

Southern California Edison (SCE)
Model Documentation
Prepared for 2026-2028 WMP
Appendix B

Transmission Class Models (CFO & EFF)

5/16/25

Table of Contents

1. EXECUTIVE SUMMARY	3
1.1 Model Purpose and Intended Use	3
1.2 Model Description Summary	3
1.3 Model Risk Rating	4
1.4 Model Dependency and Interconnectivity	5
1.5 Model Assumptions	6
1.6 Model Limitations	6
1.7 Overall Model Performance Assessment	6
1.8 Contingency Plan for Vendor Model	7
2. MODEL FRAMEWORK AND THEORY	7
2.1 Model Inputs and Data Quality	7
2.2 Methodology	11
2.3 Suitability	13
2.4 Assumptions	14
2.5 Limitations and Compensating Controls	14
2.6 Model Outputs	15
3. MODEL PERFORMANCE AND TESTING	17
3.1 Model Specification Testing	17
3.2 Sensitivity Analysis and Scenario Testing	20
3.3 Outcome Analysis / Backtesting	22
3.4 Benchmarking Analysis	23
4. MODEL MANAGEMENT AND GOVERNANCE	25
4.1 Ongoing Monitoring Plan	25
4.2 Security and Control	25
5. REFERENCES	26

1. EXECUTIVE SUMMARY

1.1 Model Purpose and Intended Use

Transmission Class Models are a Probability of Ignition (POI) Sub-Model developed by Southern California Edison (SCE). The wildfire risk associated with SCE's transmission model is further measured in two sub-models, i.e., Equipment / Facility Failure (EFF) and Contact with Foreign Object (CFO). At SCE, models are developed using Machine Learning (ML) algorithms at the same granularity as the asset inventory. Transmission models are refreshed annually and are used to predict the probability of failure (POF) for transmission equipment.

The calibrated outputs of the Transmission model—i.e., failure events—are broadly used by two categories of programs described below:

1. Inspections and Remediations programs that consider POI as an element in prioritization and scoping.
2. Risk analyses via SCE's Multi Attribute Risk Scoring (MARS) Framework.

1.2 Model Description Summary

The Transmission model predicts the probability of failure of transmission equipment using Extreme Gradient Boosting (XGBoost)—a ML technique—with two sub-models, i.e., EFF and CFO.

- **EFF sub-model:**
The EFF sub-model is a binary classification model. The EFF component predicts the probability of a conductor igniting a spark due to equipment failure.
- **CFO sub-model:**
The CFO sub-model is a binary classification model. This differs from the distribution CFO model because of the availability of CFO events. ML models perform best when given a bigger set of events to model. Due to the nature of transmission CFO notifications all CFO sub-drivers are consolidated to generate enough events to build a well-performing model. The CFO component predicts the probability of a conductor producing a spark because of contact with any type of foreign object, i.e., animal, vegetation, balloon, vehicle, unknown, and others.

Both EFF and CFO sub-models use different variables to produce their failure targets but most of the variables are shared. Some of the common features used by both models are historical notification data, available conductor attributes and condition data (i.e., age, voltage, etc.) and other conductor and environmental attributes (i.e., historical wind, temperature, humidity etc.) as well as vegetation data (proximity to lines, tree growth rate, etc).

The model is programmed in Python using the libraries pandas, geopandas, xarray, and scikit-learn and is connected to various data sources such as Map3d, SAP, ADS Weather, etc. The model is run once a year manually by the Advanced Predictive Modeling team. The model is calibrated every year with annualized count of notifications.

Please refer to Section 2.1 for more information about the inputs used by the transmission model along with data processing details.

Southern California Edison (SCE) Model Documentation Prepared for 2026-2028 WMP Appendix B Transmission Class Sub-Model

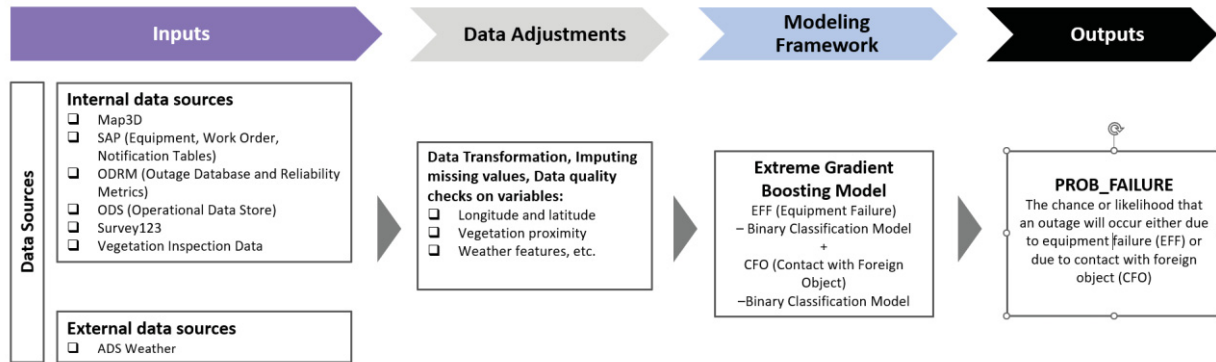


Figure 1: Transmission model framework

The transmission model uses the XGBoost methodology for both EFF and CFO sub-models. Since the prediction is a classified event (i.e., failure) and the XGBoost methodology can perform both classification and regression tasks, the XGBoost methodology is considered a viable choice for the Transmission model. This methodology predicts output with high accuracy, runs efficiently on large datasets, and maintains accuracy with minimal adjustments for missing values and data treatments. Furthermore, it has a modest number of hyperparameters to tune and benefits from more transparent explainability inherent in tree-based models.

1.3 Model Risk Rating

There is no defined mechanism to identify model risk rating at SCE. However, certain factors—like frequency of risk events and use case—are considered when flagging model risk. Based on the Wildfire Mitigation Plan quarterly report, the frequency of outages in a year from both EFF and CFO averages around 685. Figure 2 provides a snapshot of the count of outages over the years by the causes captured in the Transmission model. In addition, the output of this model importantly informs the strategy of several programs, discussed in section 1.1. Hence, the Transmission model is deemed to be a high-risk model.

				Number of risk events																Projected risk events											
					2015	2016	2017	2018	2019	2020	2020	2020	2020	2020	2021	2021	2021	2021	2021	2022	2022	2022	2022	2022	2022	2022	2022	2022	2022	2022	
Risk Event category	Cause category	#	Sub-cause category	Are risk events tracked for ignition driver? (yes / no)	2015	2016	2017	2018	2019	2020	2020	2020	2020	2020	2021	2021	2021	2021	2021	2022	2022	2022	2022	2022	2022	2022	2022	2022	2022	2022	
Outage - Transmission	25. Contact from object - Transmission	25.a.	Veg. contact - Transmission	Yes	15	16	13	9	8	0	0	1	5	2	1	0	1	2	2	2	2	2	2	2	2	2	2	2	2	2	
	25.b.	Animal contact - Transmission	Yes	79	75	67	70	33	8	20	5	12	6	14	10	6	7	10	9	8	7	10	9	8	7	10	9	8	7		
	25.c.	Balloon contact - Transmission	Yes	23	39	56	50	24	3	14	7	8	10	14	4	8	2	20	7	8	11	10	7	8	11	10	7	8	11		
	25.d.	Vehicle contact - Transmission	Yes	36	39	39	38	18	3	5	6	4	8	6	3	5	5	5	4	5	6	5	4	5	6	5	4	5	6		
	25.e.	Other contact from object - Transmission	Yes	77	36	36	45	28	7	4	5	3	1	2	4	9	7	4	4	6	3	4	4	6	3	4	4	6	3		
26. Equipment / facility failure - Transmission	26.a.	Capacitor bank damage or failure - Transmission	Yes	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0		
	26.b.	Conductor damage or failure - Transmission	Yes	23	16	91	47	38	6	3	13	7	9	5	6	20	7	8	8	13	9	8	8	13	9	8	8	13	9		
	26.c.	Fuse damage or failure - Transmission	Yes	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	26.d.	Lightning arrester damage or failure - Transmission	Yes	2	5	2	4	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	26.e.	Switch damage or failure - Transmission	Yes	5	3	4	5	2	3	2	0	0	0	0	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1		
26.f.	26.f.	Pole damage or failure - Transmission	Yes	13	13	18	10	14	3	0	3	3	3	3	3	3	3	0	5	3	3	3	3	3	3	3	3	3	3	3	
	26.g.	Insulator and bushing damage or failure - Transmission	Yes	10	13	20	4	9	2	3	1	2	0	1	0	3	2	2	2	3	2	2	3	2	2	2	3	2	2		
	26.h.	Crossarm damage or failure - Transmission	Yes	11	7	8	7	8	2	1	2	0	0	1	0	4	2	1	1	2	1	1	2	1	1	1	1	1	1		
	26.i.	Voltage regulator / booster damage or failure - Transmission	Yes	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	26.j.	Recloser damage or failure - Transmission	No	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
26.k.	26.k.	Anchor / guy damage or failure - Transmission	Yes	3	8	8	1	4	0	1	2	4	0	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1		
	26.l.	Sectionalizer damage or failure - Transmission	No	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	26.m.	Connection device damage or failure - Transmission	Yes	1	1	3	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	26.n.	Transformer damage or failure - Transmission	Yes	0	1	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	26.o.	Other - Transmission	Yes	14	26	10	25	41	3	8	9	10	10	5	5	6	3	5	5	6	7	5	5	6	7	5	5	6	7		
27. Wire-to-wire contact - Transmission	27.a.	Wire-to-wire contact / contamination - Transmission	Yes	14	20	17	29	42	10	10	1	3	1	9	4	2	3	7	4	3	2	7	4	3	2	7	4	3	2		
	28. Contamination - Transmission	28.a.	Contamination - Transmission	No	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	29. Utility work / Operation	29.a.	Utility work / Operation	Yes	11	25	12	9	11	1	1	1	1	2	2	3	0	2	2	2	1	2	2	1	2	2	2	1	2		
	30. Vandalism / Theft - Transmission	30.a.	Vandalism / Theft - Transmission	Yes	4	7	3	10	2	0	0	1	2	0	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1		
	31. Other - Transmission	31.a.	All Other - Transmission	Yes	189	212	224	244	187	38	63	47	55	47	54	48	53	36	53	50	53	50	53	50	53	50	53	50	53		
32. Unknown - Transmission	32.a.	Unknown - Transmission	Yes	369	335	311	134	267	38	70	36	59	53	55	48	66	29	60	58	64	60	60	58	64	60	60	58	64			

Figure 2: Key recent and projected risk events due to Transmission Model from the Quarterly report as of May 2022

References: Refer to link [\[RF 1\]](#) in Section 5 for SCE’s Wildfire Mitigation Plan Q1 2022 Quarterly Data Report submission.

1.4 Model Dependency and Interconnectivity

Transmission models are “Ignition Likelihood” models that use Atmospheric Data Solutions (ADS) modeling output along with other data sources to calculate the probability of ignition.

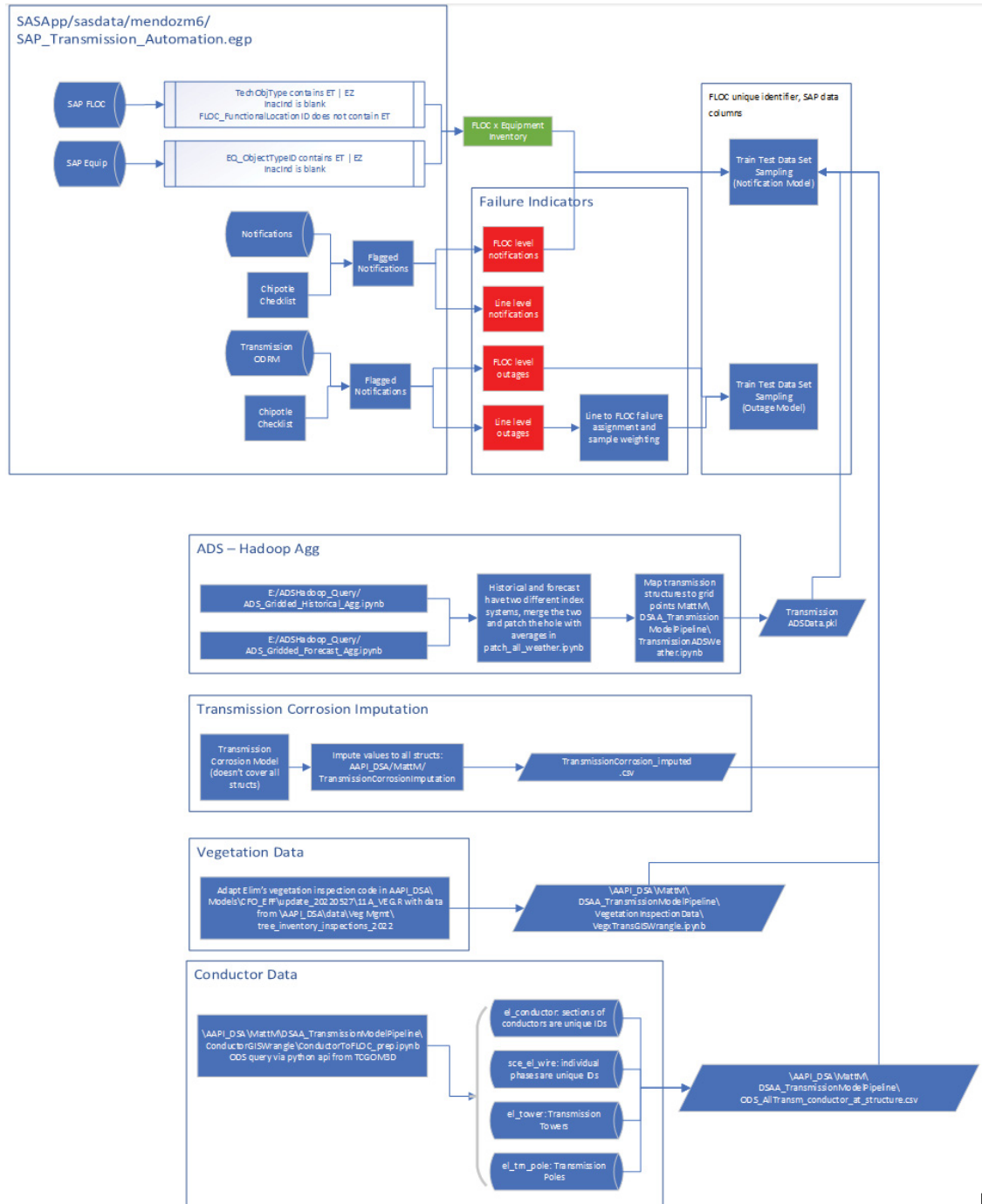


Figure 3: Model Interconnectivity Schema

ADS weather variables are used as one input in the Transmission models. ADS' Next Generation Weather Modeling System (NGWMS) upgrades SCE's in-house weather modeling capabilities and enhances SCE's ability to make more targeted PSPS decisions. The ADS model generates 10 years of hourly weather data between 2013 and 2022. That information is then processed and aggregated to calculate statistical measures such as mean and standard deviation of wind, humidity, rain, snow, etc. These are used as locational measures and are matched to the conductors by their latitude and longitude coordinates.

The output data from the Transmission models (i.e., POI) is used to inform the strategic decisions of the two categories of programs, discussed in Section 1.1.

1.5 Model Assumptions

The business and model assumptions for both the Transmission EFF and CFO sub-models are summarized below:

1. There is no change in the transmission conductor technical specification over time.
2. The contact types that can cause a spark will remain the same throughout the prediction period.
3. The calibration methodology assumes that fires are a subset of failures.
4. The model is designed to work in both base weather and extreme weather conditions.
5. The feature variables in the dataset should have some actual values so that the classifier model can predict accurate results.
6. The predictions from each tree must have very low correlations.

A detailed explanation of these assumptions is available in Section 2.4.

1.6 Model Limitations

The model limitations for both the Transmission EFF and CFO sub-models are summarized below:

1. Unavailability of linear/non-linear representation in the form of intuitive equation or correlation statistics.
2. Resource utilization in terms of system capacity and higher configuration for model execution is high.
3. Model accuracy may reduce if the dataset experiences covariate shift.

A detailed explanation of these data limitations is available in Section 2.5.

1.7 Overall Model Performance Assessment

The ML model used to build the Transmission Class model is the XGBoost algorithm. The model's overall performance is determined by the Area Under the ROC Curve (AUC) value and Confusion Matrix results.

The performance of the Transmission Class model was evaluated on test data using full transmission inspection notifications information through 2023.

- The AUC values for the EFF and CFO sub-models are 0.92 and 0.90 respectively.
- Confusion matrix results capture the recall as 67.4% and 64.2% for EFF and CFO sub-model respectively.

The above metrics were derived at the time of the model refresh in September 2024 to capture an exhaustive set of statistical results for documentation purposes.

1.8 Contingency Plan for Vendor Model

A contingency plan is not applicable for this model as it is an in-house SCE model. This is not a vendor model.

2. MODEL FRAMEWORK AND THEORY

The EFF and CFO sub-models are binary classification models pertaining to equipment failures and contact with foreign objects (i.e., animal, vegetation, balloon, vehicle, unknown, and others), respectively. Both sub-models of the Transmission model employ an XGBoost algorithm to predict the likelihood of a segment experiencing an ignition event. The XGBoost approach was chosen for the classification task over other modeling approaches—such as logistic regression, random forest, etc.—because it predicts output with high accuracy, runs efficiently on large datasets, and maintains accuracy with minimal adjustments for missing values and data treatments.

2.1 Model Inputs and Data Quality

Data Sources

This model refers to multiple internal and external data sources. The internal data sources used by the model are:

- **Map3D** is an internal geospatial data source used for geospatial display, and it contains the geospatial attributes of the assets as well as conductor related features like conductor type, conductor size, conductor length, conductor material, etc. as inputs to the Transmission model. The conductor data is updated through operations and field verifications and the circuit connectivity data is monitored/updated by the grid-applications support team.
- **SAP equipment/FLOC** houses circuit¹, structure, and equipment characteristics. It contains latitude and longitude information about the assets which are used to determine the location of the equipment which is used to geospatial data enrichment.
- **SAP Notifications** are used to flag increased risk conditions for the model as found and detailed by crew inspections. These are the failure targets for the classifier.
- **Survey123** is used to fetch the Vegetation information such as Tree density², tree proximity³ etc. are used only by CFO sub-model. It also houses information about the tree inventory on SCE's territory.

The external data sources used by the model are:

- **ADS** (Atmospheric Data Solutions) model provides gridded weather data using 10 years of hourly data from 2013-2023. These are used as a single locational measure and matched to the

¹ Circuit comprises segments that collectively form a path for electrical current floating from the power source (including, but not limited to, a substation) to another power source or circuit endpoint.

² Tree density is the total count of existing trees (includes hazard and dead and dying trees) within 50 feet to a segment. The trees are mapped to a segment using ArcGIS spatial join.

³ Tree proximity represents the closest distance to the segment. It is fetched by translating the CLEARANCE flag to numeric clearances thereby obtaining the closest distance to the segment.

conductors through spatial join to the nearest grid by the latitude and longitude data as a part of the data engineering step.

Quality Checks

SCE has internal data management teams for ensuring data quality, including Enterprise Asset Data (EAD) and Master Data. They work on processing asset data corrections (E2 notifications) in SAP and fixing known data issues like missing or erroneous latitude and longitude information for assets in their territory. Some of the data quality checks performed in the Transmission model to ensure accuracy, validity, integrity, and consistency are provided below. Quality checks (QC) are incorporated coded in Python.

The QC steps performed by Python code are as follows:

- The FLOC data file that is used to fetch the latitude and longitude values of the FLOCs from Map3D would have duplicates entries in it. Those duplicates in latitude and longitude values are removed to improve data quality by considering the distinct values in data processing.
- Vegetation data that is fetched from the tree inventory file might contain duplicate entries. These duplicates are removed by Tree_ID to improve its quality. Additionally, the tree inventory data is filtered for relevant, recent data marked by record status fields and within 50ft of conductor, and details about the trees are included as features, e.g. growth rate.

The manual QC steps are as follows:

- ADS weather data is validated against actual weather station observations.
- Asset data obtained from SAP is validated and updated through inspections and other programs.
- Routine tree data and hazard tree data from Survey123 are validated by QC and field verifications.

Data Sampling

Since this is a classification model to predict the failures, there are no sampling strategies used in the model other than the random split strategy to bifurcate the train and test data. The dataset used for the model is randomly sampled to have 67% in train data and 33% in test data.

Data Cleansing and Transformation

The data cleansing and transformation activities that are incorporated in the Python scripts as a part of automation to ensure the completeness of data used for model training and estimation are provided below.

- Missing data for the below specified numeric variables are handled by imputing the mean value.
 - FLOC_Latitude
 - FLOC_Longitude
 - DaysInService
- ADS weather data is aggregated to the year and then matched to the year of the event in the train/test set.

Data Assumptions

The accuracy of the predicted results depends on the accuracy of the data used to build the predictive models. The following are the data assumptions:

1. The assumptions used for the data imputation utilized SCE's Distribution Design Standard (DDS), engineering judgement, manufacturer data, and acceptable engineering practices.
2. The target labeling process used to label the failures and non-failures as '1' and '0' is considered accurate. This is performed by comparing the notification count in a segment against the mean value of total notification. If notification count in a segment is greater than the mean value of total notifications, the '1' is assigned which represents a failure. Else '0' is assigned for non-failures.
3. Input data with respect to asset, weather, and engineering information are assumed to be stable and will not change over time until the subsequent data refresh. Example: If there is an update in the structure information specific to an asset, that updated information will be reflected only in the subsequent data refresh. So, it is assumed that the updated structure information is not drastically different from the previous information which might alter the model outcomes.

Data Limitations

Following are data limitations across internal and external data sources:

1. Some of the data used by the model faces accuracy issues in terms of consistency in data labeling or missing values that may impact model prediction power.
 - Data labeling issues might be caused due to manual errors during data entry. While updating different categorical variables, incorrect labels might be mistakenly entered. This affects the consistency of the data and needs to be addressed before using the data in the model.
 - Missing data is handled by imputing the mode of the circuit for specific features.
2. The starting location of a failure is not accurately tracked every time, so the failures associated with a structure need to be mapped based on approximations that mirror the DOTS 2.2 Unknown Outage Mapping (Figure 4) method if the direct structure is missing.

Mapping Unknown Outages with DOTS2.2 Method

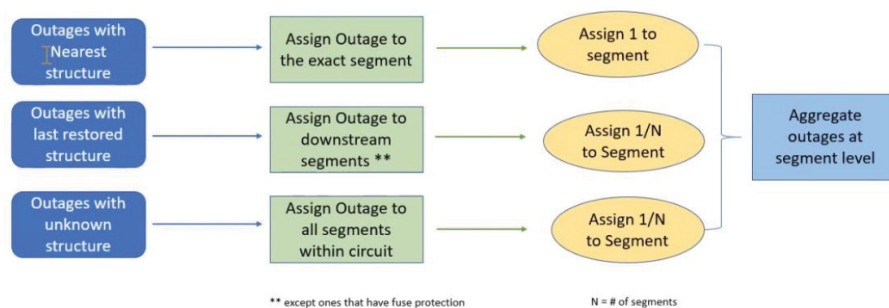


Figure 4: Unknown outage mapping process

Southern California Edison (SCE) Model Documentation Prepared for 2026-2028 WMP Appendix B
Transmission Class Sub-Model

The mapping process utilizes structure locations from the notification database to map the failures for a pass/fail in the predictive model. The techniques used to map notifications to structures can be detailed with the following steps:

1. Inner join notifications (with unique notification IDs) to their listed structures.
2. Assign a value of 1 to each row and create a pivot table that counts the number of structures assigned per unique notification ID.
3. Assign each row a sample weight of $1/N$ where N is the number of structures associated with a unique ID.

This allows each notification to be assigned to every associated structure without overcounting notifications that may affect multiple structures.

Independent variables

The Transmission model uses multiple variables/features. Most of the features are commonly used in both EFF and CFO sub-models. Some of these features are created based on engineering knowledge and some are selected based on expert advice like the logic to calculate tree density. A subset of the independent variables (inclusive of EFF and CFO sub-models) used in the Transmission model along with its data source and description, is provided below.

Feature	Data Source	Description
FLOC_Latitude	SAP	Geographical coordinate
FLOC_Longitude	SAP	Geographical coordinate
EQ_SystemVoltage	SAP	System voltage
DaysInService	SAP	Number of days between date at start of model year and the structure in service date
Sum(hourly_precipitation_amount)	ADS	Total yearly precipitation
N_hours_temp_lt36	ADS	Number of hours in a year when the temperature was less than 36° F
N_hours_temp_gt100	ADS	Number of hours in a year when the temperature was greater than 100° F
Std_wind_magnitude	ADS	Standard deviation of the hourly wind magnitude
Std_air_temperature	ADS	Standard deviation of the hourly air temperature
RCD_clearance_bucket_trees	Survey123/Inspections	Line clearance between vegetation and nearby lines
Avg_soil_temp_level_1	ADS	Average hourly subsurface temperature of soil
Stddev_pop_soil_temperature_level_1	ADS	Standard deviation of hourly soil temperature
Sum_Snow_depth	ADS	Total snowfall

10 years of hourly data fetched from ADS Weather model is processed and aggregated to calculate the numerical measures like mean, max and standard deviation for wind, temperature, water vapor, turbulence kinetic energy, humidity, rain and storm. Map3D and SAP provide data for location, equipment features, and start up dates.

Dependent Variable

In a typical classification risk model, defining the dependent variable is key for both model development and model performance assessment. The dependent variable in Transmission model is PROB_FAILURE. PROB_FAILURE represents the chance or likelihood that a failure will occur either due to failure in conductors or due to contact from foreign objects. The probability value ranges from 0 to 1 where '0' represents the least likelihood of a notification and '1' represents the high chance of a notification.

- EFF sub-model is a classification model, the target variable represents the chance or likelihood that a failure will occur due to failure of equipment.
- CFO sub-model is a classification model, the target variable represents the chance or likelihood that a failure will occur due to contact from a foreign object.

2.2 Methodology

SCE uses ML to identify patterns that may lead to failures causing sparks from conductors and use the trained model to predict Probability of Ignition (POI)s at asset level. The Transmission model predicts the POI arising from two ignition drivers viz., asset and contact type separately. The POI model with asset as the driver is categorized as sub-model EFF whereas contact type is categorized as sub-model CFO. Both the EFF and CFO components of the Transmission model employ an XGBoost algorithm to predict failure events.

XGBoost is a supervised ML algorithm that is constructed from many decision trees. It can be used to solve both classification and regression problems. This approach uses ensemble learning, which is a technique that combines many classifiers to achieve greater predictive accuracy than that of a single classifier. A decision tree is a decision support technique that forms a tree-like structure. It consists of three components: decision nodes, leaf nodes, and a root node. The following diagram shows the three types of nodes in a decision tree.

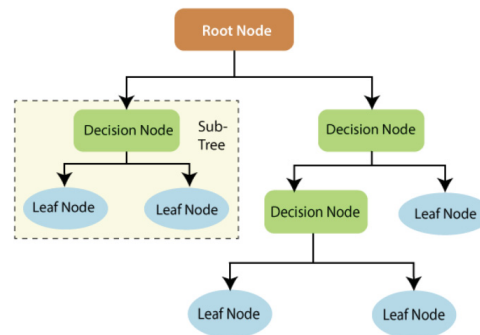


Figure 5: Decision Tree Structure

A decision tree algorithm divides observations of a dataset into branches, which further segregate into other branches. This sequence continues until a leaf node is attained. A leaf node cannot be segregated further. In more detail, the root node is the base of a decision tree, where the first of a chain of decisions is made. A branch is the connection path between nodes. A node is a potential splitting point on a tree. Decision nodes provide a link to the leaves. On the other hand, leaves, also known as terminal nodes, are the ends of a tree, representing the resulting classification or value for the sample.

The ‘forest’ generated by the XGBoost algorithm is trained through boosting. Boosting is an ensemble meta-algorithm that fits multiple models sequentially, where each new model corrects the errors of the previous ones. The diagram below shows the contrast between bagging and boosting.

Bagging VS Boosting

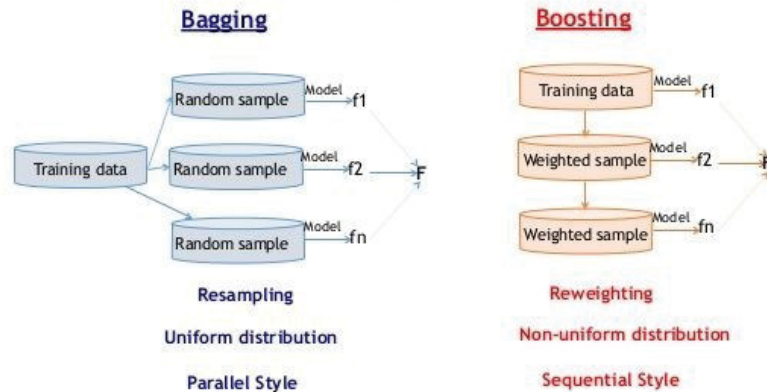


Figure 6: Structure of Boosting vs. Bagging

The selection of the final output in boosting follows a weighted voting system. In this classification model case, each decision tree contributes to the final output based on its accuracy, with more accurate trees having a greater influence. The XGBoost system combines the outputs of all the trees, correcting errors from previous trees to improve overall accuracy. The sequential addition of trees in the boosting process leads to higher accuracy and helps reduce bias, while also mitigating the risk of overfitting.

Train test split is a model validation procedure that allows to simulate how a model would perform on new/unseen data. Figure 7 provides the logic about dividing the dataset into train test data. First the data is consolidated and prepared for train test split. Then the historical input datasets are split into a training dataset (67%) and testing dataset (33%) based on simple random sampling strategy with a split ratio of 2:1 without replacement. Simple random sampling is a technique that ensures each observation has an equal likelihood of being selected for a set. It is a fair strategy as it helps in avoiding any bias involved compared to other modeling techniques and it has no restrictions on the sample size which makes it suitable to handle vast amounts of input data. The predictive algorithm is developed using the training dataset and built by looking at the interactions between all the features to find patterns and predict the likelihood of equipment failure.

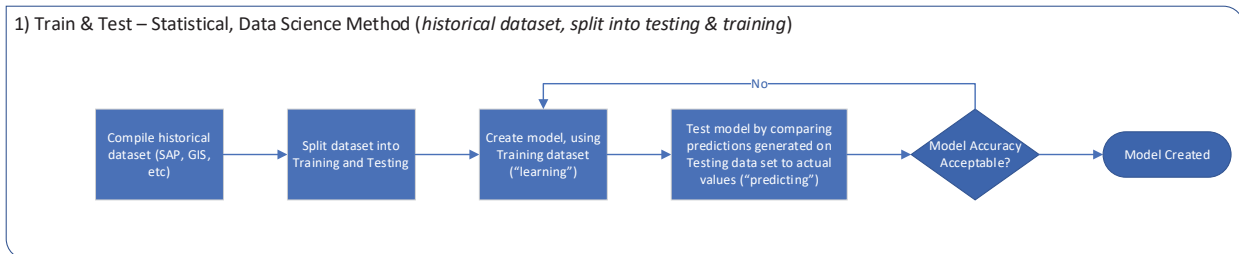


Figure 7: Train and Test data split logic

In the next step, the algorithm is tested on the 'testing' dataset. The model is run on the test dataset to make a prediction of failure or non-failure. Then an internal validation of the model is conducted by comparing the predicted results to the actual results which indicates the predictive capabilities of the features as well as the model. Area Under the Curve (AUC) is the metric used to assess the performance of the model on test data.

AUC – Area Under the Receiver Operating Characteristic (ROC) Curve estimates the model discriminatory power (i.e., degree of separability) for the binary classification problem. The ROC curve plots True Positive Rate against different thresholds with False Positive Rate (FPR) or True Negative Rate (TNR). The higher the AUC, the better the model is at predicting True Negatives (non-events) and True Positives (events).

Hyperparameter Tuning:

Hyperparameters are parameters that are explicitly defined by the user to control the learning process. The process of selecting the optimal hyperparameters is known as hyperparameter tuning, and the tuning process to achieve the best-defined performance statistic is known as hyperparameter optimization. The transmission models' hyperparameters are tuned manually to strike a balance between maximizing AUC and model recall (True Positive Rate).

The criterion used for the hyperparameter tuning in Transmission Model are:

- **N_estimators:** Total number of trees constructed by the ensemble model.
- **max_depth:** Specifies the maximum size of the sample data drawn for training each tree. A higher value for this feature will make the model more complex but can lead to overfitting.
- **Scale_pos_weight:** This parameter assigns weight to the positive class to make these classifications more influential to the model. This is a way to handle imbalanced data to increase the recall of the model at the expense of type I and II errors.
- **Learning_rate:** A parameter that controls the rate at which the optimizer updates weight. It's a value between 0 and 1. The learning rate affects the model's accuracy and speed of learning:
 - Smaller learning rate: Results in slower but more accurate updates. This can lead to a more optimal outcome.
 - Larger learning rate: Results in faster but less accurate updates. This can lead to underfitting.
- **Subsample:** The subsample parameter in XGBoost controls the fraction of observations used for each tree. By adjusting subsample, you can influence the model's performance and its ability to generalize.

2.3 Suitability

During the development of the model in 2024, both Random Forest and XGBoost methods were tested. XGBoost yielded better AUC scores than Random Forest and ran more efficiently. XGBoost was tested in previous years and was close to outperforming Random Forest. The test results showed that the XGBoost methodology fits well with the data and the results desired from the latest refresh. See Section 3.4 for the AUC comparison of these two approaches.

XGBoost methodology can solve classification and regression problems and works well with categorical and continuous variables. Among the main advantages of the XGBoost methodology is that it runs

efficiently for large datasets and maintains accuracy with minimal adjustments for missing values and data treatments. Theoretically, the XGBoost methodology exhibits a higher level of accuracy, stability and handles non-linear parameters and missing values more efficiently than other approaches. XGBoost also runs faster in Google Cloud compared to other models.

Hence, the usage of XGBoost for the Transmission model is deemed to be a suitable fit.

2.4 Assumptions

The key business assumptions that were considered during the model development are specified below:

BA 01: There is no change in Transmission technical specification over time. The model assumes the type of Transmission equipment used in the model building process have the same characteristics in terms of build and quality.

BA 02: The contact types that can cause a spark to remain the same throughout the prediction period. The six sub-drivers included in the CFO component are Animal, Balloon, Vegetation, Vehicle, Unknown and Other. The list of specific sub-drivers might not be exhaustive, but it is the best representation of the contact types that are the main drivers of ignitions based on SME judgment. It is assumed that there will not be any requirement to add a new sub-driver to the existing list of six CFO sub-drivers.

BA 03: The Calibration model assumes that fires are a subset of failures. Notifications are the representative failure targets used in place of the relatively sparse number of ignition events. Notifications indicate equipment with an increased risk of failure which can potentially spark an ignition, but not all notifications will result in a fire. Hence, fire can be treated as a subset of failure.

BA 04: The model is designed to work in both base weather and extreme weather conditions. The weather variables considered by the model are represented as various statistical aggregations like max, mean, and standard deviation on wind, wind speed, humidity, rain, and snow. Hence the model results can be used under both base weather and extreme weather conditions.

The functional/model methodology assumptions that were considered during model development are discussed in detail below:

MA 01: The feature variables in the dataset should have some actual values so that the classifier model can predict accurate results. In an ideal scenario, all variables would have actual, not estimated, values. The current model provides accurate results even after using estimates as they are derived through imputation using actual values from other variables. Example: Estimating equipment age based on equipment startup/in service dates.

MA 02: The predictions from each tree must have very low correlations. It is difficult to differentiate between a real interaction effect, marginal effects, and random variations in XGBoost. Hence, highly correlated variables in XGBoost approach will impact its ability to identify strong predictors.

2.5 Limitations and Compensating Controls

The key model limitations that would impact the accuracy and performance of the model are discussed in detail below:

Limitation ID: L01

Limitation Title: Unavailability of linear/non-linear representation in the form of intuitive equation or correlation statistic.

Description: The XGBoost algorithm does not explain any linear or non-linear relationship in the form of an equation or correlation statistic that would allow enable measurement of the scalability of impact of independent variables on the dependent variable.

Compensating Controls: The XGBoost model is considered a black box as it is difficult to understand the relationship between independent and dependent variables and how the independent variables influence predictions. Since black box is a common limitation with most ML algorithms, usage of the model is considered appropriate as it provides better AUC results than other models.

Limitation ID: L02

Limitation Title: Resource utilization for model execution is high.

Description: Since XGBoost models process many decision trees, they need more resources with respect to system configuration and system capacity to store that data.

Compensating Controls: The resource utilization factor will have a major impact for the real time models as it would run more frequently. Since the Transmission model is run only once in a year with reasonable use cases, the impact of resource utilization is low. Since the model is not executed automatically through computer programs with a defined frequency and is instead run only once a year manually, usage of the model is considered appropriate.

Limitation ID: L03

Limitation Title: Model accuracy might reduce if the dataset experiences covariate shift.

Description: Covariate shift is a type of model drift which occurs when the distribution of independent variables changes between the training environment and live/test environment. Since XGBoost cannot extrapolate (i.e., predict outside the training space), the model performance might decrease if there is covariate shift in the dataset.

Compensating Controls: The covariate shift affects most ML models, as test data is never going to be the same as training data. Detecting and addressing covariate shift is therefore a key step to the ML process. The current model is run only once a year along with data refresh. It uses random sampling mechanism to split the dataset into train (67%) and test (33%) data whenever it is run. The random sampling mechanism is used to resolve covariate drift and maintain the accuracy of model results. Hence, the XGBoost methodology and the random sampling mechanism to split train/test data are considered appropriate.

2.6 Model Outputs

The Transmission model is a downstream model that predicts the probability of ignition (POI) arising from asset (conductor) and contact from foreign objects separately. Both POI models categorized as EFF and

CFO have a single output respectively characterized by a continuous number between 0 and 1 for each transmission conductor.

The probabilities across different asset failure predictive models cannot be aggregated or compared and hence are calibrated to derive frequencies of ignition. The sum of the resulting frequencies of ignition for a sub-driver equals the total expected ignitions for the specified year.

$$\text{Frequency of Ignition} = \text{Probability of Ignition} \times \frac{\text{Calibrated Targets}}{\sum \text{Probability of Ignition}}$$

where Calibrated Targets = Forecasted Ignitions for that sub-driver

The output from this calibration exhibits the following features:

- Frequency: Each value can be specified as the frequency of fires per year.
- Comparability: The frequencies are comparable against sub-drivers and models.
- Additivity: The frequencies can be added across models to derive the aggregated fire forecast in a year.

This is achieved by forecasting fires by sub-drivers and using these forecasts to scale the model probabilities such that the sum of probabilities from each calibrated model equals the forecast by sub-driver.

Error! Reference source not found. provides the calibration steps that are performed using the failure probability results from the Transmission model. This methodology followed in the calibration model is provided below:

- A. Aggregate the probability output from each sub-driver model.
- B. Based on the forecast logic specified above, find the forecast results (expected fires) for each sub-driver.
- C. Generate the calibration factor for each sub-driver based on the values calculated in the above steps (B/A).
- D. Multiply each model probability by its calibration factor to arrive at the estimated frequency of fires from each sub-driver.

Southern California Edison (SCE) Model Documentation Prepared for 2026-2028 WMP Appendix B Transmission Class Sub-Model

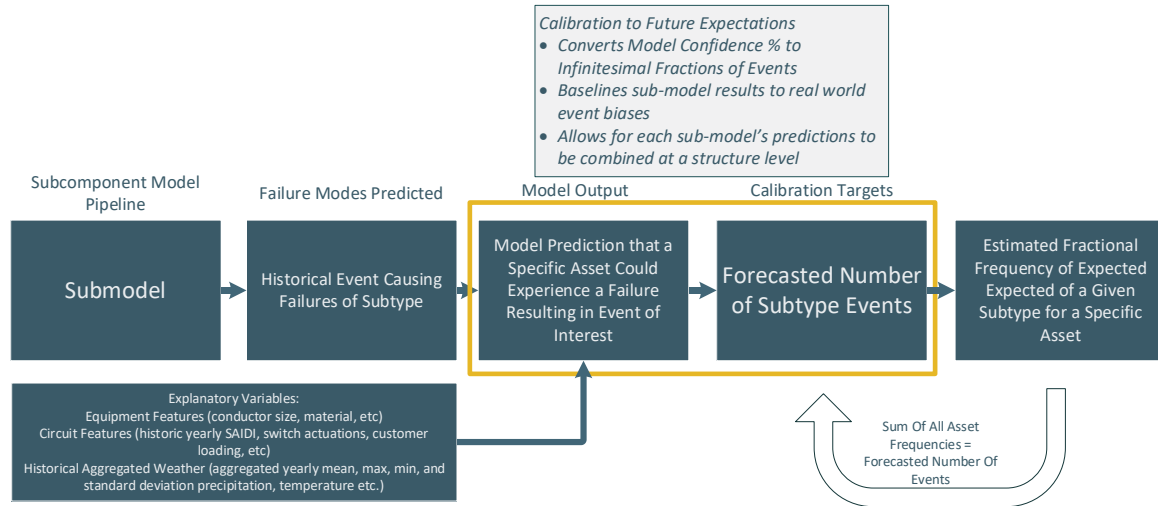


Figure 8: Calibration model schema

This estimated frequency of fires from each sub-driver can be added across the models to derive the expected frequency of ignition for each location.

The data from the calibrated probabilities—frequencies of events—based on the output from the Transmission model is used to inform the programs mentioned in Section 1.1.

3. MODEL PERFORMANCE AND TESTING

For each machine learning model developed, SCE tries to select the best algorithm based on the model train/test performance, which can be measured in one way by Area Under the Curve (AUC), indicating the model's accuracy, and other types of accuracy metrics such as the Confusion Matrix with particular attention being given to precision and recall.

3.1 Model Specification Testing

The model is developed and tested in Python using the library sci-kit learn. The model is run once a year manually by the Advanced Predictive Modeling team. The model probabilities are calibrated every year in SAS scripts to annualized counts of notification data.

SCE performs verification of the model implementation by checking the variable importance results (see Model Estimation section below for a detailed explanation of variable importance results). The performance of the model is validated through the AUC, defined in Section 2.2 and provided in Section 3.3.

The validity and impact of the Model Assumptions are discussed below:

- XGBoost is a powerful method for variable selection in high-dimensional data. It can handle variables with high correlation due to its tree-based structure, which considers dependencies hierarchically. However, highly correlated variables can still pose challenges, as they might lead

to redundancy and reduce the model's interpretability. While XGBoost can capture interaction effects, distinguishing between real interactions, marginal effects, and random variations can be complex. Therefore, it is often beneficial to filter out highly correlated features to improve the model's performance and interpretability.

Model Estimation:

The EFF and CFO sub-models employ several independent variables. Section 2.1 contains a list of the independent variables utilized in this model.

The variable importance test results for the EFF and CFO sub-models, Figures 9 and 10, show the order of which features provide the most information gain in informing the correct prediction of failure or non-failure. The variable importance features test estimates the relative influence of each variable by calculating whether that variable was chosen to split during the tree building process and how much the squared error over all trees improved, or decreased, as a result.

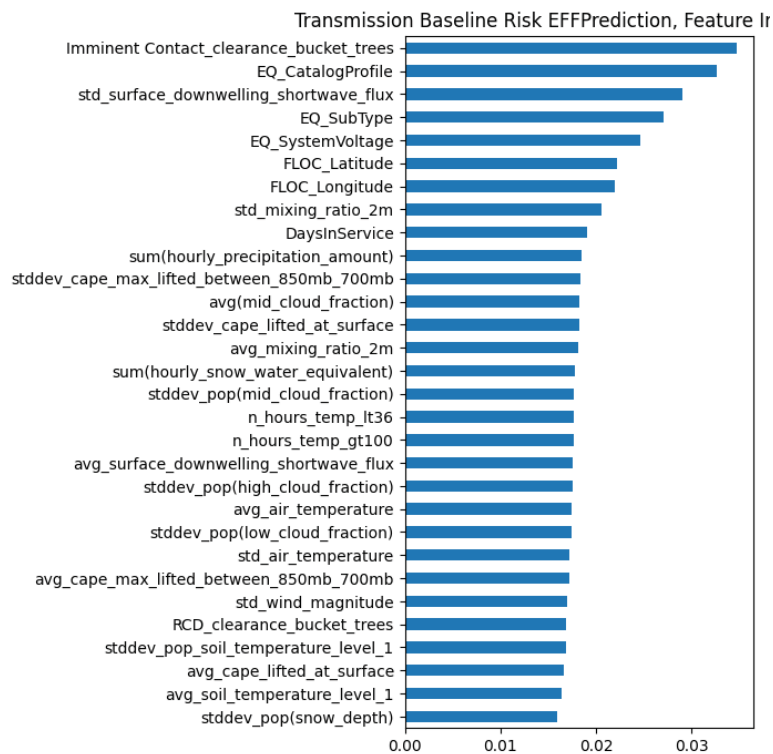


Figure 9: Variable Importance test results for EFF sub-model

Southern California Edison (SCE) Model Documentation Prepared for 2026-2028 WMP Appendix B Transmission Class Sub-Model

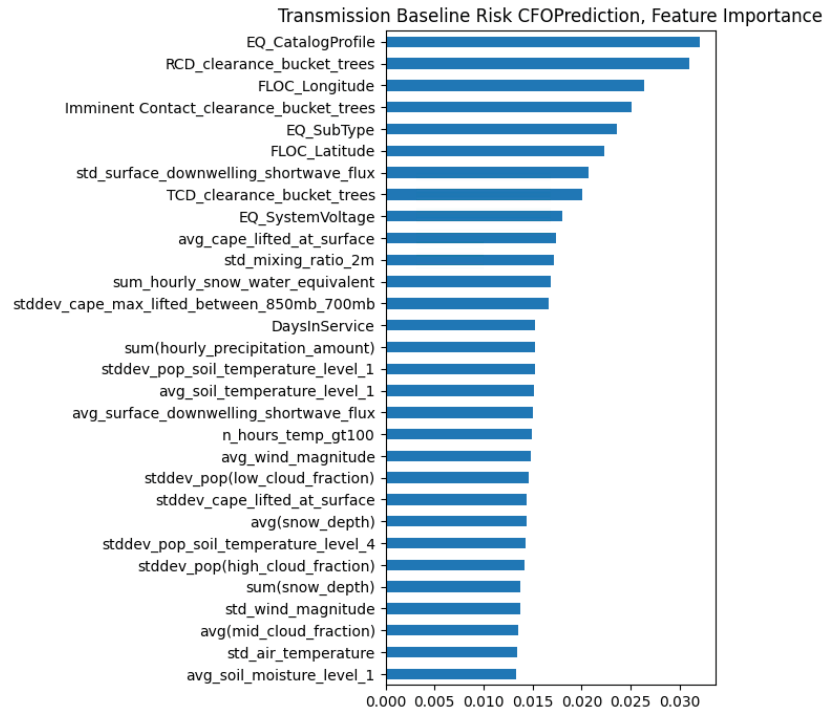


Figure 10: Variable Importance test results for CFO sub-model

References: Refer to link [RF 3] in Section 0 for description on the methodology used to perform the Variable Importance for tree-based methods.

The accuracy of the model prediction can be determined using the Confusion Matrix, Precision, Recall and Classification Error Rate results.

- A confusion matrix presents a table layout of the different outcomes of the predicted and actual values of a classifier model.

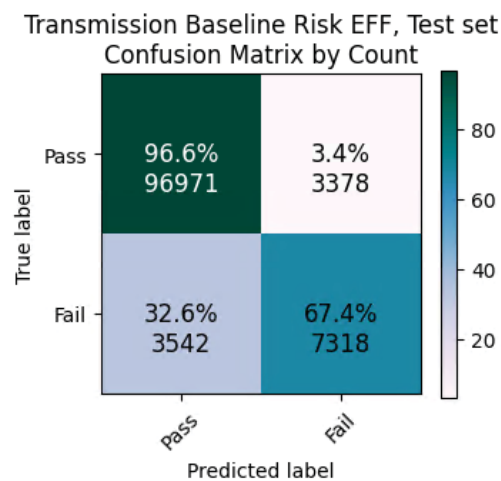


Figure 11: Confusion matrix results for EFF sub-model

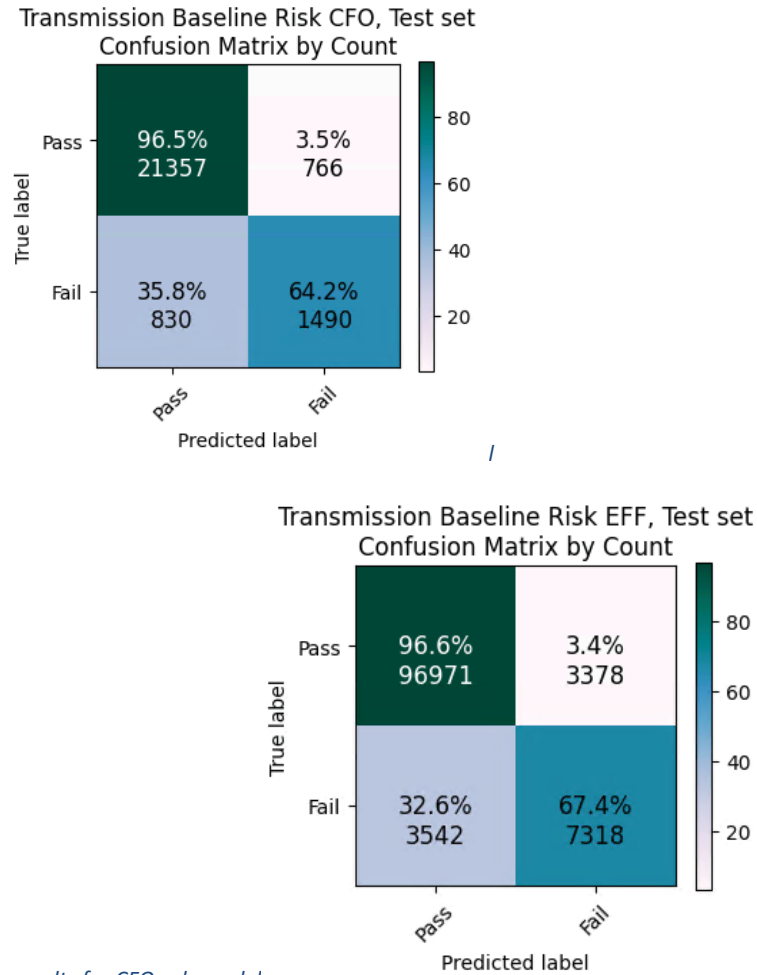


Figure 12: Confusion matrix results for CFO sub-model

- Figure 11 and 1Error! Reference source not found. provide the confusion matrix results for EFF and CFO sub-models respectively. It captures the accuracy rate as 93.78% and 93.47% for EFF and CFO sub-model respectively.
- Classification error rate is used to estimate the proportion of instances misclassified over the whole set of instances. It is estimated using the formula:

$$\text{Error Rate} = \frac{\text{False Positives} + \text{False Negatives}}{\text{True Positives} + \text{True Negatives} + \text{False Positives} + \text{False Negatives}} * 100$$

The error rate for EFF and CFO sub-model turns out to be 6.22% and 6.53% respectively. This means that the failure rate of the model prediction is low and under control.

All these test results are performed on test datasets with full historical notification data till 2023.

A detailed explanation of the compensating controls employed for these limitations is available in Section 2.5.

3.2 Sensitivity Analysis and Scenario Testing

Southern California Edison (SCE) Model Documentation Prepared for 2026-2028 WMP Appendix B Transmission Class Sub-Model

Shapley Additive Explanations (SHAP) is a method that provides an explanation of a model's output by attributing the contribution of each feature to the model's prediction. SHAP is based on the concept of Shapley values, which is a method for distributing the contribution of each player in a cooperative game. In the context of an ML model, the players are the features, and the game is to predict the output. For sensitivity analysis, SHAP values were calculated for the input features to quantify how each feature impacted model predictions.

To calculate the Shapley values for a feature, SHAP generates a set of all possible feature combinations, called coalitions. For each coalition, SHAP calculates the model's output and the difference between the output of the coalition with and without the feature. These differences are averaged over all possible coalitions, giving a measure of the feature's contribution to the model's prediction. This process is repeated for each feature in the model.

The result is a set of Shapley values that describe the contribution of each feature to the model's prediction. Positive Shapley values indicate that the feature increases the model's prediction, while negative values indicate that the feature decreases the prediction. The magnitude of the Shapley value indicates the importance of the feature. These values can be used to provide an explanation of the model's output, either by showing the contribution of each feature for a specific prediction or by calculating the average contribution of each feature over the entire dataset. By transforming variables into additive factors that drive probability, SHAP can analyze the sensitivity of a model to different variables, which can help identify which features are the most important in making predictions. Overall, SHAP provides a powerful method for understanding and interpreting the behavior of complex ML models.

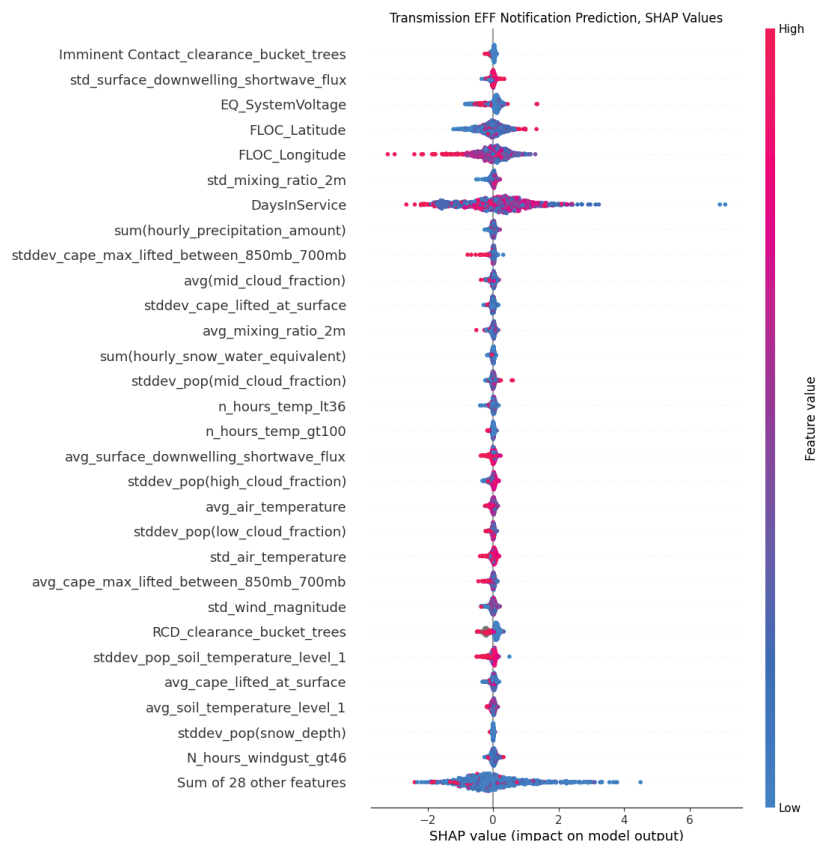


Figure 13: SHAP values for the most important features in the EFF sub-model

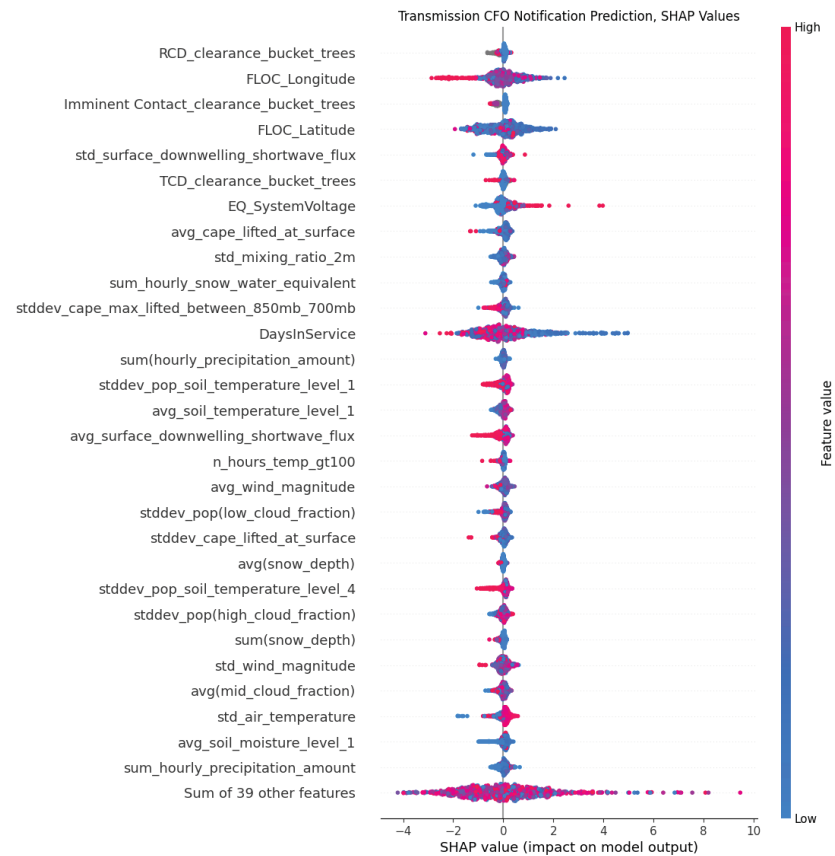


Figure 14: SHAP values for the most important features in the CFO sub-model

3.3 Outcome Analysis / Backtesting

The subset of historical data on which a model is trained and optimized is referred to as the in-sample data, while the subset of the dataset that has been reserved to test the model is known as the out-of-sample data. The Transmission model uses a random sampling approach to split the dataset into Train (67%) and Test (33%) data. The results of the train data are considered in-sample backtesting and the results of the test data are considered out-of-sample backtesting.

Once the ML model is built with the training data, it is evaluated using a separate test dataset that has not yet been studied. The performance of the model is determined by the AUC value. Figure and Figure show the AUC ROC for EFF and CFO sub-models based on the test dataset ran with full historical notification data until 2023.

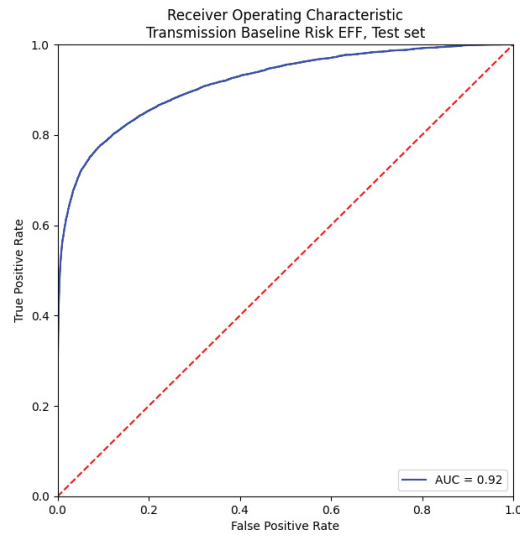


Figure 15: Out-sample back testing result for EFF sub-model based on test dataset

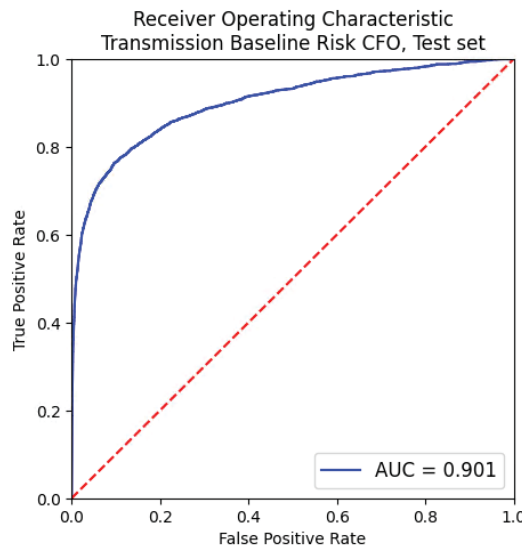


Figure 16: Out-sample back testing result for overall CFO sub-model based on test dataset

The AUC values for the EFF and CFO sub-models are 0.92 and 0.901 respectively. The AUC values of both sub-models are higher than 0.901 which implies that the models possess high accuracy in terms of predicting the results.

The impact of uncertainty in model inputs and parameters on model outputs are tested as a part of the sensitivity analysis and the results are captured in Section 3.2. In addition, the data imputations that are incorporated to address missing values before running the model are defined in Section 2.1.

3.4 Benchmarking Analysis

For Transmission Models, different approaches like Extreme Gradient Boosting (XGBoost) and Random Forest were considered during the model development phase in 2024. The analysis on these supervised ML approaches and the results are provided below.

- **Random Forest** is a popular ML algorithm that can be used for both classification and regression problems. Random Forest is another ensemble method that combines the predictions of several decision trees to improve the predictive accuracy of the model. The individual decision trees are created based on a randomly selected subset of features at each node prior to determining the optimal split so each tree differs. The final output is determined by taking the majority vote of the predictions from the individual decision trees. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.
- **Extreme Gradient Boosting (XGBoost)** is an advanced implementation of gradient boosting designed for speed and performance. It is highly efficient and scalable, making it suitable for large datasets and complex models. XGBoost includes regularization to prevent overfitting, handles missing values well, and provides detailed feature importance scores. Additionally, it is optimized to run efficiently on cloud platforms like Google Cloud, leveraging distributed computing to handle large-scale ML tasks with reduced computation time and cost.

Historically, Random Forest has been used in the transmission model due to its ability to handle large datasets with significant dimensionality while providing robust performance and interpretability. Its ensemble approach, which aggregates the predictions of multiple decision trees, allows users to gain insights into the importance of features contributing to predictions. However, as the demand for higher accuracy and efficiency in predictive modeling has grown, XGBoost has emerged as a notable upgrade to Random Forest for its advanced capability and interpretability. Utilizing gradient boosting techniques, XGBoost not only enhances predictive accuracy through a more focused learning approach but also retains a high level of explainability. By offering features such as built-in support for calculating feature importance and visualizing decision paths, XGBoost allows practitioners to maintain transparency in their models while benefiting from improved performance metrics. Figures 17 and 18 show the accuracy of transmission modeling using Random Forest compared to previously shown XGBoost ROC curves (Figures 15 and 16).

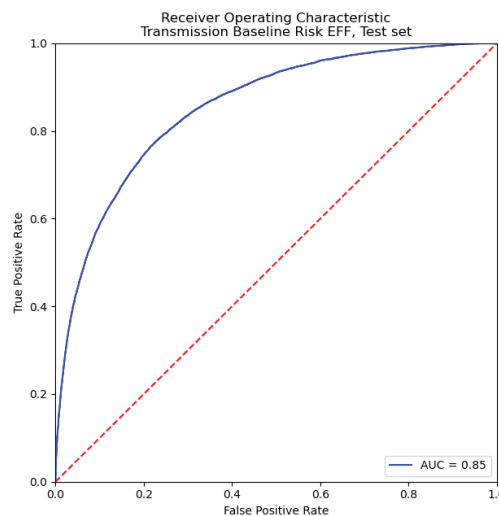


Figure 17: Out-sample back testing EFF sub-model using Random Forest, compare to Fig 15

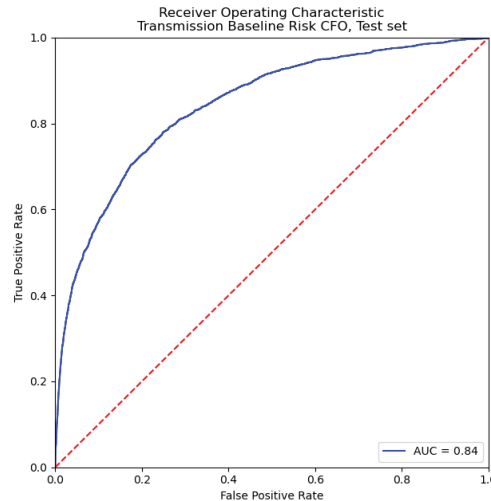


Figure 18: Out-sample back testing CFO sub-model using Random Forest, compared to Fig 16

4. MODEL MANAGEMENT AND GOVERNANCE

4.1 Ongoing Monitoring Plan

Ongoing monitoring is important for ML models especially when used to make predictions or when they are run on datasets with high volatility in variable values. The Transmission model is run manually once a year, incorporating updated input datasets to reflect the latest available data and implementing any specific model enhancements, e.g., inclusion/replacement/removal of a feature, optimization of the code, evaluation of a new performance metric, etc. During the model refresh, the limitations and assumptions of the model are also revisited by the model developers and necessary actions are taken to address them.

Performance monitoring is required only after running the model. The AUC and accuracy rate from Confusion Matrix results obtained after model refresh are compared against a threshold of 70%. If the value drops below this threshold, the reason behind the performance dip is investigated. Post-investigation, the steps required to improve the model's performance will be carried out. To monitor the model performance more thoroughly, the developers of the model plan to additionally evaluate metrics like Precision and Recall. Precision is the positive predictive value which represents the proportion of predicted failures that were predicted correctly. Recall is the true positive rate which represents the proportion of actual failures that were predicted correctly.

The model documentation and the performance results are updated once a year immediately after the model refreshes.

4.2 Security and Control

The Advanced Predictive Modeling team has access to data inputs, code, and implementation for the model. Other business units, like the Grid Hardening Strategy team, are provided with access to the model outputs upon request but cannot update or modify the code.

The model is run using Python programming and it can be executed in Python 3. The current model versioning is labeled by year of refresh (e.g., 2024 refresh). The code is saved on GitHub, a platform that

facilitates version control by tracking changes to the source code. Users with write or admin privileges to the repository can review proposed changes and approve them.

A contingency plan is not applicable for this model as it is an in-house model for SCE.

5. REFERENCES

RF 1: SCE's WMP 2022 Q1 Quarterly Data Report submission

<https://www.sce.com/sites/default/files/AEM/Data%20Requests/2022/SCE%20Q1%202022%20Tables%201-12.xlsx>

RF 2: Boosting vs. Bagging Visual Reference

Odibat, O. (n.d.). Boosting Algorithms. *SlideShare*. Retrieved May 2, 2025, from [\[URL\]](#)

RF 3: *Variable Importance methodology for tree-based methods*

[Feature Importance — scikit learn documentation](#)