Learning statistical information from images: a proof of concept

Organisation: Statistics Netherlands

Author(s): Joep Burger (in collaboration with Tim de Jong, Lyana Curier, Marc Ponsen, Dewi Peerlings, Han van Leeuwen, Krzysztof Cybulski and Jan van den Brakel)

Date: May 4, 2020

Version: 2

1. Background and why and how this study was initiated

(as many sentences as necessary, as few as possible)

In 2015, the United Nations launched a 15-year plan to meet 17 sustainable development goals, among others to end poverty, promote well-being and reduce inequality. Indicators are needed to monitor the progress. MAKSWELL is a project funded by the European Union to harmonize indicators on sustainable development and well-being. Work Package 3 of MAKSWELL focuses on the measurement of regional poverty. Register-rich countries like the Netherlands can compile regional income-based poverty statistics from a complete enumeration of tax income data. Many countries do not have this luxury. Luckily, the earth is observed by satellites since the 1970's, unaffected by national borders and legislation. Sometimes aerial images are also available. Moreover, computer vision has evolved spectacularly during the past decades due to the development of specialized hardware. Therefore, the research question we are addressing is: can we learn poverty from aerial or satellite images? This question serves two objectives: 1) learn how to use machine learning to exploit imagery as a new data source in the production of official statistics and 2) assist other countries who do not have income data in measuring poverty from imagery. In this proof of concept we replace poverty with open data such as the number of inhabitants, households and dwellings. Labelling images with sensitive data that require strict protection, such as income-related poverty, presents new challenges that will be dealt with later.

2. Data

2.1 Input Data **(short description)**

Two image data sources were used: aerial images and Landsat satellite images. Aerial images have a resolution of 0.25 m per pixel, three color channels (RGB), 8-bit color depth (256 bits per channel) and are available since 2016, whereas Landsat images have a lower resolution (30 m

per pixel) but more color channels (11), a more precise color depth (16-bit, i.e. 65536 bits per channel) and a longer history (Landsat 8 was launched in 2013). In addition, open grid statistics published by Statistics Netherlands were used to label the images (https://www.cbs.nl/nl-nl/dossier/nederland-regionaal/geografische-data/kaart-van-100-meter-bij-100-meter-met-statistieken).

2.2 Data Preparation

(e.g.: Data Cleaning, Normalisation… – or: none)

Aerial images were cut into 1-ha squares (100 m × 100 m). Given the resolution, each image therefore has a dimension of 400 pixels × 400 pixels × 3 bands (red, green, blue). The Netherlands covers about 40 thousand km^2. A 1-ha grid will therefore contain about 40000 / (0.1 × 0.1) = 4 million squares. To limit storage and processing time for the proof of concept, a simple random sample of 300.000 squares was drawn from the 1-ha grid. Grid statistics were linked, i.e. each 1-ha square was assigned a number of people, households and dwellings. A yet unknown but presumably selective subset of squares was lost because the statistical information was unknown, unreliable or undisclosed. The corresponding aerial images were retrieved, i.e. for the exact same bounding box. So far only 70 thousand images were used. Data augmentation was applied, including 45° random rotation, 10% width shift, 10% height shift, 20% random zoom, 0.2° shear and horizontal and vertical flip. Sensitivity of the results to these hyperparameters has not been studied yet.

Satellite pixels that fully or partially cover each square were collected, including the fraction of the square covered by each pixel. Pixels with a high cloud score were excluded. Given the 30-m resolution, each 1-ha square contains (sections of) about 16 pixels, although in practice this may be higher because the pixels are rotated relative to the orientation of the grid, or lower because of clouds.

2.3 Feature Selection

(yes/no, if yes: how, why)

The aerial images were used in a Convolutional Neural Network. The advantage of this technique is that features are automatically learned, so no manual feature extraction is needed.

The Landsat images were analyzed with more traditional machine learning techniques, requiring manual engineering of features from the set of pixel values in each square. For the proof of concept, the weighted mean, standard deviation and range were calculated of each channel

and of a number of additional remote sensing indices derived from the channels, such as the normalized difference vegetation index ($NDVI = \frac{nir-red}{nir+red}$). The proportion of the square covered by the pixel was taken as pixel weight.

2.4 Output data

(short description)

Squares will be labeled with income-related poverty indices but for the proof of concept we used open data to avoid privacy issues. The number of inhabitants, households and dwellings were categorized into five classes: four quartiles and a rest category for unknown, unreliable or undisclosed. We took quantiles to prevent class imbalance. (Class imbalance will drive the algorithm towards "predicting" the most common class.) The rest category was ignored for the proof of concept. Thus, we attempted to predict in which quartile an out-of-sample square falls into.

In a closed environment with income data, the proportion of households under the poverty threshold will be categorized into classes, each square will be labeled with one of these classes and the algorithm will be trained again to predict in which class an out-of-sample square falls into. These predictions could be mapped to identify poor neighborhoods, in other countries or before register data is updated, they could be followed over time and the learned features could be used as auxiliary information to improve model-based estimators.

3. Machine Learning Solution

3.1 Models tried

(e.g.: Multi-Layer-Perceptron, Random Forest, SVM, ...)

For the aerial images, we trained Convolutional Neural Networks (CNNs). Two existing architectures were used: VGG16 and ResNet50 with weights transferred from the model pre-trained on ImageNet, a dataset with over 1.2 million images and 1000 object categories. Object categories include animals, plants, activities, materials, instrumentation, scenes and food, sometimes with a surprising amount of detail (Persian cat, Model T, Granny Smith; http://image-net.org/challenges/LSVRC/2015/browse-synsets). The final, fully-connected layer was adjusted to learn four instead of 1000 weights corresponding to the number of output classes. These weights were learned from the aerial images. Transfer learning speeds up training by exploiting a larger dataset that can help identifying generic patterns such as lines. On the other hand, images and object categories in ImageNet differ substantially from our study.

Performance is expected to improve by fine-tuning the initial weights in earlier or possibly all layers with the aerial images.

For the satellite images, we trained a random forest (RF) and a support vector machine (SVM).

3.2 Model(s) finally selected and the criterion

(i.e.: which model was why seen being the best?)

It is too early to make this call.

3.3 Hardware used

(e.g.: Intel Core i5-6300U, 2.4GHz)

To be completed.

3.4 Runtime to train the model

(e.g.: 2 hours for 500,000 training samples and 25 features)

To be completed.

4. Results

(e. g. in terms of RMSE, MAE, distributional accuracy [*], F1 (micro or macro), recall, accuracy, (threshold,) ..., perhaps as a table for different situations (if available))

[*]: If used: How did you measure distributional accuracy? By proportions, moments, quantiles, correlations, …?

Since the classes were balanced we used accuracy as an overall quality measure. Random guessing is expected to yield an accuracy of 0.25. Using VGG16 an accuracy of 0.68 was achieved on the number of households quartile and 0.65 on the number of inhabitants quartile. Using ResNet an accuracy of 0.69 was achieved on the number of dwellings quartile. A clean comparison between methods on the same set of images and the same outcome variable has not been done yet. Increasing the training set from 20 thousand to 50 thousand images (and the test set from 10 thousand to 20 thousand) increased the accuracy to 0.74. Simplifying the classification task to two classes increased the accuracy to 0.87 (using the training set of 20 thousand images). The effect of sample size and sample composition (through sampling design) will be studied more thoroughly in a follow-up study.

The more traditional machine learning methods RF and SVM on hand-crafted features from Landsat satellite pixels (see Section 2.3) reached an accuracy of around 0.3 to 0.5. It is better than random

guessing so something is being learned, but the scores are lower than with the CNNs on aerial images. It is currently not possible to say if this due to the data source (satellite versus aerial image), the resolution (30 m versus 0.25 m) or the method (RF and SVM on manually extracted features versus CNN).

5. Code/programming language

(e.g. the Python code is stored in GitHub)

Most of the code was written in Python using the keras library and stored at https://gitlab.com/CBDS/deepstat.

6. Evolution of this study inside the organisation

(e. g.: Collaboration within the organisation? Has this study advanced ML within the organisation?)

Statistics Netherlands has a methodology department with a long history and strong expertise in survey methodology and statistics. Our Center for Big Data Statistics was launched in 2016 and has attracted data scientists with strong expertise in machine learning and computer science. This project has greatly stimulated the collaboration between the two groups. The study has also stressed the importance of specialized hardware and IT skills needed to be able to apply deep learning.

7. Is it a proof of concept or is it already used in production?

(If it is a proof of concept: Was it successful? How will its results prospectively be used in the future?)

This study is clearly a proof of concept. To us it was successful because it has improved the collaboration between methodologists and data scientists and it has shown that some statistical information can be learned from images. The most important next step will be to move to a closed environment where we can link income-based poverty indices.

7.1 What is now doable which was not doable before?

(**e. g.: Is something faster or cheaper or more exact? What is the added value using this machine learning?**)

Convolutional Neural Networks allow the exploitation of images that have never been used in the production of official statistics. Features may be learned that can help improve existing model-based estimation methods.

7.2 Is there already a roadmap/service journey available how to implement this?

(as many sentences as necessary, as few as possible)

No, there are too many issues to be resolved and studied before anything can be implemented.

7.3 Who are the stakeholders?

Countries without income registers but with access to aerial or satellite images, and departments within Statistical Netherlands responsible for measuring SDGs or producing regional statistics.

7.4 Fall Back

Is a fall back plan in place or planned to mitigate the risk of the ML solution failing in production? Will there be resource left in place to go back to e.g. manual imputation or the use of rule-based scripts? (as many sentences as necessary, as few as possible)

The two major conditions that should be met are access to the imagery data and the hardware and IT skills to train CNNs. When taken into production, access to data can be guaranteed by a supply agreement. Hardware can be rented or bought. However, privacy-sensitive data require special attention and IT support (Unix) will require a different mindset in a Windows-oriented organization. The fall back scenario is the current situation where the imagery data are left unexploited.

7.5 Robustness

What fail checks are in place or planned to ensure that the ML solution is consistently meeting or exceeding the set gold standard? (as many sentences as necessary, as few as possible)

By splitting the data into training and test data, the models can be tested for their performance on out-of-sample images within the same period. As more recent images and tax data become available, models can be applied to the newer data and tested for their performance on out-of-sample images from a different period. More recent images and tax data can also be used to expand the training set, making the model more robust, or to update the training set, making the model more up to date.

8. Conclusions and lessons learned

(e.g.: ML can be used for editing but one has to have the following points in mind ...)

Our proof of concept has shown that statistical information can be learned from images. The main bottleneck in training a convolutional neural network is the availability of specialized hardware (Unix, GPUs) in a secure environment. The input is big (up to TBs), the models are complex and the output is privacy-sensitive.

9. Potential organisation risk if ML solution not implemented

(as many sentences as necessary, as few as possible)

There is no risk to Statistics Netherlands, but we will miss out an opportunity to make use of a rich and open data source that is out there waiting to be explored. Other countries may not be able to estimate income-based poverty because they do not have the labels needed to train a CNN.

10. Has there been collaboration with other NSIs, universities, etc?

(yes/no, if yes: which ones?)

We had fruitful discussions with, and received useful suggestions and Landsat data from Abel Coronado and Jimena Juarez from INEGI.

11. Next Steps

(as many sentences as necessary, as few as possible)

Several next steps are considered:

- Move the data and model to a closed environment where we can link income-related poverty labels.
- Use an ordinal loss function. Default classification considers labels as nominal variable, where each wrong category is equally wrong. Income-related poverty quantiles and many other labels are, however, ordinal variables. The loss function should take into account that for a rank-1 label, predicting rank 2 (almost right) is less wrong than predicting rank 5.
- Quantify the effect of sample size on prediction quality. How much performance does the model gain from more images?
- Quantify the effect of grid size on prediction quality. A 25-ha grid (500 m × 500 m) will show more of the environment than a 1-ha grid, but will reduce the number of potential images (to about 160 thousand). At a resolution of 0.25 m per pixel, the resulting 2000 × 2000 × 3 matrix per image will be too big. Images will have to be resized using some interpolation algorithm, compromising detail, or the first convolutional layer could use larger filter sizes or larger stride to reduce spatial dimensions. A comparison with Landsat satellite images (about 17 × 17 × 11 per image) could become feasible.
- Quantify the effect of sampling design on prediction quality. Given a sample size, how much can the performance of a model trained on a simple random sample of images be improved by using a more sophisticated sampling design? The latter requires some clustering algorithm or auxiliary information correlated with the output labels. The potential of transfer learning could be tested by sampling selectively, e.g. train on images from the south and test on images from the north.
- Expand the number of channels by adding indices derived from the color bands, such as the normalized difference vegetation index.
- Visualize the features learned in each network layer. Can we transform them into auxiliary variables to improve time series models and small area estimation models?
- Validate the model on images outside the country.