AGENDA

Getting Started

Background

Scipy Code

Step through SciPy Gauss Spline code

CuPy Code

Step through Elementwise Kernel modifications

Nsight Systems

Using code profile to evaluate performance

WHAT ARE ELEMENTWISE OPERATIONS?

Same operation done element by element

```
import numpy as np

x = np.random.rand(4)    →    [0.97605699 0.44687052 0.64403474 0.37204669]
y = np.random.rand(4)    →    [0.58380232 0.63697089 0.79787468 0.50245494]


np.sqrt(x)               →    [0.98795597 0.66848375 0.80251775 0.6099563 ]
np.exp(y)                →    [1.79284244 1.89074492 2.22081596 1.65277376]
np.maximum(x, y)         →    [0.97605699 0.63697089 0.79787468 0.50245494]
```

SCIPY PYTHON CODE

```
import numpy as np
from numpy.core.umath import (sqrt, exp)

def gauss_spline(x, n):
   """ Gaussian approximation to B-spline basis
function of order n.

   Parameters
   ----------
   x : array_like
       a knot vector
   n : int
       The order of the spline. Must be
non-negative, i.e., n >= 0

Returns
   -------
   res : ndarray
       B-spline basis function values approximated by
       a zero-mean Gaussian function.

x = np.asarray(x)
   signsq = (n + 1) / 12.0
   return 1 / sqrt(2 * np.pi * signsq) * exp(-x ** 2
/ 2 / signsq)
```

Source: https://github.com/scipy/scipy/blob/master/scipy/signal/bsplines.py

CUPY COMPARISON

Results of baseline SciPy code

	First Pass (µs)	Average over 10,000 runs (µs)	Speed Up
SciPy	1118.15	724.45	1
CuPy Elementwise Kernel	-	-	-

CUPY ELEMENTWISE KERNEL

class **cupy.ElementwiseKernel**(in_params, out_params, operation, name='kernel', reduce_dims=True, preamble='', no_return=False, return_tuple=False, **kwargs)

- **in_params** (*str*) – Input argument list.
- **out_params** (*str*) – Output argument list.
- **operation** (*str*) – The body in the loop written in CUDA-C/C++.
- **name** (*str*) – Name of the kernel function. It should be set for readability of the performance profiling

Source: https://docs.cupy.dev/en/stable/reference/generated/cupy.ElementwiseKernel.html

CUPY COMPARISON

	First Pass (µs)	Average over 10,000 runs (µs)	Speed Up (µs)
SciPy	1118.15	724.45	1
CuPy Elementwise Kernel	69260.336	37.38	19.38

* First pass include JIT compilation

CUPY COMPARISON

	First Pass (µs)	Average over 10,000 runs (µs)	Speed Up
Naive CuPy	70165.56	69.76	1
CuPy Elementwise Kernel	66803.28	34.96	2.01

* First pass include JIT compilation