Automatic Production of Large Spatial Price Rasters with Python

A lightning presentation with a focus on geo-processing tools found in the python library arcpy.

GASP Workshop
Kevin Hunt, Geographer, Research & Development Division, NASS
November 6, 2020 | Washington, DC

Overview

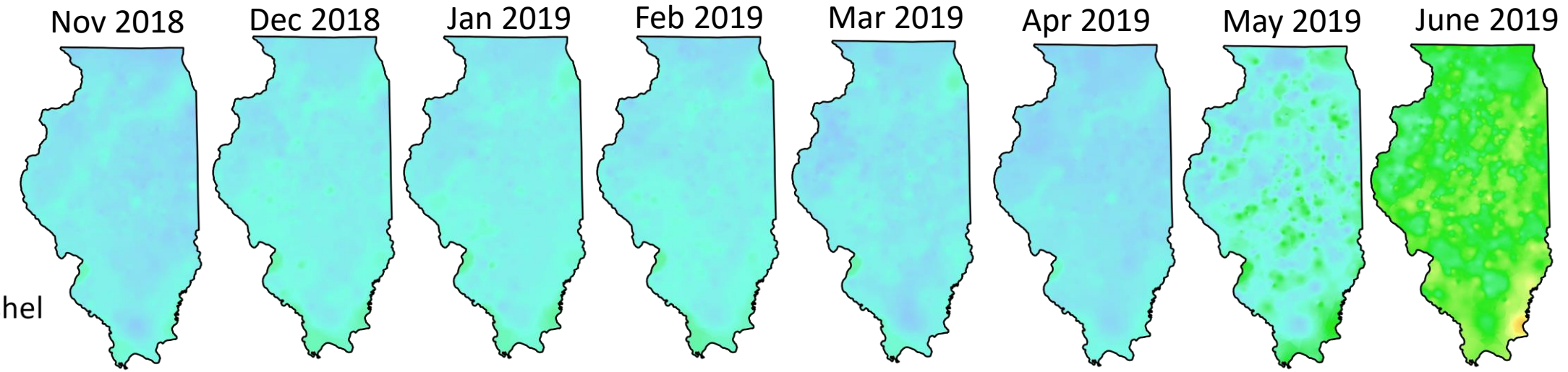
USDA National Agricultural Statistics Service (NASS) is developing a machine-learning early season forecasting model for pre-season acreage estimation that includes geo-located agricultural commodity prices.

Steps to automating the creation of this data:

1. Scrape data from host website and format into monthly averages
2. Convert dbf tables into point features
3. Convert point features into IDW raster surfaces
4. Format data for the machine-learning model

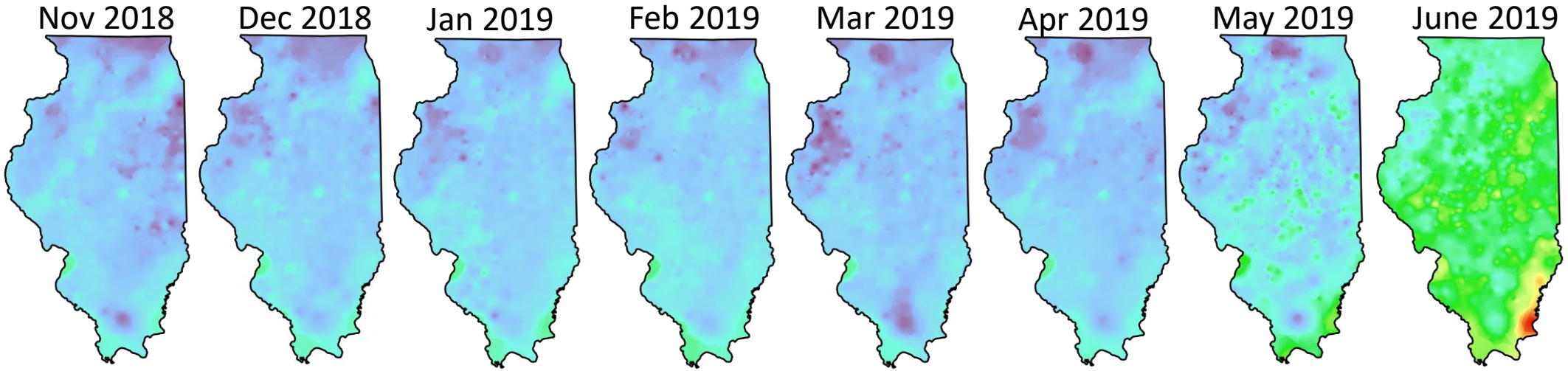
Monthly averages:

Basis price:



Price per bushel

Future contract price:



Step 1: Scape data from web

RStudio

- Set workspace
- Define url

- Set temporal parameters for daily data
- Average into monthly values

- Output as dbf tables with values and geo-locations

Step 2: Convert tables to points

Import arcpy
Import arcpy management

Set workspace
Set output workspace

Use arcpy "list files"

Format file to a dbf

In arcpy management
Use XYTableToPoint
Set parameters

```
import arcpy
from arcpy import management

arcpy.env.workspace = r"drive:\example\DBF"
outWorkspace = r"drive:\example\Points"

dbfList = arcpy.ListFiles()
For dbf in dbfList:
        outputDBF = "{}\\{}".format(outWorkspace, dbf)

arcpy.management.XYTableToPoint(in_table=dbf,
out_feature_class=outputDBF, x_field="longitude",
y_field="latitude", coordinate_system="GCS_WGS_1984")
```

Step 3: Convert points to IDW surfaces

Import arcpy
Import arcpy environment
Import Spatial analyst
Import os

Use arcpy
"list feature classes"

Set parameters for IDW

```
import arcpy
from arcpy import env
from arcpy.sa import *
import os
arcpy.env.workspace = r"drive:\example\Points"
outWorkspace = r"drive:\example\IDW"

points = arcpy.ListFeatureClasses("", "Point")

zField = "prc"
cellSize = 30
Power = 2
searchRadius = RadiusVariable(5, 112654)
```

Step 3: Convert points to IDW surfaces

Check out Spatial Analyst

Loop through list Using the IDW tool

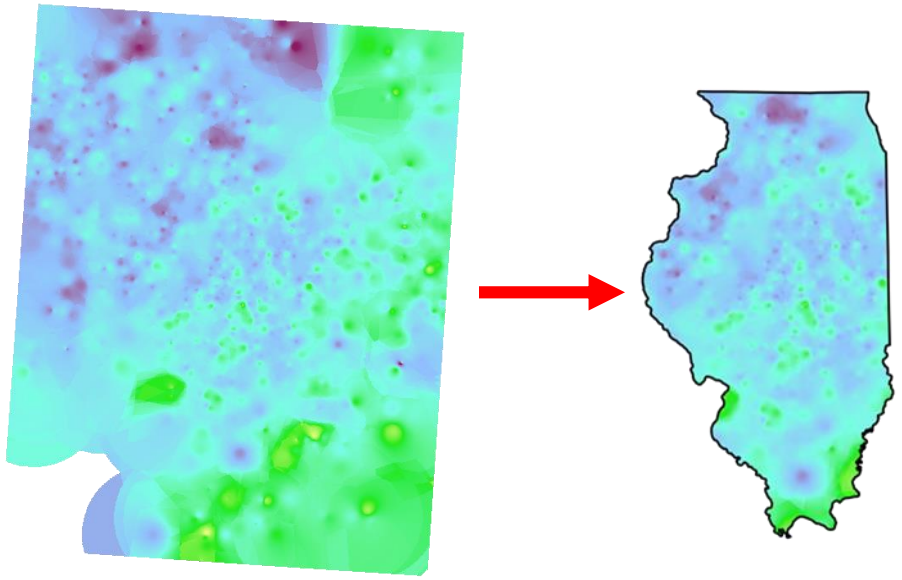
Use os to join '.img' and save files as IMAGINE files

```
arcpy.CheckOutExtension("Spatial")

for fc in points:
        with arcpy.EnvManager(outputCoordinateSystem="Equal
        Area Conic Projection")
        outIDW = arcpy.sa.Idw(fc, zfield, cellSize, power,
                 searchRadius)

        IMGname = fc[:-4]
        IDW_OUT=pm os.path.join(outWorkspace, '{}.img.format
                  (IMGname))
        outIDW.save(IDW_OUT)
```

Step 4: Format data



ERDAS IMAGINE 2018 (64-bit)

- Mask to IL shape
- Snap pixels to Cropland Data Layer

- Make first process in Model Maker and save a ‘gmd’ file
- Generate scrip and save as an ‘mdl’ file
- Edit script and save as ‘bcf’ file which can be run as a batch command.

- Files now can be run in machine-learning model.

Summary

- Streamlined workflow
- Automation increased speed
- This process can be applied to other data

Thank you!

For questions contact:
Kevin.A.Hunt@usda.gov