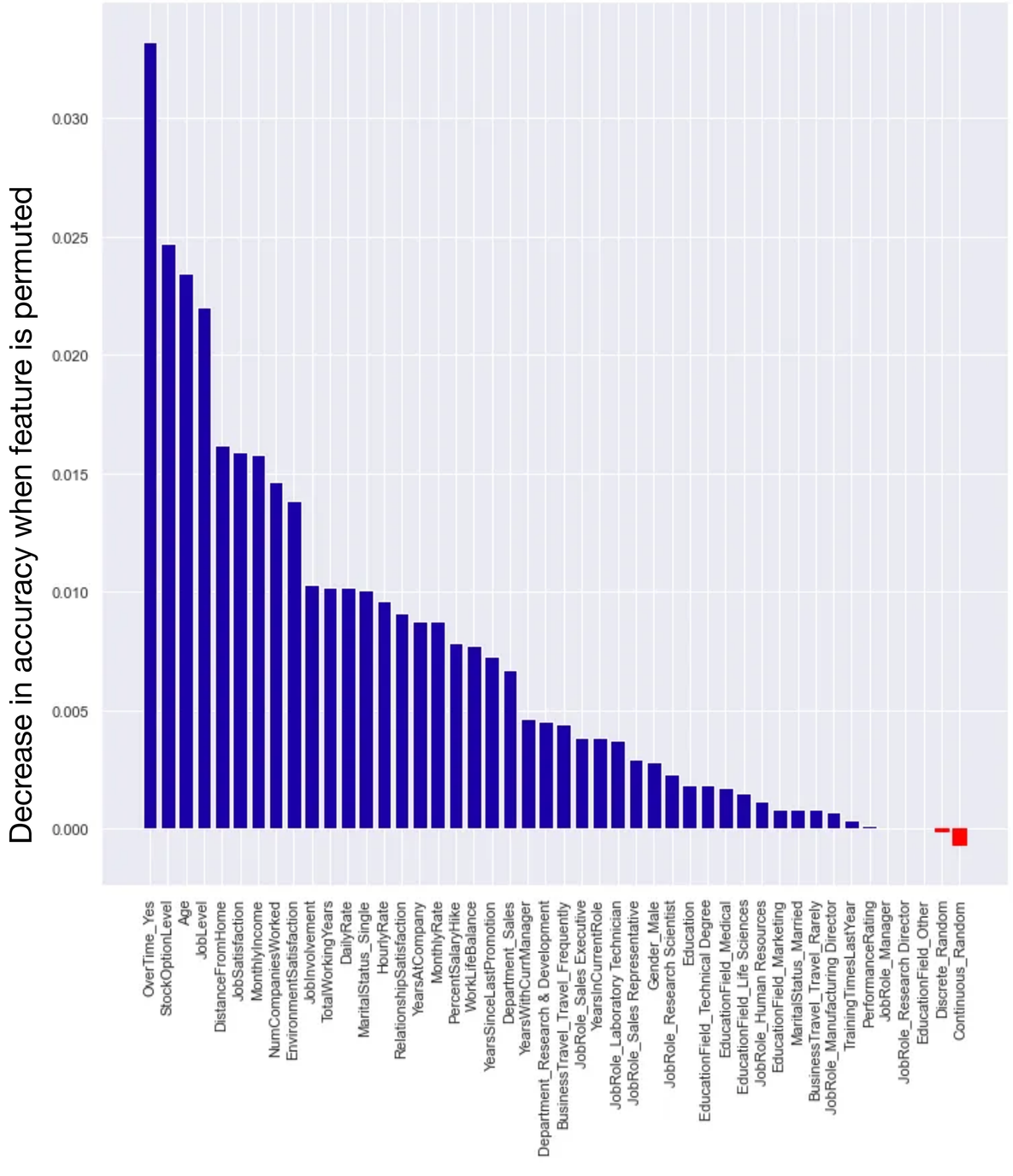
DR is NOT feature selection!! So what is :)

Feature selection for supervised ML

Permutation importance
(Not the only way to do this)

- Inputs: fitted predictive model m, tabular dataset (training or validation) D.
- Compute the reference score s of the model m on data D (for instance the accuracy for a classifier or the R^2 for a regressor).
- For each feature j (column of D):
    - For each repetition k in $1, \ldots, K$:
        - Randomly shuffle column j of dataset D to generate a corrupted version of the data named $\tilde{D}_{k,j}$.
        - Compute the score $s_{k,j}$ of model m on corrupted data $\tilde{D}_{k,j}$.
    - Compute importance i_j for feature f_j defined as:

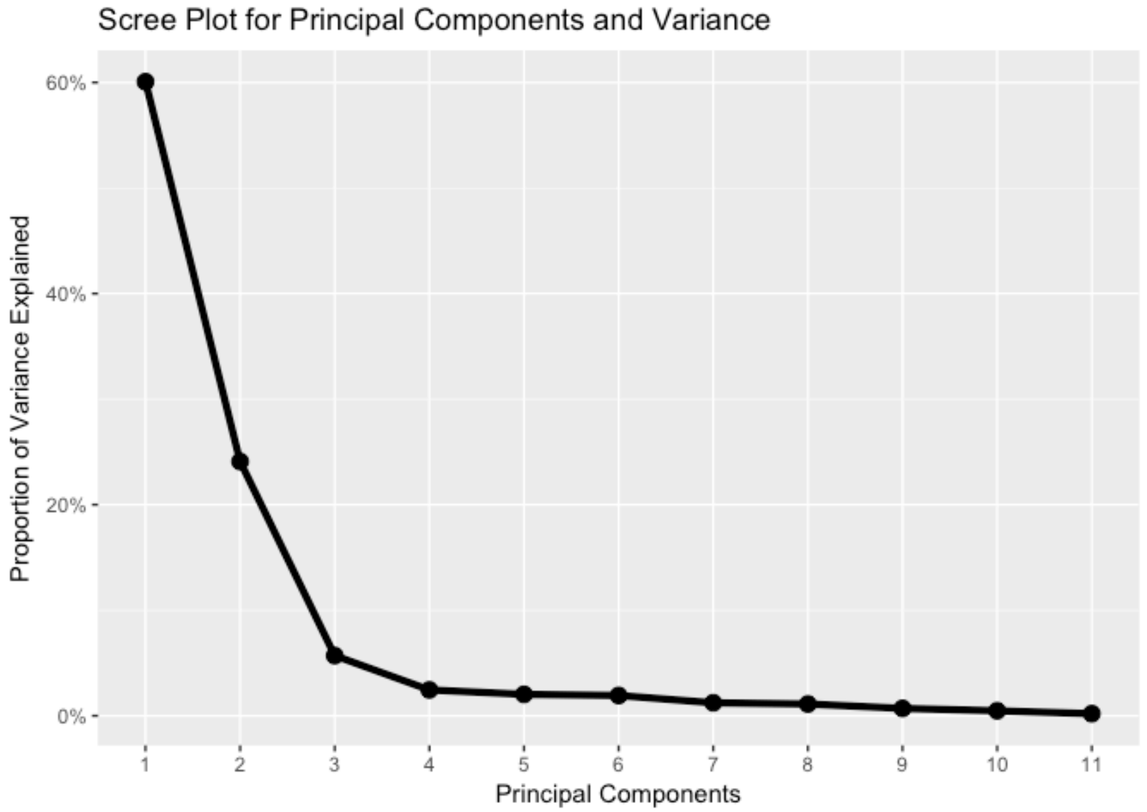
$$i_j = s - \frac{1}{K} \sum_{k=1}^{K} s_{k,j}$$



DR Summary

PCA

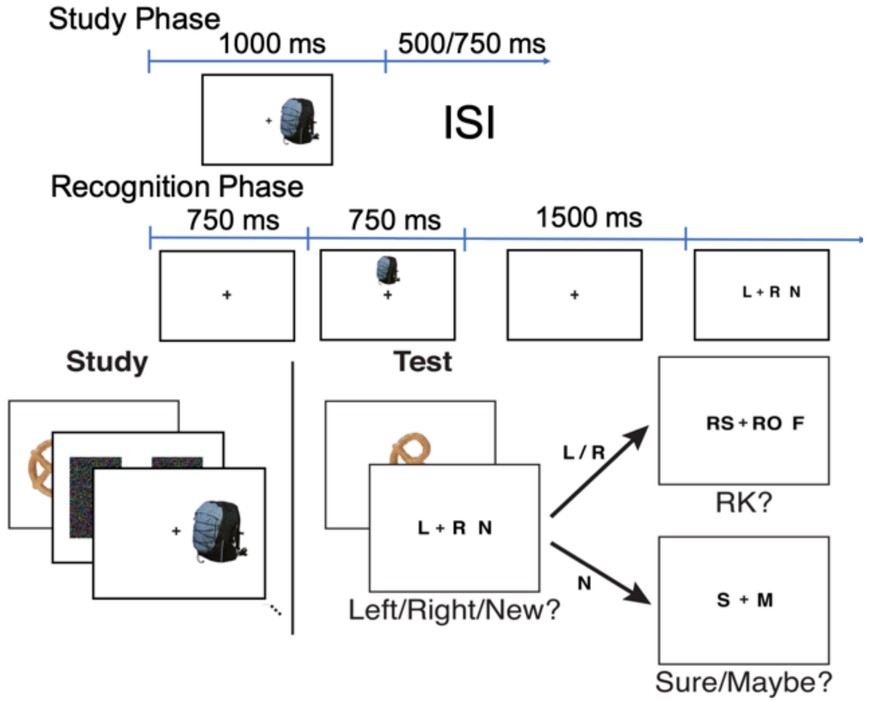
- Motivated by variance
- Linear transformation, analytic solution, no hyper parameters
- Can be used to transform new data. Can be inverted back to original data space
- In-sample ways to select number of dimensions
  - % var explained to threshold
  - % var explained elbow
- Cross validation out of sample
  - against reconstruction error
  - against a supervised task on top

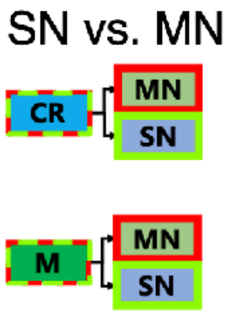


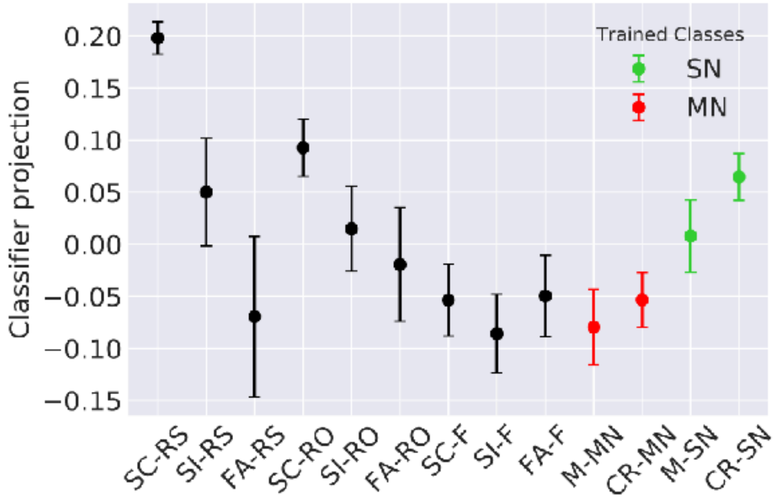
LDA

To do

- Motivated by supervised classification
- Linear transformation, analytic solution, no hyper parameters
- Can be used to transform new data. Can be inverted back to original data space
- In-sample ways to select number of dimensions
  - % var explained to threshold
  - % var explained elbow
- Cross validation out of sample
  - against reconstruction error
  - against a supervised task on top







t-SNE / UMAP

- Motivated by preserving local distances inside a neighborhood
- Non-linear transformation in the sense that neighborhood sizes change depending on the density of the data in the local region
- Hyper-parameter of the number of neighbors to consider. UMAP also has a hyper-parameter related to manifold density.
- May be limited to 2 or 3 dim depending on solver
- In vanilla forms there is no way to transform new data and there is no way to invert back to the original data space.
- Parametric versions allow transform of new data and inversion… think of these as a different solver where the solution is approximate
- No way to select number of dimensions in-sample for vanilla, can do for parametric but errors are approximate ¯_(ツ)_/¯
- Cross validation out of sample to select number of dimensions against reconstruction error or a supervised task on top of DR