

CMPE 422/Spring 08/Project 1: Z Transform, Least Squares, and sequence searching

Samir Chettri

January 21, 2008

Abstract

The Z transform has been taught in class. We are going to apply it to the discovery of signals. We will consider how the Z transform, and the least squares method can be used to design the response function of a filter to be able to detect specific signals. After considering an example that we can solve by hand, we will derive equations using the well known technique of “least-squares.” Students will then create MATLAB programs to solve the general case.

Students will hand in both their derivations and program that does the calculations. Upto fifty percent of the points for the project will be given for derivations and upto fifty percent for the MATLAB programs. The maximum grade a student can get on this project is one hundred points.

Keywords: Least Squares, impulse response, Z transform, MATLAB.

Contents

1	The problem	1
2	Input data	2
3	What to do?	3

1 The problem

Our textbook, on page 151, writes a transfer function as a ratio of two polynomials in z :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1z^{-1} + \dots + b_Nz^{-N}}{1 + a_1z^{-1} + \dots + a_Dz^{-D}}. \quad (1)$$

$X(z)$ is the z -transform of the input signal and $Y(z)$ is the z -transform of the output signal. We also assume that the system is causal. Hence the difference equation in the time domain (Ambardar, p. 62) is:

$$y[k] = -a_1y[k-1] - \dots - a_Dy[k-D] + b_0x[k] + \dots + b_Nx[k-N]. \quad (2)$$

Let us consider the following signal $x[n] = [0, 0, 0, \mathbf{2}, -5, 2, 0, 0, \dots]$, where the bold-face indicates the origin. We wish to design a filter $h[n]$ which takes this input signal and produces a spike ($[1, 0, 0, \dots]$) as its output. Working in the Z domain, we have $X(z) = 2 - 5z + 2z^2$ and $Y(z) = 1$. From this we can calculate

$$H(Z) = \frac{1}{(2-z)(1-2z)}.$$

After some manipulation we find that

$$H(Z) = \frac{1}{3} \sum_{n=0}^{\infty} (2^{n+1} - 2^{-(n+1)})z^n. \quad (3)$$

It should be evident from the solution for $H(z)$, that the impulse response is infinitely long and that it increases as n increases. Thus a filter such as this, would be impossible to implement on the computer. On the other hand, if the filter were decreasing, we could truncate it at some finite number of terms. An alternative to simple truncation exists which could easily be implemented as discussed below.

Let us start by assuming that all sequences we shall work with are causal and real. As discussed in previous paragraphs, the aim is to design a causal, finite-length filter so that a given input sequence, $[x[0], x[1], \dots, x[n], 0, 0, \dots]$ gives rise to a known output sequence (say a Kronecker delta function). Call this the desired output $d[r], r \in [0 \infty]$. Denote the filter by $[b[0], b[1], \dots, b[m], 0, 0, \dots]$. As indicated by our first example, this could lead to infinite, increasing filters. Therefore a compromise is in order - we might wish to approximate $d[r]$ with an output $[y[0], y[1], \dots, y[m+n], 0, 0, \dots]$. The issue then becomes, how closely does $y[\cdot]$ approximate $d[\cdot]$.

This is where “least-squares” comes in. Start with the error term E

$$E = \sum_{r=0}^{\infty} (d[r] - y[r])^2 = \sum_{r=0}^{m+n} \left(d[r] - \sum_{s=0}^m h[s]x[r-s] \right)^2 + \sum_{r=m+n+1}^{\infty} d[r]^2. \quad (4)$$

The student should minimize E with respect to $h[\cdot]$ to get a set of equations that are written as:

$$\sum_{s=0}^m h[s] \left(\sum_{r=0}^{m+n} x[r-s]x[r-j] \right) = \sum_{r=0}^{m+n} d[r]x[r-j]. \quad (5)$$

Let

$$\rho[j-s] = \sum_{r=0}^{m+n} x[r-s]x[r-j]$$

and

$$g[j] = \sum_{r=0}^{m+n} d[r]x[r-j].$$

This is now a linear system of equations that can be solved relatively easily for the unknowns $h[\cdot]$.

2 Input data

Given an input $x[\cdot] = [2, -1, 0, 0, \dots]$ that we wish to transform to $\delta[0]$. Use the technique of the previous section to design a finite-length filter $h[\cdot]$ of at most $m+1$ non-zero terms, i.e., $[h[0], h[1], \dots, h[m]]$. Tabulate your results as follows: From the basic theory, the student should write a MATLAB function, **LSQZTrans**

Table 1: Table of $h[\cdot]$ for different filter lengths, m . What are the values of $h[\cdot]$ tending toward?

$h[\cdot]$	$m = 0$	$m = 1$	$m = 2$	$m = 3$	$m = 10$	$m = 20$
$h[0]$						
$h[1]$						
$h[2]$						
$h[3]$						
.						
.						
$h[20]$						

that will accept the input signal $x[\cdot]$, the desired output $d[\cdot]$, the size of the filter denoted by m , and will produce the filter itself $h[\cdot]$. In order to obtain Table 1, the student will write a main program that will call the function as many times as needed to populate its entries. The main program will also produce the output sequences for $m = 0, 1, 2, 3, 10, 20$.

3 What to do?

This project has two parts. The first part deals with performing the derivations needed to understand the basic theory while the latter one deals with programming in MATLAB.

Derivations - 40%.

1. To show that you understand the least squares method, obtain equation (5) from equation (4). How did we get the second part of equation (4)? Write the basic least-squares equations (in matrix form). Comment on the form of the matrix.
2. **Due date:** xxx March 2008 in class. Your submissions should be neatly written (or typed). Make all assumptions clear to the reader and show all steps in derivations. Points will be taken off for missing steps, mistakes etc.

MATLAB - 60%

1. Write a MATLAB function `function [h] = LSQZTrans(m, d, x)` that solves the matrix problem in equation (5). Your file name should be the same as the function name, i.e., `LSQZTrans.m`.
2. Write a calling program `Project01Spring2008Main.m` that calls `LSQZTrans.m` with the appropriate inputs, receives the outputs and creates Table 1. The calling program will also produce the output sequences for $m = 0, 1, 2, 3, 10, 20$. Your main program will compare the $m = 20$ case by plotting it and simultaneously plotting the Kronecker delta function.
3. **Due date:** xxx May 2008 by email only.

Remember, points will be taken off for not following the format described above, regardless of whether you get the right results or not. Late submissions will not be accepted and will be given a grade of zero points.