Informe Técnico: Pipeline de Datos COVID-19 – Ecuador vs Perú
Proyecto Final
Fecha: 31 de agosto de 2025
Estudiante: Kevin Cajeca

1. Arquitectura del Pipeline

El pipeline sigue una arquitectura modular basada en Dagster, con enfoque en assets para facilitar la orquestación, monitoreo y trazabilidad. Cada asset representa una transformación clara de datos.

Descripción de assets

leer_datos:
Descarga el dataset completo de OWID desde la URL canónica usando
requests.No aplica limpieza.

datos_procesados:
Filtra por Ecuador y Perú, elimina nulos en new_cases y people_vaccinated, y elimina duplicados.

metrica_incidencia_7d:
Calcula la incidencia acumulada por 100k habitantes con promedio móvil de 7 días.

metrica_factor_crec_7d:
Calcula el factor de crecimiento semanal (casos semana actual vs anterior).

reporte_excel_covid:
Exporta un reporte comparativo en Excel con dos hojas: incidencia y factor de crecimiento.

2. Justificación de decisiones de diseño

Elección de tecnologías
Dagster: Ideal para pipelines de datos modulares, con UI de monitoreo, asset checks y ejecución incremental.
Pandas: Adecuado para datasets de tamaño medio (OWID tiene ~100k filas), prototipado rápido y fácil integración con Excel.
Requests: Ligero y eficiente para descargar CSVs sin necesidad de subir archivos manualmente.
OpenPyXL: Permite formatear celdas en Excel (fechas, alineación), evitando errores visuales como #####
.

No se usó DuckDB ni Soda porque:

- El volumen de datos no requiere optimización de rendimiento
- Pandas es suficiente para las operaciones de ventana y agregación-
- Los asset checks de Dagster cubren las validaciones necesarias

3. Decisiones de validación

3.1 Validación de entrada (chequeos_entrada)

Se aplicaron las siguientes reglas para garantizar la calidad de los datos crudos:

- location presente
- date presente
- new_cases presente
- population presente
- datos de Perú o Ecuador presentes
- fecha máxima razonable Máxima: 2023-03-17
- unicidad (location, date)
- population > 0
- new_cases ≥ 0

Motivación de reglas:

- max(date) ≤ hoy + 30: evita fechas futuras no razonables
- unicidad (location, date): asegura que no haya múltiples observaciones por fecha
- population > 0: evita división por cero en métricas
- new_cases ≥ 0: valida que no haya casos negativos (salvo correcciones documentadas)

Estrategia para duplicados:

- El check falla por diseño para alertar sobre datos crudos no limpios
- En datos_procesados, se eliminan duplicados con drop_duplicates(keep="last")
- Se asume que la última observación es la versión corregida

3.2 Validación de salida (chequeos_salida_incidencia)
incidencia_7d en rango [0, 2000]: (Rango epidemiológicamente razonable)

Motivación:

- Valores fuera de rango indicarían errores de cálculo o datos extremos
- 2000 casos por 100k en 7 días es un pico extremo (ej: ola Delta)

4. Descubrimientos importantes en los datos
1. Perú tuvo picos más altos que Ecuador en 2021 y 2022
- Máxima incidencia 7D: ~330 casos/100k (Perú, enero 2022)
- Ecuador: ~50 casos/100k (enero 2022)

2.Ecuador mostró mayor estabilidad
- Menor variabilidad en el factor de crecimiento
- Menos olas pronunciadas

3.Factor de crecimiento >1 en olas, <1 en descensos
- Ej: Perú en enero 2022 → factor ~3.7 (crecimiento acelerado)
- Ecuador en abril 2022 → factor ~0.4 (descenso)

4. Datos crudos con 7,770 duplicados
- Común en fuentes como OWID por actualizaciones retroactivas
- Manejado correctamente en el pipeline

5. Consideraciones de arquitectura
Elección de pandas vs. DuckDB vs. Soda
Pandas: Ideal para prototipado, fácil de usar, soporta operaciones de ventana y pivotado
DuckDB: No necesario para este volumen de datos
Soda: Los asset checks de Dagster son suficientes para validaciones simples

Conclusión: Pandas es la mejor opción para este caso por simplicidad, flexibilidad y compatibilidad con Excel.

6. Resultados
6.1 Métricas implementadas
Incidencia 7D: (new_cases / population) * 100000
- Tasa de nuevos casos por 100k habitantes (suavizada)

Factor de crecimiento:casos_semana_actual / casos_semana_previa
- Indica si la epidemia está creciendo (>1) o decreciendo (<1)

6.2 Resumen de resultados
Ecuador
Incidencia 7d titpica: 5–15
Factor crecimiento tipico: 0.8–1.3
Perú
Incidencia 7d titpica : 10–30
Factor crecimiento tipico:0.7–1.8

Interpretación:
- Perú tuvo una carga de enfermedad más alta
- Ambos países mostraron olas similares, pero Perú con mayor intensidad
- Ecuador mostró mayor control en 2022

6.3 Resumen del control de calidad
Entrada: unicidad (location, date)
Falló
Se eliminaron duplicados en datos_procesados
Entrada: new_cases ≥ 0
Pasó
No hubo casos negativos
Salida: incidencia_7d en rango
Pasó
Todos los valores dentro de [0, 2000]
Procesamient: inf en factor_crec_7d
Detectado
Se reemplazó inf con 0 o None

7. Conclusiones
- El pipeline es robusto, modular y reproducible
- Se detectaron y corrigieron problemas en datos crudos
- Se generó un reporte comparativo claro y profesional
- Se cumplió con todos los requisitos del proyecto

Nota final:
El archivo reporte_covid.xlsx se genera automáticamente al ejecutar el pipeline.