1)Datos

El tema de nuestros datos centra en 2 temas la distribución socioeconómica 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/kwullum/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:

**(agragar cambios hechos a los datos)**

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. Utilzando la llave (estado, ciudad). Se eleminó 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 numerico, 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

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:boolean, threat\_level:string, flee:string, body\_camera:boolean)

socioeconomics(**state**:string, **city**:string, **poverty\_rate**:float, **percent\_completed\_highschoo**l:float, **median\_income**:float)

En la imagen anterior se muestra ek 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) las cuales **(tambien son foraneas?)** , 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 socioeconómicos de las distintas ciudades y cuenta con la llave primaria (state,city) las cuales **(también son foraneas?)**

Estas tablas cuentan con las siguientes restricciones:

(agregar restricciones, creo que se refiere a los CHECK)

3)Indıces, Vistas, etc.

Para hacer eficiente la búsqueda usamos un índice de tipo **(diria que deberiamos usar un índice tipo arbol sobre los estados ya que son los que más afectan las búsqueda, sin embargo debo revisarlo antes)**, 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.

4)Consultas

1.La primera consulta que podemos ver a continuacion nos permite visualizar la cantidad de asesinatos policiales según el estado elegido por el usuario (esta es la consulta más simple)

**SELECT state , COUNT(state) AS cantidad FROM killings WHERE state = --------- (** aqui va el estado que quiere el usuario)

2.tengo que traspasarlo

3..tengo que traspasarlo

5)Implementacion de la aplicacion

Describir la implementacion de la aplicacion: que software usaron, como implementaron la conexion a la base de datos, como implementaron la aplicacion Web, como aseguraron la ´ seguridad del sistema contra inyecciones, etc. Con respecto a la seguridad, hay que dar un fragmento de codigo donde se crea(n) y se manda(n) la(s) consultas a la base de datos. [aprox. 1/ 2 pagina de discusi on].

Se utiliza python con flask, sqlalchemy, psycopg2, html y css. La conexion a la base de datos viene por psycopg2, lo cual crea una conexion y luego se ejecuta las consultas a través de dicha conexion. La aplicacion web fue implementado con Flask porque es muy minimo, 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 limpiar lo que ingresa el usuario antes de que se agrega a la consulta SQL.

6)Ejemplos de la Aplicacion

Den algunos ejemplos concretos de busqueda usando su aplicacion con los resultados. Se pueden incluir ´ screenshots. Hay que incluir un enlace a la aplicacion tambıen. [aprox. 1 /2 pagina de discusion sin considerar el espacio tomado por los ´ screenshots].

7)Lecciones aprendidas

Esta parte la hago cuando completemos lo demas