Submodular optimization and compressed sensing

A ~~nighmare~~comparison

Let’s start with a matrix

- I have some large matrix $X \in \mathcal{R}^{n \times d}$
- I want to reduce the dimensionality of this matrix to $X \in \mathcal{R}^{n \times m}$ where

$$m << d$$

- Common approaches to this task include *feature selection* methods and *feature engineering* methods

Selection versus engineering

- Feature *selection* is when you select a subset of features and maintain their original values. Filtering by variance or correlation with a target are examples of feature selection.

- Feature *engineering* is when you use the existing features to craft new features you anticipate will be more informative or denoised. All embedding methods (e.g. PCA and matrix factorization but also MMD-MA), model stacking, etc. are feature engineering.

- Either or both can be used prior to transform raw data into input to analysis or machine learning models

Overview

- ***Submodular optimization*** with a cardinality constraint can be used to select features that are minimally redundant with each other and so can be used as a feature selection method.

- ***Compressed sensing*** studies properties of features derived from linear combinations of the old feature space, which is a feature engineering method.

Defining submodularity

Submodular functions take the following form:

$$f(X \cup \{v\}) - f(X) \geq f(Y \cup \{v\}) - f(Y)$$

where $X \subseteq Y$ and $v \notin Y$

Defining submodularity

Submodular functions take the following form:

$$f(X \cup \{v\}) - f(X) \geq f(Y \cup \{v\}) - f(Y)$$

where $X \subseteq Y$ and $v \notin Y$

For example, consider a function that counts the number of scenes each character is in:

f( ) = 10

f( ) = 17

Defining submodularity

Submodular functions take the following form:

$$f(X \cup \{v\}) - f(X) \geq f(Y \cup \{v\}) - f(Y)$$

where $X \subseteq Y$ and $v \notin Y$

For example, consider a function that counts the number of scenes each character is in:

f(,) = 20

Some scenes have both characters so the number of scenes either character is in is less than 27.

Defining submodularity

Submodular functions take the following form:

$$f(X \cup \{v\}) - f(X) \geq f(Y \cup \{v\}) - f(Y)$$

where $X \subseteq Y$ and $v \notin Y$

For example, consider a function that counts the number of scenes each character is in:

f(, ,) = 21

Jenny doesn’t add much to the function, just like she doesn’t add much to the show.

The max-coverage function

$$f(X) = \sum_{i=1}^{d} \left(1_{\sum_{x \in X} x_i > 1} \right)$$

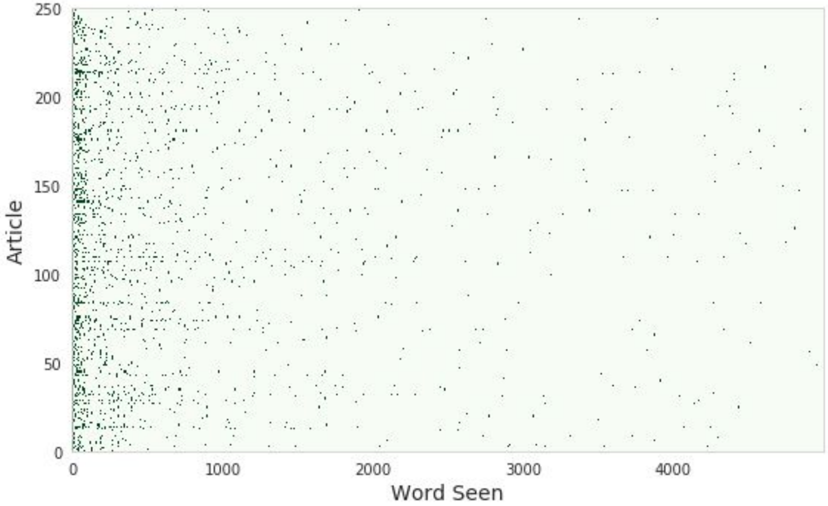
This submodular function tries to select items that have the most newly observed entries.

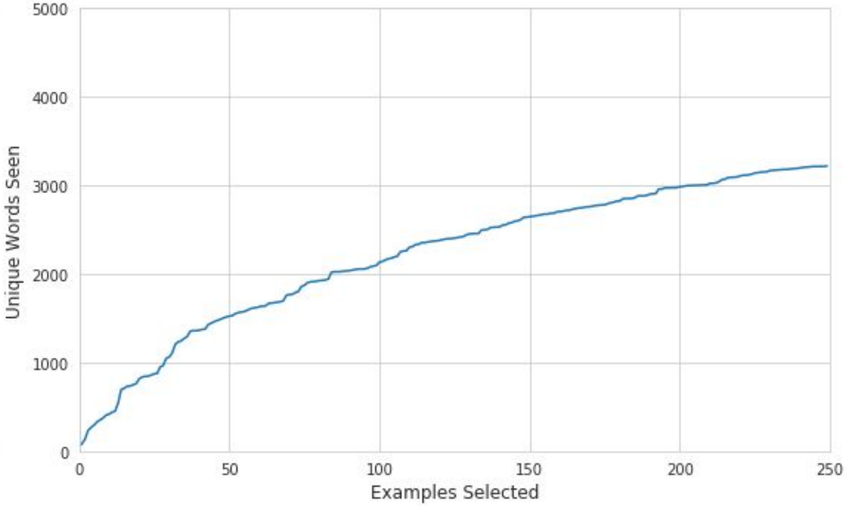
If optimized over examples (selecting rows) it can select examples that, together, have a large number of features with at least 1 non-zero.

If optimized over features (select columns) it can select features that, together, are present in the maximum number of examples.

The max-coverage function

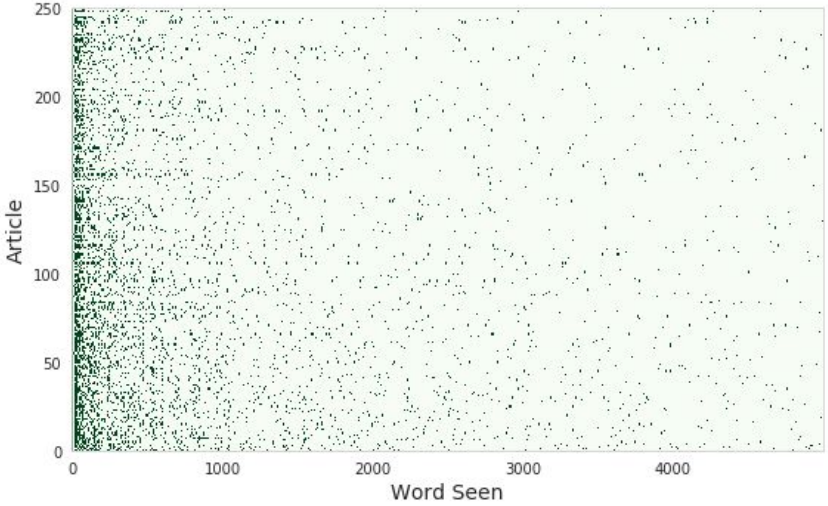
$$f(X) = \sum_{i=1}^{d} \left(1_{\sum_{x \in X} x_i > 1} \right)$$

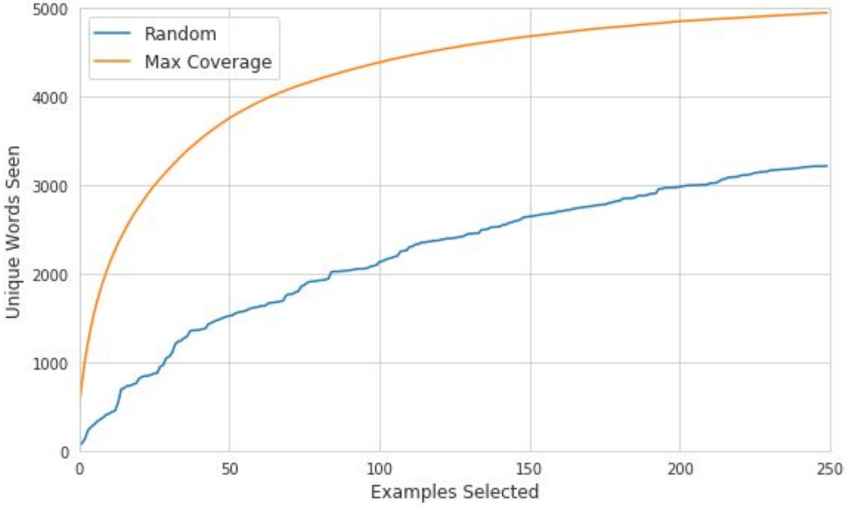




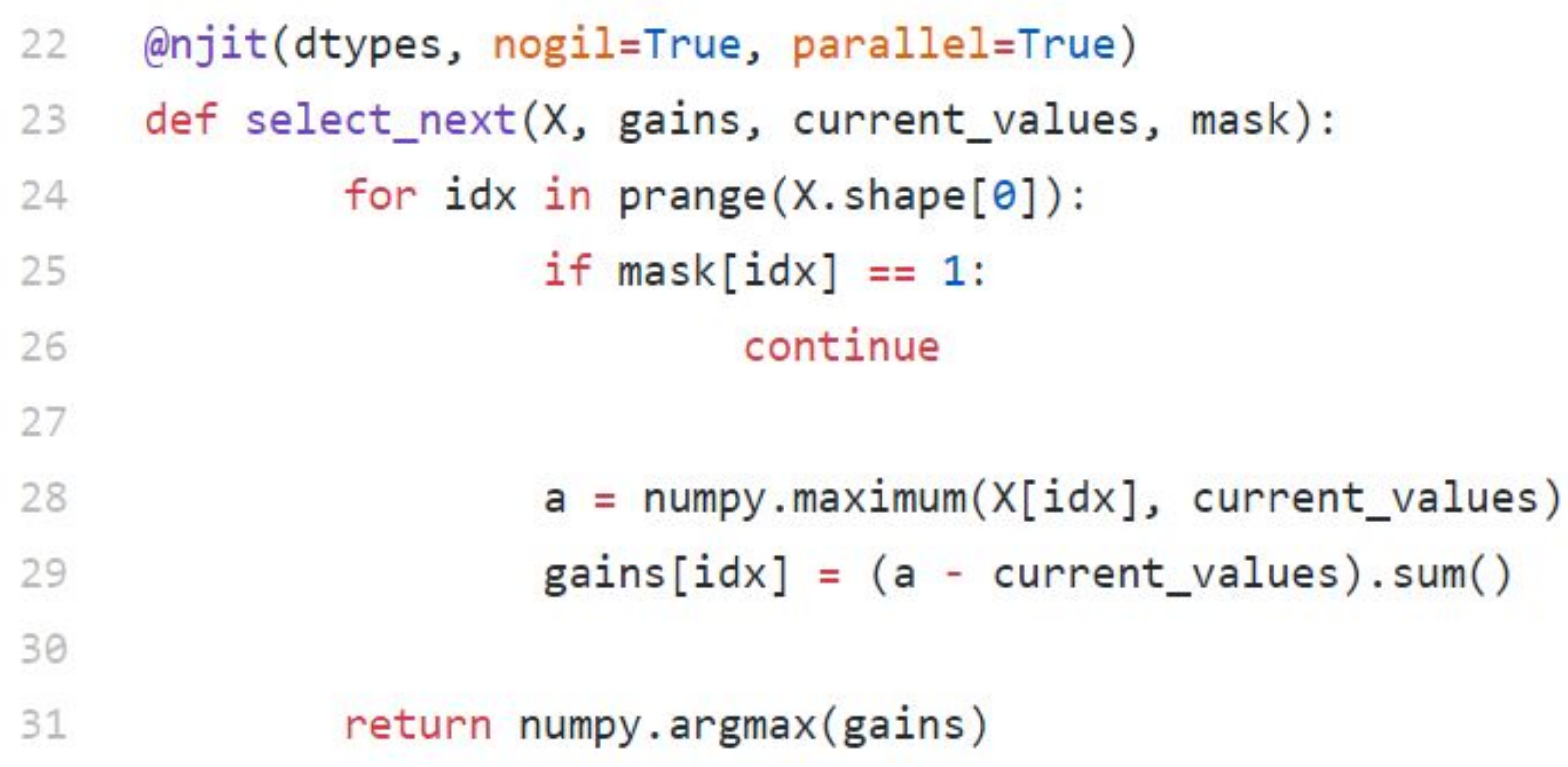
The max-coverage function

$$f(X) = \sum_{i=1}^{d} \left(1_{\sum_{x \in X} x_i > 1} \right)$$





Submodular functions can be optimized greedily



```
@njit(dtypes, nogil=True, parallel=True)
def select_next(X, gains, current_values, mask):
        for idx in prange(X.shape[0]):
                if mask[idx] == 1:
                        continue

                a = numpy.maximum(X[idx], current_values)
                gains[idx] = (a - current_values).sum()

        return numpy.argmax(gains)
```

Submodular optimization has theoretical guarantees

- The function value of the selected elements is guaranteed to be within **$1 - e^{-1}$** of the optimal selection of elements (through a non-greedy method)

- Because these features are selected to be minimally redundant with each other (according to a submodular measure of redundancy) they can compress the data into its more distinct parts

- Can operate on any type and distributions of data but no guarantees on how well the selected elements can reconstruct the original data

Defining compressed sensing

- Compressed sensing studies the compression of a sparse signal **x** into a dense signal **y** using linear combinations of **x**, i.e.

$$y = Ax$$

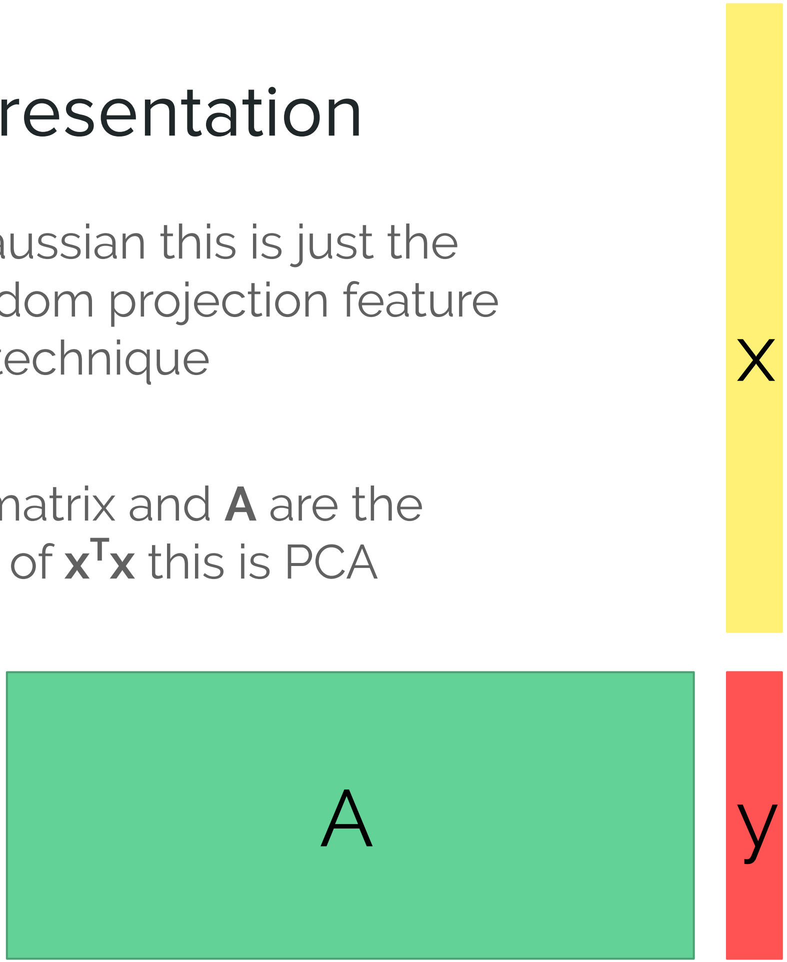
- **x** and **y** can be either a single vector or a matrix and **x** is **k**-sparse if it has at most **k** non-zero entries

- The “sensing matrix” **A** can be e.g. Gaussian or Bernoulli

A visual representation

- When **A** is Gaussian this is just the Gaussian random projection feature engineering technique

- When **x** is a matrix and **A** are the eigenvectors of **x**^{T}**x** this is PCA



sparse :(

$$y = Ax$$

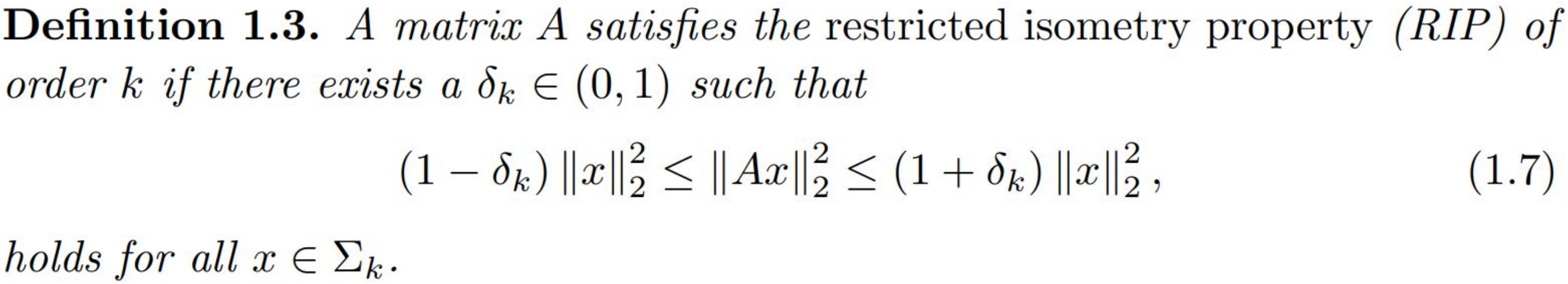
dense :)

An unnatural detour into the mathematical.

All figures, definitions, and theorems are from http://www.ecs.umass.edu/~mduarte/images/IntroCS.pdf

How do we construct the sensing matrix?

- We want an **A** such that distances in **x** are preserved in **y**, a property known as the “restricted isometry property” (RIP). Let’s assume that there is no noise in **y**.

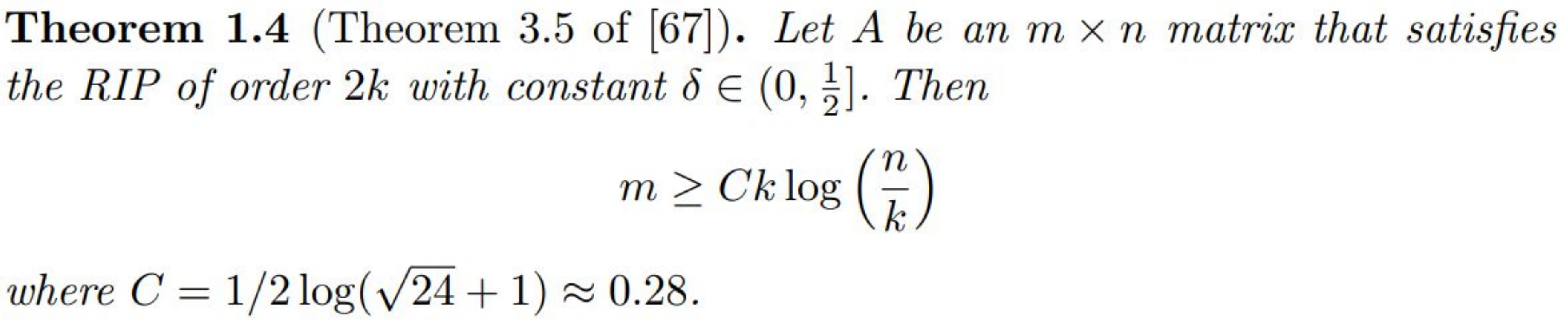


Definition 1.3. *A matrix A satisfies the* restricted isometry property *(RIP) of order k if there exists a $\delta_k \in (0,1)$ such that*

$$(1-\delta_k)\left\|x\right\|_2^2 \leq \left\|Ax\right\|_2^2 \leq (1+\delta_k)\left\|x\right\|_2^2, \qquad (1.7)$$

holds for all $x \in \Sigma_k$.

From the RIP we can derive a good size for A



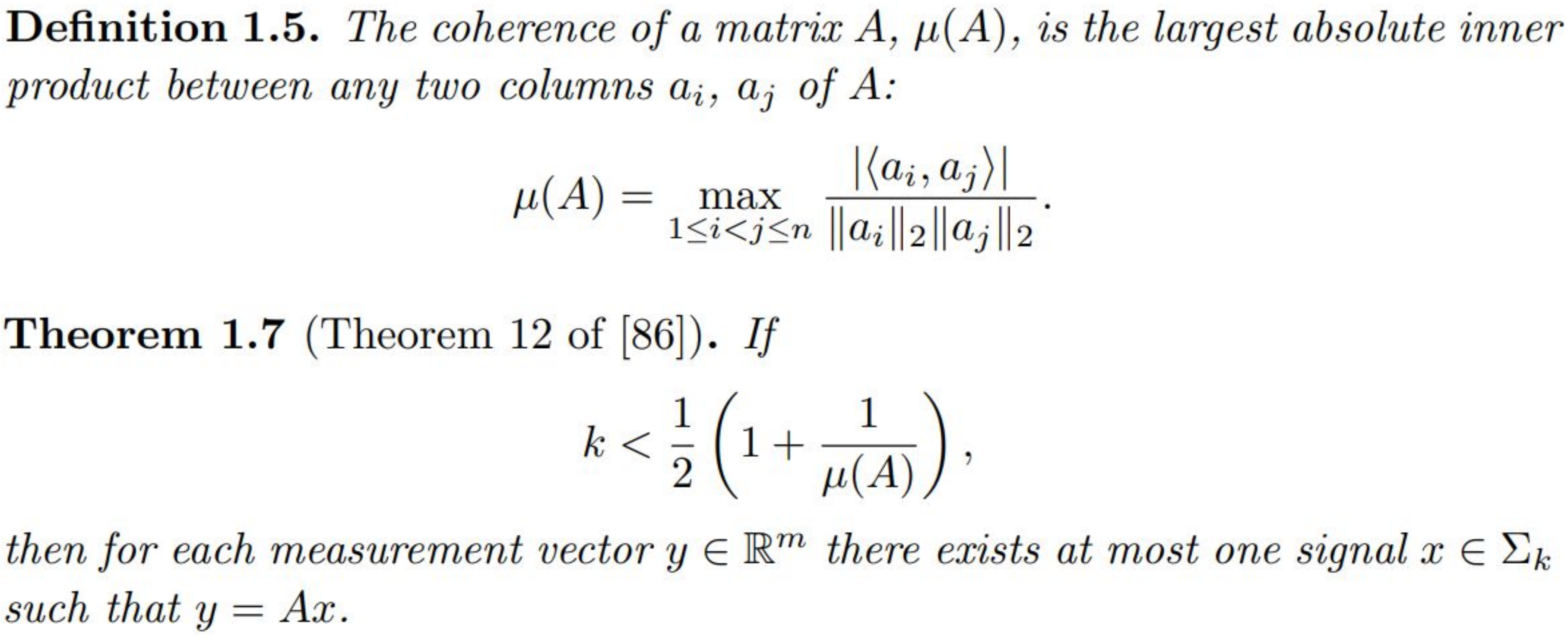
Theorem 1.4 (Theorem 3.5 of [67]). *Let A be an $m \times n$ matrix that satisfies the RIP of order $2k$ with constant $\delta \in (0, \frac{1}{2}]$. Then*

$$m \geq Ck \log\left(\frac{n}{k}\right)$$

where $C = 1/2 \log(\sqrt{24} + 1) \approx 0.28$.

- Given the sparsity of **x** and the number of dimensions in the original space we can derive the size of **y** that is needed for exact recovery

We can account for sparsity using the coherence



Definition 1.5. *The coherence of a matrix A, $\mu(A)$, is the largest absolute inner product between any two columns a_i, a_j of A:*

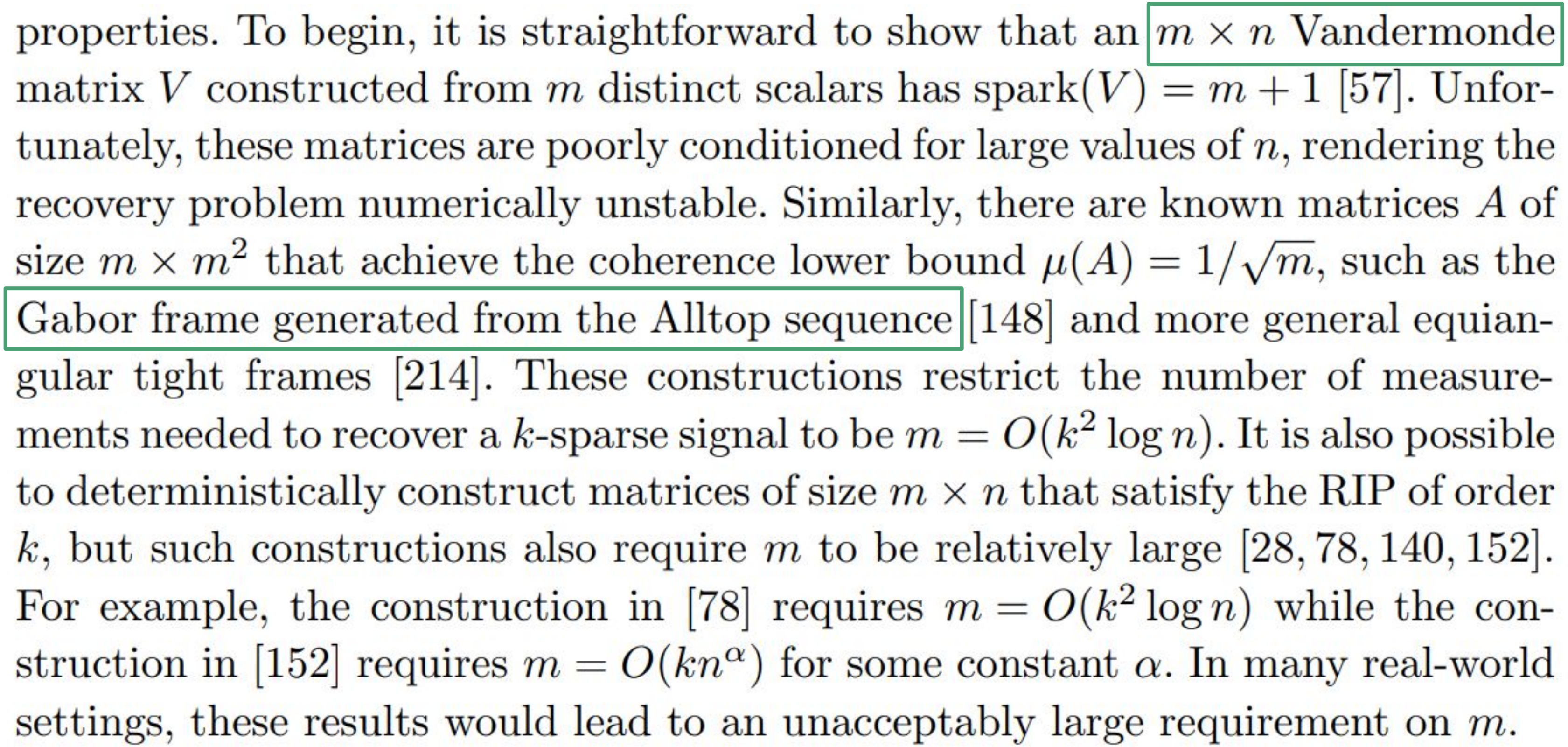
$$\mu(A) = \max_{1 \leq i < j \leq n} \frac{|\langle a_i, a_j \rangle|}{\|a_i\|_2 \|a_j\|_2}.$$

Theorem 1.7 (Theorem 12 of [86]). *If*

$$k < \frac{1}{2}\left(1 + \frac{1}{\mu(A)}\right),$$

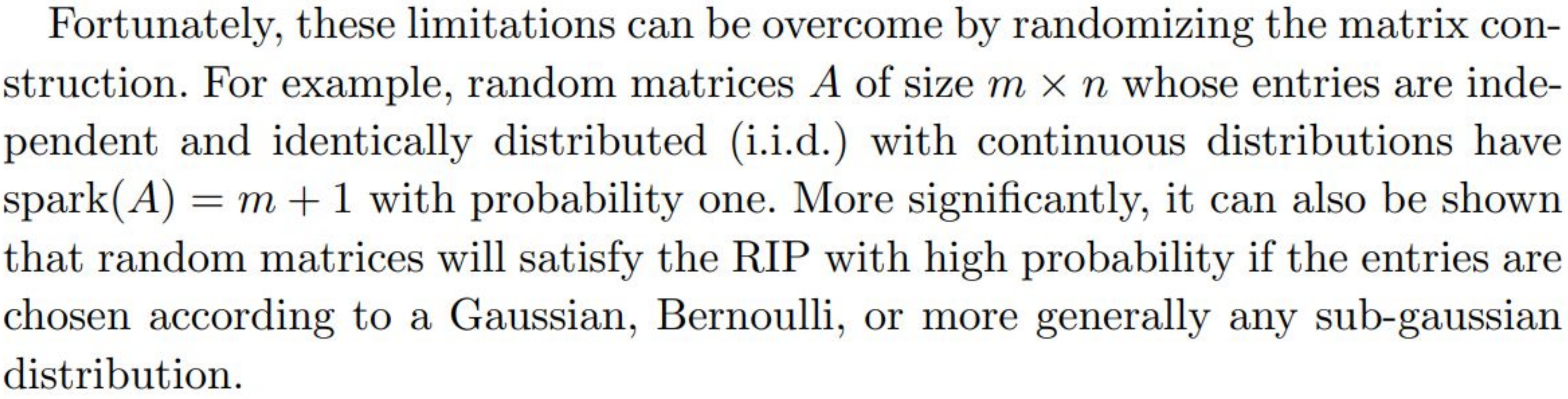
then for each measurement vector $y \in \mathbb{R}^m$ there exists at most one signal $x \in \Sigma_k$ such that $y = Ax$.

There are complicated ideas for how to construct **A**



properties. To begin, it is straightforward to show that an $m \times n$ Vandermonde matrix V constructed from m distinct scalars has $\operatorname{spark}(V) = m + 1$ [57]. Unfortunately, these matrices are poorly conditioned for large values of n, rendering the recovery problem numerically unstable. Similarly, there are known matrices A of size $m \times m^2$ that achieve the coherence lower bound $\mu(A) = 1/\sqrt{m}$, such as the Gabor frame generated from the Alltop sequence [148] and more general equiangular tight frames [214]. These constructions restrict the number of measurements needed to recover a k-sparse signal to be $m = O(k^2 \log n)$. It is also possible to deterministically construct matrices of size $m \times n$ that satisfy the RIP of order k, but such constructions also require m to be relatively large [28, 78, 140, 152]. For example, the construction in [78] requires $m = O(k^2 \log n)$ while the construction in [152] requires $m = O(kn^\alpha)$ for some constant α. In many real-world settings, these results would lead to an unacceptably large requirement on m.

There are also simple ways to construct **A**



Fortunately, these limitations can be overcome by randomizing the matrix construction. For example, random matrices A of size $m \times n$ whose entries are independent and identically distributed (i.i.d.) with continuous distributions have $\mathrm{spark}(A) = m + 1$ with probability one. More significantly, it can also be shown that random matrices will satisfy the RIP with high probability if the entries are chosen according to a Gaussian, Bernoulli, or more generally any sub-gaussian distribution.

- Key point: There is math to back up just using random numbers for **A**

Compression is cool but reconstruction is super cool

- The key point about compressed sensing is that in noise-free circumstances **x** can be recovered exactly from **y**

- Even under realistic data generation conditions (uniform and Gaussian noise) recovery can still be very good

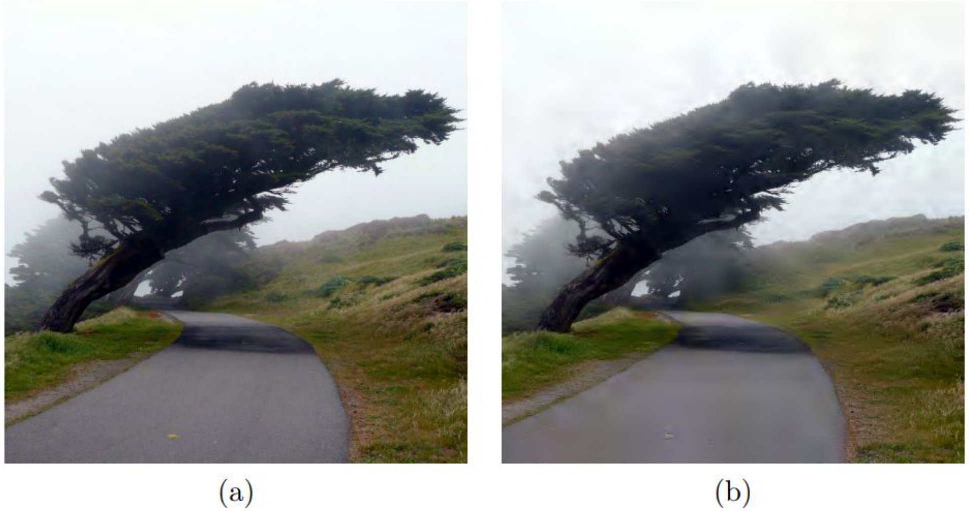


Figure 1.4 Sparse approximation of a natural image. (a) Original image. (b) Approximation of image obtained by keeping only the largest 10% of the wavelet coefficients.

A series of unfortunate examples

- Many introductions to compressed sensing involve reconstructing images, which are not sparse?

- These examples require the transformation of the image (or other signal) into a sparse signal.

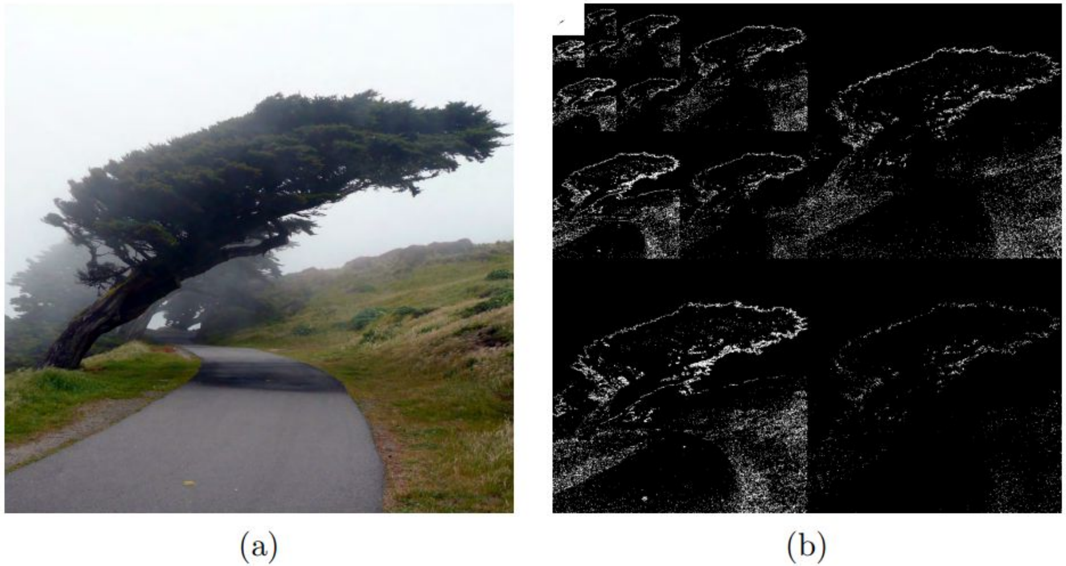


Figure 1.3 Sparse representation of an image via a multiscale wavelet transform.

Reconstruction is a simple optimization problem

- Given **y** and **A** where **y** is known to be noise-free

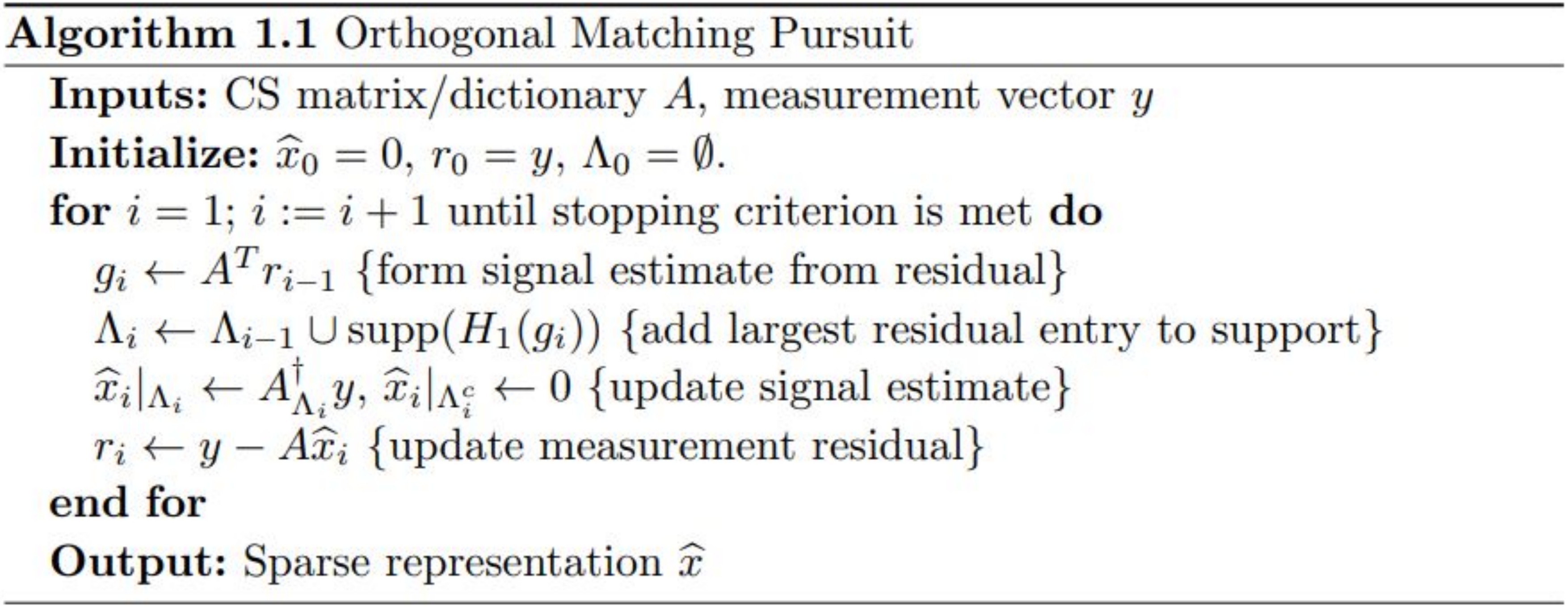
$$\hat{x} = \min_x ||x||_1 \quad \text{subject to} \quad y = A\hat{x}$$

- When **y** is known to be noisy

$$\hat{x} = \min_x ||y - Ax||_2 + \gamma||x||_1$$

- Note the L1 norm. L0 would be preferable but difficult to optimize.

Alternatively reconstruction is a greedy algorithm

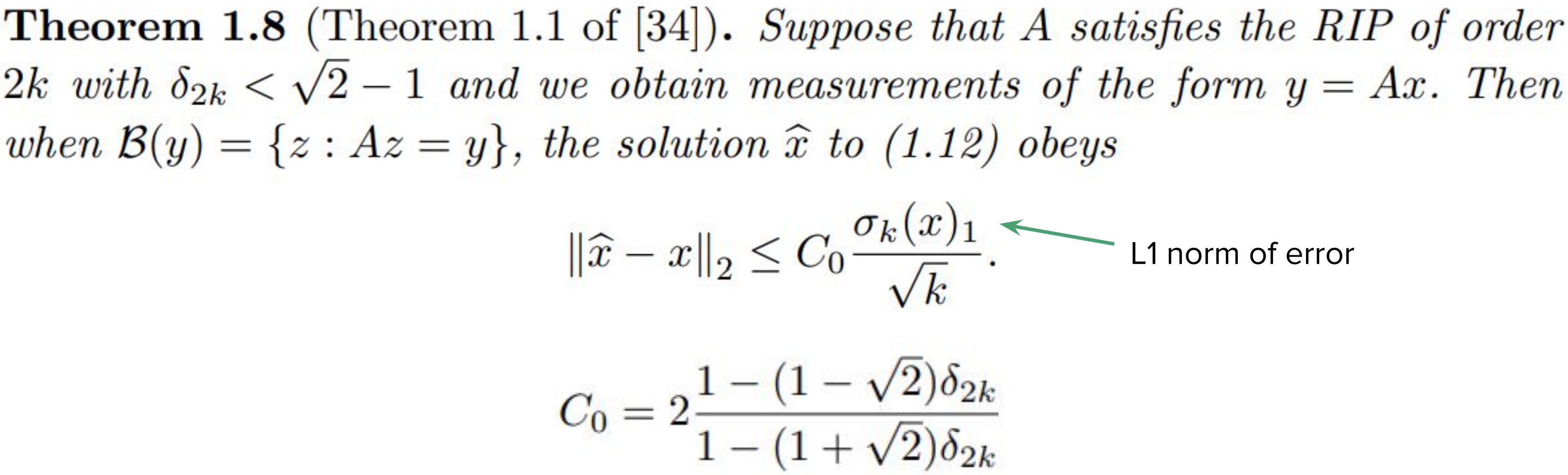


Algorithm 1.1 Orthogonal Matching Pursuit

```
Inputs: CS matrix/dictionary $A$, measurement vector $y$
Initialize: $\widehat{x}_0 = 0$, $r_0 = y$, $\Lambda_0 = \emptyset$.
for $i = 1; i := i + 1$ until stopping criterion is met do
    $g_i \leftarrow A^T r_{i-1}$ {form signal estimate from residual}
    $\Lambda_i \leftarrow \Lambda_{i-1} \cup \mathrm{supp}(H_1(g_i))$ {add largest residual entry to support}
    $\widehat{x}_i|_{\Lambda_i} \leftarrow A^{\dagger}_{\Lambda_i} y$, $\widehat{x}_i|_{\Lambda_i^c} \leftarrow 0$ {update signal estimate}
    $r_i \leftarrow y - A\widehat{x}_i$ {update measurement residual}
end for
Output: Sparse representation $\widehat{x}$
```

Provably good reconstruction if A follows RIP

- To me, **THIS IS THE KEY POINT (1/2).**



Theorem 1.8 (Theorem 1.1 of [34]). *Suppose that A satisfies the RIP of order $2k$ with $\delta_{2k} < \sqrt{2} - 1$ and we obtain measurements of the form $y = Ax$. Then when $\mathcal{B}(y) = \{z : Az = y\}$, the solution $\widehat{x}$ to (1.12) obeys*

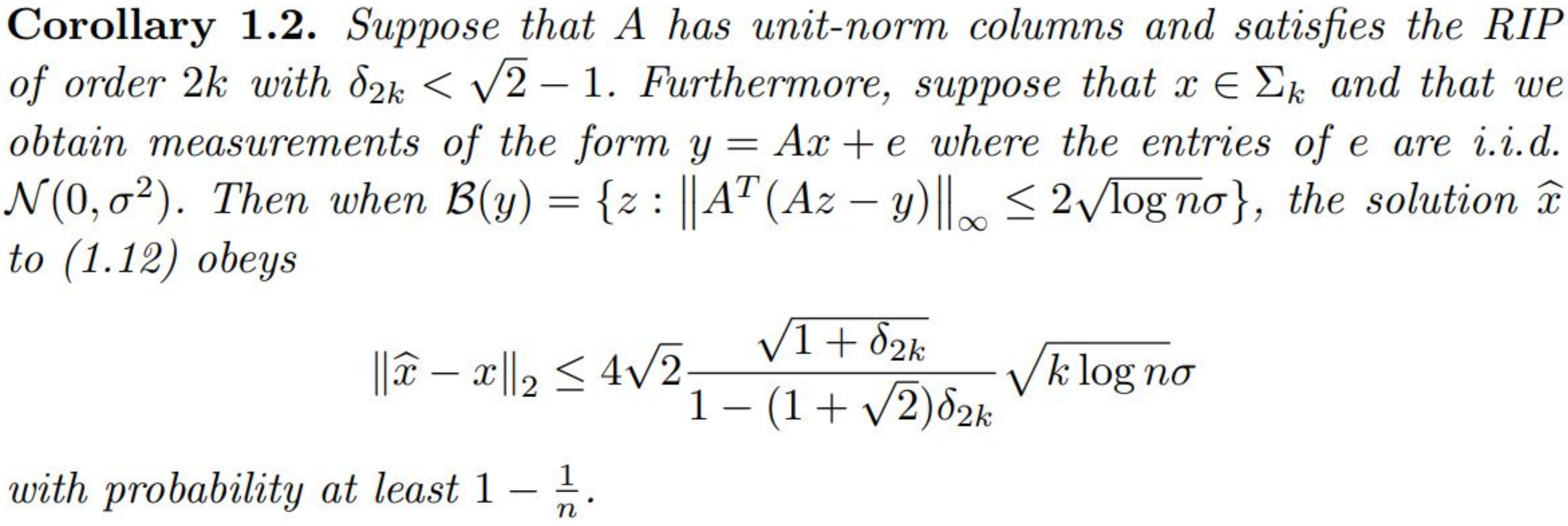
$$\|\widehat{x} - x\|_2 \leq C_0 \frac{\sigma_k(x)_1}{\sqrt{k}}.$$

L1 norm of error

$$C_0 = 2\frac{1 - (1 - \sqrt{2})\delta_{2k}}{1 - (1 + \sqrt{2})\delta_{2k}}$$

Provably good even under Gaussian noise

- To me, **THIS IS THE KEY POINT (2/2).**



Corollary 1.2. *Suppose that A has unit-norm columns and satisfies the RIP of order $2k$ with $\delta_{2k} < \sqrt{2} - 1$. Furthermore, suppose that $x \in \Sigma_k$ and that we obtain measurements of the form $y = Ax + e$ where the entries of e are i.i.d. $\mathcal{N}(0, \sigma^2)$. Then when $\mathcal{B}(y) = \{z : \left\| A^T(Az - y) \right\|_\infty \leq 2\sqrt{\log n}\sigma\}$, the solution $\widehat{x}$ to (1.12) obeys*

$$\|\widehat{x} - x\|_2 \leq 4\sqrt{2}\frac{\sqrt{1 + \delta_{2k}}}{1 - (1 + \sqrt{2})\delta_{2k}}\sqrt{k \log n}\sigma$$

with probability at least $1 - \frac{1}{n}$.

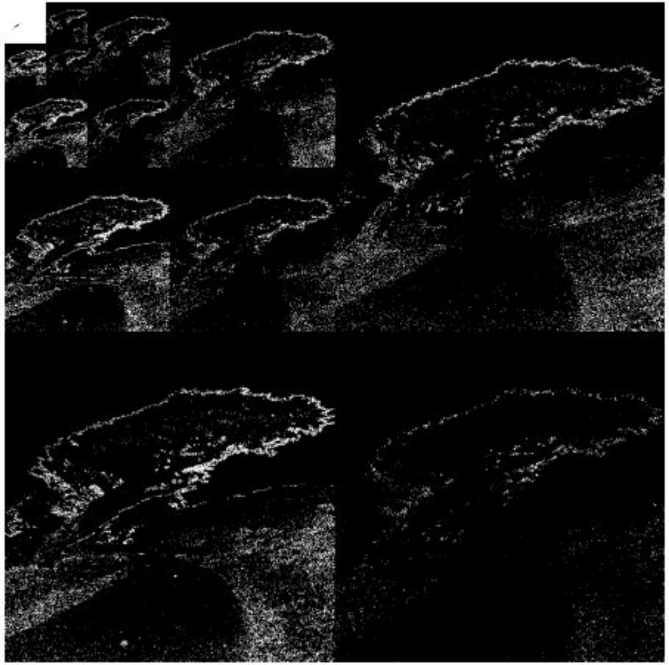
Aren’t there a million methods for compressing data?

- Compressed sensing studies the **recovery of signals that are sparse** when **you can only measure linear combinations** of these sparse signals

- If your goal is solely to compress some data X as much as possible you’ll likely want to use a non-linear method such as an autoencoder

- If you can design a data acquisition process that can collect these combinations cheaper than the raw data you can save time and money

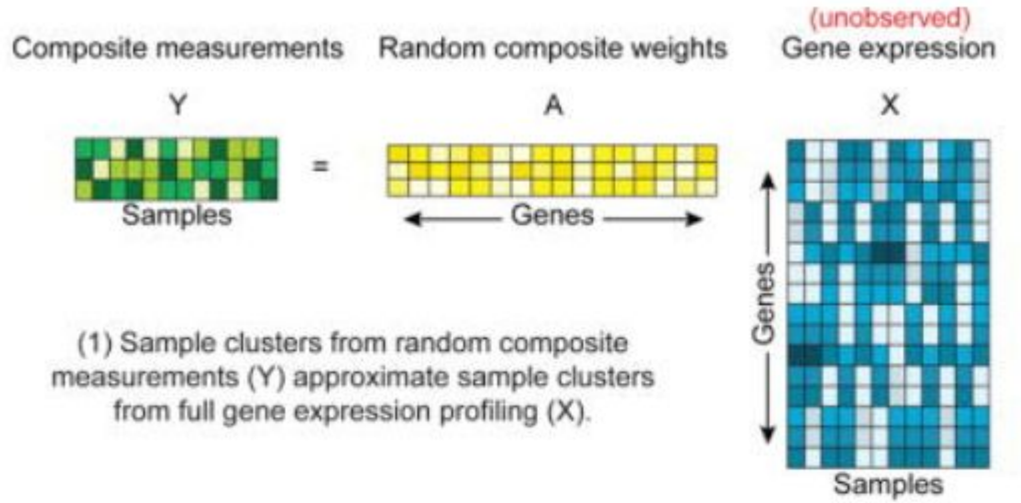
For example a camera that captures wavelets



Acquire and store

Reconstruct when necessary

For example measuring gene expression



(1) Sample clusters from random composite measurements (Y) approximate sample clusters from full gene expression profiling (X).

Efficient generation of transcriptomic profiles by random composite measurements https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5726792/#!po=19.7674

Summary

- At first glance both are related to compressing a matrix into one with fewer columns.

- Submodular optimization subject to a cardinality constraint can be used to select a diverse set of features but has no guarantees about reconstruction of the original signal.

- Compressed sensing provides strong guarantees on the ability to reconstructing sparse signals from linear combinations of that signal that are directly measured. It is only somewhat concerned with compressing an acquired signal.