Documento descriptivo

1. Para poder optimizar y mejorar el proceso de producción de vacunas hemos creado un modelo de predicción, el cual, nos ayudara a disponer de una mejor visión para la futura mejora. El reconocimiento de cuáles son los puntos y parámetros más influyentes del proceso, será un punto fundamental para la toma de decisiones y la mejora en la producción de vacunas.
2. El proceso de ETL de nuestros datos está basado en una unión de todas las bases de datos obtenidas para su limpieza conjunta, hemos dejado como base madre la tabla de cultivos\_productivos como muestra la imagen.

A screenshot of a computer screen

Description automatically generated

A continuación, hicimos la limpieza de datos bajo los siguientes criterios.

* **Revisión de valores faltantes**: Identifica y manejar datos incompletos (eliminar, imputar o reemplazar).
* **Eliminación de duplicados**: Detectar y eliminar registros duplicados que podrían distorsionar los resultados.
* **Corrección de errores**: Corregir errores tipográficos, inconsistencias de formato y outliers.
* **Normalización**: Uniformar formatos (fechas, unidades de medida, etc.).
* **Codificación de variables**: Convertir variables categóricas en variables numéricas.
* **Escalado**: Ajustar los rangos de valores para algoritmos sensibles.

1. Con esto hecho, iniciamos con la creación de varios modelos de predicción para su prueba y saber cuál es el que mejor actúa frente al objetivo, ya que el proceso de prueba y evaluación es clave para la elección adecuada del modelo.

Para esto definimos el problema y obtuvimos que los mejores modelos son: regresión lineal o logística, arboles de decisión, redes neuronales y clustering.

Probando varios modelos y utilizando métricas como precisión, AUC o RMSE para comparar resultados hemos llegado a la conclusión de que el mejor modelo es DIXON and COLES. **(imágenes de modelos)**

1. Decisión y resultados del modelo elegido
2. Conclusión.

**Notas:**

**Ojo con la hyper\_parametrizacion de los modelos.**

La regresión lineal (comprobar si funciona para problemas de regresión) es importante usarla como modelo comparativo, si un modelo más complejo no da mejores resultados que la regresión es mejor quedarse con esta, es fácil de entrenar y alto poder de interpretabilidad. (sklearn) **pruebas AUC al igual que RMSE**

El modelo árbol de decisión sirve como base para otros modelos más complejos.

Modelo ARIMA: series temporales, predecir valores temporales, forecasting. (no se si son datos estacionarios)

Suavizado exponencial: Series temporales micro, ejemplo (demanda de una empresa) no es necesario que los datos sean estacionarios.

Random forest: genera modelos muy estables y no sobreajusta contra: tarda mucho en entrenarse.

XGBBOOST: también compara muchos arboles de decisión, cada nuevo mini árbol intenta corregir los del árbol anterior, puede sobre ajustar, es el mejor en el momento, pero tiene tiempos de entrenamiento muy largos.

LIGHTGBM: Es el número uno en proyectos empresariales con datos tabulares, también usa arboles de decisión, funciona más rápido que XGBBOOST construye ramas verticales. Datos Supervisados

Referencias para estudiar modelos: <https://keyrus.com/sp/es/insights/las-11-tecnicas-mas-utilizadas-en-el-modelado-de-analisis-predictivos>

<https://digibuo.uniovi.es/dspace/bitstream/handle/10651/64418/TFM_PelayoSuarezDosantos.pdf?sequence=4>

Revisar las oportunidades en

* INTELIGENZ,
* AI & Data hub wavespace
* Flower.ai
* Cajamar tecnología
* Innova-tsn

**VITACORA**

Crear carpeta Data\_Base

Agrupamos todos los archivos de biorreactores y de centrifugación en folders diferentes, para hacerle un append a todo lo que está en bio y centri y los que no están en estos grupos los dejamos por separado, después leímos todos los archivos en Python y separamos las hojas extras incluyendo las ocultas y las volvimos un dataframe independiente.

Problema, DIFICL ver cuál es cual df por los nombres

Solución, A mano acortar los nombres

Ya cambiamos a mano los nombres de los DF sueltos

Mirar si cambiamos los nombres de biorreactores y centrifugación, ya que en biorreactores.

Para los folders de centrifugación y biorreactores usamos la librería openpyxl, la cual nos permitió omitir las hojas ocultas que no servían.

PROBLEMA: las columnas de los de los biorreactores y centrifugación tienes diferentes nombres de columnas lo cual hace que se multipliquen a la hora de concatenarlas.

Posible solución: no tome el nombre de las columnas, sacar índice a cada columna.

La idea principal es **unificar los nombres de las columnas** eliminando o normalizando los IDs o partes variables antes de concatenar los DataFrames.

Extraer el valor numérico del nombre del archivo, para quitarlo del nombre de las columnas y además hacer una columna con el mismo ID, para extraer el número del archivo usamos la librería RE.

Creamos las columnas con B y C para el ID

PROBLEMON hay lotes repetidos hasta más de 11 veces

Además los hijos de puta nos pusieron / en los lotes, no todos los datos están regulares hay símbolos y mayúsculas que tenemos que arreglar.

Al cruzar la columna orden de la of con la columna de inicio de centrifugación tenemos que hallar la manera muy seguramente por la fecha de poder unirla y saber cuál es el lote para usar.

Nos enfocamos en tratar de unir una sola orden, con todas las fases y procesos en la manufactura.

Tratar de que por lote exista una sola fila.