Critical Temperature in Superconductor

September 7, 2022

Joaquin Dominguez, Richard Kim, Hien Lam

1 Introduction

Paragraph here on background of superconductor and why critical temperature is important.

2 Methods

2.1 Data Preprocessing

The superconductivity dataset consisted of two files: train and material. The former file contained 82 relevant feature information from 21,263 superconductors including the response feature, `critical_temp`. The material file contained 88 features which represented the chemical formula for `material` that were one hot encoded for all 21,263 superconductors. A simplistic example for water (H2O) would have a two under the hydrogen column, one under oxygen, and zero for all the other elements. The two files were concatenated and `material` subsequently dropped since this feature denoted the full chemical formula and deemed unnecessary after hot encoding. Next, we identified the nine features that had a single value and consequently dropped them due to their lack of usefulness in predictive performance. Lastly, we confirmed that the data was free of missing values and duplicated records. The final data frame resulted in 21263 rows and 159 columns. We will note the datatypes were all numerical and proceeded with exploratory data analysis.

2.2 Exploratory Data Analysis

Mention/discuss regression assumptions?

2.3 Feature Scaling

Given that the features followed a normal distribution, we felt confident Scikit-Learn's standard scaler class was the most appropriate feature scaling for the dataset. This estimator bounded each feature to maintain a mean of zero and standard deviation of one. We conducted linear regression with scaled and unscaled data and compared their performance by way of five-fold internal cross validation. Looking at R2 and root mean squared error (RMSE), both models produced equivalent metrics at 0.45 and 23.8, respectively. We split the data into X and y (training features and target feature, respectively) and proceeded with standard scaler going forward.

2.4 Modeling

As stated above, we would like to predict the critical temperature of a superconductor using linear regression with regularization then investigate the important features. Specifically, lasso and ridge regularization were explored. Briefly explain what lasso and ridge is here. We utilized Scikit-Learn's pipeline class to set up the gridsearch workflow, lasso and ridge classes to train the models, gridsearchcv class to execute the search and refit the best performing model and cross_validate class to ascertain their mean RMSE from five-fold, shuffled internal cross validation. We tuned lasso's alpha to be between 0.1-10 with 200 samples and ridge's alpha to be 0.1-100 with 300 samples. The feature importance was extracted from the best model of each algorithm and its values transformed to the absolute root because we care about the magnitude of the coefficients, not the sign of the coefficient.

3 Results

3.1 Regularization

Displayed below is a table of the optimal alpha from each model and their respective five-fold cross validation metrics.

	Lasso	Ridge
Alpha	2.03	100
R2	0.66	0.53
RMSE	20.09	22.43

3.2 Feature Importance

Words

4 Conclusion

Linear regression with lasso regularization produced the lowest RMSE, albeit quite small difference, but also 13% R2 increase with fewer parameters.

5 Code

Please refer to the attached Python notebook.