**Final Project Report: A Model to Predict Netflix Subscription Cancellations**

**1. Dataset overview and EDA findings**

This Netflix dataset is consisted of about 3,600 subscribers with attributes: customer\_id, age, gender, subscription\_type, watch\_hours, last\_login\_days, region, device, monthly\_fee, payment\_method, number\_of\_profiles, avg\_watch\_time\_per\_day, favorite\_genre, and Churned (0 = not churn, 1 = churn).

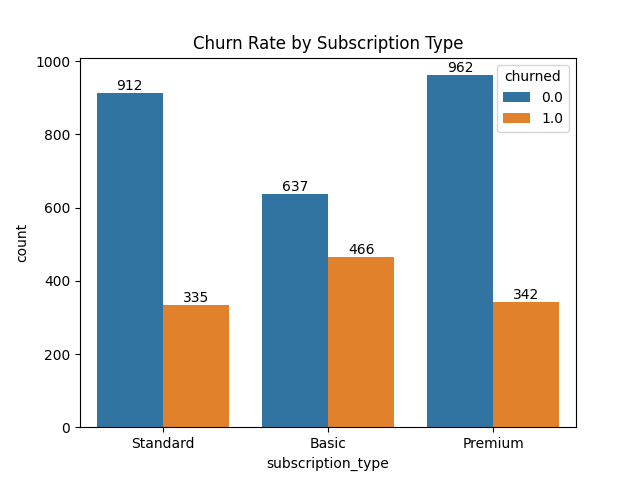
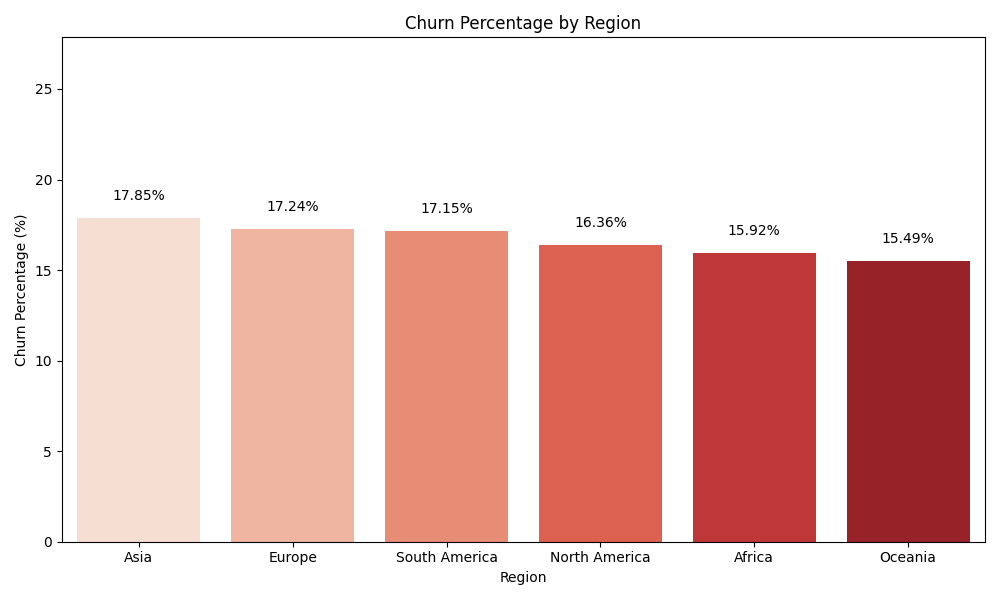
The target variable in the customer churn prediction model is whether a customer has **churned or not**.

1. **Handling missing value and cleaning up data**

* ‘watch\_hours’, ‘age’: impute missing value by using the mean.
* ‘favorite\_genre’: Fill missing values by assigning the label **"Unknown"**.
* Standardize the 'region' column by replacing values like "EUROPE" and "Europa" with "Europe".

1. **Exploratory Analysis Findings**

* **~31.28% churn rate** overall. This is a significant number; a predictive model may help identify the factors related to user cancellation.
* **Churn rate by subscription type:** Among churned users (label 1), Basic subscribers are the largest group. Meanwhile active user group (label 0), the most users are in Premium plan. Understanding the factors driving churn in each subscription tier can help improve the platform and retain users.
* **Churn percentage by region**: In the churned user group, the top three regions with the highest cancellation rates are Asia (17.85%), Europe (17.24%), and South America (17.15%).



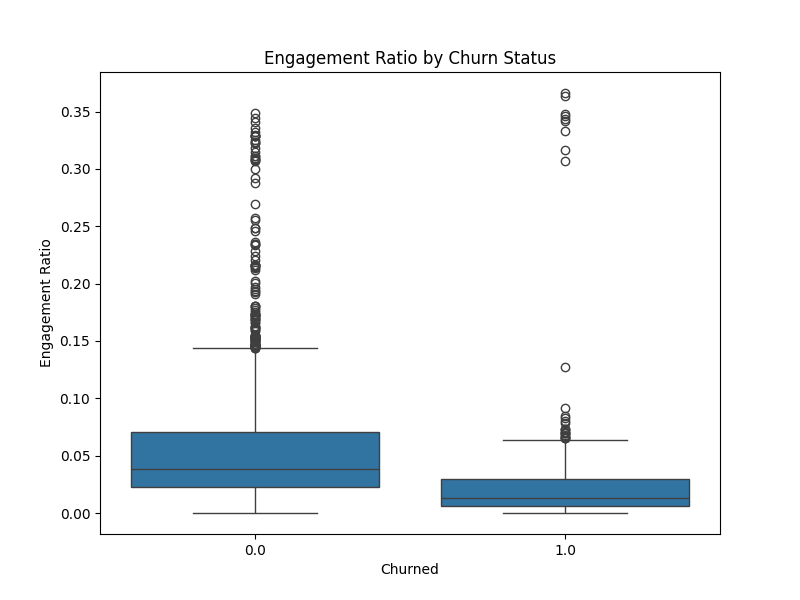
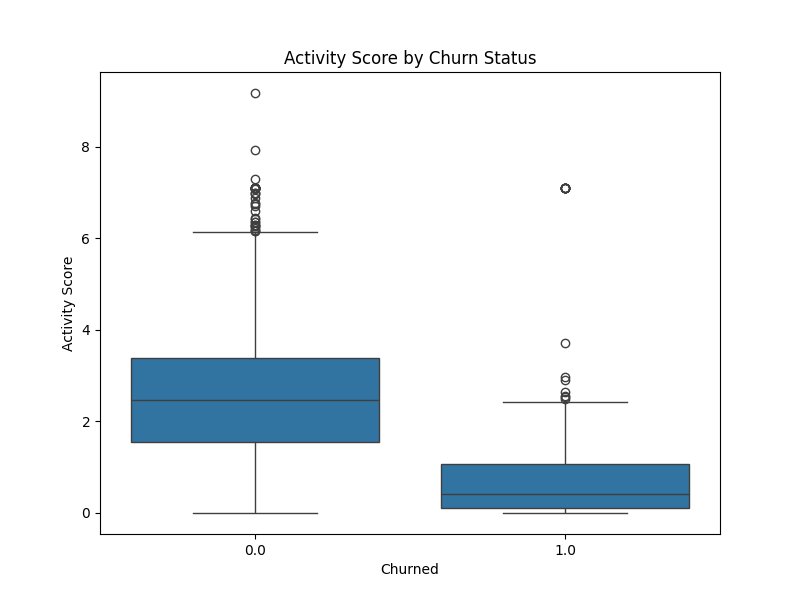
1. **Feature engineering and EDA**

From the provided features, I also performed additional feature engineering to make the data more insightful:

* **Estimated Tenure:** it represents the approximate active days within a year (365 - last\_login\_days).
* **Engagement Ratio:** it shows average daily viewing intensity (watch\_hours / estimated\_tenure).
* **Payment Frequency:** It is account usage relative to profile count (estimated\_tenure / number\_of\_profiles).
* **Activity Score:** it is a composite measure of user engagement. (watch\_hours \* avg\_watch\_time\_per\_day). To reduce skewness and improve the distribution for modeling, a log transformation was applied to the activity score.

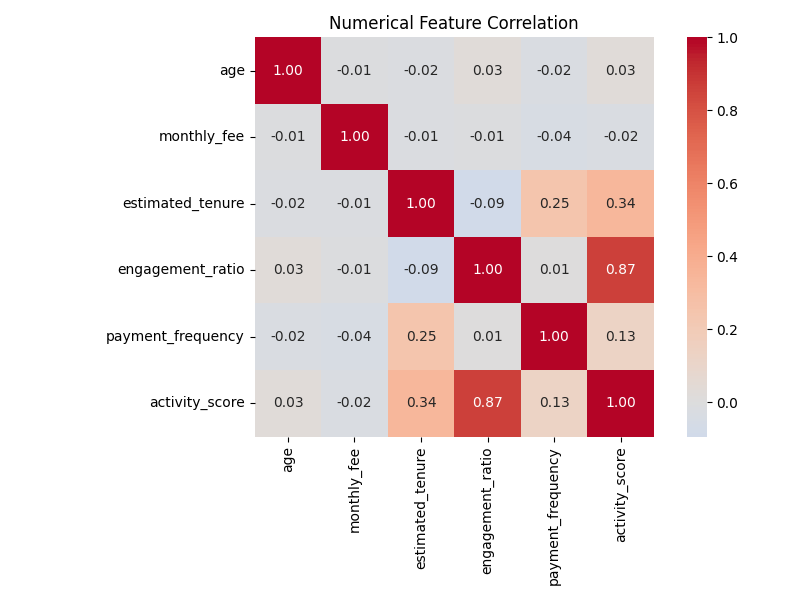
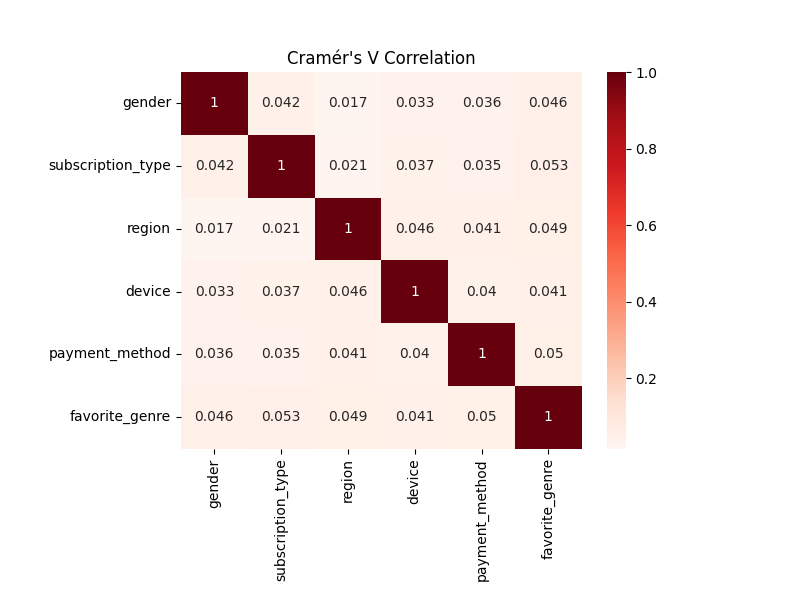
Exploratory Analysis Findings from feature engineer

* **Engagement ratio by churn:** it represents daily/weekly platform use**. Median engagement** is **higher** for **active users**, meaning they tend to watch more content relative to their estimated tenure than churned users. There are a **lot of outliers**, especially for active users—some people watch a lot more than average. In conclusion, **higher engagement ratios** may correlate with **user retention.** Users who watch more consistently are less likely to churn.
* **Activity score by churn:** it represents total time spent and depth engagement. Again, active users have higher median activity scores than churned ones. Less active users are more likely to churn, which reinforces the insight from the engagement ratio plot.



**2. Preprocessing steps**

1. **Drop target variable (Churned).**
2. **Finding feature correlation for numerical features.** Since some original columns are components of engineered features, so I keep the engineered features and drop the original columns that are fully represented inside engineered features to avoid multicollinearity.



**B**

**A**

**Result:** (A) The correlation heatmap shows that most numerical features are weakly related to each other. A strong positive correlation is seen between engagement\_ratio and activity\_score (0.87). There are moderate correlations between estimated\_tenure and both activity\_score (0.34) and payment\_frequency (0.25), suggesting that users who stay longer tend to be more active and pay more often. (B) The Cramér’s V heatmap shows very weak associations between the categorical features in the dataset, with all values below 0.06. This indicates that these categorical variables are independent of one another.

1. **Feature Encoding:** Applied one-hot encoding to categorical variables. And scaled numeric features using StandardScaler.

**4) Multicollinearity:** Dropped engagement\_ratio due to its high correlation with activity\_score.

**5) Train/Test Split:** Used an 80/20 stratified split on the churn variable to preserve class balance.

**3. Models selection and evaluation**

**1) Model Selection** For the classification task (churn vs. not churn), I selected four models that are well-suited for this problem:

* **Logistic Regression**: A simple, interpretable baseline model.
* **K-Nearest Neighbors (KNN)**: A non-parametric model that is sensitive to local data structures.
* **Random Forest**: An ensemble model that captures non-linear relationships and is robust to noise.
* **XGBoost**: A gradient boosting model known for its performance on complex datasets.

**2) Hyperparameter Tuning** To avoid underfitting and overfitting, I performed hyperparameter tuning for all four models using GridSearchCV with 5-fold cross-validation. The best combination of parameters was selected based on the ROC-AUC score.

**Result**

* **Overall metrics (positive class only, churn = 1)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **ROC-AUC** | **Precision** | **Recall** | **F1-score** |
| Logistic Regression | 0.95 | 0.83 | 0.78 | 0.8 |
| K-Nearest neighbors | 0.88 | 0.82 | 0.62 | 0.7 |
| Random Forest | 0.99 | 0.98 | 0.88 | 0.92 |
| XGBoost | 0.99 | 0.98 | 0.93 | 0.96 |

* **Classification report:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Class**  **(0 = active, 1 = churn)** | **Precision** | **Recall** | **F1-score** | **support** |
| **Logistic Regression** | 0 | 0.90 | 0.93 | 0.91 | 502 |
|  | 1 | 0.83 | 0.78 | 0.80 | 229 |
| accuracy |  |  |  | **0.88** | 731 |
| macro avg |  | 0.87 | 0.85 | 0.86 | 731 |
| weight avg |  | 0.88 | 0.88 | 0.88 | 731 |
| **K-Nearest neighbors** | 0 | 0.85 | 0.94 | 0.89 | 502 |
|  | 1 | 0.82 | 0.62 | 0.71 | 229 |
| accuracy |  |  |  | **0.84** | 731 |
| macro avg |  | 0.83 | 0.78 | 0.80 | 731 |
| weight avg |  | 0.84 | 0.84 | 0.83 | 731 |
| **Random Forest** | 0 | 0.95 | 0.99 | 0.97 | 502 |
|  | 1 | 0.98 | 0.89 | 0.93 | 229 |
| accuracy |  |  |  | **0.96** | 731 |
| macro avg |  | 0.96 | 0.94 | 0.95 | 731 |
| weight avg |  | 0.96 | 0.96 | 0.96 | 731 |
| **XGBoost** | 0 | 0.97 | 0.99 | 0.98 | 502 |
|  | 1 | 0.98 | 0.94 | 0.96 | 229 |
| accuracy |  |  |  | **0.98** | 731 |
| macro avg |  | 0.98 | 0.97 | 0.97 | 731 |
| weight avg |  | 0.98 | 0.98 | 0.98 | 731 |

**4. Conclusion and Discussion**

I evaluated four classification models to predict customer churn, focusing on key metrics: ROC-AUC, Precision, Recall, and F1-score, particularly for the positive class (churn = 1).

* XGBoost and Random Forest demonstrated superior performance across all metrics, especially in recall and F1-score, indicating they are more effective at correctly identifying churned customers.
* Logistic Regression performed well as a baseline model, offering high interpretability but slightly lower recall.
* K-Nearest Neighbors (KNN) had the lowest recall, suggesting it may miss a substantial number of churned customers—possibly due to its sensitivity to data scaling and class distribution.
* Overall, all models achieved high accuracy, with XGBoost reaching 0.98 and Random Forest 0.96. Macro and weighted averages confirm that most models maintained a good balance between classes, although KNN showed a more noticeable performance drop for the minority class (churn).
* For deployment, XGBoost is preferred for its best balance of precision and recall, minimizing false positives and negatives. If interpretability is consideration, Logistic regression is a reliable baseline.

**5. Feature insights affecting churn**

**Logistic Regression**

1. *engagement\_ratio* (-3.51): Higher engagement means a lower risk of churn.
2. *estimated\_tenure* (-2.11): Users with longer subscription tenure tend to be more loyal and less likely to churn.
3. *payment\_method\_Debit Card* (-0.85): Debit card users may be more consistent and committed payers, reflecting higher retention.
4. *subscription\_type\_Standard* (-0.73): Standard subscribers might find the value or price point satisfactory, reducing their chance of churn.
5. *payment\_method\_Crypto* (+0.65): A positive coefficient means crypto payers tend to churn more, possibly due to trial usage or lower commitment.

**Random Forest**

1. *activity\_score* (0.32): The most important feature here — it reflects user activity intensity. High activity often indicates higher satisfaction and lower churn.
2. *estimated\_tenure* (0.19) & *engagement\_ratio* (0.19): Both confirm that tenure and engagement are key predictors.
3. *payment\_frequency* (0.09): How often a user pays or interacts financially; more frequent payments may signal ongoing commitment.
4. *age* (0.025): Older users might be slightly more loyal or have different usage patterns.

**XGBoost**

1. *activity\_score* (0.25): Again the top feature, confirming its strong predictive power.
2. *estimated\_tenure* (0.097): Consistent with other models; tenure matters.
3. *payment\_method\_Crypto* (0.092): Matches the logistic regression finding — crypto payers have a higher churn risk.
4. *monthly\_fee* (0.079): Higher monthly fees may be associated with premium users who tend to stay longer or churn less.
5. *payment\_method\_Gift Card* (0.062): Indicates gift card users are somewhat predictive; they could be less loyal or more likely to churn.

In summary, all models consistently show that user activity (like **engagement ratio and activity score**) and **subscription length (tenure) are the main factors affecting churn**. Payment methods, especially crypto, stand out more in Logistic Regression and XGBoost, reflecting how different models capture different details. These differences come from how each model works and measures feature importance.