AGENDA

- Exponential Moving Averages
  - Why lag is important
  - How to compute the EMA constant to produce a given lag
- Higher order filters
  - Let your computer do a superior job of smoothing
- Essence of Predictive Filters
- Linear Kalman Filters
- Nonlinear Kalman Filters
- Theoretically Optimum Predictive Filters
- Zero Lag smoothing
Fundamental Concept of Predictive Filters

- In the trend mode price difference is directly related to time lag

- Procedure to generate a predictive line:
  - Take an EMA of price (better, a 3 Pole filter)
  - Take the difference (delta) between the price and its EMA
  - Form the predictor by adding delta to the price
    - equivalent to adding 2*delta to EMA
A Simple Predictive Trading System

Rules:
- Buy when Predictor crosses EMA from bottom to top
- Sell when Predictor crosses EMA from top to bottom

Usually produces too many whipsaws to be practical
Secrets of Predictive Filters

- All averages lag (and smooth)
- All differences lead (and are more noisy)
- The objective of filters is to eliminate the unwanted frequency components
- The range of trading frequencies makes a single filter approach impractical
- A better approach divides the market into two modes
  - Cycle Mode
  - Trend Mode
    - A Trend can be a piece of a longer cycle
Simple and Exponential Moving Averages

- EMA constant is usually related to the length of an SMA
  - The equation is $\alpha = 2 / (\text{Length} +1)$

- Only delay and amplitude smoothing are important
  - Delay is the most important criteria for traders
  - An EMA has superior rejection for a given delay
Relating Lag to the EMA Constant

- An EMA is calculated as:
  \[ g(z) = \alpha \cdot f(z) + (1 - \alpha) \cdot g(z - 1) \]
  where
  - \( g() \) is the output
  - \( f() \) is the input
  - \( z \) is the incrementing variable

- Assume the following for a trend mode
  - \( f() \) increments by 1 for each step of \( z \)
    - has a value of “i” on the “i th” day
  - \( k \) is the output lag
  \[
  i - k = \alpha \cdot i + (1 - \alpha) \cdot (i - k - 1) \\
  = \alpha \cdot i + (i - k) - 1 - \alpha \cdot i + \alpha \cdot (k + 1) \\
  = \alpha \cdot (k + 1) - 1
  \]
  Then \( k = \frac{1}{\alpha} - 1 \) OR \( \alpha = \frac{1}{k + 1} \)
## Relationship of Lag and EMA Constant

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$k$ (Lag)</th>
</tr>
</thead>
<tbody>
<tr>
<td>.5</td>
<td>1</td>
</tr>
<tr>
<td>.4</td>
<td>1.5</td>
</tr>
<tr>
<td>.3</td>
<td>2.33</td>
</tr>
<tr>
<td>.25</td>
<td>3</td>
</tr>
<tr>
<td>.2</td>
<td>4</td>
</tr>
<tr>
<td>.1</td>
<td>9</td>
</tr>
<tr>
<td>.05</td>
<td>19</td>
</tr>
</tbody>
</table>

- Small $\alpha$ cannot be used for short term analysis due to excessive lag
EMA is a Low Pass Filter

\[ g(z) = \alpha f(z) + (1 - \alpha)g(z - 1) \]

Use Z Transform notation (unit lag = 1/z)
\[ g = \alpha f + (1 - \alpha)\frac{g}{z} \]

Solving the algebra: \[ g = \frac{\alpha f z}{z - (1 - \alpha)} \]

- Output is related to input by a first order polynomial
- Called 1 Pole filter because response goes to infinity when \( z = 1 - \alpha \)
- Higher order polynomials produce better filtering
  - Second order: \( g = \frac{kf}{z^2 + az + b} \)
  - Third order: \( g = \frac{kf}{z^3 + az^2 + bz + c} \)
Higher Order Filters Give Better Filtering

Smoothing increases with filter order
High Frequencies (short cycles) are more sharply rejected
Higher Order Filter Design Equations

- Delay = $N \times P / \pi^2$ (N is order, P is cutoff period)

- Second Order Butterworth equations:
  
  $$a = \exp(-1.414\pi/P)$$
  
  $$b = 2a\cos(1.414\pi/P)$$
  
  $$g = b g[1] - a^2 g[2] + ((1 - b + a^2)/4)(f + 2f[1] + f[2])$$

- Third Order Butterworth equations:
  
  $$a = \exp(-\pi/P)$$
  
  $$b = 2a\cos(1.732\pi/P)$$
  
  $$c = \exp(-2\pi/P)$$
  
  $$g = (b + c)g[1] - (c + b^2)g[2] + c^2 g[3]$$
  

where $g$ is output, $f$ is input
14 Bar Cutoff
1, 2, & 3 Pole LowPass Filters

- Increased Lag is the penalty for increased smoothing
Higher Order filters give better fidelity for an equal amount of lag.
Linear Kalman Filters

- Originally used to predict ballistic trajectories
- Basic ideal is to correct the previous estimate using the current error to modify the estimate
- Procedure for a Linear Kalman Filter:
  - Previous estimate is the EMA
  - Estimate Lag error based on price change
  - Multiply the price rate of change by the lag-related constant

\[ g(z) = \alpha f(z) + (1 - \alpha) g(z - 1) + \gamma (f(z) - f(z - 1)) \]
Computing Kalman Coefficients

- As before, increment f() by 1 for each step of z
  \[ i - k = \alpha \cdot i + (1 - \alpha) \cdot (i - k - 1) + \gamma \cdot (i - (i - 1)) \]
  \[ = \alpha \cdot i + (i - k) - 1 - \alpha \cdot i + \alpha \cdot (k + 1) + \gamma \]
  \[ 0 = \alpha \cdot (k + 1) - 1 + \gamma \]
  \[ \gamma = 1 - \alpha \cdot (k + 1) \]

<table>
<thead>
<tr>
<th>K</th>
<th>\gamma</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (Lag)</td>
<td>1 - 2*\alpha</td>
</tr>
<tr>
<td>0</td>
<td>1 - \alpha</td>
</tr>
<tr>
<td>-1 (Lead)</td>
<td>1</td>
</tr>
<tr>
<td>-2</td>
<td>1 + \alpha</td>
</tr>
</tbody>
</table>

- Now lag is under control for any EMA constant
- Leading functions are too noisy to be useful
Linear Kalman Filter
1 Day Lag

Kalman $\gamma = .5$
EMA $\alpha = .25$
Nonlinear Kalman Filter

- Take EMA of price (better, a 3 Pole filter)
- Take the difference (delta) between Price and its EMA
- Take an EMA of delta (or a 3 Pole filter)
  - Smoothing will help reduce whipsaws
  - Ideally, smoothing introduces no major trend mode lag because delta is detrended
- Add the smoothed delta to EMA for a zero lag curve.
- Add 2*(smoothed delta) to EMA for a smoother predictive line
Zero Lag
Nonlinear Kalman Filter Example

Nonlinear Kalman Response

EMA $\alpha = .25$

GC__95Q.TTD

950728
Theoretically Optimum Predictive Filters

- Optimum predictive filters are solutions to the generalized Wiener-Hopf integral equation

- Optimum Predictive filters pertain only to the market cycle mode (Must use detrended waveforms)

- Two solutions are of interest to traders
  - Pure predictor (noise free case)
  - Predicting in the presence of noise
Pure Predictor

- **Calculations start by taking two 3 Pole Low Pass filters for smoothing**
  - Period1 = .707 * Dominant Cycle
  - Period2 = 1.414 * Dominant Cycle

- **Ratio of the two periods is 2:1**
  - The second filter has twice the lag of the first

- **Take the difference of the two filter outputs**
  - The difference detrends the information
  - The resultant is in phase with the cycle component of the price

- **A very smooth (noise-free) replica of the cycle component of the price is established. This is the BandPass Filter output.**
Sinewave “momentum” phase leads by 90 degrees

“Momentum” is similar to a calculus derivative.

\[
d \left( \sin(\omega t) \right) / dt = \omega \cdot \cos(\omega t)
\]

\[
1/\omega = P/(2*\pi) \text{ must be used as an amplitude normalizer.}
\]
Computing the Noise-Free Predictor

- Take the “momentum” of the BandPass Filter output (simple one day difference).
- Normalize amplitude by multiplying the “momentum” by \( P_0 / (2\pi) \)
- Produce 30 degree leading function
  - Multiply normalized “momentum” by \( .577 \) (\( \tan(30) = .577 \))
  - Add product to BandPass Filter output
- Reduce leading function amplitude
  - Multiply by \( .87 \) to normalize vector amplitude
  - Multiply again by \( .75 \) to reduce amplitude below BandPass amplitude.
    - Crossover entry signal always leads by 1/8th of a cycle
Noise-Free Predictor Vector Construction

Normalized

.577 * Normal

"Momentum"

30°

BandPass
The Complete BandPass Indicator

The BandPass Indicator is automatically tuned in:
- MESA for Windows
- 3D for Windows
BandPass Indicator Crossings Give Buy/Sell Signals
Optimum Predictive Filter in the Presence of Noise

- **Start with RSI or Stochastic Indicator**
  - Provides detrended waveform
  - Adjust length until the waveform resembles a sinewave

- **Technique is useful only when the waveform has a Poisson probability distribution**
  - The midpoint crossings must be relatively regular

- **Take an EMA of the RSI**
  - $\alpha = .25$ is nominally correct (gives a 3 day lag)

- **Subtract the EMA from the RSI to produce the predictor**
  - Remember the fundamental premise in constructing predictive filters?
RSI and Optimum Predictive Filter
Zero Lag Filters

- Zero Lag filters are constructed using cycle theory
- A phasor accurately depicts cyclic amplitude and phase characteristics
- Phasors ignore the cyclic rotation and examine only relative lead and lag relationships
Zero Lag Filter Construction

- Phasor A has a lag of DominantCycle/16
- Phasor B has twice the lag of Phasor A
- Subtract B from A by reversing B and adding
- Resultant is detrended leading angle Phasor C
- Vector add C to A
- Resultant is zero lag, non-detrended Phasor D
Zero Lag Filter Example
A Zero Lag Filter Application

- Take a 3 Pole zero lag filter of price highs
- Take a 3 Pole zero lag filter of price lows
- Calculate statistics of the high and low variations
  - Add 2 Standard Deviations to the Highs Zero Lag Filter
  - Subtract 2 Standard Deviations to the Lows Zero Lag Filter
- Resultant channels can be used as stop values for a stop-and-reverse system
- Remove the +/- Std Deviations near cycle turns
- SUMMIT for Windows uses this procedure
SUMMARY

What you have learned:

• How to relate filter lag to EMA constant
• How to compute Higher Order Butterworth Filters
• How to control lag using a Linear Kalman Filter
• How to compute a Nonlinear Kalman Filter
  – Possible start for a crossover system
• How to compute Optimum Predictive Filters for the cycle mode
  – Pure Predictor (Noise-Free, using higher order filters)
  – With RSI or Stochastics
• How to compute a zero lag filter