Subject Index

A

ABORT statement
  exiting IML, 476
ABS statement
  absolute value, 476
ALL statement
  checking for nonzero elements, 476
ANY function
  checking for nonzero elements, 477
APPEND statement
  SAS data sets, 479
APPLY function, 480
ARMACOV call
  autocovariance sequence, 482
ARMALIK call
  log likelihood and residuals, 484
ARMASIM function
  simulating univariate ARMA series, 485

B

Bessel function
  finding nonzero roots and derivatives of, 577
BLOCK function
  forming block-diagonal matrices, 487
box-and-whisker plots, 822
BRANKS function
  computing bivariate ranks, 487
BTRAN function
  computing the block transpose, 488
BYTE function
  translating numbers to ordinal characters, 489

C

CALL statement
  calling a subroutine or function, 490
CHANGE call
  replacing text in an array, 491
CHAR function
  character representation of a numeric matrix, 491
Character Functionality, 454
CHOOSE function
  choosing and changing elements, 492
CLOSE statement
  closing SAS data sets, 493
CLOSEFILE statement
  closing a file, 494
CONCAT function
  performing elementwise string concatenation, 496
CONTENTS function
  obtaining the variables in SAS data sets, 497
Control Statements, 458
CONVEXIT function
  calculating convexity of non contingent "cash-flows", 498
CONVMOD function
  converting modules to character matrices, 499
correlation coefficient computation, 822
COVLAG function
  computing autocovariance estimates, 499
CREATE statement
  creating new SAS data sets, 500
CSHAPE function
  reshaping and repeating character values, 502
CUSUM function
  calculating cumulative sums, 504
CVEXHULL function
  finding a convex hull, 504

D

Dataset and File Functionality, 459
DATASETS function
  obtaining names of SAS data sets, 505
DELETE call
  deleting SAS data sets, 506
DELETE statement
  marking observations for deletion, 507
DESIGN function
  creating a design matrix, 509
DESIGNF function
  creating a full-rank design matrix, 509
DET function
  computing determinants of a square matrix, 510
DIAG function
  creating a diagonal matrix, 511
DISPLAY statement
  displaying fields in display windows, 512
DO and END statements
  grouping statements as a unit, 514
DO DATA statement
  repeating a loop until, 516
DO function
  producing an arithmetic series, 513
DO statement
  DATA clause, 516
  UNTIL clause, 517
  WHILE clause, 518
DO statement, iterative
  iteratively executing a DO group, 515

DO UNTIL statement
conditionally executing statements iteratively, 517
DO WHILE statement
conditionally executing statements iteratively, 518
DURATION function
calculating modified duration of non-contingent cash-flows, 519

E

ECHELON function
reducing a matrix to row-echelon normal form, 520
EDIT statement
opening a SAS data set for editing, 520
EIGEN call
computing eigenvalues and eigenvectors, 523
eigenvalue decomposition
compared with ODE call, 674
EIGVAL function
computing eigenvalues, 525
EIGVEC function
computing right eigenvectors, 526
END statement
ending a DO loop or DO statement, 526
EXECUTE call
executing SAS statements immediately, 527
EXP function
calculating the exponential, 527

F

FFT function
performing the finite Fourier transform, 528
FILE statement
opening or pointing to an external file, 529
FIND statement
finding observations, 530
FINISH statement
denoting the end of a module, 532
forward rates, 532
FREE statement
freeing matrix storage space, 533

G

GBLKVP call
defining a blanking viewport, 533
GBLKVPD call
deleting the blanking viewport, 534
GCLOSE call
closing the graphics segment, 535
GDELETE call
deleting a graphics segment, 535
GDRAW call
drawing a polyline, 536
GDRAWL call
drawing individual lines, 537
GENEIG call
generalized eigenproblems, 538
GGRID call
drawing a grid, 539
GINCLUDE call
including graphics segments, 540
GINV function
computing generalized inverses, 540
GOPEN call
opening graphics segments, 541
GOTO statement
jumping to a new statement, 542
GPIE call
drawing pie slices, 543
GPIEXY call
converting coordinates, 544
GPOINT call
ploting points, 545
GPOLY call
drawing and filling a polygon, 546
GPORT call
defining a viewport, 547
GPORTPOP call
popping viewports, 547
GPORTSTK call
stacking viewports, 548
Graphics and Window Functions, 459
GSCALE call
calculating round numbers for labeling axes, 548
GSCRIPT call
writing multiple text strings, 549
GSET call
setting attributes for graphics segments, 550
GSHOW cal
showing a graph, 551
GSORTH call
computing the Gram-Schmidt orthonormalization, 552
GSTART call
initializing the graphics system, 553
GSTOP call
deactivating the graphics system, 554
GSTRLEN call
finding the string length, 554
GTEXT and GVTEXT calls
placing text on a graph, 555
GWINDOW call
defining the data window, 556
GXAXIS and GYAXIS calls
drawing an axis, 557

H

HALF function
computing Cholesky decomposition, 558
HANKEL function
generating a Hankel matrix, 559
HDIR function
performing a horizontal direct product, 561
HERMITE function
reducing a matrix to Hermite normal form, 562
HOMOGEN function
solving homogeneous linear systems, 563

I

I function
  creating an identity matrix, 564
IF-THEN/ELSE statement
  conditionally executing statements, 564
IFFT function
  computing the inverse finite Fourier transform, 566
IMLMLIB Module Library
  modules reference, 822–824, 826–827
  overview, 819–821
INDEX statement
  indexing a variable in a SAS data set, 567
INFILE statement
  opening a file for input, 567
INPUT statement
  inputing data, 568
INSERT function
  inserting one matrix inside another, 570
INT function
  truncating a value, 571
INV function
  computing a matrix inverse, 572
inverses
  Moore-Penrose inverse, 496, 727, 733–735
INVUPDT function
  updating a matrix inverse, 573
IPF call
  performing an iterative proportional fit, 574
ISM TIMSAC packages, 270–272

J

J function
  createing a matrix of identical values, 576

K

Kalman filter subroutines
  examples, 280
  one-step forecast for SSM, 583–586
  one-step predictions, 578–580
  overview, 279–280
  smoothed estimate, 580–583
  smoothed state vectors, 586–587
  syntax, 578

L

LABEL
  quadratic form maximization, 619
LCP call
  solving the linear complementarity problem, 593
least absolute value regression, 589–592
LENGTH function
  finding the lengths of character matrix elements, 596
Linear Algebra Functionality, 456
linear least squares
  full rank example, 696–697
  QR decomposition, 710
  rank deficient solutions, 727, 729–732
LINK and RETURN statements
  jumping to another statement, 596
LIST statement
  displaying observations of a data set, 597
LMS and LTS calls
  performing robust regression, 600
LOAD statement
  loading modules and matrices, 608
LOC function
  finding nonzero elements of a matrix, 609
LOG function
  takeing the natural logarithm, 610
LP call
  solving the linear programming problem, 611
LTS call
  performs robust regression, 613
LUPDT call, 613

M

MAD function
  univariate median-absolute-deviation, 614
MARG call
  evaluating marginal totals, 616
matrix decomposition
  Cholesky decomposition, 727–729
  complete orthogonal decomposition, 478, 494–496
  downdating and updating, 709–712, 727
  QR decomposition, 679–682, 693–694, 696–697
Matrix Inquiry Functions, 453
Matrix Reshaping Functions, 454
MATTRIB statement
  associating printing attributes with matrices, 617
MAX function
  finding the maximum value of matrix, 618
MCD and MVE calls, 621
median computation, 824
MIN function
  finding the smallest element of a matrix, 628
MOD function
  computing the modulo (remainder), 629

N

NAME function
  listing the names of arguments, 629
NCOL function
  finding the number of columns of a matrix, 630
NLENG function
  finding the size of an element, 630
nonlinear optimization subroutines
  advanced examples, 336
  conjugate gradient optimization, 634
  control parameters vector, 332–334
  double dogleg optimization, 635, 637
  feasible point computation, 641
  finite difference approximations, 314–316, 637–640
  global vs. local optima, 307
  hybrid quasi-Newton optimization, 642–644
  Kuhn-Tucker conditions, 308

- least-squares methods, 642–645, 647
- Levenberg-Marquardt optimization, 645, 647
- Nelder-Mead simplex optimization, 648–649, 652
- Newton-Raphson optimization, 652–653, 655
- Newton-Raphson ridge optimization, 655, 658
- objective function and derivatives, 309–314
- options vector, 319–321, 323–325
- parameter constraints, 317–319
- printing optimization history, 334, 336
- quadratic optimization, 663, 665, 667
- quasi-Newton optimization, 658–659, 661, 663
- return codes, 309
- termination criteria, 325–331
- trust-region optimization, 667–668

NORMAL function
- generating a pseudo-random normal deviate, 668

NROW function
- finding the number of rows of a matrix, 669

NUM function
- producing a numeric representation of a character matrix, 669

Numerical Analysis Functionality, 456

numerical integration, 697–703
- adaptive Romberg method, 698
- of differential equations, 670–675
- specifying subintervals, 698
- two-dimensional integration, 701

O

OPSCAL Function, 676

Optimization Subroutines, 457

ORPOL function
- generating orthogonal polynomials, 678

orthogonal factorization, 727–729

orthogonalization
- by ORTVEC call, 679–682

P

PARSE statement
- parsing matrix elements as statements, 682

PAUSE statement
- interrupting module execution, 683

PGRAF call
- producing scatter plots, 684

POLYROOT function
- finding zeros of a real polynomial, 685

PRINT statement
- printing matrix values, 686

printing matrices, 827

probability contour plot, 823

PRODUCT function
- multiplying matrices of polynomials, 688

PURGE statement
- removing observations marked for deletion, 689

PUSH call, 689

PUT statement
- writing data to an external file, 690

PV function
- calculating present value, 691

Q

quadratic form maximization, 620–621

quartile computation, 825

QUEUE call
- queueing SAS statements, 703

QUIT statement
- exiting from IML, 704

R

RANK function
- ranking elements of a matrix, 705

RANKTIE function
- ranking matrix elements using tie-averaging, 706

RATES function
- converting interest rates, 706

RATIO function
- dividing matrix polynomials, 707

READ statement
- reading observations from a data set, 713

Reduction Functions, 453

regression, 825
- best subsets, 621
- least absolute value, 589–592
- response surface, 824

REMOVE function
- discarding elements from a matrix, 716

REMOVE statement
- removing matrices from storage, 717

RENAME call
- renaming SAS data sets, 717

REPEAT function
- creating a new matrix of repeated values, 718

REPLACE statement
- replacing values, 718

RESET statement
- setting processing options, 721

reshaping matrices, 822, 826

response surface regression, 824

RESUME statement
- resuming execution, 723

RETURN statement
- returning to caller, 723

ROOT function
- performing the Cholesky decomposition of a matrix, 724

ROWCAT function
- concatenating rows without blank compression, 724

ROWCATC function
- concatenating rows with blank compression, 725

RUN statement
- executing statements in a module, 726

S

SAVE statement
- saving data, 735

Scalar Functions, 453

scatter plots, 823

sequential tests, 736–746, 748–749

group sequential methods, 742–743
minimizing average sample number (ASN), 746, 748
randomized clinical trials, 746, 748
scaling, 738
shifting, 738
Set Functions, 458
SETDIF function
comparing elements of two matrices, 749
SETIN statement
making a data set current for input, 750
SETOUT statement
making a data set current for output, 750
SHAPE function
reshaping and repeating values, 751
SHOW statement
printing system information, 753
SOLVE function
solving a system of linear equations, 754
SORT statement
sorting a SAS data set, 755
SOUND call
producing a tone, 756
splines, 756, 758–760
integration of splines, 758
SPOT function
calculating spot rates, 760
SQRSYM function
converting to a square matrix, 761
SQRT function
calculating the square root, 761
SSQ function
calculating the sum of squares, 762
standardizing numeric data, 827
START and FINISH statements
defining a module, 762
Statistical Functionality, 454
STOP statement
stopping execution of statements, 763
STORAGE function
listing names of matrices and modules, 763
STORE statement
storing matrices and modules, 763
SUBSTR function
taking substrings of matrix elements, 764
substring replacement, 826
SUM function
summing all elements, 765
SUMMARY statement
computing summary statistics, 765
SVD call
computing the singular value decomposition, 768
SWEEP function
sweeping a matrix, 770
SYMSQR function
converting to a symmetric matrix, 771

T

T function
transposing a matrix, 772
TEIGEN call
computing eigenvalues and eigenvectors, 772
TEIGVAL functions
computing eigenvalues, 773
TEIGVEC functions
computing eigenvectors, 773
time series analysis and control
AR model selection, 225–227, 795
ARMA model prediction, 245–246, 792
Bayesian constrained least squares, 265–267
Bayesian seasonal adjustment, 242–244, 254–256, 782–783
instantaneous response model, 229–230, 274–276
ISM TIMSAC packages, 270–272
least squares and Householder transformation, 264–265
locally stationary multivariate time series, 789–790
locally stationary time series, 788
minimum AIC method, 225–230, 250–252
missing values, 269
multivariate time series, 245–246, 259–261, 792
nonstationary covariance function analysis, 794
nonstationary data analysis, 231–232, 234–242
nonstationary time series, 256–259, 784–787
overview, 223
periodic AR model, 791–792
roots of AR and MA equations, 247–248, 793
smoothness priors modeling, 252–253, 784–787
spectral analysis, 261–264
state space and Kalman filter method, 267–269
VAR model, 228–230, 273–275, 790–791
Time Series Functionality, 455
TOEPLITZ function
generating a Toeplitz matrix, 773
TPSPLINE call
computing thin-plate smoothing splines, 775
TPSPLNEV call
evaluating thin-plate smoothing splines, 777
TRACE function
summing diagonal elements, 781
triangular linear systems, 781
TYPE function
determining matrix types, 796

U

UNIFORM function
generating pseudo-random uniform deviates, 797
UNION function
performing unions of sets, 797
UNIQUE function
sorting and removing duplicates, 798
USE statement
opening SAS data sets, 798

V

VALSET call
performing indirect assignments, 800

VALUE function
assigning values, 801
VARMACOV Call
computing auto-cross covariance matrices, 801
VARMASIM Call
generating a VARMA(p,q) series, 803
VECDIAG function
creating vectors, 805
VNORMAL Call
generating multivariate normal random series, 805
VTSROOT Call
calculating characteristic roots, 806

W
WINDOW statement
opening a display window, 807

X
XMULT function
performing accurate matrix multiplication, 809
XSECT function
intersecting sets, 810

Y
YIELD function
calculating yield-to-maturity of a cash-flow streams, 810

Syntax Index

A
ABORT statement, 476
ABS statement, 476
ALL statement, 476
ANY function, 477
APPCORT call, 478
APPEND statement, 479
APPLY function, 480
ARMACOV call, 482
ARMALIK call, 484
ARMASIM function, 485

B
BLOCK function, 487
BRANKS function, 487
BTRAN function, 488
BYTE function, 489

C
CALL statement, 490
CHANGE call, 491
CHAR function, 491
CHOOSE function, 492
CLOSE statement, 493
CLOSEFILE statement, 494
COLVEC function, 822
COMPORT call, 494–496
CONCAT function, 496
CONTENTS function, 497
CONVEXIT function, 498
CONVMOD function, 499
CORR function, 822
COVLAG function, 499
CREATE statement, 500
CSHAPE function, 502
CUSUM function, 504
CVEXHULL function, 504

D
DATASETS function, 505
DELETE call, 506
DELETE statement, 507
DESIGN function, 509
DESIGNF function, 509
DET function, 510
DIAG function, 511
DISPLAY statement, 512
DO and END statements, 514
DO DATA statement, 516
DO function, 513
DO statement, iterative, 515
DO UNTIL statement, 517
DO WHILE statement, 518
DURATION function, 519

E
ECHELON function, 520
EDIT statement, 520
EIGEN call, 523
EIGVAL function, 525
EIGVEC function, 526
END statement, 526
EXECUTE call, 527
EXP function, 527

F
FFT function, 528
FILE statement, 529
FIND statement, 530
FINISH statement, 532
FORWARD function, 532
FREE statement, 533

G
GBLKVP call, 533
GBLKVPD call, 534
GBXWHSKR call, 822
GCLOSE call, 535
GDELETE call, 535
GDRAW call, 536
GDRAWL call, 537
GENEIG call, 538
GGRID call, 539
GINCLUDE call, 540
GINV function, 540
GOPEN call, 541
GOTO statement, 542
GPIE call, 543
GPIEXY call, 544
GPOINT call, 545
GPOLY call, 546
GPORT call, 547
GPORTPOP call, 547
GPORTSTK call, 548
GPROBCNT call, 823
GSCALE call, 548
GSCRIPT call, 549
GSET call, 550

GSHOW call, 551
GSORTH call, 552
GSTART call, 553
GSTOP call, 554
GSTRLEN call, 554
GTEXT and GVTEXT calls, 555
GWINDOW call, 556
GXAXIS and GYAXIS calls, 557
GXYPLOT call, 823

H

HALF function, 558
HANKEL function, 559
HDIR function, 561
HERMITE function, 562
HOMOGEN function, 563

I

I function, 564
IF-THEN/ELSE statement, 564
IFFT function, 566
IMLMLIB Module Library
  modules reference, 822–824, 826–827
  overview, 819–821
INDEX statement, 567
INFILE statement, 567
INPUT statement, 568
INSERT function, 570
INT function, 571
INV function, 572
INVUPDT function, 573
IPF call, 574

J

J function, 576
JROOT function, 577

K

KALCVF call, 281, 286, 288–289, 578–580
KALCVS call, 286, 580–583
KALDFF call, 289, 583–586
KALDFS call, 586–587
Kalman filter subroutines
  examples, 280
  overview, 279–280
  syntax, 578

L

LAV call, 589–592
LCP call, 593
LENGTH function, 596
LINK and RETURN statements, 596
LIST statement, 597
LMS and LTS calls, 600
LOAD statement, 608
LOC function, 609
LOG function, 610
LP call, 611
LTS call, 613
LUPDT call, 613

M

MAD function, 614
MARG call, 616
MATTRIB statement, 617
MAX function, 618
MAXQFORM call, 619–621
MCD and MVE calls, 621
MEDIAN function, 824
MIN function, 628
MOD function, 629

N

NAME function, 629
NCOL function, 630
NLENG function, 630
nonlinear optimization subroutines
  advanced examples, 336
  details, 307
  introductory examples, 299
  NLPCG Call, 341
  NLPCG call, 634
  NLPDD Call, 346, 349, 366, 635
  NLPDD call, 635
  NLPDD Call, 637
  NLPFDD Call, 355
  NLPFDD call, 637
  NLPFDD Call, 638
  NLPFDD call, 639
  NLPFDD Call, 640
  NLPFEA call, 641
  NLPHQN call, 642
  NLPHQN Call, 643–644
  NLPLM Call, 357, 645
  NLPLM call, 645
  NLPLM Call, 647
  NLPNMS call, 648
  NLPNMS Call, 649, 652
  NLPNRA call, 652
  NLPNRA Call, 653, 655
  NLPNRR Call, 342
  NLPNRR call, 655
  NLPNRR Call, 658
  NLPQN Call, 351–352, 369
  NLPQN call, 658
  NLPQN Call, 659, 661, 663
  NLPQUA call, 663
  NLPQUA Call, 665, 667
  NLPTR Call, 337, 354
  NLPTR call, 667
  NLPTR Call, 668
  overview, 297
  syntax, 630
NORMAL function, 668
NROW function, 669
NUM function, 669

O

ODE call, 670–675
OPSCAL function, 676

ORPOL function, 678
ORTVEC call, 679–682

P
PARSE statement, 682
PAUSE statement, 683
PGRAF call, 684
POLYROOT function, 685
PRINT statement, 686
PRODUCT function, 688
PURGE statement, 689
PUSH call, 689
PUT statement, 690
PV function, 691

Q
QR call, 693–694, 696–697
QUAD call, 697–703
QUADREG call, 824
QUARTILE function, 825
QUEUE call, 703
QUIT statement, 704

R
RANK function, 705
RANKTIE function, 706
RATES function, 706
RATIO function, 707
RDODT call, 709–712
READ statement, 713
REGRESS call, 825
REMOVE function, 716
REMOVE statement, 717
RENAME call, 717
REPEAT function, 718
REPLACE statement, 718
RESET statement, 721
RESUME statement, 723
RETURN statement, 723
ROOT function, 724
ROWCAT function, 724
ROWCATC function, 725
ROWVEC function, 826
RSUBSTR function, 826
RUN statement, 726
RUPDT call, 709–712, 727
RZLIND call, 727–735

S
SAVE statement, 735
SEQ call, 736–746, 748
SEQSCALE call, 736–746, 748
SEQSHIFT call, 736–746, 748–749
SETDIF function, 749
SETIN statement, 750
SETOUT statement, 750
SHAPE function, 751
SHOW statement, 753
SOLVE function, 754
SORT statement, 755
SOUND call, 756
SPLINE call, 756, 758–759
SPLINEC call, 756, 758–759
SPLINEV function, 758–760
SPOT function, 760
SQRSYM function, 761
SQRT function, 761
SSQ function, 762
STANDARD function, 827
START and FINISH statements, 762
STOP statement, 763
STORAGE function, 763
STORE statement, 763
SUBSTR function, 764
SUM function, 765
SUMMARY statement, 765
SVD call, 768
SWEEP function, 770
SYMSQR function, 771

T
T function, 772
TABPRT call, 827
TEIGEN call, 772
TEIGVAL functions, 773
TEIGVEC functions, 773
TIMSAC subroutines
  advanced examples, 273
  details, 250
  introductory examples, 225
  overview, 223
  syntax, 249
  TSBAYSEA subroutine, 242–244, 782–783
  TSDECOMP subroutine, 241, 784–787
  TSMLOCAR subroutine, 231, 788
  TSMLOMAR subroutine, 239, 789–790
  TSMULMAR subroutine, 228–230, 273–276, 790–791
  TSPEARS subroutine, 791–792
  TSPRED subroutine, 245–246, 275, 792
  TSROOT subroutine, 247–248, 793
  TSTVCAR subroutine, 794
  TSUNIMAR subroutine, 795
TOEPLITZ function, 773
TPSPLINE call, 775
TPSPLNEV call, 777
TRACE function, 781
TRISOLV function, 781
TYPE function, 796

U
UNIFORM function, 797
UNION function, 797
UNIQUE function, 798
USE statement, 798

V
VALSET call, 800

VALUE function, 801
VARMACOV Call, 801
VARMASIM Call, 803
VECDIAG function, 805
VNORMAL Call, 805
VTSROOT Call, 806

W
WINDOW statement, 807

X
XMULT function, 809
XSECT function, 810

Y
YIELD function, 810