1 Introducción

La predicción de propiedades moleculares representa uno de los desafíos más fascinantes en la intersección entre la Inteligencia Artificial y la química computacional. En esta práctica, nos enfrentaremos al problema de predecir la collision cross section (CCS) de diferentes moléculas, una propiedad física fundamental que determina cómo interactúan las moléculas con otras partículas en su entorno.

1.1 ¿Por qué es importante?

Hoy en día, para desarrollar nuevos fármacos o compuestos químicos, los científicos deben realizar una gran cantidad de experimentos de laboratorio para caracterizar las propiedades de las moléculas. Sin embargo, estos experimentos son costosos y consumen mucho tiempo. Imagínate poder predecir estas propiedades de forma precisa usando tan solo la fórmula química de la molécula.

Más concretamente, la CCS es una propiedad fundamental en espectrometría de masas para la identificación de moléculas. Cuando los científicos analizan un tejido biológico, inicialmente desconocen qué moléculas están presentes en él. Para identificarlas, utilizan un espectrómetro de masas que mide propiedades moleculares como la CCS. Posteriormente, comparan estos valores medidos con valores de referencia existentes en bases de datos (por ejemplo, buscan qué molécula conocida tiene un CCS de 1000). Si encuentran una coincidencia, pueden inferir que la molécula encontrada en el tejido es la misma que la molécula de referencia.

El problema es que este procedimiento asume que la CCS de la molécula de referencia está en la base de datos. Para que esto ocurra, alguien tuvo que realizar un experimento de laboratorio para medir la CCS de esa molécula y, como ya hemos mencionado, este es un procedimiento caro. Si pudiéramos predecir la CCS de cualquier molécula a partir de su fórmula química, podríamos poblar bases de datos con valores predichos y acelerar el proceso de identificación de moléculas.

1.2 El desafío: predicción de CCS

1.2.1 ¿Qué es la CCS?

La Collision Cross Section (CCS) es una propiedad física que describe la probabilidad de que una molécula colisione con otras partículas en un entorno gaseoso. Cuanto mayor sea la CCS (medido en Åˆ2), mayor será la probabilidad de colisión.

1.2.2 ¿Cómo se mide la CCS en un experimento?

Durante el análisis en un espectrómetro de masas, una molécula puede combinarse con diferentes iones (como H+, Na+, etc.) mientras viaja por el instrumento. Estos iones, conocidos como “aductos”, modifican las propiedades de la molécula

original. Específicamente, la CCS de una molécula M variará dependiendo de si se ha combinado con H+ o con Na+. Por lo tanto, el tipo de aducto es un factor crucial que debemos considerar en nuestras predicciones.

1.2.3 Características disponibles en el dataset

Para facilitar el trabajo, los datos han sido preprocesados y se han extraído características útiles a partir de la fórmula química de las moléculas. El dataset contiene las siguientes columnas:

1. **Descriptores moleculares** (columnas `desc_x`): Propiedades numéricas calculadas que incluyen:
   - Peso molecular
   - Número de átomos
   - Propiedades topológicas
   - Características geométricas
   - etc.
2. **Fingerprints moleculares** (columnas `fgp_x`): Representaciones binarias de la estructura molecular, donde cada bit indica la presencia (1) o ausencia (0) de un patrón estructural específico.
3. **Aductos** (columna `adduct`): Identifica el tipo de ion que se ha combinado con la molécula original.
4. **CCS** (columna `ccs`): Variable objetivo a predecir, medida en Å^2.

2 Objetivo de la competición

El reto consiste en desarrollar un modelo de machine learning para predecir con la mayor precisión posible la *collision cross section* (CCS) de nuevas moléculas y sus aductos.

2.1 Datos disponibles

- **Conjunto de entrenamiento**: Archivo `public_train.csv` que contiene tanto las features como las etiquetas (valores CCS) para entrenar vuestro modelo.
- **Conjunto de test**: Archivo `public_test.csv` que contiene únicamente las features. Los valores CCS de este conjunto no se proporcionan y serán utilizados para evaluar vuestro modelo.

2.2 Formato de las predicciones

Debéis generar un archivo CSV con las predicciones de vuestro modelo sobre el conjunto de test. El archivo debe:

- Contener una única columna sin cabecera

- Cada fila debe contener la predicción del valor CCS correspondiente
- El orden de las predicciones debe corresponder con el orden de las muestras en `public_test.csv`

2.3 Rendimiento de los modelos

El rendimiento de vuestros modelos se evaluará utilizando la métrica MEDAE (Median Absolute Error): Esta métrica es robusta frente a valores atípicos y penaliza las desviaciones en las predicciones de manera uniforme, independientemente de la magnitud del valor CCS.

3 Entregables y Evaluación

3.1 Entregables

1. Repositorio GitHub
   - Crear un repositorio privado que incluya todo el código del proyecto
   - Todos los miembros del equipo deben figurar como colaboradores
   - Invitar al profesor (usuario: constantino-garcia) como colaborador
2. Jupyter Notebook Entregar a través del campus virtual un notebook que contenga, como mínimo, las siguientes secciones:
   - **Preprocesamiento de datos**: Documentación de todas las transformaciones y limpieza realizadas
   - **Entrenamiento y estimación del error**: Desarrollo y validación del modelo
   - **Generación de predicciones**: Proceso de obtención de las predicciones finales
   - **Conclusiones**: Análisis crítico del trabajo realizado, incluyendo:
     - Limitaciones identificadas
     - Posibles mejoras
     - Lecciones aprendidas
3. Predicciones
   - Archivo `test_preds.csv` con las predicciones del modelo sobre el conjunto de test
   - Entregar a través del campus virtual
4. Evaluación del trabajo en equipo
   - Completar la rúbrica de evaluación del trabajo en equipo
   - El enlace a la rúbrica se proporcionará más adelante
   - Los alumnos que no completen la rúbrica de todos los demás miembros del equipo no recibirán su nota de prácticas

3.2 Criterios de Evaluación

3.2.1 Documentación y justificación

- Clara documentación de cada paso del proceso
- Justificación sólida de todas las decisiones tomadas
- Análisis crítico de los resultados obtenidos

3.2.2 Calidad del código

- Legibilidad y organización del código
- Seguimiento de buenas prácticas de programación
- Eficiencia en la implementación

3.2.3 Sofisticación técnica

- Creatividad en el enfoque del problema
- Implementación de técnicas avanzadas
- Desarrollo de ideas más allá del contenido cubierto en clase

3.2.4 Rendimiento

- Precisión del modelo en el conjunto de test
- Robustez de la solución propuesta

3.2.5 Bonus por rendimiento

Los tres primeros clasificados en la competición recibirán puntos adicionales sobre su nota media de prácticas.