Exercise 1.1

Question:

1. Look at the first 5 rows of this table. What do you notice about their chemical formula prevents us from this dataset as-is to train a machine learning model?

Check or Submit Answers

Answer:

In the mastml_df["chemicalFormula Clean"], the same chemical formula shows up multiple times. If the same chemical formula appears much more frequently than other formulas, it can lead to an imbalance in the dataset's class distribution. In certain machine learning tasks, class imbalance can impact the model's performance, causing it to be biased towards the more frequently occurring class. If duplicate data represents errors or redundant information in the dataset, it may be necessary to perform data cleaning to ensure data accuracy and consistency. This is also the reason why the groupby and mean functions are used down below.

Exercise 1.2

Question:
1. Look at the starting dataframe mastml_df, How Many data points did we start with? (make sure to look at the very first dataframe we imported and not the one after filtering for reliability) 2. Now look at the cleaned dataframe mastml_df_clean, how many data points do we have now?

Check or Submit Answers

```
[17]
# Answer 1.2
# Look at the starting dataframe mastml_df, How Many data points did we start with?
print("There are", len(mastml_df), "data points in the mastml_df.")

# look at the cleaned dataframe mastml_df_clean, how many data points do we have now?
print("There are", len(mastml_df_clean), "data points in the mastml_df_clean.")
```

Python

```
There are 1447 data points in the mastml_df.
There are 467 data points in the mastml_df_clean.
```

Exercise 1.3

Question:
1. What is the range of band gap values?
2. Think ahead to once we build our model. Would a predicted error of 5 eV be considered small enough to be an accurate or useful prediction?
3. How about a predicted error of 0.5 eV?

Check or Submit Answers

Answer:

1. What is the range of band gap values?

   range = max - min = 13.105 - 0.009 = 13.096

2. Would a predicted error of 5 eV be considered small enough to be an accurate or useful prediction?

   - Because the mean value (mean) represents the average level of all band gap values in the dataset, which is approximately 2.231 eV. The standard deviation (std) measures the dispersion of these band gap values, and a std of 2.287 eV indicates a relatively large degree of dispersion of data points around the mean. Taking these two statistical measures into consideration, when we evaluate an error of 5 eV, we can observe that this error is relatively large compared to the overall range of band gap values. Specifically, a 5 eV error is equivalent to more than twice the mean value, and it is also a significant proportion of the standard deviation. This suggests that the error value has a relatively large level of variation compared to the band gap values, which may indicate that it is not sufficiently accurate or useful. Therefore, from a statistical perspective, a 5 eV error is considered relatively large in relation to the band gap values in the dataset;
   - 25%, 50%, and 75%: in statistics, all numerical values are arranged from small to large and divided into four equal parts, located at three segmentation points. In the dataset, 50% of the band gap values are less than or equal to 1.435 eV. Therefore, if our model's prediction error is 5 eV, this error is significantly larger in comparison to the range of 50% of the band gap values because it exceeds the range of most band gap values. Consequently, from a statistical perspective, a 5 eV error is considered relatively large;
   - range = 13.096, the range of actual band gap values in the dataset is relatively small, thus the error of 5 convers a large range.

3. How about a predicted error of 0.5 eV?

   Considering the three points I made in question 2, a predicted error of 0.5 eV would be considered smaller and potentially more useful than a 5 eV error, but its utility would still depend on the specific application and requirements.

Exercise 1.4

Questions:

1. Is our band gap data balanced (i.e. uniformly distributed across its range)?
2. Given your answer to question 1, Would you expect that the model has similar performance between 0-2 eV as between 10-12 eV?

Challenges (optional questions): 3. What is one thing we can do to address this issue?

Check or Submit Answers

Answer:

1. Is our band gap data balanced (i.e. uniformly distributed across its range)?

   According to the histogram polt, we have more data in 0–4 than 5–12. It appears to be right-skewed or positively skewed, meaning that there are more data points at lower band gap values, and the frequency of data points decreases as band gap values increase. Thus, it is not uniformly distributed across its range and our band gap data is not balanced.

2. Would you expect that the model has similar performance between 0-2 eV as between 10-12 eV?

   Given the imbalance in the band gap data distribution, we would not expect the model to have similar performance between the 0-2 eV range and the 10-12 eV range. Since there are more data points in the lower band gap range (0-2 eV), the model would likely have more data to learn from in that range and may perform better there. Conversely, in the 10-12 eV range, where there are fewer data points, the model may have limited examples to learn from, potentially leading to lower performance in that range.

Exercise 1.5

Questions:

1. What are the five most common elements in the dataset?
2. What are the five least common elements in the dataset?
3. Rank your confidence in the following predictions:
   - predictions containing Oxygen (oxides)
   - predictions containing Iridium
   - predictions containing an element that doesn't appear in the dataset at all

Check or Submit Answers

Answer:

1. What are the five most common elements in the dataset?

   O, Se, S, Te, As

2. What are the five least common elements in the dataset?

   Ta, Y, Ir, Tm, Lu

3. Rank your confidence in the following predictions:

   Scale 0–10, with 10 being very confident and 0 being no hope at all

   - predictions containing Oxygen (oxides): almost 10, very confident, because it is the most common element in the dataset, and its frequent presence suggests it plays a significant role in various compounds;
   - predictions containing Iridium: about 3—with some hope or moderate confidence, but not too much, because it is the third least common element, making predictions involving it less common but still plausible;
   - predictions containing an element that doesn't appear in the dataset at all: Lu, because it is the least common element. As elements not present in the dataset are virtually impossible to predict accurately, given no training data for reference.

Exercise 2.1

Questions:

1. How many features have we generated?

Check or Submit Answers

```
[28]  print("There are", features_df.shape[1], "features that we generated.")
                                                                                Python
```

```
...  There are 87 features that we generated.
```

Exercise 2.2

Questions:

1. Calculate the $AtomicNumber\_CompositionAverage$ for the chemical Formula $\mathrm{Li_3Sb_1}$.$(AtomicNumber_{\mathrm{Li}} = 3, AtomicNumber_{\mathrm{Sb}} = 51)$

Check or Submit Answers

Answer:

Based in the formula above:

AtomicNumber_CompositionAverage = (3 X 3 + 1 X 51) / (3 + 1) = 15

Exercise 3.1

Questions:

1. How many features do we have left?
2. Should you worry about having too few useful features?

Check or Submit Answers

Answer:

1. 86 features;
2. You shouldn't be overly concerned about having too few useful features, as columns with constant values typically don't provide any information about the data in machine learning models and may even have a negative impact on model performance. Therefore, removing these columns with constant values is typically part of data preprocessing to improve model efficiency and performance. In some cases, reducing the number of features can also lower model complexity and prevent overfitting. However, it's important to ensure that a sufficient number of relevant and informative features are retained for the model to make accurate predictions. Therefore, the key is to ensure that the feature selection process is carefully considered to find the most relevant and useful features.

Exercise 3.2

Questions:

1. After filtering for highly correlated features how many features do we have left?
2. Are we worried about having too few useful features?

Check or Submit Answers

Answer:

1. 71;
2. After filtering for highly correlated features, there is a potential concern about having too few useful features left in the dataset. Removing highly correlated features is a common data preprocessing step to reduce multicollinearity and enhance the interpretability and performance of machine learning models. However, it should be done carefully. If a large number of features were highly correlated and were subsequently removed, it's possible that the remaining set of features may be insufficient to capture the complexity of the underlying data. This could lead to a loss of important information, reduced model expressiveness, and potentially poorer predictive performance. Therefore, it's essential to strike a balance between removing redundant or correlated features and retaining enough relevant features to build an effective predictive model. Careful feature selection and domain knowledge can help ensure that the final feature set is both informative and manageable for the machine learning task at hand.

Exercise 4.1

Questions:

1. Does it look like we have enough data that the test split data is representative of the full dataset and the train split data?

Check Answers

Answer:

The test dataset should have a similar data distribution as the training dataset. This means that the test data should encompass similar features and class distributions as those present in the training data in order to reflect the model's generalization ability during testing. According to the bar chart above, the testing dataset and training dataset don't have quite a similar distribution. Therefore, I think we don't have enough data that the test split data is representative of the full dataset and the train split data.

Exercise 5.1

Questions:

1. Take a look at the training parity plot. Evaluate model prediction on training data by answering the following:
    1. Is there enough information in the features to make predictions? Do the features model this data (training) well?
    2. Are there any outliers?
    3. Does it consistently overpredict/underpredict bandgap values in any particular range?

Take a look at all the training data statistics (RMSE, RMSE/std, MAE, R2). Assume these metrics, specifically RMSE, can be used to put an error bar on any predictions we make. Recall that we identified two ML applications for this model, and answer:

2. Can we use this model to predict bandgap values of materials for making single-junction solar cell, which requires a bandgap between 1.1 and 1.7eV?
3. Can we use this model to predict high bandgap materials above 3 eV?

Check or Submit Answers

Answer:

1.
    - From the training data plot, if the model performs well on the training data, the predicted values should closely align with the actual values, forming a distribution that closely follows a 45-degree line. If the plot shows that most data points align along the 45-degree line, it can be inferred that the features contain sufficient information, and the model has effectively fit these features into the training data. Therefore, I think there is enough information in the features to make predictions.
    - Through the training plot, it is possible to observe whether there are outliers that are distinctly different from the other data points. From the graph above, there are no outliers shown. Therefore, there are no outliers.
    - If the predicted values deviate from the 45-degree line and consistently exhibit overestimation or underestimation within a certain range of bandgap values, then the model may perform poorly within that range. From the training data plot, there is no predicted value deviate from the 45-degree line. There is no overpredict/underpredict bandgap values in any particular range.
2. RMSE (Root Mean Square Error) for the training data is 0.0003 eV, while for the test data, it's 1.0492 eV. The RMSE is a measure of prediction accuracy, and in this case, the RMSE for the test data is significantly higher than for the training data. The RMSE value indicates the average prediction error, and an RMSE of 1.0492 eV for the test data is relatively high compared to the desired bandgap range of 1.1 - 1.7 eV. Given these statistics, it's likely that the model may not be accurate enough to reliably predict bandgap values within the required range for single-junction solar cells. The prediction error appears to be larger than what is acceptable for this application.
3. The model has an RMSE of 1.0492 eV in the test data, while it has an RMSE of 0.0003 eV in the training data. This indicates that the model's performance is relatively poor in predicting high bandgap materials, as it struggles to generalize well to the test data, resulting in a significant error range. The model's RMSE and RMSE/std for the test data are 1.0492 eV and 0.4884, respectively. Both of these values indicate relatively high prediction errors for the test data. For applications that require high bandgap materials (greater than 3 eV), the model's errors may be too substantial, making it potentially unreliable in such cases.

Exercise 5.2

Questions:

1. Compare both the parity plots and performance statistics for the training and test set. Is the model performing better on one set than the other, or is there no difference? (No calculation needed.)
2. Which of the following most accurately describes this model: Underfit, overfit, or neither?
3. Should we use training data or test data to estimate model prediction performance?

Check or Submit Answers

Answer:

1. Ideally, data points should align along the 45-degree line in the parity plot, indicating that the model's predictions match the actual values. If, in the training and testing plot, data points for both the training and test data closely follow the 45-degree line, it suggests that the model's performance is likely similar on both datasets. However, if data points for the test data deviate from the 45-degree line while those for the training data are closer to it, it may indicate that the model's performance is poorer on the test data. Therefore, the model is performing better on training data than testing data.
2. If the model fits the training data extremely well but performs poorly on the test data, it may be overfitting. If the model performs poorly on both the training and test data, it may be underfitting. If the model performs well on both datasets, it is likely an appropriate model. Based on the graphs, it should be overfitting.
3. In general, the test dataset is used to assess the model's performance on unseen data, making it suitable for evaluating the model's generalization ability. The training dataset is primarily used for model training and is not a reliable dataset for performance estimation because the model may perform well on the training data but poorly on new, unseen data. Therefore, we should use test data.

Exercise 5.3

Questions:

1. Looking at the Density_composition_average node identified above, how many samples from the training data are included at this node?
2. Trace the branch down from this node until you find a leaf node. How many samples are included at this leaf node?
3. Inspect a few other leaf nodes until you identify how many samples are at the majority of leaf nodes, how many samples is this?
4. based on your previous analysis of overfitting, underfitting, or well fit does this support that previous claim about how the model is fit?

Check or Submit Answers

Answer:

1. At the first density composition average (≤ 0.08) node, there are samples = 420 from the training data are included.
2. At the leaf node, there is samples = 1 are included at the leaf node.
3. samples = 1.
4. If the model performs well on the training data but poorly on the test data, and most leaf nodes contain a similar number of samples, it may be a sign of overfitting. If the model performs poorly on both the training and test data, and the distribution of samples in the leaf nodes is uneven, it may be a sign of underfitting. If the model performs well on both the training and test data, and the distribution of samples in the leaf nodes is even, it supports a well-fit model. Based on the result, it should be overfitting.

Exercise 6.1

Before diving into the questions lets run a few different grid searches to get a feel for how it works. By default the notebook is setup with a very rough grid search which should run quickly. be careful adding too many grid points because it is possible to slow down the grid search to the point of not finishing in hours or days. As a rule of thumb lets not set number of trees to be above 100, and lets not include more than 10 individual grid points in any one search. Try to get a sense of how performance varies with number of trees. Before answering questions below set your grid to be [1,3,5,7,10,15,20,50] which should give a reasonable spread of values. As a reminder these edits are made in the section above "defining a parameter space"

Questions:

1. Looking just at the training data curve above, what is the number for `n_estimators` that gives the smallest MSE?
2. How about for the validation data? Which value for `n_estimators` gives the smallest MSE?
3. Look at the trend in validation data. If n_estimators was increased beyond 50 do you think the model would change in performance significantly?
4. Do we expect the model get more complex as we increase the number of trees?
5. Looking at just the validation curve, does increasing number of trees seem to increase overfitting at any point on the curve? This would show up as worsening performance in the validation data as trees increase.

Check or Submit Answers

Answer:

1. 50
2. 50
3. If the MSE stabilizes and no longer significantly decreases beyond 50 trees, adding more trees may not significantly change the model's performance. Typically, the curve levels off after a certain number of trees. Therefore, I think the model would not change in performance significantly.
4. The complexity of a Random Forest model does increase to some extent as the number of trees is increased, but it does not increase linearly. Random Forest is an ensemble learning method composed of multiple decision trees, each of which is trained on the data. Each tree is a basic classification or regression model, and their complexity is typically not very high.
5. If the MSE of the validation data continues to decrease as the number of trees increases, the model may be overfitting to the validation data. Overfitting can lead to a decrease in performance on new, unseen data. Therefore, if the MSE of the validation data starts to rise after a certain number of trees, it may be a sign of overfitting. When the curve shows signs of stabilization or decreasing validation performance, it typically means that as the number of trees increases, the Mean Squared Error (MSE) or other performance metrics start to level off or even increase. This may indicate that the model no longer improves significantly with an increased number of trees and might even start to overfit the data. Therefore, as the number of trees increases, and the validation performance no longer significantly improves, it's a signal that should be approached with caution.

Exercise 6.2

Questions:

1. Do we get improvement in the RMSE between the default and optimized model? What is the percentage improvement ($|RMSE_{testopt} - RMSE_{testdefault}|/RMSE_{testdefault}$)?

For this lab we'll assume that the RMSE metric gives a reasonable error estimate on predictions. So any prediction we made we'll assume it has + or - the CV RMSE value.
2. Assuming this level of accuraccy from the optimized model. Is our model accurate enough to predict single-junction solar materials? where the key design metric is having a band gap between 1.1 eV and 1.7 eV?
3. How about our other task. Is the optimized model accurate enough to predict high bandgap materials? where the key metric is ensuring predictions are above 3 eV?

Check or Submit Answers

Answer:

1. Percentage Improvement = |1.1311 − 1.4691| / 1.4691 × 100 ≈ 23.01%;

   Yes, we have indeed observed an improvement in RMSE between the default model and the optimized model. The average test RMSE for the optimized model is 1.1311 eV, whereas for the default model, it is 1.4691 eV. Therefore, the RMSE of the optimized model is lower than that of the default model, indicating an enhancement in model performance on the test data.

2. With the optimized model's RMSE of 1.1311, it indicates that, on average, the model's predictions have an error of about 1.1311 eV. While this represents an improvement over the default model, it's still relatively high for predicting single-junction solar materials, where the key design metric is having a band gap between 1.1 eV and 1.7 eV. The model's accuracy may not be sufficient for this task.

3. Similarly, for predicting high bandgap materials (above 3 eV), the optimized model's performance with an RMSE of 1.1311 may still not be accurate enough. High bandgap materials require precise identification, and the model's average error of about 1.1311 eV may not meet this requirement effectively.

Exercise 6.3

Questions:

1. Look at both the parity plots and the training data statistics. Does the optimized model do better or worse at predicting the training data than the default model?
2. Do prediction performances on training data give you enough information to decide which model is more likely to give better predictions on Si and SiO_2, which are not in the training set or the test set? Another way to ask this is does the training data result tell us anything about the predictive power of the model?

Check or Submit Answers

Answer:

1. The default model has an RMSE of 0.0003 eV on the training data, while the optimized model has an RMSE of 0.4226 eV. The RMSE/std for the default model on the training data is 0.0001, while for the optimized model, it's 0.1839. The default model has an MAE of 0.0 eV on the training data, while the optimized model has an MAE of 0.274 eV. The R2 for the default model on the training data is 1.0, while for the optimized model, it's 0.9662. From these data, it appears that the optimized model performs relatively worse at predicting the training data compared to the default model. The RMSE, RMSE/std, MAE, and R2 for the optimized model are all larger than those of the default model, indicating that the optimized model has a larger prediction error on the training data and may not have fit the training data very well.

   If the model performs well on the training data, the predicted values should closely align with the actual values, forming a distribution that closely follows a 45-degree line. If the predicted values deviate from the 45-degree line and consistently exhibit overestimation or underestimation within a certain range of bandgap values, then the model may perform poorly within that range. According to the graph, the distribution closely follows a 45-degree line in the default model. However, in the optimized model, the predicted values deviate from the 45-degree line. Again, the optimized model has a larger prediction error on the training data and may not have fit the training data very well.

2. The results on the training data are primarily used to assess the model's fit to the training data but may not adequately reflect the model's performance on unknown materials such as Si and SiO2, which are not included in the training set or test set. To evaluate the model's predictive ability on materials like Si and SiO2, it is best to directly make predictions on these materials and assess the model's performance. Therefore, the performance of the training data typically cannot serve as a sufficient basis for predicting the model's performance on unknown materials.

Exercise 6.4

Questions:

1. Just looking at the testing data statistics, does the optimized model do better or worse at predicting the testing data?
2. Compare the difference between train and test RMSE for the default and optimized model. Did the difference between training and test performance increase or decrease after hyperparameter optimization?
3. Is this evidence that the optimized model is more overfit or less overfit?

Check or Submit Answers

Answer:

1. Looking at the testing data statistics, the optimized model performs better at predicting the testing data. The optimized model has a testing data RMSE of 0.6062 eV, whereas the default model has a testing data RMSE of 1.0492 eV. Additionally, the optimized model outperforms the default model in terms of testing data RMSE/std, MAE, and R2.

2. The difference in RMSE between the training and test sets for the default model is 1.0492 - 0.0003 = 1.0489 (eV), while for the optimized model, it is 0.6062 - 0.4226 = 0.1836 (eV). The difference between training and test RMSE decreased after hyperparameter optimization. Specifically, the difference decreased from approximately 1.0489 (eV) for the default model to approximately 0.1836 (eV) for the optimized model.

3. This decrease in the difference between training and test RMSE suggests that the optimized model is less overfit compared to the default model. The smaller gap between training and test performance indicates improved generalization of the optimized model to unseen data, which is evidence of reduced overfitting.

Exercise 6.5

Questions:

1. Based on your answers to exercise 6.3 and 6.4, if you were to choose between the default model and the optimized model, what decision will you make? Give at least 3 criteria for your evaluation.

Check or Submit Answers

Answer:

- Lower Test RMSE: The optimized model has a lower Test RMSE (0.6062 eV) compared to the default model (1.0492 eV). Lower RMSE indicates that the optimized model provides more accurate predictions on the test data, which is a critical criterion for model selection.
- Reduced Overfitting: The optimized model exhibits a smaller difference between the Training and Test RMSE (0.1836 eV) compared to the default model (1.0489 eV). This suggests that the optimized model is less prone to overfitting and generalizes better to unseen data, which is an essential factor for model reliability.
- Higher Test R2 Score: The optimized model has a higher Test R2 score (0.9204) compared to the default model (0.7614). A higher R2 score indicates that the optimized model explains a larger portion of the variance in the test data, signifying its superior predictive power.

Exercise 7.1

Questions:

The first task was a use case where we want to predict single-junction solar materials and they need to have a band gap between 1.1 eV and 1.7 eV to have decent efficiency.

1. In the Test dataset how many materials do we have with band gaps within the range of being a good solar material? Note in terms of classification we are identifying the number of positive cases in the dataset.
2. Now we'll compare to the Predictions. Find the number of times the positive cases identified in question 1 are predicted to be single-junction solar materials (between 1.1 and 1.7 eV). This is referred to as the number of true positives, the number of times the positive cases were predicted correctly. Divide this number of true positives by the total number of positive cases (from question 1) to obtain the recall value. What is the recall of our pseudo-classifier to predict single-junction solar materials?

The second task was to identify high bandgap materials such as GaN, where the bandgap at or above 3 eV.
3. In the Test dataset how many materials do we have with band gaps at or above 3 eV?
4. Perform the same process from question 2 (remember are classes are now defined differently for this new task) and calculate the recall for predicting high bandgap materials. What is the recall in this case?

Think about the performance of the model in correctly completing both of the prediction tasks on the test dataset. 5. Based on the evidence from questions 1-4 which tasks can the model succeed at?

Challenges (optional questions): 6. There are a number of other classification metrics we could have used to assess the performance of the model on the two potential tasks. Calculate the precision and false discovery rate for the two tasks. Do these metrics support the conclusion you made in question 5? We ignored this earlier in favor of simplicity, but potentially there is a more nuanced understanding of how the models might perform.

Check or Submit Answers

Answer:

1. 7

2. 2 is the number of true positives; recall value = 2 / 7 = 0.2857

3. 9

4. 9 is the number of true positives; recall value = 9 / 9 = 1

5. The model can identify all high bandgap materials (band gaps at or above 3 eV) with a recall of 1, indicating perfect performance in this task. The model can predict single-junction solar materials (band gaps between 1.1 eV and 1.7 eV) with a recall of approximately 0.2857, which is relatively low but still provides some predictive capability. So, the model succeeds in identifying high bandgap materials but has limited success in predicting single-junction solar materials.