…please install the Bitwise Font available here.

Lynker Analytics

- Based in Wellington, New Zealand
- Formed in late 2018 in a partnership with US based, **Lynker**
- 3 primary delivery areas
    - Artificial Intelligence
    - Geospatial Analytics
    - Data Visualisation

@LynkerAnalytics LynkerAI

Lynker Analytics

- A team of 3 with 20+ years experience in the Geospatial & Data Science domains
- **Matt Lythe** ~ Managing Director
- **David Knox** ~ Principal, Data Science
- **Phil Woods** ~ Principal, Analytics & Visualisation

...an **"AI First"** geospatial business who specialise in unlocking insights from geospatial datasets through the use of advanced location-informed, **machine learning.**

@LynkerAnalytics LynkerAI

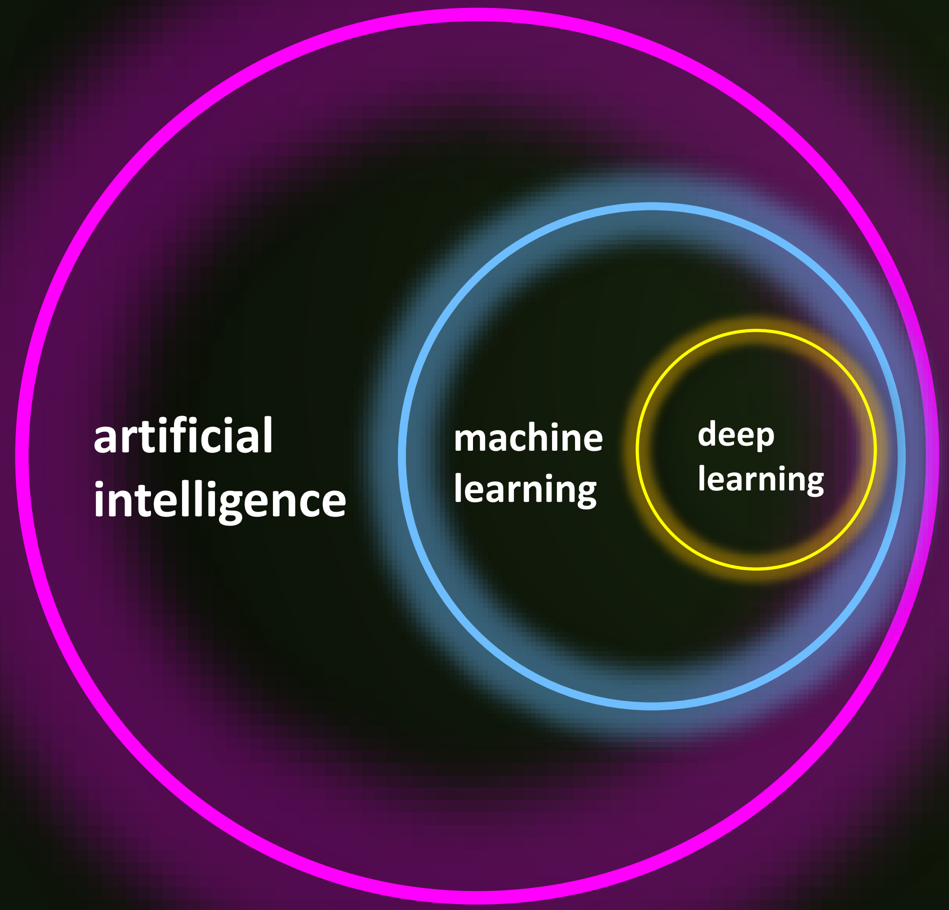
Lynker Analytics

- Recent R&D focused on porting proprietary Machine Learning technology into Esri's ArcGIS Pro
  - Computer Vision
  - Earth Observation (EO) Machine Learning

...an **"AI First"** geospatial business who specialise in unlocking insights from geospatial datasets through the use of advanced location-informed, **machine learning.**

@LynkerAnalytics   LynkerAI

AI vs Machine Learning or Deep Learning



- Both “Machine Learning” & “Deep Learning” are subsets of the greater, Artificial Intelligence domain
- **Machine Learning and Deep Learning are** where most “data science” occurs
- Both are very useful tools for geospatial analytics
- Esri have implemented several Machine Learning / Deep Learning tools within the ArcGIS platform

Machine Learning?

- an evolving way of communicating with computers that allows us to program them by example rather than instruction
- It’s an automation tool - consistent performance and high accuracy
- *“Anything you can do…”* machine learning processes can do almost any easily repeatable, task oriented procedure humans carry out today
- It’s also an analytics tool that’s good at identifying patterns and correlations across large, unstructured datasets
- Can be applied to a variety of geospatial disciplines including remote sensing, geospatial analysis and data extraction

How Machines Learn…

In most cases, machine learning processes are "trained" by relating a proportionally representative amount of the desired output against an equivalent amount of the available input… or something like that.

- **Supervised**
  - Labelled data provided to achieve a good model
- **Semi Supervised**
  - Human-in-the-loop (active learning)
  - Label propagation
- **Unsupervised**
  - No labelling required

Machine Learning +

- Esri provide several tools that can be used within or to create Machine Learning workflows, including:
  - Export Training Data for Deep Learning
  - Detect Objects & Classify Pixels using Deep Learning
  - Forest-based Classification and Regression
- Esri don't - at this stage - provide pre-built models for Object Detection or Classification although there's a simple example for Buildings using RCNN masks here: https://github.com/Esri/raster-deep-learning/tree/master/examples/keras/mask_rcnn

Computer Vision

- **Object Detection**
  - An object within an image is identified with a bounding geometry
  - Commonly used to locate objects within large imagery datasets for quantitative analytics

- **Pixel Classification** or **Sematic Segmentation**
  - Pixels within an image are classified as belonging to a specific object
  - Commonly used within geospatial workflows to extract features from imagery

Computer Vision

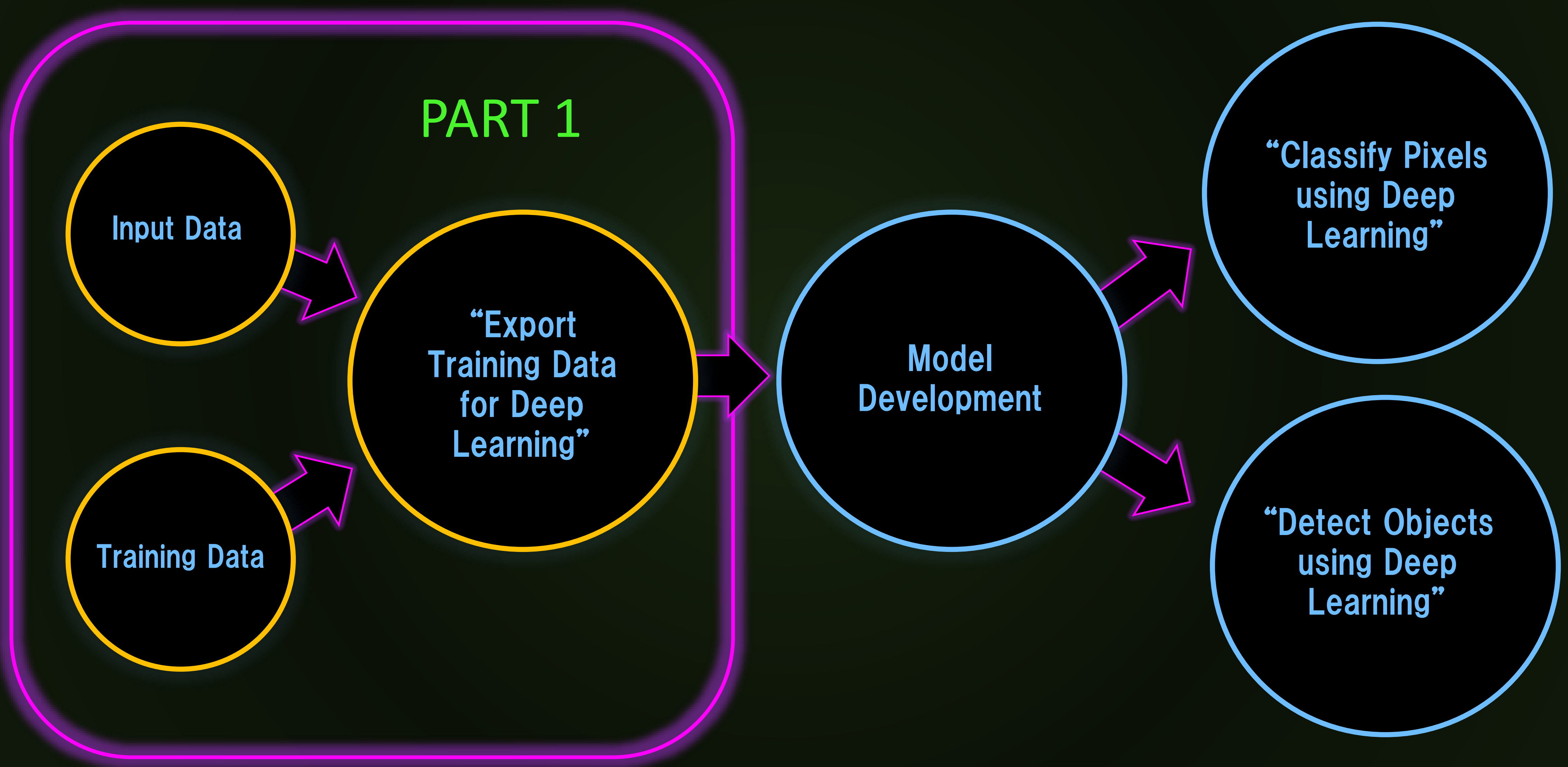
Object Detection

- Quantitative Analytics

Semantic Segmentation

- Data Capture

Computer Vision ...in Esri



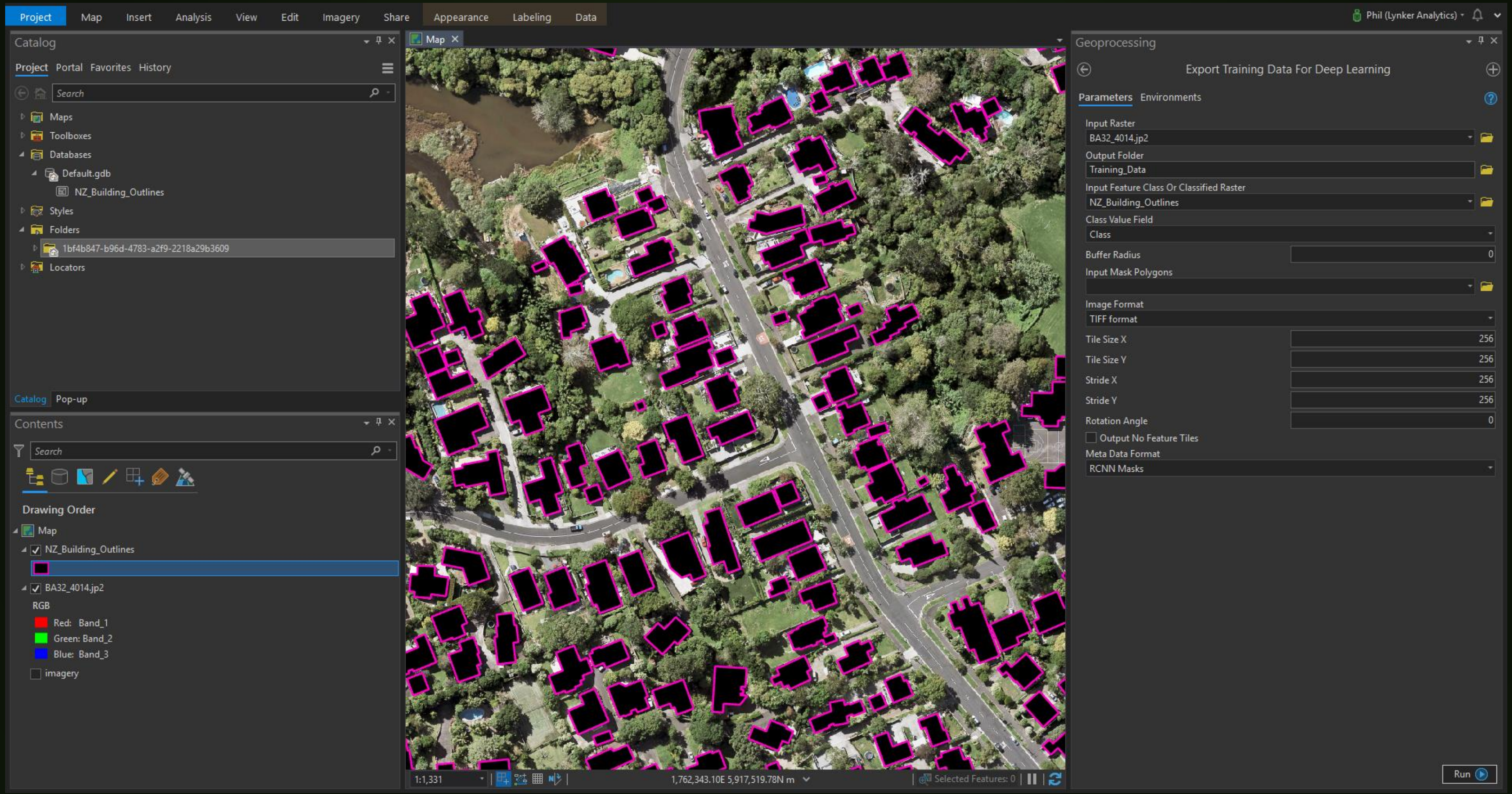
Export Training Data for Deep Learning

Export Training Data For Deep Learning (Image Analyst Tools)
Uses a remote sensing image to convert labeled vector or raster data into deep learning training datasets. The output is a folder of image chips and a folder of metadata files in the specified format.

- Converts labelled vector or raster data into training datasets (known as 'image chips')
- Training data produced by this tool is provided to a data scientist to create the associated machine learning model

Export Training Data for Deep Learning

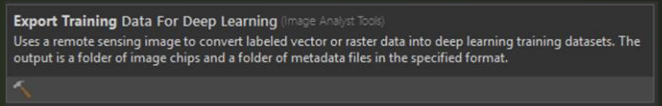
Export Training Data For Deep Learning (Image Analyst Tools)
Uses a remote sensing image to convert labeled vector or raster data into deep learning training datasets. The output is a folder of image chips and a folder of metadata files in the specified format.

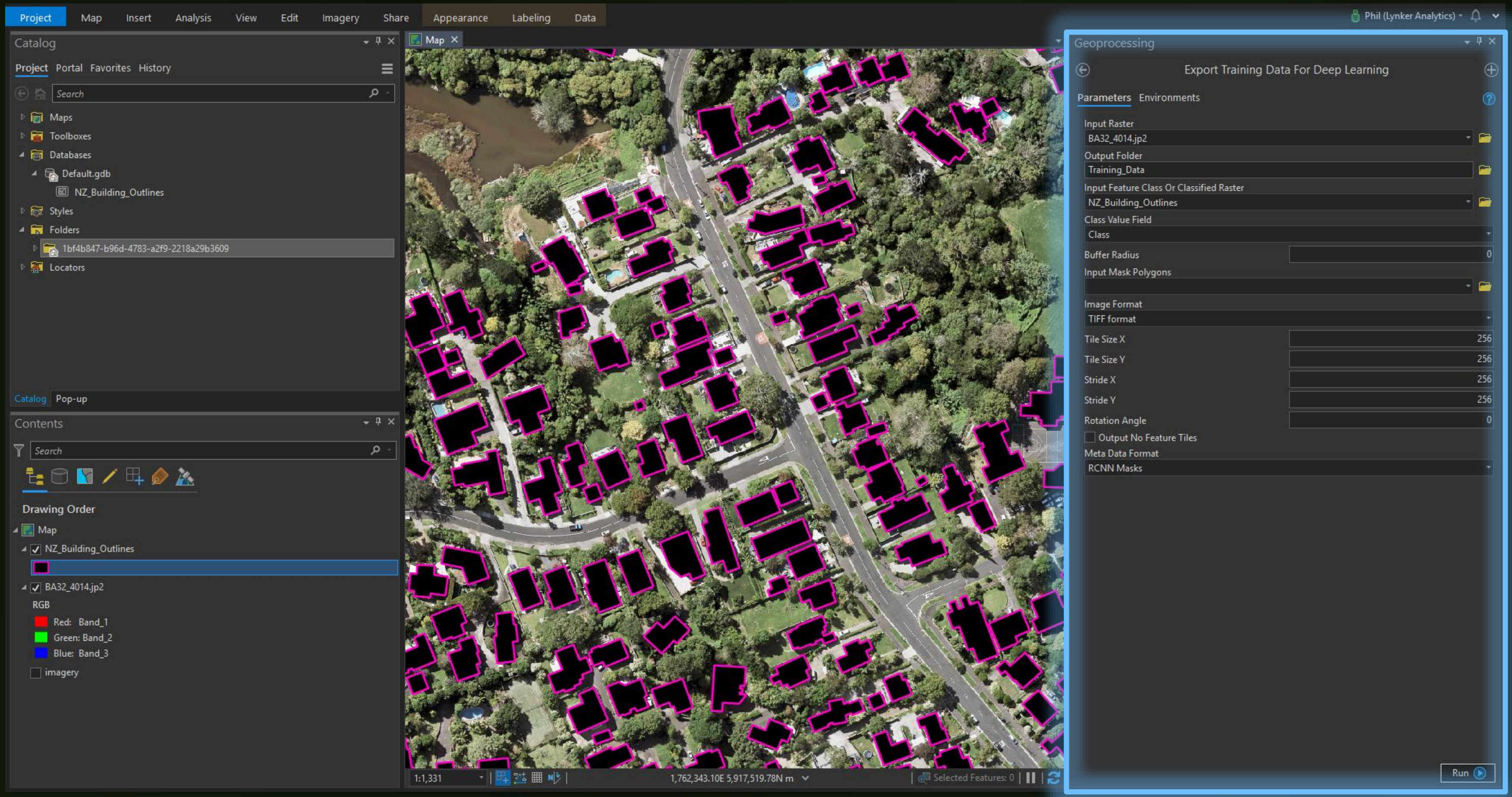


Labels: buildings

Example: imagery

Export Training Data for Deep Learning





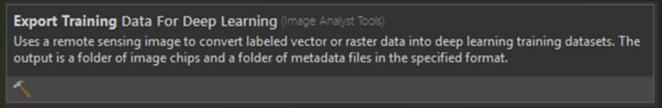
⇐ Training Image – RGB

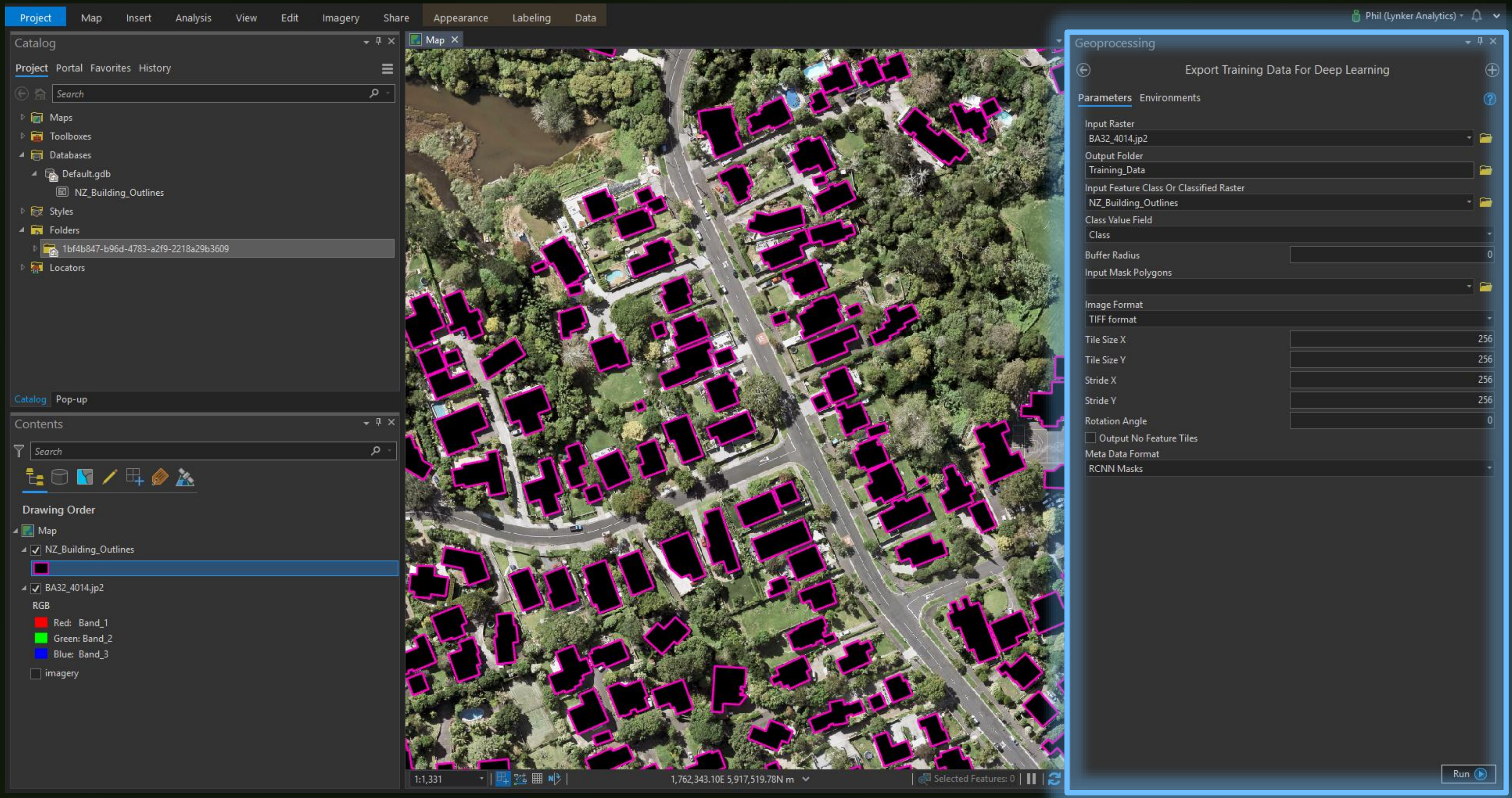
⇐ Training Data – Buildings

⇐ Class Value – Single Value

⇐ Format – RCNN Masks

Export Training Data for Deep Learning





Formats:

- **RCNN Masks**
- **Classified Tiles**
- **KTTI Rectangles**
- **Pascal VOC**

Export Training Data for Deep Learning

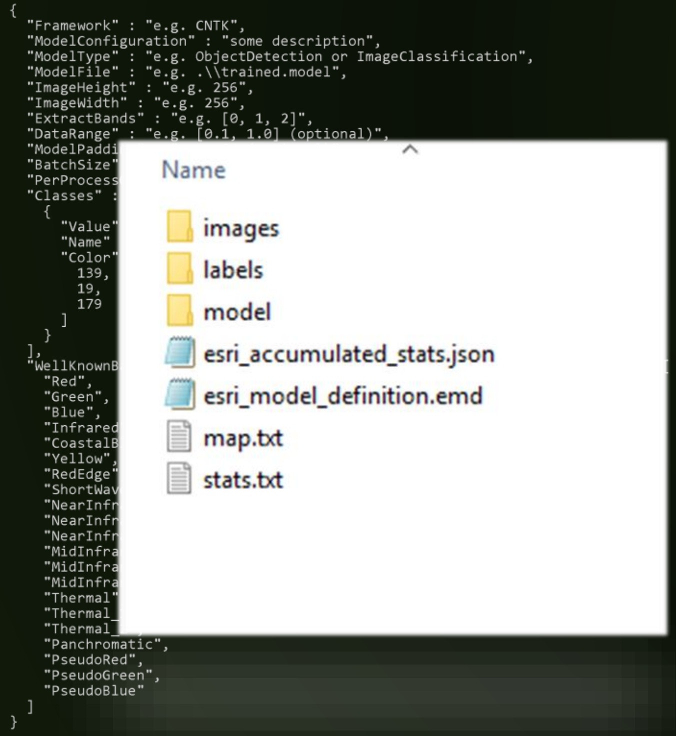
```
import arcpy
from arcpy.ia import *
from datetime import datetime
import fnmatch
import os

# arcpy.env.overwriteOutput = True
start = datetime.now()

try:
   arcpy.CheckOutExtension("SpatialAnalyst")
except:
   e = sys.exc_info()[1]
   print(e.args[0])
   arcpy.AddError(e.args[0])

# variables - system
project = "C:/Projects/Esri_Machine_Learning"
geodatabase = project + "/MachineLearning.gdb"

# variables - export training data tool
in_raster = project + "/mosaic.tif"
out_folder = project + "/buildings_training_data"
in_class_data = geodatabase + "/training_buildings"
image_chip_format = ""
tile_size_x = "256"
tile_size_y = "256"
stride_x = "256" # chip side overlap
stride_y = "256" # chip forward overlap
output_nofeature_tiles = "ONLY_TILES_WITH_FEATURES" # ALL_TILES, ONLY_TILES_WITH_FEATURES
metadata_format = "RCNN_Masks" # KITTI_rectangles, PASCAL_VOC_rectangles, Classified_Tiles, RCNN_Masks, Labeled_Tiles
start_index = "0"
class_value_field = "CLASS" # attribute in featureclass
buffer_radius = ""
in_mask_polygons = "" # geodatabase + "/bounding_box"
rotation_angle = "0"
```

```
# export training data tool
try:
   print("Exporting training data...")
   ExportTrainingDataForDeepLearning(in_raster
                                     , out_folder
                                     , in_class_data
                                     , image_chip_format
                                     , tile_size_x
                                     , tile_size_y
                                     , stride_x
                                     , stride_y
                                     , output_nofeature_tiles
                                     , metadata_format
                                     , start_index
                                     , class_value_field
                                     , buffer_radius
                                     , in_mask_polygons
                                     , rotation_angle)

   print("Process completed in " + str(datetime.now() - start))
   result = out_folder + "/images/"
   file_count = (len(fnmatch.filter(os.listdir(result), '*.tif')))
   if file_count == 0:
      print("No errors but something's wrong - no training data was produced!")
   else:
      print(str(file_count) + " training tiles were produced.")

except:
   e = sys.exc_info()[1]
   print(e.args[0])
   arcpy.AddError(e.args[0])
```

Export Training Data for Deep Learning

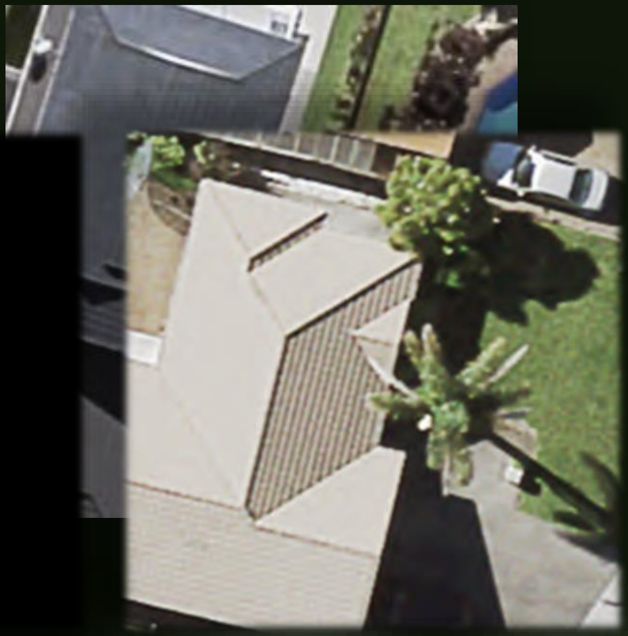
Export Training Data For Deep Learning (Image Analyst Tools)
Uses a remote sensing image to convert labeled vector or raster data into deep learning training datasets. The output is a folder of image chips and a folder of metadata files in the specified format.

.emd (Esri Model Definition) package

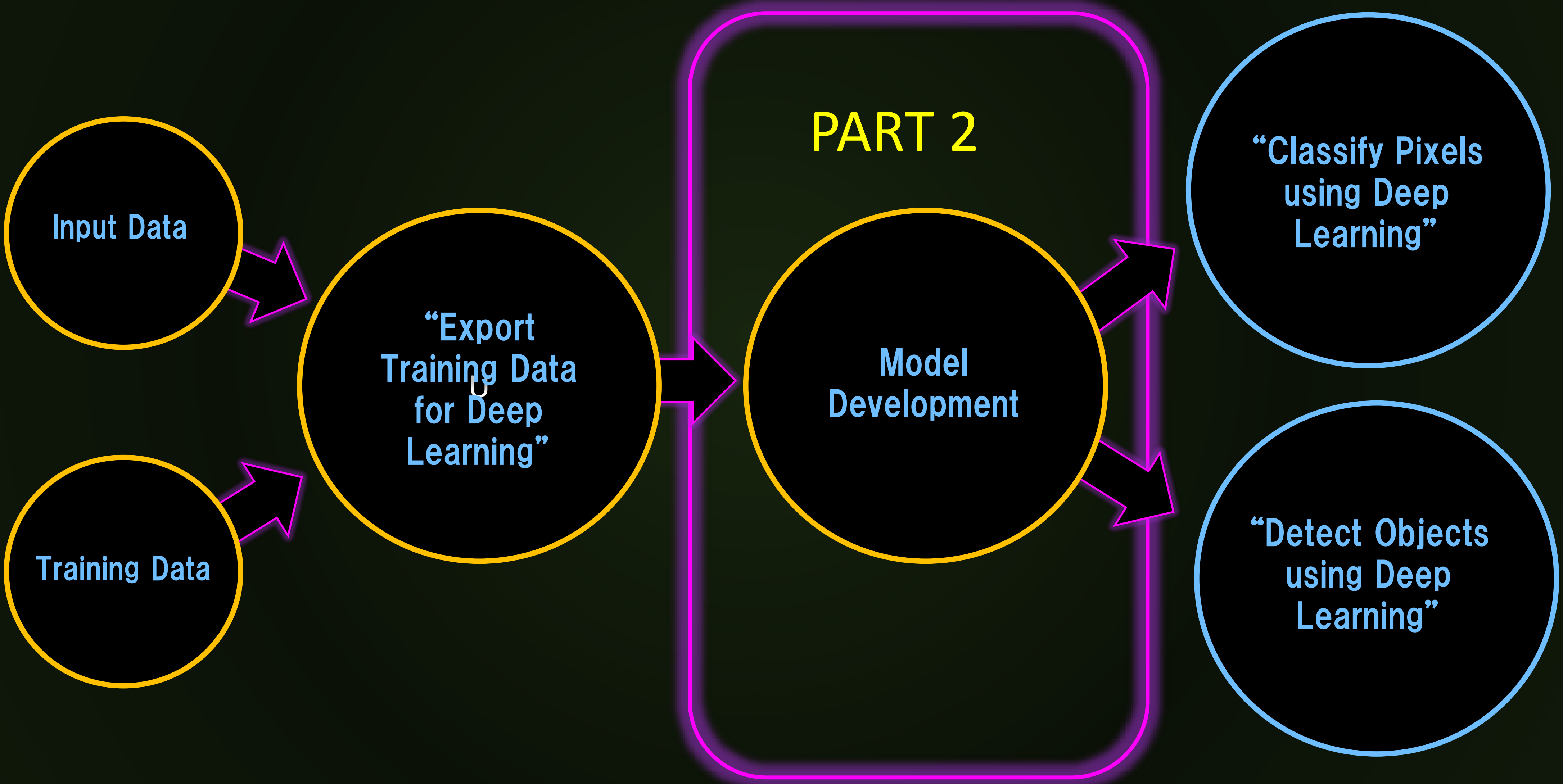


Labels

Images



Computer Vision ...in Esri



The Data Scientist...

.emd package


```
{
  "Framework" : "e.g. CNTK",
  "ModelConfiguration" : "some description",
  "ModelType" : "e.g. ObjectDetection or ImageClassification",
  "ModelFile" : "e.g. .\\trained.model",
  "ImageHeight" : "e.g. 256",
  "ImageWidth" : "e.g. 256",
  "ExtractBands" : "e.g. [0, 1, 2]",
  "DataRange" : "e.g. [0.1, 1.0] (optional)",
  "ModelPadding" : "e.g. 64 (optional)",
  "BatchSize" : "e.g. 8 (optional)",
  "PerProcessGPUMemoryFraction" : "e.g. 0.8 (optional)",
  "Classes" : [
    {
      "Value" : 1,
      "Name" : "1",
      "Color" : [
        139,
        19,
        179
      ]
    }
  ],
  "WellKnownBandNames (FYI, these band names can be used in ExtractBands)" : [
    "Red",
    "Green",
    "Blue",
    "Infrared",
    "CoastalBlue",
    "Yellow",
    "RedEdge",
    "ShortWaveInfrared",
    "NearInfrared",
    "NearInfrared_1",
    "NearInfrared_2",
    "MidInfrared",
    "MidInfrared_1",
    "MidInfrared_2",
    "Thermal",
    "Thermal_1",
    "Thermal_2",
    "Panchromatic",
    "PseudoRed",
    "PseudoGreen",
    "PseudoBlue"
  ]
}
```


+

=

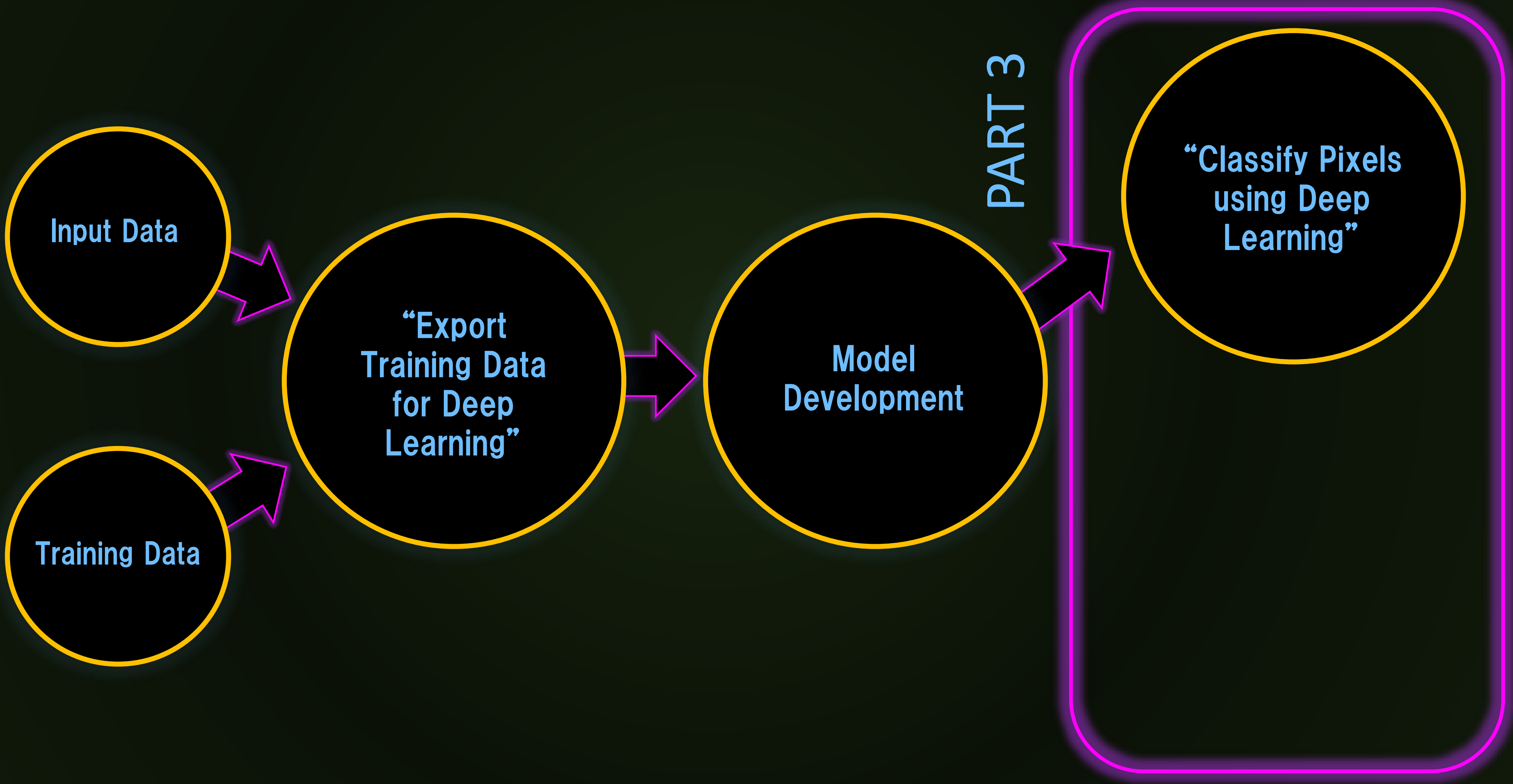
Trained Models

Keras

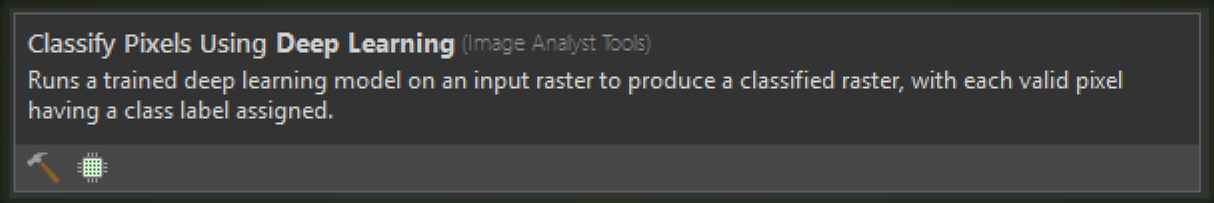
TensorFlow

PyTorch

Computer Vision ...in Esri

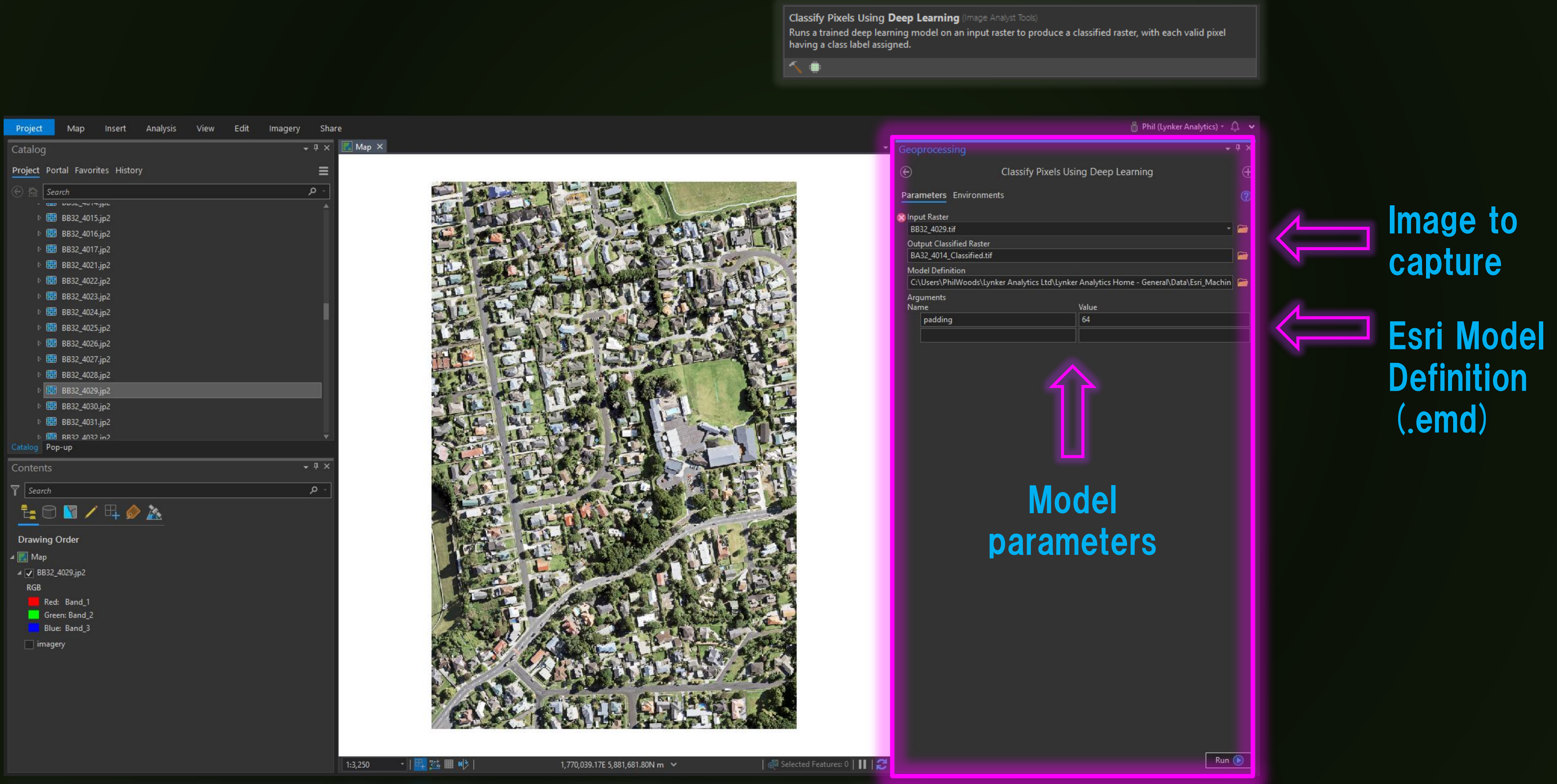


Classify Pixels Using Deep Learning

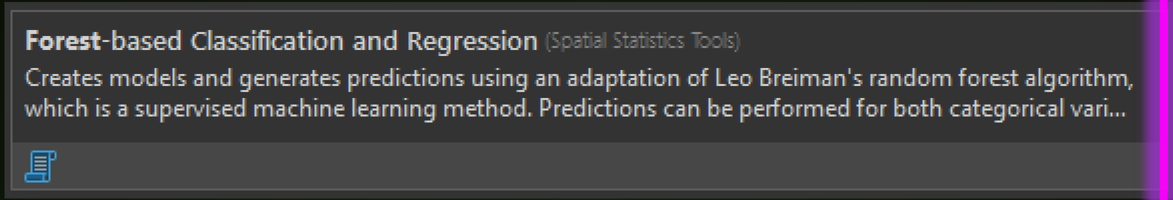


- Classifies pixels within an image to produce a segmented image
- **Highly** recommend a computer with a high-end Nvidia GPU with CUDA and Tensor Core support. This makes a huge difference.

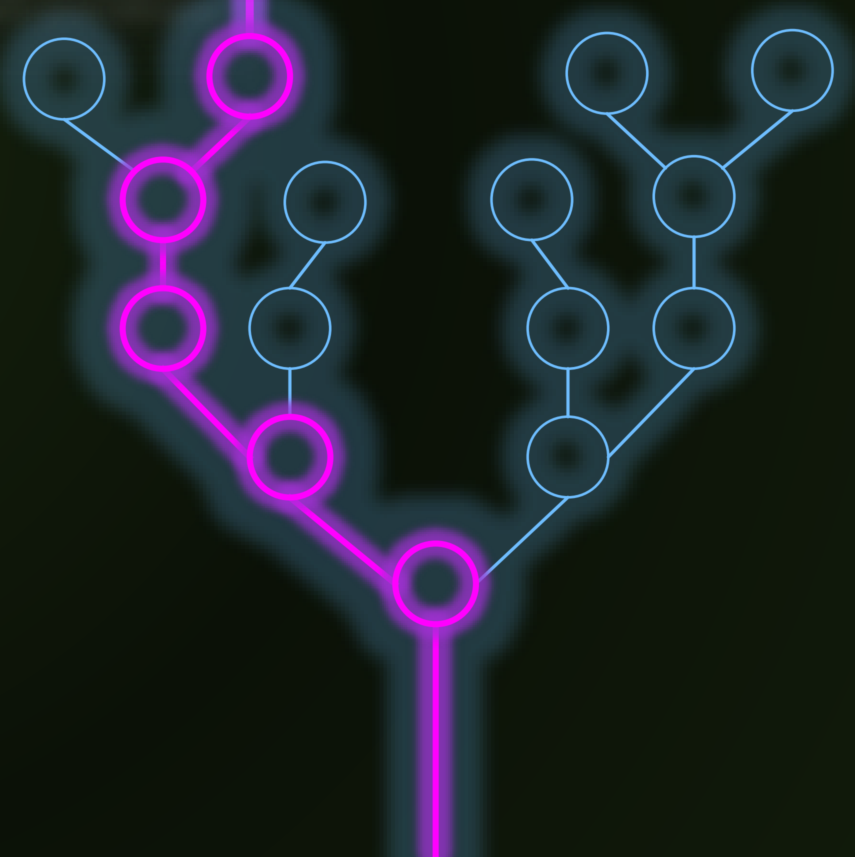
Classify Pixels Using Deep Learning



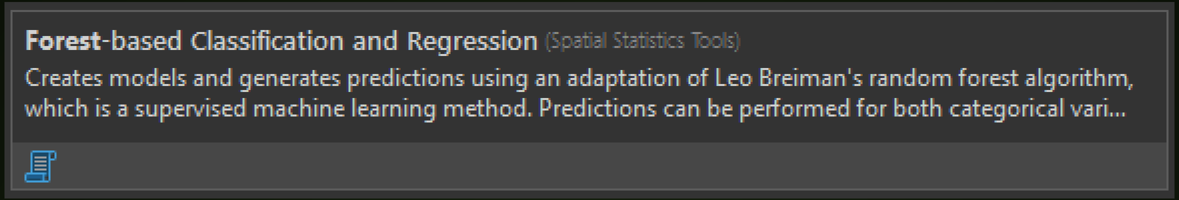
Forest-based Classification and Regression



- An ensemble-based learning method primarily used for classification and regression
- "Random Forests" are made up of multiple "decision trees" and can be used to rank the importance of variables in a regression or classification problem
- Interpolate / predict missing data
- Requires Scikit Learn Python library

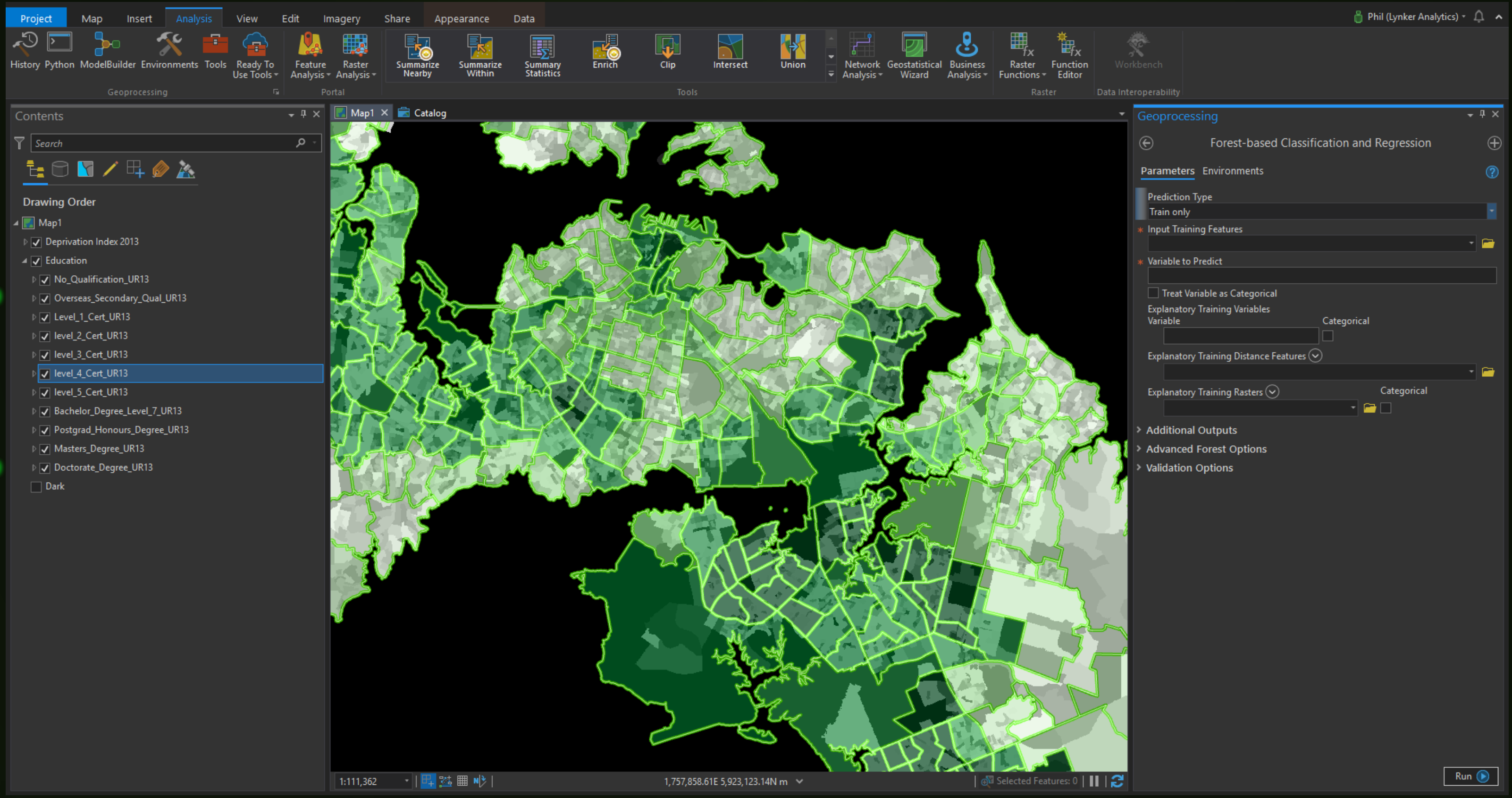


Forest-based Classification and Regression

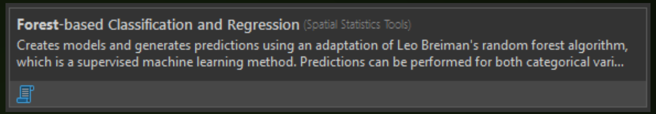


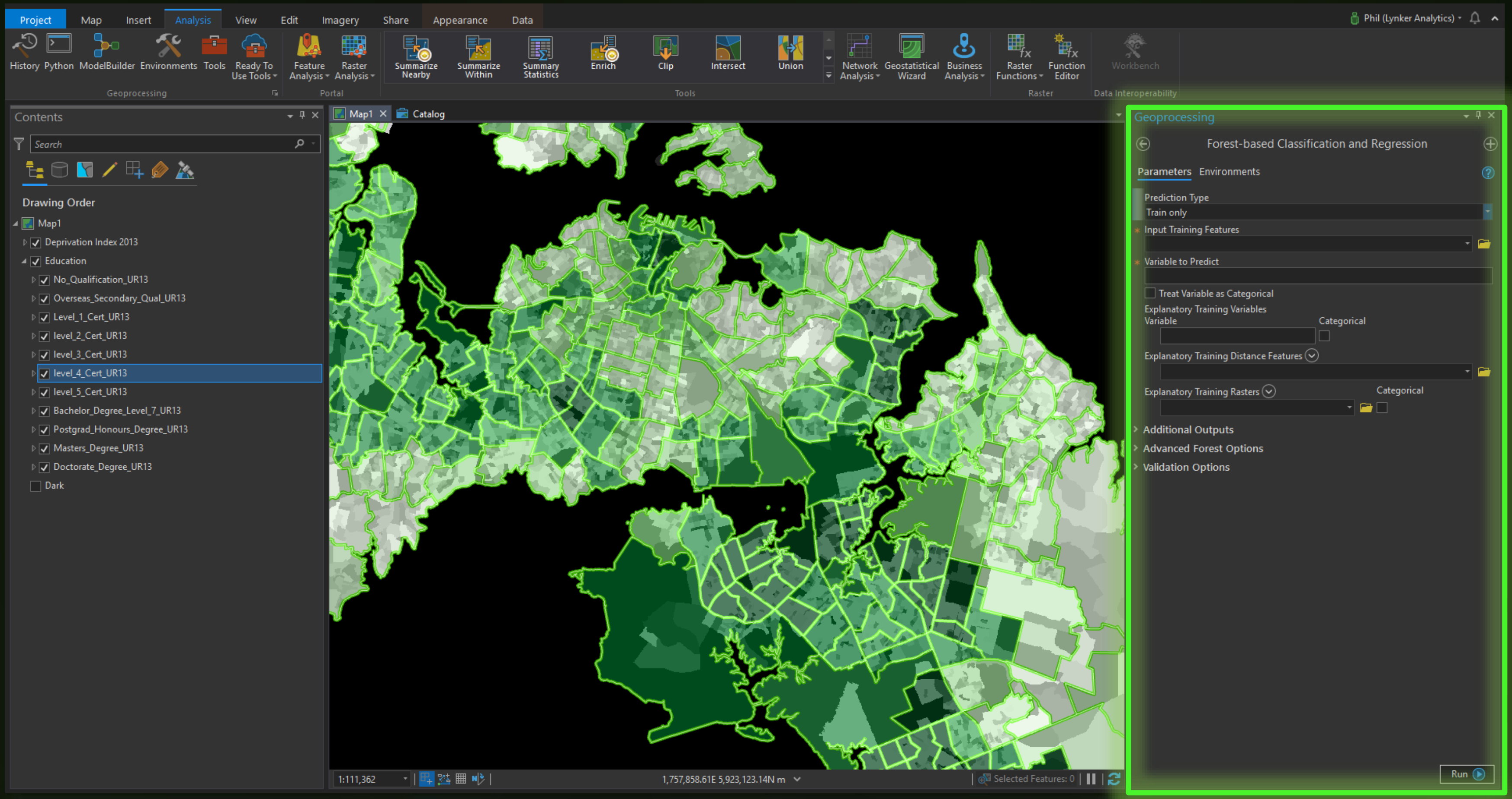
Deprivation Index

Census 2013 Education



Forest-based Classification and Regression

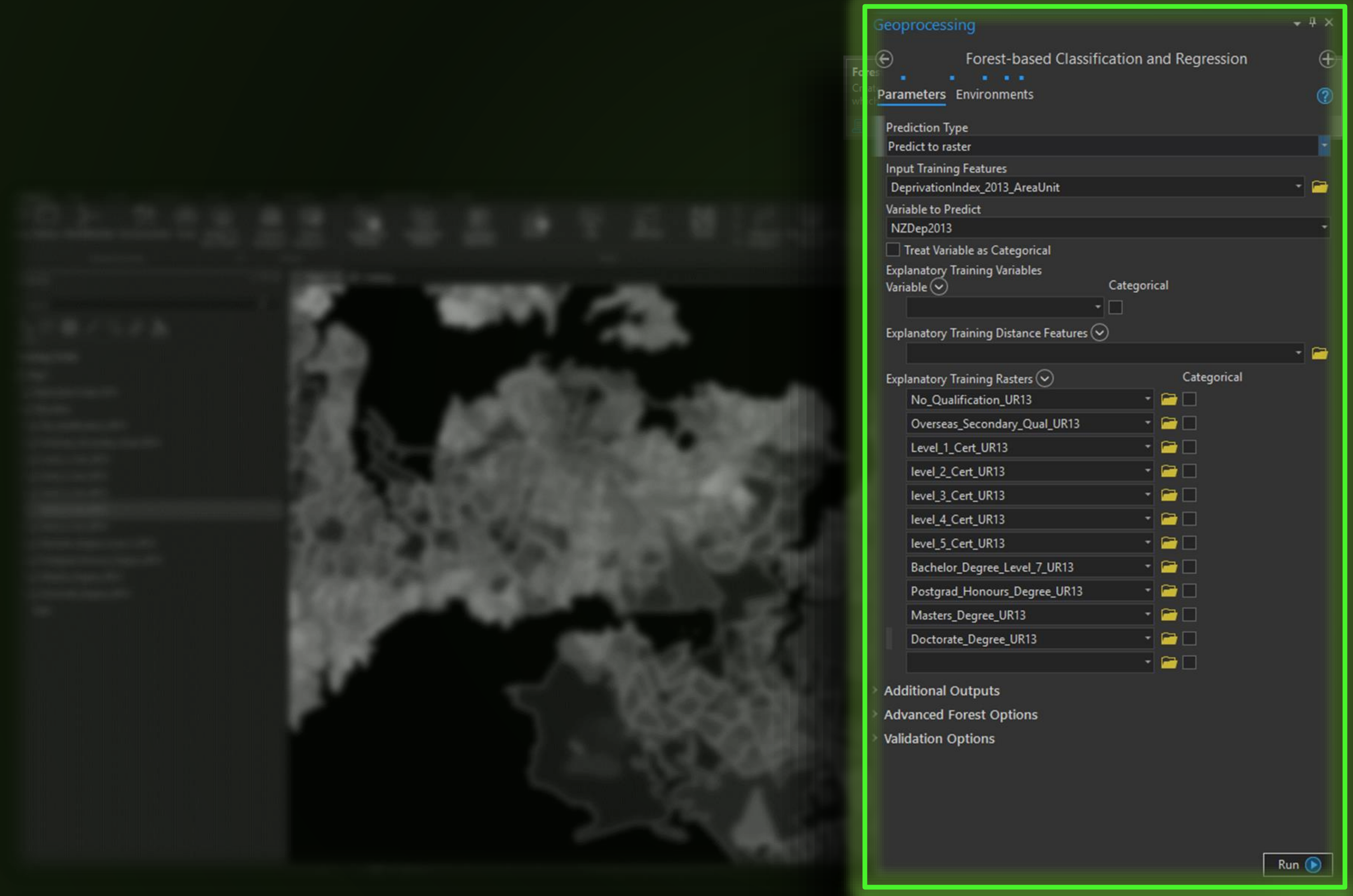




Key variables:

- Prediction Type
- Input Features
- Variable to Predict
- Explanatory Rasters
- Validation Options

Forest-based Classification and Regression



Key variables:

- Prediction Only
- Deprivation Index
- NZDep2013
- Census - Education
- 10% data excluded for validation

Forest-based Classification and Regression

Messages

```
Start Time: Thursday, 8 August 2019 11:46:15 PM
Running script Forest...
Random Seed: 444885
-------------- Model Characteristics ---------------Number of Trees
100Leaf Size                                            5Tree Depth Range
17-25Mean Tree Depth                                     20% of Training Available per Tree
100Number of Randomly Sampled Variables                  3% of Training Data Excluded for
Validation     10------------ Model Out of Bag Errors ------------Number of Trees
50          100MSE                           3.764       3.670% of variation explained
60.721       61.706--------------------- Top Variable Importance ---------------------
Variable                                 Importance             %NO_QUALIFICATION_UR13
3398.78                21POSTGRAD_HONOURS_DEGREE_UR13              1780.47
11LEVEL_1_CERT_UR13                         1666.52              10LEVEL_5_CERT_UR13
1599.27                10BACHELOR_DEGREE_LEVEL_7_UR13              1508.00
9LEVEL_3_CERT_UR13                         1303.68              8LEVEL_2_CERT_UR13
1222.29                 8LEVEL_4_CERT_UR13                          1193.90
7DOCTORATE_DEGREE_UR13                      989.44              6MASTERS_DEGREE_UR13
820.52                 5OVERSEAS_SECONDARY_QUAL_UR13               747.33              5--
--- Training Data: Regression Diagnostics ------R-Squared
0.920p-value                                  0.000Standard Error
0.006*Predictions for the data used to train the model compared to the observed categories
for those features---- Validation Data: Regression Diagnostics -----R-Squared
0.622p-value                                  0.000Standard Error
0.031*Predictions for the test data (excluded from model training) compared to the observed
values for those test features------------------------- Explanatory Variable Range
Diagnostics ------------------------                        Training
Validation              Training  Validation         Variable                      Minimum
Maximum    Minimum       Maximum   Share(a)      Share(b)NO_QUALIFICATION_UR13        0.00
113.61     0.00          51.26       1.00           0.45*OVERSEAS_SECONDARY_QUAL_UR13   0.00
63.28      0.00          24.24       1.00           0.38*LEVEL_1_CERT_UR13             0.00
88.94      0.00          32.23       1.00           0.36*LEVEL_2_CERT_UR13             0.00
72.57      0.00          29.66       1.00           0.41*LEVEL_3_CERT_UR13             0.00
185.23     0.00          158.20      1.00           0.85*LEVEL_4_CERT_UR13             0.00
81.14      0.00          26.98       1.00           0.33*LEVEL_5_CERT_UR13             0.00
53.29      0.00          23.41       1.00           0.44*BACHELOR_DEGREE_LEVEL_7_UR13   0.00
86.91      0.00          37.02       1.00           0.43*POSTGRAD_HONOURS_DEGREE_UR13   0.00
20.54      0.00          8.76        1.00           0.43*MASTERS_DEGREE_UR13           0.00
21.68      0.00          10.55       1.00           0.49*DOCTORATE_DEGREE_UR13         0.00
6.69       0.00          5.53        1.00           0.83*(a) % of overlap between the ranges of
the training data and the input explanatory variable(b) % of overlap between the ranges of
the validation data and the training data*  Data ranges do not coincide. Training or
validation is occurring with incomplete data.+  Ranges of the training data and prediction
data do not coincide and the tool is attempting to extrapolate.
Completed script Forest-based Classification and Regression...
Succeeded at Thursday, 8 August 2019 11:53:40 PM (Elapsed Time: 7 minutes 25 seconds)
```

Key variables:

- **Prediction Only**
- **Deprivation Index**
- **NZDep2013**
- **Census - Education**
- **10% data excluded for validation**

Thanks

matt.lythe@lynker-analytics.com
david.knox@lynker-analytics.com

lynker-analytics.com

@LynkerAnalytics

LynkerAI