Prediction of High-resolution Maps of Storm-driven Coastal Flooding Using Deep Learning

Tomás Cuevas López[1], Brandon Tucker[1], Casey Dietrich[1] & Dylan Anderson[2]

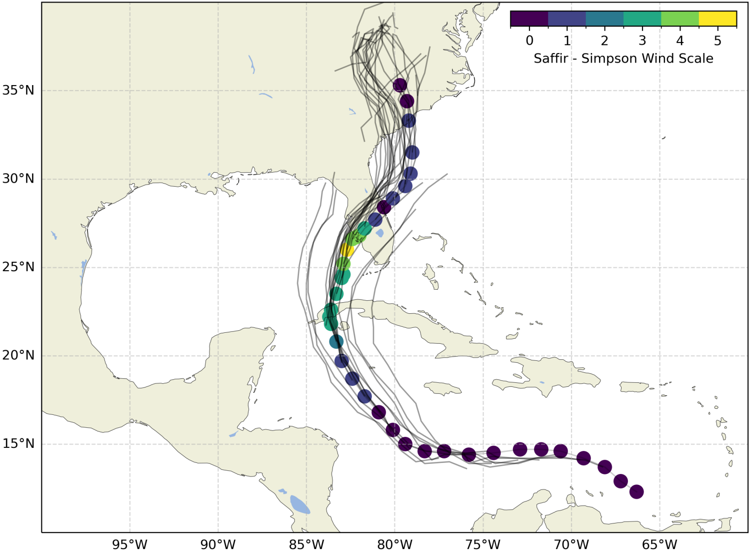
[1]Department of Civil, Construction, and Environmental Engineering, NC State University
[2]Coastal Hydraulics Laboratory, U.S. Army Engineer Research and Development Center, Duck, NC, USA

ADCIRC Users Group Meeting
June 2023

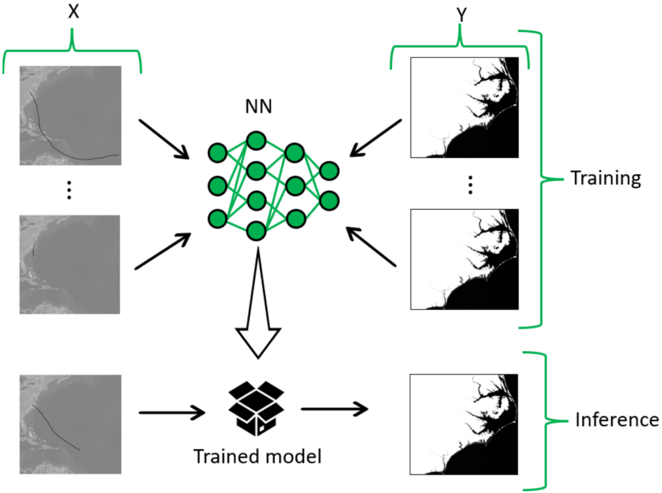
Agencia Nacional de Investigación y Desarrollo
Ministerio de Ciencia, Tecnología, Conocimiento e Innovación
Gobierno de Chile

COASTAL RESILIENCE CENTER

Motivation: Real-time forecasting uncertainty



Motivation: Accelerate Predictions with Deep Learning



What is new?

- Astronomical tides
- Prediction of maps

Motivation: Proposed Workflow

Step 1

Data Collection

- Dataset of synthetic tropical cyclones
- MDA

Step 2

Training Library

- Hydrodynamic modeling
- Downscaling

Step 3

Deep Learning

- RNN for time
- 2D CNN for space
- Train/Test/Validate

Step 4

Final Validation

- Compare NN against hindcasts

Motivation: Proposed Workflow

Step 1

Data Collection

- Dataset of synthetic tropical cyclones
- MDA

Step 2

Training Library

- Hydrodynamic modeling
- Downscaling

Step 3

Deep Learning

- RNN for time
- 2D CNN for space
- Train/Test/Validate

Step 4

Final Validation

- Compare NN against hindcasts

Step 1

Data Collection

- Dataset of synthetic tropical cyclones
- MDA

Step 2

Training Library

- Hydrodynamic modeling
- Downscaling

Step 3

Deep Learning

- RNN for time
- 2D CNN for space
- Train/Test/Validate

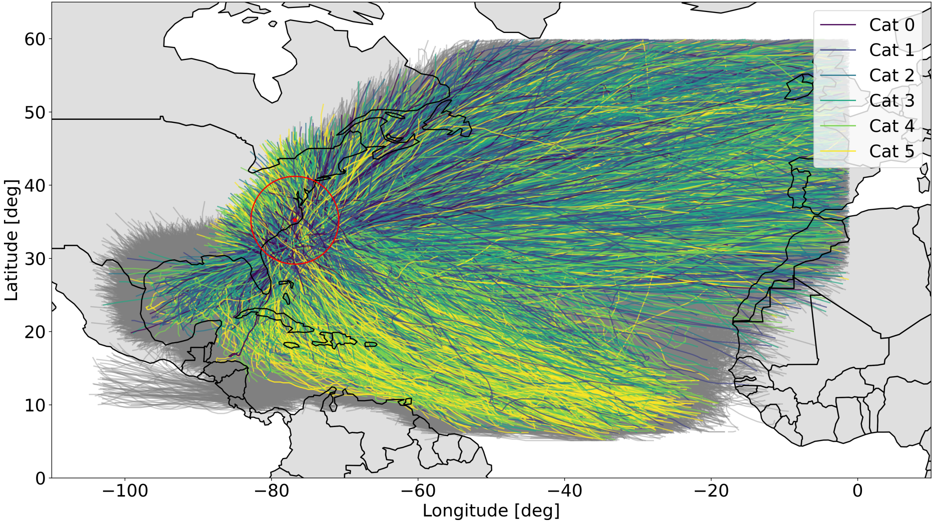
Step 4

Final Validation

- Compare NN against hindcasts

Step 1: Dataset of synthetic tropical cyclones

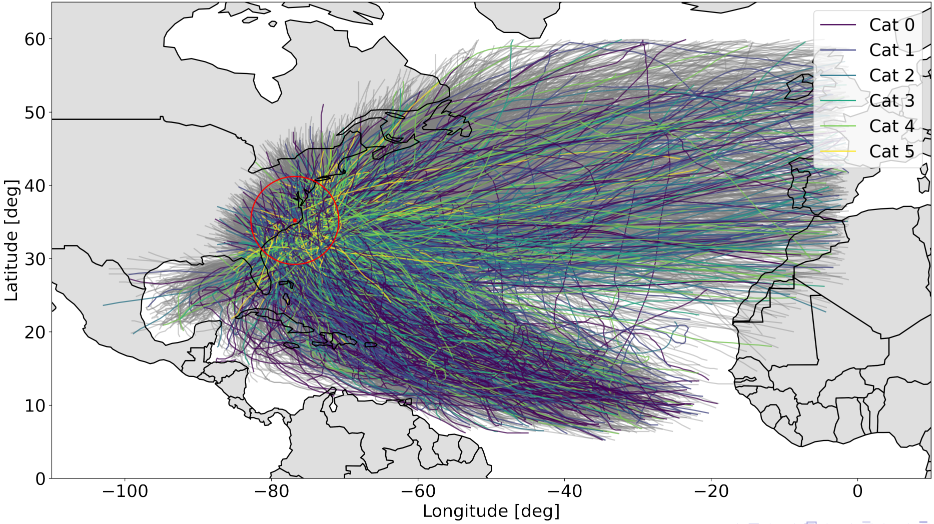
Filter storms by duration and proximity to NC



Bloemendaal et al. (2020)

Step 1: Maximum Dissimilarity Algorithm

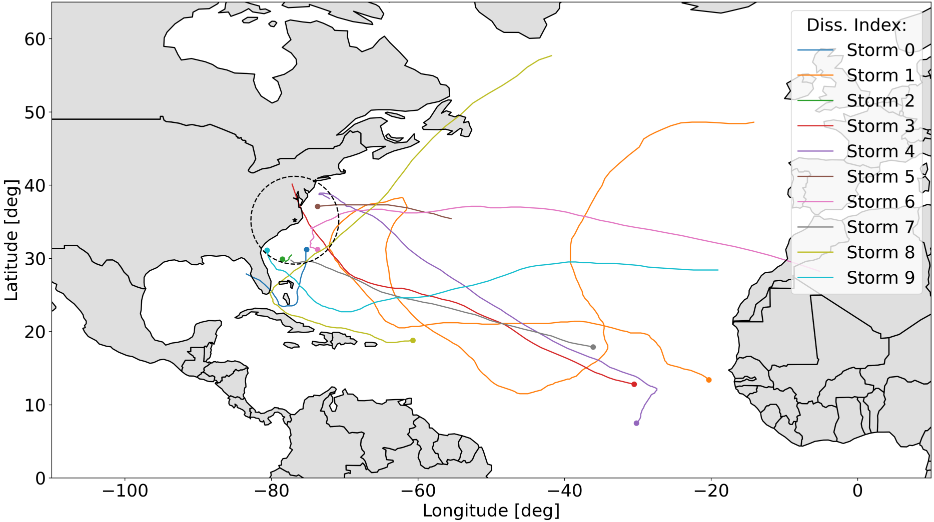
Selection of a representative set of storms – 1000 most dissimilar



Camus et al. (2011)

Step 1: Maximum Dissimilarity Algorithm

Selection of a representative set of storms – 10 most dissimilar



Step 1

Data Collection

- Dataset of synthetic tropical cyclones
- MDA

Step 2

Training Library

- Hydrodynamic modeling
- Downscaling

Step 3

Deep Learning

- RNN for time
- 2D CNN for space
- Train/Test/Validate

Step 4

Final Validation

- Compare NN against hindcasts

Step 2: Hydrodynamic Modeling

Setup of ADCIRC simulations

Fort.13 & **Fort.14**: same for all (NC9[a])

Fort.15: Variable given tides

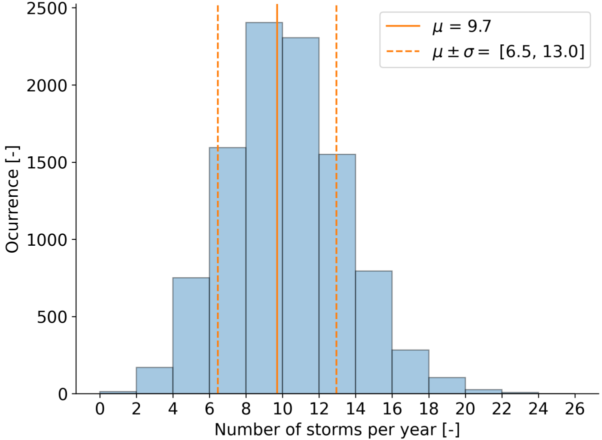
Random date $\Longrightarrow$ random tide

- Find a yearly range of storms
- Iterate from 2020 to past
  - Month is given
  - Day is chosen randomly
  - Remove storms and continue

Fort.22: Symmetric Holland Model

No need to compute extra parameters

- Coords, WS, P, and RMW



[a] Blanton and Luettich (2008)

Step 2: Hydrodynamic Modeling

Postprocessing ADCIRC simulations – Log files

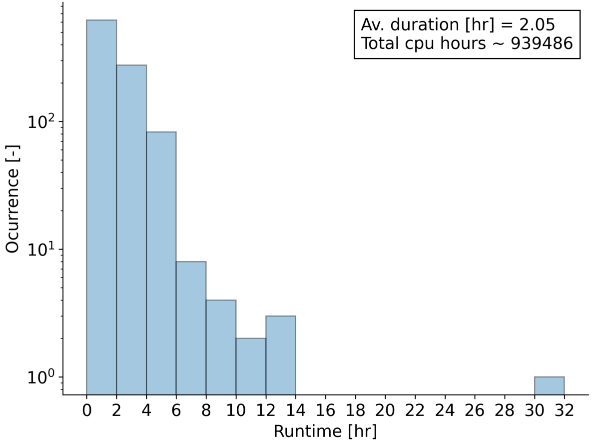
HPC systems

- NCSU Hazel
- Purdue Anvil

Automated log files reading

- Run to completion or fail
  - Type or error
- Runtime
- CPU hours

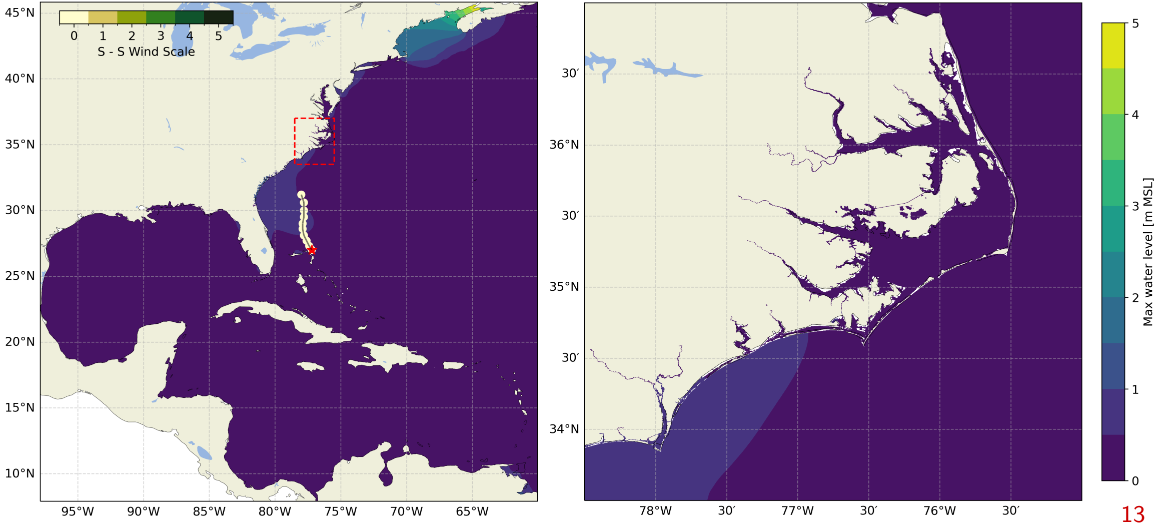
$\approx$ 1M CPU hours and 2 months



Step 2: Hydrodynamic Modeling

Postprocessing ADCIRC simulations – 2D Maps

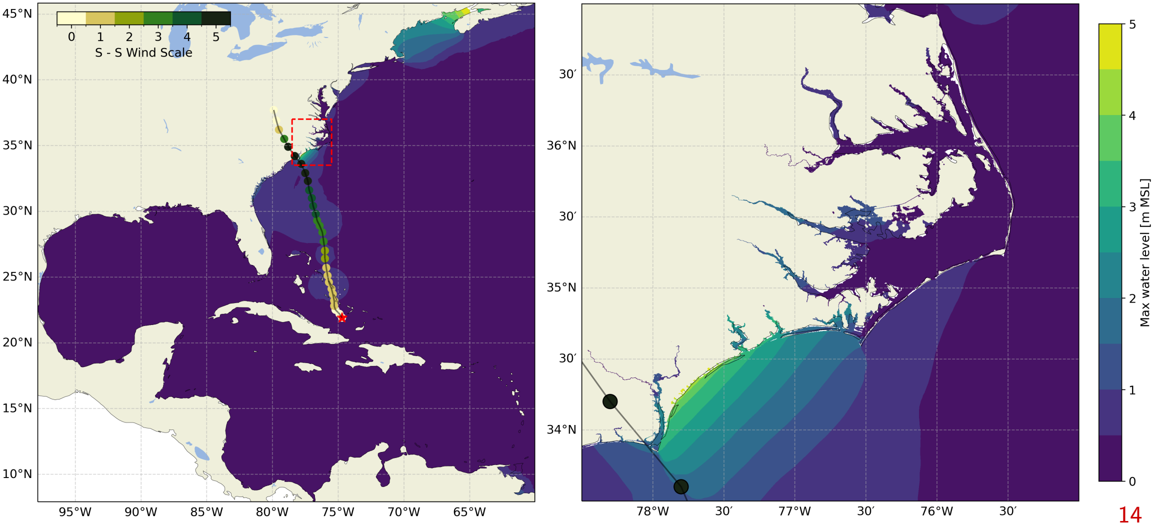
"Boring storm"



Step 2: Hydrodynamic Modeling

Postprocessing ADCIRC simulations – 2D Maps

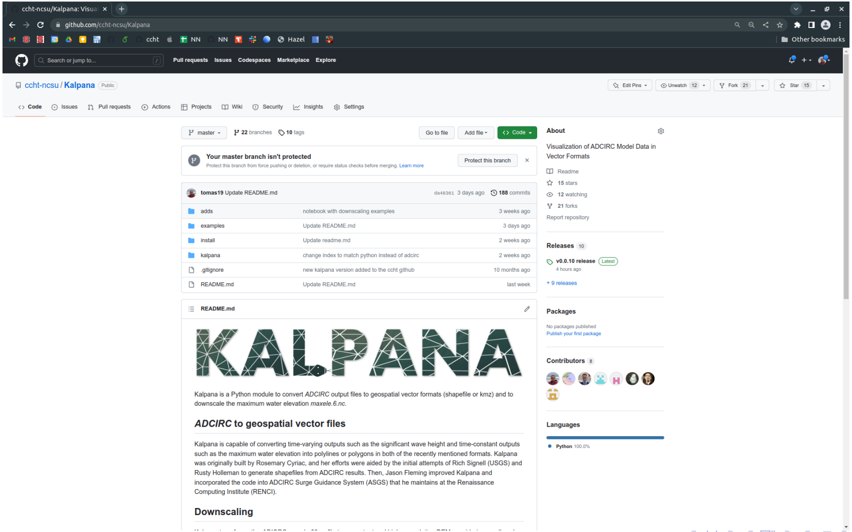
Strong storm



"An A.I.-Generated Picture Won an Art Prize. Artists Aren't Happy", The New York Times.

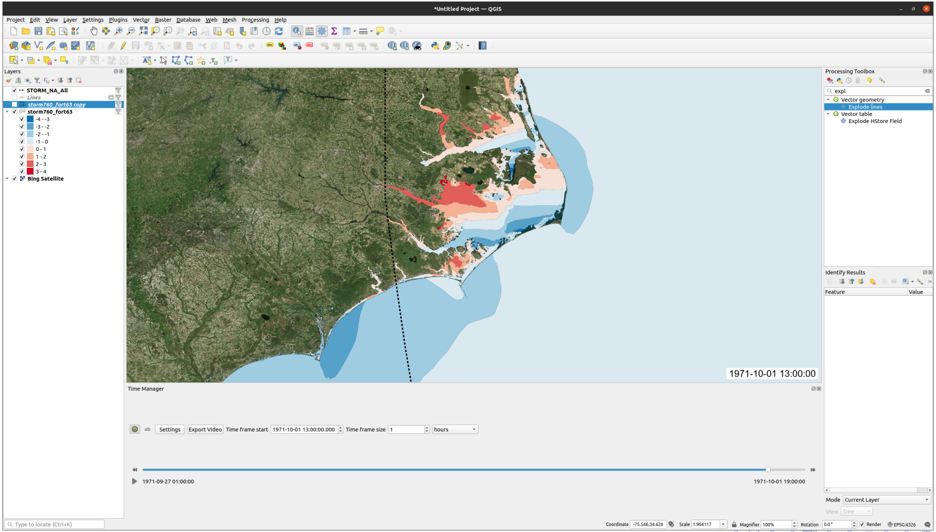
Kalpana

Software to visualize and downscale ADCIRC



Kalpana

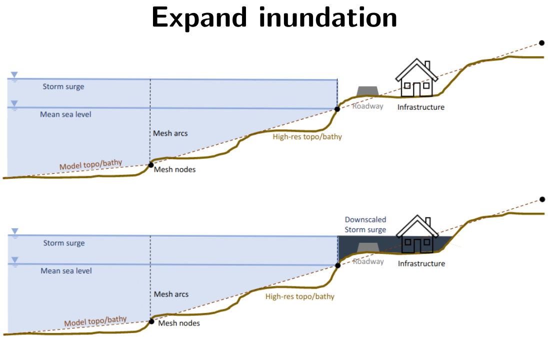
Kalpana allows easy visualization in GIS – netCDF to shapefile

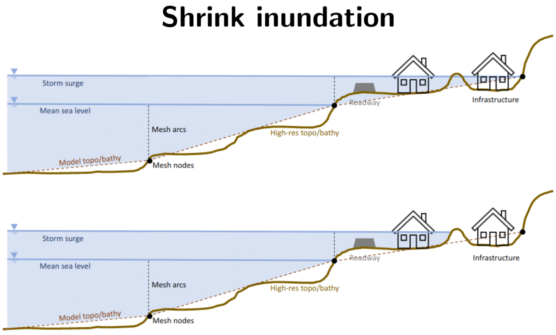


Kalpana

Static Downscaling Method

Use of a high-resolution topo DEM to increase ADCIRC resolution and to expand or shrink the inundation extent[1]





Rucker et al. (2021)

Kalpana

New version of the code

Main changes:

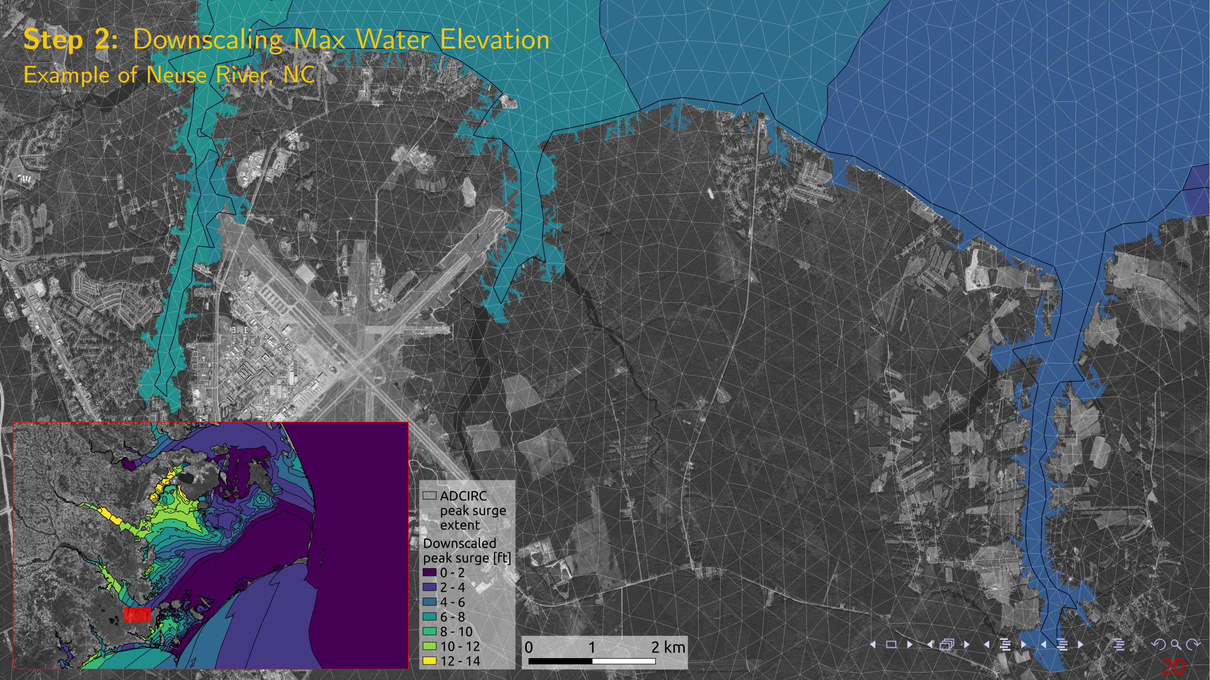
- **From Python 2.7 to 3.9 (or higher)**
- Pandas, GeoPandas, Shapely, rioxarray and Dask
- Some parallelization
- From 1 single script to 2 modules
- All functions available on GitHub and documented.

Mayor improvements:

- Preprocess to accelerate the downscaling
- From 45 to 7 minutes on a 15m res DEM of NC
- **Less user-defined inputs**
- **Inputs related to mesh**

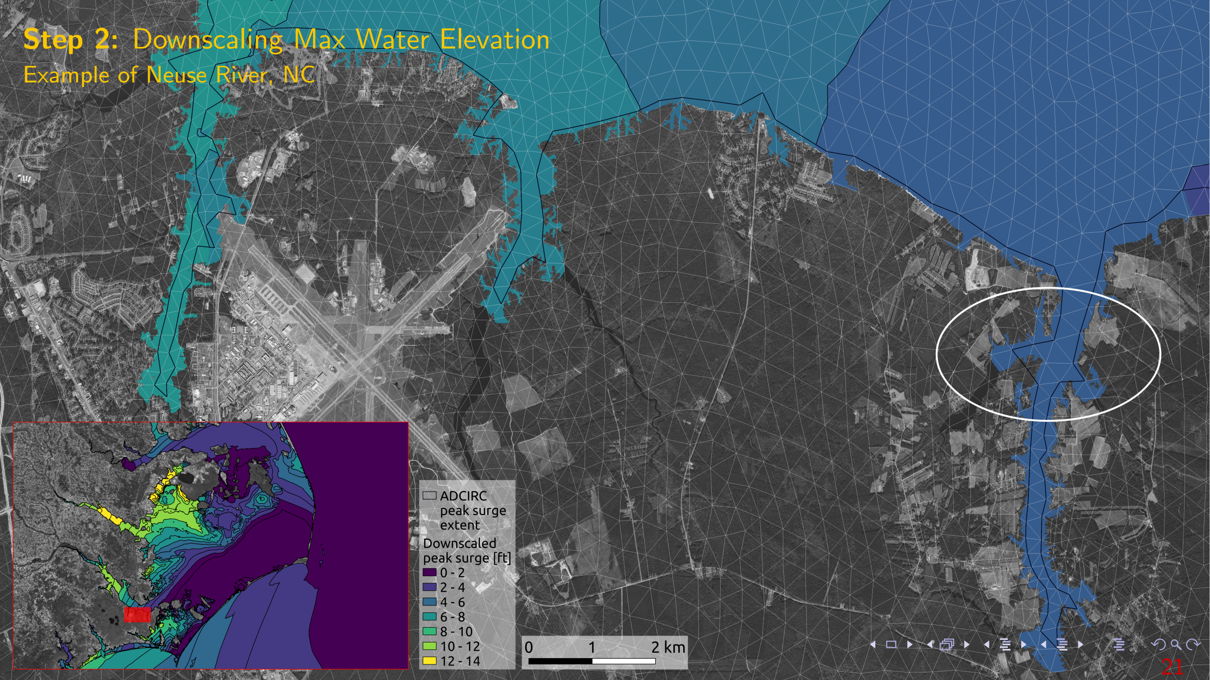
Step 2: Downscaling Max Water Elevation

Example of Neuse River, NC



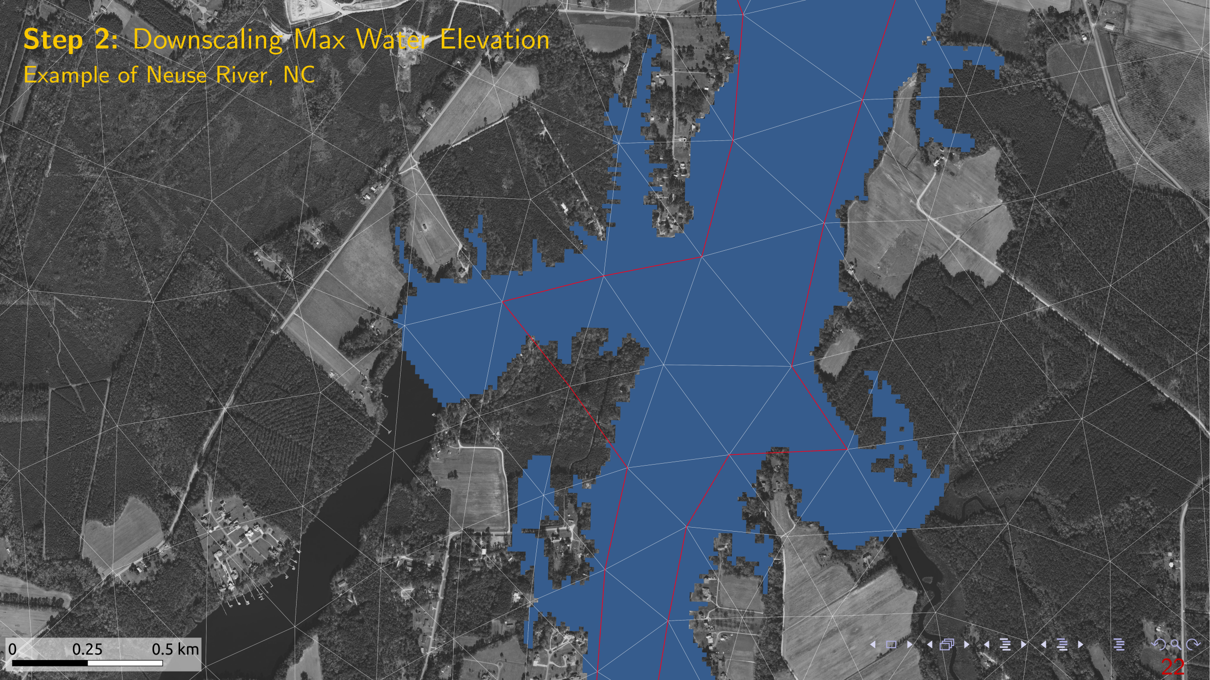
Step 2: Downscaling Max Water Elevation

Example of Neuse River, NC



Step 2: Downscaling Max Water Elevation

Example of Neuse River, NC



Step 2: Downscaling Max Water Elevation

Downscaled DEM to greyscale image

Step 1

Data Collection

- Dataset of synthetic tropical cyclones
- MDA

Step 2

Training Library

- Hydrodynamic modeling
- Downscaling

Step 3

Deep Learning

- RNN for time
- 2D CNN for space
- Train/Test/Validate

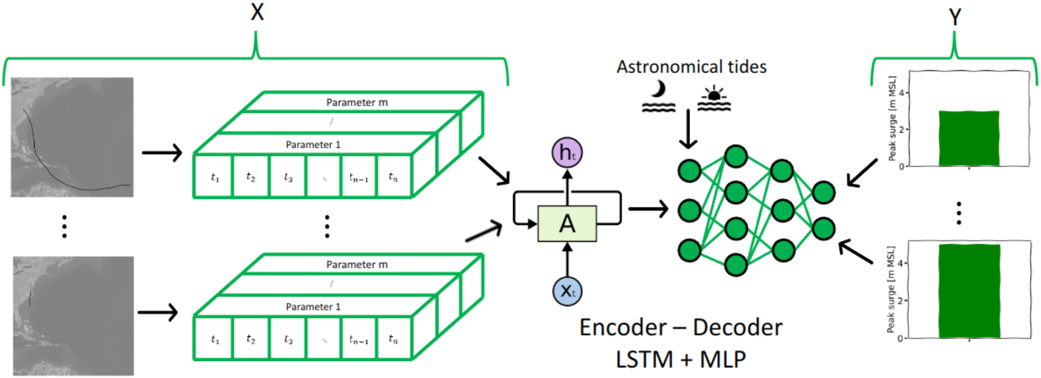
Step 4

Final Validation

- Compare NN against hindcasts

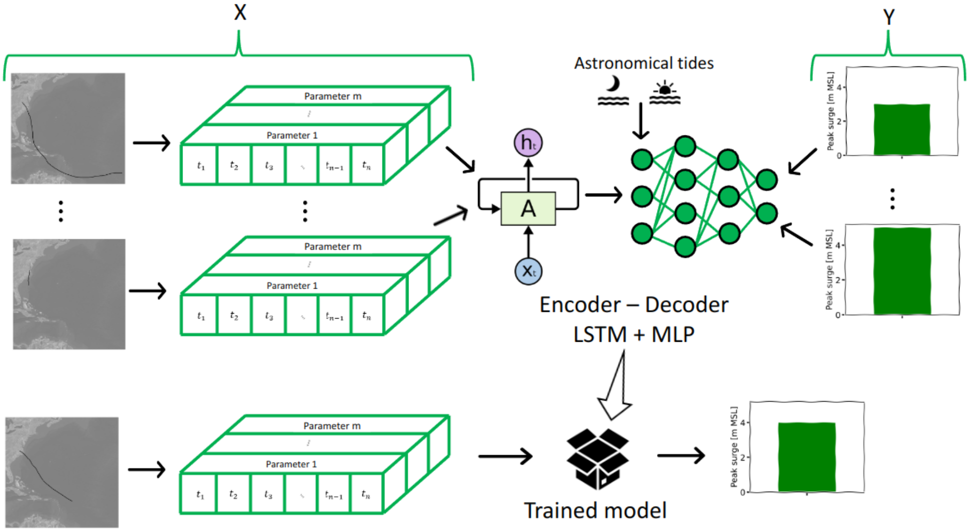
Step 3: Neural Network Development

Long-Short Term Memory (LSTM) + Multi-Layer Perceptron (MLP) to predict peak surge at a single point



Step 3: Neural Network Development

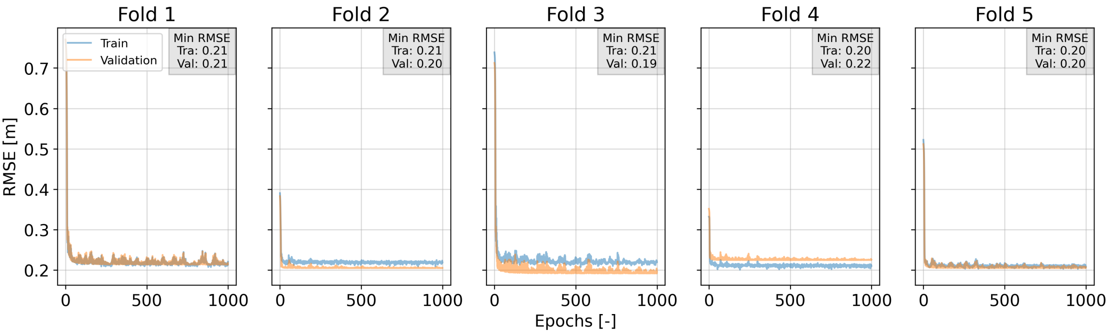
Long-Short Term Memory (LSTM) + Multi-Layer Perceptron (MLP) to predict peak surge at a single point



Step 3: Neural Network Predictions

RMSE for training and validation is close to 20 cm

Prediction at Duck (NC) NOAA tide gauge



Proposed Workflow

Step 1

Data Collection

- Dataset of synthetic tropical cyclones
- MDA

Step 2

Training Library

- Hydrodynamic modeling
- Downscaling

Step 3

Deep Learning

- RNN for time
- 2D CNN for space
- Train/Test/Validate

Step 4

Final Validation

- Compare NN against hindcasts

Summary and Future Work

NN to predict high-res maps of storm-driven coastal flooding

- Selected a subset of tracks that represents the max and avg of the tropical cyclone conditions in NC
- Simulated 1000 storms with ADCIRC using $\approx$ 1M cpu hours
- Downscaled the peak surge output to produce high and constant-resolution maps
- Implemented a NN based on an LSTM and MLP layers to predict peak surge at single points

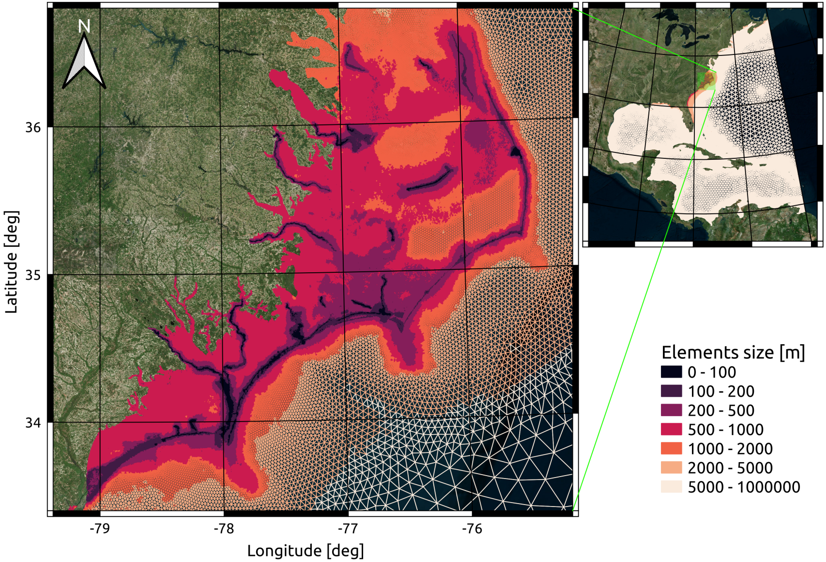
Coming soon:

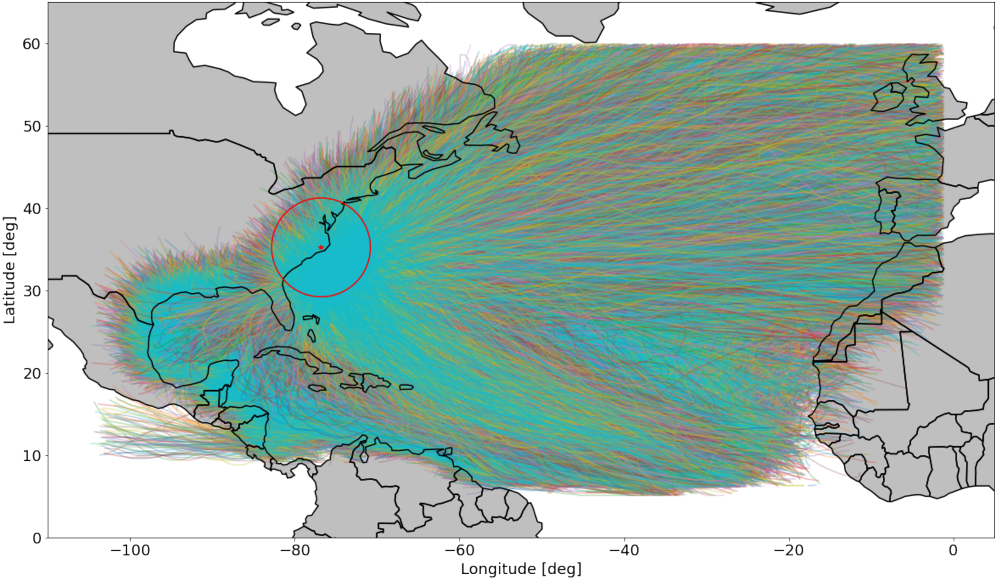
- Define a new set of storms to simulate
- Simulate 1,000 more storms with a mesh developed by Johnathan Woodruff
- **Neural network development**
  - 2D CNN to generate maps

Thank you – Gracias!

github.com/ccht-ncsu/Kalpana

Motivation: Resolution (Accuracy) vs Model RunTime




60
50
40
30
20
10
0
Latitude [deg]
−100
−80
−60
−40
−20
0
Longitude [deg]

Tracks parametrization

Track ID	Time step
Storm 1	1
$\vdots$	$\vdots$
Storm 1	p
Storm 2	1
$\vdots$	$\vdots$
Storm 2	q
$\vdots$	$\vdots$
Storm N	1
$\vdots$	$\vdots$
Storm N	r

Track ID	Param 1	...	Param 2
Storm 1	P_{11}	$\dots$	P_{1m}
Storm 2	P_{21}	$\dots$	P_{2m}
$\vdots$	$\vdots$	$\ddots$	$\vdots$
Storm N	P_{N1}	$\dots$	P_{Nm}

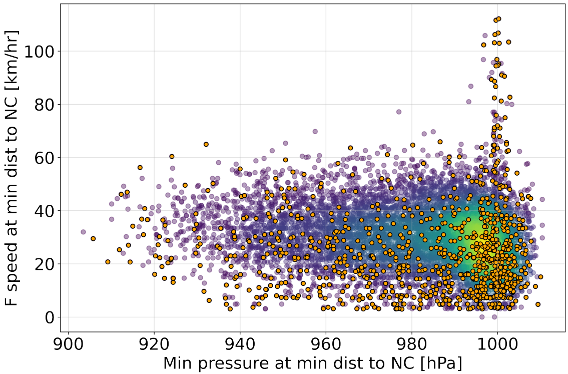
A key challenge is how to compute the distance for circular variables

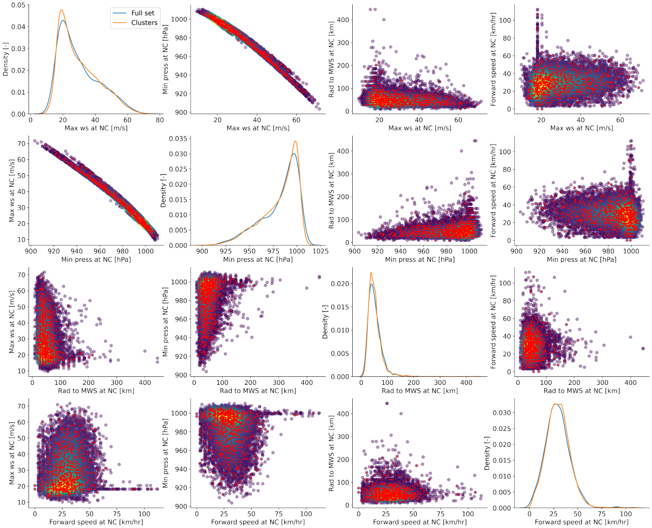
"Normal variables":
Euclidean distance

$$d = \sqrt{(v_1 - v_2)^2}$$

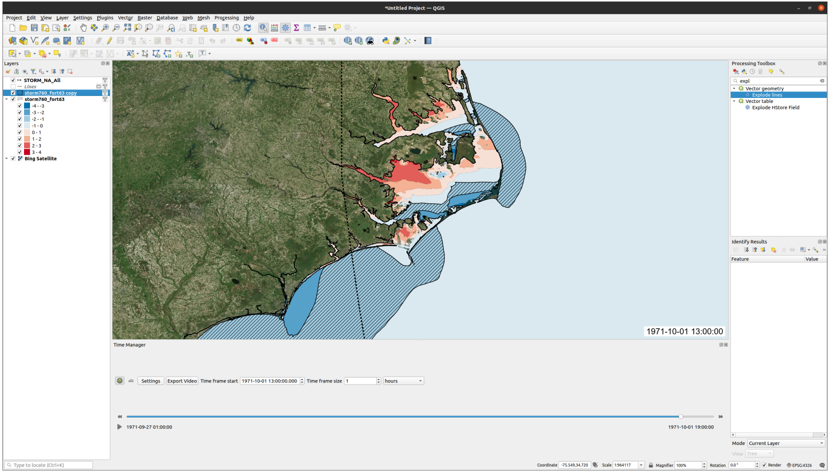
Circular variables: metric by Camus et al. (2011)

$$d = \sqrt{min(|d_1 - d_2|, 2 - |d_1 - d_2|)^2}$$



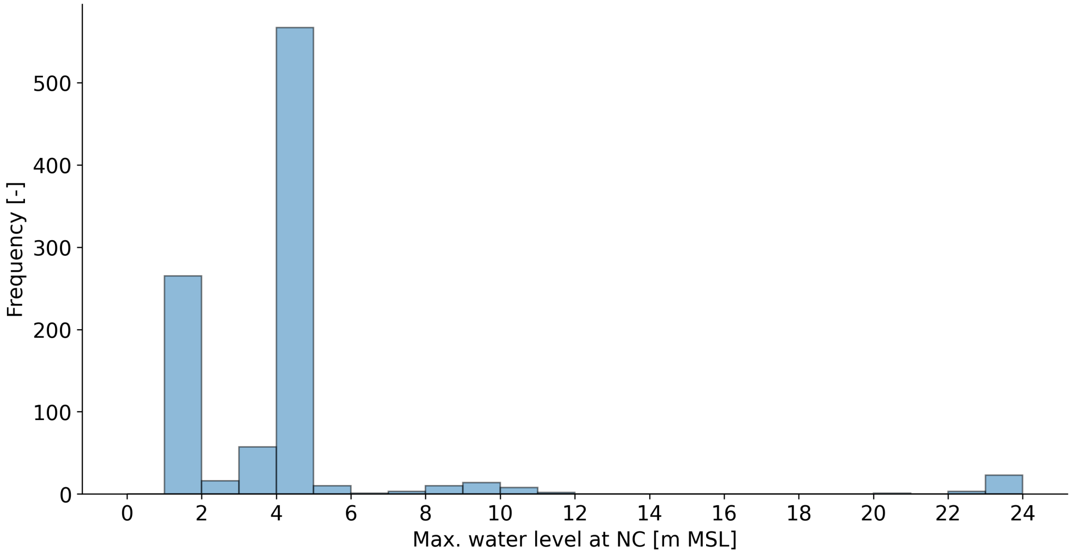

Full set
Clusters
Density [-]
Max ws at NC [m/s]
Min press at NC [hPa]
Rad to MWS at NC [km]
Forward speed at NC [km/hr]

Easy visualization of ADCIRC outputs in QGIS



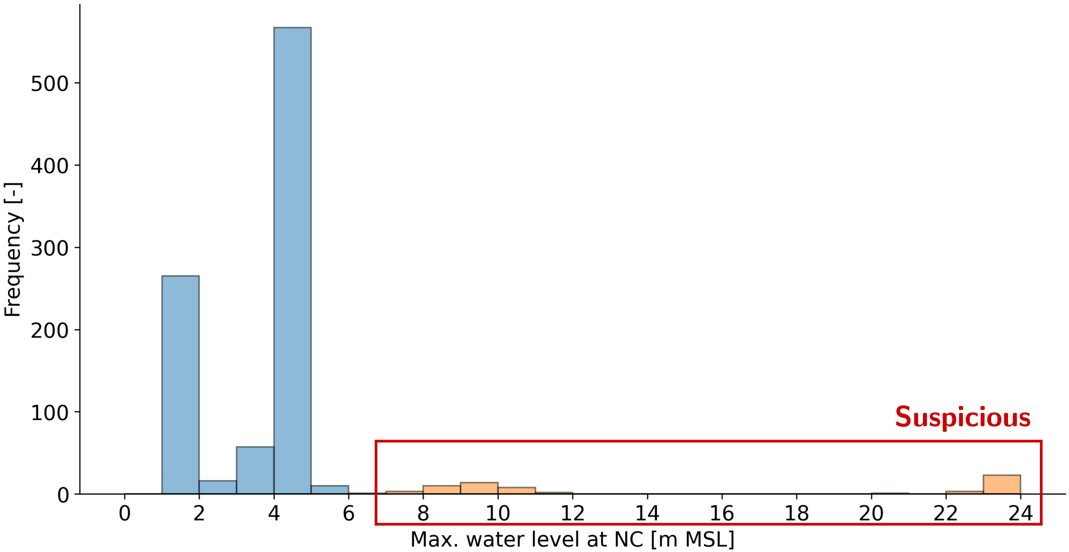
Step 2: Hydrodynamic Modeling

Postprocessing ADCIRC simulations – Peak Surge at NC



Step 2: Hydrodynamic Modeling

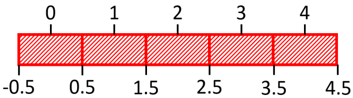
Postprocessing ADCIRC simulations – Peak Surge at NC



Step 2: Downscaling Max Water Elevation

Downscaled DEM to greyscale image

Water levels are binned: 1, 2, 3, 4, ...



Water levels are discrete now, and the values given by the input downscaling levels

If input levels can be scaled to integers $\implies$ grayscale png (only if max level is below 255)

Neural Network Architecture