How to create an inundation/flood hazard map

Kelsey L. Ruckert
Earth and Environmental Systems Institute, The Pennsylvania State University, University Park, Pennsylvania, USA
Email: klr324@psu.edu

Please read the instructions CAREFULLY, as each step in this tutorial is dependent on the previous step. Please take your time in running the tutorial and note that this tutorial was written under ArcMAP version 10.3.1.

Introduction

Inundation maps can be a helpful part of understanding coastal inundation issues, assessing flood adaptation strategies, and floodplain management. Today, the Federal Emergency Management Agency (FEMA) creates inundation maps called FIRMS (Flood insurance rate maps). These maps can be used in decision-making. For example, FIRMS were used in the proposal of the BIG U (being built at the Battery in New York City) in the aftermath of Hurricane Sandy. This tutorial will show how to create the flood risk maps in Ruckert et al. [1] using ArcGIS.

Following this tutorial you will create an inundation map for the San Francisco Bay area in California. To do this you will obtain Digital Elevation Models (DEMs) for the San Francisco Bay area and use previously calculated storm surge heights. These calculated storm surge heights are the 100-year storm surge accounting for several different scenarios of sea-level rise in the year 2100. This process will require you to:

- integrate raster and vector data layers,
- integrate python with ArcGIS to increase reproducibility,
- calculate the area inundated by the different flood levels for specific counties.

Procedure

Obtaining the data files

A description of how to obtain the data files is provided below, however, all files can be downloaded at https://download.scrim.psu.edu/Ruckert_etal_San_Francisco/.

In this tutorial, you will use 1/9 arc-second (roughly 3 meter) topobathy (merged topographic and bathymetric surface) DEMs (digital elevation models). DEMs are used to record the Earth's physical terrain surface. Each grid cell in a DEM has an elevation value. The elevation data serve as the base layer for mapping inundation. The DEMs can be downloaded from the USGS National Elevation dataset.

In the Product Search Filter click on the subfilter: **1/9 arc-second DEM**, zoom into the San Francisco area, click **Find Products**, and download the following DEMs:

- Ned19_n37x75_w122x25_ca_sanfrancisco_topobathy_2010
- Ned19_n37x75_w122x50_ca_sanfrancisco_topobathy_2010
- Ned19_n37x75_w122x75_ca_sanfrancisco_topobathy_2010
- Ned19_n38x00_w122x25_ca_sanfrancisco_topobathy_2010
- Ned19_n38x00_w122x50_ca_sanfrancisco_topobathy_2010
- Ned19_n38x00_w122x75_ca_sanfrancisco_topobathy_2010.

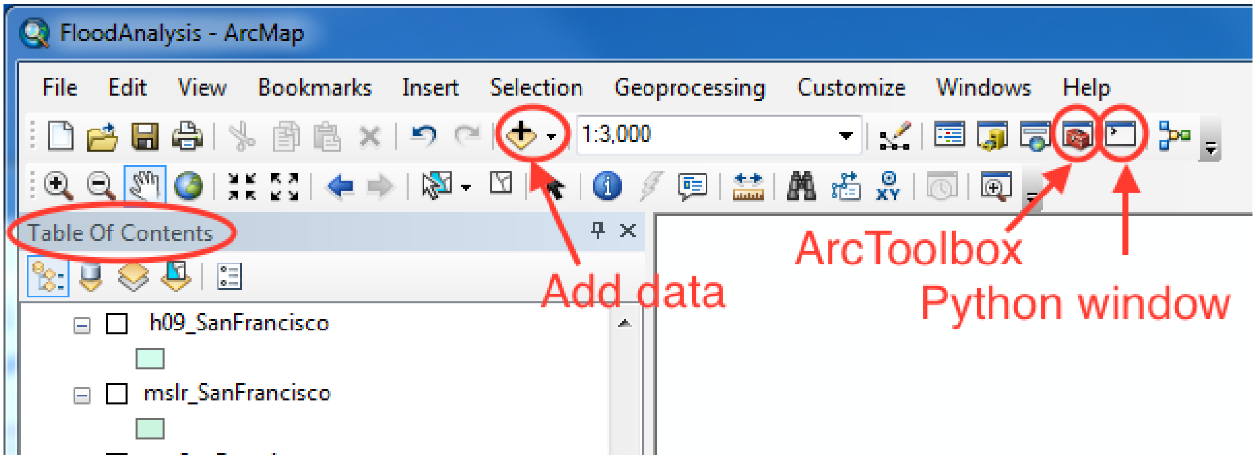
Files including counties, states, and water area are shapefiles that come from the U.S Census Bureau TIGER/Line Shapefiles. On the server, go to Download > Web Interface and select the year 2015 or the latest year. Then download the national file for the layer type: **Counties (and equivalent)** and also the layer type: **States (and equivalent)**. Additionally download the water data by selecting Water > Area Hydrography > California > ... and select the following counties:

- Alameda
- Contra Costa
- Marin County
- San Francisco County
- San Mateo County.

The files used to make up the satellite images shown in the figures of Ruckert et al. [1] are U.S.G.S. Imagery-1 meter (NAIP) images. On the server, zoom into the San Francisco area, click on Imagery - 1 meter (NAIP) > Find Products, and download the files in that area.

Setup the workspace

The point of this tutorial is to create an inundation map. In order to do this effectively, the data must be in an appropriate coordinate system that preserves area in the location of interest (i.e., NAD 83 California Zone 3 for San Francisco Bay). First, **open ArcMap** under ArcGIS and choose New Maps > Blank Map. This will give you an empty ArcMap screen. From the Toolbar, go to Geoprocessing > Environments > Workspace. Make sure that both the current workspace and scratch workspace are set to the same location as the DEM and shapefiles. Now that your map and workspace are setup, it is a good idea to save your map to the current workspace. Doing so will help setup the environment when using the python window.



Next, re-project the county, state, and water shapefiles according to the NAD 83 California Zone 3 projection. To help increase reproducibility, we will use the python window. In the python window type the following code (note that spacing will be messed up if you copy and paste directly from this document). A separate file (`Flood_Analysis_Python.py`) including all of the python code is provided along with all the other data needed at https://download.scrim.psu.edu/Ruckert_etal_San_Francisco/. The following code block will setup the environment, save the shapefiles with the NAD 83 California Zone 3 projection, and load the shapefiles to the map.

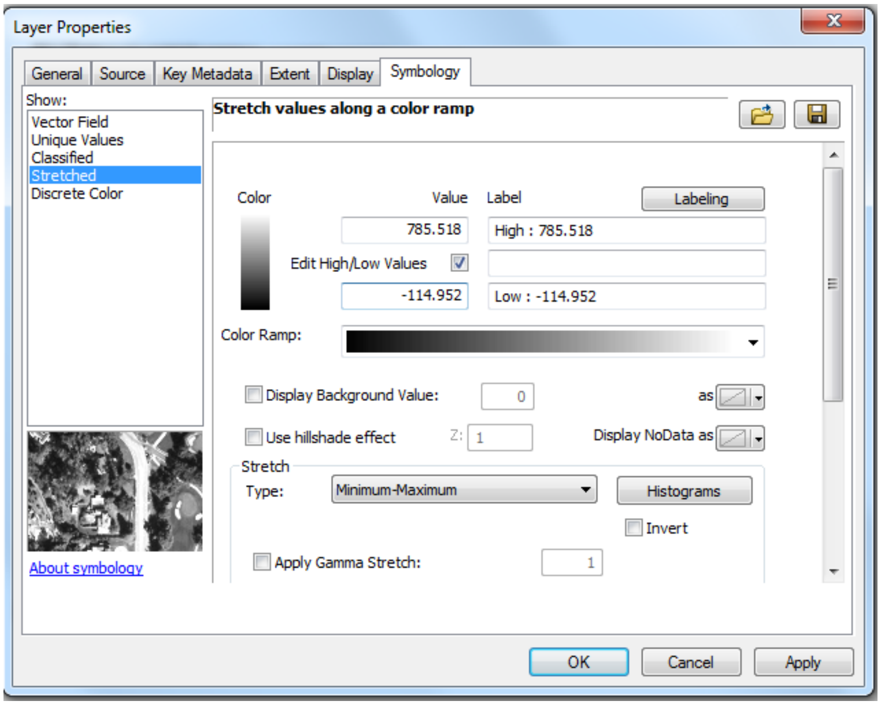
```
# ===========================================================================
# Import system modules
import arcpy
import os
from arcpy import env

# Set current working directory (cwd)
cwd = os.getcwd()
# ===========================================================================

# Input data is in the NAD 1983 coordinate system
# Create a spatial reference object for the output coordinate system: NAD 83
# California Zone 3
out_coordinate_system = arcpy.SpatialReference('NAD 1983 (2011) StatePlane
    California III FIPS 0403 (Meters)')

# Project the counties shapefile
arcpy.Project_management(os.path.join(cwd, "Data/tl_2015_us_county/tl_2015_us_county.shp"),
    os.path.join(cwd, "Pre_analyzed_files/counties_Nad83z3.shp"), out_coordinate_system)

# Project the states shapefile
arcpy.Project_management(os.path.join(cwd, "Data/tl_2015_us_state/tl_2015_us_state.shp"),
    os.path.join(cwd, "Pre_analyzed_files/state_Nad83z3.shp"), out_coordinate_system)

# Project the sub-counties shapefile
arcpy.Project_management(os.path.join(cwd, "Data/tl_2015_06_cousub/tl_2015_06_cousub.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Subcounties_Nad83z3.shp"), out_coordinate_system)
```

```

# Merge the water shapefiles
arcpy.Merge_management([os.path.join(cwd,
    "Data/tl_2015_06001_areawater/tl_2015_06001_areawater.shp"), os.path.join(cwd,
    "Data/tl_2015_06013_areawater/tl_2015_06013_areawater.shp"), os.path.join(cwd,
    "Data/tl_2015_06041_areawater/tl_2015_06041_areawater.shp"), os.path.join(cwd,
    "Data/tl_2015_06075_areawater/tl_2015_06075_areawater.shp"), os.path.join(cwd,
    "Data/tl_2015_06081_areawater/tl_2015_06081_areawater.shp")], os.path.join(cwd,
    "Pre_analyzed_files/Water_area.shp"))

# Project the merged water shapefile
arcpy.Project_management(os.path.join(cwd, "Pre_analyzed_files/Water_area.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Water_Nad83z3.shp"), out_coordinate_system)
```

Setup the DEMs

Unfortunately, the raster analysis will be done manually instead of using the python window. However, we discuss the steps in detail below. First, load the DEM layers by clicking **Add data** from the Toolbar, navigate to the correct directory, and upload the **.img** file. If prompted change the “Pyramid resampling technique” from **Nearest Neighbor** to **Cubic Convolution** and click “yes”. If a window pops up in regards to the coordinate system, then close the window. This issue will be addressed later.

After loading the DEM files, note the highest and lowest elevation value. These values should be 785.518 and -114.952. Using these values, modify the elevation range in each DEM file to reflect this range in elevation. In the Table of Contents, right click one of the DEM layers, click on **Properties**, and click on **Symbology**. In this tab, we can reset the elevations manually. Under the **Stretch Type:** choose **Minimum-Maximum**. Check the **Edit high/low Values** and edit these values to match the highest and lowest elevation from the DEM layer. Click **Apply**, **ok**, and then repeat this step for all the DEM layers.



Stretch values along a color ramp

Color | Value | Label
785.518 | High : 785.518
Edit High/Low Values ☑
-114.952 | Low : -114.952

Color Ramp:

□ Display Background Value: 0 as

□ Use hillshade effect Z: 1 Display NoData as

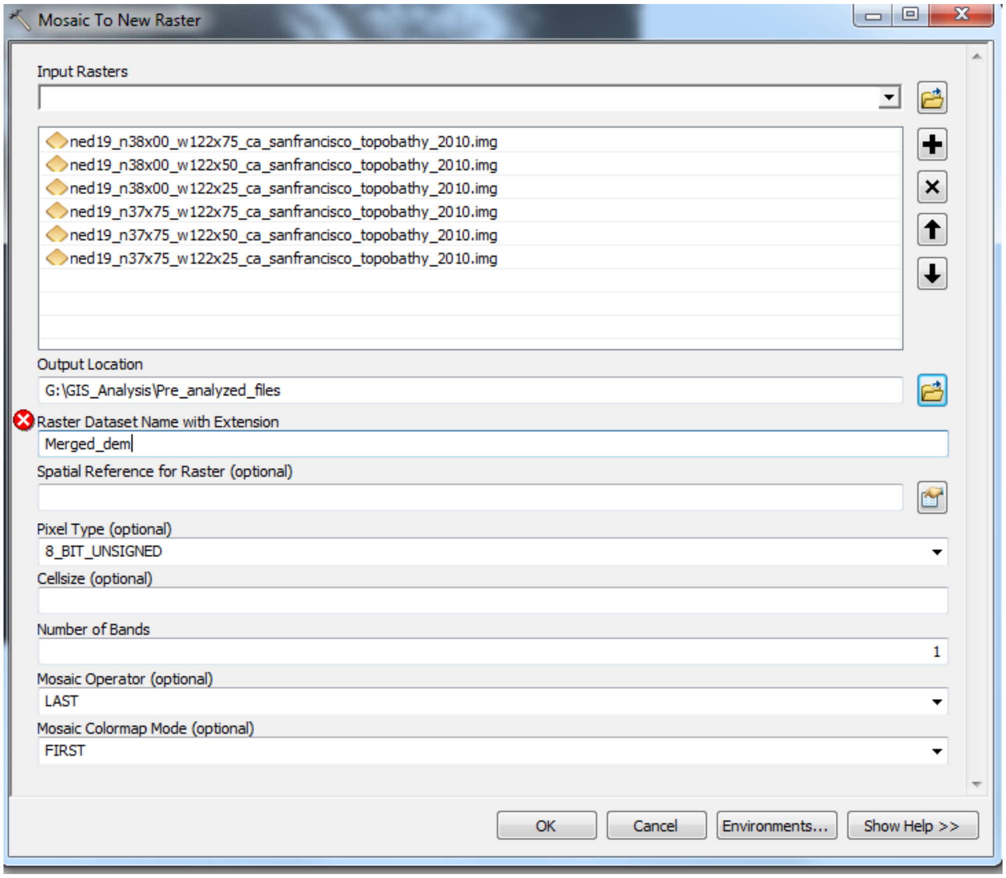
Stretch
Type: Minimum-Maximum

□ Invert

□ Apply Gamma Stretch: 1

Mosaic

To do the analysis it will be easier to work with one layer rather than 6 DEMs. We can merge the multiple raster datasets into one using the **Mosaic To New Raster** tool. Navigate in the ArcToolbox to Data Management Tools » Raster » Raster Dataset Toolset » Mosaic To New Raster. For the inputs, add all the DEM layers to merge together. Then set the output file path and save the merged raster with a logical name (i.e., Merged_dem; no extension is needed). Set the number of bands to 1 and leave everything else as is. This may take a few minutes after clicking **ok**.



Mosaic To New Raster

Input Rasters

- ned19_n38x00_w122x75_ca_sanfrancisco_topobathy_2010.img
- ned19_n38x00_w122x50_ca_sanfrancisco_topobathy_2010.img
- ned19_n38x00_w122x25_ca_sanfrancisco_topobathy_2010.img
- ned19_n37x75_w122x75_ca_sanfrancisco_topobathy_2010.img
- ned19_n37x75_w122x50_ca_sanfrancisco_topobathy_2010.img
- ned19_n37x75_w122x25_ca_sanfrancisco_topobathy_2010.img

Output Location: G:\GIS_Analysis\Pre_analyzed_files

Raster Dataset Name with Extension: Merged_dem

Spatial Reference for Raster (optional):

Pixel Type (optional): 8_BIT_UNSIGNED

Cellsize (optional):

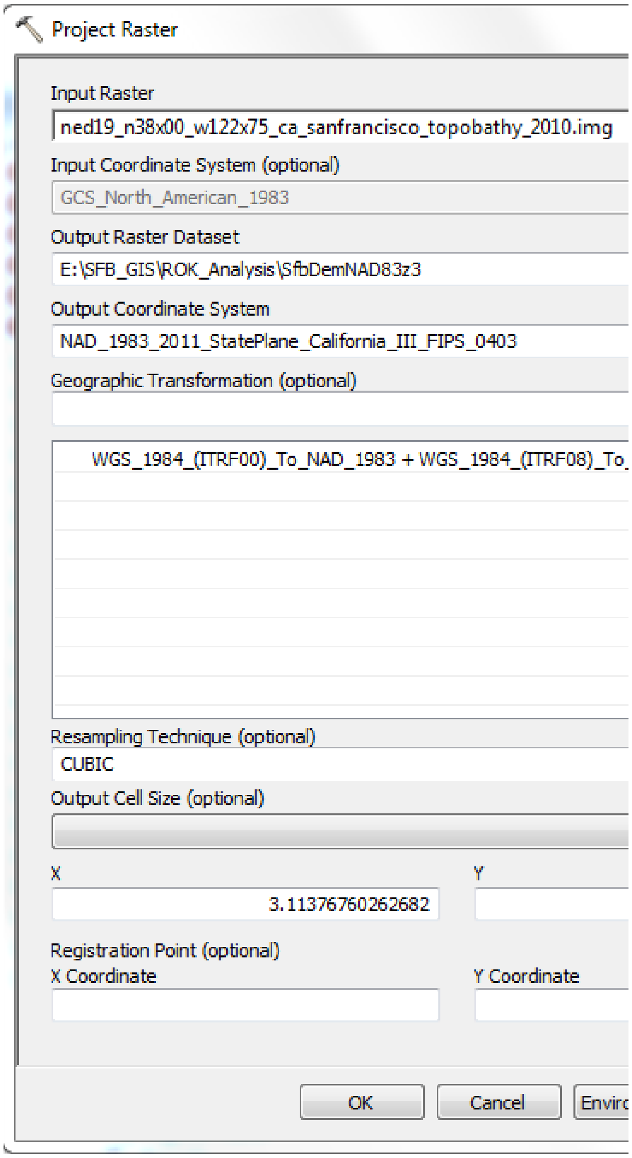
Number of Bands: 1

Mosaic Operator (optional): LAST

Mosaic Colormap Mode (optional): FIRST

Re-project the DEM

Re-project the mosaicked DEM into the same coordinate system as our other layers. Navigate in the ArcToolbox to Data Management Tools » Projections and Transformations » Raster » Project Raster. For the "Input Raster" select the merged DEM (the second field will fill in automatically). In the Output raster dataset field, navigate to your folder and save your new raster with a logical name of 13 or less characters (for example, DEM_NAD83z3). In the output coordinate system, click the button to the right of the field and choose **NAD 1983 (2011) StatePlane California III FIPS 0403 (Meters)**. Choose 'CUBIC' convolution for the resampling technique. Leave the output size as the default. Click **ok** and remove the unprojected DEM layers from your project.



Project Raster

Input Raster
ned19_n38x00_w122x75_ca_sanfrancisco_topobathy_2010.img

Input Coordinate System (optional)
GCS_North_American_1983

Output Raster Dataset
E:\SFB_GIS\ROK_Analysis\SfbDemNAD83z3

Output Coordinate System
NAD_1983_2011_StatePlane_California_III_FIPS_0403

Geographic Transformation (optional)

WGS_1984_(ITRF00)_To_NAD_1983 + WGS_1984_(ITRF08)_To_

Resampling Technique (optional)
CUBIC

Output Cell Size (optional)

X
3.11376760262682

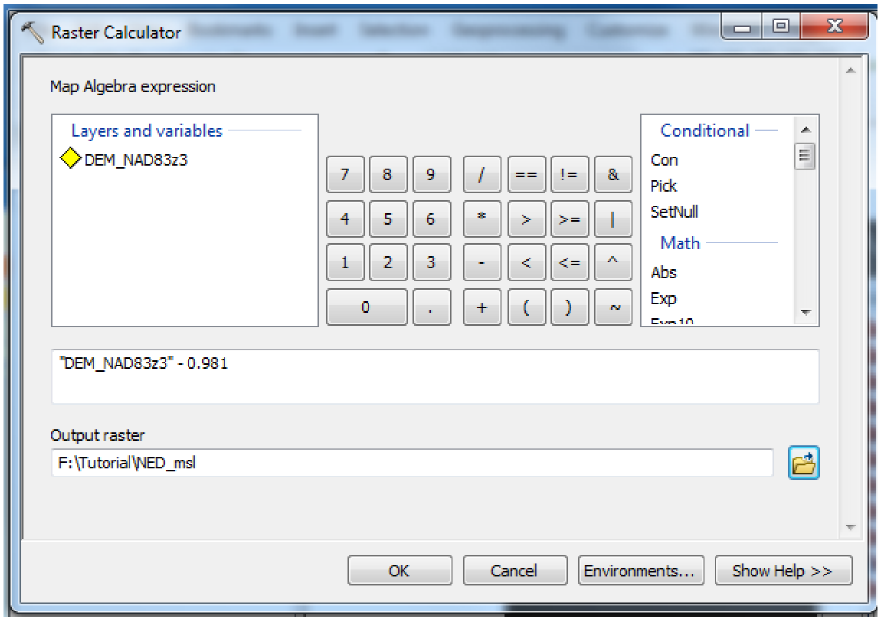
Y

Registration Point (optional)
X Coordinate

Y Coordinate

Adjusting the Elevation to Mean Sea-level

The DEMs from the USGS NED are referenced to the North American Datum of 1988 (NAVD88). For our flood maps, we want to reference mean sea level as 0 m rather than NAVD88. According to the San Francisco tide gauge (#9414290) the NAVD88 is at 1.804 m while mean sea level occurs at 2.773 m. To adjust the mosaic DEM to mean sea level, subtract the raster by the difference between mean sea level and NAVD88 (0.981 m). Navigate in the ArcToolbox to Spatial Analyst Tools ⟩ Map Algebra ⟩ Raster Calculator. For input raster, add the projected DEM, select the **subtraction operator**, and type **0.981**. Save the output with a sufficient name (i.e., NED_msl) and click **ok**.



Map Algebra expression

"DEM_NAD83z3" - 0.981

Output raster

F:\Tutorial\NED_msl

Applying the elevation data

We are interested in the area that is prone to flooding from certain flood heights. To find the areas at risk, we have to analyze the elevation data and output the results.

The first step is the identify which parts of the elevation data that are between mean sea level and the baseline storm surge or future flood height (storm surge and sea-level rise). Here, we will use the flood height estimates for the year 2100 in Ruckert et al. [1].

- Baseline 100-year storm surge = 1.59 m
- 100-year flood height accounting for mean sea-level rise = 2.21 m
- 100-year flood height accounting for Heberger et al. (2009) estimate = 2.42 m
- 100-year flood height accounting for uncertainty in sea-level rise = 2.72 m

Reclassify the elevation data

By reclassifying the elevation data, we turn the ranges of elevations into categories or ranks. First, we will create a surface layer that has a ranking based on the ranges specified above. Navigate to Spatial Analyst Tools > Reclass > Reclassify. Ensure that the input raster is set to the mosaicked layer and set the Reclass field to Value. Name the output raster as NED_reclass. You will have to manually edit the values in the Reclassification dialog box. See the table below on how to change the values. NOTE: the ranges have to be typed in as number, space, hyphen, space, and then the number. Selecting each row and clicking the "delete entries" button can remove the extra rows. Leave the "NoData" row. Run the tool and then study the resulting layer. Remember what each value stands for before clicking **Apply**. Repeat this process for the other elevations (Reclass_mslr; Reclass_h09; Reclass_uslr)

Table 1: Reclassify for mean sea level and baseline storm surge.

Old values	New values
-115.932782 - 0	1
0 - 1.59	2
1.59 - 784.537	3

Table 2: Reclassify for flood height accounting for mean sea-level rise.

Old values	New values
-115.932782 - 0	1
0 - 2.21	2
2.21 - 784.537	3

Table 3: Reclassify for flood height accounting for Heberger et al. (2009) estimate.

Old values	New values
-115.932782 - 0	1
0 - 2.42	2
2.42 - 784.537	3

Table 4: Reclassify for flood height accounting for sea-level rise uncertainty.

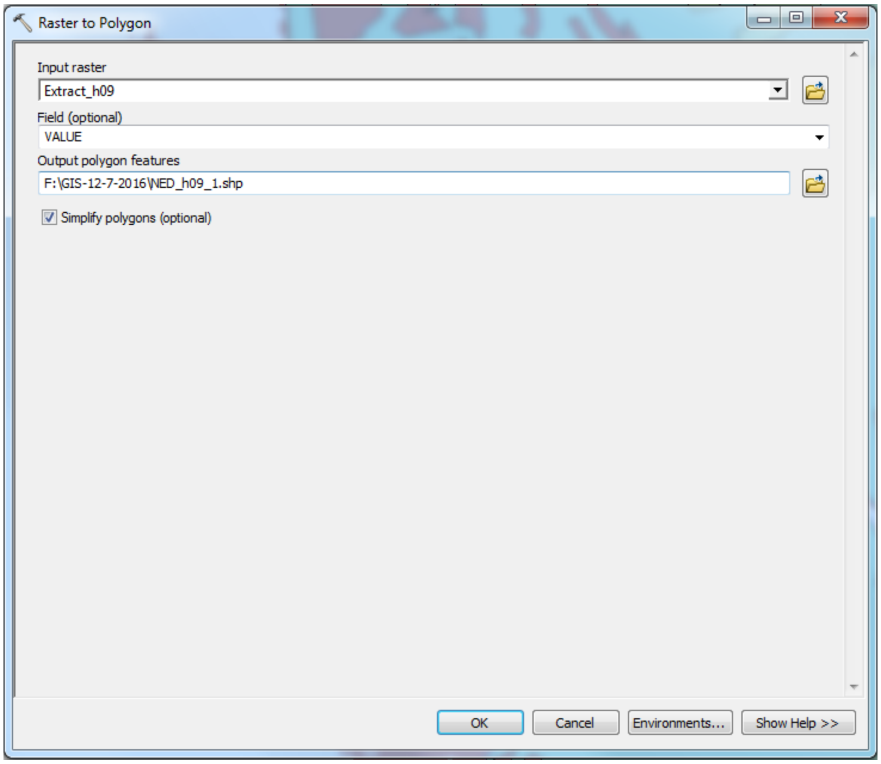
Old values	New values
-115.932782 - 0	1
0 - 2.72	2
2.72 - 784.537	3

Extract the elevation ranges

To turn the new datasets into workable forms, we want to extract the appropriate elevation ranges. For ease, you may want to turn off the raster layers. Navigate to Spatial Analyst Tools » Extraction » Extract by Attributes. First, we will extract the elevation range signifying standing water. Select the output raster named NED_reclass. Click on the small SQL query builder button to the right of the second field to build a . Only the "where clause" is needed. Select from 'Value' where "Value" = 1 (remember the standing water was ranked as 1). Name the extracted dataset as Extract_sl and click ok. Again navigate to the **Extract by Attributes** tool and select the output raster called NED_reclass. This time select from 'Value' where "Value"=2 (the storm surge value was ranked 2). Name the extracted dataset as Extract_ss and click **ok**. Repeat this process for the other elevations ranges; Extract_mslr ("Value"=2), Extract_h09 ("Value"=2), and Extract_uslr ("Value"=2).

Convert the results from raster to vector format

To enable futher analysis, convert the results from the extracted raster to a vector polygon. Navigate to Conversion Tools » From Raster » Raster to Polygon. Select extract_sl as the input and set the 'Field' to 'Value'. Name your new vector as NED_Water, leave the 'simplify polygons (option)' option checked, and run the tool. Repeat this step for the other extracted files and give them appropriate names (e.g., NED_stormsurge, NED_mslr, NED_h09, and NED_uslr). Since we are working with shapefiles, we can use the Python window to re-project them as NAD 83 California Zone 3.



Raster to Polygon

Input raster

Extract_h09

Field (optional)

VALUE

Output polygon features

F:\GIS-12-7-2016\NED_h09_1.shp

☑ Simplify polygons (optional)

```
# Project the new shapefiles for the output coordinate system: NAD 83
# California Zone 3
# ===================================================================
arcpy.Project_management(os.path.join(cwd, "Pre_analyzed_files/NED_Water.shp"),
    os.path.join(cwd, "Pre_analyzed_files/NED_Water_nad83z3.shp"), out_coordinate_system)

arcpy.Project_management(os.path.join(cwd, "Pre_analyzed_files/NED_stormsurge.shp"),
    os.path.join(cwd, "Pre_analyzed_files/NED_ss_Nad83z3.shp"), out_coordinate_system)

arcpy.Project_management(os.path.join(cwd, "Pre_analyzed_files/NED_mslr.shp"),
    os.path.join(cwd, "Pre_analyzed_files/NED_mslr_Nad83z3.shp"), out_coordinate_system)

arcpy.Project_management(os.path.join(cwd, "Pre_analyzed_files/NED_h09.shp"),
    os.path.join(cwd, "Pre_analyzed_files/NED_h09_Nad83z3.shp"), out_coordinate_system)

arcpy.Project_management(os.path.join(cwd, "Pre_analyzed_files/NED_uslr.shp"),
    os.path.join(cwd, "Pre_analyzed_files/NED_uslr_Nad83z3.shp"), out_coordinate_system)
```

At this time, you may remove or turn off the raster layers and nonprojected shapefiles from the table of contents. Lastly, we want to remove the flood risk area that is currently underwater (i.e., in the bay, ocean, rivers, or lakes). To do this, we will erase the known water area (Water_Nad83z3) from the flood risk areas. Again this can be done using the Python window.

```
# Erase the known water area from the flood risk areas:
# ============================================================
# Baseline storm surge
arcpy.Erase_analysis(os.path.join(cwd, "Pre_analyzed_files/NED_ss_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Water_Nad83z3.shp"), os.path.join(cwd,
    "Pre_analyzed_files/ss_Nad83z3.shp"), "")

# Mean sea-level rise
arcpy.Erase_analysis(os.path.join(cwd, "Pre_analyzed_files/NED_mslr_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Water_Nad83z3.shp"), os.path.join(cwd,
    "Pre_analyzed_files/mslr_Nad83z3.shp"), "")

# Estimate from Heberger et al. (2009)
arcpy.Erase_analysis(os.path.join(cwd, "Pre_analyzed_files/NED_h09_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Water_Nad83z3.shp"), os.path.join(cwd,
    "Pre_analyzed_files/h09_Nad83z3.shp"), "")

# Accounting for sea-level rise uncertainty
arcpy.Erase_analysis(os.path.join(cwd, "Pre_analyzed_files/NED_uslr_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Water_Nad83z3.shp"), os.path.join(cwd,
    "Pre_analyzed_files/uslr_Nad83z3.shp"), "")
```

Calculate the amount of area at risk of flooding

At this point it is reasonable to quantify the amount of land at risk of flooding in San Francisco, Oakland, and Alameda. First, you will need to know the boundary of San Francisco, Oakland, and Alameda. We can extract and save the area of the county/ sub-county using their **FID** number. Turn on the counties_Nad83z3 and Subcounties_Nad83z3 shapefiles if not already on.

```
# Select and write the selected features to a new feature class
#  for San Francisco #635, Oakland #319, and Alameda #320
# ============================================================
# San Francisco #635
arcpy.SelectLayerByAttribute_management("counties_Nad83z3", "NEW_SELECTION",
    '"FID" = 635 ')
arcpy.CopyFeatures_management("counties_Nad83z3", "SanFrancisco")

# Clear the selection
arcpy.SelectLayerByAttribute_management("counties_Nad83z3", "CLEAR_SELECTION")

# Oakland #319
arcpy.SelectLayerByAttribute_management("Subcounties_Nad83z3", "NEW_SELECTION",
    '"FID" = 319 ')
arcpy.CopyFeatures_management("Subcounties_Nad83z3" , "Oakland")

# Clear the selection
arcpy.SelectLayerByAttribute_management("Subcounties_Nad83z3", "CLEAR_SELECTION")

# Alameda #320
arcpy.SelectLayerByAttribute_management("Subcounties_Nad83z3", "NEW_SELECTION",
    '"FID" = 320 ')
arcpy.CopyFeatures_management("Subcounties_Nad83z3", "Alameda")
```

```python

# Clear the selection
arcpy.SelectLayerByAttribute_management("Subcounties_Nad83z3", "CLEAR_SELECTION")
```

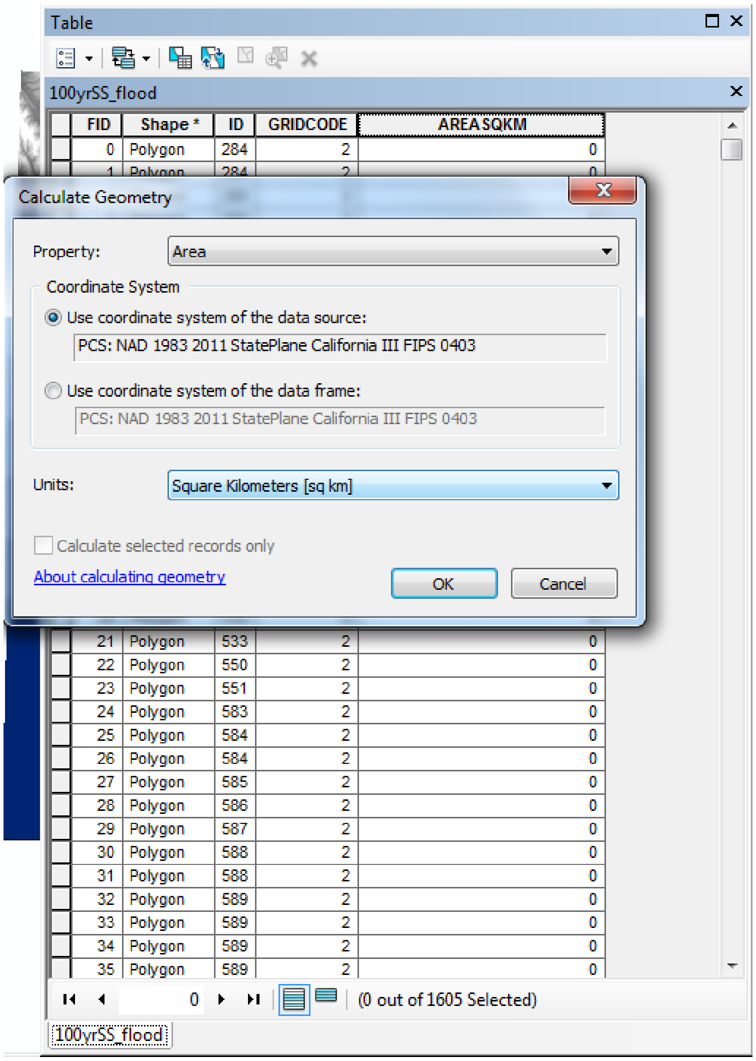
The second step is to restrict the flooding polygons to the specific counties/ sub-counties. This can be done using multiple functions. Using a different function may produce slightly different results. Here, we will use **Clip** to clip the polygon to the area of the county/ sub-county specified.

```python
# Clip the flood area polygons with the San Francisco, Oakland, and Alameda
# counties/ sub-counties
# ==============================================================
arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/ss_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/SanFrancisco.shp"), os.path.join(cwd,
    "Pre_analyzed_files/ss_SanFrancisco.shp"))
arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/mslr_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/SanFrancisco.shp"), os.path.join(cwd,
    "Pre_analyzed_files/mslr_SanFrancisco.shp"))
arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/h09_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/SanFrancisco.shp"), os.path.join(cwd,
    "Pre_analyzed_files/h09_SanFrancisco.shp"))
arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/uslr_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/SanFrancisco.shp"), os.path.join(cwd,
    "Pre_analyzed_files/uslr_SanFrancisco.shp"))

arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/ss_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Oakland.shp"), os.path.join(cwd,
    "Pre_analyzed_files/ss_Oakland.shp"))
arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/mslr_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Oakland.shp"), os.path.join(cwd,
    "Pre_analyzed_files/mslr_Oakland.shp"))
arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/h09_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Oakland.shp"), os.path.join(cwd,
    "Pre_analyzed_files/h09_Oakland.shp"))
arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/uslr_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Oakland.shp"), os.path.join(cwd,
    "Pre_analyzed_files/uslr_Oakland.shp"))

arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/ss_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Alameda.shp"), os.path.join(cwd,
    "Pre_analyzed_files/ss_Alameda.shp"))
arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/mslr_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Alameda.shp"), os.path.join(cwd,
    "Pre_analyzed_files/mslr_Alameda.shp"))
arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/h09_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Alameda.shp"), os.path.join(cwd,
    "Pre_analyzed_files/h09_Alameda.shp"))
arcpy.Clip_analysis(os.path.join(cwd, "Pre_analyzed_files/uslr_Nad83z3.shp"),
    os.path.join(cwd, "Pre_analyzed_files/Alameda.shp"), os.path.join(cwd,
    "Pre_analyzed_files/uslr_Alameda.shp"))
```

Next, create an additional field for each flood polygon for San Francisco, Alameda, and Oakland. This additional field will be populated with estimates of the area in km^2 for the polygon.

```
# Create an additional field to quantify the area in km^2 flooded
# in each county/ sub-county
# ==========================================================
arcpy.AddField_management("ss_SanFrancisco", "AREASQKM", "DOUBLE")
arcpy.AddField_management("mslr_SanFrancisco", "AREASQKM", "DOUBLE")
arcpy.AddField_management("h09_SanFrancisco", "AREASQKM", "DOUBLE")
arcpy.AddField_management("uslr_SanFrancisco", "AREASQKM", "DOUBLE")

arcpy.AddField_management("ss_Oakland", "AREASQKM", "DOUBLE")
arcpy.AddField_management("mslr_Oakland", "AREASQKM", "DOUBLE")
arcpy.AddField_management("h09_Oakland", "AREASQKM", "DOUBLE")
arcpy.AddField_management("uslr_Oakland", "AREASQKM", "DOUBLE")

arcpy.AddField_management("ss_Alameda", "AREASQKM", "DOUBLE")
arcpy.AddField_management("mslr_Alameda", "AREASQKM", "DOUBLE")
arcpy.AddField_management("h09_Alameda", "AREASQKM", "DOUBLE")
arcpy.AddField_management("uslr_Alameda", "AREASQKM", "DOUBLE")
```

Last, populate the new field. In the table of contents, right click one of the layers and open the **Attribute Table**. In the attribute table, right click the new field named **AREASQKM** and click Calculate Geometry > Square Kilometers [sq km] > OK. If you right click on the field again and go to **statistics** you can see the sum of the total area at risk for the county/ sub-county. Repeat this process for each polygon.



Table

100yrSS_flood

FID	Shape *	ID	GRIDCODE	AREA SQKM
0	Polygon	284	2	0
1	Polygon	284	2	0
21	Polygon	533	2	0
22	Polygon	550	2	0
23	Polygon	551	2	0
24	Polygon	583	2	0
25	Polygon	584	2	0
26	Polygon	584	2	0
27	Polygon	585	2	0
28	Polygon	586	2	0
29	Polygon	587	2	0
30	Polygon	588	2	0
31	Polygon	588	2	0
32	Polygon	589	2	0
33	Polygon	589	2	0
34	Polygon	589	2	0
35	Polygon	589	2	0

Calculate Geometry

Property: Area

Coordinate System

● Use coordinate system of the data source:
PCS: NAD 1983 2011 StatePlane California III FIPS 0403

○ Use coordinate system of the data frame:
PCS: NAD 1983 2011 StatePlane California III FIPS 0403

Units: Square Kilometers [sq km]

□ Calculate selected records only

About calculating geometry

0 (0 out of 1605 Selected)

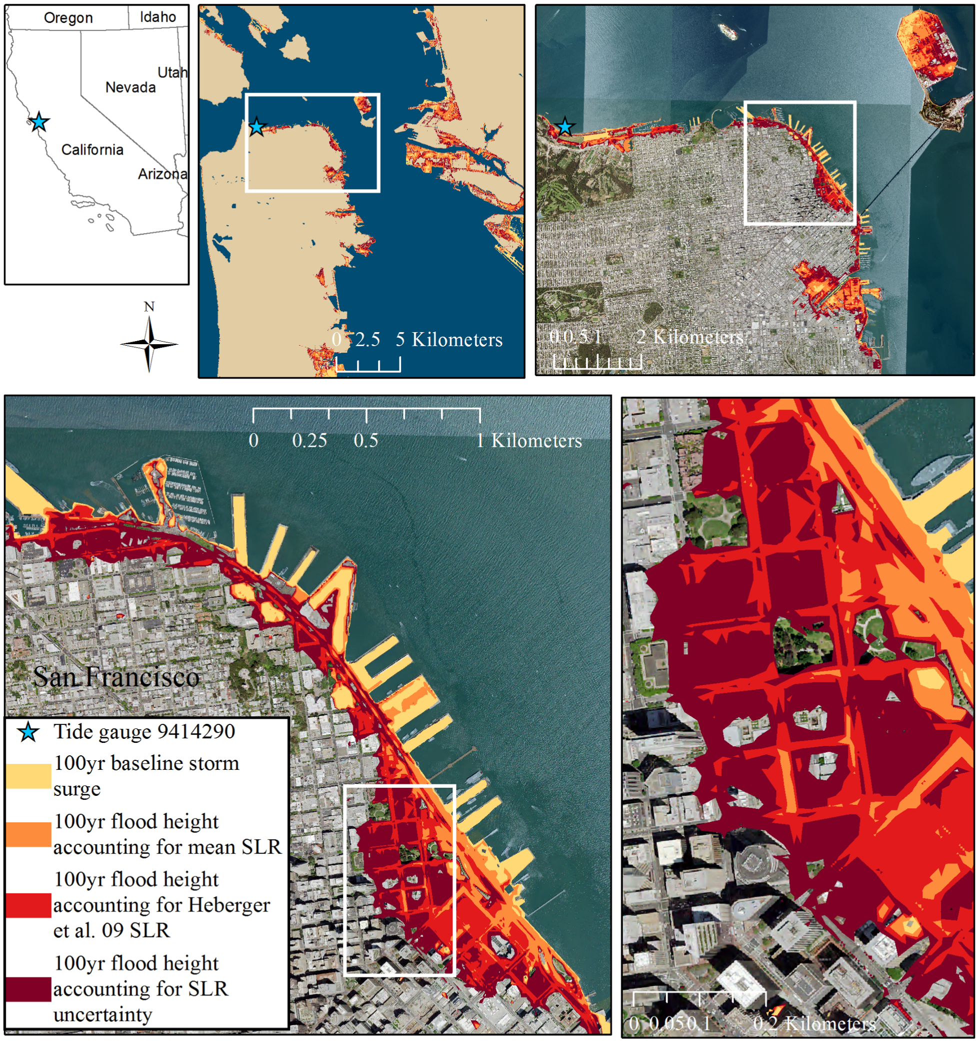
100yrSS_flood

Create a visual map

Make a map of the results of your analysis. Be sure that the map includes: a title, scale bar, legend, north arrow, and acknowledgement of the sources (for all layers of data). These additions can be found by clicking **Insert** on the Toolbar. You can change the name of the layer by clicking on the name in the Table of Contents. You can also consider using a **base map** service, available from Esri over the internet (click the down arrow beside **Add Data** and select **Add Basemap**). You can also consider using the elevation raster as the background and can make the storm surge polygons semi-transparent. Multiple screens can be added by adding a **Data Frame** (Insert › Data Frame). To make the map (Fig 5) in Ruckert et al. [1], add the tide gauge location (SF_TG9414290.shp), add the USGS Imagery-1 meter (NAIP) files, and set the polygons according to the following "rgb" colors.

Table 5: RGB numbers for map.

Polygon	Red	Green	Blue
Baseline storm surge	254	217	118
Flood height accounting for mean SLR	253	141	60
Flood height accounting for Heberger et al. (2009) SLR	227	26	28
Flood height accounting for SLR uncertainty	128	0	38


Oregon
Idaho
Utah
Nevada
California
Arizona
N
0 2.5 5 Kilometers
0 0.5 1 2 Kilometers
0 0.25 0.5 1 Kilometers
San Francisco
Tide gauge 9414290
100yr baseline storm surge
100yr flood height accounting for mean SLR
100yr flood height accounting for Heberger et al. 09 SLR
100yr flood height accounting for SLR uncertainty
0 0.05 0.1 0.2 Kilometers

Sources: U.S. Census Bureau; NOAA Tides and Currents; USGS National Elevation Dataset
Base layer: USGS Imagery-1 meter (NAIP)
Credit: U.S. Department of Agriculture, Farm Service Agency; U.S. Department of Interior, USGS

References

[1] Kelsey L. Ruckert, Perry C. Oddo, and Klaus Keller. Accounting for sea-level rise uncertainty increases the projected area at risk of flooding: An example from san francisco bay. *N.a.*, In review.