

# Dimensionality Reduction

**CS/DSA 5970: Machine Learning Practice**

# Challenges

Most data that we wish to analyze live in high-dimensional spaces

- Potentially need *really* large data sets to achieve a reasonable representation of the sample distribution
- Our intuition can go out the door quickly
- Some of our math breaks
- Computational tools may not scale to high dimensions well

# Challenges

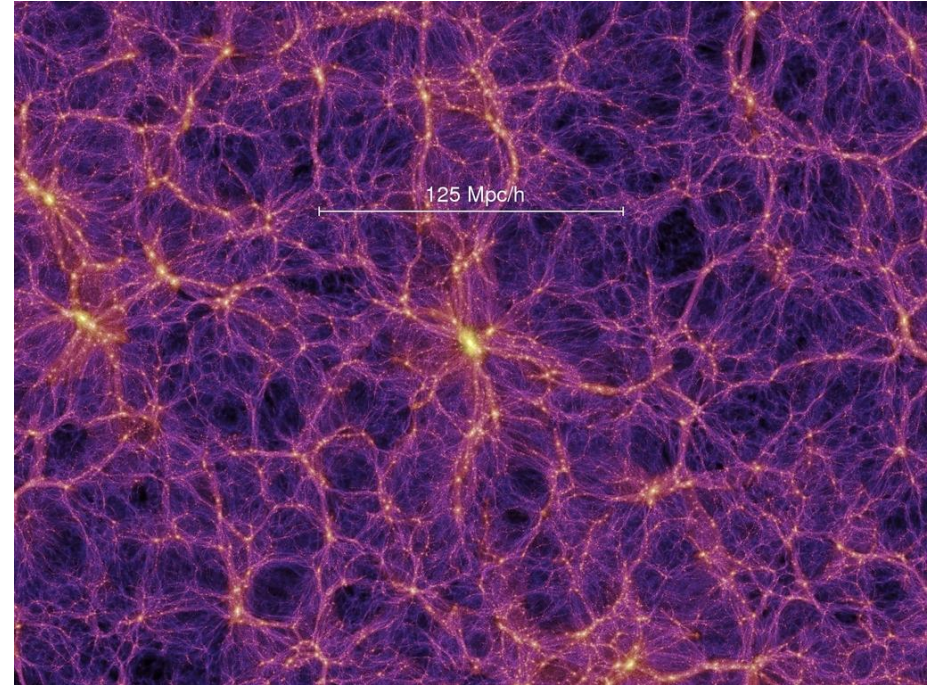
Random points selected uniformly from a unit N-cube:

- Distances become really large
- Distribution of distances becomes very narrow
  - By  $N=30$ , all uniformly selected point pairs have very similar distances
  - This suggests that the Euclidean distance metric may not have much meaning

# Sample Distribution

For many data sets, samples are not drawn uniformly from the feature space

- 0 D: clusters
- 1 D: line segments / curves
- 2 D: planes / surfaces
- :



[www.mpa-garching.mpg.de](http://www.mpa-garching.mpg.de)

Use the term *manifold* to describe a group of samples that locally vary in some dimensions, but not in others

# Dimensionality Reduction

- Goal:
  - Given a set of samples from some  $N$ -dimensional feature space
  - Re-encode the samples into a smaller  $M$ -dimensional space
- Challenge:
  - No labels

# Projection Approaches

Projection: linear transformation of a point in  $N$  space onto a nearby  $M$ -dimensional manifold (where  $M \ll N$ )

- Projection into linear subspaces
- Warping space, followed by projection

# Embedding Approaches

- Identify samples that are “near” one-another in the N-dimensional feature space
- Find a way to embed corresponding points into an M-dimensional space that respects this “nearness”
- Again:  $M \ll N$

# Benefits of Reducing the Dimensionality of a Feature Set

- Make explicit the primary variance in the samples
  - While: removing only small variance
- Through visualization of the reduced-dimensionality data:
  - Possible to reclaim some of our intuition about the data
  - Or even discover new, interesting relationships



# Benefits of Reducing the Dimensionality of a Feature Set

Can use as a means of preprocessing our data before applying other learning techniques. Smaller dimensionality implies:

- Subsequent models have fewer parameters
- Reduced potential for overfitting
- Training times can be much faster

,

# Principal Component Analysis

**CS/DSA 5970: Machine Learning Practice**

# Principal Component Analysis

Incremental process:

- Identify the one axis in a feature space along which we have the highest variance
- Subtract all variance along this axis
- Repeat

,

# Example: Principal Component Analysis

**CS/DSA 5970: Machine Learning Practice**



# **Example: PCA with Kinematics**

**CS/DSA 5970: Machine Learning Practice**





# **Kernel PCA and Kinematics**

**CS/DSA 5970: Machine Learning Practice**

# Kernel PCA

- PCA involves only linear transformations
  - This could be a problem for feature spaces that contain non-linear manifolds
- As with linear regression and SVMs:
  - We can add a set of non-linear transformations on the features
  - Then, we can perform PCA on the expanded feature vectors
  - The Kernel Trick works here, too!



