Automated Fabric Color Prediction

by

Fawzia Muhtashima
19101204
Fawzia Maksurah
19101167

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
May 2023

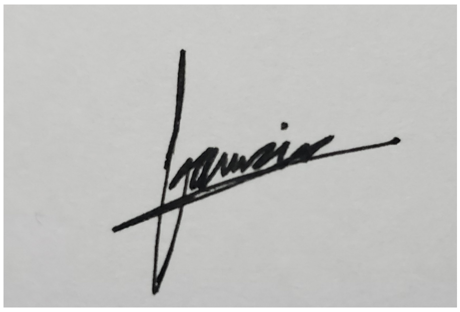
Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Fawzia Muhtashima
19101204



Fawzia Maksurah
19101167

Approval

The thesis titled "Automated Fabric Color Prediction" submitted by

1. Fawzia Muhtashima(19101204)
2. Fawzia Maksurah(19101167)

Of Spring, 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May 29, 2023.

Examining Committee:

Supervisor:
(Member)

Sifat

Sifat E Jahan
Lecturer
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Moin Mostakim

Moin Mostakim
Senior Lecturer
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:

Dr. Md. Golam Rabiul Alam, PhD
Thesis Coordinator and Professor
Department of Computer Science and Engineering
Brac University

Abstract

This paper focuses on addressing some challenges faced by colorists and explores various approaches to predict fabric color changes after dyeing processes. It emphasizes the importance of color prediction in the textile industry and proposes suitable models that can effectively carry out color prediction tasks based on given recipes. By implementing such predictive models, the textile industry can improve efficiency, reduce labor-intensive practices, and enhance the overall quality control process.
The methods used in this study are supervised machine learning techniques, including multiple linear regression, decision tree, random forest, and neural network. Among these models, the most appropriate one is selected and further optimized using feature engineering techniques to improve accuracy. **Keywords:** Machine

Learning; Color Prediction; Decision tree; Linear Regression; Neural network; Feature Engineering

Table of Contents

List of Figures

Introduction

The production of textile materials at large scale has played a significant role in the rapid economic growth and development of Bangladesh as textile exports contribute around roughly 20 percent of the country's GDP. Despite this huge success, the fabric production industry faces challenges in maintaining control over fabric color quality and accurately predicting color changes during the dyeing process. These challenges add workload and inconsistency in recipe preparation. Traditionally, determining the right amount of dye for achieving the desired sample color or creating dye recipes has relied on time-consuming and imprecise methods such as visual inspection, trial and error, and expert guesswork. These methods are physically and mentally demanding and often lead to inaccuracies.

1.1 Introduction to fabric color

Color is without a doubt one of the most essential textile properties. Even though the fabric material is outstanding, a lack of good color may still lead to a lackluster sale [4]. Many businesses still utilize visual methods for production today, which not only call for thorough training and grading of employees based on their vision but also accounts for the light source and the appropriate surroundings. Therefore, the majority of businesses still rely on colorists' skill when matching colors, and they still prepare color recipes manually.

1.2 What is color-matching?

Initially the colorists look through their inventory for examples of related colors to replicate the match of a certain piece of fabric ordered by consumers. The closest sample is chosen, and its concentrations are then adjusted based on colorist's experience to match the desired color. If a test print is unsatisfactory, the concentration is adjusted before running another test print. Due to the time and expense involved in this trial-and-error process, current approaches for its improvement were looked at.

1.3 Technical Approaches

The kubella munk theory[5] was one of the widely used color-matching techniques for dyeing processes. However this method relies on the precise fulfillment of dyeing conditions and calls for accurate knowledge of dyeing physics. Since it is highly challenging to regulate every process parameter in textile color printing, different approaches to recipe calculation were looked into.

1.4 Objectives

Finding the right amount of each dye in a dyeing recipe to achieve the intended sample color is the central objective of dyeing recipe prediction. It takes a lot of skill and patience to look through the entire collection of color cards in order to choose the sample that matches the color the best [13].Sometimes it becomes clear through the trial-and-error approach (previously outlined) that the combination of dyes chosen is not suitable for producing the desired color. Therefore one of the approaches taken is to use the artificial neural network to calculate recipe parameters like the concentration of each dye in the printing paste [11].

Problem Statement

2.1 The problem with recipe preparation

To achieve a desired color on the fabric, the colorists require a ”dyeing recipe” which is a set of appropriate dyes or pigments when applied at a specific concentration to the fabric in a particular way will render the required color [8]. In other words, the word “dye recipe” corresponds to the same meaning as the “recipe for any dish”. The process in which the colorists try to determine the recipe for the required color is known as recipe preparation and it is an ongoing quest for colorists to find the appropriate recipe for each unique color. The appropriateness of the recipe depends on various factors which include temperature, ph, type of fabric, coloring agents used for the dye combinations for the color mixture. An expert must take into account most of these criteria when determining a recipe for that color.

2.2 The aim of research

When trying to match colors for textiles, a computer needs to be taught the proper information regarding the coloristic properties of different dyes. This is done by making and testing dye mixtures and recording the results, which can be used to match colors in the future.Since the goal is to automate the color matching process to save time and resources, a database of dye mixtures is needed to train machine learning models to generalize this data and hence, accurately match colors. Using supervised learning approaches we can train various models and evaluate its performance base on the color difference(in CIELAB units) between sampled color and predicted color.

Literature Review

3.1 Color-matching

In [11], The Levenberg–Marquardt (LM) optimized backpropagation (BP) neural network is employed to analyze color changes in fabric during the fabric printing process. Initially, the fabric is printed and converted to RGB format, which is fed into the BP-NN model to obtain its corresponding CIELAB color coordinates. Subsequently, the color of the printed fabric is directly measured using a spectrophotometer, allowing for a comparison with the model's output. The accuracy of this model was 93%, indicating that neural networks excel at determining color characteristics when provided with various types of input data.
In [9], multiple linear regression model was introduced to solve the color mismatch problems during the digital printing process.100 random color samples were scanned, printed and converted to RGB format, normalized and compared with the measurement of spectrophotometer under D65 illuminant. The accuracy score was more than 90% which showed a good correlation between scanned/printed fabric color and original fabric color represented in CIELAB color space.
According to [10] It is advisable to predict reflectance rather than tristimulus values(required to calculate CIELAB coordinates) directly at the output for two primary reasons. First, direct prediction of XYZ values is unnecessary since XYZ can be easily calculated from reflectance. By avoiding direct XYZ prediction, the network's computational workload is reduced. Second, tristimulus specifications are illuminant specific, whereas reflectance values allow for the computation of tristimulus specifications under any illuminant.

In [14], a hybrid model combining Stearns-Noechal equation and neural network was introduced to detect color error and it obtained a better result when compared to both the SN model and ANN model with 1 hidden layer with ANN performing worse than SN as it was trained with only 60 samples.

3.2 Recipe calculation

In [6], an artificial neural network-based approach is used to predict the recipe for carded fiber. Typically, a recipe involves mixing dyes to obtain a color but here, the final color is obtained by mixing colored fibers in different compositions which are then carded together to make the carded fiber with a new color. The ANN-based

system is used to predict reflectance factors of the differently colored fibers that are to be carded to make the fiber. The first approach involves only the input of the same material of fibers with different colors. The second approach is the prediction of reflectance values(i.e. Colors in the form of numerical values) of colored fibers made of different materials. The two approaches were used to predict the reflectance factors of 500 different blends of fiber and eliminate the bias that can be caused due to differences in fiber material.
[16] utilizes fabric type and target colorimetric information from a DCS, such as K/S spectra and RGB values, as inputs to predict dye concentrations. The dyeing recipe recommendation system follows a modular architecture, with individual modules consisting of multiple regression models. Each model is trained for a specific fabric type and a single dye combination set
In [2], an effort was made to examine the use of an ANN in the color prediction task to map the non-linear relationship between dye concentrations and color parameters X, Y, and Z. Its performance was compared with the conventional K-M model, and the results showed that the ANN performed better at accurately predicting tristimulus values.

Hemingray and Westland [12] compared two conventional prediction models (Stearns-Noechel and Friele) with two neural network models for predicting fibre blend color. They found that the neural network models outperformed the conventional models in terms of accuracy and consistency. Additionally, the neural network models were able to predict the color of blends with a greater number of primary colors than the conventional models.
In [1] genetic algorithm was used to predict dye combination sets where the set of dyes were population and optimized towards lowest color difference scores.
[15] used Linear programming optimization to find the appropriate dyes to mix and their exact concentrations to produce the required color. Here, The objective function and all the constraints of the model are expressed linearly according to the decision variables. The objective function is to minimize the differences between the K / S spectra of the sample dyed with the proposed mixture and the target color. The constraints of the model are formulated based on Kubelka-Munk and Duncan theories in the same way as those of the single-constant model.

In a previous study [3], colored fibers were blended together to obtain a uniform color, and the reflectance values of pre-colored fibers and blended fibers were measured using an expensive spectrophotometer. The data were used to train neural networks to predict the final color coordinates. Compared to conventional prediction models(Stearns-Noechal, Friele, Kubelka-Munk), neural networks performed better [12].
The most common neural network model for color applications is the multi-layer perceptron feed-forward network, as per [12] where he quoted different sizes of neural networks that were applied for predicting the reflectance of colored sample. In a similar vein, [1] employed neural networks to predict dye concentrations from CIELAB coordinates.
[12] developed a neural network to predict spectral reflectance for six printing inks mixed on a white card. The neural network's performance surpassed that of a Kubelka-Munk model when a hidden layer containing seven units and all the 273

available training data were employed, resulting in a CIELAB color difference of 1 unit for 60 testing samples.

These reviews were focused on investigations conducted to improve the accuracy of color recipe prediction for industrial applications, specifically for colored fabric or fiber blends. The emphasis was not placed on diverse textile materials like nylon or cotton, but rather on ensuring consistent color reproduction in repetitive orders placed by clients, even when those orders are made at different times.

Workplan

Our objective is to identify a suitable model capable of predicting the resulting color of a finished fabric product based on a given set of dyes or DCS from our recipe.Out of a total of 150 samples, we utilize 144 training samples to train regression models, classifiers (Decision Tree and Random Forest), and a feed-forward neural network (specifically, a multi-layer perceptron) in order to establish the relationship between dyes and color. The performance of these models is evaluated by comparing the color differences between the tested color and the predicted color and select the most appropriate model for further optimization. We aim to enhance the learning capabilities through feature engineering. Using the obtained feature importance scores obtained from Random Forest classifier, we will introduce more features as a function of their importance_scores and feed them as well to our best model. The accompanying flowchart illustrates the sequential steps involved in constructing an appropriate model.

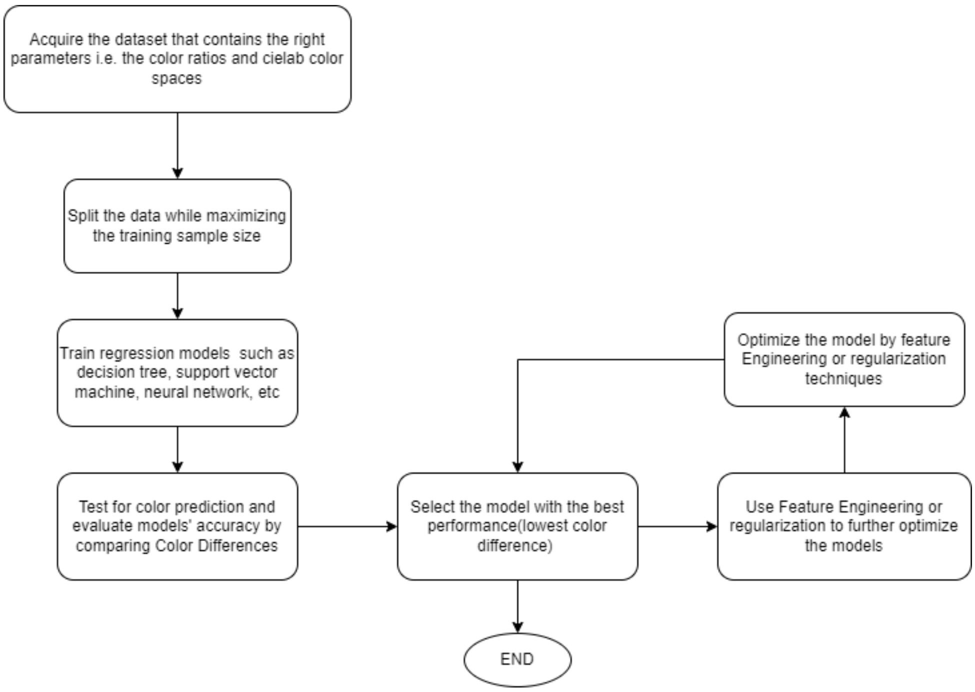


Figure 4.1: Flowchart of Workplan

4.1 Inference from scatter graphs

To better understand the complex relationship between dyes and their blend. We plotted scatter diagrams where we represented each color against its CIELAB color coordinates (L*A*B values where L represents Lightness/Greyness scales, A represents blueness/greenness scales and B represents yellowness/blueness scales in CIELAB color space).Three mulitple linear regression models are trained to better understand the trend in these graphs where the dependant variables are the three dye ratios and the independant variable are one of the scales in L, A, B each plotted separately in y-axis. The sampled coordinates and the predicted coordinates including the color mismatch metric(represented as CIELAB color difference or $E_{a,b}$) are shown in results table 6.2. The MSE (mean squared error) below 4.1 for the 'a' color coordinate in which represents redness/greenness contrast in the CIELAB color space is unacceptably large. This indicates that the linear regression is not a suitable approach for such prediction tasks. 3 of the graphs resemble a linear relationship (%Magenta vs L, %Magenta vs A, %Yellow vs B) but the rest show a non-linear relationship.Some of the points are heavily concentrated and this is because of the nature of our dataset. Since the features(different colors) interact in a complicate manner to form the blend color, it is their collective influence that determines the output coordinates, L, A, B.

Model	MSE	STD Error
Yellow:Magenta:Cyan vs L	1.47	1.21
Yellow:Magenta:Cyan vs a	7.02	2.65
Yellow:Magenta:Cyan vs b	2.27	1.51

Table 4.1: Chart showing MSEs of 3-MLR model

4.2 Scatter plots

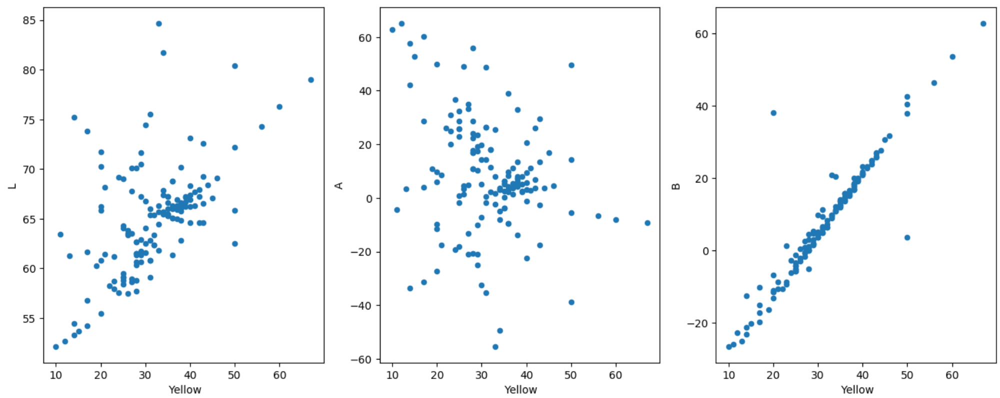


Figure 4.2: Scatter Graph showing percentage Yellow vs final color in CIELAB color spaces scales, %Yellow vs L, %Yellow vs A, %Yellow vs B

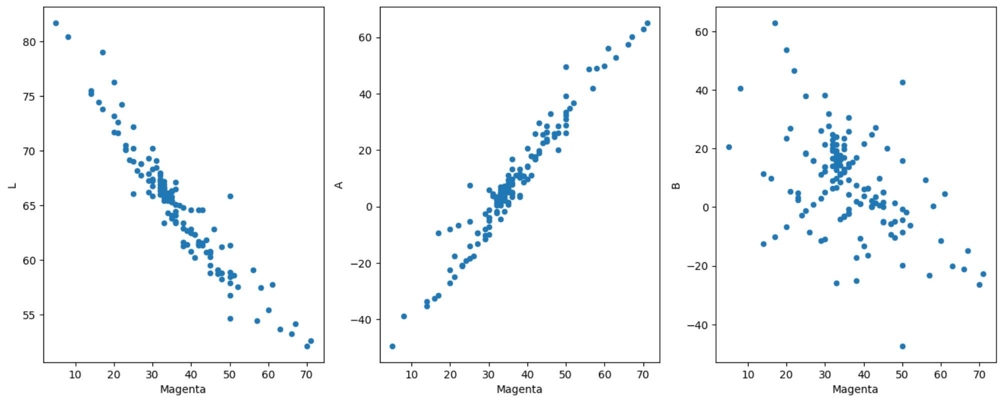


Figure 4.3: Scatter Graph showing percentage Magenta vs final color in CIELAB color spaces scales, %Magenta vs L, %Magenta vs A, %Magenta vs B

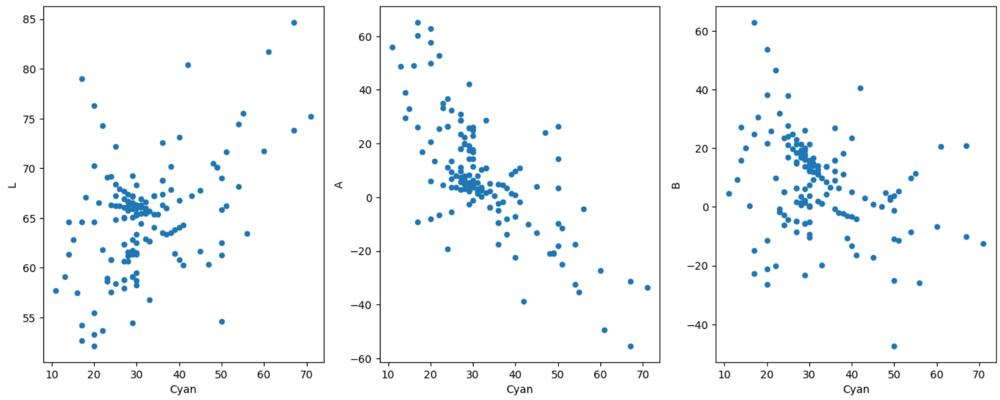


Figure 4.4: Scatter Graph showing percentage Cyan vs final color in CIELAB color spaces scales, %Cyan vs L, %Cyan vs A, %Cyan vs B

Description of data

5.1 Feature Selection

The dataset which we acquired contains 150 samples of blended color and the ratio of colorants used in percentage.It is preprocessed to select only the features required for input and output data. From the 150 samples, 146 training samples are chosen and 4 testing samples are selected as we want to maximize the training set size for the system to learn more correctly.The structure of our raw Dataset is shown Figure 5.1 and the features selected for input and output are marked with red rectangle and blue rectangle accordingly.

Yellow	Magenta	Cyan	Blend	L	A	B
	50	50	Moody Blue	54.64	26.23	-47.33
10	70	20	Mulberry	52.16	62.9	-26.46
11	33	56	Glacier	63.41	-4.38	-25.83
12	71	17	Cerise	52.66	65.18	-22.66
13	38	50	Ship Cove	61.28	3.5	-25.1
14	66	20	Mulberry	53.28	57.59	-21.23
14	57	29	Tapestry	54.44	42.01	-23.2
14	14	71	Shakespeare	75.22	-33.58	-12.45
15	63	22	Mulberry	53.67	52.8	-20.11
17	67	17	Cranberry	54.21	60.2	-14.97

Figure 5.1: Dataeset with input features and output features

5.2 Preliminary Data Analysis

Since we are training regression models, to predict the fabric's resultant color, the final color or output data must be represented in colorimetric information, such as reflectance values obtained from a spectrophotometer, CIE values, or sRGBs. It is important to note that the input features must contain information about the color recipe, whether in terms of concentration or ratios, to ensure accurate predictions.For this research we are not going through all the conditions in color recipe, except the proportions of colorful dyes required to render that particular color in fabric. This approach helps eliminate biases forms due to variance in parameters. To achieve this, the dataset needs to undergo preprocessing to select the relevant features.
The most commonly used metric for quantifying color differences in CIELAB color space is Known as CIELAB color difference or ΔE , which represents the overall

color distance between two colors. A ΔE value of 0 indicates a perfect color match, while larger values represent increasing color differences. The scales in CIE-L*a*b are:

L* (Lightness): Range: 0 to 100, 0 represents pure black, while 100 represents pure white. Values in between represent various levels of lightness, with 50 being a neutral gray.

a* (Red-Green Axis): Range: -128 to 127 Negative values represent green hues, with -128 being a strong green. Positive values represent red hues, with 127 being a strong red. Zero indicates a neutral gray along the red-green axis.

b* (Blue-Yellow Axis): Range: -128 to 127 Negative values represent blue hues, with -128 being a strong blue. Positive values represent yellow hues, with 127 being a strong yellow. Zero indicates a neutral gray along the blue-yellow axis.

The smaller the CIELAB color difference(aka error), the closer the blended color is to the target color. According to [7], if the ΔE value is equal to or less than 2.3 CIELAB units, then the difference of color between the sample and target color is negligible and will not be perceived by human vision under normal lighting comditions. The perception of color differences can vary among individuals, and there is no universally agreed-upon threshold for the smallest CIELAB color difference that would be imperceptible to all humans. However, research and standards have been established to estimate color differences that are typically considered below the threshold of human perception. However, it's important to note that this threshold can vary depending on factors such as the specific colors being compared, the lighting conditions, and the observer's individual color acuity. Thus, it has been widely used to test accuracy for color-matching. Therefore, our proposed model should work towards minimizing the value of ΔE.

If the sample color has $L_s * a_s * b_s$ and blended color has $L_b * a_b * b_b$ then the ΔEa,b or color difference can be calculated using,

$$\Delta L = L_s - L_b$$
$$\Delta a = a_s - a_b$$
$$\Delta b = b_s - b_b$$
$$\Delta E = \sqrt{\Delta L^2 + \Delta^2 + \Delta b^2}$$

To understand how much the color differs in human perception, the following figures 5.2, 5.3, 5.4, 5.5 shows the main colors and their CIElab color coodinates L*a*b* in numbers.

Figure 5.2: Colors with CIELAB color difference 3.81 CIEALB units

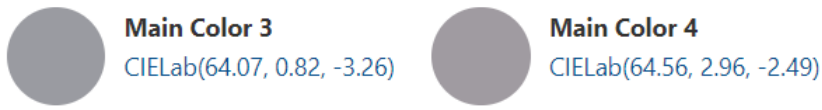


Figure 5.3: Colors with CIELAB color difference 2.34 CIELAB units

Figure 5.4: Colors with CIELAB color difference 0.61 CIELAB units

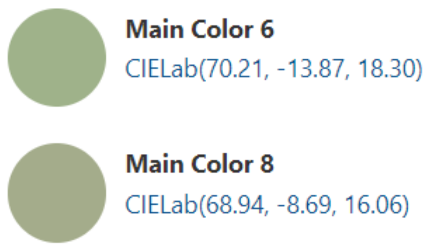


Figure 5.5: Colors with CIELAB color difference 5.78 CIELAB units

5.3 Models implemented

To find the appropriate model, we trained Decision tree (**DT**), Random Forest (**RF**), a multiple-linear regression (**MLR-3**) which ran three times for 3 output variables L*, A*, B* and tuned it with L1 and L2 regularization, a Support Vector Regression (**SVR**), a neural network (**NN**) and a Random Forest based Neural Network (**RF-NN**) that was implemented with an augmented dataset.

5.4 Desision Tree

In regression, decision tree works by building a tree recursively, by splitting the data based on selected features and thresholds, therefore aiming to minimize the variance or error in the predicted values within each leaf node. The predicted value for a new data point is determined by traversing down the tree, following the appropriate branches based on the feature values until a leaf node is reached. The predicted value at the leaf node represents the regression output.

5.5 Random Forest

Random Forest is an ensemble learning method that constructs multiple decision trees and combines their predictions. It can handle both categorical and numerical features, making it is still suitable for handling data that requires to map continuos input variables to continuos output variables. The Random Forest classifier can capture complex interactions between the color ratios and predict the corresponding color label.

5.6 Linear Regressions/Least-Squares

The goal is to minimize the difference between predicted and actual values of the dependent variable. Multiple linear regression provides insights into the collective influence of multiple independent variables on the dependent variable and is widely used for prediction, analysis, and understanding complex relationships. We trained three multiple linear regression models to analyze the trends depicted in the scatter plots.
Ridge regression and Lasso regression are both regularization techniques used in linear regression to handle multicollinearity and prevent overfitting to the training data by adding a 'bias' to the loss function. For Ridge, the bias is proportional to the square of the coefficients and for lasso, it is proportional to the absolute value of the coefficients. The proportionality constant otherwise λ can be any value from 0 to positive infinity. We varied this λ which is the regualrization parameter by employing GridSearch and 5-fold cross validation and ran the model three times to predict the three output variables.

To map a continuos multi-dimensional input to an output, we can use least squares, and improve their performance by adding biases which we do in lasso and ridge regression where the line is made to deviate slightly from best fit so it can match the testing set of data.

5.7 Neural Network

This model can effectively model complex relationships between inputs and outputs. By training a neural network on the CIELAB color spaces and percentages of colors in the dataset, it can learn patterns and correlations to predict the blend color accurately. The architecture of neural networks consists of multiple interconnected layers of processing units or neurons, combined with the iterative adjustment of weights. The loss is propagated backwards which enables the optimization of the model to minimize errors and achieve accurate results which regression models or other supervised learning fails to achieve. Unlike regression which tries to best fit a line or hyperplane to the data, the output of neurons is completely dependant on the values weights and biases which are repeatedly adjusted from the calculated loss. This help tweak the lines or hyperplane to completely fit into the non-linear relationship. Neural networks are almost completely free from outliers or anomalies which other regression types has to compromise.

To explicitly model nonlinear relationships, techniques such as polynomial regression (like support vector regression) or using nonlinear activation functions within neural networks are typically employed. These approaches introduce flexibility for the line to become curved and capture nonlinear patterns in the data.

5.8 Support Vector Regression

SVR is a regression technique that uses Support Vector Machines (SVM) for prediction. It finds a hyperplane that has the maximum margin while still fitting the data within a specified epsilon (tolerance) range. SVR can handle both linear and nonlinear regression problems through the use of different kernel functions. Linear kernel is chosen to predict the cielab values.The epsilon and C are varied using GridSearch and 5-fold cross validation.The C is the regularization parameter that controls the trade-off between fitting the training data and keeping the model simple to avoid overfitting.While the epsilon controls the margin of tolerance for the model's prediction errors. It specifies the width of the epsilon-insensitive zone, where errors within this zone are considered acceptable and do not contribute to the loss function during training.

5.9 Neural Network trained with augmented dataset based from Random Forest

In order to further reduce color differences, we employed feature engineering techniques to augment our dataset based on insights gained from training a Random Forest. We began by acquiring the feature importance scores for each input color, specifically Yellow, Magenta, and Cyan from the Random Forest which was first implemented with the original dataset. The obtained feature importance scores for the colors Yellow, Magenta, Cyan are 0.182, 0.622, and 0.196, respectively.

To augment our dataset with this imformation, we introduced three new features: Yellow_impact, Magenta_impact, and Cyan_impact. These features are created by multiplying the ratios of each features with their corresponding importance scores acquired from Random Forest. By including these newly derived features, we aim to capture the weighted influence of each color component, thereby enhancing the model's ability to minimize color differences effectively.The model's flowchart is shown in 5.6 and dataset before and after Feature Engineering are shown in 5.7 and 5.8.

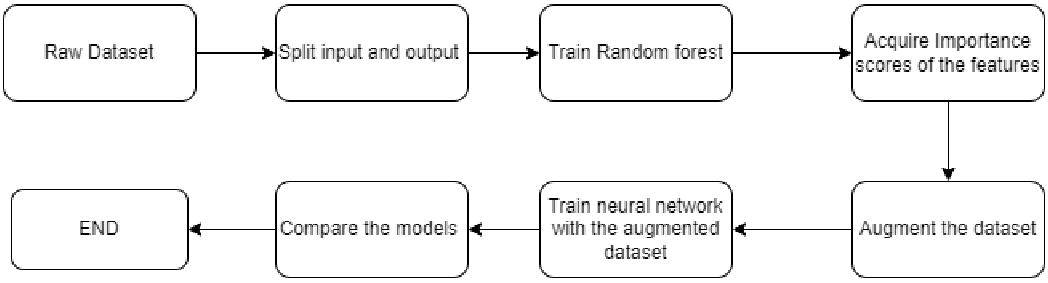


Figure 5.6: Workflow of the Random Forest based Neural Network

	Yellow	Magenta	Cyan	Blend	L	A	B
0	50.0	NaN	50.0	Pastel Green	62.49	14.44	3.75
1	33.0	33.0	33.0	Napa	65.71	1.63	9.94
2	NaN	50.0	50.0	Moody Blue	54.64	26.23	-47.33
3	50.0	50.0	NaN	Coral	65.85	49.53	42.70
4	50.0	25.0	25.0	Gimblet	72.22	-5.35	37.87
...	...	...	...	...	...	...	...

Figure 5.7: Dataset before Feature Engineering

	Yellow	Magenta	Cyan	Yellow_impact	Magenta_impact	Cyan_impact	Blend	L	A	B
0	50.0	0.0	50.0	9.121925	0.000000	9.798367	Pastel Green	62.49	14.44	3.75
1	33.0	33.0	33.0	6.020471	20.512607	6.466922	Napa	65.71	1.63	9.94
2	0.0	50.0	50.0	0.000000	31.079708	9.798367	Moody Blue	54.64	26.23	-47.33
3	50.0	50.0	0.0	9.121925	31.079708	0.000000	Coral	65.85	49.53	42.70
4	50.0	25.0	25.0	9.121925	15.539854	4.899183	Gimblet	72.22	-5.35	37.87

Figure 5.8: Dataset after Feature Engineering

Model Overview

6.1 Design of the neural network architecture

Throughout our study we found that neural networks are quite adept at handling diverse data types, such as categorical, image(in RGB format), or numeric(cielab, cmyk, reflectance, etc), makes it the preferred choice for numerous prediction tasks. Neural networks have been increasingly utilized to model color changes that occur in various industrial processes, such as spinning fibres or heating fabric, where the original color may undergo significant alterations. Our study employed a 3-16-16-3 multi-layer perceptron neural network, using a rectifier linear unit for the dense layer's activation function and linear for the output layer, with RMSprop(Root Mean Square propagation) as the optimizer.The code snippet below highlights the model's straightforward construction. After training the model for 500 epochs, the mean squared error was reduced to 0.85.

```
model = tf.keras.models.Sequential([
    tf.keras.layers.Dense(16, activation='relu', input_shape=(3,)),
    tf.keras.layers.Dense(16, activation='relu'),
    tf.keras.layers.Dense(3)
])
model.compile(optimizer='RMSprop',
              loss='mse')
model.fit(x_train,y_train,epochs=500)
model.evaluate(x_test,y_test)
y_pred = model.predict(x_test)
```

Figure 6.1: Actual code for the network model

Next, we employed add new features to our dataset by using feature engineering. We acquired the importance scores from Random forest aand used it to add 3 new columns to our dataset as shown previously in 5.8, 5.7.We then trained the RF based neural network model again and the mean squared error is found to be 1.5281, a much higher than the neural network with an MSE of 0.85.

6.2 Results and Analysis

We calculated the CIELAB color difference scores using the formula in 5.2 and discovered that the highest color differences for Neural Network and Support Vector Regression were 1.1 and 1.7 which were less than the threshold 2.3 CIELAB units. In this regard, we can conclude that the both the models performed outstandingly in target color prediction. For L1 and L2 regularization, the highest CIELAB color differences were around 4.4 and 4.1 CIELAB units. Such high error suggest that linear regression is very unsuitable for this task. [3] used friele's color mixing model to match 28 targets using 17 dyed cotton fibres, defined by CIELAB values, and three colorants in different amounts for the blends[12]. The average CIELAB error of the matches was about 5.7 CIELAB units.[3] also used Stearns-Noechel predict the colour of 234 two-component blends and a kubelka-Munk model. The average color difference scores were around 2.7 and 2.4 CIELAB units. This proves that neural networks and support vector regression can surpass conventional color mixing models at prediction.

Other regression models including decision tree and random forest could be investigated further with more number of features as they happen to give low color difference scores, 1.70 and 0.75 but high MSEs as shown in table 6.1. For NN and RF the average color differences (0.85 CIELAB units and 0.75 CIELAB units) are also the lowest, so we decided to make acombination of these two models(Rf and NN) using feature engineering. Unfortunately, its predictions did not improve. The multiple linear regression has the largest median color difference (2.85 CIELAB units) but smaller MSEs than DT and RF. For DT, the average error is 1.7 CIELAB units, for RF it is 0.75 CIELAB units. However, in many places as seen in 6.2, the error is negligeble indicating that these models are good but also provides large anomalies.

Models	MSEs	Color difference
MLR-3	3.60	2.85
L1	3.30	3.00
L2	3.70	2.90
SVR	0.50	1.05
DT	11.62	1.70
RF	3.88	0.75
NN	0.85	0.85
RF-NN	1.37	1.65

Table 6.1: Models, their MSEs and Median CIELAB color differences

Models	Tested			Predicted			Color difference
	L_t	a_t	b_t	L_p	a_p	b_p	$E_{a,b}$
MLR-3	72.61	-17.62	26.99	71.7	-13.7	25.7	**4.2**
	64.07	0.82	-3.26	64.6	3.2	-2.7	2.5
	65.43	5.21	12.42	65.7	8.0	12.9	2.8
	67.27	-3.81	13.76	67.6	-0.9	13.7	2.9
DT	62.34	18.02	6.46	62.34	18.02	6.46	0.0
	70.21	-13.87	18.31	72.61	-17.62	26.99	**9.8**
	70.12	-20.76	3.16	70.09	-20.85	2.57	0.6
	66.01	-1.84	6.37	66.35	-2.33	9.1	2.8
RF	62.34	18.02	6.46	62.3	17.5	5.7	1.0
	70.21	-13.87	18.31	68.9	-8.8	16.2	**5.6**
	70.12	-20.76	3.16	70.1	-20.5	2.9	0.4
	66.01	-1.84	6.37	65.8	-1.5	6.2	0.5
L2	72.61	-17.62	26.99	71.7	-13.8	25.8	**4.4**
	64.07	0.82	-3.26	64.6	3.2	-2.7	2.5
	65.43	5.21	12.42	65.7	8.0	12.9	2.8
	67.27	-3.81	13.76	67.6	-0.6	13.7	3.2
L1	72.61	-17.62	26.99	71.8	-13.8	25.8	**4.1**
	64.07	0.82	-3.26	64.6	3.2	-2.7	2.5
	65.43	5.21	12.42	65.7	8.0	12.9	2.9
	67.27	-3.81	13.76	67.6	-0.9	13.7	2.9
SVR	72.61	-17.62	26.99	72.3	-17.1	27.0	0.6
	64.07	0.82	-3.26	64.1	1.9	-3.4	1.1
	65.43	5.21	12.42	65.5	6.9	12.8	**1.7**
	67.27	-3.81	13.76	67.6	-2.9	13.3	1.0
NN	72.61	-17.62	26.99	71.9	-17.6	26.7	0.8
	64.07	0.82	-3.26	63.5	1.2	-3.5	0.7
	65.43	5.21	12.42	65.5	5.9	13.2	**1.1**
	67.27	-3.81	13.76	66.5	-3.4	13.8	0.9
RF-NN	72.61	-17.62	26.99	72.0	-17.7	25.8	1.3
	65.43	5.21	12.42	65.1	7.6	12.0	**2.5**
	67.27	-3.81	13.76	66.3	-2.6	12.6	2.0
	76.27	-8.13	53.68	76.3	-7.8	54.9	1.2

Table 6.2: Results from different Models (MLR-3 representing 3 Multiple Linear Regression, DT representing Decision Tree classifier, RF representing Random Forest, L1 and L2 regularization, SVR representing Support Vector Regression, NN representing Neural Network in 3-16-16-3 architecture, RF-NN representing Random Forest based Neural Network), where L_t,a_t,b_t are tested/actual L*a*b values whereas L_p,a_p,b_p are predicted L*a*b values

Summary

7.1 Discussion

The Highest Color differences in MLR-3, DT, RF are 4.2, 9.8, 5.6 CIELAB units respectively which are higher than L2, SVR, NN, RF-NN. The lower the color difference , the better is the performance of the models and MSEs indicate the magnitude of differences of each output variable. Thus, by observing both of these metrics, we can evaluate which models perform the best. The MSEs in the standard neural network model varied significantly each time we ran the model. However, the most frequently observed MSEs ranged from 0.8 to 0.70, so we decided to fix it at 0.85. Upon examining the neural network model, we concluded that it is currently the most suitable model for color prediction. Our goal was to enhance its performance through feature engineering techniques, and as a result, we found that the RF-NN model exhibited a larger median color difference compared to the standard neural networks and random forest alone. Overall, the neural network model outperforms all the other models, demonstrating both a small MSE and color difference.

The results also highlight that Support Vector Machine could be the second best in performance after Neural Network as it has the lowest MSE and a color difference below the threshold. If provided with larger dataset with more features (e.g temperature, pH, water ratio, etc), then both SVR and NN could predict with same accuracy whereas RF-NN could be used to eliminate useless features or give importance to features that directly impact the output to a greater extent. It is possible that the constraint of using only three components in the blends and a small scale dataset could have been a factor in the relatively large errors obtained as hinted in [12].

7.2 Conclusion

For improving the recipe preparation process, an accurate color prediction on fabric is necessary. If the prediction of color for each DCS is accurate, the colorists can identify what color the recipe would transmit on the fabric beforehand and it will save them a lot of time and effort by not practically carrying out the recipe preparation processes. ANN based approach to perform the prediction will increase the feasibility of the methods for achieving the demands and developing the system. So far AI induced improvements to production and quality control have benefitted textile industries and are currently receptive to adopt any kinds of improvements.

Bibliography

[1] J. M. Bishop, M. J. Bushnell, A. Usher, and S. Westland, "Genetic optimisation of neural network architectures for colour recipe prediction," in *Artificial Neural Nets and Genetic Algorithms*, R. F. Albrecht, C. R. Reeves, and N. C. Steele, Eds., Vienna: Springer Vienna, 1993, pp. 719–725, ISBN: 978-3-7091-7533-0.

[2] S. Westland, "Artificial neural networks and colour recipe prediction," in *Proceedings of the International Conference and Exhibition: Colour Science*, 1998, pp. 225–233.

[3] B. Philips-Invernizzi, D. Dupont, A.-M. Jolly-Desodt, and C. Cazé, "Color formulation by fiber blending using the stearns–noechel model," *Color Research & Application*, vol. 27, no. 2, pp. 100–107, 2002. DOI: https://doi.org/10.1002/col.10028. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/col.10028. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/col.10028.

[4] L. Thevenet, D. Dupont, and A. M. Jolly-Desodt, "Modeling color change after spinning process using feedforward neural networks," *Color Research & Application*, vol. 28, no. 1, pp. 50–58, 2003. DOI: https://doi.org/10.1002/col.10114. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/col.10114. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/col.10114.

[5] l. Yang and B. Kruse, "Revised kubelka–munk theory i. theory and application," *Journal of the Optical Society of America. A, Optics, image science, and vision*, vol. 21, pp. 1933–41, Nov. 2004. DOI: 10.1364/JOSAA.21.001933.

[6] R. Furferi and L. Governi, "Prediction of the spectrophotometric response of a carded fiber composed by different kinds of coloured raw materials: An artificial neural network-based approach," *Color Research & Application*, vol. 36, no. 3, pp. 179–191, 2011. DOI: https://doi.org/10.1002/col.20623. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/col.20623. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/col.20623.

[7] W. Mokrzycki and M. Tatol, "Color difference delta e - a survey," *Machine Graphics and Vision*, vol. 20, pp. 383–411, Apr. 2011.

[8] R. R.Convert L.Schacher, "An expert system for the dyeing recipes determination," *Journal of Intelligent Manufacturing*, vol. 11, no. 2, pp. 145–155, 2014. DOI: 10.1023/A:10089867321. eprint: 10.1023/A:1008986732187. [Online]. Available: https://doi.org/10.1023/A:1008986732187.

[9] J. P. Hwang, S. Kim, and C. K. Park, "Development of a color matching algorithm for digital transfer textile printing using an artificial neural network and multiple regression," *Textile Research Journal*, vol. 85, no. 10, pp. 1076–1082, 2015. DOI: 10.1177/0040517515569525. eprint: https://doi.org/10.1177/0040517515569525. [Online]. Available: https://doi.org/10.1177/0040517515569525.

[10] M. Jawahar, C. B. Narasimhan Kannan, and M. Kondamudi Manobhai, "Artificial neural networks for colour prediction in leather dyeing on the basis of a tristimulus system," *Coloration Technology*, vol. 131, no. 1, pp. 48–57, 2015.

[11] P.-F. Li, J. Wang, and J.-F. Jing, "Application of improved back propagation algorithm in color difference detection of fabric," *Color Research & Application*, vol. 40, no. 3, pp. 311–317, 2015. DOI: https://doi.org/10.1002/col.21895. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/col.21895. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/col.21895.

[12] C. Hemingray and S. Westland, "A novel approach to using neural networks to predict the colour of fibre blends," *Coloration Technology*, vol. 132, no. 4, pp. 297–303, 2016. DOI: https://doi.org/10.1111/cote.12220. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cote.12220. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cote.12220.

[13] J. Shen and X. Zhou, "Spectrophotometric colour matching algorithm for top-dyed mélange yarn, based on an artificial neural network," *Coloration Technology*, vol. 133, no. 4, pp. 341–346, 2017. DOI: https://doi.org/10.1111/cote.12285. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cote.12285. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cote.12285.

[14] J. Shen, X. Zhou, H. Ma, and W. Chen, "Spectrophotometric prediction of pre-colored fiber blends with a hybrid model based on artificial neural network and stearns–noechel model," *Textile Research Journal*, vol. 87, no. 3, pp. 296–304, 2017. DOI: 10.1177/0040517516629145. eprint: https://doi.org/10.1177/0040517516629145. [Online]. Available: https://doi.org/10.1177/0040517516629145.

[15] A. Moussa, "Textile color formulation using linear programming based on kubelka-munk and duncan theories," *Color Research & Application*, vol. 46, no. 5, pp. 1046–1056, 2021. DOI: https://doi.org/10.1002/col.22626. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/col.22626. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/col.22626.

[16] X. Qin and X. J. Zhang, "An industrial dyeing recipe recommendation system for textile fabrics based on data-mining and modular architecture design," *IEEE Access*, vol. 9, pp. 136 105–136 110, 2021. DOI: 10.1109/ACCESS.2021.3117261.