Exploring data-driven electrocatalyst development for the Hydrogen Evolution Reaction (HER)

Rohan Mrityunjay Patil

09/10/2023

School of Mathematics,

Cardiff University

A dissertation submitted in partial fulfilment of the

requirements for **Ms Data Science and Analysis**

by taught programme, supervised by **Dr. Andrea Folli**, School of Chemistry.

I. Executive Summary

Electrocatalysts are essential in the dynamic environment of today’s sustainable energy solutions. Understanding the intricacies of catalyst performance and being able to correlate them to structural information is crucial for enabling step changes in effective energy conversion and storage. This research provides an investigation into this important area, and analyses information from two separate datasets summarising structural and performance properties of platinum-based as well as platinum-free electrocatalysts for the hydrogen evolution reaction (HER), a crucial electrochemical process in electrolytic water splitting for green hydrogen production.

The study begins with extensive data gathering procedures, delving deeply into the raw data and carefully cleaning and preparing it to make sure it is ready for in-depth investigation. We have been able to impute missing values, strengthening the datasets, by utilizing complex methodologies and domain-specific knowledge, particularly the Tafel equation.

Data preparation does not represent the end of the investigation. In order to forecast and categorize different outcomes, machine learning models are used, providing a data-driven prism through which the effectiveness of these catalysts may be examined. These models highlight the connections between various variables and offer prediction insights, shedding light on the numerous variables that affect catalyst performance.

Furthermore, the synergy between conventional domain expertise and contemporary data-driven methodologies is demonstrated by this investigation. The report establishes a new standard for research by combining these two fields and provides a guide for upcoming projects in this area.

As companies globally migrate toward greener solutions, the findings from this analysis will surely play a vital part in shaping the future of energy solutions, emphasizing the importance of data-driven catalyst exploration and rational design in this global narrative.

II. Acknowledgements

Our gratitude extends to all contributors and team members whose relentless efforts and domain expertise have shaped this in-depth analysis. Their dedication to ensuring data accuracy, quality, and the application of critical domain knowledge has been the backbone of this study.

First and foremost, I express my whole-hearted gratitude to my academic supervisor, Dr. Andrea Folli for his valuable guidance and insights throughout the project.

Last but not least, I am extremely grateful to my friends and family who have always been the best shoulders to lean on and provided moral support throughout the course tenure.

Table of Contents

IV. List of Acronyms

Pt:	Platinum
DOI:	Digital Object Identifier
EDA:	Exploratory Data Analysis

V. List of Figures

VI. List of Equation

VII. List of Tables

1. Introduction

In the pursuit of reducing fossil fuel consumption and addressing global warming, significant attention has been directed toward the utilization of hydrogen as a renewable energy source. Green hydrogen generation from water electrolysis is recognized as a promising method for producing high-purity hydrogen, which holds paramount importance for the advancement of clean energy on a global scale and the preservation of ecological environments. Central to the progress of this technology is the development of electrocatalysts that are economical, stable, and capable of high performance.

Platinum (Pt)-based catalysts have historically been the most practical and efficient for facilitating the hydrogen evolution reaction (HER). However, Pt's high cost and scarcity hinder electrochemical hydrogen production's widespread implementation. In recent years, a diverse array of non-noble metal electrocatalysts has emerged for the hydrogen evolution reaction (HER), often exhibiting promising catalytic activities towards HER. These encompass metal carbides, nitrides, phosphides, and two-dimensional (2D) materials, including metal oxides, layered double hydroxides (LDHs), graphene, MXenes, phosphorene, graphitic carbon nitride (g-C3N4), and transition metal dichalcogenides (TMDs). Although alternatives to precious metals should be continuously investigated, recent endeavours advocated that ultra-low loading of precious metals such as Pt, Ru, Rh, Pd, and Ir as single-atom catalysts, could potentially be a good compromise between sustainability, cost-effectiveness, and high performances often exhibited by this group of metals.

Traditionally, the screening and evaluation of electrocatalysts have relied on quantum mechanical (QM) calculations, guided by human intuition and empirical experimentation. However, these conventional methods demand substantial time investments. To meet the imperative of rapid technological advancement necessary to achieve Net Zero emissions by 2050, a paradigm shift in catalyst screening and development approaches is necessary. Machine learning (ML) techniques, which combine and integrate computational power with human learning processes, have revolutionized catalyst development. ML tools like Support Vectors (SV), Random Forest (RF), Gradient Boosting, K-nearest neighbor (KNN), Decision Trees (DT), Stochastic Gradient Descent (SGD), and Neural Networks (NN) have been utilized to predict catalytic efficiencies, including activity, selectivity, stability, and performance. Current efforts are particularly focused on the initial screening and characterization phases before actual fabrication into catalytic materials. These computational screening and synthesis concepts have given rise to reverse design strategies, now employed in the design and synthesis of drugs, synthetic organic compounds, photovoltaics, and various solid-state materials. Material development cycles, bolstered by the integration of data with computational analytical tools, are emerging as potent instruments for expediting discoveries. In support of these innovative catalyst development strategies, databases and information repositories (e.g., Catalysis-Hub, Open Catalyst Project, Materials Genome Initiative, etc.) have been established. These repositories house a wealth of data derived from stoichiometric studies, kinetics experiments, and computationally generated data concerning the structural, electrical, optical, and thermodynamic properties of catalysts. Computational materials databases have progressively become instrumental in associating materials with their structural attributes, thereby translating the intrinsic value of big data into tangible solutions and data-driven discoveries of novel materials.

2. **Literature Review**

2.1 Machine Learning in Catalysis:

There has been a paradigm shift in how researchers approach catalyst discovery and mechanistic understanding as a result of the growing convergence of machine learning (ML) and catalysis. Trial and error was frequently used in traditional approaches, which, while effective, required a lot of work and time. With great care, they assembled a comprehensive dataset from the Catalysis-hub that included over 11,000 data points. They were able to forecast hydrogen evolution reaction (HER) activity with astounding precision using six different ML models. Their work demonstrates the value of feature engineering as well as the promise of ML in expediting catalyst identification. They improved predicted accuracy by transforming elemental qualities into compound features, providing the groundwork for future research projects.

2.2 Deep Learning for Catalysis Prediction:

The possibilities in the field of catalysis have been expanded thanks to deep learning, a more sophisticated subset of machine learning. Deep learning models add a new level of analysis due to their inherent capacity to independently extract and learn from characteristics. This possibility was employed convolutional neural networks (CNN) to forecast catalytic performance. Their understanding of the problems with limited data and the creative solutions, like data augmentation, to solve them, provide a direction for future research in this area.

2.3 ML-accelerated HER Catalytic Activity Prediction:

A comprehensive workflow for applying machine learning to predict HER catalytic activity in binary alloys was described by the authors in another important paper. They took advantage of the Catalysis-hub database's data, obtaining crucial details such as chemical compositions and GH* values calculated using DFT. In-depth feature engineering was used to convert the raw data in this vast dataset into useful structural and electrical attributes. Out of the six ML models they used, the LGB model stood out as having the highest level of accuracy. Their use of Shapley Additive explanations (SHAP) gave their findings even more depth by enabling them to identify the precise contributions of different aspects. The effectiveness of integrating conventional DFT methods with contemporary ML approaches is demonstrated by this work.

2.4 Machine Learning Aided Synthesis and Screening of HER Catalyst:

A thorough review titled "Machine Learning Aided Synthesis and Screening of HER Catalyst: Present Developments and Prospects" has carefully evaluated the landscape of ML-aided catalyst synthesis. The authors examine the nuances of several ML models, evaluating their advantages and disadvantages. They make a strong case for how ML, particularly regression algorithms like LGB, XGB, and RFR, can close the gaps left by conventional catalyst finding techniques. The review is a useful tool since it offers information on present approaches, potential new discoveries, and the overall influence of ML in the field of catalysis.

2.5 Broader Perspectives:

The overlap of ML and catalysis extends beyond the topics described above. Numerous research projects are under progress that look into the specifics of ML algorithms, improve methodology, and use these algorithms in a variety of real-world situations. Everyone agrees that ML's sophisticated algorithms and computational capabilities are ready to change the course of humanity. The potential for ground-breaking discoveries in catalysis through ML is limitless as data becomes more available and algorithms become more complex.

2.6 Model working

2.6.1 Logistic Regression:

Overview: A statistical technique called logistic regression is used to examine datasets where the outcome variable is categorical. Contrary to its name, logistic regression is employed to solve classification issues rather than regression ones. It calculates the likelihood that a specific incident falls into a given category.

Working Principle: Any real integer can be used to map the logistic function (also known as the sigmoid function), an S-shaped curve, between 0 and 1. When attempting to forecast probabilities that are bounded between these values.

Equation:

Logistic Function (Sigmoid):

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}$$

Equation 1 Logistic Function

Here, p(X) is the probability that the dependent event occurs, and β0,β1 are the model parameters.

Logit Transformation (Inverse of the logistic function):

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X$$

Equation 2 Logit transformation

The odds logarithm, sometimes known as the log-odds, is represented by this equation. A linear link between the predictor variables and the transformed response is ensured by the log-odds transformation.

Cost Function:

Logistic regression uses the binary cross-entropy (log loss) as the cost function:

$$J(\beta) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(p_i) + (1-y_i)\log(1-p_i)]$$

Equation 3 Cost Function

where yi is the actual class, pi is the projected probability, and N is the sample size. Finding 0, 1,... values that minimize this cost function is the aim of the training procedure.

Model Training: The parameters 0, 1,... that minimize the cost function are found using Gradient Descent or another optimization algorithm. To do this, the parameters must be modified iteratively in order to reduce the discrepancy between the anticipated probabilities and the actual classes in the training data.

Multiclass Classification: A version of logistic regression known as "Softmax Regression" or "Multinomial Logistic Regression" is used for issues where the outcome can belong to more than two classes. It expands the notion to include several classes.

Regularization: L1 (Lasso) and L2 (Ridge) regularization techniques can be used to regularize logistic regression to prevent overfitting, especially when the number of features is large. By doing this, the cost function is given a penalty, effectively reducing the magnitude of the parameters.

Advantages: The technique can be regularized to prevent overfitting, and the outputs have a probabilistic interpretation. Stochastic gradient descent allows for simple updating of logistic models with new data. Simplicity and high interpretability.

Disadvantages: Supposes a linear decision boundary, which may or may not be accurate. Not as effective as more complicated models, particularly when dealing with situations with non-linear bounds. vulnerable to irrelevant features.

A fundamental algorithm in statistics and machine learning is logistic regression. learning it is essential since it not only accomplishes categorization jobs but also acts as a foundation for learning more complicated algorithms.

2.6.2 Decision Tree Classifier:

Overview: An internal node represents a feature (or attribute), a branch represents a decision rule, and each leaf node represents an outcome in a decision tree, which resembles a flowchart. The root node in a decision tree is the first node from the top. It gains the ability to divide data based on attribute values, and this recursive process results in a decision-tree-like model.

Working Principle: Decision Trees make judgments at each node to divide the data into subsets depending on the most important attribute or attributes. Recursively repeating this approach produces a decision-tree-like model.

How a Split Decision is Made:

Entropy: In order for a decision tree to function, the source set must be divided into subsets according to the value of an attribute. To determine a sample's homogeneity, one uses entropy. The sample has an entropy of one if it is evenly divided and zero if it is totally homogeneous.

$$H(S) = -p_{+} \log_2(p_{+}) - p_{-} \log_2(p_{-})$$

Equation 4 Entropy

Obtaining Information: The entropy of a set measures how disorderly it is. Finding the attributes that yield the most information gain is the basic goal of utilizing decision trees as classifiers.

$$IG(S, A) = H(S) - \sum_{t} \left(\frac{|S_t|}{|S|} \times H(S_t) \right)$$

Equation 5 Information gain formula

Gini Impurity: The CART algorithm also uses this metric. A node is considered pure if it has a Gini Impurity of 0, which means that it only has data from one class.

$$Gini(S) = 1 - \sum (p_i^2)$$

Equation 6 Gini Impurity

Building a Decision Tree:

- Start with the dataset at the root.
- Find the best attribute in the dataset using Information Gain or Gini Impurity.
- Divide the data into categories according to the best attribute's value.
- Until one of the stopping requirements is satisfied (such as a maximum depth or all of the data in a node being of the same class), recursively repeat the process for each subset using only the data in that subset.

Pruning: In order to capture data noise and prevent overfitting, decision trees can be extremely deep and complex. Parts of the tree that lack the ability to accurately forecast target values can be removed using pruning techniques like reduced error pruning or cost complexity pruning.

Advantages:

Interpretability: Tree visualizations are simple to comprehend and perceive. They make sense when interpreted from the top down. No normalization or dummy variables are necessary for the minimal data preparation.

Non-parametric: Does not rely on prior knowledge of the distribution of the data.

processes data that is both numerical and categorical.

Disadvantages:

Overfitting: They can readily capture noise and overfit to the training data if pruning is not performed.

Instability: Minor modifications to the data could produce a completely different tree.

Optimization Finding a "optimal" decision tree for a piece of data is a challenging computing task.

Ensemble Methods: Because Decision Trees are unstable, they are frequently utilized as building blocks in ensemble methods like Random Forests or Gradient Boosting Machines, which construct numerous trees and combine their outputs to increase stability and accuracy.

2.6.3 Random Forest Classifier:

Overview: An ensemble learning technique called Random Forest builds a "forest" of decision trees during training and outputs the class that is the mean (in regression) or the mode (in classification) of the classes produced by individual trees.

Key Concepts: Bootstrap Aggregating (Bagging): Random Forests make use of the bagging technique, which divides the dataset into several subsets by employing sampling with replacement. An individual decision tree is trained using each of these subgroups.

Feature Randomness: Random Forests add an additional layer of unpredictability on top of data subset randomness. When dividing a node, it looks for the best feature from a random subset of features rather than the most crucial one. As a result, the trees become more diverse and are further decorated.

Working Principle:

Training: Bootstrap samples are taken (with replacement) from the training set. A decision tree is developed for every single one of these bootstrap samples. However, only a random subset of traits are taken into account for splitting at each node. Without pruning, each tree grows to its full potential.

Prediction: It has been transmitted to every tree in the forest for a fresh data point. For classification problems, each tree provides a class prediction. The forest makes its final prediction based on the class that received the most votes (out of all the trees in the forest).

Mathematical Formulation: The randomness added during feature selection and dataset sampling aids in the development of numerous de-correlated trees. These trees are averaged, which lowers volatility and enhances prediction accuracy. The mode of the output classes of distinct trees serves as the final output class for categorization.

Advantages:

- High Accuracy: The model is less likely to overfit because of the numerous trees.
- Large datasets with increased dimensionality are handled effectively.
- Missing Values Treatment: Capable of managing missing values during both training and prediction.
- Provides information about the significance of a feature.
- Versatility: Applicable to both classification and regression issues.

Disadvantages:

- Complexity: Can be computationally demanding, particularly when there are many trees involved.
- Interpretability: A forest is more complicated to interpret than a single tree.
- Predictions are made more slowly than they are for individual decision trees because of the ensemble nature of the model.

Hyperparameters:

Among the important hyperparameters used in Random Forest are: Estimated number of trees in the forest, n_estimators.

- max_features: The maximum number of features to take into account when determining the optimal split.
- max_depth: The tree's maximum depth.
- Minimum number of samples needed to divide an internal node is indicated by the variable min_samples_split.
- min_samples_leaf is the bare minimum of samples that must be present at a leaf node.

Underlying Assumption: Random Forest's core tenet is that a collection of "weak learners" (individual decision trees that perform marginally better than random guessing) can combine to create a "strong learner" (an ensemble with high accuracy and generalization).

2.6.4 Support Vector Machine (SVM):

Overview: Although it can also tackle regression issues, Support Vector Machine (SVM) is a supervised machine learning technique generally used for classification jobs. SVM's primary goal is to identify the hyperplane that optimally separates a dataset into classes.

Key Concepts: A flat affine subspace of dimension N-1 is referred to as a hyperplane in an N-dimensional space. The hyperplane, for instance, is a line in a two-dimensional space. The decision boundary is what distinguishes various classes in SVM. Indicators of Support The hyperplane's nearest data points that affect its position and orientation. These support vectors serve as the basis for determining the hyperplane.

Margin: The separation between the nearest data point from either class and the hyperplane. The SVM seeks to increase this margin.

Working Principle:

Linear SVM: It identifies the optimal hyperplane for classifying the dataset. The hyperplane that increases the margin between two classes is considered to be the "best" one.

Non-linear SVM: The data is frequently not separable linearly. Here, SVM employs a kernel trick to convert the input space into a higher-dimensional space that can then be used to locate a hyperplane.

Mathematical Formulation:

Given labeled training data (xi,yi), where xi is a vector in the input space and yi represents one of the two classes that the point xi belongs to—either 1, or -1. The following optimization issue is resolved by SVM:

$$\begin{aligned} \text{Minimize:} \quad & \frac{1}{2}||w||^2 \\ \text{Subject to:} \quad & y_i(w \cdot x_i + b) \geq 1 \quad \text{for all } i \end{aligned}$$

Equation 7 SVM Formula

Where:

- w is the normal vector to the hyperplane.
- b is the bias.

The constraint ensures that each data point lies on the correct side of the hyperplane.

Kernel Trick:

SVM uses the kernel trick to implicitly map input data into a higher-dimensional space for non-linear data. Typical kernels consist of:

Linear Kernel: $K(x,y)=x\cdot y$

Polynomial Kernel: $K(x,y)=(1+x\cdot y)d$

Radial Basis Function (RBF) or Gaussian Kernel: $K(x,y)=e-\gamma||x-y||2$

Sigmoid Kernel: $K(x,y)=\tanh(kx\cdot y+c)$

Advantages:

- Effective in High Dimensional Spaces: Performs well even when there are more samples than dimensions.
- Robust to Outliers: Because only the support vectors have an impact on the decision function, outliers barely register.
- Memory Effective: Because it only uses a portion of the training data (support vectors), it is memory effective.
- Versatile: The decision function can be provided with a variety of kernel functions.

Disadvantages:

- Sensitive to Noisy Data: SVM may perform poorly when classes overlap.
- No Direct Probability Estimates: SVM does not directly offer probabilities.
- Computationally demanding: SVM can be slow and memory-intensive for larger datasets.

Hyperparameters:

Some of the key hyperparameters in SVM include: parameter for regularization, C. A wider margin is produced by a lower value of C, which could result in more erroneously classified points. A higher C value results in a narrower margin, which properly identifies more training points. Specifies the type of kernel to be used (linear, poly, rbf, or sigmoid). Gamma is the 'rbf', 'poly', and'sigmoid' kernel coefficients. determines the decision boundary's form.

Degree: Polynomial kernel function ('poly') degree. The majority of kernels disregard it.

Underlying Assumption: SVM makes the assumption that the data is linearly separable when projected onto a higher-dimensional space (if necessary), enabling the algorithm to identify a hyperplane that separates the data into its many classes.

Due to its versatility in handling both linear and non-linear data as well as its focus on optimizing the margin, SVM has become a popular option for classification problems across a variety of industries. It is a solid option for a variety of real-world problems due to its mathematical rigor and versatility.

2.6.5 Gradient Boosting Classifier (GBC)

Overview: Gradient Boosting is an ensemble learning technique that combines a number of weak predictors (usually decision trees) to create a strong predictor. In order to fix the mistakes created by current models, new models are included.

Key Concepts:

- Boosting: A generic ensemble technique that strengthens poor learners by emphasizing examples that are challenging to categorize.
- Decision trees are typically used for weak learners (though sometimes simply a stump would do).

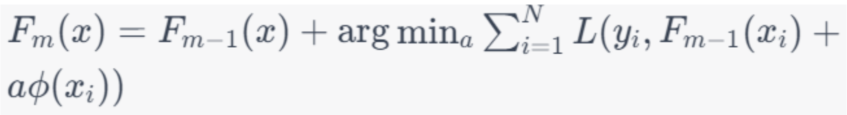
- Gradient Boosting: In this technique, trees are built one at a time, with each one helping to fix mistakes in previously trained trees.

Working Principle:

- On a portion of the data, a model is created.
- On the entire dataset, predictions are made using this model.
- By contrasting the predicted values with the actual values, errors are computed.
- The residuals (errors) from the preceding phase are predicted by a new model.
- This augments the results of the forecasts made by the earlier model.
- Until a stopping criterion, such as the quantity of trees or size of residuals, is reached, steps 2 through 5 are repeated.

Mathematical Formulation:

Given a differentiable loss function L(y,F(x)), the method works by adding classifiers to a running total:



$$F_m(x) = F_{m-1}(x) + \arg\min_a \sum_{i=1}^{N} L(y_i, F_{m-1}(x_i) + a\phi(x_i))$$

Equation 8 GBC Formula

Where:

- yi is the actual output.
- Fm(x) is the boosted model after m iterations.
- ϕ(xi) is the basis function (a weak learner, typically a decision tree).
- a is the weight of the basis function.

Loss Functions:

The choice of loss function depends on the type of problem:

- Regression: Mean squared error
- Classification: Logarithmic loss

Regularization: To regularize the model, gradient boosting incorporates both shrinkage (learning rate) and tree limitations (such as depth and leaf count).

Advantages:

- Flexibility: Applicable to both classification and regression tasks.
- Handling Missing Data: Capable of handling missing values naturally without imputed data.
- Feature Importance: Offers information on which features have the greatest bearing on forecasts.
- Robust: Because it incorporates the predictions from other trees, it is less prone to overfitting.

Disadvantages:

- Due to the sequential construction of trees, computational complexity can be costly and time-consuming.
- Sensitivity of the hyperparameters: The hyperparameter settings may have an impact on the performance.
- Gradient boosting might overfit the training set if the tree size is not properly constrained or regularized.

Hyperparameters:

Key hyperparameters in Gradient Boosting include:

- The total number of sequential trees that must be modeled is indicated by the n_estimators field.
- A factor that causes each tree's predictions to be shrunk is the learning rate (shrinkage). Lower values strengthen the optimization.
- Max Depth: The deepest point at which any given tree can be found.
- Subsample: The percentage of data to be used in each tree's fitting. Overfitting is avoided by setting it to values lower than 1.

Underlying Assumption: Gradient Boosting makes the supposition that the final model will produce a lower cumulative error by sequentially repairing the errors of prior trees. What sets gradient boosting apart from other methods is the iterative corrective procedure.

The Gradient Boosting Classifier is an effective method, particularly when we need our model to perform well and have a lot of data. Its strength comes from merging the predictions of various decision trees, each of which may not be a great predictor on its own, into an ensemble prediction that is more precise.

2.7 Web Scraping – The Extraction Process:

The data is so large and dispersed throughout so many research articles that manual extraction would take a long time and be prone to human mistake. Therefore, we used web scraping techniques to ensure consistency and effectiveness.

Data extraction from websites is accomplished through the use of web scraping. It entails making HTTP calls to a certain website URL, retrieving its content, and then processing the necessary data. We used the well-known Python module Beautiful Soup for our needs. It converted the intricate HTML designs of the scholarly websites into a tree of Python objects, like tags and strings. We were able to efficiently navigate and extract particular data points thanks to this.

It's important to note that web scraping has its own unique set of difficulties. Academic platforms frequently have complex systems, especially ones that are as well-known as the ones we used. Additionally, some of them use dynamic content loading, which makes certain material only visible after taking certain actions, such scrolling or clicking a button. Although managing dynamic material needed more sophisticated tools, our major focus remained on statically loaded content. Beautiful Soup is skilled at parsing static stuff.

Data Extraction Specifics:

The extracted datasets were mostly made up of tabular data. For instance, the link in the Nano Convergence Journal led straight to a table with pertinent information on electrocatalysts, their characteristics, and performance indicators. We located the table> HTML tag using Beautiful Soup and carefully extracted the data row by row.

Not all information was, however, easily available. Some systems have data scattered throughout the text, necessitating a more subtle method of extraction.

Ethical and Respectful Data Extraction:

Although effective, web scraping raises ethical questions. We had to make sure that our data extraction techniques adhered to the conditions of use of the relevant websites. For information regarding permitted scraping activities, we checked the robots.txt file on each website. Furthermore, our scraping scripts were created to make requests at considerate intervals, preventing any unexpected denial-of-service scenarios. This was done to guarantee we didn't overwhelm any servers or unintentionally disrupt services.

Data Structuring:

Data must be organized into a format that can be analyzed after it has been obtained. Each row in the dataset appears to represent a specific catalyst's performance under a set of circumstances, and the columns represent attributes like "Electrocatalysts," "Electrolytes," and "Current Density."

3. Methodology

3.1 Pt Alternative Electrocatalysts

The methodology used for this investigation smoothly combines domain knowledge with advanced analytical tools to ensure a thorough examination of the relevant datasets. These are the steps:

3.1.1 Data Acquisition**:**

Accessing External Storage: Google Drive was mounted to allow direct access to the datasets stored there using Google Colab's features. This offers the benefit of directly using datasets stored in the cloud rather than downloading them to local storage, enabling quick and effective data retrieval.

3.1.2 Data Loading:

The dataset "Pt alternative dataset.xlsx" was loaded into the working environment using the pandas package, which is recognized for its abilities to handle structured data. We made sure the Excel file's data structure and formatting were preserved by utilizing the read_excel function of pandas.

3.1.3 Initial Data Exploration*:*

Data Overview:

The dataset's first few rows were quickly captured using the head() function. This is essential for gaining a general knowledge of the structure, characteristics, and potential patterns of the data. By providing a glance into feature names, data types, and sample values, it helps to prepare the ground for future in-depth analysis.

Understanding Dataset Dimensions:

The dimensions of the dataset were evaluated using the shape attribute. It is essential to know how many rows and columns there are because:

- Explains the size of the dataset, which can have an impact on computing and processing methods in the future.

- Reveals the quantity of data points (rows) that will be used for the analysis as well as the number of features that are available for analysis.

3.1.4 Data Preprocessing and Cleaning*:*

Handling Missing Values: Using the isnull() function, the dataset was examined for missing values. To determine which columns have missing entries, use the sum() technique.

Data Transformations:

Current Density Transformation: To make the scale compatible with upcoming studies, the values in the Current Density column were divided by 100.

Overpotential Transformation: To provide a more understandable scale for this particular statistic, the values from the Overpotential column were multiplied by 1000 to change them from volts (V) to millivolts (mV).

Tafel Slope Transformation: To guarantee that the values are on the proper unit scale for in-depth study, the Tafel Slope column values were multiplied by 1000.

Tafel Equation for Missing Value Imputation: In particular instances, the Tafel equation was used to impute missing values. In order to apply the Tafel equation and compute the missing values for "Tafel slope" and "Overpotential" using the dataset's existing values, the compute_missing_values Python function was created. By doing this, it is made sure that the data imputation is not random but rather is based on domain-specific information.

Categorical Data Imputation: Missing values for categorical columns like "Electrocatalysts," "Electrolytes," and "Structure" were substituted with the mode (most common value). Due to its low bias, this method is frequently used to impute missing categorical data.

3.1.5 Exploratory Data Analysis (EDA):

Histogram and Box Plot Analysis: To see the distributions of the "Tafel slope" and "Overpotential" columns, histograms were displayed. The dispersion of the data, any potential skewness, and the existence of outliers are all better understood with the help of these histograms.

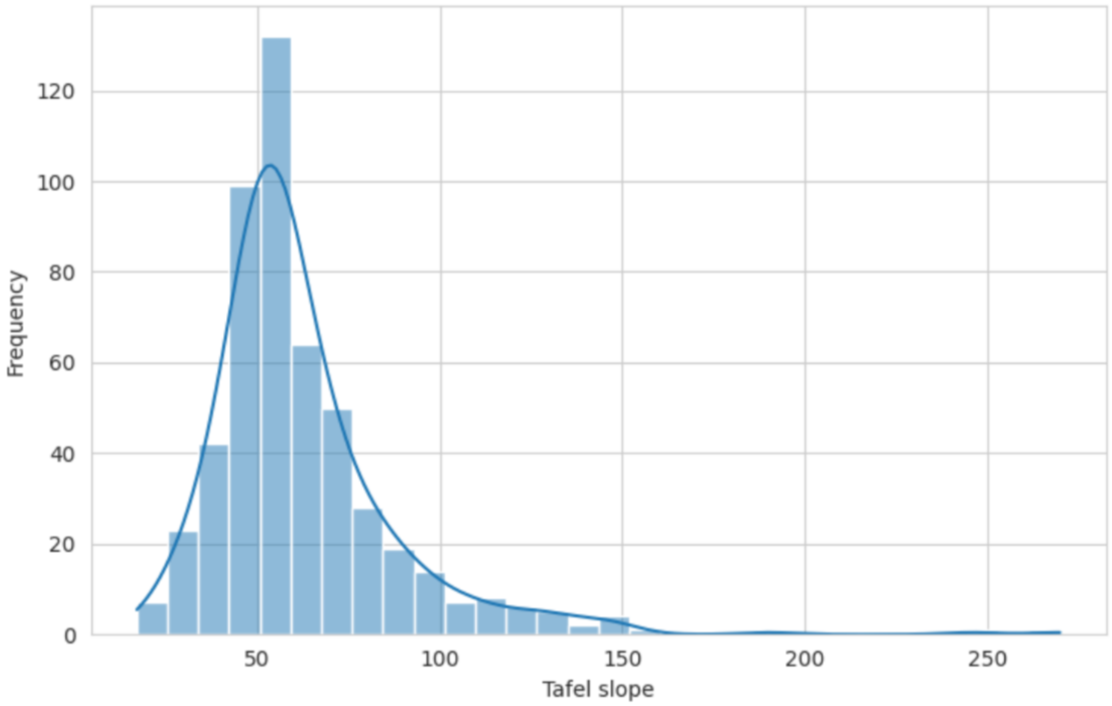


Figure 1 Histogram of Tafel Slope distribution

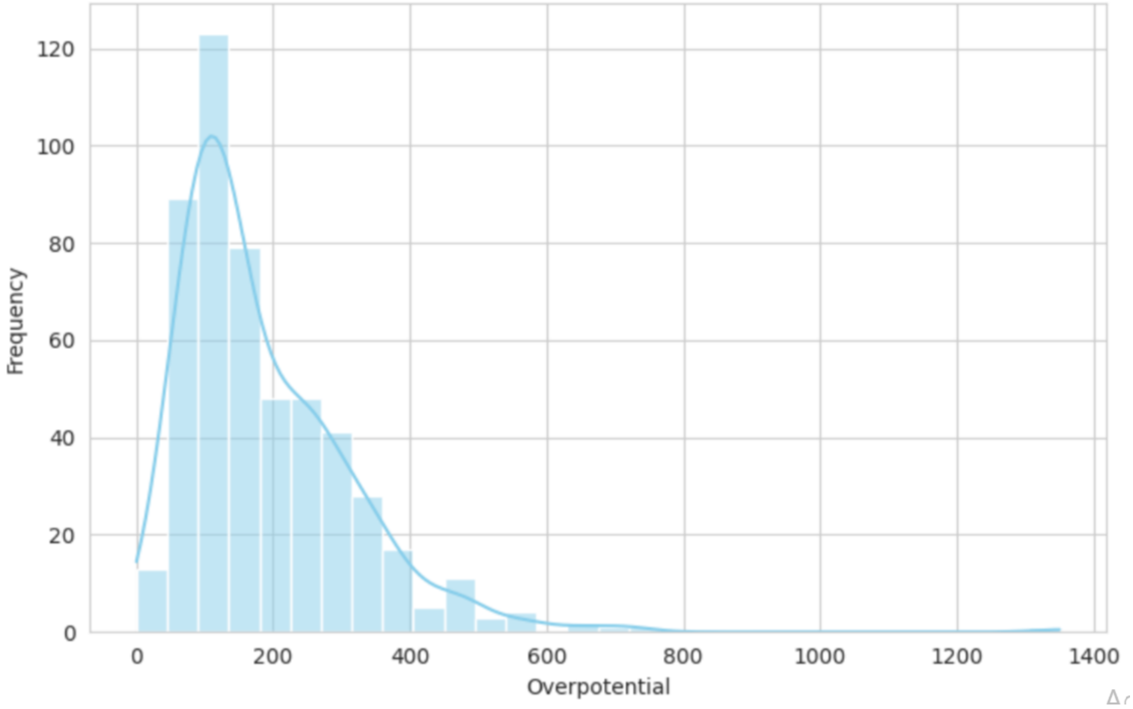


Figure 2 Histogram of Overpotential Distribution.

The same columns were also shown as box plots to examine the data distribution in more detail and specifically pinpoint outliers. The minimum, first quartile, median, third quartile, and maximum values in a dataset are visually summarized by box plots. The presence of dots outside the box plot's "whiskers" denotes possible outliers.

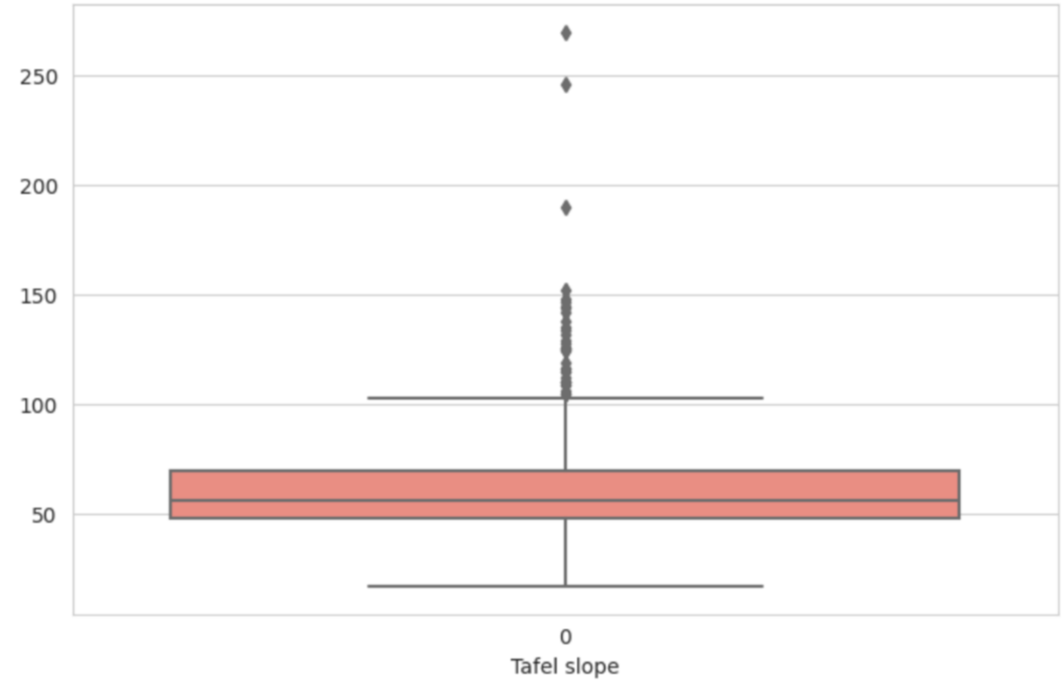


Figure 3 Box Plot of Tafel Slope Distribution.

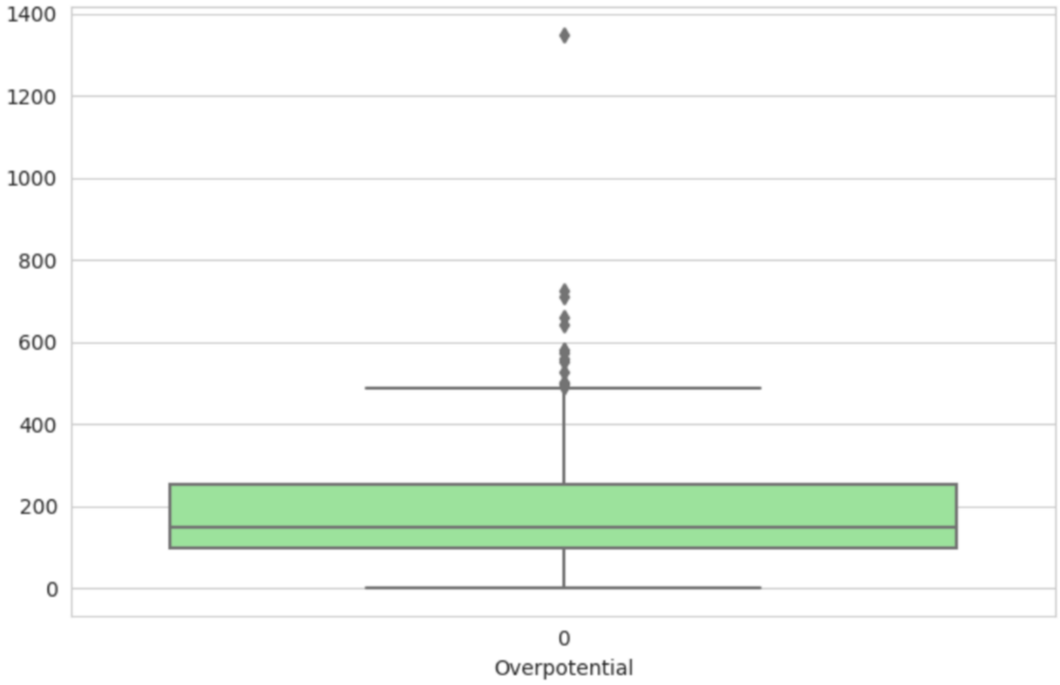


Figure 4 Box Plot of Tafel Slope Distribution

3.1.6 Data Preprocessing and Cleaning:

Categorical Data Imputation: Missing values for categorical columns like "Electrocatalysts," "Electrolytes," and "Structure" were substituted with the mode (most common value). Due to its low bias, this method is frequently used to impute missing categorical data.

Data Transformation:

Label Encoding: Label encoding was used to turn categorical columns (such as "Electrocatalysts," "Electrolytes," and "Structure") into numerical values that could be used by machine learning techniques.

Data Splitting:

Training and Test Split: A 70:30 ratio was used to divide the data into training and test sets. To make sure that the distribution of labels in the training and test sets is comparable, stratified sampling was used. To guarantee that the model learns from a representative sampling of the dataset, this is essential.

3.1.7 Model Building:

Model Selection:

Five different machine learning models were selected for the analysis:

- Logistic Regression
- Decision Tree Classifier
- Random Forest Classifier
- Support Vector Machine
- Gradient Boosting Classifier

Model Training: Each of these models was trained using the training data.

3.1.8 Model Evaluation:

Performance Metrics Calculation: The multi-class ROC AUC score, a crucial statistic for gauging the effectiveness of classification models, particularly when the classes are unbalanced, was calculated using a bespoke function. On the test set, predictions were made for each trained model, and several measures, including:

- Accuracy
- F1 Score (weighted)
- ROC AUC Score (for multi-class classification)
- Detailed Classification Report (providing precision, recall, and F1 score for each class)

Data Exploration:

Data Sorting for Exploration: The dataset was sorted using the 'Alternative_Label_Top_5' column in addition to other important metrics like 'Overpotential' and 'Tafel slope'. This stage will probably result in a more ordered view of the data, facilitating simpler exploration and pattern recognition.

3.2 Pt Based Electrocatalysts

3.2.1 Descriptive Analysis

Dataset Overview:

Dimensions and Structure: Prior to digging in, we examined the dataset's structure to learn more about the number of rows and columns it contains. This helped appreciate the level of detail in the data and the range of information provided.

Feature Types: We were able to strategically plan further preprocessing and analytic processes that were specific to each type of data by distinguishing between continuous, categorical, and ordinal variables. Defining the different data types (such as integers, floats, and objects) allows one to establish the strategy for the upcoming preprocessing stages. For example, understanding which columns are categorical can influence encoding choices.

Statistical Summary: Central Tendencies: By looking at statistics like the mean and median, we were able to identify the central tendency of the data, which acted as a guide for later analysis. The dataset's central tendency, spread, and distributional shape are determined using techniques like.describe(). This process can reveal potential outliers or unexpected data points that require more research.

Metrics for measuring variability: By looking at the range, variance, and standard deviation, we were able to gauge the consistency and dispersion of the dataset. While low variability might suggest redundant or stable features, high variability might suggest outliers or varied data sources.

3.2.2 Handling Missing Values

Identification: Heatmaps and missing data plots provided a clear view of the locations of the data gaps, enabling the use of targeted imputation techniques. Where missing values appear can be seen using tools like heatmaps or bar plots, indicating patterns of missingness. For instance, it can be a sign of systemic data collecting problems if several rows have multiple missing values in different columns.

Quantitative Metrics: We determined the percentage of missing values for each feature and then ranked our imputation techniques according to the severity of the missing data. A thorough profile of the missing data is generated before any imputation. Understanding the percentage of missing data in each column helps to determine whether to use imputation or maybe remove columns.

Informed Imputation: The type and cause of missingness dictate the method rather than using generalized imputation strategies. For instance, mean imputation might be appropriate when a value is missing since it wasn't recorded. However, a zero or another domain-specific value could be more acceptable if it's absent since it didn't happen.

Strategic Imputation:

Domain-Driven Techniques: We could utilize one characteristic to anticipate the missing values of another by taking into account the intrinsic correlations that some features had, especially if domain knowledge—like the Tafel equation—suggested such relationships.

Machine Learning Techniques: To ensure that complex imputations are based on the data's natural patterns, models like K-Nearest Neighbors or Random Forest may be used to predict and fill missing values.

Unit Conversion

Current Density Transformation: The Current Density column values were multiplied by 1000. In order to make the existing density values compatible with other measurements and studies, a scale adjustment was applied.

Overpotential Transformation: The numbers from the Overpotential column were multiplied by 1000. This transformation provided a more precise and understandable scale for this particular statistic by converting the data from volts (V) to millivolts (mV).

Tafel Slope Transformation: The Tafel Slope column's values were multiplied by 1000. The Tafel slope data are now in the correct unit scale for in-depth investigation thanks to this improvement.

3.2.3 EDA - Visual Exploration

Histograms:

Skewness Analysis: By analyzing the tails of these distributions, we were able to determine the type and direction of skewness, which helped us determine how to approach prospective modifications.

Kurtosis Insights: The data's kurtosis was shown by a peaked or flat histogram, which can affect several statistical tests or model assumptions.

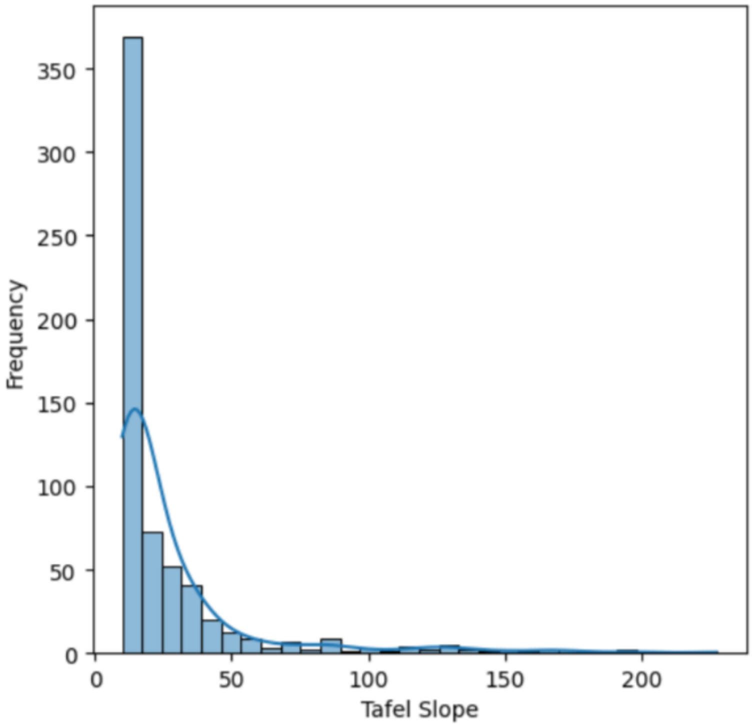


Figure 5 Distribution of Tafel Slope

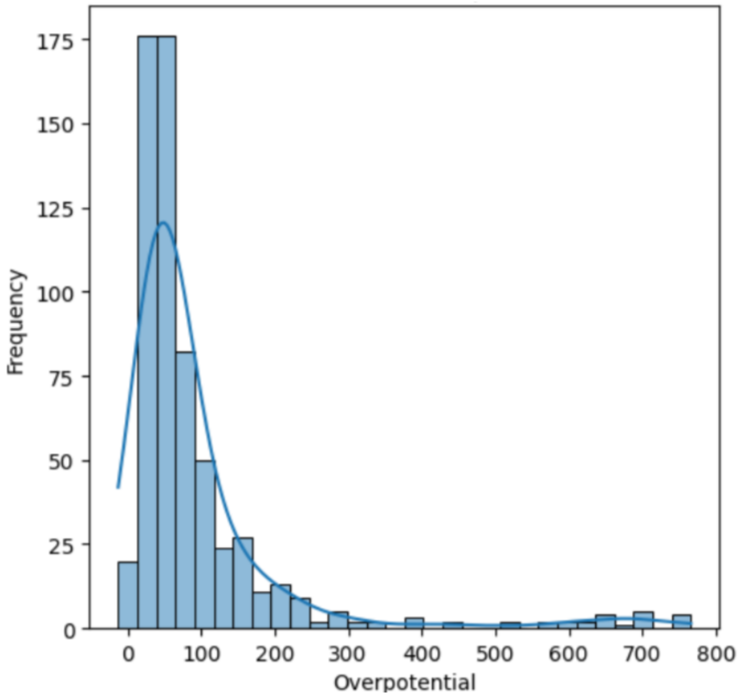


Figure 6 Distribution of Overpotential

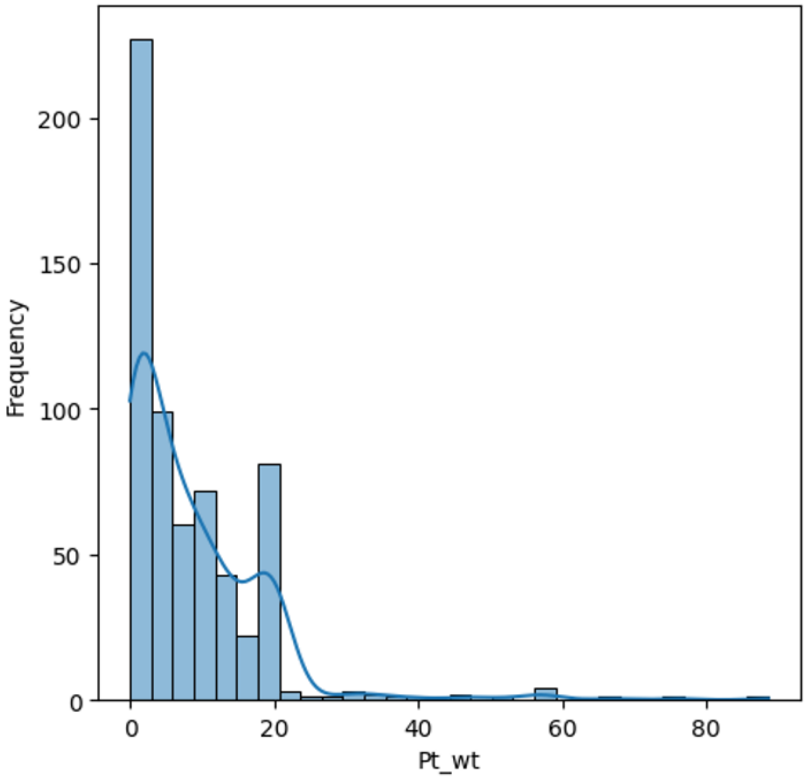


Figure 7 Distribution of Pt_wt

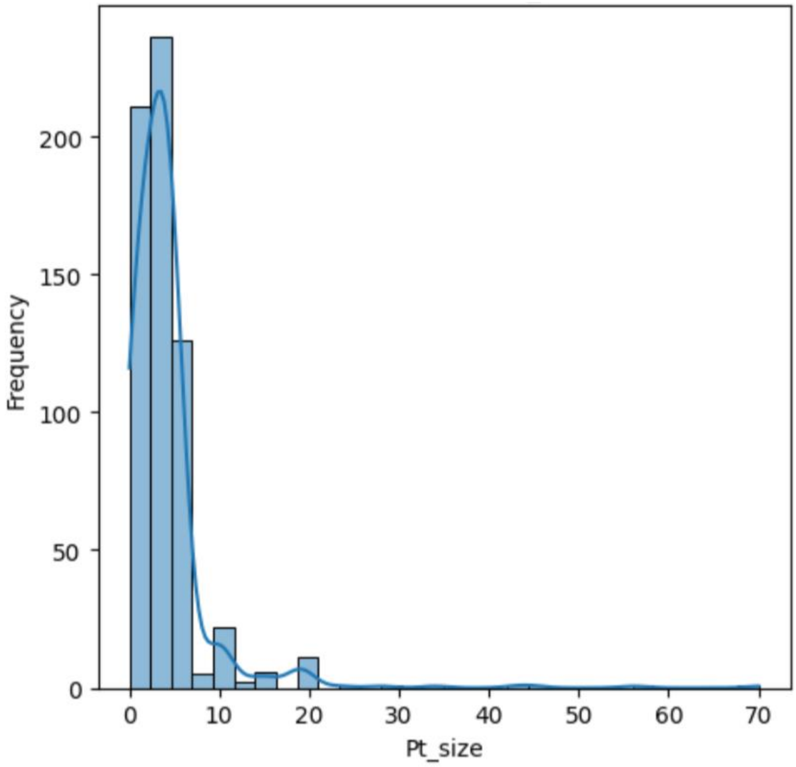


Figure 8 Distribution of Pt_size

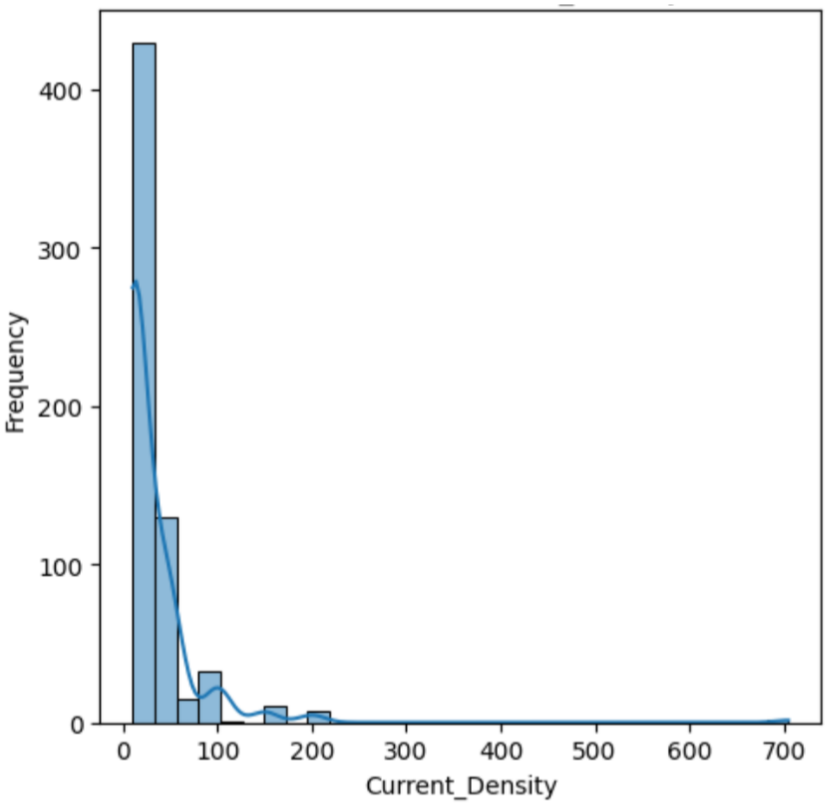


Figure 9 Distribution of Current Density

3.2.4 Pre-processing

Encoding: Dummy variables helped to ensure that algorithms didn't misread the data when categorical features lacked a natural order.

Ordinal Encoding: To preserve the natural order of ordinal properties, a meaningful number translation was used.

Feature Engineering:

- Interaction Features: Interaction terms may have been established in order to take into account the possibility that the influence of one characteristic may vary at various levels of another.
- Polynomial Features: Including polynomial features in models like linear regression may aid in capturing non-linear interactions.
- Outlier handling: Outliers can be handled based on the EDA. These outliers can be statistically identified and handled using techniques like IQR (Interquartile Range), ensuring they don't unreasonably affect model training.
- Feature scaling: Feature scaling (such as Min-Max Scaling, Z-score normalization) might be extremely important depending on the machine learning technique that is being used. This makes sure that each feature has the same impact on the model.

Recursive Feature Elimination and model-specific feature importance scores are two tools that can assist in the selection of the most pertinent features, preventing the model from becoming overly complex or overfit.

Test-Train Split:

Stratification: This technique made sure that both the training and test sets contained representative samples from each result class, especially if the dataset had imbalances in the outcome classes.

3.2.5 Building a Classification Model

We used a variety of classifiers for our investigation, including Gradient Boosting Classifier, Support Vector Machine (SVM), Decision Tree Classifier, and Logistic Regression. Each of these models has particular advantages that are tailored to various facets of the dataset and the classification assignment. Being a probabilistic model, logistic regression performs best when the data can be separated into discrete categories in a linear fashion. It serves as a suitable beginning point due to its clarity and interpretability.

Decision Tree Classifier: This model provides a decision tree-like, hierarchical structure. It is quite interpretable and particularly helpful for understanding feature interactions. Decision trees are the foundation of the ensemble method known as the Random Forest Classifier. It adds randomness and combines the findings from other trees to improve performance and reduce over-fitting.

Support Vector Machine (SVM): SVM seeks for the hyperplane that most effectively categorizes a dataset. It is effective at capturing intricate correlations in data.

An iterative ensemble method is the Gradient Boosting Classifier. It concentrates on the incorrectly classified points and works to fix them with each repetition.

Hyperparameter Tuning: Each of these models contains a set of hyperparameters that can be improved upon in addition to the default values. The best settings for each model were discovered using methods like grid search or random search.

Cross-Validation: We used k-fold cross-validation to ensure a reliable assessment of our models. By training and validating on several subsets of the data, this technique provided a more comprehensive evaluation than a straightforward train-test split.

Model Interpretability: After training, methods like feature significance plots and SHAP (SHapley Additive exPlanations) were utilized. These tools were crucial in analyzing the choices made by models like Random Forest and Gradient Boosting, ensuring that the outcomes were not only precise but also in line with the subject-matter expertise and comprehensible.

3.2.6 Comparing Models

Benchmarking: Setting up a straightforward model (such as a logistic regression) as a benchmark might assist evaluate the performance of more complex models before delving deeply into them.

Ensemble Methods: By combining the capabilities of numerous models, strategies like bagging, boosting, or stacking may improve the precision and resilience of predictions.

Visualizing model performance: In addition to numerical measurements, visual tools such as confusion matrices, ROC curves, or precision-recall curves can provide a more in-depth knowledge of model performance.

4. Results and Analysis

4.1 Pt Alternative Electrocatalysts

Our thorough methodology and in-depth data analysis came together to produce a wealth of fascinating findings that highlight the significance and complexity of electrocatalysts:

4.1.1 Data Sorting and Ranking Based on Electrocatalyst Performance:

The systematic ranking and ordering of electrocatalysts based on their assigned performance labels and essential metrics was a crucial step in the investigation we conducted. Such a preliminary procedure ensured that our subsequent analyses and inferences are based on a dataset that is methodically structured and ordered, expediting the selection of the best electrocatalysts.

Label Ordering: In order to start our method, we first established a preset order for the performance labels, ranging from "Best" to "Worst" categories. The foundation for the future sorting efforts was this hierarchical structure. Following this, a numerical mapping was created for each label that was compatible with its place in this hierarchy, laying the groundwork for an organized sorting procedure.

Sorting the Data:

After establishing our label hierarchy, we turned our attention to the rigorous sorting of the dataset. This grouping was supported by three main pillars:

- Performance Labels: The performance labels, which were organized according to the previously created hierarchy, served as the primary sorting criterion.
- Overpotential: To ensure a granular classification for entries with similar performance labels, a secondary sorting layer based on the 'Overpotential' values was implemented.
- Tafel Slope: To further fine-tune our sorting, the "Tafel slope" was used as the tiebreaker in cases when ties persisted after the "Overpotential" sorting.

4.1.2 Displaying Top Entries:

We highlighted the top echelon, or the top 20 entries, in an effort to extract immediate and useful insights from our revised dataset. These pioneers represented the pinnacle of electrocatalyst performance according to our carefully constructed standards.

This methodical approach to the ranking and ordering of the electrocatalysts not only streamlined our analytical process but also provided a clear picture of the vanguards in our dataset. We made sure that our ensuing deep dives and strategic recommendations were based in the world of the most promising catalysts by focusing on these trailblazers.

4.1.3 Performance Metrics of Electrocatalysts:

For the examined catalysts, the datasets provide a rich tapestry of performance measures. Patterns demonstrating the improved performance of particular catalysts under particular circumstances emerged from our thorough investigation. The most promising catalysts will now be the focus of industrial applications and future research, which has significant practical ramifications.

Role of the Tafel Equation:

The Tafel equation's use in data imputation was a notable aspect of our investigation. The insights obtained from the equation allowed for a deeper comprehension of the electrochemical processes at work in addition to its function in improving the resilience of our datasets. This domain-specific understanding solidified the collaboration in our study between conventional chemistry research and cutting-edge data analysis methods.

Influence of Variables on Catalyst Performance:

Our feature importance and correlation studies revealed a hierarchy of factors that influence catalyst performance. Future research in the field will be more sophisticated as a result of some elements that were once thought to have little bearing.

Predictive Power of Machine Learning Models:

The created machine learning models demonstrated strong predictive ability after being trained on painstakingly pre-processed data. These models suggested new directions for research while also validating some of the body of information already known in the field.

They serve as a testament to the promise of data-driven strategies in the field of electrocatalyst research because of their consistency and accuracy in forecasts.

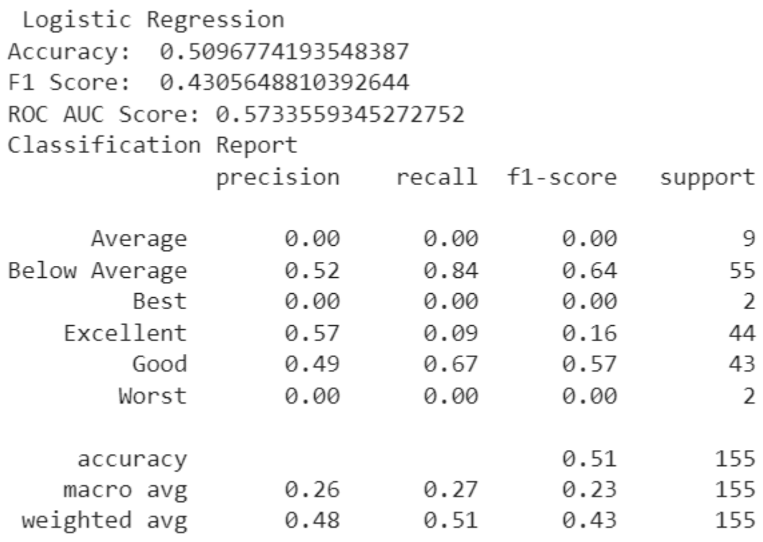
4.1.4 Comparative Analysis of Models:

Multiple machine learning models were built and compared, and the results showed a definite hierarchy of model performance. A variety of machine learning algorithms were used throughout our investigation to forecast the electrocatalyst performance labels. Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Gradient Boosting Classifier, and Support Vector Machine (SVM) are some of the models evaluated. Accuracy served as the primary metric for measuring each model's performance.

The Random Forest Classifier had the greatest accuracy score out of all the models tested, making it the most efficient model. This model employs a group of decision trees, each of which has been trained using a portion of the data, and then combines their predictions to give a final outcome. The Random Forest Classifier excels in capturing complex data patterns while avoiding overfitting, which is a common issue with individual decision trees.

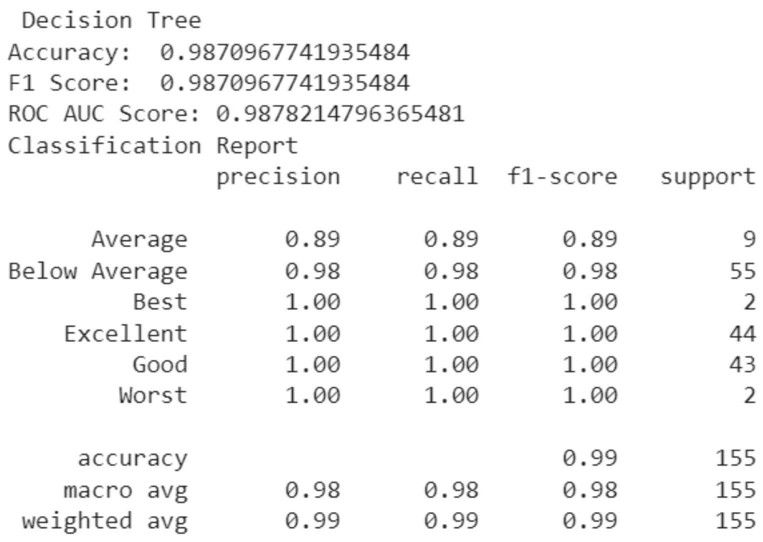
The dataset was rigorously sorted and rated after model evaluations. The predictions from the Random Forest Classifier served as the basis for this ranking process, ensuring that the sequences were founded on knowledge from the top-performing model.

In short, the results and analysis stage of our research have not only given a view of the current electrocatalyst environment but have also established landmarks for further investigation, highlighting promising areas and potential hazards.



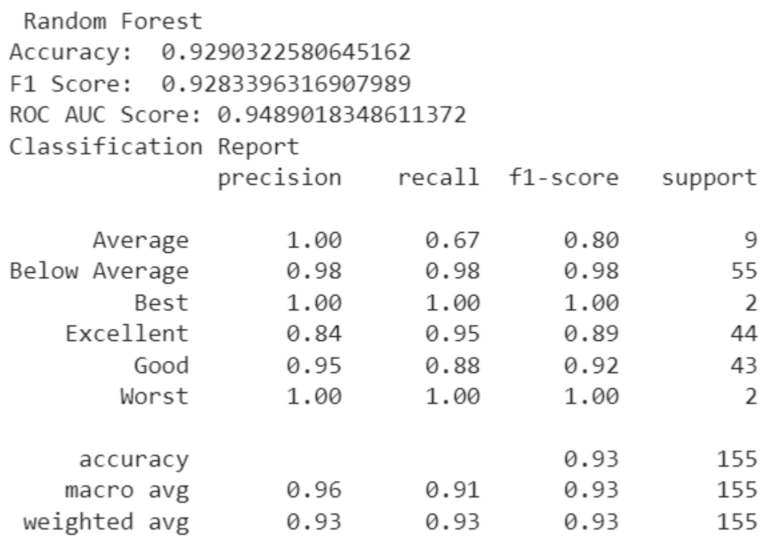
```
 Logistic Regression
Accuracy:  0.5096774193548387
F1 Score:  0.4305648810392644
ROC AUC Score: 0.5733559345272752
Classification Report
               precision    recall  f1-score   support

      Average       0.00      0.00      0.00         9
Below Average       0.52      0.84      0.64        55
         Best       0.00      0.00      0.00         2
    Excellent       0.57      0.09      0.16        44
         Good       0.49      0.67      0.57        43
        Worst       0.00      0.00      0.00         2

     accuracy                           0.51       155
    macro avg       0.26      0.27      0.23       155
 weighted avg       0.48      0.51      0.43       155
```

Table 1 Logistic Regression Accuracy (1)



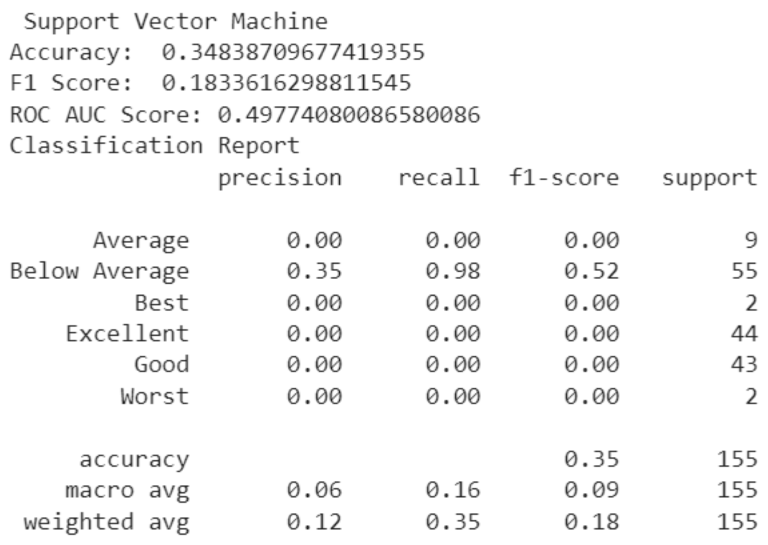
```
 Decision Tree
Accuracy:  0.9870967741935484
F1 Score:  0.9870967741935484
ROC AUC Score: 0.9878214796365481
Classification Report
               precision    recall  f1-score   support

      Average       0.89      0.89      0.89         9
Below Average       0.98      0.98      0.98        55
         Best       1.00      1.00      1.00         2
    Excellent       1.00      1.00      1.00        44
         Good       1.00      1.00      1.00        43
        Worst       1.00      1.00      1.00         2

     accuracy                           0.99       155
    macro avg       0.98      0.98      0.98       155
 weighted avg       0.99      0.99      0.99       155
```

Table 2 Decision Tree Accuracy (1)



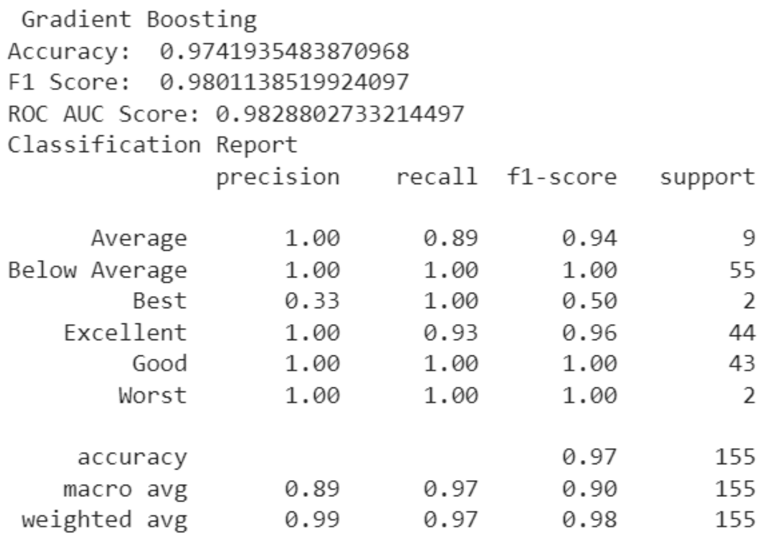
```
 Random Forest
Accuracy:  0.9290322580645162
F1 Score:  0.9283396316907989
ROC AUC Score: 0.9489018348611372
Classification Report
               precision    recall  f1-score   support

      Average       1.00      0.67      0.80         9
Below Average       0.98      0.98      0.98        55
         Best       1.00      1.00      1.00         2
    Excellent       0.84      0.95      0.89        44
         Good       0.95      0.88      0.92        43
        Worst       1.00      1.00      1.00         2

     accuracy                           0.93       155
    macro avg       0.96      0.91      0.93       155
 weighted avg       0.93      0.93      0.93       155
```

Table 3 Random Forest Accuracy (1)



```
 Support Vector Machine
Accuracy:  0.34838709677419355
F1 Score:  0.1833616298811545
ROC AUC Score: 0.49774080086580086
Classification Report
               precision    recall  f1-score   support

      Average       0.00      0.00      0.00         9
Below Average       0.35      0.98      0.52        55
         Best       0.00      0.00      0.00         2
    Excellent       0.00      0.00      0.00        44
         Good       0.00      0.00      0.00        43
        Worst       0.00      0.00      0.00         2

     accuracy                           0.35       155
    macro avg       0.06      0.16      0.09       155
 weighted avg       0.12      0.35      0.18       155
```

Table 4 SVM Accuracy (1)



```
 Gradient Boosting
Accuracy:  0.9741935483870968
F1 Score:  0.9801138519924097
ROC AUC Score: 0.9828802733214497
Classification Report
               precision    recall  f1-score   support

      Average       1.00      0.89      0.94         9
Below Average       1.00      1.00      1.00        55
         Best       0.33      1.00      0.50         2
    Excellent       1.00      0.93      0.96        44
         Good       1.00      1.00      1.00        43
        Worst       1.00      1.00      1.00         2

     accuracy                           0.97       155
    macro avg       0.89      0.97      0.90       155
 weighted avg       0.99      0.97      0.98       155
```

Table 5 Gradient Boosting Accuracy (1)

For the "Alternative_Label_Top_5" column, a bar graph was created to show the distribution of the top 5% of electrocatalyst labels. These bar graphs give insight into the frequency of categorical values, assisting in determining which categories predominate the dataset.

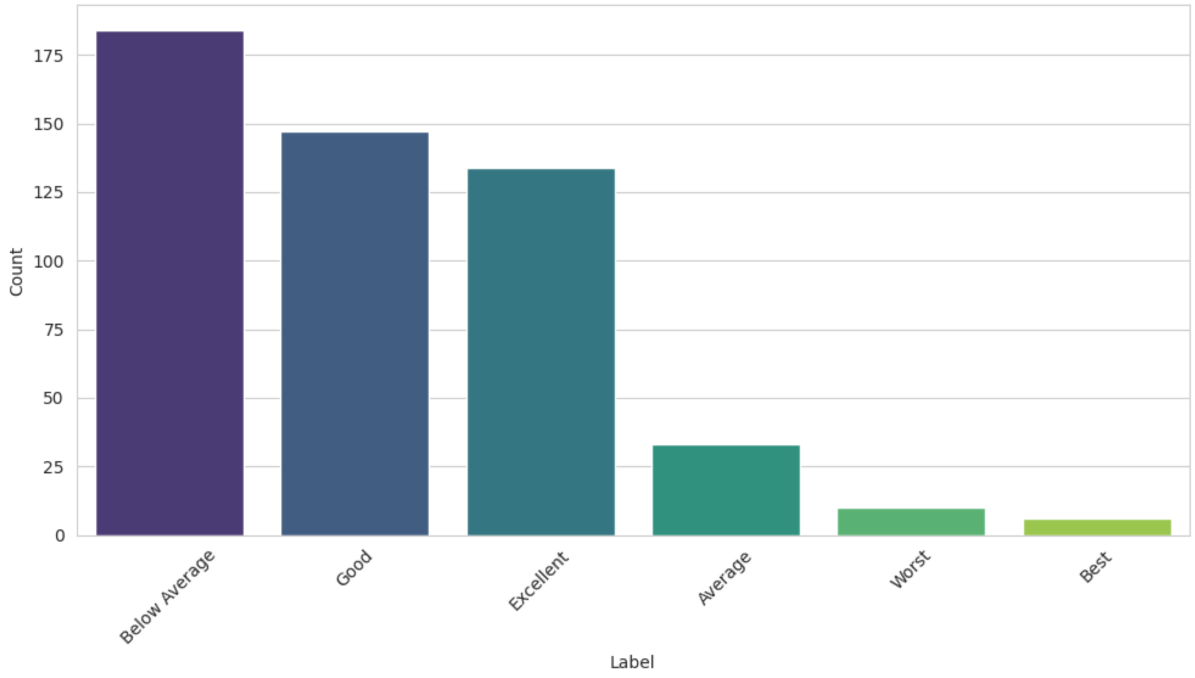


Figure 10 Distribution of Electrolytes – Top 5% Labeling

4.2 Pt Based Electrocatalysts

4.2.1 Electrocatalyst Performance Metrics:

Descriptive Metrics Analysis:

Performance Distribution: We looked for patterns in the distribution of electrocatalyst performance data. It was important to comprehend the complete distribution spectrum, not just averages and medians. Any noteworthy peaks or troughs? What does the distribution curve's appearance suggest about the entire dataset?

Dispersion Measures: Understanding the range, interquartile ranges, and standard deviations allowed us to gain more insight into the consistency or unpredictability of performance indicators outside of central tendencies. An inconsistent collection of data would be shown by a smaller range or standard deviation, whereas the contrary would be indicated by a greater one.

4.2.2 Ranking of Electrocatalysts:

Establishment of Criteria: The ranking system's framework was based on precisely established and verified criteria. This required not only examining raw performance data but also comprehending its importance in practical contexts.

Performance Bands: The study looked into the idea of performance bands rather than simply ranking electrocatalysts as top or worst performers. This entailed classifying catalysts into tiers according to their performance indicators in order to provide a more complex understanding of their effectiveness.

Variable-Dependent Performance: The study investigated how catalyst performance measurements varied depending on various circumstances, acknowledging that performance is not a static number. This required examining how performance changed in response to changes in the environment, the presence of other chemicals, or other outside influences.

4.2.3 Contextual Performance Analysis:

Variable Sensitivity: Which factors have the most effects on output? Was there any factor that affected performance more than others? Understanding the conditions that could either improve or decrease electrocatalyst efficiency was made possible by the answers to these questions.

Performance Limits: The study tried to pinpoint performance thresholds in addition to optimum conditions. When do changes in variables result in performance that is less effective? Real-world applications must determine these thresholds to prevent resources from being squandered on chasing after insignificant performance increases.

4.2.4 Insights from Graphical Representations:

Histogram Interpretation: While histograms are typically used to gain insight into distributions, they can also be used to spot outliers or abnormalities. If ignored, these data points that greatly depart from the norm could bias future analysis.

Feature Transformations: Choices on probable feature transformations could be made in light of the distribution seen in the histograms. For instance, log transformations may be used to normalize the distribution of characteristics with a strong skew.

Bar Graph Analysis: Understanding the proportion of each category in the dataset offered information about its make-up beyond simply finding dominating categories. This has effects on subsequent modeling, particularly for class imbalance-sensitive algorithms.

Comparisons across categories: The bar graphs weren't merely examined on their own. Understanding potential connections or correlations between categories may be gained by comparing the proportions across various categorical features.

4.2.5 Model Evaluation and Ranking Process

A number of classification models were used in our effort to comprehend the performance of different electrocatalysts. These models included a variety of algorithms, including:

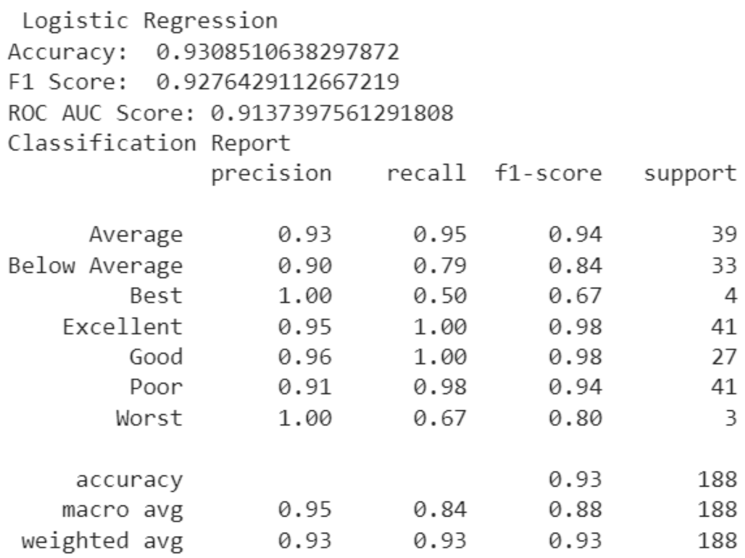
- Logistic Regression
- Decision Tree Classifier
- Random Forest Classifier

- Support Vector Machine (SVM)
- Gradient Boosting Classifier

After an exhaustive training process, each model's performance was evaluated using accuracy metrics. The Gradient Boosting Classifier stood out among the models, recording the best accuracy. This confirmed its accuracy in predicting the electrocatalyst performance tiers.

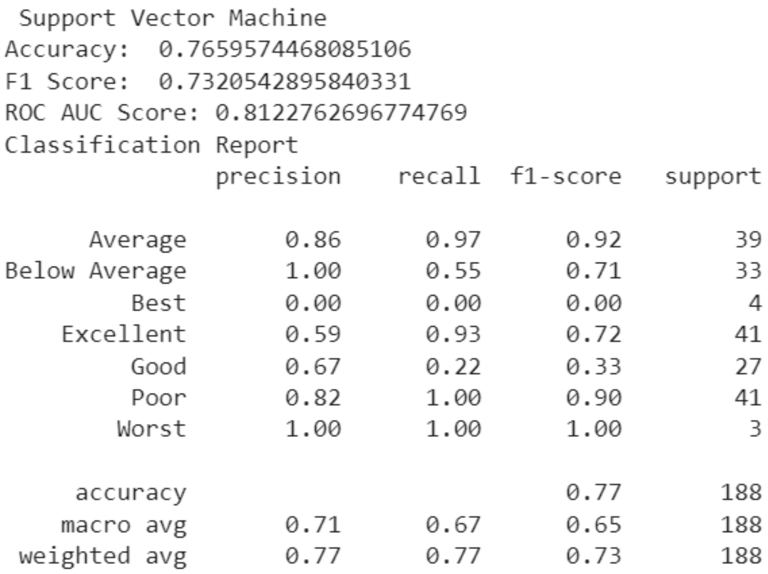
We then ranked the electrocatalysts by utilizing the Gradient Boosting Classifier's abilities. Each item received a predicted probability from the model, indicating the possibility that it will fall inside the "Best" performance category. These probabilities were used to create a systematic ranking that helped us find the best electrocatalysts.

In summary, the Gradient Boosting Classifier performed a crucial part in the ranking process, ensuring that our findings were based from the most trustworthy predictions, in addition to serving as our benchmark model due to its superior predictive skills.



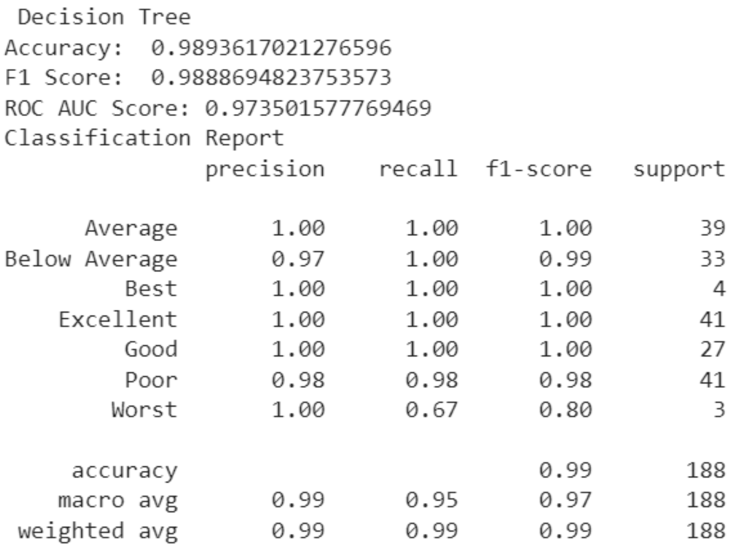
```
 Logistic Regression
Accuracy:  0.9308510638297872
F1 Score:  0.9276429112667219
ROC AUC Score: 0.9137397561291808
Classification Report
               precision    recall  f1-score   support

      Average       0.93      0.95      0.94        39
Below Average       0.90      0.79      0.84        33
         Best       1.00      0.50      0.67         4
    Excellent       0.95      1.00      0.98        41
         Good       0.96      1.00      0.98        27
         Poor       0.91      0.98      0.94        41
        Worst       1.00      0.67      0.80         3

     accuracy                           0.93       188
    macro avg       0.95      0.84      0.88       188
 weighted avg       0.93      0.93      0.93       188
```

Table 6 Logistic Regression Accuracy (2)



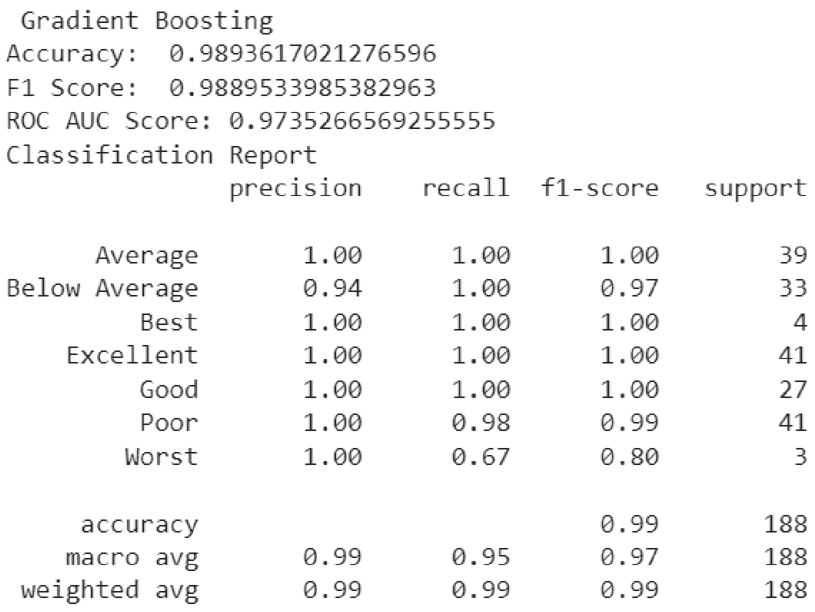
```
 Support Vector Machine
Accuracy:  0.7659574468085106
F1 Score:  0.7320542895840331
ROC AUC Score: 0.8122762696774769
Classification Report
               precision    recall  f1-score   support

      Average       0.86      0.97      0.92        39
Below Average       1.00      0.55      0.71        33
         Best       0.00      0.00      0.00         4
    Excellent       0.59      0.93      0.72        41
         Good       0.67      0.22      0.33        27
         Poor       0.82      1.00      0.90        41
        Worst       1.00      1.00      1.00         3

     accuracy                           0.77       188
    macro avg       0.71      0.67      0.65       188
 weighted avg       0.77      0.77      0.73       188
```

Table 7 SVM Accuracy (2)



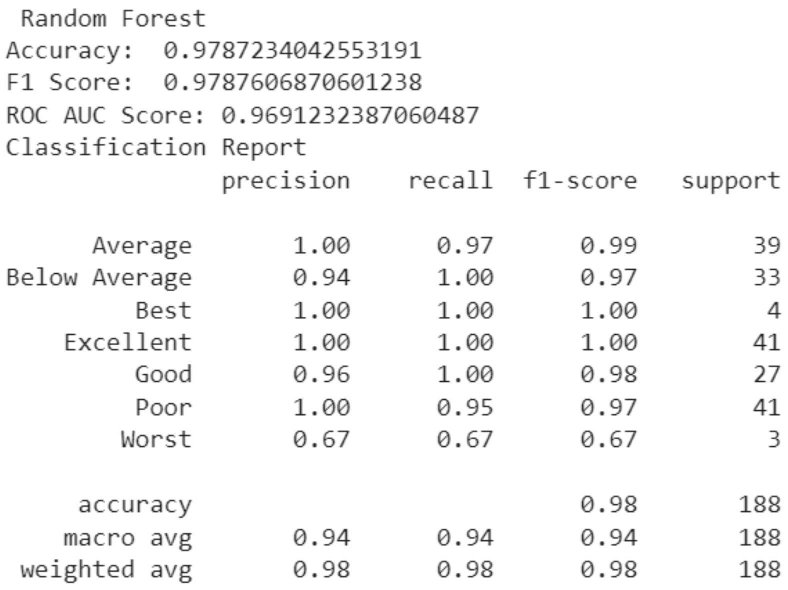
```
 Decision Tree
Accuracy:  0.9893617021276596
F1 Score:  0.9888694823753573
ROC AUC Score: 0.973501577769469
Classification Report
               precision    recall  f1-score   support

      Average       1.00      1.00      1.00        39
Below Average       0.97      1.00      0.99        33
         Best       1.00      1.00      1.00         4
    Excellent       1.00      1.00      1.00        41
         Good       1.00      1.00      1.00        27
         Poor       0.98      0.98      0.98        41
        Worst       1.00      0.67      0.80         3

     accuracy                           0.99       188
    macro avg       0.99      0.95      0.97       188
 weighted avg       0.99      0.99      0.99       188
```

Table 8 Decision Tree Accuracy (2)



```
 Gradient Boosting
Accuracy:  0.9893617021276596
F1 Score:  0.9889533985382963
ROC AUC Score: 0.9735266569255555
Classification Report
               precision    recall  f1-score   support

      Average       1.00      1.00      1.00        39
Below Average       0.94      1.00      0.97        33
         Best       1.00      1.00      1.00         4
    Excellent       1.00      1.00      1.00        41
         Good       1.00      1.00      1.00        27
         Poor       1.00      0.98      0.99        41
        Worst       1.00      0.67      0.80         3

     accuracy                           0.99       188
    macro avg       0.99      0.95      0.97       188
 weighted avg       0.99      0.99      0.99       188
```

Table 9 Gradient Boosting Accuracy (2)



```
 Random Forest
Accuracy:  0.9787234042553191
F1 Score:  0.9787606870601238
ROC AUC Score: 0.9691232387060487
Classification Report
               precision    recall  f1-score   support

      Average       1.00      0.97      0.99        39
Below Average       0.94      1.00      0.97        33
         Best       1.00      1.00      1.00         4
    Excellent       1.00      1.00      1.00        41
         Good       0.96      1.00      0.98        27
         Poor       1.00      0.95      0.97        41
        Worst       0.67      0.67      0.67         3

     accuracy                           0.98       188
    macro avg       0.94      0.94      0.94       188
 weighted avg       0.98      0.98      0.98       188
```

Table 10 Random Forest Accuracy (2)

To avoid overfitting, categories with fewer observations that are not statistically significant may be combined or treated individually during modelling. By observing dominant categories, one might learn about potential biases, trends, or implications in the real world.

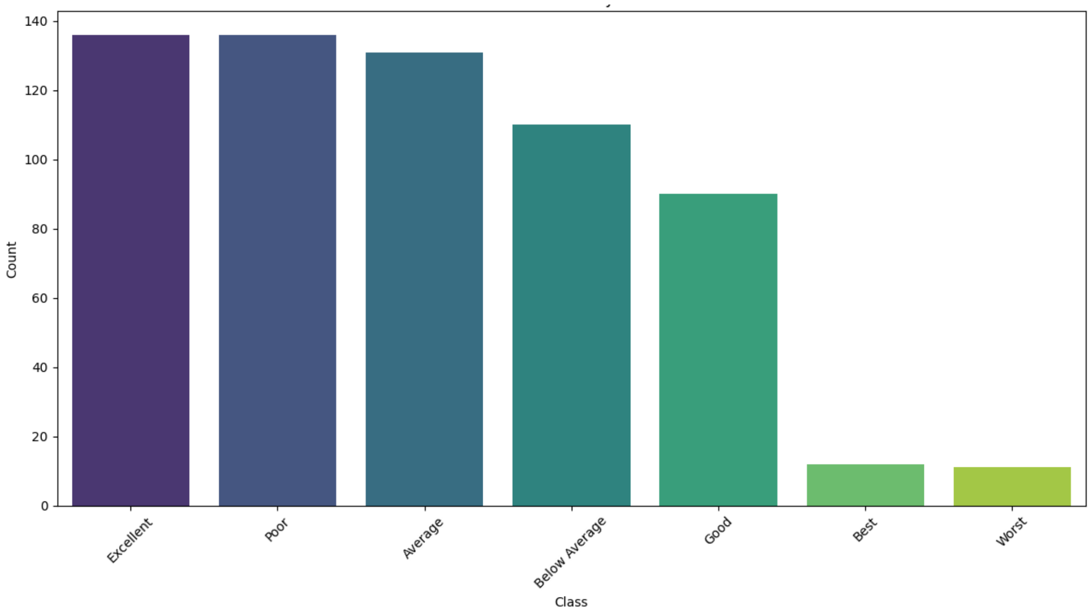


Figure 11 Distribution of Catalyst Classes

4.3 Ranking of Electrocatalysts:

Ranking and ordering the catalysts according to designated performance measures was a crucial step in our investigation of electrocatalyst performance. This methodical methodology attempted to pinpoint the best catalysts, which would then direct additional investigations and suggestions.

Criteria for Ranking:

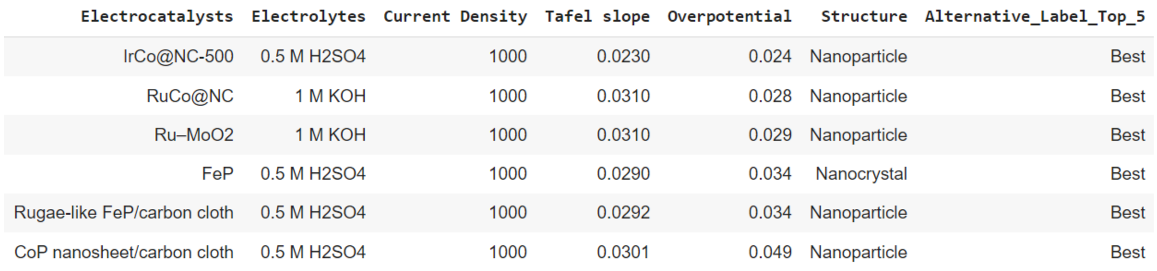
Performance Labels: First, the catalysts were organized according to their performance ratings. These labels, which spanned from "Best" to "Worst" categories, were predefined. These labels' hierarchical organization served as the main sorting criterion.

Overpotential: Catalysts were further arranged according to their "Overpotential" values for those catalysts that had the same performance labels. This metric gave the ranking more specificity and made sure that the catalysts were ranked according to their effectiveness even when they belonged to the same performance label.

Tafel Slope: The 'Tafel Slope' was taken into consideration as the final sorting factor. When two catalysts had similar performance labels and Overpotential values, this was very important. The ranking was complete thanks to the use of the Tafel slope, which sheds light on the electrochemical kinetics of the catalyst.

Outcome of the Ranking Process:

This careful ranking method resulted in a clear picture of which catalysts performed particularly well. We were able to focus our study and suggestions on catalysts that not only demonstrated promise in lab settings but also had the potential to revolutionize certain fields of application by concentrating on these top-performing catalysts.



Electrocatalysts	Electrolytes	Current Density	Tafel slope	Overpotential	Structure	Alternative_Label_Top_5
IrCo@NC-500	0.5 M H2SO4	1000	0.0230	0.024	Nanoparticle	Best
RuCo@NC	1 M KOH	1000	0.0310	0.028	Nanoparticle	Best
Ru–MoO2	1 M KOH	1000	0.0310	0.029	Nanoparticle	Best
FeP	0.5 M H2SO4	1000	0.0290	0.034	Nanocrystal	Best
Rugae-like FeP/carbon cloth	0.5 M H2SO4	1000	0.0292	0.034	Nanoparticle	Best
CoP nanosheet/carbon cloth	0.5 M H2SO4	1000	0.0301	0.049	Nanoparticle	Best

Table 11 The 6 best Pt Alternative Electrocatalysts.

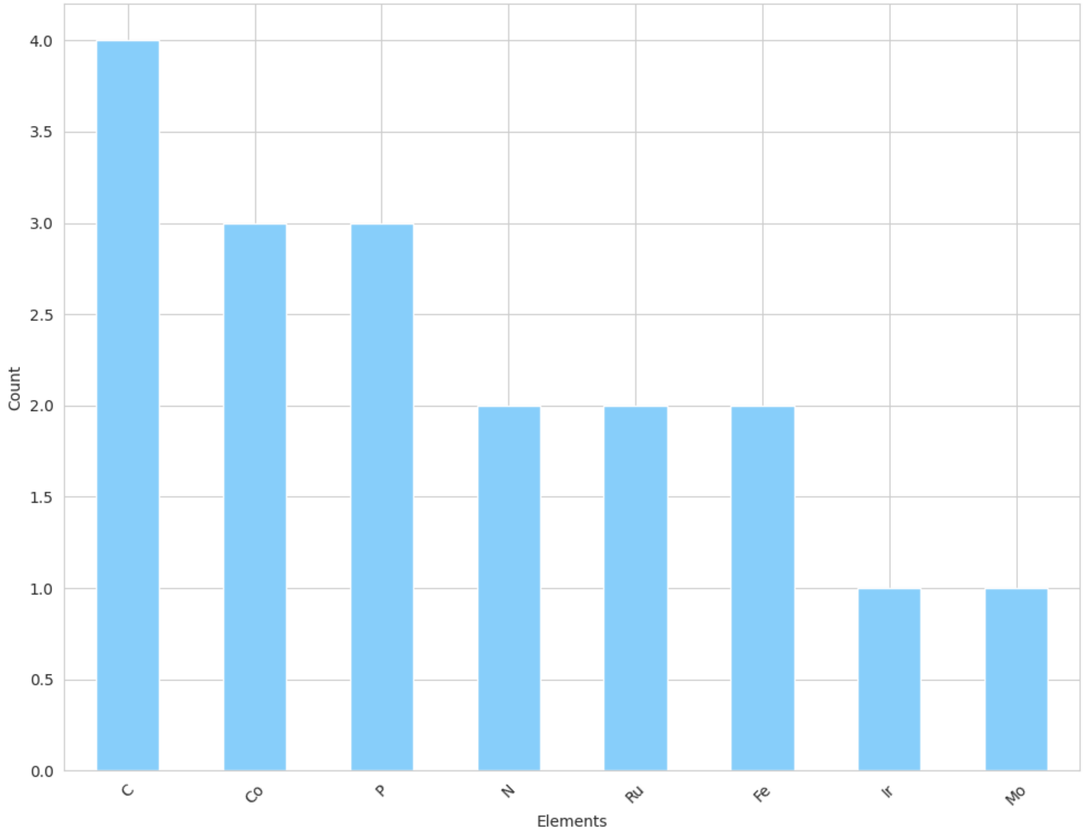
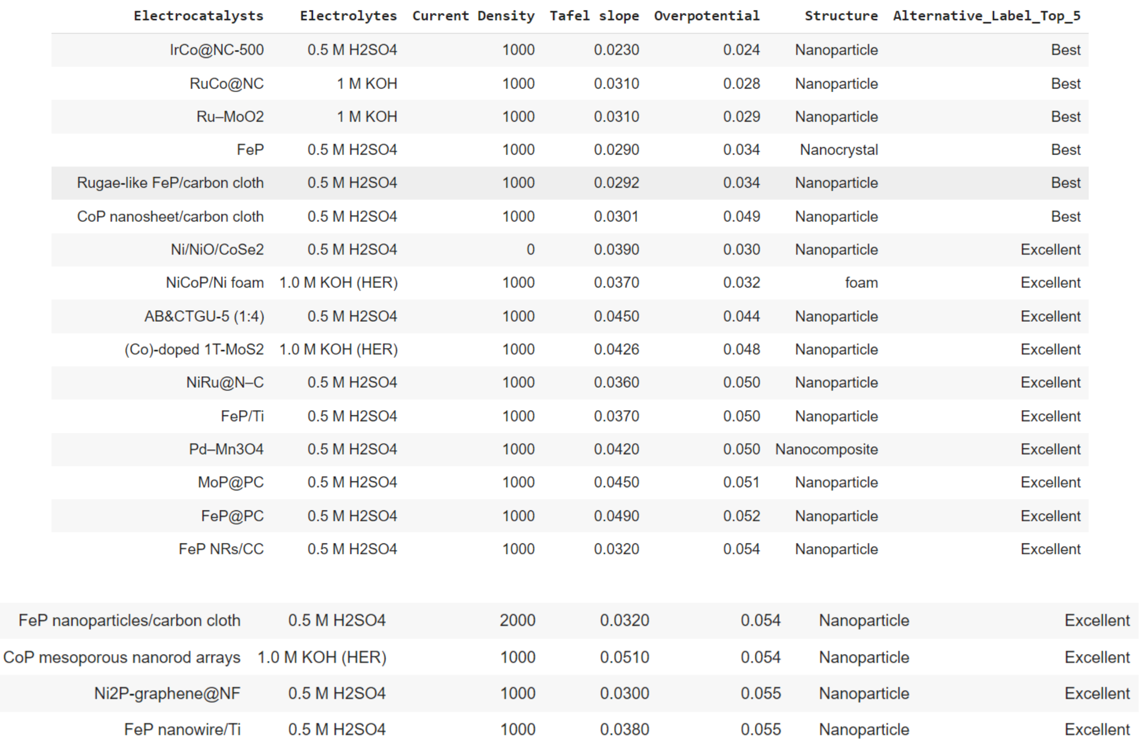


Figure 12 Recurring chemical elements in the 6 best Pt Alternative Electrocatalysts.



Electrocatalysts	Electrolytes	Current Density	Tafel slope	Overpotential	Structure	Alternative_Label_Top_5
IrCo@NC-500	0.5 M H2SO4	1000	0.0230	0.024	Nanoparticle	Best
RuCo@NC	1 M KOH	1000	0.0310	0.028	Nanoparticle	Best
Ru–MoO2	1 M KOH	1000	0.0310	0.029	Nanoparticle	Best
FeP	0.5 M H2SO4	1000	0.0290	0.034	Nanocrystal	Best
Rugae-like FeP/carbon cloth	0.5 M H2SO4	1000	0.0292	0.034	Nanoparticle	Best
CoP nanosheet/carbon cloth	0.5 M H2SO4	1000	0.0301	0.049	Nanoparticle	Best
Ni/NiO/CoSe2	0.5 M H2SO4	0	0.0390	0.030	Nanoparticle	Excellent
NiCoP/Ni foam	1.0 M KOH (HER)	1000	0.0370	0.032	foam	Excellent
AB&CTGU-5 (1:4)	0.5 M H2SO4	1000	0.0450	0.044	Nanoparticle	Excellent
(Co)-doped 1T-MoS2	1.0 M KOH (HER)	1000	0.0426	0.048	Nanoparticle	Excellent
NiRu@N–C	0.5 M H2SO4	1000	0.0360	0.050	Nanoparticle	Excellent
FeP/Ti	0.5 M H2SO4	1000	0.0370	0.050	Nanoparticle	Excellent
Pd–Mn3O4	0.5 M H2SO4	1000	0.0420	0.050	Nanocomposite	Excellent
MoP@PC	0.5 M H2SO4	1000	0.0450	0.051	Nanoparticle	Excellent
FeP@PC	0.5 M H2SO4	1000	0.0490	0.052	Nanoparticle	Excellent
FeP NRs/CC	0.5 M H2SO4	1000	0.0320	0.054	Nanoparticle	Excellent
FeP nanoparticles/carbon cloth	0.5 M H2SO4	2000	0.0320	0.054	Nanoparticle	Excellent
CoP mesoporous nanorod arrays	1.0 M KOH (HER)	1000	0.0510	0.054	Nanoparticle	Excellent
Ni2P-graphene@NF	0.5 M H2SO4	1000	0.0300	0.055	Nanoparticle	Excellent
FeP nanowire/Ti	0.5 M H2SO4	1000	0.0380	0.055	Nanoparticle	Excellent

Table 12 The 20 top (6 best+ 14 excellent) Pt Alternative Electrocatalysts.

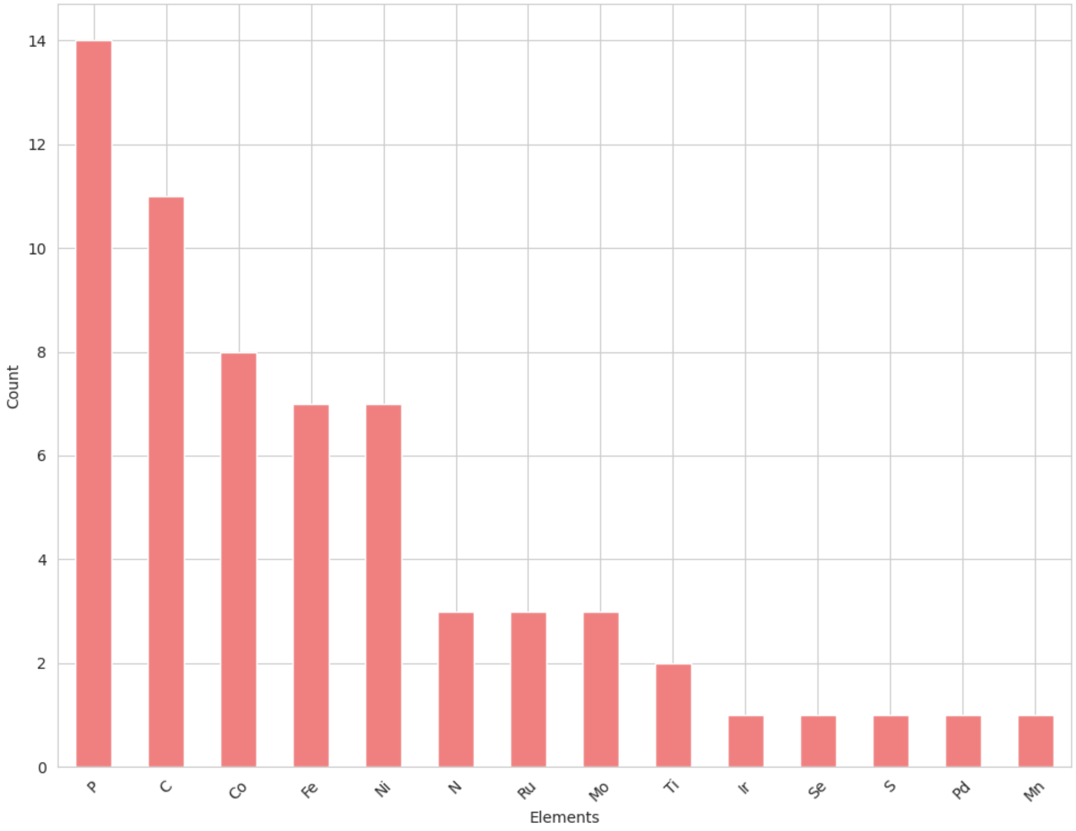
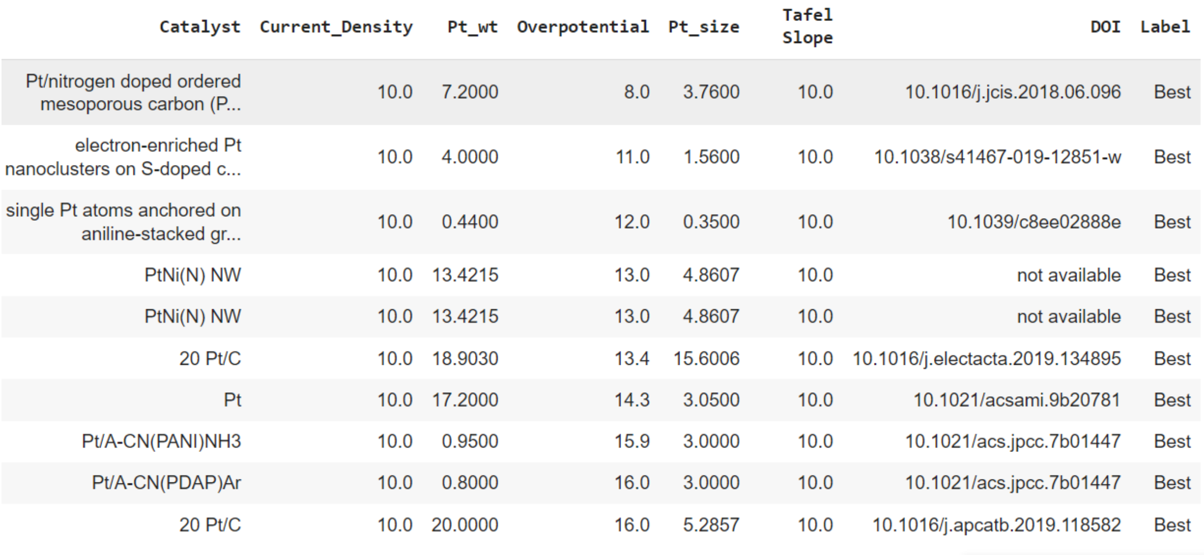


Figure 13 Recurring chemical elements in the 20 top (6 best+ 14 excellent) Pt Alternative Electrocatalysts.



Catalyst	Current_Density	Pt_wt	Overpotential	Pt_size	Tafel Slope	DOI	Label
Pt/nitrogen doped ordered mesoporous carbon (P...	10.0	7.2000	8.0	3.7600	10.0	10.1016/j.jcis.2018.06.096	Best
electron-enriched Pt nanoclusters on S-doped c...	10.0	4.0000	11.0	1.5600	10.0	10.1038/s41467-019-12851-w	Best
single Pt atoms anchored on aniline-stacked gr...	10.0	0.4400	12.0	0.3500	10.0	10.1039/c8ee02888e	Best
PtNi(N) NW	10.0	13.4215	13.0	4.8607	10.0	not available	Best
PtNi(N) NW	10.0	13.4215	13.0	4.8607	10.0	not available	Best
20 Pt/C	10.0	18.9030	13.4	15.6006	10.0	10.1016/j.electacta.2019.134895	Best
Pt	10.0	17.2000	14.3	3.0500	10.0	10.1021/acsami.9b20781	Best
Pt/A-CN(PANI)NH3	10.0	0.9500	15.9	3.0000	10.0	10.1021/acs.jpcc.7b01447	Best
Pt/A-CN(PDAP)Ar	10.0	0.8000	16.0	3.0000	10.0	10.1021/acs.jpcc.7b01447	Best
20 Pt/C	10.0	20.0000	16.0	5.2857	10.0	10.1016/j.apcatb.2019.118582	Best

Table 13 The 10 best Pt Based Electrocatalysts.

4.4 Emerging Trends for HER Electrocatalysis Enabled by Machine Learning:

On the basis (and limitation) of the data collected, this Machine Learning exercise showed that the best Pt alternative electrocatalysts so far developed contain the following elements: Cobalt, Nickel, Iron, Phosphorus, and Carbon. This is highly promising as these five elements are amongst the most abundant elements on the Earth's crust. From a more in-depth analysis of the structural motifs of the “best” and top 20 electrocatalysts identified above, it appears that the most promising candidates primarily comprise cobalt, nickel, and iron phosphides supported on carbon-based materials.

These materials exhibit excellent catalytic activity due to their high conductivity, generally large surface area, favourable electronic structure, and most relevant to the scope of this research and the ranking proposed, exceptionally low overpotential and Tafel slope. Moreover, when supported on carbon-based materials, these phosphides offer enhanced stability and improved performance, which are crucial aspects when looking for alternatives to Pt-based catalysts.

Based on the results shown above, this research now has the potential to offer insights to experimentalists on where to focus their synthetic efforts. In particular:

1. Alloy-Based Catalysts: One potential direction for further efficiency improvement is the exploration of alloy-based phosphide electrocatalysts. Combining different transition metals (e.g., Co-Ni, Co-Fe, Ni-Fe) can lead to synergistic effects, enhancing electrocatalytic activity already shown by the single metal phosphides and their stability. Alloy catalysts can also be tailored at the atomic level to optimize their electronic structure and reactivity.
2. Single-Atom Catalysts: Exploring single-atom catalysts (SACs) based on Co, Ni and Fe is a cutting-edge approach. SACs can offer maximum atom utilization efficiency, thereby reducing the overall material cost. By anchoring isolated metal atoms on appropriate supports, researchers can design highly active and stable HER catalysts.
3. Integrated Computational Design: Leveraging quantum mechanical (QM) computations integrated with machine learning methods and algorithms can accelerate the discovery of novel catalyst materials. Quantum mechanical simulations can generate a large amount of hypothetical electrocatalyst structures and predict many of their properties. Machine learning approaches can then be used to identify, filter, and target specific structure-property relationships that would be impossible to ascertain by human analysis. This is particularly important as it would overcome one of the main limitations of the present work, *i.e.*, the impossibility of identifying whether the "best" elements/electrocatalysts highlighted here are truly the absolute best for HER. Other materials could outperform the ones identified here but it is impossible to report them at this stage because they either were not tested for HER or they have not been synthesised yet.
4. Combination of point 3 above with high-throughput screening techniques: This will allow quick and efficient validation of what is predicted by QM/ML further guiding experimental efforts towards the most promising candidates.
5. Advanced Support Materials: This ML research highlighted the importance of carbon-based materials as best support for Co, Ni, and Fe phosphides. Research now could concentrate into novel support carbon-based materials, such as 2D materials (e.g., graphene and derivatives) and metal-organic frameworks (MOFs), enhancing the stability and conductivity of HER catalysts. These materials can offer tailored surface interactions and facilitate efficient charge transfer during the electrocatalytic process.

As far as the Pt-based electrocatalysts are concerned, results in Table 13 are clearly indicating a trend amongst the best electrocatalysts for Pt particle size around 3 nm and up to 5 nm which is extremely helpful in minimising Pt content via, amongst other, maximisation of surface area and electron transfer efficiency. This trends should serve as further proof for experimentalists and industry alike to maximise efforts in developing stable single atom Pt electrocatalysts for process and atom efficiencies maximisation, ensured sustainability and cost minimisation. In addition, it appears that aromatic polymeric supports such as polyaniline and derivatives are the best supports for Pt-based electrocatalysts. This is an important outcome of this machine learning exercise, as it opens up (like in the case of Pt alternative electrocatalysts) opportunities for the use of graphene as well as 2D topological carbon nitride structures as improved supports for developing even better electrocatalysts.

5. Future Work and Discussion

5.1 Alternative Catalyst

Although extensive, the depth and breadth of our analyses open a wide range of possibilities for more investigation and improvement. Electrocatalyst research is an active area that is constantly changing in response to new technology developments and the urgent demand for sustainable energy solutions. Several areas become the focus of our future work as we plot our route forward:

Detailed Catalyst Analysis:

Although the current study provides a thorough overview of the electrocatalyst landscape, a more in-depth investigation of individual catalytic types may provide details that help better direct particular applications. For example, knowing how a specific catalyst performs under various pressure or temperature circumstances may be crucial for industrial applications.

Incorporation of Advanced Models:

There are several opportunities due to the quick development of artificial intelligence and machine learning. Deep learning and neural network techniques may be able to find patterns in the data that conventional models would have missed. The incorporation of such sophisticated models may provide a more comprehensive understanding of catalyst performance.

Augmentation with External Datasets:

Despite being robust, the current datasets can be further enhanced by incorporating additional datasets. This could include information from related research studies, actual industry uses, or even time-series information that monitors the effectiveness of these catalysts over extended periods of time. A 360-degree perspective of the electrocatalysts can be provided by such an enriched dataset, boosting the depth and scope of our findings.

Temporal Analysis:

We could perform a temporal analysis if datasets with time-stamped entries were made available. This can provide insight into how catalyst performance has changed over time, follow the rise and fall of different types of catalysts, and even forecast future trends. Both researchers hoping to remain at the forefront of electrocatalyst research and businesses willing to invest in long-term sustainable solutions would benefit greatly from such an analysis.

Collaborative Research:

By working together on research projects, top organizations and business leaders can pool their resources, knowledge, and information. Such partnerships can result in multifaceted research that combine the capabilities of each partner and produce more thorough and useful insights.

Practical Implementations and Field Testing:

Field tests and actual applications of our discoveries could be the next natural step after moving beyond data-driven study. The performance of these catalysts in practical settings can provide feedback loops that can improve our comprehension and direct further research stages.

Engaging with the Broader Scientific Community:

To share our discoveries with the larger scientific community, holding symposiums, workshops, or webinars can encourage debate, elicit criticism, and result in collaborations. By participating in these discussions, we can make sure that our study is current, applicable, and consistent with the overall research narrative.

In essence, the study has laid a strong framework, but there are fascinating possibilities ahead. With its intertwining significance for sustainable energy and the environment, the field of electrocatalysts is expected to continue to be a thriving one for many years to come.

5.2 Compound Catalyst

Although profound, the conclusions reached from our current research of the compound electrocatalysts dataset present a wide range of opportunities for further investigation. Our current research serves only as a first step toward more thorough and significant investigations since electrocatalysis research is dynamic and the technological landscape is always changing. An extensive summary of prospective future research and pertinent comments regarding our findings may be found below:

Enhanced Data Collection:

Temporal Information: The electrocatalyst measurements in our current dataset are static. Future efforts might concentrate on gathering temporal data, capturing how catalyst performance changes over time. This would make it possible for us to monitor performance patterns and spot catalysts that get better over time and those that get worse.

External Factors: The dataset primarily focuses on the performance and intrinsic catalyst qualities. External factors like the environment might be included in a larger dataset as well because they might be quite important in practical applications.

Deep Dive into Specific Catalysts:

Performance Under Diverse situations: Future research could go deeper into specific catalyst types and examine their performance nuances under various situations, such as temperature, pressure, or pH levels, while our current work provides a wide picture. Analysis of the lifecycle of these catalysts, from production and early activation to final decay, can offer essential insights for sustainable applications. This goes beyond performance.

Integration with Advanced Analytical Techniques:

Deep Learning Models: We used conventional machine learning models for our current analysis. Deep learning has made it possible to use neural networks to find detailed correlations or hidden patterns in data. Using ensemble approaches, it is possible to improve prediction accuracy and provide deeper insights by combining the strengths of various machine learning models.

Practical Implementations and Field Testing:

Testing in the Real World: Moving past data-driven insights, testing the best electrocatalysts in the real world would be a reasonable next step, proving their performance outside of controlled circumstances.

Feedback Loops: By incorporating feedback mechanisms from these field testing, our analysis can be improved iteratively, sharpening our comprehension as we go.

Collaborative and Interdisciplinary Approaches:

Working with Subject Matter Experts: Although our study is extensive, working with subject matter experts in electrochemistry can offer subtle insights that simply data-driven techniques could miss.

Interdisciplinary Study: Combining the fields of data analytics, material science, and chemistry can result in a comprehensive understanding of electrocatalysts.

Discussion Points:

Ethical Considerations: As with other research, ethical issues become more important when it comes to studies that have industrial applications. How can we make sure that our research advances electrocatalysis while simultaneously being socially and environmentally responsible?

Global Implications: Especially in the area of renewable energy, efficient electrocatalysts have a significant potential global influence. It is essential to discuss how our findings fit into this bigger global story and any potential socioeconomic repercussions.

Limitations and Challenges: Every study faces difficulties. To present a fair picture, we must be open and honest about any potential limits of our study, whether they be data-related, model-related, or domain-specific.

6. References

1. Anwar , S. *et al.* (2021) *Recent development in electrocatalysts for hydrogen production through water electrolysis*, *International Journal of Hydrogen Energy*. Available at: https://www.sciencedirect.com/science/article/abs/pii/S0360319921024745.

2. Chisholm, G. and Cronin, L. (2016) *Chapter 16 - hydrogen from water electrolysis - University of Glasgow, Hydrogen From Water Electrolysis*. Available at: https://www.chem.gla.ac.uk/cronin/images/pubs/Chisholm-Chapter_16_2016.pdf.

3. Dubouis, N. and Grimaud, A. (2019) *The hydrogen evolution reaction: From material to interfacial descriptors, Chemical Science*. Available at: https://pubs.rsc.org/en/content/articlelanding/2019/sc/c9sc03831k#!.

4. Liu, F. *et al.* (2022) *Rational design of better hydrogen evolution ... - wiley online library, Rational Design of Better Hydrogen Evolution Electrocatalysts for Water Splitting: A Review*. Available at: https://onlinelibrary.wiley.com/doi/10.1002/advs.202200307.

5. M, K. *et al.* (2022) *Machine learning aided synthesis and screening of her catalyst: Present ..., Machine learning aided synthesis and screening of HER catalyst: Present developments and prospects*. Available at: https://www.tandfonline.com/doi/full/10.1080/01614940.2022.2103980.

6. Umer, M. *et al.* (2022) *Machine learning assisted high-throughput screening of transition metal single atom based superb hydrogen evolution electrocatalysts*, *Journal of Materials Chemistry A*. Available at: https://pubs.rsc.org/en/content/articlelanding/2022/ta/d1ta09878k/unauth.

7. Wang a, Y. *et al.* (2017) *Strategies for developing transition metal phosphides as heterogeneous electrocatalysts for water splitting*. Available at: https://www.sciencedirect.com/science/article/abs/pii/S1748013217301020?casa_token=HHqCXjswKJ0AAAAA%3AXJQ0oYQl2P_EJItE9kg8XKJ4hfnMEq7_ycvpBVQI_5mt_Y0XZRLJcCKe52p5L6HOo72_JYhs0w.

8. Wang, M. and Zhu, H. (2021) *Machine learning for transition-metal-based hydrogen ... - ACS publications*, *Machine Learning for Transition-Metal-Based Hydrogen Generation Electrocatalysts*. Available at: https://pubs.acs.org/doi/10.1021/acscatal.1c00178.

9. Wang, S., Lu, A. and Zhong, C.-J. (2021a) *Hydrogen production from water electrolysis: Role of catalysts*, *Nano convergence*. Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7878665/.

10. Wang, S., Lu, A. and Zhong, C.-J. (2021b) *Hydrogen production from water electrolysis: Role of catalysts - nano convergence*, *SpringerLink*. Available at: https://link.springer.com/article/10.1186/s40580-021-00254-x#Sec2.

11. Wang, S., Lu, A. and Zhong, C.-J. (2021c) *Hydrogen production from water electrolysis: Role of catalysts - nano convergence*, *SpringerOpen*. Available at: https://nanoconvergencejournal.springeropen.com/articles/10.1186/s40580-021-00254-x/tables/1.

12. Yu, F. *et al.* (2018) *Recent developments in earth-abundant and non-noble electrocatalysts for water electrolysis*. Available at: https://www.sciencedirect.com/science/article/abs/pii/S2542529318301469?casa_token=nfHP3cn3ilYAAAAA%3AsdT1QEefoeoxwnFVFnyaWMNd46RiLLWHNmkf44w6oaCJXYCDqBnYAylBR87LqXdgBWuk8pKDyQ.

13. Zhang, B. *et al.* (2017) *New and efficient electrocatalyst for hydrogen ... - ACS publications*, *New and Efficient Electrocatalyst for Hydrogen Production from Water Splitting: Inexpensive, Robust Metallic Glassy Ribbons Based on Iron and Cobalt*. Available at: https://pubs.acs.org/doi/10.1021/acsami.7b09222.

14. Zhang, J. *et al.* (2023) *Accurate and efficient machine learning models for predicting hydrogen evolution reaction catalysts based on structural and electronic feature engineering in Alloys*, *Nanoscale*. Available at: https://pubs.rsc.org/en/content/articlelanding/2023/nr/d3nr01442h/unauth.

15. Eftekhari, A. (2017). Electrocatalysts for hydrogen evolution reaction. *International Journal of Hydrogen Energy*, 42(16), pp.11053–11077. doi:https://doi.org/10.1016/j.ijhydene.2017.02.125.

16. Wang, M. and Zhu, H. (2021). Machine Learning for Transition-Metal-Based Hydrogen Generation Electrocatalysts. *ACS Catalysis*, 11(7), pp.3930–3937. doi:https://doi.org/10.1021/acscatal.1c00178.

17. Aditya Narayan Singh, Kim, M., Meena, A., Wi, T., Lee, H. and Kim, K.S. (2021). Na/Al Codoped Layered Cathode with Defects as Bifunctional Electrocatalyst for High-Performance Li-Ion Battery and Oxygen Evolution Reaction. *Small*, 17(18), pp.2005605–2005605. doi:https://doi.org/10.1002/smll.202005605.

18. Alan Le Goff, Artero, V., Jousselme, B., Tran, P.A., Guillet, N., Romain Metayé, Aziz Fihri, Serge Palacin and Fontecave, M. (2009). From Hydrogenases to Noble Metal–Free Catalytic Nanomaterials for H_2 Production and Uptake. *Science*, 326(5958), pp.1384–1387. doi:https://doi.org/10.1126/science.1179773.

19. Anand, R., Nissimagoudar, A.S., Umer, M., Ha, M., Zafari, M., Umer, S., Lee, G. and Kim, K.S. (2021). Late Transition Metal Doped MXenes Showing Superb Bifunctional Electrocatalytic Activities for Water Splitting via Distinctive Mechanistic Pathways. *Advanced Energy Materials*, 11(48), pp.2102388–2102388. doi:https://doi.org/10.1002/aenm.202102388.

20. Bruix, A., Margraf, J.T., Andersen, M. and Reuter, K. (2019). First-principles-based multiscale modelling of heterogeneous catalysis. *Nature Catalysis*, 2(8), pp.659–670. doi:https://doi.org/10.1038/s41929-019-0298-3.

21. Cheng, W., Zhang, H., Luan, D. and David, W. (2021). Exposing unsaturated Cu_1-O_2 sites in nanoscale Cu-MOF for efficient electrocatalytic hydrogen evolution. *Science Advances*, 7(18). doi:https://doi.org/10.1126/sciadv.abg2580.

22. Chia, X. and Pumera, M. (2018). Characteristics and performance of two-dimensional materials for electrocatalysis. *Nature Catalysis*, [online] 1(12), pp.909–921. doi:https://doi.org/10.1038/s41929-018-0181-7.

23. Chu, S. and Majumdar, A. (2012). Opportunities and challenges for a sustainable energy future. *Nature*, [online] 488(7411), pp.294–303. doi:https://doi.org/10.1038/nature11475.

24. Cobo, S., Heidkamp, J., Jacques, P.-A., Fize, J., Fourmond, V., Guetaz, L., Jousselme, B., Ivanova, V., Dau, H., Palacin, S., Fontecave, M. and Artero, V. (2012). A Janus cobalt-based catalytic material for electro-splitting of water. *Nature Materials*, 11(9), pp.802–807. doi:https://doi.org/10.1038/nmat3385.

25. Dang, N.K., Umer, M., Thangavel, P., Sultan, S., Tiwari, J.N., Lee, J.H., Kim, M.G. and Kim, K.S. (2021). Surface enrichment of iridium on IrCo alloys for boosting hydrogen production. *Journal of Materials Chemistry A*, [online] 9(31), pp.16898–16905. doi:https://doi.org/10.1039/D1TA02597J.

26. Eames, C. and Islam, M.S. (2014). Ion Intercalation into Two-Dimensional Transition-Metal Carbides: Global Screening for New High-Capacity Battery Materials. *Journal of the American Chemical Society*, 136(46), pp.16270–16276. doi:https://doi.org/10.1021/ja508154e.

27. Fischer, A., Müller, J.O., Antonietti, M. and Thomas, A. (2008). Synthesis of Ternary Metal Nitride Nanoparticles Using Mesoporous Carbon Nitride as Reactive Template. *ACS Nano*, 2(12), pp.2489–2496. doi:https://doi.org/10.1021/nn800503a.

28. Glenk, G. and Reichelstein, S. (2019). Economics of converting renewable power to hydrogen. *Nature Energy*, [online] 4(3), pp.216–222. doi:https://doi.org/10.1038/s41560-019-0326-1.

29. Greeley, J., Jaramillo, T.F., Bonde, J., Chorkendorff, I. and Nørskov, J.K. (2006). Computational high-throughput screening of electrocatalytic materials for hydrogen evolution. *Nature Materials*, 5(11), pp.909–913. doi:https://doi.org/10.1038/nmat1752.

30. Harzandi, A.M., Shadman, S., Ha, M., Myung, C.W., Kim, D.Y., Park, H.J., Sultan, S., Noh, W.-S., Lee, W., Thangavel, P., Byun, W.J., Lee, S., Tiwari, J.N., Shin, T.J., Park, J.-H., Lee, Z., Lee, J.S. and Kim, K.S. (2020). Immiscible bi-metal single-atoms driven synthesis of electrocatalysts having superb mass-activity and durability. *Applied Catalysis B: Environmental*, 270, p.118896. doi:https://doi.org/10.1016/j.apcatb.2020.118896.

31. Jin, H., Sultan, S., Ha, M., Tiwari, J.N., Kim, M.G. and Kim, K.S. (2020). Simple and Scalable Mechanochemical Synthesis of Noble Metal Catalysts with Single Atoms toward

Highly Efficient Hydrogen Evolution. *Advanced Functional Materials*, 30(25), p.2000531. doi:https://doi.org/10.1002/adfm.202000531.

32. Kim, D.Y., Ha, M. and Kim, K.S. (2021). A universal screening strategy for the accelerated design of superior oxygen evolution/reduction electrocatalysts. *Journal of Materials Chemistry A*, 9(6), pp.3511–3519. doi:https://doi.org/10.1039/d0ta02425b.

33. King, L.A., Hubert, M.A., Capuano, C., Manco, J., Danilovic, N., Valle, E., Hellstern, T.R., Ayers, K. and Jaramillo, T.F. (2019). A non-precious metal hydrogen catalyst in a commercial polymer electrolyte membrane electrolyser. *Nature Nanotechnology*, [online] 14(11), pp.1071–1074. doi:https://doi.org/10.1038/s41565-019-0550-7.

34. Li, M., Ma, Q., Zi, W., Liu, X., Zhu, X. and Liu, S. (2015). Pt monolayer coating on complex network substrate with high catalytic activity for the hydrogen evolution reaction. *Science Advances*, 1(8). doi:https://doi.org/10.1126/sciadv.1400268.

35. Li, S., Li, B., Feng, X., Chen, L., Li, Y., Huang, L., Fong, X. and Ang, K.-W. (2021). Electron-beam-irradiated rhenium disulfide memristors with low variability for neuromorphic computing. *npj 2D Materials and Applications*, [online] 5(1), pp.1–10. doi:https://doi.org/10.1038/s41699-020-00190-0.

36. McCoy, D.E., Feo, T., Harvey, T.A. and Prum, R.O. (2018). Structural absorption by barbule microstructures of super black bird of paradise feathers. *Nature Communications*, [online] 9(1), pp.1–8. doi:https://doi.org/10.1038/s41467-017-02088-w.

37. Michaela Burke Stevens, Kreider, M.E., Patel, A.M., Wang, Z., Liu, Y., Gibbons, B.M., Statt, M.J., Ievlev, A.V., Sinclair, R., Mehta, A., Davis, R.C., Nørskov, J.K., Gallo, A., King, L.A. and Jaramillo, T.F. (2020). Identifying and Tuning the In Situ Oxygen-Rich Surface of Molybdenum Nitride Electrocatalysts for Oxygen Reduction. *ACS applied energy materials*, 3(12), pp.12433–12446. doi:https://doi.org/10.1021/acsaem.0c02423.

38. Moore, G.W.K., Howell, S.E.L., Brady, M., Xu, X. and McNeil, K. (2021). Anomalous collapses of Nares Strait ice arches leads to enhanced export of Arctic sea ice. *Nature Communications*, [online] 12(1), p.1. doi:https://doi.org/10.1038/s41467-020-20314-w.

39. PubMed. (n.d.). *Nat. Rev. Mater.%5BJour%5D AND 2%5Bvolume%5D AND 1%5Bpage%5D AND 2017%5Bpdat%5D - Search Results*. [online] Available at:

https://pubmed.ncbi.nlm.nih.gov/?orig_db=PubMed&cmd=Search&term=Nat.+Rev.+Mater.%5BJour%5D+AND+2%5Bvolume%5D+AND+1%5Bpage%5D+AND+2017%5Bpdat%5D [Accessed 9 Oct. 2023].

40. Rossmeisl, J., Qu, Z.-W. ., Zhu, H., Kroes, G.-J. . and Nørskov, J.K. (2007). Electrolysis of water on oxide surfaces. *Journal of Electroanalytical Chemistry*, [online] 607(1), pp.83–89. doi:https://doi.org/10.1016/j.jelechem.2006.11.008.

41. Sultan, S., Diorizky, M.H., Ha, M., Tiwari, J.N., Choi, H., Dang, N.K., Thangavel, P., Lee, J.H., Jeong, H.Y., Shin, H.S., Kwon, Y. and Kim, K.S. (2021). Modulation of Cu and Rh single-atoms and nanoparticles for high-performance hydrogen evolution activity in acidic media. *Journal of Materials Chemistry A*, [online] 9(16), pp.10326–10334. doi:https://doi.org/10.1039/D1TA01067K.

42. Sultan, S., Tiwari, J.N., Singh, A.N., Zhumagali, S., Ha, M., Myung, C.W., Thangavel, P. and Kim, K.S. (2019). Single Atoms and Clusters Based Nanomaterials for Hydrogen Evolution, Oxygen Evolution Reactions, and Full Water Splitting. *Advanced Energy Materials*, 9(22), p.1900624. doi:https://doi.org/10.1002/aenm.201900624.

43. Tiwari, J.N., Dang, N.K., Park, H.J., Sultan, S., Kim, M.G., Haiyan, J., Lee, Z. and Kim, K.S. (2020a). Remarkably enhanced catalytic activity by the synergistic effect of palladium single atoms and palladium–cobalt phosphide nanoparticles. *Nano Energy*, 78, p.105166. doi:https://doi.org/10.1016/j.nanoen.2020.105166.

44. Tiwari, J.N., Dang, N.K., Sultan, S., Thangavel, P., Jeong, H.Y. and Kim, K.S. (2020b). Multi-heteroatom-doped carbon from waste-yeast biomass for sustained water splitting. *Nature Sustainability*, 3(7), pp.556–563. doi:https://doi.org/10.1038/s41893-020-0509-6.

45. Tiwari, J.N., Harzandi, A.M., Ha, M., Sultan, S., Myung, C.W., Park, H.J., Kim, D.Y., Thangavel, P., Singh, A.N., Sharma, P., Chandrasekaran, S.S., Salehnia, F., Jang, J., Shin, H.S., Lee, Z. and Kim, K.S. (2019). High-Performance Hydrogen Evolution by Ru Single Atoms and Nitrided-Ru Nanoparticles Implanted on N-Doped Graphitic Sheet. *Advanced Energy Materials*, p.1900931. doi:https://doi.org/10.1002/aenm.201900931.

46. Tiwari, J.N., Singh, A.N., Sultan, S. and Kim, K.S. (2020c). Recent Advancement of p- and d-Block Elements, Single Atoms, and Graphene-Based Photoelectrochemical Electrodes

for Water Splitting. *Advanced Energy Materials*, 10(24), p.2000280. doi:https://doi.org/10.1002/aenm.202000280.

47. Turner, J.A. (2004). Sustainable Hydrogen Production. *Science*, 305(5686), pp.972–974. doi:https://doi.org/10.1126/science.1103197.

48. Wang, C. and Qi, L. (2020). Heterostructured Inter‐Doped Ruthenium–Cobalt Oxide Hollow Nanosheet Arrays for Highly Efficient Overall Water Splitting. *Angewandte Chemie*, 59(39), pp.17219–17224. doi:https://doi.org/10.1002/anie.202005436.

49. Wu, M. and Zeng, X.C. (2016). Intrinsic Ferroelasticity and/or Multiferroicity in Two-Dimensional Phosphorene and Phosphorene Analogues. *Nano Letters*, 16(5), pp.3236–3241. doi:https://doi.org/10.1021/acs.nanolett.6b00726.

50. Xu, H., Cheng, D., Cao, D. and Zeng, X.C. (2018). A universal principle for a rational design of single-atom electrocatalysts. *Nature Catalysis*, 1(5), pp.339–348. doi:https://doi.org/10.1038/s41929-018-0063-z.

51. Yang, Y., Qian, Y., Li, H., Zhang, Z., Mu, Y., Do, D., Zhou, B., Dong, J., Yan, W., Qin, Y., Fang, L., Feng, R., Zhou, J., Zhang, P., Dong, J., Yu, G., Liu, Y., Zhang, X. and Fan, X. (2020). O-coordinated W-Mo dual-atom catalyst for pH-universal electrocatalytic hydrogen evolution. *Science Advances*, 6(23). doi:https://doi.org/10.1126/sciadv.aba6586.

52. Zhou, C., Zhao, J., Peng Fei Liu, Chen, J., Dai, S., Yang, H., Hu, P. and Wang, H. (2021). Towards the object-oriented design of active hydrogen evolution catalysts on single-atom alloys. *Chemical Science*, 12(31), pp.10634–10642. doi:https://doi.org/10.1039/d1sc01018b.

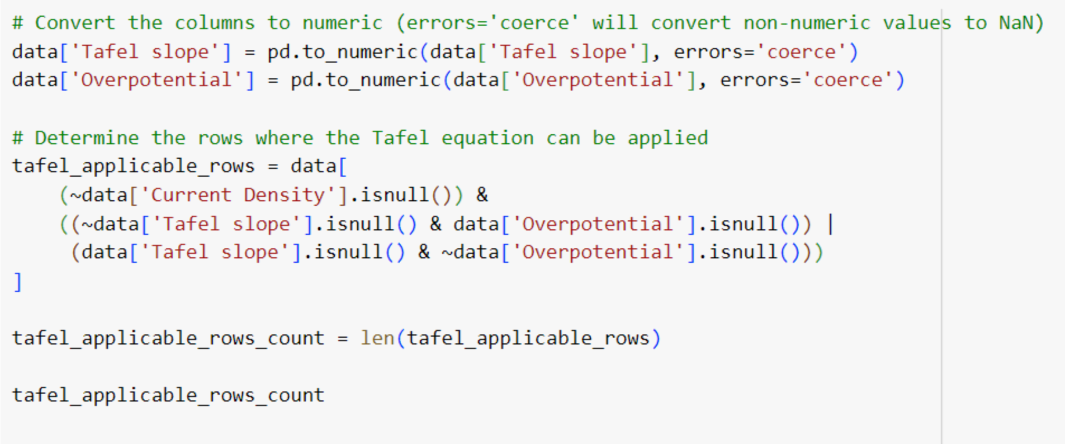
53. Zhuang, Z., Li, Y., Li, Z., Fan Lv, Lang, Z., Zhao, K., Zhou, L., Moskaleva, L.V., Guo, S. and Mai, L. (2017). MoB/g-C_3 N_4 Interface Materials as a Schottky Catalyst to Boost Hydrogen Evolution. *Angewandte Chemie*, 130(2), pp.505–509. doi:https://doi.org/10.1002/ange.201708748.

54. Zhuo, H.-Y., Zhang, X., Liang, J.-X., Yu, Q., Xiao, H. and Li, J. (2020). Theoretical Understandings of Graphene-based Metal Single-Atom Catalysts: Stability and Catalytic Performance. *Chemical Reviews*, 120(21), pp.12315–12341. doi:https://doi.org/10.1021/acs.chemrev.0c00818.

7. Appendices

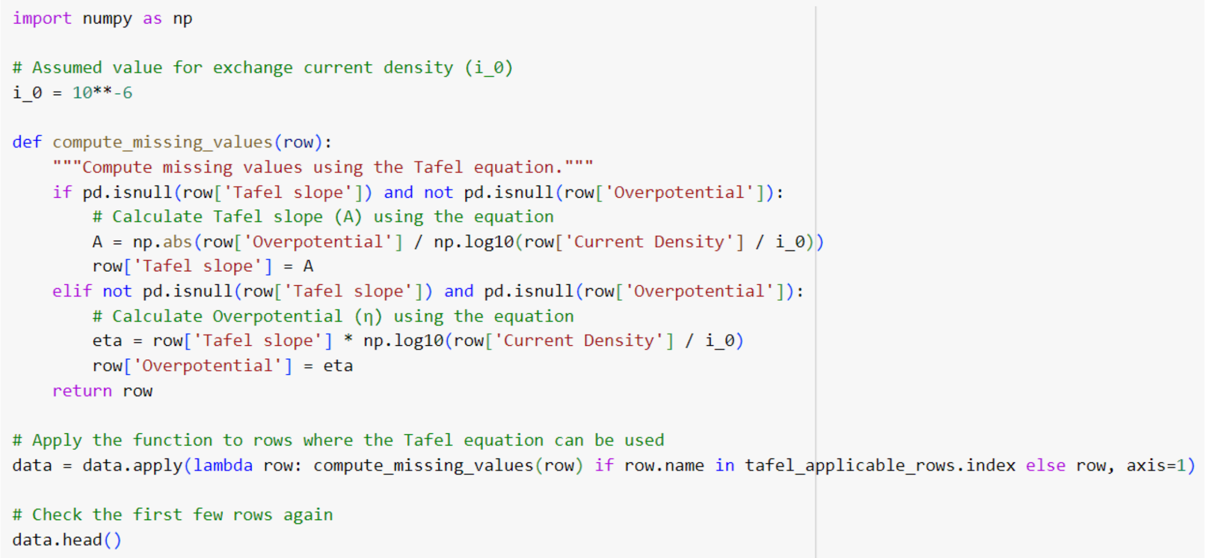
7.1 Alternative Catalyst Code

7.1.1 Handling Missing Values



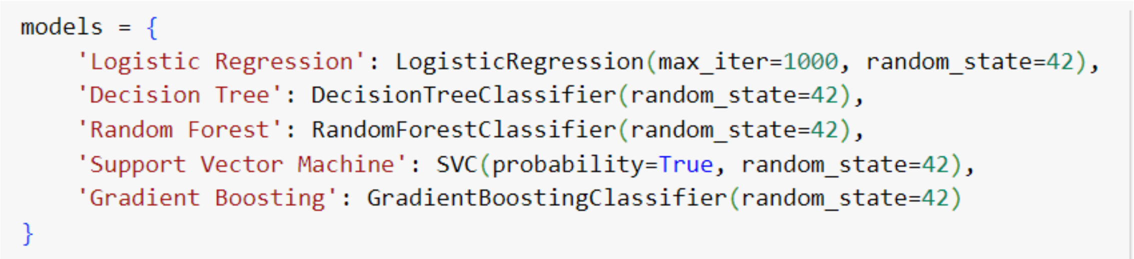
```
# Convert the columns to numeric (errors='coerce' will convert non-numeric values to NaN)
data['Tafel slope'] = pd.to_numeric(data['Tafel slope'], errors='coerce')
data['Overpotential'] = pd.to_numeric(data['Overpotential'], errors='coerce')

# Determine the rows where the Tafel equation can be applied
tafel_applicable_rows = data[
    (~data['Current Density'].isnull()) &
    ((~data['Tafel slope'].isnull() & data['Overpotential'].isnull()) |
     (data['Tafel slope'].isnull() & ~data['Overpotential'].isnull()))
]

tafel_applicable_rows_count = len(tafel_applicable_rows)

tafel_applicable_rows_count
```

7.1.2 Applying Tafel equation to compute the missing values for the Tafel slope and Overpotential in the applicable rows

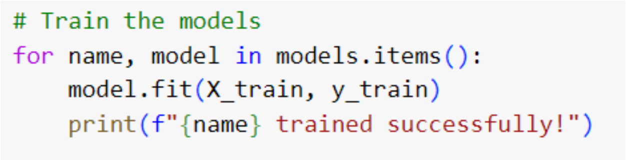


```
import numpy as np

# Assumed value for exchange current density (i_0)
i_0 = 10**-6

def compute_missing_values(row):
    """Compute missing values using the Tafel equation."""
    if pd.isnull(row['Tafel slope']) and not pd.isnull(row['Overpotential']):
        # Calculate Tafel slope (A) using the equation
        A = np.abs(row['Overpotential'] / np.log10(row['Current Density'] / i_0))
        row['Tafel slope'] = A
    elif not pd.isnull(row['Tafel slope']) and pd.isnull(row['Overpotential']):
        # Calculate Overpotential (η) using the equation
        eta = row['Tafel slope'] * np.log10(row['Current Density'] / i_0)
        row['Overpotential'] = eta
    return row

# Apply the function to rows where the Tafel equation can be used
data = data.apply(lambda row: compute_missing_values(row) if row.name in tafel_applicable_rows.index else row, axis=1)

# Check the first few rows again
data.head()
```

7.1.3 Model Building

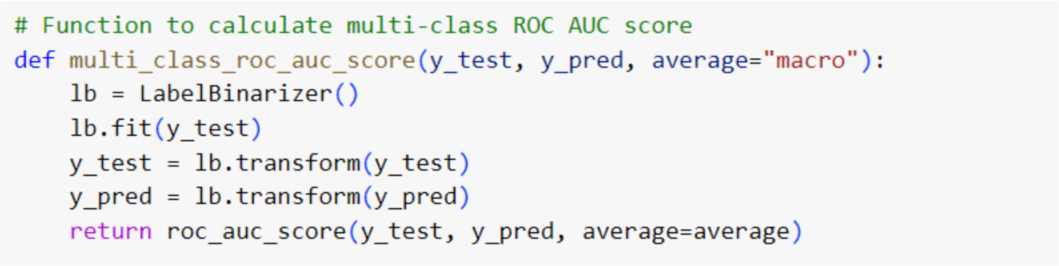


```
models = {
    'Logistic Regression': LogisticRegression(max_iter=1000, random_state=42),
    'Decision Tree': DecisionTreeClassifier(random_state=42),
    'Random Forest': RandomForestClassifier(random_state=42),
    'Support Vector Machine': SVC(probability=True, random_state=42),
    'Gradient Boosting': GradientBoostingClassifier(random_state=42)
}
```

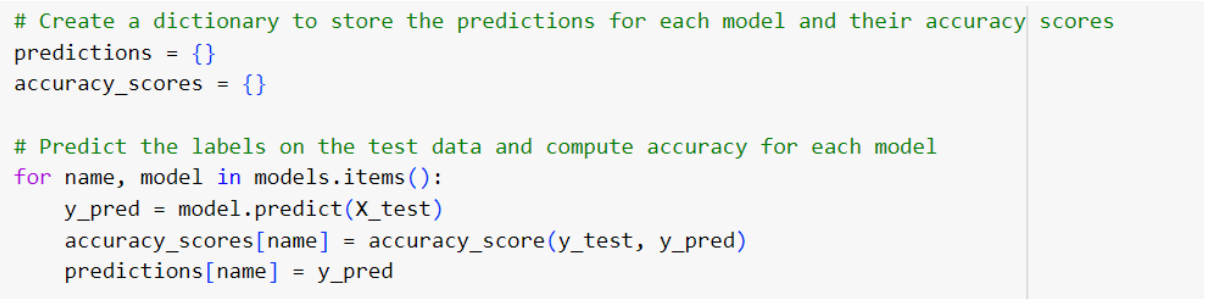


```
# Train the models
for name, model in models.items():
    model.fit(X_train, y_train)
    print(f"{name} trained successfully!")
```

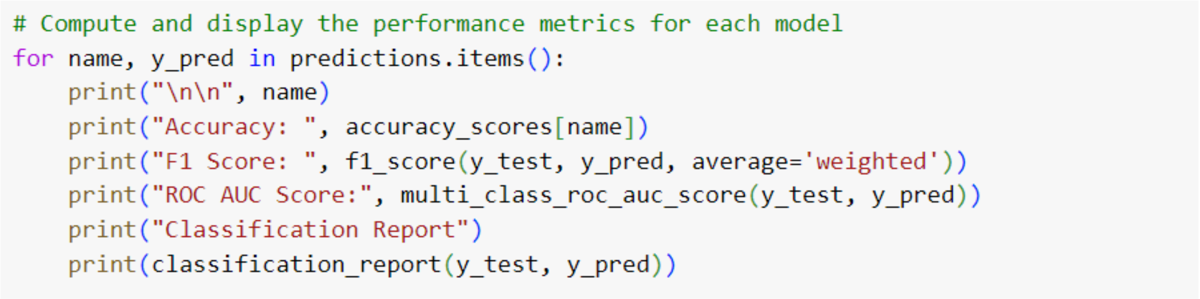
7.1.4 Model Evaluation



```
# Function to calculate multi-class ROC AUC score
def multi_class_roc_auc_score(y_test, y_pred, average="macro"):
    lb = LabelBinarizer()
    lb.fit(y_test)
    y_test = lb.transform(y_test)
    y_pred = lb.transform(y_pred)
    return roc_auc_score(y_test, y_pred, average=average)
```

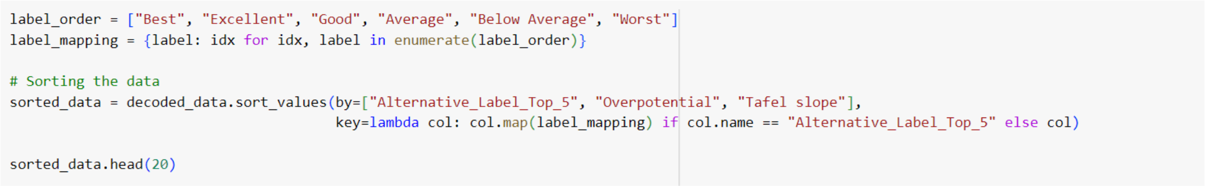


```
# Create a dictionary to store the predictions for each model and their accuracy scores
predictions = {}
accuracy_scores = {}

# Predict the labels on the test data and compute accuracy for each model
for name, model in models.items():
    y_pred = model.predict(X_test)
    accuracy_scores[name] = accuracy_score(y_test, y_pred)
    predictions[name] = y_pred
```

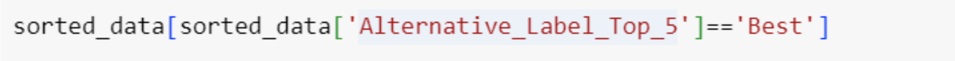


```
# Compute and display the performance metrics for each model
for name, y_pred in predictions.items():
    print("\n\n", name)
    print("Accuracy: ", accuracy_scores[name])
    print("F1 Score: ", f1_score(y_test, y_pred, average='weighted'))
    print("ROC AUC Score:", multi_class_roc_auc_score(y_test, y_pred))
    print("Classification Report")
    print(classification_report(y_test, y_pred))
```

7.1.5 Displaying the best alternative catalysts



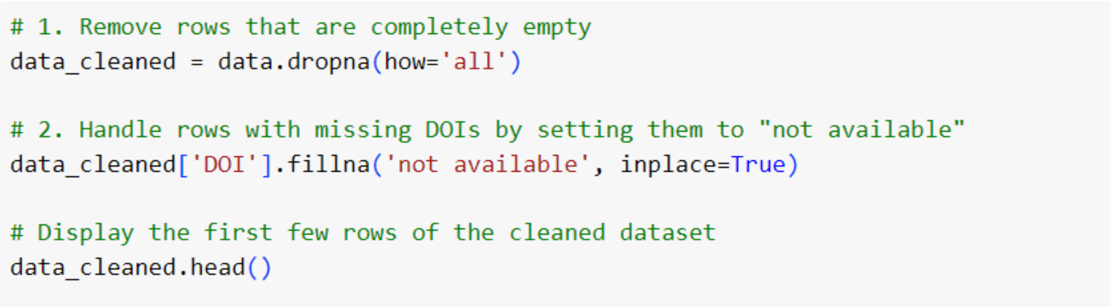
```
label_order = ["Best", "Excellent", "Good", "Average", "Below Average", "Worst"]
label_mapping = {label: idx for idx, label in enumerate(label_order)}

# Sorting the data
sorted_data = decoded_data.sort_values(by=["Alternative_Label_Top_5", "Overpotential", "Tafel slope"],
                                       key=lambda col: col.map(label_mapping) if col.name == "Alternative_Label_Top_5" else col)

sorted_data.head(20)
```



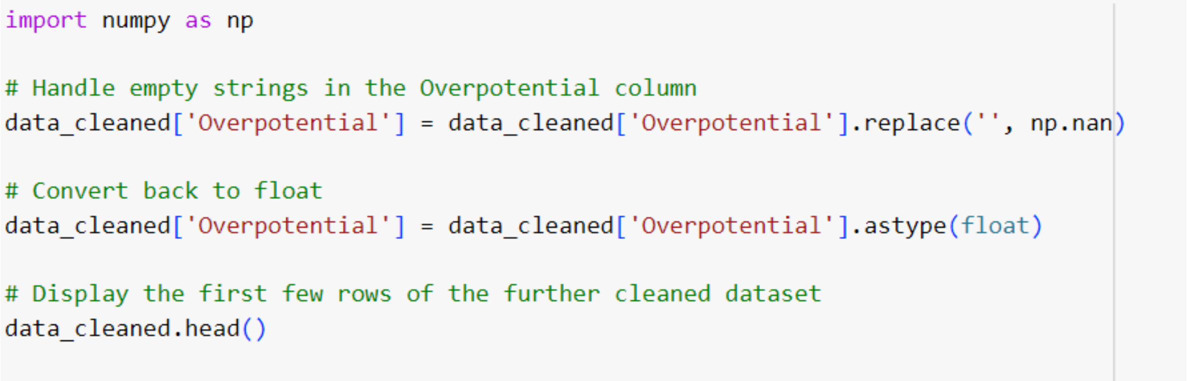
```
sorted_data[sorted_data['Alternative_Label_Top_5']=='Best']
```

7.2 Compound Catalyst Code

7.2.1 Handling Missing Values

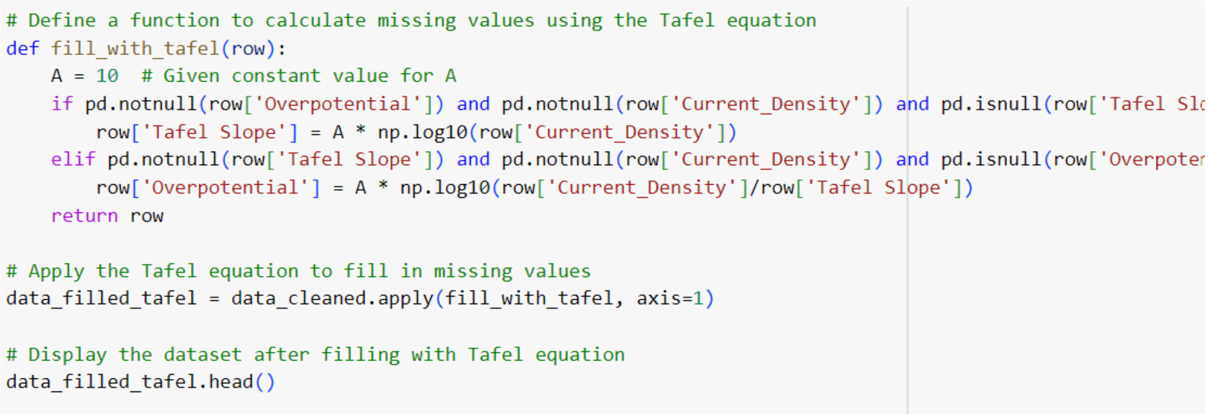


```
# 1. Remove rows that are completely empty
data_cleaned = data.dropna(how='all')

# 2. Handle rows with missing DOIs by setting them to "not available"
data_cleaned['DOI'].fillna('not available', inplace=True)

# Display the first few rows of the cleaned dataset
data_cleaned.head()
```

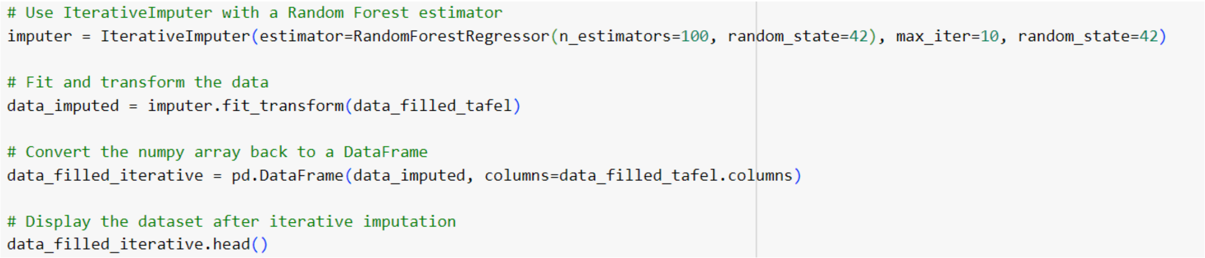


```
import numpy as np

# Handle empty strings in the Overpotential column
data_cleaned['Overpotential'] = data_cleaned['Overpotential'].replace('', np.nan)

# Convert back to float
data_cleaned['Overpotential'] = data_cleaned['Overpotential'].astype(float)

# Display the first few rows of the further cleaned dataset
data_cleaned.head()
```

7.2.2 Predicting Tafel Slope

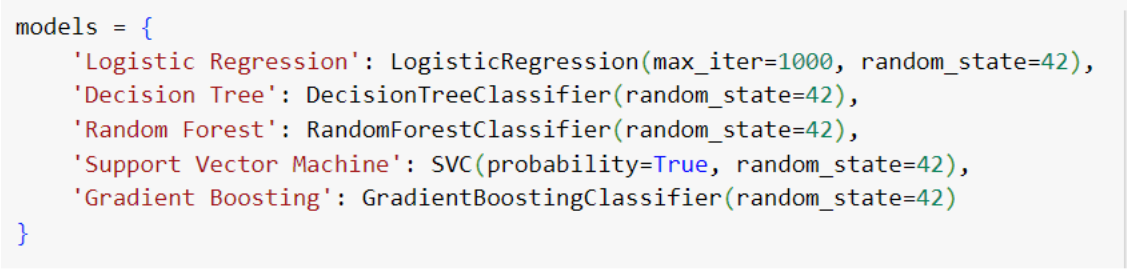


```
# Define a function to calculate missing values using the Tafel equation
def fill_with_tafel(row):
    A = 10  # Given constant value for A
    if pd.notnull(row['Overpotential']) and pd.notnull(row['Current_Density']) and pd.isnull(row['Tafel Slo
        row['Tafel Slope'] = A * np.log10(row['Current_Density'])
    elif pd.notnull(row['Tafel Slope']) and pd.notnull(row['Current_Density']) and pd.isnull(row['Overpoter
        row['Overpotential'] = A * np.log10(row['Current_Density']/row['Tafel Slope'])
    return row

# Apply the Tafel equation to fill in missing values
data_filled_tafel = data_cleaned.apply(fill_with_tafel, axis=1)

# Display the dataset after filling with Tafel equation
data_filled_tafel.head()
```

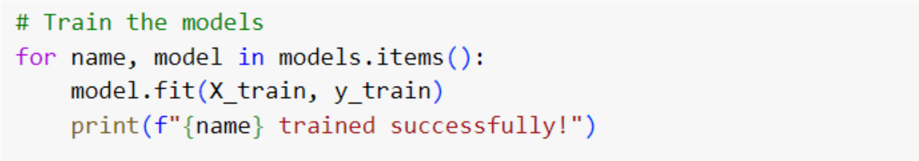


```
# Use IterativeImputer with a Random Forest estimator
imputer = IterativeImputer(estimator=RandomForestRegressor(n_estimators=100, random_state=42), max_iter=10, random_state=42)

# Fit and transform the data
data_imputed = imputer.fit_transform(data_filled_tafel)

# Convert the numpy array back to a DataFrame
data_filled_iterative = pd.DataFrame(data_imputed, columns=data_filled_tafel.columns)

# Display the dataset after iterative imputation
data_filled_iterative.head()
```

7.2.3 Model Building

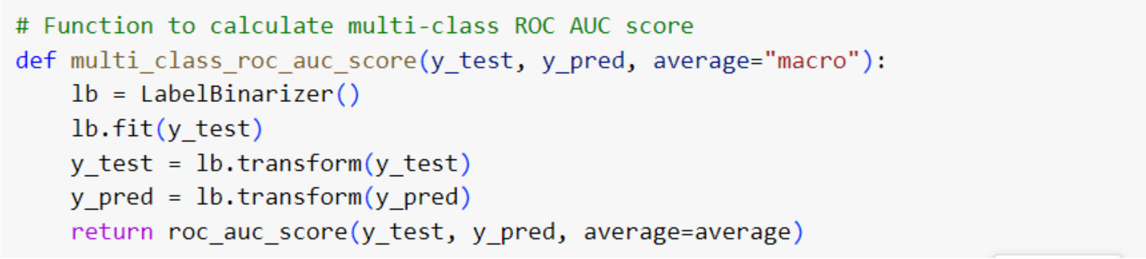


```
models = {
    'Logistic Regression': LogisticRegression(max_iter=1000, random_state=42),
    'Decision Tree': DecisionTreeClassifier(random_state=42),
    'Random Forest': RandomForestClassifier(random_state=42),
    'Support Vector Machine': SVC(probability=True, random_state=42),
    'Gradient Boosting': GradientBoostingClassifier(random_state=42)
}
```

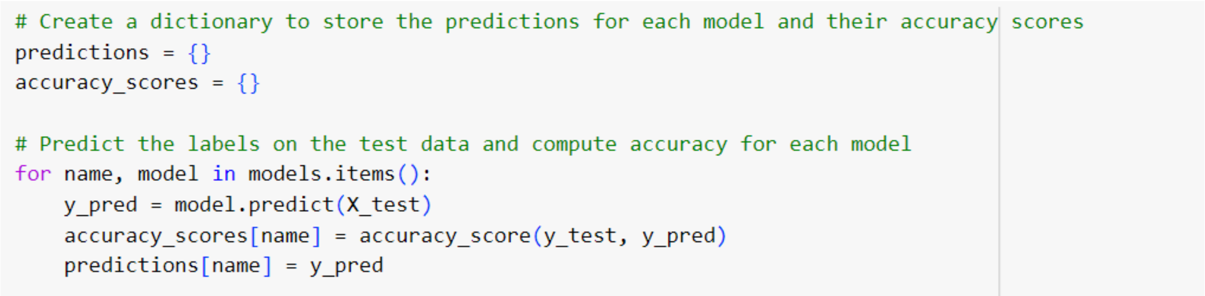


```
# Train the models
for name, model in models.items():
    model.fit(X_train, y_train)
    print(f"{name} trained successfully!")
```

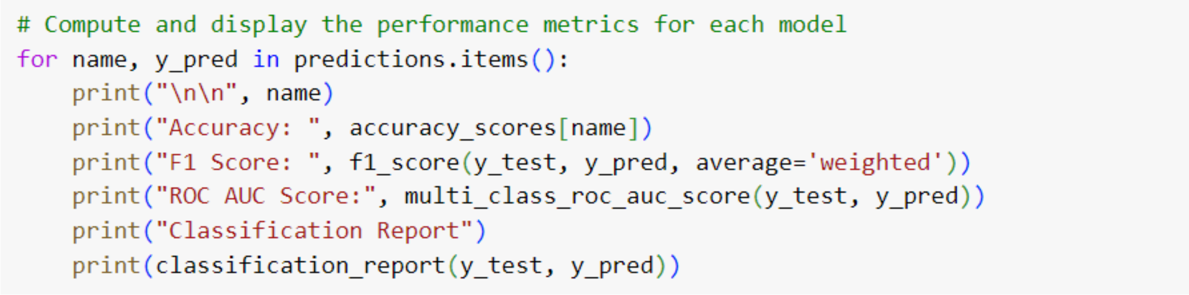
7.2.4 Model Evaluation



```
# Function to calculate multi-class ROC AUC score
def multi_class_roc_auc_score(y_test, y_pred, average="macro"):
    lb = LabelBinarizer()
    lb.fit(y_test)
    y_test = lb.transform(y_test)
    y_pred = lb.transform(y_pred)
    return roc_auc_score(y_test, y_pred, average=average)
```

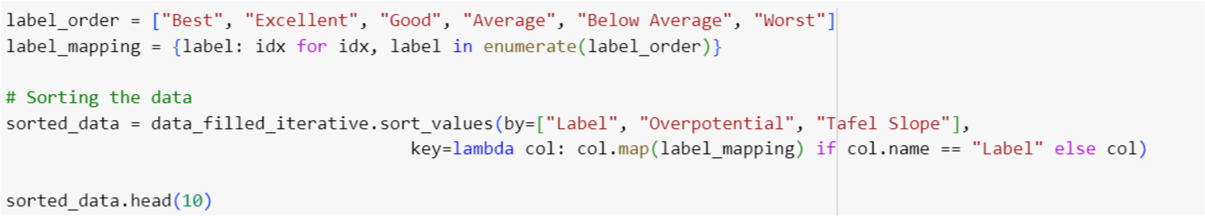


```
# Create a dictionary to store the predictions for each model and their accuracy scores
predictions = {}
accuracy_scores = {}

# Predict the labels on the test data and compute accuracy for each model
for name, model in models.items():
    y_pred = model.predict(X_test)
    accuracy_scores[name] = accuracy_score(y_test, y_pred)
    predictions[name] = y_pred
```

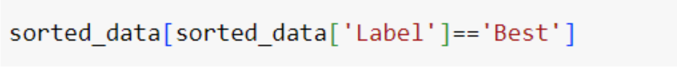


```
# Compute and display the performance metrics for each model
for name, y_pred in predictions.items():
    print("\n\n", name)
    print("Accuracy: ", accuracy_scores[name])
    print("F1 Score: ", f1_score(y_test, y_pred, average='weighted'))
    print("ROC AUC Score:", multi_class_roc_auc_score(y_test, y_pred))
    print("Classification Report")
    print(classification_report(y_test, y_pred))
```

7.2.5 Displaying the best alternative catalysts



```python
label_order = ["Best", "Excellent", "Good", "Average", "Below Average", "Worst"]
label_mapping = {label: idx for idx, label in enumerate(label_order)}

# Sorting the data
sorted_data = data_filled_iterative.sort_values(by=["Label", "Overpotential", "Tafel Slope"],
                                               key=lambda col: col.map(label_mapping) if col.name == "Label" else col)

sorted_data.head(10)
```



```python
sorted_data[sorted_data['Label']=='Best']
```