

[Proyecto Final CC3201] (Grupo 33)

Victor Arias
Benjamín Herrera
Jonathan Vikbladh

July 17, 2020

1 Datos

El tema de nuestros datos centra en 2 temas la distribución socio-económica entre las ciudades de los estados de Estados Unidos y su relación con los asesinatos llevados a cabo por la fuerza policial de ese país, para ello se cuenta con datos como porcentaje de pobreza, de escolaridad, de ingreso promedio de los hogares y la distribución por razas en las distintas ciudades, además de datos sobre las causas en las que se produjeron los asesinatos por parte de la policía. Estos datos fueron extraído desde la página <https://www.kaggle.com/kwulum/fatal-police-shootings-in-the-us> donde se hayan en formato CSV y constan de 5 tablas en las que se muestran mas de 24000 ciudades y mas de 800 casos de asesinato policial.

A estos datos se le introdujeron los siguientes cambios: Utilizando el paquete pandas en python. Se fusionaron con un INNER JOIN las tablas de educación, ingreso mediano y tasa de pobreza a una tabla. Utilizando la llave (estado, ciudad). Se elimino las tuplas conteniendo nulos totalmente. Se cambió el nombre de la columna “Geographic area” a “State”. Se aplicó funciones por cada entrada verificando que sean numérico, o transformando el formato de la fecha de “dd/mm/yy” a “yyyy-mm-dd”. También reemplazando “W” y “M” en la columna “gender” por “female” y “male”. Igual para la columna “race”, “W” fue reemplazado por “white” etc. para todas las razas.

2 Esquema Relacional

En la figura 1, se muestra el esquema relacional de proyecto que se compone por 3 tablas: la tabla demographics que muestra la distribución demográfica y cuenta con la llave primaria (state,city), la tabla killing que muestra la información de los asesinatos y cuenta con la llave primaria “id” y la tabla socioeconomics que une los datos socio-económicos de las distintas ciudades y cuenta

con la llave primaria (state,city) Estas tablas cuentan con las siguientes consideraciones para evitar que haya datos erróneos: para la primera (demographics) se tiene las siguientes restricciones share-white ≤ 100 AND share-white ≥ 0 , share-black ≤ 100 AND share-black ≥ 0 , share-native-american ≤ 100 AND share-native-american ≥ 0 , share-asian ≤ 100 AND share-asian ≥ 0 , share-hispanic ≤ 100 AND share-hispanic ≥ 0 y para la ultima (socioeconomics) poverty-rate ≤ 100 AND poverty-rate ≥ 0 , percent-completed-highschool ≤ 100 AND percent-completed-highschool ≥ 0 ,todas con el motivo de verificar que el porcentaje este entre los limites correctos.

```

demographics(state:string,city:string,share-white:float,share-black:float,
share-native-american:float,share-asian:float,share-hispanic:float)
killings(id:int,name:string,date:date,manner-of-death:string,
armed:string,age:int,gender:string,race:string,city:string,state:string,
signs-of-mental-illness,threat-level:string,flee:string,body-camera)
socioeconomics(state:string,city:string,poverty-rate:float,
percent-completed-highschool:float,median-income:float)

```

Figure 1: Esquema relacional

3 Indices, Vistas, etc.

Se usaron las vistas mostviolent que nos muestra todos los estados y ciudades con su cantidad de homicidios y su ranking después de ordenar los en orden descendente, rankpov que muestra estado, ciudad, la tasa de pobreza de esta ciudad y su raking, el cual viene de la tasa de pobreza ordenada ascendente-mente, lo menor la tasa de pobreza implica rango más alto y rankedu que nos muestra estado, ciudad, porcentaje con educación secundaria, ranking y el cual viene de ordenar las ciudades en orden descendente, así que una ciudad con una porcentaje de personas con educación secundaria, tiene un rango más alto, esta vistas nos ayudan a la hora de realizar la consulta 2.

También utilizamos una vista nombrada PromedioPorRaza a partir de los datos de la tabla demographics para agilizar el proceso de calcular el promedio de las distintas razas por estado y ayudarnos en la consulta 3

4 Consultas

En total tenemos 3 consultas:

1. La primera consulta está en la figura 2. Esta consulta nos permite visualizar la cantidad de asesinatos policiales según el estado elegido por el usuario, elegimos esta consulta simple para comenzar.
2. La segunda consulta, mostrada en la figura 3 no posee un input del usuario dada la complejidad de la consulta. Esta nos muestra que el actuar de

la policía es más violento en lugares que tiene menos formación y más pobreza, pues como se puede ver en ella podemos ver que una ciudad con mas violencia se encuentra en los últimos lugares tanto del ranking de formación como de riqueza.

3. La tercera consulta, mostrada en la figura 4 nos permite visualizar el ingreso promedio de los hogares por estado y su distribución racial, para así mostrar si existe una relación de el salario y la raza. Además esta búsqueda hace uso de una vista y anidación para obtener su resultado.

```
SELECT state, COUNT(state) AS cantidad
FROM killings
WHERE state = consultaDelUsuario GROUP BY state;
```

Figure 2: Consulta 1

```
SELECT mostviolent.state, mostviolent.city,
mostviolent.RANK AS ranking_most_violent,
foo.edu AS ranking_most_educated, foo.edu AS ranking_least_poor
JOIN (SELECT rankedu.state AS state, rankedu.city AS city,
rankedu.RANK AS edu, rankpov.RANK AS pov
FROM rankedu
JOIN rankpov
ON rankedu.city=rankpov.city AND rankedu.state=rankpov.state) AS foo
ON mostviolent.city=foo.city AND mostviolent.state=foo.state;
```

Figure 3: Consulta 2

```
SELECT RazaPorEstado.state, psw, psn, psa,
psh, ipe.ingreso
FROM RazaPorEstado, (SELECT state, FLOOR(AVG(median_income)) AS ingreso
FROM socioeconomics GROUP BY state) AS ipe
WHERE RazaPorEstado.state=ipe.state AND
RazaPorEstado.state= consultaDelUsuario;
```

Figure 4: Consulta 3

5 Implementación de la Aplicación

Se utiliza python con flask, sqlalchemy, psycopg2, html y css. La conexión a la base de datos viene por psycopg2, lo cual crea una conexión y luego se ejecuta

las consultas a través de dicha conexión. La aplicación web fue implementada con Flask porque es muy simple, por cada route o url se implementó un template en html para mostrar los datos. La seguridad de inyecciones es garantizada por `psycopg2`, sus funciones se encargan de limpiar lo que ingresa el usuario antes de que se agregue a la consulta SQL.

```
public static int consultar(String stringEntrada) {
    Statement stmt = ...
}
```

Figure 5: El código para consultar la base de datos (ejemplo)

6 Ejemplos de la Aplicación

DAR ALGUNOS EJEMPLOS CONCRETOS DE BÚSQUEDA USANDO SU APLICACIÓN CON LOS RESULTADOS. SE PUEDEN INCLUIR *screenshots*. HAY QUE INCLUIR UN ENLACE A LA APLICACIÓN TAMBIÉN. [APROX. $\frac{1}{2}$ PÁGINA DE DISCUSIÓN SIN CONSIDERAR EL ESPACIO TOMADO POR LOS SCREENSHOTS].

La aplicación está disponible aquí: <http://miapp.com/example/>.
 El usuario puede hacer consultas como ...
 En la figura 6, se muestra la interfaz para hacer consultas donde ...
 En la figura XX, se muestran ejemplos de resultados para una búsqueda ...

Figure 6: El *screenshot* de la página inicial (ejemplo)

7 Lecciones aprendidas

Durante el desarrollo de las consultas nos dimos cuenta que los datos no eran los más propicios para desarrollar consultas de alto interés, pero dado que ya habíamos realizado la normalización de estos tuvimos que seguir el proyecto con los mismos, por ello una de las partes más difíciles del proyecto fue pensar en las consultas. Por otra parte lo que resultó de cierta manera más sencillo fue la búsqueda y obtención de datos pues en internet se puede hallar una gran variedad de datos interesantes.

Algo interesante de este proyecto fue que los datos nos mostraron las diferencias socio económicas y su relación con la violencia en un país como Estados Unidos. Este proyecto nos sirvió de gran experiencia pues nos permitió ver como era la

implementación de una aplicación web que usa una base de datos lo cual nos
acercó de una forma mas practica a lo que se hace en la industria.