Computational Assignment

Problem 1: Polynomial Approximations [10 Points]

In class we observed that the space of polynomials is a dense subset of the space of continuous functions $C([0,1])$ under the supremum norm. One class of polynomials which can be used to constructively provide an approximation is the family of Bernstein polynomials, defined as follows: Let for some $f \in C([0,1])$:

$$B_{n,f}(t) = \sum_{k=0}^{n} f(\frac{k}{n}) \binom{n}{k} t^k (1-t)^{n-k}$$

See our discussion in class, as well as Exercise 2.5.14 of the Lecture Notes, for a probability theoretic proof.

In this exercise, you are asked to write a Matlab (or an alternative program) function which admits a trigonometric function $f \in C([0,1])$, and n as its inputs and generates the Bernstein polynomial approximation of the signal.

You may take f to be for example $f = 2\sin(\frac{1}{4}\pi t) + 3\cos(\frac{1}{2}\pi t)$.

Given a trigonometric function of your choice, compute the Bernstein approximations of order n, where n can take three values and numerically verify that the difference

$$\sup_{t\in[0,1]} |f(t) - B_{n,f}(t)|,$$

decreases as n increases.

Problem 2: Fourier Transform [15 Points]

In this assignment, you are asked to generate (with the help of Matlab) the Fourier series expansion of the following function: $x \in l_2(\{-N, -(N-1), \cdots -1, 0, 1, \ldots, N-1, N\})$ where $x(n) = 1$ for $|n| \leq N_1$ and 0 elsewhere. That is

$$x(n) = 1_{\{|n|\leq N_1\}}, \quad -N \leq n \leq N$$

with $1_{\{.\}}$ denoting the indicator function.

You are asked to obtain a representation of the form:

$$\tilde{x}(n) = \frac{1}{\sqrt{2N+1}} \sum_{k=-N}^{N} a_k e^{ik(2\pi/(2N+1))n}$$

where

$$a_k = \frac{1}{\sqrt{2N+1}} \sum_{n=-N}^{N} x(n) e^{-ik(2\pi/(2N+1))n}.$$

By defining a FourSeries function, obtain the expansion coefficients with Matlab. Your function should take the signal as its input and generate the Fourier series coefficients as the output.

Now, let $(N = 2, N_1 = 1)$; $(N = 20, N_1 = 5)$; and $(N = 20, N_1 = 20)$ and obtain $\tilde{x}(n)$ above for $n \in \mathbb{Z}, n \in [-25, 25]$. Plot the Fourier series coefficients for x.

For $N = 20, N_1 = 1$, plot $\tilde{x}(n)$ for $n \in \mathbb{Z}, n \in [-25, 25]$. What is the relationship between $x(n)$ and $\tilde{x}(n)$?

Problem 3: Fourier Transform [15 Points]

a) Let $x \in L_2([0, 2]; \mathbb{C})$ be given by:

$$x(t) = 1_{\{t \leq 1\}} t + 1_{\{1 \leq t \leq 2\}}(1 - t)$$

Let $\hat{x}(\frac{k}{2})$ denote the Fourier series coefficient corresponding to $\{\frac{1}{\sqrt{2}} e^{i2\pi \frac{k}{2} t}, t \in [0, 2]\}$.

With Matlab, generate the plot of the signal

$$x_N(t) = \sum_{k=-N}^{N} \hat{x}(\frac{k}{2}) \frac{1}{\sqrt{2}} e^{i2\pi \frac{k}{2} t}, \quad t \in [0, 2]$$

for $N = 5, 10$ and 15. Here $\hat{x}(\frac{k}{P})$ are the Fourier Series expansion coefficients.

Observe that, the signal looks more and more like the original signal as N gets larger.

b) Prove that $\lim_{N \to \infty} \int (x(t) - x_N(t))^2 dt = 0$.

Hint: Use the properties of Hilbert spaces and the fact that $\{\frac{1}{\sqrt{P}} e^{i2\pi \frac{k}{P} t}\}$ forms a complete orthonormal sequence. You could invoke this result directly in your argument.

c) Does for a general $x \in L_2([0, 2]; \mathbb{C})$,

$$\sup_{t \in [0,2]} |x(t) - x_N(t)| \to 0,$$

as $N \to \infty$? Explain your argument.

Problem 4: Low-Pass Filters and Noise Removal [40 Points]

Suppose we have signal $x \in \mathcal{S}$ given by:

$$x(t) = 10 e^{-\frac{t^2}{2}}$$

Suppose this signal is perturbed by some high-frequency noise signal given by:

$$n(t) = 20 \cos(8\pi t) + 2 \sin(8\pi t)$$

Suppose we have a receiver which receives:

$$y(t) = x(t) + n(t),$$

The goal of the receiver is to reconstruct the original signal with low distortion.

(i) Plot x and y as a function of time for some positive time interval to see that y is a distorted version of x.

(ii) We wish to remove the noise term from the signal y: Observe that the signal $x \in \mathcal{S}$ does not have compact support. As such, we could work with CCFT ($\mathcal{F}_{CC}$) and exploit the fact that the signal's CCFT

bandwidth also decays to zero very fast: Recall that signals in $\mathcal{S}$ are transformed to signals in $\mathcal{S}$ under CCFT; and in fact, the CCFT of a Gaussian signal is also Gaussian (see Exercise 5.7.1).

One way to extract the signal is to sample y with some period sufficiently small to obtain a discrete time representation, using the Nyquist-Shannon Sampling Theorem (by approximating the bandwidth of the original signal by some finite value and justifying that the approximation error will be small via Parseval's Theorem). We should be able to reconstruct the continuous-time signal back from the samples if we can take the noise terms out.

Now that we have a discrete-time signal, we could apply it to a low-pass filter to take the noise terms out of the signal. Use an ideal low-pass filter for this step, and convert the signal back to the time domain.

(iii) Plot the resulting signal. Plot the resulting continuous time signal.

(v) We will now repeat steps (iii-iv) using a more practical filter. The ideal low-pass filter is not easy to implement and it requires a non-causal system.

There are a number of commonly used filtering methods used in practice. One common practical method is known as the Butterworth filter. This filter has a linear system description:

$$\hat{h}(f) = \frac{\sum_{k=0}^{N} b_k e^{-2\pi f k}}{\sum_{m=0}^{N} a_m e^{-2\pi f m}}.$$

Note that this is corresponds to the following discrete-time system (see Section 6.2):

$$\sum_{m=0}^{N} a_m y(n-m) = \sum_{k=0}^{N} b_k u(n-k)$$

It is easier to conceptually visualize such a filter in continuous time: An Nth order Butterworth filter with cut-off frequency f_c in continuous time has the following frequency response magnitude:

$$|\hat{h}(f)| = \frac{1}{1 + |\frac{f}{f_c}|^N}$$

Observe that, the filter is such that, for large N values, the filter resembles an ideal low pass filter.

In Matlab, the command, *butter(n,W)* returns a filter of order n - an nth order linear system - with the desired characteristics approximating a cut-off frequency W. That is, this function returns a linear system $[A, B, C, D]$ with description:

$$x(n+1) = Ax(n) + Bu(n)$$
$$y(n) = Cx(n) + Du(n).$$

Here, u is the input signal and y is the output.

Using Matlab, pick an appropriate W and an order n such that the resulting plot resembles the noise free signal. Plot the output.

Problem 5: Sampling [20 Points]

The operation of signal deletion (or decimation or down sampling) is defined by the pointwise mapping from $l_2(\{0, 1, 2, \ldots, N-1\}; \mathbb{R})$ to $l_2(\{0, 1, 2, \ldots, \frac{N}{M} - 1\}; \mathbb{R})$, N divisible by M:

$$y(n) = x(nM), \quad n \in \{0, 1, 2, \cdots, \frac{N}{M} - 1\}$$

so that the signal x is down-sampled by an integer factor M. For example if

$$x = \{\cdots, -2, 4, 3, -6, 5, -1, 8, \cdots\}$$

then the down-sampled sequence by a factor 2 are given by

$$y = \{\cdots, -2, 3, 5, 8, \cdots\}$$

a) Write a MATLAB function dnsample that has the form

$$y = dnsample(x, M)$$

to implement the above operation. Use the indexing mechanism of MATLAB with careful attention to the origin of the time axis $n = 0$.

b) Generate

$$x(n) = \cos(0.16\pi n), \quad -100 \leq n \leq 100.$$

Decimate x by a factor of 4 to generate y. Plot both $x(n)$ and $y(n)$ using the subplot command and comment on the results.

c) Repeat the above using

$$x(n) = \cos(0.8\pi n), \quad -100 \leq n \leq 100.$$

Qualitatively discuss the effect of sampling on the signals.

d) Using the FFT function in Matlab calculate and plot the Fourier transform of all the above signals. Compare the Fourier transform of the original and down-sampled signals in parts (b) and (c). In which case, can the original signal be recovered using the down-sampled version? Calculate the maximum frequency of the sinusoid such that one can recover the original signal using the down-sampled signal.

FFT is a very important algorithm to implement DTFT ($\mathcal{F}_{DD}$) in a computationally efficient fashion. The *fft* command in Matlab generates the transform.