UC San Diego

Fit Predictions on Clothing Data

Team 6: Yingning Jia, Ketki Patankar, Joseph Yeh, Omar Khan, Zihao Feng

ECE 143 Programming for Data Analysis | March 14, 2025

UC San Diego

CONTENTS

Motivation and Objective

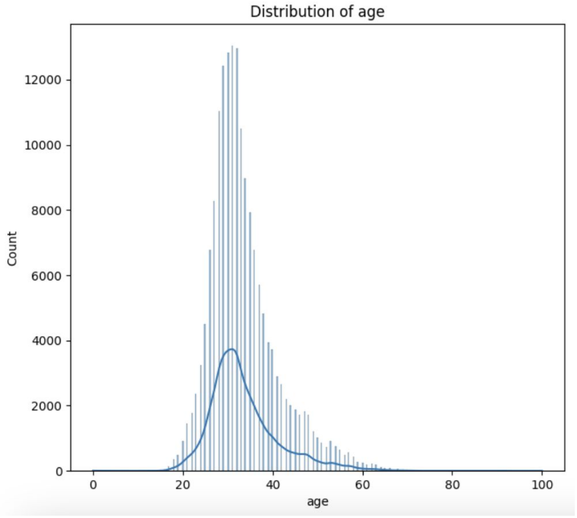
Objective

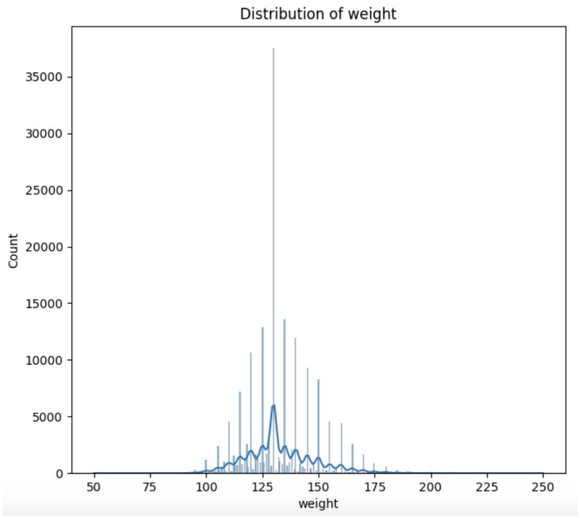
- Perform exploratory data analysis (EDA) on the Clothing Fit Dataset
- Develop a fit prediction system leveraging various machine learning algorithms
- Enhance fit prediction by integrating insights from sentiment analysis on reviews

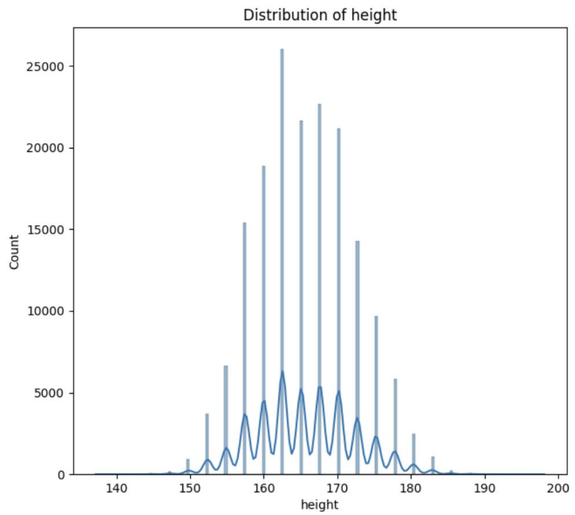
Motivation

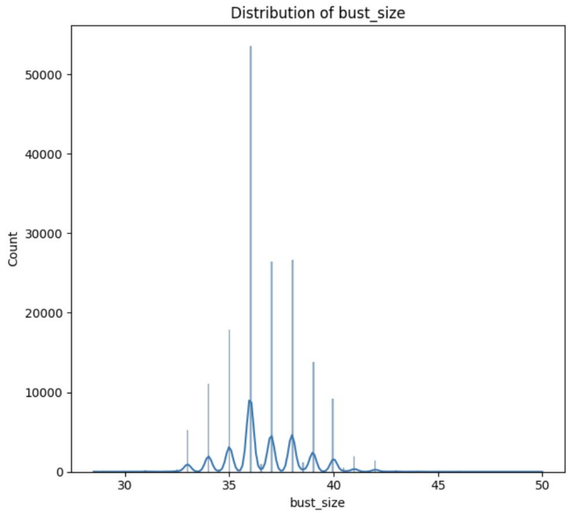
- Difficulties in choosing the right size due to inconsistent sizing across brands
- Higher return rates drive up operational costs for online retailers
- Improving customer satisfaction

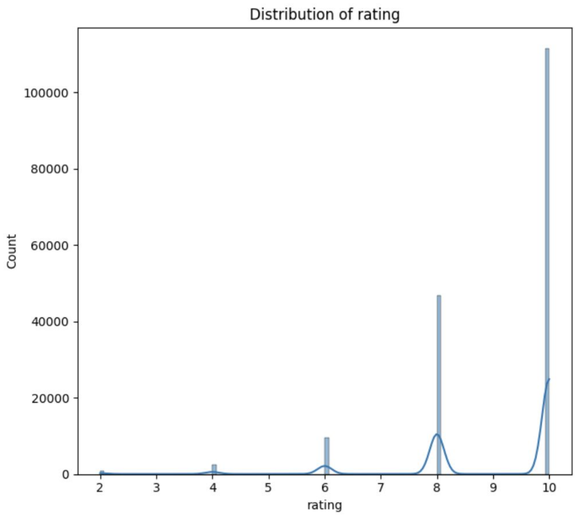
Exploratory Dataset Analysis - Distributions

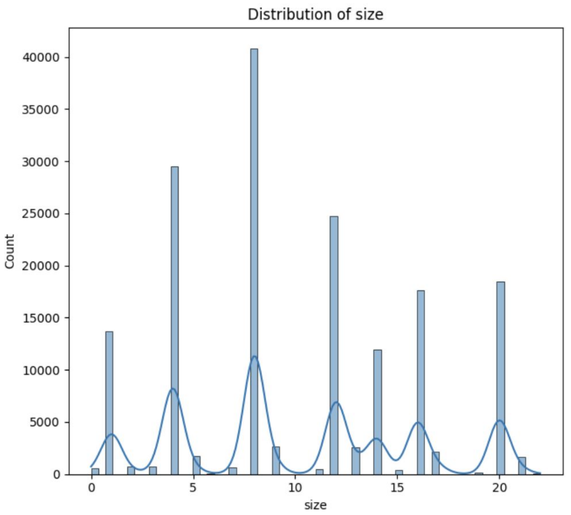




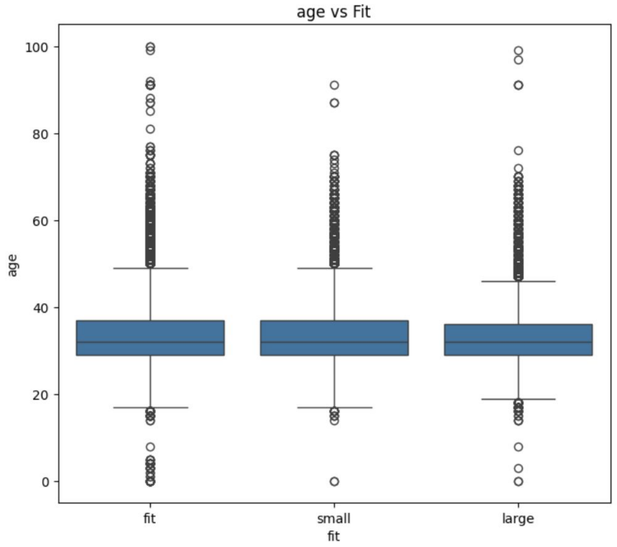


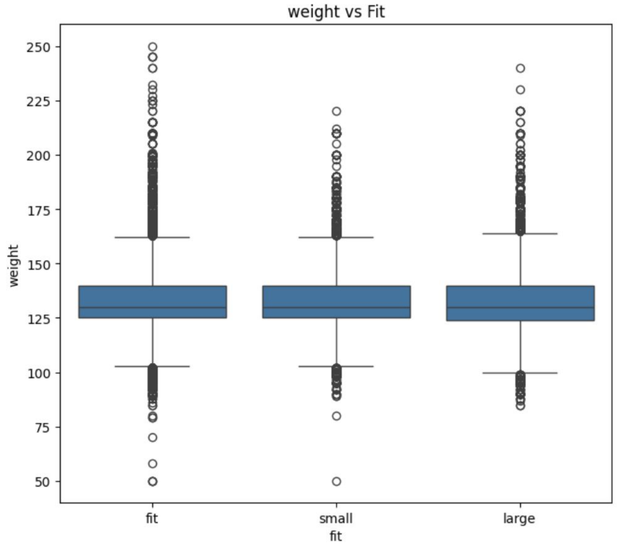


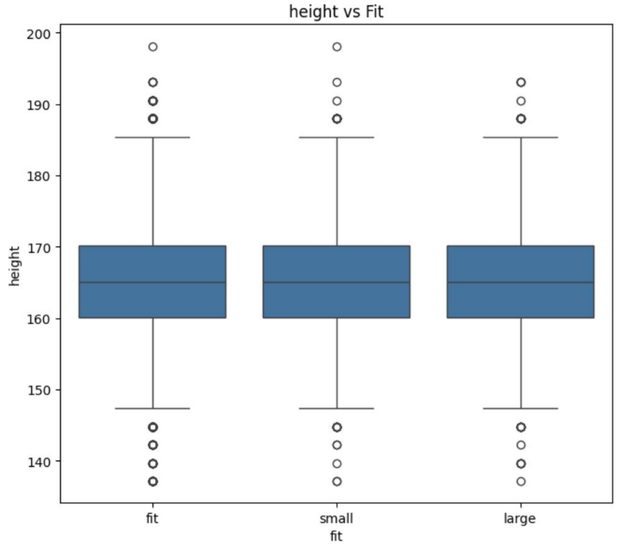


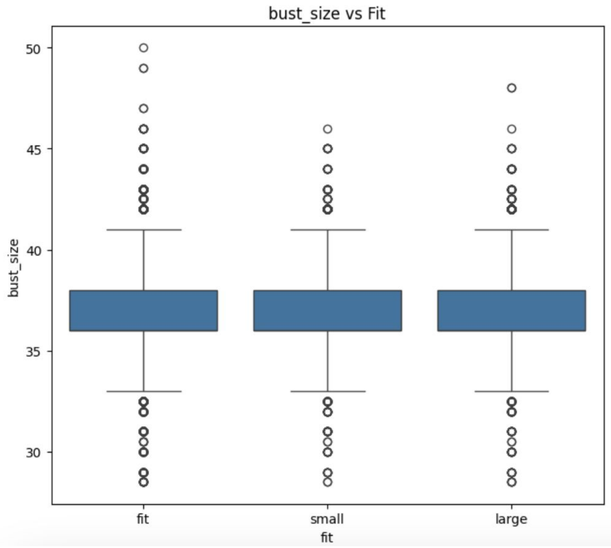


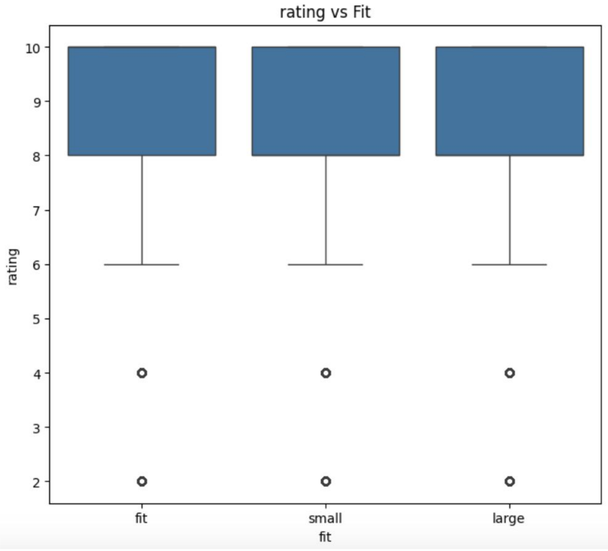
Exploratory Dataset Analysis - Box Plot

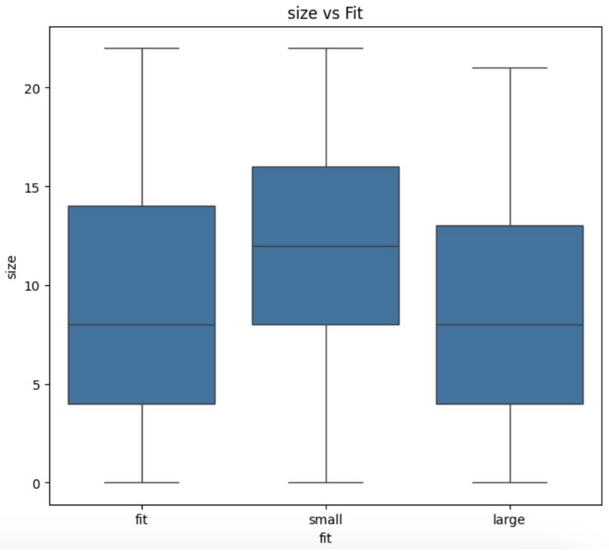




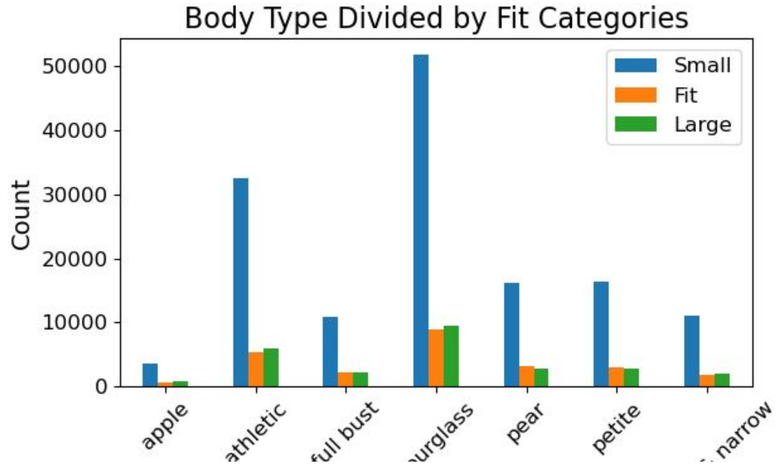


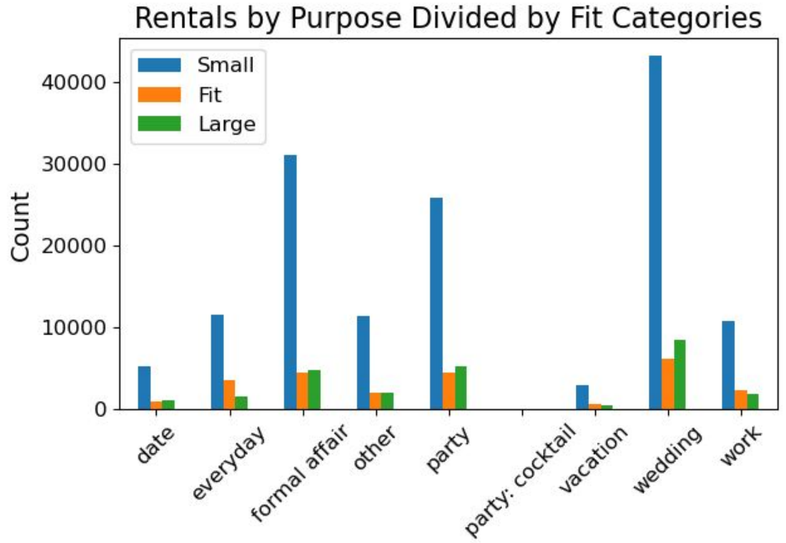


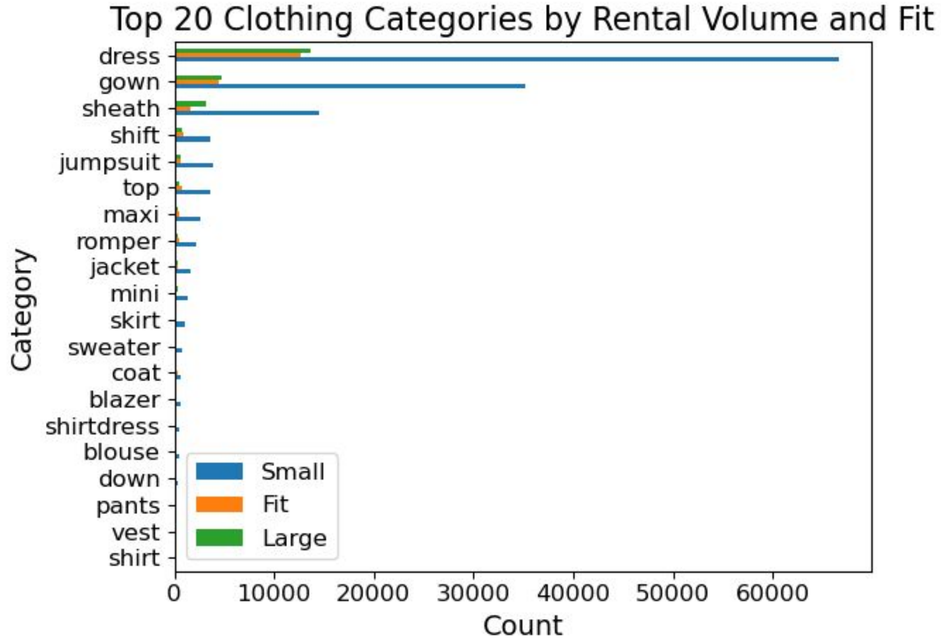


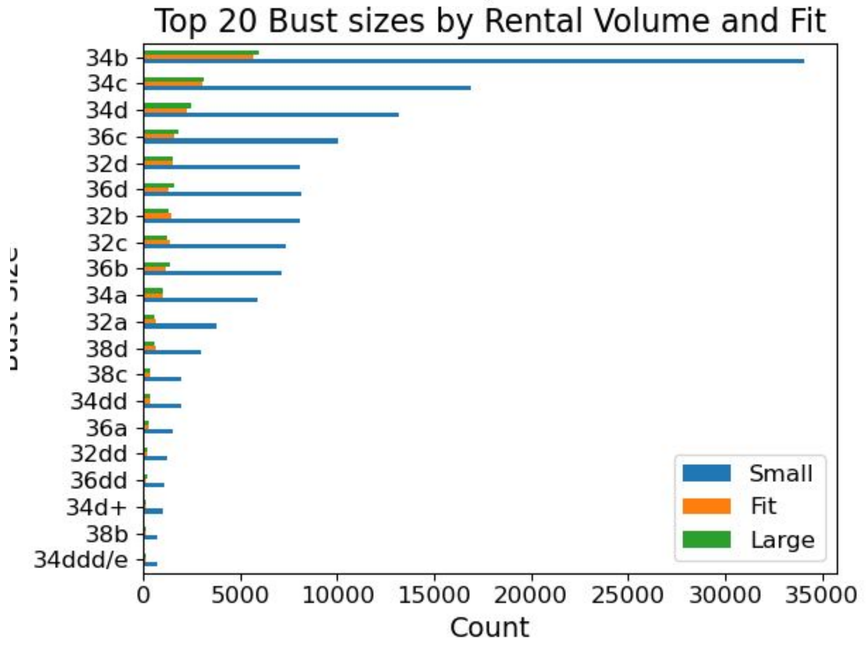


Exploratory Dataset Analysis

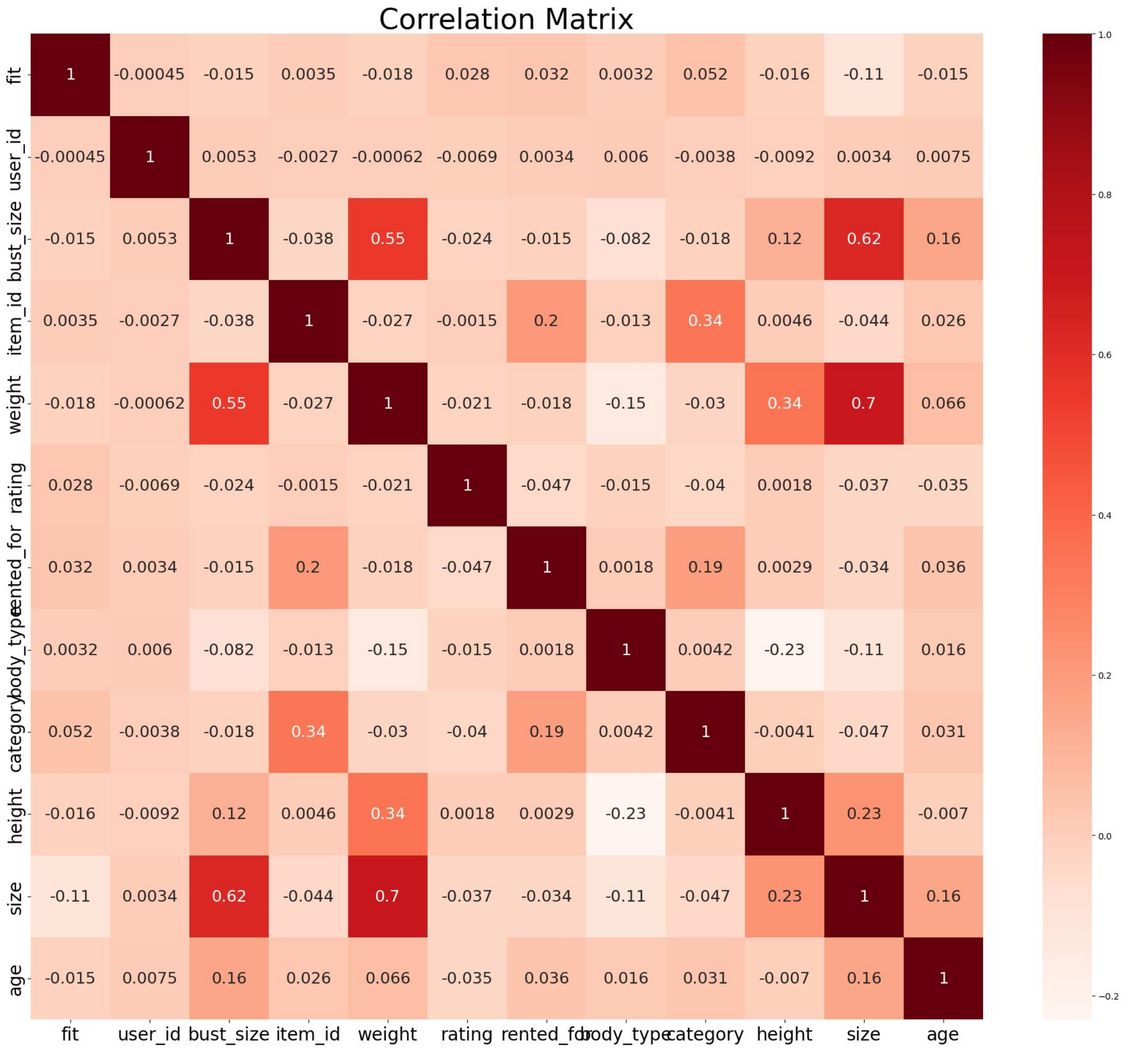








Exploratory Dataset Analysis



This heatmap shows the **Pearson correlation coefficients** between our numerical variables.

The color intensity represents the strength of the relationship, **with darker red indicating stronger correlations**.

There's a strong positive correlation between bust size and clothing size, as bust size increases, customers typically require larger clothing sizes to accommodate their proportions.

There's a strong positive correlation between weight and clothing size, indicating that heavier customers generally require larger sizes.

Training Pipeline

UC San Diego

A) Data Preparation and Cleaning

1. **Data Loading**
2. **Handling Missing Values**: Missing values are imputed using the most frequent values for categorical features.
3. **Feature Engineering**: Transform categorical features into numerical representations.
4. **Outlier Removal**: Data points with extreme values are filtered out.

B) Model Preparation and Evaluation Framework

1. **Feature Selection**: Based on correlation analysis, nine features are selected including size, category, rental purpose, and physical attributes.
2. **Feature Scaling**: Normalize the numerical features.
3. **Train-Test Split**: Data is split into training (75%) and testing (25%) sets with stratification to maintain class distribution.
4. **Evaluation Function**: A common evaluation function calculates accuracy, MSE, and F1 score, and generates confusion matrices for consistent comparison across models.

C) Model Training

1. **SVD**
2. **KNN**
3. **Random Forest**
4. **XGBoost**
5. **MLP with TF-IDF**

D) Model Comparison

Methodology - Baselines

UC San Diego

1. Latent Factor Model using Singular Value Decomposition

- capturing global patterns through dimensionality reduction, simplifying the user-item relationship into latent factors
- 'n_factors': 2, 'n_epochs': 30, 'lr_all': 0.005, 'reg_all': 0.02

2. K-Nearest Neighbors

- finding the k nearest neighbors in the feature space for a given data point based on a similarity metric
- k small, focusing on the closest neighbors, detecting small clusters data better
  k large, reducing sensitivity to noise, providing smoother decision boundaries and removing overfitting
- k = 9, 'algorithm': 'ball_tree', 'leaf_size': 40, 'n_neighbors': 40, 'p': 1, 'weights': 'uniform'

3. Random Forest

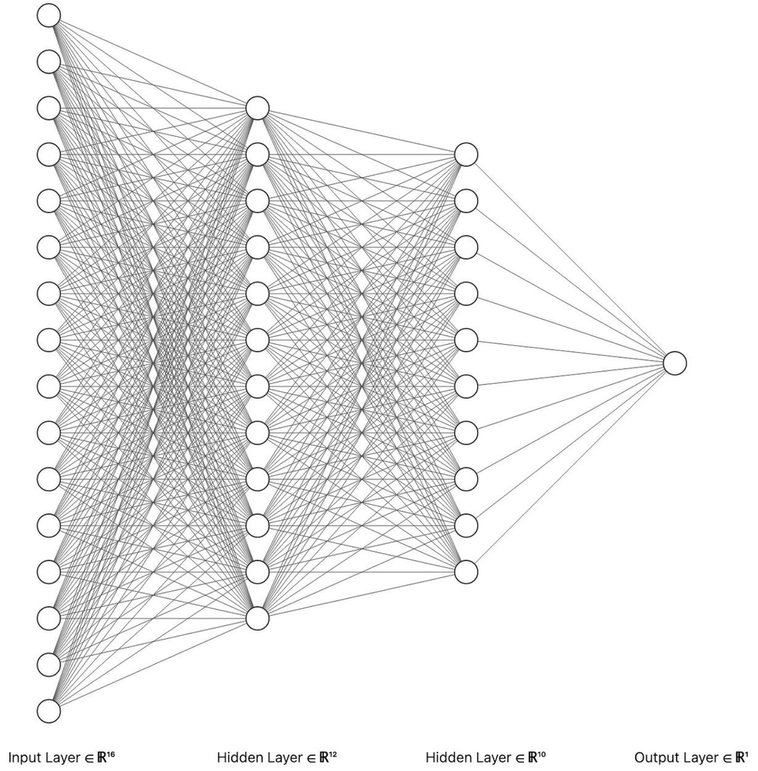
- constructing multiple decision trees and combining their predictions to make a final decision
- using a random subset to build each tree, introducing diversity and removing overfitting
- 'max_depth': 10, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 5, 'n_estimators': 100, 'n_jobs': 4, 'random_state': 42

4. XGBoost

- building an ensemble of decision trees sequentially to correct the errors of the previous ones
- using gradient descent algorithm to minimize the loss function and optimize model performance
- 'n_estimators': 300, 'max_depth': 3, 'learning_rate': 0.075, 'reg_lambda': 2, 'reg_alpha': 0.01, 'min_child_weight': 3, 'gamma': 0.5, 'colsample_bytree': 0.8

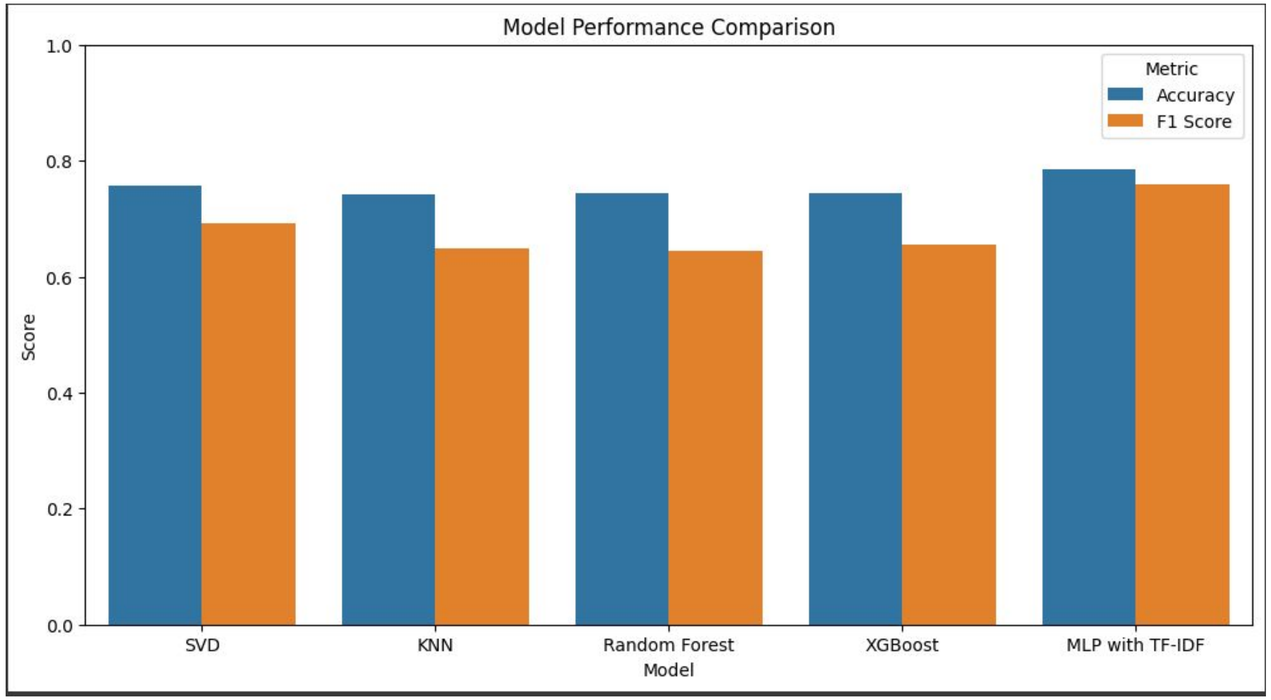
Methodology - Sentimental Analysis

- TF-IDF (Term Frequency - Inverse Document Frequency)
  - Term Frequency (TF): Measures how often a word appears in a document
  - Inverse Document Frequency (IDF): Reduce the weight of common words across all documents
- Feature Extraction with TF-IDF
  - Convert reviews text into TF-IDF vectors
  - Capture term frequency while reducing impact of common words
  - Highlights distinctive words in each review, helps MLP learn which terms correlates with positive or negative feedback
- MLP Architecture
  - Hidden layer size: [150, 100, 50]
  - Activation: ReLU
  - L2 Regularization: 0.001



Results - Comparison Table

1. Similar accuracy across all models due to class imbalance as fit labels are highly skewed
2. MLP with TF-IDF outperforms due to incorporating customer review information
3. Slightly better performance by Latent factor model with SVD indicates presence of sparse data



	SVD	KNN	Random Forest	XGBoost	MLP with TF-IDF
Accuracy	0.756253	0.742532	0.742929	0.743233	**0.785046**
F1 Score	0.691648	0.648772	0.643942	0.654646	**0.758488**

Summary and Future Scope

Summary

- MLP with TF-IDF outperformed the other models in both accuracy and F1 score, showing its effectiveness in learning from customer reviews.
- It’s important to understand the characteristics of dataset when selecting recommendation models.

Future Scope

- Fit Prediction for New Items
- Targeted Marketing using sentimental analysis
- Demand Forecasting and optimizing inventory based of size preferences
- Integrate with AR

Reference

[1] Rishabh Misra, Mengting Wan, and Julian McAuley. 2018. Decomposing fit semantics for product size recommendation in metric spaces. In Proceedings of the 12th ACM Conference on Recommender Systems (RecSys '18). Association for Computing Machinery, New York, NY, USA, 422–426.

[2] Abdul-Saboor Sheikh, Romain Guigour` es, Evgenii Koriagin, Yuen King Ho, Reza Shirvany, Roland Vollgraf, and Urs Bergmann. 2019. A deep learning system for predicting size and fit in fashion e-commerce. In Proceedings of the 13th ACM Conference on Recommender Systems (RecSys '19). Association for Computing Machinery, New York, NY, USA, 110–118.

[3] Andrea Nestler, Nour Karessli, Karl Hajjar, Rodrigo Weffer, and Reza Shirvany. 2021. SizeFlags: Reducing Size and Fit Related Returns in Fashion E-Commerce. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD '21). Association for Computing Machinery, New York, NY, USA, 3432–3440.

[4] Karl Audun Kagnes Borgersen, Morten Goodwin, Morten Grundetjern, and Jivitesh Sharma. 2024. A Dataset for Adapting Recommender Systems to the Fashion Rental Economy. In Proceedings of the 18th ACM Conference on Recommender Systems (RecSys '24). Association for Computing Machinery, New York, NY, USA, 945–950.

UC San Diego

THANK YOU!