Product Recommendation System for British Cosmetics

A Thesis by

Group 08	D/DBA/21/0006	MVD Nimsilu
	D/DBA/21/0025	NV Amarasinha
	D/DBA/21/0029	MDR PEIRIS
	D/DBA/21/0031	I Pandigama
	D/DBA/21/0037	WWAGK Palagolla
	D/DBA/21/0043	AABB Athukorala
INTAKE	38	

Supervised by

Mrs. ERC Sandamali

Mrs. SMM Lakmali

Submitted in partial fulfillment of the requirements for the award of BSc (Honours) in Data Science and Business Analytics Degree

GENERAL SIR JOHN KOTELAWALA DEFENCE UNIVERSITY

2024

Declaration, copyright statement and the statement of the supervisor

1 declare that this is my own work and this thesis does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.
Also, I hereby grant to KDU the non-exclusive right to reproduce and distribute my thesis, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books)

Signature :

Student Name : MVD Nimsilu
Student Registration Number : D/DBA/21/0006
Date : 22/01/2024

Signature :
Student Name : NV Amarasinha
Student Registration Number : D/DBA/21/0025
Date : 22/01/2024

Signature :
Student Name : MDR Peries
Student Registration Number : D/DBA/21/0029
Date : 22/01/2024

Signature :
Student Name : I Pandigama
Student Registration Number : D/DBA/21/0031
Date : 22/01/2024

Signature :

Student Name : WWAGK Palagolla

Student Registration Number : D/DBA/21/0037

Date : 22/01/2024

Signature :

Student Name : AABB Athukorala

Student Registration Number : D/DBA/21/0043

Date : 22/01/2024

The above candidates have carried out the project for the partial fulfillment of the BSc Honours in Data Science and Business Analytics Degree thesis under my supervision.

Signature :

Supervisor's Name : Mrs. ERC Sandamali

Date : 22/01/2024

Signature :

Supervisor's Name : Mrs. SMM Lakmali

Date : 22/01/2024

Abstract

This thesis explores the development of an automatic skincare product recommendation system to enhance the customer experience for British Cosmetics (Pvt) Ltd. The project involves data collection through a comprehensive survey, focusing on customer details, concerns, and preferences. The collected data is then processed and analyzed, utilizing exploratory data analysis (EDA) techniques, data preprocessing, and visualization tools.

The methodology incorporates machine learning models, including Random Forest Classifier, Support Vector Machine (SVM), Logistic Regression, and K-Nearest Neighbors (KNN) and Decision Tree Classifier models each tailored to predict the most suitable skincare products based on user attributes. The models undergo hyperparameter tuning and evaluation using cross-validation techniques.

The study emphasizes the significance of model interpretability and generalizability, ensuring recommendations for diverse customer profiles. The implementation involves standardization of features, handling categorical variables, and optimizing model performance through grid search and cross-validation. The evaluation metrics include accuracy, precision, recall, and F1-score, offering a comprehensive assessment of model effectiveness. Overall, this research demonstrates the feasibility and effectiveness of employing data-driven approaches to enhance customer satisfaction in the skincare product domain, providing valuable insights and a foundation for future advancements in personalized recommendation systems.

Acknowledgement

We extend our sincere appreciation to all who contributed to the completion of this thesis on customer recommendations for British Cosmetics (Pvt) Ltd.

Our gratitude goes to our supervisors Mrs. ERC Sandamali and Mrs. SMM Lakmali and to all other lecturers of our faculty for the invaluable guidance and support throughout the research process.

We also thank the management and staff of British Cosmetics (Pvt) Ltd for their cooperation, enriching our understanding of the cosmetics industry, and providing us with necessary information.

This project would not have been possible without the collective effort and support of all those mentioned above. Thank you for being an integral part of our academic journey.

Table of Contents

List of Figures

Chapter 01: Introduction

1.1 Prologue

This project aims at constructing a Product Recommendation System for British Cosmetics (Pvt) Ltd, a prominent beauty care company specializing in skincare. The research focuses on developing an advanced system utilizing innovative technologies to provide personalized skincare recommendations. By analyzing individual customer preferences, skin types, and concerns, the system aims to offer tailored suggestions from the extensive British Cosmetics product line. The primary objective is to enhance the skincare experience, ensuring that customers receive precise recommendations aligning with their gender, skin type, skin concerns and age.

In essence, the thesis explores how we can use machine learning techniques to predict what customers might like. This sophisticated approach goes beyond just following industry trends – it's a deliberate effort by British Cosmetics to exceed customer expectations. The aim is to provide a product recommendation that fits each person's preferences and needs. As skin care is mostly codependent on the customers unique skin texture, the system has the ability to help the customers find the correct product for their own skin. This prolegomenon is like opening the door to a new way of thinking about beauty, where technology and personal care come together.

1.2 Background and Motivation

With 19 years of experience, British Cosmetics is well known among the beauty realm as one of the country's leaders in cosmetics. Their vision is to provide uncompromised **world** class skin and beauty care products to their client base. Respecting and caring for our customers' well-being has always been the key to success at British Cosmetics. Today, their commitment remains strong as we strive to offer the finest solutions to the Sri Lankan market. This commitment has led them to finding solutions for their customers, to identify the best products for both the online and on store platforms. Their beauty products range from, makeup and cosmetics, haircare products, eyecare products, skincare, etc.

The beauty industry, known for its creativity and trends, is going through a big change with technology. Using data analytics and machine learning has become a powerful way to bring in new ideas. British Cosmetics sees how successful this approach has been in similar areas and wants to use predictive analytics to understand customers better. The goal is to have more interactions and build stronger connections with its diverse customers.

This idea comes from realizing that in today's digital age, using data insights can really change how the beauty industry works. By using predictive analytics, British Cosmetics wants to not just keep up but stay ahead in meeting the changing needs of its customers. The appeal of using data-driven strategies is that they can reveal patterns and trends that traditional methods might miss. Through this effort, British Cosmetics aims to be a leader in the industry, adapting its strategies to the tech changes happening in the beauty market.

1.3 Aims

•Conduct a predictive analysis for British Cosmetics.

•Utilize data-driven insights to enhance marketing strategies and customer experiences.

1.4 Objectives of the Project

The project at British Cosmetics is designed with several key objectives in mind. Firstly, we aim to predict the best product that tailors to need of the customers based on their conditions and concerns. This allows us to tailor our offerings to each individual's concerns, providing a more personalized experience.

Customer satisfaction is a focal point of our efforts, and the project aims to elevate customer experiences by understanding and responding to their behaviors. This personalized approach not only strengthens our connections with customers but also fosters loyalty and satisfaction.

To remain agile in the ever-changing beauty industry, we emphasize continuous monitoring and adaptation. This ensures that our strategies stay relevant and responsive to evolving trends and consumer preferences.

Finally, ethical considerations are paramount in our project objectives. We are dedicated to the responsible use of data, prioritizing customer privacy and maintaining transparency. This commitment reflects our values and contributes to building trust and positive relationships with our clientele.

1.5 Problem Domain

The beauty industry is unique, with diverse customer needs, and our goal is to go beyond generic recommendations.

The challenge here is that beauty preferences are unique to each person, and traditional one-size-fits-all recommendations don't work. We need a more detailed approach to figure out what each customer really wants in their beauty products. This project aims to dive deep into this domain, uncovering the subtleties of beauty preferences and behaviors. Our objective is to build a recommendation system that meets customer expectations, making the shopping experience at British Cosmetics truly exceptional.

1.6 Significance of the study

The significance of this study lies in its potential to redefine the beauty and skincare industry through the utilization of age, gender, skin type, and skin concerns as key parameters for product recommendations.

Improved Customer Satisfaction - By leveraging age, gender, skin type, and skin concerns, the study aims to provide more accurate and tailored product recommendations, leading to increased customer satisfaction.

Enhanced Relevance - The utilization of specific customer attributes allows for a more precise understanding of individual needs, ensuring that product recommendations align closely with customers' unique profiles.

Deeper Customer Connections - Tailoring recommendations based on age, gender, skin type, and concerns fosters a deeper connection with customers, as they perceive the brand's understanding of their specific skincare requirements.

Conducting comprehensive testing and evaluation is a pivotal step in the implementation process. Rigorous testing is necessary to ensure the accuracy and reliability of the recommendation system, identifying and rectifying potential issues before deployment.

1.7 Deliverables

The most important deliverable of the project is the machine learning model that has the ability to predict the best skincare product according to the customer's gender, age, skin type, and the present skin concern.

1.8 Timeline

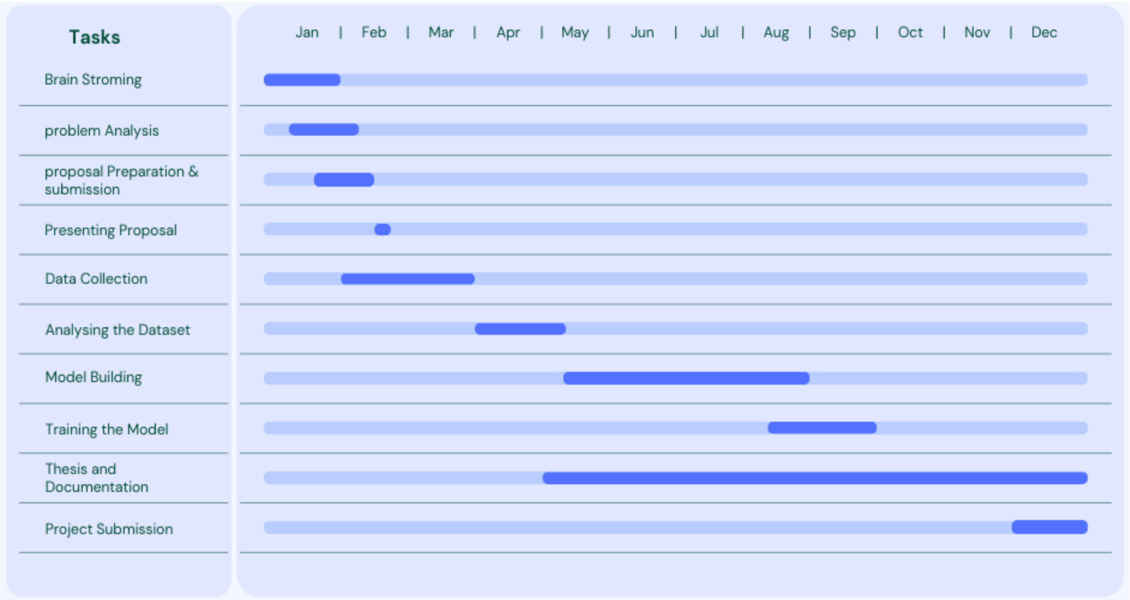

Figure 1. 1 Timeline of the project

1.9 Summary

In summary, this project signifies a significant step forward for British Cosmetics, introducing an advanced product recommendation system. In response to the changing beauty industry, the study highlights the importance of delivering tailored recommendations based on key factors like age, gender, skin type, and specific concerns.

Going beyond industry trends, the project aims to surpass customer expectations by filling a noticeable gap in the market. The introduction outlines challenges in implementing the system, including ensuring data accuracy, managing complex algorithms, and seamlessly integrating the new system with existing processes.

Conducting comprehensive testing and evaluation is a pivotal step in the implementation process. Rigorous testing is necessary to ensure the accuracy and reliability of the recommendation system, identifying and rectifying potential issues before deployment.

Chapter 02: Literature Review

2.1 Overview

The literature review for this project explores a variety of sources, focusing primarily on methodologies for predicting optimal skincare product recommendations. Various models are examined, utilizing algorithms, mathematical models, machine learning, and deep learning approaches to forecast ideal skincare product suggestions. The emphasis is on ensuring accurate recommendations for users. The review also highlights the importance of considering factors such as individual skincare needs, dynamic skincare planning, and real-world user satisfaction metrics when evaluating the effectiveness of automated prediction models for skincare product recommendations. The goal is to present a comprehensive understanding of successful methods in automating the prediction of optimal skincare product recommendations within the existing literature, contributing valuable insights to the research.

2.2 Skincare and Machine learning

There are several literature papers that are related to skincare industry, the research by Arya Kothari, et al [1] mentions how skin spectrum is varied and about the various skin types that are significant in selecting products.

There are various recommender systems related to skincare, which use different methods of selection, Gyeongeun Lee's [2] research proposes a recommendation system that analyzes the ingredients and user preferences to suggest suitable skincare products. Many papers pay attention to personalized recommendation according to the characteristics of the customers.

2.3 Approaches used in skincare recommendation systems

Filtering methods for skincare recommendation :

Various skincare recommender systems have used the content-based filtering when building their systems. [2] The products are recommended basted on the characteristics of the items and the users' preferences. Some other recommenders have used Collaborative filtering for this [3]. Here the system identifies who are having common preferences and recommends items that those users have liked but the target user hasn't seen yet. Certain systems use hybrid filtering techniques, by using both collaborative filtering and content- based filtering together. [4] [5]. Hybrid recommendation systems aim to combine the strengths of both content-based and collaborative filtering to improve recommendation accuracy.

Machine learning algorithms used in recommendation systems :

Machine learning algorithms are the building blocks of building a model. Here we will explore a variety of algorithms used in recommendation systems. Convolutional Neural Networks (CNN) have been used in several for skin identification [6] [4] [1] [7]

K- nearest neighbors has been used to create effective models in some cases as well [4] where it has been used to group customers into similar

Supervised learning algorithms like support vector machines(SVM) have been used for prediction in skin type classification tasks [1] [7] [6]

2.4 Challenges and opportunities

In reviewing the literature on smart facial skincare products utilizing computer vision technologies, common challenges have emerged. Subjectivity and individual differences in skincare concerns pose a significant hurdle [6] [1], leading to difficulties in creating recommendation systems that accurately diagnose diverse skin conditions and determine product suitability based on personal needs and preferences. Technological limitations, including reduced recognition rates in conventional computer vision technologies and accuracy challenges in skin type classification from facial images, further complicate the development of precise skincare analysis. The overarching challenge lies in the customization of existing models to capture individual preferences, concerns, and skin characteristics accurately, limiting the potential for personalized skincare recommendations. [4]

Despite these challenges, there are several common opportunities identified across the literature. Personalization stands out as a key theme, with computer vision technologies enabling personalized skincare solutions and advanced techniques such as deep learning and collaborative filtering facilitating personalized recommendations. Additionally, there is an opportunity for innovation in skincare business models, particularly with the introduction of unmanned stores, which not only offer individualized services but also reduce operating expenses. The integration of machine learning, deep learning, and advancements in computer vision and image processing provides avenues for improving the precision of facial feature identification and enhancing the overall classification of skin types. Ethical considerations, user trust, and transparency also play a crucial role in fostering a positive user experience, emphasizing the importance of ethical behavior in data use.

Continuous improvement through feedback mechanisms and the incorporation of user feedback further contribute to the opportunities for refining skincare recommendation systems. In addition, the exploration of new technologies, such as augmented reality (AR) and virtual reality (VR), holds promise in enhancing the user experience in skincare consultations and product recommendations. The literature review highlights these common challenges and opportunities, providing insights into the complex landscape of smart facial skincare product development.

Chapter 03: Requirement Analysis

3.1 Foundational Elements of the Skincare Recommendation Project

Introduction:

The skincare recommendation project always takes place in the always changing and innovative skincare industry, where consumer demands are always changing in a short time period. With the use of customer ratings on skincare products, British Cosmetics is leading an effort to improve consumer happiness by introducing a novel skincare suggestion system with the use of ever evolving technology.

Motivation:

Motivated by the challenges faced by consumers about their skincare, the project responds to diverse skin types, multiple skin concerns, age and gender with their range of skincare products. The motivation is to improve the entire skin care experience by recommending a product by the system, according to the needs of each consumer.

Market Trends and Consumer Expectations:

A thorough examination of the current market trends and consumer expectations reveals a huge demand for personalized skincare solutions. In present, consumers influenced by factors like skin concerns and beauty standards, are increasingly seeking tailored recommendations. The project seeks to align with these expectations and deliver a product recommendation system under skincare category for British Cosmetics.

3.2 Challenges in the skincare domain

Navigating the Skincare Market:

The skincare market presents challenges to individuals due to the huge number of products available. The vast diversity of skincare products makes it challenging for consumers to identify products suitable for their specific concerns.

Product Effectiveness and Compatibility:

Choosing a product might be difficult when it comes to compatibility and effectiveness. Customers have doubts about which product will work for their skin type, concerns according to their gender and age. The project addresses these challenges by focusing on recommending products that are not only highly rated but also effective and compatible. By analyzing the products that used by customers and considering the most used products by consumers.

Changing Beauty Standards:

Making decisions becomes more difficult because of the dynamic nature of beauty standards. Now a days customers mostly use skincare products to maintain their skin rather than using makeup. These days people mainly focus on the healthiness of the skin and use less makeup. So customers are looking for skincare solutions that not only fit in with the latest trends but also they are concerned about the standard of the products.

3.3 Importance of Product Recommendations

Meeting Diverse User Needs:

Tailored product recommendations are essential for meeting consumers' various needs. Recognizing that every person has different concerns, taste, choices. This project aims to give the most suitable product to the new customer according to their skin concerns.

Increasing User Satisfaction and Outcomes:

Product recommendations are important when choosing a product. By recommending a product according to the user's preferences, it helps to increase customer satisfaction and the trust about the company. When the customer becomes satisfied about the outcome, it also helps to increase the brand loyalty.

3.4 Functional Requirements for the Skincare Recommendation System

Defining System Features:

When considering skincare product recommendation system, the functional requirements are the specific features and capabilities required to fulfill user goals and provide an effective recommendation experience. Here are some functional requirements for the skincare recommendation project,

- By using the survey details, making a dataset like customer profiles of existing customer base of British Cosmetics
- Make a product recommendation model for new customers
- Predicting the products by analyzing the patterns of the existing customers dataset and recommending those products to new customers by matching their concerns with existing customers dataset.
- Recommend the suitable product, when new users input their details like age, gender, skin type and skin concerns.

3.5 Non-Functional Requirements for the Skincare Recommendation System

Providing High Performance:

Providing high-performance capabilities to deliver quick and accurate recommendations that enhance the user experience.

Scalability and Reliability:

The system needs to be capable of managing large amounts of consumer data and be scalable enough to adapt accordingly when increasing the range of skincare products. System should guarantee consistent performance.

Usability and Maintainability:

The system should be user friendly for both end users and staff of British Cosmetics. Allowing individuals to easily understand recommendations. The system should be maintained to keep the system current and functional and to easily modify and update the system smoothly.

3.6 Hardware Requirements

Development:

There isn't much hardware needed for the skincare recommendation project. It is sufficient to have a computer with a lot of computing capacity for tasks like training and implementing machine learning models. Predicting the best skincare product based on the user inputs is the main objective.

3.7 Software Requirements

Programming Language – Python:

Python has been selected as the programming language for the skincare product recommendation project. In today's technical scene, Python is a prominent platform; its compatibility with British Cosmetics' systems guarantees a smooth integration.

Python is used throughout the project for data preprocessing, model training, and implementation.

Integrated Development Environment (IDE) - Google Co-laboratory:

The selected IDE for this project is Google Co-laboratory, a well-known environment for developing machine learning models. Known for its user-friendly interface and suitability for machine learning tasks, Google Co-laboratory streamlines the process of data preprocessing and subsequent tasks involved in the skincare recommendation system. The entire project is executed within the Google Co-laboratory environment, ensuring consistency and ease of use.

3.8 Conclusion of Requirement Analysis:

In summary, the requirement analysis stage has revealed important details of the skincare product recommendation project for British Cosmetics. Motivated by the always changing skincare industry and challenges faced by consumers, the project aims to enhance user satisfaction through product recommendations. The analysis outlined challenges in navigating the skincare market, addressed the importance of tailored recommendations, defined both functional and non-functional requirements and addressed both hardware and software requirements to the system.

In summary, a strong skincare recommendation system that satisfies user expectations and market trends is made possible by the requirement analysis stage. The identified requirements will guide subsequent phases in the project, ensuring the successful implementation of an effective and user-friendly skincare product recommendation system for British Cosmetics.

Chapter 04: Methodology

The skincare product recommendation project for British Cosmetics (Pvt) Ltd followed a structured methodology beginning with comprehensive data collection through a customer survey, spanning three months. The dataset, inclusive of customer details and preferences, underwent meticulous preprocessing, including data cleaning and transformation. Exploratory Data Analysis (EDA) utilizing diverse visualizations unveiled key relationships. The subsequent model-building phase featured the implementation and optimization of machine learning models leading to the development of a sophisticated skincare recommendation system aligned with individual customer profiles.

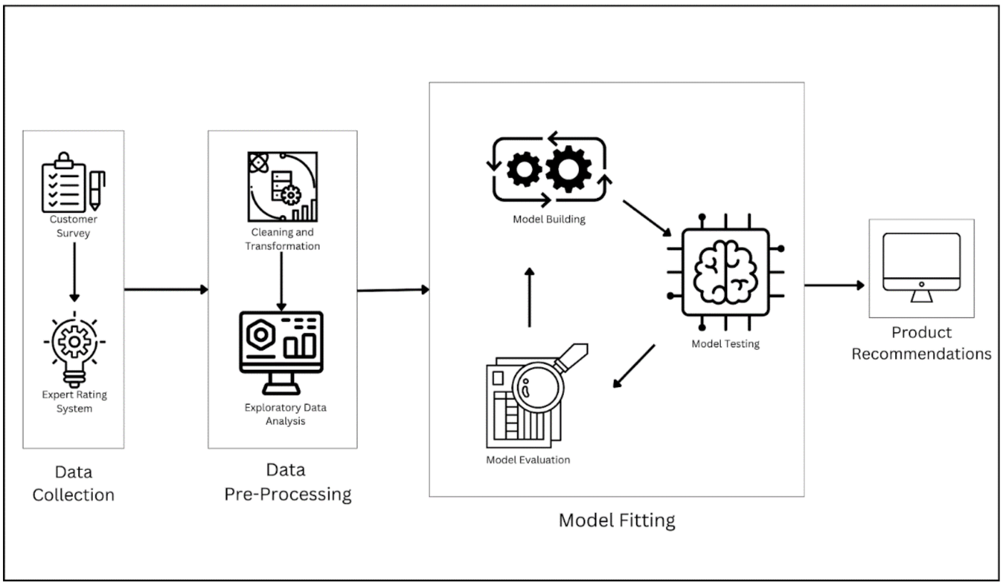

Figure 4. 1 Methodology Flowchart

4.1 Data Collection

4.1.1 Source of Data

In response to the imperative need for an enhanced customer experience and a personalized skincare product recommendation system, our collaboration with British Cosmetics (Pvt) Ltd initiated a meticulous data collection process.

The primary data source was a thoughtfully designed survey, created through collaborative efforts with company representatives and our academic supervisor. This survey aimed to capture comprehensive customer details, skin type and their respective concerns. To encourage participation, a gift giveaway format was employed, fostering engagement. The survey, administered over a three-month period, resulted in a dataset reflecting diverse customer profiles and preferences. Ethical considerations were paramount, with participants providing informed consent, and measures taken to ensure anonymity and confidentiality.

Quality assurance procedures were rigorously implemented to maintain data integrity. Regular checks and validation processes identified and rectified inconsistencies. Continuous collaboration with British Cosmetics (Pvt) Ltd facilitated real-time adjustments to survey strategies, ensuring alignment with business objectives. Data security was prioritized, employing robust measures for storage and access control.

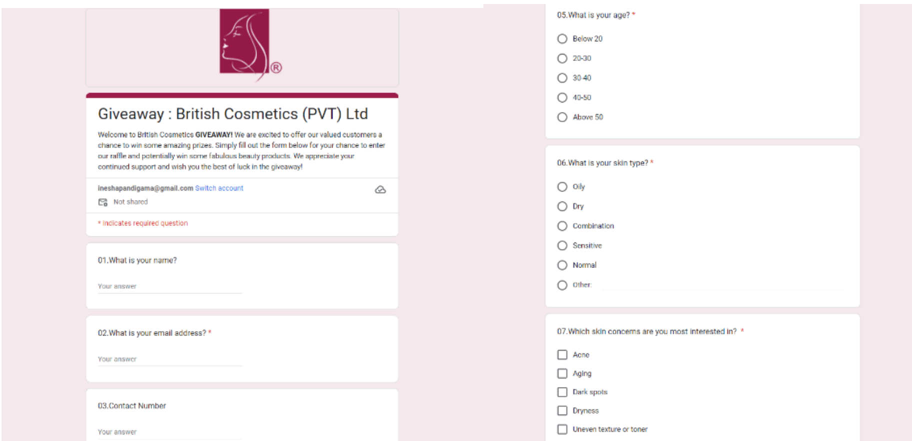

Figure 4. 2 Survey Form

4.1.2 Description of Data

The data, collected through the designed survey, was systematically recorded and stored in Excel sheets to ensure data integrity and reliability. Essential customer details, including names, emails, and telephone numbers, were meticulously gathered during the survey administration process.

British Cosmetics (Pvt) Ltd envisioned a predictive model that could intelligently recommend skincare products tailored to individual customer needs. The predictive factors encompassed customer gender, skin type, age, and specific skin concerns highlighted by the participants. Following the survey's completion, skincare experts, formulated product recommendations personalized for each customer, implementing a rating system.

The age variable was stratified into five categories to capture the diverse age groups effectively.

1. Below 20
2. 20 - 30
3. 30 - 40
4. 40 - 50
5. Above 50

Skin types were categorized into five groups

1. Normal Skin Type
2. Oily Skin Type
3. Combination Skin Type
4. Dry Skin Type
5. Sensitive Skin Type

Skin concerns that were taken into consideration in the survey are

1. Acne
2. Ageing
3. Dark spots
4. Dryness
5. Uneven texture or toner

This thorough consideration aimed to enhance the precision and relevance of the subsequent product recommendations.

Beyond the core dataset, supplementary information regarding skincare use frequency and customer satisfaction with the company's products was diligently collected. This additional layer of insights serves a dual purpose: supporting internal company analysis and providing a nuanced comprehension of the intricate dynamics between customers and the products offered by British Cosmetics (Pvt) Ltd.

	A	B	C	D	E	F	G	H	I
1	Time	Name	Email		Gender	Age	Skin Type	Skin Concerns	
2	4/17/2023 16:38:03	Test - BC	supuni.britishcosmetics@gmail.com		Female	20-30	Combination	Acne, Dryness, Uneven texture or toner	
3	4/17/2023 16:44:14	BC-Test	anushkaoffice22@gmail.	0769260907	Female	20-30	Combination	Dark spots	
4	4/28/2023 17:05:05	Chamathka maddumarac	Maddumarachchichamat	0762074466	Female	20-30	Combination	Dryness	
5	4/28/2023 17:05:48	Sayuni Senadeera.	sayunisenadeera4@gma	0740803785	Female	20-30	Oily	Uneven texture or toner	
6	4/28/2023 17:12:54	Sandul Tharana	sandultharana21@gmail.	0714801247	Male	20-30	Normal	Acne, Dark spots	
7	4/28/2023 17:14:17	Shalini Peiris	shalinisubodha07@gmai	0769997506	Female	20-30	Normal	Dark spots, Dryness, Uneven texture or toner	
8	4/28/2023 17:17:48	Eumi Perera	eumiperera@gmail.com	754919924	Female	20-30	Normal	Uneven texture or toner	
9	4/28/2023 17:18:47	Amandi Greshani	greshaniamandi@gmail.	0740025897	Female	20-30	Dry	Dryness	
10	4/28/2023 17:20:32	Bisandi Jayasundera	Bisandi@hotmail.com	0710548192	Female	20-30	Oily	Acne	
11	4/28/2023 17:20:37	Praveen Dissanayake	praveeen924@gmail.con	0713771791	Male	20-30	Oily	Aging, Uneven texture or toner	
12	4/28/2023 17:23:08	Tharindi Theekshani	tharinditheekshani@gma	0763741869	Female	20-30	Dry	Dryness	
13	4/28/2023 17:23:33	Chamathi	chamathimaneesha@gm	0766477973	Female	20-30	Combination	Dark spots, Dryness, Uneven texture or toner	
14	4/28/2023 17:26:21	Tiromi Gunarathne	tiromigunarathne@gmail.	0703733484	Female	20-30	Oily	Acne	
15	4/28/2023 17:29:52	Madusha Wijesundera	Madushaw305@gmail .c	0714348155	Female	30-40	Sensitive	Dark spots	
16	4/28/2023 17:33:26		Gawiniw@gmail.com		Female	20-30	Oily	Dark spots	
17	4/28/2023 17:33:27	Milani Weerakkody	juliathomas194@gmail.c	0772695088	Female	20-30	Oily	Acne, Aging, Dark spots, Uneven texture or toner	
18	4/28/2023 17:34:02	Helani Batuwantudawa	helanibatu2017@gmail.c	0777286814	Female	20-30	Oily	Acne, Dark spots	
19	4/28/2023 17:34:05	Sasmi Sumanasekara	sasmigayangasumanase	0766055645	Female	20-30	Normal	Acne, Dark spots	
20	4/28/2023 17:34:34	nethma peiris	nethmapeiris@gmail.con	0762309986	Male	20-30	Oily	Acne, Dark spots	
21	4/28/2023 17:35:32	Pradeepika Hettiarachch	Pradeepika.h.76@gmail.	0718066727	Female	40-50	Combination	Acne, Aging	
22	4/28/2023 17:37:24	Ravindi Hewage	ravindihewage0724@gm	0702104447	Female	20-30	Normal	Dryness, Uneven texture or toner	
23	4/28/2023 17:37:52	Kaushalya	kaushiwanigatunga2000	0779049162	Female	20-30	Oily	Acne, Uneven texture or toner	
24	4/28/2023 17:38:03	DMKS Dissanayake	kaveeshasupunsara@gn	0786777647	Female	20-30	Oily	Dark spots	
25	4/28/2023 17:39:32	Deenath	deenathrajapaksha454@	0779286910	Male	20-30	Combination	Acne, Dark spots, Dryness	
26	4/28/2023 17:40:31	Sashani	sashaniliyanage@gmail.	0715581604	Female	20-30	Oily	Dark spots	
27	4/28/2023 17:41:18	Senethma Gunathilake	senesaku2003@gmail.cc	0713231490	Female	Below 20	Normal	Acne, Aging, Dark spots, Dryness, Uneven texture or toner	
28	4/28/2023 17:44:57	Lakisha Kaluthotage	lakishakaluthotage@gma	0741118828	Female	20-30	Combination	Acne, Dark spots, Uneven texture or toner	
29	4/28/2023 17:45:14	Ravindu Jayalath	jayalathravindu726@gma	0718953803	Male	20-30	Normal	Acne	
30	4/28/2023 17:45:30	Tharushi	tharushiindrakumara@gmail.com		Female	20-30	Dry	Dryness	
31	4/28/2023 17:46:20	Prarthana Sewmini	Prarthanasewmini2001@	0714340688	Female	20-30	Sensitive	Acne, Dark spots, Dryness, Uneven texture or toner	
32	4/28/2023 17:46:33	Hiruni Peiris	hirunipeiris323@gmail.cc	0711772853	Female	20-30	Combination	Uneven texture or toner	
33	4/28/2023 17:46:35	MS.Dushani Rajapakshe	dushanirajapaksa72@gn	0773293977	Female	Above 50	Normal	Aging	

Figure 4. 3 Dataset after data collection

4.2 Data Preprocessing

4.2.1 Cleaning and Transformation

Upon completion of the data collection phase, a critical step involved the cleaning and transformation of the acquired dataset. To enhance the dataset's relevance to the problem at hand, extraneous data columns such as email and telephone numbers were omitted, streamlining the dataset for optimal usability. The elimination of unnecessary columns contributed to a more focused dataset, aligning with the specific requirements of the predictive modelling task. Systematic examination of the dataset was executed to identify and address potential discrepancies, particularly in terms of null or erroneous values. No substantial instances of missing or erroneous data were found, affirming the overall usability of the dataset.

In addressing the challenge of recommending the most suitable products to customers, we prioritized the product with the highest rating within the rated product sets after discussion and advices from the company representatives. This approach ensured that the recommended products held the highest perceived efficacy, as indicated by the expert ratings. To facilitate the integration of categorical data into the modeling phase, we used label encoders. These encoders systematically converted categorical variables, such as gender, skin type, and other relevant factors, into numerical representations.

```
# Label Encoding for 'Age', 'Skin Type', and 'Gender' columns
label_encoder = LabelEncoder()

columns_to_encode = ['Age', 'Skin Type', 'Gender']
for col in columns_to_encode:
    data_processed[col] = label_encoder.fit_transform(data_processed[col])

# Display the processed dataset
print("\nProcessed Dataset:")
print(data_processed.head())

# Save the processed dataset to a new file
data_processed.to_csv('processed_dataset.csv', index=False)  # Save the processed dataset to a new CSV file
print("\nProcessed dataset saved as 'processed_dataset.csv'")
```

Figure 4. 4 Use Example of label encoder code in numerical conversions

In summary, the data preprocessing phase emerged as a crucial preparatory step, characterized by strategic cleaning, thoughtful product selection, and intelligent encoding of categorical data. These meticulous efforts laid the foundation for a robust dataset poised for effective utilization in subsequent modeling endeavors.

4.2.2 Exploratory Data Analysis

The Exploratory Data Analysis (EDA) phase was carried out to take the the intrinsic patterns within the dataset. By employing statistical techniques and visualizations, we aimed to gain valuable insights into the distribution and interrelationships among various features.

One of the primary objectives during EDA was to delve into the distribution of individual features. Histograms were extensively used to visually inspect the spread and central tendencies of key features such as age groups, skin types, and skin concerns. Additionally, box plots provided a concise summary of the distribution's quartiles, aiding in the identification of potential outliers.

Correlation analysis was carried out to identify relationships between different features. By computing correlation coefficients, we gauged the degree and direction of linear relationships among variables. A correlation matrix, coupled with a heatmap representation, provided a comprehensive overview of pairwise associations, highlighting potential dependencies. Scatter plots and pair plots further enriched our understanding by showcasing bivariate relationships.

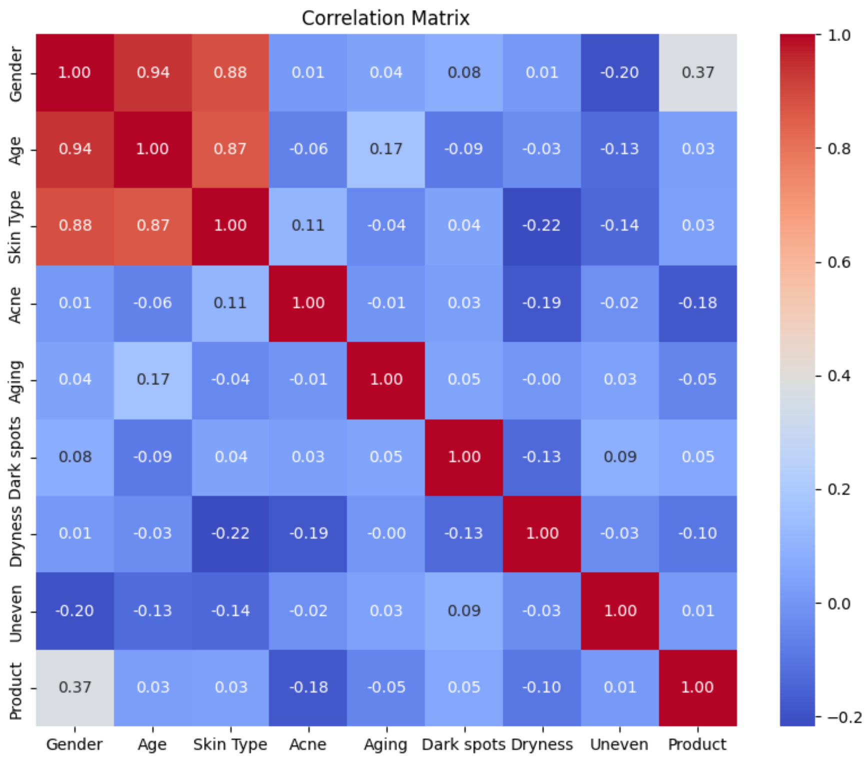

Figure 4. 5 Correlation Heat

Data visualizations were diverse, employing a range of plots for different types of variables. Bar charts and count plots were instrumental in examining the frequency distribution of various categorical variables, offering a clear representation of the popularity of different product choices among customers. Pair plots allowed for a multivariate exploration of relationships, providing a holistic view of feature interactions.

Histograms: Utilized to check the numerical distributions of features.

Box Plots: Provided a visual summary of distribution quartiles, aiding in outlier identification.

Scatter Plots: Examined relationships between two variables.

Pair Plots: Offered a comprehensive multivariate exploration of relationships.

Bar Plots and Count Plots: Visualized the frequency distribution of categorical variables.

Heatmaps: Represented correlation matrices visually, highlighting associations between features.

In essence, the EDA and correlation analysis stages served as a crucial precursor to modeling, dynamics within the dataset and steering subsequent decision-making processes. The combination of descriptive statistics and visually appealing plots enriched our understanding of the dataset's nuances, laying the groundwork for informed modeling strategies.

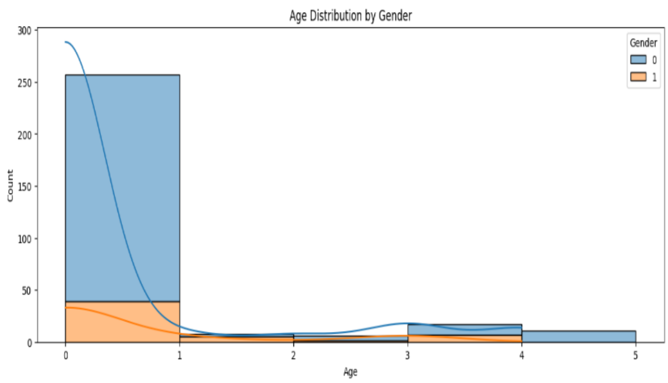

Figure 4. 6 Age Distribution by Gender

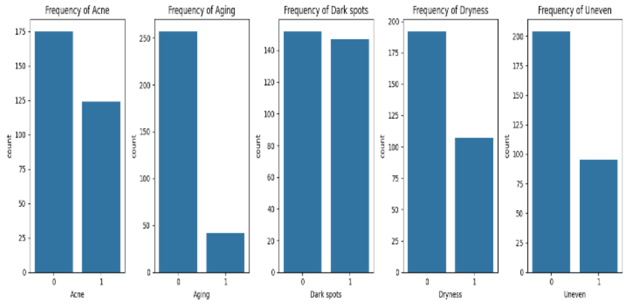

Figure 4. 7 Skin concern frequencies by gender

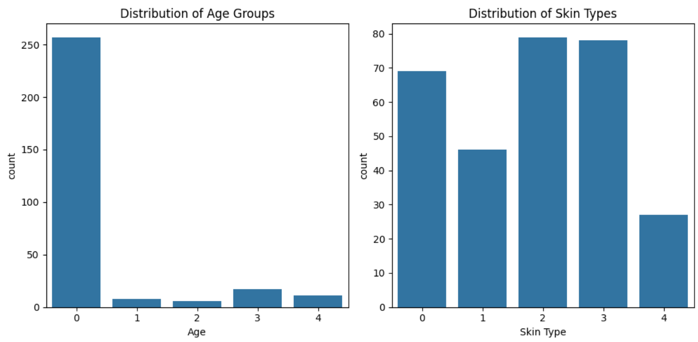

Figure 4. 8 Distribution of age and skin types

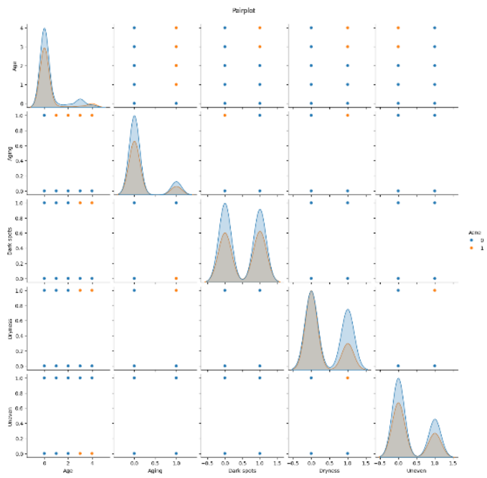

Figure 4. 9 Pair plot diagram

4.2.3 Attribute Handling

After the EDA was carried out we examined the significance of each of the variables for the final outcomes. Then the product use frequency was examined and we identified that only 8 products out of the 29 products in the company's product range had a significant use frequency. There we decided to consider all the other products as a one category for the accuracy of mode so the model will be focusing its prediction on 9 product categories.

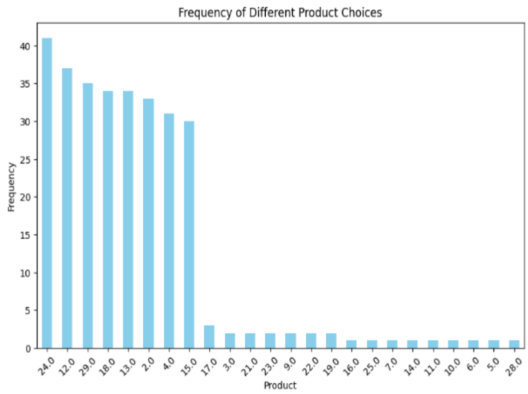

Figure 4. 10 Product use frequency

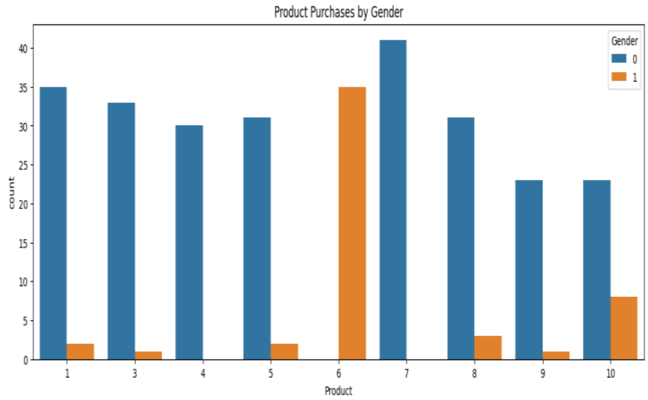

Figure 4. 11 Product use frequency after attribute handling

4.3 Machine Learning Models

4.3.1 Overview of Selected Models

In the model-building phase of our project, our objective was to develop a robust skin care product recommendation system for British Cosmetics (Pvt)Ltd. We began by exploring various machine learning models to capture the complex relationships between customer demographics, skin types, and skin concerns, aiming to predict the

most suitable products for individual customers. Our approach included models such as Logistic Regression, Random Forest Classifier, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and a Decision Tree. Each model was carefully selected to leverage its unique strengths in handling different aspects of the data and accommodating diverse patterns. Additionally, we performed hyperparameter tuning to optimize the performance of certain models. The model-building process was guided by a comprehensive understanding of the dataset obtained through exploratory data analysis and preprocessing. By employing a range of models, we sought to identify the most effective solution that aligns with the requirements of British Cosmetics, providing accurate and personalized product recommendations to enhance
customer satisfaction and experience.

1. <u>Logistic Regression Model</u>

We began by splitting the dataset into training and testing sets using the `train_test_split` function, maintaining an 80-20 ratio. Feature scaling was then applied to standardize the features using the `StandardScaler` from scikit-learn. This step is crucial for Logistic Regression, as it involves minimizing the impact of varying feature scales on the algorithm's performance.

The model underwent hyperparameter tuning using `GridSearchCV`, exploring different combinations of hyperparameters such as regularization strength (`C`) and penalty type (`penalty`). The objective was to identify the optimal configuration that maximizes accuracy.

Once the best hyperparameters were determined, the Logistic Regression model was instantiated and trained on the training set. Predictions were made on the test set, and the model's performance was evaluated using accuracy metrics. The accuracy score, calculated with `accuracy_score`, provides an overall measure of the model's correctness. Additionally, the detailed classification report, generated by `classification_report`, offers insights into the precision, recall, and F1-score for each skincare product class.

```python
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Standardize the features (important for some algorithms like Logistic Regression)
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Train a Logistic Regression model with hyperparameter tuning using GridSearchCV
param_grid = {
    'C': [0.001, 0.01, 0.1, 1, 10, 100],
    'penalty': ['l1', 'l2']
}

logreg_model = LogisticRegression(random_state=42)
grid_search = GridSearchCV(logreg_model, param_grid, cv=3, scoring='accuracy', n_jobs=-1)
grid_search.fit(X_train_scaled, y_train)

# Get the best parameters from the grid search
best_params = grid_search.best_params_
print(f"Best Hyperparameters: {best_params}")

# Train the model with the best parameters
best_logreg_model = LogisticRegression(**best_params, random_state=42)
best_logreg_model.fit(X_train_scaled, y_train)

# Make predictions on the test set
y_pred = best_logreg_model.predict(X_test_scaled)
```

Figure 4. 12 Logistic Regression Model Code

2. <u>K-Nearest Neighbor(KNN) Model</u>

The first step involved splitting the dataset into training and testing sets using the `train_test_split` function, maintaining an 80-20 ratio for robust model evaluation. Given the sensitivity of KNN to variations in feature scales, we standardized the features using the `StandardScaler` from scikit-learn.

Hyperparameter tuning was performed using `GridSearchCV`, exploring different combinations of hyperparameters like the number of neighbors (`n_neighbors`), the weight function (`weights`), and the Minkowski distance parameter (`p`). This rigorous process aimed to identify the optimal hyperparameters that maximize accuracy and effectiveness.

Once the best hyperparameters were determined, the KNN model was instantiated and trained on the scaled training set. Predictions were then generated on the test set, and the model's performance was assessed using key metrics. The accuracy score, calculated with `accuracy_score`, provides a comprehensive measure of the model's correctness. Additionally, the detailed classification report, generated by `classification_report`, furnishes insights into the precision, recall, and F1-score for each skincare product class.

```
# Train a KNN model with hyperparameter tuning using GridSearchCV
param_grid = {
    'n_neighbors': [3, 5, 7, 10],
    'weights': ['uniform', 'distance'],
    'p': [1, 2]
}

knn_model = KNeighborsClassifier()
grid_search = GridSearchCV(knn_model, param_grid, cv=3, scoring='accuracy', n_jobs=-1)
grid_search.fit(X_train_scaled, y_train)

# Get the best parameters from the grid search
best_params = grid_search.best_params_
print(f"Best Hyperparameters: {best_params}")

# Train the model with the best parameters
best_knn_model = KNeighborsClassifier(**best_params)
best_knn_model.fit(X_train_scaled, y_train)

# Make predictions on the test set
y_pred = best_knn_model.predict(X_test_scaled)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy}")
```

Figure 4. 13 K-Nearest Neighbor Model Code

3. Support Vector Machine(SVM) Model

To ensure the SVM's optimal performance, we started by splitting the dataset into training and testing sets using the `train_test_split` function, maintaining an 80-20 ratio. Feature scaling, accomplished through the `StandardScaler` from scikit-learn, was crucial for SVM models, as they are sensitive to variations in feature scales.

The SVM model underwent hyperparameter tuning via `GridSearchCV`, exploring different combinations of hyperparameters such as regularization parameter (`C`), kernel type (`kernel`), gamma coefficient (`gamma`), polynomial degree (`degree`), and coefficient `coef0`. This process aimed to identify the optimal hyperparameter configuration that maximizes accuracy.

Upon identifying the best hyperparameters, the SVM model was instantiated and trained on the scaled training set. Predictions were then made on the test set, and the model's performance was assessed using accuracy metrics. The accuracy score, calculated with `accuracy_score`, provides a comprehensive measure of the model's correctness. Additionally, the detailed classification report, generated by `classification_report`, furnishes insights into the precision, recall, and F1-score for each skincare product class.

```
# Standardize the features (important for SVM)
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Train an SVM model with hyperparameter tuning using GridSearchCV
param_grid = {
    'C': [0.1, 1, 10, 100],
    'kernel': ['linear', 'rbf', 'poly', 'sigmoid'],
    'gamma': ['scale', 'auto', 0.1, 0.01, 0.001],
    'degree': [2, 3, 4],
    'coef0': [0.0, 1.0, 2.0]
}

svm_model = SVC(random_state=42)
grid_search = GridSearchCV(svm_model, param_grid, cv=3, scoring='accuracy', n_jobs=-1)
grid_search.fit(X_train_scaled, y_train)

# Get the best parameters from the grid search
best_params = grid_search.best_params_
print(f"Best Hyperparameters: {best_params}")

# Train the model with the best parameters
best_svm_model = SVC(**best_params, random_state=42)
best_svm_model.fit(X_train_scaled, y_train)

# Make predictions on the test set
y_pred = best_svm_model.predict(X_test_scaled)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy}")

# Display classification report for more detailed evaluation
print("Classification Report:")
print(classification_report(y_test, y_pred))
```

Figure 4. 14 Support Vector Machine Model Code

4. <u>Decision Tree Classifier Model</u>

Initially, the dataset was divided into training and testing sets using the `train_test_split` function, maintaining an 80-20 ratio for robust model evaluation. To ensure uniformity in the treatment of features, especially for decision trees, the features were standardized using the `StandardScaler` from scikit-learn.

Subsequently, a Decision Tree Classifier model was instantiated and trained on the scaled training set. No hyperparameter tuning was conducted for this model, as decision trees are not as sensitive to scaling and usually don't require extensive hyperparameter adjustments.

Predictions were generated on the test set using the trained Decision Tree Classifier, and the model's performance was evaluated using key metrics. The accuracy score, calculated with `accuracy_score`, provides an overall measure of the model's correctness. Additionally, the detailed classification report, generated by `classification_report`, offers insights into the precision, recall, and F1-score for each skincare product class.

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, classification_report
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import GridSearchCV


data=pd.read_csv('Final.csv')
# Assuming 'data' is your DataFrame
X = data[['Age', 'Gender', 'Skin Type', 'Acne', 'Aging', 'Dark spots', 'Dryness', 'Uneven']]
y = data['Product']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Standardize the features (important for some algorithms like Decision Trees)
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Train a Decision Tree Classifier model
decision_tree_model = DecisionTreeClassifier(random_state=42)
decision_tree_model.fit(X_train_scaled, y_train)

# Make predictions on the test set
y_pred_decision_tree = decision_tree_model.predict(X_test_scaled)
```

Figure 4. 15 Decision Tree Classifier Model Code

5. <u>Random Forest Classification Model</u>

In this specific model, we employed the Random Forest Classifier, a powerful ensemble learning algorithm.

To ensure robust model performance, we initially split the dataset into training and testing sets, with an 80-20 ratio, utilizing the `train_test_split` function. Feature scaling was applied using the `StandardScaler` from scikit-learn, which standardizes the features to have zero mean and unit variance. This step is crucial, especially for algorithms sensitive to the scale of input features, such as the Random Forest Classifier.

The model was fine-tuned for optimal performance through hyperparameter tuning using `GridSearchCV`. We explored various hyperparameter combinations, including the number of estimators, maximum depth, minimum samples split, and minimum samples leaf, aiming to identify the configuration that yields the highest accuracy.

After identifying the best hyperparameters, we instantiated and trained the Random Forest Classifier with these parameters on the training set. Subsequently, we made predictions on the test set and evaluated the model's performance. The accuracy score, calculated using `accuracy_score`, provides a measure of the model's overall correctness. Additionally, the detailed classification report, generated by `classification_report`, offers insights into the model's precision, recall, and F1-score for each class (skincare product) in the dataset.

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import GridSearchCV

data = pd.read_csv('Final.csv')
# Assuming 'data' is your DataFrame
X = data[['Age', 'Gender', 'Skin Type', 'Acne', 'Aging', 'Dark spots', 'Dryness', 'Uneven']]
y = data['Product']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Standardize the features (important for some algorithms like Logistic Regression)
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Train a Random Forest Classifier model with hyperparameter tuning using GridSearchCV
param_grid = {
    'n_estimators': [50, 100, 200],
    'max_depth': [None, 10, 20],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4]
}

rf_model = RandomForestClassifier(random_state=42)
grid_search = GridSearchCV(rf_model, param_grid, cv=3, scoring='accuracy', n_jobs=-1)
grid_search.fit(X_train_scaled, y_train)

# Get the best parameters from the grid search
best_params = grid_search.best_params_
print(f"Best Hyperparameters: {best_params}")

# Train the model with the best parameters
best_rf_model = RandomForestClassifier(**best_params, random_state=42)
best_rf_model.fit(X_train_scaled, y_train)
```

Figure 4. 16 Random Forest Classifier Model Code

4.3.2 Evaluation Metrics

In the evaluation of the skincare product recommendation models, several key metrics were employed to gauge their performance. Commonly used metrics such as accuracy were calculated to determine the overall correctness of predictions. The classification report provided a detailed breakdown of precision, recall, and F1-score for each product category, offering insights into the models' ability to correctly classify instances. For Random Forest and Logistic Regression, hyperparameter tuning through GridSearchCV enhanced model efficiency. Additionally, ROC curves and confusion matrices provided a comprehensive view of model performance, aiding in the identification of potential improvements. This multifaceted evaluation approach ensured a robust understanding of the models' capabilities in recommending skincare products effectively.

Chapter 05: Technology

This chapter describes the technological aspects that are used in the development of the product recommendation system for the company British Cosmetics. Here, the technical considerations and tools that had been used will be explored in detail

5.1 Technical tools

Python was used as the key technology tool in the development of our recommendation models. The following tools were used, throughout our project.

5.1.1 Python for data visualization, analysis and model building

Python Programming Language

The main programming language used in the project was Python, which was used I data processing, exploratory data analysis, and the model building.

Google Colaboratory

Google Colab is a free, cloud-based platform for collaborative Python coding. It was used to create shareable notebooks containing code, visualizations, and text. Integrated with Google Drive, it offers GPU acceleration for machine learning tasks. Colab provides an accessible and efficient environment for collaborative coding and data analysis.

Libraries that were used:

- Pandas - Pandas is a versatile data manipulation and analysis library, we used this for efficient handling and manipulation of structured data.
- Numpy – Used for numerical operations and data manipulation.
- Matplotlib - Matplotlib is a popular plotting library that enables the creation of static, animated, and interactive visualizations in Python.

- Scikit-learn - Scikit-learn is a machine learning library that provides simple and efficient tools for data mining and data analysis, implementing various machine learning algorithms.
- Label Encoders from scikit-learn - Label Encoders in scikit-learn are tools for converting categorical labels into numerical format.
- GridSearchCV - GridSearchCV is a method in scikit-learn for hyperparameter tuning, systematically searching over a specified parameter grid.
- StandardScaler - StandardScaler is a feature scaling tool in scikit-learn used for standardizing numerical features.
- Seaborn - Seaborn is a powerful Python data visualization library based on Matplotlib, designed for creating informative and attractive statistical graphics.

Chapter 06: Testing And Evaluation

6.1 Testing

System testing

To guarantee the stability, precision, and practicality of the model in our Skincare Recommender System, extensive system testing is essential. The goals of the testing phase, the tactics used, and the results obtained are described in this section.

Objectives of Testing

The testing goals were carefully crafted to evaluate the Skincare Recommender System's dependability and efficacy.

- Accuracy Assessment:

Assess the model's accuracy in recommending skincare products according to the severity of skin types and skin concerns of the customers.

Testing Strategies

- Train-Test Split Function:

In machine learning, the train-test split function is a widely used method for evaluating a model's performance on hypothetical data. One subset of the dataset is used to train the model, and the other is used to evaluate the model's performance. By using this method, the recommender system is kept from overfitting to the training set and is able to make good generalizations to newly discovered data. It aids in assessing how well the model can recommend actions based on inputs that it hasn't seen during training.

- GridSearchCV

GridSearchCV is a hyperparameter tuning method that finds the optimal model performance by thoroughly searching through a predefined set of hyperparameter values. Hyperparameter-based machine learning algorithms are frequently employed with it

The objective of the project is to improve the predictive power of your recommender system by optimizing its hyperparameters using GridSearchCV. Making recommendations for skincare products based on user input may become more accurate and efficient as a result.

- Accuracy Score

A fundamental metric used to assess a classification model's overall correctness is its accuracy score. The ratio of accurately predicted instances to all instances in the dataset is computed. The accuracy score gives a broad overview of the recommender system's performance. It is helpful in determining whether the suggestions for skincare products are generally accurate. It might not be enough, though, in situations where there are class disparities.

- Classification Report

An in-depth assessment of a classification model's performance can be found in a classification report. It contains metrics for every class, including F1-score, recall, and precision.

The classification report helps to understand how well the recommender system works for each distinct aspect, such as various skin types or levels of acne severity. It provides information on the system's strong points and potential areas for development.

- Confusion Matrix

A table that compares expected and actual classifications to provide a clear picture of the model's performance is called a confusion matrix. A confusion matrix can be used to identify the different kinds of mistakes that the skincare recommender system makes. It may indicate, for example, whether the system frequently misclassifies particular skin types or degrees of severity. This data is essential for focused enhancements.

6.2 Evaluation

1. <u>Logistic Regression Model for Multi-Class Classification</u>

Our Logistic Regression model for multi-class classification has been fine-tuned with the best hyperparameters to improve the overall performance of the model. To further understanding, Let's break down the evaluation metrics:

```
Best Hyperparameters: {'C': 100, 'penalty': 'l2'}
Accuracy: 0.6333333333333333
Classification Report:
              precision    recall  f1-score   support

           1       0.90      0.75      0.82        12
           3       0.38      0.50      0.43         6
           4       0.71      1.00      0.83         5
           5       0.60      0.75      0.67         8
           6       0.78      1.00      0.88         7
           7       0.67      0.40      0.50         5
           8       0.60      0.75      0.67         4
           9       0.00      0.00      0.00         7
          10       0.43      0.50      0.46         6

    accuracy                           0.63        60
   macro avg       0.56      0.63      0.58        60
weighted avg       0.59      0.63      0.60        60
```

Figure 6. 1 Classification Report of Logistic Regression Model

Best Hyperparameters:

- In this model the chosen hyperparameters were {'C': 100, 'penalty': 'l2'}.
- The 'C' hyperparameter in logistic regression controls the regularization strength.
- The 'penalty' hyperparameter determines the type of regularization to be applied. In this model, it's set to 'l2', indicating L2 regularization.

Accuracy:

The overall accuracy of the model is calculated to be approximately 63.33%. According to this model, 63.33% of the proportion correctly predicted among all instances in the dataset. In general, accuracy above 50% can be considered as that model is performing better than random chance. But only using accuracy, we cannot tell the performance of the model especially when dealing with imbalanced classes.

Precision:

Precision measures the accuracy of the positive predictions. As an example, precision for class 1 is 90% which means that 90% of instances that are predicted as class 1 are correct. This is a high precision value; the model is reliable when finding instances of class

1.

Recall:

Measures the ability of the model to capture all relevant instances of a class. Class 3 has a recall of 50%, which means that only half of actual instances in class 3 were identified by model. Improvement of recall is needed for better performance.

F1-score:

It provides a balanced measure that considers both false positives and false negatives. When considering class 4, it has a good balance between precision and recall.

Macro and Weighted Averages:

The macro-average gives equal weight to each class, while the weighted average considers class imbalance.

Macro Avg:

The macro-average F1-score is 58%, providing an overall measure of the model's ability to perform well across all classes. This low macro-average F1-score indicates that the model's performance varies across different skincare product categories

Weighted Avg:

The weighted-average F1-score of 60% considers class imbalance, giving more weight to classes with larger support. This low weighted-average F1-score implies that the model is not effectively handling the imbalance in the number of users across different skincare product categories.

To gain insights into the model's performance concerning confusion matrix was plotted as follows:

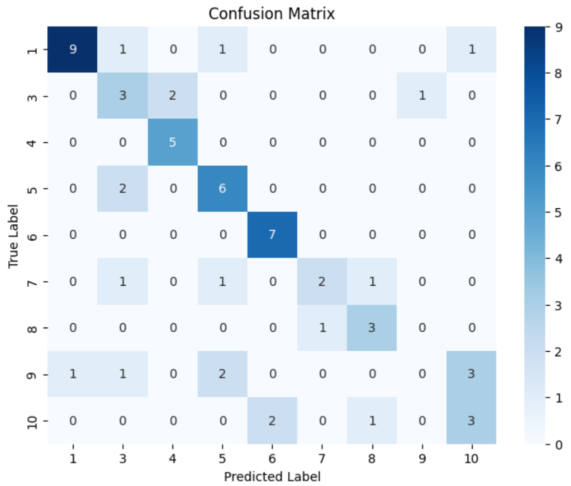

Figure 6. 2 Confusion Matrix of Logistic Regression Model

- Each element on the diagonal represents the number of instances correctly classified for the corresponding class. For example, the element in the first row and first column indicates that 9 instances of class 1 were correctly classified.

- Off-diagonal elements indicate misclassifications. For example, the element in the first row and second column indicates that 1 instance of class 1 was misclassified as class 3. In this model there are comparatively high number of misclassifications.
- The intensity of the color is proportional to the count in each cell, where darker colors indicate higher counts and lighter colors indicate lower counts. According to the above confusion matrix, most of the cells in the diagonal are in lighter shades which means lot of classes contain lower instances.
- Overall performance of the model was 0.633 by getting summation of the diagonal elements and dividing it by the total number of instances.

With the consideration of difference metrices, overall performance of this logistic regression model is low due to higher misclassifications.

2. <u>K-Nearest Neighbors (KNN) model for multi-class classification</u>

Our K-Nearest Neighbors (KNN) model for multi-class classification has undergone fine-tuning, resulting in the selection of the following hyperparameters to enhance overall performance:

```
Best Hyperparameters: {'n_neighbors': 10, 'p': 1, 'weights': 'distance'}
Accuracy: 0.8
Classification Report:
              precision    recall  f1-score   support

           1       0.82      0.75      0.78        12
           3       0.83      0.83      0.83         6
           4       0.83      1.00      0.91         5
           5       0.80      1.00      0.89         8
           6       0.86      0.86      0.86         7
           7       0.83      1.00      0.91         5
           8       0.80      1.00      0.89         4
           9       1.00      0.14      0.25         7
          10       0.62      0.83      0.71         6

    accuracy                           0.80        60
   macro avg       0.82      0.82      0.78        60
weighted avg       0.83      0.80      0.77        60
```

Figure 6. 3 Classification Report of KNN Model

Best Hyperparameters:

- The chosen hyperparameters for the KNN model are {'n_neighbors': 10, 'p': 1, 'weights': 'distance'}.
- The 'n_neighbors' hyperparameter indicates the number of neighbors to consider when making predictions. In this model, it's set to 10, which means that it considers the 10 nearest neighbors.
- The 'p' hyperparameter, set to 1, indicates the use of Manhattan distance (L1 distance) as the distance metric for determining neighbors
- The 'weights' hyperparameter, set to 'distance', implies that closer neighbors have a higher influence on the prediction.

Accuracy:

The overall performance of the KNN model is calculated to be 80%. This indicates that 80% of the instances in the dataset are correctly predicted, suggesting a relatively good overall performance.

Let's break down the evaluation metrics using the classification report:

Precision:

- Precision measures the accuracy of positive predictions.

- For instance, class 1 has a precision of 82%, meaning 82% of instances predicted as class 1 are correct. Similar high precision values are observed for other classes.

Recall:

- Recall measures the ability of the model to capture all relevant instances of a class.
- Class 9 has a recall of 14%, indicating that only a small proportion of actual instances in class 9 were identified by the model. Improvements in recall are needed for better performance in this class. Rathe rthan class 9, thera are higher recall values in other classes.

F1-score:

- The F1-score provides a balanced measure that considers both false positives and false negatives.
- Classes 4, 5, and 7 have high F1-scores, indicating a good balance between precision and recall.

Macro and Weighted Averages:

- The macro-average F1-score is 78%, indicating the model's ability to perform well across all classes.
- The weighted-average F1-score of 77% considers class imbalance, suggesting that the model is handling the imbalance in the number of instances across different classes effectively.

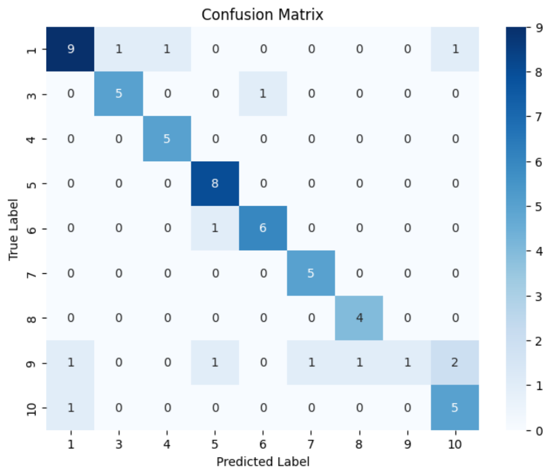

Figure 6. 4 Confusion Matrix of Logistic KNN Model

- Diagonal Elements: Each element on the diagonal represents the number of instances correctly classified for the corresponding class. For example, the element in the first row and first column (9) indicates that 9 instances of class 1 were correctly classified.
- There are certain number of misclassifications that indicates in off diagonal elements
- The color intensity in this model is better than logistic regression model. Most of the cells in the diagonal contain darker shades, that means it contains a higher count of instances in each class.

In summary, the KNN model demonstrates a promising overall performance with room for improvement in specific classes, particularly in terms of recall for class 9.

3. <u>Decision Tree model for multi-class classification</u>

   Our Decision Tree model for multi-class classification has undergone optimization, resulting in the following evaluation metrics:

Decision Tree Accuracy: 0.85
Decision Tree Classification Report:

	precision	recall	f1-score	support
1	1.00	0.75	0.86	12
3	1.00	0.83	0.91	6
4	0.83	1.00	0.91	5
5	0.73	1.00	0.84	8
6	0.86	0.86	0.86	7
7	1.00	1.00	1.00	5
8	0.67	1.00	0.80	4
9	1.00	0.43	0.60	7
10	0.75	1.00	0.86	6
accuracy			0.85	60
macro avg	0.87	0.87	0.85	60
weighted avg	0.89	0.85	0.84	60

Figure 6. 5 Classification Report of Desicion Tree Model

Accuracy:

The overall accuracy of the Decision Tree model is 85%. This implies that 85% of the instances in the dataset are correctly predicted, suggesting a strong overall performance.

Precision:

Notably, classes 1, 3, 7, and 9 have high precision values, indicating that the model is reliable when predicting instances of these classes. Generally, there are higher precision values in every class.

Recall:

Class 9 has a recall of 43%, indicating that less than half of the actual instances in class 9 were identified by the model. Improvement in recall for this class may be necessary.

F1-score:

High F1-scores are observed across various classes, highlighting a good balance between precision and recall.

Macro and Weighted Averages:

- The macro-average F1-score is 85%, indicating the model's ability to perform well across all classes.
- The weighted-average F1-score of 84% considers class imbalance, suggesting that the model is effectively handling the imbalance in the number of instances across different classes.

Confusion matrix:

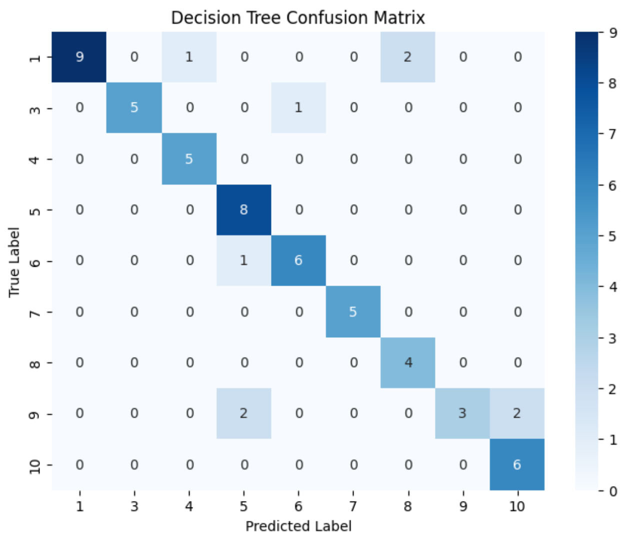

Figure 6. 6 Confusion Matrix of Decision Tree Model

- Each element on the diagonal represents the number of instances correctly classified for the corresponding class. For example, the element in the first row and first column indicates that 9 instances of class 1 were correctly classified.
- Off-diagonal elements indicate misclassifications. For instance, the element in the first row and third column suggests that 1 instance of class 1 was misclassified as class 4. The misclassification cells in decision tree are lower than both logistic regression and decision tree models.
- The color intensity in diagonal cells is much darker due to higher count of instances in each class
- The overall performance of the Decision Tree model can be assessed by summing up the diagonal elements and dividing by the total number of

instances. In this case, the sum of (9 + 5 + 5 + 8 + 6 + 5 + 4 + 3 + 6) divided by the total number of instances (60) is 0.85.

In summary, the Decision Tree model exhibits a strong overall performance with high accuracy and balanced F1-scores. While precision is generally high, there is room for improvement in recall for class 9. Further details on hyperparameters would provide additional insights into the model's behavior and potential areas for fine-tuning.

4. <u>Support Vector Machine (SVM) model for multi-class classification</u>

```
Best Hyperparameters: {'C': 1, 'coef0': 1.0, 'degree': 4, 'gamma': 'scale', 'kernel': 'poly'}
Accuracy: 0.85
Classification Report:
              precision    recall  f1-score   support

           1       1.00      0.75      0.86        12
           3       0.83      0.83      0.83         6
           4       0.83      1.00      0.91         5
           5       0.80      1.00      0.89         8
           6       0.86      0.86      0.86         7
           7       0.83      1.00      0.91         5
           8       0.80      1.00      0.89         4
           9       1.00      0.57      0.73         7
          10       0.71      0.83      0.77         6

    accuracy                           0.85        60
   macro avg       0.85      0.87      0.85        60
weighted avg       0.87      0.85      0.85        60
```

Figure 6. 7 Classification Report of SVM Model

Best Hyperparameters:

The hyperparameters selected for the SVM model are {'C': 1, 'coef0': 1.0, 'degree': 4, 'gamma': 'scale', 'kernel': 'poly'}.

- C: The regularization parameter, controlling the trade-off between achieving a low training error and a low testing error.
- Coef0: The independent term in the kernel function.
- Degree: The degree of the polynomial kernel function.
- Gamma: The kernel coefficient for 'rbf', 'poly', and 'sigmoid'.

- Kernel: The type of kernel used in the SVM. In this case, it's a polynomial kernel.

Accuracy:

The overall accuracy of the SVM model is 85%. This implies that 85% of the instances in the dataset are correctly predicted, indicating a strong overall performance.

Classification Report:

Let's break down the evaluation metrics using the classification report:

Precision:

Classes 1, 4, 5, 6, and 7 have high precision values, indicating that the model is reliable when predicting instances of these classes. Generally, precision values are higher in all classes.

Recall:

Class 9 has a recall of 57%, indicating that more than half of the actual instances in class 9 were identified by the model. Improvement in recall for this class compared to the previous models.

F1-score:

High F1-scores are observed across various classes, suggesting a good balance between precision and recall.

Macro and Weighted Averages:

The macro-average F1-score is 85%, indicating the model's ability to perform well across all classes.

The weighted-average F1-score of 85% considers class imbalance, suggesting that the model is effectively handling the imbalance in the number of instances across different classes.

To gain insights into the model's performance concerning confusion matrix was plotted as follows:

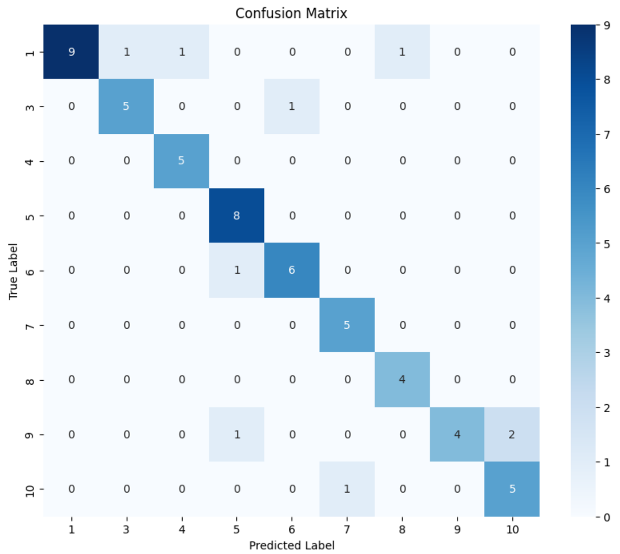

Figure 6. 8 Confusion Matrix of SVM model

Diagonal Elements: Each element on the diagonal represents the number of instances correctly classified for the corresponding class. For example, the element in the second row and second column indicates that 5 instances of class 3 were correctly classified.

- Off-diagonal Elements: These elements indicate misclassifications. The misclassifications in this model are comparatively low when compared to the above models.
- Color Intensity of the cells in the diagonal darker than previous models. It implies that there is a higher count of instances in each class.
- The overall performance of the SVM model can be assessed by summing up the diagonal elements and dividing them by the total number of instances. In this case, it would be 0.85.

In summary, the SVM model exhibits a strong overall performance with high accuracy and balanced F1-scores. The selected hyperparameters, including the polynomial kernel with a degree of 4, contribute to the model's effectiveness in capturing patterns in the data.

5. <u>Random Forest model for multi-class classification</u>

   Our Random Forest model for multi-class classification has been fine-tuned with the following hyperparameters, leading to the provided evaluation metrics:

```
Best Hyperparameters: {'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 5, 'n_estimators': 200}
Accuracy: 0.8666666666666667
Classification Report:
              precision    recall  f1-score   support

           1       1.00      0.75      0.86        12
           3       0.83      0.83      0.83         6
           4       0.83      1.00      0.91         5
           5       0.89      1.00      0.94         8
           6       0.88      1.00      0.93         7
           7       1.00      1.00      1.00         5
           8       0.67      1.00      0.80         4
           9       1.00      0.43      0.60         7
          10       0.75      1.00      0.86         6

    accuracy                           0.87        60
   macro avg       0.87      0.89      0.86        60
weighted avg       0.89      0.87      0.86        60
```

Figure 6. 9 Classification Report of Random Forest Model

Best Hyperparameters:

The selected hyperparameters for the Random Forest model are {'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 5, 'n_estimators': 200}.

- Max Depth: The maximum depth of the trees in the forest. In this case, it is set to None, allowing nodes to expand until they contain fewer than the minimum samples required for a split.
- Min Samples Leaf: The minimum number of samples required to be at a leaf node.
- Min Samples Split: The minimum number of samples required to split an internal node.
- N Estimators: The number of trees in the forest.

Accuracy:

The overall accuracy of the Random Forest model is approximately 86.67%. This implies that 86.67% of the instances in the dataset are correctly predicted, indicating a strong overall performance.

Precision:

- Precision measures the accuracy of positive predictions.
- High precision values are observed across various classes, indicating that the model is reliable when predicting instances of these classes. All the precision values are relatively high.

Recall:

- Class 9 has a recall of 43%, indicating that less than half of the actual instances in class 9 were identified by the model. Similar to other models, improvement in recall for this class may be necessary. Other classes contain high recall values

F1-score:

- The F1-score provides a balanced measure that considers both false positives and false negatives.
- High F1-scores are observed across various classes, suggesting a good balance between precision and recall. Better F1- Scores can be seen in this model than previous models.

Macro and Weighted Averages:

- The macro-average gives equal weight to each class, while the weighted average considers class imbalance.
- The macro-average F1-score is approximately 86%, indicating the model's ability to perform well across all classes.
- The weighted-average F1-score of approximately 86% considers class imbalance, suggesting that the model is effectively handling the imbalance in the number of instances across different classes.

Both macro and weighted averages are high, so it suggests that the skincare product recommendation model is robust and effective in catering to various user preferences.

Confusion Matrix:

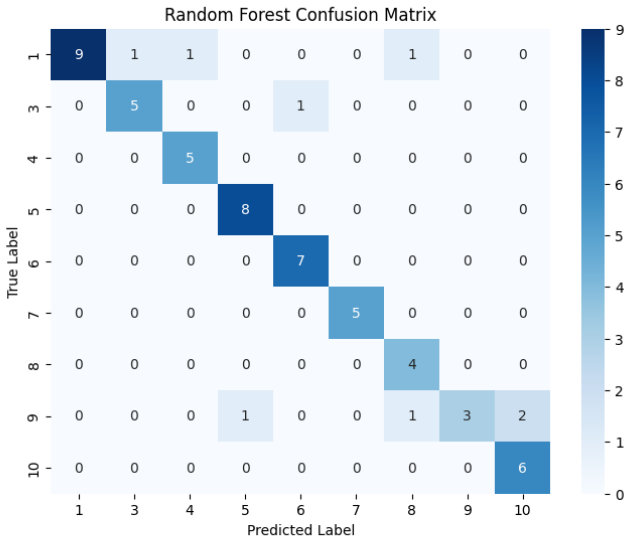

Figure 6. 10 Confusion Matrix of Random Forest model

- Each element on the diagonal represents the number of instances correctly classified for the corresponding class. For example, the element in the first row and first column indicates that 9 instances of class 1 were correctly classified.
- There are only 6 off-diagonal cells that are misclassified.
- All the diagonal cells are in the 5-9 color intensity range.
- The overall performance of the Random Forest model can be assessed by summing up the diagonal elements and dividing them by the total number of instances. In this case, it would be 0.87.

Random Forest model demonstrates a strong overall performance with high accuracy and balanced F1-scores. This model has achieved the highest accuracy among the models discussed. The selected hyperparameters contribute to the model's effectiveness in capturing complex patterns in the data, and overall, the model performs well across various evaluation metrics. This model will give accurate product to the customer by considering their concerns.

Chapter 07: Conclusion and Recommendation

7.1 Summary of Findings:

In the process of trying to make the beauty culture sector in Sri Lanka better, we found that it is quite helpful to use innovative technology to analyze consumer behavior. We discovered that our suggested approach, which anticipates consumer preferences and makes tailored product recommendations, has a distinct edge over other systems through analysis of current ones. In the beauty industry, using machine learning for segmentation and suggestions can increase commercial success.

7.2 Implications of the Study:

This study has significant implications for companies in the beauty culture industry since it gives them a useful tool for understanding and interacting with their wide range of customers. Beauty businesses may increase customer satisfaction, improve sales, and create long-term connections by utilizing segmentation strategies and personalized recommendations. The report emphasizes how crucial it is to implement technology-driven solutions to adjust to the evolving circumstances of Sri Lanka's beauty culture sector.

7.3 Limitations of the Study:

Furthermore, the completeness and quality of the data collected may impact how accurate the recommendations are, highlighting the need for continuous data quality monitoring. The study assumes that previous behavior predicts future preferences and mostly depends on historical data. The model might not carefully consider outside variables such as abrupt changes in consumer trends or the economy. There were ethical concerns of collecting sensitive data. Furthermore, the completeness and quality of the data collected may have an impact on how accurate the recommendations are, highlighting the necessity of continuous data quality monitoring.

7.4 Suggestions for Future Research:

In the beauty culture industry, future research might investigate the integration of emerging technologies such as augmented reality (AR) for virtual try-ons and sentiment analysis for real-time customer feedback, to further improve the efficacy of customer segmentation and product recommendation systems. Future research directions include examining how cultural quirks affect consumer choices and implementing dynamic pricing methods that adjust to demand fluctuations. Furthermore, follow-up studies may shed light on how customized recommendations affect client loyalty and business performance over a lengthy period of time.

7.5 Conclusion:

In conclusion, this project aims to establish an innovative approach in Sri Lanka's beauty culture sector. The results highlight how consumer segmentation, predictive modeling, and customized recommendations can influence business strategy and customer interaction in the future. Businesses may meet the difficulties of a changing industry, create enduring connections, and guarantee long-term success in the ever-evolving beauty culture scene by adopting these technology breakthroughs. For Sri Lankan beauty companies, this study represents a first step towards a more based on data and customer-focused future.

References

[1] D. S. T. S. a. S. D. A. Kothari, "Cosmetic Skin Type Classification Using CNN With Product Recommendation," in *12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Kharagpur, India., 2021 .

[2] G. Lee, "A Content-based Skincare Product Recommendation," in *Earlham College Computer Science Senior Capstone*, New York, NY, USA, 2020.

[3] R. F. R. A. K. Chaira Qalbyassalam, "Skincare Recommender System Using Neural Collaborative Filtering with Implicit Rating," in *2022 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*, 2022.

[4] V. M. M. S. P. T. S. a. M. D. C. J. Akshya, "Efficient Net-based Expert System for Personalized Facial Skincare Recommendations," in *2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, 2023.

[5] Y.-L. C. M.-H. H. H.-C. K. Yi-Fan Wang, "A personalized recommender system for the cosmetic business," *Expert Systems with Applications,* vol. 26, no. 3, pp. 427-434, 2024.

[6] T.-Y. L. S.-C. D. C.-H. H. a. C.-F. L. Hung-Tse Chan1, "SMART FACIAL SKINCARE PRODUCTS USING COMPUTER VISION TECHNOLOGIES," in *Proceedings, APSIPA Annual Summit and Conference 2021*, Tokyo, Japan, 2021,.

[7] H. M. W. T. C. B. Samarakoon, A. U. Ovitigala, G. A. T. T. Wijesinghe, K. M. L. P. Weerasinghe, G. K. B. V. Karunathilake and I. Weerathunga, "Facial Skincare

Product Suggestion with Product Popularity and Post Recommendation Care," in *4th International Informatics and Software Engineering Conference (IISEC)*, Ankara, Turkiye, 2023.