Predicting True Regional Income via Satellite Images of Lights at Night

Leo Goldman (leo1) [*]
leo.goldman@columbia.edu

Abstract

Predicting economic activity via satellite images of nights at light is a promising tool for constructing accurate global poverty measures and assessing currently used measurements, such as GDP or survey consumption data. The current literature largely focuses on linear regressions that conclusively establish a relationship between lights and economic activity but do not offer highly accurate predictions. More recent work has used deep learning tools to predict binary components of infrastructure. I propose a CNN architecture that achieves superior performance in predicting GDP to regressions in the literature. Furthermore, the CNN achieves similar accuracy predicting GDP as income survey results, suggesting that each measure may be similarly correlated to lights and therefore true income. Similar to existing models, however, the deep learning model does poorly at predicting income of small countries, for which predictions would be most useful.

1 Introduction

Measuring global poverty world requires high quality estimates on true regional income. The two primary methods of estimating income are surveys and official accounts of GDP. These two measurements, however, can differ greatly, leading to a wide range of estimates.

Using exogenous variables, such as nighttime lights captured by satellites, researchers can track human activity and derive income estimates that are not susceptible to the same measurement errors and biases as GDP accounts and surveys. For example, surveys can suffer from misreporting and selection bias. GDP accounts can suffer from biases due to government influence. Moreover, gathering either measure can be expensive and difficult, which could introduce uncertainty and bias that may be systematically affected by region or industry.

Reliable measures of true income are needed because in many areas of the developing world accurate measurements of income are unavailable. The ability to predict income without costly measures (like most surveys) would make it easier to derive global poverty estimates. In this paper, I show how deep neural networks, specifically convolutional neural networks (CNNs), can take in satellite images of light at night and output accurate predictions of GDP or surveys.

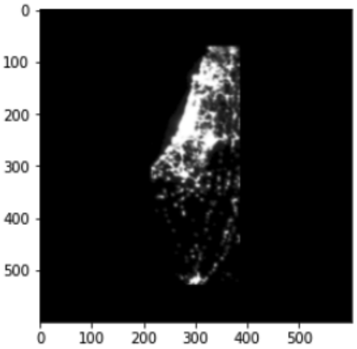
2 Related Work

For decades, researchers have used nighttime lights data to enhance income predictions (Ghosh et al. 2013). The strong, nonlinear correlation between lights data and income is well-known. Henderson et al. (2012) present these correlations through regression analysis, taking logs of income and lights in most cases, and adding time- or entity-fixed controls in some models. Even in recent years, however, the literature, has largely relied on linear regression (Ivan et al. 2020). These models fail to exploit the full richness of the data by reducing satellite images over a country to one statistic, often the sum of light (pixel radiance), or some transformation thereof. Although this choice is sensible, it prevents learning from spatial context that may correlate with income. For example, one could imagine that a country with pockets of intense light, generated by cities, may have a different income from a country with a similar sum of light spread uniformly.

[*] I am a current Research Analyst at the Federal Reserve Bank of New York. Any views expressed in this paper are the author's and do not necessarily reflect those of the Federal Reserve Bank of New York or the Federal Reserve System. All errors are my own. For the fall of 2021, I can also be reached at leo1@stanford.edu.

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

Figure 1: Nighttime Lights in Israel, 2010



In 2018, Oshri et al. took a deep learning approach, using CNNs with satellite imagery as input, but they formulated their problem as a classification task. Specifically, the authors used Residual Neural Networks to predict whether or not a country had quality components of infrastructure, such as electricity, roads, sewage, etc. Perhaps unsurprisingly, they found that nighttime lights was only a good predictor of access to electricity. This complements findings by Henderson et al. (2012), but does not rule out the ability of lights or electricity data to predict income.

Regarding the relationship between GDP and survey income data, Pinkovskiy and Salah-i-Martin (2016) used the nightlights data as an independent proxy of true income to study whether GDP or surveys are better signals of true income. Primarily using the log of light sum, the authors found that an optimal prediction of true income would weight GDP much more heavily than surveys, although the weight on surveys is still statistically significant. In Section 5, I will discuss how my model could not only contribute to the main prediction task at hand, but also to the discussion on relative usefulness of GDP versus survey data.

3 Dataset

3.1 Satellite Images

Satellite images of lights at night collected by the DMSP-OLS satellite program and maintained by the National Oceanic and Atmospheric Administration (NOAA) can be found here (NOAA's NGDC). Satellite images of every location between 65 degrees south latitude and 65 degrees north latitude are available at a resolution of 30 arcseconds (each pixel corresponds to about one square kilometer at the equator). The NOAA processes images to remove cloud cover, snow, and ephemeral lights, such as forest fires (Elvidge et al. 1997).

This data is accessible via the Google Earth Engine (GEE) API (Gorelick 2017). Using GEE, one can pull NOAA nighttime light images, as well as Global Administrative Unit Layers boundaries for each country in the form of shape files. With these boundaries, the satellite images can be "clipped," or restricted to a particular country.

Elvidge et al. (2009) offer a methodology to calibrate the satellite images to account for differences between satellites across the years. The authors fit a quadratic model so that two calibrated satellites in the same year produce the same output, reducing noise in comparisons across years or satellites.

For my project, after clipping and calibrating images for different countries and years, to generate a dataset with consistent dimensions, I resize each image to 600 pixels by 600 pixels. A pixel in the satellite images represents roughly one square kilometer, so many countries are within these dimensions, but some scaling is inevitable. For larger countries, I scale down the image so that the larger of height and width becomes 600 pixels. For the shorter dimension or smaller countries, I pad the image with zero-valued pixels, which represent a black pixel with no light (see Figure 1). This process aims to achieve a uniform image size with as little distortion as possible. An image's total sum of light is certainly important, so I sought to introduce minimal pre-processing so that a neural network could at least learn to use a simple transformation of the sum of light.

Importantly, the largest countries, including the USA, Russia, and Canada, were intentionally left out of the dataset for multiple reasons. First, pixels in the satellite images are top-coded, meaning the satellites observation of light in the most intensely lit areas is limited, which distorts the relationship between light and income in countries with aggregately wealthy cities. Second, resizing these images to a similar size as other countries in the dataset would lead to greater distortion or require much more memory for training. Finally, these countries tend to have large GDPs, giving them great influence in training with a mean squared error loss function despite being less similar to lower-income countries where income predictions may be most useful.

3.2 National GDP and Survey Data

For training labels, I use GDP accounts recorded in the Penn World Tables and mean survey consumption from surveys of households collected by the World Bank. Although there is some discussion about the quality of measurments in the Penn World Tables (Pinkovskiy and Salah-i-Martin 2016), the CNNs I train achieve fairly high accuracy, suggesting a true common component between the lights and GDP measurements.

4 Method

4.1 Loss Function

Predicting GDP and survey data, each of which is fairly continuous and wide-ranging, is a regression task and therefore requires a loss function that takes into account error size, unlike in classification tasks. The main loss function I will use is the mean squared error (MSE) loss function, accounting for both target variables, shown below. This will be a useful for comparing a deep learning model to linear regressions, which minimize the same the loss function, as each observation's weight will be the same for both models. The cost function for the entire training set will be the average loss evaluated over the training samples.

$$L(\hat{y}_{GDP}, \hat{y}_{Survey}, y_{GDP}, y_{Survey}) = (\hat{y}_{GDP} - y_{GDP})^2 + (\hat{y}_{Survey} - y_{Survey})^2$$

The MSE loss often gives more weight to outliers than linear loss functions, as outliers often have large prediction errors that are given a quadratically proportional weight. It is important, therefore, to be aware of outliers in the dataset, and this is the main reason for dropping countries like the United States and China as described in Section 3.

For GDP and survey data, the distribution is skewed right, meaning most outliers are far above the mean. This presents a challenge for predicting income in the lowest income regions (the regions where predictions tend to be the most useful), but for now I accept the weak performance on low-income countries to show the overall superiority of a deep learning model relative to linear regressions for this task.

4.2 Model Architecture

Since the satellite images are grayscale images, each input's shape is 600 x 600 x 1. A dense neural network, therefore, would have 360,000 parameters in its first layer alone, but CNNs can effectively share parameters across inputs (pixels).

Using a 5 x 5 filter size in each convolutional layer greatly reduces the number of parameters as compared to a dense network (i.e. using a 1 x 1 filter), as does using a stride greater than 1 x 1. Note that as long as the stride is less than or equal to the filter size, the convolutional layer still takes in each pixel as an input. Each convolutional layer is followed by a batch normalization layer, which improves performance and keeps the data "centered." For nonlinearity, I use a rectified linear unit (ReLU) activation function in each convolutional layer. The number of filters used in each convolutional layer increases from 16 to 32 to 64 so that the model can first capture simpler elements of the image and then possibly more complex ones.

Importantly, I do not use any pooling layers, such as average or max pooling, even though these are common layers in a CNN. Because the sum of light in an image is a commonly used statistic in the literature, I wanted to make sure that the model could at least learn to use some transformation of the sum of light in an image. An average pooling layer may not hurt the chances of this too much, but a max pooling layer certainly would. Consider, for example, two countries with equal light sum, but one spread uniformly and one spread unevenly. A max pooling layer would output dissimilar values for each of these countries, even though that may be harmful to prediction.

After three CONV - BATCHNORM - RELU layers, I wanted the output size of the third layer to be relatively small before adding a dense layer. With a small input, the dense layer has a greater chance of exploiting spacial context that was captured by the convolutional layers, which could be key to outperforming a transformation of the sum of light. The filter and stride sizes were chosen to achieve this and after the three sets of layers, a 7 x 7 x 64 dimension output is passed into a dense layer with 32 units. Before the final output layer, a dropout layer is included for regularization (see Section 5 for discussion on the dropout rate). Finally, an output layer with one (when predicting only GDP) or two (when predicting GDP and survey data) is used without an activation so that the range of normalized outputs is not limited. Since the training labels were scaled down to have a mean of zero and standard deviation of one, the scalar outputs are then converted back to the original units by multiplying by the standard deviation and adding the mean, each of which was saved before training. In total, the model has 165,377 parameters.

5 Experiments and Results

To test the ability of deep learning models to predict GDP and survey data, I began by training the model described in Section 4 on each GDP and survey labels. Then, to simultaneously contribute to the discussion on the relative correlations between GDP and surveys with lights, I modified the model to have an output layer with two units and simultaneously predict GDP and survey data. The multi-task model achieved similar results to the two individual models, and since GDP and surveys are similar targets, predicting them together makes sense. Finally, to make use of more of the available data, I show that despite the multi-task model's strong performance on GDP prediction, there is still room for improvement by increasing the size of the training set, which is possible for GDP prediction. All models were trained using Tensorflow in Python with an Adam optimizer on a Google Colab Pro virtual machine with GPU-acceleration and 25GB of RAM. A random 30 percent of the training set was set aside for out of sample testing and is referred to in the rest of the paper as the "validation set."

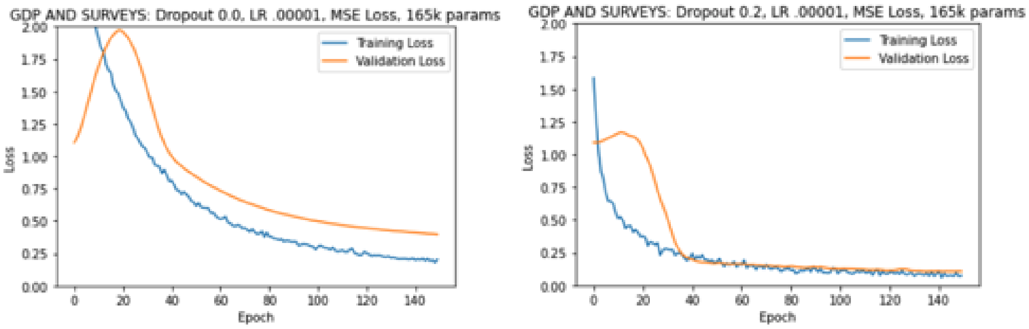
5.1 Hyperparameters

The main hyperparameters that required tuning were the learning rate and the dropout rate. After a random search of learning rates, it appeared that learning rates greater than 10^{-3} kept the model from learning, and learning rates below 10^{-5} drastically increased model training time without evidence of improved performance, even when allowed more time to train.

As expected, the larger learning rates (from right to left in Figure A2) seem to converge fastest, but after after about 150 epochs of training with a dropout rate of 0.25, each model's loss converges to roughly the same value for learning rates $\alpha \approx 10^{-4}$. As a result, I decided to go with 10^{-4} for each specification moving forward.

Figure A2 shows learning rate experiments with a non-negligible dropout rate in the final hidden layer because initial experiments showed a clear need for regularization. With a fairly small dataset, the model without dropout could overfit to the training set, but accuracy did not fully carry over to the validation set.

Figure 2: Dropout Rate Experiments



With a fixed learning rate, Figure 2 shows the training and validation set losses without dropout (left) and with a dropout rate of 0.2 (right). By randomly setting some weights to zero just before output, the dropout layer hurts the model's ability to overfit to the training data and instead helps it learn useful parameters. Increasing the dropout rate beyond 0.2 (or decreasing the keep probability below 0.8) did not improve performance. The vanishing gap between the orange and blue curves in Figure 2 shows that a dropout rate of 0.2 greatly reduces the variance problem present without dropout.

5.2 Evaluation

When using the MSE loss function, a common evaluation metric is the coefficient of determination, or R^2. Like any single metric, it is imperfect (e.g. it says nothing about consistent goodness of fit across the data), but it is useful as a simple way to compare models. Moreover, since I am primarily comparing my model to regressions, it is a useful metric since almost every regression analysis presents an R^2.

$$R^2 = 1 - \frac{\text{Sum of Square Residuals}}{\text{Total Sum of Squares}} = 1 - \frac{(\hat{y}_i - y_i)^2}{(y_i - \bar{y})^2}$$

Intuitively, the R^2 metric measures the ratio of a model's sum of squared residuals to the variance in the target variable. As a baseline, constantly predicting the target variable's mean will always achieve an R^2 of zero, and predicting each point exactly correctly will achieve an R^2 of 1.

Table A1 shows the results of regressing GDP and survey values on radiance, or the square root of the pixel sum cubed, as these transformations are common in the literature (Pinkovskiy and Salah-i-Martin 2016).

These regressions are fairly simple and give an idea of the R^2 for a simple linear model, around 0.3-0.5. With fixed effects for countries and year, as well as adjustments for country area or population, linear models can achieve R^2 values closer to 0.80 (Pinkovskiy and Salah-i-Martin 2016 and Henderson et al. 2012).

Table 1: Performance of Linear and Deep Learning Models (Error in $ Millions)

	Model	R^2	Avg Error	Median Error
GDP	Log Log Model	0.120	216,113	44,284
GDP	Linear Log Model	0.310	253,115	127,961
Survey	Log Log Model	0.110	88,971	20,244
Survey	Linear Log Model	0.296	103,808	52,009
GDP	CNN	0.890	121,279	81,062
Survey	CNN	0.897	50,420	32,851
GDP	CNN (larger train set)	0.937	49,689	29,579

My multi-task model, however, which simultaneously predicts GDP and survey values using only the lights data, achieves out-of-sample (on the validation set) R^2 values of .89 and .90, respectively, demonstrating the potential that deep neural networks have for predicting income. Moreover, training the model (with identical architecture and hyperparameters) on a training set of almost twice the size (which I have for GDP but not surveys), increases the out-of-sample R^2 to 0.94. Table 1 reports errors (in millions of dollars) and R^2 values for each model, in which the dependent variable, when applicable, has been transformed to the original scale.

Qualitatively, CNNs seem to have more potential for income prediction than regressions. Both types of models, however, perform very poorly on small, low-income countries. Each model listed in Table 1 has a negative R^2 when restricting predictions trained on the entire dataset to the bottom half of countries (by GDP or survey income). For the regressions, this is mostly due to overestimating, as predictions are drawn to the mean. For the neural networks, the residuals are more symmetric, but there are a decent number of negative predictions, which are nonsensical for the task at hand. In my experiments, I attempted adding a "leaky ReLU" activation function to the network's output layer. This helped accuracy for the lowest income countries but did not seem to help overall performance, which was sensitive to the scaling parameter.

6 Conclusion

Using lights at night from satellite images is an inexpensive and method of predicting economic activity with great potential in terms of accuracy. I have shown that a relatively simple CNN outperforms models in the literature, which primarily focuses on regressions. Moreover, the same CNN can predict GDP accounts and survey results with similar accuracy, suggesting that, under the assumptions that lights are only caused by true income, both measures correlate well with true income. A much deeper analysis would be needed to make any causal assumptions, but these results do not suggest that either GDP or surveys are obviously weak measurements.

For useful predictions in low-income countries, often the most challenging to predict accurately, there are several avenues for future work. Increasing the training set size seemed to have substantial returns for GDP prediction, so collecting more data could continue to improve accuracy. Further, closer attention could be paid to smaller countries in the creation of the loss function. Countries could be given weights that are inversely proportional to their existing income labels so that the model does not easily overlook these observations. Additionally, the model could input more than just a satellite image, but also time- or country-specific controls as many of the regression analyses in the literature have done. This would help a deep network quickly discriminate between small and large countries and make use of other existing data types. For now, however, many of the relationships between light and economic activity have been firmly established, and empirically-focused policymakers should look towards deep learning tools for deriving income and poverty estimates at a new tier of accuracy.

7 Contributions

I individually completed this project, including all of the research for and writing of the paper. All errors are my own. I would like to thank Dr. Maxim Pinkovskiy for introducing me to the nighttime lights data, invaluable discussions on how the data has been used in earlier work, and comments on my project. I would also like thank my TA, Fenglu Hong, for her incredible advice and meeting with me on several times, including outside of her allotted office hours.

References

Elvidge, Christopher D., Kimberly E. Baugh, Eric A. Kihn, Herbert W. Kroehl, Ethan R. Davis. 1997. "Mapping City Lights with Nighttime Data from the DMSP Operational Linescan System.

Ghosh, Tilottama., Sharolyn J. Anderson, Christopher D. Elvidge, Paul C. Sutton. 2013. "Using Nightitme Satellite Imagery as a Proxy Measure of Human Well-Being." Sustainability, 5, 4988-5019.

Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., Moore, R. 2017. Google Earth Engine: Planetary-scale geospatial analysis for everyone. Remote Sensing of Environment.

Henderson, J. Vernon, Adam Storeygard, and David N. Weil. 2012. "Measuring Economic Growth from Outer Space." American Economic Review, 102 (2): 994-1028.

Ivan, Kinga et al. "VIIRS Nighttime Light Data for Income Estimation at Local Level." Remote Sensing 12.18 (2020): 2950.

NOAA's NGDC (National Geophysical Data Center). (DMSP data collected by US Air Force Weather Agency.

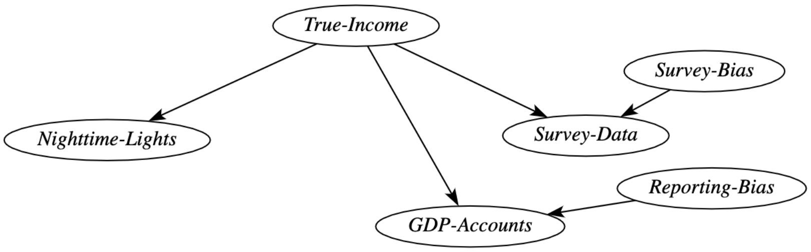
Oshri, Barak, Annie Hu, Peter Adelson, Xiao Chen, Pascaline Dupas, Jeremy Weinstein, Marshall Burke, David Lobell, Stefano Ermon. 2018. "Infrastructure Quality Assessment in Africa using Satellite Imagery and Deep Learning." KDD '18: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, July 2018: 616-225.

Pinkovskiy, Maxim L., Sala-i-Martin, Xavier. 2016. "Lights, Camera ... Income! Illuminating the National Accounts-Household Surveys Debate." The Quarterly Journal of Economics, 131, no. 2: 579–631.

Pinkovskiy, Maxim L. 2017. "Growth discontinuities at borders." Journal of Economic Growth, 22, 145-192.

Appendix

Figure A1: Causal Graph for True Income and Its Proxies



If Nighttime-Lights is only caused by True-Income and noise, and if Nighttime-Lights is highly predictive of GDP-Accounts or Survey-Data, then that measure should also be highly predictive of True-Income.

Figure A2: Learning Rate Experiments

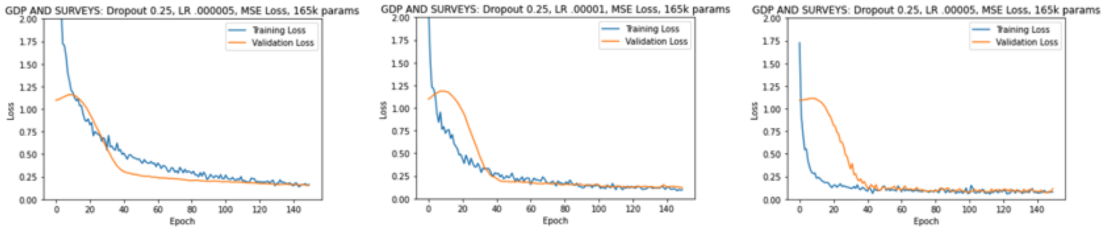


Table A2: Predicting GDP and Survey Measures with Lights (Linear Models)

	GDP	Log GDP	Log GDP	Survey	Log Survey	Log Survey
	(1)	(2)	(3)	(4)	(5)	(6)
Radiance	0.00***		0.00***	0.00***		0.00***
	(0.00)		(0.00)	(0.00)		(0.00)
Log Radiance		0.50***	0.40***		0.45***	0.34***
		(0.02)	(0.02)		(0.02)	(0.02)
Observations	819	819	819	819	819	819
R^2	0.30	0.49	0.52	0.28	0.45	0.48
Note:	*p<0.1; **p<0.05; ***p<0.01 Radiance is defined as the square root of pixel values cubed.					