A Case-Study on

PREDICTING MECHANICAL PROPERTIES OF LOW-ALLOY STEEL USING MACHINE LEARNING

by
SHREYAS KUMAR TAH

Visit my Github https://github.com/skt-shreyas for the notebook and the codes.

Case Study

I. Abstract : Selection of an appropriate material by any design engineer is very essential and to do so it is necessary to understand the mechanical properties of the material to select them for the engineering applications. Currently there are no precise theoretical methods to predict the mechanical properties of metals. All the methods available, include statistical and extensive physical testing of the materials which require a lot time along with patience. Since the testing of each material with different composition is a highly tedious task so their is a need for the usage of the advanced tools to predict and analyse some of the major properties metals using it. This paper includes determining the mechanical properties of various alloys of Carbon(6), Silicon(14), Manganese(25), Phosphorous(15), Sulphur(16), Nickel(28), Chromium(24), Molybdenum(42), Copper(29), Vanadium(23), Aluminium(13), Nitrogen(7) using their weight percentage taken at different temperature in celsius for each test. Lastly mechanical properties including tensile strength, yield strength, elongation and reduction in area are predicted using regression algorithms of machine learning. The challenge is to predict the mechanical properties using the alloy composition and temperature.

II. Introduction : An alloy is a substance that is combined or mixed with more than one metal or nonmetal. For example, brass is an alloy of two metals: copper and zinc. Steel is an alloy of a metallic element (iron) and a small amount of a non-metallic element (carbon - upto 2%). The mechanical properties of these metals are associated with the ability of the material to resist mechanical forces and load and are determined by their microstructures, which are governed by its chemical constituents and manufacturing processes. These properties determine the range of usefulness of a material and establish their period of service life. Choosing the right alloy for any particular usage is very important. One should know the environmental conditions it will be working in, and the different engineering properties it would be showcasing in that environment as they may be substantially different with respect to its constituent elements. This is the reason metal alloys are in high demand in a variety of applications and industries such as manufacturing, electronics, domestic goods, architecture, plumbing, and the automotive and aerospace industries.

However, the selection of an appropriate materials for a product design is a difficult and complex task. This problem can be summarized into two categories, materials selection based on the material properties, and another based on the design requirements. New sustainable methods and research are being conducted to compensate for these problems. This paper discuses one such method by using of advanced tools like machine learning algorithms to determine or predict the enigeering properties of different alloys using its composition and temperature. The general outcome of the material selection process is to identification that one or more materials with properties which satisfy the functional requirements of our need, such as strength, stiffness, etc. Through this study we would discover such advanced techniques using machine learning that could tell us the properties of any type of alloy easily.

Apart from predicting the properties of materials, machine learning provides a new means of screening novel materials with good performance, developing quantitative structure-activity relationships (QSARs), discovering new materials and performing other materials relateds studies. Artificial intelligence, especially machine learning (ML) and deep learning (DL) algorithms, is becoming an important tool in the fields of materials and mechanical engineering. Recent breakthroughs in ML techniques have created vast opportunities for not only overcoming long-standing mechanics problems but also for developing unprecedented

materials design strategies.

III. Case Study :

III.1 Title : To predict the mechanical properties of alloys using different metal composition and temperature.

III.2 Objective : To understand the application of machine learning algorithms in the field of materials and mechanical engineering.

III.3 Tools used : Kaggle Notebooks, Numpy, Pandas, Matplotlib, Seaborn, Plotly, Scikit Learn, Machine Learning Algorithms, Regression Models.

III.4 Outcome :

Students are able to perform the required data preprocessing, data cleaning, feature engineering, etc to make the dataset suitable for model building.

To get familiar with different types of machine learning models, regression and classification and its applications.

Finanlly to understand the application of machine learning in the field of materials and mechanical engineering.

III.5 Dataset :

Dataset contains 20 columns and 915 rows. This comprises compositions by weight percentages of low-alloy steels along with the temperatures at which the steels were tested and the values mechanical properties observed during the tests. The alloy code is a string unique to each alloy. Weight percentages of alloying metals and impurities like Aluminum, copper, manganese, nitrogen, nickel, cobalt, carbon, etc are given in columns. The temperature in celsius for each test is mentioned in a column. Lastly mechanical properties including tensile strength, yield strength, elongation and reduction in area are given in separate columns. The 915 rows implies 915 set of unique entries. The predicting features in the dataset are Yield strength as 0.2% Proof Stress in (MPa), Tensile Strength in (MPa), Elongation in (%), Reduction in Area in (%). Their are in total 95 types of alloys, all coded with unique names.

```
df= pd.read_csv("../input/mechanical-properties-of-low-alloy-steels/MatNavi Mechanical properties of low-alloy steels.csv")
```

```
#first 10 rows of the dataset
df.head(10)
```

	Alloy code	C	Si	Mn	P	S	Ni	Cr	Mo	Cu	V	Al	N	Ceq	Nb + Ta	Temperature (°C)	0.2% Proof Stress (MPa)	Tensile Strength (MPa)	Elongation (%)	Reduction in Area (%)
0	MBB	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	27	342	490	30	71
1	MBB	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	100	338	454	27	72
2	MBB	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	200	337	465	23	69
3	MBB	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	300	346	495	21	70
4	MBB	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	400	316	489	26	79
5	MBB	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	450	287	461	25	81
6	MBB	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	500	274	431	28	85
7	MBB	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	550	262	387	32	87
8	MBB	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	600	220	314	42	88
9	MBB	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	650	152	226	59	92

[6]:
```
#checking number of rows and columns
df.shape
```

[6]: (915, 20)

III.6 Theory :

III.6.1 Python: Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. Python is a high-level, interpreted, interactive and object-oriented
scripting language. Python is designed to be highly readable. It uses English keywords frequently whereas other languages use punctuation, and it has fewer syntactical constructions than other languages.

III.6.2 Numpy : NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random sitmulation and much more.

III.6.3 Pandas : Pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.

III.6.4 Matplotlib : Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

III.6.5 Seaborn : Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.
For a brief introduction to the ideas behind the library, you can read the introductory notes or the paper.

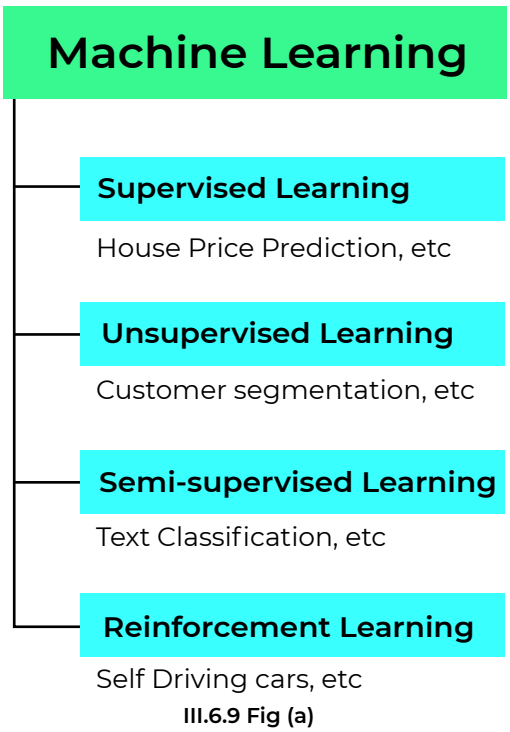
III.6.6 Plotly : The Plotly Python library is an interactive open-source library. This can be a helpful tool for data visualization and understanding the data simply and easily. Plotly graph objects are a high-level interface to plotly which are easy to use. It can plot various types of graphs and charts like scatter plots, line charts, bar charts, box plots, histograms, pie charts, etc.

III.6.7 Scikit Learn : Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python.

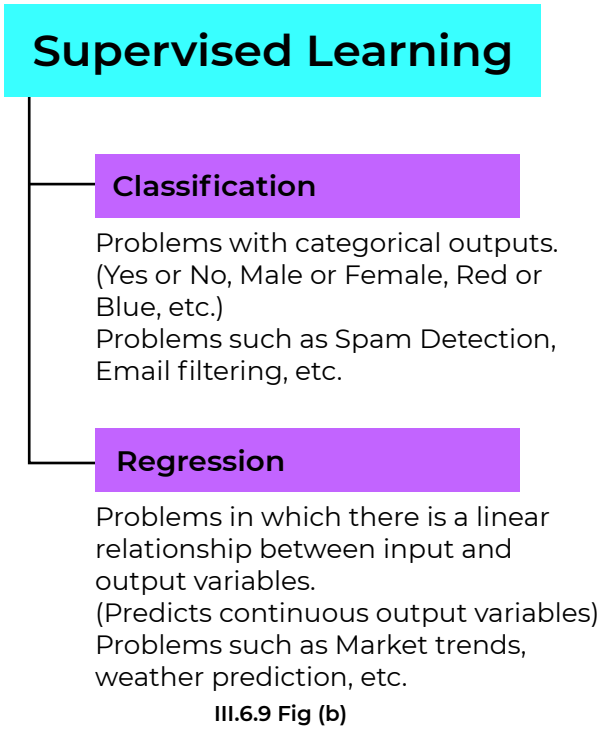
III.6.8 Kaggle Notebooks : Kaggle is a Platform where one can perform multiple hands On towards Data Science and ML problems. Kaggle notebook is a cloud computed notebook, where all your code/processes are computed on their cloud servers.

III.6.9 Machine Learning Algorithms : Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. A machine learning algorithm is the method by which the AI system conducts its task, generally predicting output values from given input data.

There are four types of machine learning algorithms:
> Supervised learning,
> Semi-supervised learning,
> Unsupervised learning, and
> Reinforcement learning.



III.6.9 Fig (a)



III.6.9 Fig (b)

Here in this case study we will be dealing with supervise learning. The two main types of supervised machine learning algorithms are classification and regression and here we will use the regression algorithms of Supervised learning to predict the mechanical properties of alloys.

IV Regression Algorithms :

IV.1 Simple Linear Regression Algorithm : It is one of the most-used regression algorithms in Machine Learning. It learns a model based on a training dataset to make predictions about unknown or future data. It is represented as y = b*x + c. This regression technique finds out a linear relationship between a dependent variable and the other given independent variables. Hence, the name of this algorithm is Linear Regression.

IV.2 Decision Tree : It is widely used algorithm for non-linear regression in Machine Learning. They are good at capturing non-linear interaction between the features (independent features) and the target variable (dependent features). The main function of the decision tree regression algorithm is to split the dataset into smaller sets.

IV.3 Random Forest Regressor : Unlike decision tree regression (single tree), a random forest uses multiple decision trees for predicting the output. It is a Supervised Learning algorithm used for classification and regression. The input data is passed through multiple decision trees. Since there are multiple decision trees, multiple output values will be predicted via a random forest algorithm. For the final output of new datas we have to find the average of all the predicted values. Random Forest algorithm requires more input in terms of training because the large number of decision trees mapped under this algorithm requires more computational power.

IV.4 KNN : It is popularly used for non-linear regression in Machine Learning. The new data point is compared to the existing categories and is placed under a relatable category. The average value of the k nearest neighbors is taken as the input in this algorithm.

V Data Preprocessing / Data Cleaning :

Data cleaning is the process of removing incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset to improve the accuracy while model building. “Dirty” data does not produce the accurate and good results. So it becomes very important to handle this data. Professionals spend a lot of their time on this step.

In our dataset we have done label encoding to change the categorical values of column with “alloy code” into numerical values as for building ML regression model the feature values in our dataset should be numerical values only.

For this we used LabelEncoder from sklearn.preprocessing package of sklearn library to encode the alloy codes.

[19]:
```
#Seperating dependent and independent features for model building
X=df3.iloc[:,1:-4]
y=df3.iloc[:,-4:]
```

V. Fig (a) - Splitting the dataset

```
y.head()
```

[20]:

	0.2% Proof Stress (MPa)	Tensile Strength (MPa)	Elongation (%)	Reduction in Area (%)
0	342	490	30	71
1	338	454	27	72
2	337	465	23	69
3	346	495	21	70
4	316	489	26	79

V. Fig (b) - Target Features

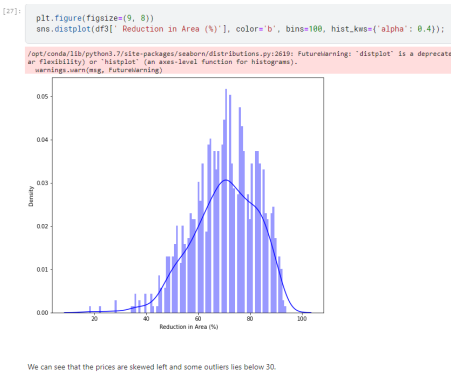
[21]:
```
X.head()
```

[21]:

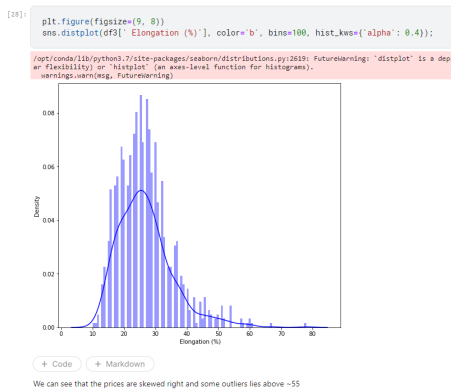
	C	Si	Mn	P	S	Ni	Cr	Mo	Cu	V	Al	N	Ceq	Nb + Ta	Temperature (°C)
0	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	27
1	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	100
2	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	200
3	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	300
4	0.12	0.36	0.52	0.009	0.003	0.089	0.97	0.61	0.04	0.0	0.003	0.0066	0.0	0.0	400

V. Fig (c) - Independent Features

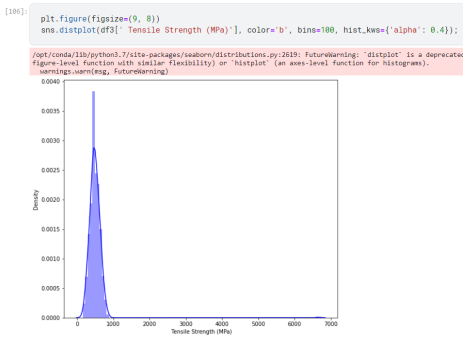
VI Dataset Visualization : We check for the outliers in our dataset as outliers may lead to bad training of our model. If we have many outliers we must do something to remove them.



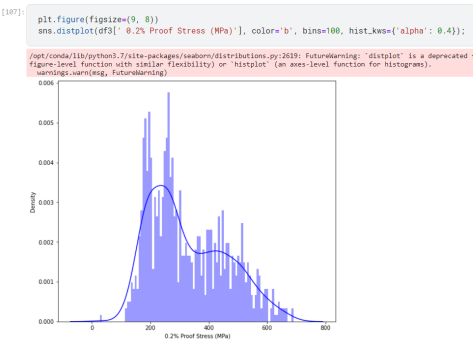
VI. Fig (a)



VI. Fig (b)



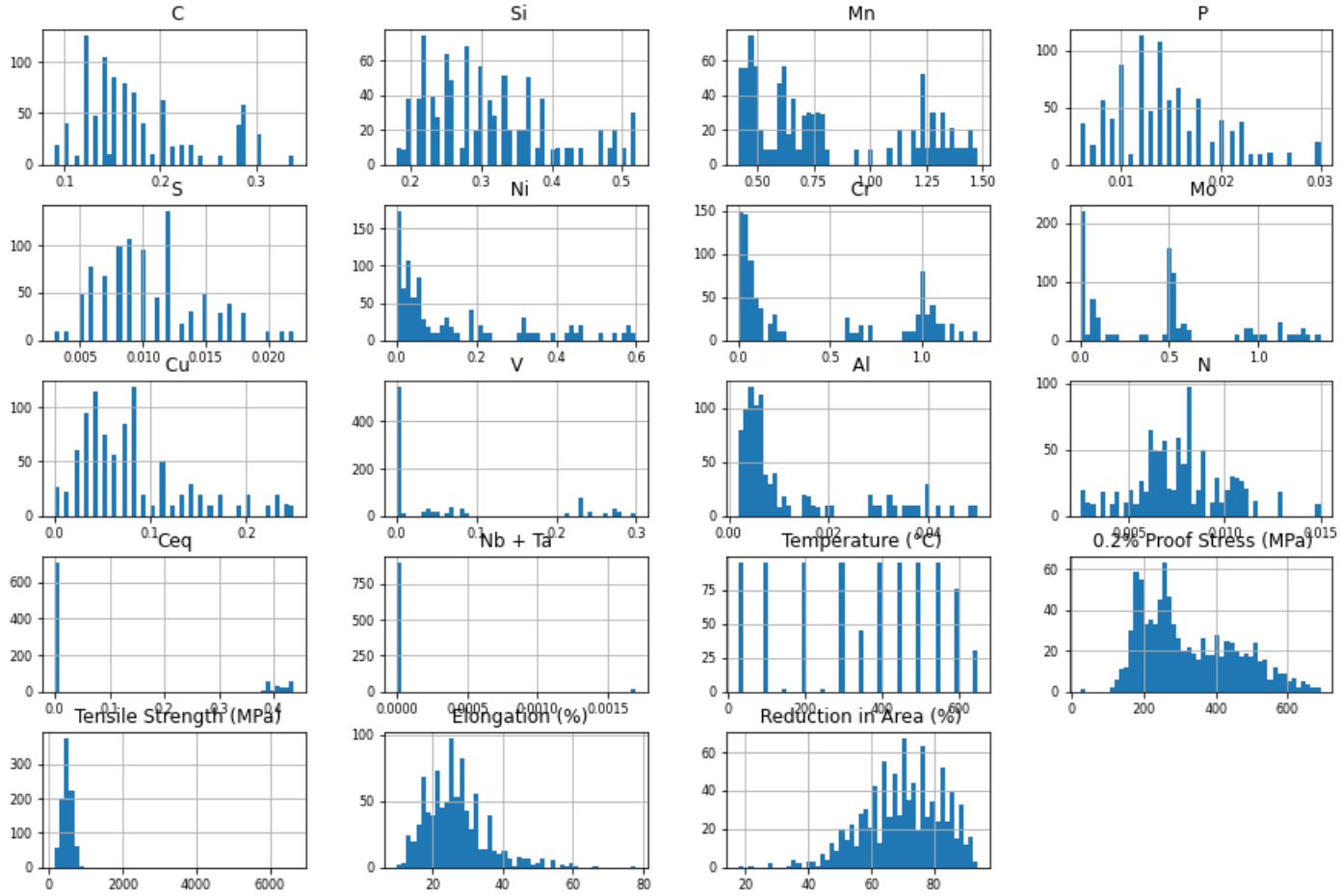
VI. Fig (c)



VI. Fig (d)

VI.1 Hist Plot

```
[116]: #visualizing the numerical values of our dataset
       num_data.hist(figsize=(15, 15), bins=50, xlabelsize=8, ylabelsize=8);
```



+ Code + Markdown

As we already know from theoritical aspect, the elongation and reduction in area is inversely distributed.

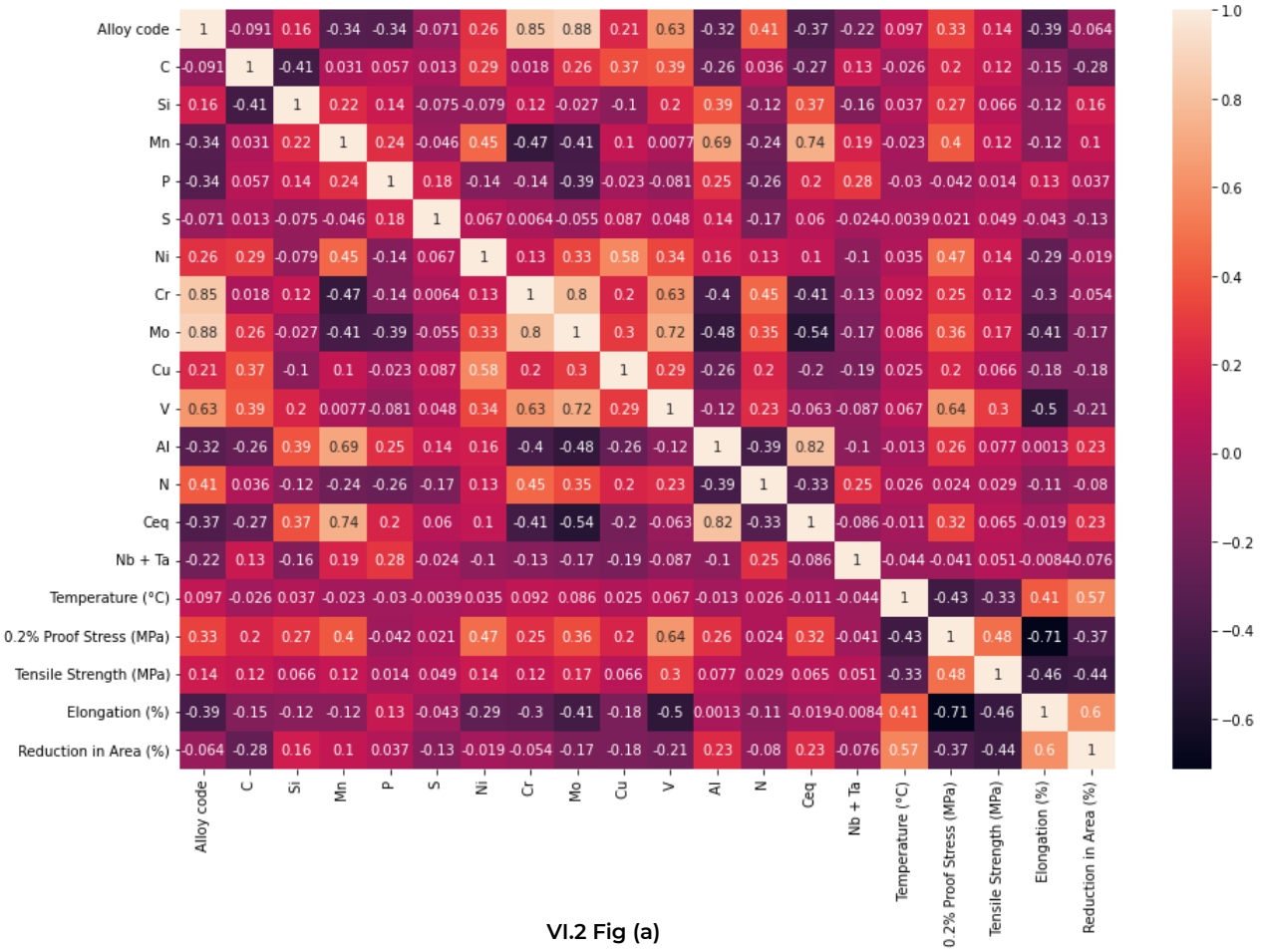
VI.1 Fig (a)

Figiure VI.1 Fig (a), represents the frequency of numerical data using rectangles. The height of a rectangle (the vertical axis) represents the distribution frequency of the variables (the amount, or how often that variable appears) of our dataset. This shows how frequently every value in our data set occurs in a relatively unbiased way.

VI.2 Correlation Matrix

```
corelation = df3.corr()
plt.subplots(figsize=(15,10))
sns.heatmap(corelation, xticklabels=corelation.columns, yticklabels=corelation.columns, annot=True)
```

[24]: <AxesSubplot:>



VI.2 Fig (a)

A heatmap is a graphical representation of a correlation matrix representing the correlation between different variables. The value of correlation can take any value from -1 to 1.
A correlation matrix is simply a table which displays the correlation coefficients for different variables in our dataset. The matrix depicts the correlation between all the possible pairs of values in the dataset. It is a powerful tool to summarize a large dataset and to identify and visualize patterns in the given data.

Here through this correlation matrix, in figure VI.2 Fig. (a), we can coclude that -

1) Temperature has high effect on % Elongation and % Reduction in area.
2) 0.2% Proof Strength is more corelated with presence of V, Ni, Mn, Mo, and Ceq.
3) Tensile Strength is highly related by the presence of V and moderately influenced by presence of Mo, Ni, Cr, C and Mn.
4) Tensile Strength is also highly related to 0.2% Proof Strength.
5) % Elongation is moderately influenced by presence of P.
6) % Reduction in Area is highly influenced by temperature and %Elongation.

VII Model Building : We use Regression Algorithms to build our model because variables in our dataset contains continuous values and for such type of problems regression machine learning algorithms are the most suitable choice. We begin by acalig our dataset in order to make every value in our dataset in a common scale. Scaling helps to increase the accuracy and reduces the loss error. We scale our data using Standard Scaler by importing it from the preprocessing package of sklearn library.

Standard Scaler resizes the distribution of the values so that the mean of the observed values is 0 and the standard deviation is 1. Here we apply standardization instead of normalization (MinMaxScaler) because standardized data is usually preferred when the data is being used for multivariate analysis i.e. when we want all the variables of comparable units. It is usually applied when the data has a bell curve i.e. it has gaussian distribution. Hence here the standardization is much more effective than Normalization.

[32]:
```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
y = sc.fit_transform(y)
X = sc.fit_transform(X)
```

[33]:
```
#splitting our dataset for model building
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,test_size=0.2, random_state=42)
print(X_train.shape, X_test.shape, y_train.shape, y_test.shape)
```

```
(732, 15) (183, 15) (732, 4) (183, 4)
```

We split our data into training and testing sets, namely, X_train, X_test, y_train, y_test .
Separating data into training and testing sets is an important part of evaluating our machine learning models. When we separate a data set into a training set and testing set, most of the data is used for training, and a smaller portion of the data is seperated which is used for the purpose of testing. This is done using the train-test split procedure which is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to train the model (test dataset).

Train Dataset :
Used to fit the machine learning model.

Test Dataset :
Used to evaluate the fit machine learning model.

After training the X_train and y_train dataset using different Regression machine learning models, it is tested using the X_test dataset and then the predicted result is compared with the y_test dataset to get the accuracy for our trained model.

We train our model using the four most popularly used regression machine learning algorithms which have been discussed in the *section IV* of this paper i.e.
1) Linear Regression
2) Decision Tree Regressor
3) Random Forest Regressor
4) KNearest Neighbour Regressor

VIII Cnclusion : The highest accuracy received was from Random Forest Rgressor of about 0.897% accuracy with 0.0796 as the mean squared error and 0.174 as the mean absolute error.

Mean Squared Error is used as a default metric for evaluation of the performance of most regression algorithms. It is calculated by the sum of square of errors in prediction that is actual output minus predicted output (here, y_test is the actual output) and then divide by the number of data points. It gives us an absolute number on how much our predicted results deviate from the actual number.

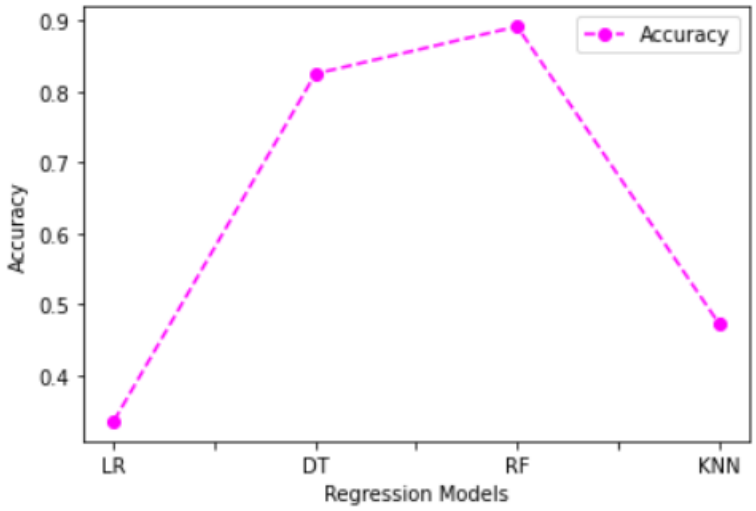
Mean Absolute Error is the average of the magnitude of the difference between the prediction of an observation and the true value of that observation. It measures the accuracy for the continuous variables.

Hence, the Random Forest Regression machine learning model makes an ideal choice for the prediction of mechanical properties of low-alloy steels with R2 score of 89.7% which is significantly greater than R^2 score of other popularly known regression models.

[113...

	Model	Accuracy
0	LR	0.334844
1	DT	0.824939
2	RF	0.891674
3	KNN	0.472835

VIII Fig (a)



VIII Fig (b)

References (Blogs & Papers) :

1. Mechanical Properties of Materials: Definition, Testing and Application (https://www.researchgate.net/publication/344287957_Mechanical_Properties_of_Materials_Definition_Testing_and_Application)
2. https://www.nde-ed.org/Physics/Materials/Mechanical/Mechanical.xhtml
3. Machine learning prediction of the mechanical properties of γ-TiAl alloys produced using random forest regression model (https://www.sciencedirect.com/science/article/pii/S2238785422002800)
4. What is an Alloy? - Mead Metals (https://www.meadmetals.com/blog/what-is-an-alloy)
5. https://www.sciencedirect.com/topics/engineering/material-selection-problem
6. Materials discovery and design using machine learning (https://www.sciencedirect.com/science/article/pii/S2352847817300515#:~:text=Machine%20learning%20provides%20a%20new,performing%20other%20materials%2Drelateds%20studies.)
7. Artificial intelligence and machine learning in design of mechanical materials (https://pubs.rsc.org/en/content/articlehtml/2021/mh/d0mh01451f)
8. Documentations of Pandas , Numpy, Seaborn, Matplotlib, Plotly, Scikit Learn, Kaggle.
9. https://www.techtarget.com/whatis/definition/machine-learning-algorithm
10. https://www.jigsawacademy.com/popular-regression-algorithms-ml/
11. https://www.analyticsvidhya.com/blog/2021/05/5-regression-algorithms-you-should-know-introductory-guide/
12. Towardsdatssience (https://towardsdatascience.com/data-preprocessing-e2b0bed4c7fb)
13. Vidyaanalytics (https://www.analyticsvidhya.com/blog/2021/05/feature-engineering-how-to-detect-and-remove-outliers-with-python-code/)
14. Becominghuman.ai (https://becominghuman.ai/what-does-feature-scaling-mean-when-to-normalize-data-and-when-to-standardize-data-c3de654405ed)