Etapa 3 — Experimento E-P-A (Spark MLlib)

Objetivo

Clasificar automáticamente el tipo de incidente NIBRS (THEFT, VIOLENCE, DRUG, SEX, KIDNAPPING_TRAFFICKING, OTHER, MULTIPLE) sobre el dataset 2020, extrayendo, procesando y analizando datos a escala con Apache Spark MLlib para obtener conocimiento útil sobre patrones de delito.

Extracción

- Fuente de datos: `data/Individual_Incident_2020.csv` (~1.52 GB, 9,130,711 filas).
- Entorno: clúster Spark (1 master, 2 workers) vía Docker Compose.
- Ingesta distribuida con inferencia de esquema y normalización de columnas:
  - `incident_number → A_INCIDENT_ID`, `total_offense → TOTAL_DELITOS`, `date_HRF → INCIDENT_DATE_RAW`.
- Evidencia de corridas y totales: `data/reports/etapa3_summary.txt`, `data/reports/etapa3_summary_rf.txt`.

Procesamiento

- Filtrado de calidad: `TOTAL_DELITOS > 0` → 7,561,485 filas procesadas.
- Enriquecimiento (feature engineering):
  - Fecha: `INCIDENT_DATE = to_date(yyyyMMdd)`, `TIPO_DIA` (DIA_SEMANA/FIN_SEMANA), `MONTH`.
  - Hora: `HOUR_BUCKET` desde la columna textual `hour` (e.g., “midnight”, “1900–1959”).
  - Numéricas tipadas e imputadas: `TOTAL_DELITOS`, `total_victim`, `total_offender`, `gun_involvement`, `drug_involvement`, `stolen_motor`, `property_value`.
- Etiqueta supervisada `INCIDENT_TYPE` derivada de flags: si >1 en 1 → `MULTIPLE`; si ninguna → `UNKNOWN`.

Diseño Del Modelo

- Pipeline MLlib: `StringIndexer` + `OneHotEncoder` + `VectorAssembler` + clasificador.
- Modelos comparados: Logistic Regression (multinomial) y RandomForestClassifier.
- Split 80/20 (train/test): 6,049,475 | 1,512,010 filas.
- Métricas: Accuracy y F1; por clase (precisión, recall, F1) y matriz de confusión.

Ejecución (cluster)

- LR: `src/etapa3_mllib_clasificacion.py` → modelos y reportes en `data/models` y `data/reports`.
- RF: `src/etapa3_mllib_clasificacion_rf.py` → mismos outputs para comparación.
- Visualización:
  - Reporte simple: `data/reports/etapa3_report.html`.
  - Comparativa LR vs RF: `data/reports/etapa3_report_compare.html`.
  - Gráficos: `data/reports/figures/*.png`.

Resultados

- Totales: 9,130,711 leídos; 7,561,485 procesados; 6,049,475 train | 1,512,010 test.
- Rendimiento global:
  - LR: Accuracy 0.4598 | F1 0.3787 (`data/reports/etapa3_summary.txt`).
  - RF: Accuracy 0.5979 | F1 0.5094 (`data/reports/etapa3_summary_rf.txt`).
- Distribución de etiquetas (test):
  - THEFT 641,316; OTHER 373,363; VIOLENCE 271,856; DRUG 151,042; MULTIPLE 58,350; SEX 16,083.
  - CSV: `data/reports/incidente_type_distribution.csv`.
- Por clase (RF) — `data/reports/per_class_metrics_rf.csv`:
  - THEFT: Prec 0.665 | Rec 0.920 | F1 0.772 (muy sólido).
  - VIOLENCE: Prec 0.466 | Rec 0.920 | F1 0.619 (recupera bien, precisión moderada).
  - DRUG: Prec 0.768 | Rec 0.198 | F1 0.315 (acierta cuando predice, recupera poco).
  - OTHER: Prec 0.641 | Rec 0.058 | F1 0.106 (clase heterogénea, difícil).
  - MULTIPLE: Prec 0.768 | Rec 0.214 | F1 0.335.
  - SEX: Prec 0.000 | Rec 0.000 | F1 0.000 (muy desbalanceada).
- Matriz de confusión (RF) — `data/reports/confusion_rf.csv`:
  - Confusiones destacadas: OTHER→THEFT 224,719; OTHER→VIOLENCE 125,601; DRUG→VIOLENCE 79,857; DRUG→THEFT 38,154; THEFT↔VIOLENCE (46,071 / 12,376).

Costes y Tiempos

- LR (total ~13.1 min): carga ~49s; features ~14.5s; split ~116s; fit ~485s; eval ~94s.
- RF (total ~51.3 min): carga ~48s; features ~14.6s; split ~107s; fit ~2,581s; eval ~179s.
- Gráficos: `data/reports/figures/timings.png` y comparación `data/reports/figures/algos_compare.png`.

Conocimiento Extraído

- Clasificación a gran escala viable; RF mejora significativamente a LR con el mismo feature set.
- THEFT y VIOLENCE se distinguen bien (recall alto): útil para priorización y alertas.
- OTHER y SEX requieren señales adicionales y/o manejo del desbalance.
- La etiqueta derivada desde flags ofrece una taxonomía interpretable y operativa.

Limitaciones

- Desbalance y baja señal en clases minoritarias (SEX, parte de OTHER).
- `OTHER` agrega heterogeneidad que degrada el rendimiento.
- Conjunto de features básico: no se usaron `property_description*`, `completed_attempted*`, demografía detallada, día de semana exacto, `ORI`.

Próximos Pasos

- Ampliar features:

    - Categóricas: `completed_attempted2/3`, `property_description[1..3]`, día de semana (1–7), más franjas horarias.
    - Numéricas: contadores demográficos (menores, sexo de víctima/ofensor).
- Abordar desbalance:
    - Reponderación por clase (p. ej. `weightCol` en LR) o muestreo estratificado.
    - Analizar `MULTIPLE` por separado.
- Operatividad:
    - Persistir insumos en Parquet y calendarizar entrenamientos incrementales.
    - Publicar métricas periódicas para observabilidad.

Referencias De Salida

- Resúmenes: `data/reports/etapa3_summary.json`, `data/reports/etapa3_summary_rf.json`.
- Por clase y confusión: `data/reports/per_class_metrics_*.csv`, `data/reports/confusion_*.csv`.
- HTML: `data/reports/etapa3_report.html`, `data/reports/etapa3_report_compare.html`.
- Modelos: `data/models/incidente_clf_lr`, `data/models/incidente_clf_rf`.