ELECTRIC VEHICLE DETECTION CHALLENGE

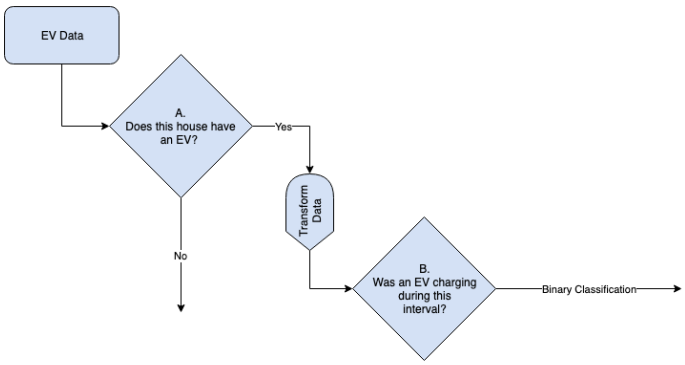
July 1, 2019

Andrea L. Keane

July 1, 2019

Abstract

Increasing elective vehicle (EV) ownership and thus electricity demand introduces challenges for existing energy infrastructure. Developing predictive models to understand and manage the demand may mitigate risks and enable benefits of increased EV ownership. Using 60 days of labeled smart meter power readings from 1590 houses with and without EVs, binary classifiers were trained to predict (a) which houses have an EV and (b) during which time intervals an EV was charging. The Logistic Regression classifier trained for part A yielded an accuracy between 0.79 and 0.86. The K-Nearest Neighbors classifier for part B yielded an accuracy between 0.75 and 0.96. Predictions were made on a blind test set of 699 houses using the trained models from parts A and B.



Classification data flow

Introduction

Background

Increasing electric vehicle (EV) ownership presents new challenges for the energy grid. Potential impacts include infrastructure failure, unstable electricity streams and power outages. Highest risk areas are those where peak demands are already approaching maximum capacity. Fortunately, the steadily increasing adoption of EVs provides an opportunity to proactively address problems and optimize solutions. With emerging technologies, EV growth delivers an opportunity to improve our energy infrastructure.

Predictive modeling plays a critical role in optimizing supply, managing demand and coordinating consumption. Introducing a fleet of EV batteries effectively increases the grid's storage capacity. The ability to store energy creates separation between energy supply and demand, buffering against unexpected fluctuations. Properly managed, the decentralization of energy storage may improve overall energy market stability. Furthermore, EV batteries may enable more efficient use of "clean" energy sources which are transient and don't necessarily align with the current demand cycles. The ability to predict when and where EVs are plugging in is critical to optimizing the energy grid for both environmental impact and consumer demand.

Problem Description

The training set contains two months of smart meter power readings from 1590 houses. The readings were taken at half-hour intervals. Some of the homes have electric vehicles and some do not. The file "EV_train_labels.csv" indicates the time intervals on which an electric vehicle was charging (1 indicates a vehicle was charging at some point during the interval and 0 indicates no vehicle was charging at any point during the interval). Can you determine:

A. Which residences have electric vehicles?

B. When the electric vehicles were charging?

C. Any other interesting aspects of the data set?

A solution to part B might consist of a prediction of the probability that an electric car was charging for each house and time interval in the test set. Please include code and explain your reasoning. What do you expect the accuracy of your predictions to be?

Model Development

Data Profile

The training data contains 2880 (60 days) power readings for 1590 houses. Of the 1590 houses, 30.5% (485 houses) charged an EV during at least one interval in the time period. Houses with EVs had an average of 220 EV charging events over the 2880 total intervals. Figure 2 demonstrates the imbalanced class distribution for both parts A and B.

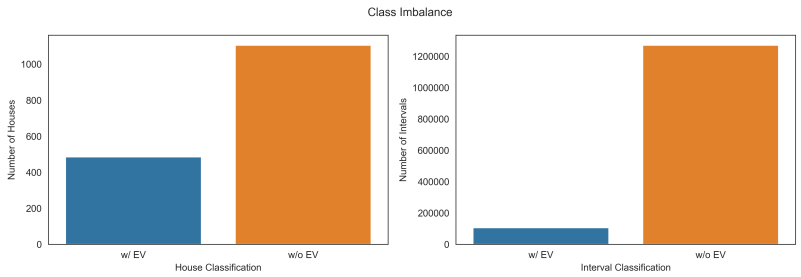


Figure 2: Imbalanced EV Ownership and Charging Events

After removing outliers (discussed below), 2.4% of all power readings contain an EV charging event. This increases to 7.7% when only considering power readings from households with EVs. In both cases, the proportion of EV charging events is significantly lower than the non-EV charging events. The labeled training data was deliberately not balanced prior to classifier development to maximize training data size and to develop a model with bias resembling the distribution of realistic data.

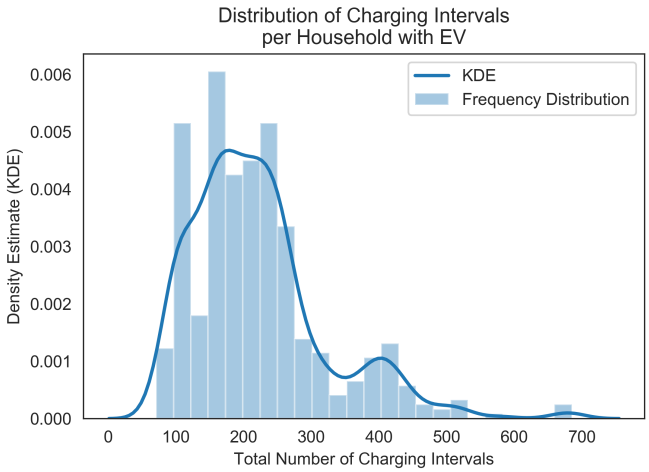


Figure 3: Distribution of total number of charging events

The total number of charging events for houses with EVs ranges from 71 to 685 events (2.5% to 23.8% of intervals). No houses have a sufficiently small number of charges to suggest that the EV charging events are due to non-residents. Per figure 3, most households have approximately 200 charging events, with a second peak just over 400 events. The maximum number of charging events is 685. This distribution may be due to multiple EV households or differing EV technologies.

Figure 4 examines the distribution of descriptive statistics based on household EV classification. As expected, the minimum power reading across all 2880 intervals is not discernibly different between the two classes. Somewhat surprisingly, the mean and median power readings appear unaffected. The maximum power reading demonstrates a difference between the two classes. These observations are further supported by the samples in Figure 5.

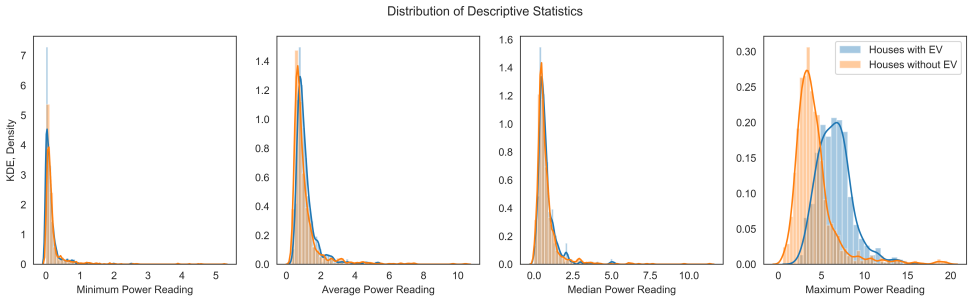


Figure 4: Distribution of descriptive statistics by house classification

Methods

Data Preparation

Initial investigation revealed outliers with exceptionally large power readings. Houses with power readings in the top 5% (> 2 stds) are excluded, resulting in the disqualification of 37 houses (2.3%). To avoid creating holes in the data, the entire house is removed instead of a single house-interval data point. Table 1 summarizes the impact of outlier removal. The maximum of the 'Max Power' field decreases from 163.1 to 19.6. The mean and standard deviation of 'Max Power' decrease accordingly.

		Total Power	**Average Power**	**Median Power**	**Min Power**	**Max Power**	**Total Charges**
With Outliers	mean	4031.3	1.4	1.1	0.3	5.7	67.2
	std	8921.2	3.1	2.8	1.6	7.1	115.7
	min	814.8	0.3	0.0	0.0	0.6	0.0
	50%	2446.7	0.8	0.6	0.1	4.5	0.0
	max	244527.1	84.9	75.8	50.9	163.1	685.0
Without Outliers	mean	3127.8	1.1	0.8	0.2	4.9	68.0
	std	2526.9	0.9	0.8	0.4	2.6	116.3
	min	814.8	0.3	0.0	0.0	0.6	0.0
	50%	2417.1	0.8	0.6	0.1	4.4	0.0
	max	30073.7	10.4	11.3	5.2	19.6	685.0

Table 1: Descriptive statistics before and after outlier disqualification

Prior to training, data is normalized using the `sklearn.preprocessing.StandardScaler`. Per scikit-learn documentation, `StandardScaler` will "standardize features by removing the mean and scaling to unit variance". Thus, for each feature the mean is set to 0 and the standard deviation to 1. The scaling operation was performed on all model input data including training, validation and testing data sets. Separate scalers are used for parts A and B, as the classifier inputs are different. For consistency, scalers from the training process are persisted via `pickle.dump(...)` and applied to the final testing data prior to classification.

Feature Engineering

Pearson's Correlation Coefficient was used to evaluate engineered features. This correlation model tests for linear correlation between a pair of features. Uncorrelated features have a coefficient near 0, while perfectly correlated features have a coefficient of +/-1.

Features were engineered to quantify observations in the 'energy signature' of sampled houses, figure 5.

Houses with EV charging events appeared to have more frequent and significant spikes in power readings. The maximum power reading in a 60-day window was generally higher for houses with EVs. Theoretically, the average and total power consumption should also be larger. The developed features aim to capture such behavior by defining a baseline and characterizing peak presence for each household. Adjusting for linear modelling limitations, features are raised to a power. In these cases, both an even and an odd power were considered to account for possible sign (+/-) dependencies.

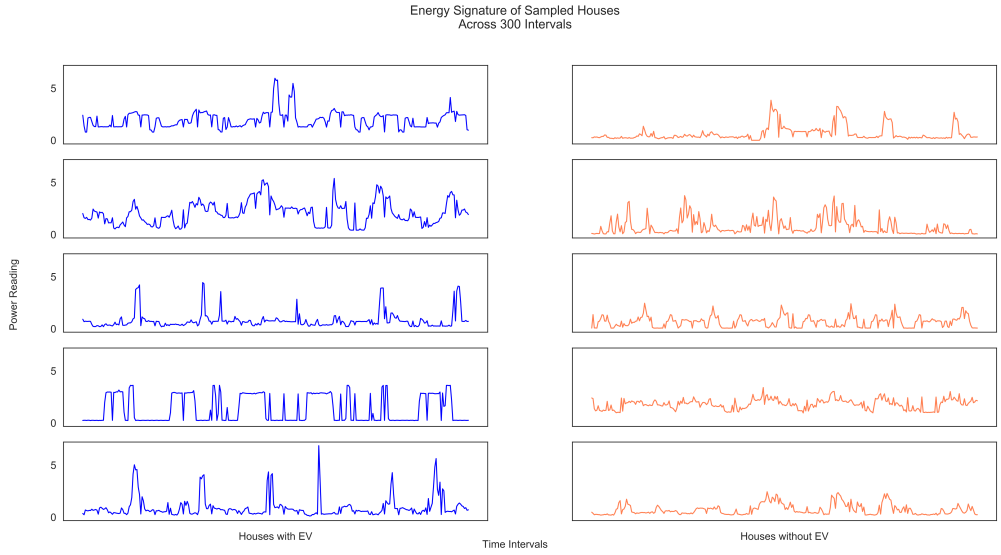


Figure 5: Energy profile of sampled households

Part A

Initially, a logistic regression model was trained with each interval in the 60-day window as a separate input variable. However, this approach presented several drawbacks. First, a prediction for a given household couldn't be made without at least 60-days (2880 consecutive intervals) of data. In a production environment where new households may be added frequently, this seems like a significant setback. Second, using 2880 input variables with similar information creates a complex model with limited analysis potential. Finally, logistic regression expects uncorrelated variables, and is restricted to linear relationships between the independent variables. To address these limitations, alternative features were engineered, table 2. The engineered features can be determined from any number

of intervals, although more intervals will likely improve prediction accuracy. Figure 6 provides the correlation coefficients for each pair of features.

Feature	**Description**
avg_pwr	Mean power reading over all intervals
avg_pwr^2	Feature 'avg_pwr' raised to the power 2
avg_pwr^3	Feature 'avg_pwr' raised to the power 3
median_pwr	Median power reading over all intervals
median_pwr^2	Feature 'median_pwr' raised to the power 2
median_pwr^3	Feature 'median_pwr' raised to the power 3
min_pwr	Minimum power reading over all intervals
min_pwr^2	Feature 'min_pwr' raised to the power 2
min_pwr^3	Feature 'min_pwr' raised to the power 3
max_pwr	Maximum power reading over all intervals
max_pwr^2	Feature 'max_pwr' raised to the power 2
max_pwr^3	Feature 'max_pwr' raised to the power 3
diff_max	Maximum difference between two adjacent intervals
diff_max^2	Feature 'diff_max' raised to the power 2
diff_max^3	Feature 'diff_max' raised to the power 3
pct_pwr<2	Proportion of intervals with power readings less than 2
pct_pwr<3	Proportion of intervals with power readings less than 3

Table 2: Part A: Engineered features

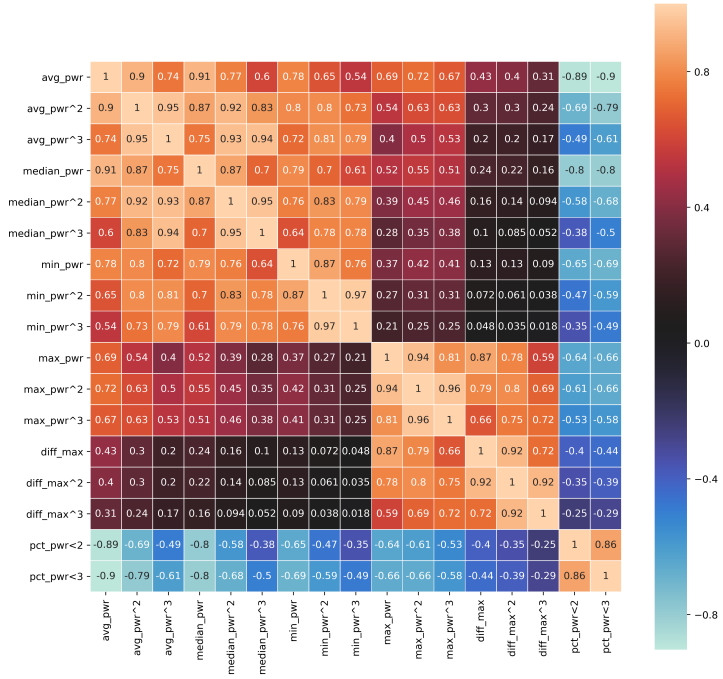


Figure 6: Part A: Feature correlation coefficients

Part B

For part B, each interval requires a separate prediction. This requirement does not lend itself to passing multiple intervals as input variables. Therefore, new features are engineered to provide additional information - beyond a singular power reading - to the model(s). Table 3 provides feature descriptions while figure 7 demonstrates the correlations.

Feature	Description
value	Power reading at a single interval
diff	Power reading difference between current and preceding intervals
h8_avg	Average power reading over preceding 8 hour interval
h24_avg	Average power reading over preceding 24 hour interval
h24_min	Minimum power reading over preceding 24 hour interval
h24_max	Maximum power reading over preceding 24 hour interval
h72_avg	Average power reading over preceding 72 hour interval
diff_2	Feature 'diff' raised to the power 2
diff_3	Feature 'diff' raised to the power 3
diff_5	Feature 'diff' raised to the power 5

Table 3: Part B: Engineered features

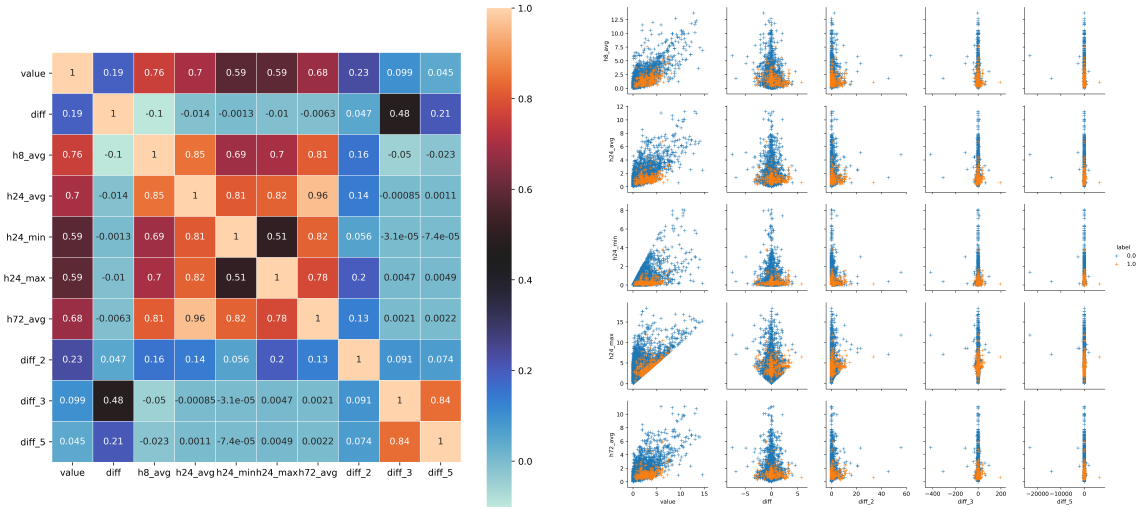


Figure 7: Part B: Feature correlation coefficients

Model Selection

Parts A and B both qualify as Binary Classification problems. Five classifiers, given below, are considered for each part (A and B) using their default configurations. All models are implemented through scikit-learn. Data is partitioned into training and validation sets once-per-part such that the data is identical for all classifiers. For initial comparison, models are evaluated using their respective `score()` methods.

- Logistic Regression (LR)
- Linear Support Vector Classification (SVM)
- Multi-layer Perceptron classifier (MLP)
- K-Nearest Neighbors (KNN)
- Random Forest Classifier (RF)

The training data is segregated according to figure 8. After removing outliers, features are constructed and predictions made for part A. Based on the originally provided labels (not predicted), houses with EVs are distilled. Using only data for houses with EVs, a secondary set of features is constructed. The data is then 'stacked' such that each unique house-interval data point can be independently classified.

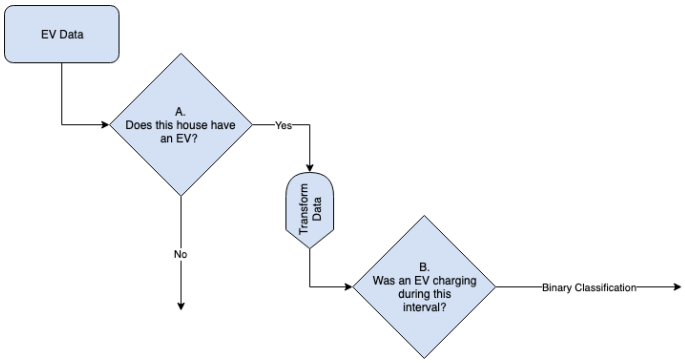


Figure 8: Classification data flow

Part A

Results are provided in Table 4. Of the models, the MLP classifier gave the highest accuracy, LR and SVM performed similarly, followed by KNN and finally the RF classifier provided the lowest accuracy. All models demonstrate slight over-fitting, with higher

training scores than test scores. Logistic regression uniquely balances reasonable performance with relative simplicity, and was therefore selected as the model for part A. The model is tuned by varying the 'Inverse of regularization strength' and 'Solver' parameters. These modifications did not significantly impact model performance.

Model	**Train Score**	**Test Score**
Logistic Regression	0.861	0.843
SVM	0.856	0.843
MLP Classifier	0.876	0.866
Random Forest	0.989	0.807
K-Nearest Neighbors	0.880	0.835

Table 4: Part A: classifier training and testing scores

Part B

Model comparison for part B is given in table 5. Models performed similarly, with only Random Forests demonstrating significant over-fitting. Because Logistic Regression is used for Part A, it was also reported on for part B. KNN was chosen for further development due to its strong score and ease of tuning.

Model	**Train Score**	**Test Score**
Logistic Regression	0.9458	0.9455
SVM	0.9452	0.9449
Neural Network	0.9575	0.9573
Random Forest	0.9964	0.9598
K-Nearest Neighbors	0.9696	0.9586

Table 5: Part B: classifier training and testing scores

Based on the model tuning results in Part A, the LR model for part B is left in its default state. The optimum number of neighbors for the KNN model is determined by evaluating performance with odd-numbers of neighbors between 1 and 19. Only the odds were evaluated, to prevent 'ties' that can occur with even neighbors. The optimum number of neighbors for the KNN model is 13.

K Neighbors	Train Score	Test Score
1.0	1.0	0.951
3.0	0.975	0.957
5.0	0.97	0.959
7.0	0.967	0.958
9.0	0.965	0.958
11.0	0.964	0.958
13.0	0.963	0.959
15.0	0.962	0.959
17.0	0.962	0.958
19.0	0.961	0.958

(a)

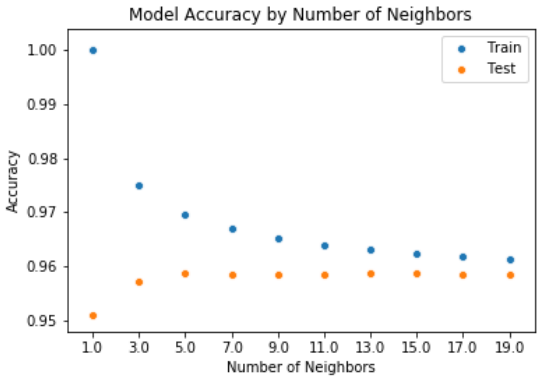


Figure 9: Feature correlation coefficients for Model B

Results

Analysis Methods

Receiver Operating Characteristic (ROC) and Precision Recall (PR) curves are used to evaluate model performance. The ROC curve demonstrates the relationship between the decision threshold and the false positive rate (FPR). The ROC curve for a perfect model gives 100% specificity at any cutoff point (Heaviside step function). The area under curve (AUC) of the ROC curve is 1 for a perfectly accurate model and 0.5 for a random model. Thus, a larger ROC AUC indicates a more accurate model.

The PR curve indicates the relationship between the Precision and Recall rates. 'Precision' measures the performance on the predicted true events whereas 'Recall' measures the performance on the actually true events. Thus Precision relates to the false positive rate (FPR) and Recall relates to the false negative rate (FNR).

Because the training data is imbalanced, the balanced accuracy score is also used to evaluate model performance. Balance accuracy is normalized by the total number of samples in each class and is therefore less susceptible to bias.

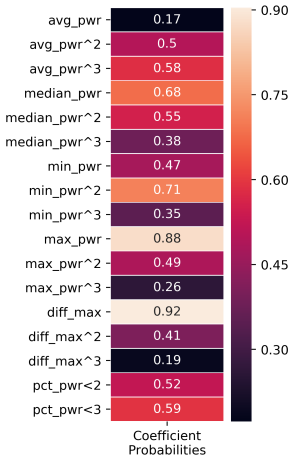
Part A: Logistic Regression Model

The logistic regression classifier for part A gave a training accuracy of 0.86 and a validation accuracy of 0.84, suggesting that over training isn't a significant concern. Measuring the area under the ROC curve (ROC AUC) gave a score of 0.90 while the average precision score was 0.797. Finally, the balanced accuracy score is the lowest, at 0.792. Thus, this model can be expected to correctly classify between 80% and 90% of samples.

Converting the intercept and coefficients from logodds into probabilities reveals the model's bias and the relative feature contributions. The intercept converts to a probability of 0.200, indicating that the model is biased towards predicting False, no EV. The most impactful features are the 'Maximum Difference', 'Maximum Power Reading' and the 'Minimum Power Reading^2'.

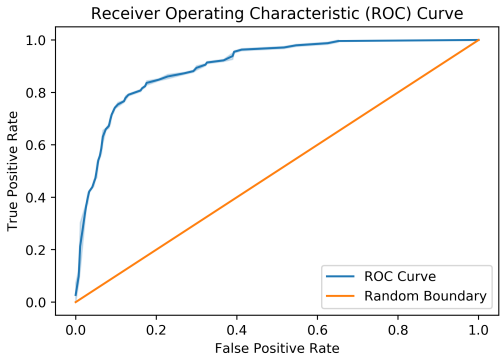
Measurement	Score
Training Score	0.861
Testing Score	0.843
ROC AUC	0.901
Average Precision Score	0.797
Balanced Accuracy Score	0.792
Linear Intercept Probability	0.200

(a) Training Scores

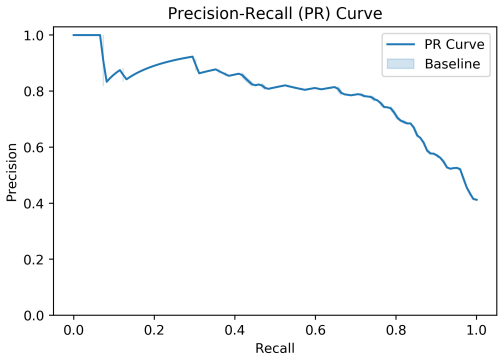


(b) Feature Coefficients

Figure 10: Part A: Performance results and model coefficients



(a) ROC Curve



(b) PR Curve

Figure 11: Part A: ROC and PR Curves

Part B: Logistic Regression Model

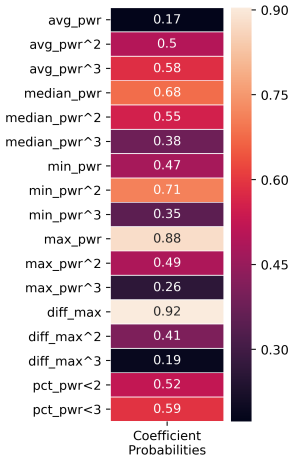
While the `score()` method and the ROC AUC score indicate accuracy >0.90, the other success metrics are less optimistic. The model's PR score is 0.624 and the balance accuracy score is 0.713. Thus, the realistic performance of this model is likely below 90%, and potentially as low as 60% to 70% accurate.

Again, the model's linear intercept and coefficients were converted from logodds to probabilities. As expected, the intercept ($p = 0.022$) confirms the classifier is highly biased toward predicting a non-EV charging event. The most indicative features are the interval power reading ('value') and the difference in power reading between the prediction interval and the preceding interval. Large power readings, and larger differences increase

the probability of an EV charging event.

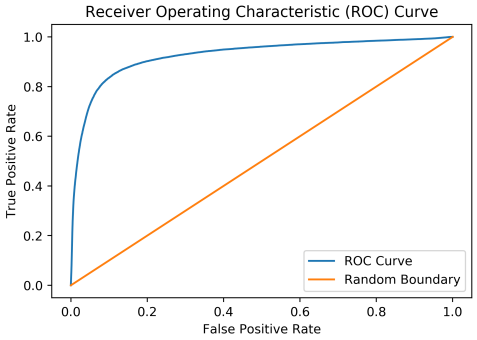
Measurement	Score
Training Score	0.946
Testing Score	0.946
ROC AUC	0.925
Average Precision Score	0.624
Linear Intercept Probability	0.022

(a) Training Scores

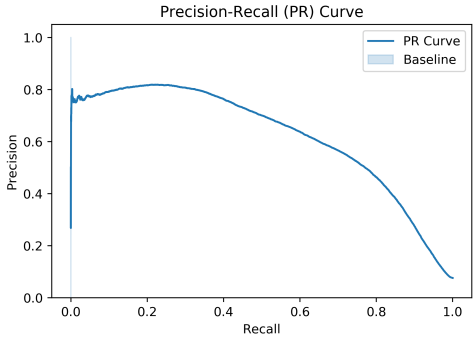


(b) Feature Coefficients

Figure 12: Part B, LR: Performance results and model coefficients



(a) ROC Curve



(b) PR Curve

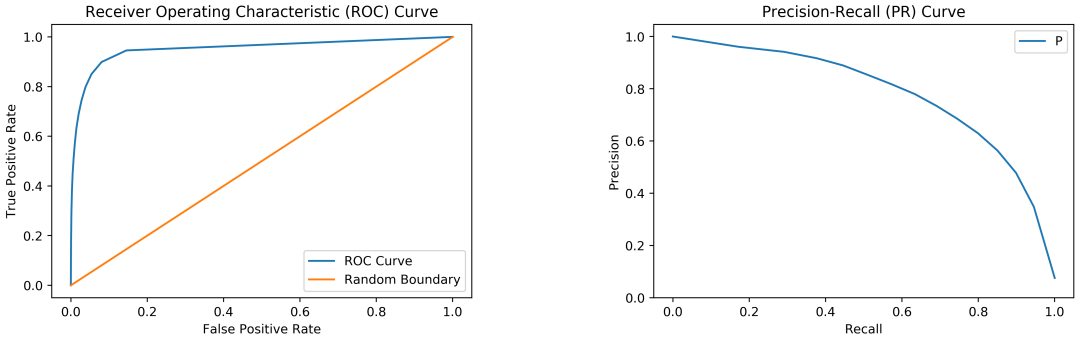
Figure 13: Part B, LR: ROC and PR Curves

Part B: K-Nearest Neighbors

The optimum number of neighbors is 13. Altering the number of neighbors did not significantly affect model performance. With 13 neighbors, both training and testing scores were 0.96. The KNN model was evaluated using ROC, PR and balanced accuracy scores. Unlike Logistic Regression, the KNN classifier does not yield linear intercepts and coefficients. The ROC AUC score is 0.951 while the PR score is 0.756 and the balanced accuracy score is 0.80. Thus, the KNN model will likely yield slightly better predictions than the LR model for part B.

Measurement	Score
Training Score	0.963
Testing Score	0.959
ROC AUC	0.951
Average Precision Score	0.756
Balanced Accuracy Score	0.80

Table 6: Training Scores



(a) ROC Curve **(b)** PR Curve

Figure 14: Part B, KNN: ROC and PR Curves

Final Test Predictions

Predictions were made for the 699 houses in the 'EV_test.csv' file. After constructing features, the data is scaled using the `StandardScaler` fitted on the training data. This ensures the scale of the inputs is consistent with the fitted coefficients and intercepts. Per the problem statement suggestion 'A solution to part B might consist of a prediction of the probability that an electric car was charging for each house and time interval in the test set', a part B prediction was made for each household, regardless of the part A results. The feature '72h_avg' restricts the part B model from making predictions on the first 144 intervals. Future iterations of this model could decrease the imposed delay to 48 hours or only 24 hours. Future iterations could also incorporate the prediction or probability from part A as a feature for part B.

Future Work

Background Knowledge

Given the mediocre classifier accuracy (all below 90% accuracy), significant improvements could be made. Adding contextual understanding may assist with further feature development. Immediate examples are (a) usage of 'charging' in the original problem statement and (b) the interpretation of 'power reading'. It is unclear whether the word 'charging' indicates that a vehicle is actively accumulating energy or whether a charging vehicle could be passively plugged in. If the 'charging' class includes passively plugged-in vehicles, the energy signature wouldn't reflect such nuance. Understanding the 'power reading' units may also add insight to the problem. Are the power readings point-in-time or cumulative over the interval? Understanding the relationship between the classification labels and the power readings may enable more sophisticated noise filtering.

Finally, the process of battery charging is complex, and influenced by on-grid and off-grid variables. Understanding the current-voltage-power relationships involved in the power readings may be beneficial. Additionally, battery charging isn't linear, and can be affected by variables such as battery capacity, age and temperature. Developing a greater understanding of these variables and their relationships will promote more informed feature development.

Additional Data Elements

Baseline electric consumption and EV ownership are subject to high environmental influence. An affluent Arizona household in August will have a significantly different energy profile than a rural Minnesota household in February. Adding contextual data to both the household and interval would likely improve model performance. Introducing (1) a date-time stamp for each interval and (2) location data (ZIP code) for each household would serve as a starting point for associating known energy-consumption influencers, such as weather patterns and affluence.

Model Optimization

Additional time could be invested selecting and optimizing the models. Regarding classifier selection, models could be optimized prior to comparison and multiple scoring methods could be used. Additionally, more effort could be invested tuning models individually.

Given the limited number of samples removed by outlier classification, additional outlier filtering would significantly improve initial scores. While removing additional outliers may improve training and validation scores, it will likely hinder the model when using production data with natural variability. In a similar vein, all models were deliberately trained with imbalanced training data. Future models could be tested with balanced training data. While this will likely reduce the models' bias, production performance may deteriorate, as the natural ratio of EV charging events to non-EV charging events is skewed proceeding data normalization.