Linear Algebra Commands in MAPLE

Description of commands in the *LinearAlgebra* Package

Linear Algebra	Description
1. General	
with(LinearAlgebra):	read in the linear algebra package
type(*expr*, Matrix)	gives *true* if *expr* has the form of a matrix, else false
type(*expr*, Vector)	gives *true* if *expr* has the form of a vector
RowDimension(a)	the number of rows of the matrix a
ColumnDimension(a)	the number of columns of a
Dimensions(a)	the number of rows and columns
Dimension(v)	the dimension of a vector
2. Constructing matrices and vectors	
Matrix($[[a_{11}, \dots, a_{1n}], \dots, [a_{m1}, \dots, a_{mn}]]$)	build an $m \times n$ matrix $a = (a_{ij})$ by listing its elements row by row
$<< a_{11}, \dots, a_{m1} > \mid \dots \mid < a_{1n} \dots \mid a_{mn} >>$	build an $m \times n$ matrix $a = (a_{ij})$ by listing its elements column by column
Matrix(m, n, f)	build an $m \times n$ matrix whose ij-th entry is $f(i,j)$
Matrix($m, n, (i,j) \to expr$)	build an $m \times n$ matrix with ij-th entry $expr(i,j)$
Vector($[a_1, \dots, a_n]$) or $< a_1, \dots, a_n >$	construct a (column) vector with entries $a_1, \dots, a_n$
3. Basic matrix operations	
$a + b$	the sum of the matrices a and b
$c * a$	multiplying the matrix a by a scalar c
$a \,.\, b$ or Multiply(a, b)	matrix multiplication
a^n	the n-th power of a matrix
$1/a$ or a^(-1)	the inverse of a matrix
map(f, a)	the matrix obtained by applying the function f to each entry of a
Transpose(a) or a^%T	the transpose of a
Determinant(a)	the determinant of a
Trace(a)	the trace of a
Rank(a)	the rank of a
Adjoint(a)	the adjoint matrix (matrix of minors) of a

4. Special matrices	
ZeroMatrix(m, n)	the $m \times n$ zero matrix
DiagonalMatrix(L)	generate a diagonal matrix with the list L as the diagonal entries
IdentityMatrix(n)	generate an $n \times n$ identity matrix
VandermondeMatrix(L)	generate a Vandermonde matrix whose 2nd column is the list L
BandMatrix(L)	create a tri–diagonal matrix (or an arbitrary band matrix) using the list L
JordanBlockMatrix$([[c, n]])$	generate an $n \times n$ Jordan block with eigenvalue c
JordanBlockMatrix$([[c_1, n_1], \ldots, [c_r, n_r]])$	generate a Jordan matrix with Jordan blocks $(c_1, n_1), \ldots, (c_r, n_r)$
CompanionMatrix(p, x)	generate the $n \times n$ companion matrix associated to a monic polynomial $p(x)$ of degree n
5. Extracting parts of a matrix	
$a[i, j]$	the (i, j)-th entry of the matrix a
Row(a, i)	the i-th row of the matrix a
Column(a, j)	the j-th column of the matrix a
SubMatrix$(a, [i_1, \ldots, i_r], [j_1, \ldots, j_s])$ or $a[[i_1, \ldots, i_r], [j_1, \ldots, j_s]]$	the $r \times s$ submatrix of a with row indices i_k and column indices j_k
SubMatrix$(a, i_0..i_1, j_0..j_1)$ or $a[i_0..i_1, j_0..j_1]$	the submatrix of a having row and column indices from i_0 to i_1 and j_0 to j_1, respectively
6. Pasting and altering matrices	
$< a, b, \ldots >$	join two (or more) matrices a, b, … vertically
$< a \mid b \mid \ldots >$	join two (or more) matrices horizontally
Matrix$(m + r, n + s, [a]$,fill$= c)$;	enlarge the $m \times n$ matrix a by r additional rows and s additional columns with the value c
Matrix(m, n, L)	partition a list L of mn elements into an $m \times n$ matrix
DiagonalMatrix$([a_1, a_2, \ldots])$	construct a block diagonal matrix using the (square) matrices $a_1, a_2, \ldots$
$a[i, j] := expr$	replace the (i, j)-th entry of matrix a by the expression *expr*
DeleteRow$(a, i_1..i_2)$	delete rows i_1 to i_2 of a
DeleteColumn$(a, i_1..i_2)$	delete columns j_1 to j_2 of a

7. Row and column operations	
RowOperation($a, [i_2, i_1], c$)	add c times row i_1 to row i_2 (thus, the result is put in row i_2)
ColumnOperation($a, [j_2, j_1], c$)	add c times column j_1 to col. j_2
RowOperation(a, i, c)	multiply row i by c
ColumnOperation(a, j, c)	multiply column j by c
RowOperation($a, [i_1, i_2]$)	interchange rows i_1 and i_2
ColumnOperation($a, [j_1, j_2]$)	interchange columns j_1 and j_2
8. Row reduction and solutions of linear systems	
LinearSolve(a, b)	find the (general) solution of the matrix equation $ax = b$
NullSpace(a)	compute a basis for the nullspace of a matrix a
GaussianElimination(a)	find an upper triangular matrix which is row equivalent to a
BackwardsSubstitution(a, b)	solve $ax = b$ by back-substitution (if a is upper triangular)
ReducedRowEchelonForm(a)	compute the reduced row echelon form of a
HermiteForm(a)	row reduce the integer matrix a to an upper-Δ integer matrix
SmithForm(a)	row reduce and column reduce the integer matrix a to a diagonal integer matrix
9. Eigenvalues, Eigenvectors	
CharacteristicMatrix(a, x)	compute the characteristic matrix $Ix - a$ (x a variable)
CharacteristicPolynomial(a, x)	find the characteristic polynomial of a
MinimalPolynomial(a, x)	find the minimal polynomial of a
Eigenvalues(a)	compute the eigenvalues of a
Eigenvectors(a)	find the eigenvectors of a
JordanForm(a) or JordanForm (a, output$=$ ['J','Q']);	compute the Jordan canonical form J of a and the matrix q such that $a = q^{-1}Jq$
FrobeniusForm(a) or FrobeniusForm(a, output $=$ ['F', 'Q'])	compute the Frobenius (or rational) canonical form F of a; find p such that $a = pFp^{-1}$
IsSimilar(a, b) or IsSimilar(a, b, output $=$ ['query', 'C'])	determine whether a is similar to b; if so, find the matrix q such that $a = q^{-1}bq$

10. Vector operations	
convert($< v, w >$, Vector)	direct sum of two vectors v, w
$v + w$	add two vectors v, w
$c * v$	multiply the vector v by scalar c
$a \; . \; v$	multiply the matrix a by the (column) vector v (on the right)
Transpose(v) . a	multiply the matrix a by the vector v (on the left)
$v \; . \; w$	the dot (scalar) product of v and w
Norm($v, 2$)	the length or norm $\|\|v\|\|$ of a vector v
Normalize(v)	divide the vector v by its length
VectorAngle(v, w)	the angle between the vectors v and w
11. Vector spaces	
Basis(L)	find a basis of the vector space spanned by list L of vectors
IntersectionBasis($[L_1, L_2 \ldots]$)	find a basis of the intersection of the spaces spanned by the lists $L_1, L_2, \ldots$
SumBasis($[L_1, L_2 \ldots]$)	find a basis of the sum/union of the spaces spanned by the lists $L_1, L_2, \ldots$
RowSpace(a), ColumnSpace(a)	find a basis for the row/column space of the matrix a
NullSpace(a)	compute a basis for the nullspace of a matrix a
GramSchmidt($[v_1, \ldots, v_n]$)	apply the Gram-Schmidt procedure to the vectors $v_1, \ldots, v_n$
12. Miscellaneous	
IsOrthogonal(a)	test whether a is an orthogonal matrix
LeastSquares(a, b)	find x such that $\| ax - b \|$ is minimal
Equal(a, b)	test whether matrices a and b are equal
convert(evalm(subs($x = a, f$)), Matrix);	evaluate the matrix polynomial $f(a)$ (where $f(x)$ is a polynomial)