**CHAPTER 3: METHODOLOGY**

* 1. **Data Collection**

Since, our objective for the project is to build supervised learning models to predict the tensile strength and thermal conductivity of alloys, the first and foremost step is to create a reliable dataset.

As discussed earlier, the tensile strength of an alloy varies with multiple factors such as chemical composition, form factors and temper factors. The thermal conductivity of an alloy is fixed for a given chemical composition, however a difference in composition will almost always bring a change in thermal conductivity. Therefore, our dataset should contain features that incorporate all the above-mentioned information for each alloy.

The data used in this project was collaboratively compiled from multiple sources including the web, educational software and past research papers. These sources are Copper Org website, CES Edu pack and past research papers.

* + 1. Data Overview :

A view of the dataset can be found in Figure. Each alloy is represented by a single row of the csv file represented by the following features :

1. Name (Identifier)
2. Temper Conditions
3. Form Conditions
4. Mass Percentage Composition of alloys –

Named as the corresponding chemical symbol of each element in the periodic table. eg. ‘Cu’ column contains the percentage (by mass) of copper (Cu) in each alloy, ‘Al’ column contains the percentage of aluminium (Al) and so on.

1. Tensile Strength (ksi units) - Measured at room temperature, 68°F (20°C)
2. Thermal Conductivity (*Btu/ sq ft/ ft hr/ °F*) - Measured at room temperature, 68°F (20°C)

The data that has been collected has been made such that there are no missing values for any of the alloys. If an alloy does not contain an element, the feature column corresponding to that element is set as 0.

* 1. **Data Pre-processing**

The data was pre-processed for our supervised modelling step using the following steps:

* Remove the identifier column (Name):

Since, identifiers like name do not have any effect on the tensile strength or the thermal conductivity of an alloy, we will remove them from the dataset.

* Encode categorical variables (Temper and Form):

This was done using the concept of dummy variables, which is a suitable method of encoding categorical variables for machine learning models (as discussed earlier)

* Removing duplicates of alloys from the dataset:

For those alloys (if any) which occurred in the dataset more than once we retained only a single instance of the alloy. (The benefits of this are as discussed earlier)

* Removing Temper and Form columns (only for Thermal Conductivity prediction):

Since, the Thermal Conductivity of an alloy is only dependent on the its chemical composition, columns containing other information (Temper and Form) were also removed for the dataset compiled for thermal conductivity.

* 1. **Train Test Split**

The data was divided into train and test sets. For the following sections (from to ) only the training dataset has been used, while the test dataset is saved for the final evaluation.

* 1. **Model Selection**

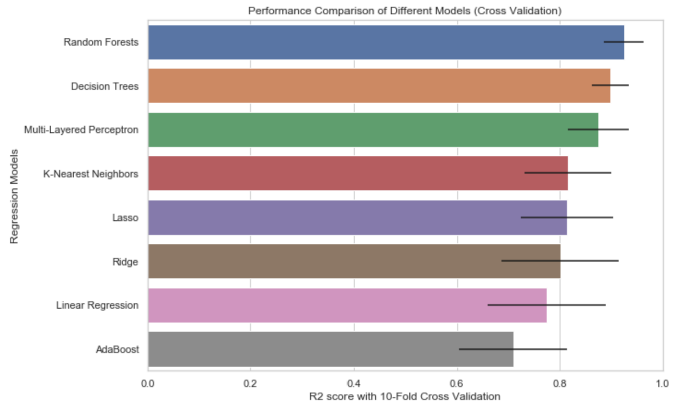
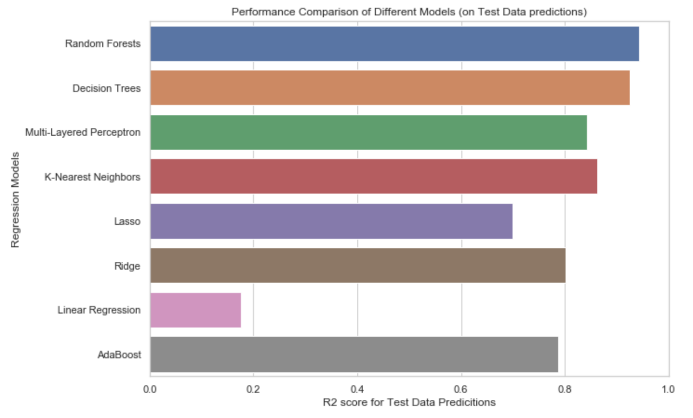
The model selection phase was carried out using the pipeline shown in figure.

* + 1. For Tensile Strength

The following regression models from scikitlearn’s machine learning library were used to predict the tensile strength of copper alloys using their composition, form, and temper:

* Random Forests
* Linear Regression
* Decision Trees
* K-Nearest Neighbours
* Multi-Layered Perceptron
* XGBoost

The figure shows a comparison of their cross-validation performances on the training data.

**** ****

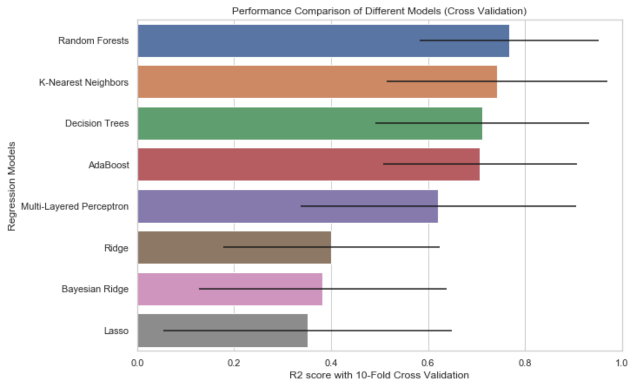
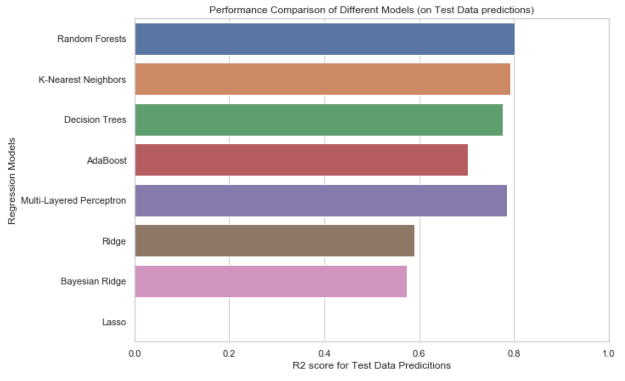
As can be seen from the figure, Random Forests had the best performance score. Therefore, the Random Forest model was selected and taken to the step of hyperparameter tuning.

* + 1. For Thermal Conductivity

The following regression models from scikitlearn’s machine learning library were used to predict the thermal conductivity of copper alloys using only their composition features:

* Random Forests
* Linear Regression
* Decision Trees
* K-Nearest Neighbours
* Multi-Layered Perceptron
* XGBoost

The figure shows a comparison of their cross-validation performances on the training data.

As can be seen from the figure, Random Forests had the best performance score. Therefore, the Random Forest model was selected and taken to the step of hyperparameter tuning.

* 1. **Hyperparameter Tuning**

The hyperparameter tuning step is necessary one since it is shown to optimise model performance (**as discussed earlier**). The tuning process was carried out separately for tensile strength and for thermal conductivity. The following sections contain the details of our process.

* + 1. For Tensile Strength

Hyperparameters of Random Forests contain the following:

* N\_estimators
* Min\_split\_size

A screenshot of a cell phone

Description automatically generatedA screenshot of a social media post

Description automatically generated

All the above parameters were used to create separate models and were evaluated on the basis of their cross-validation performances using scikitlearns’ inbuilt module GridSearchCV. (The workings of GridSearchCV are as discussed earlier). The results of the model parameters are as shown here.

The parameters which created the best performing model was ultimately selected.

The parameters of this model are here.

* + 1. For Thermal Conductivity

Hyperparameters of Random Forests contain the following:

* N\_estimators
* Min\_split\_size

All the above parameters were used to create separate models and were evaluated on the basis of their cross-validation performances using scikitlearns’ inbuilt module GridSearchCV. (The workings of GridSearchCV are as discussed earlier). The results of the model parameters are as shown here.

A screenshot of a social media post

Description automatically generated

A screenshot of a cell phone

Description automatically generated

The parameters which created the best performing model was ultimately selected.

The parameters of this model are here.

* 1. **Model Evaluation**

Once the final models optimised, they were used to make predictions on the test data we separated earlier in section above. The following tables show the summary of the model performances.

* + 1. Tensile Strength Model

A screenshot of a map

Description automatically generatedA screenshot of a social media post

Description automatically generated

* + 1. Thermal Conductivity Model

A close up of a map

Description automatically generated

**A screenshot of a cell phone

Description automatically generated**

* 1. **Inverse Modelling**

The objective of this step is to create a system with the function which takes an input of the desired value a physical property and uses the model above to output alloy compositions which are likely to have that desired property value.

The following procedure was followed for the inverse modelling:

1. Take user inputs (desired value, property, number of predictions)
2. Choose 10 samples from the dataset with values closest to the desired value
3. Generate new data by shuffling values within these samples (as shown here)
4. Use models to make predictions on newly generated samples.
5. Return alloys from dataset (if any) and from the predictions, which have the closest values of the desired property (along with a confidence percentage)

The predicted compositions were then evaluated against the data available on Copper Org website