TEMPLE UNIVERSITY COLLEGE OF ENGINEERING | ECE 3522 | SPRING 2015

Computer Assignment 2: Regression and Histograms

Tyler Berezowsky

February 1, 2015

1 Problem Statement

This computer assignment consisted of two tasks focused around “basic prediction functions”. The first task was to overlay the global mean and a linear regression on the closing Google stock prices acquired through the frame and windowing exercise of CA: 01. The second task was to develop a probability mass function (pmf) or histogram, and the cumulative distribution function for the audio signal also from the previous computer assignment.

2 Approach and Results

Google Stock Prices:
As stated in the problem statement, the stock prices were previously filtered via the frame and window technique. The frame and window metrics were 1 and 7 respectively. The global mean was calculated through the `mean()` command. The linear regression was calculated via the command `fitlm`. The resulting computations can be seen in figure 2.1.
The plot depicts the original signal (blue), the mean (orange): a single term approximation, and the linear regression (yellow): a two term approximation. The mean is constant for all time, while the linear regression attempts to crudely track the signal with time.

Audio Signal:
Generation of the histogram / probability mass function and cumulative mass function was also preformed all with a built-in MATLAB commands, `histogram`. Two histogram plots were generated. The first with a bin size of 500, and the second with the requested bin size of 10. Plots of the histograms along with their corresponding cumulative distribution function can be seen in figures 2.2 and 2.3.

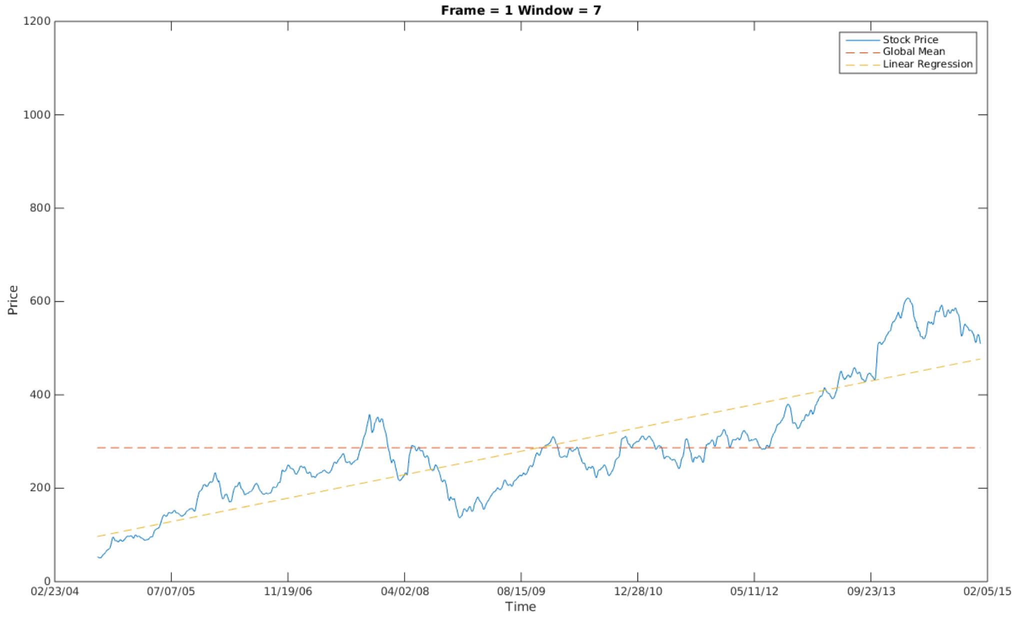

Figure 2.1: Plot of Google stock price frame and windowed, overlaid with the global mean and a linear regression.

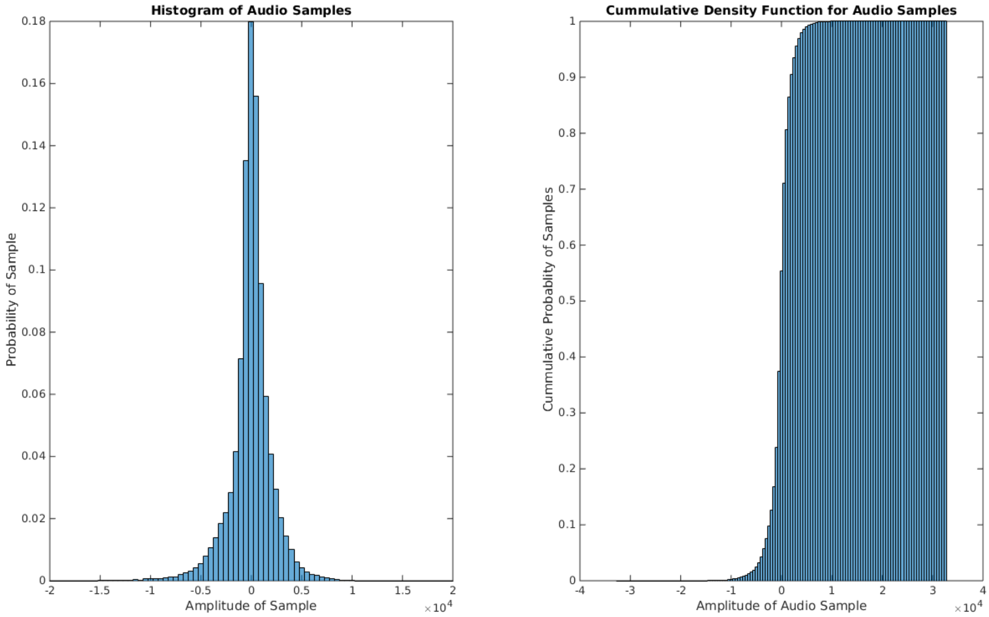

Figure 2.2: Histogram plot emulating the probability mass distribution with a bin width of 500 (left). Cumulative distribution function for the pmd (right).

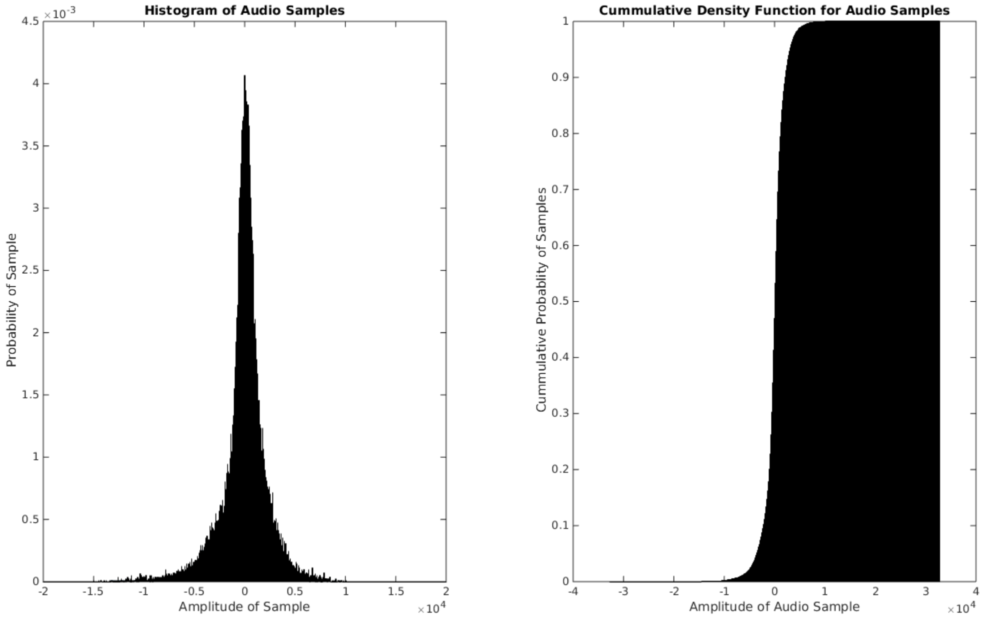

Figure 2.3: Histogram plot emulating the probability mass distribution with a bin width of 10 (left). Cumulative density function for the pmd (right).

The following relationships can be noted from the figures. The bin width of a histogram plot is proportional to the probability of bin. This can be seen through the scaling of the two figures. The bin width also determines the readability of the chart for large data sets. If the bin width is too small, as in figure 2.3, it becomes unreadable. The histograms are not symmetrical with a tendency for negative amplitudes. The distribution resembles a bell curve, but as it lacks symmetry and the decreasing rate of increase near the mean or max extrema, it cannot be classified as Gaussian. In physical correlation, the amplitude values closest to zero, have the highest probabilities. This is supported by the zero crossing nature of AC signals and the near-zero mean found in CA: 01.

3 MATLAB Code

As evident in the results section, the tasks were completed with built-in MATLAB commands primarily from the statistics toolbox or with code from the previous computer assignment. The only interesting development was the use of the `histogram()` command which generates a histogram or cumulative distribution function based on the arguments given.

Listing 1: Code for CA_02.m

```
%% Import Google Stock Prices
% Import the excel spreadsheet into an array

clear; clc; clear all;

filename = 'google_v00.xlsx'
[data, header, raw] = xlsread(filename)

% ease data manipulation with names
High = data(:,3);
Low = data(:,4);
Close = data(:,5);
Open = data(:,2);
```

```
Dates = x2mdate(data(:,1), 1);


%% Frame and Window Stock Data
% Windows which exceed frames will be trunkcated instead of zero-stuffed.
% This will "hopefully" prevent skewing the mean and variance.
% Actually no. MATLAB will not permit irregular array sizes to be
% concatenated. There maybe a work around, but I believe Python would be a
% better environment. Frames will be disregarded.
% Okay, well to plot ... how do I plot with a time-series.
% The time values corresponding to the window values selected where
% averaged to reflect the data point calculated from the window.

% N (Window Size) = 7, 30
% M (Frame Size) = 1, 7, 14, 30

clear windows
clear windowDates

N = [7];
M = [1];
googleFW = table()
fwMean = []
plotIndex = 1;

% Raster Through Windows and Frames Dimensions
%
for x = 1:length(N)
    for y = 1: length(M)

        sigLength = length(data);

        frameSize = M(y);
        windowSize = N(x);

        % initialize arrays for windows or dates
        windows = []
        windowDates = []

        for z = 1:frameSize:sigLength
          % calculate the frame center, and then the right and left window indexes
          frameCenter = floor( z + frameSize/2 ) ;
          windowLeft = floor( (frameCenter - 1) - 0.5*windowSize );
          windowRight = windowLeft + windowSize - 1;

          % insure the window never exceeds signal
          if (windowLeft >= 1) && (windowRight <= sigLength)
            windows = [windows; data(windowLeft : windowRight, 5)'];
            windowDates = [windowDates; Dates(windowLeft: windowRight)'];
          end
        end

        centerDates = mean(windowDates, 2);

        windowsMean = mean(windows, 2);
        windowsVariance = var(windows, 0, 2);
    end
end

%% Calculate Global Mean and Linear Regression Vector

globalMean = mean(windowsMean)
meanVector = linspace(globalMean, globalMean, length(windowsMean))'
```

```
linMod = fitlm(centerDates, windowsMean);


%% Plot Mean, Global Mean, and Linear Regression

figure();
plot(centerDates, windowsMean, centerDates, meanVector, '--', ...
    centerDates, linMod.Fitted, '--')
        ylim([0 1200]);
        datetick('x',2 ,'keeplimits', 'keepticks');
        xlabel('Time')
        ylabel('Price')
        plotIndex = plotIndex + 1
        titleStr = sprintf('Frame = %d Window = %d',...
            frameSize, windowSize);
        title(titleStr);
        legend('Stock Price', 'Global Mean', 'Linear Regression')

%% Import Audio File
% Import the .raw file into an array

filename = 'rec_01_speech.raw';
file = fopen(filename, 'r');
audio = fread(file, inf,'short');


 %% Calculate Histogram and Cummulative Distribution
BINWIDTH = 1000

figure()
subplot(121);
h = histogram(audio, 'BinWidth', BINWIDTH, 'Normalization', 'probability', ...
    'BinLimits', [-32767 32767])
ylabel('Probability of Sample');
xlabel('Amplitude of Sample');
title('Histogram of Audio Samples');
xlim([-2e4 2e4])

subplot(122);
cdf = histogram(audio, 'BinWidth', BINWIDTH, 'Normalization', 'cdf', ...
    'BinLimits', [-32767 32767])
ylabel('Cummulative Probablity of Samples');
xlabel('Amplitude of Audio Sample');
title('Cummulative Density Function for Audio Samples');
```

4 Conclusions

The mean as a rudimentary signal approximation is not very effective. Consisting of only a constant, it does not vary with time, and therefore lacks a non-zero first derivative delivering no rate information. The linear approximation includes time as a factor, $y = mx + b$, but delivers only the most primitive approximation ignoring any local extrema in a signal, and local extrema in stock prices is the gold of the craft. In addition, if the signal's mean was zero, as in the audio signal, a linear approximation would provide no predictive information. This type of analysis is only for signals with relatively constant trends.

The carnality of a data set will determine the extent of resolution required for a histogram plots useability. The number of bins in a histogram chart is determined by the following factors: the number of possibilities in the data set and the number of possibilities each bin is set to represent or the bin width. If the cardinality of the data set is large as with an audio signal and the bin size is small, the width of each bin will become indistinguishable. Increasing the bin width will decrease the resolution for each bin, but allows the bins to become discernible. This effect is displayed in figures 2.2 and 2.3. Also illustrated by the two example plots is the inverse relationship between the number of bins and the probability of a specific bin. As the number of bins for a histogram chart increases, the number of possibilities each bin represents decreases thus the probability for that bin decreases.