SMAI-S25-L13: LowRank, and Applications

C. V. Jawahar

IIIT Hyderabad

February 21, 2025

Recap:

- Problems of interest:
  - Learn a function $y = f(\mathbf{W}, \mathbf{x})$ from the data.
    - (a) Classification (b) Regression
  - Learn Feature Transformations $\mathbf{x}' = \mathbf{W}\mathbf{x}$ or $\mathbf{x}' = f(\mathbf{W}, \mathbf{x})$
    - (a) Feature Normalization (b) PCA
- Algorithms/Approaches:
  - Nearest Neighbour Algorithm
  - Linear Classification: $sign(\mathbf{w}^T\mathbf{x})$
  - Decide as ω_1 if $P(\omega_1|\mathbf{x}) \geq P(\omega_2|\mathbf{x})$ else ω_2.
  - Linear Regression: (a) closed form and (b) GD
- Supervised Learning:
  - Notion of Training, Validation and Testing
  - Performance Metrics
  - Low rank data matrix and SVD
  - Notion of Loss Function, (eg. MSE), Regularization.
  - Role of Optimization, Convex and non-Convex optimization
  - Closed form solution, Gradient Descent, Eigen vector solns.

SVD

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathbf{T}}$$

$$\text{with } U^T U = I; V^T V = I; \text{ and D diagonal}$$

A is $m \times n$; U is $m \times m$; D is $m \times n$ and V is $n \times n$
Read this article[1]

- Now (10 mins) and discuss/explain/summarize
- After the class more carefully
- and ofcourse before the exam

[1] `https://jonathan-hui.medium.com/machine-learning-singular-value-decomposition-svd-principal-component-anal`

Dimensionality Reduction

Reduce dimensionality

- Linear:$\mathbf{x}' = \mathbf{W}\mathbf{x}$ (eg. PCA)
- Nonlinear $\mathbf{x}' = f(\mathbf{W}, \mathbf{x})$ (eg. Autoencoder neural networks)
- Supervised (LDA, Fisher)
- Unsupervised (PCA, autoencoder)
- Dimensionality Reduction on a Manifold (LLE, ISOMAP) (out of scope)

Why Dimensionality Reduction? Many Reasons ..

- Data compression
- Less parameters to learn (less data needed)
- Less overfittiong (More reliable models/solutions)
- Faster computation
- Better visualization

Why Dimensionality Reduction? Separability

Fisher Linear Discriminant or LDA Goal (two classes):

- Classes are well separated (means are far) (Maximimize S_B)
- Each class is compact (variance is small) (Minimize S_W)

$$S_B = [\mu_1 - \mu_2][\mu_1 - \mu_2]^T$$

$$S_W = S_1 + S_2 = \sum_{\mathbf{x_i} \in \omega_1} [\mathbf{x_i} - \mu_1][\mathbf{x_i} - \mu_1]^T + \sum_{\mathbf{x_i} \in \omega_2} [\mathbf{x_i} - \mu_2][\mathbf{x_i} - \mu_2]^T$$

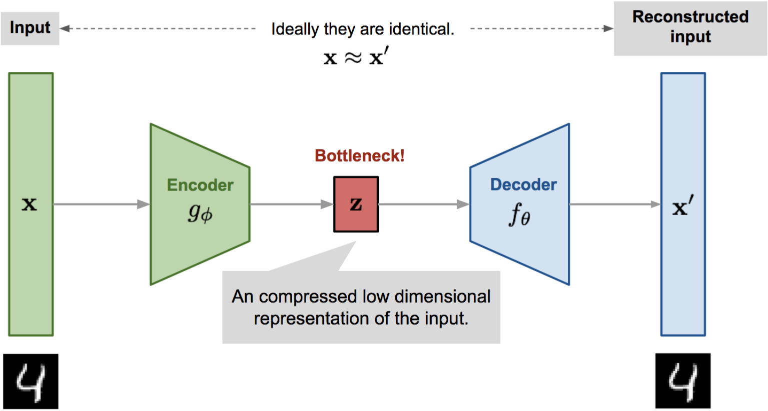
If we reduction $z = \mathbf{u}^T \mathbf{x}$ our goal is to maximize: $J(\mathbf{u}) = \frac{\mathbf{u}^T \mathbf{S_B} \mathbf{u}}{\mathbf{u}^T \mathbf{S_W} \mathbf{u}}$

$$\max_{\mathbf{u}} \frac{1}{2} \mathbf{u}^T \mathbf{S_B} \mathbf{u} - \frac{\lambda}{2} (\mathbf{u}^T \mathbf{S_W} \mathbf{u} - 1)$$

Leads to a genarlized eigen value problem $\mathbf{S_B} \mathbf{u} = \lambda \mathbf{S_W} \mathbf{u}$ with solution as

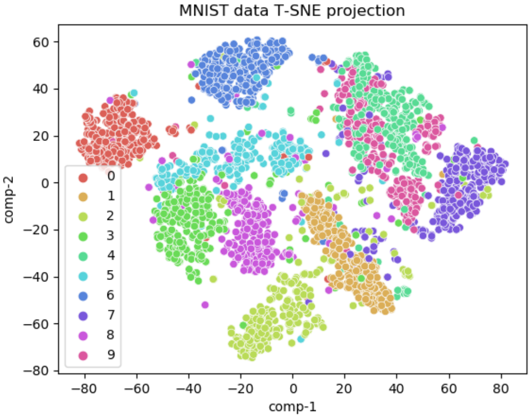
$$\mathbf{u} = \alpha \mathbf{S_W}^{-1} [\mu_1 - \mu_2]$$

Non-Linear Dimensionality Reduction: Auto Encoder



An excellent example of unsupervised learning

Why Dimensionality Reduction? Visualization: tsne[2]



[2] https://en.wikipedia.org/wiki/T-distributed_stochastic_neighbor_embedding

Why Dimensionality Reduction? TSNE

t-distributed stochastic neighbor embedding

Characteristic	t-SNE	PCA
Type	Non-linear dimensionality reduction	Linear dimensionality reduction
Goal	Preserve local pairwise similarities	Preserve global variance
Best used for	Visualizing complex, high-dimensional data	Data with linear structure
Output	Low-dimensional representation	Principal components
Use cases	Clustering, anomaly detection, NLP	Noise reduction, feature extraction
Computational intensity	High	Low
Interpretation	Harder to interpret	Easier to interpret

Matrix Problems

- Matrix Factorization/Decomposition
- Non-Negative Matrix factorization
- Matrix Completion
- Low rank Approximation

Applications

- LSI Watch (10 mins):
  - https://www.youtube.com/watch?v=rMaDnRLZAzM
- Recommendation Systems Watch (10 mins):
  - https://www.youtube.com/watch?v=tP9e0PlmMKo