Sub-Nyquist Sampling

Compressive Sampling
Roadmap

- What is it?
- Why is it important?
  - The Nyquist-Shannon sampling criterion
- How it works
- Practical example
- Applications
- Questions
What are we talking about here?

- Digitizing information
  - Representing real-world information in a digital format which can be stored and recalled with minimal effort

- Signal processing
  - Filtering
  - Smoothing
  - Spectrum Analysis
  - Feature Extraction
  - Pattern Recognition
General Signal Processing

Start here

Signal → Sampling → ADC → Compression → Transmission/Storage

On the other side

Retrival → Reconstruction → Signal
Sampling

Start with an analog signal, which can be described pretty well in some basis: sinusoids, wavelets, sinc, Dirac delta, etc...

http://www.falstad.com/fourier/

**Bandlimited Signal:**
This signal contains no components with frequencies higher than B

\[ X(f) = 0 \text{ for all } f > B \]
Digitize the Signal (Sampling)

We want to digitize the analog signal.

Sample it. Observe it at discrete points in time, and assign a value to the signal at that point.

Record the signal value and corresponding point in time.
Reconstruction

Without even talking about compression.

Rebuild the signal from the recorded data.

- For example, using the sinc basis
  - Multiply by the amplitude, \( x[n] \)
  - Phase shift it
    (move it along the x-axis to \( nT \))
  - Add them all together
Whittaker-Shannon Interpolation (WSI)

This process is called Whittaker-Shannon Interpolation

\[ x(t) = \sum_{n=-\infty}^{\infty} x[n] \cdot \text{sinc}\left(\frac{t - nT}{T}\right) \]
Problems with WSI

Idealized theory.

Cannot be implemented in practice. Requires infinite summation because all terms are considered to be important.

So we use square pulses, essentially a binary on or off.

Completely accurate reconstruction so long as the Shannon-Nyquist condition is met.
Sampling Rate

- Jumping ahead, better signal reconstruction with more points, right?
  - Less chance of missing out on high frequency components
  - Other constraints as well
- Cannot sample continuously
  - Sample at a fixed rate (or not so fixed)
  - Time between sampling is $T$
  - So we have

  $T \overset{\text{def}}{=} \frac{1}{f_s}$,

  where, $f_s$, is the sampling frequency.
Sampling Rate

How do we pick the sampling rate?

- Nyquist-Shannon Sampling Theorem
  - A **sufficient** condition for complete (accurate) signal reconstruction, sample at: \( f_s > 2B \)

- Example, human voice contains very small elements at >10kHz
  - Just sample at 20kHz and you're guaranteed accurate reconstruction

- But what if you don't...?
Aliasing

Adequate Sampling

Inadequate Sampling
Aliasing

- Examine the Fourier transform of the reconstructed signal

A properly sampled signal, the spectrum of the signal (blue), accompanied by its images (green)
Aliasing

- If $f_s$ is not large enough the images begin to overlap
  - These copies create **ambiguity**
  - We cannot discern a frequency component above $f_s/2$ from a lower frequency component, otherwise known as an **alias**...
Example of Ambiguity
Critical Frequency

Strict inequality \((f_s > 2B)\) as demonstrated by considering, Given this sinusoid:

\[
x(t) = \cos(2\pi Bt + \theta) = \cos(2\pi Bt)\cos(\theta) - \sin(2\pi Bt)\sin(\theta).
\]

With \(f_s = 2B\), \(T = 1/(2B)\), the samples are given as

\[
x(nT) = \cos(\pi n)\cos(\theta) - \sin(\pi n)\sin(\theta) = \cos(\pi n)\cos(\theta).
\]

Which are indistinguishable from

\[
x_A(t) = \cos(2\pi Bt)\cos(\theta).
\]
Critical Frequency

So various waveforms, with different phase shifts and amplitudes all begin to look exactly like the same thing.

This is why the criterion is strict.
Solving Aliasing

- Sample according to the Criterion
- Sample at very high rates
- Apply an anti-aliasing filter: a low pass filter
- Sample using different techniques?
Check our roadmap

Recall where we are in the signal processing process

Start here

Signal

Sampling

ADC

Compression

Transmission/Storage

On the other side

Retrival

Reconstruction

Signal
Compression

- Most systems use a form of compression
  - Reduce transmission or storage size
  - DFT, FFT, JPEG (DCT)
    - Basically, determine the values of the sinusoidal coefficients and frequencies
    - Store only the coefficients that are important
      - Use a cutoff
  - These coefficients are stored as a **sparse** matrix
Store only the value and location of non-zero components

Generate a sparse matrix

FFT
Roadmap Again

Now we have the signal and it's compressed.
How can we make this better?

- The question was asked
- Sample less?
  - No, this is ruled out by the criterion

- The answer is, combine steps
  - Sampling + Compression = Compressive Sampling
  - Sample such that your resulting information is already compressed
  - Bonus, sub-Nyquist sampling can be achieved!
Enter, the actual subject of this talk

Compressive Sampling = Compressed Sensing

- Based on the idea that a small number of non-adaptive measurements of a signal that is known to be compressible will provide enough information for complete reconstruction
  - Non-adaptive means the way you sample is not specifically tuned to the kind of signal you are measuring

- In other words: a signal that is sparse can be sampled in an incoherent way and the result will contain enough information to reconstruct the signal
Layman's Terms

Take a natural signal, measure it randomly, apply L1-magic, \textbf{bam!} you get your signal back.
Requirements for CS

- **Sparse Signal**
  - Recall sparsity = a coefficient matrix full of mostly zeros
  - These signals are pretty common, they include most in nature
    - Omit noise (but the method is robust!)

- **Incoherent measurements (test functions)**
  - Incoherent with respect to the signal being measured
  - This just means that the signals are not alike, at all, there is no way to relate one to the other, one is repetitious, like a sine wave, the other is noise
Why this works

- The basic idea
  - Each random measurement picks up information about all the coefficients
  - With enough random measurements we gain enough information about all the significant coefficients to reconstruct the complete transform
  - L1-minimization used to reconstruct the data
L1-norm Minimization

Of course, we will not be able to "fill in" the missing samples using sinc interpolation (or any other kind of linear method).
But since we know that the signal we wish to recover is "sparse", we can take a different approach to the problem. Given these 80 observed samples, the set of length-256 signals that have samples that match our observations is an affine subspace of dimension 256-80=176. From the candidate signals in this set, we choose the one whose DFT has minimum L1 norm; that is, the sum of the magnitudes of the Fourier transform is the smallest. In doing this, we are able to recover the signal exactly!
L1-norm Minimization

- When undersampling we will not be able to fill in all the gaps with just sinc functions
  - The problem is underdefined
    - Many unknowns, few equations
- But we know the signal is sparse
- So we can apply this knowledge towards an approach to a solution
- Using norm-minimization we seek a combination of series coefficients that best-fits the data we have, yet still minimizes the size of the sparse coefficient matrix
  - Assuming (for this explanation), a basis of sinusoids
L1-norm Minimization

- Each measurement reduces the number of unknowns by adding another "equation" to a system of unknown linear equations.
- If we took all the measurements we could, we would reduce the system to a point where everything is exactly determined.
  - But we don't/can't.
- If you consider the values you have already measured, and the remaining values which are undetermined, you can select values for those points to fill in the gaps.
  - Then, you create a DFT for this guess, and check the sum of the magnitudes of the Fourier coefficients.
  - Iterate and compare.
  - You want the smallest magnitude sum.
The number of measurements

● We only need to take about $m = S \log(N)$ measurements
  ○ For a camera of 1M pixels, $N = 10^6$
  ○ Say we know the image can be reduced to a sparse matrix of 10,000 coefficients, so $S = 10^4$
  ○ Therefore, using CS we only need $m = 60,000$ measurements
  ○ Incredible!

● This is the sub-Nyquist part
  ○ The number of measurements we have to take, is far less than what is defined by the sampling theorem
  ○ And, the measurement is non-adaptive
Single Pixel Camera

Low-cost, fast, sensitive optical detection

Image encoded by DMD and random basis

Compressed, encoded image data sent via RF for reconstruction

RNG

DMD

A/D

Xmtr

PD

DSP

Rcvr
Actual Single Pixel Camera
Single Pixel Camera

- Image is masked by a randomized filter
- Masked image light is focused to a single pixel
- The value is recorded
  - Some number of measurements are taken
- Recovery takes place on a computer running the l1-norm algorithm
Single Pixel Camera

- **Original**: 16384 Pixels
- **16384 Pixels**
  - **1600 Measurements** (10%)
- **16384 Pixels**
  - **3300 Measurements** (20%)
Applications

- Anything that costs a lot to measure
  - Systems where you can only take a few measurements
- Situations with excessive noise
- Make sure you have sparsity (some kind of structure in the thing you're measuring)
- Possible fields
  - MRI
  - Cheap sensor networks
  - LIDAR?
  - Wide range spectroscopy
Questions?
Sources

Nyquist

Compressive Sensing

http://en.wikipedia.org/wiki/Compressed_sensing

http://dsp.rice.edu/cs
Sidenote: Spectrum Images

- From the Poisson summation formula

\[
X_s(f) \overset{\text{def}}{=} \sum_{k=-\infty}^{\infty} X(f - kf_s) = T \sum_{n=-\infty}^{\infty} x(nT) e^{-i2\pi nf_s}.
\]

- Copies of \(X(f)\) occur at multiples of \(f_s\)
  - These copies create **ambiguity**
  - We cannot discern a frequency component above \(f_s/2\) from a lower frequency component, otherwise known as an **alias**...