# Matrix Operations

This lesson discusses various matrix operations.

# Inverse #

Suppose we have to solve the equation $AX = b$ for $b$.

If vector $b$ changes, one has to call `np.linalg.solve()` over and over. Instead, if $A^{-1}$ were calculated once, matrix multiplication with $b$ could be done over and over, more efficiently, to obtain different solutions.

The inverse of a matrix may be computed with the `inv` function of the `linalg` package.

Let's use the inverse of the matrix to obtain the solution to the problem in the first lesson.

$$AX = b$$

$$A^{-1}AX = A^{-1}b$$

$$X = A^{-1}b$$

If the inverse of matrix `A` is called `Ainv`, the solution for the equation may be obtained through matrix multiplication of `Ainv` with the right-hand side. Let's look at an implementation of this below.
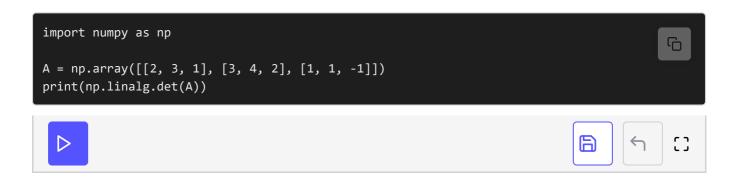
```
import numpy as np

A = np.array([[2, 3, 1], [3, 4, 2], [1, 1, -1]])
b = np.array([17, 25, 6])

Ainv = np.linalg.inv(A)
x =  np.dot(Ainv, b)

print(x)
```

# Determinant #

`det()` from the `linalg` submodule is used to compute the determinant of the input matrix.

```
import numpy as np

A = np.array([[2, 3, 1], [3, 4, 2], [1, 1, -1]])
print(np.linalg.det(A))
```

# Trace #

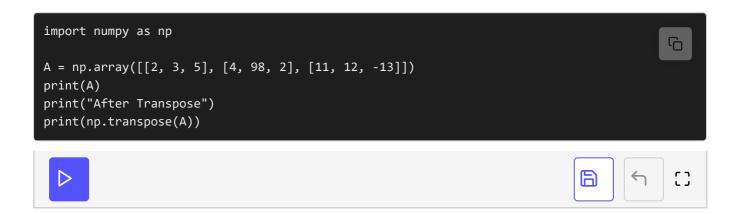`trace()` from the `numpy` module is used to compute the trace of the input matrix.

> 💡 Trace is the sum of the diagonal values of a matrix.

```
import numpy as np

A = np.array([[2, 3, 1], [3, 4, 2], [1, 1, -1]])
print(np.trace(A))
```

Transpose

# Transpose #

`transpose()` from the `numpy` module returns the transpose of the input array.

> 💡 The transpose of a matrix is a new matrix whose rows are the columns of the original.

```python
import numpy as np

A = np.array([[2, 3, 5], [4, 98, 2], [11, 12, -13]])
print(A)
print("After Transpose")
print(np.transpose(A))
```

# Euclidean norm #

The Euclidean norm of an $m \times n$ matrix is given by the following equation:

$$\| A \| = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} | a_{ij} |^2}$$

The `norm()` method from the `linalg` submodule is used to calculate the Euclidean norm of a matrix.

```python
import numpy as np

v = np.array([0, 3, 4])
M = np.array([[2, 3, 5], [4, 98, 2], [11, 12, -13]])

print("Norm of v is", np.linalg.norm(v))
print("Norm of M is", np.linalg.norm(M))
```

In the next lesson, we will discuss sparse matrices.

#