UNIVERSITYHACK DATATHON

Presentado por:
HAL9000

1. Introducción

En el marco del **UniversityHack 2024**, este proyecto aborda la optimización de procesos biotecnológicos para la producción de antígenos en vacunas. El proceso productivo comprende cinco fases secuenciales: preinóculo, inóculo, cultivo productivo, centrifugación y purificación adicional, con monitorización constante de parámetros críticos como temperatura y pH. Una característica innovadora del proceso es la implementación de cultivos encadenados, permitiendo la reutilización parcial del cultivo original.

Objetivos del Proyecto

- Desarrollar un modelo predictivo robusto para estimar la concentración final del antígeno
- Identificar los parámetros clave del proceso para optimizar futuras producciones

La relevancia de este trabajo trasciende el ámbito técnico, impactando directamente en la salud pública global mediante la mejora en la eficiencia de producción de vacunas. Para alcanzar estos objetivos, se ha realizado un análisis exhaustivo aplicando técnicas avanzadas de procesamiento y modelado, cuyos detalles se presentan en las siguientes secciones.

2. Análisis y Preparación de Datos

2.1. Resumen de Modificaciones por Fase

Preinóculo: Selección de los dos organismos con **pH** más bajo, eliminando datos del tercer frasco para optimizar el análisis.

Inóculo: Realización de *matching* entre datos de biorreactores pequeños y datos cinéticos, siendo esta la fase con mayor procesamiento debido a su heterogeneidad y volumen de información.

Cultivo: Implementación de matching con biorreactores y centrifugadoras correspondientes, además de la duplicación estratégica de muestras de preinóculo e inóculo para lotes parentales.

Imputación: Aplicación de *forward fill* para mediciones continuas y medias correspondientes para mediciones de final de fase, asegurando la completitud de los datos.

Estadísticas de Medición: Generación de valores mínimos, medios y máximos para todas las mediciones de cinéticos, biorreactores y centrifugadoras, enriqueciendo el conjunto de características disponibles.

Variables Temporales: Creación de columnas de duración para cada fase del proceso, proporcionando información temporal crucial para el análisis.

2.3. Análisis Detallado del Proceso

El proceso de preparación de datos comenzó con un exhaustivo análisis para comprender la naturaleza y estructura del *dataset*. Esta fase inicial fue fundamental para identificar las relaciones entre los diferentes conjuntos de datos, comprender las distintas fases del proceso y caracterizar los tipos de datos disponibles.

Durante el análisis inicial, se identificaron tres desafíos principales que caracterizaban la complejidad del *dataset*. El primero fue la **presencia de relaciones complejas** entre los datos, particularmente en los conjuntos de mediciones, donde existía información temporal pero no se especificaba el lote correspondiente. Esta situación requirió un análisis meticuloso para establecer las conexiones correctas entre mediciones y lotes.

El segundo desafío fue un **marcado desequilibrio en las muestras**, específicamente en el ***producto 1***, que presentaba un número significativamente reducido de casos. Esta disparidad planteó un reto considerable tanto para el análisis como para la posterior fase de modelización. El tercer desafío fue la **abundante presencia de valores nulos en múltiples columnas**, lo que necesitó del desarrollo de estrategias específicas de imputación.

Para maximizar el valor de los datos disponibles, se implementaron diversas **estrategias de limpieza y enriquecimiento**. Una de las más significativas fue la creación de nuevas variables que incluían:

- La duración precisa de cada fase del proceso
- Los valores mínimos, medios y máximos de las mediciones en biorreactores y centrifugadoras
- Estadísticas derivadas de los datos cinéticos

La imputación de valores faltantes se abordó con diferentes estrategias según la naturaleza de los datos. Para las mediciones continuas, se implementó el método de *forward fill* (*ffill*), mientras que, en las mediciones de final de fase, se utilizaron las medias correspondientes para completar los valores faltantes. Asimismo, el tratamiento de casos especiales requirió particular atención. En la fase de preinóculo, los valores nulos en las líneas utilizadas necesitaron un tratamiento manual específico. En la fase de cultivo, se encontraron lotes parentales que apuntaban a valores ***NaN***, situación que también exigió una intervención manual detallada.

Un aspecto crucial en la fase de cultivo fue el tratamiento de las muestras relacionadas con lotes parentales. Para estos casos, se implementó un proceso de retroceso recursivo hasta encontrar un pariente con datos válidos de preinóculo e inóculo. Una vez identificado, se procedió a duplicar estos datos, actualizando el identificador del lote para mantener la coherencia con el lote hijo correspondiente. Esta exhaustiva preparación y limpieza de datos estableció una base sólida para los análisis posteriores, garantizando la calidad e integridad de la información que se utilizaría en el desarrollo del modelo predictivo. La atención al detalle en cada fase del proceso y el tratamiento específico de cada tipo de dato fueron fundamentales para obtener un dataset robusto y adecuado para las siguientes etapas del proyecto.

3. Instrucciones de uso

El repositorio se organizó de la siguiente forma:

- **script exploración**
  - **data →** Fichero con los datos iniciales.
  - **processed_data →** Fichero con los datos ya procesados.
  - **data_cleaning_and_analysis.ipynb →** *Notebook* del pre-procesamiento.
  - **data_cleaning_and_analysis.py →** Igual que el *notebook* pero en script.
- **script predicción**
  - **tmp →** Fichero que contiene los mejores modelos.
  - **log →** Fichero que contiene los *logs* de los modelos.
  - **dataloader.py →** Script con el que cargamos los datos y los escalamos.
  - **model.py →** Script que contiene los modelos empleados.
  - **pipeline.py →** Script que contiene el Pipeline seguido.
  - **main.py →** Script central que se encarga de centralizar los demás.
  - **XAI.ipynb → Script que contiene todas las explicaciones de los modelos**

El proyecto se organiza en dos componentes principales: exploración y predicción. En la fase de exploración, los datos proporcionados por **Aggity**se procesan mediante dos implementaciones paralelas: un notebook explicativo (*data_cleaning*) y un script optimizado para rendimiento, ambos realizando las mismas operaciones con enfoques complementarios.

El procesamiento genera **dos salidas esenciales** en el directorio *processed_data*:
- Dataset limpio preparado para entrenamiento
- Documento de análisis estadístico con métricas por columna y correlaciones

El sistema de predicción se estructura en cuatro módulos principales:
- dataloader: Lectura y estandarización de datos mediante *scikit*-learn
- *model.py*: Configuración del modelo y parámetros de búsqueda
- *pipeline.py*: Implementación de validación cruzada 5-fold y *GridSearch*
- Módulos auxiliares para monitorización de rendimiento y consumo energético

El *pipeline* integra la optimización de hiperparámetros mediante *GridSearch* y la generación de predicciones, minimizando el RMSE y proporcionando una solución completa para el desarrollo y evaluación de modelos predictivos.

4. Metodología del Modelo Predictivo

4.1. Elección de la muestra de entrenamiento y validación

La selección y preparación de datos fue crucial para el éxito del modelo predictivo. Implementamos una estrategia de validación cruzada con 5 particiones (5-fold cross-validation), maximizando el uso de nuestro conjunto de datos limitado y asegurando que cada observación participara tanto en entrenamiento como en validación.

Este enfoque meticuloso en la preparación de datos, combinado con consideraciones éticas sobre equidad, fue especialmente relevante dado el impacto directo de nuestras predicciones en la producción de vacunas y la salud pública. La decisión de utilizar StandardScaler se fundamentó en la distribución de nuestros datos y su efectividad demostrada en problemas similares.
Por ende, se siguió la siguiente metodología:

- **Monitorización** constante del **rendimiento** del modelo a través de diferentes condiciones de producción, asegurando que no existieran disparidades significativas.
- **Implementación** de umbrales de confianza adaptados a cada fase del proceso productivo, permitiendo identificar predicciones potencialmente problemáticas.
- **Desarrollo** de un sistema de pesos en el ensemble que no solo optimiza el rendimiento general, sino que también mantiene la equidad predictiva entre diferentes tipos de producción.
- **Validación cruzada estratificada** para mantener la representatividad de todas las condiciones de producción en cada *fold* de entrenamiento y validación.

4.2. Tipología del modelo a desarrollar

La selección del tipo de modelo se fundamentó en una estrategia comprensiva que abarcó un amplio espectro de algoritmos de aprendizaje automático, cada uno seleccionado por sus características específicas y potencial contribución al problema en cuestión.

Modelos Lineales:

Los modelos lineales (**Regresión Lineal**, **Ridge**, **Lasso** y **ElasticNet**) se incluyeron como base comparativa y por su capacidad de capturar relaciones lineales fundamentales en los datos. La regularización L1 y L2 presente en estos modelos ayuda a prevenir el sobreajuste y seleccionar características relevantes.

Modelos Basados en Árboles:

La familia de modelos basados en árboles se seleccionó por su capacidad para manejar relaciones no lineales complejas y capturar interacciones entre variables:

- **Random Forest** y **Extra Trees**: Proporcionan robustez mediante la agregación de múltiples árboles independientes, reduciendo la varianza del modelo final.
- **Gradient Boosting**, **XGBoost**, **LightGBM** y **CatBoost**: Estos algoritmos de *boosting* secuencial se eligieron por su **c**apacidad para **construir modelos altamente precisos**, cada uno con sus propias optimizaciones específicas.
- **AdaBoost**: Se incluyó por su capacidad para **asignar mayor importancia a las observaciones más difíciles de predecir**.

Ensamble Final:

El modelo de votación final representa la culminación de nuestro proceso de selección, combinando los mejores modelos individuales con pesos optimizados:

- **CatBoost** (peso 6.0): Mostró el mejor rendimiento individual y la mayor estabilidad.
- **Random Forest** (peso 3.5): Aportó robustez y capacidad de generalización.
- **LightGBM** (peso 2.7): Contribuyó con predicciones rápidas y precisas.
- **XGBoost** (peso 1.0): Añadió un componente de regularización adicional.
- **Gradient Boosting** (peso 3.5): Proporcionó un balance entre velocidad y precisión.

La asignación de pesos se determinó mediante un **proceso iterativo de optimización**, evaluando no solo el rendimiento individual de cada modelo sino también su contribución al conjunto final.

4. 3. Criterios aplicados para la selección del ganador

La selección del modelo óptimo se basó en un conjunto exhaustivo de criterios diseñados para garantizar no solo la precisión de las predicciones, sino también la robustez y aplicabilidad práctica del modelo:

1. Métricas de Rendimiento:

- **RMSE como Métrica Principal**: Se eligió el Error Cuadrático Medio Raíz por su interpretabilidad y su capacidad para penalizar errores grandes. Los valores se redondearon a dos decimales para mantener la consistencia con las especificaciones del problema.
- **Análisis de Varianza**: Se evaluó la desviación estándar de las predicciones en la validación cruzada para asegurar la estabilidad del modelo.
- **Métricas Secundarias**: Se consideraron también el Error Absoluto Medio y el R^2 ajustado para tener una visión más completa del rendimiento.

2. Proceso de Optimización:

- **Búsqueda de Hiperparámetros**: Se realizó una búsqueda exhaustiva mediante *GridSearchCV*, explorando sistemáticamente el espacio de hiperparámetros de cada modelo.
- **Validación Cruzada**: La implementación de *5-fold cross-validation* aseguró una evaluación robusta y representativa del rendimiento real del modelo.
- **Monitoreo de Recursos**: Se utilizó *EmissionsTracker* para medir y optimizar el consumo energético durante el entrenamiento, considerando la sostenibilidad del modelo.

3. Criterios de Robustez:

- **Estabilidad**: Se evaluó la consistencia de las predicciones a través de diferentes particiones de datos.
- **Generalización**: Se analizó el rendimiento en datos no vistos para evitar el sobreajuste.
- **Eficiencia Computacional**: Se consideraron los tiempos de entrenamiento y predicción, así como los requisitos de memoria.
- **Interpretabilidad**: Se valoró la capacidad del modelo para proporcionar *insights* sobre la importancia de las variables.

4. Consideraciones Prácticas:

- **Mantenibilidad**: Se evaluó la facilidad de actualización y mantenimiento del modelo.
- **Escalabilidad**: Se consideró la capacidad del modelo para manejar incrementos en el volumen de datos.
- **Reproducibilidad**: Se aseguró que los resultados fueran reproducibles mediante la fijación de semillas aleatorias.

El modelo *ensemble* final fue seleccionado tras demostrar un rendimiento superior en estos criterios, proporcionando un balance óptimo entre precisión, robustez y aplicabilidad práctica. La combinación ponderada de diferentes algoritmos permite aprovechar las fortalezas específicas de cada modelo mientras mitiga sus debilidades individuales, resultando en un sistema predictivo más fiable y estable.

5. Explicabilidad y Justicia

La transparencia y comprensión del proceso de toma de decisiones de nuestros modelos es fundamental, especialmente en un contexto tan crítico como la producción de vacunas. Para ello, hemos realizado un análisis utilizando **SHAP** (*SHapley Additive exPlanations*), que nos permite comprender cómo cada variable contribuye a las predicciones individuales y su impacto global en el modelo.

5.1. Análisis Global de Variables Importantes

El análisis SHAP revela patrones consistentes a través de los diferentes modelos del ensemble:

La **glucosa mínima** emerge como la **variable más influyente** en todos los modelos, mostrando una correlación positiva especialmente fuerte con la producción del antígeno. Esta consistencia entre modelos refuerza la importancia crítica del control de los niveles de glucosa durante el proceso productivo. Además, la **turbidez** y **sus métricas derivadas** (máxima, media y end_cul) aparecen como el segundo grupo de variables más importantes. Particularmente, **CatBoost** y **LGBM** muestran una fuerte dependencia de estas mediciones, sugiriendo que son indicadores cruciales de la calidad del proceso de producción.

5.2. Patrones de Impacto Específicos

El análisis detallado de los modelos individuales revela matices importantes:

CatBoostRegressor:
- Muestra la distribución más equilibrada del impacto de las variables
- La glucosa mínima y la turbidez máxima dominan claramente las predicciones
- Las variables de temperatura mantienen una influencia moderada pero consistente

RandomForestRegressor:
- Presenta una mayor dependencia de las variables relacionadas con la turbidez
- Muestra una sensibilidad particular a las mediciones de DO y PV
- Las variables temporales tienen un impacto más moderado

LGBMRegressor:
- Exhibe una fuerte dependencia de las mediciones de turbidez
- Las variables de proceso (temperatura, DO) muestran un impacto más uniforme
- Demuestra una sensibilidad especial a las mediciones de final de cultivo

5.3. Justicia y Equidad en las Predicciones

El análisis SHAP también nos ha permitido identificar y abordar posibles sesgos en las predicciones:

1. Distribución de Impacto:
- Se observa una distribución equilibrada del impacto de las variables críticas
- No hay evidencia de dependencia excesiva de variables potencialmente sesgadas
- Los valores atípicos son tratados de manera consistente por todos los modelos

2. Casos Especiales:
- Se identificaron casos donde las predicciones mostraban alta variabilidad
- Los lotes con condiciones extremas reciben un tratamiento equilibrado
- Las predicciones mantienen su precisión incluso en casos límite

3. Robustez del Modelo:
- El ensemble mitiga efectivamente los sesgos individuales de cada modelo

- La combinación ponderada asegura predicciones más equitativas
- Se mantiene la consistencia predictiva a través de diferentes condiciones de operación

5.3. Implicaciones para la Optimización del Proceso

Este análisis de explicabilidad nos proporciona directrices claras para la optimización del proceso:

1. El **control preciso** de los niveles de glucosa debe ser una prioridad operativa
2. El **monitoreo** continuo de la turbidez es crucial para asegurar la calidad del producto
3. Las **variables** de proceso (temperatura, DO) requieren un control estable pero **menos estricto**.

La comprensión de las relaciones identificadas proporciona una base sólida tanto para la predicción precisa como para la optimización del proceso productivo. Nuestro enfoque en explicabilidad y equidad garantiza predicciones que son no solo precisas, sino también confiables y justas, aspectos fundamentales en la producción de vacunas donde la consistencia y calidad son imperativos.

6. Sostenibilidad y Eficiencia Computacional

El desarrollo sostenible y la eficiencia energética son aspectos cada vez más relevantes en el campo de la inteligencia artificial. En nuestro proyecto, mediante la biblioteca *CodeCarbon*, monitorizamos cuidadosamente tanto el tiempo de computación como la huella de carbono asociada.

El proceso de entrenamiento y validación generó emisiones totales de aproximadamente **490 gramos de CO2, equivalentes a un viaje en coche de 2.5 kilómetros**. Estas emisiones se distribuyeron entre experimentos principales (257.43g), pruebas de validación cruzada (142.86g) y ajuste final del ensemble (89.71g).

La experimentación completa requirió **18.5 horas** de computación, distribuidas en **entrenamiento inicial (8.5h**), **validación cruzada y ajuste de hiperparámetros (6.2h)**, y **ensamblado con pruebas finales (3.8h)**. Para minimizar nuestro impacto ambiental, implementamos paralelización eficiente, selección preliminar de variables y *early stopping* en modelos de boosting.

Estas medidas de optimización no solo **redujeron** las emisiones de CO2, sino que también **mejoraron la eficiencia general** del proceso de desarrollo. Este análisis demuestra nuestro fuerte compromiso con el desarrollo responsable de soluciones de inteligencia artificial, buscando así un equilibrio entre el rendimiento del modelo y su impacto ambiental.