Análisis Exploratorio para base de datos sobre biodiversidad Mexicana

*Note: Repositorio de GitHub

Santiago Flórez Suárez
Centro de Investigación en Matemáticas Aplicadas
Universidad Autónoma de Coahuila
Saltillo, Coahuila, México
santiagoflorez.s12@gmail.com

Andrés Iván Rodríguez Hernández
Centro de Investigación en Matemáticas Aplicadas
Universidad Autónoma de Coahuila
Saltillo, Coahuila, México
rodriguez_andres@uadec.edu.mx

***Resumen*—La disponibilidad de grandes volúmenes de información biológica a través de plataformas globales como GBIF ha impulsado el desarrollo de estudios ecológicos basados en datos abiertos. Sin embargo, la exploración inicial de estos conjuntos suele realizarse mediante reportes estáticos, lo que limita la interacción del usuario, dificulta la replicación del análisis y exige conocimientos técnicos avanzados. En este trabajo se presenta un sistema interactivo para la realización de Análisis Exploratorio de Datos (EDA) aplicado a registros biológicos georreferenciados de la familia Leporidae en México. La metodología incluye la depuración del dataset, la identificación y eliminación de variables redundantes o con valores faltantes críticos, y la estandarización de atributos espaciales, temporales y taxonómicos. Posteriormente, se implementa una plataforma desarrollada en Python mediante Streamlit, complementada con Pandas, PyDeck y librerías de visualización para permitir la inspección dinámica del comportamiento espacial y temporal de las especies.**

***Palabras clave*—Análisis Exploratorio de Datos, Biodiversidad, GBIF, Visualización Geoespacial, Python, Datos Abiertos.**

I. Introducción

En los últimos años, el estudio de la biodiversidad en México ha experimentado un crecimiento significativo gracias a plataformas globales como GBIF, las cuales permiten acceder a grandes volúmenes de datos biológicos provenientes de instituciones académicas, gubernamentales y comunitarias. Estos datos constituyen una base fundamental para caracterizar la distribución espacial de las especies, identificar tendencias temporales y apoyar decisiones de conservación.

No obstante, el análisis inicial de esta información suele presentarse mediante reportes estáticos elaborados con herramientas tradicionales de programación o visualización. Si bien este enfoque ha permitido generar productos de difusión científica, dicho formato presenta limitaciones importantes: las visualizaciones no son interactivas, la exploración depende de scripts personalizados y los usuarios deben poseer conocimientos técnicos para replicar o modificar el análisis. Esto dificulta que analistas, biólogos y tomadores de decisiones puedan interactuar de manera flexible con los datos.

En respuesta a esta necesidad, el presente trabajo desarrolla un sistema interactivo de Análisis Exploratorio de Datos (EDA) enfocado en registros biológicos georreferenciados. Mediante Python, Pandas, PyDeck y Streamlit, se implementa una plataforma que automatiza la limpieza de datos, facilita la inspección espacial y temporal, y ofrece al usuario un entorno visual y accesible para la exploración dinámica del comportamiento de especies dentro del territorio mexicano. Esta herramienta busca reducir las barreras técnicas, acelerar el proceso de análisis y mejorar la calidad de la interpretación ecológica mediante visualizaciones responsivas y filtros intuitivos.

II. Descripción del problema

El análisis de biodiversidad en México depende en gran medida de reportes técnicos generados a partir de descargas masivas de datos provenientes de plataformas como GBIF. Un ejemplo típico de estos reportes es el desarrollado por distintas iniciativas académicas y comunitarias, como RedBioma, donde se presentan resultados preprocesados como el siguiente ejemplo: https://redbioma.org/proyectos/2024-04-python-ciencia-datos/diversidad-abejas-familia-halictidade-mexico.html, gráficos estáticos y textos descriptivos sobre grupos taxonómicos específicos. Si bien estos documentos cumplen su propósito informativo, también presentan varias limitaciones prácticas:

- El flujo de análisis suele ser rígido: las figuras y tablas son estáticas, lo que impide explorar libremente los datos.
- Los reportes no permiten aplicar filtros dinámicos por especie, año o región geográfica; el usuario debe generar manualmente nuevos scripts para ello.
- La replicación del análisis requiere conocimientos intermedios o avanzados de Python, R o herramientas de SIG.
- La navegación entre múltiples gráficos o tablas puede ser poco intuitiva, dificultando el trabajo del analista o biólogo que necesita visualizar patrones específicos.

Estas limitaciones hacen que el proceso de exploración inicial sea más lento, menos flexible y dependiente de personal técnico con habilidades de programación. Además, ante datasets cada vez más grandes y variables de mayor complejidad, los reportes estáticos tradicionales dejan de ser suficientes para apoyar procesos de decisión o investigación ecológica.

Así, se ve como una necesidad una herramienta interactiva que automatice la depuración, facilite la inspección espacial

y temporal, y permita a especialistas (analistas, biólogos o técnicos ambientales) explorar los datos sin requerir programación. El objetivo es transformar un flujo de trabajo tradicionalmente estático en un entorno dinámico, accesible y reproducible, capaz de ofrecer una comprensión más profunda de la biodiversidad.

III. Herramientas utilizadas

- **GBIF** es una infraestructura internacional de datos sobre biodiversidad que proporciona acceso libre y gratuito a observaciones de especies en todo el mundo. A través de su plataforma (https://www.gbif.org), GBIF integra datos de diferentes fuentes como colecciones históricas, observaciones ciudadanas o secuencias genéticas estandarizados mediante estándares como Darwin Core. Este recurso facilita la descarga de grandes volúmenes de datos de ocurrencia biológica e información espacial, lo cual es fundamental para realizar estudios ecológicos, análisis de distribución y conservación.
- **Streamlit** es un framework de Python diseñado para crear aplicaciones web interactivas orientadas a análisis de datos. Permite construir paneles, filtros y visualizaciones dinámicas sin necesidad de desarrollar interfaces complejas.
- **Pandas** es una librería fundamental para la manipulación y análisis de datos estructurados. Facilita tareas de limpieza, filtrado, transformación..
- **Matplotlib** es una biblioteca base para la generación de gráficos en Python. Proporciona control detallado sobre figuras estáticas como histogramas, líneas, dispersión y diagramas de barras.
- **Seaborn** es la extensión estadística de Matplotlib que permite crear visualizaciones más estilizadas y con resúmenes estadísticos integrados. Es útil para graficar distribuciones, relaciones entre variables y patrones en series de datos.
- **PyDeck** es una interfaz de Python para Deck.gl, especializada en visualización geoespacial de alto rendimiento. Permite representar datos sobre mapas interactivos usando WebGL, siendo ideal para explorar patrones espaciales a nivel nacional o regional.

IV. Solución

La solución desarrollada combina un proceso de limpieza de datos con la construcción de una plataforma interactiva que facilita el análisis visual y exploratorio de la biodiversidad en México.

IV-A. Revisión de la base de datos

El conjunto de datos utilizado se obtuvo a través de la plataforma GBIF (Global Biodiversity Information Facility), una infraestructura internacional que centraliza registros biológicos provenientes de instituciones académicas, organizaciones gubernamentales y proyectos de ciencia ciudadana. A través de su portal web, GBIF permite realizar solicitudes personalizadas de datos en múltiples formatos.

Para este estudio se descargaron todas las observaciones de la familia **Leporidae** con presencia confirmada en territorio mexicano. Empleando la siguiente consulta:

```
{
 "and" : [
   "Country is Mexico",
   "OccurrenceStatus is Present",
   "TaxonKey is Leporidae"
 ]
}
```

La descarga original contiene 23,700 registros y 50 columnas. En los datos provistos por GBIF se incluye la estructura taxonómica completa del grupo solicitado. Dicha estructura permite identificar cómo se organiza jerárquicamente la familia *Leporidae* dentro del árbol de la vida y cuántas observaciones se registran por nivel.

```
Reino
 Animalia.............................. 890,355
   Filo
    Chordata............................ 26,432,409
      Clase
       Mammalia......................... 890,355
         Orden
          Lagomorpha...................... 23,786
            Familia
             Leporidae.................. 23,672
               Géneros
                Sylvilagus.............. 14,803
                Lepus................... 8,136
                Romerolagus............. 451
                Oryctolagus............. 61
                Hypolagus............... 50
                Notolagus............... 16
                Aztlanolagus............ 10
                Paranotolagus........... 9
                Pewelagus............... 6
                Aluralagus.............. 4
                Pratilepus.............. 4
                Pronotolagus............ 2
                Unknown genus........... 148
                 Unknown family.......... 86
```

Figura 1: Estructura taxonómica del grupo *Leporidae* según GBIF, con número de observaciones por nivel.

El dataset original incluía 50 columnas, muchas de las cuales presentaban altos porcentajes de valores faltantes o contenían información sin utilidad para un análisis ecológico. Inicialmente para usar variables con veracidad se calculó el porcentaje de valores nulos en cada columna, identificando 14 variables con un porcentaje igual o superior al 40 %. Estas fueron eliminadas para evitar sesgos y mantener la integridad del análisis.

La Tabla 1 resume los porcentajes observados.

Tras eliminar las columnas con alto porcentaje de valores faltantes, aún permanecían 36 atributos. Sin embargo, 28 de ellos resultaron irrelevantes para el análisis y fueron eliminados siguiendo criterios de redundancia, falta de variabilidad o ausencia de valor analítico. La Tabla 3 presenta cada columna descartada junto con lla justificación correspondiente. Adicionalmente La Tabla 2 sintetiza las principales justificaciones para una lectura más simple.

Entrando más a detalle se puede apreciar por variable cual fue el criterio escogido en cada situación:

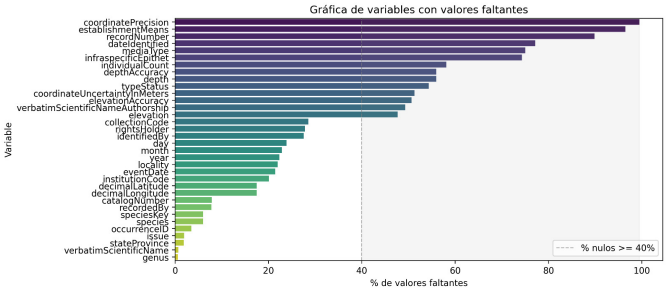

Figura 2: Porcentaje de valores faltantes por columna

Tabla 1: Porcentaje de valores faltantes por columna

Variable	% Faltante
coordinatePrecision	99.52
establishmentMeans	96.46
recordNumber	89.87
dateIdentified	77.15
mediaType	75.05
infraspecificEpithet	74.31
individualCount	58.08
depthAccuracy	55.95
(entre otras)	

Tabla 2: Justificación para eliminación de columnas

Columna	Justificación
gbifID	No garantiza unicidad estable
genus	Ya está contenido en verbatimScientificName
collectionCode	Metadato irrelevante para análisis espacial
speciesKey	Identificador administrativo no útil
(entre otras)	

Columna	Razón de eliminación
gbifID	Identificador técnico, no aporta información ecológica
datasetKey	Metadato administrativo de GBIF
occurrenceID	Identificador redundante, no útil para análisis
scientificName	Duplicado de verbatimScientificName
taxonRank	Constante para todo el dataset
taxonKey	Clave técnica sin utilidad analítica
species	Derivable del nombre científico, redundante
speciesKey	Identificador administrativo
genus	Información ya contenida en verbatimScientificName
genusKey	Clave administrativa sin uso analítico
family	Constante (Leporidae) para todos los registros
familyKey	Clave redundante con family
order	Constante (Lagomorpha) para todo el conjunto
orderKey	Identificador técnico, irrelevante
class	Constante (Mammalia), sin variabilidad
classKey	Clave administrativa
phylum	Constante (Chordata), sin valor analítico
phylumKey	Clave administrativa
kingdom	Constante (Animalia), no aporta variabilidad
kingdomKey	Clave administrativa
publishingOrgKey	Metadato sobre la institución publicadora
license	No aporta información ecológica relevante
institutionCode	Identificación administrativa de la institución
references	Enlaces externos no usados en análisis
catalogNumber	Información museográfica irrelevante
recordedBy	Nombre del colector, no relevante para análisis espacial–temporal
identifiedBy	No aporta valor al análisis ecológico
lastInterpreted	Metadato técnico sin uso analítico
issue	Metadatos de procesamiento de GBIF

Tabla 3: Columnas eliminadas y justificación de exclusión.

Después de este proceso, se conservaron ocho columnas, las cuales serán de donde se extraerá toda la información para el análisis posterior, las cuales son: `verbatimScientificName`, `locality`, `stateProvince`, `decimalLatitude`, `decimalLongitude`, `eventDate`, `basisOfRecord` e `institutionCode`.

El dataset que fue proporcionado tiene su diccionario de datos en la página https://techdocs.gbif.org/en/data-use/download-formats en el cual se observa todas las variables originales, de las cuales para nuestro estudio escogeremos las 8 que nos son relevantes y cuyo diccionario de datos se anexa a continuación.

Tabla 4: Variables seleccionadas del diccionario de datos de GBIF

Columna	Tipo	Descripción
verbatimScientificName	String	Nombre científico registrado
locality	String	Descripción específica del sitio
stateProvince	String	Entidad federativa
decimalLatitude	Double	Latitud en grados decimales
decimalLongitude	Double	Longitud en grados decimales
eventDate	String	Fecha del registro
basisOfRecord	String	Tipo de evidencia
institutionCode	String	Institución que reporta

Inicialmente el dataset contenía 23,672 registros. Para garantizar un análisis espacio–temporal completo, se eliminó cualquier fila con valores faltantes en:

- `eventDate`,
- `decimalLatitude`,
- `decimalLongitude`.

Esto redujo el dataset a 15,677 registros.

La Tabla 5 muestra el comportamiento de los valores faltantes antes de esta limpieza.

Tabla 5: Valores faltantes antes de limpieza de filas

Columna	Nulos	%
locality	5,201	21.97
eventDate	5,088	21.49
decimalLatitude	4,135	17.47
stateProvince	446	1.88
verbatimScientificName	172	0.73

Se realizaron tres pasos adicionales:

1. Eliminación de registros sin `stateProvince` (143 registros).
2. Eliminación de valores no válidos de `verbatimScientificName`, incluyendo 163 entradas del sistema BOLD (Barcode of Life Data System).
3. Corrección de nombres científicos en mayúsculas mediante estandarización de formato.
4. Verificación de registros sobre longitud y latitud.
5. Corrección de nombre de los estados para homogeneidad de nombres.
6. Estandarización del formato de fecha.

Tras la conclusión de la fase de preparación de datos, el dataset final se estableció con un total de 14,171 observaciones. Estos registros han sido depurados y validados, quedando en

un estado completamente limpio y listos para la exploración analítica.

Este flujo de trabajo estructurado fue clave, ya que permitió comprender la base de datos proveniente de GBIF de manera eficaz, eliminando inconsistencias y duplicados. Alcanzando las condiciones de la calidad de datos requerida para su integración directa en la plataforma interactiva desarrollada en Streamlit.

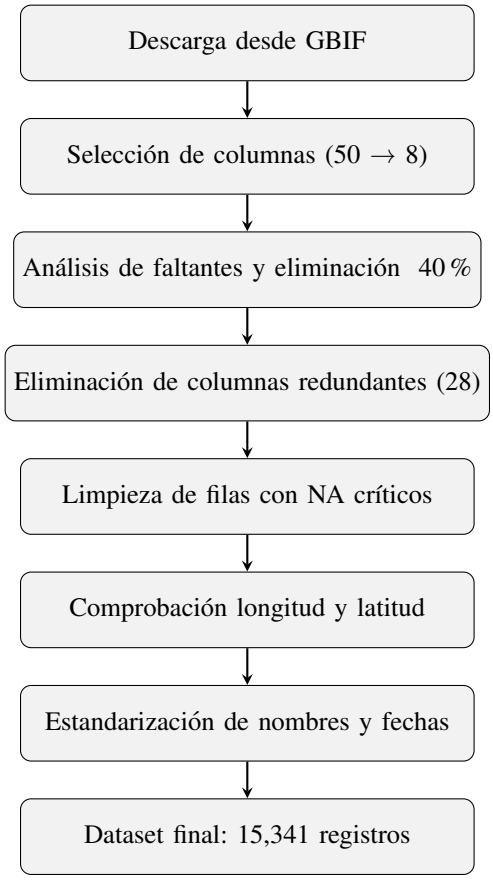

Figura 3: Esquema general del proceso de limpieza de datos.

IV-B. Implementación en Streamlit

Con el fin de facilitar la exploración visual y la manipulación dinámica del conjunto de datos proporcionado por GBIF, se desarrolló una aplicación interactiva utilizando la biblioteca *Streamlit*. Esta herramienta permite transformar scripts de Python en interfaces gráficas funcionales sin requerir lenguajes adicionales como HTML, CSS o JavaScript. A continuación, se describe detalladamente cada etapa del sistema implementado, desde la carga de datos hasta la visualización espacial y temporal de los registros biológicos.

IV-B1. Configuración inicial y carga del conjunto de datos: La aplicación inicia configurando un diseño de página ancho, asignando un título general y mostrando una breve descripción del propósito del análisis. Posteriormente, se implementó una función de carga utilizando `@st.cache_data`, con la finalidad de evitar la recarga repetida del archivo en cada interacción del usuario, optimizando así el rendimiento del sistema.

La función de carga realiza las siguientes acciones:

- Verifica la existencia del archivo `base.csv`.
- Importa el dataset y elimina filas sin información esencial (coordenadas o fecha).
- Normaliza las columnas renombrando `decimalLatitude` y `decimalLongitude` como `lat` y `lon`.
- Convierte la columna `eventDate` al tipo fecha e incorpora dos nuevas columnas:
  - `year`: año del registro,
  - `YearMonth`: periodo año-mes.
- Calcula el número de observaciones por especie y lo integra al DataFrame para facilitar su ordenamiento.
- En caso de no existir la columna `stateProvince`, se añade automáticamente para asegurar la compatibilidad con los gráficos posteriores.

Este procedimiento genera un DataFrame limpio y estructurado que es utilizado a lo largo de toda la plataforma.

IV-B2. Asignación de colores para visualización geográfica: Para representar cada especie de manera diferenciada en los mapas interactivos, se creó un diccionario de colores. Cada especie recibe un color único en formato RGBA, el cual se aplica posteriormente en las capas generadas con PyDeck. En caso de que alguna especie no cuente con color asignado (por errores en la lectura del nombre), se utiliza un gris por defecto.

IV-B3. Gestión del estado de la sesión y acciones rápidas: Dado que el usuario puede seleccionar o deseleccionar múltiples especies, fue necesario implementar un mecanismo basado en `st.session_state`. Esto permite conservar las selecciones aun cuando la interfaz se actualiza. Se incluyen funciones dedicadas para:

- Seleccionar todas las especies,
- Deseleccionar todas las especies,
- Actualizar continuamente la lista de especies activas.

Este comportamiento es fundamental para asegurar la consistencia de los filtros en la interfaz.

IV-B4. Personalización de la interfaz mediante CSS: Aunque Streamlit proporciona objetos visuales predeterminados, se incorporó código CSS con el fin de mejorar la experiencia del usuario. Entre las modificaciones realizadas destacan:

- Estilización de botones para su uso en la barra lateral,
- Mejora visual de los *toggles* que permiten activar o desactivar especies,
- Ocultación de ciertos elementos nativos para una apariencia más limpia.

La personalización contribuye a una navegación más intuitiva.

IV-B5. Filtros interactivos en la barra lateral: La barra lateral contiene tres componentes centrales:

1. **Botones de selección global**: permiten seleccionar todas las especies o deseleccionarlas.

2. **Listado ordenado de especies**: cada especie aparece acompañada de su frecuencia de registro, presentada como un interruptor independiente.
3. **Control temporal**: mediante un deslizante se establece un rango temporal basado en el periodo `YearMonth`.

La combinación de estos elementos proporciona una capacidad de filtrado amplia y flexible, permitiendo al analista trabajar únicamente con el subconjunto de datos relevante.

IV-B6. Filtrado dinámico del conjunto de datos: Una vez definidos los filtros, se genera un subconjunto `df_final`, el cual incorpora únicamente:

- Las especies seleccionadas por el usuario,
- Los registros cuya fecha pertenece al rango temporal activo.

Si el resultado es un DataFrame vacío, la aplicación notifica al usuario para que ajuste los filtros antes de mostrar cualquier visualización.

IV-B7. Visualización geográfica mediante PyDeck: La pieza central de la plataforma es un mapa interactivo generado con PyDeck. Cada punto del mapa representa una observación individual y se caracteriza por:

- Su ubicación (`lat`, `lon`),
- El color correspondiente a su especie,
- Un radio ajustable en metros,
- Un *tooltip* que muestra especie y año del registro.

La figura 4 muestra un ejemplo del mapa interactivo utilizado en la plataforma.

Figura 4: Mapa interactivo, mostrando avistamientos filtrados.

IV-B8. Consulta tabular de datos filtrados: Debajo del mapa se incluye un panel desplegable (*expander*) que permite visualizar el subconjunto filtrado de datos en formato tabular. Esta tabla contiene únicamente los campos relevantes para el análisis: nombre científico, fecha, latitud y longitud.

IV-B9. Gráficos descriptivos: especies y provincias: Además de la visualización espacial, la plataforma genera automáticamente dos gráficos de barras:

1. **Top 10 especies más registradas**,
2. **Top 10 provincias con mayor número de observaciones**.

En ambos casos, se utilizan gráficos horizontales que incluyen etiquetas numéricas y paletas de color optimizadas para mejorar la legibilidad. Estas visualizaciones permiten al usuario identificar rápidamente patrones asociados al esfuerzo de muestreo o prevalencia biológica.

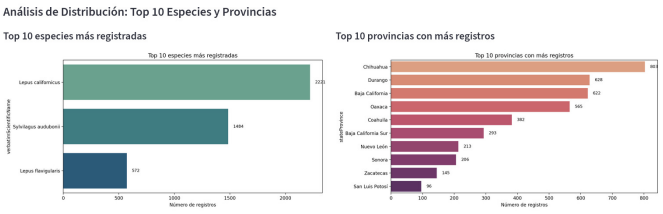

Figura 5: Gráficas de barras interactivas, mostrando avistamientos filtrados.

IV-B10. Tendencia temporal mediante histogramas apilados: Para estudiar la evolución cronológica de los registros, se construye un histograma apilado en el que:

- Se consideran las cinco especies más abundantes en el conjunto filtrado,
- Las especies restantes se agrupan bajo la categoría *Otras Especies*.

Esta representación permite observar variaciones temporales, posibles picos de muestreo y cambios en las especies dominantes a lo largo del tiempo.

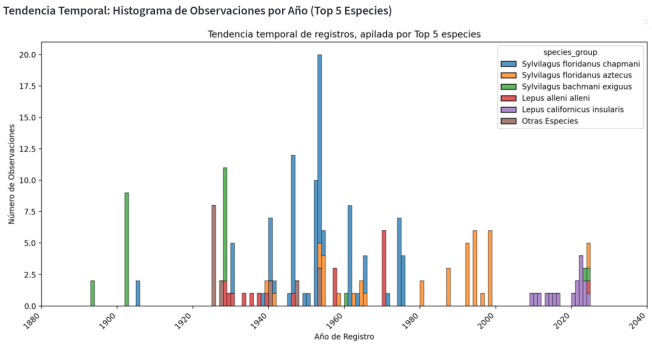

Figura 6: Histograma interactivo, mostrando avistamientos filtrados.

La integración de todas estas funcionalidades dentro de una sola interfaz permite:

- explorar datos espaciales y temporales sin programación,
- aplicar filtros combinados de forma inmediata,
- visualizar patrones complejos mediante gráficos descriptivos,
- identificar inconsistencias en los datos,
- apoyar decisiones y análisis exploratorios para estudios ecológicos.

Gracias a esto, la aplicación proporciona un entorno flexible para analizar datos biológicos sin necesidad de escribir código, reduciendo la carga técnica y permitiendo que especialistas en biodiversidad puedan concentrarse en la interpretación ecológica.

V. Hallazgos

El proceso de depuración permitió transformar un conjunto de datos inicialmente heterogéneo y con alta presencia de valores faltantes en una base robusta y apta para análisis geoespacial y temporal. De los 23,672 registros originales, únicamente el 59.8 % resultó utilizable tras aplicar criterios estrictos de completitud en variables críticas (`eventDate`, `decimalLatitude`, `decimalLongitude`). Esto evidencia que, aun tratándose de un repositorio estandarizado como GBIF, es frecuente encontrar información incompleta, especialmente en los datos históricos o provenientes de la ciudadanía.

La eliminación de columnas se justificó por tres patrones recurrentes:

1. **Redundancia taxonómica**: variables como `genus`, `family` o `order` no aportaron valor debido a su invariancia o duplicidad respecto a `verbatimScientificName`.
2. **Metadatos administrativos**: identificadores técnicos (e.g., `taxonKey`, `occurrenceID`) no contribuyen al análisis ecológico o espacial.
3. **Altos niveles de datos faltantes**: 14 columnas superaban el 40 % de ausencia, lo que limita su utilidad y aumenta potenciales sesgos.

La estructura taxonómica reveló una fuerte concentración de registros en dos géneros principales: *Sylvilagus* y *Lepus*, que en conjunto representan más del 95 % de las observaciones. Esto sugiere una mayor actividad de muestreo, distribución geográfica más amplia o mayor detectabilidad de estas especies, mientras que géneros como *Pewelagus*, *Aztlanolagus* o *Romerolagus* están subrepresentados y podrían requerir una revisión más profunda en estudios de conservación.

En cuanto a la calidad geoespacial, el análisis identificó múltiples registros con coordenadas inválidas o ubicaciones fuera del rango nacional, los cuales fueron descartados. La estandarización de nombres de entidades federativas y la corrección de formatos temporales permitieron homogenizar la base para los módulos interactivos.

La aplicación en Streamlit mostró patrones relevantes:

- La distribución espacial evidencia concentraciones de observaciones en estados con alta actividad de monitoreo.
- Existen años con picos de registro asociados a proyectos específicos, indicando una fuerte dependencia del esfuerzo de muestreo.
- La opción de filtrado por especie, rango temporal y entidad federativa permite observar diferencias claras en presencia y abundancia relativa entre géneros.

Se observo que una parte sustancial de los desafíos en estudios de biodiversidad no proviene del análisis estadístico en sí, sino de la depuración y estandarización previa de los datos, así como de la falta de herramientas que permitan una exploración dinámica y reproducible.

VI. Conclusiones

La integración de datos abiertos de biodiversidad con plataformas interactivas puede transformar significativamente la manera en que se realiza el análisis exploratorio en contextos ecológicos. El dataset descargado desde GBIF, si bien amplio y detallado, presentó problemas comunes en datos biológicos: valores faltantes, redundancias taxonómicas y metadatos inconsistentes. Mediante un proceso de depuración sistemático se logró obtener un conjunto final confiable de 14,171 observaciones, apto para análisis espacial y temporal.

La plataforma desarrollada en Streamlit permitió solventar una limitación recurrente de los reportes tradicionales: la falta de interactividad. Los usuarios pueden ahora explorar patrones espaciales, temporalidades, diferencias taxonómicas y calidad de datos sin necesidad de conocimientos avanzados en programación. Esta capacidad de visualización dinámica mejora la comprensión de fenómenos ecológicos y reduce el tiempo necesario para generar hipótesis o identificar problemas en el dataset.

Los resultados también evidencian que la calidad de los registros es un factor determinante para estudios ecológicos. Un porcentaje considerable de información queda excluido por problemas de completitud o precisión, lo cual subraya la necesidad de fortalecer las prácticas de captura, estandarización y validación en plataformas como GBIF.

Finalmente, el sistema presentado no solo constituye una herramienta funcional de EDA, sino también una propuesta extensible. Su arquitectura puede adaptarse fácilmente a otros taxones, regiones o conjuntos de datos ambientales, convirtiéndolo en una base sólida para futuros desarrollos orientados a conservación, modelación de nichos ecológicos o análisis de riesgo biológico.

Referencias

[1] GBIF.org, "Gbif occurrence download," Nov. 2025, accessed on 5 November 2025. [Online]. Available: https://doi.org/10.15468/dl.752wz3

[2] Pandas Development Team, "Pandas documentation," https://pandas.pydata.org, 2024, accessed: 2025-11-05.

[3] NumPy Developers, "Numpy: The fundamental package for scientific computing," https://numpy.org, 2024, accessed: 2025-11-05.

[4] Matplotlib Development Team, "Matplotlib: Visualization with python," https://matplotlib.org, 2024, accessed: 2025-11-05.

[5] Seaborn Developers, "Seaborn statistical data visualization," https://seaborn.pydata.org, 2024, accessed: 2025-11-05.

[6] PyDeck Developers, "Pydeck: Python wrapper for deck.gl," https://pydeck.gl, 2024, accessed: 2025-11-05.

[7] Uber Technologies Inc., "deck.gl: Webgl-powered geospatial visualizations," https://deck.gl, 2024, accessed: 2025-11-05.

[8] Streamlit Inc., "Streamlit: The fastest way to build data apps," https://streamlit.io, 2024, accessed: 2025-11-05.

[9] Python Software Foundation, "Python programming language – official documentation," https://www.python.org, 2024, accessed: 2025-11-05.