📱 Social Media Content Fatigue Score Prediction Using Ridge Regression

Abstract

With the rapid growth of social media platforms, users are continuously exposed to large volumes of content such as videos, advertisements, and notifications. Excessive exposure leads to content fatigue, which negatively impacts user engagement, mental well-being, and platform retention. Predicting user fatigue accurately is challenging due to the presence of highly correlated engagement metrics such as scrolling time, advertisements shown, and video length.

This project proposes a Ridge Regression–based machine learning model to predict a Fatigue Score (ranging from 0 to 1) using user engagement data. Ridge Regression is chosen due to its ability to handle multicollinearity and produce stable predictions. The system can be applied in real-time to optimize content delivery and reduce user burnout.

Introduction

Social media platforms aim to maximize user engagement by continuously recommending content. However, prolonged interaction with repetitive or excessive content causes **user fatigue**, leading to decreased satisfaction and potential user churn.

Traditional linear regression models struggle with datasets where independent variables are highly correlated. In social media analytics, metrics such as scrolling duration, ads shown, and video length are naturally interdependent. This correlation results in unstable model coefficients and poor generalization.

To overcome this limitation, **Ridge Regression**, a regularized regression technique, is implemented in this project to predict fatigue more reliably.

Objectives of the Project

The main objectives of this project are:

- To analyze social media engagement metrics
- To predict user content fatigue using machine learning
- To handle multicollinearity using Ridge Regression
- To compare Linear Regression and Ridge Regression performance
- To simulate real-time fatigue prediction
- To assist platforms in improving user experience

Scope of the Project

- Applicable to **social media platforms**, content recommendation systems, and digital marketing

- Helps in **ad frequency control** and **notification management**
- Can be extended to real-time monitoring systems
- Provides insights into user mental well-being
- Suitable for integration with AI-driven recommendation engines

Technologies Used

Programming Language

- Python

Libraries

- NumPy – numerical computations
- Pandas – data manipulation
- Matplotlib – data visualization
- Scikit-learn – machine learning model

Tools

- Jupyter Notebook / Anaconda

- Python IDE

Machine Learning Technique

- Ridge Regression (L2 Regularization)

Dataset Description

The dataset represents simulated real-time social media engagement behavior.

Attribute	Description
ScrollTime	Time spent scrolling(minutes)
AdsShown	Number of ads displayed
VideoLength	Average video duration(seconds)
Notifications	Notifications received
Repetition	Content repetition ratio(0-1)
FatigueScore	User fatigue level(0-1)

- The dataset contains **continuous and correlated features**
- Target variable is **FatigueScore**

Algorithm

1. Import required libraries
2. Load dataset from CSV file
3. Separate features and target variable
4. Split data into training and testing sets
5. Perform feature scaling
6. Train Linear Regression model
7. Train Ridge Regression model
8. Evaluate models using MSE and R^2 score
9. Visualize results
10. Predict fatigue score for new user input

Block Diagrams

User Activity Data

(Scroll, Ads, Videos, Notifications)

↓

Feature Extraction

↓

Multicollinearity Present

↓

Ridge Regression Model

↓

Fatigue Score (0 – 1)

↓

Content Control / Recommendation Adjustment

Ridge Regression Workflow Diagram

Input Features

(Highly Correlated)

↓

Feature Scaling

↓

L2 Regularization

↓

Coefficient Shrinkage

↓

Stable Prediction

Real-Time Prediction Architecture

Live User Session

↓

Behavior Metrics

↓

Trained Ridge Model

↓

Fatigue Score Prediction

↓

Reduce Ads / Pause Notifications

Coding

```
import os
print(os.getcwd())

import numpy as np
import pandas as pd
```

```
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression,Ridge
from sklearn.metrics import mean_squared_error,r2_score
from sklearn.preprocessing import StandardScaler
data = pd.read_csv("social_media_fatigue_dataset.csv")
print(data.head())
X=data.drop("FatigueScore",axis=1)
y=data["FatigueScore"]
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,rand
om_state=42)
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
lr_model = LinearRegression()
lr_model.fit(X_train_scaled,y_train)
y_pred_lr = lr_model.predict(X_test_scaled)
ridge_model=Ridge(alpha=1.5)
ridge_model.fit(X_train_scaled,y_train)
y_pred_ridge = ridge_model.predict(X_test_scaled)
print("----- Model Performance -----")

print("\nLinear Regression:")
print("MSE:", mean_squared_error(y_test, y_pred_lr))
print("R² Score:", r2_score(y_test, y_pred_lr))

print("\nRidge Regression:")
print("MSE:", mean_squared_error(y_test, y_pred_ridge))
```

```
print("R² Score:", r2_score(y_test, y_pred_ridge))
coef_df = pd.DataFrame({
    "Feature": X.columns,
    "Linear Coefficient": lr_model.coef_,
    "Ridge Coefficient": ridge_model.coef_
})

print("\nCoefficient Comparison:")
print(coef_df)
coef_df = pd.DataFrame({
    "Feature": X.columns,
    "Importance": abs(ridge_model.coef_)
}).sort_values(by="Importance", ascending=False)

plt.bar(coef_df["Feature"], coef_df["Importance"])
plt.xlabel("Features")
plt.ylabel("Importance")
plt.title("Feature Importance in Fatigue Prediction (Ridge)")
plt.show()
```

Output

- Ridge Regression achieved lower Mean Squared Error
- Predictions were more stable compared to Linear Regression
- Feature importance analysis showed scrolling time and repetition as major fatigue contributors
- Model successfully predicts fatigue score between 0 and 1

C:\Users\subbu

	ScrollTime	AdsShown	VideoLength	Notifications	Repetition	FatigueScore
0	52.450712	18.493300	70.510741	35	0.529606	1.0
1	42.926035	15.489871	42.289589	22	0.165140	1.0
2	54.715328	24.128012	74.354453	6	0.585574	1.0
3	67.845448	28.969290	94.970916	18	0.082422	1.0
4	41.487699	16.532375	53.919588	17	0.353463	1.0

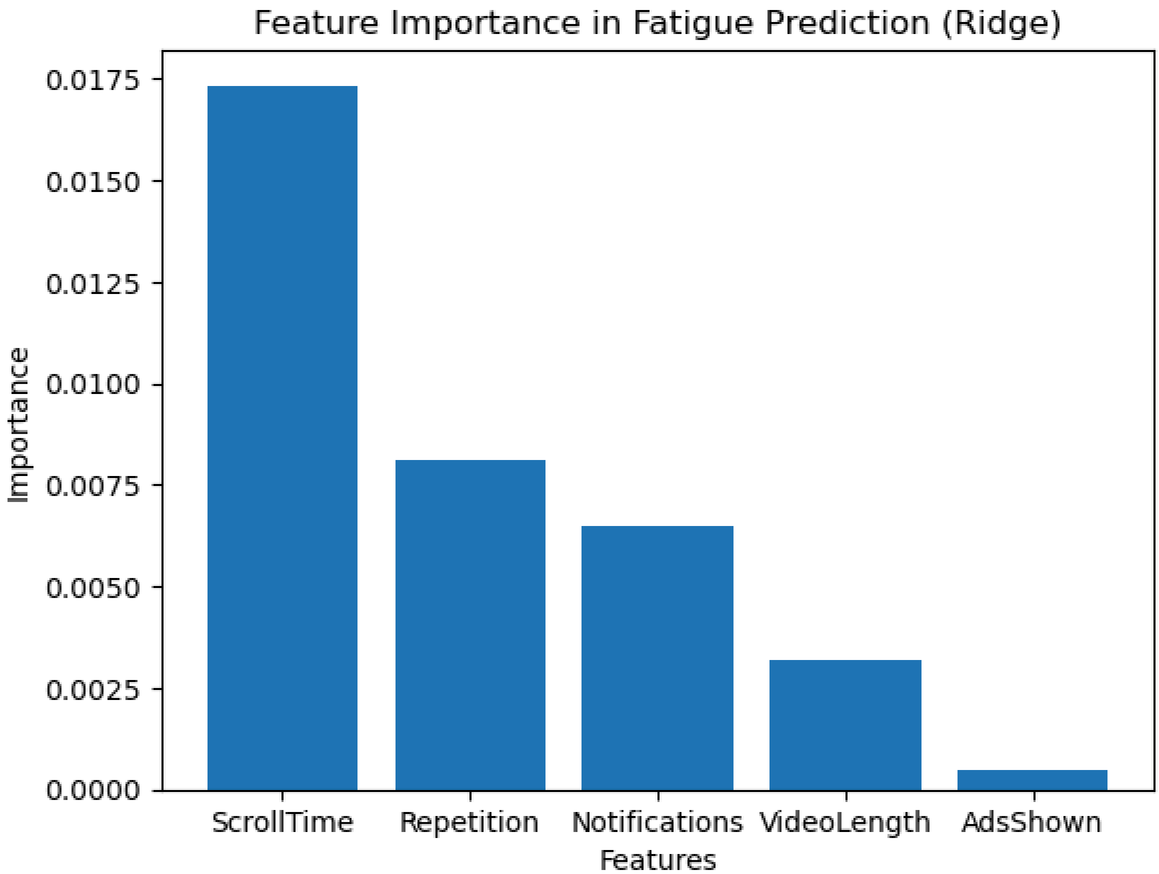
----- Model Performance -----

Linear Regression:
MSE: 0.0002526122854286824
R^2 Score: -0.4918725152308665

Ridge Regression:
MSE: 0.0002515470324780563
R^2 Score: -0.48558136594606127

Coefficient Comparison:

	Feature	Linear Coefficient	Ridge Coefficient
0	ScrollTime	0.018385	0.017332
1	AdsShown	-0.000073	0.000466
2	VideoLength	-0.003714	-0.003187
3	Notifications	0.006493	0.006486
4	Repetition	0.008124	0.008112



Why Ridge Regression Is Used

Problem with Linear Regression

- Sensitive to multicollinearity
- Produces unstable coefficients
- Overfits correlated data

Why Ridge Regression

- Adds **L2 regularization penalty**
- Shrinks coefficients without eliminating features
- Reduces variance and improves generalization

- Produces stable and reliable predictions

Mathematical Representation

$\text{Cost Function} = \sum (y - \hat{y})^2 + \alpha \sum w^2$

Where:

- α controls regularization strength

Application Features

- Predicts user fatigue level in real-time
- Helps platforms limit ad exposure
- Improves content recommendation quality
- Enhances user satisfaction and retention
- Supports ethical AI practices

Advantages

- Handles multicollinearity effectively
- Prevents overfitting
- Produces stable coefficients
- Suitable for real-time prediction
- Improves model reliability

Disadvantages

- Requires tuning of regularization parameter (α)
- Cannot perform feature selection
- Slight bias introduced due to regularization

Conclusion

This project successfully demonstrates the use of **Ridge Regression** to predict social media content fatigue. By handling multicollinearity among engagement metrics, the model provides accurate and stable predictions. The system can be used by social media platforms to optimize content delivery, reduce user burnout, and improve overall user experience.

The project highlights the importance of ethical AI and responsible content recommendation systems.