# CMPE 422/Spring 08/Project 2: Statistics and the FFT

Samir Chettri

March 16, 2008

**Abstract**

Many text books discuss the DFT and FFT ( Digital Signal Processing by Ashok Ambardar; DSP by Oppenheim and Schafer etc. or go to the following URL `http://www.csee.umbc.edu/čhettri/ENE610/fourier_new.pdf` for some fairly comprehensive notes). The FFT has been named one of the top 10 algorithms of the $20^{th}$ century (go to `http://www.siam.org/news/news.php?id=637` for details). It would not have gotten this title if was purely of use in the DSP realm (wide enough though this is). The FFT belongs in this rare company because it has wide applications in the sciences and engineering. In this project we'll explore an application to the generation of probability distribution functions.

Students will first understand simple derivations (and fill in the blanks) followed by actual numerical calculations for PDF generation in MATLAB. They will hand in their program that does the calculations. The maximum grade a student can get on this project is one hundred points.

**Keywords:** FFT, probability and statistics, charactersitic function, normal distribution, chi–square distribution, PDF, MATLAB.

## Contents



## 1 The problem

Define the characteristic function, $\phi_X(s)$, of a probability density function, $f_X(x)$ as

$$\phi_X(s) = \int_{-\infty}^{\infty} e^{isx} f_X(x)dx \approx \int_{l}^{u} e^{isx} f_X(x)dx \approx \sum_{n=0}^{T-1} e^{isx_n} P_n. \tag{1}$$

To make the above discretization work, we need the PDF values at the sample points;

$$P_n = f_X(x_n)\Delta x, \tag{2}$$

the sample points starting with the lower bound;

$$x_n = l + n(\Delta x), \tag{3}$$

and the step length;

$$\Delta x = \frac{u-k}{T} \tag{4}$$

T represents the total number of points used in the summation.

The first approximation in equation (1) can be made increasingly exact if we assume $\lim_{x\to\infty} f_X(x) = \lim_{x\to-\infty} f_X(x) = 0$ and if $u$ and $l$ are sufficiently large. Similarly, the second approximation can be made arbitarily accurate by increading $T$ to a large enough value.

The problem then, is as follows: Assume $\phi_X(s)$ is available for any $s$ and from $\phi_X(s)$ we wish to obtain $P_n$ from which the PDF can be recovered. This is where DSP comes in. Equation (1) should remind you of a DFT (p. 339, Ambardar or my notes).

Divide the last equation in (1) by $e^{isl}$ and let $sn\Delta x = 2\pi nt/T$. This gives

$$\Phi_X(s)e^{-isl} \approx \sum_{n=0}^{T-1} e^{isn(\Delta x)} P_n = \sum_{n=0}^{T-1} e^{2\pi int/T} P[n] = g[t]. \tag{5}$$

Pay attention to the slight change of notation from $P_n$ to $P[n]$ - simply to maintain consistency with our book. Thus $g[t] = \mathcal{F}^{-1}(P[n])$, or in words, $g[t]$ is the inverse discrete fourier transform of $P[n]$, with $g[t] = [g[0]\ g[1]\ \ldots\ g[T-1]]$ and $P[n] = [P[0]\ P[1]\ \ldots\ P[T-1]]$.

Now, $s(\Delta x) = 2\pi t/T$, so the $s$ values can be chosen

$$s_t = \frac{2\pi t}{T(\Delta x)}, t = -\frac{T}{2}, -\frac{T}{2}+1, \ldots, \frac{T}{2}+1 \tag{6}$$

Therefore,

$$g[t] \approx \Phi_X(s_t)e^{-is_t l} \tag{7}$$

Basically, the above discussion shows how to get the probability density function from the characteristic function via the FFT. The student should be able to write a code to do this in the general case. This project deals with only two cases - the normal and chi–square distributions - though the method can be generalized to any case. Let us present a road–map for that case.

Start with with the moment generating function (MGF) of a probability density function (PDF):

$$M_X(r) = \int_{-infty}^{-\infty} f_X(x)e^{rx}dx. \tag{8}$$

$r$ is a complex variable that takes on values such that the integral converges, called the "region of convergence." Some will note that this looks like a Laplace transform of the (PDF) and that, formally at least, the CF can be obtained from the MGF with the substitution $\Phi_X(t) = M_X(it)$.

The student is asked to derive $M_X$ and $\Phi_X$ for the normal distribution with mean $\mu$ and variance $\sigma^2$, i.e.,

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

and for the chi–square density with $n$ degrees of freedom (DOF):

$$f_{\chi_n^2}(x) = \frac{1}{2^{n/2}\Gamma(n/2)} e^{-x/2} x^{(n/2)-1}.$$

Then write a program that uses the DFT/FFT to calculate the PDF of the normal distribution and the chi–square distribution. MATLAB has a statistics toolbox that can also calculate these two distributions. Compare your answers by plotting both your calculated PDF and the PDF of the normal and chi–squared distributions from MATLAB. Also, in the same graph, plot the error (the difference between your distribution and that from MATLAB).

## 2 What to do?

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

**Derivations and Theory - 30%**.

1. Derive the CF for the normal and chi–square distributions
2. **Due date**: **1** April 2008 in class or no later than 11:59 pm of the same day. Your submissions should be neatly written (or typed). Make all assumptions clear to the reader and show all steps in derivations.

**MATLAB - 70%**

1. Write a program that uses the DFT/FFT to calculate the PDF of the normal distribution and the chi–square distribution. MATLAB has a statistics toolbox that can also calculate these two distributions. Compare your answers by plotting both your calculated PDF and the PDF of the normal and chi–squared distributions from MATLAB. Also, in the same graph, plot the error (the difference between your distribution and that from MATLAB). Call your program `FFTForNormalAndChi.m`.
2. **Due date**: 15 April 2008 by email only.

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