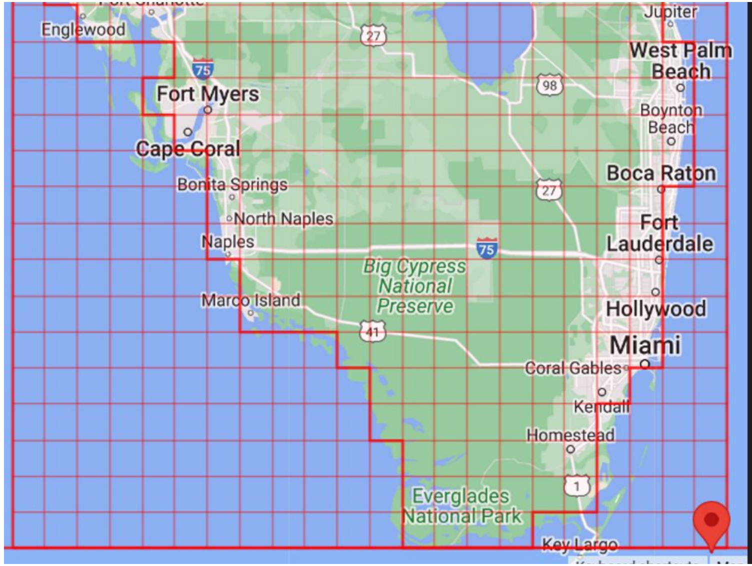
Predicting Tomorrow’s Rain

Isabela Yepes

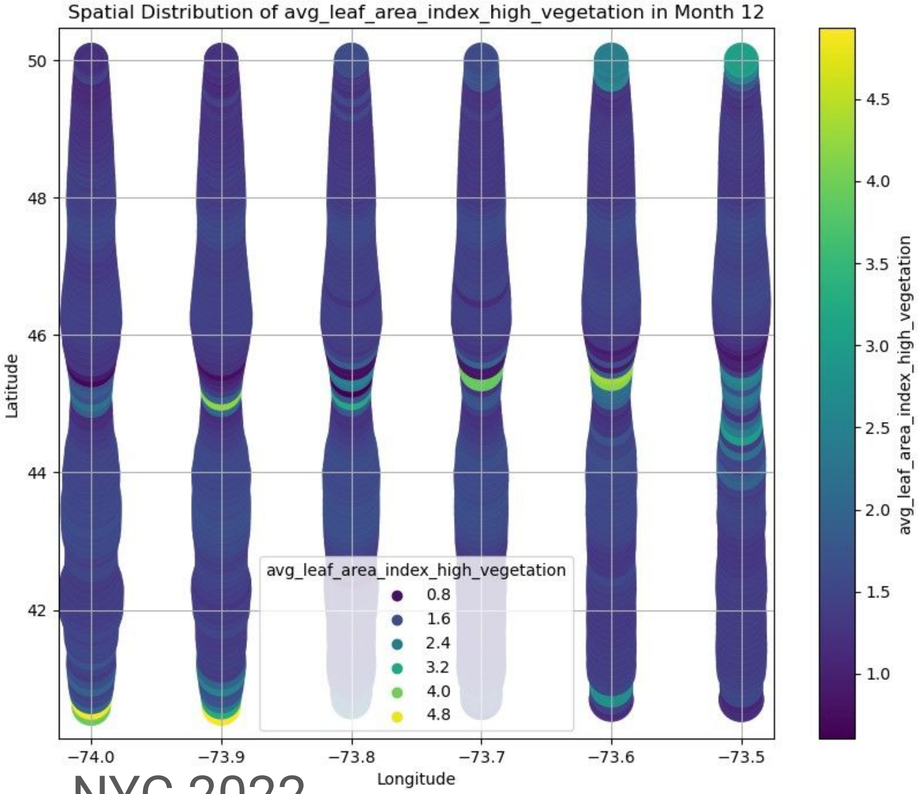
Project Definition

- Goal: Predict tomorrow’s rain ‘next_day_prcp_total’ using current day’s total rainfall, temperature, wind, surface pressure, and vegetation features.
- Regressions and classifiers in machine learning
- Models trained and tested on South Florida 2015 data
- Tested on NYC and South Florida 2022 December data.



- Github: https://github.com/isabelayepes/PredTomorrowsRain

Project Examples

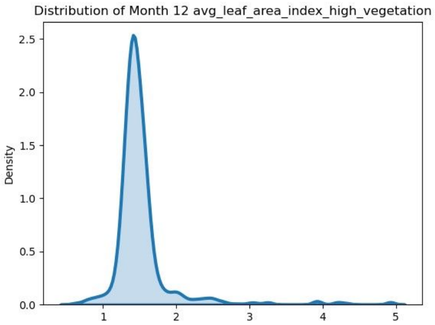


NYC 2022

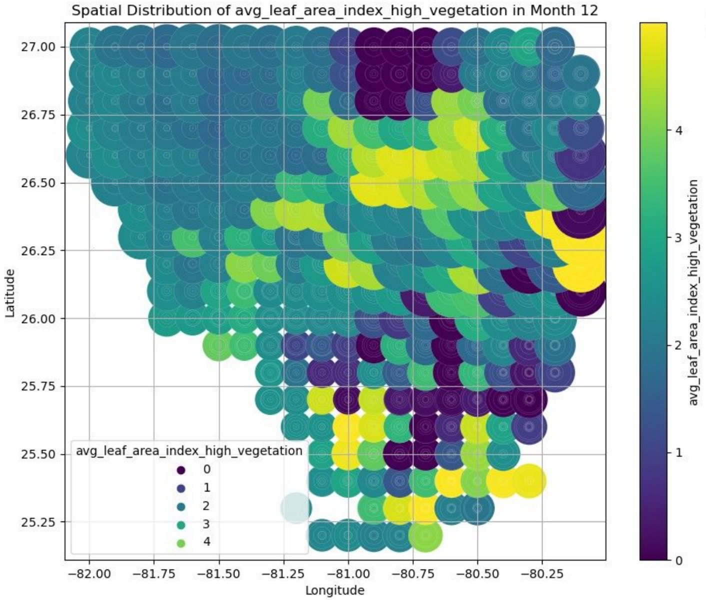
-Hourly netCDF data from a climate API with a Python script

- converted to daily (avg, min, max, 75%, 25%) CSV calculated target variable with R script

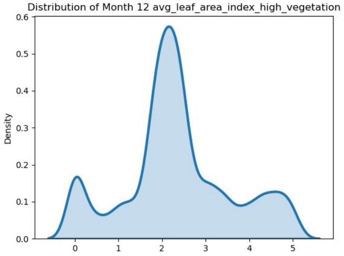
-cleared NaN data



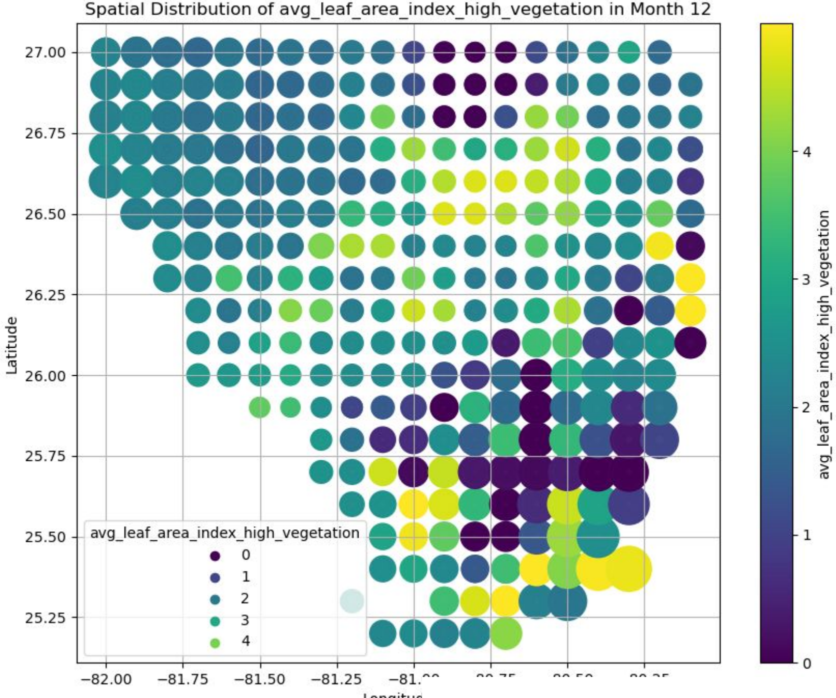
Project Examples



Spatial Distribution of avg_leaf_area_index_high_vegetation in Month 12



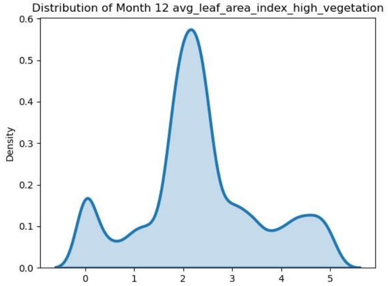
Distribution of Month 12 avg_leaf_area_index_high_vegetation



Spatial Distribution of avg_leaf_area_index_high_vegetation in Month 12

South FL
2015

South FL
2022

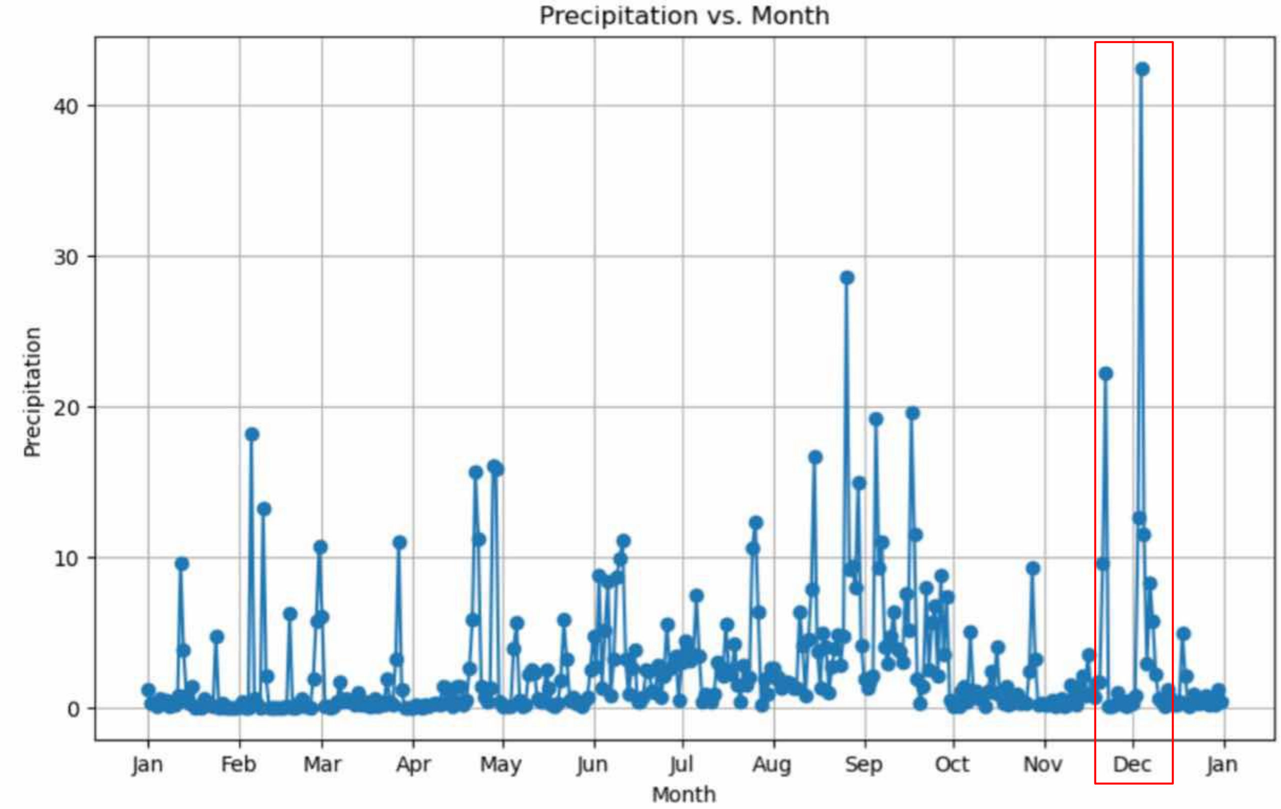


Distribution of Month 12 avg_leaf_area_index_high_vegetation

Project Details

-standardized & outlier removal

South FL 2015



Project Details

-Data Points: 87941 (2015 South Florida)

-Model 1: Neural Net Regression (10 epochs, 2 hidden layers Relu activation)

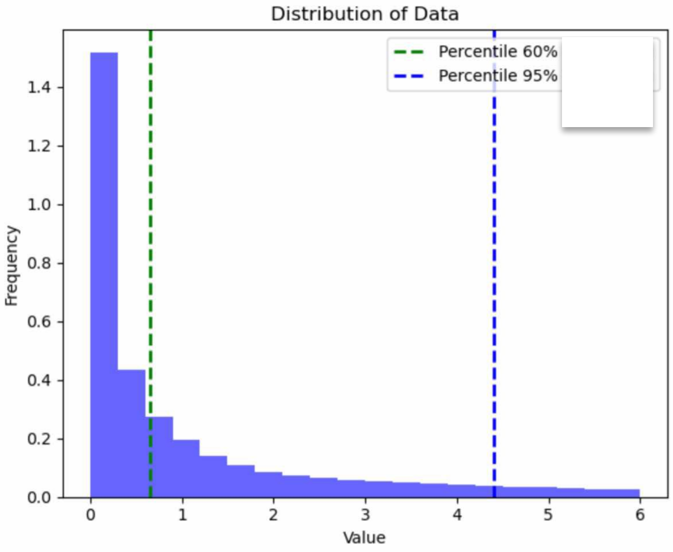
-Result with Lat and Lon: test MAE 0.443 mm

-Results: test MAE: 0.4996 mm

Categorized (creates class imbalance)

-Model 2: Random Forest Classifier 88% accurate (44 features)

-Model 3: XGBoost Classifier 89-92% acurate (44 features)

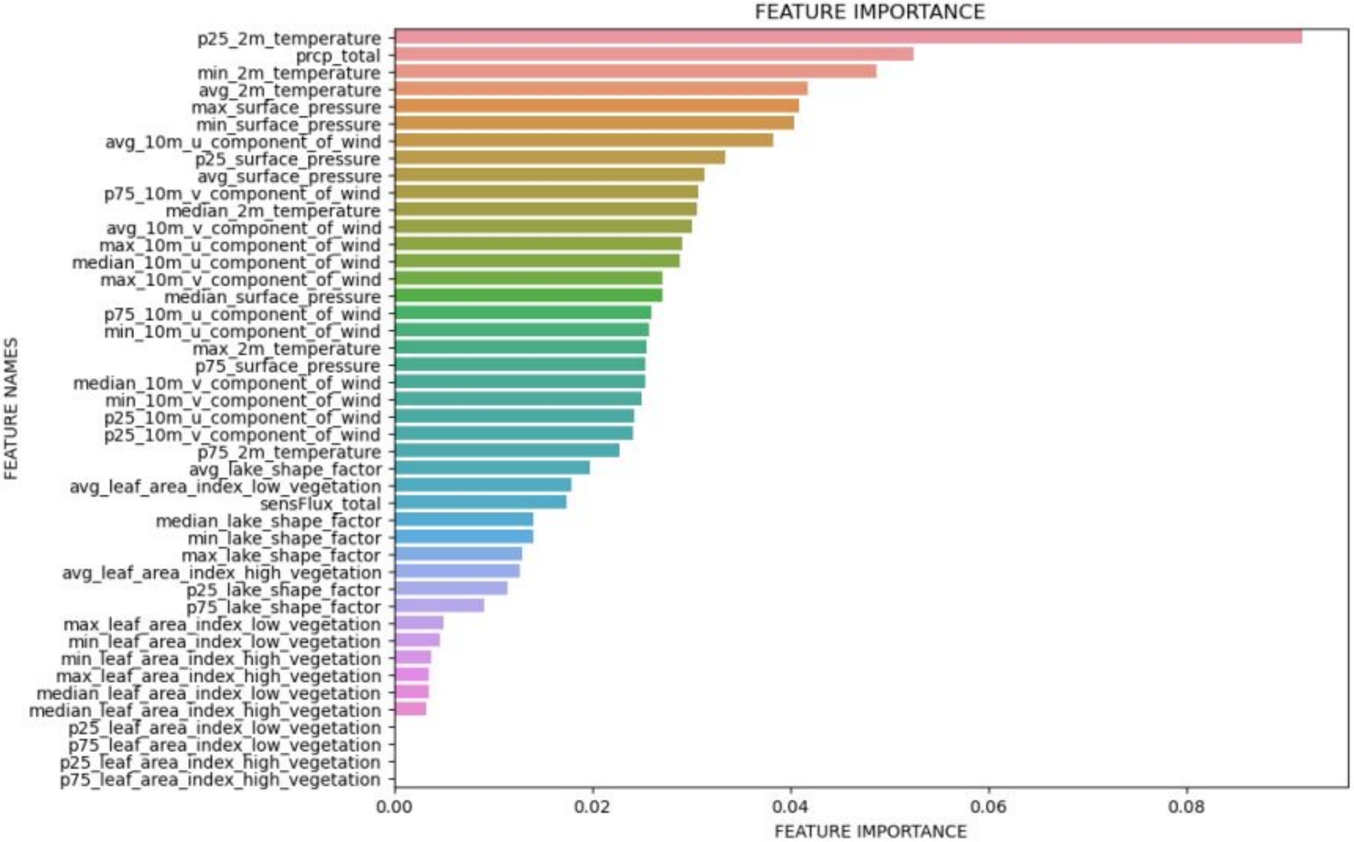


Project Examples

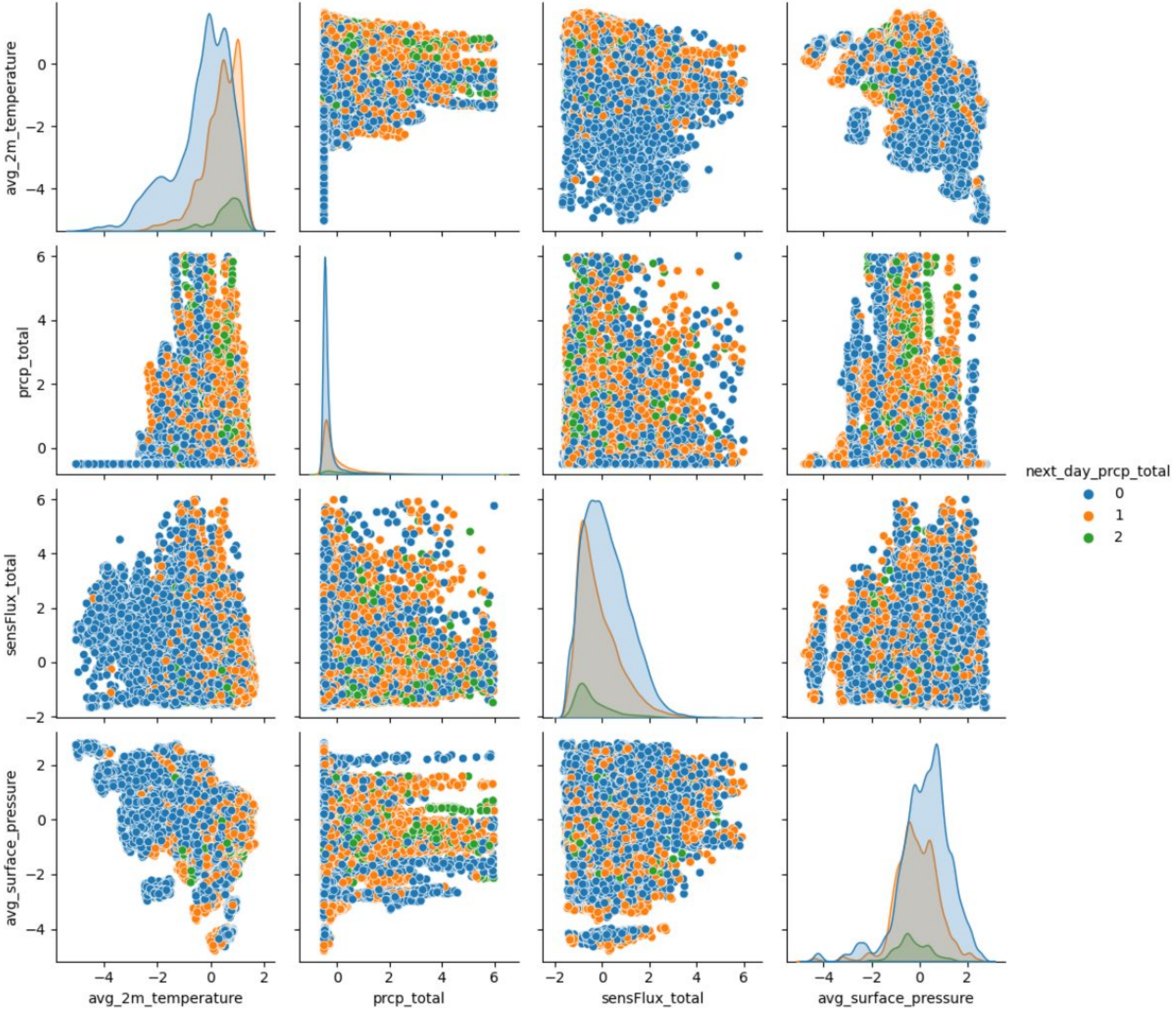
South FL 2015

XGBoost

Feature importance



Project Examples



2015 South Florida Models on 2022 South Florida & 2022 NYC

Project Details

True: next_day_prcp_total, nxtpr_cat

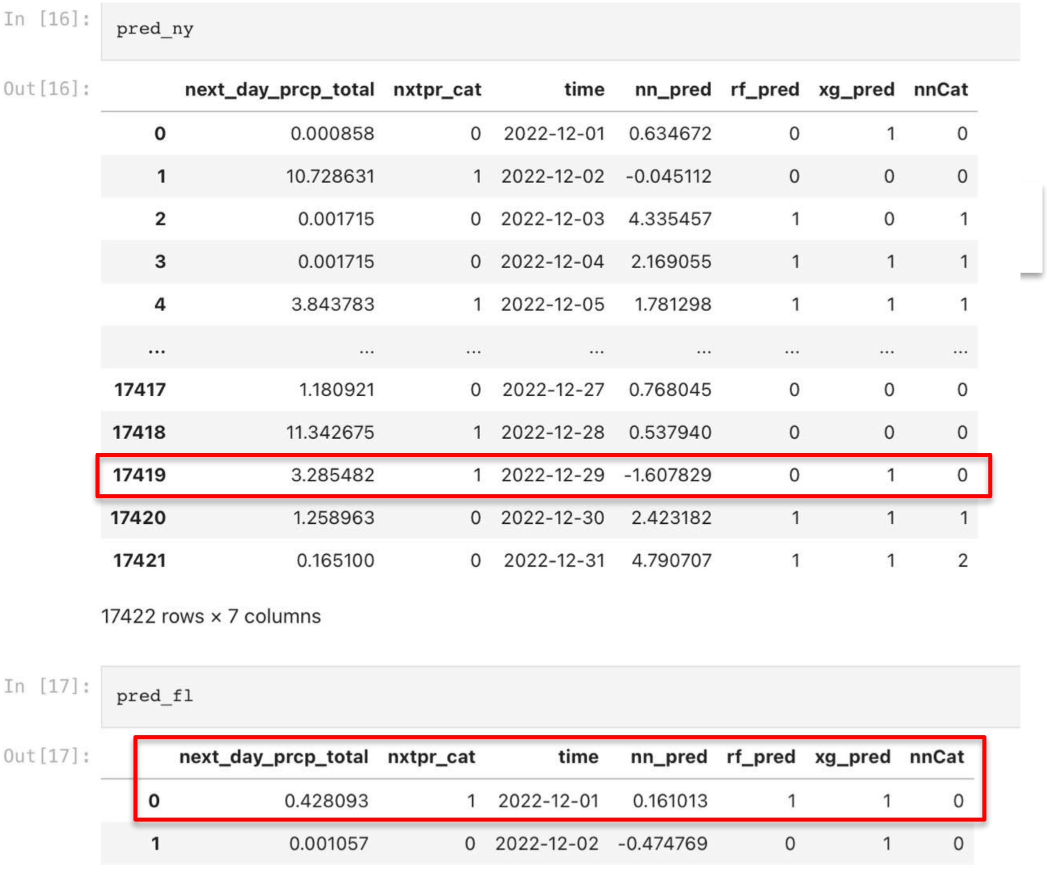
Predictions:

Neural Net Regression, nn_pred

Random Forest Classifier, rf_pred

XGBoost Classifier, xg_pred

Neural Net Classifier, nnCat



In [16]:
```
pred_ny
```

Out[16]:

	next_day_prcp_total	nxtpr_cat	time	nn_pred	rf_pred	xg_pred	nnCat
0	0.000858	0	2022-12-01	0.634672	0	1	0
1	10.728631	1	2022-12-02	-0.045112	0	0	0
2	0.001715	0	2022-12-03	4.335457	1	0	1
3	0.001715	0	2022-12-04	2.169055	1	1	1
4	3.843783	1	2022-12-05	1.781298	1	1	1
...	...	...	...	...	...	...	...
17417	1.180921	0	2022-12-27	0.768045	0	0	0
17418	11.342675	1	2022-12-28	0.537940	0	0	0
17419	3.285482	1	2022-12-29	-1.607829	0	1	0
17420	1.258963	0	2022-12-30	2.423182	1	1	1
17421	0.165100	0	2022-12-31	4.790707	1	1	2

17422 rows × 7 columns

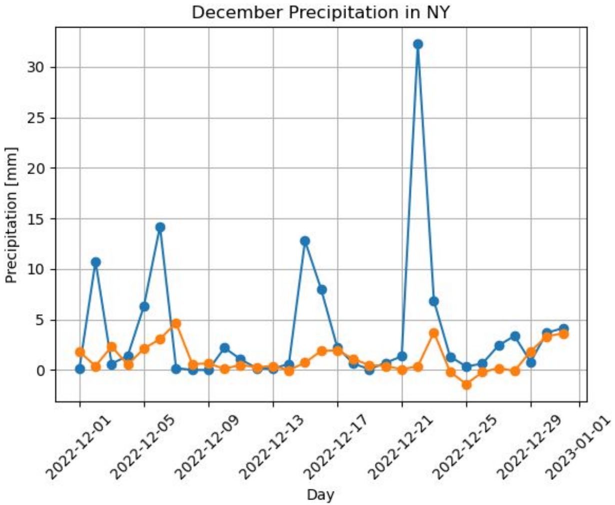
In [17]:
```
pred_fl
```

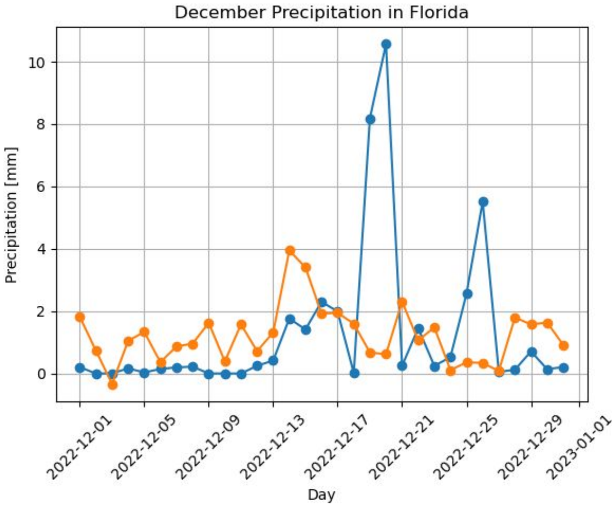
Out[17]:

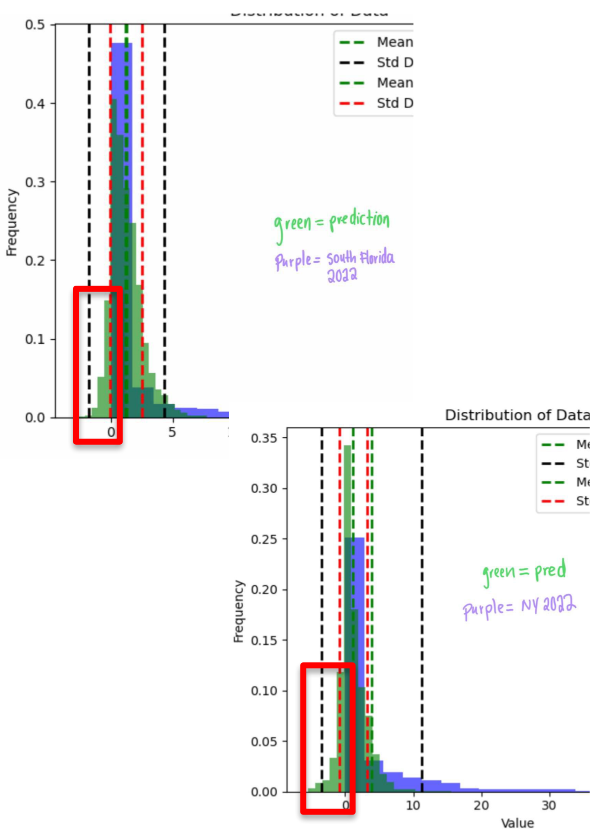
	next_day_prcp_total	nxtpr_cat	time	nn_pred	rf_pred	xg_pred	nnCat
0	0.428093	1	2022-12-01	0.161013	1	1	0
1	0.001057	0	2022-12-02	-0.474769	0	1	0

Project Details

Model 1 Regression result:


December Precipitation in NY


December Precipitation in Florida



Blue = true

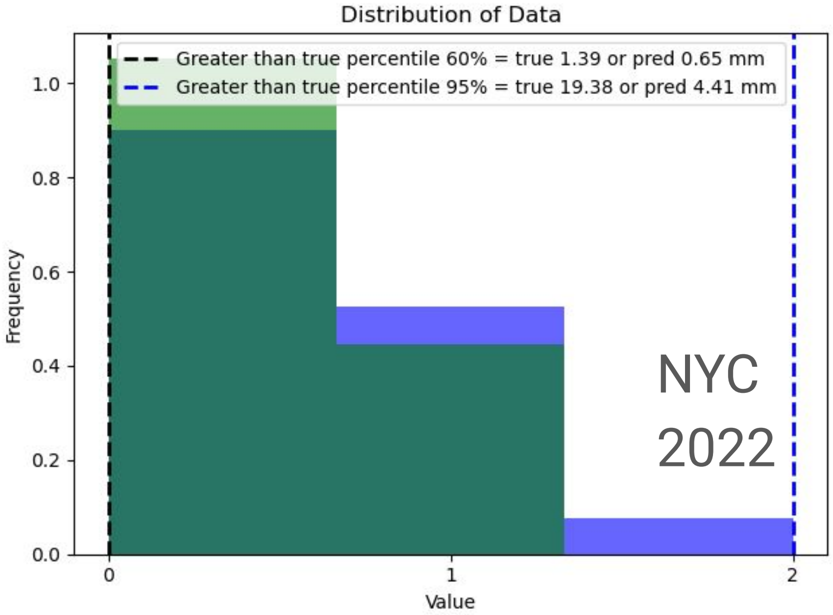
Orange = neural net regression prediction

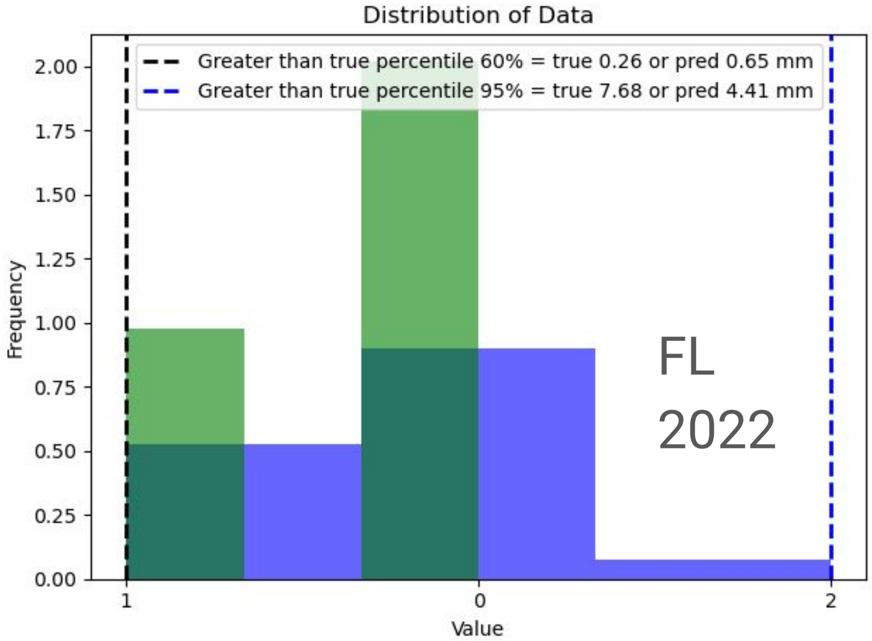
Project Examples

Green = pred

Purple = true

Model 2: Random Forest Classifier Result



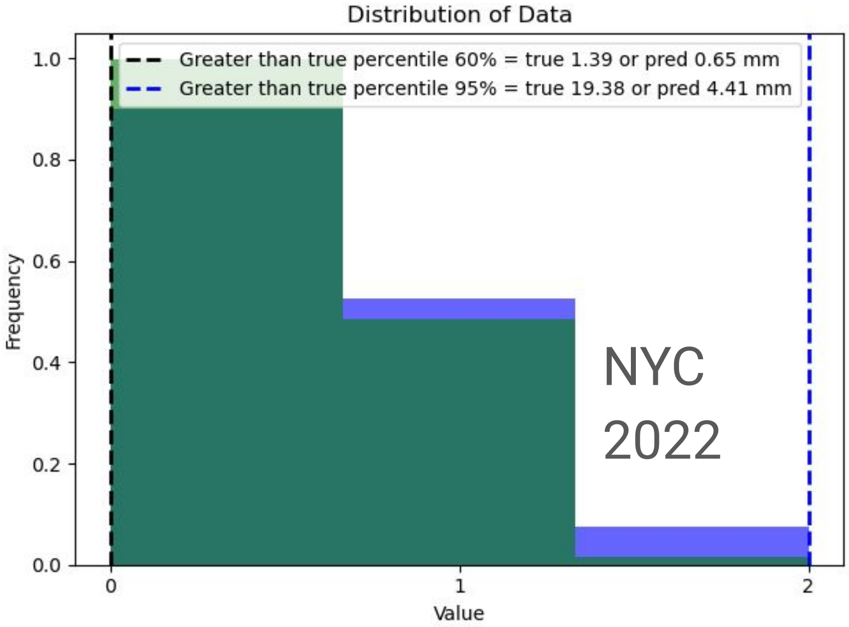


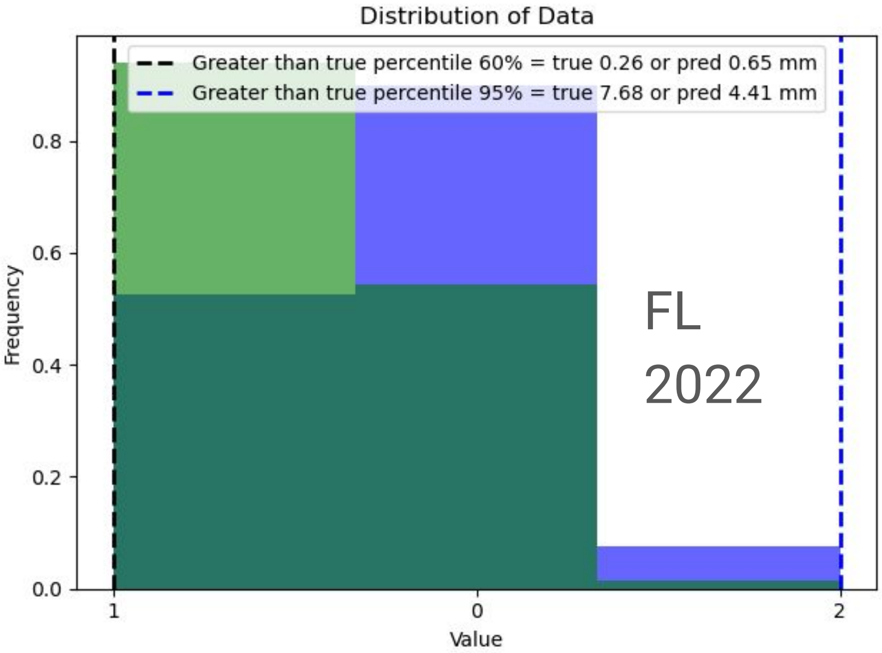
Project Examples

Green = pred

Purple = true

Model 3: XGBoost Classifier Result



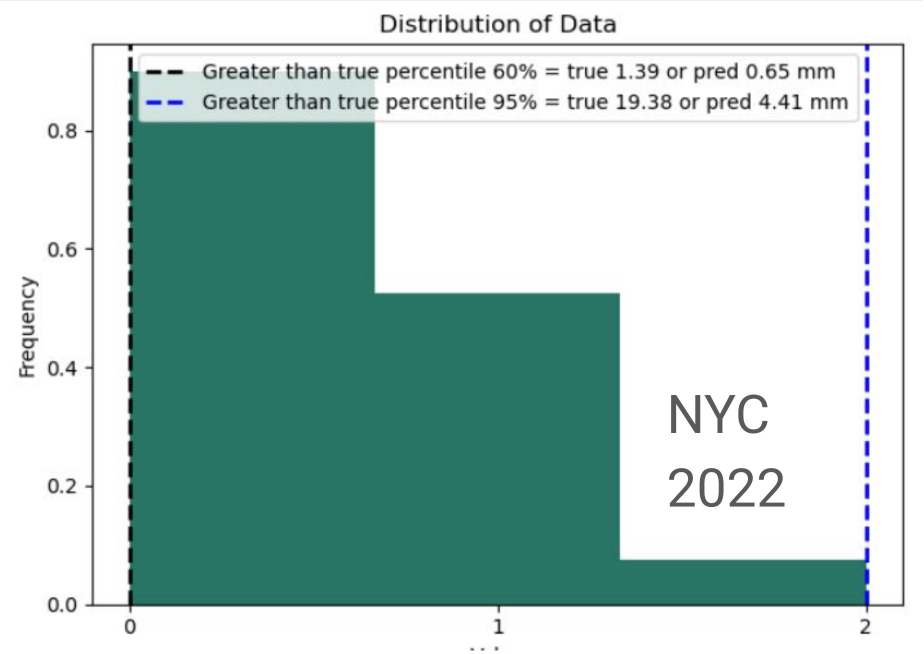


Project Examples

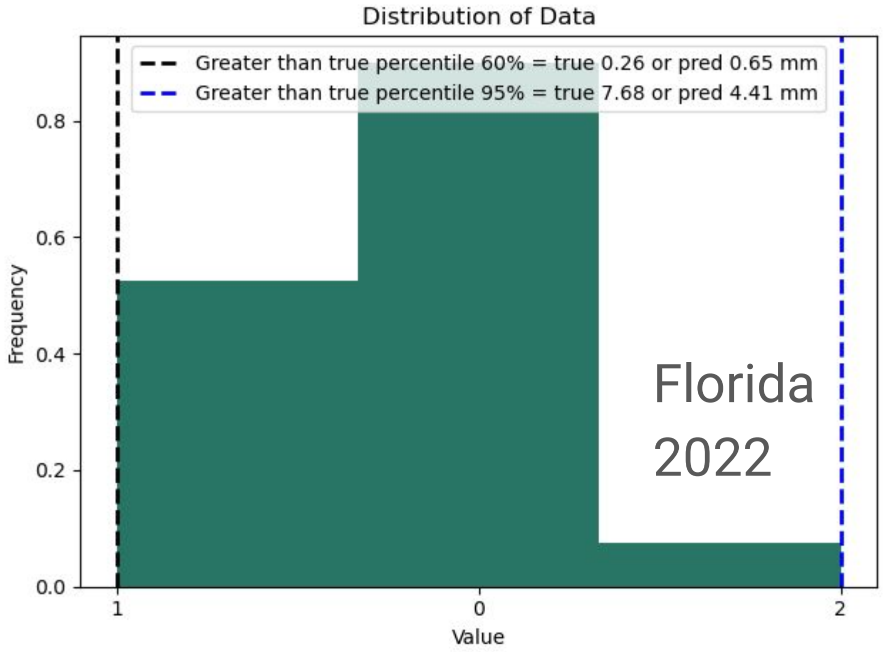
Green = pred

Purple = true

Model 1: Neural Net Classifier Result:



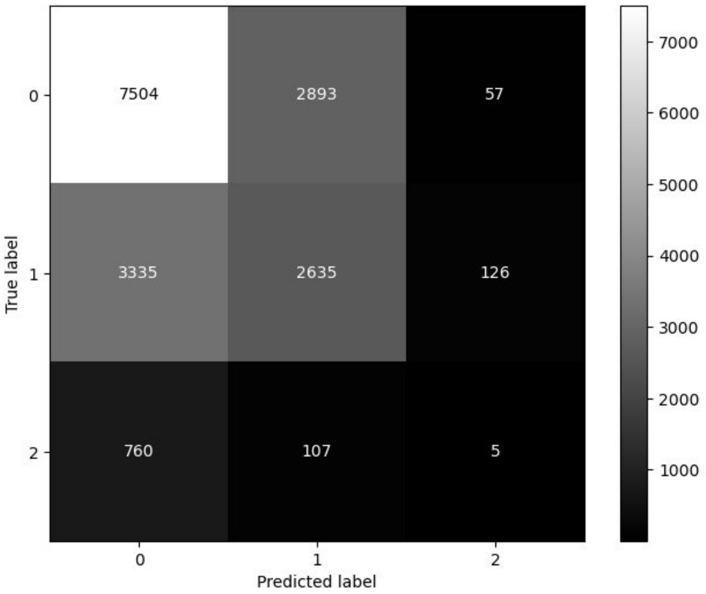
NYC 2022



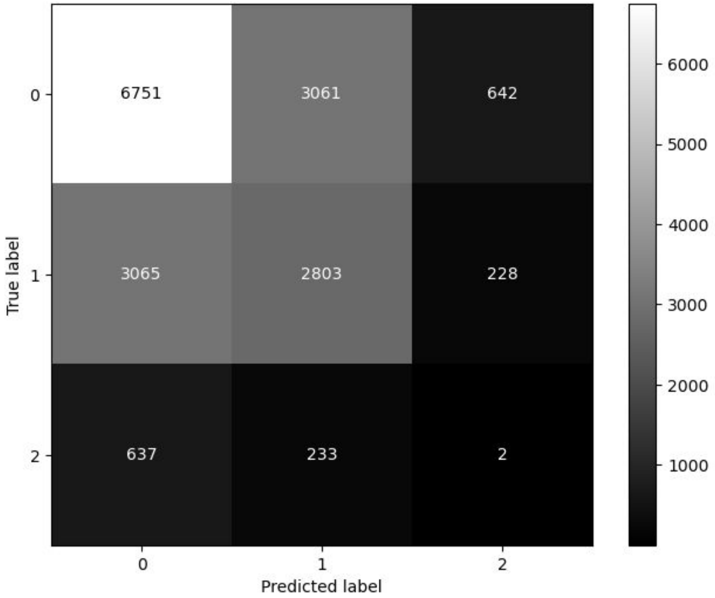
Florida 2022

Project Examples

NYC 2022



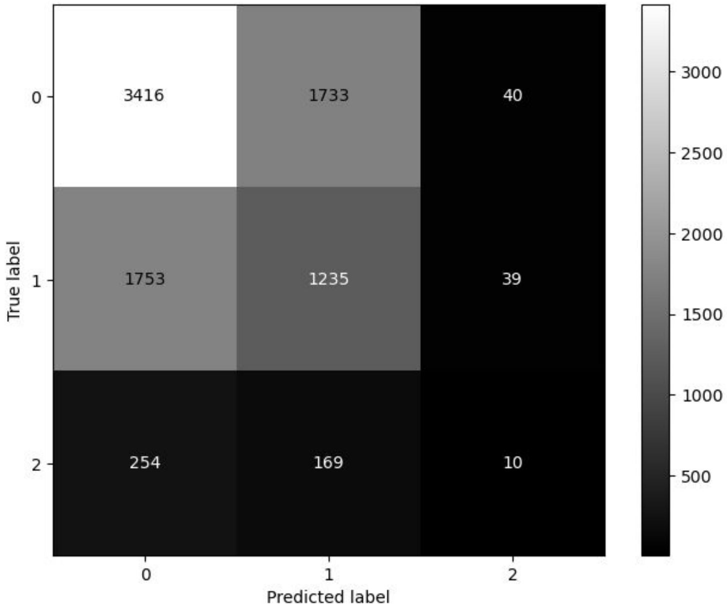
XGBoost



Neural Net

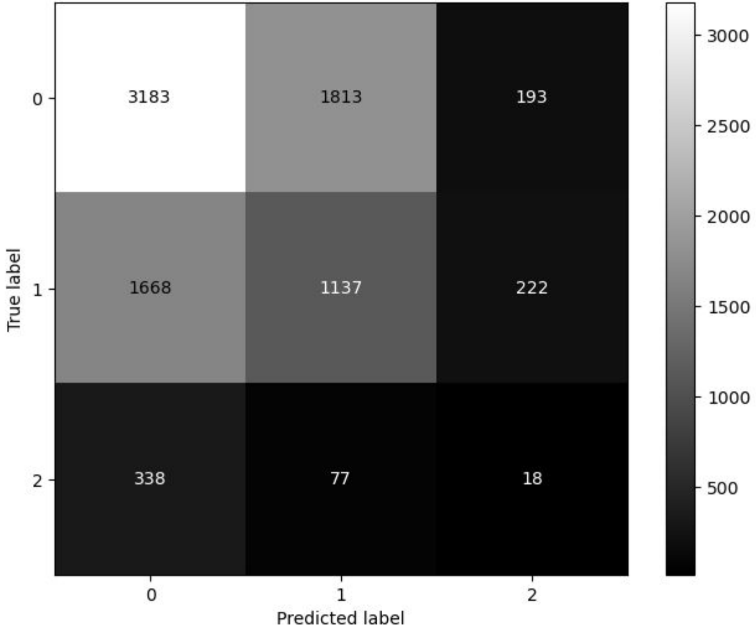
Project Examples

South Florida 2022



	0	1	2
0	3416	1733	40
1	1753	1235	39
2	254	169	10

XGBoost



	0	1	2
0	3183	1813	193
1	1668	1137	222
2	338	77	18

Neural Net

Classifier Comparison

Model	NYC 2022			South Florida 2022		
f1-score	0	1 (65%)	2 (95%)	0	1	2
1: Neural Net	0.64581	**0.45977**	0.00229	0.61341	0.37562	**0.04157**
2: Random Forest	**0.69831**	0.44907	0.00000	**0.66140**	0.37823	0.00000
3: XGBoost	0.68054	0.44924	**0.00943**	0.64380	**0.40071**	0.03831

Project Examples

- visually the neural network classifier is the best, but based on f-1 score XGBoost is most accurate. Classifiers preferred.
- confusion matrix harder to interpret due to class imbalance by design (60 and 95 percentiles)

Further work:

- model building
  - Increasing number of epochs
  - see whether the outlier removal is actually beneficial
  - Try different percentiles
  - Conserve lat and lon (might not work NYC)
  - Add previous prcp total feature

Further Reading

- Github: https://github.com/isabelayepes/PredTomorrowsRain
- Data documentation: ERA5-Land hourly data from 1950 to present: https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-land?tab=doc

References

- Run model function/ roc and auc curve: https://github.com/azalahmadkhan/Precipitation-Prediction-using-ML/blob/main/Precipitation_Prediction.ipynb
- Outlier removal, training and validation accuracy, standardization, confusion matrix: https://www.kaggle.com/code/karnikakapoor/rain-prediction-ann#MODEL-BUILDING
- Pairwise correlation, heatmap, confusion matrix: https://www.kaggle.com/code/chandrimad31/rainfall-prediction-7-popular-models#Model-Comparison
- Plot Feature Importance: https://www.analyseup.com/learn-python-for-data-science/python-random-forest-feature-importance-plot.html
- Saving the model to reuse it again: https://github.com/Biswajit6844/rainfall-prediction/blob/master/Xgboost%20model.ipynb