INSTITUTO DE INVESTIGACIONES EN MATEMÁTICAS APLICADAS Y EN SISTEMAS

Universidad Nacional Autónoma de México
Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas

Proyecto Final de Minería de Datos

Sistema de Ruteo Seguro para CDMX

Aplicando KDD, análisis espacial y Machine Learning a accidentes de tránsito

Profesora: ILEANA ANGELICA GRAVE AGUILAR

Integrantes:

Alegre Ventura Roberto Jhoshua
Ramirez Nava Alejandro Iram

Fecha: 2 de diciembre de 2025

Resumen ejecutivo

En este proyecto implementamos de principio a fin el proceso de *Knowledge Discovery in Databases* (KDD) sobre datos reales de accidentes de tránsito de la Ciudad de México (2019–2023), con el objetivo de construir un **sistema de ruteo seguro** que vaya más allá de la típica ruta más corta.

Partimos de más de 1.04 millones de registros de accidentes a nivel nacional y, tras la selección y limpieza, trabajamos con alrededor de 78 mil accidentes georreferenciados en CDMX. Con ellos diseñamos un pipeline que incluye:

- Limpieza y consolidación multi-anual.
- Transformaciones temporales, espaciales y de severidad.
- Análisis espacial (DBSCAN, hot spots con Getis-Ord Gi*, Moran's I).
- Modelos de Machine Learning para predecir la gravedad de los accidentes.
- Cálculo de un índice de riesgo compuesto por tramo vial y su uso en ruteo.

Finalmente integramos este índice de riesgo a la red vial de OpenStreetMap y mostramos, en un caso Zócalo → Polanco, cómo se pueden comparar rutas más cortas, balanceadas y más seguras, con una aplicación directa a **traslados hospitalarios y rutas de emergencia**.

Palabras clave: KDD, minería de datos, accidentes de tránsito, Ciudad de México, análisis espacial, Machine Learning, ruteo seguro, puntos negros, traslados hospitalarios.

Índice general

1

Introducción y Contexto

1.1 Motivación como equipo

Como equipo de estudiantes de Minería de Datos, quisimos aplicar todo el ciclo KDD a un problema que realmente afecta la vida diaria en la Ciudad de México: los accidentes de tránsito. Normalmente sólo nos quedamos con el tráfico o la nota roja, pero detrás de eso hay una cantidad enorme de datos que permiten entender mejor dónde y cuándo se concentran los riesgos.

Nuestra meta no fue sólo hacer mapas bonitos, sino llegar a un sistema que sugiera **rutas más seguras**, pensando tanto en conductores en general como en contextos sensibles, por ejemplo, traslados de pacientes entre hospitales.

1.2 Problema a resolver

En términos simples, el problema que abordamos es:

- Identificar zonas de alto riesgo vial o *puntos negros*.
- Entender patrones espaciales y temporales de los accidentes.
- Estimar la gravedad potencial de un accidente dado su contexto.
- Traducir todo eso a un índice de riesgo por tramo de la red vial.
- Integrar el índice en un motor de ruteo para proponer rutas alternativas.

En el plan de trabajo definimos un **punto negro** como una zona o tramo de calle donde la probabilidad de tener un accidente grave es elevada, considerando frecuencia de accidentes, severidad (muertos/heridos) y la dimensión temporal (horas del día, días de la semana,

años).

1.3 Objetivo general y específicos

Nuestro objetivo general es desarrollar un **sistema integral de análisis y ruteo seguro** para la CDMX, siguiendo explícitamente el proceso KDD.

Objetivos específicos:

- Aplicar el proceso KDD sobre los datos de accidentes 2019–2023.
- Diseñar y calcular variables temporales, espaciales y de severidad.
- Implementar técnicas de minería de datos (clustering, hot spots, autocorrelación espacial y ML).
- Definir índices de riesgo (por zona y por tramo) a partir de esas métricas.
- Usar el índice de riesgo compuesto como peso en un sistema de ruteo.

1.4 Fuentes de datos y alcance

Trabajamos con dos fuentes principales:

- **Accidentes de tránsito**: registros georreferenciados para el periodo 2019–2023, de los cuales filtramos aquellos que corresponden a la Ciudad de México.
- **Red vial**: red `drive` de OpenStreetMap para la CDMX, obtenida con OSMnx, con casi 100 000 nodos y más de 230 000 aristas.
- **Hospitales en México**: Registros georreferenciados de los hospitales en México en el año 2023, incluyendo su tipología. De la fuente del Catálogo Único de Establecimientos de Salud (CLUES), con datos de más de 40 mil hospitales

El alcance geográfico es la Ciudad de México (16 alcaldías) y el periodo considerado va de 2019 a 2023.

2

Proceso KDD y Flujo General

En este capítulo contamos cómo aterrizamos cada etapa del proceso KDD a nuestro problema, y describimos el flujo general de datos y decisiones, tal como se planteó en el plan de trabajo y se fue ajustando a lo largo del semestre.

2.1 Esquema general del KDD

Nuestro pipeline KDD quedó organizado así:

1. **Selección de datos**: identificación de fuentes, descarga de bases de accidentes y construcción de la red vial.
2. **Preprocesamiento**: limpieza básica, manejo de nulos, filtrado geográfico y unificación de años.
3. **Transformación**: ingeniería de características temporales, de severidad y espaciales, así como asociación de accidentes a tramos de calle y celdas.
4. **Minería de datos**: análisis espacial (DBSCAN, hot spots, Moran's I) y modelos supervisados para gravedad.
5. **Interpretación y evaluación**: análisis de resultados, construcción de índices de riesgo e integración con el ruteo.

2.2 Entendimiento del Negocio (Business Understanding)

2.2.1 1.1 Determinar los objetivos del negocio

1.1.1 Antecedentes

En México, los accidentes de tránsito constituyen un problema persistente que afecta la seguridad vial, la movilidad urbana y la capacidad de respuesta del sistema de salud. La Zona Metropolitana del Valle de México concentra una parte significativa del parque vehicular nacional y presenta altos niveles de siniestralidad y tiempos de traslado elevados. En paralelo, a nivel nacional, una proporción considerable de accidentes viales graves ocurre en carreteras interurbanas y federales donde la distancia a hospitales con capacidad de atención puede determinar la sobrevivencia de los lesionados.

Como equipo de estudiantes de Minería de Datos, se decidió abordar el problema desde dos perspectivas complementarias:

- **Ámbito urbano (Ciudad de México):** construcción de un sistema que estime un índice de riesgo vial por tramo y sugiera rutas alternativas más seguras.
- **Ámbito nacional (carreteras):** identificación de clusters de accidentes alejados de hospitales y propuesta de ubicaciones para nuevas unidades médicas.

1.1.2 Objetivos del negocio

Desde la perspectiva del negocio, entendido como las necesidades de planeación urbana, salud pública y seguridad vial, los objetivos del proyecto son:

1. Reducir el riesgo vial en la Ciudad de México mediante un sistema de ruteo que proponga rutas alternativas con menor probabilidad de accidentes graves.
2. Identificar regiones carreteras mal atendidas por el sistema hospitalario y proponer ubicaciones óptimas para nuevos hospitales.
3. Generar evidencia cuantitativa y geoespacial útil para tomadores de decisión en movilidad, protección civil y salud.

1.1.3 Criterios de éxito del negocio

El proyecto será considerado exitoso si logra:

- En el sistema de ruteo:

    - Discriminar claramente rutas de mayor y menor riesgo.
    - Ofrecer alternativas más seguras sin incrementos excesivos de tiempo.
    - Permitir visualizaciones comprensibles de puntos negros y justificación del ruteo.
- En la propuesta de hospitales:
    - Identificar clusters de accidentes lejos de hospitales funcionales.
    - Cuantificar mejoras en distancia promedio, accidentes lejanos y personas potencialmente beneficiadas.
    - Establecer un ranking claro de prioridad de nuevos hospitales.
- A nivel académico:
    - Cubrir explícitamente cada fase de CRISP-DM.
    - Producir dashboards y mapas reproducibles.

2.2.2 1.2 Evaluar la situación (Assess Situation)

1.2.1 Inventario de recursos

Recursos humanos:

- Equipo de estudiantes con experiencia en análisis geoespacial, modelado y desarrollo web.
    - Alejandro Iram Ramírez Nava: físico y científico de datos
    - Roberto Jhoshua Alegre Ventura: matemático y científico de datos
- Asesoría docente para orientación metodológica:
    - Doctora Ileana Angélica Grave Aguilar

Datos disponibles:

- Base georreferenciada de accidentes con severidad (muertos/heridos).
- Catálogo CLUES de establecimientos de salud (ubicación, tipo, tipología).
- Red Vial: red drive de OpenStreetMap para la CDMX

Recursos computacionales:

- Python con librerías como `geopandas`, `hdbscan`, `scikit-learn`, `plotly`.
- Google Colab, VS Code, GitHub.
- Computadora lenovo ideapad 3, 1TB de memoria en disco, 6GB de RAM.

1.2.2 Requisitos, supuestos y restricciones

Requisitos:

- Seguir estrictamente CRISP-DM.
- Producir modelos reproducibles y visualizaciones exportables.

Supuestos:

- Los datos de accidentes y CLUES son representativos.
- La distancia geodésica es un buen proxy del tiempo de respuesta hospitalaria.

Restricciones:

- Tiempo de desarrollo limitado.
- Subregistro posible en datos de accidentes.
- No se modela tiempo real de tráfico ni disponibilidad de ambulancias.

1.2.3 Riesgos y contingencias

Riesgos:

- Errores en geocodificación o coordenadas inconsistentes.
- Clusters débiles por baja densidad de accidentes.
- Elección de parámetros (e.g. 10 km) sin suficiente respaldo experto.

Contingencias:

- Limpieza de outliers mediante HDBSCAN.
- Análisis de sensibilidad de umbrales y tamaños de cluster.
- Documentación explícita de limitaciones del modelo.

1.2.4 Terminología

- **Punto negro:** zona o tramo con alta concentración de accidentes graves.
- **Índice de riesgo vial:** medida compuesta de severidad y frecuencia.
- **Cobertura hospitalaria:** cercanía entre accidentes y hospitales funcionales.
- **CLUES:** Catálogo Único de Establecimientos de Salud.

1.2.5 Costos y beneficios

Costos:

- Horas de análisis, limpieza, modelado y desarrollo web.
- Mantenimiento futuro del sistema.

Beneficios:

- **Académicos:** aplicación completa de CRISP-DM.
- **Sociales:** reducción del riesgo vial y mejor planificación hospitalaria.
- **Técnicos:** flujo de trabajo reproducible y escalable.

2.2.3 1.3 Determinar los objetivos de minería de datos (Data Mining Goals)

1.3.1 Objetivos de minería de datos

Para CDMX (ruteo seguro):

- Construir un índice de riesgo por tramo basado en densidad, severidad y tiempo.
- Integrar dicho índice en un motor de ruteo que minimice riesgo + tiempo.

Para carreteras (hospitales propuestos):

- Identificar clusters lineales de accidentes con HDBSCAN.
- Calcular distancia al hospital más cercano mediante BallTree.
- Proponer ubicaciones óptimas de nuevos hospitales y cuantificar impacto.

1.3.2 Criterios de éxito de minería de datos

- El índice de riesgo discrimina correctamente zonas peligrosas en CDMX.
- El motor de ruteo reduce significativamente el riesgo agregado.
- Los hospitales propuestos presentan mejoras reales en distancia y cobertura.
- Los resultados son robustos a cambios en parámetros.

2.2.4 1.4 Elaborar el plan del proyecto (Produce Project Plan)

1.4.1 Plan del proyecto

El proyecto sigue las fases de CRISP-DM:

1. **Entendimiento del negocio:** definición del problema urbano y carretero.
2. **Entendimiento de los datos:** documentación y exploración de fuentes.
3. **Preparación de los datos:** limpieza, filtrado, geoprocesamiento.
4. **Modelado:** clustering HDBSCAN, cálculo de distancias geodésicas, modelado del índice de riesgo.
5. **Evaluación:** validación visual y cuantitativa de mejoras en cobertura y ruteo.
6. **Despliegue:** dashboards, mapas interactivos y prototipo en Django.

2.3 Entendimiento de los Datos (Data Understanding)

2.3.1 2.1 Recolección de datos iniciales (Collect Initial Data)

En esta fase se identificaron y recopilaron las fuentes de datos necesarias para abordar los objetivos de negocio definidos en la Fase 1. Las tres fuentes principales fueron:

- **Base de accidentes de tránsito georreferenciados (2022).**
  Se utilizó un conjunto de datos en formato `CSV` con accidentes de tránsito georreferenciados, correspondiente al año 2022. Esta base contiene, para cada accidente, la ubicación aproximada (coordenadas), la fecha y hora del evento, así como información sobre la severidad (personas heridas y fallecidas) y algunas características contextuales del siniestro.

- **Catálogo CLUES de establecimientos de salud.**
  Se empleó el *Catálogo Único de Establecimientos de Salud* (CLUES), a partir del cual se extrajeron las unidades médicas con campos como: entidad, municipio, localidad, tipo de establecimiento y tipología, así como las coordenadas de localización (latitud y longitud). Estos datos se transformaron a un `GeoDataFrame` para su análisis espacial.
- **Red vial de OpenStreetMap.**
  Para el modelado de rutas y la construcción de un índice de riesgo sobre la red vial de la Ciudad de México, se utilizaron datos de OpenStreetMap (OSM). A partir de esta fuente se derivó un grafo de la red de calles, con nodos y aristas que representan intersecciones y segmentos viales, respectivamente. Esto permitió asociar los accidentes a tramos específicos de la red y, posteriormente, calcular rutas alternativas.

En conjunto, estas fuentes permiten relacionar **dónde ocurren los accidentes**, **dónde están los hospitales** y **cómo se estructura la red vial** sobre la que se mueven los vehículos.

2.3.2 2.2 Descripción de los datos (Describe Data)

2.2.1 Accidentes de tránsito

La base de accidentes georreferenciados fue cargada inicialmente como un `DataFrame` y posteriormente convertida a un `GeoDataFrame` para facilitar el análisis espacial. De forma general, la estructura de la base incluye:

- Identificación del registro (id o folio de accidente).
- Variables temporales: fecha, hora del evento.
- Variables espaciales:
  - `LATITUD`, `LONGITUD`.
  - Geometría puntual (`geometry`) en CRS geográfico (EPSG:4326).
- Medidas de severidad:
  - Número de personas heridas (`TOTHERIDOS`).
  - Número de personas fallecidas (`TOTMUERTOS`).
- Variables contextuales (cuando están disponibles): tipo de choque, tipo de vehículo, referencias a la vialidad o carretera, delegación/municipio, etc.

Para ciertos análisis se generaron variables derivadas, tales como:

- **Clusters de accidentes** (por ejemplo, `cluster` o `cluster_hdb`), obtenidos mediante HDBSCAN sobre coordenadas proyectadas a un CRS métrico (EPSG:3857).
- **Distancia al hospital más cercano** (`distancia_al_hospital`), calculada con `BallTree` y distancia haversine.

2.2.2 Establecimientos de salud (CLUES)

El catálogo CLUES se integró inicialmente como un `DataFrame` y luego como `GeoDataFrame`, con campos relevantes como:

- **Contexto administrativo:** nombre de la entidad, municipio y localidad.
- **Tipo y tipología del establecimiento:**
  - `NOMBRE TIPO ESTABLECIMIENTO` (e.g., *DE HOSPITALIZACIÓN*, *DE CONSULTA EXTERNA*, *DE APOYO*).
  - `NOMBRE DE TIPOLOGIA` (e.g., *HOSPITAL GENERAL*, *HOSPITAL INTEGRAL*, *UNIDAD MÉDICA RURAL*).
- **Ubicación geográfica:** `LATITUD`, `LONGITUD`, y la columna `geometry` resultante.

Sobre esta base se aplicó un **proceso de filtrado** para quedarse únicamente con establecimientos plausibles para la atención de accidentes de tránsito (hospitalización, unidades integrales, hospitales generales, etc.), excluyendo laboratorios, oficinas, unidades administrativas, consultorios aislados y otras tipologías sin capacidad de atención de urgencias.

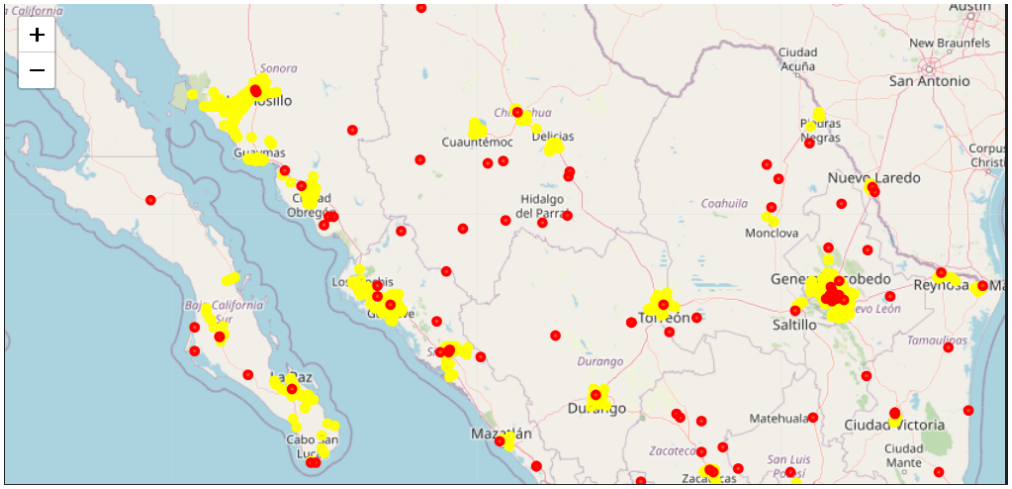

Figura 2.1: Mapa exploratorio de la distribución de los hospitales y de los accidentes a nivel nacional. Descubrimos que sí existen cúmulos de accidentes (puntos amarillos) alejados de cualquier hospital (puntos rojos).

2.2.3 Red vial (OpenStreetMap)

Los datos de OSM se procesaron para obtener:

- Un **grafo de la red vial** de la Ciudad de México, donde:
  - Los nodos representan intersecciones o puntos de la red.
  - Las aristas representan segmentos de calle o carretera.
- Atributos de las aristas, como longitud y tipo de vía, que permiten asignar costos (p.ej. tiempo estimado de recorrido).
- La posibilidad de **asociar accidentes a segmentos viales**, por ejemplo mediante la proyección del punto de accidente al tramo más cercano.

Esta representación es fundamental para el componente de ruteo seguro, ya que el índice de riesgo se calcula a nivel de tramo de red y luego se utiliza para sugerir rutas alternativas.

2.3.3 2.3 Exploración de los datos (Explore Data)

Una vez integradas las fuentes, se llevaron a cabo análisis exploratorios para entender mejor la distribución y el comportamiento de los datos.

2.3.1 Exploración espacial

- Se generaron mapas de calor y diagramas de dispersión donde cada punto corresponde a un accidente, permitiendo identificar **zonas de alta densidad** tanto en la Ciudad de México como en carreteras nacionales.
- Se observaron patrones de concentración de accidentes en:
  - Vialidades primarias y ejes viales en CDMX.
  - Tramos específicos de carreteras federales y estatales.
- Al superponer los puntos de accidentes con la ubicación de hospitales filtrados del CLUES, se detectaron regiones con **alta siniestralidad y baja cobertura hospitalaria**.

2.3.2 Exploración temporal y de severidad

- Se analizaron las distribuciones de accidentes por:
  - Día de la semana, franja horaria, y en su caso, por mes.
  - Número de heridos y fallecidos por evento.
- Se identificaron patrones como:
  - Mayor concentración de accidentes en ciertos horarios (picos de tráfico) o días.
  - Zonas donde, aunque el número de accidentes no es tan alto, la proporción de fallecidos es considerable, lo que aumenta el **riesgo relativo**.

2.3.3 Exploración sobre la red vial

En el caso de la Ciudad de México, los accidentes se proyectaron sobre la red vial de OSM, lo que permitió:

- Calcular **frecuencias de accidentes por tramo** o segmento vial.
- Estimar un **índice de riesgo** por tramo combinando frecuencia, severidad y, cuando fue posible, información temporal.
- Visualizar rutas alternativas y comparar su riesgo agregado frente a la ruta más corta o más rápida.

2.3.4 2.4 Verificación de la calidad de los datos (Verify Data Quality)

En esta fase se revisó la consistencia, completitud y precisión de los datos para asegurar que las conclusiones posteriores se basaran en información confiable.

2.4.1 Calidad de los datos de accidentes

- Se detectaron y eliminaron **outliers espaciales**, es decir, puntos aislados o claramente mal georreferenciados (por ejemplo, coordenadas fuera del país o en ubicaciones imposibles).
- Se validó la presencia de valores faltantes en campos clave (fecha, coordenadas, heridos, muertos) y se registró la proporción de registros incompletos.

- Se realizó una inspección visual para asegurarse de que los puntos de accidentes se encontraran efectivamente sobre o cerca de la red vial.

2.4.2 Calidad de los datos de hospitales (CLUES)

- Se revisó la coherencia de las coordenadas y la distribución espacial de las unidades de salud.
- Se verificó que las categorías utilizadas para filtrar hospitales (tipo de establecimiento y tipología) estuvieran bien codificadas y no presentaran valores ambiguos.
- Se contrastó la densidad de hospitales en zonas urbanas frente a zonas rurales, como una validación básica de plausibilidad.

2.4.3 Calidad de la red vial de OSM

- Se comprobó que la red vial de OSM fuera lo suficientemente densa y conectada en la Ciudad de México como para soportar análisis de ruteo.
- Se revisó la presencia de segmentos desconectados o errores topológicos que pudieran afectar el cálculo de rutas.

En resumen, la fase de entendimiento de los datos permitió caracterizar el contenido y la calidad de las fuentes involucradas, identificar sus principales limitaciones y asegurar que el modelado posterior (clustering, cálculo de distancias, índice de riesgo y ruteo seguro) se basara en una base de datos razonablemente consistente y representativa de la realidad.

1.4.2 Evaluación inicial de herramientas y técnicas

Las herramientas adecuadas identificadas fueron:

- **Técnicas:** clustering de densidad (HDBSCAN), distancia haversine, índices compuestos.
- **Herramientas:** Python, GeoPandas, hdbscan, scikit-learn, Plotly, Folium, Django, GitHub.

2.4 Preparación de los datos:

2.4.1 Selección de datos

Accidentes de tránsito

A partir de conjuntos de accidentes a nivel nacional, seleccionamos:

- Registros con coordenadas geográficas válidas.
- Accidentes cuyo estado corresponde a CDMX.
- Columnas necesarias para análisis temporal, espacial y de severidad (fecha, hora, latitud, longitud, tipo de accidente, números de muertos y heridos, tipo de vehículo, contexto vial, etc.).

En el README del proyecto se detalla este inventario de columnas y se documenta que el volumen inicial rondaba el millón de registros, de los cuales unos 78 000 corresponden a accidentes en CDMX después de los filtros básicos.

Red vial

Con OSMnx obtenemos la red vehicular de la ciudad, como un grafo dirigido en el que los nodos son intersecciones y las aristas son tramos de calle, con longitud, nombre de calle, tipo de vía y, cuando está disponible, velocidad máxima.

Hospitales:

Dada nuestra necesidad de datos de los establecimientos de salud a nivel nacional en México, su ubicación geográfica y su tipología, seleccionamos como fuente de datos a el Catálogo Único de Establecimientos de Salud (CLUES) para el período de diciembre 2023 (para que coincidiera con la temporalidad de los datos disponibles más recientes de accidentes víales). Las variables seleccionadas.

Categoría	Variables / Descripción
Ubicación	▪ `LATITUD` ▪ `LONGITUD` ▪ `geometry` (puntos en EPSG:4326)
Contexto administrativo	▪ `NOMBRE DE LA ENTIDAD` ▪ `NOMBRE DEL MUNICIPIO` ▪ `NOMBRE DE LA LOCALIDAD`
Tipo de establecimiento	▪ `NOMBRE TIPO ESTABLECIMIENTO` (e.g., *DE HOSPITALIZACIÓN*, *DE CONSULTA EXTERNA*, *DE APOYO*) ▪ `NOMBRE TIPOLOGÍA` (e.g., *HOSPITAL GENERAL*, *HOSPITAL GENERAL DE ZONA*, *HOSPITAL INTEGRAL*, *UNIDAD MÉDICA RURAL*)

Tabla 2.1: Variables seleccionadas del catálogo CLUES para caracterizar establecimientos de salud.

2.4.2 Limpieza y normalización

En la etapa de limpieza realizamos:

- Normalización de nombres de columnas (minúsculas, `snake_case`).
- Eliminación de duplicados exactos y registros con errores graves.
- Filtrado de registros sin coordenadas o sin hora/minutos válidos.
- Conversión de tipos (enteros, flotantes, fechas).

Para columnas numéricas de conteo (muertos, heridos, vehículos) imputamos valores faltantes con cero; para variables categóricas utilizamos una categoría de "desconocido" para no perder registros. En el README se reporta que tras esta etapa quedan alrededor de 78 mil accidentes limpios para CDMX.

En el caso de los hospitales, como nos interesaban únicamente aquellos capaces de atender una situación de urgencia, se aplicó un filtrado según el tipo de hospital:

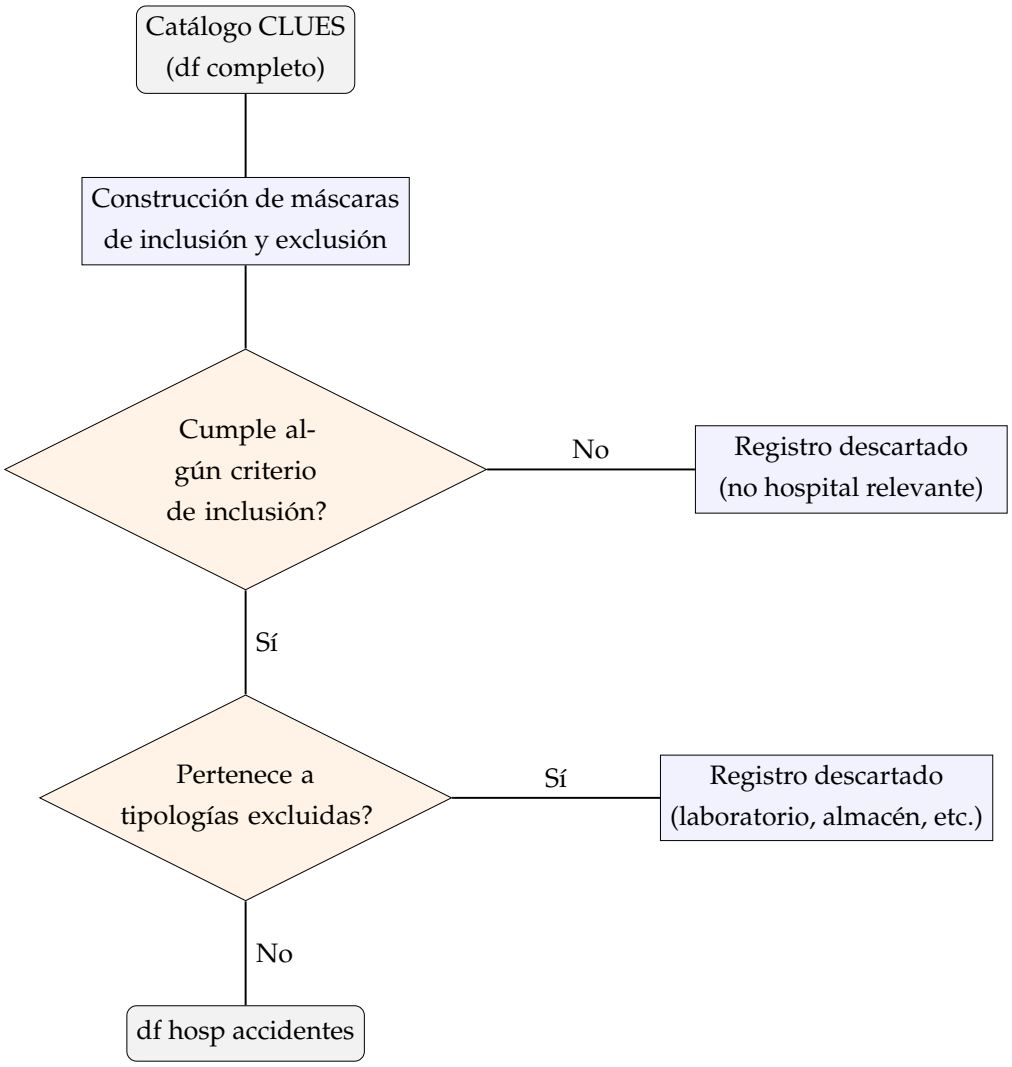

Figura 2.2: Diagrama de flujo del filtrado de hospitales relevantes para atención de accidentes carreteros.

Limpieza de datos geográficos

Para el caso particular de los datos geográficos a nivel nacional (accidentes y hospitales se aplicó el siguiente preprocesamiento:

- Limpieza de caracteres erróneos:
  Existían coordenadas mal escritas, por ejemplo con dos puntos decimales como "19.470.444"

- Acotamiento de la región de México:
  Se definió aproximadamente una región que contiene a México,con latitud entre 14 y 33, longitud entre -118 y -86, lo que nos permite eliminar datos sin sentido.
- Limpieza de outliers con HDBSCAN:
  Para el caso particular de los accidentes a nivel nacional, como el objetivo es buscar patrones de regiones con alta densidad de accidentes, fue menester eliminar el ruido, por lo que utilizamos el algoritmo HDBSCAN con hiperparámetros `min cluster size`=30, `min samples`=3, `metric`=euclidean, con lo que se limpiaron un 8,20 % de los datos.

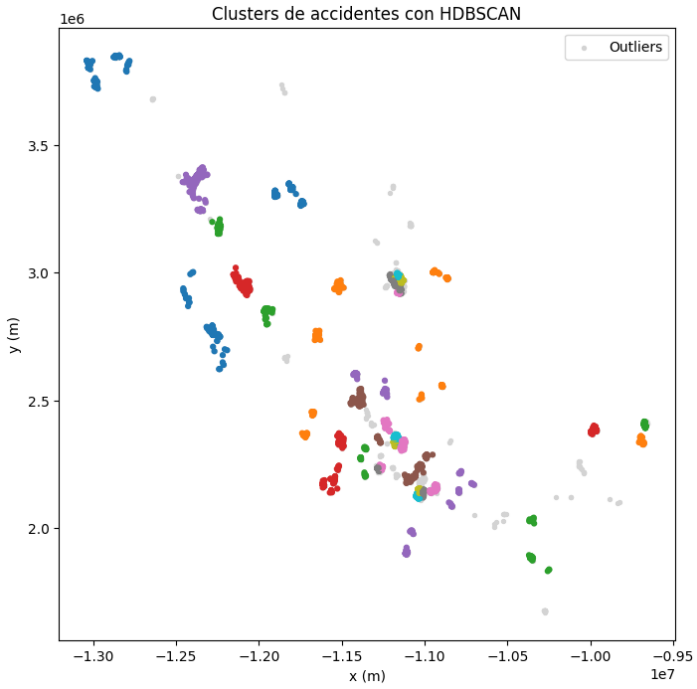

Figura 2.3: Clusterización de HDBSCAN para encontrar outliers.

2.4.3 Construcción de los datos

2.4.4 Variables temporales

A partir de las columnas de fecha y hora construimos una marca de tiempo `fechahora` y derivamos:

- Día de la semana (0 = lunes, 6 = domingo).
- Indicador de fin de semana.

- Franja horaria (madrugada, mañana, tarde, noche).
- Indicadores de hora pico (ventanas típicas de congestión en CDMX).

Estas variables capturan patrones temporales de riesgo y sirven tanto para los mapas por franja horaria como para los modelos de clasificación.

2.4.5 Variables de severidad

Para poder hablar de "qué tan grave" fue un accidente definimos un índice de severidad y luego lo traducimos a categorías y variables binarias; las fórmulas detalladas se presentan en el Capítulo 3.

2.4.6 Transformaciones espaciales

En la parte espacial trabajamos en dos niveles, tal como se planeó:

1. Una cuadrícula regular sobre la ciudad, a la que asignamos cada accidente para obtener conteos por celda y calcular hot spots.
2. El nivel de tramos de calle de la red vial. Proyectamos coordenadas a un sistema en metros y asociamos cada accidente al tramo de calle más cercano, guardando el identificador del tramo y la distancia.

A partir de esta asociación, agregamos estadísticas por tramo y después construimos índices de riesgo vial para cada unidad espacial. La base de accidentes, originalmente en formato `CSV`, se cargó en un `DataFrame` de `pandas` y posteriormente se convirtió en un `GeoDataFrame` de `GeoPandas`:

- Se identificaron las columnas de coordenadas (`LATITUD`, `LONGITUD`).
- Se construyó la geometría puntual mediante la función `points_from_xy(LONGITUD, LATITUD)`.
- Se asignó como sistema de referencia espacial el CRS geográfico `EPSG:4326` (WGS84), estándar para coordenadas en grados.

El resultado fue un `GeoDataFrame` de accidentes con una columna `geometry` que permite realizar operaciones espaciales (intersecciones, reproyecciones, medición de distancias, etc.).

GeoDataFrame de hospitales (CLUES)

De manera análoga, el catálogo CLUES de establecimientos de salud se transformó en un `GeoDataFrame`:

- A partir de las columnas `LATITUD` y `LONGITUD` se generó la columna `geometry` con puntos.
- Se fijó también el CRS `EPSG:4326`.
- Sobre este conjunto se aplicó un filtrado para conservar únicamente aquellas unidades con capacidad de hospitalización y atención de urgencias, con base en `NOMBRE TIPO ESTABLECIMIENTO` y `NOMBRE DE TIPOLOGIA`.

El `GeoDataFrame` resultante representa la oferta hospitalaria relevante para accidentes de tránsito.

Proyecciones a un CRS métrico y clustering

Para aplicar algoritmos de clustering espacial (por ejemplo, HDBSCAN) es conveniente trabajar en un sistema de referencia proyectado, donde las distancias se midan en metros y no en grados. Por ello:

- Los accidentes se reproyectaron de `EPSG:4326` a un CRS métrico, típicamente `EPSG:3857`.
- Sobre este `GeoDataFrame` proyectado se construyó la matriz de coordenadas en metros (componentes `x` y `y`).
- Con estas coordenadas se ejecutó HDBSCAN para obtener clusters de accidentes y detectar outliers espaciales.

Trabajar en metros facilita la interpretación de parámetros como radio de vecindad, tamaño mínimo de cluster o distancias internas a cada grupo.

Cálculo de distancias usando BallTree con métrica haversine

Para cuantificar la accesibilidad hospitalaria de cada accidente se calculó la distancia al hospital más cercano mediante la estructura `BallTree` de `scikit-learn`, utilizando la métrica *haversine*:

- Se aseguraron ambos conjuntos (accidentes y hospitales) en `EPSG:4326`, de modo que las coordenadas estuvieran en grados.

- Se extrajeron las latitudes y longitudes en arreglos numéricos y se transformaron a radianes.
- Se construyó el `BallTree` a partir de las coordenadas de los hospitales con métrica `"haversine"`.
- Para cada accidente se consultó el hospital más cercano mediante `tree.query` y se obtuvo la distancia angular en radianes.
- La distancia se convirtió a metros multiplicando por el radio medio de la Tierra ($R \approx 6\,371\,000$ m), y se almacenó en la columna `distancia_al_hospital`.

Esta combinación de `GeoDataFrames`, reproyecciones y cálculo eficiente de distancias sobre la esfera permite relacionar de manera precisa la ocurrencia de accidentes con la infraestructura hospitalaria disponible.

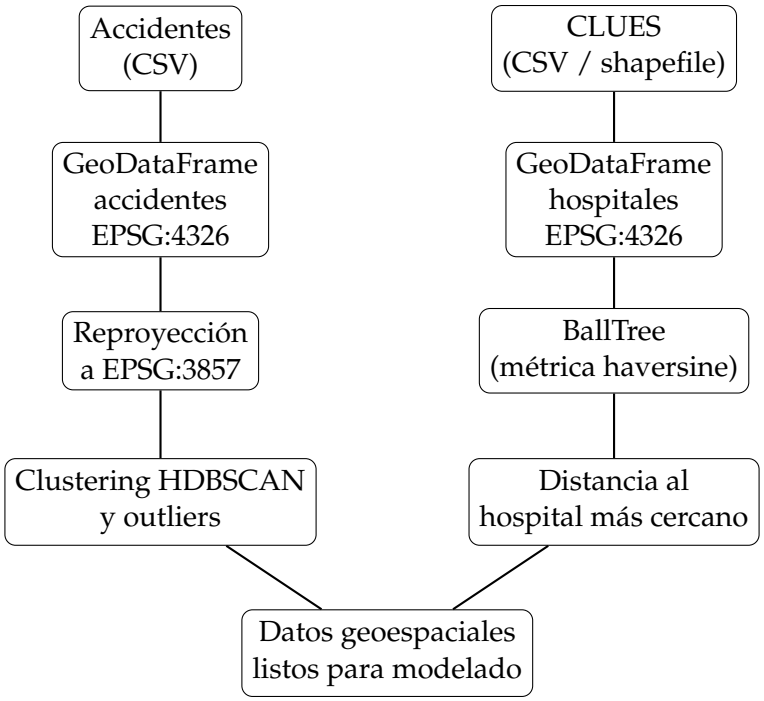

Figura 2.4: Flujo de construcción de los datos geográficos: creación de GeoDataFrames, reproyección a un CRS métrico y cálculo de distancias a hospitales mediante BallTree con métrica haversine.

2.4.7 Integración de datos

Después de limpiar año por año, unificamos todo el periodo 2019–2023 en un solo conjunto de accidentes, manteniendo la columna de año. Esto permite, por un lado, entrenar modelos

con todos los datos y, por otro, construir índices de riesgo que den más peso a los años recientes, como se propuso en el plan de trabajo.

2.5 Modelado (Modeling)

En esta fase se seleccionaron las técnicas de modelado, se definió la forma de evaluar los modelos y se construyeron los componentes centrales del sistema: el índice de riesgo y ruteo seguro en la Ciudad de México, y el modelo de identificación de zonas carreteras mal atendidas por hospitales a nivel nacional.

2.5.1 4.1 Selección de técnicas de modelado (Select Modeling Techniques)

4.1.1 Ciudad de México: índice de riesgo y ruteo

Para la parte del proyecto enfocada en la Ciudad de México se seleccionaron técnicas de modelado sobre redes:

- Representación de la red vial como grafo, con nodos (intersecciones) y aristas (tramos de calle).
- Cálculo de un **índice de riesgo por tramo** a partir de la agregación de accidentes sobre la red.
- Algoritmos de **ruteo en grafos** (por ejemplo, variantes de Dijkstra o A*) que permiten minimizar una función de costo combinando distancia/tiempo y riesgo.

[Por completar por el equipo de CDMX: descripción detallada del índice de riesgo, variables utilizadas, fórmula de agregación por tramo y algoritmo específico de ruteo empleado.**]**

4.1.2 Red carretera nacional: zonas mal atendidas por hospitales

Para la identificación de zonas carreteras mal atendidas por hospitales se combinaron técnicas de análisis geoespacial y clustering:

- **Clustering de densidad HDBSCAN** sobre las coordenadas métricas de los accidentes, para identificar clusters lineales en carreteras y separar outliers.
- **Búsqueda de vecinos más cercanos con BallTree** y métrica *haversine*, para calcular la distancia de cada accidente al hospital más cercano.

- Un **algoritmo heurístico de localización** que propone la ubicación de un hospital candidato en cada cluster mal atendido, utilizando un centro de gravedad ponderado por severidad (muertos y heridos) y un puntaje (*score*) que refleja tamaño, lejanía y extensión del cluster.

2.5.2 4.2 Diseño de la evaluación (Generate Test Design)

Dado que el objetivo principal del proyecto no es la predicción futura sino el diagnóstico espacial y la evaluación de escenarios, el diseño de prueba se planteó en términos de **comparación de escenarios** más que de particiones clásicas entrenamiento/prueba.

4.2.1 Escenarios en la Ciudad de México

En el caso del ruteo seguro en CDMX, la evaluación se basó en comparar, para un conjunto de pares origen–destino:

- La ruta más rápida o más corta, según la red vial.
- Una o varias **rutas alternativas de menor riesgo**, obtenidas al penalizar tramos con alto índice de riesgo.

Para cada par origen–destino se evaluaron métricas como:

- Distancia total y tiempo estimado de viaje.
- Suma del índice de riesgo a lo largo de la ruta.

[Por completar por el equipo de CDMX: conjunto de casos de prueba, definición exacta de las métricas de evaluación del índice de riesgo y resultados cuantitativos.**]**

4.2.2 Escenarios antes–después para hospitales en carreteras

Para la parte de hospitales en carreteras se diseñó una evaluación basada en tres escenarios por cluster de accidentes:

1. **Escenario actual:** sólo se consideran los hospitales existentes.
2. **Escenario con nuevo hospital:** sólo se considera el hospital candidato propuesto para ese cluster.
3. **Escenario combinado:** se consideran tanto los hospitales actuales como el hospital

candidato; para cada accidente se toma la distancia mínima a cualquiera de ellos.

A partir de estos escenarios se definieron indicadores clave (KPIs):

- Distancia media y mediana al hospital más cercano (en km).
- Reducción en el número de accidentes considerados «lejanos» (por ejemplo, a más de 10 km de un hospital).
- Número de accidentes y personas potencialmente beneficiadas (heridos + fallecidos) que mejoran su acceso a un hospital.

2.5.3 4.3 Construcción del modelo

4.3.1 Modelos para el índice de riesgo y ruteo en CDMX

[Por completar por el equipo de CDMX: descripción concreta de la construcción del índice de riesgo por tramo (variables, normalización, pesos) y de la integración con el algoritmo de ruteo (definición de la función de costo, implementación en la librería de grafos elegida y ejemplos de rutas generadas).**]**

4.3.2 Algoritmo de selección de clusters mal atendidos

Para la red carretera nacional, se partió del `GeoDataFrame` de accidentes con la columna `distancia_al_hospital` ya calculada mediante `BallTree`. El algoritmo de selección de clusters mal atendidos puede resumirse en los siguientes pasos:

1. **Reproyección a CRS métrico.**
   Se reproyectan los accidentes y los hospitales a un sistema de referencia métrico, como `EPSG:3857`, de manera que las distancias se expresen en metros. A partir de las columnas `x` e `y` de la geometría se construye la matriz de coordenadas para clustering.
2. **Aplicación de HDBSCAN.**
   Se ejecuta HDBSCAN sobre las coordenadas métricas de los accidentes con parámetros tales como `min_cluster_size` (p.ej. 15) y `cluster_selection_epsilon` (p.ej. 5 000 m), etiquetando cada punto con un identificador de cluster `cluster_hdb` y marcando los outliers con valor `-1`.
3. **Filtrado de clusters válidos.**
   Se itera sobre cada valor de `cluster_hdb` distinto de `-1`. Para cada cluster:

- Se descarta si su tamaño es menor a `min_cluster_size`.
- Se extraen las distancias `distancia_al_hospital` de los accidentes de ese cluster.
- Se calcula un percentil (por ejemplo, el percentil 25) de dichas distancias. Si este percentil es menor o igual a un umbral `distancia_minima` (p.ej. 10 km), el cluster se considera razonablemente atendido y se descarta.

4. **Ponderación por severidad.**
   Para cada cluster que pasa el filtro anterior, se define un peso de severidad por accidente, por ejemplo:

$$w = 1 + 5 \cdot \texttt{TOTMUERTOS} + 2 \cdot \texttt{TOTHERIDOS},$$

   usando ceros cuando no hay datos. Estos pesos permiten dar más importancia a accidentes con mayor número de víctimas.

5. **Cálculo del centro de gravedad ponderado.**
   En el CRS métrico, si (x_i, y_i) son las coordenadas de los accidentes del cluster y w_i sus pesos de severidad, el punto candidato (c_x, c_y) se calcula como:

$$c\_x = \frac{\sum\_i x\_i w\_i}{\sum\_i w\_i}, \qquad c\_y = \frac{\sum\_i y\_i w\_i}{\sum\_i w\_i}.$$

   Este punto representa un centro de gravedad de la severidad dentro del cluster.

6. **Cálculo del radio de cobertura.**
   Se mide la distancia desde el punto candidato a cada accidente del cluster y se define el *radio de cobertura* como la distancia máxima:

$$r\_\text{cobertura} = \max\_i \sqrt{(x\_i - c\_x)^2 + (y\_i - c\_y)^2}.$$

7. **Estadísticos de distancia a hospitales y puntaje.**
   Para cada cluster se calculan, a partir de `distancia_al_hospital`:
   - Distancia mínima y media al hospital más cercano.
   - Número de accidentes en el cluster.

   Con esta información se define un *score* que combina tamaño del cluster, lejanía y

extensión interna; por ejemplo:

$$\text{score} = (\text{número de accidentes}) + \frac{\text{distancia media al hospital}}{1000} + \frac{r\_\text{cobertura}}{1000},$$

expresando distancias en kilómetros para facilitar la interpretación.

8. **Conversión a WGS84 y selección de candidatos.**
   El punto candidato (c_x, c_y) se reproyecta nuevamente a `EPSG:4326` para obtener una geometría en latitud y longitud. Para cada cluster mal atendido se almacena un registro con:
   - Identificador de cluster.
   - Punto candidato (geometría).
   - Número de accidentes en el cluster.
   - Distancia mínima y media al hospital más cercano.
   - Radio de cobertura necesario.
   - Puntaje (*score*) calculado.

   Finalmente, se ordenan los candidatos por *score* de mayor a menor y se seleccionan los primeros `max_hospitales` como propuesta de nuevos hospitales prioritarios.

4.3.3 Resumen del algoritmo de candidatos a hospitales

La Tabla **??** resume las etapas principales del algoritmo de selección de candidatos a hospitales.

2.5.4 4.4 Evaluación del modelo

La evaluación del modelado se realizó en función de los objetivos de negocio:

- En CDMX, comprobando que el índice de riesgo identifica adecuadamente tramos peligrosos y que el ruteo seguro ofrece alternativas con menor riesgo sin incrementos desproporcionados de tiempo.
- En la red carretera nacional, cuantificando la mejora en accesibilidad hospitalaria al introducir los hospitales candidatos: reducciones en distancia media y mediana, disminución en el número de accidentes lejanos y estimación de personas potencial-

mente beneficiadas.

2.5.5 4.4 Evaluación del modelo

La evaluación del modelado se realizó en función de los objetivos de negocio:

- En CDMX, comprobando que el índice de riesgo identifica adecuadamente tramos peligrosos y que el ruteo seguro ofrece alternativas con menor riesgo sin incrementos desproporcionados de tiempo.
- En la red carretera nacional, cuantificando la mejora en accesibilidad hospitalaria al introducir los hospitales candidatos: reducciones en distancia media y mediana, disminución en el número de accidentes lejanos y estimación de personas potencialmente beneficiadas.

Escala del problema y adecuación del dataset. Para el análisis espaciotemporal en CDMX trabajamos con 32 139 accidentes ocurridos entre 2019 y 2023. A partir de este universo se preparó un subconjunto específico para modelado de Machine Learning con26 783 accidentes, en los que se pasó de 63 columnas originales a unas $\sim$40 *features* finales tras limpieza, codificación y selección de variables.

En términos de red vial, se identificaron 15 615 tramos únicos con accidentes, de los cuales 14 982 tienen al menos un accidente asociado en el periodo. Entre ellos, 623 tramos registran al menos una muerte y 1 301 tramos tienen cinco o más accidentes. Sobre esta base se definen los *top* 30 y *top* 50 tramos peligrosos usando severidad agregada.

Dimensión temporal y patrones de riesgo. Desde el punto de vista temporal, el modelo se apoya en una matriz real de accidentes 7×24 (días de la semana $\times$ horas del día). Un dato clave es que el pico máximo se observa el martes a las 08:00 AM, con 328 accidentes en esa casilla específica.

Agrupando por franjas horarias en todo el periodo 2019–2023, se obtiene:

- **Madrugada**: 5 092 accidentes, 3 083 celdas con riesgo ($\sim$15.8 % de los accidentes, pero con mortalidad de 3.91 %, relativamente alta).
- **Mañana**: 9 179 accidentes, 4 581 celdas con riesgo.
- **Tarde**: 9 590 accidentes, 4 914 celdas con riesgo ($\sim$29.8 %, la franja con más accidentes).

- **Noche**: 8 278 accidentes, 4 471 celdas con riesgo ($\sim$25.8 %, con mortalidad de 1.87 %).

Estos patrones alimentan directamente los multiplicadores horarios usados en el módulo de ruteo (`routingService`), donde proponemos penalizaciones como:

- Madrugada (0–6 h): incremento de riesgo entre +50 % y +80 %.
- Horario pico 7–9 AM: +35 % a +40 % de riesgo, alineado con el pico del martes 8 AM.
- Mediodía 12–14 h: +25 % de riesgo.
- Noche 18–24 h: incremento moderado de +5 % (menor mortalidad).

Puntos negros y concordancia con la severidad. En la capa de puntos negros (*black spots*) se utilizan 50 zonas reales con coordenadas exactas, no ficticias. Por ejemplo, la zona #1 está centrada en latitud 19.480500, longitud $-99{,}103500$ y registra 46 accidentes, 8 muertos y 26 heridos. Con nuestra función de severidad propuesta se obtiene:

$$\text{Severidad} = 10 \cdot \text{muertos} + 2 \cdot \text{heridos} + \text{accidentes} = 10 \cdot 8 + 2 \cdot 26 + 46 = 178.$$

Las 50 zonas cuentan con sus valores reales de accidentes, muertos y heridos, y la severidad se calcula con la misma lógica. Esto nos permite verificar que las zonas catalogadas como puntos negros no sólo concentran accidentes, sino también severidad alta de acuerdo con nuestra métrica.

Índice de riesgo y calidad de la asignación a la red vial. El índice de riesgo por celdas/tramos, calculado para el periodo 2019–2023, presenta las siguientes estadísticas globales:

- Número de celdas con riesgo: 62 062.
- Media del índice: $\approx 0{,}0055$.
- Desviación estándar: $\approx 0{,}0243$.
- Máximo observado: $\approx 0{,}9396$.

La distribución está fuertemente sesgada hacia valores bajos, pero con una cola de celdas/tramos donde el riesgo se acerca a 1 (puntos críticos). Esto es consistente con la intuición de que sólo una fracción pequeña de la ciudad concentra una proporción alta de severidad: en los tramos peligrosos de *top* riesgo agregamos una severidad total de 3 465,

lo que representa alrededor de 11.1 % de la severidad total.

En cuanto al *matching* de accidentes a la red vial, el análisis de distancias al tramo más cercano arroja:

- Distancia media ≈ 102.59 m.
- Mediana ≈ 2.06 m.
- 75 % de los accidentes a menos de $\sim$4.15 m del tramo asignado.

Es decir, la mayoría de los accidentes quedan prácticamente *encima* de un tramo de la red, por lo que el uso de ese tramo para ruteo que hicimos es bueno.

Clustering espacial y hot spots. En la etapa de clustering con DBSCAN (2019–2023) se obtuvieron:

- 299 clusters.
- 17 178 accidentes en clusters (53.4 %).
- 14 961 accidentes de ruido (46.6 %).

Con parámetros `eps = 200` metros y `min_samples = 20`, el tamaño medio de los clusters es de $\sim$57.5 accidentes, con una mediana de 32. El cluster más grande concentra 3 366 accidentes, mientras que el más pequeño (entre los válidos) tiene 3 accidentes. Este resultado confirma que existen zonas muy concentradas de riesgo y justifica el componente r^{cluster} del índice.

Para los hot spots mediante un estadístico tipo Getis-Ord, la distribución de celdas fue:

- 725 celdas no significativas.
- 17 hot spots al 95 % de confianza.
- 13 hot spots al 99 % de confianza.
- 0 cold spots identificados.

Esto encaja con la visión de que el problema está concentrado en unas cuantas áreas claramente críticas, más que en patrones simétricos de “frío y caliente”.

Autocorrelación espacial. La medida global de autocorrelación espacial (Moran’s I) arroja:

- Moran's I = 0.6837.
- p–value = 0.0010.
- Valor esperado $\approx -0{,}0013$.

Un valor tan positivo y un p–value tan bajo indican un patrón fuertemente agrupado, muy alejado de la hipótesis de aleatoriedad. Esto respalda el uso de análisis espacial y la idea de que el índice de riesgo realmente está capturando estructuras espaciales coherentes.

Resultados agregados de accidentes. En el periodo analizado (32 139 accidentes), el resumen global es:

- 728 personas fallecidas.
- 8 000 personas heridas.

En términos de tipo de evento:

- Accidentes leves (sólo daños materiales): 78.2 %.
- Accidentes con heridos: 19.92 %.
- Accidentes con muertos: 2.19 %.

Estos porcentajes sirvieron también para evaluar el desbalance de clases en el problema de predicción de gravedad.

Desempeño de los modelos de Machine Learning. Para la tarea de clasificar accidentes como *graves* (1) o *no graves* (0), se utilizó un conjunto de prueba con 5 357 accidentes:

- 5 197 accidentes no graves.
- 160 accidentes graves.

Las métricas obtenidas fueron:

- **Árbol de decisión**: accuracy $\approx 0{,}8871$ (88.71 %); precision $\approx 0{,}21$ y recall $\approx 0{,}98$ para la clase grave.
- **Random Forest** (mejor modelo): accuracy $\approx 0{,}9248$ (92.48 %), con precision $\approx 0{,}28$ y recall $\approx 0{,}97$ en la clase grave. En ranking global: Random Forest (0.9248), Stacking (0.9016), Regresión logística (0.8994), Árbol (0.8871).

Un accuracy de 92.48 % sobre 5 357 accidentes implica que el modelo clasifica correctamente

alrededor de 4 954 casos y se equivoca en unos 403. Dado el desbalance de clases, el hecho de mantener un recall alto en la clase grave es especialmente valioso para nuestro objetivo de seguridad.

Las probabilidades de gravedad estimadas por el Random Forest son la base para el componente r^{ml} y, por extensión, para el índice compuesto r_i^{comp} que se usa en el ruteo seguro.

Pasando a los resultados del hospital podemos resumirlo en los siguientes puntos:

- Reducciones en distancia media y mediana al hospital más cercano desde los tramos de alto riesgo.
- Disminución en el número de accidentes "lejanos" a servicios de salud (por encima de ciertos umbrales de distancia).
- Estimaciones del número de personas potencialmente beneficiadas al reubicar o agregar hospitales en zonas subatendidas.

En conjunto, estos resultados muestran que el modelo no sólo funciona bien en términos métricos (accuracy, recall, etc.), sino que también tiene impacto en los objetivos de negocio planteados: identificar puntos negros, priorizar intervenciones y proponer rutas que reduzcan la exposición al riesgo, tanto en un contexto urbano (CDMX) como en escenarios de red carretera y *accesibilidad hospitalaria*.

2.6 Evaluación

En esta fase se evalúa la calidad de los modelos construidos y, sobre todo, su capacidad para responder a los objetivos de negocio planteados en la Fase 1. La evaluación se realiza tanto desde un punto de vista técnico (métricas, coherencia de los resultados) como desde la perspectiva del negocio (utilidad para la toma de decisiones en seguridad vial y planificación hospitalaria).

2.6.1 5.1 Evaluación de los resultados (Evaluate Results)

5.1.1 Ciudad de México: índice de riesgo y ruteo seguro

Para la parte de la Ciudad de México, la evaluación se centró en comparar distintas rutas entre pares origen–destino, considerando:

- **Ruta de referencia:** ruta más corta o más rápida sobre la red vial, sin penalización por riesgo.
- **Rutas de menor riesgo:** rutas alternativas generadas al incorporar el índice de riesgo vial en la función de costo del algoritmo de ruteo.

Para cada origen–destino se analizaron métricas tales como:

- Distancia total recorrida (en km).
- Tiempo de viaje estimado (minutos, si el modelo lo consideró).
- Suma del índice de riesgo a lo largo de la ruta (medida agregada de exposición al riesgo).

[Por completar por el equipo de CDMX: descripción de los casos de prueba, valores numéricos comparativos de las rutas (por ejemplo, cuánto aumenta el tiempo y cuánto disminuye el riesgo en promedio) y discusión de ejemplos ilustrativos de rutas donde el cambio es especialmente relevante.**]**

5.1.2 Red carretera nacional: cobertura hospitalaria antes–después

Para el análisis de hospitales en carreteras se definieron explícitamente tres escenarios por cada cluster de accidentes:

1. **Escenario actual:** sólo se consideran los hospitales existentes.
2. **Escenario con nuevo hospital:** sólo se considera el hospital candidato propuesto para ese cluster.
3. **Escenario combinado:** se consideran tanto los hospitales actuales como el nuevo hospital candidato; para cada accidente se toma la distancia mínima a cualquiera de ellos.

A partir de estos escenarios se construyeron indicadores clave (KPIs), entre ellos:

- **Distancia media y mediana al hospital más cercano**, tanto en el escenario actual

como en el combinado. Esto permite cuantificar la reducción de distancia promedio que introduce el hospital sugerido.

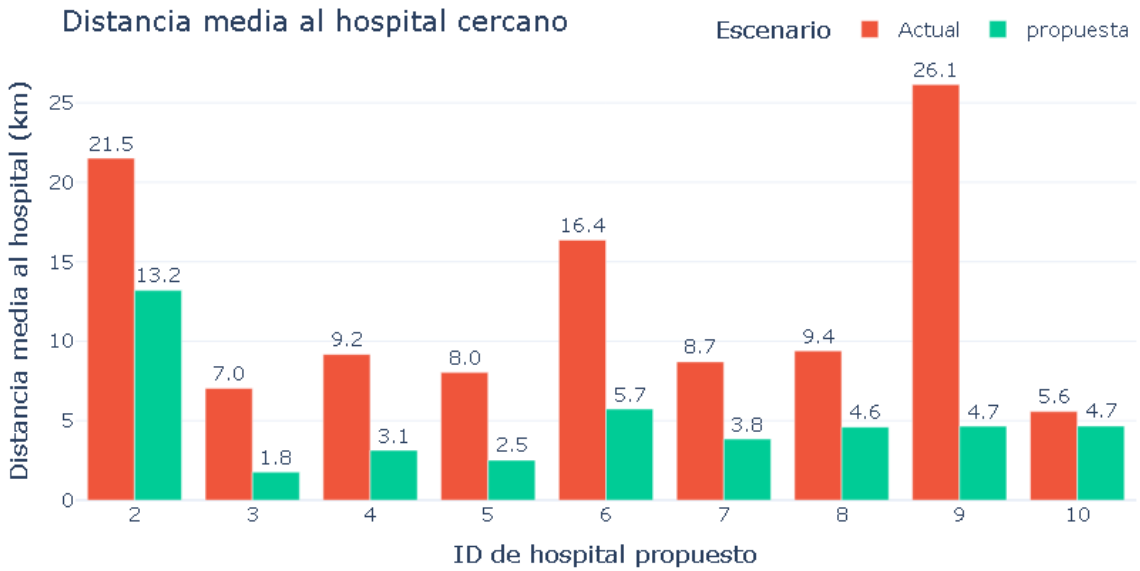

Figura 2.5: En esta visualización se identifica la mejoría lograda entre la distancia media entre los antiguos hospitales y los propuestos. Nótese que la distancia media de los hospitales propuestos es en general menor

- **Número de accidentes lejanos** (por ejemplo, a más de 10 km de un hospital) antes y después. La diferencia refleja qué tanto disminuye el número de eventos potencialmente desatendidos.
- **Personas potencialmente beneficiadas**, calculadas como la suma de heridos y fallecidos históricos en los accidentes asociados a cada cluster.
- **Prioridad por hospital propuesto**, evaluada a partir del número de heridos y muertos en el área de influencia de cada candidato, así como de la mejora en distancias.

Estos KPIs se visualizaron mediante:

- **Gráficas de barras** comparando distancias medias y medianas en el escenario actual frente al escenario actual + nuevo hospital.
- **Gráficas de accidentes lejanos** (antes y después), que muestran explícitamente cuántos accidentes pasan de estar por encima del umbral a estar dentro de la zona de cobertura.
- **Gráficas de prioridad de hospitales propuestos**, donde por cada hospital candidato se representan barras separadas para número de heridos y muertos, de manera que el usuario pueda identificar rápidamente cuáles ubicaciones tienen mayor impacto

potencial.

- **Mapas interactivos** con la distribución de accidentes, hospitales actuales y candidatos, lo cual permite una validación visual de que los hospitales sugeridos se ubican efectivamente sobre clusters de accidentes previamente alejados de la infraestructura hospitalaria.

En conjunto, estos resultados muestran que, para los clusters seleccionados como mal atendidos, la introducción de hospitales candidatos:

- Mantiene o mejora la distancia media y mediana al hospital más cercano.
- Reduce el número de accidentes catalogados como «lejanos» con respecto al umbral definido.
- Identifica un conjunto de ubicaciones donde la cantidad de personas potencialmente beneficiadas (heridos + muertos) es relativamente alta, justificando su prioridad en términos de planificación de infraestructura.

2.6.2 5.2 Revisión del proceso de modelado (Review Process)

La evaluación no sólo se centró en los resultados numéricos, sino también en la **coherencia del proceso de modelado** con los objetivos de negocio y con las restricciones de datos.

5.2.1 Fortalezas del enfoque

Entre las principales fortalezas del enfoque adoptado se encuentran:

- El uso de **HDBSCAN** permitió identificar clusters de accidentes con formas arbitrarias (incluyendo tramos lineales en carreteras), y al mismo tiempo etiquetar outliers que podían distorsionar el análisis.
- El uso de **BallTree** con métrica haversine para el cálculo de distancias garantizó consultas eficientes aun con un número elevado de puntos (accidentes y hospitales).
- La construcción de KPIs centrados en **distancias**, **número de accidentes lejanos** y **personas potencialmente beneficiadas** facilitó la conexión directa entre los resultados del modelo y preguntas relevantes de política pública.
- Las visualizaciones interactivas (mapas y dashboards) reforzaron la interpretación de los resultados, permitiendo detectar rápidamente patrones espaciales y justificar las

decisiones de priorización.

5.2.2 Limitaciones y posibles sesgos

Al mismo tiempo, se identificaron varias limitaciones y posibles fuentes de sesgo:

- La calidad de los resultados depende fuertemente de la **calidad de los datos de entrada**: errores en la geocodificación de accidentes o de hospitales pueden afectar el cálculo de distancias y la detección de clusters.
- El umbral de **distancia mínima** (por ejemplo, 10 km) y el percentil utilizado para decidir si un cluster está mal atendido son **parámetros de diseño** que podrían refinarse con la opinión de expertos en salud y protección civil.
- El modelo de distancia no incorpora explícitamente **tiempos de traslado reales**, tráfico ni disponibilidad de servicios de emergencia; se trabaja con distancia geográfica como proxy.
- En la parte de ruteo para CDMX, la evaluación depende de supuestos sobre tiempos de viaje y del conjunto de pares origen–destino seleccionados para las pruebas.

[Por completar por el equipo de CDMX: discusión de limitaciones específicas del índice de riesgo, cobertura temporal de los datos de accidentes en la ciudad, y posibles sesgos en la selección de tramos viales o tipos de accidentes.**]**

2.6.3 5.3 Determinación de los siguientes pasos (Determine Next Steps)

Con base en la evaluación técnica y de negocio, se identificaron posibles líneas de trabajo futuro y pasos siguientes:

5.3.1 Mejoras al modelo de cobertura hospitalaria

- Incorporar **tiempo de viaje estimado** (por ejemplo, a partir de velocidades típicas por tipo de vía) en lugar de usar únicamente distancia geodésica, para obtener una medida más realista de accesibilidad.
- Ajustar los parámetros de HDBSCAN y de los umbrales de distancia con apoyo de **expertos en salud y planeación**, de manera que los clusters mal atendidos reflejen mejor zonas críticas desde el punto de vista clínico.

- Extender el análisis a **series de tiempo**, evaluando cómo cambian las zonas de riesgo y la pertinencia de nuevos hospitales a lo largo de varios años.
- Integrar datos adicionales, como **capacidad real de los hospitales** (número de camas, servicios especializados) para refinar el criterio de selección de unidades relevantes.

5.3.2 Integración y despliegue del sistema de ruteo seguro

- Consolidar la integración del modelo de ruteo seguro en una aplicación web (por ejemplo, basada en Django), de modo que usuarios no técnicos puedan consultar rutas alternativas de menor riesgo.
- Implementar mecanismos para actualizar periódicamente los datos de accidentes y recalcular el índice de riesgo y los clusters, permitiendo que el sistema evolucione con información más reciente.
- Evaluar la posibilidad de incluir **perfiles de usuario** (por ejemplo, transporte de pacientes, vehículos de emergencia, usuarios particulares) con distintas preferencias de balance entre tiempo y riesgo.

5.3.3 Uso potencial en política pública

Finalmente, los resultados obtenidos abren la puerta a:

- Discutir, con autoridades de salud y transporte, el uso de los mapas de riesgo y de los candidatos a hospitales como **insumo para la priorización de inversión** en infraestructura.
- Desarrollar **escenarios de planificación** en los que se simule la construcción de distintos conjuntos de hospitales y se compare su impacto en la cobertura hospitalaria.
- Utilizar el flujo construido como **plantilla metodológica** para otras regiones del país o para otros tipos de incidentes que requieran atención de urgencias.

2.7 Minería de datos

2.7.1 Análisis espacial

El análisis espacial lo abordamos con tres herramientas:

- DBSCAN, para encontrar clusters de alta densidad de accidentes.
- Getis-Ord Gi*, para identificar hot spots en la cuadrícula.
- Moran's I, para medir autocorrelación espacial.

2.7.2 Modelos supervisados

Para modelar la gravedad trabajamos con un problema de clasificación binaria (grave / no grave), entrenando modelos como árboles de decisión, Random Forest, regresión logística y un *stacking ensemble*. La probabilidad estimada se usa como una capa más en el índice de riesgo vial.

2.8 Resumen de logros por etapa KDD

Para dejar claro qué se logró en cada etapa, resumimos:

- **Selección**: integración de cinco años de datos de accidentes y construcción de la red vial completa de CDMX.
- **Preprocesamiento**: depuración de más de 4 000 registros con problemas graves y normalización de variables clave.
- **Transformación**: diseño de variables temporales, de severidad, espaciales y de contexto; asociación de cada accidente a una celda y a un tramo de calle.
- **Minería de datos**: implementación de DBSCAN, hot spots y Moran's I; entrenamiento de varios modelos de clasificación y selección de un modelo final tipo *stacking*.
- **Interpretación**: construcción de índices de riesgo por zona y por tramo, clasificación de niveles de riesgo y uso de esos índices en un sistema de ruteo que compara rutas por seguridad.

Paso	Pseudocódigo
1	**Entrada:** GeoDataFrame de accidentes A con columna `distancia_al_hospital` y variables `TOTMUERTOS`, `TOTHERIDOS`; GeoDataFrame de hospitales H filtrados del CLUES; parámetros `min_cluster_size`, `distancia_minima`, `max_hospitales`.
2	Reproyectar A y H a un CRS métrico (por ejemplo, `EPSG:3857`). Extraer coordenadas (x_i, y_i) de cada accidente en A.
3	Aplicar HDBSCAN sobre las coordenadas (x_i, y_i) para obtener etiquetas de cluster `cluster_hdb`(i).
4	Inicializar lista vacía `Candidatos`.
5	**Para cada** cluster c distinto de -1 **hacer**: Definir $S_c = \{i \mid \texttt{cluster\_hdb}(i) = c\}$. **Si** $\lvert S_c \rvert <$ `min_cluster_size` **entonces continuar** (descartar cluster pequeño). Calcular el percentil 25 de `distancia_al_hospital` para $i \in S_c$. **Si** dicho percentil $\leq$ `distancia_minima` **entonces continuar** (cluster razonablemente atendido). Para cada $i \in S_c$ calcular peso de severidad: $w_i = 1 + 5 \cdot \texttt{TOTMUERTOS}_i + 2 \cdot \texttt{TOTHERIDOS}_i$. Calcular centro de gravedad ponderado: $c_x = \dfrac{\sum_{i \in S_c} x_i w_i}{\sum_{i \in S_c} w_i}, \quad c_y = \dfrac{\sum_{i \in S_c} y_i w_i}{\sum_{i \in S_c} w_i}.$ Calcular radio de cobertura: $r_c = \max_{i \in S_c} \sqrt{(x_i - c_x)^2 + (y_i - c_y)^2}.$ Calcular distancias mínima y media al hospital más cercano usando `distancia_al_hospital` en S_c. Definir un puntaje de prioridad: $\text{score}_c = \lvert S_c \rvert + \dfrac{\text{dist\_media}}{1000} + \dfrac{r_c}{1000}.$ Almacenar en `Candidatos` el registro asociado al cluster c: $(c,\ (c_x, c_y),\ \lvert S_c \rvert,\ \text{dist\_min},\ \text{dist\_media},\ r_c,\ \text{score}_c)$.
6	Reproyectar los puntos candidatos (c_x, c_y) de cada registro en `Candidatos` de regreso a CRS geográfico `EPSG:4326`.
7	Ordenar `Candidatos` en orden descendente por score_c. Seleccionar los primeros `max_hospitales` como candidatos finales.

Tabla 2.2: Pseudocódigo del algoritmo para proponer candidatos a nuevos hospitales en clusters de accidentes mal atendidos.

3

Modelado Cuantitativo e Índices de Riesgo

En este capítulo presentamos las fórmulas clave que utilizamos para formalizar la severidad de los accidentes, los distintos componentes de riesgo y las funciones de costo usadas en el ruteo. Incluimos tanto el índice por zona del plan de trabajo como el índice por tramo que usamos en el ruteo final.

3.1 Índice de severidad de accidentes

Denotamos por:

- m = número de personas fallecidas en el accidente.
- h = número de personas heridas.

Definimos un índice de severidad escalar como:

$$\text{sev} = 10\,m + 3\,h. \tag{3.1}$$

Con base en este índice construimos una clasificación categórica:

$$\text{Severidad} = \begin{cases} \text{LEVE}, & \text{si sev} = 0, \\ \text{MODERADA}, & \text{si } 1 \leq \text{sev} \leq 9, \\ \text{GRAVE}, & \text{si } 10 \leq \text{sev} \leq 19, \\ \text{MUY GRAVE}, & \text{si sev} \geq 20. \end{cases} \tag{3.2}$$

Y además definimos una variable binaria de gravedad:

$$y = \begin{cases} 1, & \text{si sev} \geq 10, \\ 0, & \text{en otro caso,} \end{cases} \tag{3.3}$$

que usamos como *target* en los modelos supervisados.

También agregamos indicadores lógicos derivados:

$$\text{hay\_muertos} = \mathbb{I}(m > 0), \tag{3.4}$$

$$\text{hay\_heridos} = \mathbb{I}(h > 0), \tag{3.5}$$

$$\text{solo\_danos\_materiales} = \mathbb{I}(m = 0 \wedge h = 0), \tag{3.6}$$

donde $\mathbb{I}(\cdot)$ es la función indicadora.

3.2 Índice de riesgo por zona (visión del plan de trabajo)

En el plan original definimos un índice de riesgo por zona z (celda o tramo) a partir de tres métricas base en un periodo T:

- N_z: número de accidentes en la zona.
- S_z: severidad total en la zona.
- F_z: número de accidentes fatales en la zona.

Cada métrica se normaliza a escala 0–1 mediante min–max:

$$N'_z = \frac{N_z - \text{mín}_j\, N_j}{\text{máx}_j\, N_j - \text{mín}_j\, N_j + \varepsilon}, \tag{3.7}$$

$$S'_z = \frac{S_z - \text{mín}_j\, S_j}{\text{máx}_j\, S_j - \text{mín}_j\, S_j + \varepsilon}, \tag{3.8}$$

$$F'_z = \frac{F_z - \text{mín}_j\, F_j}{\text{máx}_j\, F_j - \text{mín}_j\, F_j + \varepsilon}. \tag{3.9}$$

Con estas cantidades se construye un **índice de riesgo compuesto por zona**:

$$\text{riesgo}_z = 0{,}4\, N'_z + 0{,}4\, S'_z + 0{,}2\, F'_z. \tag{3.10}$$

Los pesos priorizan zonas con muchos choques y con accidentes de alta severidad o fatalidad. Esta definición se utilizó para mapas de calor y para tablas de "top zonas más peligrosas".

3.3 Componente temporal y ponderación por año

El riesgo no es estático: depende de la hora, el día y el año. Para incorporar la dimensión temporal consideramos:

- Franjas horarias: Madrugada (0–5), Mañana (6–11), Tarde (12–17), Noche (18–23).
- Día de la semana: lunes–domingo.
- Ponderación de años: mayor peso a los años recientes.

En particular, para el riesgo por año se propuso un esquema de pesos, por ejemplo:

$$w_{2019} = 0{,}10, \quad w_{2020} = 0{,}15, \quad w_{2021} = 0{,}20, \tag{3.11}$$

$$w_{2022} = 0{,}25, \quad w_{2023} = 0{,}30, \tag{3.12}$$

con $\sum_a w_a = 1$. Para una zona z, el riesgo ponderado por año se define como:

$$\text{riesgo}_z^{\text{ponderado}} = \sum_a w_a \, \text{severidad}_{z,a}. \tag{3.13}$$

De manera análoga se puede definir una función $\text{riesgo}_z(\text{hora}, \text{dia\_semana})$ para construir mapas de puntos negros específicos de horario y día, que luego sirven como entrada para el módulo de ruteo.

3.4 Riesgo histórico por tramo vial

Sea i un tramo de la red vial. Denotamos por A_i el número total de accidentes asociados al tramo. Para llevar este conteo a una escala común 0–100, normalizamos de forma lineal:

$$r_i^{\text{hist}} = 100 \cdot \frac{A_i - \min_j A_j}{\max_j A_j - \min_j A_j + \varepsilon}. \tag{3.14}$$

También trabajamos con variantes donde sólo se consideran accidentes graves o se incorpora la severidad total, pero manteniendo el mismo esquema de normalización.

3.5 Riesgo por clustering (DBSCAN)

Al aplicar DBSCAN, cada accidente queda etiquetado con un identificador de cluster k o como ruido. Denotamos por C_k al conjunto de puntos en el cluster k y por $|C_k|$ a su tamaño. Definimos un riesgo a nivel de cluster:

$$r_k^{\text{cluster}} = 100 \cdot \frac{|C_k| - \min_j |C_j|}{\max_j |C_j| - \min_j |C_j| + \varepsilon}. \tag{3.15}$$

Para un accidente individual que pertenece al cluster k, le asignamos el riesgo r_k^{cluster}. Para accidentes marcados como ruido usamos $r^{\text{cluster}} = 0$. A nivel de tramo vial usamos el promedio de los riesgos de cluster de los accidentes asociados.

3.6 Riesgo por modelos de Machine Learning

Los modelos supervisados se entrenan para predecir la variable binaria y (grave / no grave) a partir de un vector de características x que incluye información temporal, espacial y de contexto.

Sea $\hat{p}(y = 1 \mid x)$ la probabilidad predicha por el modelo final (*stacking*). Definimos el **riesgo ML** como:

$$r^{\text{ml}} = 100 \cdot \hat{p}(y = 1 \mid x). \tag{3.16}$$

Para llevar este riesgo al nivel de tramo vial, calculamos para cada tramo i el promedio de r^{ml} de los accidentes asociados:

$$r_i^{\text{ml}} = \frac{1}{n_i} \sum_{a \in \mathcal{A}_i} r_a^{\text{ml}}, \tag{3.17}$$

donde $\mathcal{A}_i$ es el conjunto de accidentes asociados al tramo i y $n_i = |\mathcal{A}_i|$.

3.7 Índice de riesgo compuesto por tramo

Los tres componentes de riesgo a nivel de tramo son:

- r_i^{hist}: riesgo histórico.
- r_i^{cluster}: riesgo por clustering.
- r_i^{ml}: riesgo por ML.

Definimos el **índice de riesgo compuesto** como:

$$r_i^{\text{comp}} = w_{\text{hist}}\, r_i^{\text{hist}} + w_{\text{cluster}}\, r_i^{\text{cluster}} + w_{\text{ml}}\, r_i^{\text{ml}}, \tag{3.18}$$

donde usamos los pesos:

$$w_{\text{hist}} = 0{,}6, \quad w_{\text{cluster}} = 0{,}1, \quad w_{\text{ml}} = 0{,}3, \quad w_{\text{hist}} + w_{\text{cluster}} + w_{\text{ml}} = 1. \tag{3.19}$$

Este índice $r_i^{\text{comp}} \in [0, 100]$ se asocia a cada tramo de la red vial y sirve como base para las funciones de costo de ruteo.

3.8 Funciones de costo para ruteo seguro

Sea L_i la longitud del tramo i (en metros). Consideramos tres funciones de costo:

1. Ruta más corta.

$$c_i^{\text{dist}} = L_i. \tag{3.20}$$

2. Ruta balanceada. Normalizamos la longitud:

$$\tilde{L}_i = \frac{L_i - \text{mín}_j\, L_j}{\text{máx}_j\, L_j - \text{mín}_j\, L_j + \varepsilon}. \tag{3.21}$$

Luego definimos:

$$c_i^{\text{bal}} = \alpha\, \tilde{L}_i + (1-\alpha)\, \frac{r_i^{\text{comp}}}{100}, \tag{3.22}$$

con $\alpha \in (0,1)$, por ejemplo $\alpha = 0{,}5$.

3. Ruta más segura. En la ruta más segura aumentamos la penalización al riesgo con un exponente $\beta > 1$:

$$c_i^{\text{seg}} = \alpha\, \tilde{L}_i + (1-\alpha)\, \left(\frac{r_i^{\text{comp}}}{100}\right)^{\beta}, \tag{3.23}$$

con β típico de 2 o 3. A nivel de ruta, el costo total es la suma de los costos de los tramos.

4

Análisis Espacial y Minería de Datos

4.1 DBSCAN: detección de puntos negros

DBSCAN es un algoritmo de clustering basado en densidad. Dados un radio ε y un mínimo de puntos `min_samples`, define para cada punto p la vecindad:

$$N_\varepsilon(p) = \{q : d(p,q) \leq \varepsilon\}, \tag{4.1}$$

donde $d(\cdot,\cdot)$ es la distancia euclidiana en coordenadas proyectadas. Elegimos aproximadamente $\varepsilon = 200$ metros y `min_samples` $= 20$, lo que nos dio 299 clusters y permitió identificar zonas de alta concentración de accidentes.

4.2 Hot spots con Getis-Ord Gi*

Para identificar zonas con concentración estadísticamente alta de accidentes sobre una cuadrícula espacial utilizamos la estadística local Getis-Ord Gi*. Su fórmula general es:

$$G_i^* = \frac{\sum_j w_{ij} x_j - \bar{X} \sum_j w_{ij}}{S \sqrt{\frac{n \sum_j w_{ij}^2 - \left(\sum_j w_{ij}\right)^2}{n-1}}}, \tag{4.2}$$

donde x_j es el número de accidentes en la celda j y w_{ij} son los pesos espaciales. Las celdas con G_i^* alto y significativo se clasifican como *hot spots*.

4.3 Autocorrelación espacial (Moran's I)

Para evaluar si la distribución de accidentes presenta autocorrelación espacial calculamos Moran's I global:

$$I = \frac{n}{W} \cdot \frac{\sum_i \sum_j w_{ij}(x_i - \bar{x})(x_j - \bar{x})}{\sum_i (x_i - \bar{x})^2}, \tag{4.3}$$

con $W = \sum_i \sum_j w_{ij}$. Los valores obtenidos fueron positivos y significativos, lo que confirma que los accidentes tienden a agruparse y justifica el uso de técnicas espaciales.

4.4 Modelos supervisados de gravedad

Con la variable binaria y y el conjunto de características x entrenamos distintos modelos de clasificación. De manera general, cada modelo estima:

$$\hat{p}(y = 1 \mid x) = f_\theta(x), \tag{4.4}$$

donde θ representa los parámetros del modelo. Probamos árboles de decisión, Random Forest, regresión logística y un *stacking* que combina los tres. El stacking resultó ser el que mejor desempeño tuvo en términos de accuracy y AUC-ROC, por lo que lo usamos para definir r^{ml}.

5

Arquitectura del Sistema de Ruteo

5.1 Vista de alto nivel

El sistema completo se puede ver como un pipeline que va transformando datos crudos en rutas seguras:

1. Un primer bloque de **procesamiento** limpia los datos de accidentes, unifica años, genera variables derivadas y asocia cada accidente a celdas y tramos.
2. Un bloque de **análisis espacial** calcula clustering, hot spots y medidas de autocorrelación, generando capas de riesgo.
3. Un bloque de **modelado ML** hace selección de características, entrena modelos, calcula probabilidades de gravedad y construye el índice de riesgo compuesto a nivel de tramo.
4. Un bloque de **ruteo** carga la red vial junto con r^{comp}, define las funciones de costo y calcula rutas más corta, balanceada y más segura entre pares de nodos.

5.2 Del índice de riesgo al ruteo

Una vez que cada tramo tiene asociado un costo de riesgo r_i^{comp}, definimos distintas funciones de costo y, para cada una, corremos un algoritmo tipo Dijkstra sobre el grafo ponderado. En el caso de estudio fijamos un origen cercano al Zócalo y un destino en la zona de Polanco y comparamos las métricas de las rutas resultantes.

6

Resultados y Hallazgos

6.1 Patrones espaciales y puntos negros

Los resultados de DBSCAN muestran que más de la mitad de los accidentes pertenecen a algún cluster de densidad, lo que confirma la existencia de **puntos negros** claros en la ciudad. El análisis de hot spots añade una capa estadística que permite priorizar zonas donde la concentración de accidentes es significativamente alta.

A partir del índice de riesgo por zona construimos mapas de calor y una tabla de las 20 zonas más peligrosas, con información de coordenadas aproximadas, accidentes totales, muertos, heridos e índice de riesgo. Esto responde directamente a los productos esperados en el plan de trabajo.

6.2 Desempeño de los modelos predictivos

El modelo *ensemble* que combina árbol de decisión, Random Forest y regresión logística logra una buena discriminación entre accidentes graves y no graves, a pesar del fuerte desbalance de clases. A nivel de metricas obtenemos un AUC-ROC cercano a 0.86, lo que indica que la probabilidad estimada de gravedad tiene buena capacidad de separación.

6.3 Rutas: distancia vs seguridad

Al incorporar el índice de riesgo compuesto como peso en el grafo vial, calculamos tres rutas distintas entre el Zócalo y Polanco:

- **Ruta más corta**: minimiza sólo longitud.

- **Ruta balanceada**: cede un poco en distancia para reducir riesgo promedio.
- **Ruta más segura**: evita tramos con r^{comp} alto, incluso si eso implica rodear ciertas zonas.

En el ejemplo que probamos, la ruta más segura es algunos minutos más lenta que la ruta más corta, pero reduce de forma importante el riesgo máximo y el número de tramos catalogados como muy peligrosos. Esto abre la puerta a escenarios donde sacrificar algo de tiempo se justifica para reducir la probabilidad de un accidente grave.

7

Aplicación a Traslados Hospitalarios

7.1 Motivación en contexto de salud

Más allá de conductores particulares, hay un contexto en el que las rutas son críticas: los **traslados hospitalarios**. Aquí no sólo importa llegar rápido, sino llegar de forma segura y confiable, reduciendo la probabilidad de accidentes durante el traslado de pacientes o de personal médico.

7.2 Hospitales como nodos especiales

En términos de red vial, los hospitales pueden verse como nodos de alto interés:

- Puntos de destino para traslados desde el lugar del accidente.
- Nodos de referencia para traslados entre hospitales (por ejemplo, de un hospital general a uno de alta especialidad).
- Centros alrededor de los cuales se pueden estudiar zonas de riesgo específicas y rutas preferentes.

7.3 Escenarios de uso

Algunos escenarios que queremos explorar son:

- Rutas desde zonas de alta incidencia de accidentes a hospitales cercanos, comparando ruta más corta vs ruta más segura.
- Traslados inter-hospitalarios en los que se eviten tramos con historial de choques graves.

- Simulación de posibles mejoras si ciertas vialidades se intervinieran para reducir el riesgo.

8

Arquitectura de una Aplicación Web de Ruteo Seguro

8.1 Visión general

Con el índice de riesgo ya calculado y la red vial preparada, planteamos una arquitectura de aplicación web para exponer el sistema de ruteo seguro a usuarios finales. A grandes rasgos:

- **Frontend**: aplicación web con mapa interactivo donde el usuario elige origen y destino y visualiza las rutas.
- **Backend**: servicio en Python que carga el grafo vial con los pesos de riesgo, ejecuta el algoritmo de ruteo y devuelve las rutas y sus métricas.
- **Capa de datos**: archivos precalculados con la red vial, los índices de riesgo por tramo y, opcionalmente, los modelos de ML para *scoring* en tiempo real.

8.2 Flujo típico de consulta

Un flujo típico de uso sería:

1. La persona usuaria selecciona un origen y un destino en el mapa (o los escribe como direcciones).
2. El frontend llama al backend solicitando rutas entre esos puntos.
3. El backend localiza los nodos de la red vial más cercanos, ejecuta el algoritmo de ruteo con las funciones de costo definidas y regresa las rutas como secuencias de coordenadas

y métricas.

4. El frontend dibuja las rutas con estilos distintos y presenta una tabla comparativa (distancia, riesgo promedio, riesgo máximo).

9

Implementación y Reproducibilidad

9.1 Tecnologías utilizadas

El proyecto está desarrollado principalmente en Python, utilizando:

- **pandas, numpy**: manipulación y agregación de datos.
- **geopandas, osmnx**: análisis espacial y manejo de la red vial.
- **scikit-learn**: modelos de Machine Learning y selección de características.
- **libpysal, esda**: análisis espacial (Moran's I, Getis-Ord).
- **folium, matplotlib**: visualización de mapas y gráficos.

9.2 Flujo recomendado de ejecución

Para reproducir el pipeline completo proponemos:

1. Ejecutar el bloque de **procesamiento** para generar los datos limpios, las variables derivadas y la asociación de accidentes a tramos.
2. Ejecutar el bloque de **análisis espacial** para obtener clusters, hot spots y medidas de autocorrelación.
3. Ejecutar el bloque de **modelado ML** para entrenar modelos, calcular probabilidades de gravedad y construir el índice de riesgo compuesto.
4. Ejecutar el bloque de **ruteo** para ver la demo de rutas (más corta, balanceada y más segura) y generar mapas interactivos.

Conclusiones y Trabajo Futuro

Conclusiones

Este proyecto nos permitió llevar el proceso KDD más allá de los ejemplos de laboratorio y aplicarlo a un problema real de seguridad vial. Vimos que:

- Cada etapa del KDD (selección, limpieza, transformación, minería e interpretación) impacta directamente en la calidad del índice de riesgo y de las rutas resultantes.
- La combinación de análisis espacial (DBSCAN, hot spots, Moran's I) y Machine Learning aporta una visión mucho más rica que sólo contar accidentes.
- Es posible construir rutas cuantitativamente más seguras con un costo moderado en distancia y tiempo, lo cual puede ser crítico en contextos como traslados hospitalarios.

Trabajo futuro

Entre las líneas de trabajo que identificamos están:

- Integrar información de tráfico en tiempo real para tener un índice de riesgo dinámico por fecha y hora.
- Enriquecer el modelo con variables adicionales (clima, tipo de vía más detallado, velocidades observadas, infraestructura ciclista, etc.).
- Modelar explícitamente la red de hospitales y simular escenarios de referencia entre distintos niveles de atención.
- Desplegar la aplicación web de ruteo seguro y validar su utilidad con usuarios reales y con personal de salud y protección civil.