

Reporte Semana # 2

Data Engineering

Índice

Página 2. Fuentes

Página 3. Carga incremental de los datos

Página 4. ETL completo













Página 5. Diccionario de datos










Página 7. Stack elegido






Página 8. Estructura de datos implementada

Fuentes

Para el desarrollo de los requerimientos decidimos usar la data que nos proveyó el cliente. Consideramos que los datos recibidos son suficientes y confiables para presentar el producto que queremos ofrecer, así como también toda la arquitectura asociada.

-  INPUTEVENTS_CV
-  INPUTEVENTS_MV
-  LABEVENTS
-  MICROBIOLOGYEVENTS
-  NOTEEVENTS
-  OUTPUTEVENTS
-  PATIENTS
-  PRESCRIPTIONS
-  PROCEDUREEVENTS_MV
-  PROCEDURES_ICD
-  SERVICES
-  TRANSFERS

-  D_CPT
-  D_ICD_DIAGNOSES
-  D_ICD_PROCEDURES
-  D_ITEMS
-  D_LABITEMS
-  DATETIMEEVENTS
-  DIAGNOSES_ICD
-  DRGCODES
-  ICUSTAYS

-  ADMISSIONS
-  CALLOUT
-  CAREGIVERS
-  CHARTEVENTS
-  CPTEVENTS

Carga Incremental de los Datos

El aplicativo de la carga incremental de datos hace referencia a la actualización de los mismos en la medida que las fuentes principales sean modificadas a lo largo del tiempo.

En nuestro contexto, el análisis del área de cuidados intensivos en cuestión, se espera que esta actualización sea instantánea, ya que el personal médico necesitara tener los datos actualizados en todo momento.

Para esta carga incremental se dispuso del siguiente orden de ejecución:

- Importación de los datos a MySQL
- Limpieza y normalización de datos
- Eliminación de tablas que no usaremos

ETL completo

Origen de los Datos

Los datos que se utilizan en este proyecto provienen de múltiples archivos CSV que contienen información de diversas variables de la Unidad de Cuidados Intensivos. Dichas variables son estudiadas según su importancia e influencia positiva o negativa para un mejor rendimiento de esta área. Los datos fueron extraídos de CSVs. que el cliente proveyó directamente.

Proceso de Transformación

Una vez que se importaron los archivos, se realizaron varias transformaciones para prepararlos para su uso. Estas incluyeron:

- Renombrar columnas para mayor claridad
- Completar los datos faltantes
- Revisar y eliminar filas duplicadas
- Eliminar columnas que consideramos no son necesarias

Destino Final

Los datos transformados son luego consumidos por la plataforma Google Cloud Platform, para su posterior uso y evaluación en el Dashboard y en el sistema de Machine Learning

Conclusiones

El proceso de ETL fue exitoso, tanto en la recolección como en la transformación de datos provenientes de diferentes fuentes. Las transformaciones realizadas permiten que los datos se analicen de manera efectiva para obtener información valiosa sobre el efecto que distintas variables tienen sobre el rendimiento de las UCI.

Diccionario de Datos

patients	patient_id	Esta columna contiene probablemente un identificador único para cada paciente o sujeto de su base de datos. Este ID puede haber sido asignado por el sistema sanitario o el proyecto de investigación que recopiló los datos.
	gender	Es probable que esta columna contenga información sobre el sexo o género de cada paciente de su base de datos. Puede utilizar un sistema de codificación como "M" para hombre y "F" para mujer.
	date_birth	Esta columna contiene probablemente la fecha de nacimiento de cada paciente de su base de datos. Puede almacenarse como un tipo de dato de fecha/hora o en un formato específico como AAAA-MM-DD HH-mm-ss.
	date_decease	Esta columna contiene probablemente la fecha de defunción de cada paciente de su base de datos, si procede. Al igual que dob, puede almacenarse como un tipo de dato fecha/hora o en un formato específico.
	deceased	Esta columna puede contener un valor binario (por ejemplo, 0 ó 1) que indica si un paciente ha fallecido o no, según la información disponible en la base de datos.
	deceased_hosp	Esta columna puede contener un valor binario (por ejemplo, 0 ó 1) que indica si un paciente ha fallecido o no en el hospital.

icustays	icu_id	Esta columna probablemente contiene un identificador único para cada estancia en la UCI asociada con cada ingreso hospitalario y paciente en su base de datos.
	patient_id	Esta columna contiene probablemente un identificador único para cada paciente o sujeto en su base de datos.
	hadm_id	Esta columna contiene probablemente un identificador único para cada ingreso hospitalario asociado a cada paciente en su base de datos.
	first_careunit	Esta columna probablemente contiene el nombre o código de la primera unidad asistencial o departamento donde el paciente ingresó en la UCI.
	last_careunit	Esta columna contiene probablemente el nombre o código de la última unidad asistencial o departamento donde el paciente fue tratado durante su estancia en la UCI.
	first_wardid	Esta columna contiene probablemente el identificador de la primera sala o ubicación dentro de la unidad asistencial donde el paciente ingresó en la UCI.
	last_wardid	Esta columna contiene probablemente el identificador de la última sala o ubicación dentro de la unidad asistencial donde el paciente fue tratado durante su estancia en la UCI.
	intime	Esta columna contiene probablemente la fecha y hora en que el paciente ingresó en la UCI.
	outtime	Esta columna contiene probablemente la fecha y hora en que el paciente fue dado de alta o trasladado fuera de la UCI.
	los	(Lenght of stay)Esta columna contiene probablemente la duración de la estancia (LOS) para cada estancia en la UCI, medida en unidades de tiempo (por ejemplo, horas, días).

admissions	hadm_id	Se trata probablemente de un identificador único para cada ingreso hospitalario en la base de datos. Puede ser un identificador generado por el sistema o un identificador específico del hospital.
	admittime	Esta columna probablemente representa la fecha y hora de admisión del paciente, y podría almacenarse como un valor datetime.
	diagnosis	Esta columna representa probablemente el diagnóstico principal del ingreso hospitalario del paciente.
	hospital_expire_flag	Esta columna probablemente indica si el paciente falleció durante su estancia en el hospital o no, y podría tener valores como "Y" o "N" para representar "sí" o "no", o 1 y 0 para representar booleanos verdadero y falso.

callout	call_id	Numero identificador de registro de callout del hospital
	patient_id	Esta columna contiene probablemente un identificador único para cada paciente o sujeto en su base de datos.
	careunit_id	Numero identificador de la unidad de cuidados intensivos.
	hadm_id	Esta columna contiene probablemente un identificador único para cada ingreso hospitalario asociado a cada paciente en su base de datos.
	insurance_id	Numero identificador de la aseguradora del paciente.
	date	columna de fechas
	createtime	fecha de solicitud de alta
	acknowledgetime	fecha de notificación de la dada de alta al paciente
	outcometime	fecha de salida del paciente
	los	tiempo en horas de la salidad del paciente.

Diccionario de Datos

careunit	careunit_id	Codigo identificador del area de cuidados intensivos.
	careunit	Nombre asociado al codigo del area de cuidados intensivos
insurance	insurance_id	Codigo identificador de la aseguradora
	insurance	Nombre de la aseguradora

Stack Elegido

Python

Lenguaje de alto nivel de programación interpretado cuya filosofía se centra en la legibilidad del código, se utiliza para desarrollar aplicaciones de todo tipo. Es un lenguaje de programación multiparadigma. Soporta parcialmente la orientación a objetos, programación imperativa y en menor medida, programación funcional. Es un lenguaje dinámico, interpretado y multiplataforma. Es considerado uno de los lenguajes de programación más populares.

Visual Studio Code (VSC)

Editor de código fuente desarrollado por Microsoft. Incluye soporte para la depuración, control integrado de GIT, resaltado de sintaxis, finalización inteligente de código, entre otras funciones idóneas para la construcción de este proyecto.

Microsoft Power BI

Software de visualización de datos interactivo desarrollado por Microsoft con un enfoque principal en la inteligencia empresarial.

Streamlit

Biblioteca de Python de código abierto que facilita la creación de aplicaciones web personalizadas para el aprendizaje automático y la ciencia de datos.



Estructura de Datos Implementada

Considerando los datos del cliente, en este caso el departamento de Engineering determino que la opción más eficaz consiste en montar un DataLake (DL) y un DataWarehouse (DW) en los servicios de Google Cloud Platform