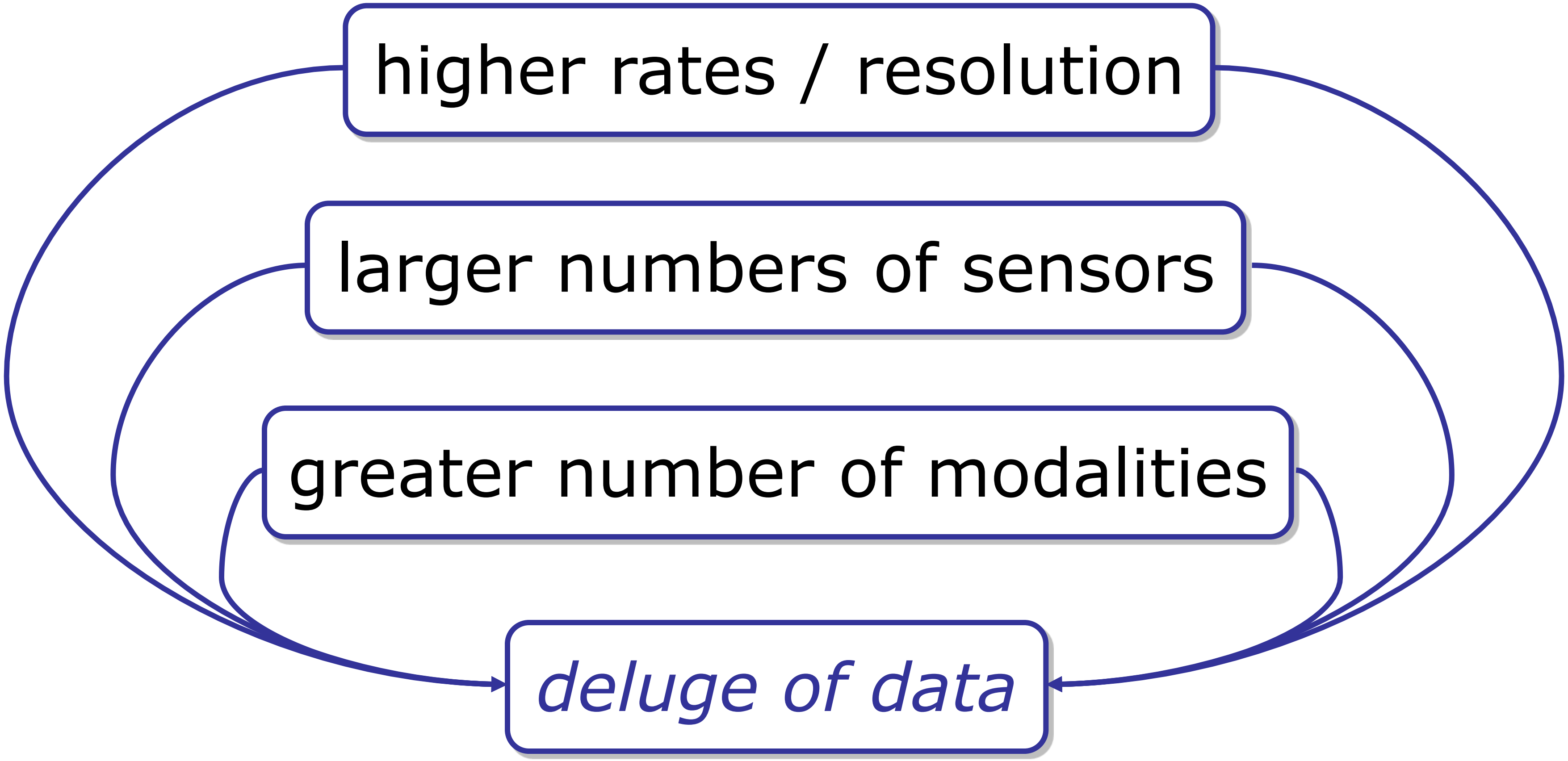
Efficient Machine Learning Using Random Projections

Chinmay Hegde
Mark Davenport
Richard Baraniuk

Michael Wakin

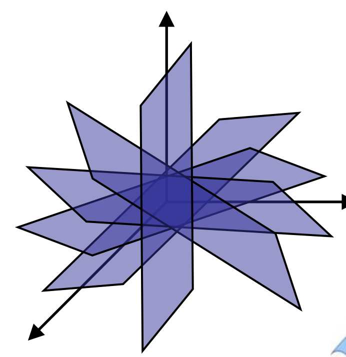
Pressure is on…

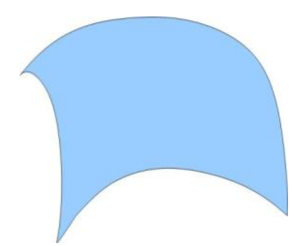
Increasing pressure on machine learning algorithms to support



Models and conciseness

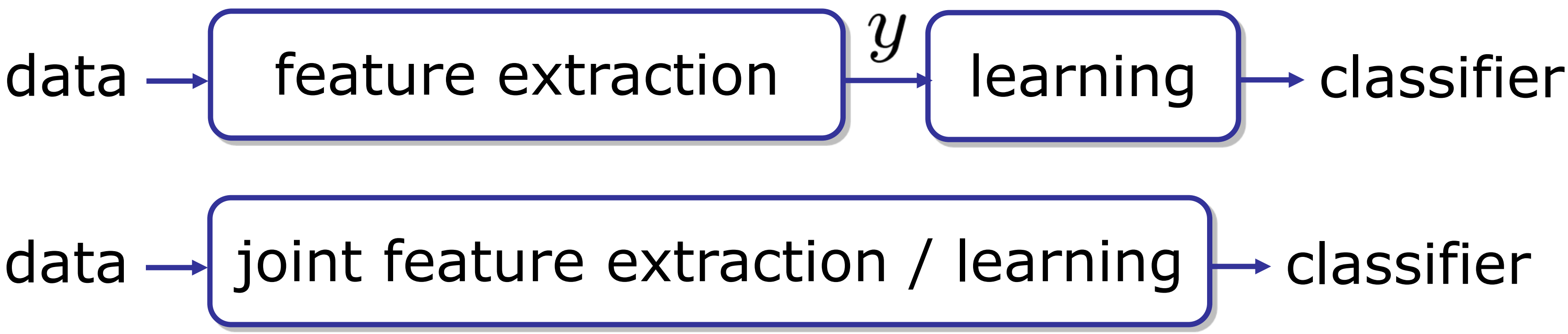
- We often have *models* for our data
- These models are usually *concise*
- Data vector $x \in \mathbf{R}^N$
- Can be described with K pieces of information, $K \ll N$
  - lies in a *subspace*
  - lies in a *union of subspaces*
  - lies on a *manifold*





Feature extraction and learning

We want a small set of features that contain as much information as possible: $y = \Phi x$



- joint feature extraction / learning is hard
- in some cases, feature extraction is an easy way to exploit prior knowledge
- splitting the process into two steps may actually help

Dimensionality reduction

- Nonlinear, adaptive
  - manifold-learning
  - learn a local set of features
  - model = manifold
- Linear, adaptive
  - PCA
  - learn a fixed set of features
  - model = subspace
- Linear, non-adaptive
  - fix a subspace, independent of the data
  - random projections
  - model = ???

Johnson-Lindenstrauss Lemma

For any set Q of points in $\mathbf{R}^N$ and $\epsilon \in (0, 1)$, w.h.p. a random $M \times N$ matrix Φ will satsify

$$(1-\epsilon) \leq \frac{\|\Phi(u-v)\|^2}{\|u-v\|^2} \leq (1+\epsilon)$$

for all $u, v \in Q$, provided $M = O(\ln(\#(Q))/\epsilon^2)$.

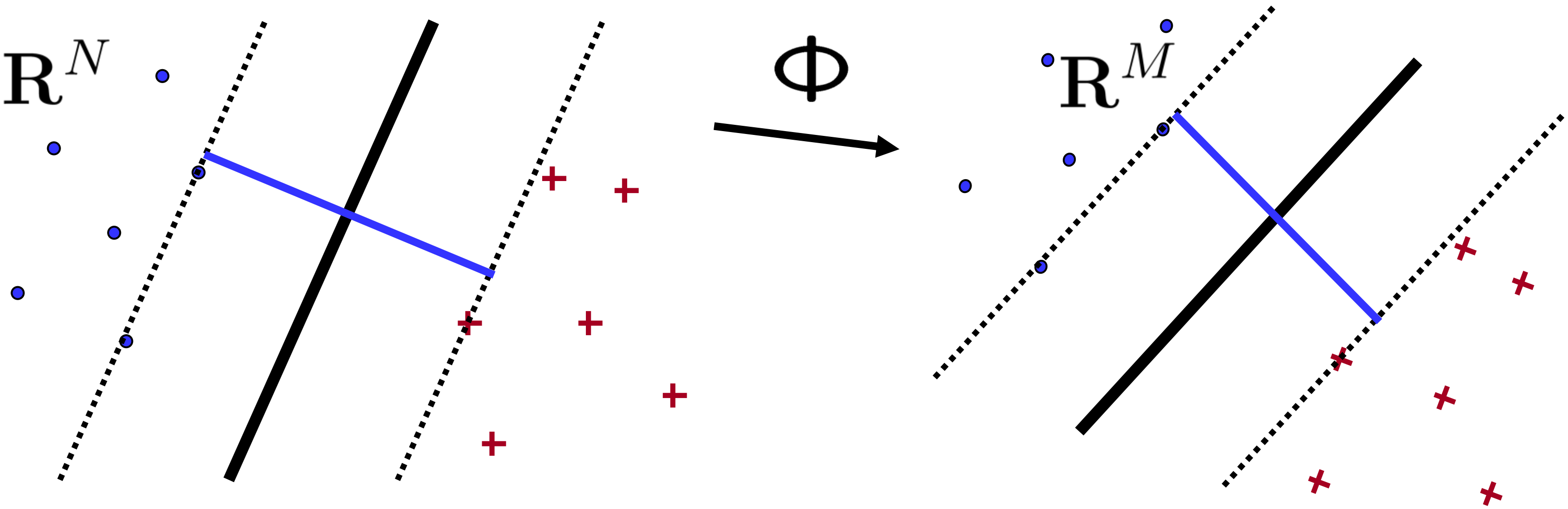
Key ingredients:

$$\mathbf{E}(\|\Phi x\|^2_{\ell_2^M}) = \|x\|^2_{\ell_2^N}$$

$$\mathbf{P}(|\,\|\Phi x\|^2_{\ell_2^M} - \|x\|^2_{\ell_2^N}| \geq \epsilon \|x\|^2_{\ell_2^N}) \leq 2e^{-CM\epsilon^2}$$

Classification

- If our classes are separable in $\mathbf{R}^N$, then they should remain separable in $\mathbf{R}^M$

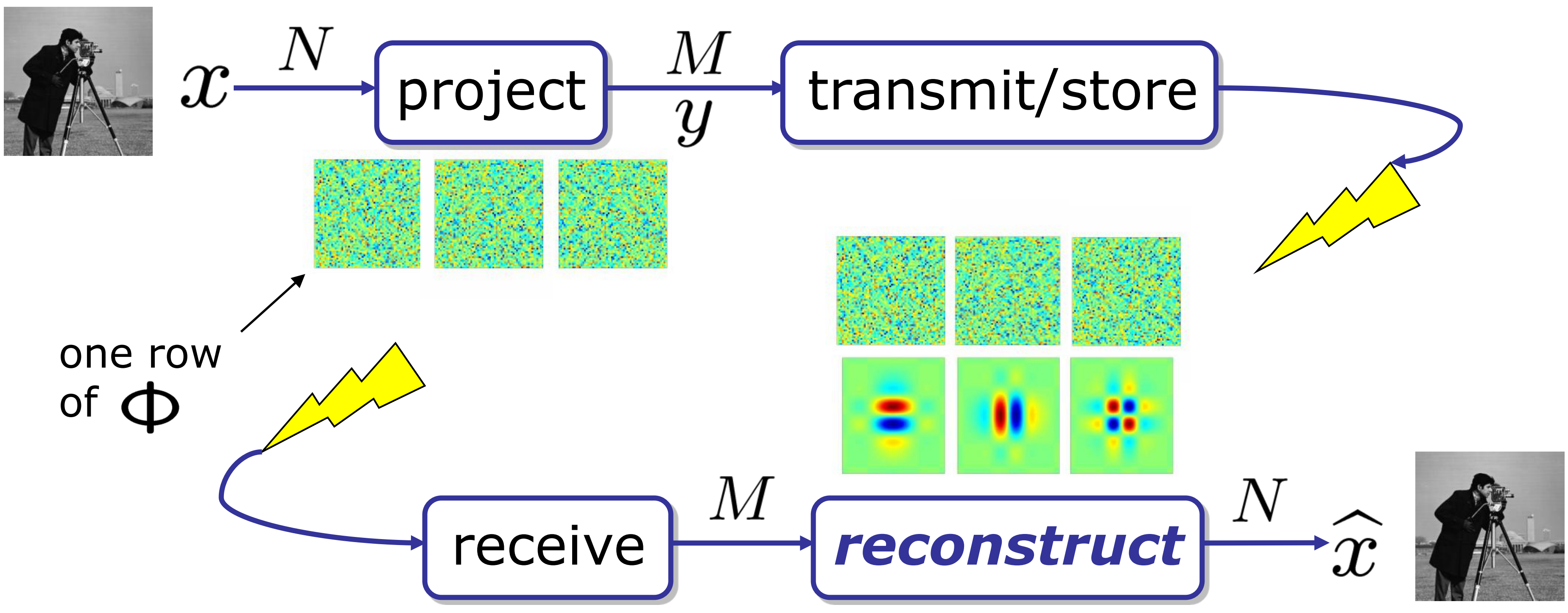


  - [Balcan, Blum, Vempala – 04, 05, 06]
  - [Rahimi and Recht – NIPS 07]

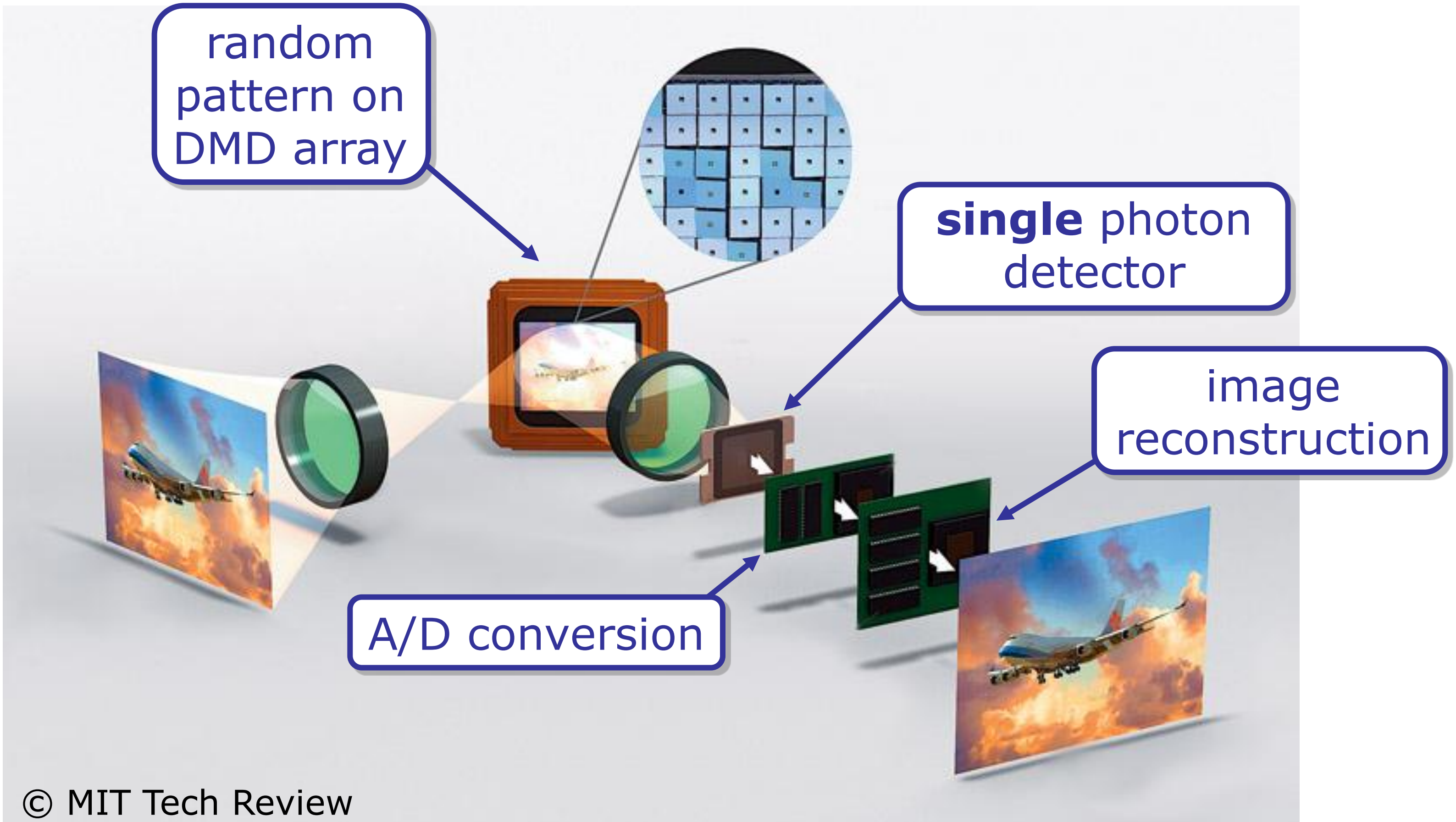
- How many projections do we need?

Compressive sensing

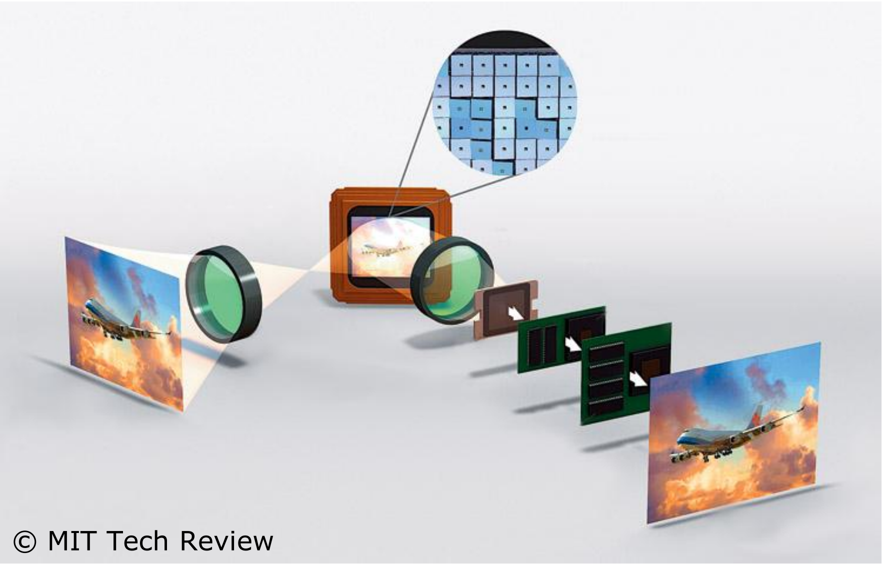
"***sparse*** signals can be recovered from a small number of ***nonadaptive linear measurements***"



“Computing” random projections



First image acquisition



ideal
256x256 pixels

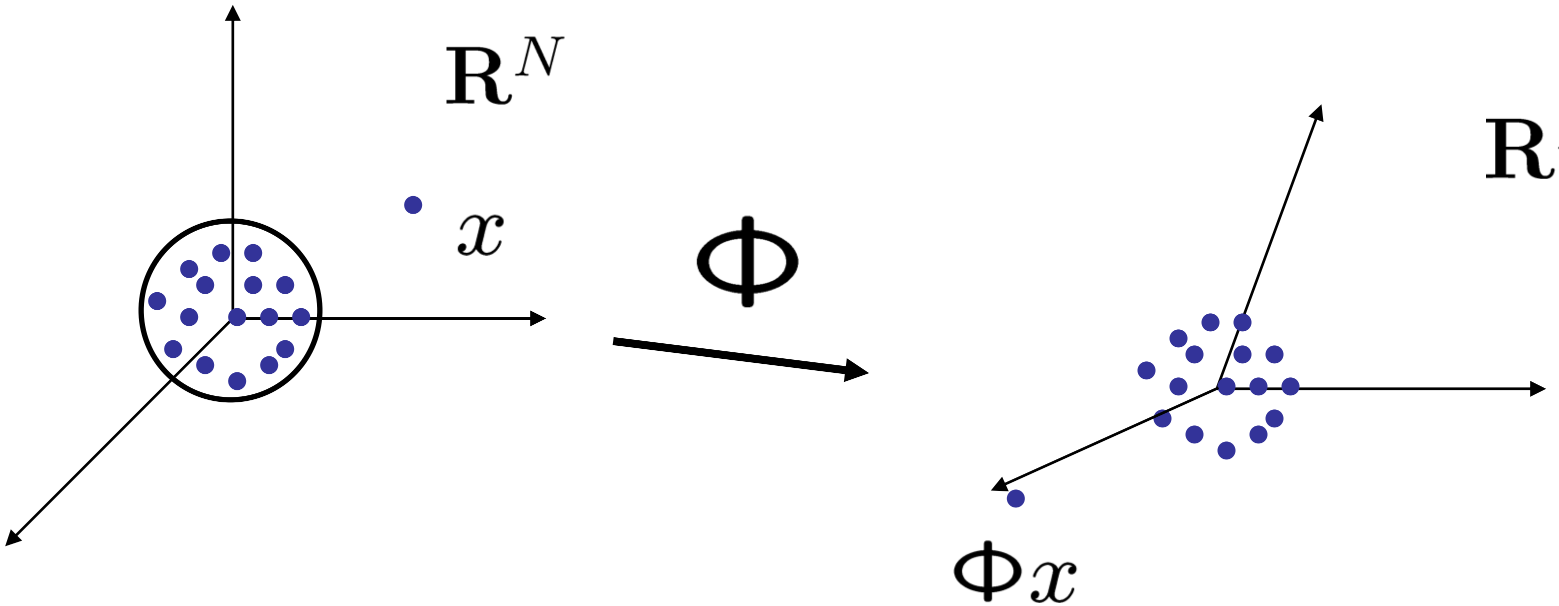
20x
sub-Nyquist

50x
sub-Nyquist

Embedding a subspace

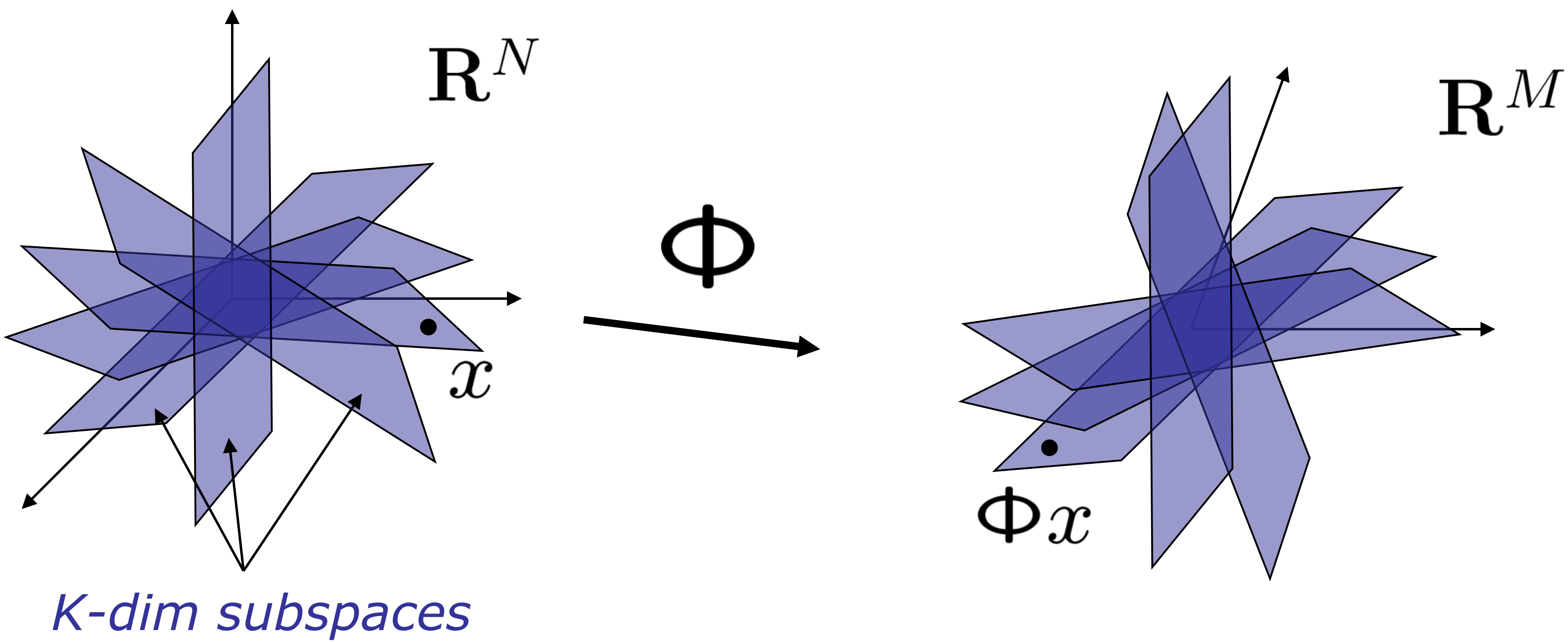
Effect of random projections on a subspace

- construct ϵ-net of points on S^{K-1} : Q
- JL: union bound $\rightarrow$ isometry for all $q \in Q$
- extend to isometry for entire subspace
- Q should have $O(N^K)$ points $\rightarrow M = O(K \ln(N))$



Embedding a union of subspaces

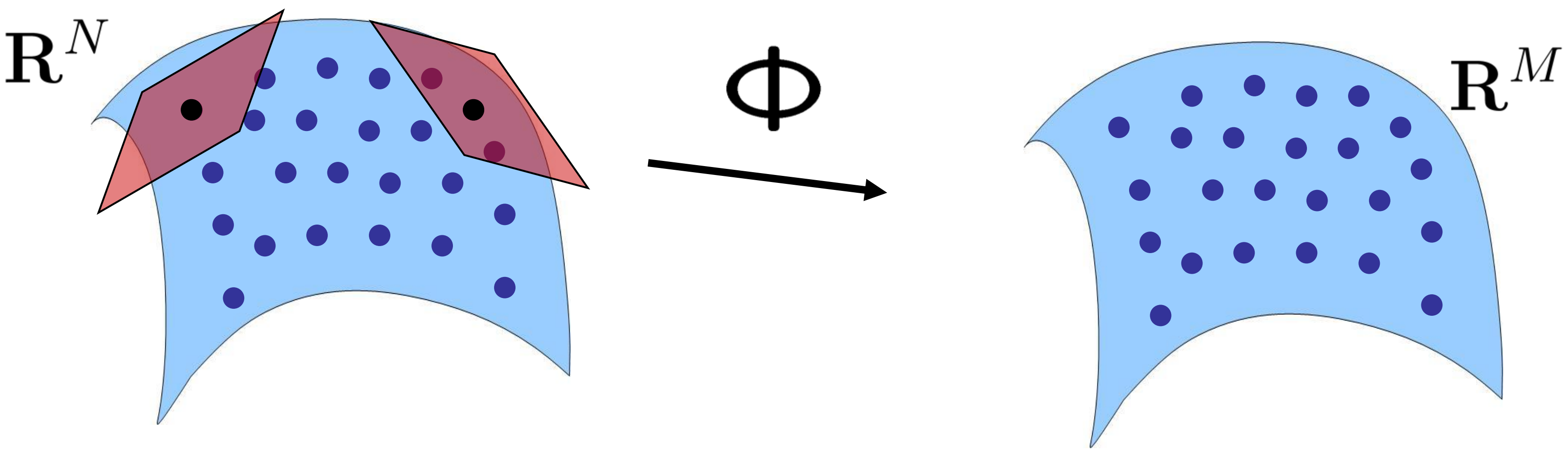
- Take a union over all $\binom{N}{K}$ subspaces
- Random projections are (near) isometries for the class of sparse signals
- Still only need $M = O(K \ln(N))$



Embedding a manifold

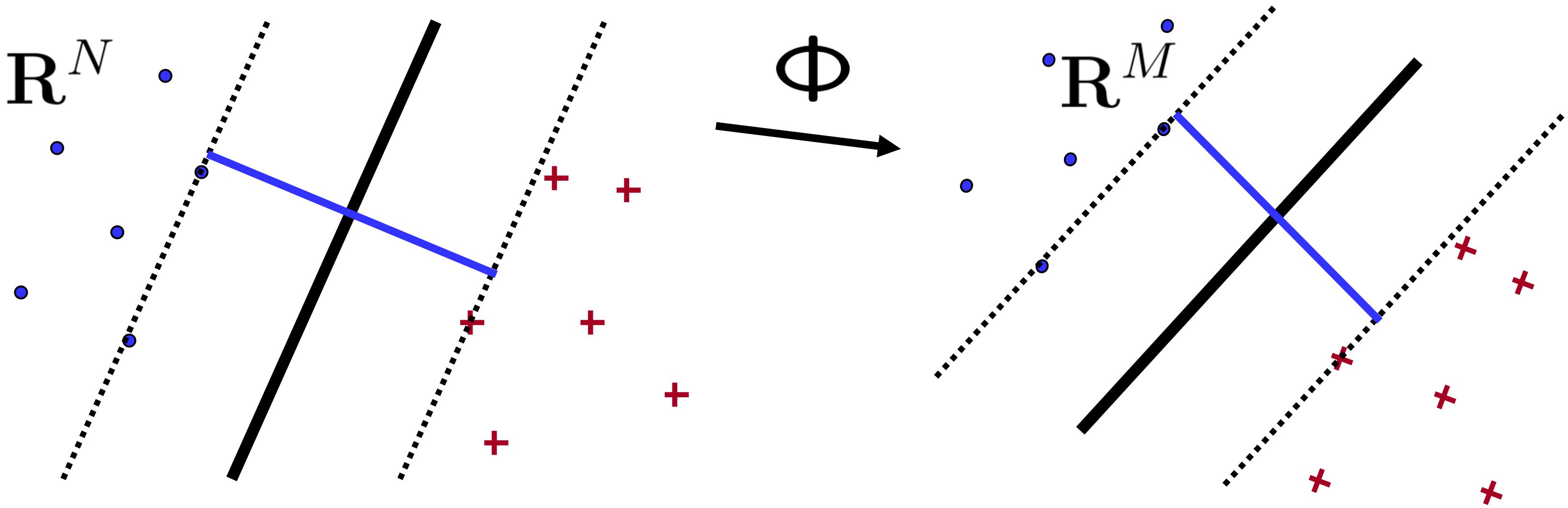
Suppose K-dim manifold is *compact, smooth*

- construct a sampling of points on manifold
- construct a sampling of points from local tangent spaces
- need $O(N^K)$ points $\rightarrow$ $M = O(K \ln(N))$



Classification

- If our classes are separable in $\mathbf{R}^N$, then they should remain separable in $\mathbf{R}^M$

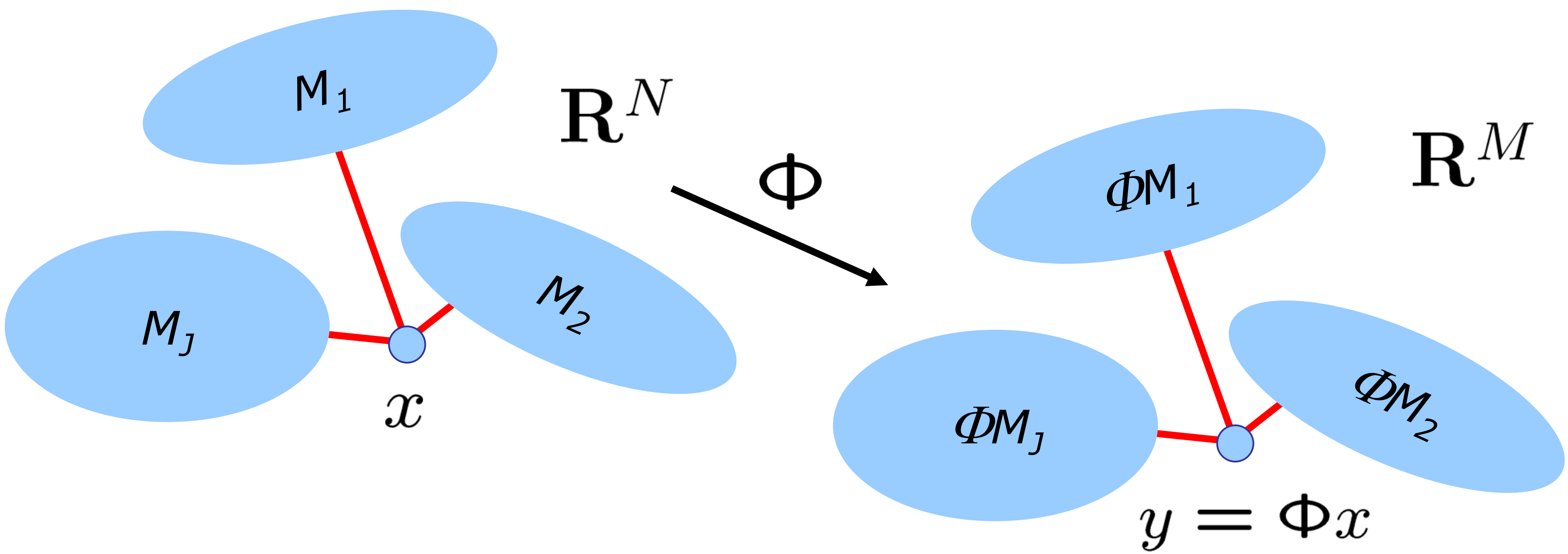


  - [Balcan, Blum, Vempala – 04, 05, 06]
  - [Rahimi and Recht – NIPS 07]

- How many projections do we need?
  - potentially many fewer than previously thought

Smashed filtering

- Many classification problems can be posed as a “nearest manifold” search
  - classical matched filter
  - object recognition
  - speaker identification

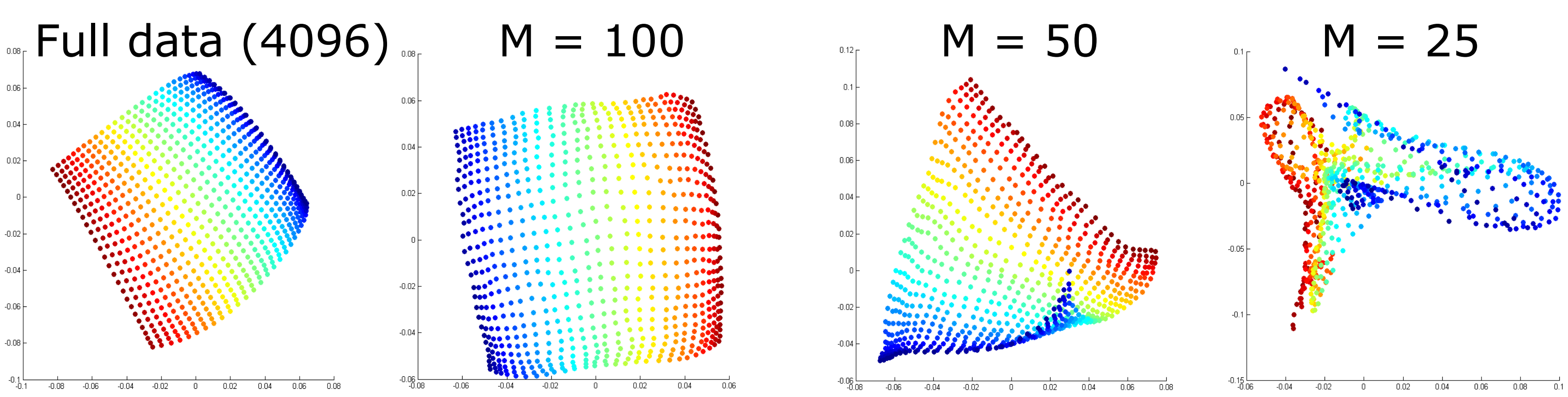


Manifold learning

- ISOMAP
  - uses pairwise distances between data points

If $M > O(K \ln N/\delta^2)$, then the ISOMAP residual variance estimate in the projected domain is bounded by an additive error factor:

$$R_{\Phi} < R + C\delta$$



Intrinsic dimension estimation

- Grassberger-Procaccia Algorithm for estimation of intrinsic dimension
  - also uses pairwise distances between data points

If $M > O(K \ln N/\delta^2)$, then the GP estimate in the projected domain is bounded by a multiplicative error factor:

$$(1-\delta)\bar{K} < K_\Phi < (1+\delta)\bar{K}$$

- Many more possibilities
  - [Hegde – NIPS 07]

Conclusions

Random projections

- useful feature extraction technique when the data obeys a *simple model*
- number of projections required *does not* grow with size of the data set
- in some cases, can be obtained at *almost zero computational cost*
- important baseline to compare against

dsp.rice.edu
dsp.rice.edu/cs