A Semi-Supervised Active Learning Approach for Block-Status Classification.

Atul Rawal, James McCoy, Andrew Duvall, and Elvis Martinez.

U.S Census Bureau

Abstract

The Census Bureau, as a part of its decennial census must maintain and update all the addresses present within the United States and its territories. These addresses help formulate policies and allocate valuable resources from the federal government. For the 2020 Census, in-office staff manually canvassed address coverage in every block. While this process was effective, it also brought about challenges associated with cost and time. To help aide the Census Bureau in labelling and classifying blocks, we have proposed a machine learning approach via semi-supervised learning. We present a robust machine learning solution to improve both data labeling and classification of parcel data to enable new data-driven insight while reducing costs and effort for data assessment. Towards this goal, we have employed an active-learning scheme to make accurate and precise classifications using the <1% (~50,000) labelled blocks out of the 8,000,000+ blocks within the country. We utilized multiple machine learning models including Logistic Regression, Random Forest, Gradient Boosting, Extreme Gradient Boosting, Light Gradient Boosting, and Categorical Boosting to make predictions on unlabeled data by training the model on the smaller set of labelled data. Predictions from all the models are then compared to pinpoint the blocks where there is a mismatch between the different models. These blocks are then forwarded to the human labelers to make a final prediction. Once the subset of predicted data has been validated by human labelers, it is then added to the training data before making predictions on the next subset of the data. We also discuss the different challenges associated with working on real-world data at this scale such as class-imbalance and data completeness, integrity.

Introduction

The U.S. Census Bureau is the largest statistical organization in the nation. It collects data on almost every aspect of the economy, demography, and geography. This data is provided by the bureau to provide a snapshot of the country's population size and growth along with the characteristics of the economy. It also performs vital statistical analysis that informs an accurate portrait of the population. All this information is collected through numerous surveys such as the American Community Survey (ACS), the economic census, and the decennial census. The decennial census, as the name suggests is performed every ten years ending in zero to count each resident of the country and where they live on April 1. This is mandated by the US Constitution to determine the apportion of the House of Representatives among the states. It also provides valuable data for a number of other applications such as informing policies for the distribution of hundreds of billions of dollars in federal funds for state and local governments.

As a part of the decennial census, each address within the US and its territories must be maintained and updated. In the past, in-field staff canvassed all the addresses by going door to door throughout the entire nation. For the 2020 census, in-office staff were able to manually canvass all the addresses using computational tools. In-office staff were able to validate approximately 65% of addresses while in-field staff were able to canvass the rest. While both of these approaches have been effective, there were major challenges associated with cost and time. The cost of hiring thousands of staff for both in-field and in-office canvasing, along with the cost of maintaining the resources for the staff was too ineffective. An estimated 800,000 hours were

spent manually canvassing the addresses by in-office staff. To help aide the Census Bureau in labelling and classifying blocks, we have proposed a machine learning approach via semi-supervised learning to alleviate the burden of manual canvassing.

Artificial intelligence (AI) and Machine Learning (ML) have made tremendous progress in the past decade. AI/ML systems have been used for a plethora of applications ranging from fraud detection to tumor detection. Here we propose the use of ML techniques to classify the block status utilizing administrative data helping alleviate the burden of manual canvassing each address. The process itself is a human-in-the-loop process as we believe the presence of an SME within the process provides more efficiency and potential for bias identification and mitigation. Here we highlight a semi-supervised active learning approach where we use a very small subset of data that is labeled to make robust predictions on the larger unlabeled portion of the data. Using an active learning approach with the human labelers in the loop we ensure that the predictions made by the ML classifiers are accurate and precise.

Overview of Machine Learning/XAI

Machine learning is a subset of AI that allows computational models to learn from training data and make predictions on unseen data without being explicitly programmed. ML models are capable of detecting and deriving patterns from the data to make truly data-driven predictions. These algorithms learn through the data rather than the provided algorithmic instruction based on IF-THEN structures. ML models enable an automated problem-solving process with limited or no-human input, solely based on the observational training data.

While artificial intelligence and machine learning are often used interchangeably, they are two different concepts. AI is the broader concept – machines making decisions, learning new skills, and solving problems in a similar way to humans – whereas machine learning is a subset of AI that enables intelligent systems to autonomously learn new things from data. Machine learning models fall into three primary categories.

Supervised machine learning

Supervised learning, also known as supervised machine learning, is defined by its use of labeled datasets to train algorithms to classify data or predict outcomes accurately. As input data is fed into the model, the model adjusts its weights until it has been fitted appropriately. This occurs as part of the cross-validation process to ensure that the model avoids overfitting or underfitting. Supervised learning helps organizations solve a variety of real-world problems at scale, such as classifying spam in a separate folder from your inbox. Some methods used in supervised learning include neural networks, naïve bayes, linear regression, logistic regression, random forest, and support vector machine (SVM). [1]

Unsupervised machine learning

Unsupervised learning, also known as unsupervised machine learning, uses machine learning algorithms to analyze and cluster unlabeled datasets (subsets called clusters). These algorithms discover hidden patterns or data groupings without the need for human intervention. This method's ability to discover similarities and differences in information make it ideal for exploratory data analysis, cross-selling strategies, customer segmentation, and image and pattern recognition. It's also used to reduce the number of features in a model through the process of dimensionality reduction. Principal component analysis (PCA) and singular value decomposition (SVD) are two

common approaches for this. Other algorithms used in unsupervised learning include neural networks, k-means clustering, and probabilistic clustering methods. [1]

Semi-supervised machine learning

Semi-supervised learning offers a happy medium between supervised and unsupervised learning. During training, it uses a smaller labeled data set to guide classification and feature extraction from a larger, unlabeled data set. Semi-supervised learning can solve the problem of not having enough labeled data for a supervised learning algorithm. It also helps if it's too costly to label enough data. [2]

Explainable AI

The increasing sophistication and number of AI/ML applications has given a rise to new challenges associated with trust and fairness. One of these challenges is the black-box nature of AI/ML systems, making it difficult for end-users to trust the predictions/decisions made by these systems. Even though explanation systems for AI have been around for some time, the lack of explanations and consequently, trust for AI/ML systems predictions still hinders the growth of these systems [3]. To this end, XAI proposes a new research direction to interpret and explains the behavior/predictions of AI/ML systems to the end-users [3]. The U.S Defense Advanced Research Projects Agency started the XAI program in 2017 to further the research on explanations of AI/ML systems [4]. It defines XAI systems as: *AI systems that can explain their rationale to a human user, characterize their strengths and weakness, and convey an understanding of how they will behave in the future*. The program highlights explainability as a vital component for trustworthy AI/ML systems.

In more layman terms, XAI is an AI/ML system that includes detailed instructions about how it works. Not just the algorithms themselves, but the actual detailed steps of the system must be understandable by the end user who will be impacted by the model itself. The end user must understand why and how the system came to an end result, why not a different result. The end user must be able to catch when the results are inaccurate and why they are inaccurate, allowing them to decide when to trust an AI/ML system and when not to. Explainability must be viewed as an essential pillar of accountable AI where explainable systems can aid in identifying and mitigating bias from real world data [3].

Explainability can be achieved via two different methods: transparent models, or post-hoc explanations. For transparent models, there are three levels of transparency: *Simulatability, Decomposability, and Algorithmic transparency*. Whereas post-hoc explanations can be generated via a variety of methods, such as *visual explanations, text explanations, explanations by example, explanation by knowledge, rule-based explanation, global and local explanations, and feature relevance* [3]. Since AI/ML models are often black-box systems, model transparency isn't always feasible. Therefore, post-hoc explanations are more commonly utilized [3]. Open-source XAI libraries and models for post-hoc explanations of AI/ML models have been presented in the literature, such as the ones from Lundberg et al., and Ribeiro et al [5, 6]. Examples include LIME, SHAP, DeepLIFT, and DeepSHAP [5-8].[1]

[1] For an in-depth review on XAI readers are encouraged to read the survey paper " Recent Advances in Trustworthy Explainable Artificial Intelligence: Status, Challenges and Perspectives" 3. Rawal, A., et al., *Recent advances in trustworthy explainable artificial intelligence: Status, challenges, and perspectives.* IEEE Transactions on Artificial Intelligence, 2021. **3**(6): p. 852-866..

For this study, SHAP was utilized for generating feature relevance and beeswarm plots. SHapley Additive exPlanations (SHAP) presented by Lundberg, et al., is an open source post-hoc XAI framework for interpreting predictions. It generates Shapley values to calculate the feature relevance score of the input variables for AI/ML models. These scores can be compared and ranked to gain insight into the impact each feature had on the model's decision/prediction. It provides additive feature importance values to generate insights into the relevance of each feature for the models predictions [6].

Overview of Block Status Classification

The Census Bureau manually evaluated coverage of living quarters in each tabulation block leading into the operations for the 2020 Decennial Census. These coverage estimations were made in an office reviewing imagery and flagging levels of activity for where updates were needed and indicating which blocks were passive and did not need an update. The process is time consuming and expensive. The evaluation is also only relevant for a moment of time as construction is pervasive in many areas of the country.

The 2030 Decennial Census is looking to new innovations with Machine Learning to provide more frequent assessments on block coverage that can be run on nationwide as updates are made biannually to reflect the new changes made to the Master Address File (MAF). This approach requires training a model using staff to provide training data based on assessments of block coverage on a smaller scale than the 8.2 million blocks that were drawn from the 2020 Census. Their assessments are made from data sources that are updated with each benchmark, so this data contains the benchmark that the assessment is relevant for.

Data Sources

The main data-source for this project is the Census Bureau's Intelligence Database (ID). It contains data from multiple sources including the MAF, DSF, Parcels, ACDC and TIGER [*Figure 1*]. It stores metrics from a variety of sources at the block-level to provide greater context to the state of the Master Address File (MAF) as it corresponds to an always-changing built environment. Another important aspect of the ID is that it is populated with a static copy of the MAF created approximately every six months after the MAF/TIGER benchmarking cycle. Given that MAF/TIGER system can be changed in real-time between those benchmarks, the data in the ID does not reflect the current state of the database. Any updates that occur between benchmarks are reflected on the following version of the ID.

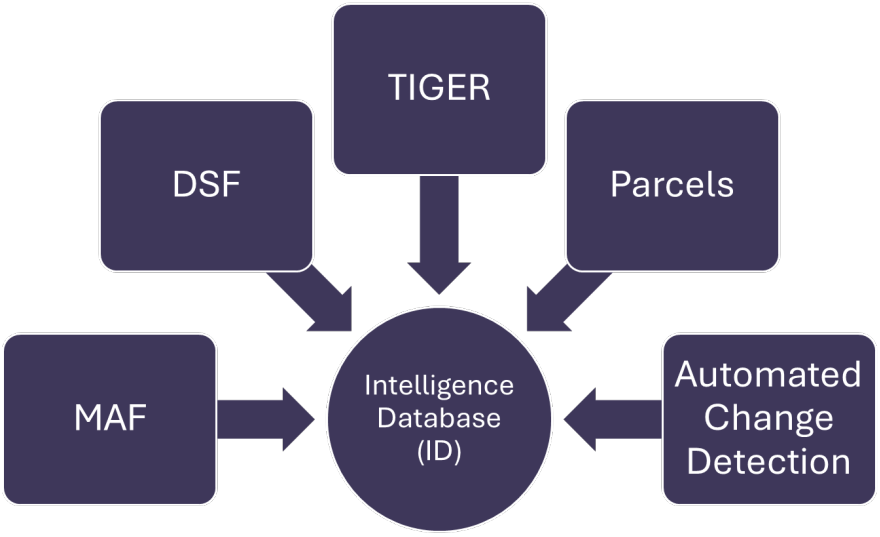

Figure 1: Data sources for the Intelligence Database (ID).

The block data is the lowest level of census geography which is aggregated to the tract, county and state levels. Blocks vary in size, which is crucial for analysis. The MAF data contained in the ID provides a general overview of the state of the MAF addresses and geocodes in a particular block. It contains millions of addresses, but only preferred ones are displayed in the ID. The rest of the records are filtered via the Census Filter which is designed to support the Decennial Census and the American Community Survey (ACS) filter, which is designed to serve the mid-decade ACS operations. The Delivery Sequence File (DSF) data is a product of the United States Postal Service and contains addresses and locations, residential/commercial status and whether it is receiving mail or not. MSP data contains spatial representations of structures that have one or more MAF Units. It represents specific structures and not the MAF Units within that structure. The parcel data used in the ID is provided by third-party organization. The data was pre-processed and normalized before being matched with the MAF. Because parcels are consolidation of multiple locality data, the quality varies from entity to entity. The TIGER data consists of road features that have both street names and address ranges that can be compared to the MAF Units in each block. The Automated Change Detection (ACD) data within the ID comprises of polygons of areas with detected changes. It uses ESRI's Land Use/Land Cover (LULC) product with changes from 2021 and 2022. The change polygons were created by classifying the land use throughout the world (vegetation, bare surface, water, cropland, built area, etc.) and then monitoring for change in those classifications (i.e., cropland that is now a built area).

The dataset contains over 8 million blocks with 92 columns of data from the different sources mentioned above. Out of the 8 million blocks from the entire U.S. approximately 40,000 have been manually labeled by staff, with the rest of the blocks remaining unlabeled. For each block the labels are split into: *Passive* (no change), *Over-coverage* (negative change) *or Under-coverage* (positive change). The labelled dataset is unbalanced and presents a challenge for making bias-free predictions [*Figure 2*]. To address this issue, we utilized both over-sampling and under-sampling approaches to compare the results and move forward with the best performing model.

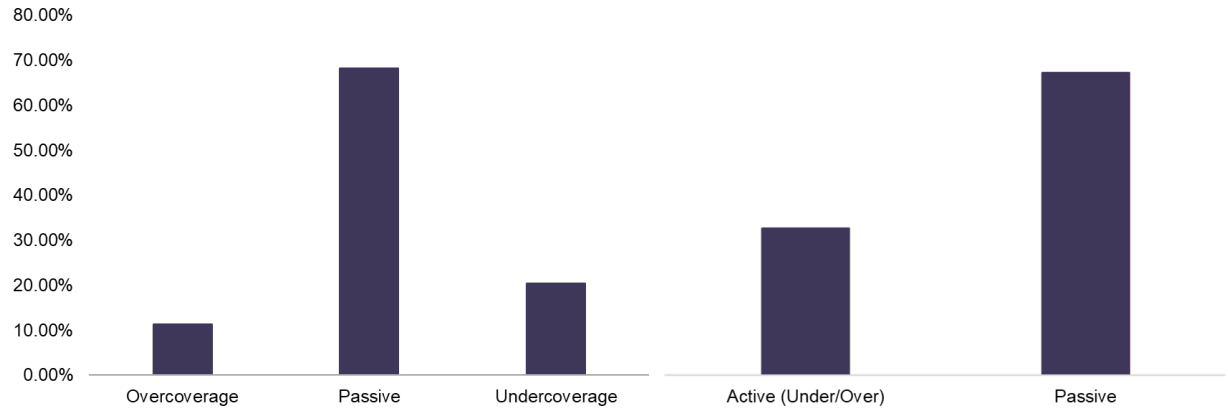

Figure 2: Class imbalance for multiclass (left) and binary class (right) models.

Another method to address the data imbalance taken for this project was to combine the two minority classes into one larger minority class. So, we combined the *under-coverage* and *over-coverage* classes into a larger minority class of *active* blocks [*Figure 4*]. Multiple different ML predictions were performed to pick the best performing model for further analysis.

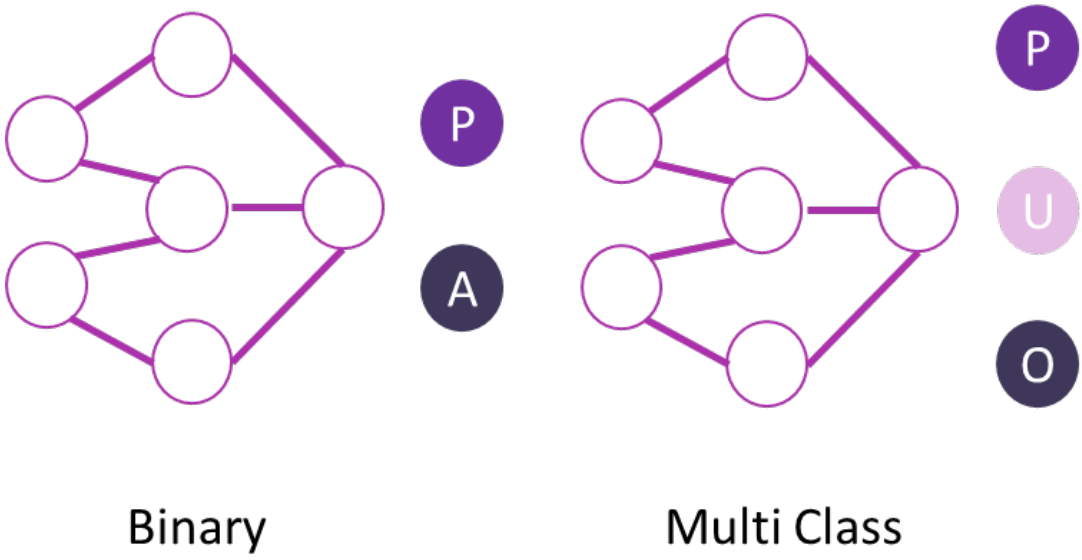

Figure 3: Binary (left) and multi-class (right) models for the project.

Methods

To address the issues of data-imbalance we utilized both under-sampling and over-sampling to compare the performance from the two techniques. For under-sampling we used the random under-sampling to bring the majority class in balance with the minority class. The random undersampler package from imblearn was imported into the python notebook. For over-sampling we utilized the k-means smote over-sampling technique to bring the minority class in balance with the majority class. The k-means SMOTE package from imblearn was imported into the notebook. We performed under and over sampling for both the minority and majority class models to compare the performance.

For this project we've taken a semi-supervised active learning approach as highlighted in [*Figure 4*]. Here we aim to take the data that has been manually labeled by human labelers and use that for the supervised and semi-supervised learning models. For the supervised learning model, we train and validate using the same data. Whereas, for the semi-supervise learning model we train using the entire labelled dataset and make predictions on batches (100K) of unlabeled dataset. With the active learning scheme, we take the data that has already been labeled for training our ML classifiers. Predictions for unlabeled data are then made using the trained ML classifiers. Block predictions with a low class-confidence or difference between the classifiers, are then sent back to human labelers for validation. Once validated, the predicted blocks are then added to the training data to make predictions on the next batch of unlabeled data. This cycle is repeated until we have predicted labels for all 8 million blocks.

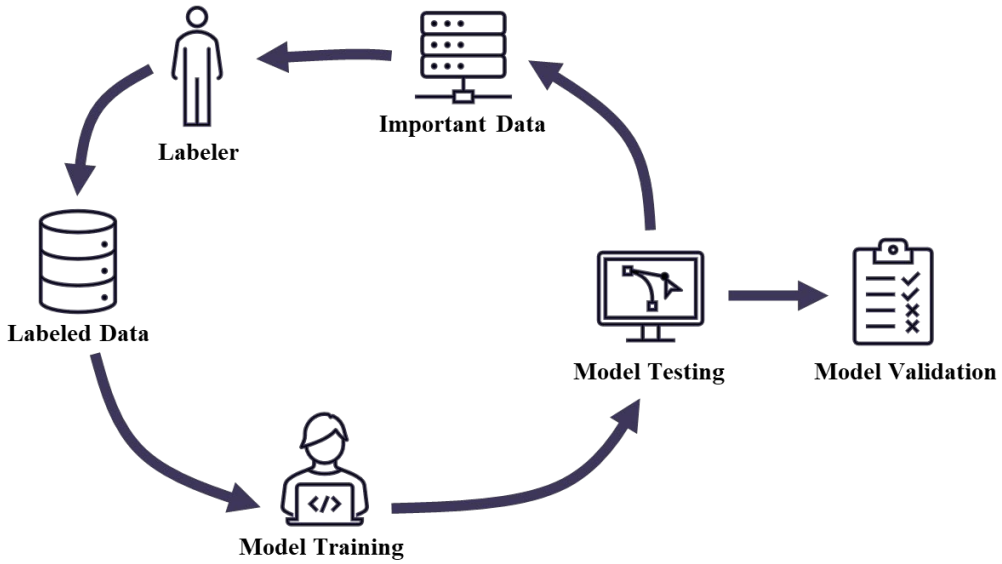

Figure 4: Active learning scheme for the project.

Upon completion of dimensionality reduction by filtering down the datasets for the respective models, pandas data frame library was utilized to import the data into a python notebook. For the supervised learning model an 80%-20% split was used for training and testing the ML classifiers with 80% of the dataset used for training the classifier and 20% used for validation. For the semi-supervised model, we trained on the entire labelled dataset and made predictions for batches of unlabeled data. Six different ML classifiers were utilized for the active learning scheme to ensure uniformity of predictions. For machine learning, we utilized both binary and multi-class classification models to classify the bock status. For the binary class the labels were passive (no change) or active (change), while the multi-class labels were passive (no change), over-coverage (negative change) or under-coverage (positive change).

Different ML libraries were imported into the notebook, including Scikit-learn (sklearn), pandas, NumPy, Keras, CausalML, and SHAP to build the classifier models. The models were then trained and validated on the dataset to identify the top performing classifiers. The ML classifiers included Logistic Regression (LR), Random Forest (RF), Gradient Boosting (GB), Light Gradient Boosting (LGB), Extreme Gradient Boosting (XGB), and CatBoost (CB) classifiers. LR, RF, and GB classifiers were imported from the sklearn package. LGB, XGB and CB classifiers were

imported from LightGBM, XGBoost and CatBoost packages respectively. Once the training and validation was completed the models were evaluated using performance metrics of accuracy, precision, recall and F1-score for the supervised learning model.

- Accuracy - The number of correct predictions over all predictions.

$$\text{Accuracy} = \frac{True\ Positve + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative}$$

- Precision – Measure of correct positive predictions made.

$$\text{Precision} = \frac{True\ Positve}{True\ Positive + False\ Positive}$$

- Recall - Measure correct positive predictions over all the positive cases in the data.

$$\text{Recall} = \frac{True\ Positve}{True\ Positive + False\ Negative}$$

- F1-Score - Combination of precision and recall. Is a value between 0 and 1, where 0 is the worst score and 1 signifies correct predictions for each observation.

$$\text{F1 Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

For the semi-supervised learning model, we calculated the cross-validation accuracy for the training data. Once the training and testing is completed, model predictions for each batch are compared and blocks with a large difference in class confidence or difference in the predicted class are then chosen as important blocks that need to be validated by a human labeler. These blocks are then exported into a data frame and sent for human labeling/validation. We chose the RF, GBR, XGB and LGB classifiers as our main classifiers to focus on the active learning portion. Blocks that had uniform predictions across all four classifiers are labeled accordingly. However, blocks that had a large difference in class confidence of $\geq$0.10 between more than 2 classifiers, were sent to human labelers for validation. Once the labels have been validated/edited by human labelers they are added back onto the training dataset to train the classifiers and make predictions on the next batch of unlabeled blocks.

The best performing classifier from the six classifiers is chosen for further analysis via the application of explainability. Explainability in this study is achieved via feature relevance with both global and local explanations to identify the important features for block classification. As mentioned previously, SHAP is an open-source post-hoc XAI library for explaining ML classifiers via feature relevance and Shapley values. For the current study, a linear explainer was used to explain the logistic regression classifier and a tree explainer was used for the RF classifier and kernel explainer is used for the boosting classifiers.

To address the issues of data-imbalance we utilized both under-sampling and over-sampling to compare the performance from the two techniques. For under-sampling we used the random under-sampling to bring the majority class in balance with the minority class. For over-sampling we utilized the k-means SMOTE (Synthetic Minority Oversampling Technique) over-sampling technique to bring the minority class in balance wit the majority class. We performed under and over sampling for both the multiclass and binary models to compare the performance.

Results & Discussion

Using oversampling and under-sampling we achieved a balanced dataset for both the multiclass and binary models. *Figure 5* and *Figure 6* present the results for the class balancing with both under-sampling and oversampling methods for the multiclass and binary models.

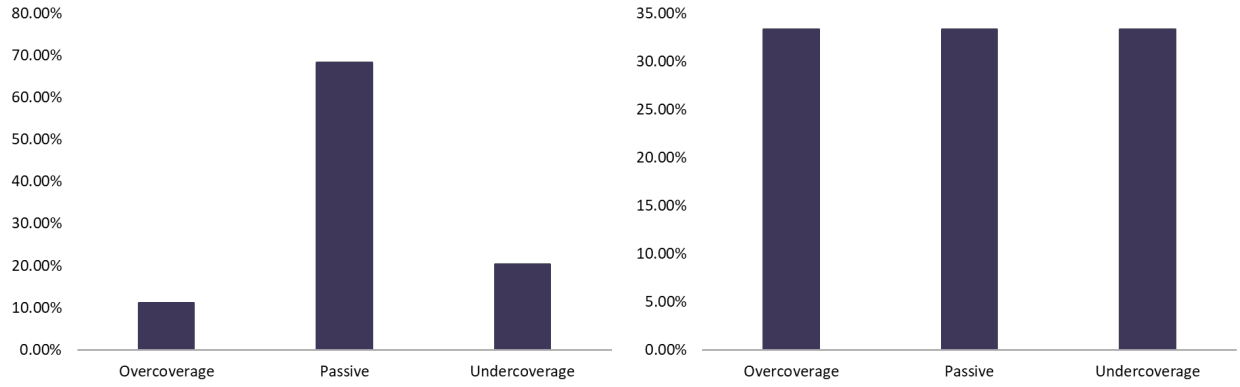

Figure 5: Class balancing for multiclass models using both under-sampling and over-sampling techniques.

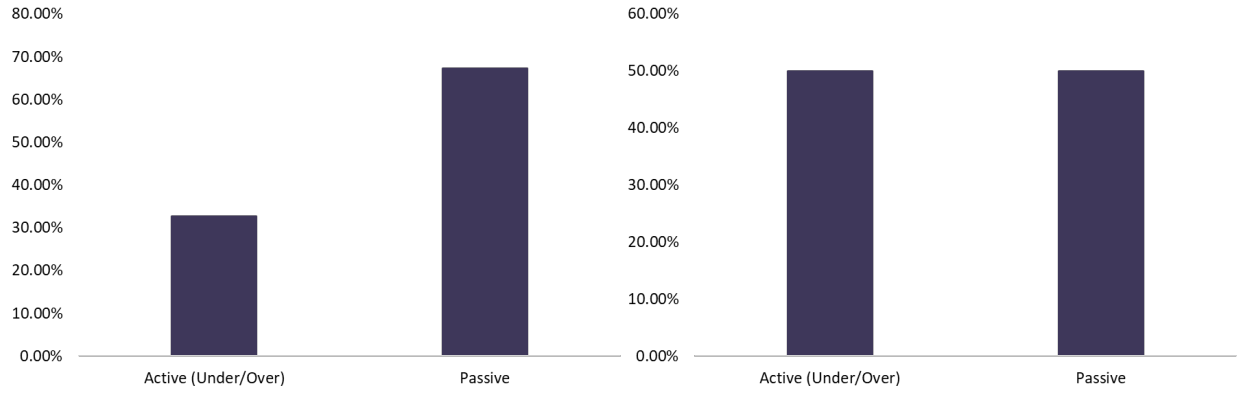

Figure 6: Class balancing for binary class models using both under-sampling and over-sampling techniques.

Supervised Learning

For the supervised learning model with training and validating on the labeled data with an 80%-20% split of the 40K labeled blocks we achieve great performance across all classifiers with some achieving a 0.99 F1 score for both the multi-class and binary class models. *Table 1* and *Figure 7* highlight the performance of all the classifiers for both the multiclass and binary models.

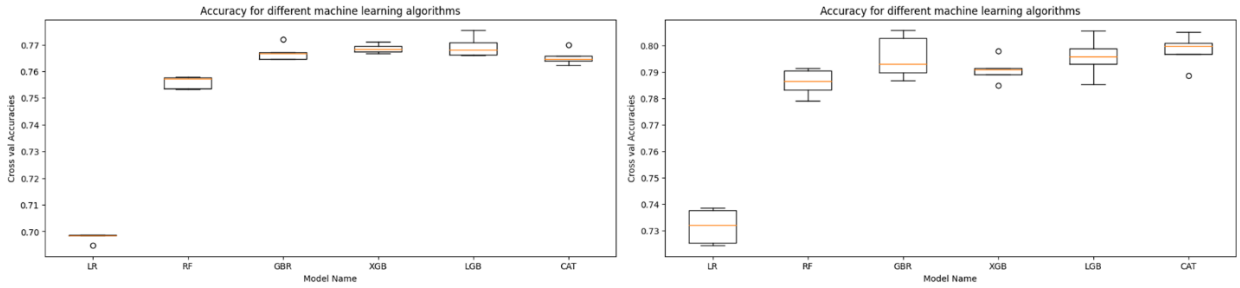

Figure 7: Cross validation accuracy for all multi-class (left) and binary (right) classification models for supervised learning.

Table 1: Performance metrices for both multiclass and binary classification models for supervised learning models.

Model	Accuracy		Precision		Recall		F1-Score	
	Multi	Binary	Multi	Binary	Multi	Binary	Multi	Binary
Logistic Regression	0.6978	0.7316	0.64	0.71	0.70	0.73	0.64	0.71
Light Gradient Boosting	0.7693	0.7958	0.82	0.83	0.83	0.83	0.81	0.83
Gradient Boosting	0.7670	0.7956	0.85	0.85	0.85	0.85	0.84	0.85
Cat Boosting	0.7653	0.7983	0.94	0.92	0.95	0.92	0.94	0.92
Extreme Gradient Boosting	0.7686	0.7909	0.98	0.97	0.98	0.97	0.98	0.97
Random Forest	0.7559	0.7862	0.99	0.99	0.99	0.99	0.99	0.99

The performance for the both the multi-class and the binary class models are comparable with no significant difference to each other with minor differences mainly in the cross-validation accuracy. Binary class models had the same performance for precision, recall and F1-score for the RF classifier. A slight (non-significant) increase in performance across all metrics is observed for the LR, LGBM, and GBR classifiers whereas a decrease is observed for the XGB classifier.

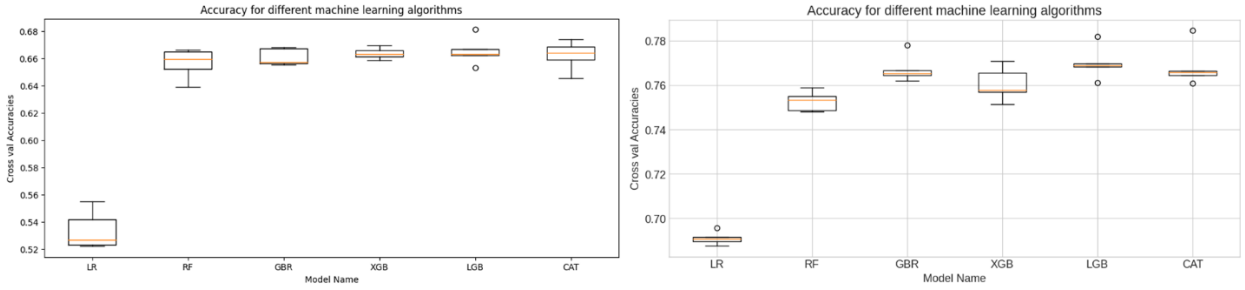

Figure 8: Cross-validation accuracy for all the ML models for supervised learning with under-sampling.

Table 2: Performance metrices for all the ML models for supervised learning with under-sampling.

Model	Accuracy		Precision		Recall		F1-Score	
	Multi	Binary	Multi	Binary	Multi	Binary	Multi	Binary
Logistic Regression	0.5336	0.6909	0.53	0.68	0.51	0.64	0.32	0.62
Light Gradient Boosting	0.6652	0.7700	0.76	0.81	0.67	0.78	0.67	0.78
Gradient Boosting	0.6609	0.7673	0.80	0.83	0.73	0.81	0.73	0.80
Cat Boosting	0.6622	0.7685	0.92	0.90	0.89	0.88	0.89	0.88
Extreme Gradient Boosting	0.6636	0.7606	0.97	0.96	0.96	0.96	0.96	0.96
Random Forest	0.6564	0.7685	0.99	0.99	0.99	0.99	0.99	0.99

Figure 8 and Table 2 highlight the performance of all the classifiers for models with data under-sampling. For both multi-class and binary class, once again we did not observe any significant differences between the performance of the classifiers other than the LR classifier which had a significant increase in performance with the binary model. A decrease in performance is observed between the imbalanced and the under-sampling models for most of the classifiers except for the

RF classifier which yielded similar performance with only a slight decrease in the cross-validation accuracy. The decrease in performance is expected with the random under-sampling method as the models are not able to generalize well over the undersampled data. Even though the RF under-sampling classifiers yield similar performance to the imbalanced data, the decrease in the performance across the other classifiers highlights the non-feasibility of the under-sampling models when scaling to the continuous US. When running the models for the continuous US, a potential bias is introduced when they are not able to generalize well over the entire data. To avoid the introduction of this potential selection bias, we did not utilize the under-sampling models going forward.

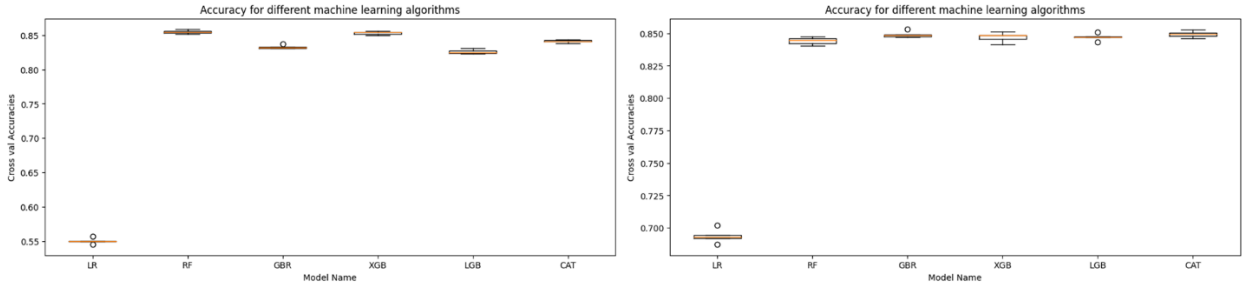

Figure 9: Cross-validation accuracy for all the ML models for supervised learning with over-sampling.

Table 3: Performance metrices for all the ML models for supervised learning with over-sampling.

Model	Accuracy		Precision		Recall		F1-Score	
	Multi	Binary	Multi	Binary	Multi	Binary	Multi	Binary
Logistic Regression	0.5501	0.6937	0.64	0.71	0.64	0.73	0.64	0.71
Light Gradient Boosting	0.8261	0.8475	0.75	0.83	0.68	0.83	0.67	0.83
Gradient Boosting	0.8329	0.8490	0.75	0.85	0.67	0.85	0.67	0.85
Cat Boosting	0.8411	0.8493	0.85	0.92	0.79	0.92	0.79	0.92
Extreme Gradient Boosting	0.8534	0.8470	0.98	0.97	0.98	0.96	0.98	0.96
Random Forest	0.8547	0.8442	0.98	0.99	0.99	0.99	0.98	0.99

Finally *Figure 9 and Table 3* display the performance results for the k-means SMOTE over-sampling classifiers. Once again, no significant difference is observed between the multi-class and binary results. All the classifiers yielded a decrease in performance for the multi-class compared to the binary class with the LGBM classifier resulting in the largest decrease between the two. However, the XGB and RF classifiers performed similarly for both the classes. When compared to the imbalanced classifiers the binary class SMOTE classifiers had similar results. However, the multi-class classifiers yielded a decreased performance as the classifiers. This can be attributed to the possible over-fitting on the imbalanced data, resulting in a decreased performance when the data is balanced, and the predictions are no longer biased towards the majority class. With the RF and XGB classifiers for both the binary and multi-class classifiers yielding excellent performance we chose the multi-class SMOTE classifiers for further utilization for the XAI and semi-supervised portion of the project. Once the predictions are made for the supervised learning portion, we export them to be integrated into the ID dashboard where block status can be displayed for the predicted blocks along with the actual status. *Figure 10* highlights the predictions made by the model for

the block status when applied to the ID dashboard. Here the end user can click on the block of interest and display the predicted and the actual status of the block.

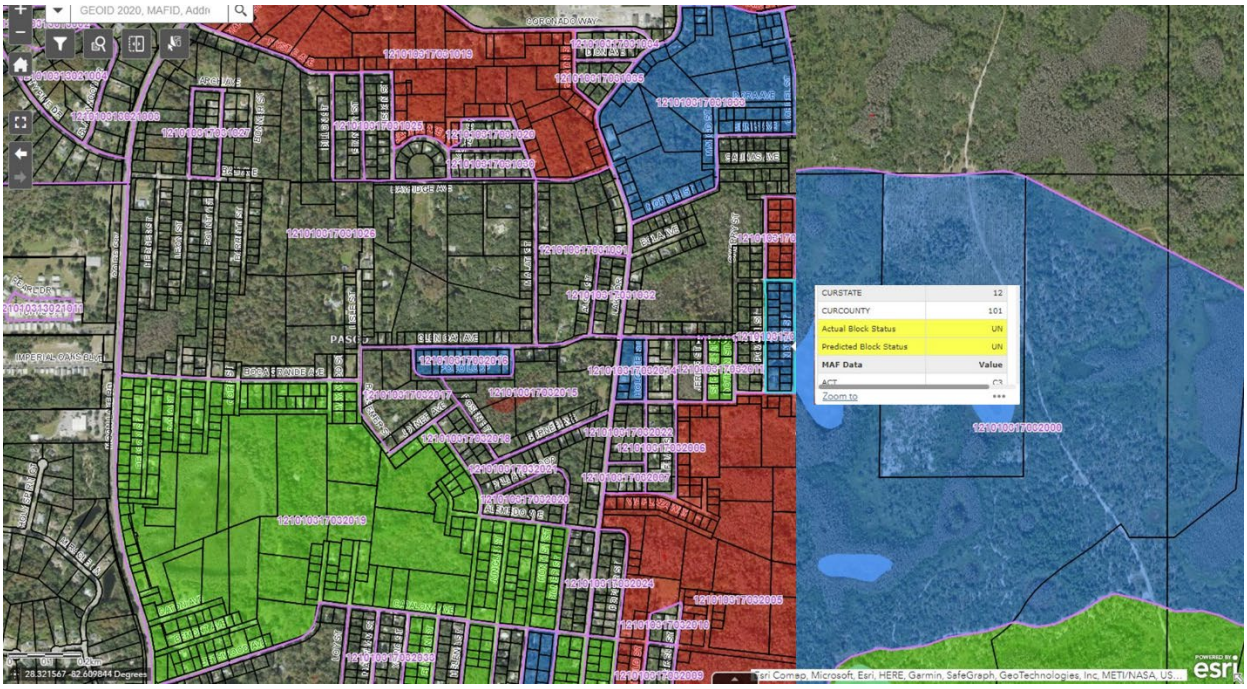

Figure 10: Example of the ID dashboard with prediction results for various blocks with actual and the predicted block status highlighted in the info tab.

XAI via SHAP is performed on the multi-class RF classifier to provide insights into both the global and local predictions. At the global scale feature relevance and beeswarm plots are generated for all the blocks included in training data, whereas at the local scale feature relevance plots are generated for each individual blocks. Feature relevance plots for each class are also generated to highlight the different features that have an impact on each class. We used SHAP to identify and rank the variables associated with the classification of the block status. SHAP allows for the calculation of SHAP values which represent the impact each variable has on the prediction made by the model. The SHAP value plot shows the relationship between the predictor features (variables) and the target classification (block status). The SHAP value plot depicts the effect in for positive and negative directions, i.e., it shows how much of an impact, both positive and negative, the specific variable makes on the target classification. The plot is made up of all the sample points from the training dataset. We ranked all the variables from top to bottom according to the impact they have on the inhibition classification. The mean absolute feature (variable) values are shown as ranked vertical bar to the right, which shows the color gradient for the values going from blue for low to red for high, which corresponds to the low and high values for the individual variables as they have been numerically encoded for the model. The features highlighted via XAI are then added to the ID dashboard visualization to help end users understand the features that made the highest impact on the prediction of the block status. Both the global and the individual class feature relevance plots highlight the top 20 features that have the highest impact on the block status. This provides valuable insights for human labelers as they can focus on the specific features highlighted by XAI for labeling/validating the block predictions generated for the semi-supervised learning model.

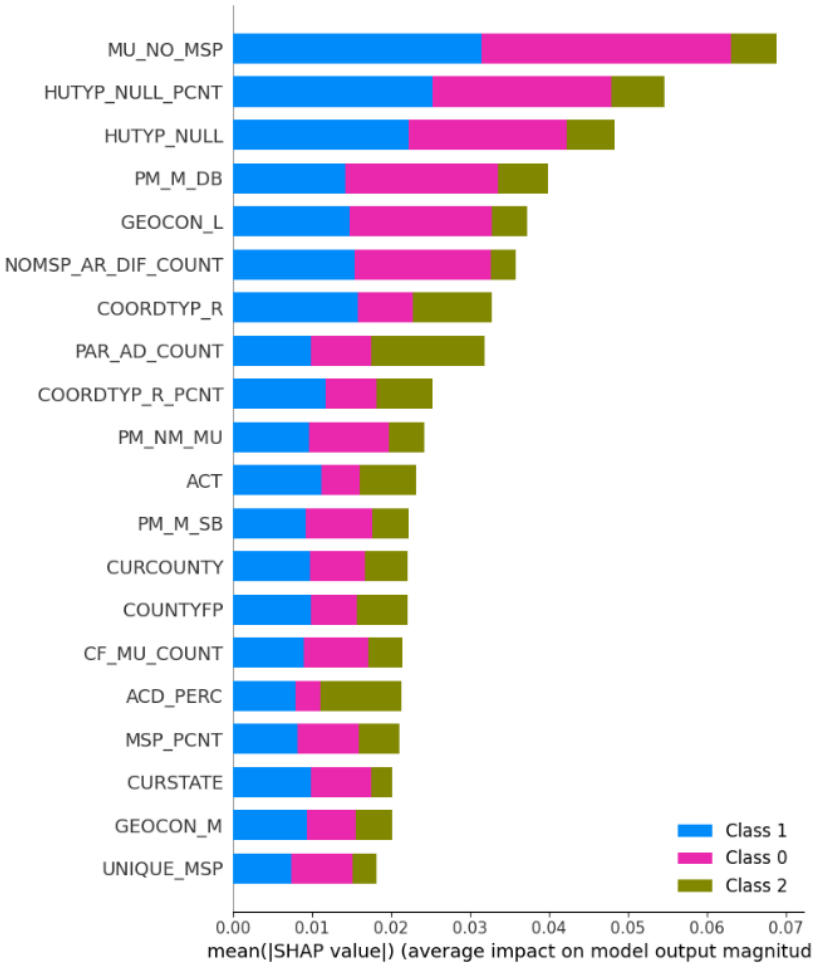

Figure 11: Feature importance plot for labelled blocks via SHAP for supervised learning model.

Figure 11 highlights the global feature relevance plot generated for the RF classifier. Here the top twenty features are highlighted from top to bottom in the order of the impact they have on the block status classification for the entire dataset. As shown in the plot each feature has a different impact on each label for the block status. *Figure 12* presents the feature relevance plots for all three labels of passive, undercoverage and overcoverage. As shown in the plots the top variables and their impact on the classification changes with each label since different features play different roles in the classification of the block status. While the MU_NO_MSP variable was the top variable for both the passive and over-coverage labels, it was listed much lower in the ranking for the undercoverage label.

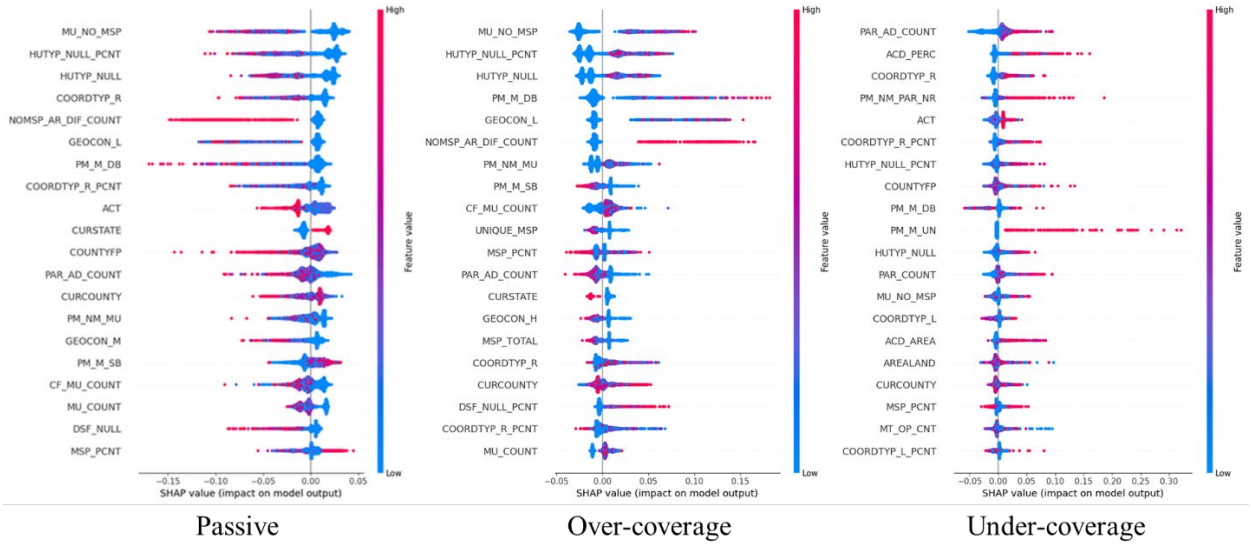

Figure 12: Feature importance plot for labelled blocks via SHAP for individual classes in the supervised learning model.

Once the feature relevance plots have been generated, they can be integrated directly into the dashboard as shown in *Figure 13*. For the block of interest, the info tab highlights the specific features that were relevant in the status classification. This additional layer of information can be helpful to human labelers for identifying potential biases when features that are not relevant are highlighted as impactful and vice-versa. Ideally, the highlighted features should have the highest impact. However, XAI features are correlation-based predictions dependent on ML classifiers and do not identify specific cause-and-effect relationships.

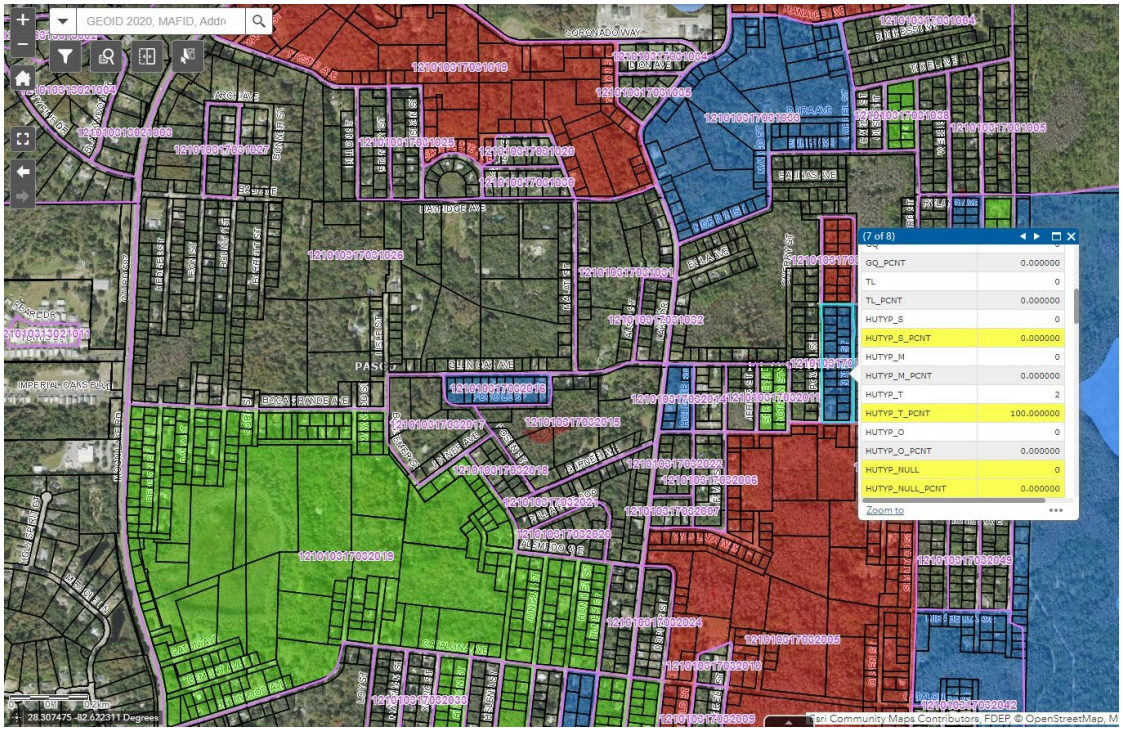

Figure 13: Example of the ID dashboard with prediction results for various blocks with the top features for the chosen block status highlighted in the info tab.

Semi-supervised Learning

Once the supervised learning models were trained and validated on the labeled data, the next step in the project was to apply the trained models to make predictions on the unlabeled data. For this we used multi-class with SMOTE oversampling models. As mentioned before and highlighted in *Figure 4* the semi-supervised models are trained using the entire labelled dataset for making predictions on batches (100K) of unlabeled dataset. Utilizing the active learning scheme, ML classifiers trained on labeled data are used to predict the block status on unlabeled blocks. To ensure robust predictions across all the classifiers we use a confidence interval of 95%. Block predictions with a low class-confidence or difference between the classifiers, are then sent back to human labelers for validation. The validated blocks are then added back to the training dataset to make predictions on the next batch until all the blocks have been classified. To do this, we export the predictions from all the classifiers into table for each block and ensure that there is uniformity between the classifiers. We mainly focus on the RF, XGB and CAT models as these were highlighted in the supervised learning portion to have the best performance. *Table 4* highlights the exported predictions for a sample subset of blocs that were predicted on using the different classifiers. The table highlights the similarity/uniformity in the predictions made by classifiers which we can add back to the training data. It also highlights the blocks that we need to send to the human labelers for validation as the predictions from two or more classifiers are in contrast to the rest of the predictions. Block 34567 highlights a block that would be sent back to the human labeler.

Table 4: Example predictions across all classifiers for different blocks.

Block ID	LR	RF	GBR	XGB	LGB	CAT
12354	0	0	0	0	0	0
23456	1	1	1	1	1	1
34567	1	1	**0**	1	**0**	1
45678	1	1	1	1	1	1
56789	1	1	**0**	1	1	1
67891	1	1	1	1	1	1
78910	1	1	1	1	1	1

After the predictions are validated by the human labelers, they are added to the ID dashboard to highlight the predictions. *Figure 14* highlights the predictions added to the dashboard where the end-user can easily check on the block status for each block. Here the blue blocks are all passive blocks, with red being over-coverage and green blocks classified as the under-coverage blocks.

With this project we aim to develop a machine learning solutions pipeline to improve both data labeling and classification of parcel data to enable new data-driven insight while reducing costs and effort for data assessment. Using the ID dashboard we're able to integrate a single user interface that allows both technical and non-technical users to train, evaluate and interact with both machine learning models and data assets. This project highlights the usability of the new emerging technologies such as AI/ML and XAI for helping automate processes that traditionally require a massive amount of human input and labelling hours. Because manually labeling each block requires 100,000+ hours of manual labor, using this approach we can cut costs using automation

approaches where AI/ML techniques using in conjunction with human-in-the-loop. The use of XAI techniques to highlight the relevant features also provides the ability to identify potential biases with classification of the block status.

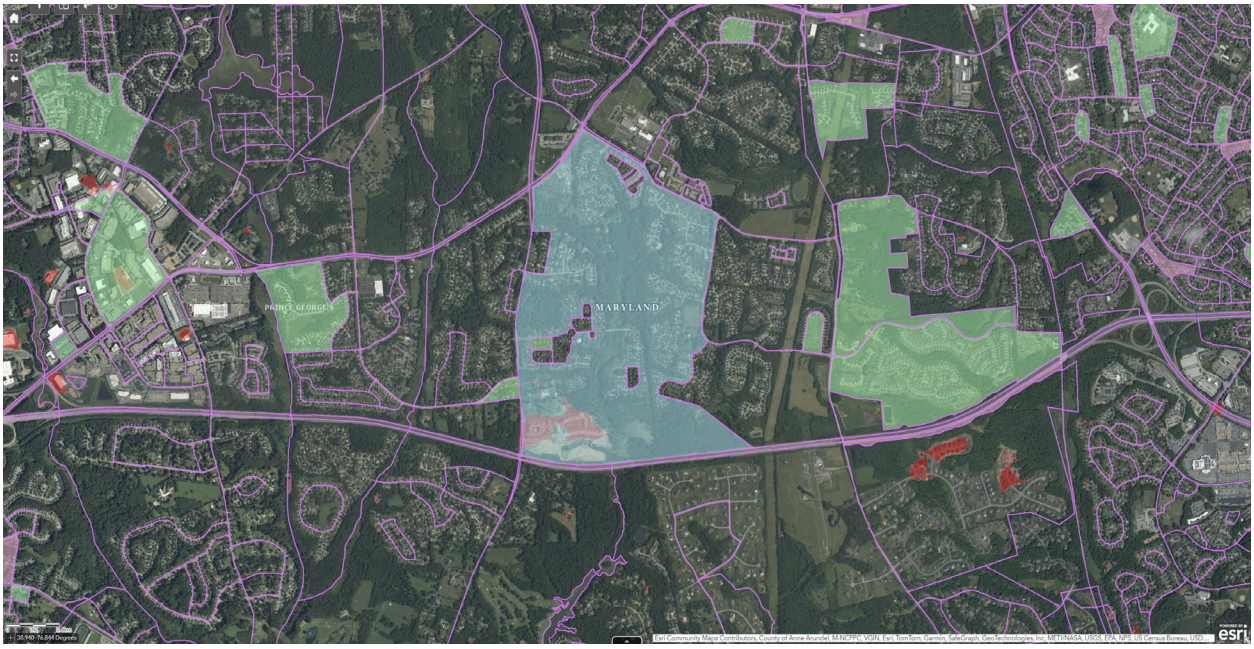

Figure 14: Example of the ID dashboard with prediction results for various blocks for the semi-supervised learning model.

There are still challenges that exist with this project as we are addressing a major issue with no one-solution fits all or magic bullet. Mainly the data itself presents a major challenge as we are using less than 1% of labelled blocks to make predictions on 99% of the remaining blocks. Manually labelling these blocks would cost the project years and years of effort to completely provide coverage for all the blocks within the US. Even with the labeled portion of the data, data-cleanliness has been a major challenge as administrative data can be very messy. We've also faced challenges with methodologies/techniques for the project as we initially utilized unsupervised learning methods but were not able to generate robust results. So, we pivoted into new methodologies/techniques as we progressed into the project as some of the things, we planned early on did not work. Finally, progress takes time. Creating the ML pipeline from data ingest to predictions to exporting to the dashboard took us a better part of the year.

Validating the predictions made by the classifiers without the use of human labelers also presents a unique challenge. We're exploring different options related to validating our predictions by using results from another project that uses image datasets to predict changes in blocks. Here we can validate any changes in either passive/active blocks which provides labels for a majority of the blocks, but the challenge still remains for the classification of the active blocks as either under or over coverage. Another validation method being utilized is the use of a list of blocks that are known to be passive as both a validation for the passive labeled predictions and additional labelled data for the classifiers. Here as the blocks are predicted to be passive, they can be validated, and then added back to the training data.

Conclusion

Manually canvassing and labelling each individual block within the US is a very large project. In the past traditional methods were utilized such as door to door canvassing. In-office address canvassing was started in 2015 and utilized for the 2020 census due to the ongoing pandemic, computational approaches were utilized for the first time resulting in a new and enhanced method of block-labeling. With all the advances in AI/ML within the past decade, the CB is looking to adapt its utilization for block labeling. To this end, we have developed a machine learning solution to improve both data labelling and classification of parcel data to enable new data-driven insight while reducing costs and effort for data assessment. Here we are able to accurately and precisely label blocks within the US. The project highlights the proof-of-concept for using AI/ML to core CB operational tasks. It also provides the benefit of removal of human bias involved in interactive review. It showcases Census' expertise and evolution in applying AI/ML in geographic data.

References

1. Zhang, J., et al., *Machine Learning Overview.* Broad Learning Through Fusions: An Application on Social Networks, 2019: p. 19-75.
2. Zhu, X. and A.B. Goldberg, *Introduction to semi-supervised learning*. 2022: Springer Nature.
3. Rawal, A., et al., *Recent advances in trustworthy explainable artificial intelligence: Status, challenges, and perspectives.* IEEE Transactions on Artificial Intelligence, 2021. **3**(6): p. 852-866.
4. Gunning, D. and D. Aha, *DARPA's Explainable Artificial Intelligence (XAI) Program*. AI Magazine, 2019. **40**(2): p. 44-58.
5. Ribeiro, M.T., S. Singh, and C. Guestrin, *"Why Should I Trust You?"*, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016. p. 1135-1144.
6. Lundberg, S. and S.-I. Lee *A Unified Approach to Interpreting Model Predictions*. 2017. arXiv:1705.07874.
7. Ribeiro, M.T., S. Singh, and C. Guestrin. *Anchors: High-Precision Model-Agnostic Explanations*. in *AAAI*. 2018.
8. Shrikumar, A., et al. *Not Just a Black Box: Learning Important Features Through Propagating Activation Differences*. 2016. arXiv:1605.01713.