## Memory Layout of NumPy/PyTorch Tensors

**Tensors**: Multi-dimensional arrays stored in **contiguous memory**

### Row-Major (C)

Default

Last dimension varies fastest

| 0,0 | 0,1 | 1,0 | 1,1 |
|---|---|---|---|

### Column-Major (Fortran)

First dimension varies fastest

| 0,0 | 1,0 | 0,1 | 1,1 |
|---|---|---|---|

### Stride

Number of bytes to jump to next element in each dimension

### Contiguous Tensors ✓

✓ Faster operations

✓ Better cache locality

✓ Optimized GPU performance

### Non-Contiguous ⚠

⚠ After transpose(), view()

⚠ May need .contiguous()

⚠ Slower memory access

### Memory Layout Impact

- Cache hits efficiency
- Vectorization performance
- GPU operation speed

### Check Methods

`.is_contiguous()`

`.stride()`