Aplicativo Web en Python sobre el análisis estadístico de Siniestralidad Vial En La Ciudad De Bogotá

Juan Sebastian Mancera Gaitán 20171020047
Jeison Jara Sastoque 20162020461
Facultad de Ingeniería
Universidad Distrital Francisco José de Caldas
TeleInformática I
22/02/2021

I. INTRODUCCIÓN:

No es para nada un secreto que en las grandes capitales los índices de accidentalidad son bastante frecuentes, según los datos del Centro de Gestión de Tránsito - CGT y de la Secretaría de Movilidad ocurre un siniestro vial cada 5,6 minutos, en cifras tomadas entre Enero y Septiembre de 2019 se presentaron 726 homicidios y 393 muertes por accidentes de tránsito.

Usualmente no somos conscientes de las cifras que estos datos implican, tampoco conocemos cuales son las causas de estos accidentes, los lugares en los que ocurren o los involucrados que participan en estos incidentes, por lo que seria de gran importancia para los ciudadanos conocer estas estadísticas y tener fácil acceso a ellas para de cierto modo mitigar los riesgos conociendo en qué lugares tener más cuidado, que acciones evitar para correr menos riesgos o con qué vehículos hay que tener la mayor precaución.

Es por ello que planteamos una solución por medio de un aplicativos que recolecta los datos suministrados por los entes que controlan este tipo de incidentes y que muestre de manera fácil y acertada la información que un ciudadano necesite, y para ello se hará uso de de algunas de las temáticas tratadas en la clase de Teleinformática

I.OBJETIVOS

Objetivo General:

1. Desarrollar una aplicación web la cual mediante los datos encontrados en la página de Datos Abiertos de Bogotá permita identificar las causas, involucrados y los siniestros ocurridos en las calles de la ciudad, mostrando de manera clara y objetiva los resultados que un ciudadano quiere obtener.

Objetivos Específicos:

1. Realizar la limpieza de datos respectiva para el dataset de Siniestros Viales Consolidados entre 2015 a 2019, mediante la librería para la ciencia de datos Pandas.
2. Llevar a cabo un estudio de los datos obtenidos a través de diferentes métodos y procedimientos estadísticos pertenecientes a la librería Numpy y Scipy.
3. Realizar distintos tipos de gráficos estadísticos para mostrar al usuario de forma entendible y concreta los resultados de su consulta aplicando la librería Matplotlib y Seaborn para este procedimiento..
4. Desarrollar aplicación Web mediante el framework Django o Flask perteneciente al lenguaje de programación Python, para visualizar los datos de forma interactiva.

II. T*RABAJOS RELACIONADOS*

*A. Articulo **Traffic Accidents Analyzer Using Big Data** [1]*

El artículo se centra en la problemática de la siniestralidad vial, es por ello que para entender del todo en cuanto a un un problema tan amplio se debe conocer el entorno del mismo, para esto se cuentan con millones de datasets con información relevante a los accidentes de tráfico en distintos países y ciudades, por lo cual se posee la facilidad de almacenar, manipular y analizar grandes cantidades

de datos que ayuden a la toma de decisiones referentes a evitar la siniestralidad vial.

Para ello en el artículo se muestra la realización de una aplicación capaz de analizar grandes cantidades de datos (Big Data) mediante técnicas de minería de datos, y a partir de diferentes resultados estadísticos obtenidos mostrar de una forma intuitiva al usuario los resultados encontrados de acuerdo a la información que él mismo desee consultar acompañado de la visualización de datos (gráficos) a partir de tecnologías como Hadoop, Python, Matplotlib y Numpy.

B. *Articulo* ***Safe Driving: A Mobile Application for Detecting Traffic Accidents [2]***

Se desarrolla una aplicación para alertar sobre posibles accidentes de tráfico, esto mediante la monitorización de datos como la velocidad y su variación, ondas acústicas y ondas de vibración, la aplicación detecta estos datos y activa una alarma cuando considera que es posible un choque.

El paper indica cómo la aplicación debe interactuar con el usuario además que implementa distintas funcionalidades útiles a partir de los datos como, historial de accidentes del usuario, historial de accidentes de un sitio, localización de accidentes en tiempo real y su ubicación mediante mapas.

C. Articulo ***A review on road accident data analysis using data mining techniques*** [3]

En el artículo se utiliza la minería de datos como base para la predicción de accidentes de tráfico, para ello se utilizan técnicas de clasificación, reglas de asociación , SMV, K-mean en las variables más significativas en el campo de estudio.

Para garantizar la validez de los datos anteriores se debe primero corroborar el correcto tratamiento y uso de los datos, para ello se realizan distintos datasets a partir de la fuente original de datos que permitan convertir información resumida en información útil.

D. Articulo ***On the Analysis of Work Accidents Data by Using Data Preprocessing and Statistical Techniques[***4]

El artículo plantea que es necesario el escenario de preprocesamiento de datos antes de realizar aplicaciones de machine learning, ya que datos perdidos, molestos e inconsistentes en las variables cambiarán los posibles resultados de la investigación.

El artículo realiza distintos procedimientos estadísticos de acuerdo a los datos como por ejemplo promedio de accidentes de diarios, correlaciones y covarianzas para encontrar variables significativas, historial de accidentes basado en histogramas y en líneas de tiempo, como estudio a futuro el artículo concluye que con la correcta visualización de datos en mayor medida es más fácil encontrar el camino correcto hacia la predicción de variables y es aquí donde se evidencia la importancia del preprocesamiento de datos.

E. Articulo ***RTAIS: Road Traffic Accident Information System*** [5]

El artículo muestra el proceso de creación de un sistema de información enfocado a los datos de siniestralidad vial, al igual que las funcionalidades de implementar más viables y la forma de llevarlas a cabo, lo que permite reducir costos, aumentar la eficiencia, digitalizar los recursos y mejorar la precisión del análisis.

III. HERRAMIENTAS:

Para la realización del análisis estadístico,se maneja el lenguaje de programación Python en su versión 3.9, siendo esta la más reciente hasta la fecha, adicionalmente se utilizarán las siguientes librerías presentes en el lenguaje.

- Pandas:

  Autor: Wes McKinney

  Descripción: Pandas es una biblioteca de software escrita como extensión de NumPy para manipulación y análisis de datos para el lenguaje de programación Python. En particular, ofrece estructuras de datos y operaciones para manipular tablas numéricas y series temporales.

  Fecha de Creación: 2008

- Numpy:

  Autor: Travis Oliphant

  Descripción: Es una biblioteca para el lenguaje de programación Python que da soporte para crear vectores y matrices grandes multidimensionales, junto con una gran colección de funciones matemáticas de alto nivel para operar con ellas.

  Fecha de Creación: 1995

- Geopandas:

  Autor: Joris Van den Bossche

  Descripción: Es un proyecto de código abierto para facilitar el trabajo con datos geoespaciales en python. GeoPandas amplía los tipos de datos utilizados por pandas para permitir operaciones espaciales en tipos geométricos.

  Fecha de Creación: 2013

- Matplotlib:

  Autor: John D. Hunter

  Descripción: Es una biblioteca para la generación de gráficos a partir de datos contenidos en listas o arrays en el lenguaje de programación Python y su extensión matemática NumPy.

  Fecha de Creación: 2003

- Flask:

  Autor: Armin Ronacher

  Descripción: Es un framework minimalista escrito en Python que permite crear aplicaciones web rápidamente y con un mínimo número de líneas de código. Está basado en la especificación WSGI de Werkzeug y el motor de templates Jinja2

  Fecha de Creación: 2010

- Seaborn:

  Autor: Michael Waskom

  Descripción: Es una biblioteca de visualización de datos de Python basada en matplotlib. Proporciona una interfaz de alto nivel para dibujar gráficos estadísticos atractivos e informativos.

  Fecha de Creación: 2012

- datetime:

  Autor: Victor Stinner

  Descripción: El módulo datetime incluye funciones y clases para hacer análisis, formateo y aritmética de fecha y hora.

  Fecha de Creación: 2008

IV. METODOLOGIA

El presente estudio se basa a partir de los datos y la información obtenida mediante procedimientos pertenecientes a la ciencia de datos, por lo cual el proyecto seguirá las etapas respectivas a la metodología para proyectos de ciencias de Datos,esta metodología es conocida como CRISP-DM (Cross Industry Standard Process for Data Mining), consistiendo en las siguientes fases:

1.Entendimiento de los datos
2.Preparación de los datos
3.Visualización de datos
4.Modelamiento de los datos
5.Evaluación de los datos
6.Despliegue

De acuerdo a la metodología mencionada, el apartado de Diseño seguirá esta fase excluyendo las #4 y #5 que corresponden a la realización de predicciones de variables significativas mediante modelos de machine learning en los datos almacenados, y la posterior evaluación del modelo predictivo en cuanto a la precisión del mismo, es por ello que se excluyen, sin embargo se espera ser abarcadas en estudios futuros.

V. DISEÑO

-Entendimiento de los datos:

En este caso se utilizará el conjunto de datos de la página de Datos Abiertos de Bogotá Siniestros Viales consolidados Bogotá D.C contando con un número mayor a 173.444 registros entre los años 2015 y 2019 clasificados por los siguientes factores:

-Siniestros: fecha, hora, gravedad,clase, choque, objeto fijo, dirección, total muertos, total heridos, localidad, diseño del lugar.
-Actor Vial: fecha, condición, estado, edad, sexo,vehículo.
-Vehículo: fecha, clase, servicio, fuga.
-Hipótesis: Código de causa, descripción, código de causa 2 y descripción 2

-Preparación de los datos:
Se crean las estructuras de datos llamadas dataframes de acuerdo a la hoja de cada excel, quedando cuatro tipos de dataframes mediante la librería Python, siendo estos accidentes_siniestros,accidentes_actor_vial, accidentes_vehiculos y accidentes hipótesis.
Una vez convertida la base de datos a diferentes dataframes se revisa el tipo de datos de las diferentes variables presentes, las variables como fechas y días de la semana son convertidas a formato datetime, variables numéricas a tipo entero, esto con la finalidad de facilitar la realización de operaciones con los datos.
Finalmente se eliminan los valores nulos, o que no cuenten con el registro completo de la información del accidente, se revisa la congruencia de los datos y se generan nuevas variables a partir de la variable fecha y hora como dia_semana y hora del accidente.

Visualización de datos:

Con la ayuda de la librería Matplotlib, se realizan mapas de calor, histogramas, polígonos de frecuencias entre otras gráficas para visualizar las variables cualitativas y cuantitativas de forma más entendible.
Se realizan gráficas de mapas. mediante la librería geopandas, esto mediante los datos del accidente y las coordenadas de geolocalización de las localidades de bogotá, este archivo de coordenadas fue tomado del repositorio de laboratorio urbano Bogotá.[]

Despliegue:

Se realiza una aplicación web, la cual almacena la información estadística, gráficas, análisis y el código utilizado para el estudio estadístico, esta aplicación web está implementada sobre la librería Flask, para el servidor web http, la interfaz gráfica está realizada mediante una plantilla de HTML Y CSS.

VI. IMPLEMENTACIÓN

Para analizar los datos y hallar las respectivas medidas estadísticas, se crean Jupyter notebooks, este formato permite crear celdas ejecutables en las cuales se puede evidenciar progresivamente el tratamiento de datos, y los resultados obtenidos por cada línea de código necesaria para la realización del estudio.

Imagen 1: Formato Jupyter Notebook

Como se ve en la Imagen 1, se crea una celda en la cual se ejecuta el código, en este caso el código realiza los histogramas de las localidades de la ciudad en comparación a la cantidad de accidentes presentes en cada una en la parte inferior de la celda.

Utilizando este formato se procede a calcular la cantidad de accidentes presentes en la base de datos, e información de interés, con las funciones existentes en las herramientas anteriormente mencionadas,jupyter notebook permite dejar el archivo ejecutado con los resultados de cada uno de las celdas, para que el usuario en caso de desear el código fuente no le sea necesario recurrir a la ejecución de este código.

Posteriormente se crea el servidor Flask para el aplicativo web ,este servidor posee las siguientes rutas las cuales podrán ser consultadas por el usuario, según su tema de interés.

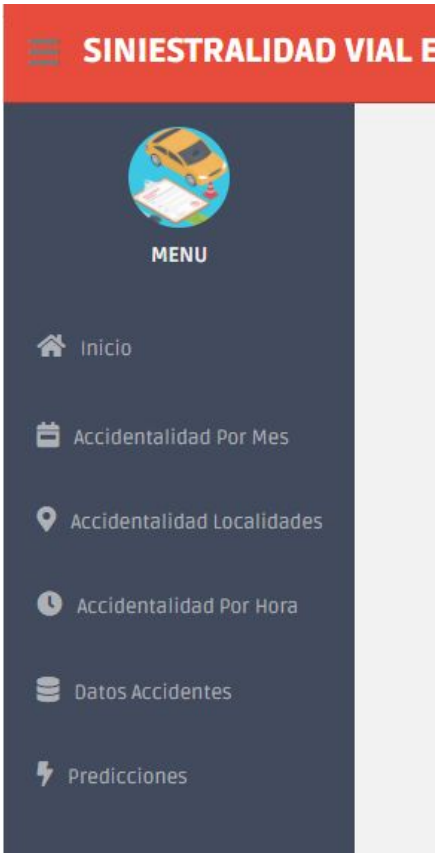


Imagen 2: Rutas de Información en el servidor Flask

En el caso de que el usuario, seleccione la opción de Accidentalidad por Mes, encontrará información relevante de acuerdo a este tema, como se puede observar en la imágen 3.

Imagen 3: Visualización Aplicativo Web.

El usuario encontrará un apartado completo,en la cual se describe las variables presentes en la opción elegida, además se acompaña esta información con gráficas útiles para la representación de la información, es por ello que el usuario tiene a su disposición un aplicativo web con los datos más relevantes de los accidentes de tráfico ocurridos en la ciudad de Bogotá entre los años 2015-2019 a partir de más de 100.000 registros de accidentes en la ciudad.

VII. RESULTADOS

Se presentan en el informe, algunos de los datos estadísticos de interés encontrados,la totalidad de estos datos se encuentran en el aplicativo web de forma gráfica y en el código fuente encontrado en los cuadernos de Jupyter.

- Factor Siniestros:

  La localidad con mayor cantidad de Accidentes es Kennedy con 19451, mientras que la menor localidad es Sumapaz seguida de Antonio Nariño.

	LOCALIDAD	Accidentes
0	Antonio Nariño	3431
1	Barrios Unidos	9462
2	Bosa	8274
3	Chapinero	10625
4	Ciudad Bolívar	6798
5	Engativá	17627
6	Fontibón	13883
7	Kennedy	19451
8	La Candelaria	901
9	Los Mártires	5902
10	Puente Aranda	12083
11	Rafael Uribe Uribe	4389
12	San Cristóbal	4674
13	Santa Fe	5043
14	Suba	15696
15	Sumapaz	5
16	Teusaquillo	9359
17	Tunjuelito	4701

Imagen 4: Accidentes por Localidad 2015-2019

Se obtiene el mapa de calor para visualizar en el mapa de la ciudad estos datos.

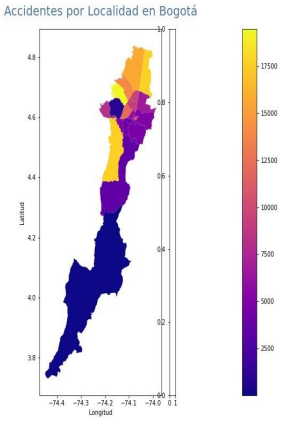
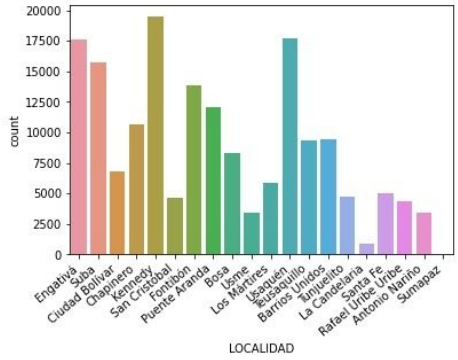
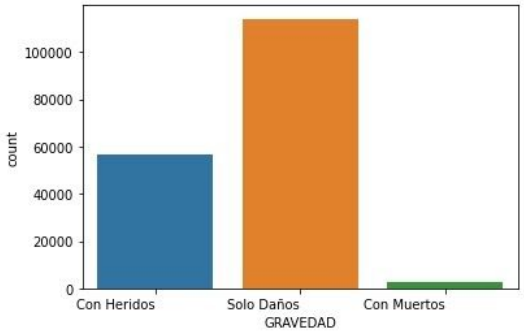


Imagen 5: Accidentes por Localidad 2015-2019

En caso de querer obtener una mejor visualización se realiza un histograma con los datos de la tabla de la imágen 4.



Imágen 6: Histograma Accidentes por Localidad



Imágen 7: Histograma Gravedad del Accidente

En la imágen 7 Se observa el histograma de la gravedad de accidentes de tránsito, como se puede evidenciar se posee tres resultados de la variable cualitativa Gravedad como lo son:

- Con heridos: Aprox 50000 Registros
- Solo Daños: Aprox 100000 Registros
- Con Muertos: Aprox 1000 Registros

Debido a la cantidad de información, los demas resultados estadísticos estarán disponibles en el aplicativo Web realizado en Python el cual es el objetivo del informe.

VIII. DISCUSIONES

El tiempo de ejecución del código referente a la conversión de los datos .xlsx (formato de excel) a dataframe (formato pandas) era bastante considerado al ser ejecutado en una máquina local, debido a que se estaba realizando la ejecución de operaciones sobre 173.444 registros, el tiempo promedio de la ejecución de cada línea de código en la cual se operara con el conjunto total de datos era de 2:30 a 4 minutos, por lo cual al realizar la comparación con otros artículos sobre data análisis se llega a la conclusión de que se debe optimizar este tiempo para mejor el tiempo de respuesta de cada una de las operaciones, para ello las alternativas viables son el almacenamiento de los datos en servicios en la nube, como los ofrecidos por Amazon Web Services e IBM Cloud, ya que esto permite hacer uso de la capacidad de procesamiento disponible por estas empresas, lo que mejoraría sustancialmente la ejecución de operaciones con este gran volumen de datos.

Por otra parte, en casos donde se presentan los datos agrupados según categorías como es nuestro caso en la cual se presentaban cuatro factores,se recomienda que estos sean agrupados en menos categorías, de ser posibles en una sola,para ello se deben realizar correlaciones y covarianzas que permitan indicar si variables entre sí poseen cierta correlación útil para ser analizadas en el estudio, aquellas variables que presentan poca o nula correlación tanto con las demás variables como con la variable objetivo, en este caso los accidentes.deben ser eliminadas del conjunto de datos.

Lo anterior se realiza con el fin de obtener los datos mejor organizados, evitar la saturación de datos con nula utilidad que generen mayor tiempo de

procesamiento y almacenamiento y en aplicaciones con machine learning se puedan realizar modelos predictivos más certeros con variables significantes en el campo de estudio.

En cuanto al despliegue del servidor , se observa en los documentos de investigación que el lenguaje Python es utilizado solo para el análisis estadístico y creación de servicios a partir de estos, es decir se utiliza el lenguaje Python para creación de diferentes API´s,estas API permiten comunicar distintos tipos de tecnologías como aplicaciones web y móviles implementadas en distintos lenguajes a las funciones y servicios creados, con la finalidad de aplicar todo este análisis de datos en diferentes aplicaciones, en este caso como aplicaciones en geolocalización, rutas óptimas de transporte,estado del tráfico, consultas de información vehicular en tiempo real entre otras.

IX. CONCLUSIONES

El lenguaje de programación Python permite crear un entorno de herramientas enfocadas al desarrollo de proyectos de ciencias de datos, ya que posee cantidad de librerías y funciones para todo tipo de cálculos de tipo estadístico, matemático y operaciones con estructuras de datos, además con estos datos obtenidos permite la graficación y visualización de los mismos en distintos tipos de gráficas, con toda esta información es posible analizar las posibles causas de los accidentes y siniestros vehiculares en la ciudad de Bogotá.

Teniendo en cuenta los resultados anteriores, es evidente encontrar correlaciones entre los accidentes y la localidad, fecha, hora, el tipo de tramo y hasta la época del año, es por ello que se propone realizar un análisis predictivo, en el cual se utilice la información resultante del análisis estadístico y como ya se conocen que tan incidentes son estos factores en la ocurrencia de un accidente, se pueda predecir la gravedad del accidente, esto con el fin de desarrollar estrategias en el campo de la movilidad distrital con el fin de facilitar el conocimiento de las problemáticas que ocasionan estos accidentes, y con ello realizar estrategias que permitan prevenir los accidentes en la ciudad, de la mano de la ciencia de datos y de la rama de la inteligencia artificial conocida como Machine Learning.

X. ANEXOS

Repositorio Github del Proyecto:

https://github.com/SebastianMancera/AnalisisEstadisticoVial

Contiene:

Gráficas Estadísticas generadas.

Información Estadística obtenida.

Jupyter Notebooks con el código fuente.

Aplicativo Web realizado en Flask.

REFERENCIAS

[1]E. Abdullah and A. Emam, "Traffic Accidents Analyzer Using Big Data," 2015 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, 2015, pp. 392-397, doi: 10.1109/CSCI.2015.187.

[2]S. Jamal, H. Zeid, M. Malli and E. Yaacoub, "Safe driving: A mobile application for detecting traffic accidents," 2018 IEEE Middle East and North Africa Communications Conference (MENACOMM), Jounieh, 2018, pp. 1-6, doi: 10.1109/MENACOMM.2018.8371000.

[3] A. V. Sakhare and P. S. Kasbe, "A review on road accident data analysis using data mining techniques," 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, 2017, pp. 1-5, doi: 10.1109/ICIIECS.2017.8275920.

[4]Aksehir, Z.D., Oruç, Y., Elıbol, A., Akleylek, S., & Kili÷, E. (2018). On the Analysis of Work Accidents Data by Using Data Preprocessing and Statistical Techniques. 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), 1-6.

[5] Tai, W., Wang, H., Chiang, C., Chien, C., Lai, K., & Huang, T. (2018). RTAIS: Road Traffic Accident Information System. 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 1393-1397.

[6]Laboratorio Urbano/Bogotá,Coordenadas Georeferenciadas deBogotá.https://bogota-laburbano.opendatasoft.com/explore/dataset/georeferencia-puntual-por-localidad/table/?flg=es