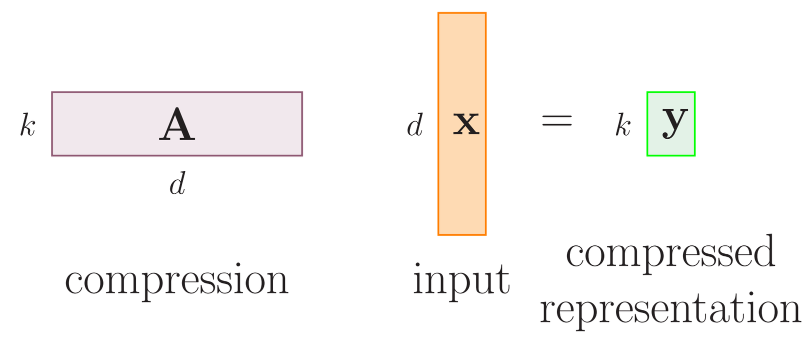
Randomized Dimensionality Reduction in Machine Learning

Sebastian Mayer
Institute for Numerical Simulation, University of Bonn

Introduction

Typical situation in machine learning:

- Feature space $\mathcal{X} \subseteq \mathbb{R}^d$ with d **very large** (e.g. $d = 40k$):
  - text represented as *bag-of-word* (BOW),
  - image represented as *histogram of gradients* (HOG),
  - property list in *computational biology*.
- **Subproblem:** computing distances or dot products
- Working on $\mathcal{X}$ difficult $\Longrightarrow$ **data compression** $y = Ax$.



Here:

Compression = randomized dimensionality reduction

Motivations for (randomized) dimensionality reduction:

- computation speed-up
- memory or storage constraints
- trick in algorithm design
- better compression guarantees

Can we compress while preserving important properties?

Johnson-Lindenstrauss embeddings (JLE): Basic version [JL '84; Dasgupta, Gupta '99]

Observation:

- Let $\mathcal{A}$ be a $(k \times d)$-**random matrix** with *i.i.d.* entries $\mathcal{A}_{ij} \sim \mathcal{N}(0, 1/k)$.
- For fixed $x \in \mathbb{R}^d$, we have $\mathbb{E}_{\mathcal{A}} \|\mathcal{A}x\|_2^2 = \|x\|_2^2$.

Given:

- **finite $S \subset \mathbb{R}^d$,**
- **Accuracy parameter $\varepsilon \in (0, 1)$,**
- Failure probability $\delta \in (0, 1)$,

Theorem: If

$$k = \mathcal{O}\big(\varepsilon^{-2} \ln(|S|/\delta)\big),$$

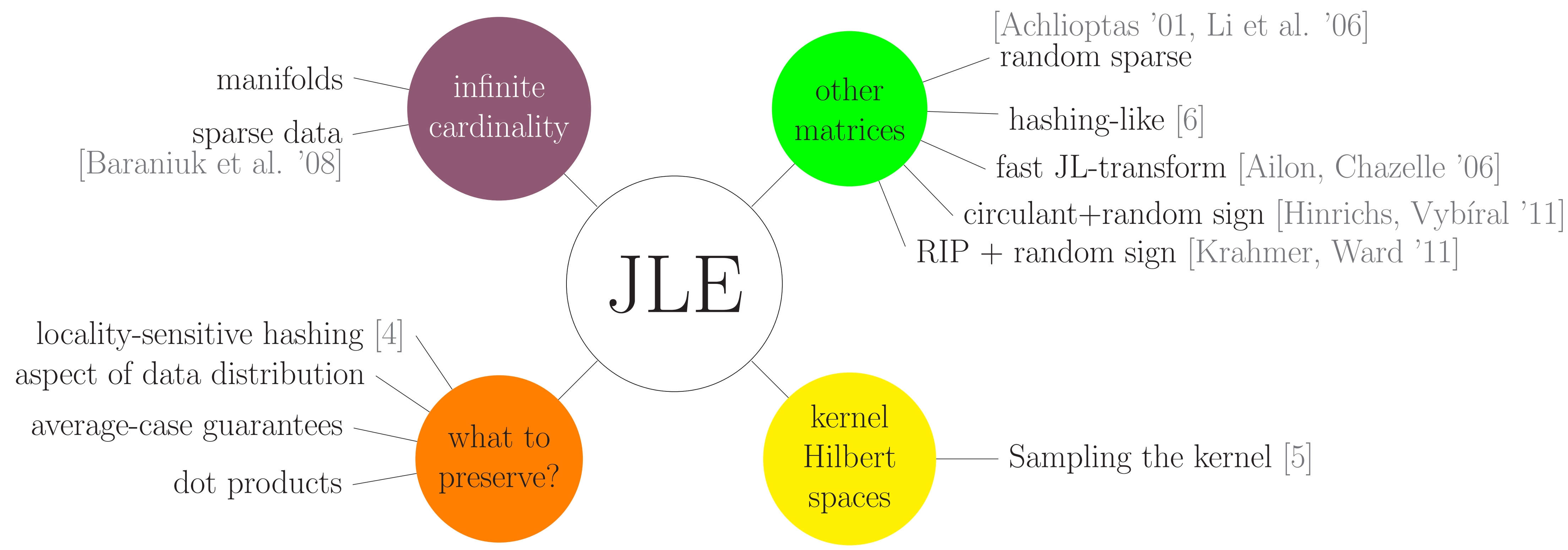
then a realization $A = \mathrm{draw}(\mathcal{A})$ is with probability $1 - \delta$ an ε-**isometry** on S:

$$(1-\varepsilon)\|x-y\|_2^2 \le \|A(x-y)\|_2^2 \le (1+\varepsilon)\|x-y\|_2^2,$$

for all $x, y \in S$.

Preservation of *relative* distances!

JLE: Extensions and Generalizations



Example 1: Nearest neighbour problems [e.g. Indyk '01]

Given: *finite* set of n points $S \subset \mathbb{R}^d$.

r-similarity graph: Compute $G = (S, E)$ such that for $p, q \in S$,

$$(p,q) \in E \iff d(p,q) = \|p-q\|_2 \le r.$$

Computation time: $\mathcal{O}(dn^2)$

Can we speed up computations?

Relax problem:

(r, R, δ)-similarity graph: Compute $G' = (S, E')$ such that with probability $1 - \delta$,

$$\begin{aligned} (p,q) \in E' &\implies d(p,q) \le r \\ (p,q) \notin E' &\implies d(p,q) \ge R. \end{aligned}$$

Computation time: *JLE with sparse matrix:* $C = \left(\frac{R^2+r^2}{R^2-r^2}\right)^2$

$$\underbrace{\mathcal{O}\left(C \log n\, n^2\right)}_{\text{dist. comp.}} + \underbrace{\mathcal{O}\left(C \sqrt{d}\, n \log n\right)}_{\text{preprocessing}}$$

Example 2: Learning mixtures of Gaussians [Dasgupta '99]

Given: n independent, $\mathbb{R}^d$-valued Gaussians $\mathcal{N}(\mu_1, \Sigma), \ldots, \mathcal{N}(\mu_n, \Sigma)$.

Goal: Estimate $\mu_1, \ldots, \mu_n$ from observations $X_1, \ldots, X_m$ originating from *any* Gaussian with equal probability.

- **γ-separated Gaussians:**
  $$\|\mu_i - \mu_j\|_2 \ge \gamma \sqrt{d\, \lambda_{\max}(\Sigma)}.$$
- **Eccentricity:**
  $$\mathrm{ecc}(\Sigma) = \sqrt{\lambda_{\max}(\Sigma)/\lambda_{\min}(\Sigma)}.$$

If the Gaussians are sufficiently separated and have low eccentricity, then there is a relatively simple algorithm which needs $\mathcal{O}(\rho^{-d})$ observations to estimate with accuracy $\rho\sqrt{d\, \lambda_{max}(\Sigma)}$.

Can we overcome the curse of dimensionality and even handle higher eccentricity?

$$A = \mathrm{draw}(\mathcal{A})$$
$$\Longrightarrow$$
auxiliary Gaussians
$$\tilde{\mu}_i = A\mu_i,\ \tilde{\Sigma} = A\Sigma A^T$$

Observation: If $X \sim \mathcal{N}(\mu, \Sigma)$, then

$$\mathbb{E}_{\mathcal{A}} \|\mathbb{E}_X \mathcal{A}X\|_2^2 = \|\mu\|_2^2$$

$$\mathbb{E}_{\mathcal{A}} \mathrm{Cov}(\mathcal{A}X) = \frac{\mathrm{trace}(\Sigma)}{k} \mathrm{Id}_k\,.$$

Theorem: If

$$k = \mathcal{O}\big(\varepsilon^{-2} \log(2n/\delta)\big)$$

and

$$\mathrm{ecc}(\Sigma) = \mathcal{O}\left(\frac{\tau\sqrt{d}}{\log(k/(\delta\tau))}\right),$$

then with probability $1 - \delta$

- **Preservation of separability:**
  $$\|\tilde{\mu}_i - \tilde{\mu}_j\|_2^2 \ge (1-\varepsilon)(1-\tau)\gamma^2 k \lambda_{\max}(\tilde{\Sigma}),$$
- **Reduction of eccentricity:**
  $$(1-\tau)\lambda \le |v^T \tilde{\Sigma} v| \le (1+\tau)\lambda$$
  for all $v \in \mathbb{S}_2^{k-1}$ and $\lambda = \mathrm{trace}(\Sigma)/k$.

References

[1] Arriaga, Vempala (2006): *An algorithmic theory of learning: Robust concepts and random projection.*

[2] Balcan, Blum, Vempala (2006): *Kernels as features: On kernels, margins, and low-dimensional mappings.*

[3] Calderbank, Jafarpour (2012): *Finding needles in compressed haystacks.*

[4] Gionis, Indyk, Motwani (1999): *Similarity search via hashing in high dimensions.*

[5] Rahimi, Recht (2007): *Random features for large-scale kernel machines.*

[6] Weinberger, Dasgupta, Langford, Smola, Attenberg (2009): *Feature hashing for large scale multitask learning.*

Example 3: Learning a linear classifier

Related work: [1, 2, 3]

Given: Classification problem $(\mathcal{D}, \ell)$, where

- (unknown) distribution $\mathcal{D}$ over $\mathcal{X} \subseteq \{x \in \mathbb{R}^d : \|x\|_2 \le R\}$,
- labelling function $\ell : \mathcal{X} \to \{-1, 1\}$.

Properties:

- $(\mathcal{D}, \ell)$ **linearly separable**: there is $w \in \mathbb{R}^d$
  $$\mathcal{R}_{0,\mathcal{D}}(w) := \mathbb{P}_{X\sim\mathcal{D}}(\ell(X)\langle w, X\rangle \ge 0) = 1.$$
- $(\mathcal{D}, \ell)$ **linearly separable at margin γ**:
  $$(\exists w \in \mathbb{S}_2^{d-1}) \quad \mathbb{P}_{X\sim\mathcal{D}}(\ell(X)\langle w, X\rangle \ge \gamma) = 1.$$

Hinge risk:

- $\mathcal{R}_{\gamma,\mathcal{D}}(w) = \mathbb{E}_{X\sim\mathcal{D}}\left(1 - \frac{\ell(X)\langle w, X\rangle}{\gamma}\right)_+$
- convex, $1/\gamma$-Lipschitz
- *Surrogate* for misclassification rate: $\mathcal{R}_{0,\mathcal{D}}(w) \le \mathcal{R}_{\gamma,\mathcal{D}}(w)$.

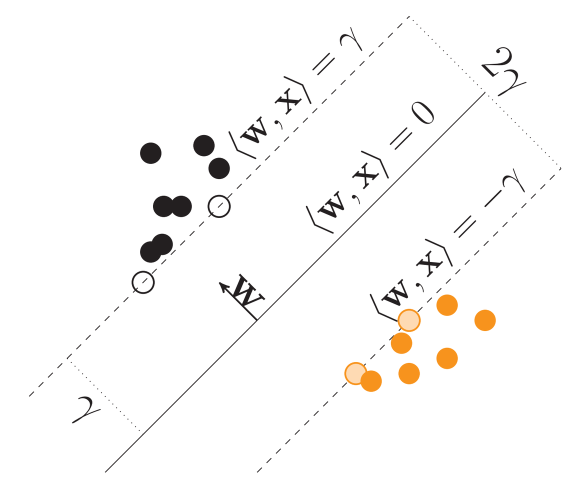
Learning task: Draw *i.i.d.* samples $X_1, \ldots, X_m \sim \mathcal{D}$, learn $\hat{w} \in \mathbb{R}^d$ with minimal **misclassification rate**:

$$\hat{w} = \mathrm{argmin}_{w \in \mathbb{R}^d} \mathcal{R}_{0,\mathcal{D}}(w).$$

- Required amount of samples m increases with decreasing γ.

Can we learn and predict on compressed data?

$$A = \mathrm{draw}(\mathcal{A})$$
$$\Longrightarrow$$
compressed classification problem $(A\mathcal{D}, \ell)$



Learning a linear classifier: Some results [M. '13]

Theorem (Preservation of linear separability):
Fix $w \in \mathbb{R}^d$ with $\|w\|_2 \le R$. If

$$k = \mathcal{O}\left(\left(\frac{1+R^2}{\varepsilon\gamma}\right)^2 \log(1/\delta)\right),$$

then a realization $A = \mathrm{draw}(\mathcal{A})$ fulfils

$$|\mathcal{R}_{\gamma,A\mathcal{D}}(Aw) - \mathcal{R}_{\gamma,\mathcal{D}}(w)| \le \varepsilon$$

with probability $1 - \delta$.

Theorem (Preservation of sample complexity):
Let $\mathcal{X} \cong \mathbb{R}_s^d$ and $w \in \mathbb{S}_2^{d-1}$. If

$$k = \mathcal{O}\left(\left(\frac{1+R^2}{\tau\gamma}\right)^2 s \log(d/s) + \log(1/\delta)\right),$$

then a realization $A = \mathrm{draw}(\mathcal{A})$ fulfils

$$\mathbb{P}_{X\sim\mathcal{D}}(\ell(X)\langle Aw, AX\rangle \ge (1-\tau)\gamma) = 1$$

with probability $1 - \delta$.

Numerical experiment: Support Vector Machines (SVM)

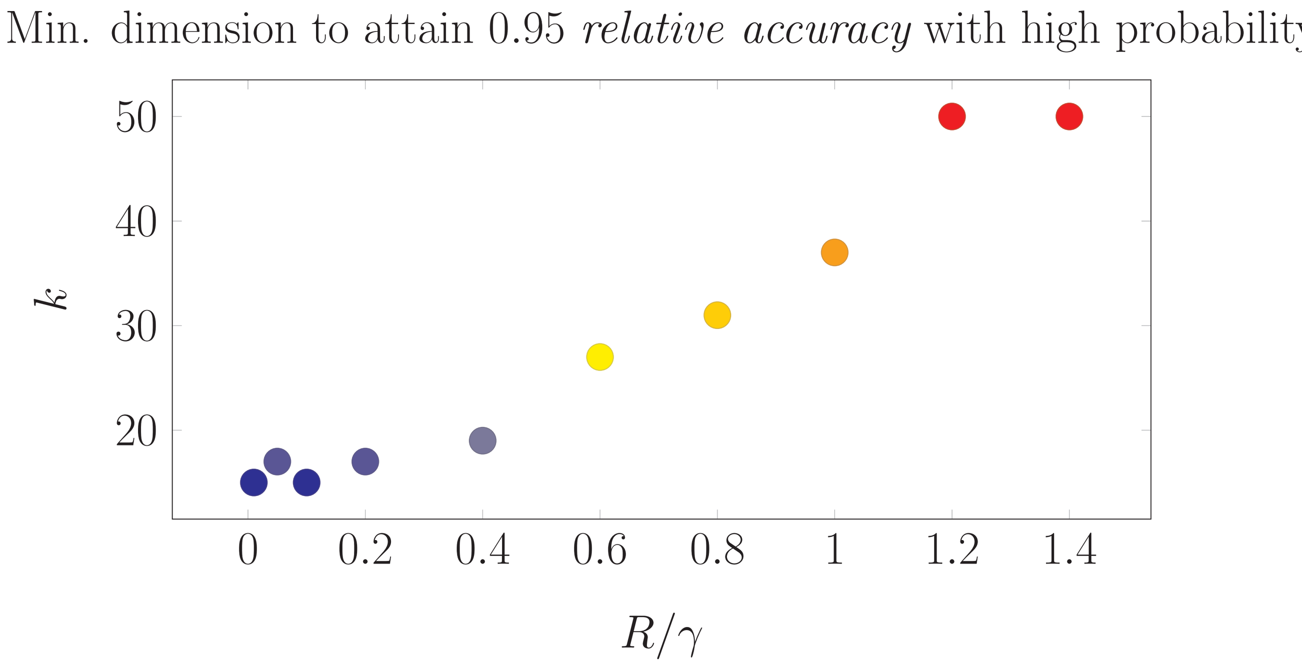
- Model: $\mathcal{D}$ uniform distribution over $\mathcal{C}_1 \cup \mathcal{C}_2$, where
  $$\mathcal{C}_1 := (R+\gamma)e_1 + [-R, R]^{50}, \quad \mathcal{C}_2 := -(R+\gamma)e_1 + [-R, R]^{50}.$$
- Training: fixed number of training samples; train both in ambient and projected space.
- Evaluation: Accuracy (=1-misclass. rate) in projected space relative to accuracy in ambient space.

- Interpretation:

  Good:
  - easy learning problems,
  - labelling cheap.

  Bad:
  - harder learning problems,
  - labelling expensive.

Min. dimension to attain 0.95 *relative accuracy* with high probability



Research Group Ullrich - Endenicher Allee 62, 53115 Bonn, Germany - mayer@ins.uni-bonn.de