Classification Report

Model: Predict Calorie Expenditure

**\*\****\*Introduction\****\*\***

*Ushbu loyihada ovqatlanish ma'lumotlari to'plamidan foydalangan holda kaloriya iste'molini bashorat qilish uchun mashinani o'rganish usullari qo'llanildi. Loyiha ovqatlanish ma'lumotlarini tahlil qilish va modellashtirish jarayonini o'z ichiga oladi. Asosiy e'tibor ma'lumotlarni oldindan qayta ishlash, model tanlash va ishlashni baholashga qaratilgan*

**\*\****\*Development\****\*\***

-Ushbu vazifa uchun RandomForestRegressor modeli tanlangan. Model trening ma'lumotlaridan foydalangan holda o'qitildi va tasdiqlash to'plami bo'yicha bashoratlar qilindi. Modelning to'g'riligini baholash uchun ishlash o'rtacha kvadrat xato va R² score kabi ko'rsatkichlar yordamida tahlil qilindi.

**\*\****\*Conclusion\****\*\***

*Loyiha yakunida ishlab chiqilgan modelning validatsiya to‘plami bo‘yicha ishlashi baholandi va natijalar haqida hisobot berildi. Tekshirish to'plami uchun o'rtacha kvadratcha xatosi 14,61, R² balli esa 0,996 deb hisoblangan. Bundan tashqari, test natijalari xususiy va davlat uchun mos ravishda 0,06281 va 0,06201 ball berdi. Ushbu loyiha oziqlanish ma'lumotlarini tahlil qilish va kaloriyalarni bashorat qilishda mashinani o'rganish dasturlarini qanday amalga oshirish mumkinligiga misol bo'lib xizmat qiladi. Topilmalar keyingi takomillashtirish va tahlillar uchun asos bo'lib xizmat qiladi.*

Data bilan tanishuv

importpandasaspd

importnumpyasnp

importmatplotlib.pyplotasplt

importseabornassns

importmatplotlib.pyplotasplt

importwarnings

warnings.filterwarnings('ignore')

Kutubxonalarni chaqirgan

Load datas et qilganda alohida

train\_df=pd.read\_csv('/kaggle/input/playground-series-s5e5/train.csv')

test\_df=pd.read\_csv('/kaggle/input/playground-series-s5e5/test.csv')

Read and analyze data , Import data with pandas, observe the overall situation with commands such as df.info(), df.describe() and df.isnull().sum().

Data Visualization

Bunda juda kop grafik chizgan

Model Building

fromsklearn.model\_selectionimporttrain\_test\_split

fromsklearn.ensembleimportRandomForestRegressor

fromsklearn.metricsimportmean\_squared\_error, r2\_score

Training

Testing my model on train data

model=RandomForestRegressor(random\_state=42)

Evaluate the performance on the validation set

mse\_val = mean\_squared\_error(y\_val, y\_val\_pred)

r2\_val = r2\_score(y\_val, y\_val\_pred)

# Convert categorical data to numerical data

label\_encoder = LabelEncoder()

train\_df['Sex'] = label\_encoder.fit\_transform(train\_df['Sex'])

# Separate features and target variable

X = train\_df.drop(['id', 'Calories'], axis=1) # Remove 'id' and 'Calories' columns

y = train\_df['Calories'] # Target variable

# Split the training data into training and validation sets

X\_train, X\_val, y\_train, y\_val = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Initialize the model

# Train the model with training data

model.fit(X\_train, y\_train)

# Make predictions on the validation set

y\_val\_pred = model.predict(X\_val)

#

# Print results

print(f'Mean Squared Error for Validation Set: {mse\_val}')

print(f'R² Score for Validation Set: {r2\_val}')