MDPI

Article

A Two-Stage Approach to Improve Poverty Mapping Spatial Resolution

Joaquín Salas [1], Marivel Zea-Ortiz [1], Pablo Vera [1], and Danielle Wood[2]

[1] Instituto Politécnico Nacional; jsalasr@ipn.mx, mzeao2000@alumno.ipn.mx, pvera@ipn.mx
[2] Massachusetts Institute of Technology ; drwood@media.mit.edu
* Correspondence: Joaquín Salas, jsalasr@ipn.mx; Cerro Blanco 141, Colinas del Cimatario, Querétaro, 76090, México

Abstract

Global extreme poverty has fallen dramatically over the past two centuries, yet hundreds of millions remain impoverished, underscoring the need for scalable monitoring tools. In Mexico, poverty metrics are available only sporadically in terms of time and space (*e.g.*, every 5 years at the municipal level), making it difficult for decision-makers to access reliable, up-to-date, and sufficiently detailed information, highlighting the need for higher-resolution, timely methods. To address this problem, we propose a two-stage approach that combines socioeconomic and Earth Observations-based data. Initially, a machine learning model maps census variables to official poverty indicators belonging to a multidimensional model, yielding fine-scale poverty estimates. A census-based model trained with XGBoost achieved $R^2 \approx 0.842$, indicating strong agreement with official poverty figures and providing high-resolution proxies. Afterward, we use features based on remote observations to predict these poverty estimates at a 469 m grid scale. In this case, advanced foundation models outperformed other ML-based approaches, achieving an R^2 of 0.683. While foundation models enable more accurate, fine-scale poverty mapping and could accelerate poverty assessments, their use comes at a heavy price in terms of carbon emissions.

Keywords: Poverty assessment, Foundation Models, Remote Observations

1. Introduction

From 1800 to 2024, extreme poverty, defined as earning less than $2.15 per day in terms of 2017 currency, fell from affecting approximately 80% of the population in 1800 to just 8.5% [1]. While appreciable worldwide, this decline has been particularly notable in China, where extreme poverty has disappeared [2]. Despite these advances, poverty persists, sometimes localized [3] and at other times in the aftermath of catastrophic events [4], limiting individuals' ability to realize their potential fully. To advance the human aspiration to eradicate poverty, one of the goals outlined in the United Nations Sustainable Development Goals [5], we need fast, reliable, and cost-effective tools that provide timely information to inform decisions. Recent advancements in computing power [6], automatic inference algorithms [7], and remote sensing observations [8] are opening vast possibilities.

Despite recent advances [9], Mexico continues to face challenges in reducing poverty. To address this, a multidimensional model was developed [10]. The model is managed by an independent state agency responsible for measuring the effects of social policies and the state of poverty in Mexico. In its methodology, the agency conducts a survey measurement of poverty, divided into 16 indicators categorized into areas such as poverty, social lag,

Received:
Revised:
Accepted:
Published:

Citation: Salas, J. *et al.* . A Two-Stage Approach to Improve Poverty Mapping Spatial Resolution. *Remote Sens.* **2025**, *1*, 0. https://doi.org/

social deficiencies, and economic well-being [11]. The evaluation results for these indicators are published every five years at the municipal level and every two years at the state level. The agency derives these indicators mainly from the National Survey of Household Income and Expenditures (ENIGH) [12]. In the 2022 edition, the survey included 105,525 selected dwellings (totaling 106,893 households), with an average of approximately 3,300 surveys per federal entity [13]. The surveys were conducted by 2,712 field interviewers between August 11 and November 28, 2022, with support from 412 administrative staff members. In 84.3% of cases, a complete interview was obtained, of which 61.6% were conducted with a direct informant and 22.7% with an indirect informant. The highest rate of successful interviews was recorded in Michoacán, at 88.5%, while the lowest was observed in Colima, at 76.8%. The analyses conducted to assess poverty based on the collected data represented a significant effort by many individuals over more than three and a half months, achieving a substantial portion of responses within the sample space. The results of the 2022 survey were published approximately nine months after the survey was completed. In 2022, 9.1 million people lived in extreme poverty, while 11.7 million experienced housing quality deficiencies. As the traditional form to carry on the assessment of poverty relies on surveys, requires much human labor, and takes much time to be processed, this research aims to provide a tool to increase the temporal and spatial resolution of poverty evaluations, at a reduced cost and in a faster manner, by applying current Earth observations (EO) and modern machine learning (ML) techniques.

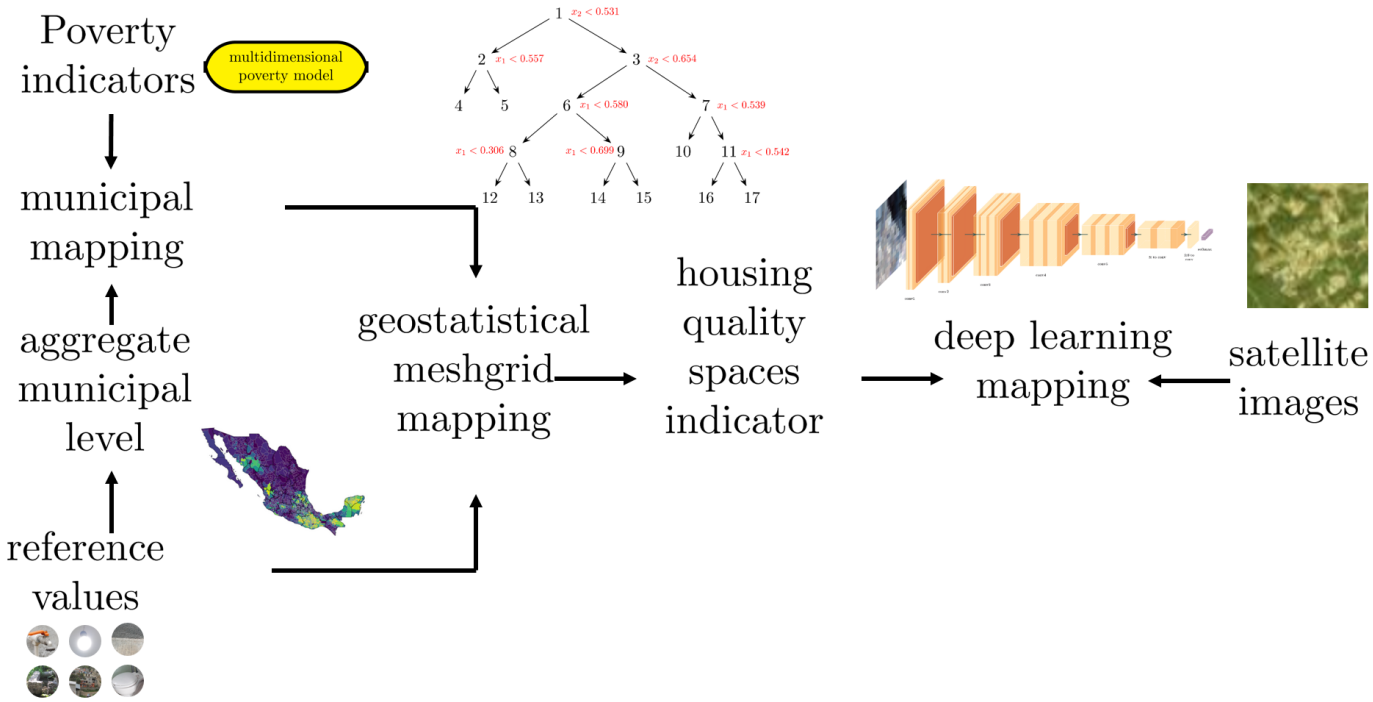


Figure 1. Mapping poverty indicators using remote observations in Mexico. Census reference values are aggregated at the municipal level to map them with indicators from the multidimensional poverty model. This mapping is then expressed at the 469 m cellular level within the geostatistical grid. Next, multispectral Earth observations (EO) are used to map these estimates. As a result, the housing quality indicator can be obtained at a higher spatiotemporal resolution.

In this research, we develop a model to assess poverty following a two-stage approach. First, we approximate the value of poverty indicators using census estimates (see Figure 1). We do this by aggregating census data at the municipal level, which is the highest resolution at which poverty indicators are published. Next, we use this mapping to derive cell-level indicators within a geostatistical grid with a 469 m resolution. Using this geospatial region

as a base, we train ML architectures on remote sensing data to infer poverty-related housing quality. The main contributions of this work include the following:

- *Poverty assessment:* We predict an official multidimensional poverty indicator (housing quality) defined by Mexico's national methodology and project it onto a 469 m equal-area geostatistical grid, which differs from prior EO-poverty studies that target wealth indices, asset scores, or survey-derived proxies on irregular administrative units or coarser grids.
- *Poverty characterization framework:* We explicitly separate (i) census to official poverty indicator (municipal level) and (ii) EOs to grid-cell proxy (469 m). The design provides a reproducible way to generate dense training labels while maintaining consistency with the national poverty framework, addressing the sparse nature of official measurements.
- *ML baseline:* We develop a baseline including a battery of ML techniques that span classic, CNN, Transformers, Graph-based, Capsules, and Foundation models, which should serve as a basis of comparison and selection of the best performing techniques.
- *Computer code and data*: We share the documented code and EO-based data used to perform the analysis described in the study, allowing other researchers to verify the results and use them as a stepping stone for further understanding of the phenomenon.

In the remainder of the paper, we survey the literature related to this study in §2, focusing primarily on the mappings between observations and poverty indicators. Then, in §3, we detailed the tabular data and EO-based features employed in the study. Additionally, we provide further details on the mapping of census data and EO to poverty indicators. Following suit, we describe the battery of ML algorithms employed to explore the dataset in §4. A description of the results for the mappings and feature selection follows in §5. These results are placed in the context of the current state of the art in §6. The manuscript concludes with final remarks and future research directions. Some appendices are added to further detail the dataset's structure and spectral indices models employed.

2. Related work

This section provides an overview of papers addressing the assessment of poverty using classical models and deep learning, the extraction of features that can best be used to map observations to poverty characterizations, and recent advances in employing foundation models to address these mappings.

2.1. Assessing socioeconomic conditions with ML

Several studies focused on poverty assessment are based on the analysis of census and survey data using ML [14–22]. Using the household expenditure and income surveys conducted in Jordan since the beginning of the millennium, Alsharkawi *et al.* [14] demonstrated the effectiveness of ML-based classifiers in tracking and targeting poverty across that country. In a study using an intercensal survey in Mexico, Corral *et al.* [16] assessed the performance of ML methods for estimating income poverty. Similarly, using survey data from the Tanzania National Bureau of Statistics, Sende *et al.* [20] implemented a variety of ML classification models to infer multidimensional poverty status. They achieved the best scores using an Ensemble Learning Model (EM) that combined the outputs of ML algorithms, including Support Vector Machine (SVM) with Radial Basis Function (RBF) and linear kernels, Naive Bayes, K-Nearest Neighbors, Gradient Boosting, Logistic regression, and CatBoost. Kuffer *et al.* [22] note that most indicators related to quality housing deprivation require a statistical database, such as census data, for their evaluation. However, these databases might not include inhabitants of informal settlements. To circumvent this problem, they monitor existing slums and informal settlements in Europe using EO data.

Wang *et al.* [23] estimate district- and county-level poverty in China by combining nighttime lights, land cover, and DEM data with socioeconomic statistics, comparing a partial least squares (PLS) model, which performs principal component analysis (PCA) to build a linear regressor ($R^2 = 0.65$), against several ML methods (*e.g.*, decision trees, SVM, Gaussian processes, random forests), showing that PLS capture poverty patterns. Wang *et al.* [24] integrate accessibility, block vitality, unit rent, green coverage, and service access into a multiplicative spatial index clustered via Self-Organizing Maps (SOM) to delineate urban poverty spaces, and contrast this with a PCA-based approach using remote sensing texture features for validation.

2.2. Deep Learning Models

Deep learning (DL) models, such as CNNs and Transformers, allow the identification of poor regions using remote sensing data [17,19,25–33]. Daoud *et al.* [17] used both census data and surveys to train DL models to estimate living conditions in India. Their methods, based on EO data and ML, measure human development at the village level. Using daytime and nighttime satellite imagery, as well as survey data from five African countries, Jean *et al.* [25] trained a CNN and a ridge regression model to estimate poverty. Similarly, Pandey *et al.* [27] estimated poverty using an Indian census database to train a CNN model that learns visual features from satellite images associated with different levels of economic development, including household roof material, source of lighting, and source of drinking water. Yeh *et al.* [30] used DL to infer a wealth index from survey data on household asset wealth in 23 African countries. They used daytime and nighttime satellite imagery to train two separate CNN models, then combined their outputs into a fully connected layer. Perez *et al.* [32] used a semi-supervised approach to estimate poverty. First, they collected multi-spectral satellite images covering the African continent and labeled 5% of them using sparse data from the Demographic and Health Surveys (DHS) program. Then, they trained a Wasserstein Generative Adversarial Network (WGAN) to extract features from both labeled and unlabeled images, which they utilized in a multitask framework to estimate poverty metrics. Putri *et al.* [33] compared the performance of ML and DL algorithms in estimating poverty on a grid of 1.5 km resolution using satellite imagery. Hu *et al.* [19] explored the potential of using open geospatial data to identify village-level poverty in Hubei, China. Their study integrated high-resolution satellite imagery (HRI), point-of-interest (POI) data, OpenStreetMap (OSM) data, and digital surface model (DSM) data. They found that the poorest areas are associated with specific natural geographical conditions and inadequate public services.

2.3. Feature selection in poverty characterization

Some studies have analyzed the relative importance of variables that characterize poverty [34–37]. Li *et al.* [34] utilized a collection of Google Earth Engine (GEE) images, including satellite, aerial, and Street View images, to investigate whether features extracted from these images significantly differentiate urban poverty in China. They used 25 predictors derived from geometric, shape, and texture features and trained four ML regression models using Random Forest (RF), Gaussian Process Regression (GPR), Support Vector Regression (SVR), and Neural Network (NN). They found differences in feature importance among the ML models. Olearo *et al.* [35] determined the probability that older adults in Lombardy, Italy, were poor and then trained an XGBoost classification model to predict three levels of risk. Among their objectives was to identify the most important variables related to health, housing, maintenance capacity, social deprivation, and consumption deprivation to estimate poverty among older people. Garza-Rodriguez *et al.* [36] studied the relative importance of the variables determining poverty by examining the case of

households in Mexico. They found a strong correlation between households in extreme poverty and their location in the southern region of the country, as well as with households where the head of the family spoke an indigenous language or was an older person. Mohamud and Gerek [37] assessed the socioeconomic status of households using data from the Inter-American Development Bank of Costa Rica. They identified the characteristics best describing a household as non-vulnerable, vulnerable, moderately poor, or extremely poor.

2.4. Foundation Models

Foundation models are ML architectures that typically contain a large number of parameters and are trained, mostly in a self-supervised manner, on a large data corpus to address tasks in an agnostic manner [38]. To employ them, one needs to fine-tune them using the available data. Usually, they excel in few-shot and zero-shot learning. Until recently [39], they performed better than task-specific fully supervised models for geospatial tasks involving only text, but did not perform well on other tasks, such as image classification. Nowadays, they can reason through various geospatial data by aligning them. Xiao *et al.* [40] provided a systematic review of foundation models for EO. Lately, there has been promising progress in foundation models. For instance, DINOv3 [41] offers a vision backbone that yields dense embeddings of small square image regions. It achieves state-of-the-art performance on diverse EO benchmarks, including classification, segmentation, canopy height estimation, and depth estimation, surpassing domain-specific models even when restricted to RGB inputs. AlphaEarth Foundations [42] introduced a universal embedding field model that integrates diverse geospatial data sources (optical, SAR, LiDAR, climate, DEM, and text) into time-continuous embeddings, consistently outperforming both domain-specific and general featurization methods on classification and regression benchmarks, and providing annual global embedding layers (2017–2024) for efficient large-scale mapping.

There have been some efforts to employ foundation models to estimate poverty, for instance, Lu *et al.* [43]. They utilized high-resolution satellite images to define candidates. Then, streetview photos were examined to validate them. This process generated 750 image patches for labeling, separating low-income areas (LIA) from non-LIA regions. Then, Sentinel-2 images were employed to feed foundation models, including U-Net, ViT, and SegFormer. More recently, Agarwal *et al.* [44] introduce the Population Dynamics Foundation Model (PDFM), a graph neural network trained on multimodal geospatial data (maps, busyness, aggregated search trends, weather, air quality, and remote sensing) to generate embeddings that achieve state-of-the-art performance on 27 health, socioeconomic, and environmental tasks, and that, when combined with the TimesFM forecasting model [45], improve unemployment and poverty predictions beyond fully supervised baselines.

3. Materials and Methods

The primary problem is to establish a mapping between remote sensing-based perception and poverty assessment. The methodology employs a two-step approximation in which several predictors, including census data, are related to poverty indicators at the municipal level. Using the resulting mapping function, we assess the poverty indicator at a 469 m grid cell level corresponding to the Mexican geostatistical meshgrid. Remote sensing-based features and indices are mapped to this assessment. This section describes the datasets used and further details how the mappings are established.

3.1. Input data

We proceed to describe the various data sources employed, including census data represented in the geostatistical meshgrid [46], multidimensional poverty model indicators [47], and remote sensing-based features [48], along with details about how they fit into the overall framework.

3.1.1. Geostatistical Meshgrid

The geostatistical meshgrid census information is released through a rHEALPix [49] mesh, a Discrete Global Grid System (DGGS) made of equal-area square cells that starts with the six faces of a cube projected onto the terrestrial ellipsoid (level 0) and for which a 1→9 hierarchical refinement is applied: every *parent* cell is split into a 3×3 grid of *children*, so spatial resolution increases exponentially while preserving the basic shape. Recursively, level L contains $N_L = 6 \cdot 9^L$ cells, and, given the Earth's surface area A_T, the mean area per cell is $A_L = A_T / N_L$; for Earth's $\approx 5.1022 \times 10^{14}\text{m}^2$, this ranges from $\approx 8.5037 \times 10^{13}\text{m}^2$ at L = 0 down to $\approx 3.7172\text{m}^2$ at $L = 14$, with the side length shrinking by a factor of 3 each level (*e.g.*, 9 221.6 km → 3 073.4 km → ⋯ → 1.9280 m). Thus, moving between levels simply divides the cell area by 9 and the side length by 3, providing a nested, temporally stable multiresolution coverage suitable for integrating and comparing statistical and geospatial data. This study employs level 9, or square cells with a side of ≈ 468.5042 m, which we will now call the 469 m grid.

The Mexican National Census of Population and Housing (CPH) has 277 variables, whereas the cells of the geostatistical grid have 40 predictors. It would be desirable to use the 40 grid cell predictors to construct the estimators. Nonetheless, the finest resolution at which multidimensional poverty model information is provided is the municipal level. Aggregating grid cells to match the resolution of the multidimensional model poses several challenges. One is that the grid cells may be defined across municipal boundaries, making the aggregates inaccurate, if naively accumulated (see § 4.3). Perhaps more importantly, due to privacy concerns, between 23% and 40% of the predictors have missing values, making it unreliable to obtain precise representations of their values at the municipal level. Thus, we identify the predictors that are common to both the CPH at the municipal level and the geostatistical 469 m grid, leaving 29 predictors available (see Table A1). Some variables relate to population, while others relate to housing. We normalize the former by the total population and the latter by the total number of inhabited private dwellings. The average occupancy $\bar{o}$ predictor is bounded and normalized by $o_n = \min(o, M)/M$. Experimentally, we have observed that fewer than 0.03% of observations in the grid cell have an average occupancy greater than $M = 7$. We normalize the poverty indicators by dividing them by the total population, resulting in values ranging from 0 to 1.

The CPH includes information at the spatial resolution of housing with a temporal frequency of ten years. Correspondingly, the 16 indicators of the multidimensional poverty model [11] are generated every two years at the state level and every five years at the municipal level. In this document, we study the coincident event in 2020.

3.1.2. Remote Sensing-based Features

Sentinel-2 is a twin satellite system with a polar, sun-synchronous orbit at a nominal altitude of 768 km and 180 degrees out of phase with each other. Its revisit time with two satellites is five days at the equator and two to three days at mid-latitudes [48]. Solorzano *et al.* [50] determined that the repetition time for cloud-free observation of a pixel for the seven ecoregions of Mexico varied between 0 and 6.58 days between 2015 and 2019. Its latitude coverage ranges from 56° S to 84° N. The instruments on the Sentinel-2 satellites sample 13 spectral bands at three spatial resolutions: four at 10 m, six at 20 m, and three at

60 m [51]. In this study, B10 (1373.3 nm and 1376.9 nm) is not used because it is not present in Google Earth Engine collections. At its current orbit, the satellite coverage is 290 km. The instruments capture images over land, coasts, islands larger than 100 km^2, islands of the European Union, other islands within 20 km of the coastline, the Mediterranean Sea, all inland water bodies, and all enclosed seas.

For this research, we used the blue, green, red, near-infrared (NIR), shortwave infrared (SWIR) 1, and SWIR 2 bands to compute and incorporate into the predictors a series of spectral indices related to vegetation, built-up areas, and the presence of water (see Appendix B for further detail) [52]. Concretely, ten indices are derived and grouped into five thematic blocks: *Vegetation condition*, *Soil exposure*, *Built-up and impervious surfaces*, *Water and moisture availability*, and *Disturbance and burn severity*. This thematic organization allows us to capture contrasts such as sparse or stressed vegetation around informal settlements, patches of bare soil in marginal agricultural plots, high impervious-surface fractions in low-quality housing clusters, water-scarcity signals, and fire scars from slash-and-burn practices.

For foundation models, we explore AlphaEarth [42] and DINOv3 [41]. In the case of the AlphaEarth feature stack, for each grid cell, we extract a vector with c=64 channels, an internally curated multispectral/multifeature stack derived from satellite remote sensing. Each tile has an arbitrary spatial size $(H \times W)$ and per-band float intensities. The stack is used as-is (no resampling to a fixed shape), preserving the native per-pixel information across all 64 layers. For DINOv3 inputs, we construct a 3-channel image for feature extraction by selecting RGB-like bands to obtain three orthogonal components for each Sentinel-2A Level-2A tile. Before, we apply robust percentile clipping (*e.g.*, 0.5–99.5th percentile) per band to suppress outliers. The resulting 3-channel image is then resized to a square canvas (*e.g.*, 224–256 px) and standardized using satellite image mean and standard deviation.

3.2. Mapping to poverty indicators

Two important mappings need to be established. One of them concerns census and poverty indicators at the municipal level. And another one, from remote observation features to poverty indicators at the 469 m geostatistical grid level. In this section, we delve into the details of how this is performed.

3.2.1. Poverty characterization

Mexico employs a multidimensional model to estimate the state of poverty [47]. That model is currently run by the Mexican agency whose objective is to produce statistics and geographic information, among others, the national census summaries [46]. We characterize poverty by mapping reference variables provided by the national census to poverty indicators defined by the multidimensional poverty model. To perform this mapping, we aggregate the reference values at the 469 m grid cell level to the municipal level (see 4.3). From the perspective of ML, Bishop [53] suggests that a sensible selection of the most appropriate regressor should involve an ensemble of them; preferably selected among those providing an enriched criterion for building the estimations. Bishop distinguishes broad families including those regressors based on decision trees [54], kernels [55], and neural networks [56]. Examples of algorithms based on decision trees include random forests and extreme gradient boosting. Examples of kernel-based methods include support vector machines and Gaussian processes.

In Mexico, publicly available national census information includes more than 200 variables. However, some may be unimportant for establishing the mapping between census reference values and the indicators of the poverty multidimensional model, which

requires a characteristic selection analysis. This selection reduces the number of processing schemes, improves performance by eliminating irrelevant predictors, and helps explain the regressor's deductions. Since the feature selection problem has a non-polynomial computing complexity, heuristics such as SHAP (*SHapley Additive ExPlanations*), Boruta, or RFE (*Recursive Feature Elimination*) are used for its solution [57].

Once the mapping between reference values and poverty indicators is established, we obtain the poverty index for each 469 m grid cell by inputting its values into the regressor and obtaining the estimate. The privacy of individuals is preserved by inferring poverty indicators only for 469 m grid cells with more than two households.

3.2.2. Mapping satellite images to poverty indicators

We employ a battery of ML architectures, including classical neural networks, Convolutional Neural Networks (CNNs), Transformers, capsule attention networks, graph-based methods, and Foundation models, to establish a mapping between satellite images and poverty indicators. The aim is to study a broad set of paradigms to identify the most effective approach for establishing the mapping. In this approach, we customize ML architectures to analyze remote-observation-based data, extracting features indicative of poverty, particularly the multidimensional poverty model's *housing quality* indicator.

The choice of *housing quality* in the second stage is deliberate and motivated by physical observability, data validity, and methodological scope. Regarding physical observability, we noticed that among the 16 indicators of Mexico's official multidimensional poverty model, *housing quality* is the component most directly observable from optical EO data. In contrast, other dimensions such as income, social security, food access, or education depend primarily on household-level economic and institutional processes that are not directly observable from space. Furthermore, the census-to-poverty mapping results show that *housing quality* is one of the most reliably predicted indicators from census variables. At the same time, the SHAP analysis confirms that it is driven by interpretable, physically grounded features (*e.g.*, dirt floors, drainage, and access to electricity). Finally, the objective of the second stage is to demonstrate how EO and modern ML can increase the spatial and temporal resolution of a well-defined, policy-relevant component of the official poverty framework, rather than to reconstruct the full multidimensional poverty index from satellite data alone. We view housing quality as a gateway indicator: it is intrinsically important and strongly correlated with other deprivation dimensions.

To establish the best mapping, the training process adjusts the parameters that define the relationship between satellite images and poverty indicators to minimize a loss function. We use the determination coefficient (R^2), which quantifies the extent to which the dependent variable is explained by the independent variables, as a loss function. R^2 is more informative for regression analysis than other measures, such as SMAPE (Symmetric Mean Absolute Percentage Error), MAPE (Mean Absolute Percentage Error), MSE (Mean Squared Error), MAE (Mean Absolute Error), and RMSE (Root Mean Square Error) [58]. Notwithstanding, we provide performance values for the latter two. In modern ML, the number of parameters is often larger than the amount of data used, which can lead to overfitting [59]. We mitigate the overfitting by including sparsity-promoting regularization terms in the loss function to control parameter values [60], using dropout to prevent the creation of complex co-adaptations [61], and implementing early stopping during training when the loss on the validation data stops decreasing, preventing the model from overfitting to noisy labels [62].

4. ML algorithms setup

To describe the preparation of ML models, we distinguish between models that map census information to poverty indicators and those that map remote observations to poverty indicators.

4.1. Census information to poverty indicators

To optimize the estimation process, we fine-tuned the hyperparameters of the regressors mapping census data to the multidimensional poverty model values at the 469 m cell grid level.

XGB [63]. We performed hyperparameter tuning for the XGBoost model using a random search with 10,000 iterations and 3-fold cross-validation. An optimization is performed to estimate multiple response variables related to poverty indicators based on census data. This process aims to identify the optimal configuration of model parameters to improve its predictive capability. The following hyperparameters were optimized during tuning: the learning rate, explored within a range of 0.01 to 0.3; the fraction of randomly selected columns for constructing each tree, evaluated within the interval of 0.1 to 1.0; the maximum tree depth, ranging from 1 to 6; the proportion of samples used to construct each tree, within a range of 0.1 to 1.0; and the regularization parameter, adjusted within the interval of 0 to 1. We split the dataset into two subsets: 50% for training and 50% for testing. This process is repeated 30 times with different random partitions. Before training, we removed observations with missing values in the predictor variables and the corresponding response variable. Subsequently, we normalized the training and test data using the parameters from the training set.

NN [59]. We trained multiple neural network architectures to identify the one with the best generalization ability and to optimize its estimation performance. The hyperparameter selection was conducted via an exhaustive search over combinations of hidden layers and neurons to find the optimal configuration for capturing the underlying relationships in the data. The implemented neural network model consists of multiple densely connected layers with ReLU nonlinearities. We tested configurations with 1 to 4 hidden layers and different numbers of neurons per layer, ranging from 5 to 100 in increments of 5. The model used the Adam optimizer with a fixed learning rate of 0.001 during training. The data partition consisted of three sets: 50% for training, 20% for validation, and 30% for testing. This partitioning process was repeated 30 times with different random configurations to evaluate the model's stability and consistency. Before training, we normalized the data, using only the training set for adjustment, and then applied the parameters to the validation and test sets.

SVR [64]. We optimized the model hyperparameters through a random search. The implemented SVR model utilized different kernel configurations, including radial basis function (RBF) and linear kernels. For the selection of optimal hyperparameters, we defined a search space, including the following parameters: the regularization hyperparameter (C), adjusted within a logarithmic range between 0.01 and 10; the kernel function coefficient for polynomial and sigmoid types (coef0), explored within a uniform range of -1 to 1; the polynomial degree, evaluated between 2 and 5; the margin-of-error control parameter (epsilon), defined within a range of 0.001 to 0.1; and the tolerance for the stopping criterion (tol), varying between 10^{-3} and 10^{-2}. We also considered whether to use the dimensionality-reduction heuristics (shrinking) and the maximum number of iterations (explored at 5 000, 10 000, and 20 000). The training procedure involves splitting the data into two sets: 50% for training and 50% for testing. This split follows Afendras and Markatou [65], who show that, for broad loss functions and models, the variance of the generalization-error estimate is minimized when the training size equals half the sample size, independent of

the data distribution and task. The data is normalized using parameters adjusted with the training set, and these parameters are applied to the test data. Hyperparameter search was conducted using 3-fold cross-validation. This process was repeated 30 times for each poverty indicator, using random partitions in each execution. We selected the optimal model based on cross-validation performance, then trained and evaluated it on the training and test sets, respectively.

Ensemble. The ensemble model integrates predictions from multiple single models (a neural network, an XGBoost model, and an SVR model) to improve the accuracy of poverty indicator estimation. The primary objective of the tuning process is to identify the optimal neural network architecture configuration that captures the relationships between input and output variables. During the tuning process, we optimized the following hyperparameters of the ensemble neural network: the number of hidden layers explored within a range of 1 to 4 layers; the number of neurons per layer, varying between 3 and 20 neurons; the learning rate, fixed at 0.001 for the Adam optimizer; and the loss function used, which is the mean squared error. We assessed the model's robustness by partitioning the data into three sets: 50% for training, 20% for validation, and 30% for testing. This step was repeated 30 times with different random partitions. We normalized the data using scalers previously adjusted for each base model (NN, XGB, SVR).

4.2. Remote observations to poverty indicators

We tested a variety of ML architectures, including classical approaches, CNNs, Capsule Attention Networks, Graph-based methods, Transformers, and Foundation Models. Across all models, the target variable was defined as the mean of 30 ensemble predictions per sample. Unless otherwise noted, we used 12-band Sentinel-2 tiles radiometrically scaled (by 5×10^{-5} or 5×10^{-3} depending on the backbone), optionally augmented with per-pixel spectral indices (see Appendix B), yielding *bands-only* and *bands+indices* variants. For capsule- and attention-based models, inputs were additionally normalized per band using robust percentile clipping followed by min–max scaling to $[0, 1]$, which improved optimization stability. Images were cropped or resized according to the architecture requirements (*e.g.*, 24×24 patches for tree-based models, 224×224 for CNNs and ViTs). The data were split into training, validation, and test partitions (50%–20%–30% or 50%–50%), again based on Afrendas and Markatou [65]. Features were standardized or normalized using training statistics. Training was performed with Adam or AdamW optimizers under an MSE loss, with learning rate schedules (OneCycleLR [66] or cosine annealing [67]) and early stopping on validation R^2. Performance is consistently reported as R^2 , MAE and RMSE on the held-out test set.

Classical models. We experimented with XGBoost (XGB) [63] and linear regression. The latter did not yield meaningful results. For XGB, we tuned hyperparameters via a 3-fold random search (200 draws over learning rate, max depth, subsample ratios, γ, and min child weight) starting from 100 decision trees with an MSE objective.

Graph-based methods. We employed two state-of-the-art unsupervised learning algorithms: Structured Doubly Stochastic Graph-based Clustering (SDSGC) [68] and Multi-order based Clustering via Dynamical Low Rank Tensor approximation (MCDLT) [69]. To evaluate their performance, we assigned poverty values to each cluster and tested configurations ranging from 10 to 100 clusters. For image features, we tested raw pixel intensities (24×24-pixel images with 12 bands, flattened to 6,912-dimensional vectors) and 1,536-dimensional embeddings from an EfficientNet-B3 model pretrained on ImageNet. The model generated embeddings from the Sentinel-2 RGB bands or from a 3-channel projection of all 22 bands (*bands+indices*) via Kernel PCA with a Radial Basis Function (RBF) kernel.

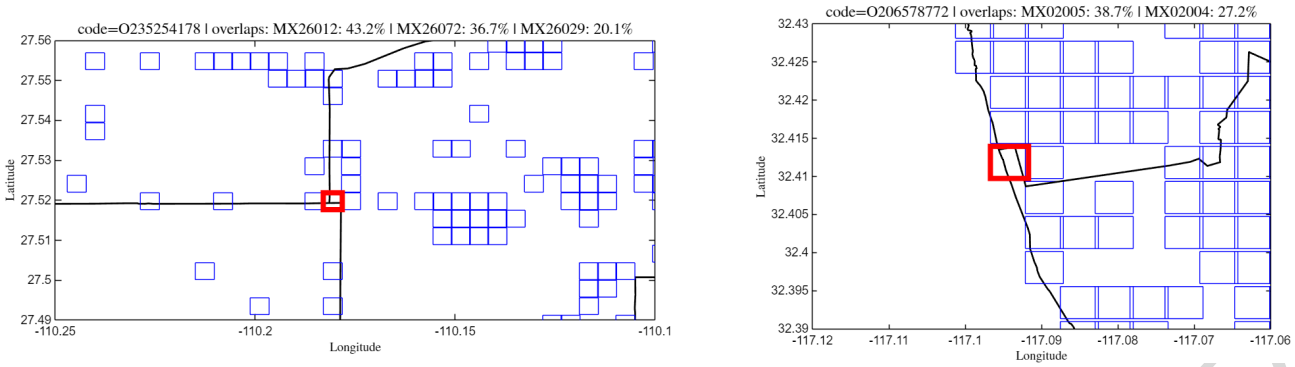
Convolutional Neural Networks. We evaluated DenseNet-169 [70], EfficientNet-B3 [71], and ResNet-50 [72], all initialized with ImageNet-pretrained weights. Architectures were adapted by replacing the classifier head with a single-unit regressor (sigmoid or linear activation) and, when required, modifying the first convolution to accept 12 or 22 channels. DenseNet and EfficientNet used resized 224×224 inputs with dropout regularization, while ResNet duplicated the last band to satisfy its 13-channel input and employed OneCycleLR scheduling.

Capsule Attention Networks. To explicitly capture part–whole relationships and spatial interactions beyond fixed receptive fields, we implemented a Capsule Attention Network (CAN) regressor [73]. Following an initial convolutional stem, feature maps were reorganized into sets of primary capsules using a convolutional operator that generates K capsule types of dimension D at each spatial location, yielding $N = KH'W'$ capsule vectors after reshaping. Capsule vectors were transformed with a squash nonlinearity to encode presence via vector length. Global interactions among capsules were then modeled through L stacked self-attention blocks operating directly in capsule space, each combined with residual MLP updates. The final prediction was obtained by aggregating capsule representations via mean pooling and projecting the resulting embedding to a scalar output with a shallow regression head.

Transformers. We explored two transformer-based architectures: the Vision Transformer (ViT) [74] and the Swin Transformer [75]. For ViT, Sentinel-2 tiles were padded to 13 channels and resized to 224×224, while in the *bands+indices* case the input comprised 22 channels. Training batches were larger (64) and optimized with OneCycleLR. The Swin Transformer ingested 96×96 patches with a convolutional embedding layer, alternating standard and shifted window self-attention, and GELU activations.

Foundation models. Two embedding-based approaches were tested. AlphaEarth [42] provided 64-dimensional geospatial embeddings at 10 m resolution. We trained (i) an EfficientNet-B3 baseline adapted to 64 input channels and (ii) a sequence model (S5) operating on embeddings treated as token sequences with masked mean pooling. DINOv3 [41] generated patch-level embeddings from RGB inputs resized to 256×256. These were projected and fed into an S5 regressor. We compared two DINOv3 backbones: the 300M (ViT-L/16) and the 7B (ViT-7B/16) variants, isolating the effect of model scale.

Note that comparing model families can be confounded by differences in preprocessing, channel count, and spatial context. This research intends to provide a meaningful benchmark in which each model is evaluated under its operating regime, while holding constant the task, labels, splits, and evaluation protocol rather than an architecture-intrinsic ranking. Specifically, all models estimate the same target (a housing-quality proxy at 469 m), using the same training/validation/test partitions, loss (MSE), and metrics (R^2, MAE, RMSE). Furthermore, performance is reported on the identical held-out test set, ensuring comparability for deployment. Except for Foundation Models, we explicitly evaluate both bands-only and bands+indices variants. This evaluation isolates the effect of additional channels within model families and shows that performance trends are consistent across input dimensionality. Architectural requirements constrain differences in patch size, channel count, or resizing. For instance, CNNs/ViTs require fixed-size tensors and benefit from larger spatial context; Swin operates on smaller windows by design; XGBoost operates on flattened local summaries; and foundation models consume pretrained embeddings. Thus, preprocessing differences reflect how these models are actually used in practice, which is the comparison we aim to make.



(a) 0235254178 (land) (b) 0206578772 (coast/ocean)

Figure 2. Two examples illustrating boundary-aware aggregation for grid cells. In both panels, the red square denotes the target grid cell (469 m side length), black lines indicate municipality boundaries, and blue squares show neighbouring grid cells. (a) `0235254178` is fully over land but intersects three municipalities, with overlap fractions 43.2% for `MX26012` (Bácum), 36.7% for `MX26072` (San Ignacio Río Muerto), and 20.1% for `MX26029` (Guaymas). (b) `0206578772` intersects the coastline, so only the land portion overlaps municipalities; the computed overlaps are 38.7% for `MX02005` (Playas de Rosarito) and 27.2% for `MX02004` (Tijuana), with the remaining area falling over the ocean.

4.3. Grid-cell aggregation to municipal level

The workflow separates (i) the estimation of the mappings from (ii) the spatial support used for evaluation. The mapping f that relates census-derived reference values to the poverty index is learned at the municipal level using aggregated grid-cell census values within municipal boundaries. The satellite mapping g is learned at the grid-cell level using census reference values provided at the same grid resolution. Accordingly, spatial aggregation is required to construct the inputs for f, while the estimation of g does not involve cross-boundary aggregation. However, aggregation is also needed for the evaluation stage, where grid-cell satellite inferences are aggregated to the municipality level to enable comparison with municipality-level poverty indices used as ground truth.

Let $k \in \{1, \ldots, K\}$ denote grid-cells and $i \in \{1, \ldots, I\}$ denote municipalities. Each cell is represented by a square footprint (side length $L = 469$ m) centered at its centroid. Using polygon–polygon intersection between the cell footprint and municipality polygons, we compute the overlap fraction as:

$$\alpha_i^k = \frac{A(\mathcal{C}_k \cap \mathcal{M}_i)}{A(\mathcal{C}_k)} \in [0, 1], \tag{1}$$

where $\mathcal{C}_k$ is the cell polygon, $\mathcal{M}_i$ is the municipality polygon, and $A(\cdot)$ denotes area. By construction, a cell can intersect multiple municipalities (up to 4 in Mexico), and $\sum_{i=1}^{I} \alpha_i^k \approx 1$ within numerical tolerance in most cases. A notable exception is cells bordering the water bodies or country limits. At any rate, occupancy is normalised when required to enforce $\sum_i \alpha_i^k = 1$.

To aggregate cell-level poverty predictions $\hat{v}^k$ from g into a municipality-level estimate, we use an overlap- and population-weighted average. Let p^k be the cell population and p_i^k the portion attributed to municipality i. The municipality's total is $p_i = \sum_{k=1}^{K} p_i^k$, and the aggregated estimate is

$$\hat{v}_i = \frac{1}{p_i} \sum_{k=1}^{K} \alpha_i^k \, \hat{v}^k \, p_i^k. \tag{2}$$

Figure 2 illustrates a representative boundary case in which a single cell overlaps three municipalities (e.g., 43.2%, 36.7%, and 20.1%), motivating overlap-aware aggregation rather than centroid-only assignment.

Satellite predictions are produced at the grid-cell level and aggregated to the municipality level using (1)–(2). We validate satellite mapping by comparing aggregated satellite estimates $\hat{v}_i$ with municipality-level poverty indices v_i in held-out areas, reporting R^2 and error distributions at the municipality level for downstream interpretation.

5. Results

This study develops a method to map satellite observations onto the Mexican multidimensional poverty measurement model indicators using data from the Census of Population and Housing [76,77]. This association involves mapping variables from census data on population and housing, collected through direct interviews, to multidimensional poverty model metrics that assess poverty conditions, such as food security and the quality of housing spaces. This first stage is followed by a mapping between satellite multispectral images and these poverty indicators at a higher spatial resolution. In this section, we evaluate the effectiveness of establishing these relationships.

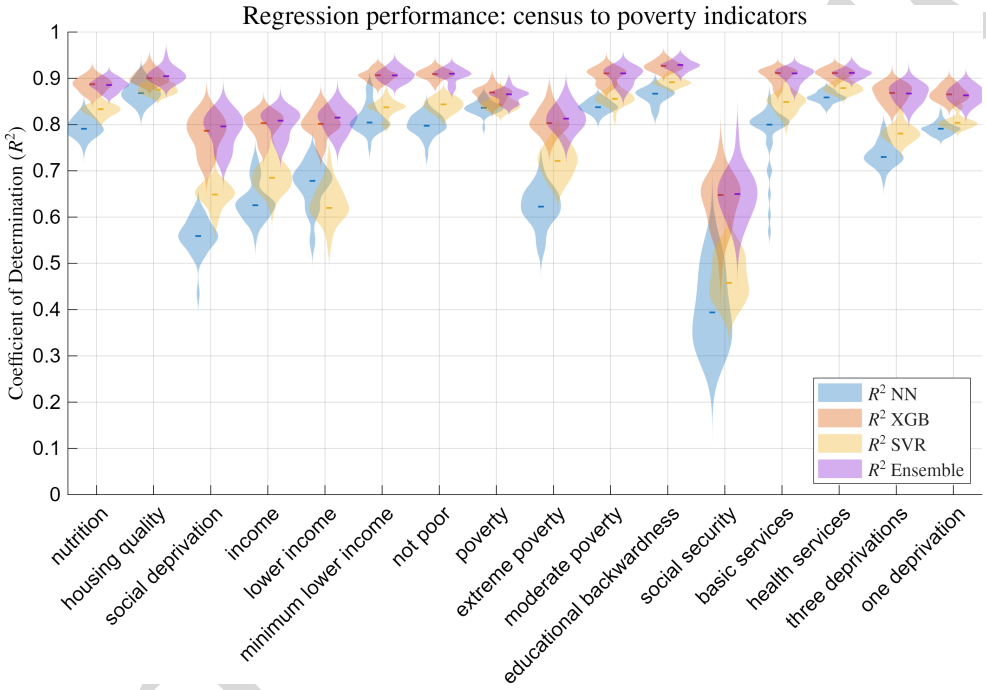


Figure 3. Performance of regression models in predicting poverty indicators from census information. The determination coefficient R^2 is reported for four modeling approaches: a neural network (NN), a decision tree-based model (XGB), a support vector machine (SVR), and an ensemble of these models based on NN. The results indicate that XGB and the ensemble achieve the best performance, with R^2 values close to 0.9 for most indicators, while NN and SVR show more variable performance, with some cases of lower accuracy. The predictors correspond to the census information described in Table A1, while the response variable corresponds to the multidimensional poverty model.

5.1. From census reference values to poverty indicators

Figure 3 shows the performance of different regressors in predicting various poverty indicators in terms of the determination coefficient R^2. The predictors correspond to the census variables described in Table A1, while the response variable corresponds to each of the indicators of the multidimensional poverty model. The distributions from the 30 runs for each regressor are shown in a violin plot. Four approaches are compared: a neural network (NN), a decision tree-based model (XGB), a support vector machine (SVR), and an ensemble model built on an NN that combines the predictions of the three previous models. To train the regression model, we first aggregated census reference variables from the 469 m cells, which are the units of the source information, to the municipal level. For

each variable, this aggregation was computed as the proportion of values corresponding to one of the optional responses in a census question.

In general, the XGB models and the ensemble achieve superior performance, with R^2 values of approximately 0.9 across half of the indicators, with slight variations. The performance curves for XGB and the ensemble overlap at almost all points; in fact, they are statistically indistinguishable, suggesting that the ensemble does not significantly improve the decision tree-based model's performance. In contrast, the neural network and SVR models exhibit more variable behavior, with R^2 values ranging from 0.4 to 0.85 across the evaluated indicators. The SVR model performs slightly better than the neural network across most indicators, achieving R^2 values of 0.75–0.85. However, the neural network outperforms the SVR on specific indicators, such as *lower income*, achieving a mean R^2 of 0.67, while the SVR remains around 0.63. In the worst cases, the performance of the individual NN and SVR models drops below 0.5, as observed in the prediction of *social security*, where both models struggle to capture data variability.

In contrast, XGB and the ensemble maintain robust performance, with mean R^2 values above 0.85 across nearly all categories, demonstrating their ability to model complex data effectively. For this reason, the XGB model and the ensemble are the most suitable approaches for predicting poverty indicators, while the NN and SVR models offer acceptable but lower performance. Hereafter, we will continue discussing results using XGB for testing.

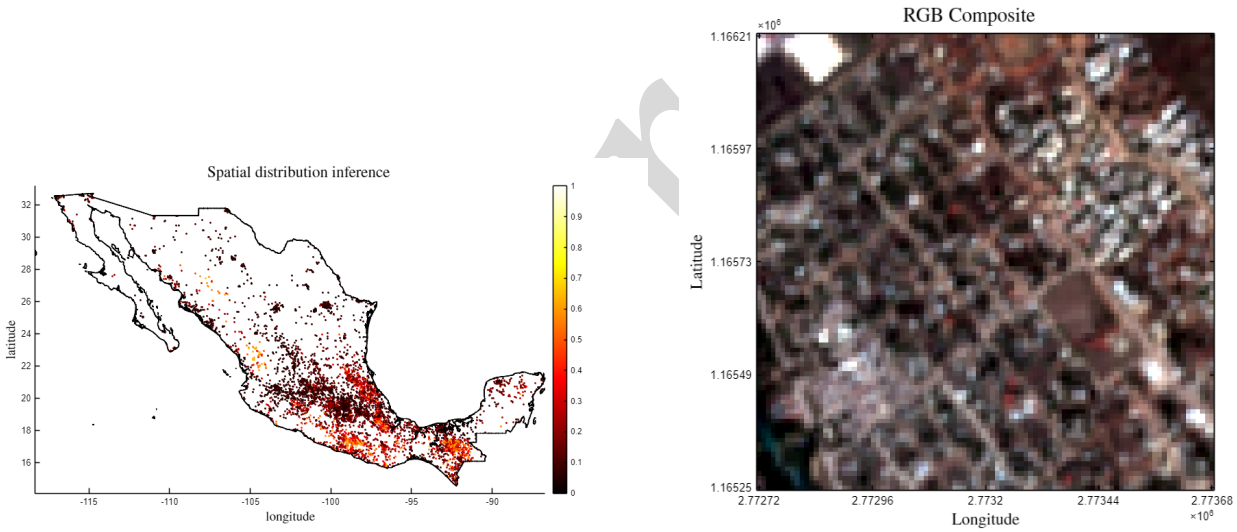


Figure 4. Mapping of multispectral images to poverty estimates. The image patches we use are geographically distributed across Mexico's territory (a). (b) shows an example of an image patch used for the training process. Only the RGB channels are displayed. The image corresponds to a 96×96 pixel patch centered on the geostatistical grid cell.

5.2. From remote observations to poverty indicators

We used geolocation data from the geostatistical 469 m grid cells to define regions of interest and crop Sentinel-2 satellite images or AlphaEarth embeddings. Each patch represents a 960×960 m^2 area centered on a cell of the geostatistical grid (see Figure 4). The extended area aims to provide enough context for the location under analysis. We cropped 12,694 image patches corresponding to complete data in the census data table. Then, we trained a variety of regressors, including classical methods, CNN, Vision Transformers, Foundation Models, Capsule Attention Networks, and graph-based algorithms. As a representative of the classical methods, we used Extreme Gradient Boosting (XGB) [63]. For CNN, we choose DenseNet-169 [70], EfficientNet-B3 [71] and ResNet50 [72] architectures. For visual transformers, we selected Swin [75] and ViT [74]. As representatives of

foundation models, we coupled the embeddings of AlphaEarth with an S5 model [78] and the EfficientNet-B3 CNN, and with the DINOv3 embeddings fed into an S5 architecture. For the graph-based methods, we evaluated the SDSGC [68] and MCDLT [69] algorithms. We additionally tested a Capsule Attention Network (CAN) as an alternative inductive bias for multispectral regression.

For the ResNet50-based model, we used an instance pretrained on the MoCo database for Sentinel-2. For the EfficientNet-B3 and DenseNet-169 CNNs, we used ImageNet-pretrained models to initialize the weights of the shape-compatible layers. We used two datasets to train the models. One set included 12-band multispectral Sentinel-2 satellite images as predictors, along with poverty indicators related to *housing quality*, previously derived as response variables at the geostatistical grid-cell level. Following USGS recommendations, image intensity values were scaled by 0.00005 for normalization. On the second dataset, we added 10 spectral indices to the multispectral images (see Appendix B). We split the dataset into training, validation, and test subsets ($50\% - 20\% - 30\%$). The training process involved adjusting the CNN parameters to minimize the mean squared error between the model's poverty predictions and the actual poverty levels of the geostatistical grid cells. We set the learning rate of the CNN models following the one cycle learning rate policy (OneCycleLR) [66,79]. For SDSGC and MCDLT, during the training stage, we grouped the satellite images using cluster counts ranging from 10 to 100. We calculated the poverty index of each cluster as the mean value of the training set samples assigned to that cluster. During inference, we assigned the poverty index of each cluster to the test samples in that cluster. As image features, we used embeddings from an EfficientNet-B3 model pretrained on ImageNet with the RGB bands as input.

Two CAN input variants were evaluated. The first uses a 13-channel tensor obtained by duplicating the last Sentinel-2 band to satisfy fixed-channel model interfaces. The second augments the input with spectral-index channels computed from six base bands (blue, green, red, NIR, SWIR1, SWIR2), yielding an extended channel stack. In both variants, we applied a per-band percentile normalization (clipping at 0.5/99.5 percentiles and rescaling) to reduce the influence of outliers and improve gradient behavior in capsule representations.

We evaluated the models' performance on the test dataset partition. Table 1 shows the determination coefficient R^2, mean absolute error (MAE) and root mean squared error (RMSE) for the models trained with the 12-band dataset and the models trained with the dataset that included the 12-band images complemented with 10 spectral indices. In general, the problem appears to be challenging for ML algorithms.

The inclusion of additional spectral bands affected the models differently. For XGB, increasing input from 12 to 22 bands improved R^2 from 0.473 to 0.495. The Vision Transformer (ViT) showed a small gain, with R^2 rising from 0.560 to 0.561. By contrast, Swin Transformer had a minor drop, with R^2 decreasing from 0.600 to 0.599. CNN–based models generally benefited from extra spectral information. DenseNet-169 saw R^2 grow from 0.632 to 0.636, and EfficientNet-B3 improved from 0.608 to 0.655. Conversely, ResNet-50 declined from 0.627 to 0.615 after adding more bands.

Given EfficientNet-B3's strong performance, we fed it AlphaEarth embeddings; however, it did not surpass S5, suggesting that the model used to process the embeddings was crucial. Thus, S5 was our model of choice for a 300-parameter model using DINO v3 embeddings. However, the 7B parameter models performed well, achieving a determination coefficient of 0.683. It was worth remembering that while AlphaEarth embeddings contain 64 channels, DINO v3 is fed with three channels, in our case, RGB. For the graph-based methods, we achieved the best performance with EfficientNet-B3 embeddings from Sentinel-2 RGB composite images, using 80 clusters for SDSGC and 44 clusters for MCDLT,

Table 1. Performance of regression models in predicting poverty using Sentinel-2 images with 12 and 22 bands. Metrics reported include the coefficient of determination (R^2), Mean Absolute Error (MAE), and Root Mean Square Error (RMSE).

		R^2		MAE		RMSE (%)	
	Method	**12 bands**	**22 bands**	**12 bands**	**22 bands**	**12 bands**	**22 bands**
Classical	XGB	0.473	0.495	0.062	0.060	0.096	0.094
CNN	ResNet50	0.627	0.615	0.047	0.049	0.081	0.082
	EfficientNet-B3	0.608	0.655	0.052	0.049	0.083	0.077
	DenseNet-169	0.632	0.636	0.049	0.048	0.081	0.080
Transformers	Swin	0.600	0.599	0.052	0.051	0.083	0.084
	ViT	0.560	0.561	0.057	0.058	0.088	0.088
Capsule Attention Networks	CAN	0.530	0.521	0.057	0.058	0.089	0.092

	Method	R^2	**MAE**	**RMSE (%)**
Foundation Models	AlphaEarth + EfficientNet-B3	0.640	0.050	0.080
	DINO v3 (small) + S5	0.644	0.049	0.079
	AlphaEarth + S5	0.677	0.050	0.076
	DINO v3 (large) + S5	0.683	0.047	0.076
Graph-based Algorithms	SDSGC	0.424	0.065	0.099
	MCDLT	0.431	0.064	0.098

yielding determination coefficients of 0.424 and 0.431, respectively. Furthermore, the proposed Capsule Attention Network (CAN) achieved a coefficient of determination R^2 of 0.530 with 12 bands and 0.521 with 22 bands.

Across all models, MAE remained within a narrow range (0.047–0.065), reflecting similar average prediction errors across methods. RMSE ranged from 0.076 to 0.099, indicating moderate sensitivity to errors of larger magnitude. In comparison, foundation models exhibited lower RMSE values than classical and graph-based approaches.

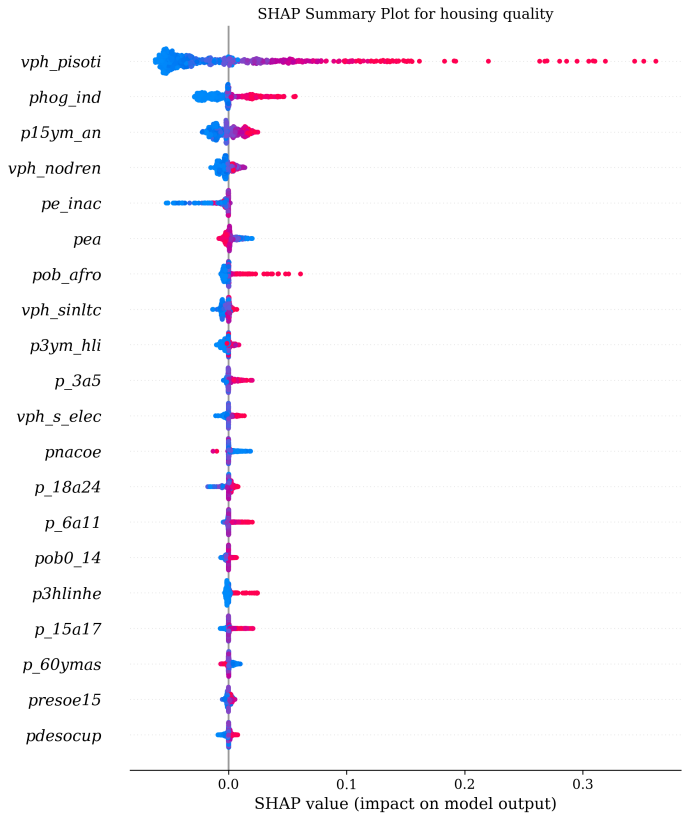


Figure 5. Importance of variables in infering *housing quality* according to SHAP Analysis. The variable vph_pisoti shows the most significant negative influence on the prediction, while factors such as p15ym_an and phog_ind also contribute significantly. Variables related to infrastructure, such as vph_s_elec and vph_nondren, have a positive impact on *housing quality*. Acronyms: *vph_pisoti* – Dirt-floor homes, *phog_ind* – Indigenous households, *p15ym_an* – Illiterate 15+, *vph_nodren* – Homes no drainage, *pe_inac* – Econ. inactive pop., *pea* – Econ. active pop., *pob_afro* – Afro-descendant pop., *vph_sinltc* – Homes no phone, *p3ym_hli* – IL speakers 3+, *p_3a5* – Pop 3–5, *vph_s_elec* – Homes no electricity, *pnacoe* – Born other state, *p_18a24* – Pop 18–24, *p_6a11* – Pop 6–11, *pob0_14* – Pop 0–14, *p3hlinhe* – 3 + no Spanish, *p_15a17* – Pop 15–17, *p_60ymas* – Pop older than 60, *presoe15* – No schooling 15+, *pdesocup* – Unemployed pop.

5.3. Features relevance analysis

To gain insight into the importance of the predictors in the inference, we performed a SHAP analysis [80] for the models trained on the census and remote sensing predictors.

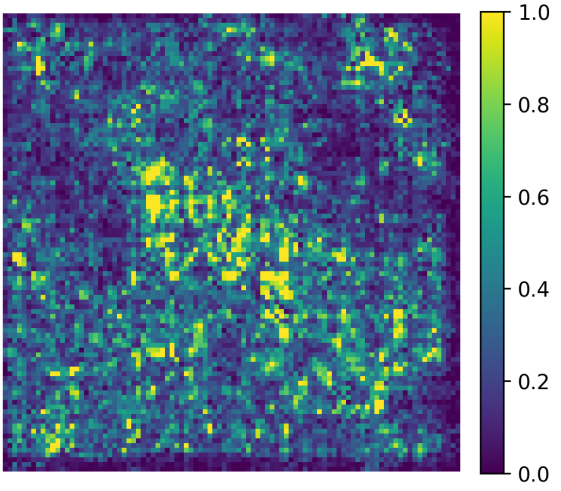
5.3.1. Census-based model interpretability

Using values from the XGB model applied to census data and the sixteen poverty indicators to infer the *housing quality* indicator, we identified several variables that significantly influence the model's predictions (see Figure 5). The variable *vph_pisoti* (homes with dirt floors) contributes the most, with SHAP values extending significantly towards negative values, indicating that its presence has a strongly negative impact on housing quality. Meanwhile, *p15ym_an* (illiterate population aged 15 and over) and *phog_ind* (indigenous households) also show significant contributions. Additionally, the variable *pob_afro* (Afro-descendant population) exhibits a moderate effect, indicating potential disparities in housing quality among different population groups.

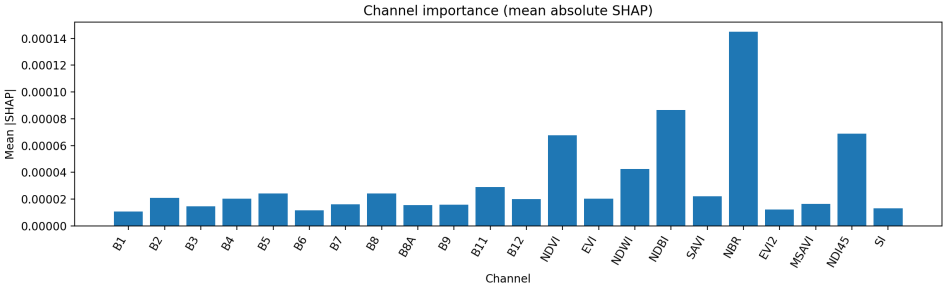
Access to basic services also appears to be a relevant predictor of housing quality. Variables such as *vph_nodren* (homes without sewage) and *vph_s_elec* (homes with electricity access) positively impact predictions, highlighting the importance of infrastructure in determining household quality of life. Similarly, *pdesocup* (total population not working) shows a moderate impact, reflecting its relationship with housing conditions across different population segments. An XGB regressor trained with the complete set of census features

results in a determination coefficient of 0.834 ± 0.004, while one trained with a reduced feature set results in 0.842 ± 0.006. The variability corresponds to 30 different train/test data splits.

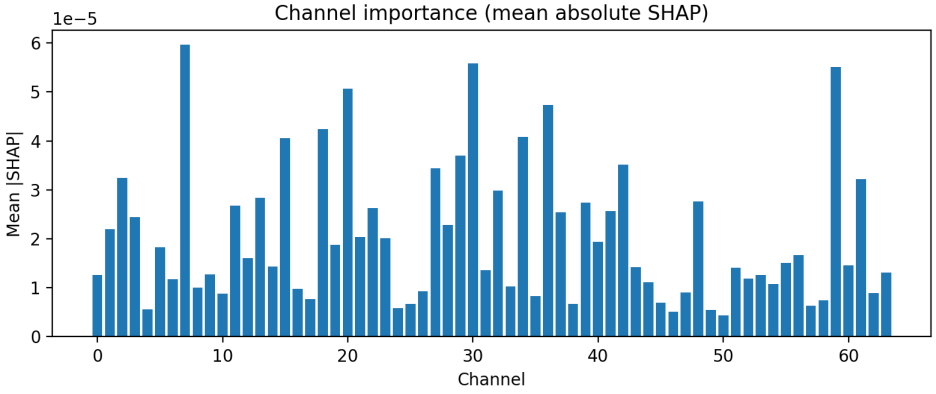
(a) RGB visualization (B4/B3/B2).



(b) $\Sigma_c \mid \mathrm{SHAP}_c \mid$ (normalized).



(c) Sentinel-2 ResNet-50 SHAP channel importance.



(d) AlphaEarth+S5 SHAP embedding-channel importance.

Figure 6. Feature relevance analysis for remote sensing-based predictors. Example Sentinel-2 tile (a) and corresponding spatial attribution for the ResNet-50 regressor. The SHAP summary map (b) highlights locations within the tile that most influence the housing-quality prediction. Bars show the mean absolute SHAP per input channel, aggregated over the explained samples and averaged over the spatial dimensions, for Sentinel-2 and AlphaEarth.

5.3.2. EO-based model interpretability

To address interpretability for the satellite-based predictor, we computed SHAP attributions directly on the inputs of the ResNet-50 regressor trained on Sentinel-2 imagery. Following the inference pipeline, each sample is a multi-channel tensor built from (i) the 12

Sentinel-2 bands (scaled by a constant factor) and (ii) 10 derived spectral indices (NDVI, EVI, NDWI, NDBI, SAVI, NBR, EVI2, MSAVI, NDI45, and SI). We used a gradient-based SHAP explainer (GradientExplainer) with a small background set drawn from the training split, and computed attributions for a subset of held-out samples. The analysis produced pixel-level SHAP maps per channel, enabling both spatial and spectral interpretation (see Figure 6(a)-(b)).

We summarize spectral relevance by aggregating SHAP magnitudes over space and samples. Specifically, for each explained image, we average $|\,\text{SHAP}\,|$ over (H, W) to obtain per-channel relevance, then average across images to get a global ranking (Figure 6(c)). The resulting ranking is dominated by indices designed to separate vegetation, burned/low-reflectance surfaces, and built-up land (e.g., NBR, NDBI, NDVI), suggesting that the model relies strongly on land-cover proxies that correlate with housing quality. In addition, the per-sample spatial attribution $\Sigma_c|\text{SHAP}_c|$ highlights where the model extracts evidence within each tile. For the illustrated sample, the highest attributions concentrate around textured, high-contrast structures and their boundaries (consistent with dense built-up patterns), rather than being uniformly distributed across the patch.

5.3.3. Foundation-model embeddings interpretability

We further provide interpretability for the foundation-model pathway by computing SHAP attributions on AlphaEarth embeddings. Each tile is represented as a 64-channel embedding image and reshaped into a sequence of tokens (one token per pixel, optionally subsampled/padded to a fixed token budget). We trained an S5-based sequence regressor that performs masked mean pooling over valid tokens and maps the pooled representation to the housing-quality target. Using GradientExplainer on this S5 regressor, we compute SHAP values over the token-by-channel inputs and then aggregate over valid tokens (masking padded positions) to obtain mean |SHAP| per embedding channel (Figure 6(d)). These 64 channels are not directly human-interpretable like spectral bands. However, the channel-importance profile quantifies which latent embedding dimensions most influence predictions and may be useful in downstream applications such as feature selection, manifold representations, or inference optimization.

5.4. Municipal-level evaluation

To evaluate satellite-based predictions against the municipality-aligned ground truth provided by the multidimensional poverty model, we adopt a two-stage workflow that (i) builds a census-to-indicator reference at the grid-cell level and (ii) performs validation after aggregating cell predictions to the municipal support. This separation ensures that learning the mappings does not require a cross-boundary averaging; aggregation is introduced only at evaluation time to make satellite inferences directly comparable to municipality-level ground-truth indicators (see § 4.3). We first reduce the census predictor space using SHAP-based importance values computed from an XGBoost regressor trained to predict the housing-quality indicator. Figure 5 shows that a small subset of variables concentrates most of the explanatory power, with *vph_pisoti* (dirt floors) dominating the SHAP magnitude and exhibiting a long positive tail in impact, followed by *phog_ind* and *p15ym_an*, among others. Guided by the observed performance gains from progressively adding the most important features, we set an importance threshold of 0.002550, reducing the predictors from 29 to 9 while preserving predictive performance.

Using these 9 predictors, we train multiple XGBoost models under repeated random splits and hyperparameter search. The resulting R^2 values as a function of the SHAP threshold indicate that the selected-feature regressor matches the performance obtained with the full set of predictors. We then deploy the trained models to infer cell-level

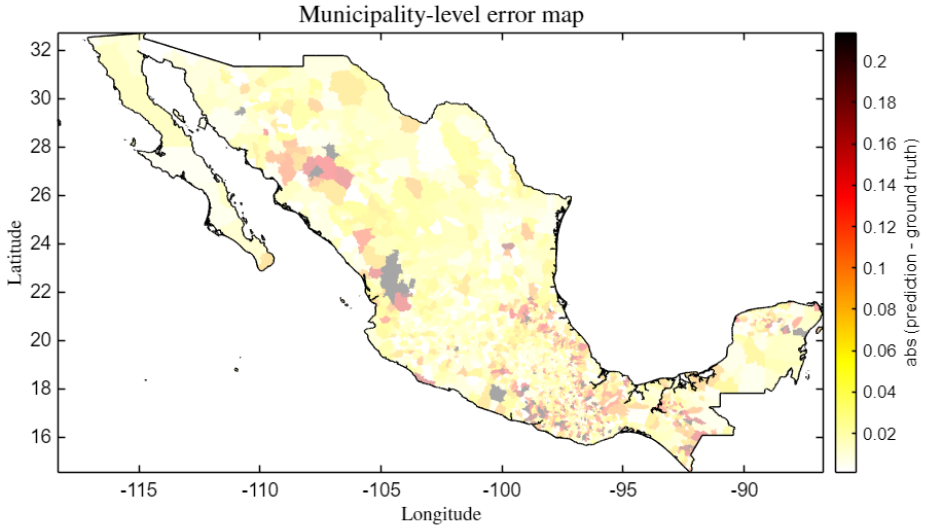


Figure 7. Municipality-level absolute error map for housing quality inference in Mexico. Colors indicate the absolute difference between model predictions and census-based ground truth ($|\hat{y} - y|$), with lighter tones denoting smaller errors and darker tones indicating larger discrepancies. The black contour delineates the national boundary. The spatial concentration of higher errors in limited regions, alongside broadly low errors elsewhere, supports the use of Sentinel-2 spectral patterns as meaningful proxies for housing-quality–related conditions.

reference values across the full grid and use them as reference targets for downstream satellite learning. Next, an EfficientNetB3-based regressor is trained to map Sentinel-2 reflectance (12 bands) augmented with the ten multispectral indices to the inferred cell-level reference targets. After training, the model produces a housing-quality prediction $\hat{v}^k$ for each grid cell k. To compare these predictions with municipality-level ground truth v_i, we aggregate cell estimates into municipality estimates $\hat{v}_i$ using the overlap- and population-weighted occupancy model in § 4.3 (Eqs. (1)–(2)). We quantify agreement between aggregated predictions $\hat{v}_i$ and municipality ground truth v_i using the coefficient of determination computed across municipalities. This municipal-level evaluation yields an R^2 of 0.610.

5.5. Error evaluation

To qualitatively assess whether EO patterns encode meaningful proxies of housing quality, we inspected the spatial distribution of the municipality–level absolute error, defined as $|\hat{v}_m - v_m|$, where $\hat{v}_m$ denotes the model prediction and v_m the census–derived housing quality indicator for municipality m (Figure 7). Municipality-level predictions $\hat{v}_m$ are obtained by aggregating cell-level predictions using a population-weighted spatial occupancy model (see § 4.3). The resulting error map shows that most municipalities exhibit low absolute errors, with a median error of 0.041 and an interquartile range of [0.041, 0.071]. Across municipalities, the mean absolute error is 0.055, while 90% have errors below 0.121 and 98% below 0.213.

Spatially, low-error regions (lighter tones) dominate large portions of the country, indicating that the model captures consistent spectral-socioeconomic relationships across diverse geographic contexts. Higher errors (darker tones) appear sparsely, suggesting systematic challenges in specific settings, such as heterogeneous urban–rural transition zones or municipalities with mixed roofing materials and land–cover compositions. Importantly, the absence of widespread, high-magnitude errors supports the assumption that EO-based data provide informative proxies for housing quality at the municipal scale.

5.6. Regional validation

A total of 96,479 samples were used to construct the training, validation, and test partitions, following a strict territorial splitting strategy at the municipal level rather than a random assignment of grid cells. These samples correspond to information from 2, 418 municipalities (m). Of these, 1, 209 municipalities were assigned to the training set, 484 to the validation set, and 727 to the test set, ensuring no spatial overlap between the partitions. In all cases, the full set of geostatistical grid cells associated with each municipality was included, preserving the spatial coherence of municipal administrative units and preventing spatial information leakage. As a result, the training set comprised 49,999 grid cells, the validation set 18,458, and the test set 28,022. Under this territorial partitioning scheme, the EfficientNet-B3 model was retrained. Using the trained weights, inference was subsequently performed on a nationwide dataset comprising 697,581 grid cells, representative of the national scale. Municipality-level predictions, $\hat{v}_m$ were obtained by aggregating grid-cell predictions using a population-weighted aggregation scheme (see Section 4.3) when these aggregated estimates were compared with the municipal-level ground-truth values of housing quality poverty, v_m, an R^2 value of 0.619 was obtained, which closely matches the previously reported value of 0.610 using randomly selected partitions.

5.7. Carbon footprint emissions

The use of artificial intelligence models consumes significant energy, resulting in substantial carbon emissions. To estimate the emissions of the ML models, we used CodeCarbon [81], a tool that calculates carbon dioxide (CO_2) emissions from computing resources used to execute code.

This analysis focuses on inference-time emissions, which are directly observable, reproducible, and representative of the operational costs incurred during large-scale deployment. Training-phase emissions are an important component of environmental impact assessments. However, for models and embeddings pretrained or released by third parties, such as large foundation models, a complete and accurate accounting of training emissions is not feasible due to the lack of disclosure regarding training duration, hardware configuration, optimization strategies, and energy sources. As a result, the training carbon footprint of these models is effectively unknown and cannot be reliably reconstructed. In Table 2, we aggregated the average emissions of each model when obtaining one image prediction from the test set. The estimated CO_2 emissions per instance during the inference stage vary notably across model families (see Figure 8). Among the regression models, the XGBoost regressor exhibited the lowest carbon footprint, emitting approximately 1.056×10^{-6} g CO_2 per instance with 12 bands, and 1.103×10^{-6} g CO_2 with 23 bands. In contrast, the DINOv3 (large)+S5 model recorded the highest emissions, with 1.602×10^{-2} g CO_2 per instance, reflecting the substantially higher computational demand of foundation models. Intermediate emissions were observed for convolutional and transformer-based architectures, with values increasing from the lightweight ResNet50 (2.7×10^{-5}–3.4×10^{-5} g/instance) to heavier models such as DenseNet-169, CAN and EfficientNet-B3 (up to 9.4×10^{-4} g/instance). Graph-based algorithms produced the lowest emissions, averaging 1.4×10^{-7} g/instance. In this context, models such as EfficientNet-B3 were selected for error-propagation verification and regional validation, as they offer a favorable trade-off between predictive performance and environmental efficiency compared to foundation models.

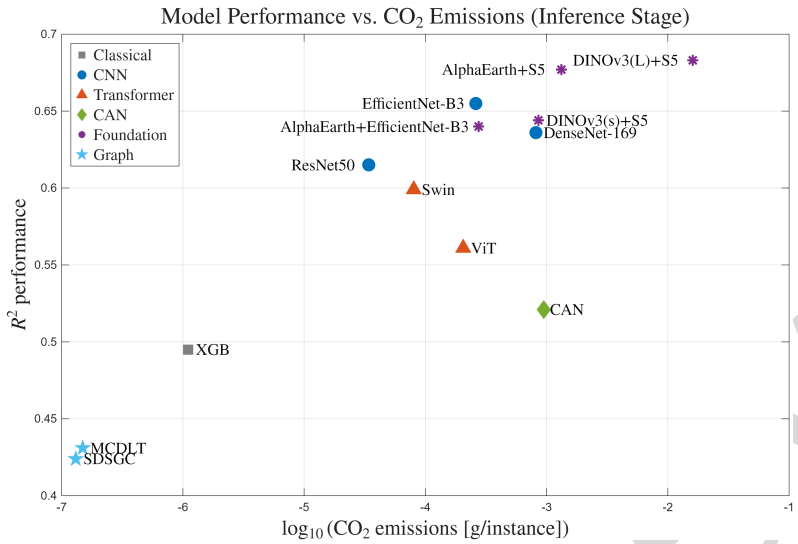


Figure 8. Relationship between model performance and carbon emissions during inference. Each point represents a regression model used to predict poverty from a 469 m grid cell in the geostatistical meshgrid, plotted by its coefficient of determination (R^2) against the base-10 logarithm of CO_2 emissions per processed instance. Marker shapes denote model families: square (Classical), circle (CNN), triangle (Transformer), diamond (Foundation Models), star (Capsule Attention Network, CAN), and pentagram (Graph-based models). The figure illustrates a trade-off between predictive performance and inference-related carbon emissions: foundation and transformer-based architectures generally achieve higher accuracy at the cost of increased emissions, while classical, graph-based, and lightweight convolutional models exhibit lower emissions but reduced performance. The CAN model occupies an intermediate regime, balancing moderate predictive accuracy with comparatively restrained emissions.

Table 2. CO_2 emissions during the inference stage for regression models predicting poverty using 12-band Sentinel-2 images and 22-band images, which include 10 spectral indices.

	Method	CO_2 (g/instance) 12 bands	CO_2 (g/instance) 22 bands
Classical	XGB	1.056×10^{-6}	1.103×10^{-6}
CNN	ResNet50	2.722×10^{-5}	3.405×10^{-5}
	EfficientNet-B3	1.945×10^{-4}	2.606×10^{-4}
	DenseNet-169	5.714×10^{-4}	8.178×10^{-4}
Transformers	Swin	5.840×10^{-5}	8.040×10^{-5}
	ViT	2.012×10^{-4}	2.053×10^{-4}
Capsule Attention Networks	CAN	9.451×10^{-4}	9.453×10^{-4}
Foundation Models	AlphaEarth + EfficientNet-B3	2.746×10^{-4}	
	DINO v3 (small) + S5	8.581×10^{-4}	
	AlphaEarth + S5	1.323×10^{-3}	
	DINO v3 (large) + S5	1.602×10^{-2}	
Graph-based Algorithms	SDSGC	1.308×10^{-7}	
	MCDLT	1.491×10^{-7}	

6. Discussion

Various approaches have been used to determine socioeconomic conditions, including poverty, through the application of ML [14–22]. This research employs, for the first time, census data summaries on a regular, equal-area geostatistical grid at 469 m resolution [46]. To evaluate inference error propagation through the two-stage learning setup, predictions generated at the geostatistical grid-cell level from satellite imagery (697,581 samples) were aggregated to the municipal level and compared with ground-truth poverty indicators, allowing quantification of error accumulation. A somewhat similar approach is employed by Jean *et al.* [25], who aggregate survey values to compare with predicted values. It is also the case of Yeh *et al.* [30], who aggregate predictions and ground measurements to the district level and use this strategy to assess accumulated error. This study exemplifies how ML models and remote sensing can serve as components of a socioeconomic module, enabling the generation of poverty indicators for data-driven decision support [82]. The generation of geostatistical grid-level poverty proxies updated at satellite passes provides a blueprint for these models to encompass human well-being.

The definition of poverty is complicated by its multidimensional nature. This study adopts the definition from the Mexican national agency responsible for measuring poverty [11]. This approach is consistent with others [23,24] that rely on poverty definitions established by national statistical agencies. One aspect that sets us apart is that we increase the resolution at which these statistics are processed through a two-steps learning process: first, a mapping from census data to poverty indicators is approximated; then the resolution is enhanced, and finally, a mapping to remote sensing observations is established. Similar to other approaches [17,19,25–33], we have studied various architectures for poverty estimation, including classical methods, CNNs, Transformers, Capsule Attention Networks, Graph-based algorithms, and Foundation models, and utilized remote sensing features at 10 m per pixel. In addition, we have explored the use of spectral indices [52], aiming to provide a richer description of the available information by embedding physical models of spectral band interactions, thereby achieving more detailed characterizations of the phenomenon.

The feature analysis highlights those most important for mapping census data to multidimensional poverty models, particularly those reflecting the quality of housing conditions. This issue is a natural concern arising from the initial use of a diverse set of predictors [34–37]. This exercise is included for the first mapping (census to poverty indicators) and for the mapping from remote observations to poverty indicators. On the other hand, the rise of foundation model embedding makes feature interpretation more abstract. While it is true that building smaller models, which could potentially perform inference faster and with lower environmental impact, might result from determining the manifold in which the features reside [83], this exercise is valuable in its own right. It has been left as an extension of this research.

Our study contributes to the growing body of evidence highlighting the relevance of foundation models in current ML frameworks. Their level of performance compared to other ML architectures underscores their importance [41,42]. Notably, the results obtained with DINOv3 were somewhat surprising, as it outperformed AlphaEarth despite using only three bands. In both cases, the volume and richness of the training data, as well as the large-scale capacity of inference models, emphasize their value in learning processes [38,40]. The modest performance gain for the foundation models over stronger CNNs, despite their much higher complexity, could be explained by the inherent limitations of predicting the poverty index from satellite images. Mapping census data to a poverty indicator introduces uncertainty into image labels, restricting the maximum performance achievable by any

model. Furthermore, the low spatial resolution makes it difficult to capture fine features related to housing quality, so some relevant information is missing from the data.

The emissions of ML models have increased over time, from 0.01 tons for AlexNet to 8,930 tons for Llama 3.1405B [84]. As a consequence, starting in 2019 [85], more studies have focused on analyzing the effect of using ML techniques on greenhouse gas production. To obtain CO_2 estimates, we evaluated emissions during inference across different models [81]. We did this to enable comparison during operation, given limited knowledge of foundation model emissions during training and the fact that most emphasis has traditionally been placed on the latter phase [86]. Our analysis evaluates individual models during inference. Recently, techniques have emerged that estimate training-related emissions for thousands of open-source models, providing an industry-scale perspective [87]. Our representation follows the spirit of Gowda *et al.* [88], who used the logarithm of electricity consumption instead of the logarithm of CO_2 emissions. We believe that the latter offers a more direct appreciation of the greenhouse gas emission problem, allowing for a clearer understanding of the trade-off between performance and carbon footprint.

Conclusion

The extensive technological resources available today, including remote sensing and GPU-based computers, facilitate the application of ML-based techniques to address socioeconomic sustainability challenges. This document describes a framework for estimating housing quality poverty using EO, pre-trained model weights, and fine-tuning with local data. The predictive model's performance enables higher spatio-temporal-resolution estimates, surpassing those of the current approach based on house surveys. This new scenario allows the preparation of a preliminary estimate, which can then be refined with field data. Our results indicate that the framework provides a fast and reliable tool for poverty assessment at the geostatistical grid cell level. One interesting surprise is the remarkable performance achieved with foundation models, which highlights the importance of data expressiveness and the capabilities of ML, albeit at the expense of substantial greenhouse gas emissions for the largest models.

One important issue is the potential for bias in the data used to train estimation models, as we only used uncensored information. Therefore, it may be worthwhile to complement these models with other sources of information to enhance the accuracy of the estimates. Additionally, the ethical use of the results from automated data analysis must be considered and delimited. At any rate, the strategy presented in this study enables rapid, cost-effective, and accessible estimates, serving as a first step in guiding survey-based efforts.

Future research will focus on estimating inter-census poverty assessments, including validating them with independent sources such as high-frequency household surveys, mobile phone mobility data, or alternative proxies. Furthermore, we will explore the disentanglement of environmental and socioeconomic drivers of poverty, possibly incorporating additional information on climate anomalies and land-use change. Additionally, it would be interesting to assess the impact of intervention policies. The proposed models can be combined with records of public investment, infrastructure programs, or conditional cash transfer coverage to detect statistically significant improvements in predicted poverty indicators over time.

Appendix A. Grid aggregated descriptors

Each 469 m grid cell is annotated with the 2020 Population and Housing Census [46] variables shown in Table A1. The mnemonics reproduce original field names, while the English labels summarise the underlying definition. We group the predictors into *Population* and *Dwellings* thematic blocks to mirror the demographic and housing dimensions of the

Mexican multidimensional poverty framework [11]. These 29 variables serve as explanatory features in subsequent modeling stages, combined with remotely sensed descriptors to estimate deprivation levels at the 469 m grid cell scale.

Appendix B. Spectral Indices Incorporated into the Satellite Predictors

This research expresses the physical footprint of poverty in terms of the *quality of housing spaces*, incorporating descriptors associated with low vegetation cover, extensive bare soil, sub-standard roofing materials, scarce water infrastructure, or fire scars caused by slash-and-burn practices. Sentinel-2–derived spectral indices provide spatially-explicit proxies for these features [52]; stacked as model inputs, they have been shown to improve household- or settlement- level poverty prediction by up to 10 pp in R^2 when compared with raw reflectance alone [89]. Below, the indices used in this study are organised by *Theme* so they can be mixed and matched to target specific poverty pathways.

The spectral indices reported in this section include a small positive constant ε in the denominator. This term is introduced to ensure numerical stability and prevent undefined values in pixels where the sum of spectral bands approaches zero, which may occur over water bodies, in deep shadows, due to sensor noise, or in masked areas. In practice, ε is set to a fixed, dimensionless constant ($\varepsilon = 10^{-8}$) that does not alter the physical interpretation of the index but guarantees stable computation across heterogeneous land-cover types. The same value of ε is used consistently across the indices to preserve comparability.

Theme: Vegetation condition

Normalized Difference Vegetation Index (NDVI) [90]. NDVI is widely used to assess vegetation health and density. It helps monitor agricultural areas, assess vegetation cover, and study environmental changes associated with human settlements. It is computed as

$$\mathrm{NDVI} = \frac{\mathrm{NIR} - \mathrm{Red}}{\mathrm{NIR} + \mathrm{Red} + \varepsilon}.$$

Enhanced Vegetation Index (EVI) [91]. EVI enhances the vegetation signal in high-biomass regions and reduces atmospheric effects, making it useful for urban planning and poverty assessment related to vegetation cover. It is computed as

$$\mathrm{EVI} = 2.5 \cdot \frac{\mathrm{NIR} - \mathrm{Red}}{\mathrm{NIR} + 6 \cdot \mathrm{Red} - 7.5 \cdot \mathrm{Blue} + 1 + \varepsilon}.$$

Table A1. Subset of 2020 Census indicators requested (English)

No.	Mnemonic	Indicator
		Population
1	p15ym_an	Illiterate population aged 15 +
2	p3hlinhe	Pop. (3 + yrs) who speak an Indigenous language and not Spanish
3	p3ym_hli	Pop. (3 + yrs) who speak an Indigenous language
4	p_0a2	Population aged 0–2
5	p_12a14	Population aged 12–14
6	p_15a17	Population aged 15–17
7	p_18a24	Population aged 18–24
8	p_3a5	Population aged 3–5
9	p_6a11	Population aged 6–11
10	p_60ymas	Population aged 60 +
11	pcon_disc	Population with a disability
12	pclim_pmen	Population with a mental problem/condition
13	pea	Economically active population (12 + yrs)
14	pe_inac	Non-economically active population (12 + yrs)
15	pdesocup	Unemployed population (12 + yrs)
16	phog_ind	Population in Indigenous households
17	pob0_14	Population aged 0–14
18	pob_afro	Population identifying as Afro-Mexican / Afro-descendant
19	pocupada	Employed population (12 + yrs)
20	pnacoe	Population born in another state
21	presoe15	Residents (5 + yrs) living in another state in 2015
22	psinder	Population without health-service affiliation
		Dwellings
23	prom_ocup	Avg. occupants per inhabited private dwelling
24	vph_aguafv	Inhabited private dwellings without piped water
25	vph_ndeaed	Inhabited private dwellings w/o electricity, water or drainage
26	vph_nodren	Inhabited private dwellings without drainage
27	vph_pisoti	Inhabited private dwellings with a dirt floor
28	vph_s_elec	Inhabited private dwellings without electricity
29	vph_sinltc	Inhabited private dwellings w/o landline or cell phone

Enhanced Vegetation Index 2 (EVI2) [92]. EVI2 provides a simplified vegetation index that does not require the blue band, making it useful for vegetation monitoring in urban and rural environments. It is computed as

$$\mathrm{EVI2} = 2.5 \cdot \frac{\mathrm{NIR} - \mathrm{Red}}{\mathrm{NIR} + 2.4 \cdot \mathrm{Red} + 1 + \varepsilon}.$$

Normalized Difference Index 45 (NDI45) [92]. NDI45 helps detect vegetation stress and urban heat islands. It is computed as

$$\mathrm{NDI45} = \frac{\mathrm{Red} - \mathrm{Blue}}{\mathrm{Red} + \mathrm{Blue} + \varepsilon}.$$

Shadow Index (SI) [93]. SI estimates the radiation damping factor caused by shadows in vegetation areas. It is computed as

$$\mathrm{SI} = \frac{\mathrm{Blue} + \mathrm{Green} + \mathrm{Red}}{3}.$$

Theme: Soil exposure and land degradation

Soil Adjusted Vegetation Index (SAVI) [94]. SAVI is helpful in areas with sparse vegetation and bare soil, aiding in land degradation and poverty analysis. It is computed as

$$\text{SAVI} = \left(\frac{\text{NIR} - \text{Red}}{\text{NIR} + \text{Red} + 0.5 + \varepsilon} \right) \cdot 1.5.$$

Modified Soil Adjusted Vegetation Index (MSAVI) [95]. MSAVI minimizes soil brightness effects in areas with sparse vegetation, useful for monitoring agricultural productivity and land use. It is computed as

$$\text{MSAVI} = \frac{2 \cdot \text{NIR} + 1 - \sqrt{(2 \cdot \text{NIR} + 1)^2 - 8 \cdot (\text{NIR} - \text{Red})}}{2}.$$

Theme: Built-up and impervious surfaces

Normalized Difference Built-up Index (NDBI) [96]. NDBI is primarily used for mapping built-up areas, which supports urban expansion studies and the identification of poverty-related infrastructure. It is computed as

$$\text{NDBI} = \frac{\text{SWIR}_1 - \text{NIR}}{\text{SWIR}_1 + \text{NIR} + \varepsilon}.$$

Theme: Water and moisture availability

Normalized Difference Water Index (NDWI) [97]. NDWI is useful in detecting water bodies, assessing water availability, and analyzing potential flood-prone areas in urban and rural landscapes. It is computed as

$$\text{NDWI} = \frac{\text{Green} - \text{NIR}}{\text{Green} + \text{NIR} + \varepsilon}.$$

Theme: Disturbance and burn severity

Normalized Burn Ratio (NBR) [98]. NBR monitors fire-affected areas and vegetation regrowth, which is important for land management and reconstruction planning. It is computed as

$$\text{NBR} = \frac{\text{NIR} - \text{SWIR}_2}{\text{NIR} + \text{SWIR}_2 + \varepsilon}.$$

Author Contributions: For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used "Conceptualization, Salas, Vera, Wood; methodology, Salas; software, Salas, Vera, Zea-Ortiz; validation, Salas, Vera, Zea-Ortiz; formal analysis, Salas, Wood; investigation, Salas, Vera; resources, Salas, Wood; data curation, Salas, Zea-Ortiz; writing—original draft preparation, Salas; writing—review and editing, Salas, Vera, Zea-Ortiz, Wood; visualization, Salas; supervision, Salas, Wood; project administration, Salas, Wood; funding acquisition, Salas, Wood. All authors have read and agreed to the published version of the manuscript.", please turn to the CRediT taxonomy for the term explanation. Authorship must be limited to those who have contributed substantially to the work reported.

Funding: This research was partially funded by SIP-IPN grant number 20250024 for Joaquín Salas and 20250065 for Pablo Vera.

Institutional Review Board Statement: "Not applicable".

Informed Consent Statement: "Not applicable".

Data Availability Statement: We encourage all authors of articles published in MDPI journals to share their research data. In this section, please provide details regarding where data supporting reported results can be found, including links to publicly archived datasets analyzed or generated during the study. Where no new data were created, or where data is unavailable due to privacy or ethical restrictions, a statement is still required. Suggested Data Availability Statements are available in section "MDPI Research Data Policies" at https://www.mdpi.com/ethics. The code can be accessed at https://github.com/joaquinsalas/poverty

Acknowledgments: We would like to express our gratitude to Elio Atenógenes Villaseñor García, Alejandra Figueroa Martínez, and Ranyart Rodrigo Suárez Ponce de León for their support during this research. During the preparation of this manuscript/study, the authors used ChatGPT and Grammarly to translate from Spanish to English and correct English grammar. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. World Bank. Poverty, Prosperity, and Planet Report 2024: Pathways Out of the Polycrisis. Technical report, World Bank, 2024.
2. Pan, Y.; Shi, K.; Zhao, Z.; Li, Y.; Wu, J. The effects of China's poverty eradication program on sustainability and inequality. Humanities and Social Sciences Communications **2024**, 11, 1–15.
3. Buheji, M.; Migdad, M.; Hassoun, A. From Crisis to Catastrophe: Poverty Dynamics in Gaza Before and During the Recent War. In War on Gaza: Consequences on Sustainability and Global Security; Springer, 2025; pp. 15–28.
4. Dunga, H.M. A nexus of access to information and household poverty post Covid-19. International Journal of Business Ecosystem & Strategy (2687-2293) **2025**, 7, 250–259.
5. Archer, E.; Males, J. Advancing action on the UN Sustainable Development Goals, 2024.
6. Sedona, R.; Cavallaro, G.; Riedel, M.; Benediktsson, J.A. Proven Approaches of Using Innovative High-Performance Computing Architectures in Remote Sensing. In Signal and Image Processing for Remote Sensing; CRC Press, 2024; pp. 8–22.
7. Diana, L.; Dini, P. Review on hardware devices and software techniques enabling neural network inference onboard satellites. Remote Sensing **2024**, 16, 3957.
8. Qiao, Y.; Teng, S.; Luo, J.; Sun, P.; Li, F.; Tang, F. On-orbit DNN distributed inference for remote sensing images in satellite Internet of Things. IEEE Internet of Things Journal **2024**.
9. Pobreza Multidimensional 2024. Technical Report Comunicado de prensa 118/25, "Instituto Nacional de Estadística y Geografía", Ciudad de México, México, 2025. Publicado el 13 de agosto de 2025.
10. Reyes, M.; Teruel, G.; Vilar, M.; López, O.; Pérez, V. Official multidimensional measurement of poverty in Mexico: Scope and limitations. Revista mexicana de sociología **2024**, 86, 43–76.
11. CONEVAL. Metodología para la medición multidimensional de la pobreza en México; Coneval, 2016.
12. Serván-Mori, E.; Wirtz, V.J. Monetary and nonmonetary household consumption of health services and the role of insurance benefits: An analysis of the Mexico's National Household Income and Expenditure Survey. The International journal of health planning and management **2018**, 33, 847–859.
13. Instituto Nacional de Estadística y Geografía (INEGI). Encuesta Nacional de Ingresos y Gastos de los Hogares 2022 (ENIGH). Technical report, Instituto Nacional de Estadística y Geografía (INEGI), 2022. Accedido el 8 de marzo de 2024.
14. Alsharkawi, A.; Al-Fetyani, M.; Dawas, M.; Saadeh, H.; Alyaman, M. Poverty classification using machine learning: The case of Jordan. Sustainability **2021**, 13, 1412.
15. Li, Q.; Yu, S.; Échevin, D.; Fan, M. Is poverty predictable with machine learning? A study of DHS data from Kyrgyzstan. Socio-Economic Planning Sciences **2022**, 81, 101195.
16. Corral, P.; Henderson, H.; Segovia, S. Poverty mapping in the age of machine learning. Journal of Development Economics **2025**, 172, 103377.
17. Daoud, A.; Jordan, F.; Sharma, M.; Johansson, F.; Dubhashi, D.; Paul, S.; Banerjee, S. Measuring poverty in India with machine learning and remote sensing. arXiv preprint arXiv:2202.00109 **2021**.
18. Espín-Noboa, L.; Kertész, J.; Karsai, M. Interpreting wealth distribution via poverty map inference using multimodal data. In Proceedings of the Proceedings of the ACM Web Conference 2023, 2023, pp. 4029–4040.
19. Hu, S.; Ge, Y.; Liu, M.; Ren, Z.; Zhang, X. Village-level poverty identification using machine learning, high-resolution images, and geospatial data. International Journal of Applied Earth Observation and Geoinformation **2022**, 107, 102694.
20. Sende, N.B.; Saha, S.; Ruganzu, L.; Kar, S. Prediction of Multidimensional Poverty Status with Machine Learning Classification at Household Level: Empirical Evidence from Tanzania. IEEE Access **2025**.
21. Solís-Salazar, M.; Madrigal-Sanabria, J. A machine learning proposal to predict poverty. Revista Tecnología en Marcha **2022**, 35, 84–94.
22. Kuffer, M.; Proietti, P.; Siragusa, A. Monitoring slums and informal settlements in Europe: Opportunities for geospatial and Earth observation techniques **2023**.
23. Wang, M.; Wang, Y.; Teng, F.; Li, S.; Lin, Y.; Cai, H. China's poverty assessment and analysis under the framework of the UN SDGs based on multisource remote sensing data. Geo-Spatial Information Science **2024**, 27, 111–131.
24. Wang, K.; Zhang, L.; Cai, M.; Liu, L.; Wu, H.; Peng, Z. Measuring urban poverty spatial by remote sensing and social sensing data: A fine-scale empirical study from Zhengzhou. Remote Sensing **2023**, 15, 381.

25. Jean, N.; Burke, M.; Xie, M.; Alampay Davis, W.M.; Lobell, D.B.; Ermon, S. Combining satellite imagery and machine learning to predict poverty. Science **2016**, 353, 790–794.
26. Burke, M.; Driscoll, A.; Lobell, D.B.; Ermon, S. Using satellite imagery to understand and promote sustainable development. Science **2021**, 371, eabe8628.
27. Pandey, S.; Agarwal, T.; Krishnan, N.C. Multi-task deep learning for predicting poverty from satellite images. In Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence, 2018, Vol. 32.
28. Hall, O.; Ohlsson, M.; Rögnvaldsson, T. A review of explainable AI in the satellite data, deep machine learning, and human poverty domain. Patterns **2022**, 3.
29. Huang, L.Y.; Hsiang, S.M.; Gonzalez-Navarro, M. Using satellite imagery and deep learning to evaluate the impact of anti-poverty programs. Technical report, National Bureau of Economic Research, 2021.
30. Yeh, C.; Perez, A.; Driscoll, A.; Azzari, G.; Tang, Z.; Lobell, D.; Ermon, S.; Burke, M. Using publicly available satellite imagery and deep learning to understand economic well-being in Africa. Nature communications **2020**, 11, 2583.
31. Liu, H.; He, X.; Bai, Y.; Liu, X.; Wu, Y.; Zhao, Y.; Yang, H. Nightlight as a proxy of economic indicators: Fine-grained GDP inference around Chinese mainland via attention-augmented CNN from daytime satellite imagery. Remote Sensing **2021**, 13, 2067.
32. Perez, A.; Ganguli, S.; Ermon, S.; Azzari, G.; Burke, M.; Lobell, D. Semi-supervised multitask learning on multispectral satellite images using wasserstein generative adversarial networks (gans) for predicting poverty. arXiv preprint arXiv:1902.11110 **2019**.
33. Putri, S.R.; Wijayanto, A.W.; Pramana, S. Multi-source satellite imagery and point of interest data for poverty mapping in East Java, Indonesia: Machine learning and deep learning approaches. Remote Sensing Applications: Society and Environment **2023**, 29, 100889.
34. Li, G.; Cai, Z.; Qian, Y.; Chen, F. Identifying urban poverty using high-resolution satellite imagery and machine learning approaches: Implications for housing inequality. Land **2021**, 10, 648.
35. Olearo, L.; D'Adda, F.; Messina, E.; Cremaschi, M.; Bandini, S.; Gasparini, F. Facing multidimensional poverty in older adults: An artificial intelligence approach that reveals the variable relevance. Intelligenza Artificiale **2024**, 18, 51–65.
36. Garza-Rodriguez, J.; Ayala-Diaz, G.A.; Coronado-Saucedo, G.G.; Garza-Garza, E.G.; Ovando-Martinez, O. Determinants of poverty in Mexico: A quantile regression analysis. Economies **2021**, 9, 60.
37. Mohamud, J.H.; Gerek, O.N. Poverty level characterization via feature selection and machine learning. In Proceedings of the 2019 27th signal processing and communications applications conference (siu). IEEE, 2019, pp. 1–4.
38. Awais, M.; Naseer, M.; Khan, S.; Anwer, R.M.; Cholakkal, H.; Shah, M.; Yang, M.H.; Khan, F.S. Foundation models defining a new era in vision: a survey and outlook. IEEE Transactions on Pattern Analysis and Machine Intelligence **2025**.
39. Mai, G.; Huang, W.; Sun, J.; Song, S.; Mishra, D.; Liu, N.; Gao, S.; Liu, T.; Cong, G.; Hu, Y.; et al. On the opportunities and challenges of foundation models for geospatial artificial intelligence. arXiv preprint arXiv:2304.06798 **2023**.
40. Xiao, A.; Xuan, W.; Wang, J.; Huang, J.; Tao, D.; Lu, S.; Yokoya, N. Foundation models for remote sensing and Earth observation: A survey. IEEE Geoscience and Remote Sensing Magazine **2025**.
41. Siméoni, O.; Vo, H.V.; Seitzer, M.; Baldassarre, F.; Oquab, M.; Jose, C.; Khalidov, V.; Szafraniec, M.; Yi, S.; Ramamonjisoa, M. DINOv3. arXiv preprint arXiv:2508.10104 **2025**.
42. Brown, C.; Kazmierski, M.; Pasquarella, V.; Rucklidge, W.; Samsikova, M.; Zhang, C.; Shelhamer, E.; Lahera, E.; Wiles, O.; Ilyushchenko, S. AlphaEarth Foundations: An embedding field model for accurate and efficient global mapping from sparse label data. arXiv preprint arXiv:2507.22291 **2025**.
43. Lu, W.; Li, Z.; Xie, Y. High-Resolution Poverty Mapping with Foundation Models: A Cost-effective Approach from Street Views to Satellite Images. In Proceedings of the 2024 IEEE International Conference on Big Data (BigData). IEEE, 2024, pp. 7364–7368.
44. Agarwal, M.; Sun, M.; Kamath, C.; Muslim, A.; Sarker, P.; Paul, J.; Yee, H.; Sieniek, M.; Jablonski, K.; Mayer, Y. General geospatial inference with a population dynamics foundation model. arXiv preprint arXiv:2411.07207 **2024**.
45. Vishwas, B.V.K.; Macharla, S.R. TimesFM: Time Series Forecasting Using Decoder-Only Foundation Model. In Time Series Forecasting Using Generative AI: Leveraging AI for Precision Forecasting; Springer, 2025; pp. 195–210.
46. Instituto Nacional de Estadística y Geografía (INEGI). Metodología de la malla para la publicación de información estadística y geográfica. Technical report, Instituto Nacional de Estadística y Geografía (INEGI), 2023.
47. Consejo Nacional de Evaluación de la Política de Desarrollo Social (CONEVAL). Medición de la Pobreza 2022, 2023. Accedido el 8 de marzo de 2024.
48. Li, J.; Roy, D.P. A global analysis of Sentinel-2A, Sentinel-2B and Landsat-8 data revisit intervals and implications for terrestrial monitoring. Remote Sensing **2017**, 9, 902.
49. Gibb, R. The rHEALPix discrete global grid system. In Proceedings of the IOP Conference Series: Earth and Environmental Science. IOP Publishing, 2016, Vol. 34, p. 012012.
50. Solórzano, J.V.; Mas, J.F.; Gao, Y.; Gallardo-Cruz, J.A. Patrones espaciotemporales de las observaciones de Sentinel-2 a nivel de imagen y píxel sobre el territorio mexicano entre 2015 y 2019. Revista de Teledetección **2020**, 1, 103–115.

51. Phiri, D.; Simwanda, M.; Salekin, S.; Nyirenda, V.R.; Murayama, Y.; Ranagalage, M. Sentinel-2 data for land cover/use mapping: A review. Remote sensing **2020**, 12, 2291.
52. Montero, D.; Aybar, C.; Mahecha, M.; Martinuzzi, F.; Söchting, M.; Wieneke, S. A standardized catalogue of spectral indices to advance the use of remote sensing in Earth system research. Scientific Data **2023**, 10, 197.
53. Bishop, C. Pattern recognition and machine learning. Springer **2006**, 2, 5–43.
54. Staus, L.P.; Komusiewicz, C.; Sommer, F.; Sorge, M. Witty: An efficient solver for computing minimum-size decision trees. In Proceedings of the AAAI Conference on Artificial Intelligence, 2025, Vol. 39, pp. 20584–20591.
55. Mou, X.; Zhang, H.; Arshad, S.H. Generalized Bayesian kernel machine regression. Statistical Methods in Medical Research **2025**, 34, 243–257.
56. Chen, P.Y.; Liu, S. Neural networks. In Introduction to Foundation Models; Springer, 2025; pp. 13–23.
57. Lott, B.; Gallagher, M.A.; Cox, B.A. Generalized Robust Approach to Feature Selection. Available at SSRN 4494520 **2023**.
58. Chicco, D.; Warrens, M.J.; Jurman, G. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. Peerj computer science **2021**, 7, e623.
59. Prince, S. Understanding Deep Learning; MIT press, 2023.
60. Louizos, C.; Welling, M.; Kingma, D. Learning sparse neural networks through L_0 regularization. arXiv:1712.01312 **2017**.
61. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research **2014**, 15, 1929–1958.
62. Li, M.; Soltanolkotabi, M.; Oymak, S. Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. In Proceedings of the International Conference on Artificial Intelligence and Statistics. PMLR, 2020, pp. 4313–4324.
63. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794.
64. Scholkopf, B.; Smola, A.J. Learning with kernels: support vector machines, regularization, optimization, and beyond; MIT press, 2018.
65. Afendras, G.; Markatou, M. Optimality of training/test size and resampling effectiveness in cross-validation. Journal of Statistical Planning and Inference **2019**, 199, 286–301.
66. Smith, L.N.; Topin, N. Super-convergence: Very fast training of neural networks using large learning rates. In Proceedings of the Artificial intelligence and machine learning for multi-domain operations applications. SPIE, 2019, Vol. 11006, pp. 369–386.
67. Liu, Z. Super convergence cosine annealing with warm-up learning rate. In Proceedings of the CAIBDA 2022; 2nd International Conference on Artificial Intelligence, Big Data and Algorithms. VDE, 2022, pp. 1–7.
68. Wang, N.; Cui, Z.; Li, A.; Lu, Y.; Wang, R.; Nie, F. Structured doubly stochastic graph-based clustering. IEEE Transactions on Neural Networks and Learning Systems **2025**.
69. Wang, N.; Cui, Z.; Li, A.; Xue, Y.; Wang, R.; Nie, F. Multi-order graph based clustering via dynamical low rank tensor approximation. Neurocomputing **2025**, p. 130571.
70. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.
71. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International conference on machine learning. PMLR, 2019, pp. 6105–6114.
72. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
73. Mazzia, V.; Salvetti, F.; Chiaberge, M. Efficient-capsnet: Capsule network with self-attention routing. Scientific reports **2021**, 11, 14634.
74. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 **2020**.
75. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 10012–10022.
76. Instituto Nacional de Estadística y Geografía (INEGI). Censo de Población y Vivienda 2010, 2010. Accessed: 2024-04-16.
77. Instituto Nacional de Estadística y Geografía (INEGI). Censo de Población y Vivienda 2020, 2020. Accessed: 2024-04-16.
78. Smith, J.T.; Warrington, A.; Linderman, S.W. Simplified state space layers for sequence modeling. arXiv:2208.04933 **2022**.
79. Smith, L.N. Cyclical learning rates for training neural networks. In Proceedings of the 2017 IEEE winter conference on applications of computer vision (WACV). IEEE, 2017, pp. 464–472.
80. Lundberg, S.; Erion, G.; Chen, H.; DeGrave, A.; Prutkin, J.; Nair, B.; Katz, R.; Himmelfarb, J.; Bansal, N.; Lee, S.I. From local explanations to global understanding with explainable AI for trees. Nature Machine Intelligence **2020**, 2, 2522–5839.

81. Bouza, L.; Bugeau, A.; Lannelongue, L. How to estimate carbon footprint when training deep learning models? A guide and review. Environmental Research Communications **2023**, 5, 115014.
82. Reid, J.; Zeng, C.; Wood, D. Combining social, environmental and design models to support the sustainable development goals. In Proceedings of the IEEE aerospace conference. IEEE, 2019, pp. 1–13.
83. Różycki, R.; Solarska, D.A.; Waligóra, G. Energy-Aware Machine Learning Models—A Review of Recent Techniques and Perspectives. Energies **2025**, 18, 2810.
84. Maslej, N.; Fattorini, L.; Perrault, R.; Gil, Y.; Parli, V.; Kariuki, N.; Capstick, E.; Reuel, A.; Brynjolfsson, E.; Etchemendy, J. Artificial intelligence index report 2025. arXiv:2504.07139 **2025**.
85. Strubell, E.; Ganesh, A.; McCallum, A. Energy and policy considerations for modern deep learning research. In Proceedings of the AAAI conference on artificial intelligence, 2020, Vol. 34, pp. 13693–13696.
86. Verdecchia, R.; Sallou, J.; Cruz, L. A systematic review of Green AI. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **2023**, 13, e1507.
87. for Blind Review, A. Hugging Carbon: Quantifying the Training Carbon Emissions of AI Models at Scale. In Proceedings of the ICLR 2026 (under review), 2025. Under review at ICLR 2026.
88. Gowda, S.N.; Hao, X.; Li, G.; Gowda, S.N.; Jin, X.; Sevilla-Lara, L. Watt for what: Rethinking deep learning's energy-performance relationship. In Proceedings of the European Conference on Computer Vision. Springer, 2024, pp. 388–405.
89. Tang, B.; Liu, Y.; Matteson, D.S. Predicting poverty with vegetation index. Applied Economic Perspectives and Policy **2022**, 44, 930–945.
90. Rouse, J.W.; Haas, R.H.; Schell, J.A.; Deering, D.W.; et al. Monitoring vegetation systems in the Great Plains with ERTS. NASA Spec. Publ **1974**, 351, 309.
91. Huete, A.; Didan, K.; Miura, T.; Rodriguez, E.P.; Gao, X.; Ferreira, L.G. Overview of the radiometric and biophysical performance of the MODIS vegetation indices. Remote sensing of environment **2002**, 83, 195–213.
92. Jiang, Z.; Huete, A.R.; Didan, K.; Miura, T. Development of a two-band enhanced vegetation index without a blue band. Remote sensing of Environment **2008**, 112, 3833–3845.
93. Ono, A.; Kajiwara, K.; Honda, Y.; et al. Development of new vegetation indexes, shadow index (SI) and water stress trend (WST). Intern. Arch. Photogramm. Remote Sens. Spat. Inf. Sci **2010**, 38, 710–714.
94. Huete, A.R. A soil-adjusted vegetation index (SAVI). Remote sensing of environment **1988**, 25, 295–309.
95. Qi, J.; Chehbouni, A.; Huete, A.R.; Kerr, Y.H.; Sorooshian, S. A modified soil adjusted vegetation index. Remote sensing of environment **1994**, 48, 119–126.
96. Zha, Y.; Gao, J.; Ni, S. Use of normalized difference built-up index in automatically mapping urban areas from TM imagery. International journal of remote sensing **2003**, 24, 583–594.
97. Gao, B.C. NDWI—A normalized difference water index for remote sensing of vegetation liquid water from space. Remote sensing of environment **1996**, 58, 257–266.
98. Key, C.H.; Benson, N.C. The Normalized Burn Ratio (NBR): A Landsat TM radiometric measure of burn severity. United States Geological Survey, Northern Rocky Mountain Science Center: Bozeman, MT, USA **1999**.