NumPy Cheatsheet

NumPy is a fundamental library for numerical computing in Python.

1. Array Creation

- **np.array(list)** - Creates an array from a list or tuple.
- **np.zeros(shape)**: Creates an array filled with zeros.
- **np.ones(shape)**: Creates an array filled with ones.
- **np.full(shape, value)**: Creates an array filled with a specific value.
- **np.arange(start, stop, step):** Creates an array with a range of values.
- **np.linspace(start, stop, num):** Creates an array of evenly spaced values.
- **np.eye(N):** Creates an identity matrix of size N.
- **np.random.rand(d0, d1, ...):** Creates an array with random values between 0 and 1.
- **np.random.randint(low, high, size):** Generates random integers in a given range.
- **np.random.normal(mean, std, size):** Creates an array with normally distributed values.

---

2. Array Attributes

- **arr.shape** - Returns the shape of the array.
- **arr.ndim** - Returns the number of dimensions of the array.
- **arr.size** - Returns the total number of elements in the array.
- **arr.dtype** - Returns the data type of the array elements.
- **arr.itemsize** - Returns the memory size of each array element.

---

3. Array Manipulation

- **np.reshape(arr, new_shape)** - Reshapes an array.
- **np.ravel(arr)** - Flattens an array into a 1D array.
- **np.transpose(arr)** - Transposes the given matrix.
- **np.hstack((arr1, arr2))** - Stacks arrays horizontally.
- **np.vstack((arr1, arr2))** - Stacks arrays vertically.
- **np.concatenate((arr1, arr2), axis=n)** - Concatenates arrays along an axis.
- **np.split(arr, sections, axis)** - Splits an array into multiple sub-arrays.
- **np.expand_dims(arr, axis)** - Expands the dimensions of an array.

- **np.squeeze(arr)** - Removes dimensions of size one.

---

4. Mathematical Operations

- **np.add(arr1, arr2)** - Adds two arrays element-wise.
- **np.subtract(arr1, arr2)** - Subtracts two arrays element-wise.
- **np.multiply(arr1, arr2)** - Multiplies two arrays element-wise.
- **np.divide(arr1, arr2)** - Divides two arrays element-wise.
- **np.power(arr, n)** - Raises each element of an array to the power n.
- **np.sqrt(arr)** - Computes the square root of each element.
- **np.exp(arr)** - Computes the exponential of each element.
- **np.log(arr) -** Computes the natural logarithm.
- **np.log10(arr)** - Computes the logarithm base 10.
- **np.sin(arr)** - Computes the sine of each element.
- **np.cos(arr)** - Computes the cosine of each element.
- **np.tan(arr)** - Computes the tangent of each element.

---

5. Statistical Functions

- **np.mean(arr)** - Computes the mean.
- **np.median(arr)** - Computes the median.
- **np.std(arr)** - Computes the standard deviation.
- **np.var(arr)** - Computes the variance.
- **np.min(arr)** - Returns the minimum value.
- **np.max(arr)** - Returns the maximum value.
- **np.sum(arr)** - Computes the sum of elements.
- **np.prod(arr)** - Computes the product of elements.
- **np.cumsum(arr)** - Computes the cumulative sum.
- **np.cumprod(arr)** - Computes the cumulative product.
- **np.percentile(arr, q)** - Computes the q^{th} percentile.
- **np.corrcoef(arr1, arr2)** - Computes correlation coefficient.
- **np.histogram(arr, bins)** - Computes the histogram.

---

6. Linear Algebra

- **np.dot(arr1, arr2)** - Computes the dot product.
- **np.matmul(arr1, arr2)** - Computes matrix multiplication.
- **np.linalg.inv(arr)** - Computes the inverse of a matrix.
- **np.linalg.det(arr)** - Computes the determinant.
- **np.linalg.eig(arr)** - Computes the eigenvalues and eigenvectors.
- **np.linalg.svd(arr)** - Computes Singular Value Decomposition.
- **np.linalg.solve(A, b)** - Solves a linear equation system Ax = b.

---

7. Boolean Masking and Filtering

- **np.where(condition, x, y)** - Returns elements based on condition.
- **np.nonzero(arr)** - Returns the indices of nonzero elements.
- **np.unique(arr)** - Finds unique elements in an array.
- **np.isin(arr, values)** - Checks for elements in an array.
- **np.any(arr)** - Checks if any element is True.
- **np.all(arr)** - Checks if all elements are True.

---

8. Sorting and Searching

- **np.sort(arr)** - Sorts an array.
- **np.argsort(arr) -** Returns the indices that would sort the array.
- **np.argmax(arr)** - Returns the index of the maximum value.
- **np.argmin(arr)** - Returns the index of the minimum value.
- **np.searchsorted(arr, value) -** Finds indices where elements should be inserted.
- **np.bincount(arr)** - Counts occurrences of values in an array.

---

9. File I/O

- **np.save(filename, arr)** - Saves an array to a binary file.
- **np.load(filename)** - Loads an array from a binary file.
- **np.savetxt(filename, arr, delimiter=',')** - Saves an array to a text file.
- **np.loadtxt(filename, delimiter=',')** - Loads an array from a text file.

---

10. Random Number Generation

- **np.random.seed(seed)** - Sets the random seed.
- **np.random.shuffle(arr)** - Shuffles an array randomly.
- **np.random.choice(arr, size)** - Selects random elements from an array.

Official documentation: https://numpy.org/doc/