1 Black-box Testing

1.A linalg.dot

1.A.1 documentation

For 2-D arrays it is equivalent to matrix multiplication, and for 1-D arrays to inner product of vectors (without complex conjugation). For N dimensions it is a sum product over the last axis of a and the second-to-last of b:

dot(a, b)[i,j,k,m] = sum(a[i,j,:] * b[k,:,m]) takes: two arrays a,b

1.B linalg.multidot

Compute the dot product of two or more arrays in a single function call, while automatically selecting the fastest evaluation order.

multi_dot chains numpy.dot and uses optimal parenthesization of the matrices [R44] [R45]. Depending on the shapes of the matrices, this can speed up the multiplication a lot.

If the first argument is 1-D it is treated as a row vector. If the last argument is 1-D it is treated as a column vector. The other arguments must be 2-D.

1.B.1 tests

1.C linalg.vdot

1.C.1 documentation

1.C.2 tests

1.D linalg.inner

1.D.1 documentation

1.D.2 tests

1.E linalg.outer

1.E.1 documentation

1.E.2 tests

1.F linalg.matmul

1.F.1 documentation

1.F.2 tests

1.G linalg.tensordot

1.G.1 documentation

1.G.2 tests

1.H linalg.einsum

1.H.1 documentation

1.H.2 tests

1.I linalg.matrix_power

1.I.1 documentation

1.I.2 tests

1.J linalg.kron

1.J.1 documentation

1.J.2 tests

1.K linalg.eig

1.K.1 documentation

1.K.2 tests

1.L linalg.eigh

1.L.1 documentation

1.L.2 tests

1.M linalg.eigvals

1.M.1 documentation

1.M.2 tests

1.N linalg.eigvalsh

1.N.1 documentation

1.N.2 tests