16. Adaptive Filters

16.15 The adaptive predictor may be considered as the linearly constrained minimization problem
\[ \hat{E} = E[\epsilon_n^2] = \min, \text{ subject to the constraint that the first element of } a = [1, a_1, \ldots, a_M]^T \text{ be unity.} \]
This constraint may be written compactly as \( u^T a = 1 \), where \( u = [1, 0, \ldots, 0]^T \).
Rederive the adaptation equations of Sec. 16.11 using the formalism and results of Problem 16.2.

16.16 Computer Experiment. A complex-valued version of the LMS adaptive predictor of Sec. 16.11 is defined by
\[
e_n = y_n + a_1(n)y_{n-1} + a_2(n)y_{n-2} + \cdots + a_M(n)y_{n-M}
\]
\[
a_m(n + 1) = a_m(n) - 2\mu a_m(n)y_{n-M}, \quad m = 1, 2, \ldots, M
\]
Let \( y_n \) consist of two complex sinusoids in zero-mean white noise
\[
y_n = A_1 e^{i \omega_1 n} + A_2 e^{i \varphi_2 n} + v_n
\]
where the frequencies and the SNRs are
\[
\omega_1 = 0.3\pi, \quad \omega_2 = 0.7\pi \text{ [radians/sample]}
\]
\[
10 \log_{10}|A_1|^2/\sigma_v^2| = 10 \log_{10}|A_2|^2/\sigma_v^2| = 20 \text{ dB}
\]
(a) Generate a realization of \( y_n \) (using a complex-valued \( v_n \)) and process it through an \( M \)th order LMS adaptive predictor using an adaptation constant \( \mu \). Experiment with several choices of \( M \) and \( \mu \). In each case, stop the algorithm after convergence has taken place and plot the AR spectrum \( S(\omega) = 1/|A(\omega)|^2 \) versus frequency \( \omega \). Discuss your results.
(b) Using the same realization of \( y_n \), iterate the adaptive Pisarenko algorithm defined by Eqs. (16.12.5) and (16.12.6). After convergence of the Pisarenko weights, plot the Pisarenko spectrum estimate \( S(\omega) = 1/|A(\omega)|^2 \) versus frequency \( \omega \).
(c) Repeat (a) and (b) when the SNR of the sinewaves is lowered to 0 dB. Compare the adaptive AR and Pisarenko methods.

16.17 Computer Experiment. Reproduce the results of Figs. 7.19 and 7.20.

16.18 Derive Eqs. (16.14.8) and (16.14.9) that describe the operation of the adaptive linear combiner in the decorrelated basis provided by the Gram-Schmidt preprocessor.

16.19 Computer Experiment. Reproduce the results of Fig. 16.12.2.

16.20 What is the exact computational cost of the conventional RLS algorithm listed in Sec. 16.15? Note that the inverse matrices \( P_0 \) and \( P_I \) are symmetric and thus only their lower-triangular parts need be updated.

16.21 Verify the solution (16.15.56) for the rank-one updating of the \( LU \) factors \( L_0 \) and \( L_1 \). Also verify that Eq. (16.15.58) is equivalent to (16.15.54).

16.22 Computer Experiment. Reproduce the results of Fig. 16.17.1. Carry out the same experiment (with the same input data) using the conventional RLS algorithm and compare with FAEST. Carry out both experiments with various values of \( \lambda \) and comment on the results.

16.23 Computer Experiment. Reproduce the results of Fig. 16.18.1.

Appendices

A Matrix Inversion Lemma

The matrix inversion lemma, also known as Woodbury’s identity, is useful in Kalman filtering and recursive least-squares problems. Consider the matrix relationship,
\[
R = A + UBV
\]
where
\[
A \in \mathbb{C}^{N \times N}, \quad U \in \mathbb{C}^{N \times M}, \quad B \in \mathbb{C}^{M \times M}, \quad V \in \mathbb{C}^{M \times N}
\]
and assume that \( A, B \) are both invertible and that \( M \leq N \). Then, the term \( UBV \) has rank \( M \), while \( R, A \) have rank \( N \). The matrix inversion lemma states that the inverse of \( R \) can be obtained from the inverses of \( A, B \) via the formula,
\[
R^{-1} = (A + UBV)^{-1} = A^{-1} - A^{-1}U[B^{-1} + VA^{-1}U]^{-1}VA^{-1}
\]

Proof: Multiply both sides of (A.1) by \( R^{-1} \) from the right, and then by \( A^{-1} \) from the left to obtain,
\[
A^{-1} = R^{-1} + A^{-1}UBV R^{-1}
\]
then, multiply both sides from the left by \( V \),
\[
VA^{-1} = VR^{-1} + VA^{-1}UBV R^{-1} \Rightarrow VA^{-1} = [I_M + VA^{-1}UB]VR^{-1}
\]
where \( I_M \) is the \( M \times M \) identity matrix, and solve for \( BVR^{-1} \),
\[
VA^{-1} = [B^{-1} + VA^{-1}U]BVR^{-1} \Rightarrow BVR^{-1} = [B^{-1} + VA^{-1}U]^{-1}VA^{-1}
\]
and substitute back into (A.3), after solving for \( R^{-1} \),
\[
R^{-1} = A^{-1} - A^{-1}UBVR^{-1} = A^{-1} - A^{-1}UBR^{-1} + VA^{-1}U - VA^{-1}
\]
Thus given \( A^{-1} \) and \( B^{-1} \), the inverse of the \( N \times N \) matrix \( R \) requires only the inverse of the smaller \( M \times M \) matrix, \( B^{-1} + VA^{-1}U \).
### B MATLAB Functions

% OSP Toolbox
% S. J. Orfanidis - 2018
%
% Local Polynomial Smoothing Filters
% ---------------------------------------------------------------
% binom - vector of binomial coefficients
% bfilter - Baxter-King bandpass filter
% cdec - classical decomposition method
% ccombfd - comb fractional-delay filter design
% compl - complement of an odd-length symmetric filter
% diffb - backward difference operator
% diffmat - difference convolution matrix
% diffpol - differentiate polynomial
% diffs - seasonal backward difference operator
% egc - ECG generator.
% ecgsim - ECG simulation
% filtdbl - filtering with double-sided FIR filter
% hahnbasis - Hahn orthogonal polynomials
% hahnc - coefficients of Hahn orthogonal polynomials
% hahnpol - Hahn orthogonal polynomial evaluation
% hahnpol - Hahn orthogonal polynomial basis
% hend - Henderson weighting function
% kmat - difference convolution matrix
% kraw - Krawtchouk binomial weighting function
% kwindows - Kaiser window for spectral analysis
% klag - Lagrange-interpolation fractional-delay filter
% npbasis - local polynomial basis
% npdiff - weighted local polynomial differentiation filters
% npfilt - local polynomial filtering - fast version
% npfilt2 - local polynomial filtering - slower version
% npinterp - local polynomial interpolation and differentiation filters
% npmat - local polynomial smoothing matrix
% npmism - weighted local polynomial filters for missing data
% npsm - local polynomial minimum-Rs smoothing filters
% npsm2 - local polynomial minimum-Rs smoothing filters (closed-form)
% npsmat - weighted local polynomial smoothing and differentiation filters
% nprev - minimum revision asymmetric filters
% npval - polynomial evaluation in factorial power series
% npfilt - robust local polynomial filtering
% sigav - signal averaging
% smadec - decomposition using seasonal moving-average filters
% smal - impulse responses of seasonal decomposition moving average filters
% smax - seasonal moving-average filtering matrix
% smake - seasonal moving average filter
% stirling - Stirling numbers of first or second kind, signed or unsigned
% swdec - seasonal Whittaker-Henderson decomposition
% trendma - trend-moving-average filter, 2XD if D is even, 1XD if D is odd
% upmat - upsample matrix of smoothing filters
% whkdec - Whittaker-Henderson-Kaiser seasonal decomposition
% x11decm - US Census X-11 decomposition method for seasonal adjustment
% x11filt - impulse responses of the US Census X-11 seasonal adjustment filters

% Local Linear Regression
% ---------------------------------------------------------------
% avobs - average repeated observations
% locband - bandwidth for local polynomial regression
% locgcv - local polynomial GCV and CV evaluation
% locgrid - uniform grid for local polynomial regression
% locpol - local polynomial regression
% locval - evaluation/interpolation of local polynomial regression
% loess - Cleveland's robust locally weighted scatterplot smoothing (loess)
% loess2 - Cleveland's robust locally weighted scatterplot smoothing (loess)

% Spline and Whittaker-Henderson Smoothing
% ---------------------------------------------------------------
% spllambda - find optimum lambda for spline smoothing using GCV
% splav - averaged repeated observations at spline knots
% splcoeff - spline coefficients
% splgcv - evaluate GCV(lambda)
% splmat - spline smoothing matrices Q,T
% splsm - spline smoothing using Reinsch's algorithm
% splsm2 - spline smoothing using Reinsch's algorithm - robust version
% whgcv - Whittaker-Henderson smoothing method
% whgen - generalized Whittaker-Henderson
% whimp - Whittaker-Henderson filter impulse response
% whsm - Whittaker-Henderson smoothing method
% whsm1 - Whittaker-Henderson smoothing method - L1 version

% Exponentially Weighted Averages
% ---------------------------------------------------------------
% binmat - binomial boost matrices for exponential smoothers
% ema - exponential moving average - exact version
% emaerr - calculate MAE, MSE, and MAPE for a range of lambda's
% emap - map equivalent lambdas between d=0 EMA and d=1 EMA
% emat - polynomial to cascaded transformation matrix
% holt - Holt's exponential smoothing
% holterr - calculate MAE, MSE, and MAPE for a range of lambda's
% mema - multiple exponential moving average
% stema - steady-state exponential moving average

% Linear Prediction & Wiener and Kalman Filtering Functions
% ---------------------------------------------------------------
% acf - sample auto-correlation function
% acmat - construct autocorrelation Toeplitz matrix from autocorrelation lags
% acsing - sinusoidal representation of singular autocorrelation matrices
% acmd - estimates dimension of signal subspace from AIC and MDL criteria
% argen - generate a zero-mean segment of an AR process
% bkwelev - backward Levinson recursion
% burg - Burg's method of linear prediction
% dir2nl - direct form to normalized lattice
% dpd - dynamic predictive deconvolution
% ddf - sample processing algorithms of direct-form Wiener filter
% dwf - direct-form Wiener filter using circular delay-line buffer
% dwwf - direct-form Wiener filtering of data
% dwwf - circular-buffer direct-form Wiener filtering of data
% faest - sample processing algorithm of adaptive lattice Wiener filter
% firw - FIR Wiener filter design
% flipv - flip a vector, column, row, or both for a matrix
% fwrlev - forward Levinson recursion
% glwf - sample processing algorithm of lattice Wiener filter
% kfilt - Kalman filtering
% ksmonth - Kalman smoothing
% latt - sample processing algorithm of analysis lattice filter
% lattfilt - lattice filtering of a data vector
% lattsync - sample processing algorithm of a single lattice section
% lattsynsect - sample processing algorithm of synthesis lattice filter
% lev - Levinson-Durbin recursion
% lms - sample processing algorithm of direct-form Wiener filter
% lpf - extract linear prediction filter from matrix L
% lpspec - compute LP spectrum of a prediction-error filter
% ltwf - sample processing algorithm of lattice Wiener filter
% ltwfilt - lattice Wiener filtering of data
% mgs - adaptive modified Gram-Schmidt
% mgsms - adaptive Gram-Schmidt using LMS
% mlnorm - minimum-norm noise subspace eigenvector
% music - MUSIC spectrum computation
% nlfilter - filtering in the normalized lattice form
% obmat - observability matrix for canonical or transposed realizations
% obmatc - observability matrix for continuous-time
% rlev - reverse of Levinson’s algorithm
% rls - RLS algorithm for adaptive linear combiner
% rls1 - sample processing algorithm of lattice Wiener filter
% rmusic - minimum-norm noise subspace eigenvector
% scatt - direct scattering problem
% schur2 - Schur algorithm for linear prediction
% schur2 - Schur algorithm for Cholesky factorization
% spike - spiking filter design
% yw - Yule-Walker method of linear prediction

% SVD, Subspace, and ARMA Modeling Functions
% arma2imp - ARMA Impulse response
% armacof - ARMA autocorrelation function
% armachol - ARMA covariance matrix Cholesky factorization
% armalfit - fitting an ARMA(p,q) model to covariance lags
% armainf - ARMA asymptotic Fisher information matrix
% armainf - ARMA modeling using the innovations method
% armaamf - Mayne-Firoozan ARMA modeling method
% armambyw - ARMA modeling by the modified Yule-Walker method
% armasim - simulate a zero-mean segment of a gaussian ARMA process
% armasim2 - simulate a zero-mean segment of a gaussian ARMA process
% beamwidth - beamwidth mapping from phi-space to phi-space
% cca - Canonical Correlation Analysis
% cacov - CCA applied to a covariance matrix
% cholgs - Cholesky factorization by Gram-Schmidt orthogonalization
% cholinnov - Cholesky factorization by innovations representation
% crb - calculate Cramer-Rao bounds for sinusoids in noise
% crb2 - calculate Cramer-Rao bounds for sinusoids in noise
% datamat - convolution data matrix of a signal vector
% datasig - extract data signal from a Toeplitz or Toeplitz/Hankel data matrix

% Wavelet Functions
% advance - circular time-advance (left-shift) of a vector
% casc - cascade algorithm for phi and psi wavelet functions
% circconv - circular convolution
% cnm - conjugate mirror of a filter
% convat - conjugate mirror of a filter
% dwtcell - cell array of sparse discrete wavelet transform matrices
% dwtdec - DWT decomposition into orthogonal multiresolution components
% dwtmat - discrete wavelet transform matrices
% dwtmat2 - discrete wavelet transform matrices
% dwtwrap - wrap a DWT matrix into a lower DWT matrix
% ebf - vector, column, row, or both for a matrix
% fwt - fast wavelet transform using convolution and downsampling
% fwtm - fast wavelet transform in matrix form
% fwtm - overall DWT orthogonal matrix
% ifwt - inverse fast wavelet transform using upsampling and convolution
% ifwtm - inverse fast wavelet transform in matrix form
% imw - inverse undecimated wavelet transform
% modwrap - wrap matrix-column-wise mod-N
% phinit - eigenvector initialization of phi
% plotdec - plot DWT/UWT decomposition or DWT/UWT coefficients
% up2 - upsample a vector by factor of two
% upw - upsample a vector by factor of 2^r
% uwt - undecimated wavelet transform
% uwtdc - UWT multiresolution decomposition
% uwtm - undecimated wavelet transform
% uwtmat - undecimated wavelet transform matrices
% uwtmat2 - undecimated wavelet transform matrices
% w2v - wavelet vector to wavelet matrix
% wcoeff - extract wavelet coefficients from DWT at given level
% wdenoise - Donoho & Johnstone's VisuShrink denoising procedure
% wduwt - wavelet denoising with UWT
% wthr - soft/hard level-dependent wavelet thresholding

% Technical Analysis Functions
% accdist - accumulation/distribution line
% atr - true range & average true range
% bbands - Bollinger bands
% bma - Butterworth moving average
% cci - commodity channel index
% chosc - Chaikin oscillator
% chvol - Chaikin volatility
% cmflow - Chaikin money flow
% cmo - Chande momentum oscillator
% delay - lag or delay or advance by d samples
% dema - steady-state double exponential moving average
% dirmov - directional movement system
% dmi - dynamic momentum index (DMI)
% donch - Donchian channels
% dpo - detrended price oscillator
% ema - exponential moving average
% fbands - fixed-envelope bands
% forosc - forecast oscillator
% gdema - generalized dema
% hma - Hull moving average
% ilrs - integrated linear regression slope indicator
% kbands - Keltner bands or channels
% lreg - linear regression, slope, and R-squared indicators
% mom - momentum and price rate of change
% ohlc - make Open-High-Low-Close bar chart
% ohlcyy - OHLC plot with other indicators on the same graph
% pbands - Projection Bands and Projection Oscillator
% pma - predictive moving average, linear fit
% pma2 - predictive moving average, polynomial order d=1,2
% pmaimp - predictive moving average impulse response
% pmaimp2 - predictive moving average impulse response, d=1,2
% pnv1 - positive and negative volume indices (PVI & NVI)
% prosc - price oscillator & MACD
% psar - Wilder's parabolic SAR
% r2crit - R-squared critical values
% rsi - relative strength index (RSI)
% sebands - standard-error bands
% sema - single exponential moving average
% shma - SMA-based Hull moving average
% sma - simple moving average
% stbands - STARC bands
% stdev - standard deviation index
% stoch - stochastic oscillator
% t3 - Tillson's T3 indicator, triple g DMA
% tcrit - critical values of Student's t-distribution
% tdistr - cumulative t-distribution
% tema - triple exponential moving average

% MATLAB Functions
% tma - triangular moving average
% trix - TRIX oscillator
% vema - variable-length exponential moving average
% vhfilt - Vertical Horizontal Filter
% wma - Wilder's exponential moving average
% wma - weighted or linear moving average
% y ylim - adjust left/right ylim & ticks

% Miscellaneous Functions
% canfilt - IIR filtering in canonical form using linear delay-line buffer
% ccan - IIR filtering in canonical form using circular delay-line buffer
% frespc - frequency response of a cascaded IIR filter at a frequency vector w
% loadfile - load data file ignoring any text lines
% taxis - define time axis
% up - upsample by a factor of L
% ustep - unit-step or rising unit-step function
% xaxis - set x-axis limits and tick marks
% yaxis - set y-axis limits and tick marks
% zmean - zero mean of each column of a data matrix (or row vector)
References

References for Chap. 1


Local Polynomial Smoothing Filters


REFERENCES


Henderson Filters


REFERENCES


Asymmetric End-Point Filters


Maximally Flat Filters


Local Polynomial Modeling and Loess


REFERENCES


Exponential Smoothing


REFERENCES

REFERENCES


\[L_1\] Trend Filtering


REFERENCES


Regularization


$L_1$ Regularization and Sparsity


REFERENCES


[590] MATLAB packages for solving the ℓ1 regularization and related problems:

https://www.mathworks.com/help/stats/las.html
ADMM http://stanford.edu/~boyd/admm.html
CVX http://cvxr.com/cvx/
FISTA http://ie.technion.ac.il/~becka/papers/rostls_package.zip
Homotopy http://www.ece.ucr.edu/~sasff/homotopy/
L1-MAGIC http://statweb.stanford.edu/~candes/l1magic/
LARS https://publib.biochem.iit.edu/xiaobw/licheng/code/lars/
https://sourceforge.net/projects/sparsetools/files/LARS/
NESTA http://statweb.stanford.edu/~candes/nesta/
REGTOOLS http://www.immt.dtu.dk/~pcha/Regtools/
SALSA http://cascai.iq.unesp.br/~mfonso/salsa.html
F. Bach http://web.stanford.edu/group/solt/software/Sal.png
Sparco http://www.cs.ubc.ca/labs/scl/sparco/
SpaRSA http://www1.cs.cornell.edu/~ravindral/sparselab.html
Sparselab http://sparselab.stanford.edu/
SPGL1 http://www.cs.ubc.ca/labs/scl/spgl1/
TwIST http://www1.cs.cornell.edu/~ravindral/SpaTIST/SpaTIST.htm
YALL1 http://yall1.blogs.rice.edu/

Comb Filters and Signal Averaging


REFERENCES


X-11 Seasonal Adjustment Method


REFERENCES


Model-Based Seasonal Adjustment


REFERENCES

Unobserved Components Models


Wavelets and Applications


REFERENCES


REFERENCES


http://www.dsp.rice.edu/software/RWT Rice Wavelet Toolbox.

http://paos.colorado.edu/research/wavelets, Torrance and Compo.

http://www.curvelet.org, Curvelets.

http://www.stats.bris.ac.uk/~wavethresh, Wavethresh in R.

http://taco.poly.edu/WaveletSoftware/, S. Cai and K. Li.

http://www2.isye.gatech.edu/~brani/wavelet.html, B. Vidakovic.


http://www.atmos.washington.edu/~wmts/, S. Cai and K. Li.


http://cam.mathlab.stthomas.edu/wavelets/packages.php, P. Van Fleet, see [685].

Wiener and Kalman Filtering

http://www-stat.stanford.edu/~wavelab/

REFERENCES


REFERENCES


REFERENCES


**References**

[1077] Proc. IEEE, 70 (9) (September 1982), Special Issue on Spectral Estimation.
REFERENCES


REFERENCES


REFERENCES


LCM and GSC Beamforming


Markowitz Portfolios


REFERENCES


SVD – Books


SVD – Applications


REFERENCES


SVD – Principal Component Analysis


REFERENCES
REFERENCES


---

**Index**

$L_1$ trend filtering, 358  
3-dB cutoff frequency, 110, 112  
accumulation-distribution, 307  
adaptive  
a posteriori RLS lattice, 901  
AR models, 86  
array processing, 878  
beamforming, 862  
channel equalizers, 868  
double-direct RLS lattice, 913  
echo cancelers, 869  
eigenvalue spread, 866  
eigenvector methods, 876, 907  
ext RLS lattice filters, 911  
FAEST algorithm, 910  
fast Kalman algorithm, 909  
FFT algorithm, 911  
gradient lattice filters, 881  
gradient projection method, 877  
Gram-Schmidt preprocessors, 889  
line enhancer, 872, 921  
linear combiner, 859  
linear predictor, 874  
noise canceler, 870  
Pisarenko’s method, 876  
sidelobe canceler, 861  
signal separator, 872  
spectrum analysis, 875, 878  
tracking of zeros, 879  
Wiener filters, 850, 862, 916  
accuracy of converged weights, 857  
conventional RLS, 904  
convergence speed, 865  
correlation canceler loop, 853  
FAEST algorithm, 910  
fast Kalman, 909  
fast RLS direct form, 907  
gradien lattice, 881  
gradien-descent method, 854  
linear prediction, 874  
LMS algorithm, 885  
Newton’s method, 866  
RLS lattice, 911  
stochastic approximation, 856  
adaptive GSC, 746  
airline data, 594

Akaikes final pr, prediction error (FPE), 678  
Akaikes information criterion (AIC), 710  
Algebraic Riccati equation, 103, 494  
analysis filter, 61, 535, 537  
analysis frame, 58  
analysis lattice filters, 537  
angle-of-arrival estimation, see superresolution  
array processing  
AR modeling of sunspot data, 88  
AR, ARMA, MA signal models, 63  
ARIMA modeling, 594  
Asymptotic statistics, 726  
eigenvector methods, 730  
linear predictors, 728  
reflection coefficients, 729  
sample covariance matrix, 21, 726, 730  
correlation  
complex-valued signals, 100  
computation by convolution, 50  
FFT computation, 96  
function, 44  
matrix, 99, 486, 512  
maximum entropy extension, 601  
method, 514  
of white noise, 54  
PARCOR coefficients, 520  
periodogram, 48  
power spectrum, 46  
reflection symmetry, 45  
sample, 48, 514  
sequence extension, 528  
singular, 529  
sinusoidal representation, 530, 694  
white noise, 45  
correlogram function of a filter, 52  
correlogram method, see Yule-Walker method, 561  
autoregressive  
models, 513  
normal equations, 513  
power spectrum, 514  
Backus-Gilbert parameter, 587  
backward prediction, 29  
bandpass signal extraction, 117  
bands, 294
INDEX

line enhancer, 872
linear estimation, 475
conditional mean, 10
correlation canceling, 8
decorrelated basis, 32
Gram-Schmidt orthogonalization, 13
jointly gaussian signals, 10
MAP, ML, LS, LMS criteria, 476
nonlinear estimation, 476
normal equations, 480
optimum estimator, 8
optimum filtering, 481
optimum prediction, 482
optimum smoothing, 481
orthogonal decomposition, 14
orthogonal projection, 8, 16
orthogonality equations, 480
signal separator, 8
unrestricted estimator, 10
Wiener filter, 484
linear phase property, 108
linear prediction
adaptivie, 874
analysis filter, 535
asymptotic statistics, 728
autocorrelation extension, 528
autoregression method, 261
backward, 27
backward Levinson recursion, 521
Burg’s method, 561
Cholesky factorization, 27, 542
covariance method, 561
decorrelated basis, 32
forward, 27
forward Levinson recursion, 519
gapped function, 512, 517
Gram-Schmidt orthogonalization, 542
lattice filters, 537
Levinson recursion, 514
LU factorization, 27
maximum entropy extension, 528, 601
minimum-phase property, 83, 539
normal equations, 513, 516
optimum filter, 510
orthogonal polynomials, 544
orthogonality of backward errors, 542
reflection coefficients, 518
reverse Levinson, 521
Schur algorithm, 547
signal classification, 566
signal modeling, 70, 509
split Schur algorithm, 551
stability test, 541
synthesis filter, 535
transfer function, 509
Yule-Walker method, 67, 561
linear regression, 275
linear regression indicator, 270
linear regression slope indicator, 270
linear trend FIR filters, 233
linearly-constrained Wiener filter, 735
LMS algorithm, 243, 855
local level filters, 270, 290
local polynomial fitting, 119
local polynomial interpolation, 206
local polynomial modeling, 197
local polynomial smoothing filters, 118
local slope filters, 270, 290
loess smoothing, 218
LPSM filters, 118
LU factorization, 18
MA and ARMA modeling, 812
MAP, ML, LS, LMS estimation criteria, 476
Market indicators:
accdist, accumulation/distribution line, 304
accdist, accumulation/distribution line, 304
acdist, linear trend autocorrelation sequence, 528
at, average true range, 299
bbands, Bollinger bands, 299
bma, Butterworth moving average, 287
ccl, commodity channel index, 304
chosc, Chaikin oscillator, 304
cchvol, Chaikin volatility, 304
cmfow, Chaikin money flow, 304
cmo, Chang momentum oscillator, 304
delay, d-delta delay, 292
dema, double EMA, 274
dirmov, directional movement system, 304
dmi, dynamic momentum index, 304
donch, Donchian channels, 290
dpo, detrended price oscillator, 304
ehma, exponential Hull moving average, 292
fbands, fixed-width bands, 299
forosc, forecast oscillator, 304
gdema, generalized DEMA, 292
hma, Hull moving average, 292
ils, integrated linear regression slope, 270
kbands, Keltner bands, 299
lreg, linear regression indicators, level, slope,
r-square, standard-errors, 278
mom, momentum, price rate of change, 304
ohlcY, OHLC chart with left/right y-axes, 278
ohlc, open-high-low-close bar chart, 278
phbands, projection bands & oscillator, 299
pma2, quadratic PMA, 272
pmaimp2, PMA impulse response, 272
pmaimp, PMA impulse response, 272
pmtracks, predictive moving average, 272
pnvdi, positive/negative volume indices, 304
proc, price oscillator and MACD, 304
psar, parabolic SAR, 302
r2crit, R-square critical values, 276
rcsd, relative strength index, 304
sebands, standard-error bands, 299
sema, single EMA, 274
shma, simple Hull moving average, 292
sma, simple moving average, 270
stbands, Starc bands, 299
stddev, length-N standard deviation, 285
stoch, stochastic, percent-K, percent-D, 304
til, Tilton’s 13 indicator, 292
tcrit, t-distribution critical values, 276
tdistr, cumulative t-distribution, 276
tema, triple EMA, 274
tma, triangular moving average, 270
tri, TRIX oscillator, 304
vema, variable-length EMA, 304
vhfill, Vertical horizontal filter, 304
wema, Wilder’s EMA, 285
yylim, adjust left/right y-axes limits, 278
yvlim, adjust left/right y-axes limits, 278
zema, zero-lag EMA, 292

MATLAB functions:
acext, autocorrelation sequence extension, 528
acf, sample autocorrelation function, 528
acmat, autocorrelation matrix from lags, 528
acsing, singular autocorrelation matrices, 528
advance, circular time-advance, 457
aicndil, AIC and MDL criteria, 528
argen, AR process generation, 528
arma2imp, ARMA impulse response, 818
armaacf, ARMA autocorrelation function, 818
armadef, ARMA modeling by innovations method, 818
armadi, ARMA modeling, 818
armainf, ARMA Fisher information matrix, 818
armainn, ARMA modeling by innovations methods, 818
armamp, ARMA by Mean-Fireouzan method, 818
armarmy, ARMA by Modified Yule-Walker, 818
armasm, ARMA process simulation, 818
armasm, ARMA process simulation, 818
avobs, average repeated observations, 218
binmat, binomial boost matrices, 263
binom, binomial coefficients, 170
bislev, backward Levinson recursion, 520
burg, Burg algorithm, 566
casc, cascade algorithm, 435
ccacov, CCA of covariance matrix, 843
cca, canonical correlation analysis, 844
chols, Cholesky factorization, 818
cholw, Cholesky factorization, 818
circov, circular convolution, 448
decov, classical decomposition method, 396
cmf, cumulative moving average, 270
comb, comb/ notch filter design, 378
compl, complementary filter, 400
convat, convolution a trous, 468
convmat, sparse convnits, 154
datamat, data matrix from signal, 810
datasig, signal from data matrix, 810
daub, Daubechies scaling filters, 432
diffmat, difference convolution matrix, 170

INDEX

dir2nl, direct form to normalized lattice, 528
dn2, downsample by factor of 2, 472
dn2, downsample by two, 457
dolphi, Dolphi-Cheryshev array, 742
dpd, dynamic predictive deconvolution, 580
dwfo2, direct-form Wiener filter, 528
dwfil2, direct-form Wiener filtering, 528
dwfilt, direct-form Wiener filtering, 528
dwf, direct-form Wiener filter, 528
dwtecol, cell array of DWT matrices, 453
dwtdct, DWT decomposition, 459
dwtnat, sparse DWT matrices, 450
eccsim, ECG simulation, 174
eema, EMA error criteria, 250
eemap, mapping equivalent lambdas, 249
etam, EMA basis transformation, 260
ema, exact EMA, 279
efaest, FAEST algorithm, 911
filt dbl, double-sided filtering, 157
fir, FIR Wiener filter, 555
filtv, flip a vector, column, row, or both, 528
frwlev, forward Levinson recursion, 520
fwtmat, DWT transformation matrix, 455
fwtn, fast DWT, 453
fwt, fast wavelet transform, 457
gfw, adaptive lattice Wiener filter, 888
gfwf, lattice filter, 528
hahnbas, Hahn polynomial basis, 182
hahnceoff, Hahn polynomial coefficients, 182
hahnpol, Hahn polynomial evaluation, 182
hend, Henderson weights, 174
holter, Holt error criteria, 265
holt, Holt’s exponential smoothing, 265
hpeq, high-order equalizer design, 384
ifwtn, inverse DWT, 453
ifwt, inverse fast wavelet transform, 457
imp2arma, impulse response to ARMA coefficient, 818
inuwt, inverse UWT in matrix form, 465
iuwt, inverse UWT, 467
kfall, Kalman filtering, 627
kssmooth, Breslow-Frazier smoothing, 656
kwindow, Kaiser window, 400
lattfilt, lattice filtering, 528
lattice, lattice realization, 537
lattsec, single lattice section, 528
lattsyn, synthesis lattice filter, 528
latt,分析 lattice filter, 528
lev, Levinson recursion, 520
lms, LMS algorithm, 863
loadfile, numerical data from file, 159
lochand, local bandwidth, 203
loccv, CV and GCV evaluation, 206
locgrid, local uniform grid, 267
locpol, local polynomial modeling, 202
local, interpolating local polynomials, 207
lowc, local weighting functions, 198
loess, loess smoothing, 219
Schur-Cohn stability test, 541
seasonal decomposition, 104
seasonal decomposition filters, 391
seasonal moving-average filters, 400
seasonal Whittaker-Henderson decomposition, 417
second-order statistics, 1
shift-invariance property, 44, 899, 908
sidelobe canceler, 861
signal averaging, 385
signal classification, 60
signal enhancement, 105
signal estimation, 476
signal extraction, 104
signal extraction, periodic, 368
signal models, see random signal models
signal separator, 872
signal subspace, 691, 699, 707
signal-to-noise ratio, 107
simulation of random vectors, 20
single, double, triple EMA, 252, 273
singular spectral analysis, SSA, 826
singular value decomposition, 763, 776
sinusoids in noise, 101
spectral analysis, 680
smoothing filters, 111, 112, 118
smoothing parameter selection, 247
smoothing splines, 315
snapshot vector, 21
SNR, 703
SNR, see signal-to-noise ratio
southern oscillation index, 846
sparse regularization, 793
sparse seasonal Whittaker-Henderson decomposition, 419
sparse Whittaker-Henderson methods, 358
spatial smoothing method, 723
spectral factorization, 82
Wiener filter, 487
spectrum estimation adaptive, 875, 878
AR estimates, 683
AR models, 514, 678
autocorrelation method, 514
classical Bartlett spectrum, 682
classical methods, 51
eigenvector methods, 689
ML estimates, Capon, 688
parametric models, 60
Pisarenko’s method, 689, 876
sinusoids, 680
windowed autocorrelation, 681
Yule-Walker method, 514
speech synthesis, 59, 566
spline filters, 329
spline, stochastic model, 331
spline, variational approach, 316
split Levinson algorithm, 532
split Schur algorithm, 551
stability and stationarity, 63
standard-error bands, 294
Starc bands, 294
stationarity, 45
steady-state EMA, 241
steepness descent, 253
steered array, 705
steering vector, 682, 697, 705
stochastic oscillator, 306
structured matrix approximations, 830
subspaces, bases, projections, 760
sunspot data, 88
sunspot numbers, 847
superresolution array processing, 694
adaptive, 878
Bartlett beamformer, 698
conventional beamformer, 695
LP spectrum estimate, 698
maximum likelihood method, 719
ML beamformer, 698
SVD and least-squares problems, 783
SVD and linear equations, 770
SVD and signal processing, 805
SVD signal enhancement, 825
synthesis filter, 58, 535
technical analysis in financial markets, 267
thriving, 254
Tikhonov regularization, 792
Tillson’s T3 indicator, 288
time constant, 113
time-series forecast indicator, 270
transient response in noise reduction filters, 108
TRIX oscillator, 309
Tukey’s twicing operation, 254
twicing, 254
twicing and zero-lag filters, 255
UL factorization, 94
unbiased estimator, 3
uncorrelated random variables, 14
uniform probability density, 2
unitarity of scattering matrix, 577
variable and adaptive bandwidth, 211
variable-length EMA, VEMA, 309
variance, 1
vector and matrix norms, 765
vector space of random variables, 14
vertical horizontal filter, VHF, 305
Vondrak filters, 341
wavetools
a trous operation, 442
analysis and synthesis filter banks, 443
analysis and synthesis with UWT, 464
decimated and undecimated filter banks, 463
denoising, 459
dilation equations, 430
discrete wavelet transform, 446
DWT in convolutional form, 456
DWT in matrix form, 448
fast DWT, 453
Haar & Daubechies scaling functions, 426
inverse DWT, 451
inverse UWT, 456
Mallat’s algorithm, 441
MATLAB functions, 472
multiresolution analysis, 425
multiresolution and filter banks, 441
multiresolution decomposition, 428, 458
orthogonal DWT transformation, 455
peroidized DWT, 450
refinement equations, 430
scaling and wavelet filters, 432, 436
symmetries, 433
UWT desymmetrization, 469
UWT matrices, 465
UWT multiresolution decomposition, 468
UWT, undecimated wavelet transform, 463
wavelet shrinkage method, 462
waves in layered media, 568
weighted local polynomial modeling, 197
weighted polynomial filters, 164
Welch method of spectrum estimation, 51
WEMA, Wilder’s EMA, 284
white noise, 45, 54
filtering of, 51
whitening filter, 61, 511
Whittaker-Henderson smoothing, 341
Wiener filter adaptive, 862
beamforming, 705
covariance factorization, 479
FIR filter, 481
gapped functions, 495
Kalman filter, 490
lattice realizations, 553
linear prediction, 514
mean-square error, 488
orthogonal basis, 553
prewhitening, 484
priori, 901
Savitzky-Golay smoothing filters, 118
scattering matrix, 570
Schur algorithm, 42, 547
Schur recursion, 553
INDEX
stationary, 484
transfer function, 488
unrealizable, 488
Wiener process, 66
Wold decomposition, 57

Yule-Walker method, 67, 514, 522, 561

zero tracking filters, 879
zero-lag EMA, 288
zero-lag filters, 253