

Vegetation AI PoC Extension Executive Summary

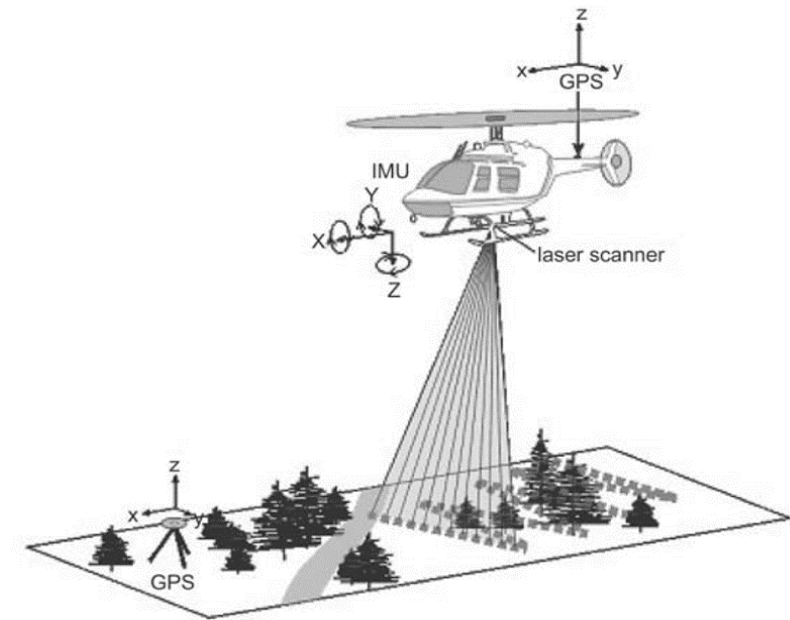
November 20th 2024

Vegetation AI POC & POC Extension Contextual Background

Contextual Background

Drivers that led to the vegetation AI POC & POC Extension:

- VM annually **assigns vegetation trim mitigations at the unique tree** level for environmental, work efficiency, access, and other business reasons
- SCE **completes & reports its vegetation mitigation work at the unique tree level**, and has established that standard as a compliance reporting precedent
- SCE has committed in its GRC to **transition from manual ground trim assessments to remote sensing assessments** in 2026, with increasing reductions in manual ground trim assessment in future years
- **Outcome:** Canopy Sense project was created to enable remote sensing to be applied at the individual tree level through the unique tree identification (crowns) & risk adjusted trim prescription (Rx) models



Vegetation AI Timeline

2024												2025			
Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	...
Proof-of-Concept						PoC Extension					Solution Analysis & Pilot				
Proved it is possible to use remote sensing to assign trims at the inventory (crown) level						Proved it is feasible to expand unique crown ID and trim Rx across SCE’s varied service territory					Plan to design & pilot a scalable solution that can be applied across SCE’s network				
												Today			

PoC Extension Phase Overview

The PoC Extension phase proved the feasibility of expanding the Canopy Sense solution across SCE's varied service territory

The POC Extension consisted of four primary work areas



Architecture

Build the foundations of the ML architecture, pipelines to run the crown workflow on new LiDAR, and identify ML data management issues.



Crowns

Rebuild and harden the crown algorithm and test its effectiveness on trees along a variety of SCE's circuits.



Trim Prescriptions

Test the effectiveness of trim Rx model on T&D circuits across SCE's service territory and across both HFRA and Non-HFRA.



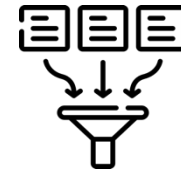
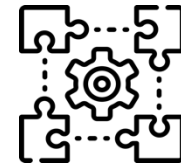
Trim Forecast

Test effectiveness of the model at describing the probability of a trim for a given crown over a 3-year time horizon.

Architecture

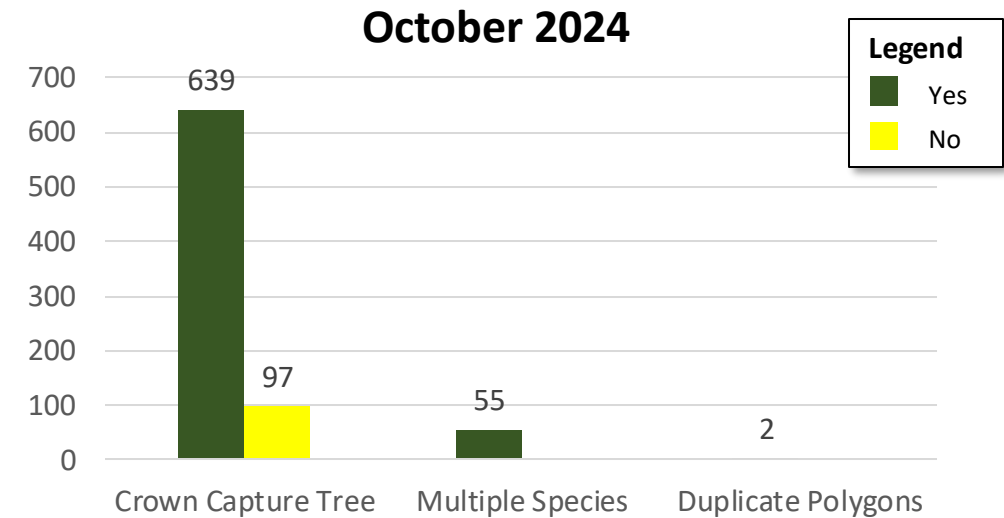
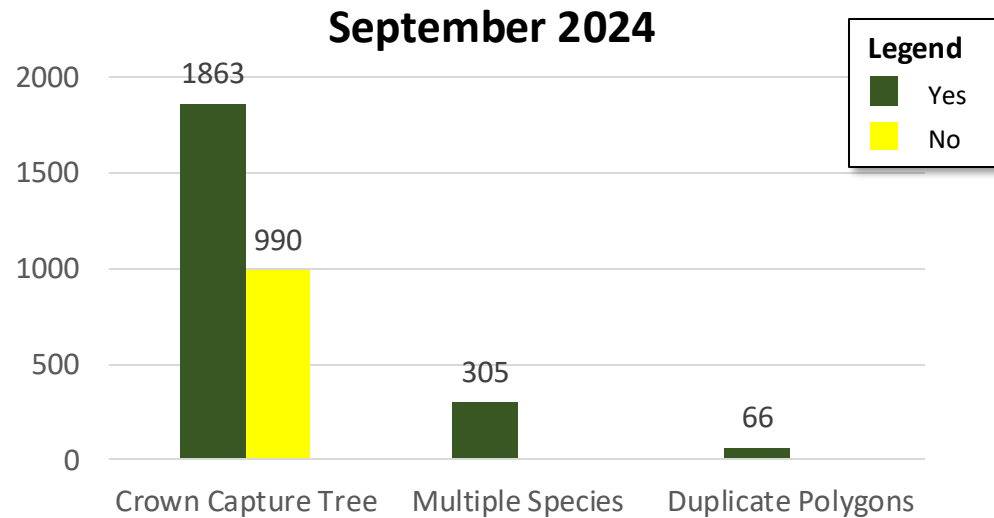
The POC Extension phase included the development of an AVD, data lakehouse platform, and an orchestration framework for producing crowns, trim RX, and trim forecast for the POC Extension circuits

	Extended POC – Achievements	Future Design Considerations to Scale
Data Lakehouse Platform	<ul style="list-style-type: none">▪ Store large variety of structured, semi-structured, and unstructured data sources in single consolidated location	<ul style="list-style-type: none">▪ Further integration with Snowflake and UDDR▪ Refine towards broader Enterprise adoption
ELT Data Pipelines	<ul style="list-style-type: none">▪ Ingest, transform, and refine data sources via iterative collection of pipelines	<ul style="list-style-type: none">▪ Add additional data sources, e.g., Earth Engine▪ Automate ingestion of new source data
Orchestration Framework	<ul style="list-style-type: none">▪ Automate, monitor, and scale distributed data pipelines and processes	<ul style="list-style-type: none">▪ Implement Cloud Composer to scale for volume▪ POC: ~ 4 circuits▪ Extended POC: ~ 40 circuits▪ Production: ~ 5000 circuits
AVD & MLOps Design	<ul style="list-style-type: none">▪ Automate, monitor, and scale machine learning systems	<ul style="list-style-type: none">▪ Implement automated model retraining▪ Implement model monitoring / feedback loop
Data Schemas & Quality	<ul style="list-style-type: none">▪ Define standard data schemas and quality expectations	<ul style="list-style-type: none">▪ Continually refine data schema definitions▪ Enforce data quality via automated system



Crowns

Crowns link remote sensing informed trim prescriptions to remote sensing identified new & existing inventory



Key Takeaways:

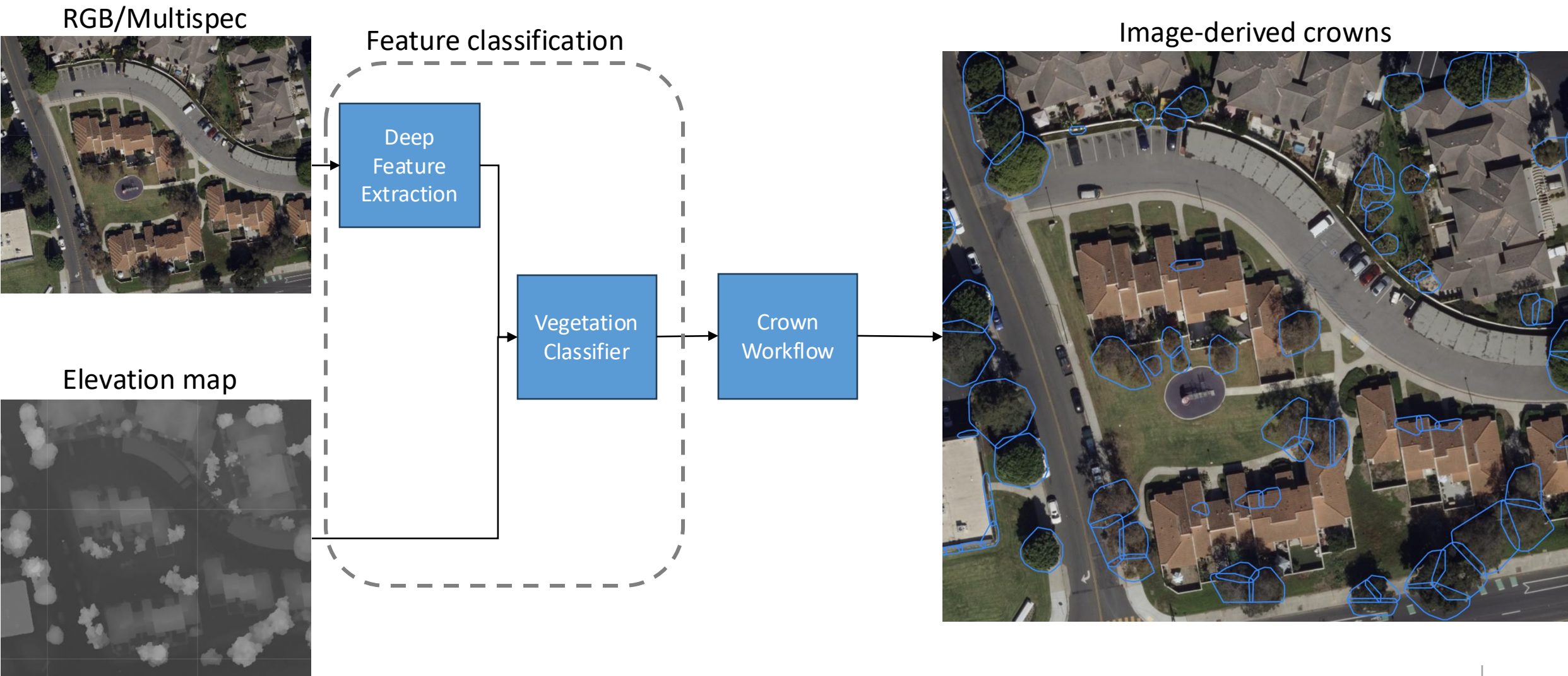
- A robust python library has been developed using SCE's LiDAR datasets as the primary driving input
- Achieved 'encroachment coverage' of 95% distribution and 96% for transmission
- Stakeholders agreed on "crawl/walk/run" approach for integrating remote sensing crowns with the existing vegetation inventory for the purposes of long-term inventory improvement and trim Rx for crowns/inventory
- Developed POC crown model code for ortho & satellite DSM to test as alternatives to LiDAR

Next Phase:

- When new feature classified LiDAR is moved to GCP production, we will want to automate the crown pipeline
- The crown polygons then will want to be integrated into Arbora throughout pilot and 2025 for "crawl" phase
- Further develop ortho & satellite variant python code

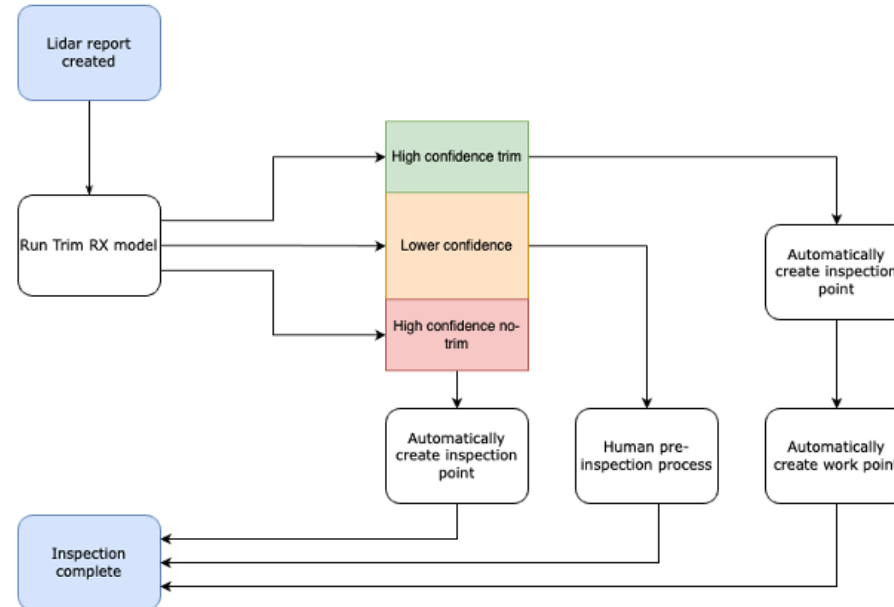
Crowns from Remote Sensing Images

Satellite and airborne stereo imagery can be used to derive crowns



Automating Trim Prescriptions

The Trim Rx model makes risk weighted trim/no-trim predictions at the unique tree level

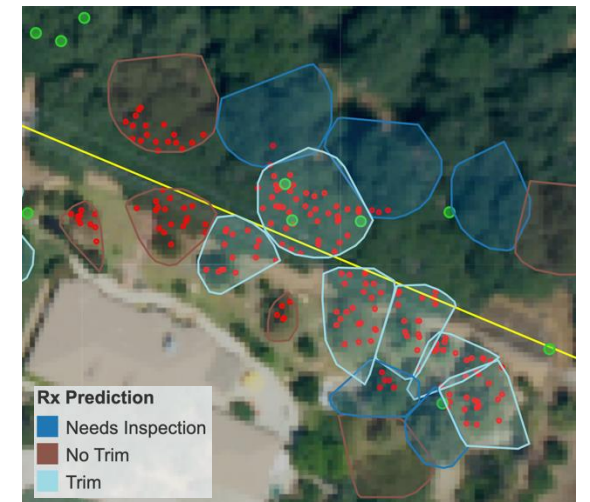
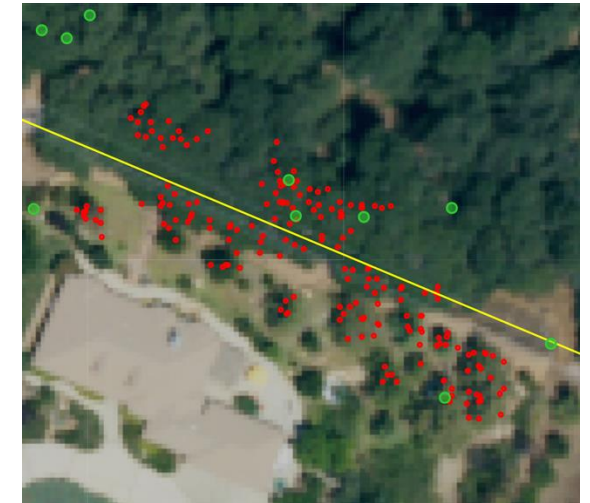


Key Takeaways:

- Integrated geospatial data drives the predictions of trim requirements at the unique tree level
- Based on the field validation data collected, 30% transmission and 12% distribution inspections can be **safely** automated

Next Phase:

- Expand the scope to ~200 circuits
- Increase the **safe** automation to 35% transmission and 12% distribution

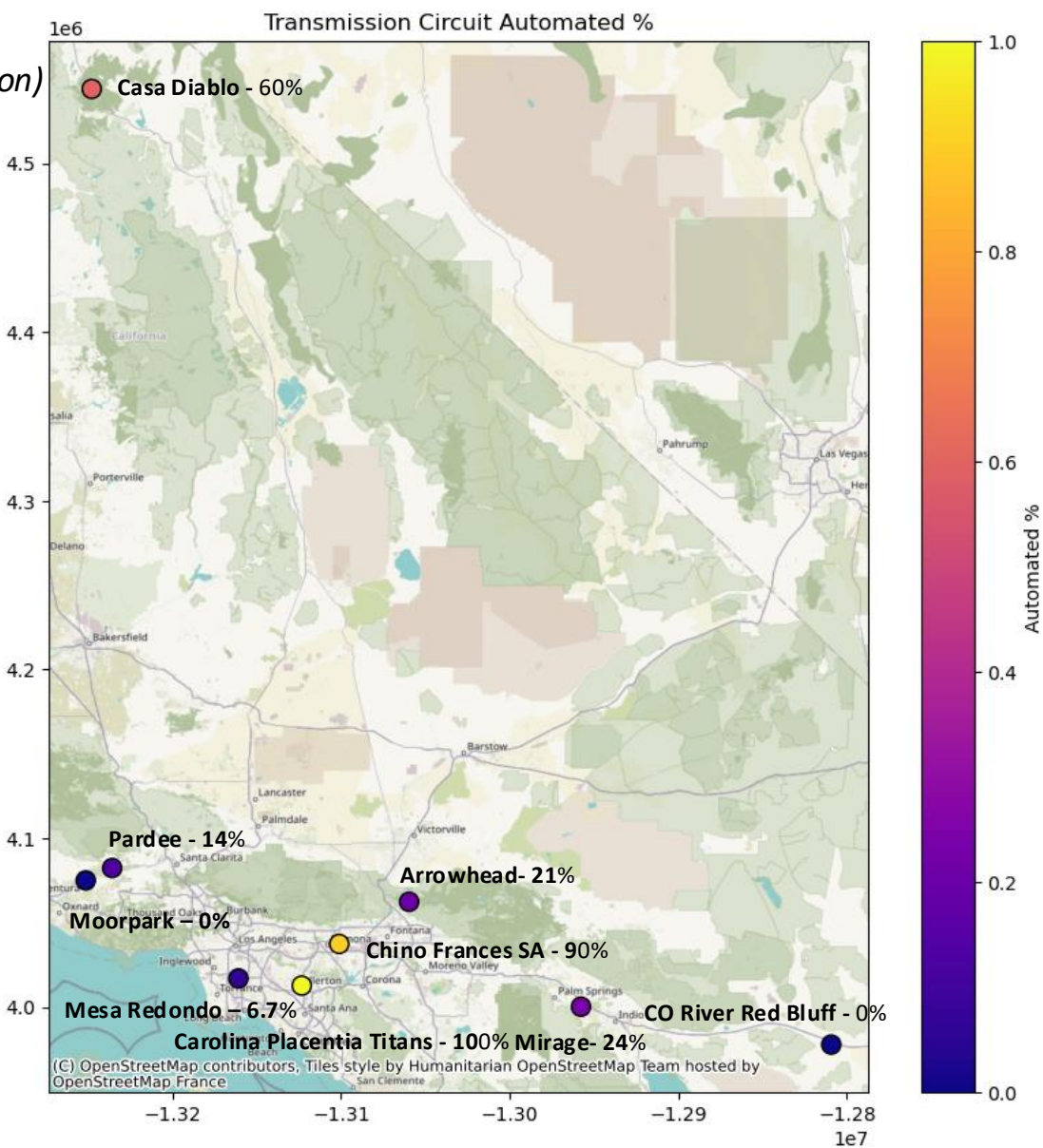


Trim RX Circuit Analysis: Transmission

Automated predictions have precision and recall of 100% (based on field validation)

Circuit ID	Name	Automated %	Total Field Observations	Automated Predictions
ET-00776	Pardee-SC	14%	22	3
ET-00886	Casa Diablo	60%	45	27
ET-01716	Arrowhead	21%	28	6
ET-00213	Mesa-Redondo	6.7%	30	2
ET-00861	Chino	90%	50	45
ET-01198	Mirage	24%	42	10
ET-01694	Villa Park	100%	29	29
ET-00775	Moorpark	0%	23	0
ET-00943	CO Red Bluff	0%	10	0

Takeaway: 35% initial automation target

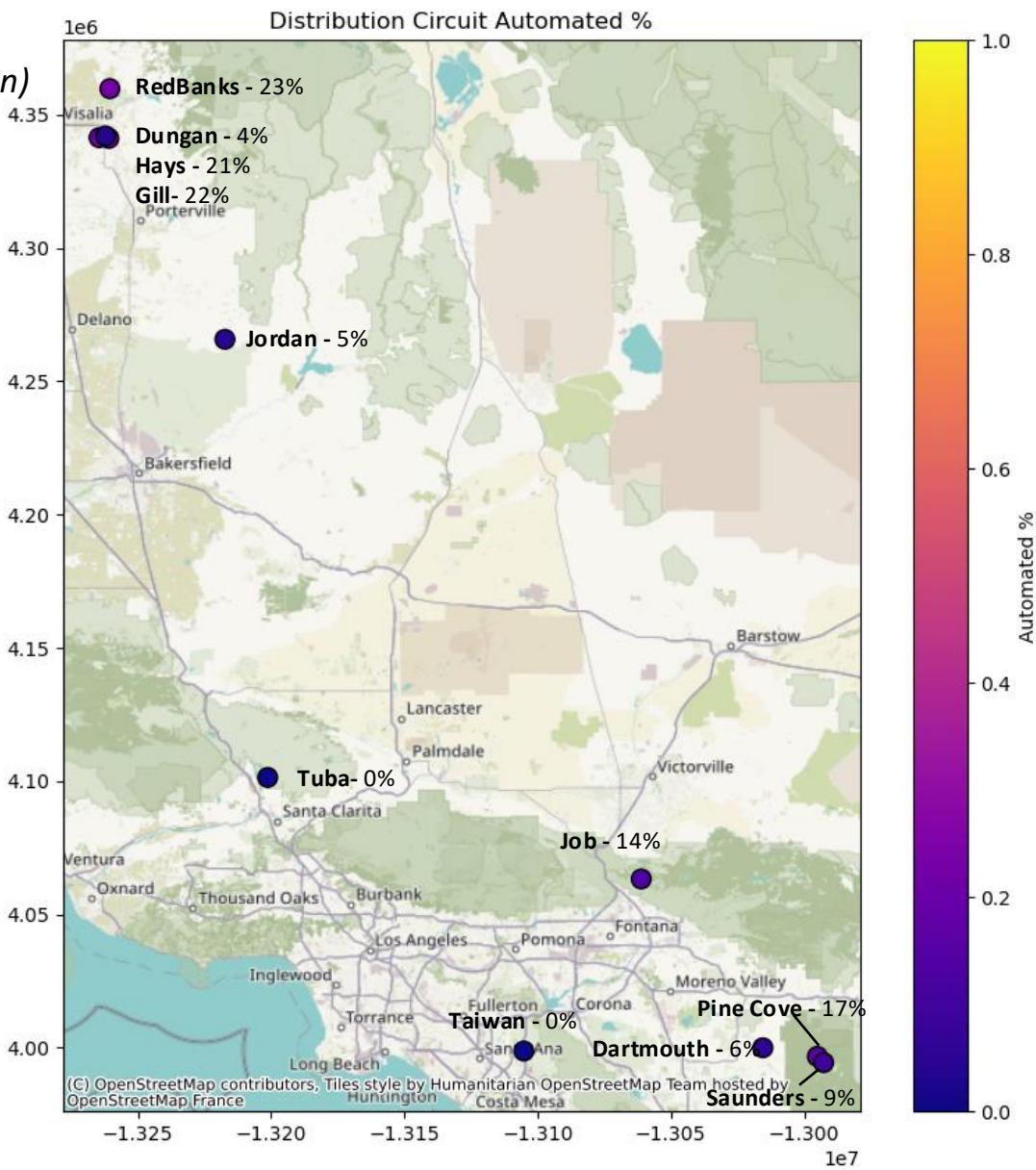


Trim RX Circuit Analysis: Distribution

Automated predictions have precision and recall of 100% (based on field validation)

Circuit ID	Name	Automated %	Total Field Observations	Automated Predictions
ED-14750	Redbanks	23%	48	11
ED-07240	Gill	22%	50	11
ED-08240	Hays	21%	53	11
ED-14097	Pine Cove	17%	30	5
ED-09275	Job	14%	28	4
ED-15922	Saunders	9%	22	2
ED-04693	Dartmouth	6%	32	2
ED-09320	Jordan	5%	40	2
ED-05400	Dungan	4%	52	2
ED-17487	Taiwan	0%	27	0
ED-18243	Tuba	0%	31	0

Takeaway: 15% initial automation target



Trim Forecast – Future Model Applications

The forecast model describes the probability of a trim for a given crown over a 3-year time horizon, which could be utilized in various applications

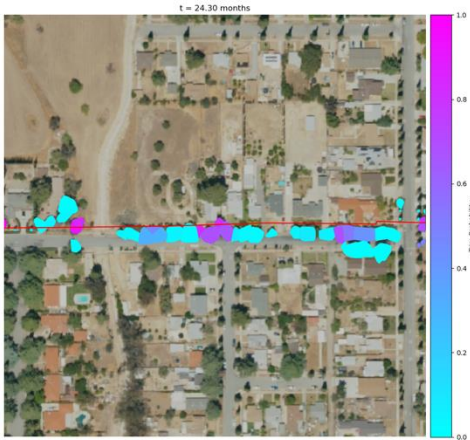
Business & Operational Applications of Trim Forecast Model

Operational Applications

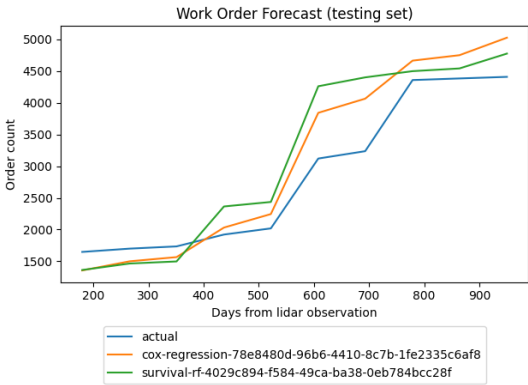
- **Work & schedule planning**
 - Determine trim & inspection month for a circuit’s inventory based on the forecasted number of trims needed
 - Identification or trim crew resources needed based on forecasted trims needed
 - Provide recommended trim depth for trees along spans given growth and remote sensing observed clearances

Business Planning Potential Applications

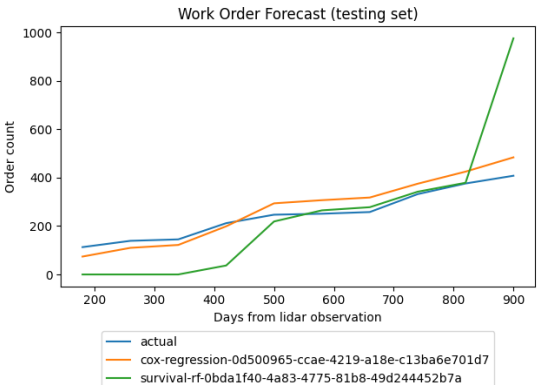
- **Risk-driven data refresh planning**
 - Predict when risk-level along a circuit will pass a threshold, triggering a new remote sensing data collection to re-establish the risk baseline
 - Differentiate when we should get a satellite / ortho / LiDAR clearance (once clearance margin of error is designated for each data type and collection medium)
- **Economic analyses**
 - Target expensive-to-maintain trees for removal through forecasted NPV of individual tree trim expenses
 - Budgeting trim and removal costs for Routine program at circuit level
- **Non-economic analyses**
 - Forecast how certain trees might impact SCE’s 2045 net-zero plan
 - Forecast how trees shading customer’s homes might impact customer energy usage



Circuit Type	Percent Error
Transmission	13.2%
Distribution	18.2%



Transmission



Distribution

¹ The AI Forecast model will rely on the Crown Segmentation process to run effectively to predict crown and future trims

POC Extension Conclusions

Four Key Takeaways

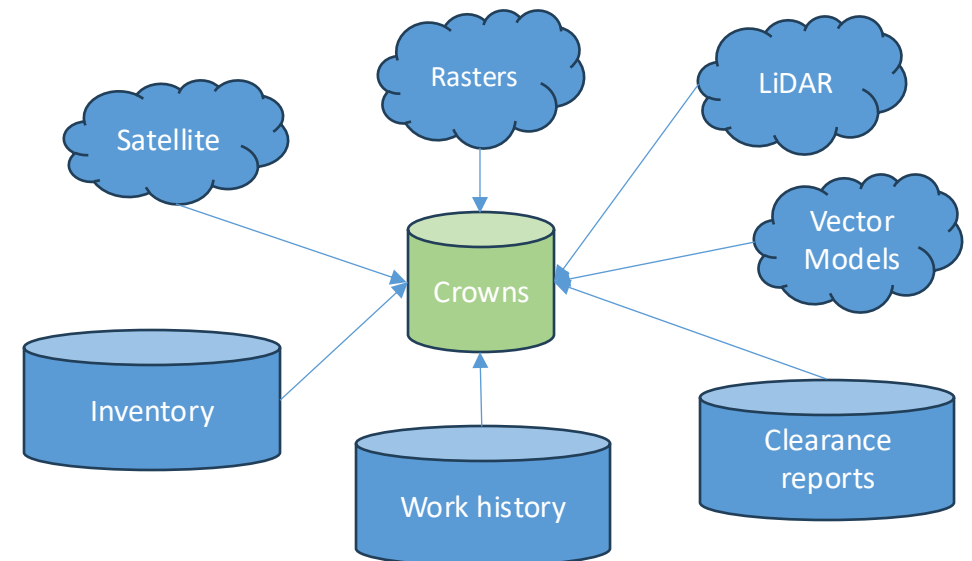
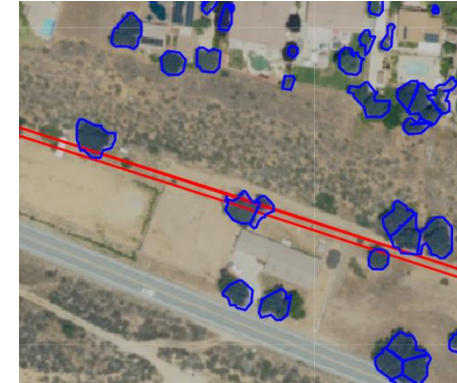
- 1 SCE Vegetation Management's **objective to transition from manual trim assignments to remote sensing trim assignments is feasible** while simultaneously maintaining unique inventory level (crowns) business and compliance requirements.
- 2 The POC and POC extension results demonstrate that with current data quality and data availability, **safe automation of 15% for Distribution and 35% for Transmission are possible for initial trim prescription automation.**
- 3 The POC Extension's **AI/ML architecture and data quality findings are primed to be further refined** to address the need to scale the solution and then subsequently piloted to prove the solution can operate at SCE's service territory scale.
- 4 **Enhancements to the POC Extension python crown and trim prescription/forecast models in addition to addressing feature classification & clearance calculation gaps are required to incorporate ortho + satellite imagery alternatives to the solution.**

Crowns

Why Canopy Crowns?

The crown grain is highly precise and enables fusion of the multi-modal vegetation management datasets for machine learning

- Algorithmically generated from classified LiDAR
- Closely aligned with physical trees
 - Best granularity for trim prescription and trim forecast modelling due to spatial accuracy
 - Better targeting of work assignments
- Enables the multi-modal data fusion required for machine learning



First PoC: [Pycrown](#)

An open-source, but non-maintained library was leveraged for the first PoC to accelerate progress, but it was unsuitable for the Extended PoC.

First PoC Rationale

- Implemented academic research on tree segmentation from LiDAR.
- Could be adapted to the first PoC use case relatively quickly
 - A small data volume made this a reasonable approach

Extended PoC Concerns

- A variety of patches and middleware were necessary to run the pycrown code in first PoC.
 - Doing this for the extended PoC would be difficult and error prone
- Code is not production ready
 - No testing
 - Highly coupled units
 - Poor separation of concerns
 - Not extensible
 - Not scalable

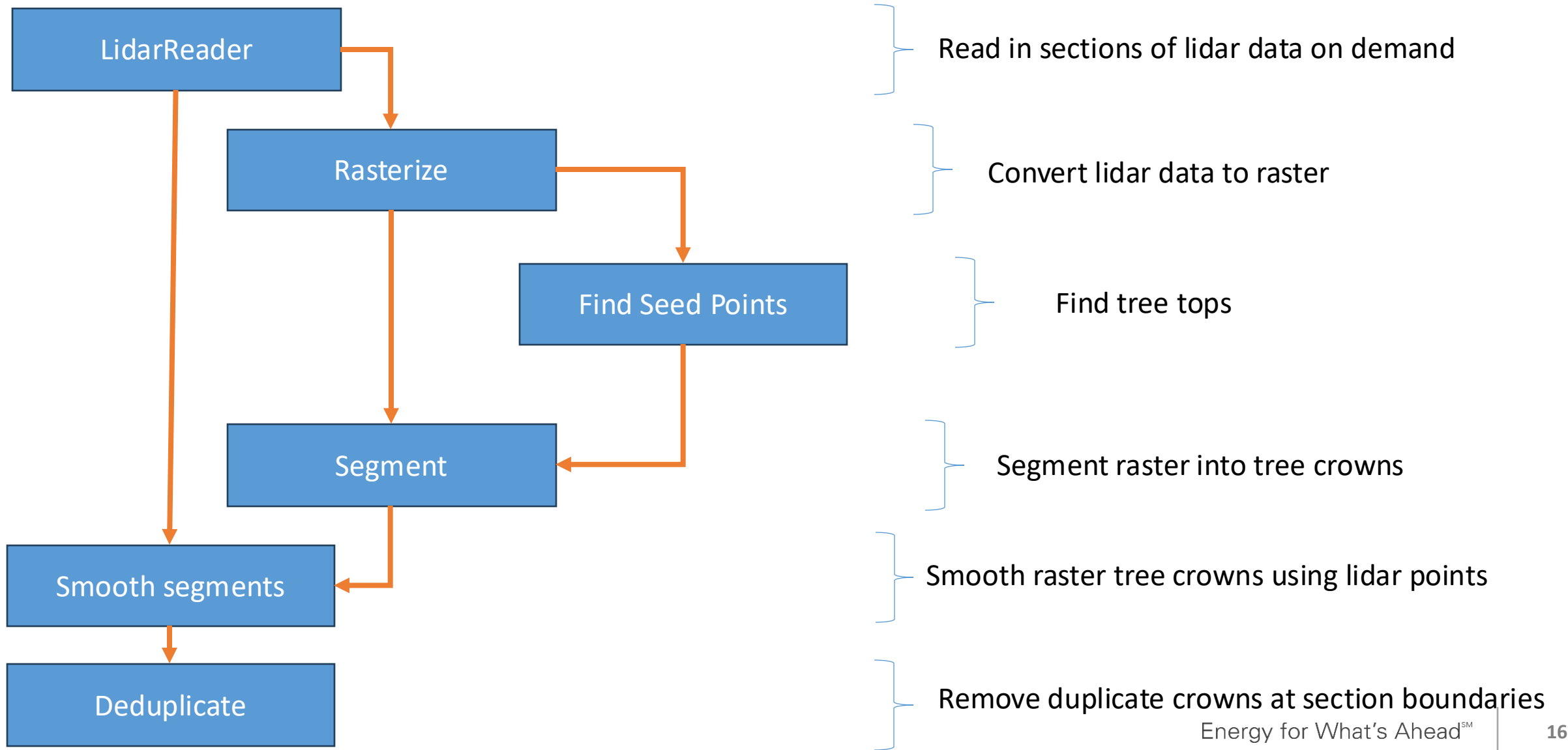
Extended PoC: [sfl](#) [sce](#) [crown](#)

A custom, production-grade Python library was created to perform crown segmentation

- Implements academic research on tree segmentation from LiDAR, just as Pycrown did
- Extensible and maintainable algorithm to run crown segmentation at scale
- Production ready
 - Fully unit tested
 - Good separation of concerns
 - Decoupled units
 - Extensible and scalable architecture

sfl_sce_crown architecture

An extensible, maintainable crowns algorithm suitable for productionalization

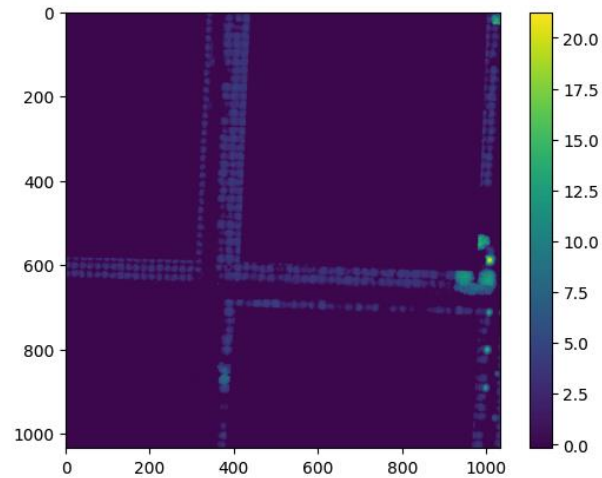
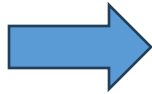


sfl_sce_crown architecture

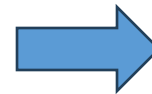
An extensible, maintainable crowns algorithm suitable for productionalization



Read in lidar data



Convert to raster data



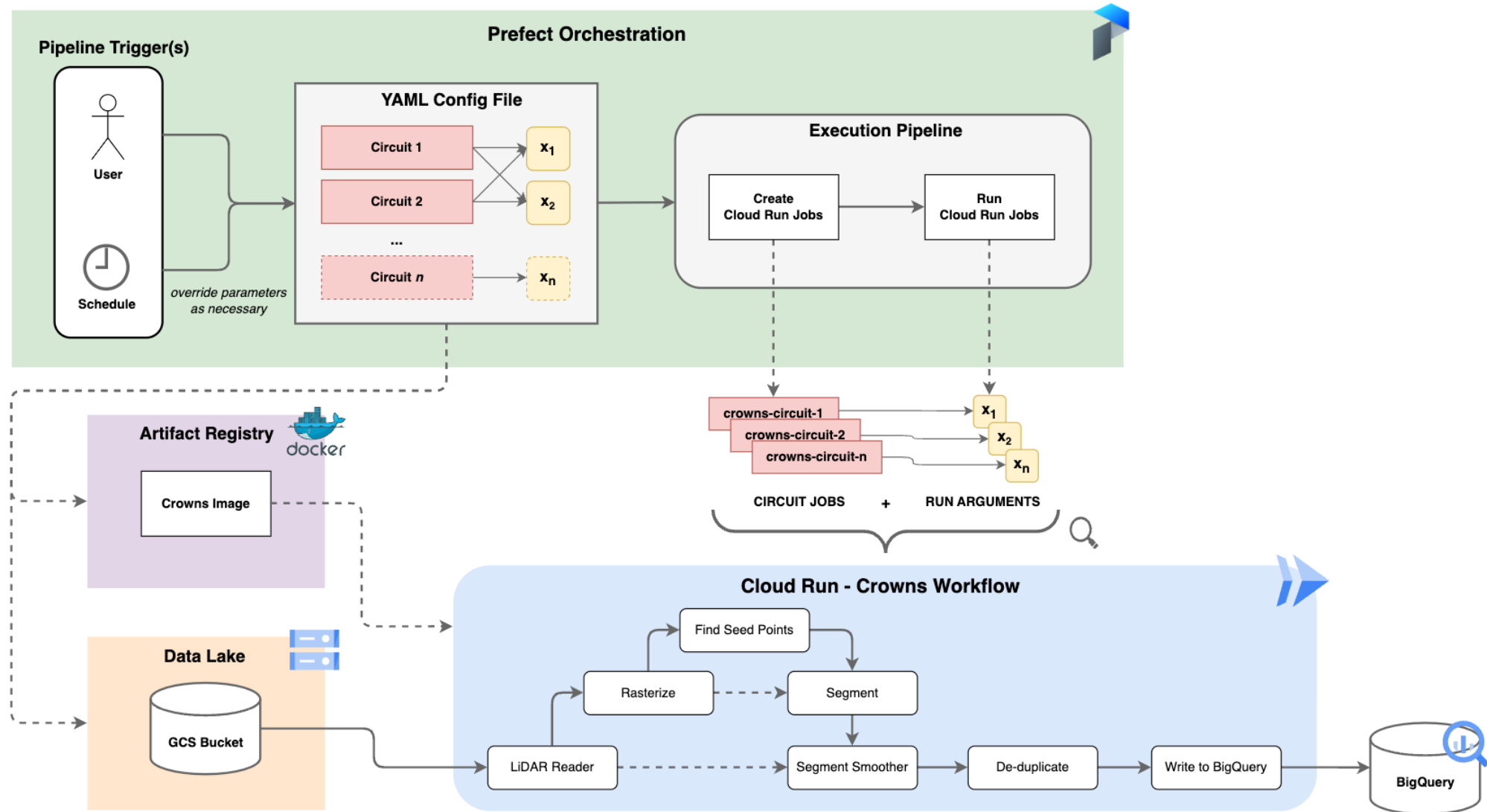
Find seed points



Segment and smooth

sfl_sce_crown deployment

Orchestrating the crown algorithm at scale



Crown segmentation assessment

Assess crown segmentation suitability for automated trim prescription and forecasting

- Clearance report encroachment points flag vegetation points that have any possibility of encroachment
- The trim RX and trim forecasting models should thus be aware of nearby encroachment points, since these are a critical input
- Lidar recall measures the crown algorithm's ability to produce crowns that cover encroachment points

lidar recall = prob(some crown captures lidar point | lidar point is an encroachment point)

Crown segmentation assessment

Overall results show the vast majority of encroachment points are captured by crowns

Assessment methodology

- Latest crowns and clearance reports for the extended PoC scope are used.
- Encroachment points must be at least 3 meters off the ground
 - Crowns algorithm is designed to detect trees, not brush or other ground cover
 - In production, these low lying points can be surfaced as possible brush, or a separate algorithm instance can be configured to detect these
- If an encroachment point is within 2 meters of a crown, it’s considered covered by said crown
 - This accounts for the fact that encroachment points are the worst offending points which may be somewhat far away from the main body of the tree.

Overall Results

circuit_type	lidar_recall	interval
overall	0.952	[0.952, 0.953]
distribution	0.951	[0.95, 0.952]
transmission	0.963	[0.961, 0.965]

Crown segmentation assessment by circuit

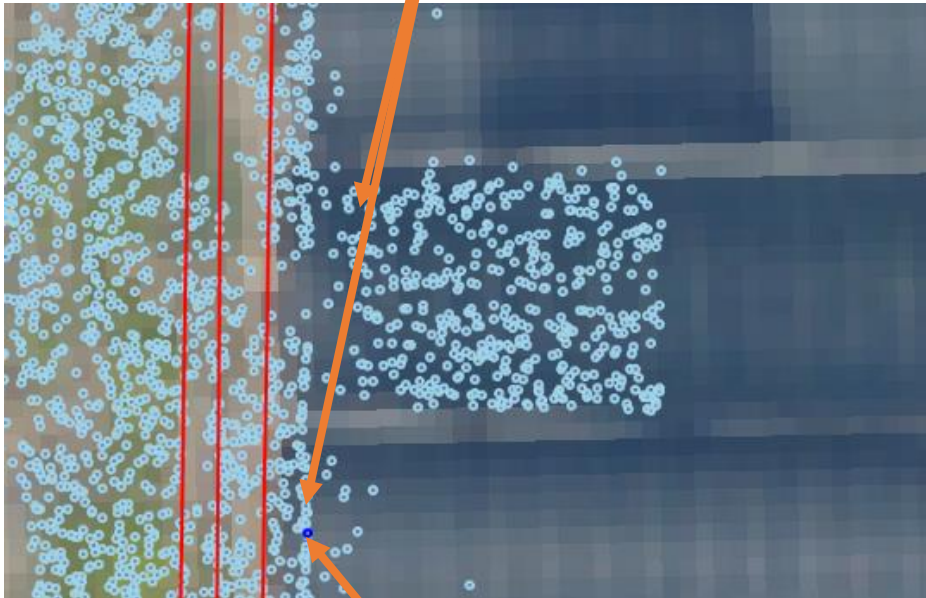
By circuit performance shows similarly strong results

circuit_id	lidar_recall	interval
ED-04693	0.962	[0.956, 0.969]
ED-05400	0.952	[0.938, 0.965]
ED-07240	0.958	[0.952, 0.964]
ED-08240	0.944	[0.935, 0.953]
ED-09275	0.937	[0.932, 0.941]
ED-09320	0.967	[0.965, 0.968]
ED-14097	0.941	[0.938, 0.943]
ED-14750	0.956	[0.949, 0.962]
ED-15922	0.94	[0.938, 0.943]
ED-17487	0.983	[0.977, 0.989]
ED-18243	0.932	[0.914, 0.953]
ET-00213	0.922	[0.888, 0.958]
ET-00775	0.975	[0.968, 0.982]
ET-00776	0.974	[0.967, 0.981]
ET-00861	0.983	[0.975, 0.991]
ET-00886	0.977	[0.974, 0.979]
ET-00943	0.984	[0.974, 0.995]
ET-01198	0.961	[0.95, 0.972]
ET-01694	0.975	[0.963, 0.985]
ET-01716	0.932	[0.927, 0.937]

Crown segmentation performance issues

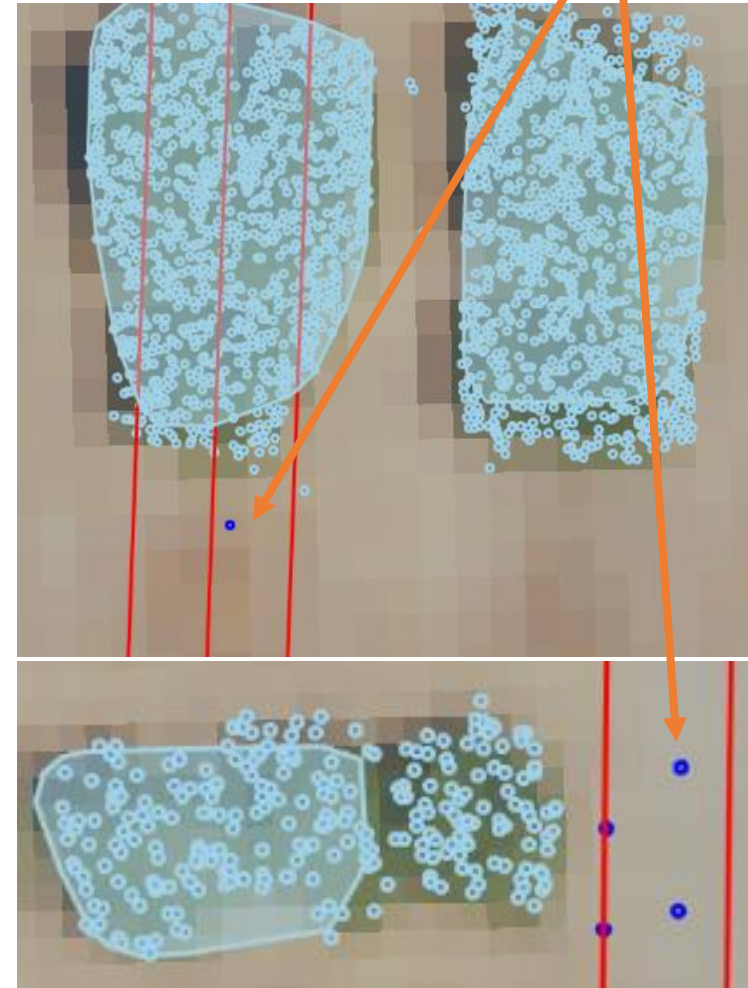
Lidar misclassifications cause some of the low performance

Misclassified as vegetation points (ED-07240)



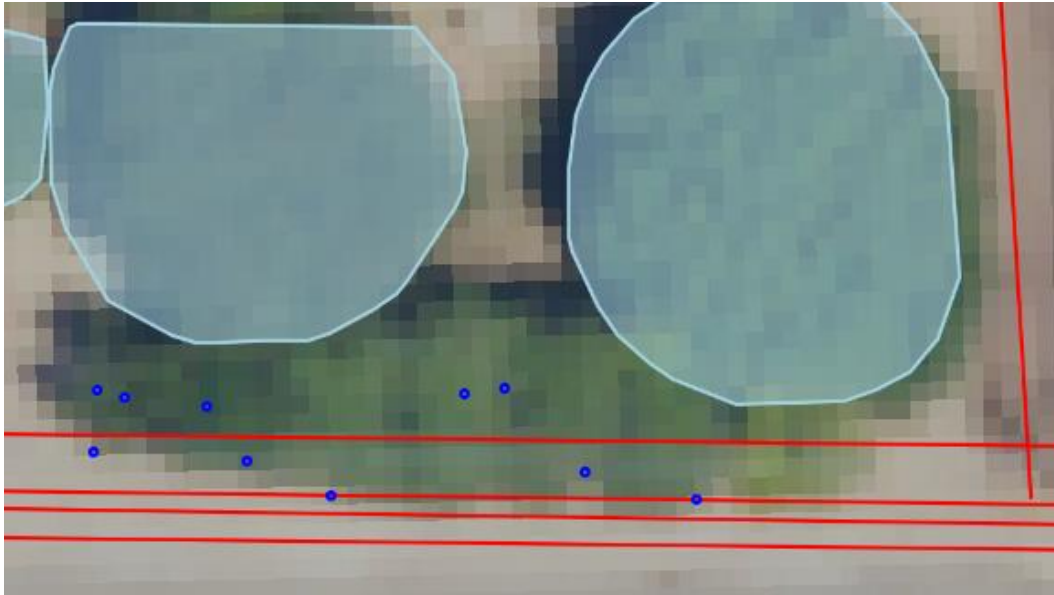
Spurious encroachment point

Encroachment point with no lidar points nearby (ED-07240)

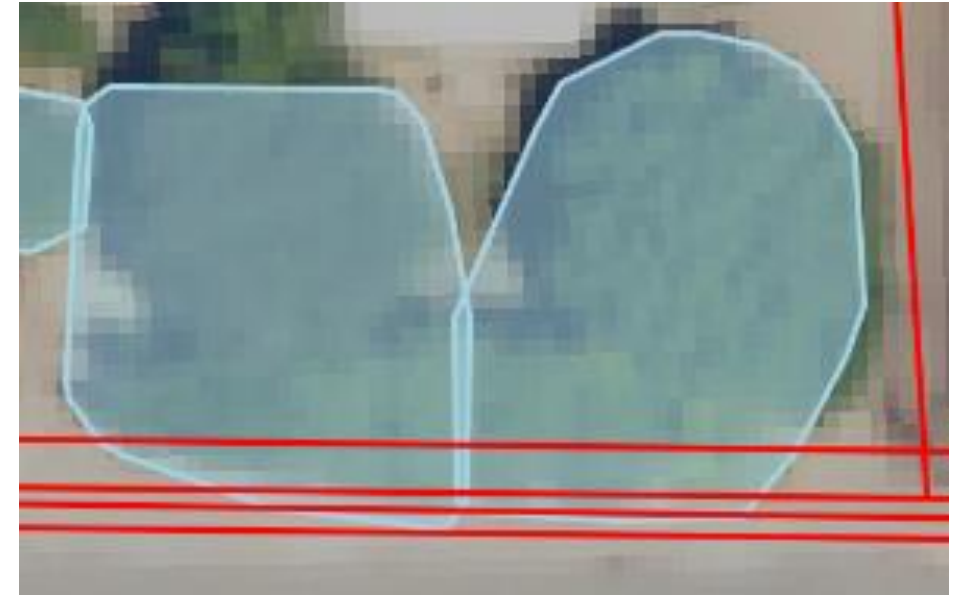


Crown segmentation performance issues

Tuning can be adjusted to improve performance



Max crown diameter: 15m



Max crown diameter: 30m

Other crown quality considerations

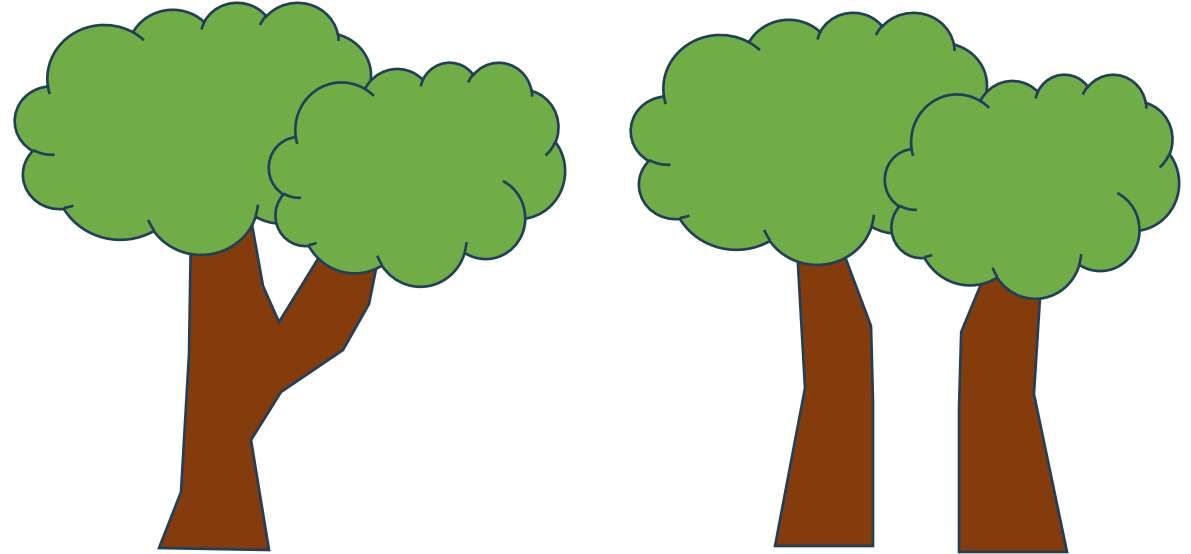
Lidar recall narrowly focuses on the Trim RX and Trim Forecasting use cases

Over segmentation

- Not critical for Trim RX and Trim Forecasting
- Can be reduced via tuning, but care should be taken not to impact lidar recall
- Difficult to eliminate due to tree growth patterns

Height

- Will be as accurate as lidar
 - Leverages a digital evaluation map computed via interpolating lidar ground points
- Hard to assess with field measurements, since it can be difficult to accurately measure height in the field
- Field test height MAPE of 36%



Tuning the crowns algorithm to distinguish between these is likely not possible to do consistently: Tunings that correctly segment the right trees will over segment the left tree.

Derived tree features

Lidar crowns additionally unlock a variety of tree features

- Tree volume
 - Volume of 3D convex hull of lidar points captured by crown polygon
- Tree surface area
 - Surface area of 3D convex hull of lidar points a captured by crown polygon
- Tree density
 - Estimate number of lidar points per unit volume
- Tree height
 - Distance between tree top raster and derived digital elevation raster
- Ground elevation
 - Elevation of tree

Crown segmentation data and code

Locations of code and crown data

Code

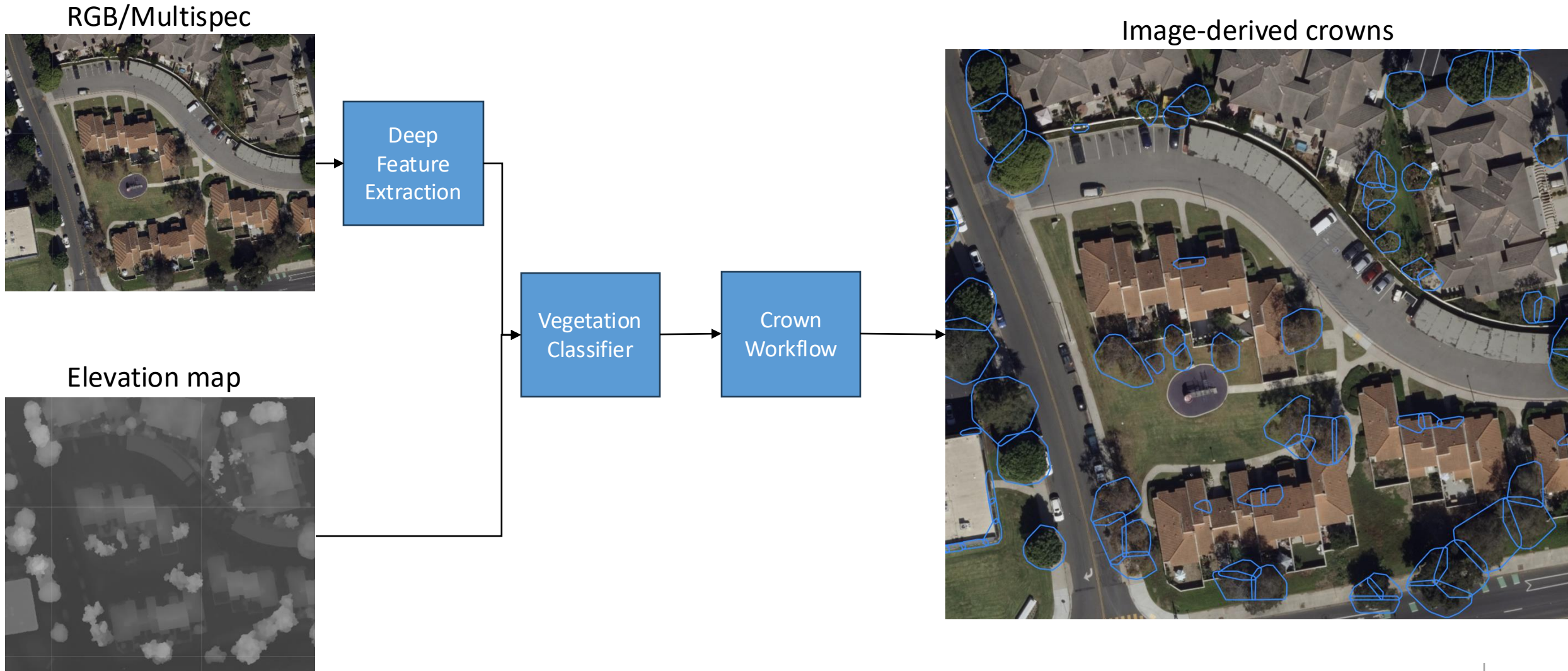
- sfl_sce_crown
 - https://github.com/EdisonInternational/sfl_sce_crown
 - General library for crown segmentation of classified LiDAR
- sfl_sce_crown_interface
 - https://github.com/EdisonInternational/sfl_sce_crown_interface
 - Interface to run the crowns algorithm in a GCP service
- sfl_sce_orchestration
 - https://github.com/EdisonInternational/sfl_sce_orchestration
 - Orchestration pipeline to run the crown algorithm at scale

Data

- CURATED.CROWNS
 - BigQuery table with results of running the crown algorithm
- ML_DEVELOPMENT.FEATURES
 - BigQuery table with fusion of crowns with many other data sources
 - Developed for ML models

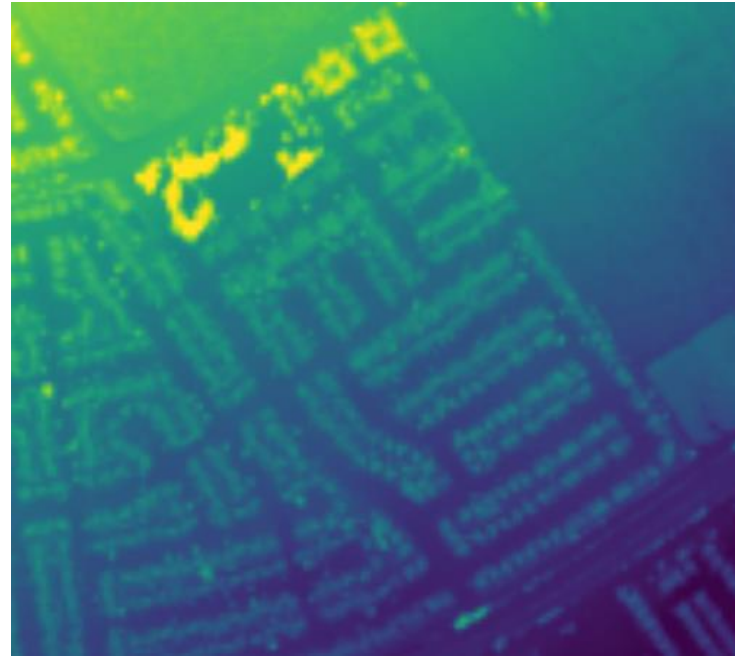
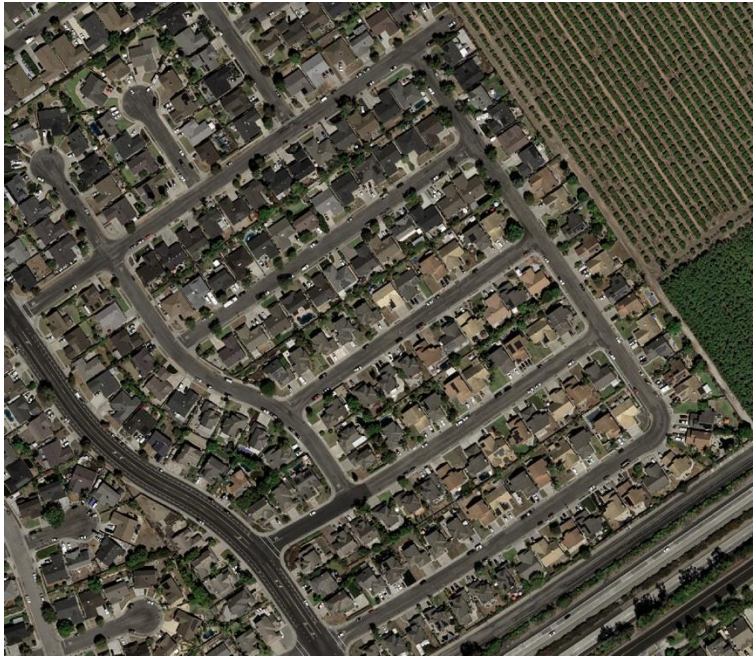
Crowns from remote sensing images

Satellite and airborne stereo imagery can be used to derive crowns



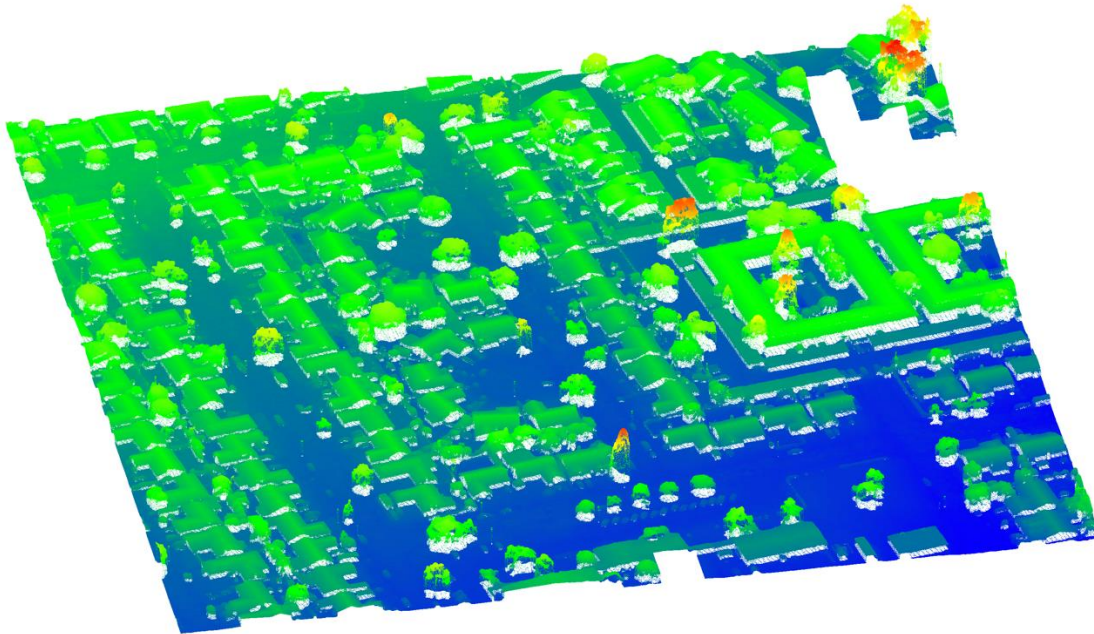
Crowns from remote sensing images

Satellite and airborne stereo imagery can be used to derive crowns



Crowns from remote sensing images

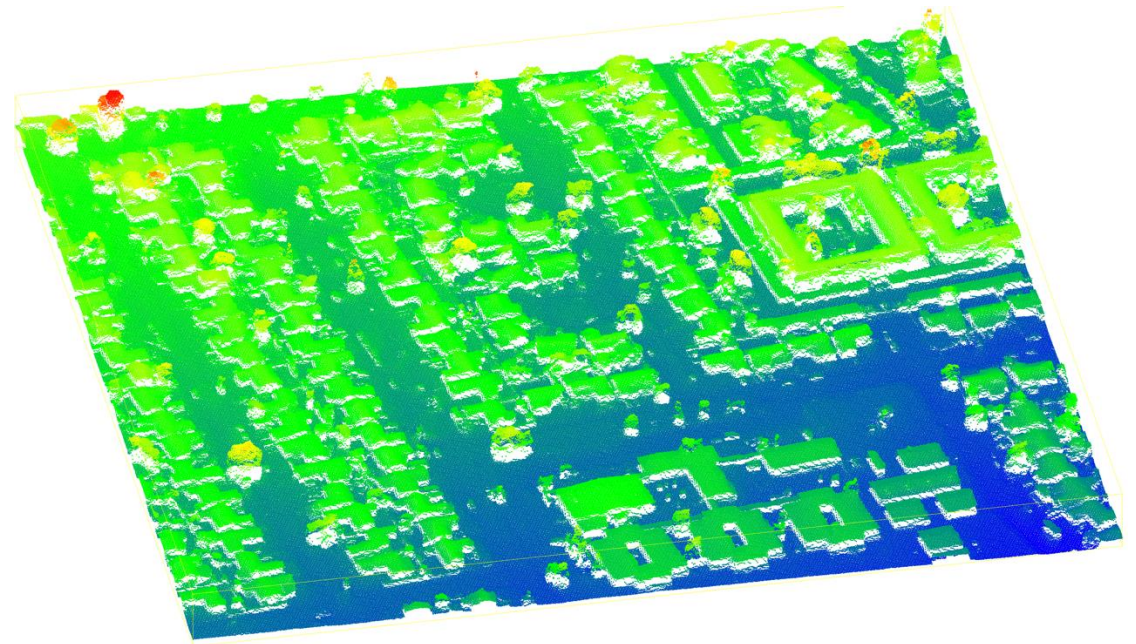
Satellite and airborne stereo imagery can be used to derive crowns



Vexcel Ortho-imagery

RGB (7.5 cm)

DSM (7.5 cm)



Airbus Tri-Stereo

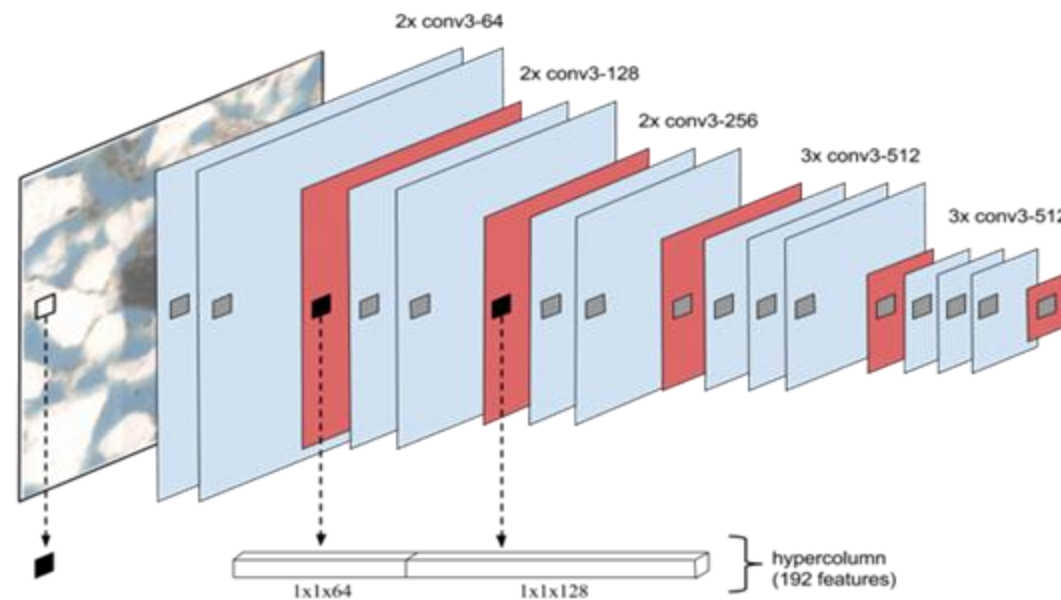
4-band multi-spec (30cm)

DSM (50 cm)

Crowns from remote sensing images

Satellite and airborne stereo imagery can be used to derive crowns

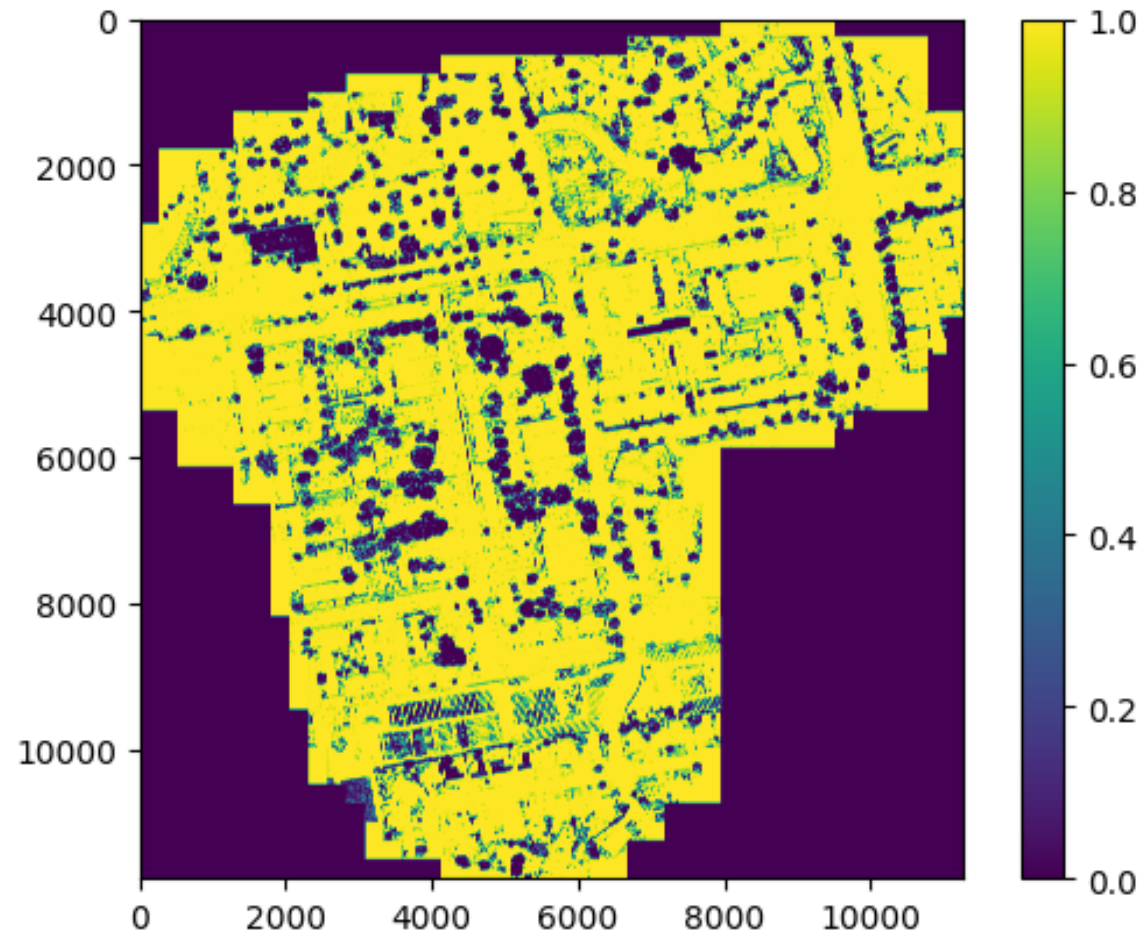
VGG-16 (pre-trained on ImageNet)



Simonyan and Zisserman (2015)
Hariharan et al (2015)

Vegetation feature classification

Satellite and airborne stereo imagery can be used to derive crowns



Crowns from remote sensing images

Additional considerations for an image-based Rx workflow

- Coordinate Reference Systems
 - Image data is typically provided with a CRS based on a dynamic datum (WGS84). Transferring that to a static datum (NAD83) is non-trivial and may require manual corrections
- Clearance Reports
 - Need to understand if clearance reports can be generated from this data, and all of the issues/requirements involved in this process

Preliminary Strategy for Improving Vegetation Inventory via Crowns

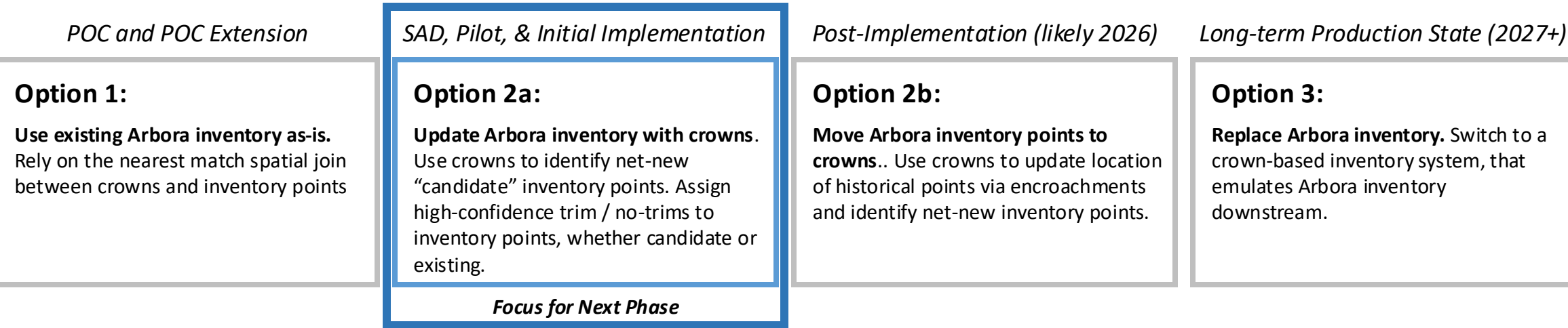
The integration strategy will take a phased approach, starting simple and building complexity as program matures

Overall Strategy Outcomes:

Team is aligned with crawl (2a), walk (2b), run (3) phased approach¹.

Team agrees 2a aligns with the Solution Analysis phase while also establishing the fields necessary for 2b, where the scope of pilot is focused on 2a. As time progresses, the project and field team will incrementally begin fusing both future state crown + existing inventory tables into 2b, eventually leading us to a final production state of 3 long-term.

Applicable Project Phase

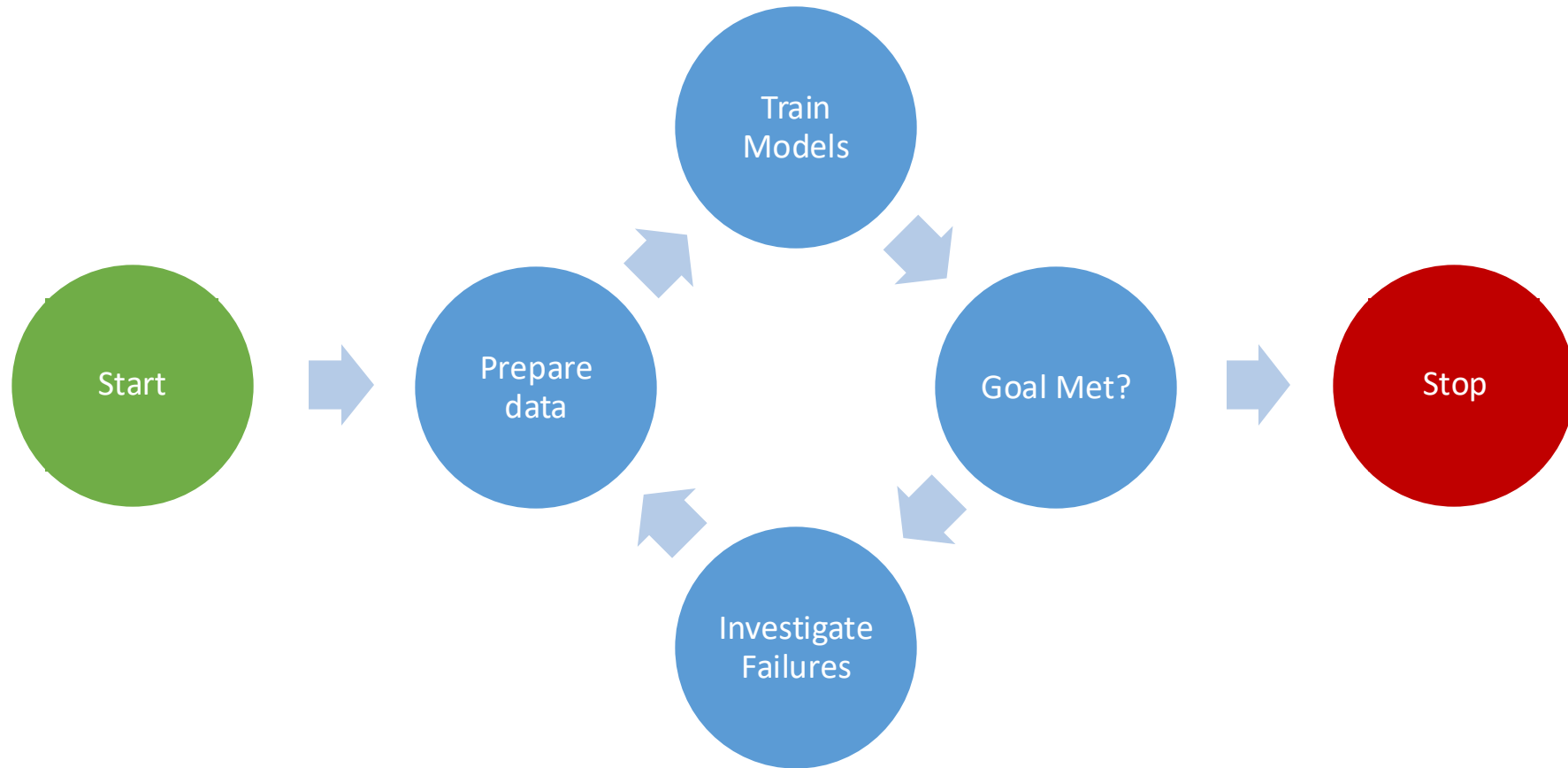


Notes: 1) Team alignment on high-level strategy from the October 2, 2024 Canopy Sense to Arbora meeting. Additional discussions on design of these integration approaches will continue. As necessary, more conversations will be had to refresh this strategy.

Trim RX

A Note on ML Training Workflows

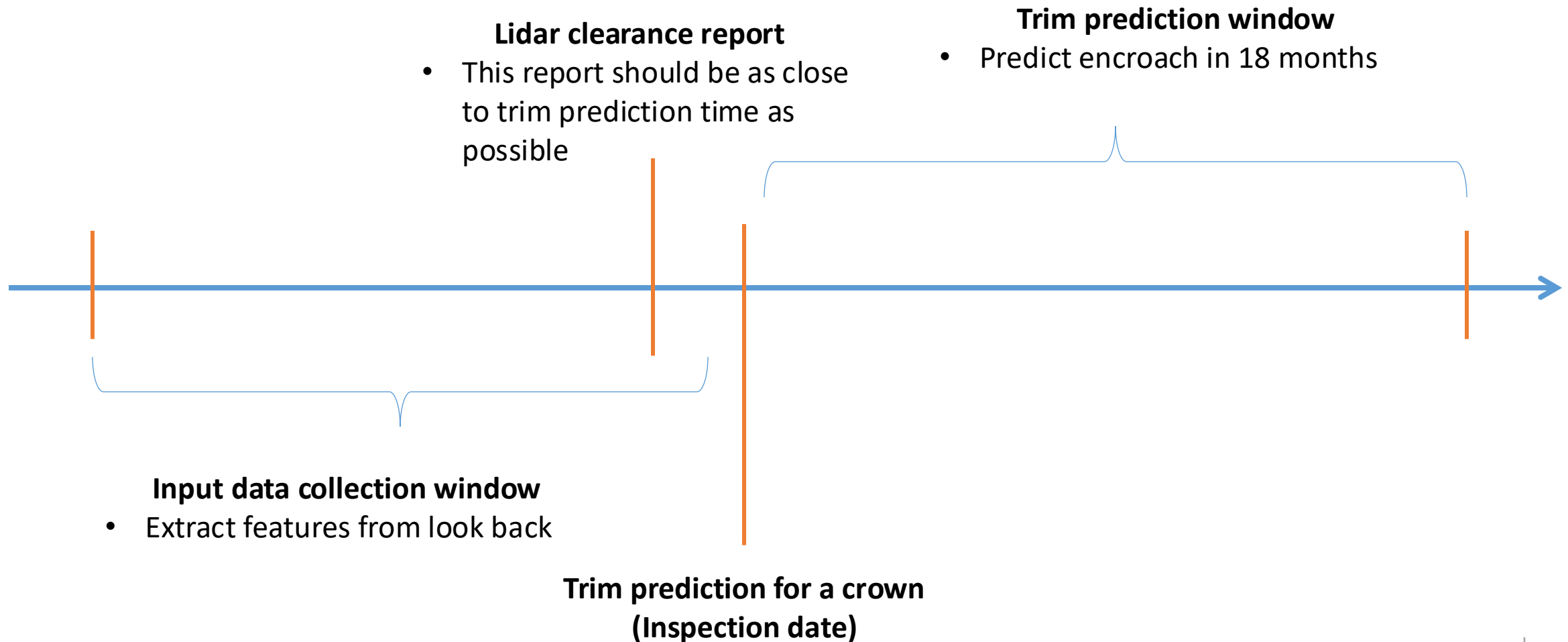
Training ML models is a cyclic process that halts when the goal of the modeling is achieved or determined to be unattainable



Goal for Extended PoC: Train models that safely automated 10% to 30% of inspections

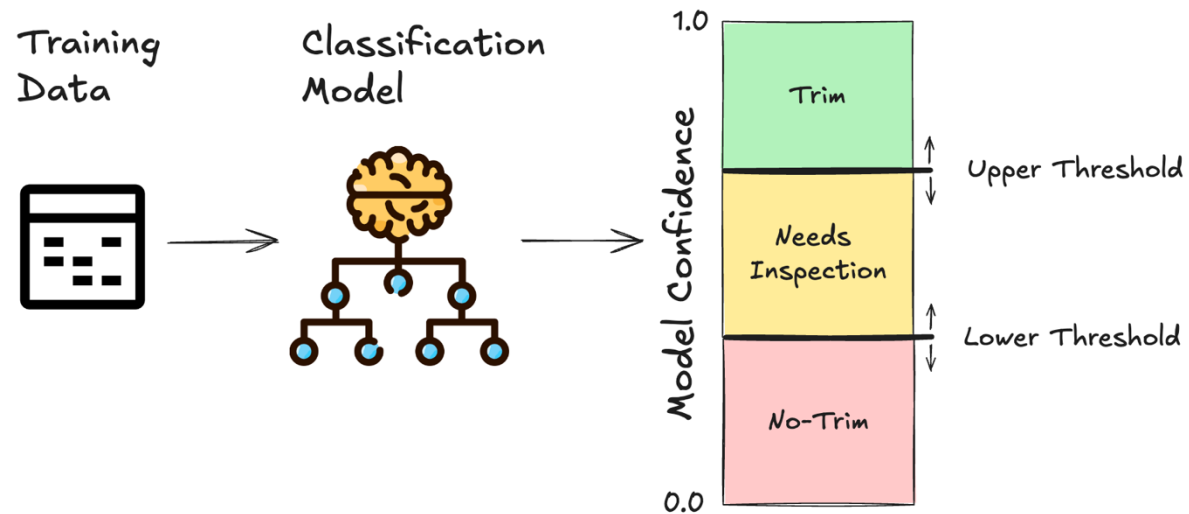
Trim Rx Model

The trim Rx modeling problem is framed as a classification problem: on the Crown grain, predict whether a tree will encroach on the circuit in the next 18 months



Model Training

Classification models output a confidence score. While training the model, we optimize for upper and lower thresholds that minimize the risk-weighted cost of inspection.



Trim Rx Business Metrics

Business metrics assess model utility

Costs

$c_{current}$ = current unit cost
 c_{model} = unit cost utilizing a model

 $c_{inspect}$ = cost of one inspection
 c_{trim} = cost of one trim
 c_{missed} = cost of one missed trim

Deployment Parameters

α = proportion of automated Rx

Model

M = model prediction
 H = human prediction
 A = actual state of the world

$$\text{unit business value} = c_{current} - c_{model}$$

$$c_{current} = \underbrace{\text{prob}(A = \text{no trim} \ \& \ H = \text{trim}) \cdot c_{trim}}_{\text{Human risk of false trim}} + \underbrace{\text{prob}(H = \text{no trim} \ \& \ A = \text{trim}) \cdot c_{missed}}_{\text{Human risk of missed trim}} + \underbrace{c_{inspect}}_{\text{Cost of pre-inspector}}$$

$$c_{model} = \underbrace{\alpha \cdot \text{prob}(A = \text{no trim} \ \& \ M = \text{trim}) \cdot c_{trim}}_{\text{Model risk of false trim}} + \underbrace{\alpha \cdot \text{prob}(M = \text{no trim} \ \& \ A = \text{trim}) \cdot c_{missed}}_{\text{Model risk of missed trim}} + \underbrace{(1 - \alpha) \cdot c_{current}}_{\text{Human inspected}}$$

Finding Optimal Thresholds

Optimal thresholds are derived during training via a brute force grid search that is bounded by percentile granularity and maximum coverage

- List all percentile pairs $\{(l_1, u_1), \dots, (l_m, u_m)\}$ such that $l_i + (1 - u_i) \leq \alpha$ where alpha is the configured maximum coverage and the percentiles are at the configured granularity.
- Apply a data splitting scheme to split training data into a collection of train and test splits

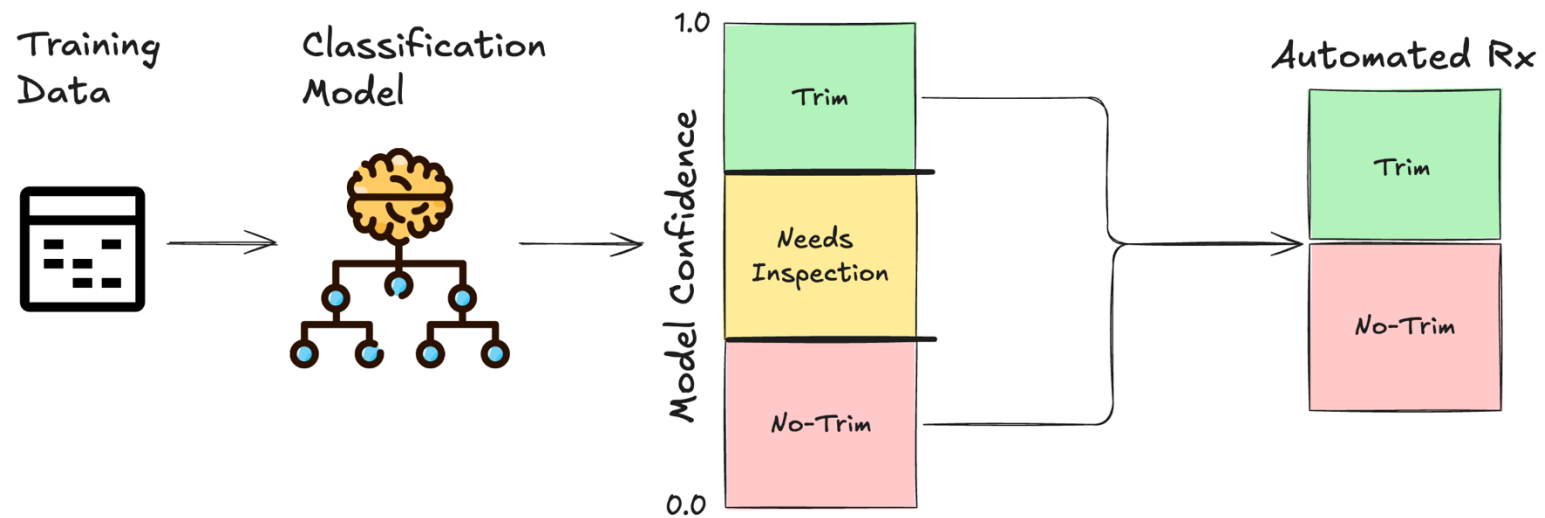
$$splits = \{(W_1, V_1), \dots, (W_n, V_n)\}$$

- Train models M_i on each training set W_i and predict on each test set V_i to get confidence value sets C_i
- Union confidence values $C = \bigcup_i C_i$ and compute upper and lower confidence bounds (lb_j, ub_j) for each percentile pair (l_j, u_j)
- For each confidence bound (lb_j, ub_j) and model M_i , compute the cost $\beta_{i,j}$ of using the model on test set V_i with the confidence bounds
- The optimal bounds are (lb_k, ub_k) where $k = \operatorname{argmin}_j(\phi(\{\beta_{i,j} \mid 1 \leq i \leq n\}))$ and ϕ is the configured aggregation function, such as max or mean

Model Training

Due to different regulations, we train 4 different models based on risk & circuit type:

- HFRA Transmission
- Non-HFRA Transmission
- HFRA Distribution
- Non-HFRA Distribution



Model Summary - Transmission

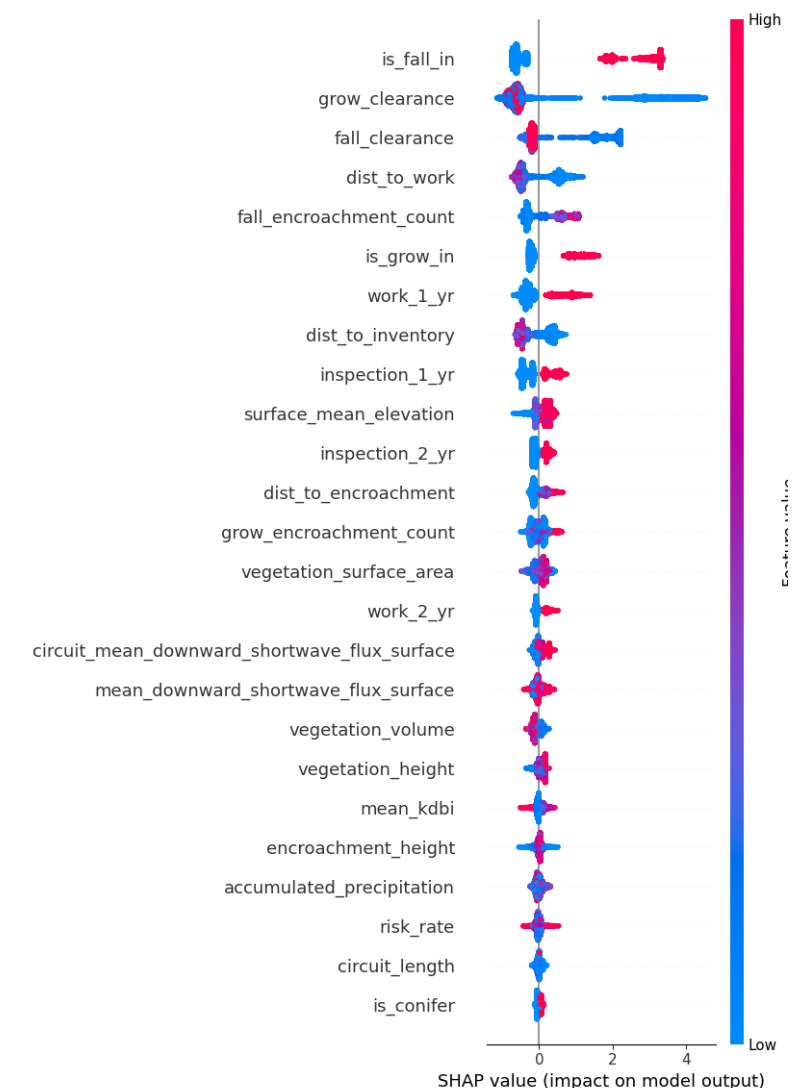
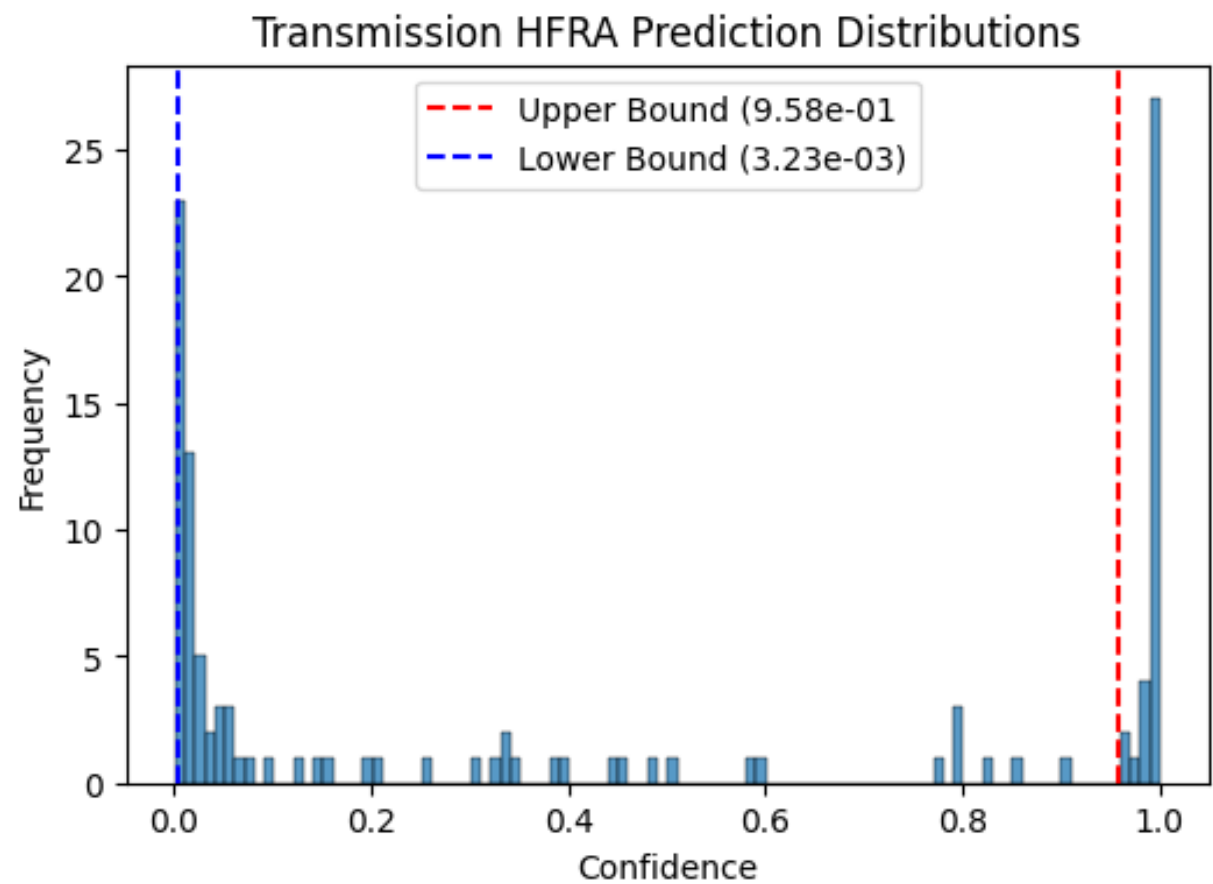
The Rx model is a **binary classification** model, with a customized risk-weighted cost function to determine confidence thresholds for automation.

We trained 3 different model types of increasing complexity. This was done for both **HFRA**, and **non-HFRA**.

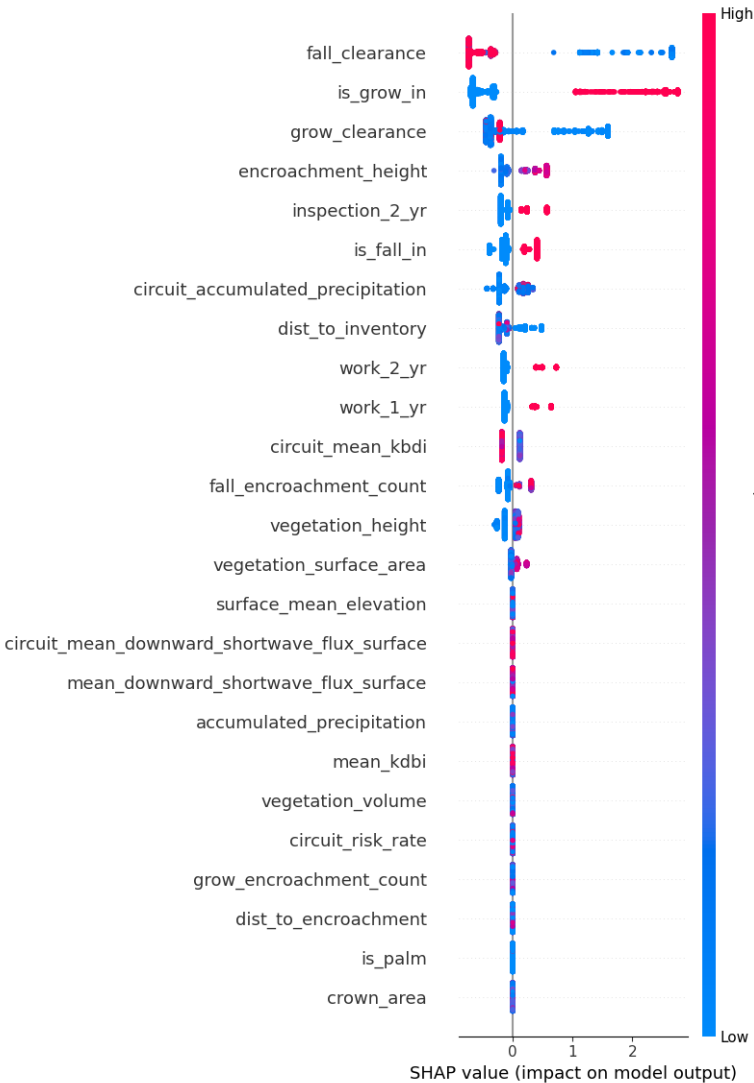
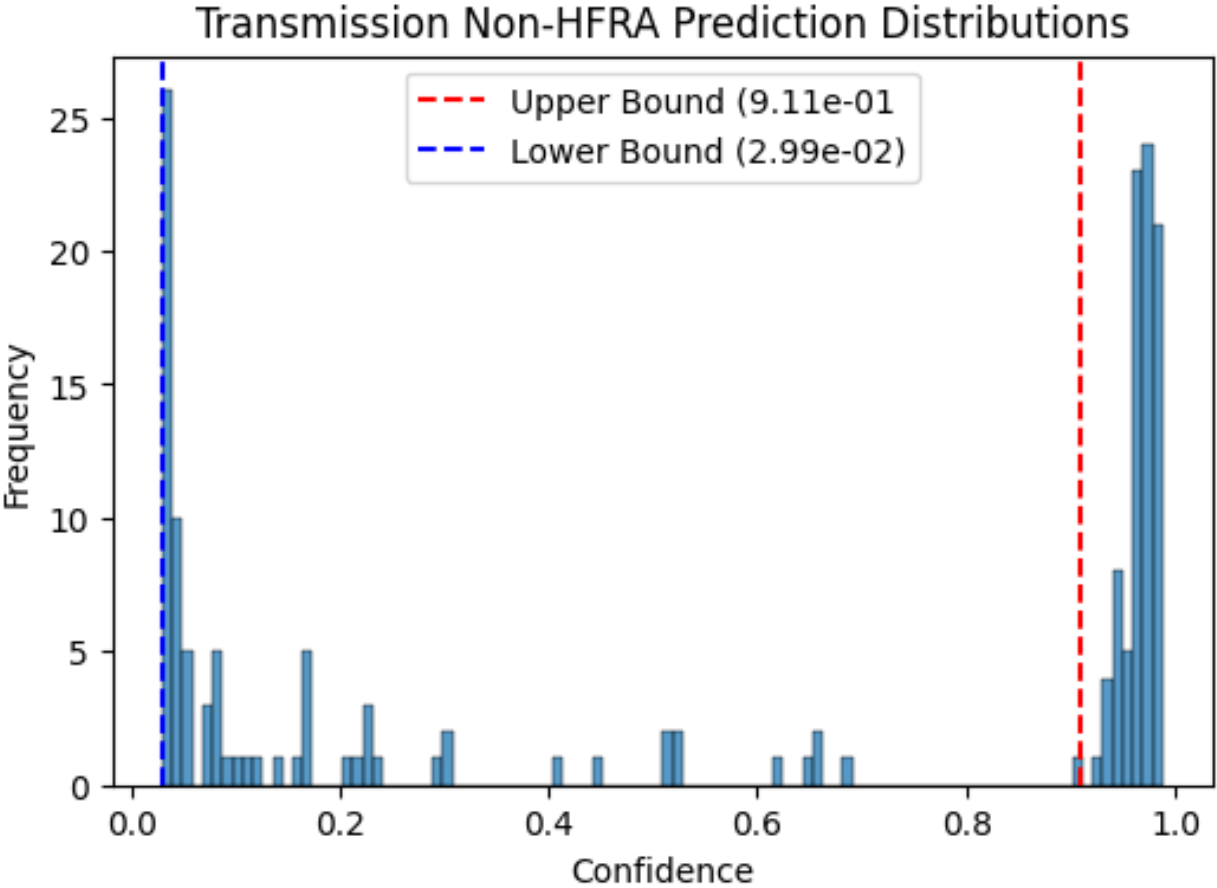
- Logistic Regression
- Linear Support Vector Machine (SVC)
- Gradient Boosted Trees (XGBoost)

Model Name	Automated %	Model Cost	Automated Precision	Automated Recall
XGBoost HFRA	26 – 40%	121	1	1
XGBoost non-HFRA	46 – 56%	88	1	1
LinearSVC HFRA	25 – 38%	123	1	1
LinearSVC non-HFRA	33 – 46%	114	0.97	1
LogReg HFRA	39 – 56%	111	1	.97
LogReg non-HFRA	37 – 49%	103	1	1

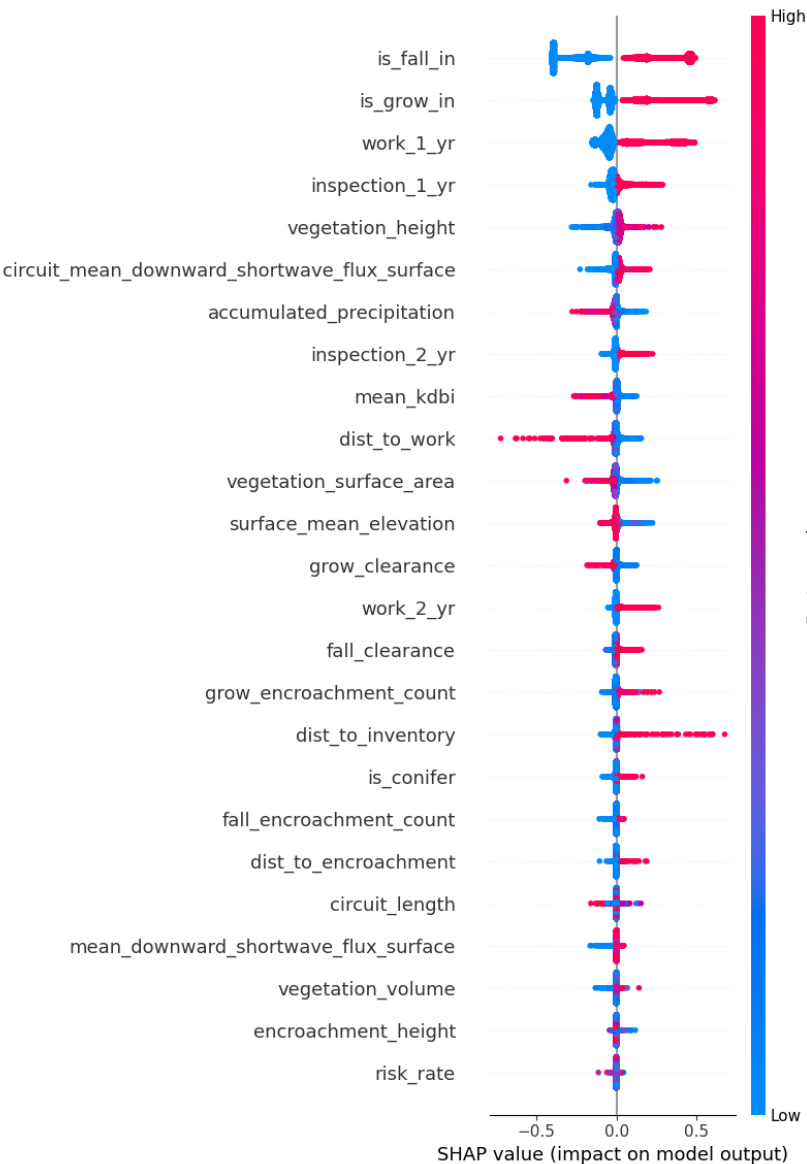
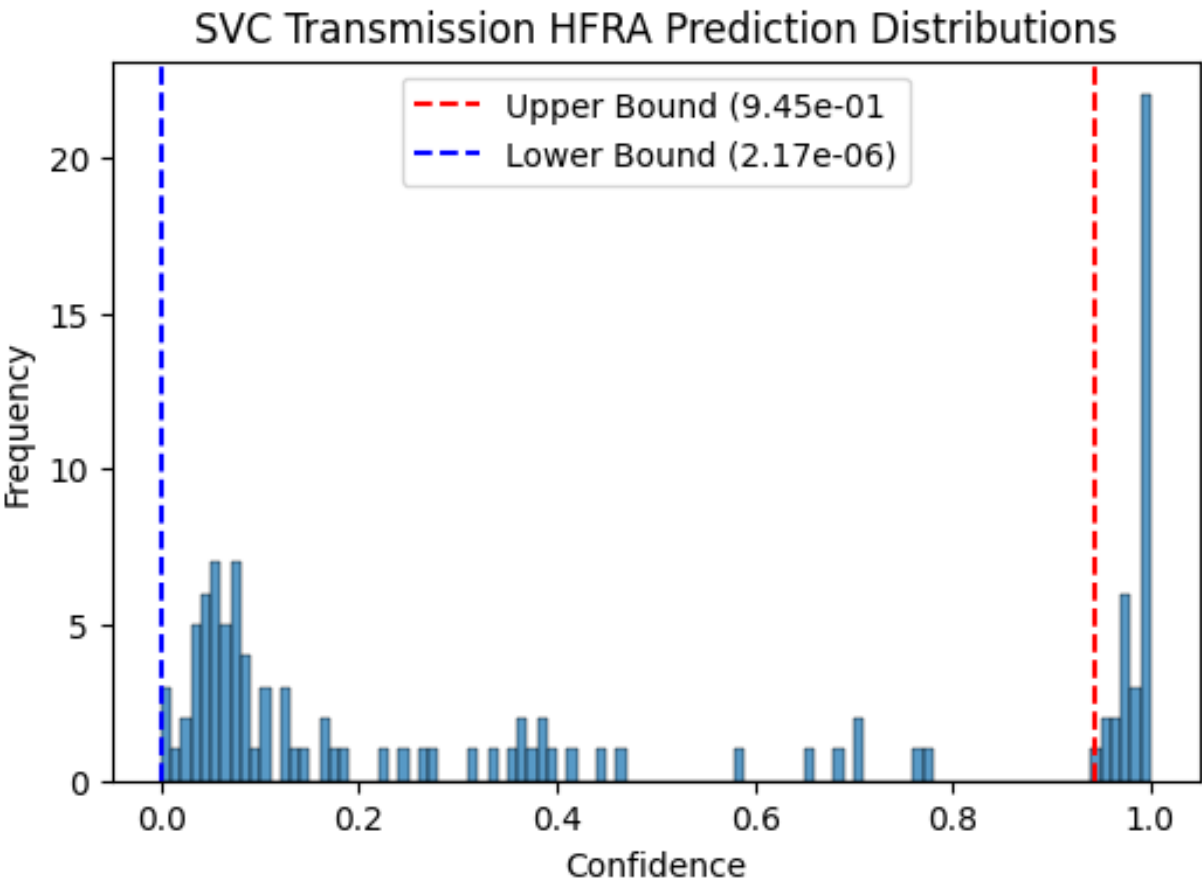
Model Analysis: Transmission XGBoost HFRA



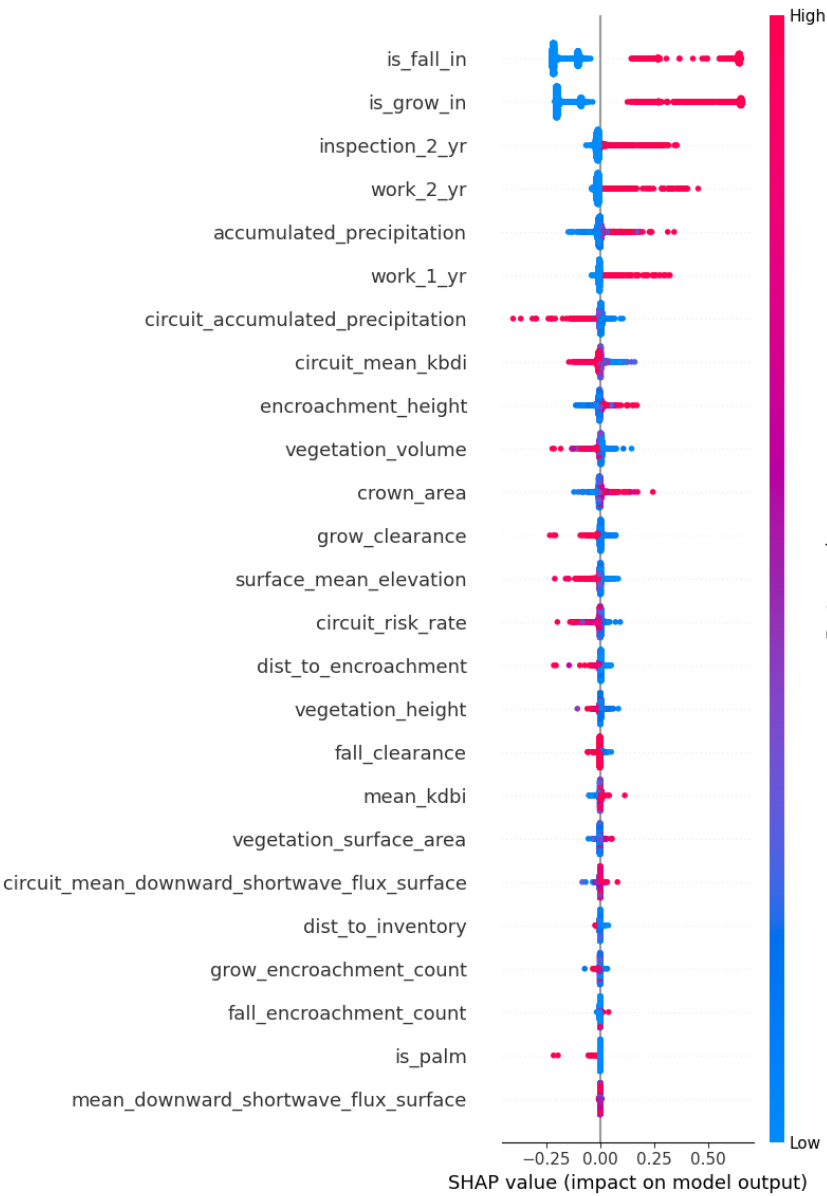
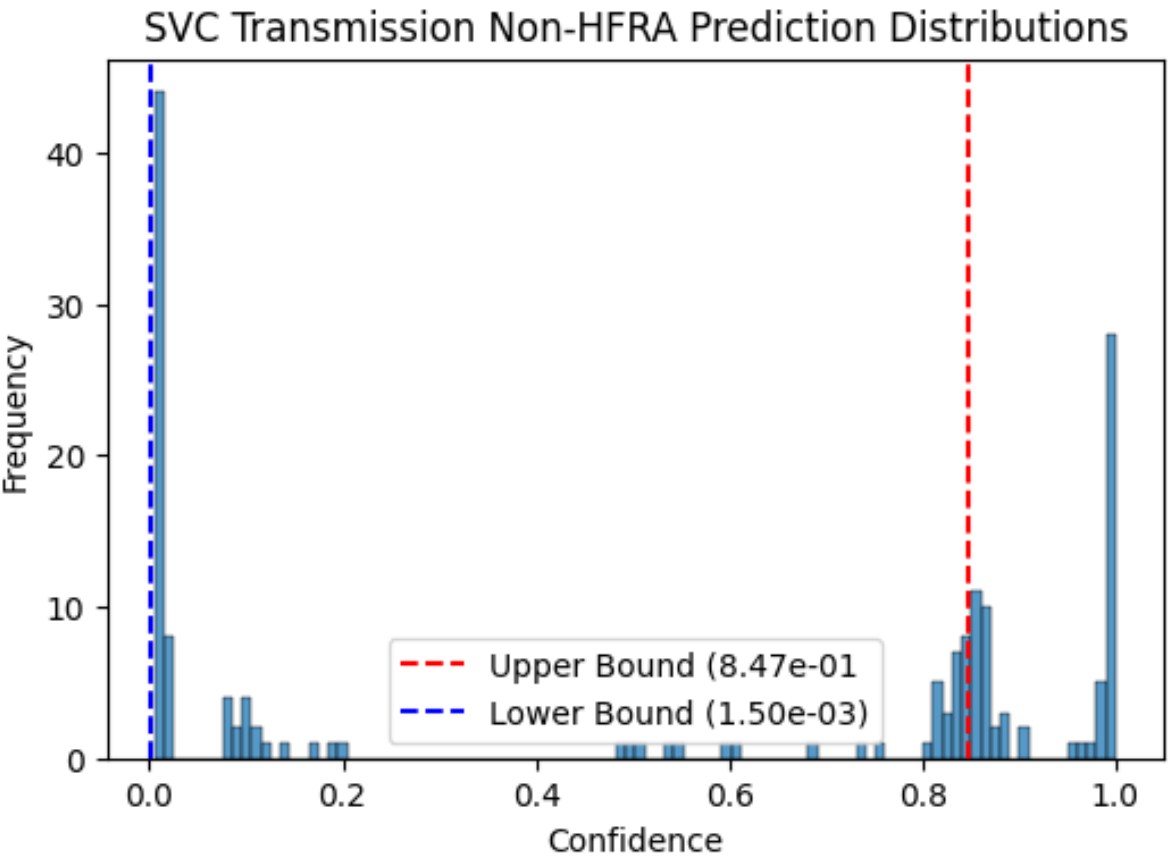
Model Analysis: Transmission XGBoost non-HFRA



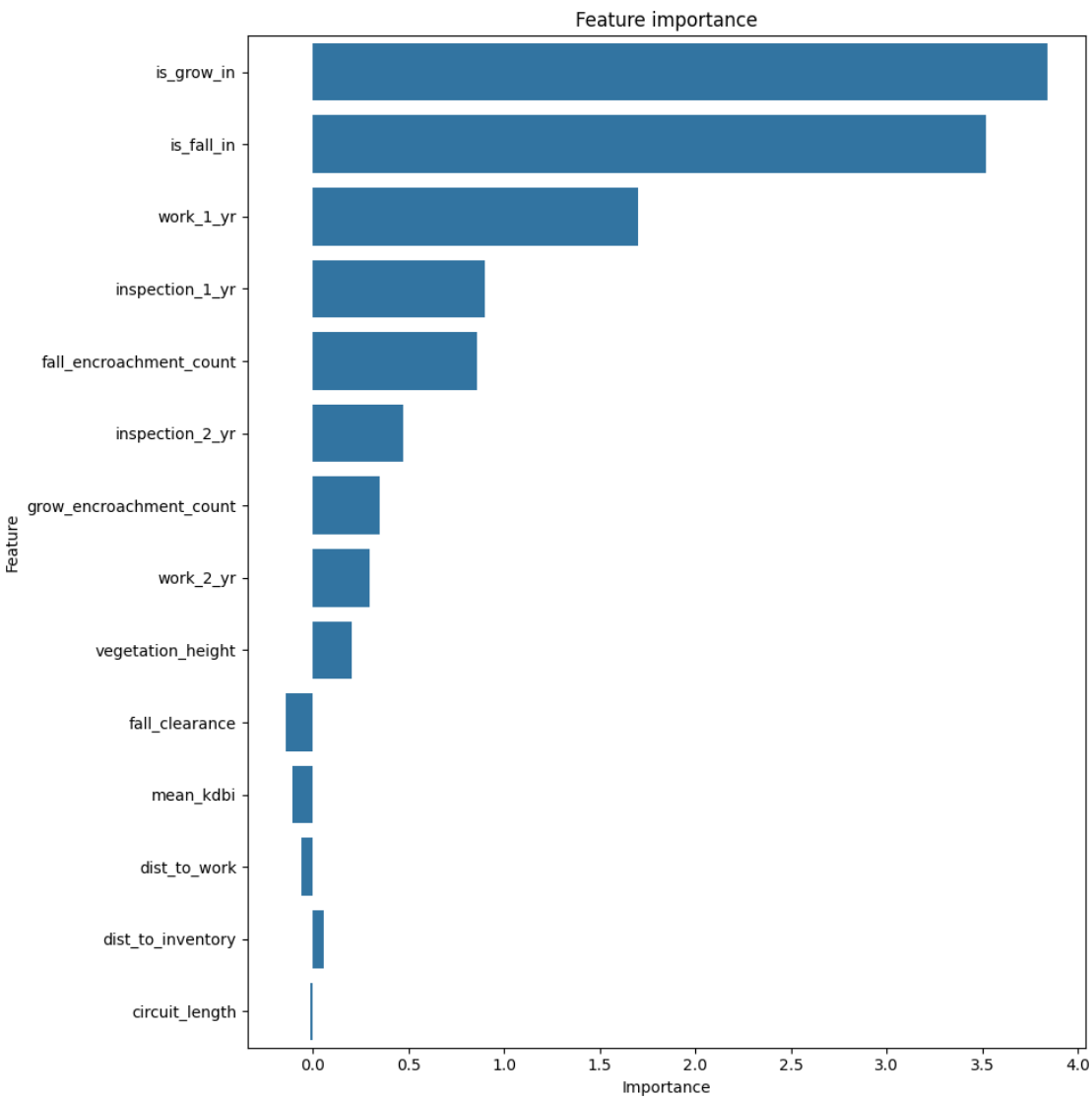
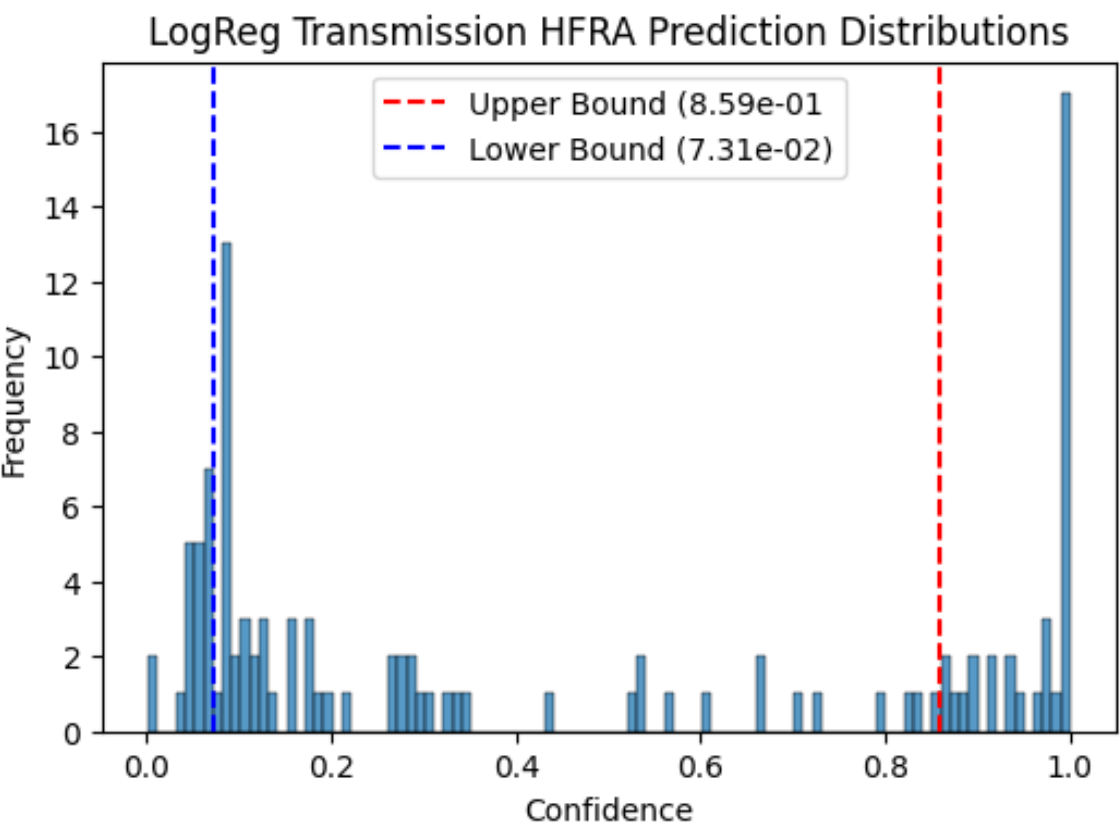
Model Analysis: Transmission Linear SVC HFRA



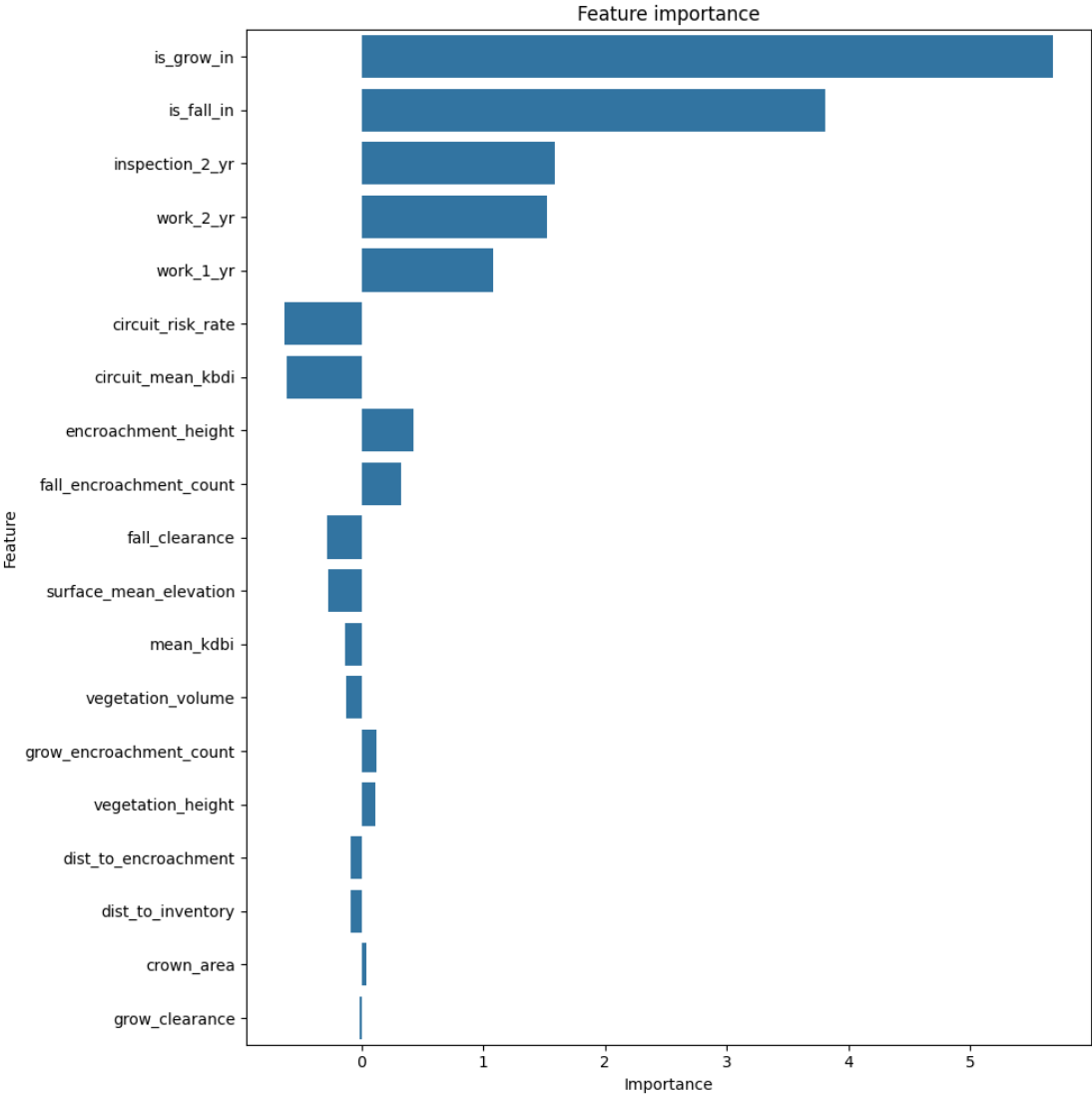
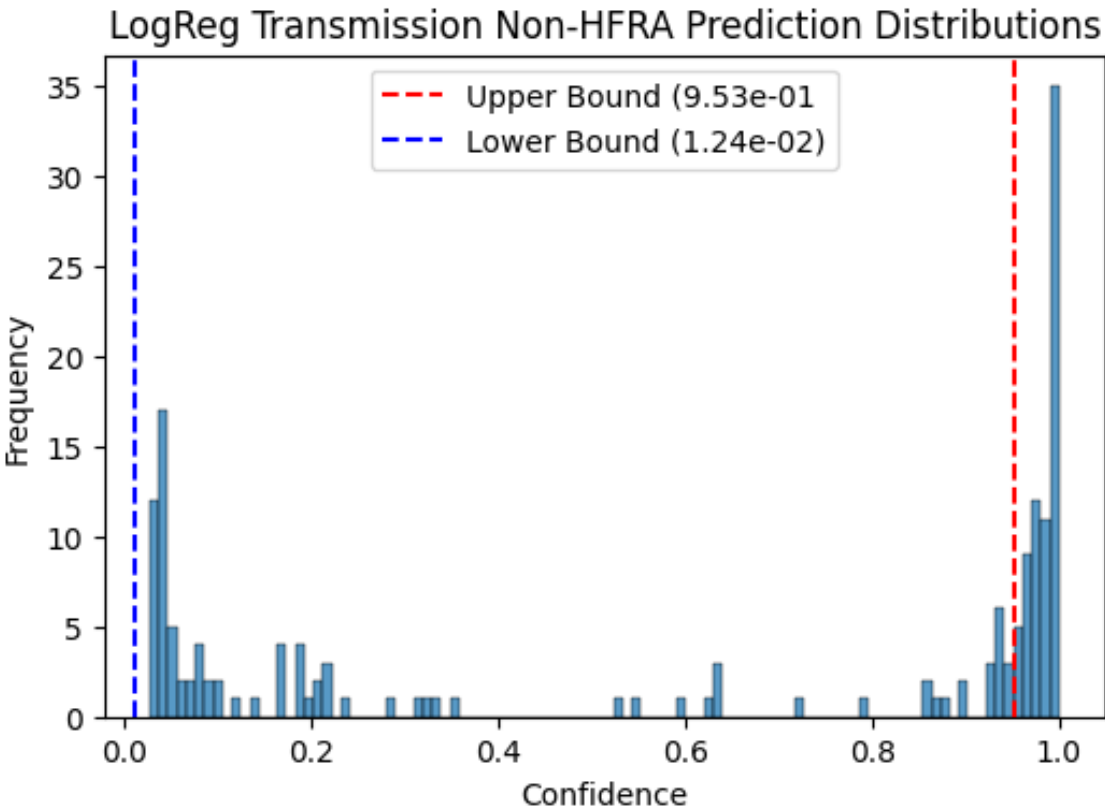
Model Analysis: Transmission Linear SVC non-HFRA



Model Analysis: Transmission LogReg HFRA



Model Analysis: Transmission LogReg non-HFRA



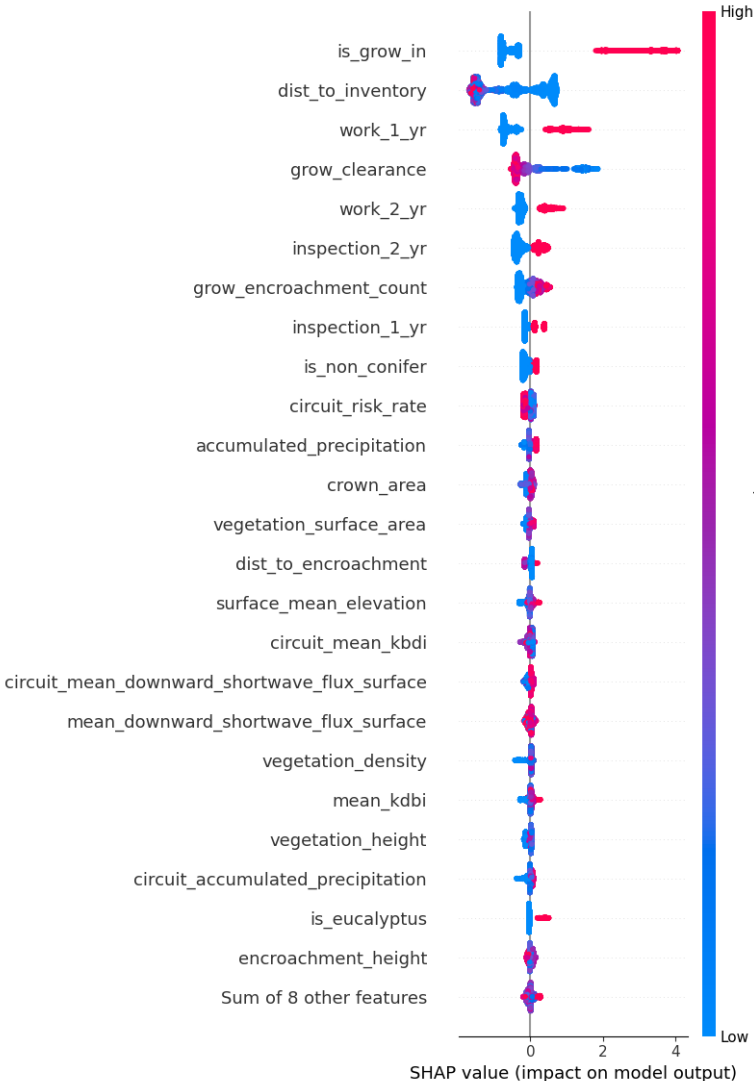
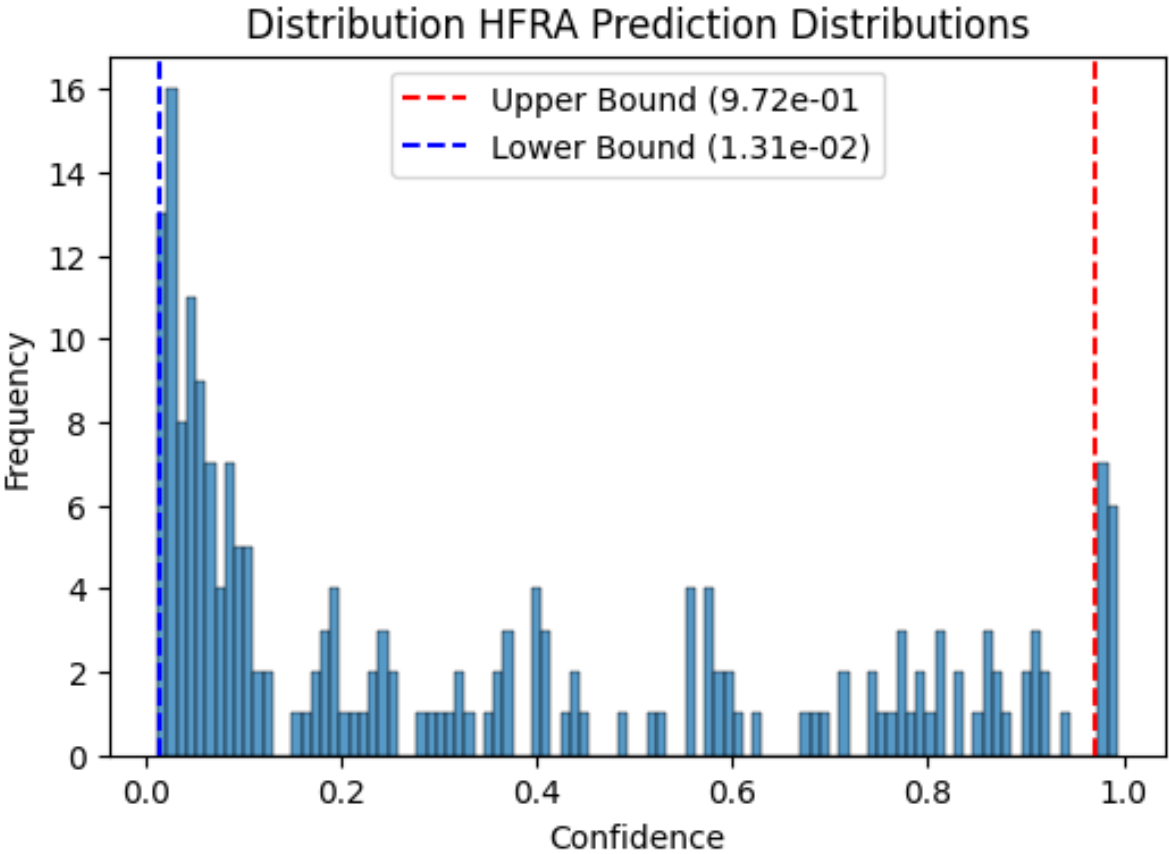
Model Summary - Distribution

For distribution, **XGBoost** models were the most effective. Two different model configurations were tested for both **HFRA**, and **non-HFRA**.

Model Name	Automated %	Model Cost	Automated Precision	Automated Recall
XGBoost Non HFRA v1	12 – 20%	50.94	1	1
XGBoost HFRA v1	5 – 11%	75.75	1	1
XGBoost Non-HFRA v2	22 – 33%	54.94	1	0.77
XGBoost HFRA v2	11 – 19%	88.03	1	0.86

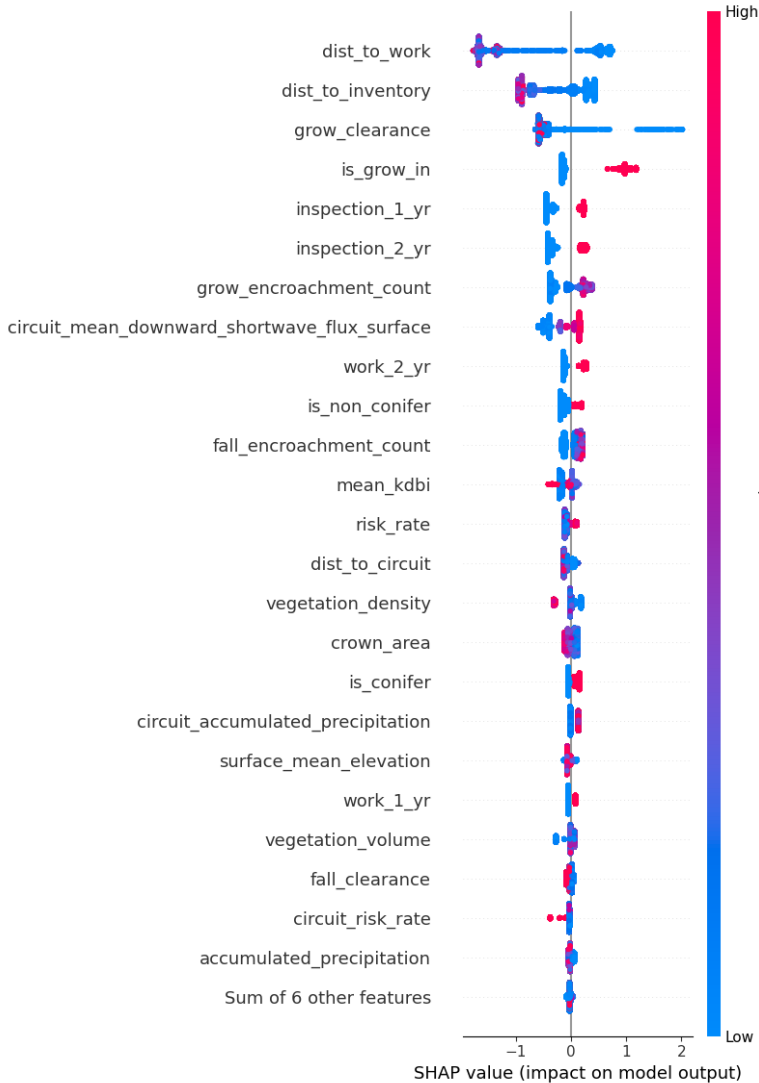
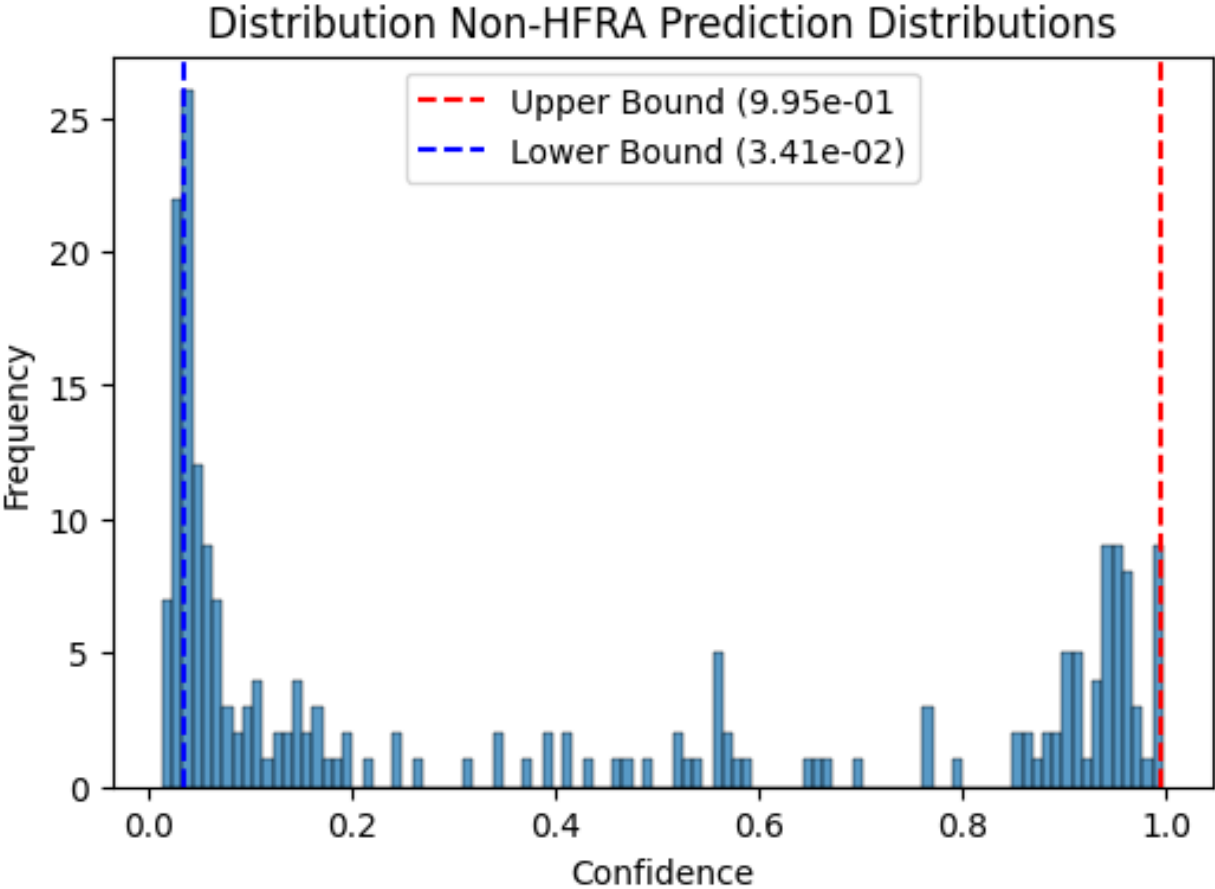
Model Analysis: Distribution XGBoost HFRA

...



Model Analysis: Distribution XGBoost non-HFRA

...



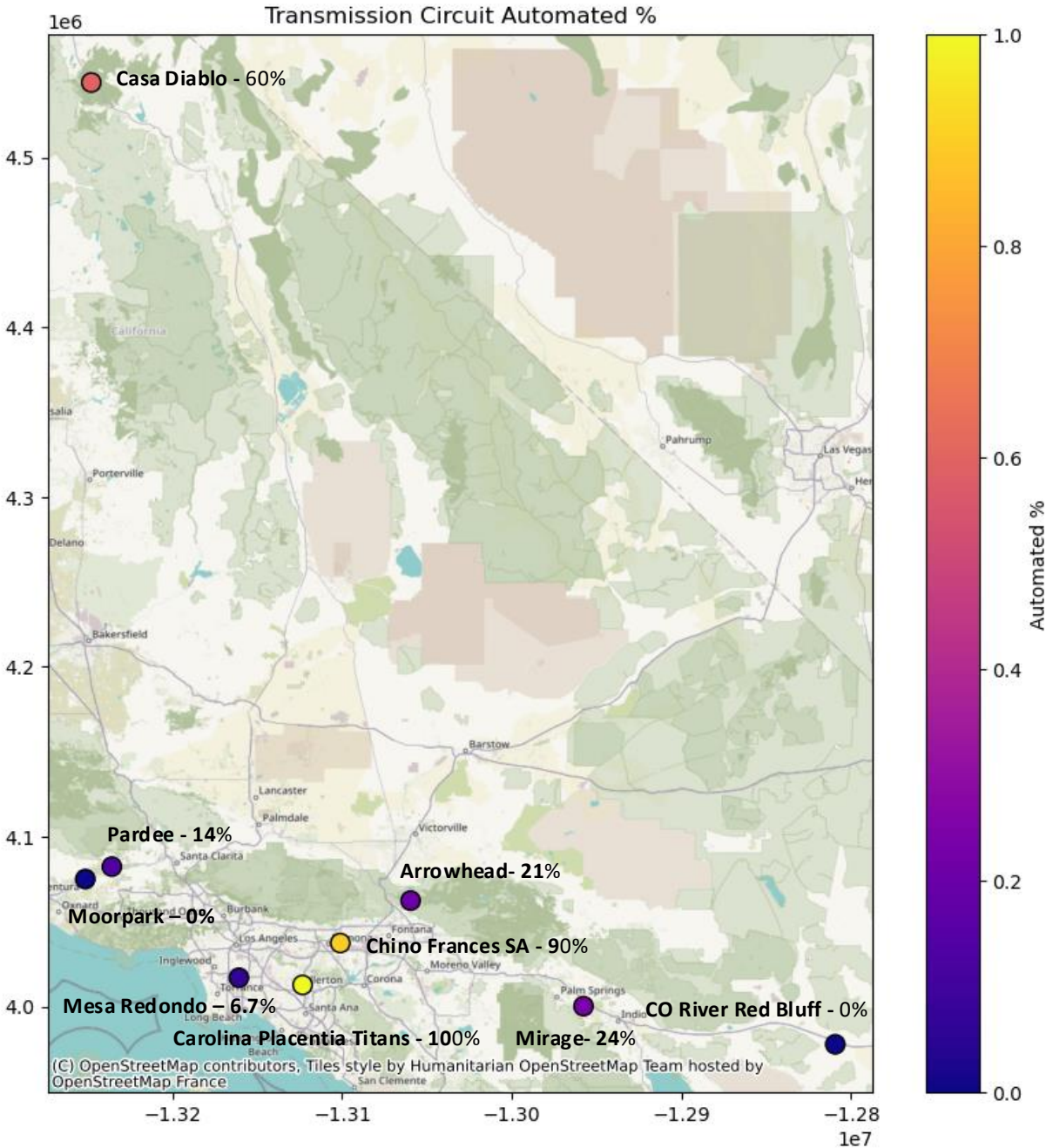
Circuit Summary – Transmission - XGBoost

The **XGBoost** models were applied to the **transmission circuits** in the PoC extension scope. The XGBoost models achieved the best overall results.

Circuit ID	Automated %	Total Field Observations	Automated Predictions	Automated Precision	Automated Recall
ET-00776	14%	22	3	1	1
ET-00886	60%	45	27	1	1
ET-01716	21%	28	6	1	1
ET-00213	6.7%	30	2	1	1
ET-00861	90%	50	45	1	1
ET-01198	24%	42	10	1	1
ET-01694	100%	29	29	1	1
ET-00775	0%	23	0	1	1
ET-00943	0%	10	0	1	1

Circuit Analysis: Transmission

Circuit ID	Name	Automated %	Total Field Observations	Automated Predictions
ET-00776	Pardee-SC	14%	22	3
ET-00886	Casa Diablo	60%	45	27
ET-01716	Arrowhead	21%	28	6
ET-00213	Mesa-Redondo	6.7%	30	2
ET-00861	Chino	90%	50	45
ET-01198	Mirage	24%	42	10
ET-01694	Villa Park	100%	29	29
ET-00775	Moorpark	0%	23	0
ET-00943	CO Red Bluff	0%	10	0



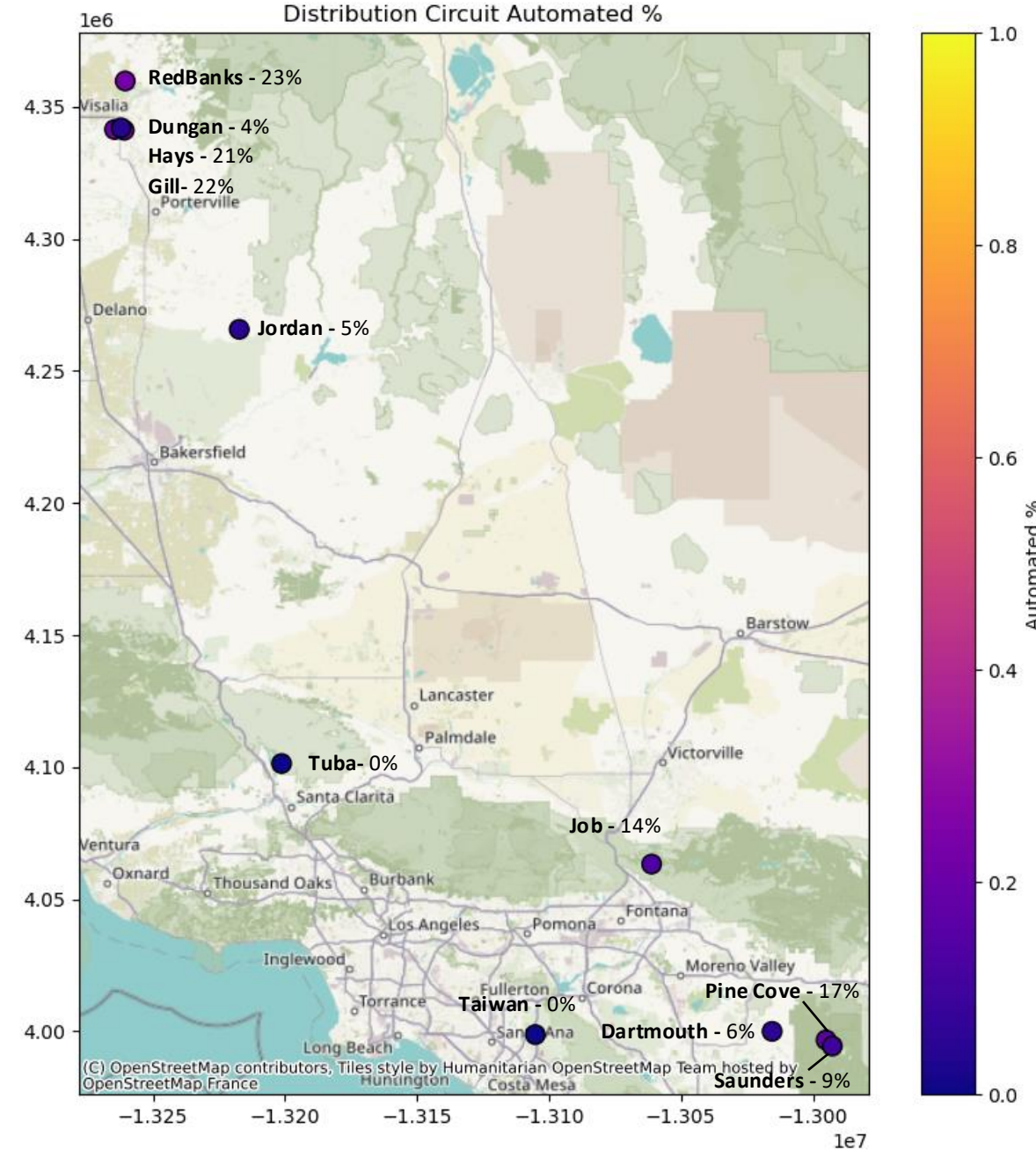
Circuit summary – Distribution - XGBoost

The top models were applied to the **distribution** circuits in the PoC extension scope:

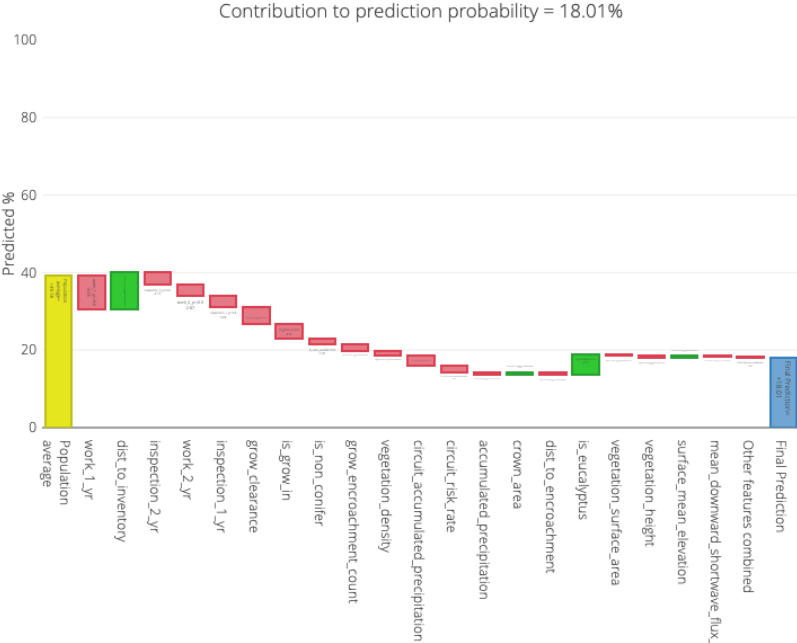
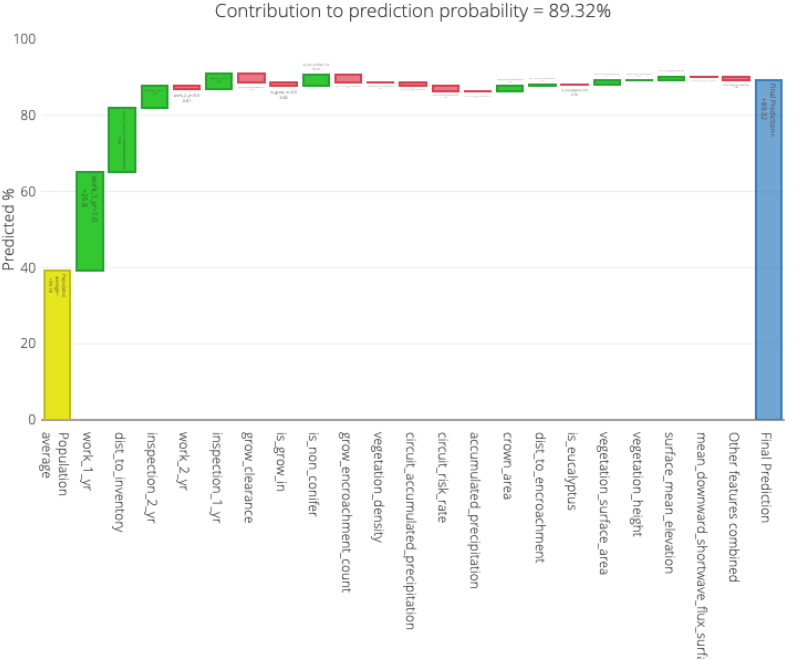
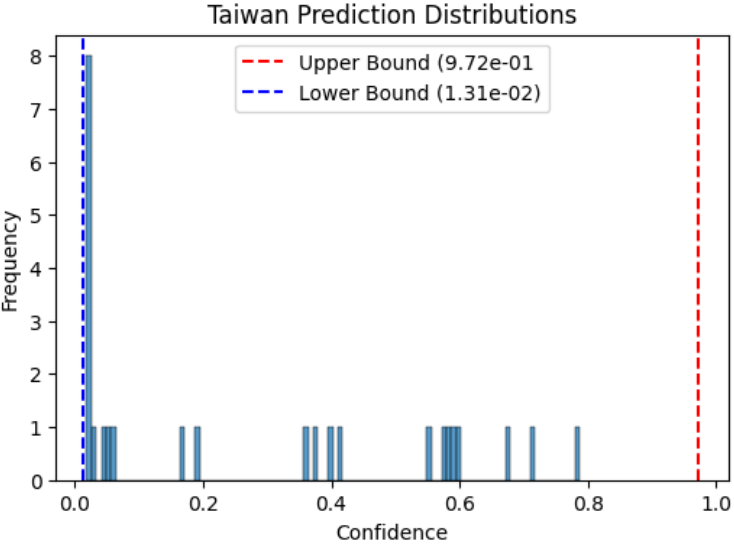
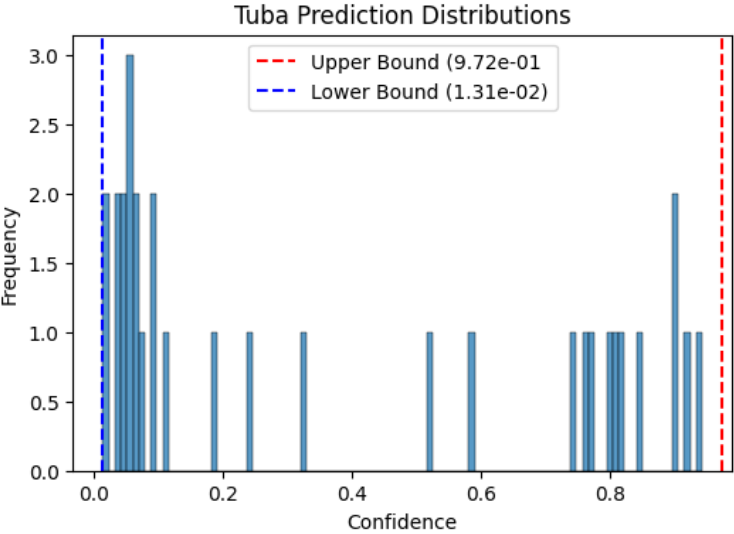
Circuit ID	Automated %	Total Field Observations	Automated Predictions	Automated Precision	Automated Recall
ED-14750	23%	48	11	1	1
ED-07240	22%	50	11	1	1
ED-08240	21%	53	11	1	1
ED-14097	17%	30	5	1	1
ED-09275	14%	28	4	1	1
ED-15922	9%	22	2	1	1
ED-04693	6%	32	2	1	1
ED-09320	5%	40	2	1	1
ED-05400	4%	52	2	1	1
ED-17487	0%	27	0	1	1
ED-18243	0%	31	0	1	1

Circuit Analysis: Distribution

Circuit ID	Name	Automated %	Total Field Observations	Automated Predictions
ED-14750	Redbanks	23%	48	11
ED-07240	Gill	22%	50	11
ED-08240	Hays	21%	53	11
ED-14097	Pine Cove	17%	30	5
ED-09275	Job	14%	28	4
ED-15922	Saunders	9%	22	2
ED-04693	Dartmouth	6%	32	2
ED-09320	Jordan	5%	40	2
ED-05400	Dungan	4%	52	2
ED-17487	Taiwan	0%	27	0
ED-18243	Tuba	0%	31	0



Circuit Analysis: Distribution

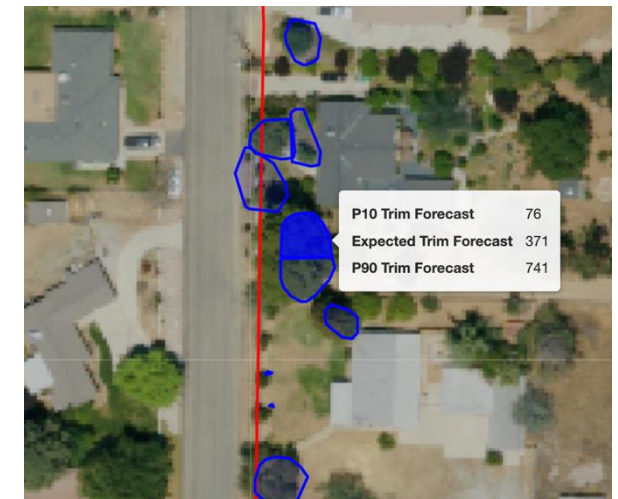
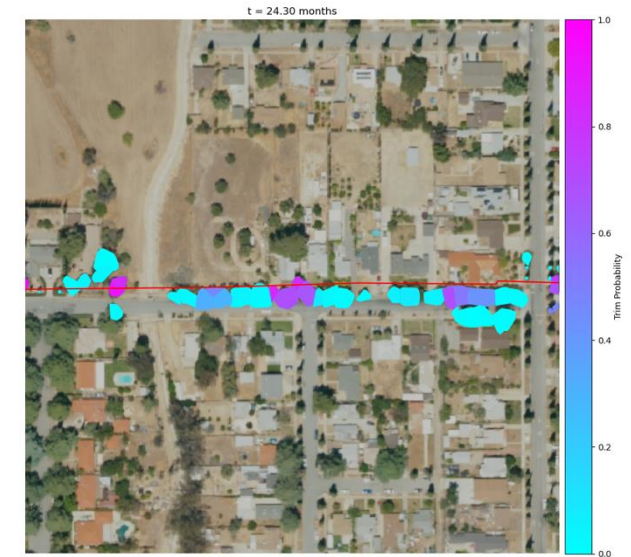


Trim Forecasting

Trim Forecast Value

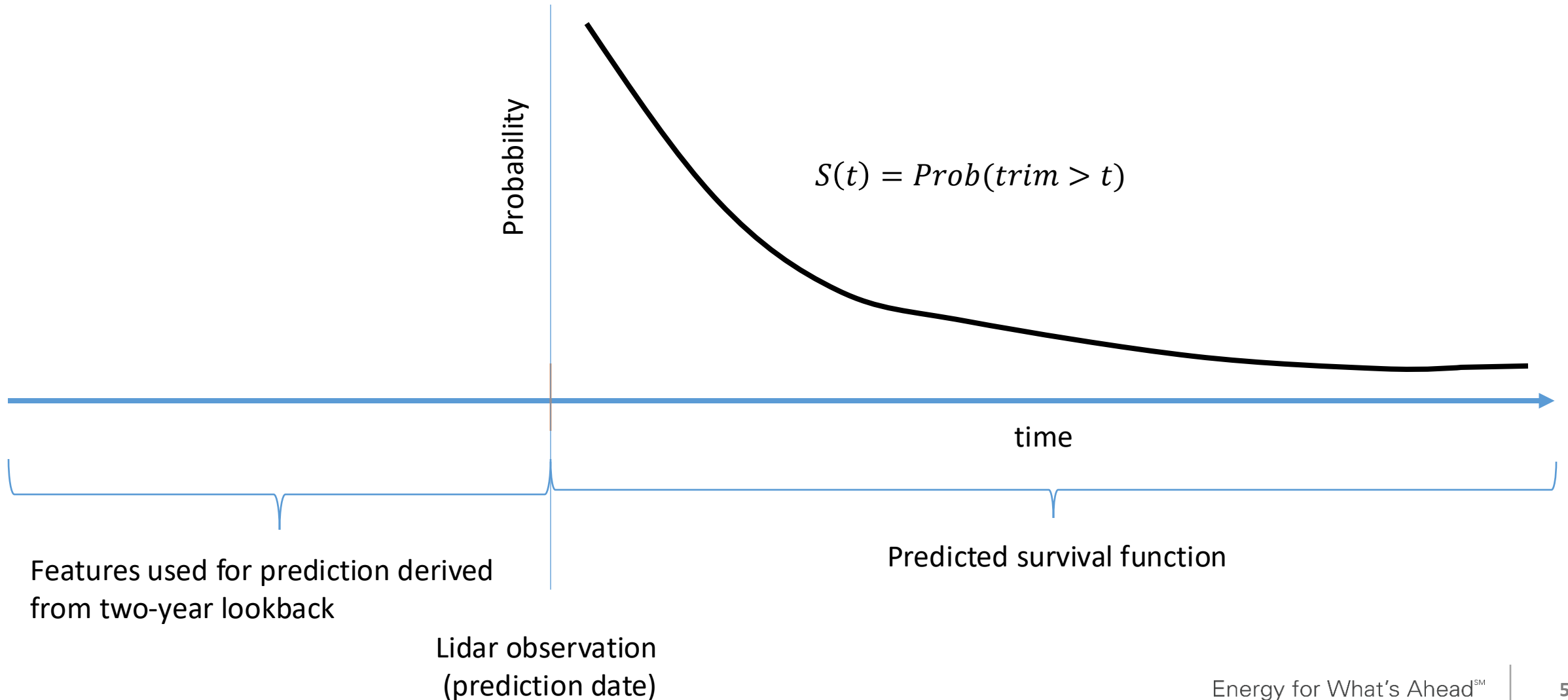
The trim forecast model describes the probability of maintenance over a long time horizon, which can be applied to long range business planning

- Understanding how the probability of trims evolves over a long time horizon time unlocks or informs many use cases
 - Work planning
 - Risk-driven data refresh planning
 - Economic analyses
 - E.g. targeting expensive-to-maintain trees for removal
- Following previous SCE research, the extended PoC focuses on forecasting future work volume



Trim Forecast Modeling Problem

The trim forecast model predicts the future survival function for each tree



Leveraging the Survival Function

The survival function allows for tree-level analyses and, with a probability threshold, aggregate predictions

Given probability threshold β and a time t and letting

$$\delta_i(t) = \begin{cases} 1, & S_i(t) < \beta \\ 0, & S_i(t) \geq \beta \end{cases}$$

be the trim threshold indicator function for the i -th tree, the number of trims becomes

$$\text{trims required by time } t = \sum_{i \in \text{trees}} \delta_i(t)$$

The threshold β can be derived during training.

Modeling Data

The modeling data consists of all circuits in Extended PoC scope with at least three lidar observations

Inclusion criterion

- Circuits with at least three lidar observations
 - Necessary to create a suitable training and testing sets

Data format

- The key targets for survival analysis is the survival_time and if the event was observed, work_required.
- If a tree is trimmed multiple times in the 3 year observation window, it will appear multiple times in the dataset.
 - This is a common phenomenon in survival analyses

geo_id	feature_1	...	survival_time	work_required
ab123	12.1		186	1
ab123	12.1		345	1
xy321	10.1	...	1098	0
...				

Assessment

A 66%/33% geospatial split is used to create train and test sets; MAPE is used to assess model performance

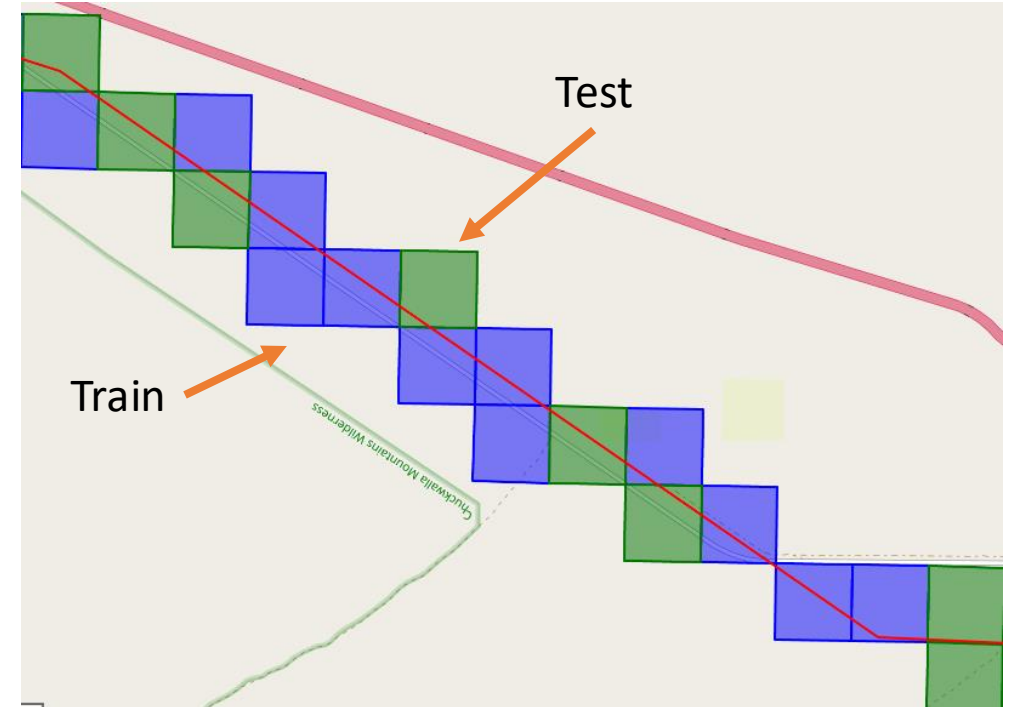
Train Test Split

- Each circuit is tiled with 1km x 1km geospatial boxes.
- These boxes are randomly split into training boxes and testing boxes
- Crowns are assigned to a split based on which type of box they intersect

Assessment

- Mean absolute percent error (MAPE) is used to assess how well the models can forecast work volume

$$mape = \frac{1}{n} \sum_{i=1}^n \left| \frac{a_i - f_i}{a_i} \right|$$



Modeling Frameworks

Two survival analysis frameworks are assessed: Cox regression and random survival forests

- Cox regression
 - Linear model
 - Relatively simple
 - See https://en.wikipedia.org/wiki/Proportional_hazards_model
- Random survival forest
 - Non-linear
 - More complex
 - See <https://arxiv.org/pdf/0811.1645>

The [scikit-survival](#) implementations are used.

Model Training

Training selects features and identifies the optimal probability threshold for work forecasting

- Training survival models is identical to training other tabular machine learning models, except the target is both the survival time and the observed outcome.
- Features selected using MRMR (minimum redundancy maximum relevance)
- Probability threshold derivation:
 - List all threshold candidates
 - Use cross validation to assess MAPE (mean absolute percentage error) for work forecasting at every candidate threshold
 - Choose the threshold which minimizes MAPE

Model Coordinates

Models are trained for transmission and distribution; separating by HFRA did not appear necessary

Transmission

model_name	model	experiment	threshold
Cox Regression	cox-regression-78e8480d-96b6-4410-8c7b-1fe2335c6af8	forecast-v4	0.642105
Survival Random Forest	survival-rf-4029c894-f584-49ca-ba38-0eb784bcc28f	forecast-v4	0.610526

Distribution

model_name	model	experiment	threshold
Cox Regression	cox-regression-0d500965-ccae-4219-a18e-c13ba6e701d7	forecast-distribution-v2	0.673684
Survival Random Forest	survival-rf-0bda1f40-4a83-4775-81b8-49d244452b7a	forecast-distribution-v2	0.515789

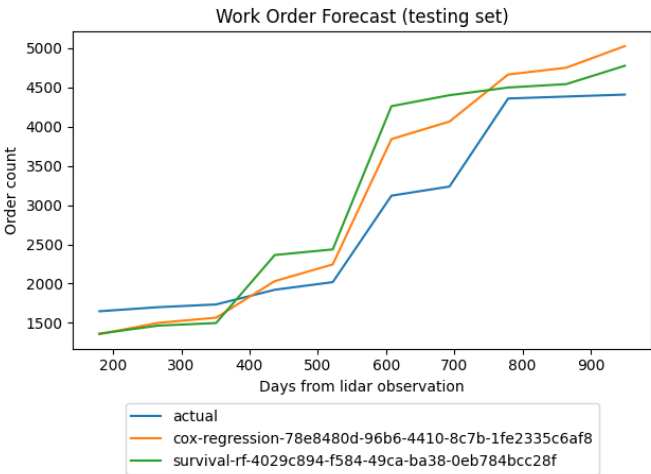
These models can be found in the Vertex AI experiment service

Model Performance

Model performance on test set indicates forecast models can be trained with less than 20% MAPE

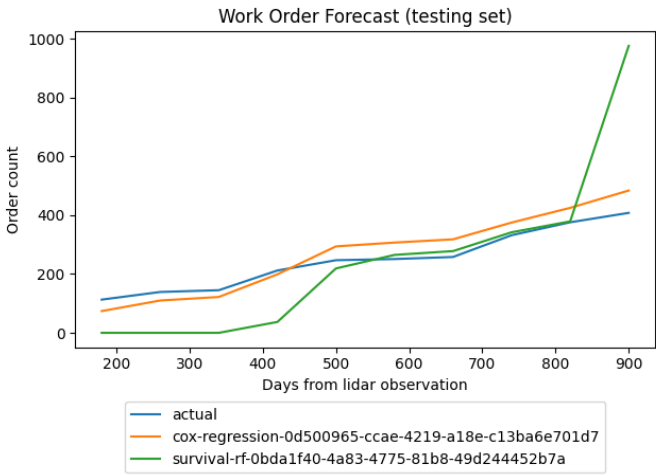
Transmission

model_name	mape	interval
Cox Regression	0.132905	[0.12, 0.14]
Survival Random Forest	0.175295	[0.17, 0.18]



Distribution

model_name	mape	interval
Cox Regression	0.18272	[0.13, 0.23]
Survival Random Forest	0.554304	[0.53, 0.57]



Model Performance

Forecast model performance on individual circuits.

Transmission

Circuit ID	Circuit Name	Count	Model Name	MAPE
ET-00213	Mesa-Redondo	839	Cox Regression	55.8%
			Survival Random Forest	58.6%
ET-00775	Moorpark	3544	Cox Regression	30.7%
			Survival Random Forest	35.3%
ET-00886	Casa Diablo	4930	Cox Regression	17.8%
			Survival Random Forest	35.3%
ET-00943	CO Red Bluff	381	Cox Regression	87.6%
			Survival Random Forest	21.8%
ET-01694	Villa Park	494	Cox Regression	42.4%
			Survival Random Forest	32.0%
ET-01912	Banning-Zanja	1594	Cox Regression	25.9%
			Survival Random Forest	30.8%

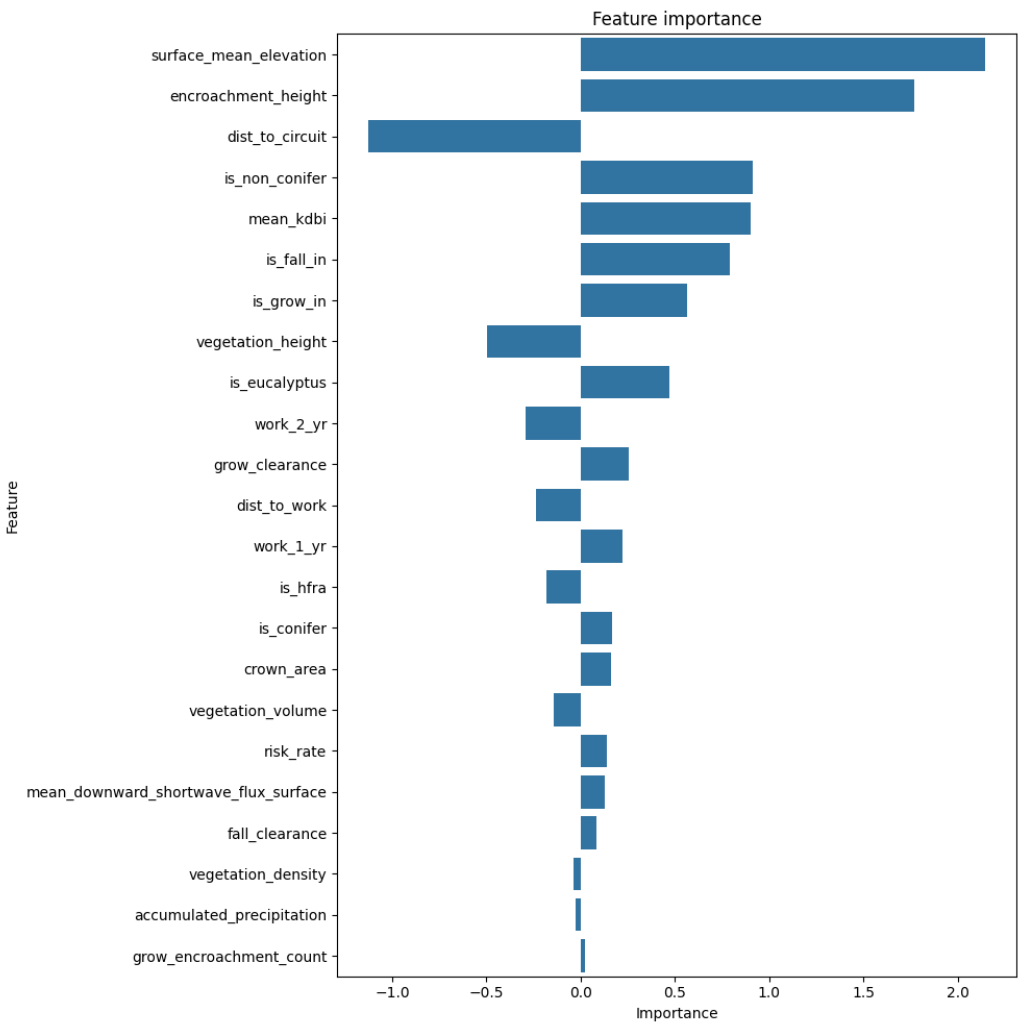
Distribution

Circuit ID		Count	Model Name	MAPE
ED-02360	Buckhorn	1162	Cox Regression	14.1%
			Survival Random Forest	40.4%
ED-17487	Taiwan	639	Cox Regression	59.8%
			Survival Random Forest	73.4%

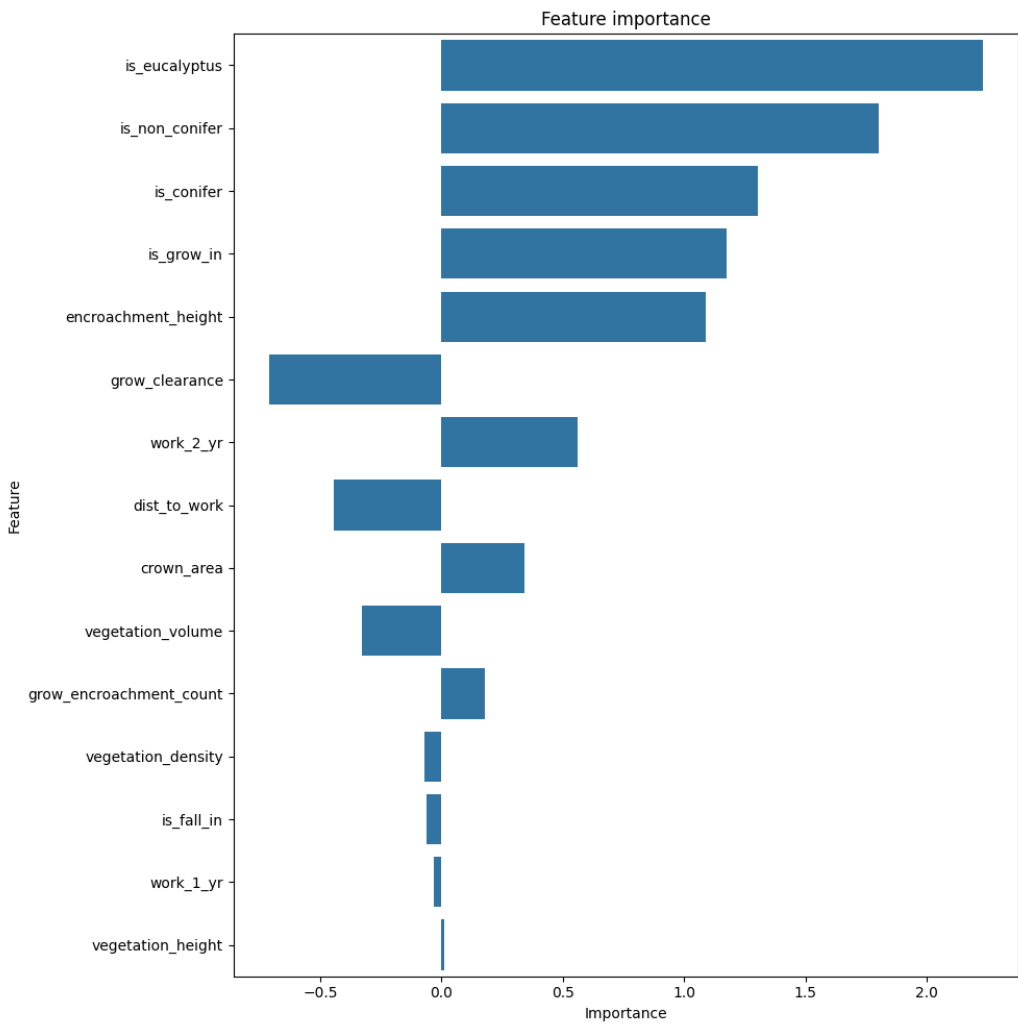
Feature importance

Feature importance plots for the Cox Regression models

Transmission



Distribution



Ortho Image Segmentation

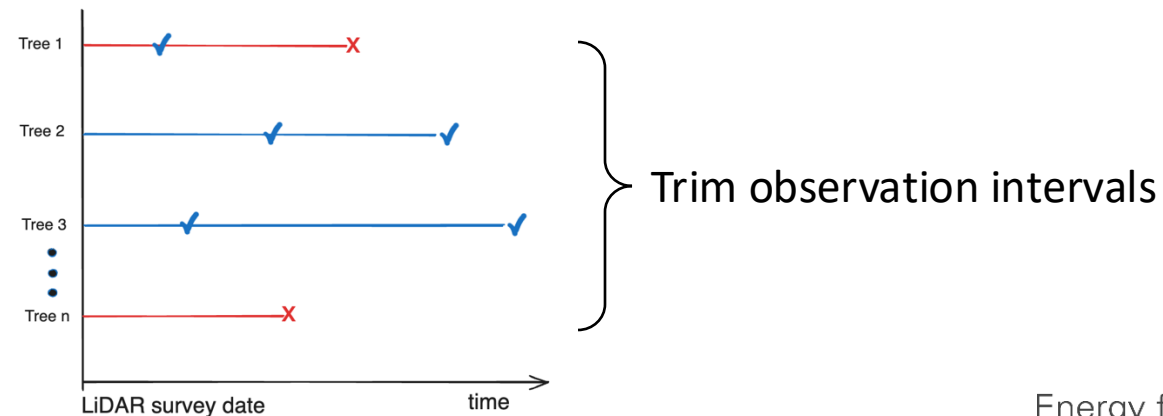
- ML_DEVELOPMENT.FORECAST_FEATURES
 - BigQuery table with fusion of crowns with many other data sources
- ML_DEVELOPMENT.FORECAST_TARGETS
 - BigQuery table with forecasting targets
- ML_DEVELOPMENT.TRIM_FORECAST_TEST_SUMMARY
 - BigQuery table with assessment results
- ML_DEVELOPMENT.TRIM_FORECAST_PREDICTIONS and ML_DEVELOPMENT.TRIM_FORECAST_D_PREDICTIONS
 - BigQuery tables with forecast model predictions

Trim Forecast Model - PoC

Trim Forecast Model

The trim forecast modeling problem is framed as a survival regression: predict the probability a tree will require a trim within a time interval

- A trim forecast prediction is an estimate of when a tree will require maintenance, accounting for inherent uncertainty (and risk)
- Forecasts can be driven by different strategies:
 - Growth models, that describe the evolution of tree dimension over time
 - Historical observations, from which growth patterns can be inferred
- Survival (or time-to event) analysis establishes a connection between covariates (features) and the time of an event
- Extract trim/no trim observation intervals from the inspection dataset, using date of LiDAR survey as the beginning of the interval



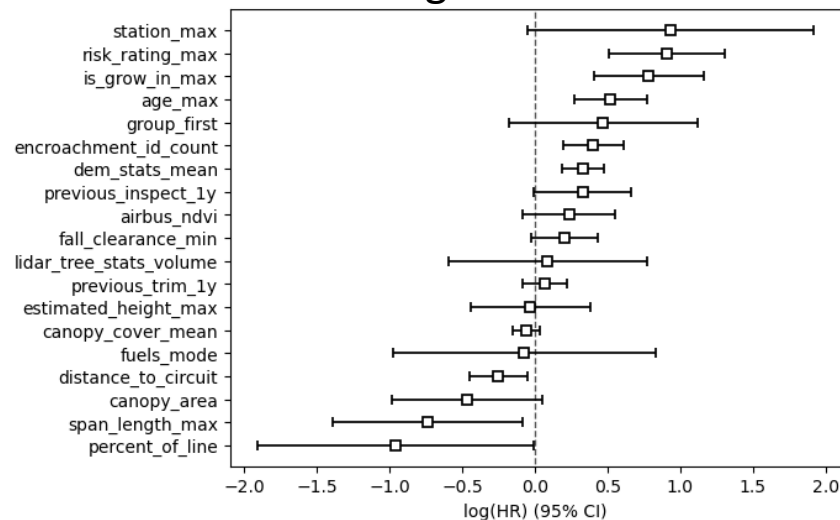


Trim Forecast Model

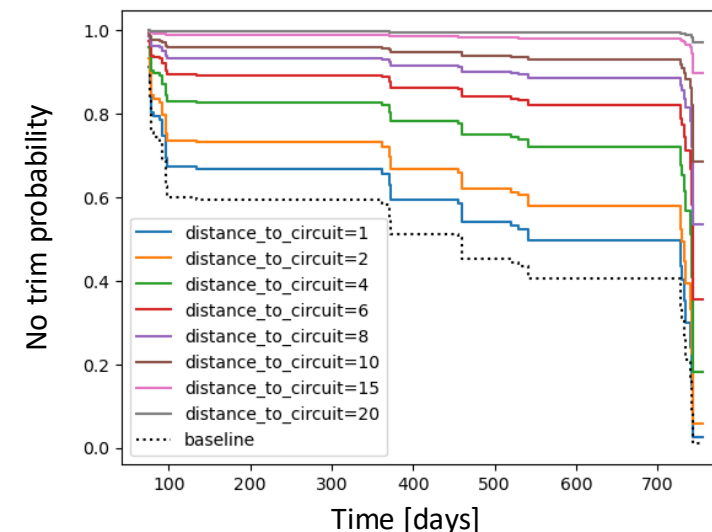
Model is trained on historical inspection data

- Multivariate Survival Analysis regression models explored: Cox Proportional Hazard Model¹
- Transmission training data
 - Banning-Zanja Feb 2022 (training)
 - Banning-Zanja Nov 2022 (test)
 - Banning-Zanja Nov 2023 (predict)
- Concordance index of 0.64 achieved on test dataset (ROC AUC measure)

Training Data coefficients



Partial dependance - distance

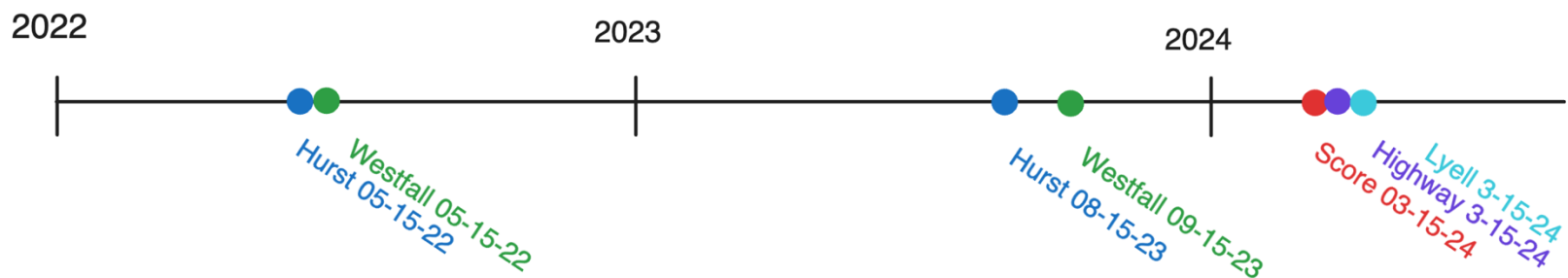


¹Davidson-Pilon, (2019). lifelines: survival analysis in Python. Journal of Open Source Software, 4(40), 1317, <https://doi.org/10.21105/joss.01317>

Trim Forecast Model

Model is trained on historical inspection data

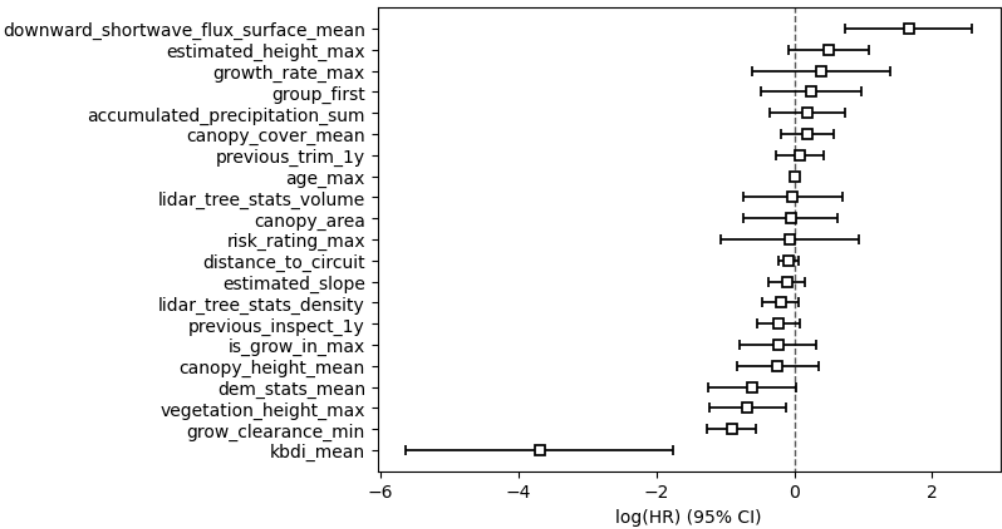
- Forecast model methodology applied to 2 HFRA distribution circuits



- Forecast model c-index on test dataset (ROC AUC measure)

Circuit	Training points	Test points	c-index
Hurst	476	163	0.553
Westfall	270	169	0.658

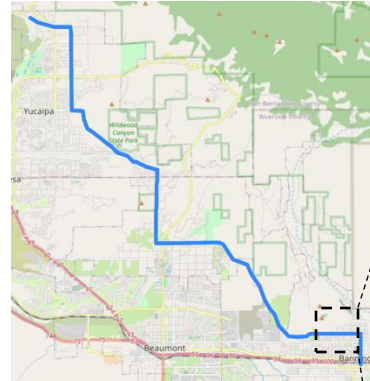
Dist. Model Coefficients



Trim Forecast Model

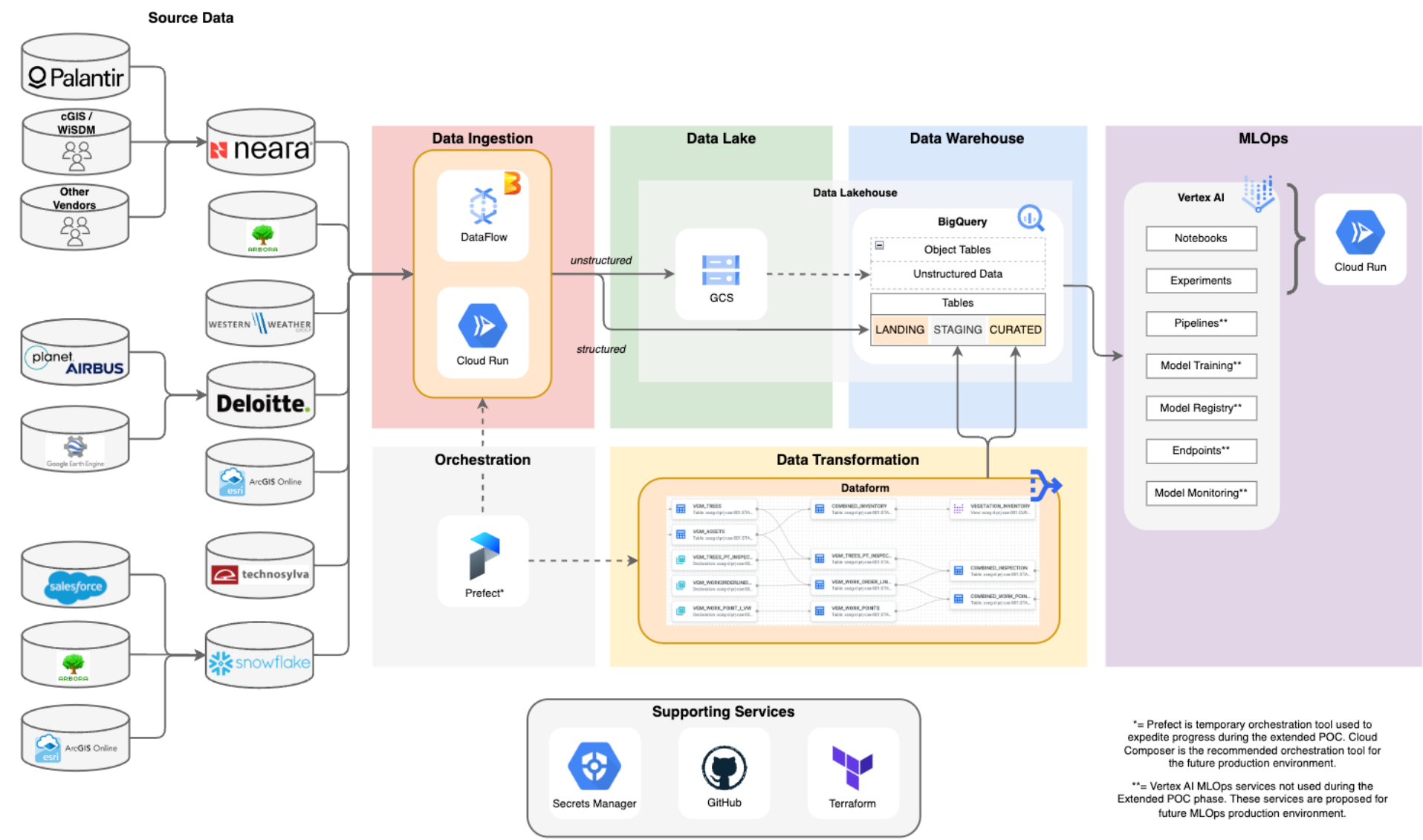
Prediction on BZ Nov 2023 LiDAR survey

- Forecast predictions give the probability of trim at different time intervals
- Expected values can also be generated for individual trees
- In addition to planning, evolution of risk can be used to assess data refresh, total cost of tree maintenance, planning scenarios

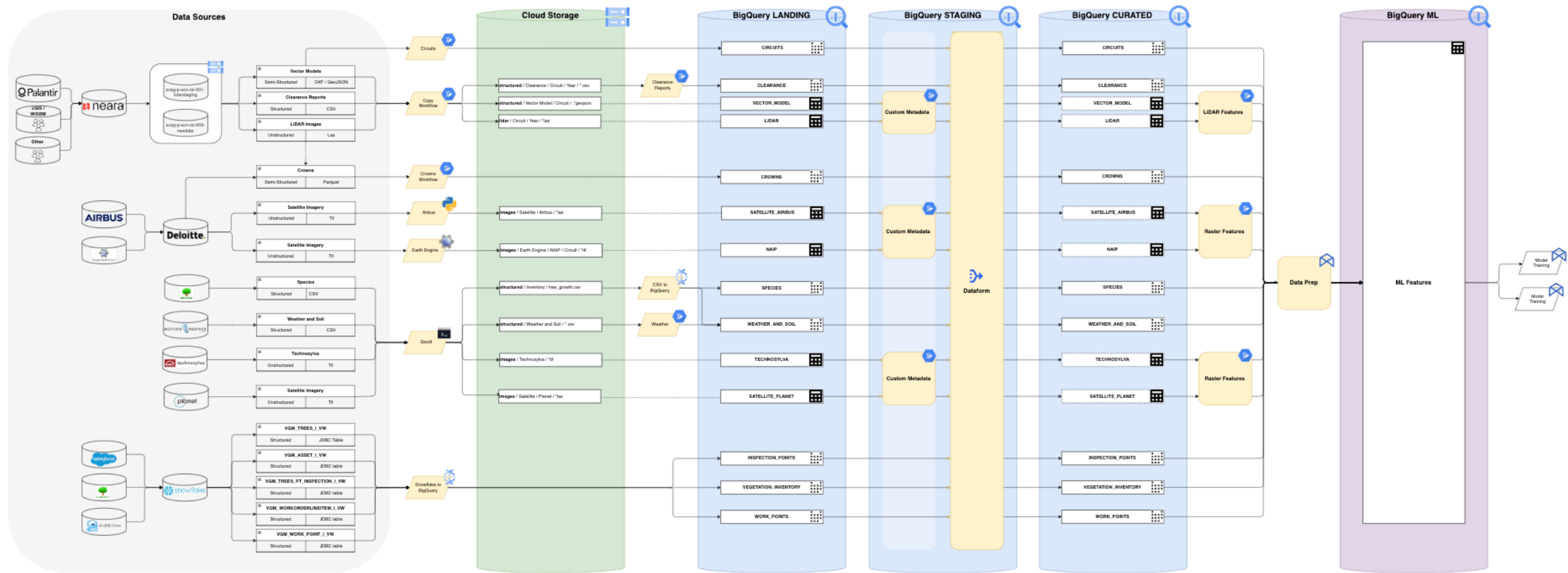


Appendix

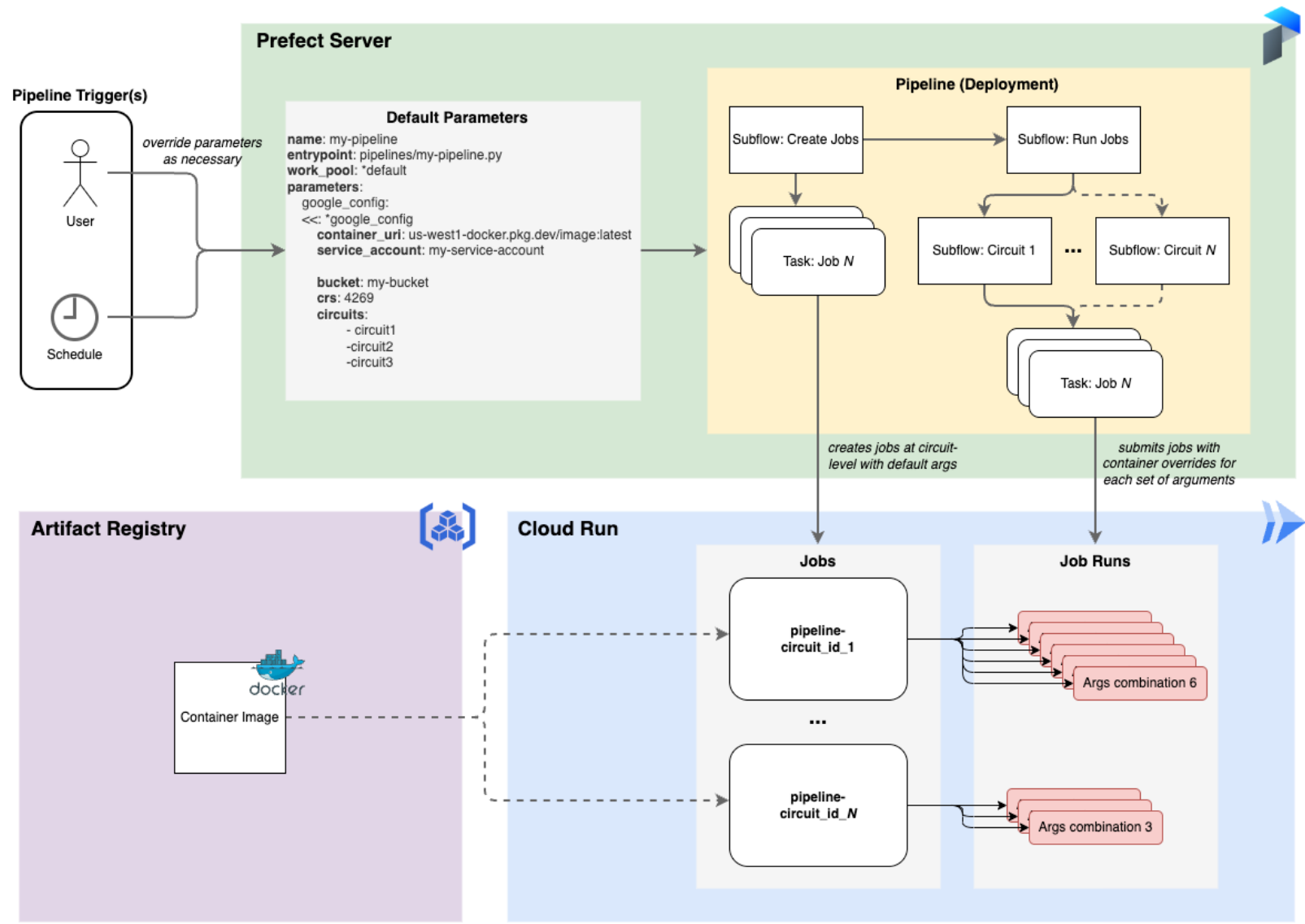
Vegetation AI Solutions Architecture



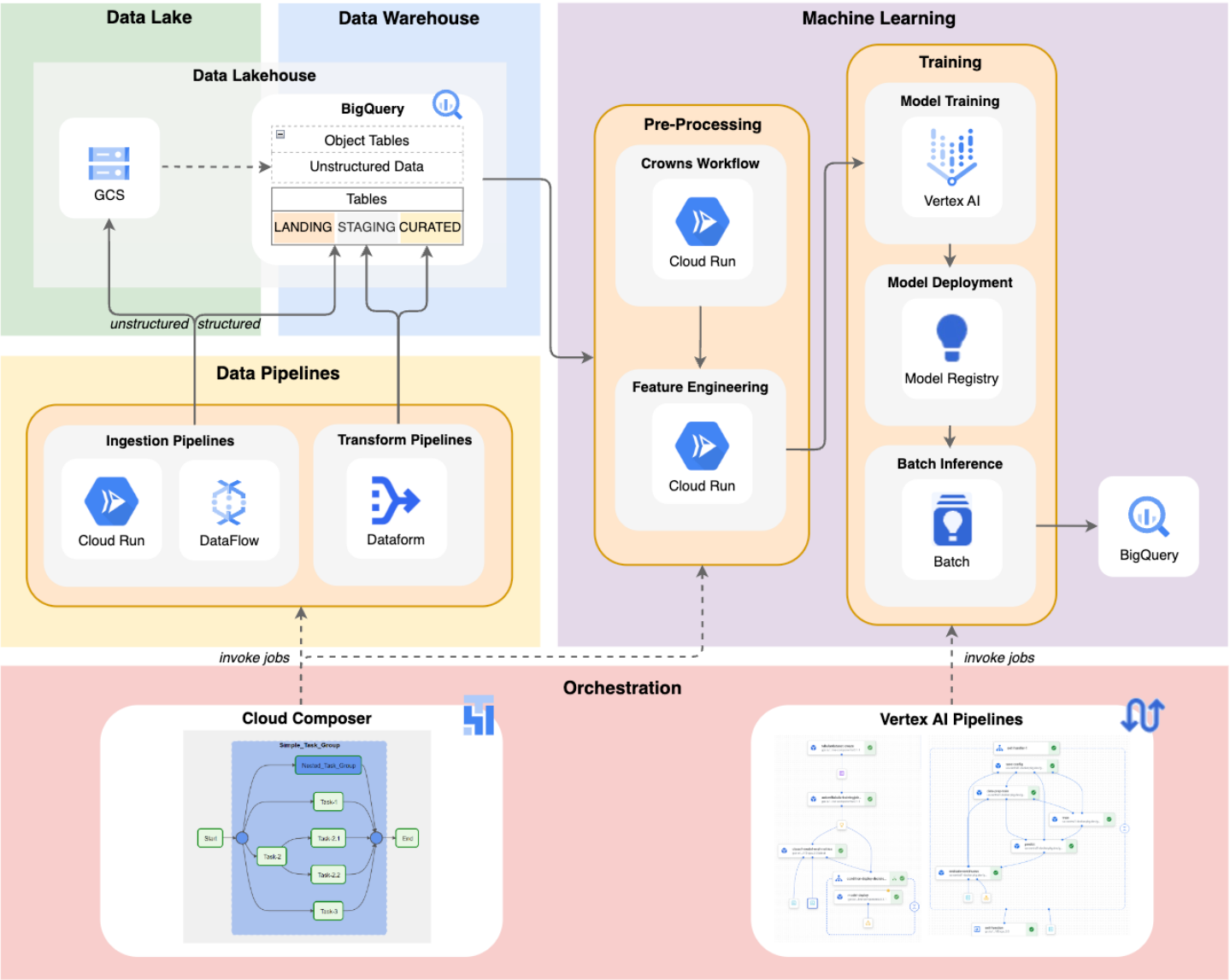
Vegetation AI Data Pipelines



Vegetation AI Orchestration Framework



Vegetation AI Proposed MLOps Architecture

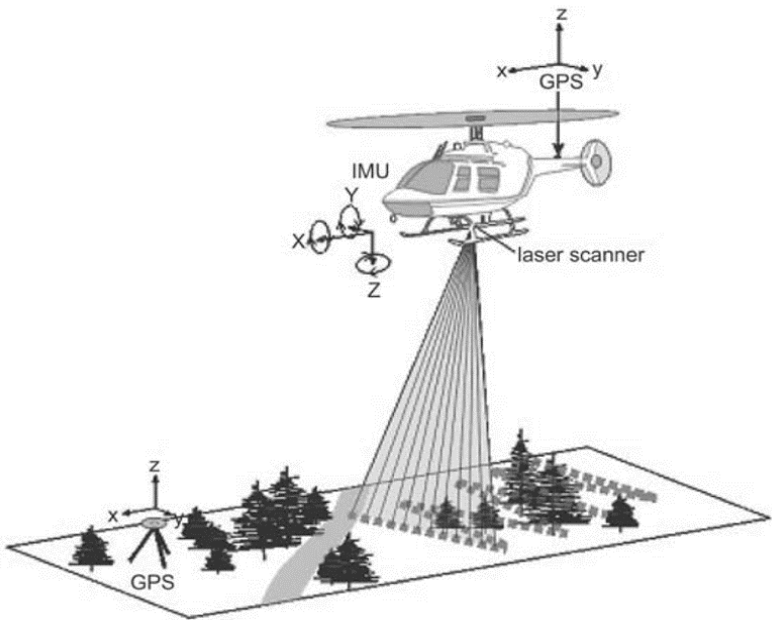


Vegetation AI POC & POC Extension Contextual Background

Contextual Background

Drivers that led to the vegetation AI POC & POC Extension:

- VM annually **assigns vegetation trim mitigations at the unique tree** level for environmental, work efficiency, access, and other business reasons
- SCE **completes & reports its vegetation mitigation work at the unique tree level**, and has established that standard as a compliance reporting precedent
- SCE has committed in its GRC to **transition from manual ground trim assessments to remote sensing assessments** in 2026, with increasing reductions in manual ground trim assessment in future years
- **Outcome:** Canopy Sense project was created to enable remote sensing to be applied at the individual tree level through the unique tree identification (crowns) & risk adjusted trim prescription (rx) models



Canopy Sense Timeline

2024				2025								2026													
FRM 1				FRM 2																					
Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	...
Proof-of-Concept				PoC Extension				Solution Analysis & Pilot				Implement: E2E Build to Release				Operate: Continued Rollout and Imp.									
Proved it is possible to use remote sensing to assign trims at the inventory (crown) level				Proved it is feasible to expand unique crown ID and trim rx across SCE’s varied service territory				Plan to design & pilot a scalable solution that can be applied across SCE’s network				Implement the Canopy Sense solution in parallel with field inspectors to minimize risk				Operate & iteratively improve the solution to prepare for ground inspection reduction				★ Finish Line: GRC goal to transition primarily to remote sensing vs. manual					
								Today																	