**Project Description**

**Dataset**

The data is downloaded from Kaggle. Its name is “Customer Experience Dataset”. It is a simulated dataset, specially designed for training machine-learning models to optimize customer experience. The data has 1000 rows (excluding the column title row), corresponding to data from 1000 customers. There are 14 columns, and according to Kaggle, they can be divided into five categories:

1. Demographics: Customer\_ID (numerical, continuous), Age (numerical, continuous), Gender(text, categorical), Location (text, categorical)
2. Interaction Data: Num\_Interactions (numerical, continuous), Feedback\_Score (numerical, continuous), Products\_Purchased (numerical, continuous)
3. Behavioral Data: Products\_Viewed (numerical, continuous), Time\_Spend\_on\_Site (numerical, continuous)
4. Satisfaction and Retention: Satisfaction\_Score (numerical, continuous), Retention\_Status (text, categorical)
5. Encoding: Gener\_Encoded (numerical, catagorical), Location\_Encoded, (numerical, catagorical), Retention\_Status\_Encoded (numerical, catagorical)

**Independent and Dependent Variable**

I chose variables that are both numerical and continuous as my interested variables (excluding customer id and age). I would like to use (1) Num\_Interactions, (2) Feedback\_Score, (3) Products\_Purchased, (4) Products\_Viewed, and (5) Time\_Spend\_on\_Site (5 independent variables) to predict Satisfaction\_Score (the dependent variable).

**Research question:**

1. Which model best predicts the satisfaction score?
2. Which independent variable is most important regarding the prediction?

**Data transformation**

I fill the potential NA values in the data using the column means. And then I applied Z-score normalization to the models used in the first research question (both training and testing data use the mean and standard deviation of the training data).

**First Research Question**

I chose three models to implement: Multivariant Linear Regression, K-Nearest Neighbors for Regression (KNN), and Multivariant Polynomial Regression. I built the first two model from scratch, and I used the sklearn library to implement the last one (I chose the power of two in the polynomial transformation).The test performance was assessed by Root Mean Square Error (RMSE) score, with lower scores indicating better performance. The results were: Linear Regression: RMSE = 2.866; KNN for Regression: RMSE = 3.330; Polynomial Regression RMSE = 2.942.

Based on these results, the conclusion is that the Linear Regression model performs the best, and the Polynomial Regression model is not much worse than the Linear Regression model.

I have also checked their performance on the training dataset. The RMSE for the linear regression, KNN, and polynomial regression were respectively 2.911, 2.347, and 2.880. These were surprising results. First, for the linear regression model, the RMSE is slightly higher for the training dataset than for the testing dataset. This is surprising, since in normal cases, it is the opposite. After carefully examining the codes, I conclude that this is due to the randomness in the data split, given that there is only a small difference. Second, for the train accuracy, the ranking (from best fit to worst fit) is: polynomial, linear, KNN, while is the test accuracy, the ranking is: linear, polynomial, KNN. KNN’s rank changed from the best to the worst, due to the overfitting effect.

**Second Research Question**

In order to decide which independent variable is most important in terms of prediction, I built a Decision Tree algorithm. In a decision tree, each branch is split based on which choice of the independent variable and where to split within that variable could best reduce the variance. The variable that reduces the most variant is the most important variable in predicting the satisfaction score.

The results were as follows: 0 (index of the independent variable): 0.104, 3: 0.206, 4:0.216, 2: 0.228, 1: 0.246. Therefore, the second variable is the most important variable in terms of predicting satisfaction score, which is the “Feedback\_Score”. Also, I have tested the performance of this model as well using RMSE, and the results show that it is good, indicating that this model is trustworthy (training RMSE: 2.458, testing RMSE = 3.317).

**Limitation**

I only used a single method to access the performance of the models. Also, the data points are too small to use more complex models like neural networks or polynomial networks with a power of higher than 2.