ESE 531: Digital Signal Processing

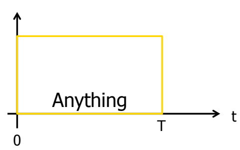
Lec 26: April 25, 2019

Compressive Sensing

Today

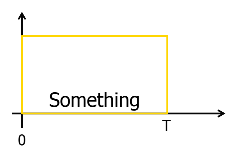
- Compressive Sampling/Sensing

Compressive Sampling



- What is the rate you need to sample at?
  - At least Nyquist

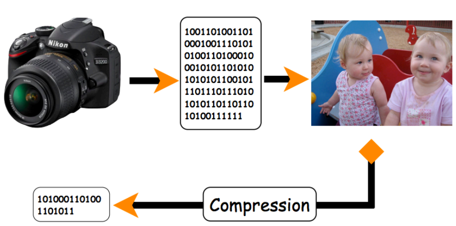
Compressive Sampling



- What is the rate you need to sample at?
  - Maybe less than Nyquist…

First: Compression

- Standard approach
  - First collect, then compress
    - Throw away unnecessary data



First: Compression

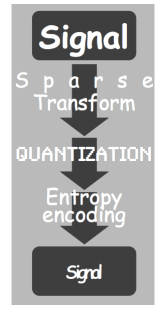
- Examples
  - Audio – 10x
    - Raw audio: 44.1kHz, 16bit, stereo = 1378 Kbit/sec
    - MP3: 44.1kHz, 16 bit, stereo = 128 Kbit/sec
  - Images – 22x
    - Raw image (RGB): 24bit/pixel
    - JPEG: 1280x960, normal = 1.09bit/pixel
  - Videos – 75x
    - Raw Video: (480x360)p/frame x 24b/p x 24frames/s + 44.1kHz x 16b x 2 = 98,578 Kbit/s
    - MPEG4: 1300 Kbit/s

First: Compression

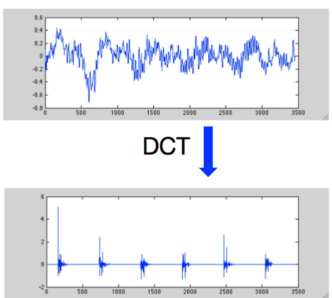
- Almost all compression algorithm use transform coding
  - mp3: DCT
  - JPEG: DCT
  - JPEG2000: Wavelet
  - MPEG: DCT & time-difference

First: Compression

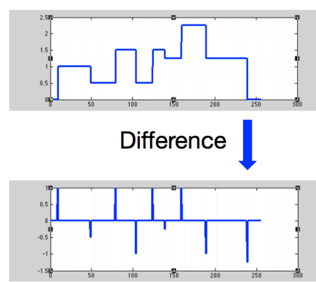
- Almost all compression algorithm use transform coding
  - mp3: DCT
  - JPEG: DCT
  - JPEG2000: Wavelet
  - MPEG: DCT & time-difference



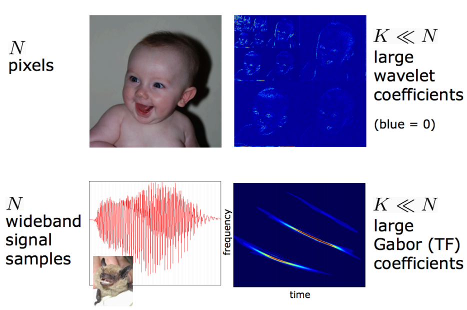
Sparse Transform



Sparse Transform



Sparsity



Signal Processing Trends

- Traditional DSP → sample first, ask questions later

Signal Processing Trends

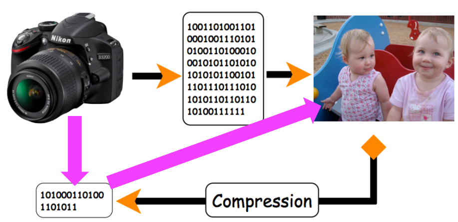
- Traditional DSP → sample first, ask questions later
- Explosion in sensor technology/ubiquity has caused two trends:
  - Physical capabilities of hardware are being stressed, increasing speed/resolution becoming expensive
    - gigahertz+ analog-to-digital conversion
    - accelerated MRI
    - industrial imaging
  - Deluge of data
    - camera arrays and networks, multi-view target databases, streaming video…

Signal Processing Trends

- Traditional DSP → sample first, ask questions later
- Explosion in sensor technology/ubiquity has caused two trends:
  - Physical capabilities of hardware are being stressed, increasing speed/resolution becoming expensive
    - gigahertz+ analog-to-digital conversion
    - accelerated MRI
    - industrial imaging
  - Deluge of data
    - camera arrays and networks, multi-view target databases, streaming video...
- Compressive Sensing → sample smarter, not faster

Compressive Sensing/Sampling

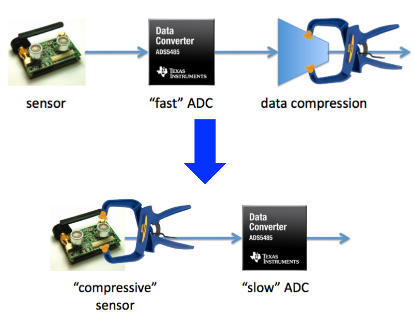
- Standard approach
  - First collect, then compress
    - Throw away unnecessary data



Compressive Sensing

- Shannon/Nyquist theorem is pessimistic
  - 2×bandwidth is the worst-case sampling rate — holds uniformly for any bandlimited data
  - sparsity/compressibility is irrelevant
  - Shannon sampling based on a linear model, compression based on a nonlinear model
- Compressive sensing
  - new sampling theory that leverages compressibility
  - key roles played by new uncertainty principles and randomness

Sensing to Data



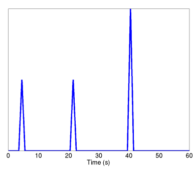
Compressive Sampling

- Sample at lower than the Nyquist rate and still accurately recover the signal, and in most cases *exactly* recover

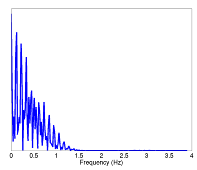
Compressive Sampling

- Sample at lower than the Nyquist rate and still accurately recover the signal, and in most cases *exactly* recover

Sparse signal in time



Frequency spectrum

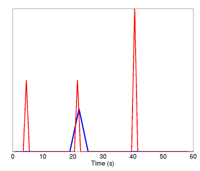


19

Compressive Sampling

- Sample at lower than the Nyquist rate and still accurately recover the signal, and in most cases *exactly* recover

Undersampled in time

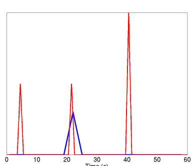


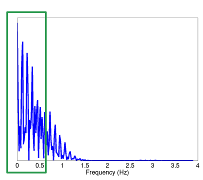
20

Compressive Sampling

- Sample at lower than the Nyquist rate and still accurately recover the signal, and in most cases *exactly* recover

Undersampled in time



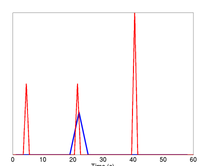


21

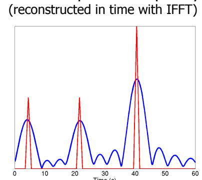
Compressive Sampling

- Sample at lower than the Nyquist rate and still accurately recover the signal, and in most cases *exactly* recover

Undersampled in time



Undersampled in frequency
(reconstructed in time with IFFT)

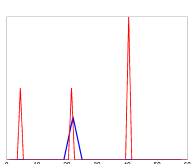


22

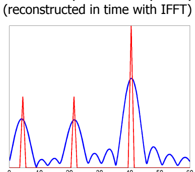
Compressive Sampling

- Sample at lower than the Nyquist rate and still accurately recover the signal, and in most cases *exactly* recover

Undersampled in time



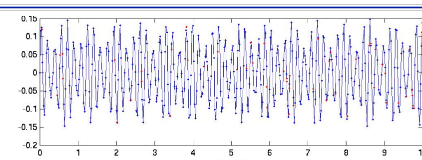
Undersampled in frequency
(reconstructed in time with IFFT)

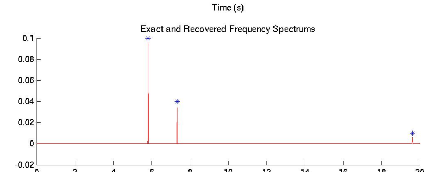


Requires sparsity and incoherent sampling

23

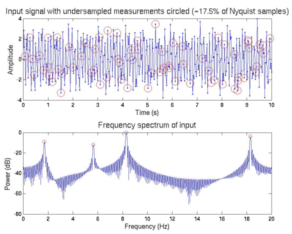
Compressive Sampling: Simple Example





24

Compressive Sampling

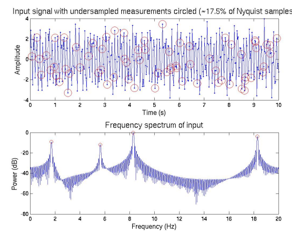


- Sense signal M times
- Recover with linear program

$$\min \sum_{\omega} |\hat{g}(\omega)| \quad \text{subject to} \quad g(t_m) = f(t_m), \quad m = 1, \ldots, M$$

Compressive Sampling

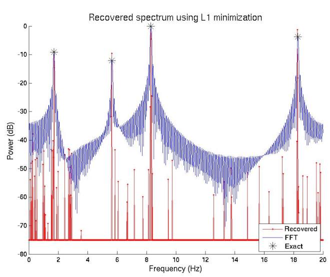
$$\hat{f}(\omega) = \sum_{i=1}^{K} \alpha_i \delta(\omega_i - \omega) \overset{\mathcal{F}}{\Leftrightarrow} f(t) = \sum_{i=1}^{K} \alpha_i e^{i\omega_i t}$$



- Sense signal M times
- Recover with linear program

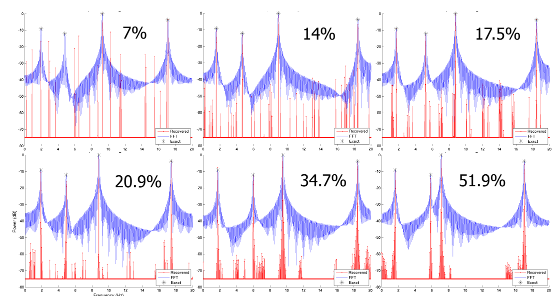
$$\min \sum_{\omega} |\hat{g}(\omega)| \quad \text{subject to} \quad g(t_m) = f(t_m), \quad m = 1, \ldots, M$$

Example: Sum of Sinusoids

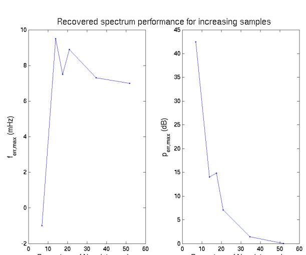


- Two relevant "knobs"
  - percentage of Nyquist samples as altered by adjusting number of samples, *M*
  - input signal duration, *T*
    - Data block size

Example: Increasing M

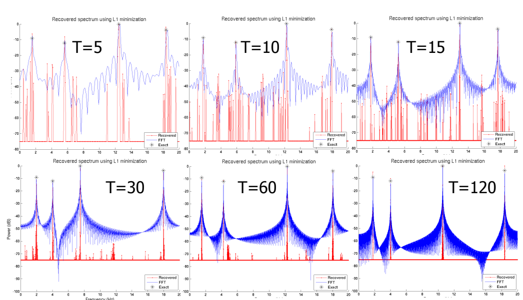


Example: Increasing M

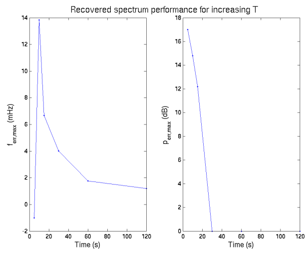


- $f_{err,max}$ **within 10 mHz**
- $p_{err,max}$ **decreasing**

Example: Increasing T



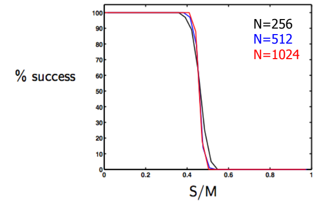
Example: Increasing T



- $f_{err,max}$ decreasing
- $p_{err,max}$ decreasing

Numerical Recovery Curves

- Sense S-sparse signal of length N randomly M times



  - In practice, perfect recovery occurs when $M \approx 2S$ for $N \approx 1000$

A Non-Linear Sampling Theorem

- Exact Recovery Theorem (Candès, R, Tao, 2004):
  - Select M sample locations $\{t_m\}$ "at random" with

$$M \geq \text{Const} \cdot S \log N$$

- Take time-domain samples (measurements)

$$y_m = x_0(t_m)$$

- Solve

$$\min_x \|\hat{x}\|_{\ell_1} \quad \text{subject to} \quad x(t_m) = y_m, \quad m = 1, \ldots, M$$

- Solution is exactly recovered signal with extremely high probability

A Non-Linear Sampling Theorem

- Exact Recovery Theorem (Candès, R, Tao, 2004):
  - Select M sample locations $\{t_m\}$ "at random" with

$$M \geq \text{Const} \cdot S \log N$$

- Take time-domain samples (measurements)

$$y_m = x_0(t_m)$$

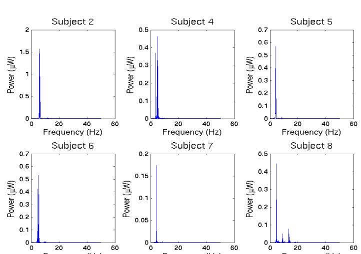
- Solve

$$\min_x \|\hat{x}\|_{\ell_1} \quad \text{subject to} \quad x(t_m) = y_m, \quad m = 1, \ldots, M$$

- Solution is exactly recovered signal with extremely high probability

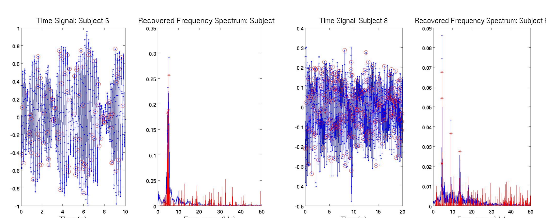
$M > C \cdot \mu^2(\Phi, \Psi) \cdot S \cdot \log N$

Biometric Example: Parkinson's Tremors

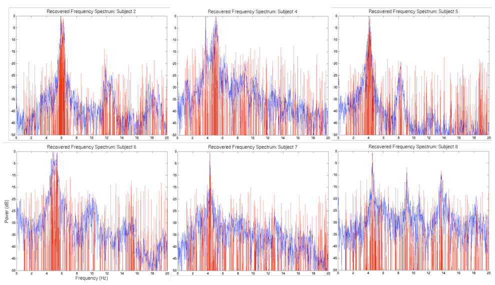


- 6 Subjects of real tremor data
  - collected using low intensity velocity-transducing laser recording aimed at reflective tape attached to the subjects' finger recording the finger velocity
  - All show Parkinson's tremor in the 4-6 Hz range.
  - Subject 8 shows activity at two higher frequencies
  - Subject 4 appears to have two tremors very close to each other in frequency

Compressive Sampling: Real Data



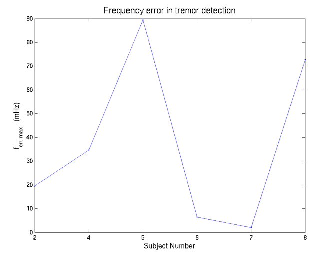
Biometric Example: Parkinson's Tremors



- **C=10.5, T=30**
  - 20% Nyquist required samples

37

Biometric Example: Parkinson's Tremors



- **Tremors detected within 100 mHz**
- **randomly sample 20% of the Nyquist required samples**

Requires post processing to randomly sample!

38

Implementing Compressive Sampling

- Devised a way to randomly sample 20% of the Nyquist required samples and still detect the tremor frequencies within 100mHz
  - Requires post processing to randomly sample!
- Implement hardware on chip to "choose" samples in real time
  - Only write to memory the "chosen" samples
    - Design random-like sequence generator
  - Only convert the "chosen" samples
    - Design low energy ADC

39

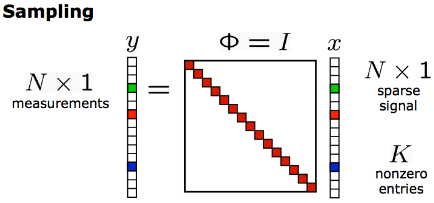
CS Theory

Why does is work?

Penn

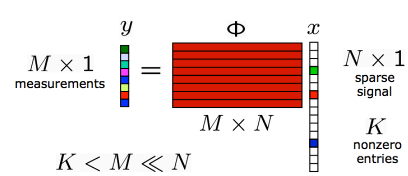
Sampling

- Signal x is K-*sparse* in basis/dictionary Ψ
  - WLOG assume sparse in space domain $\Psi = I$
- **Sampling**



Compressive Sampling

- When data is sparse/compressible, can directly acquire a ***condensed representation*** with no/little information loss through linear ***dimensionality reduction*** $y = \Phi x$

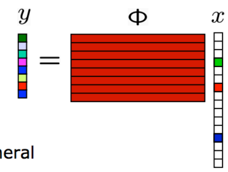


How Can It Work?

- Projection Φ
  not full rank…

  $M < N$

  … and so
  loses information in general



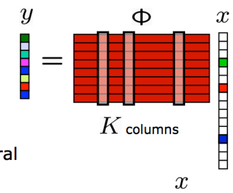
- Ex: Infinitely many x's map to the same y (null space)

How Can It Work?

- Projection Φ
  not full rank…

  $M < N$

  … and so
  loses information in general



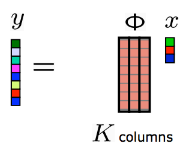
- But we are only interested in ***sparse*** vectors

How Can It Work?

- Projection Φ
  not full rank…

  $M < N$

  … and so
  loses information in general



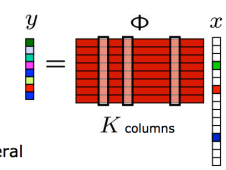
- But we are only interested in ***sparse*** vectors
- Φ is effectively *MxK*

How Can It Work?

- Projection Φ
  not full rank…

  $M < N$

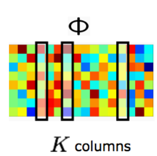
  … and so
  loses information in general



- But we are only interested in ***sparse*** vectors
- **Design** Φ so that each of its *MxK* submatrices are full rank (ideally close to orthobasis)
  - **Restricted Isometry Property (RIP)**

RIP

- Draw Φ at **random**
  - iid Gaussian
  - iid Bernoulli ± 1
  - …

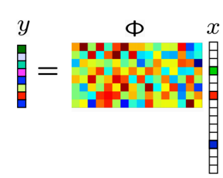


- Then Φ has the RIP with high probability provided

$$M = O(K \log(N/K)) \ll N$$

CS Signal Recovery

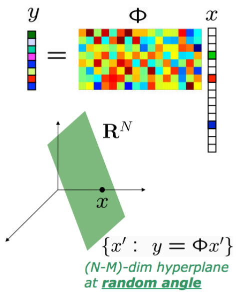
- **Goal**: Recover signal x from measurements y



- **Problem**: Random projection Φ not full rank (ill-posed inverse problem)
- **Solution**: Exploit the sparse/compressible ***geometry*** of acquired signal x

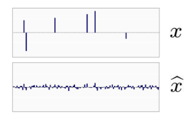
CS Signal Recovery

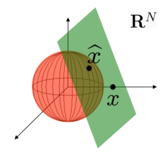
- Random projection Φ not full rank
- Recovery problem: given $y = \Phi x$ find x
- **Null space**
- Search in null space for the "best" x according to some criterion
  - ex: least squares



L_2 Signal Recovery

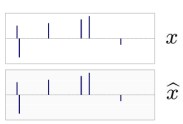
- Recovery: (ill-posed inverse problem) — given $y = \Phi x$, find x (sparse)
- Optimization: $\widehat{x} = \arg\min_{y=\Phi x} \|x\|_2$
- Closed-form solution: $\widehat{x} = (\Phi^T \Phi)^{-1} \Phi^T y$
- **Wrong answer!**

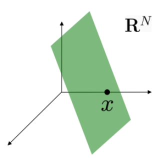




L_0 Signal Recovery

- Recovery: (ill-posed inverse problem) — given $y = \Phi x$, find x (sparse)
- Optimization: $\widehat{x} = \arg\min_{y=\Phi x} \|x\|_0$
- **Correct!** — *"find sparsest vector in translated nullspace"*

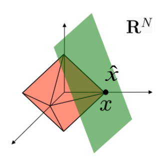




- But **NP-Complete** alg

L_1 Signal Recovery

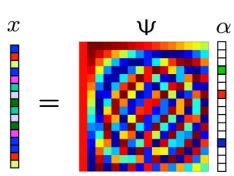
- Recovery: (ill-posed inverse problem) — given $y = \Phi x$, find x (sparse)
- Optimization: $\widehat{x} = \arg\min_{y=\Phi x} \|x\|_1$
- **Convexify** the ℓ_0 optimization
- **Correct!**
- **Polynomial time** alg (linear programming)
- Much recent alg progress
  - greedy, Bayesian approaches, ...



Universality

- Random measurements can be used for signals sparse in *any* basis

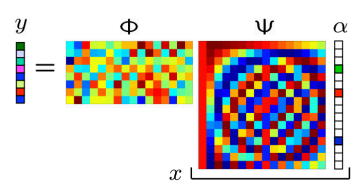
$$x = \Psi \alpha$$



Universality

- Random measurements can be used for signals sparse in *any* basis

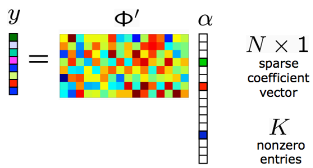
$$y = \Phi x = \Phi \Psi \alpha$$



Universality

- Random measurements can be used for signals sparse in *any* basis

$$y = \Phi x = \Phi \Psi \alpha = \Phi' \alpha$$



Reference Slide

Big Ideas

- Compressive Sampling
  - Integrated sensing/sampling, compression and processing
  - Based on sparsity and incoherency

Admin

- Final Project due – Apr 30th
  - TA advice – "The report takes time. Leave time for it."
  - No late accepted. Turn into Canvas on time.
- Last day of TA office hours – Apr 30th
  - Piazza still available
- Last day of Tania office hours – May 8th
- Final Exam Review Session – May 10th (time TBD)
  - Watch Piazza for details
- Final Exam – May 13th

Final Exam Admin

- Final – 5/13
  - Location Levine 101
  - Starts at exactly 3:00pm, ends at exactly 5:00pm (120 minutes)
  - Cumulative – covers entire course
    - Except data converters, noise shaping (lec 12), adaptive filters (lec 23), wavelet transform (lec 25), and compressive sampling (lec 26)
  - Closed book
    - Data/Equation sheet provided by me
    - 2 8.5x11 two-sided cheat sheets allowed
    - Calculators allowed, no smart phones
  - Old exams posted
  - TA Review session on 5/10, Time and Place TBD
    - Watch Piazza for details