

PREDICTING POOREST COMMUNITIES IN **AFRICA** USING SATELLITE IMAGERY



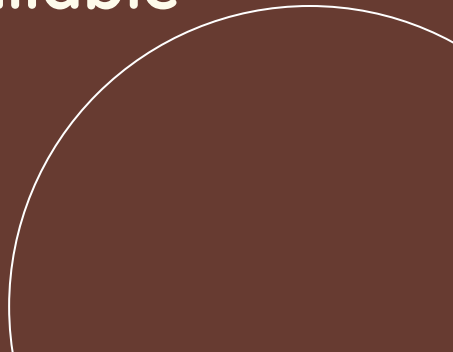

Atigon Hongchumpol
Simon Lazarus
Stanley Azuakola



PROBLEM STATEMENT

- Client: The World Bank
- Target: Anti-poverty initiative for poorest villages in Africa
- Challenge: Scarcity of relevant and timely data

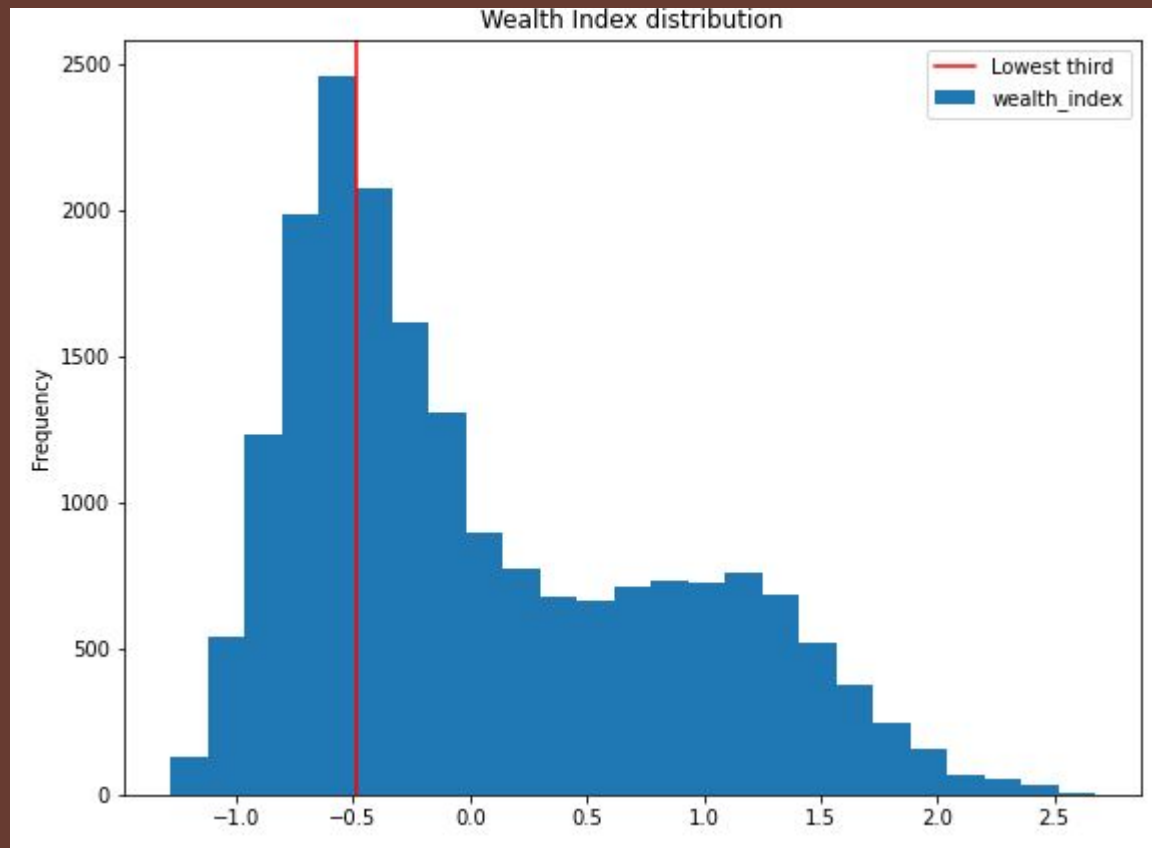
Can we train a model that correctly predicts the poorest villages in Africa, using publicly available satellite imagery?





Wealth Index







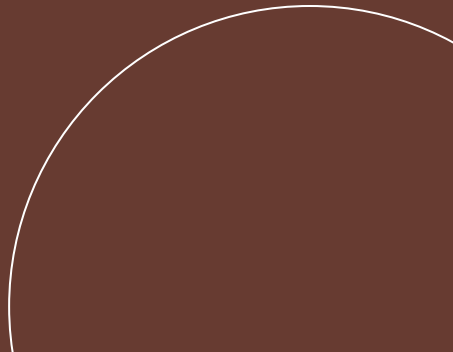
EVALUATION METRICS

We wanted a model that balanced these two metrics:

- **Accuracy:**

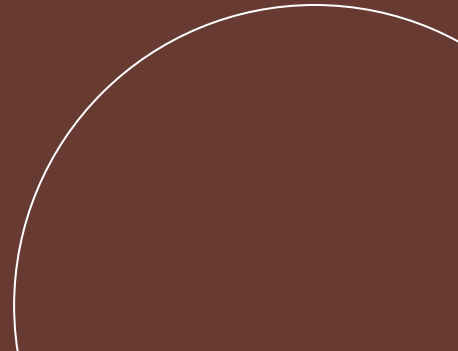
Null Model Accuracy = 66.7%

- **Recall/True Positive Rate**





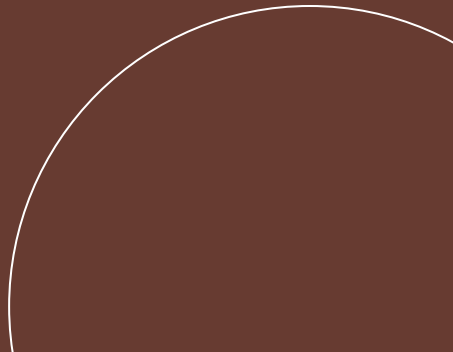
Data Collection & Details





Wealth Index Data

- Formed from Demographic & Health Surveys (DHS) data on asset wealth
 - Annual standardized survey in 23 African countries
 - Many questions about housing, water/electricity, vehicles, etc.
- Results averaged across 3-year periods in specific (latitude, longitude) locations
 - Periods: 2009-2011, 2012-2014, 2015-2017
 - A “village” in our data means a 3-year aggregate of a location.
- “Wealth Index” is 1st Principal Component of survey data
 - Data from 19,699 “villages”

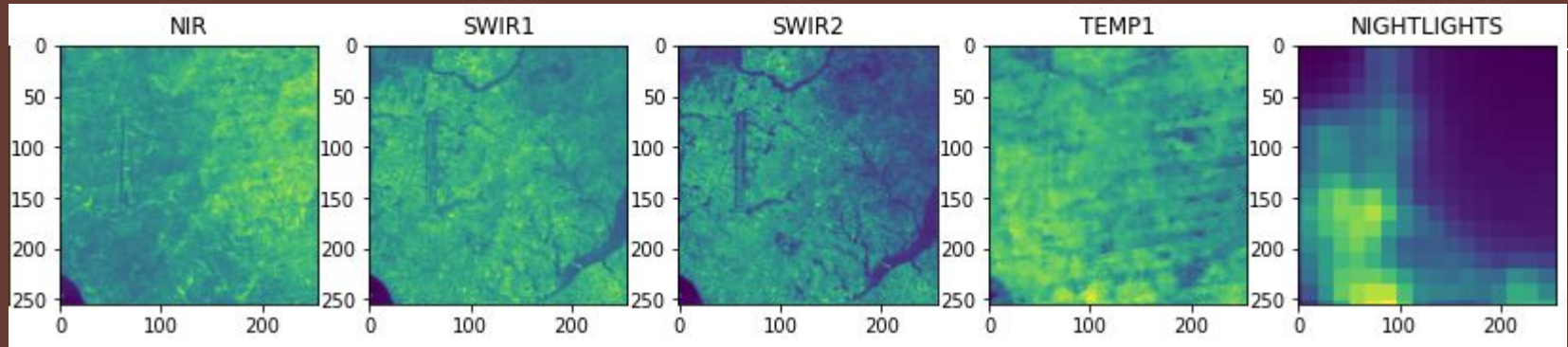
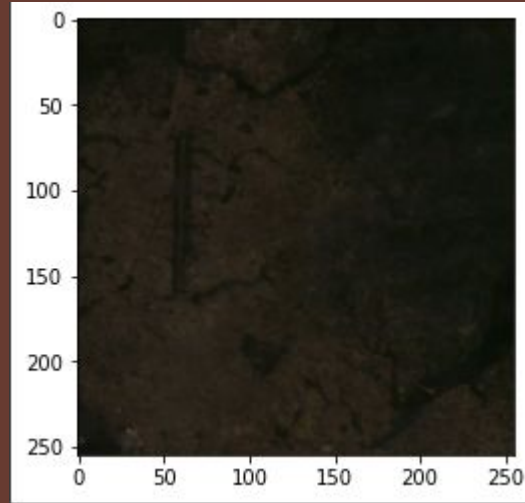


Satellite Imagery

- 8 channels: 3 RGB, 3 infrared (NIR, SWIR1, SWIR2), 1 thermal (TEMP1) and 1 nighttime lights (NIGHTLIGHTS)
 - Images from Landsat satellite (30m/pixel resolution)
 - NIGHTLIGHTS images are from DMSP (2009-11) or VIIRS (2012-14) satellites and at lower resolution
- 1 set of six 255x255 images for each “village” (RGB channels grouped)
- Pixels in each channel averaged across that village’s 3-year period
 - Median of all cloud-free pixel values used to create 1 composite
- After removing “bad” RGB images, **19,467 villages’ worth of images**


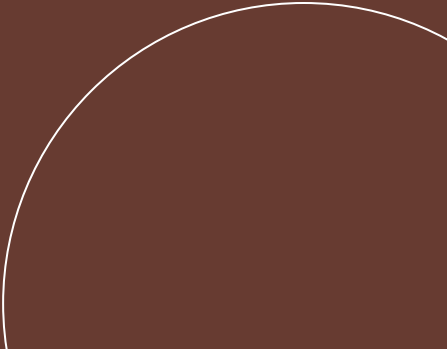
Example Imagery (1 Village in Angola)

Wealth Index: 1.88
(not in poorest third)



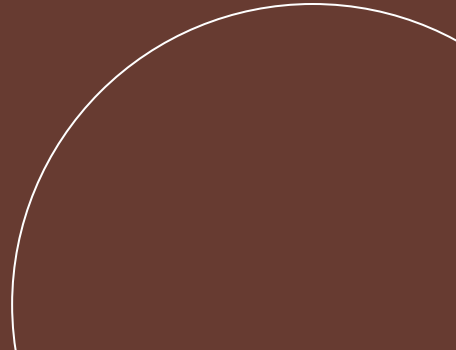


Train/Test/Validation Split

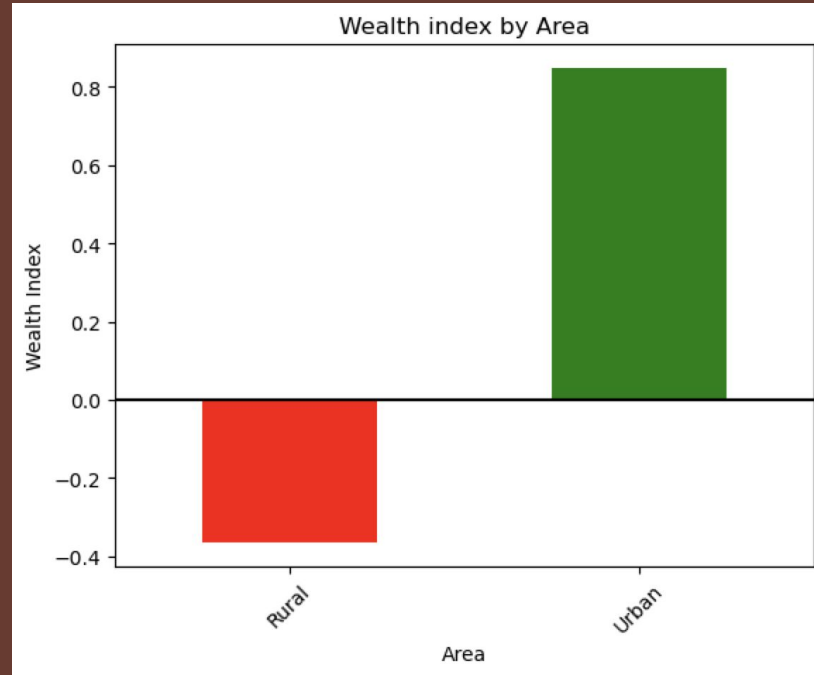
- 90% of observations (17,520) used as training data
 - 5% of observations (974) used as test data
 - Select models on the basis of Accuracy, Recall scores on test data
 - 5% of observations (973) used as validation data
 - Evaluate performance of production model vs. baseline model
- 
- 



Exploratory Data Analysis

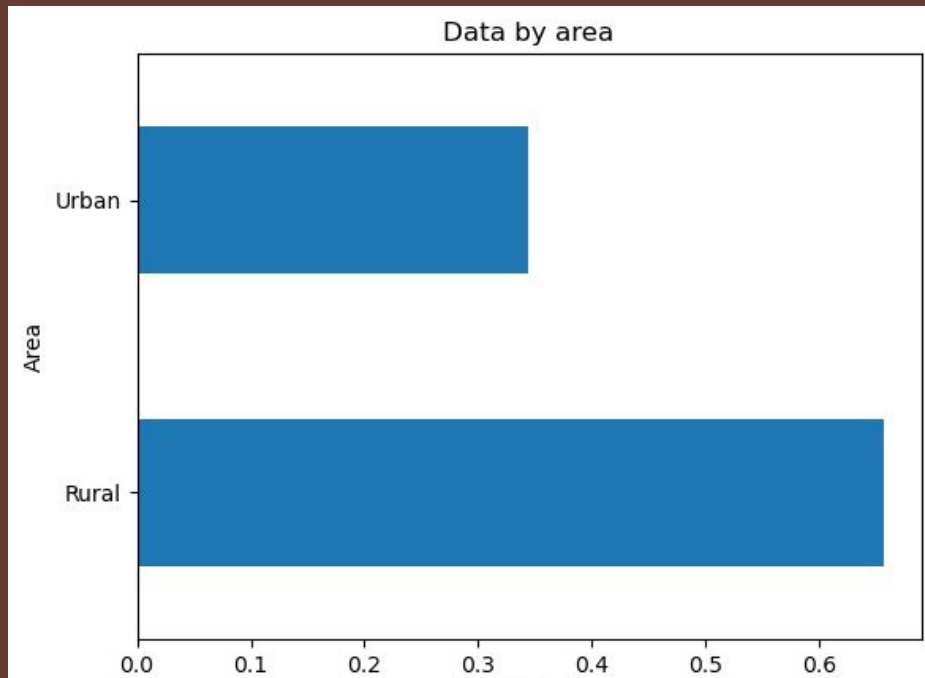


Average wealth index by living area



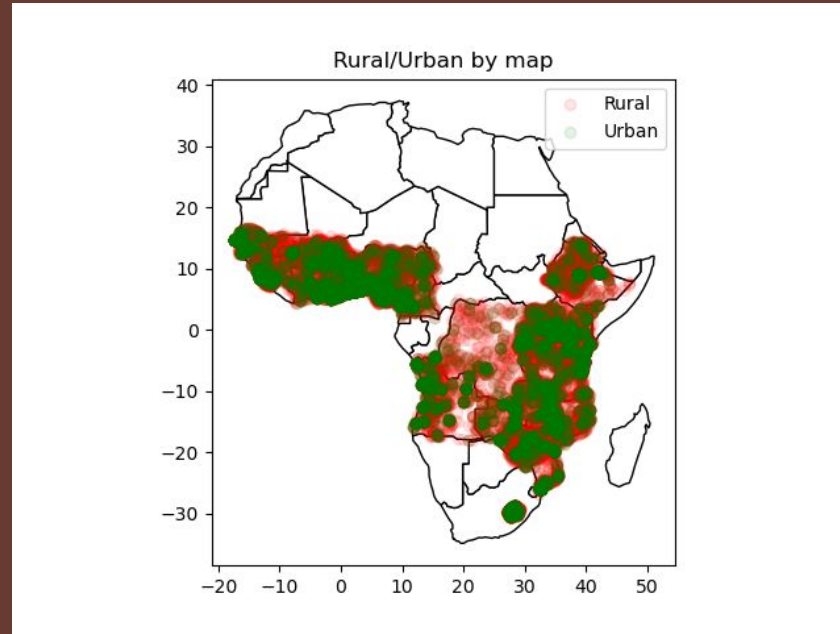
-People living in the Urban area has a lot of different wealth index by comparing to people who lived in the Rural area.

Data by area



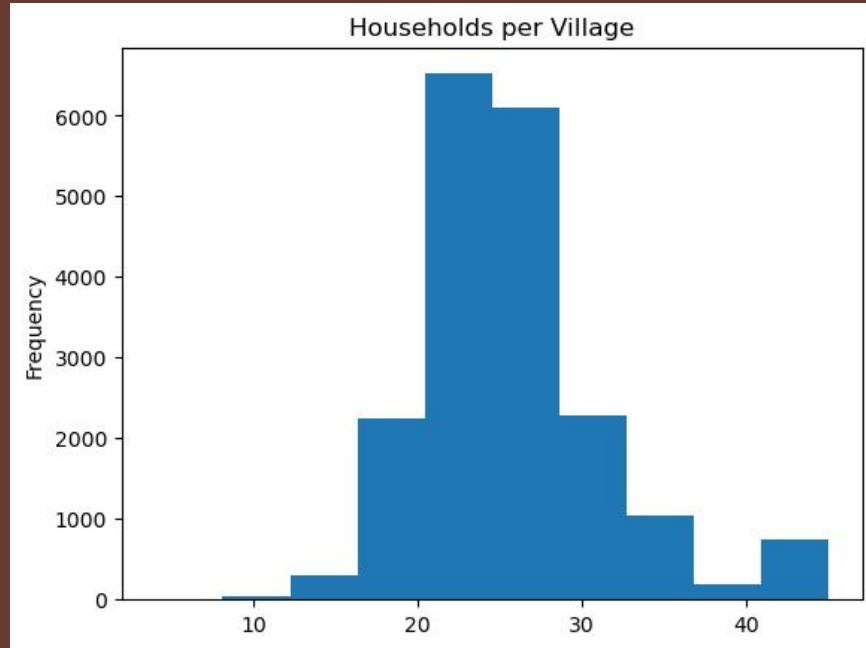
-More data in the Rural area, the size is about double from the Urban area.

Rural / Urban map



-Green color is for the Urban area and red for the Rural area, looks like people are denser close to the coastline.

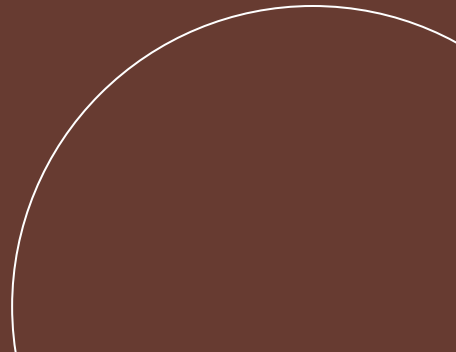
Households per village



-Average households per village is about 25



Baseline Model



All model

	LogisticRegression	RandomForestClassifier	AdaBoostClassifier	KNeighborsClassifier
only_country_train	0.709246575	0.709075342	0.709246575	0.62859589
only_country_test	0.709445585	0.704312115	0.709445585	0.605749487
country_is_urban_train	0.799885845	0.799885845	0.799885845	0.782762557
country_is_urban_test	0.783367556	0.783367556	0.783367556	0.764887064
country_is_urban_lat_long_train	0.790353881	0.999828767	0.817636986	0.880936073
country_is_urban_lat_long_test	0.781314168	0.820328542	0.808008214	0.813141684
only_lat_long_train	0.675627854	0.999771689	0.729908676	0.856164384
only_lat_long_test	0.6724846	0.794661191	0.717659138	0.790554415
country_lat_long_train	0.717294521	0.999885845	0.743436073	0.856621005
country_lat_long_test	0.719712526	0.813141684	0.743326489	0.793634497

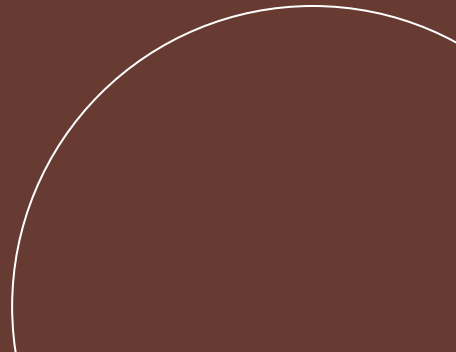
Best model score & confusion matrix

	RandomForestClassifier
country_lat_long_train	0.999885845
country_lat_long_test	0.813141684
country_lat_long_val	0.803699897
accuracy_val	0.803699897
recall_val	0.672240803

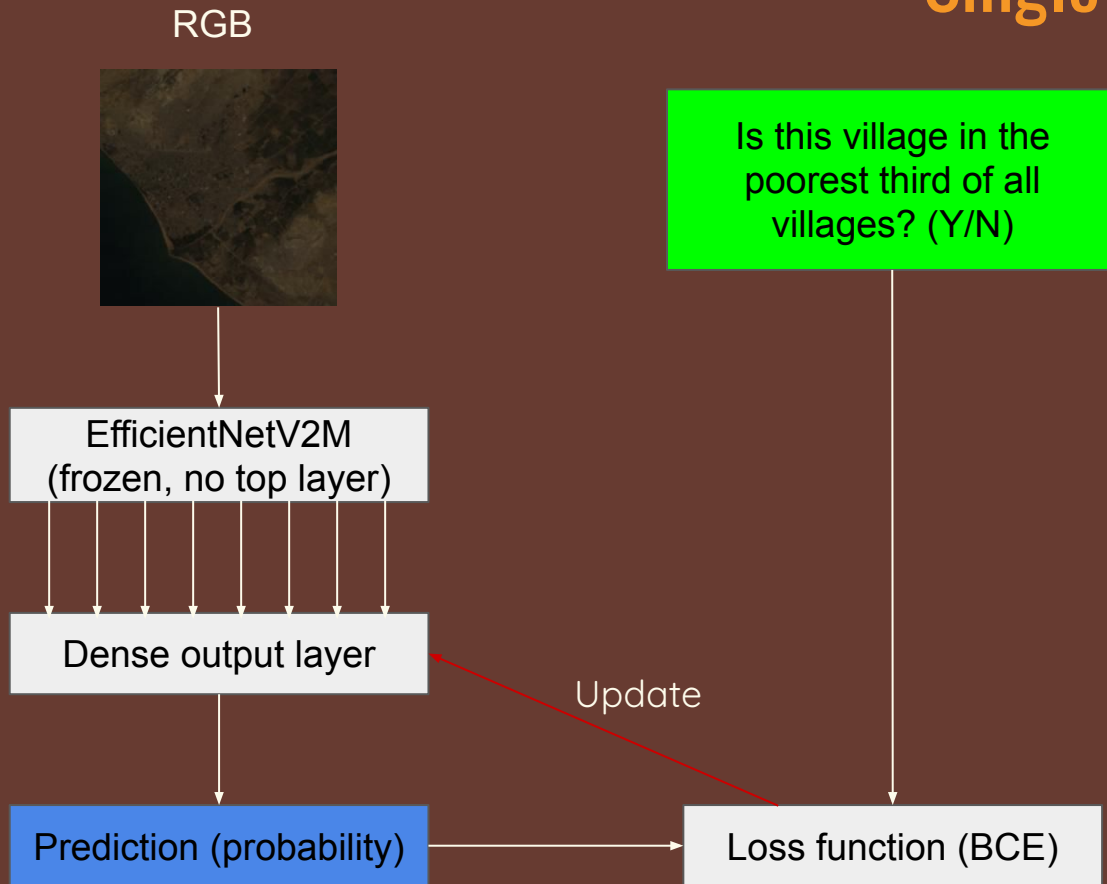
	Predicted not Poorest	Predicted Poorest
Actual not Poorest	581	93
Actual Poorest	98	201



Neural Network Models





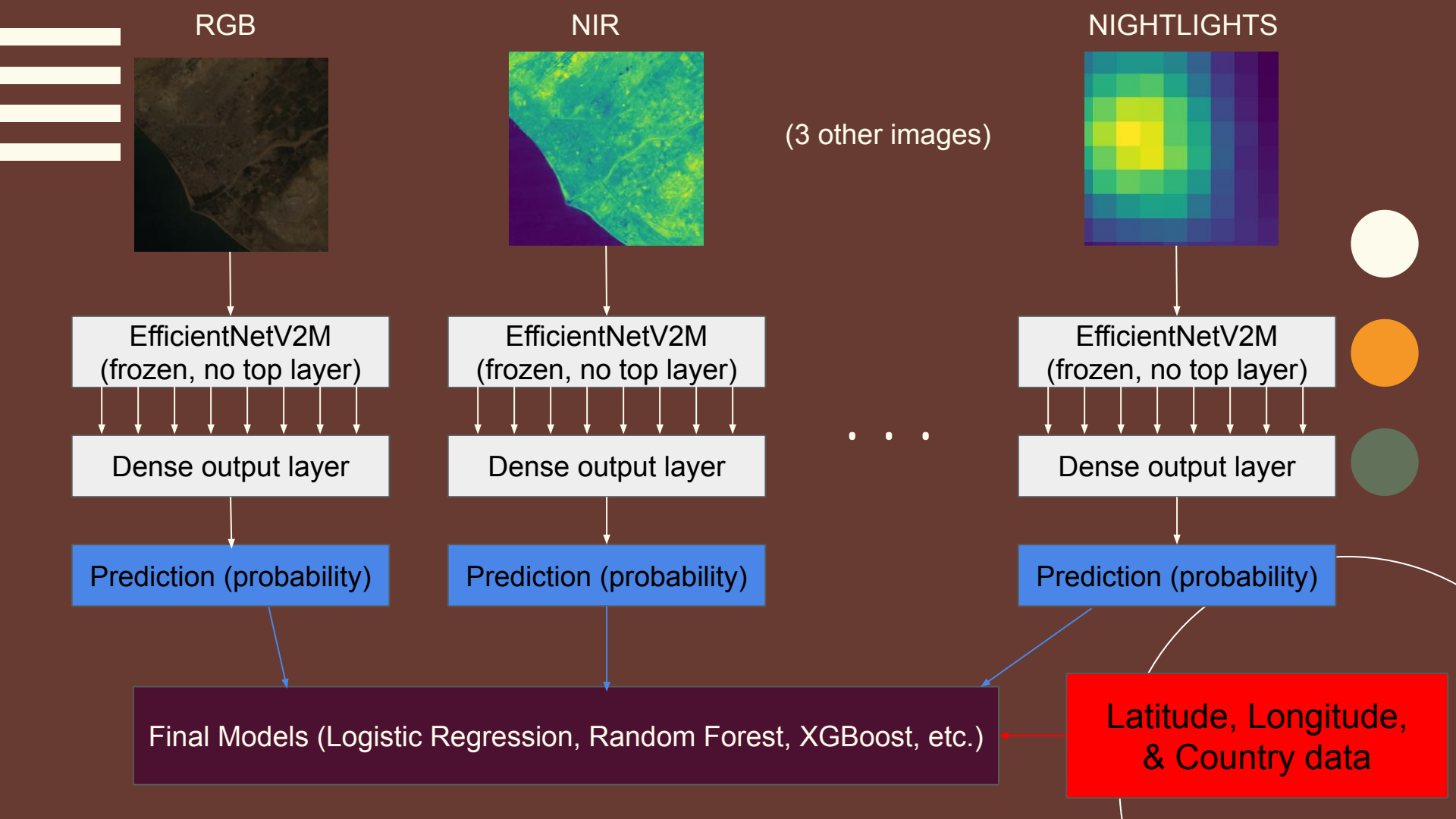
Single CNN Structure



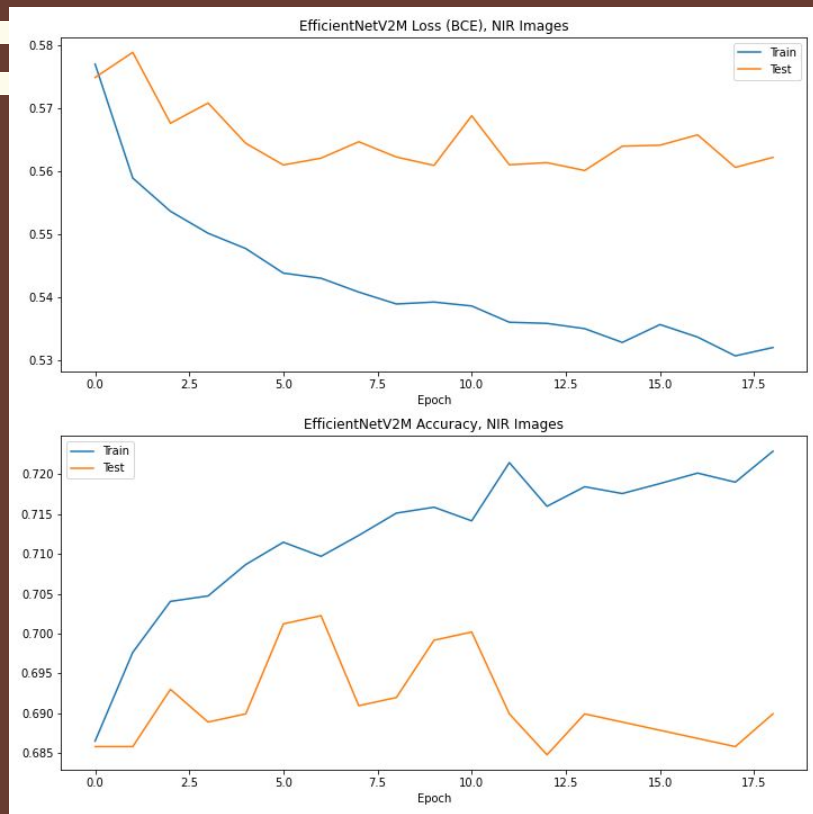


NN Modeling Approach

- Transfer Learning with Convolutional Neural Networks
 - Train one copy of EfficientNetV2M (top layer) for each image type.
 - 6 “predictors”: RGB, NIR, SWIR1, SWIR2, TEMP1, NIGHTLIGHTS
 - For each village, feed the 6 predictors’ predicted probabilities as features into a top-level model (along with coordinates + country)
 - Train top-level models on these data (see next slide)
- 
- 

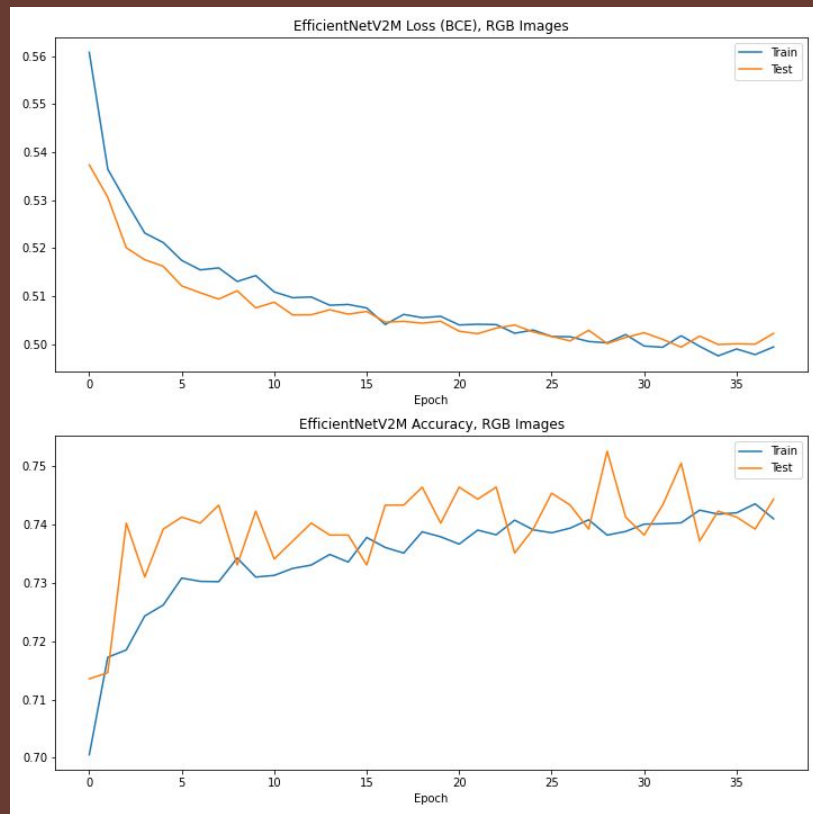


How good of predictors are the images *by themselves*?



NIR Test Accuracy: 69.0%

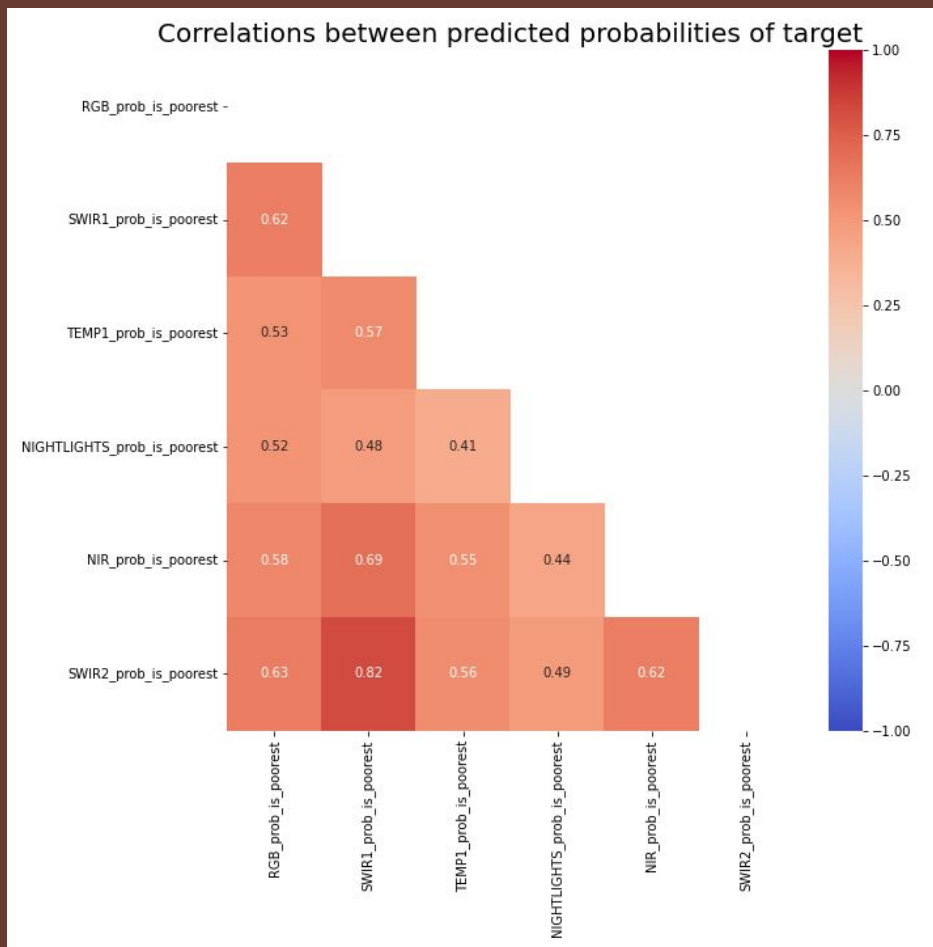
(Null model Accuracy: 66.7%)



RGB Test Accuracy: 74.4%

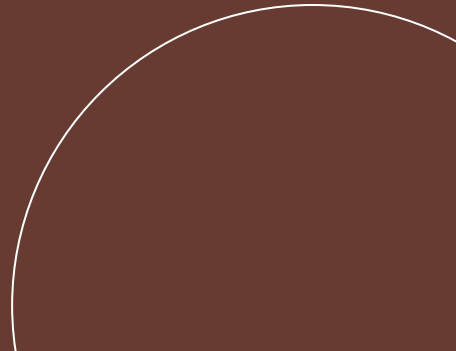
How good of predictors are the images *together*?

- Predicted probabilities are only *moderately* correlated with each other
- Reasons to believe that *combining them* will yield much better results

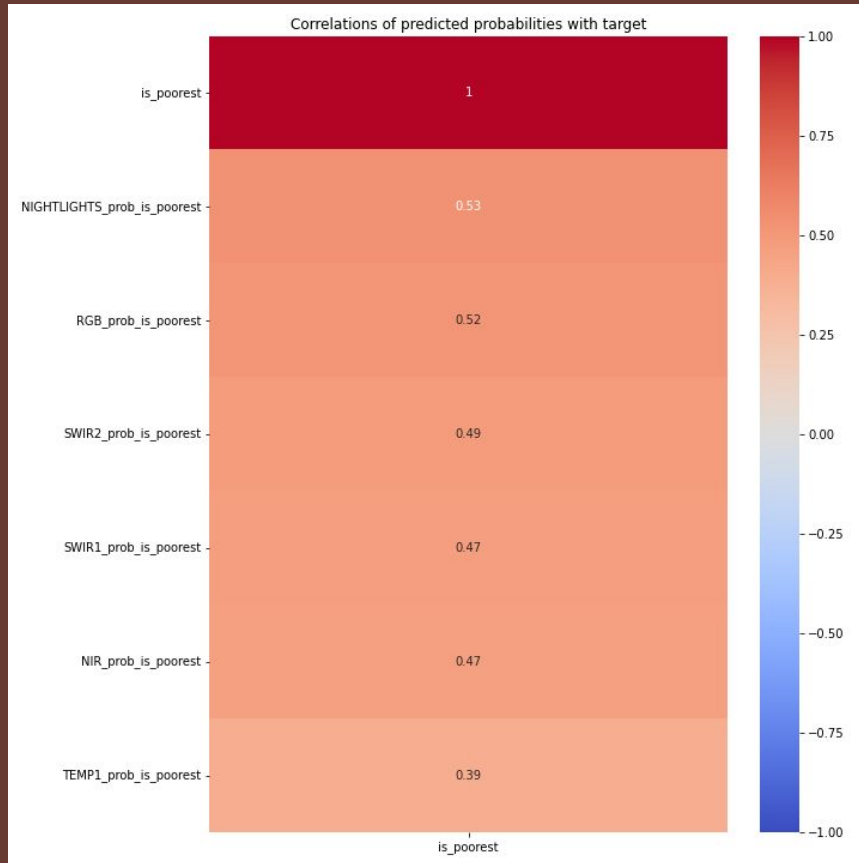




Final Models



Correlations of predicted probability with target



-Not high for correlations.
-The NIGHTLIGHT has the highest.



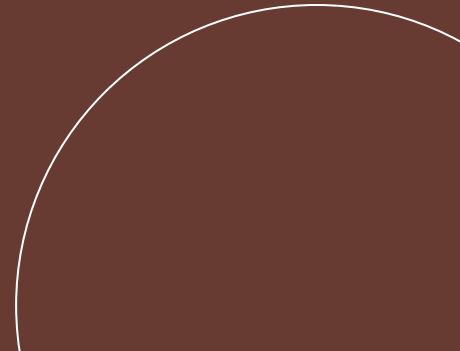
Final model

	acc_train	acc_crossval	acc_test	recall_train	recall_test
model					
Logistic Regression	0.836587	0.835445	0.820329	0.717844	0.695122
Random Forest	0.976027	0.842123	0.837782	0.951894	0.743902
Random Forest (no Country dummies)	0.945491	0.836986	0.816222	0.902252	0.710366
Logistic Regression (no Country dummies)	0.811073	0.811644	0.778234	0.668884	0.646341
XGBoost	0.891553	0.848402	0.840862	0.833675	0.762195
kNN	0.856164	0.827169	0.815195	0.770386	0.728659
Stacked Model	0.925970	0.848858	0.835729	0.872057	0.740854

- Not using “country” does not improve models
- XGBoost performs the best



Analysis of Production Model



OUR PRODUCTION MODEL IS MORE ACCURATE AND SENSITIVE

	NULL MODEL	BASELINE RF MODEL	PRODUCTION MODEL
ACCURACY	66.7%	80.4%	85.0%
RECALL	0%	67.2%	78.6%

Baseline vs **Production Model** comparison

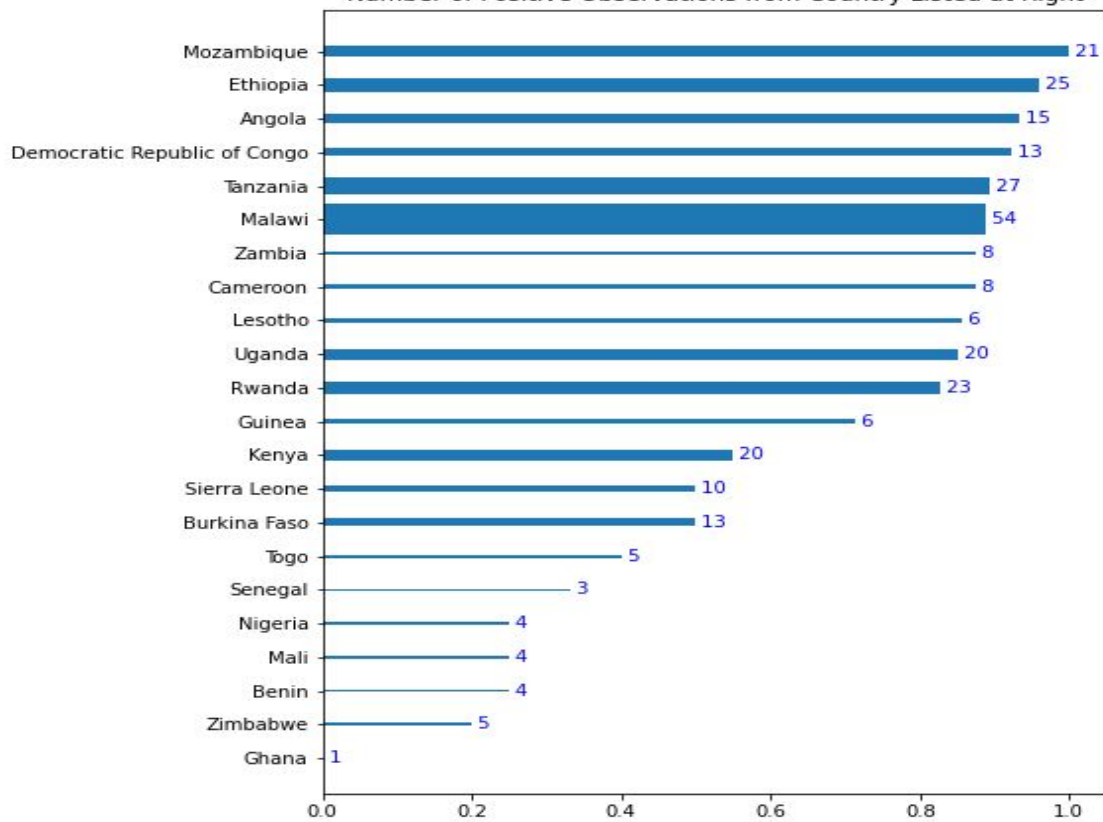
Baseline	Predict not poorest	Predict poorest
Actual not poorest	<small>TN</small> 581	<small>FP</small> 93
Actual poorest	<small>FN</small> 98	<small>TP</small> 201



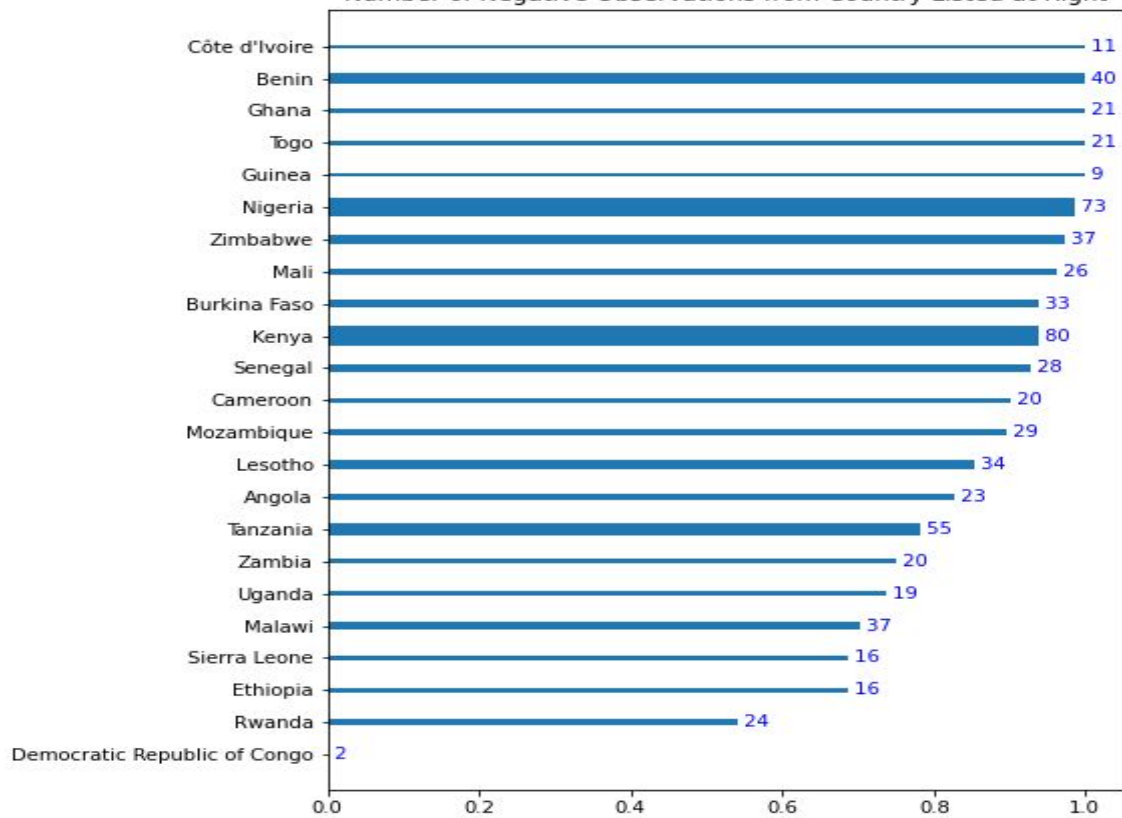
XGBoost	Predict not poorest	Predict poorest
Actual not poorest	<small>TN</small> 592 +2%	<small>FP</small> 82 -12%
Actual poorest	<small>FN</small> 64 -35%	<small>TP</small> 235 +17%

-Overall score improves

True Positive Rates from Production Model, by Country (Validation Data)
Number of Positive Observations from Country Listed at Right



True Negative Rates from Production Model, by Country (Validation Data)
Number of Negative Observations from Country Listed at Right





CONCLUSION

The problem: Can we train a model that correctly predicts the poorest villages in Africa, using publicly available satellite imagery?

- **Our solution:** YES! Using just geographic coordinates and other information that can be automatically downloaded and fed into our model - we can predict with an accuracy of 80% and a sensitivity of 78.6% whether a village in Africa suffers from extreme poverty.

