Importance of Discrete Representations

Our goal is to develop **signal processing** tools to model interesting aspects of the world, to analyze the model, and to interpret the results.

The **increasing power** and **decreasing cost** of computation makes the use of computation increasingly attractive.

However, many important signals are naturally described with continuous functions, that must be **sampled** in order to be analyzed computationally.

Today: understand relations between **continuous** and **sampled** signals.
Sampling

How does sampling affect the information contained in a signal?

\[ f(t) \]

\[ f[n] = f(n\Delta) \]

\[ \Delta = \text{sampling interval} \]

Notation:
We will use parentheses to denote functions of continuous domain \((f(t))\) and square brackets to denote functions of discrete domain \((f[n])\).
Effects of Sampling Are Easily Heard

Sampling Music

\[ f_s = \frac{1}{\Delta} \]

- \( f_s = 44.1 \) kHz
- \( f_s = 22 \) kHz
- \( f_s = 11 \) kHz
- \( f_s = 5.5 \) kHz
- \( f_s = 2.8 \) kHz

J.S. Bach, Sonata No. 1 in G minor Mvmt. IV. Presto
Nathan Milstein, violin
Effects of Sampling are Easily Seen

Sampling Images
downsampling: $64 \times 48$
Characterizing Sampling

We would like to sample in a way that preserves information. However, information is generally lost in the sampling process. Example: samples provide no information about the intervening values.

Furthermore, information that is retained by sampling can be misleading. Example: samples can suggest patterns not contained in the original.

Samples (blue) of the original high-frequency signal (green) could just as easily have come from a much lower frequency signal (red).
Characterizing Sampling

Our goal is to understand sampling so that we can mitigate its effects on the information contained in the signals we process.
**Characterizing Sampling**

Begin by sampling sinusoids with different frequencies $\omega$.

Sample $x(t) = \cos(\omega t)$ every $\Delta$ seconds to obtain $x[n]$:

$$x[n] = x(n\Delta) = \cos(\omega n\Delta) = \cos \left( (\omega \Delta) n \right)$$

$$x(t) = \cos(\omega t)$$

$$\omega \Delta = 0.0 \times 2\pi$$

$$x[n] = \cos \left( (\omega \Delta) n \right) = \cos \left( 0.0 \times 2\pi n \right)$$

$n, t/\Delta$
Characterizing Sampling

Begin by sampling sinusoids with different frequencies $\omega$.

Sample $x(t) = \cos(\omega t)$ every $\Delta$ seconds to obtain $x[n]$:

$$x[n] = x(n\Delta) = \cos(\omega n\Delta) = \cos \left( (\omega \Delta) n \right)$$

$$x(t) = \cos(\omega t)$$

$$\omega \Delta = 0.7 \times 2\pi$$

$$x[n] = \cos \left( (\omega \Delta) n \right) = \cos \left( 0.7 \times 2\pi n \right)$$
Characterizing Sampling

Begin by sampling sinusoids with different frequencies $\omega$.

Sample $x(t) = \cos(\omega t)$ every $\Delta$ seconds to obtain $x[n]$:

$$x[n] = x(n\Delta) = \cos(\omega n\Delta) = \cos \left( (\omega \Delta) n \right)$$

$$x(t) = \cos(\omega t)$$

$$\omega \Delta = 0.7 \times 2\pi$$

$$x[n] = \cos \left( (\omega \Delta) n \right) = \cos \left( 0.7 \times 2\pi n \right)$$

$$\omega \Delta \rightarrow 2\pi - \omega \Delta = 0.3 \times 2\pi$$

$$x[n] = \cos \left( (2\pi - \omega \Delta) n \right) = \cos \left( (\omega \Delta) n \right)$$

$$x(t) = \cos \left( \left( \frac{2\pi}{\Delta} - \omega \right) t \right) \quad \omega \rightarrow \frac{2\pi}{\Delta} - \omega$$
Aliasing

Sampling continuous-time signals that have different frequencies can also generate the same sequence of samples. For example, the same sequence of samples results if $\omega_2 \Delta = \omega_1 \Delta \pm 2\pi k$ for any integer value of $k$.

$$x[n] = \cos((\omega_2 \Delta) n) = \cos((\omega_1 \Delta \pm 2\pi k) n) = \cos((\omega_1 \Delta) n)$$

Each point on the lines above show a pair of frequencies ($\omega_1$ and $\omega_2$) that generate the same sequence of samples: $x[n] = \cos(\omega_1 \Delta n) = \cos(\omega_2 \Delta n)$. 
Aliasing

Sampling continuous-time signals that have different frequencies can also generate the same sequence of samples. As a second example, the same sequence of samples results if \( \omega_2 \Delta = 2\pi k - \omega_1 \Delta \) for any integer value of \( k \).

\[
x[n] = \cos((\omega_2 \Delta)n) = \cos((2\pi k - \omega_1 \Delta)n) = \cos((-\omega_1 \Delta)n) = \cos((\omega_1 \Delta)n)
\]

Each point on the lines above show a pair of frequencies (\( \omega_1 \) and \( \omega_2 \)) that generate the same sequence of samples: \( x[n] = \cos(\omega_1 \Delta n) = \cos(\omega_2 \Delta n) \).
Aliasing

Many input frequencies $\omega_1$ generate the same output sequence of samples. For example, the same samples would result if the input frequency $\omega_1$ times $\Delta$ were $0.4\pi$ or $1.6\pi$ or $2.4\pi$ or ... Therefore, it’s impossible to determine what frequency produced an output at frequency $0.4\pi$.

Since multiple frequencies $\omega_1$ generate the same discrete samples, we say that these frequencies are aliases of each other.
Anti-Aliasing

We can prevent aliasing by removing input frequencies $\omega_1 \Delta > \pi$ and disregarding output frequencies $\omega_2 \Delta > \pi$.

We call this low-frequency range of frequencies the baseband.
Anti-Aliasing

The maximum frequency that can be represented using this scheme is called the Nyquist frequency: $\omega_m = \pi/\Delta$, which equals half the sampling rate $f_s$.

$$f_m = \frac{\omega_m}{2\pi} = \frac{\pi/\Delta}{2\pi} = \frac{1}{2\Delta} = \frac{f_s}{2}$$
Consider 3 CT signals:

\[ f_1(t) = \cos(4000t) \quad ; \quad f_2(t) = \cos(5000t) \quad ; \quad f_3(t) = \cos(6000t) \]

Each of these is sampled so that

\[ f_1[n] = f_1(n\Delta) \quad ; \quad f_2[n] = f_2(n\Delta) \quad ; \quad f_3[n] = f_3(n\Delta) \]

where \( \Delta = 0.001 \).

Which list goes from lowest to highest (baseband) frequency?

0. \( f_1[n] \quad f_2[n] \quad f_3[n] \)

1. \( f_1[n] \quad f_3[n] \quad f_2[n] \)

2. \( f_2[n] \quad f_1[n] \quad f_3[n] \)

3. \( f_2[n] \quad f_3[n] \quad f_1[n] \)

4. \( f_3[n] \quad f_1[n] \quad f_2[n] \)

5. \( f_3[n] \quad f_2[n] \quad f_1[n] \)
Anti-Aliasing Demonstration

Sampling Music.

- $f_s = 11$ kHz without anti-aliasing
- $f_s = 11$ kHz with anti-aliasing
- $f_s = 5.5$ kHz without anti-aliasing
- $f_s = 5.5$ kHz with anti-aliasing
- $f_s = 2.8$ kHz without anti-aliasing
- $f_s = 2.8$ kHz with anti-aliasing

J.S. Bach, Sonata No. 1 in G minor Mvmt. IV. Presto
Nathan Milstein, violin
Importance of Discrete Representations

Our goal is to develop signal processing tools to model interesting aspects of the world, to analyze the model, and to interpret the results.

The increasing power and decreasing cost of computation makes the use of computation increasingly attractive.

However, many important signals are naturally described with continuous functions, that must be sampled in order to be analyzed computationally.

Today: understand relations between continuous and sampled signals.
Quantization

The information content of a signal depends not only with sample rate but also with the number of bits used to represent each sample.

Bit rate = (\# bits/sample) \times (\# samples/sec)
We hear sounds that range in amplitude from 1,000,000 to 1.

How many bits are needed to represent this range?

1. 5 bits
2. 10 bits
3. 20 bits
4. 30 bits
5. 40 bits
Quantization Demonstration

Quantizing Music

- 16 bits/sample
- 6 bits/sample
- 5 bits/sample
- 4 bits/sample
- 3 bits/sample
- 2 bit/sample

J.S. Bach, Sonata No. 1 in G minor Mvmt. IV. Presto
Nathan Milstein, violin
Quantizing Images

Converting an image from a continuous representation to a discrete representation involves the same sort of issues. This image has $280 \times 280$ pixels, with brightness quantized to 8 bits.
Quantization Demonstration

Quantizing Music With and Without (Robert’s) Dither

- 4 bits/sample
- 4 bits/sample with dither
- 3 bits/sample
- 3 bits/sample with dither
- 2 bits/sample
- 2 bit/sample with dither

J.S. Bach, Sonata No. 1 in G minor Mvmt. IV. Presto
Nathan Milstein, violin
Summary

We are highly motivated to develop discrete representations of signals – especially when they represent signals that are naturally described with continuous functions.

Information is generally lost in such discretization processes.

Today we discussed two mechanisms that can alter the information contained in a signal: **aliasing** and **quantization**.

Next time, we will develop representations that are specialized for discrete-time signals.