TFG

Aplicación de modelos predictivos basados en Machine Learning para la estimación de calorías quemadas en usuarios de gimnasio

1. Análisis del Dataset

El objetivo es trabajar con un dataset que contenga los atributos físicos de los usuarios de gimnasio, no los relacionados con los entrenamientos en sí. Vamos a centrarnos en las características físicas para realizar el clustering y, posteriormente, predecir las calorías quemadas.

Atributos físicos relevantes que podemos usar para el clustering:

- **IMC (Índice de Masa Corporal):** Es crucial para clasificar a los usuarios según su salud física.
- **Porcentaje de Grasa Corporal:** Ayuda a entender la composición corporal.
- **Frecuencia Cardíaca en Reposo:** Indicador de la salud cardiovascular.
- **Edad:** Importante para segmentar según grupos etarios.
- **Altura y Peso:** Junto con el IMC, ayudan a determinar la complexión corporal.
- **Nivel de Actividad Física:** Aunque está relacionado con los entrenamientos, también puede influir en el perfil físico del usuario.

2. Preprocesamiento de los Datos

Antes de aplicar cualquier algoritmo de clustering, realizaremos el siguiente preprocesamiento:

- **Limpieza de datos:**
  - Eliminación de valores nulos o tratamiento de los mismos.
  - Verificación de valores extremos o errores en los atributos.
- **Normalización de atributos:**
  - Para garantizar que todos los atributos tengan la misma escala (por ejemplo, el IMC y la frecuencia cardíaca están en escalas diferentes), normalizaremos los valores utilizando técnicas como la **normalización min-max** o la **estandarización (Z-score)**.
- **Selección de atributos:**
  - Usaremos los atributos físicos seleccionados anteriormente. Otros atributos, como el tipo de entrenamiento o la duración del entrenamiento, se eliminarán ya que no son relevantes para el clustering.

3. Clustering de los Usuarios

Ahora que tenemos los datos listos, pasaremos a la parte del clustering. Aquí, nuestro objetivo es agrupar a los usuarios según sus características físicas. Se utilizarán los siguientes métodos de clustering:

- **K-Means:**
    - Este algoritmo es útil si sabemos cuántos grupos queremos. En este caso, nos enfocamos en obtener **aproximadamente 4 clusters**, como nos sugirió el profesor. K-Means busca minimizar la distancia entre los puntos y el centroide del cluster.
- **DBSCAN (Density-Based Spatial Clustering of Applications with Noise):**
    - Este algoritmo es útil si los datos no siguen una distribución esférica y si existe ruido (outliers). DBSCAN no requiere que el número de clusters sea predefinido.
- **Clustering Jerárquico:**
    - Este enfoque nos permite construir una jerarquía de clusters y ver cómo se agrupan los usuarios en diferentes niveles. Es útil para obtener una visión más detallada de la estructura de los datos.

4. Predicción de Calorías Quemadas

Una vez que hemos realizado el clustering, pasamos a la predicción de las **calorías quemadas** para cada grupo (cluster). La predicción se realizará en dos escenarios:

Escenario A: Predicción de Calorías según el Cluster

- En este escenario, entrenaremos un modelo de regresión para predecir las calorías quemadas basándonos en el **cluster al que pertenece cada usuario**.
- **Modelo de regresión:** Usaremos un modelo como **Regresión Lineal**, **Árboles de Decisión** o **Random Forest** para predecir las calorías quemadas en función de las características físicas de los usuarios y el cluster al que pertenecen.

Escenario B: Predicción de Calorías sin Usar Clusters

- En este escenario, entrenaremos un modelo de regresión que utilice **todos los atributos físicos** de los usuarios (sin hacer uso del clustering). Aquí, no segmentaremos a los usuarios en grupos y la predicción será realizada considerando todos los datos.

5. Comparación de los Escenarios

Una vez entrenados los modelos en ambos escenarios, compararemos los resultados para determinar cuál de ellos produce las predicciones más precisas y fiables. **El objetivo es demostrar que el Escenario A, que utiliza el clustering, dará resultados más afinados y específicos.**

6. Desarrollo de una API Básica

Para la parte final del TFG, implementaremos una **API en FastAPI** para interactuar con los modelos entrenados y hacer predicciones. La API permitirá:

1. **Ingresar los datos de un usuario (atributos físicos).**
2. **Predecir el cluster al que pertenece el usuario.**
3. **Predecir las calorías quemadas para ese usuario**, tanto en el **Escenario A** (según el cluster) como en el **Escenario B** (sin usar cluster).

Estructura de la API:

- **Endpoint 1:** /predict_cluster
    - Entrada: Atributos físicos del usuario (IMC, grasa corporal, edad, etc.).
    - Salida: El número del cluster al que pertenece el usuario.
- **Endpoint 2:** /predict_calories
    - Entrada: Atributos físicos del usuario.
    - Salida: Predicción de las calorías quemadas, en dos modalidades:
        - **Escenario A:** Basado en el cluster.
        - **Escenario B:** Basado en todos los atributos físicos.

7. Resumen Final del Proyecto

1. **Análisis y preprocesamiento de datos**: Asegurarse de que los datos estén listos para el clustering.
2. **Aplicación de técnicas de clustering**: Agrupar a los usuarios en función de sus características físicas.
3. **Entrenamiento de modelos de predicción**: Predecir las calorías quemadas, comparando dos enfoques (con y sin clusters).
4. **Desarrollo de la API**: Crear una API que permita hacer predicciones interactivas.

Con esto, tendrás un modelo que no solo te ayuda a predecir las calorías quemadas de los usuarios, sino también a segmentarlos según sus características físicas, lo que debería generar predicciones más precisas y útiles.