

Coderhouse - Data Analytics

Entrega Final

Caso de estudio: **Siniestros viales
en la provincia de Buenos Aires
Entre 2019 y 2021**

Alumno: **Juan Pablo Ludueña**

ÍNDICE

Tabla de versionado	2
Temática:	3
Alcance:	3
Objetivo:	3
Hipótesis:	3
Origen del Dataset:	4
Herramientas tecnológicas implementadas:	4
Diagrama Entidad - Relación (DER)	5
Listado de Tablas	6
Detalle de columnas	7
Transformación de datos	9
Tablas auxiliares:	18
Modelo de datos en Power Bi	0
Pantallas/Solapas del dashboard	0
Futuras líneas	0

Tabla de versionado

Version	fecha	observaciones
1	15/05/2024	Primer entregable
2	05/06/2024	Segundo entregable
3	08/07/2024	Tercer entregable
4	25/07/2024	Entrega Final

Temática:

Abordaremos el desarrollo en relación a Información sobre lesiones en siniestros viales ocurridos en la Ciudad de Buenos Aires. Los datos incluyen fecha y ubicación del hecho y tipo de transporte involucrado. Además se especifica el género y edad de las víctimas y el tipo de lesión sufrida.

Alcance:

El desarrollo del dashboard apuntará a mostrar información a un nivel táctico, para que los líderes y mandos medios del Observatorio de Movilidad y Seguridad Vial de la Ciudad de Buenos Aires (OMSV) revisen, estudien y tomen las decisiones pertinentes al aumento de siniestros que mostraron un crecimiento sostenido entre los años 2019 y 2021

Objetivo:

Desarrollar un dashboard que permita a los interesados obtener información de calidad para la futura toma de decisiones en relación a los siniestros registrados en la ciudad de Buenos Aires, entre los años 2019 y 2021

Hipótesis:

Es necesario entender si se han aplicado correctas políticas y controles viales dentro del período mencionado que hayan hecho bajar la cantidad de siniestros viales y la gravedad de los mismos, en caso de no ser así es importante tener en cuenta ésta información para futuras tomas de decisiones.

Origen del Dataset:

Éste dataset toma como su principal fuente de información datos policiales, tal y como recomiendan los estándares internacionales. Las estadísticas elaboradas se realizan en base a los sumarios que instruye la Policía de la Ciudad ante la clase de delito: lesiones culposas (Arts. 94 y 94 bis del Código Penal de la Nación)

En el año 2018, comenzó a operar la Policía de la Ciudad, dependiente del Ministerio de Justicia y Seguridad del GCBA (MJyS).

La Policía de la Ciudad cubrió la totalidad del territorio e integró la información que hasta entonces recopilaban las fuerzas de manera independiente. En virtud de ello, a partir del segundo cuatrimestre de 2018, la principal fuente de información sobre siniestros viales con la que trabaja el Observatorio es la del MJyS, fuente que permite conocer una mayor cantidad de casos por cubrir la totalidad del territorio de la Ciudad.

Las estadísticas de delitos por Lesiones Culposas informadas por el MJyS son procesadas por el equipo del Observatorio para verificar que los incidentes se ajusten a las definiciones de siniestro vial vertidas en este documento.

Asimismo, con el relevamiento propio a los trece hospitales de agudos de la Ciudad de Buenos Aires se logró asignar gravedad de la lesión, considerando siniestros graves a aquellos que tuvieron víctimas que debieron ser hospitalizadas por al menos 24hs.

[Link al dataset \(xls\)](#)

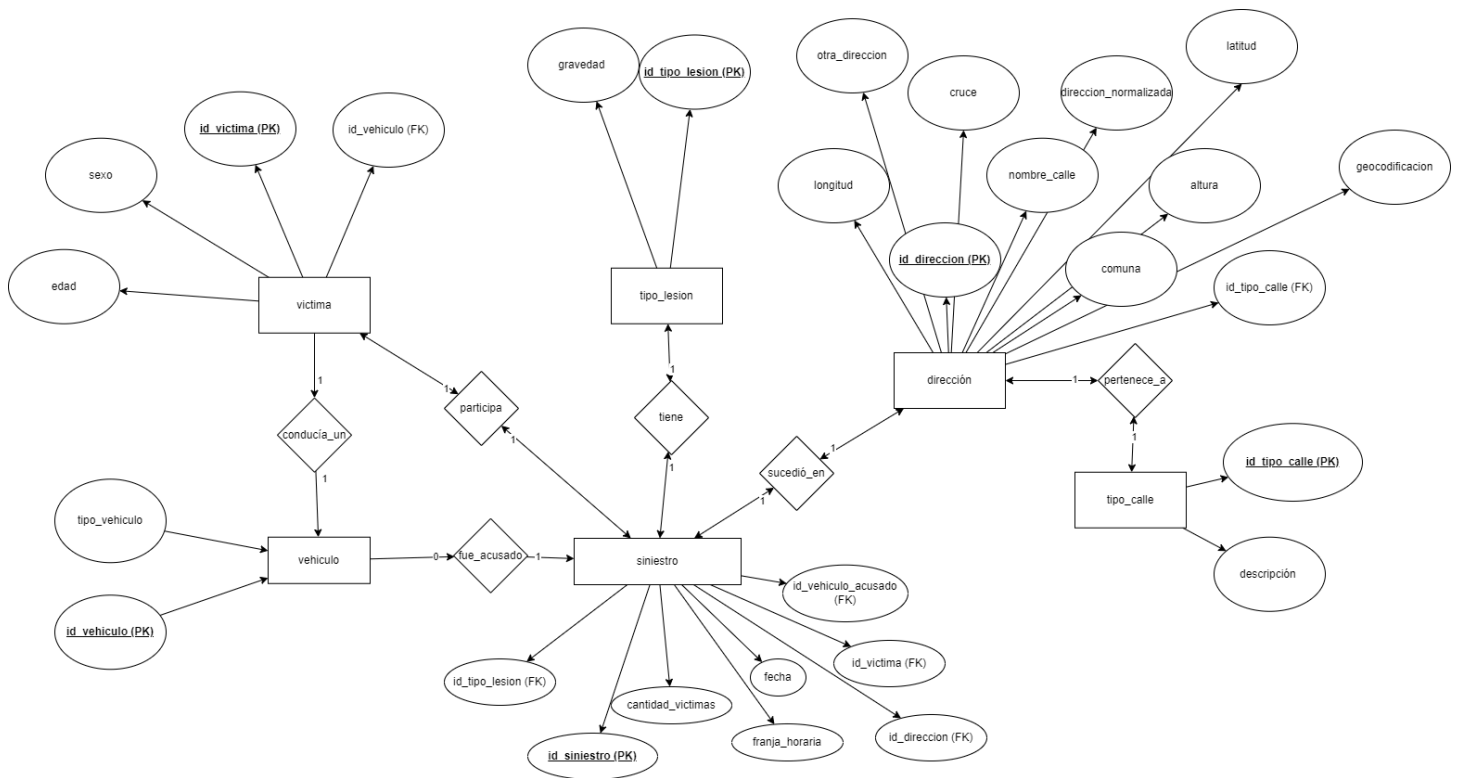
[Link al dataset de comunas \(csv\)](#)

Herramientas tecnológicas implementadas:

- Power BI
- Excel

Diagrama Entidad - Relación (DER)

Para una mejor visualización, ver el siguiente [link](#)



Listado de Tablas

En este apartado, se hará mención de cada una de las tablas junto a una breve descripción de las mismas y la definición de la clave primaria y foránea:

- **siniestro:** Incidente ocurrido en una vía de acceso público en el que participa al menos un vehículo y que resulta en al menos una persona lesionada o fallecida.
 - PK: id_siniestro
 - FK: id_tipo_lesion; id_calle; id_victima; vehiculo_acusado
- **victima:** Datos de quien haya fallecido o se haya lastimado a raíz del hecho, o bien peatón/a
 - PK: id_victima
 - FK: id_vehiculo
- **vehículo:** Datos del vehículo que ocupaba la víctima o que ocupaba quien resultó acusado/a del hecho, sin implicar culpabilidad legal
 - PK: id_vehiculo
 - FK: N/A
- **tipo_de_lesión:** Nivel máximo conocido de gravedad de la lesión de la(s) víctima(s) del siniestro en función del tiempo de hospitalización
 - PK: id_tipo_de_lesion
 - FK: N/A
- **direccion:** Dirección donde se produjo el siniestro
 - PK: id_dirección
 - FK: id_tipo_de_calle
- **tipo_de_calle:** Tipo de arteria. En el caso de intersecciones a nivel se clasifica según la de mayor jerarquía
 - PK: id_tipo_de_calle
 - FK: N/A

Detalle de columnas

En la siguientes páginas presentamos las tablas el DER del proyecto final, con sus Columnas, Descripción, Tipo de Dato, Primary key y Foreign key.

Tabla: siniestro				
columna	descripcion	tipo de dato	PK	FK
id_siniestro	identificador unico del siniestro	numerico	SI	-
cantidad_victimas	cantidad de víctimas	numerico	-	-
fecha	fecha en formato dd/mm/aaaa	fecha	-	-
id_tipo_lesion	id del tipo de lesion	numerico	-	SI
id_vehiculo_acusado	id del vehiculo acusado	numerico	-	SI
id_victima	id de la victima	numerico	-	SI
id_direccion	id de la direccion	numerico	-	SI
franja_horaria	franja horaria entera	numerico	-	-

Tabla: direccion				
columna	descripcion	tipo de dato	PK	FK
id_direccion	identificador de direccion	numerico	SI	-
longitud	longitud con separador punto. WGS84	numerico	-	-
otra_direccion	direccion en formato alternativo	texto	-	-
cruce	cruce en caso de que sea una encrucijada	texto	-	-
direccion_normalizada	direccion en formato normalizado USIG	texto	-	-
nombre_calle	nombre de la arteria donde se produjo el hecho	texto	-	-
comuna	comuna de CABA con jurisdicción según la ubicación	texto	-	-
latitud	latitud con separador punto. WGS84	numerico	-	-
altura	altura de la arteria donde se produjo el hecho	numerico	-	-
id_tipo_calle	identificador del tipo de calle	numerico	-	SI
geocodificacion	geocodificación plana	numerico	-	-

Tabla: tipo_calle				
columna	descripcion	tipo de dato	PK	FK
id_tipo_calle	identificador unico del tipo de calle	numerico	SI	-
descripcion	Tipo de arteria. En el caso de intersecciones a nivel se clasifica según la de mayor jerarquía	texto	-	-

Tabla: tipo_lesion				
columna	descripcion	tipo de dato	PK	FK
id_tipo_lesion	identificador unico del tipo de lesion	numerico	SI	-
gravedad	Nivel máximo conocido de gravedad de la lesión de la(s) víctima(s) del siniestro en función del tiempo de hospitalización	texto	-	-

Tabla: victima				
columna	descripcion	tipo de dato	PK	FK
id_victima	identificador unico de la victima	numerico	SI	-
id_vehiculo	identificador del vehiculo	numerico	-	SI
sexo	Sexo de la víctima según registro policial	texto	-	-
edad	Edad en años enterios de la víctima en registro policial	numerico	-	-

Tabla: vehiculo				
columna	descripcion	tipo de dato	PK	FK
id_vehiculo	identificador unico del vehiculo	numerico	SI	-
tipo_vehiculo	tipo de vehiculo que participó del siniestro	texto	-	-

Transformación de datos

hechos-ppal: contiene los datos del excel principal de “hechos”

- se cambian los tipos de datos
- se reemplazan valores SD por null
- se eliminan columnas "calle", "altura", "cruce", "hora", "tipo_calle", "acusado" ya que no contienen datos o son datos irrelevantes

```
let
    Source = Excel.Workbook(File.Contents("D:\coderhouse\data analytics flex\ds-lesiones-v3 - Juan Pablo Ludueña.xlsx"), null, true),
    Tabla1_Table = Source[{Item="Tabla1",Kind="Table"}][Data],
    #"Changed Type" = Table.TransformColumnTypes(Tabla1_Table,{{"id", type text}, {"n_victimas", Int64.Type}, {"aaaa", Int64.Type}, {"mm", Int64.Type}, {"dd", Int64.Type}, {"fecha", type date}, {"hora", type datetime}, {"franja_hora", Int64.Type}, {"direccion_normalizada", type text}, {"comuna", Int64.Type}, {"tipo_calle", type text}, {"otra_direccion", type text}, {"calle", type any}, {"altura", type any}, {"cruce", type any}, {"geocodificacion_CABA", type text}, {"longitud", type text}, {"latutid", type text}, {"victima", type text}, {"acusado", type text}, {"participantes", type text}, {"moto", type text}, {"auto", type text}, {"transporte_publico", type text}, {"camion", type text}, {"ciclista", type text}, {"gravedad", type text}}),
    #"Replaced Value" = Table.ReplaceValue(#"Changed Type",null,"SD",Replacer.ReplaceValue,{"direccion_normalizada"}),
    #"Removed Columns" = Table.RemoveColumns(#"Replaced Value",{"calle", "altura", "cruce"}),
    #"Replaced Value1" = Table.ReplaceValue(#"Removed Columns",null,"SD",Replacer.ReplaceValue,{"geocodificacion_CABA"}),
    #"Replaced Value2" = Table.ReplaceValue(#"Replaced Value1",null,"SD",Replacer.ReplaceValue,{"longitud"}),
    #"Replaced Value3" = Table.ReplaceValue(#"Replaced Value2",null,"SD",Replacer.ReplaceValue,{"latutid"}),
    #"Removed Columns1" = Table.RemoveColumns(#"Replaced Value3",{"hora", "tipo_calle", "acusado"}),
    #"Uppercased Text" = Table.TransformColumns(#"Removed Columns1",{{"gravedad", Text.Upper, type text}}),
    #"Replaced Value4" = Table.ReplaceValue(#"Uppercased Text","GRAVE","1",Replacer.ReplaceText,{"gravedad"}),
    #"Replaced Value5" = Table.ReplaceValue(#"Replaced Value4","FATAL","2",Replacer.ReplaceText,{"gravedad"})
in
    #"Replaced Value5"
```

victimas-ppal: carga de la base de datos de victimas

- corrección de tipos de datos
- join con el dataset "tipo_vehiculo"

```
let
    Source = Excel.Workbook(File.Contents("D:\coderhouse\data analytics flex\ds-lesiones-v3 - Juan Pablo Ludueña.xlsx"), null, true),
    Tabla2_Table = Source[{Item="Tabla2", Kind="Table"}][Data],
    #"Changed Type" = Table.TransformColumnTypes(Tabla2_Table, {{"ID hecho", type text}, {"AAA", Int64.Type}, {"MM", Int64.Type}, {"DD", Int64.Type}, {"FECHA ", type date}, {"VEHICULO_VICTIMA", type text}, {"SEXO", type text}, {"EDAD_VICTIMA", type any}, {"GRAVEDAD", type text}}),
    #"Merged Queries" = Table.NestedJoin(#"Changed Type", {"VEHICULO_VICTIMA"}, tipo_vehiculo, {"tipo_vehiculo"}, "tipo_vehiculo", JoinKind.LeftOuter),
    #"Expanded tipo_vehiculo" = Table.ExpandTableColumn(#"Merged Queries", "tipo_vehiculo", {"id_tipo_vehiculo"}, {"tipo_vehiculo.id_tipo_vehiculo"}),
    #"Renamed Columns" = Table.RenameColumns(#"Expanded tipo_vehiculo", {"tipo_vehiculo.id_tipo_vehiculo", "id_tipo_vehiculo"})
in
    #"Renamed Columns"
```

comunas: carga de base de datos de comunas, para tener mas detalle de la dirección del siniestro

- corrección de tipo de datos

```
let
    Source = Csv.Document(File.Contents("D:\pablo-descargas\comunas V2 - Juan Pablo Ludueña - comunas V2 - Juan Pablo Ludueña.csv"), [Delimiter=",", Columns=3, Encoding=65001, QuoteStyle=QuoteStyle.None]),
    #"Promoted Headers" = Table.PromoteHeaders(Source, [PromoteAllScalars=true]),
    #"Changed Type" = Table.TransformColumnTypes(#"Promoted Headers", {"ID", Int64.Type}, {"nombre", type text}, {"BARRIOS", type text})),
    #"Inserted Merged Column" = Table.AddColumn(#"Changed Type", "nombre_maps", each Text.Combine([nombre], ", Ciudad Autónoma de Buenos Aires")),
    #"Changed Type1" = Table.TransformColumnTypes(#"Inserted Merged Column", {"nombre_maps", type text})
in
    #"Changed Type1"
```

tipo-lesion: dataset que contiene la tipificación de las lesiones

```
let
    Source = Excel.Workbook(File.Contents("D:\coderhouse\data analytics flex\ds-lesiones-v3 - Juan Pablo Ludueña.xlsx"), null, true),
    Tabla1_Table = Source[{Item="Tabla1", Kind="Table"}][Data],
```

```

    #"Changed Type" = Table.TransformColumnTypes(Tabla1_Table,{{"id", type text},
{"n_victimas", Int64.Type}, {"aaaa", Int64.Type}, {"mm", Int64.Type}, {"dd", Int64.Type},
{"fecha", type date}, {"hora", type datetime}, {"franja_hora", Int64.Type},
{"direccion_normalizada", type text}, {"comuna", Int64.Type}, {"tipo_calle", type text},
{"otra_direccion", type text}, {"calle", type any}, {"altura", type any}, {"cruce", type any},
{"geocodificacion_CABA", type text}, {"longitud", type text}, {"latutid", type text},
{"victima", type text}, {"acusado", type text}, {"participantes", type text}, {"moto", type
text}, {"auto", type text}, {"transporte_publico", type text}, {"camion", type text},
{"ciclista", type text}, {"gravedad", type text}}),
    #"Replaced Value" = Table.ReplaceValue(#"Changed
Type",null,"SD",Replacer.ReplaceValue,{"direccion_normalizada"}),
    #"Removed Columns" = Table.RemoveColumns(#"Replaced Value",{"calle", "altura", "cruce"}),
    #"Replaced Value1" = Table.ReplaceValue(#"Removed
Columns",null,"SD",Replacer.ReplaceValue,{"geocodificacion_CABA"}),
    #"Replaced Value2" = Table.ReplaceValue(#"Replaced
Value1",null,"SD",Replacer.ReplaceValue,{"longitud"}),
    #"Replaced Value3" = Table.ReplaceValue(#"Replaced
Value2",null,"SD",Replacer.ReplaceValue,{"latutid"}),
    #"Removed Columns1" = Table.RemoveColumns(#"Replaced Value3",{"hora", "tipo_calle",
"acusado"}),
    #"Uppercased Text" = Table.TransformColumns(#"Removed Columns1",{{"gravedad", Text.Upper,
type text}}),
    #"Removed Columns2" = Table.RemoveColumns(#"Uppercased Text",{"id", "n_victimas", "aaaa",
"mm", "dd", "fecha", "franja_hora", "direccion_normalizada", "comuna", "otra_direccion",
"geocodificacion_CABA", "longitud", "latutid", "victima", "participantes", "moto", "auto",
"transporte_publico", "camion", "ciclista"}),
    #"Removed Duplicates" = Table.Distinct(#"Removed Columns2"),
    #"Added Index" = Table.AddIndexColumn(#"Removed Duplicates", "Index", 1, 1, Int64.Type),
    #"Renamed Columns" = Table.RenameColumns(#"Added Index",{{"Index", "id_tipo_lesion"}}),
    #"Replaced Value4" = Table.ReplaceValue(#"Renamed
Columns","SD","LEVE",Replacer.ReplaceText,{"gravedad"})
in
    #"Replaced Value4"

```

hechos-ppal-dir-norm: dataset que se crea para poder hacer union con el dataset “hechos-ppal-otra-dir” y así poder únificar las direcciones de los siniestros en una sola columna

```

let
    Source = Excel.Workbook(File.Contents("D:\coderhouse\data analytics flex\ds-lesiones-v3 -
Juan Pablo Ludueña.xlsx"), null, true),
    Tabla1_Table = Source[{Item="Tabla1",Kind="Table"}][Data],
    #"Changed Type" = Table.TransformColumnTypes(Tabla1_Table,{{"id", type text},
{"n_victimas", Int64.Type}, {"aaaa", Int64.Type}, {"mm", Int64.Type}, {"dd", Int64.Type},
{"fecha", type date}, {"hora", type datetime}, {"franja_hora", Int64.Type},
{"direccion_normalizada", type text}, {"comuna", Int64.Type}, {"tipo_calle", type text},
{"otra_direccion", type text}, {"calle", type any}, {"altura", type any}, {"cruce", type any},
{"geocodificacion_CABA", type text}, {"longitud", type text}, {"latutid", type text},
{"victima", type text}, {"acusado", type text}, {"participantes", type text}, {"moto", type
text}, {"auto", type text}, {"transporte_publico", type text}, {"camion", type text},
{"ciclista", type text}, {"gravedad", type text}}),

```

```

    #"Replaced Value" = Table.ReplaceValue("#Changed
Type",null,"SD",Replacer.ReplaceValue,{"direccion_normalizada"}),
    #"Removed Columns" = Table.RemoveColumns("#Replaced Value,{"calle", "altura", "cruce"}),
    #"Replaced Value1" = Table.ReplaceValue("#Removed
Columns",null,"SD",Replacer.ReplaceValue,{"geocodificacion_CABA"}),
    #"Replaced Value2" = Table.ReplaceValue("#Replaced
Value1",null,"SD",Replacer.ReplaceValue,{"longitud"}),
    #"Replaced Value3" = Table.ReplaceValue("#Replaced
Value2",null,"SD",Replacer.ReplaceValue,{"latutid"}),
    #"Removed Columns1" = Table.RemoveColumns("#Replaced Value3,{"hora", "tipo_calle",
"acusado"}),
    #"Uppercased Text" = Table.TransformColumns("#Removed Columns1",{{"gravedad", Text.Upper,
type text}}),
    #"Replaced Value4" = Table.ReplaceValue("#Uppercased
Text","GRAVE","1",Replacer.ReplaceText,{"gravedad"}),
    #"Replaced Value5" = Table.ReplaceValue("#Replaced
Value4","FATAL","2",Replacer.ReplaceText,{"gravedad"}),
    #"Filtered Rows" = Table.SelectRows("#Replaced Value5", each ([direccion_normalizada] <>
"SD")),
    #"Removed Columns2" = Table.RemoveColumns("#Filtered Rows,{"otra_direccion"}),
    #"Renamed Columns" = Table.RenameColumns("#Removed Columns2",{{"direccion_normalizada",
"direccion"}})
in
    #"Renamed Columns"

```

hechos-ppal-otra-dir: dataset que se crea para poder hacer union con el dataset "hechos-ppal-dir-norm" y así poder unificar las direcciones de los siniestros en una sola columna

```

let
    Source = Excel.Workbook(File.Contents("D:\coderhouse\data analytics flex\ds-lesiones-v3 -
Juan Pablo Ludueña.xlsx"), null, true),
    Tabla1_Table = Source[{Item="Tabla1",Kind="Table"}][Data],
    #"Changed Type" = Table.TransformColumnTypes(Tabla1_Table,{{"id", type text},
{"n_victimas", Int64.Type}, {"aaaa", Int64.Type}, {"mm", Int64.Type}, {"dd", Int64.Type},
{"fecha", type date}, {"hora", type datetime}, {"franja_hora", Int64.Type},
{"direccion_normalizada", type text}, {"comuna", Int64.Type}, {"tipo_calle", type text},
{"otra_direccion", type text}, {"calle", type any}, {"altura", type any}, {"cruce", type any},
{"geocodificacion_CABA", type text}, {"longitud", type text}, {"latutid", type text},
{"victima", type text}, {"acusado", type text}, {"participantes", type text}, {"moto", type
text}, {"auto", type text}, {"transporte_publico", type text}, {"camion", type text},
{"ciclista", type text}, {"gravedad", type text}}),
    #"Replaced Value" = Table.ReplaceValue("#Changed
Type",null,"SD",Replacer.ReplaceValue,{"direccion_normalizada"}),
    #"Removed Columns" = Table.RemoveColumns("#Replaced Value,{"calle", "altura", "cruce"}),
    #"Replaced Value1" = Table.ReplaceValue("#Removed
Columns",null,"SD",Replacer.ReplaceValue,{"geocodificacion_CABA"}),
    #"Replaced Value2" = Table.ReplaceValue("#Replaced
Value1",null,"SD",Replacer.ReplaceValue,{"longitud"}),
    #"Replaced Value3" = Table.ReplaceValue("#Replaced
Value2",null,"SD",Replacer.ReplaceValue,{"latutid"}),

```

```

    #"Removed Columns1" = Table.RemoveColumns(#"Replaced Value3",{ "hora", "tipo_calle",
"acusado"}),
    #"Uppercased Text" = Table.TransformColumns(#"Removed Columns1",{{"gravedad", Text.Upper,
type text}}),
    #"Replaced Value4" = Table.ReplaceValue(#"Uppercased
Text", "GRAVE", "1", Replacer.ReplaceText, {"gravedad"}),
    #"Replaced Value5" = Table.ReplaceValue(#"Replaced
Value4", "FATAL", "2", Replacer.ReplaceText, {"gravedad"}),
    #"Filtered Rows" = Table.SelectRows(#"Replaced Value5", each ([otra_direccion] <>
"Comisaria: Vecinal 13A. Calle y altura: Figueroa Alcorta, Pres. Av. 6400 Piso: Dpto:
Barrio: Belgrano" and [otra_direccion] <> "Comisaria: Vecinal 13A. Calle y altura: Pino,
Virrey Del 2200 Piso: Dpto: Barrio: Belgrano" and [otra_direccion] <> "Comisaria: Vecinal
14A. Calle y altura: Santa Fe Av. 3700 Piso: Dpto: Barrio: Palermo" and [otra_direccion] <>
"Comisaria: Vecinal 15B. Calle y altura: Cordoba Av. 4690 Piso: Dpto: Barrio: Villa
Crespo" and [otra_direccion] <> "Comisaria: Vecinal 1A. Calle y altura: Pellegrini, Carlos
1175 Piso: Dpto: Barrio: Retiro" and [otra_direccion] <> "Comisaria: Vecinal 1C. Calle y
altura: San Juan Av. 1800 Piso: Dpto: Barrio: Constitucion" and [otra_direccion] <>
"Comisaria: Vecinal 1E. Calle y altura: Manso Juana 1601 Piso: Dpto: Barrio: Puerto
Madero" and [otra_direccion] <> "Comisaria: Vecinal 7C. Calle y altura: BONIFACIO, JOSE 2948
Piso: Dpto: Barrio: Flores" and [otra_direccion] <> "Comisaria: Vecinal 8C. Calle y altura:
Roca, Cnel. Av. y Larrazabal Av. 0 Piso: Dpto: Barrio: Villa Lugano" and [otra_direccion] <>
"Comisaria: Vecinal 9A. Calle y altura: Cardenas, Cnel. Av. y Tandil 0 Piso: Dpto: Barrio:
Mataderos" and [otra_direccion] <> "SD")),
    #"Removed Columns2" = Table.RemoveColumns(#"Filtered Rows",{ "direccion_normalizada"}),
    #"Renamed Columns" = Table.RenameColumns(#"Removed Columns2",{{"otra_direccion",
"direccion"}}),
    #"Reordered Columns" = Table.ReorderColumns(#"Renamed Columns",{ "id", "n_victimas",
"aaaa", "mm", "dd", "fecha", "franja_hora", "direccion", "comuna", "geocodificacion_CABA",
"longitud", "latitud", "victima", "participantes", "moto", "auto", "transporte_publico",
"camion", "ciclista", "gravedad"})
in
    #"Reordered Columns"

```

hechos-ppal-normalizada: dataset final de siniestros es el dataset de hechos del modelo estrella

```

let
    Source = Table.Combine({#"hechos-ppal-dir-norm", #"hechos-ppal-otra-dir"}),
    #"Merged Queries" = Table.NestedJoin(Source, {"direccion"}, direcciones, {"direccion"},
"direcciones", JoinKind.LeftOuter),
    #"Renamed Columns" = Table.RenameColumns(#"Merged Queries",{{"direcciones",
"direcciones_id"}}),
    #"Expanded direcciones_id" = Table.ExpandTableColumn(#"Renamed Columns", "direcciones_id",
{"direcciones_id"}, {"direcciones_id.direcciones_id"}),
    #"Removed Columns" = Table.RemoveColumns(#"Expanded direcciones_id",{ "moto", "auto",
"transporte_publico", "camion", "ciclista"}),
    #"Split Column by Delimiter" = Table.SplitColumn(#"Removed Columns", "participantes",
Splitter.SplitTextByDelimiter("-", QuoteStyle.Csv), {"participantes.1", "participantes.2"}),
    #"Changed Type" = Table.TransformColumnTypes(#"Split Column by
Delimiter",{{"participantes.1", type text}, {"participantes.2", type text}}),
    #"Removed Columns1" = Table.RemoveColumns(#"Changed Type",{ "victima"}),

```

```

    # "Renamed Columns1" = Table.RenameColumns("#Removed Columns1",{{"participantes.1",
"victima"}, {"participantes.2", "acusado"}}),
    # "Merged Queries1" = Table.NestedJoin("#Renamed Columns1", {"victima"}, tipo_vehiculo,
{"tipo_vehiculo"}, "tipo_vehiculo", JoinKind.LeftOuter),
    # "Expanded tipo_vehiculo" = Table.ExpandTableColumn("#Merged Queries1", "tipo_vehiculo",
{"id_tipo_vehiculo"}, {"tipo_vehiculo.id_tipo_vehiculo"}),
    # "Renamed Columns2" = Table.RenameColumns("#Expanded
tipo_vehiculo",{{"tipo_vehiculo.id_tipo_vehiculo", "id_tipo_vehiculo_victima"}}),
    # "Merged Queries2" = Table.NestedJoin("#Renamed Columns2", {"acusado"}, tipo_vehiculo,
{"tipo_vehiculo"}, "tipo_vehiculo", JoinKind.LeftOuter),
    # "Expanded tipo_vehiculo1" = Table.ExpandTableColumn("#Merged Queries2", "tipo_vehiculo",
{"id_tipo_vehiculo"}, {"tipo_vehiculo.id_tipo_vehiculo"}),
    # "Renamed Columns3" = Table.RenameColumns("#Expanded
tipo_vehiculo1",{{"tipo_vehiculo.id_tipo_vehiculo", "id_tipo_vehiculo_acusado"}}),
    # "Replaced Errors" = Table.ReplaceErrorValues("#Renamed Columns3", {{"franja_hora", 0}}),
    # "Renamed Columns4" = Table.RenameColumns("#Replaced Errors",{{"gravedad",
"id_tipo_lesion"}, {"direcciones_id.direcciones_id", "direccion_id"}}),
    # "Replaced Value" = Table.ReplaceValue("#Renamed
Columns4", "SD", null, Replacer.ReplaceValue, {"longitud"}),
    # "Changed Type1" = Table.TransformColumnTypes("#Replaced Value",{{"longitud",
Int64.Type}}),
    # "Replaced Value1" = Table.ReplaceValue("#Changed
Type1", "sd", null, Replacer.ReplaceValue, {"geocodificacion_CABA"}),
    # "Replaced Value2" = Table.ReplaceValue("#Replaced
Value1", "sd", null, Replacer.ReplaceValue, {"latutid"}),
    # "Replaced Value3" = Table.ReplaceValue("#Replaced
Value2", "SD", null, Replacer.ReplaceValue, {"geocodificacion_CABA"}),
    # "Replaced Value4" = Table.ReplaceValue("#Replaced
Value3", "SD", null, Replacer.ReplaceValue, {"latutid", "victima", "acusado", "id_tipo_lesion"}),
    # "Changed Type2" = Table.TransformColumnTypes("#Replaced Value4",{{"latutid", Int64.Type},
{"victima", type text}, {"acusado", type text}, {"id_tipo_lesion", Int64.Type},
{"direccion_id", type any}}),
    # "Merged Queries3" = Table.NestedJoin("#Changed Type2", {"direccion"}, direcciones,
{"direccion"}, "direcciones", JoinKind.LeftOuter),
    # "Removed Columns2" = Table.RemoveColumns("#Merged Queries3", {"direccion_id"}),
    # "Expanded direcciones1" = Table.ExpandTableColumn("#Removed Columns2", "direcciones",
{"id_direccion"}, {"direcciones.id_direccion"}),
    # "Renamed Columns7" = Table.RenameColumns("#Expanded
direcciones1",{{"direcciones.id_direccion", "id_direccion"}}),
    # "Replaced Value5" = Table.ReplaceValue("#Renamed
Columns7", null, 1, Replacer.ReplaceValue, {"id_tipo_vehiculo_victima"}),
    # "Replaced Value6" = Table.ReplaceValue("#Replaced
Value5", null, 1, Replacer.ReplaceValue, {"id_tipo_vehiculo_acusado"}),
    # "Replaced Value7" = Table.ReplaceValue("#Replaced
Value6", 2, 3, Replacer.ReplaceValue, {"id_tipo_lesion"}),
    # "Replaced Value8" = Table.ReplaceValue("#Replaced
Value7", 1, 2, Replacer.ReplaceValue, {"id_tipo_lesion"}),
    # "Replaced Value9" = Table.ReplaceValue("#Replaced
Value8", null, 1, Replacer.ReplaceValue, {"id_tipo_lesion"})
in
    # "Replaced Value9"

```

direcciones: se separa éste dataset para tener las direcciones separado de la tabla de hechos

```
let
    Source = Table.Combine({#"hechos-ppal-dir-norm", #"hechos-ppal-otra-dir"}),
    #"Removed Columns" = Table.RemoveColumns(Source,{ "id", "n_victimas", "aaaa", "mm", "dd",
"fecha", "franja_hora", "victima", "participantes", "moto", "auto", "transporte_publico",
"camion", "ciclista", "gravedad"}),
    #"Added Index" = Table.AddIndexColumn(#"Removed Columns", "Index", 1, 1, Int64.Type),
    #"Reordered Columns" = Table.ReorderColumns(#"Added Index",{ "Index", "direccion",
"comuna", "geocodificacion_CABA", "longitud", "latutid"}),
    #"Renamed Columns" = Table.RenameColumns(#"Reordered Columns",{{"Index",
"id_direccion"}}),
    #"Removed Columns1" = Table.RemoveColumns(#"Renamed Columns",{ "id_direccion"}),
    #"Removed Duplicates" = Table.Distinct(#"Removed Columns1", {"direccion"}),
    #"Added Index1" = Table.AddIndexColumn(#"Removed Duplicates", "Index", 1, 1, Int64.Type),
    #"Renamed Columns1" = Table.RenameColumns(#"Added Index1",{{"Index", "id_direccion"},
{"comuna", "id_comuna"}}),
    #"Replaced Value" = Table.ReplaceValue(#"Renamed
Columns1", "SD", null, Replacer.ReplaceValue, {"direccion", "geocodificacion_CABA", "longitud",
"latutid"}),
    #"Replaced Value1" = Table.ReplaceValue(#"Replaced
Value", null, 12, Replacer.ReplaceValue, {"id_comuna"})
in
    #"Replaced Value1"
```

tipo_vehiculo_victima: se separa éste dataset para tener las direcciones separado de la tabla de hechos

```
let
    Source = Table.Combine({#"hechos-ppal-dir-norm", #"hechos-ppal-otra-dir"}),
    #"Merged Queries" = Table.NestedJoin(Source, {"direccion"}, direcciones, {"direccion"},
"direcciones", JoinKind.LeftOuter),
    #"Renamed Columns" = Table.RenameColumns(#"Merged Queries",{{"direcciones",
"direcciones_id"}}),
    #"Expanded direcciones_id" = Table.ExpandTableColumn(#"Renamed Columns", "direcciones_id",
{"direcciones_id"}, {"direcciones_id.direcciones_id"}),
    #"Removed Columns" = Table.RemoveColumns(#"Expanded direcciones_id",{ "moto", "auto",
"transporte_publico", "camion", "ciclista"}),
    #"Split Column by Delimiter" = Table.SplitColumn(#"Removed Columns", "participantes",
Splitter.SplitTextByDelimiter("-", QuoteStyle.Csv), {"participantes.1", "participantes.2"}),
    #"Changed Type" = Table.TransformColumnTypes(#"Split Column by
Delimiter",{{"participantes.1", type text}, {"participantes.2", type text}}),
    #"Removed Columns1" = Table.RemoveColumns(#"Changed Type",{ "victima"}),
    #"Renamed Columns1" = Table.RenameColumns(#"Removed Columns1",{{"participantes.1",
"victima"}, {"participantes.2", "acusado"}}),
    #"Removed Other Columns" = Table.SelectColumns(#"Renamed Columns1",{ "victima"}),
    #"Renamed Columns2" = Table.RenameColumns(#"Removed Other Columns",{{"victima",
"tipo_vehiculo"}})
in
    #"Renamed Columns2"
```


tipo_vehiculo_acusado: se separa éste dataset para tener las direcciones separado de la tabla de hechos

```
let
    Source = Table.Combine({#"hechos-ppal-dir-norm", #"hechos-ppal-otra-dir"}),
    #"Merged Queries" = Table.NestedJoin(Source, {"direccion"}, direcciones, {"direccion"},
"direcciones", JoinKind.LeftOuter),
    #"Renamed Columns" = Table.RenameColumns(#"Merged Queries",{{"direcciones",
"direcciones_id"}}),
    #"Expanded direcciones_id" = Table.ExpandTableColumn(#"Renamed Columns", "direcciones_id",
{"direcciones_id"}, {"direcciones_id.direcciones_id"}),
    #"Removed Columns" = Table.RemoveColumns(#"Expanded direcciones_id",{"moto", "auto",
"transporte_publico", "camion", "ciclista"}),
    #"Split Column by Delimiter" = Table.SplitColumn(#"Removed Columns", "participantes",
Splitter.SplitTextByDelimiter("-", QuoteStyle.Csv), {"participantes.1", "participantes.2"}),
    #"Changed Type" = Table.TransformColumnTypes(#"Split Column by
Delimiter",{{"participantes.1", type text}, {"participantes.2", type text}}),
    #"Removed Columns1" = Table.RemoveColumns(#"Changed Type",{"victima"}),
    #"Renamed Columns1" = Table.RenameColumns(#"Removed Columns1",{{"participantes.1",
"victima"}, {"participantes.2", "acusado"}}),
    #"Removed Other Columns" = Table.SelectColumns(#"Renamed Columns1",{"acusado"}),
    #"Renamed Columns2" = Table.RenameColumns(#"Removed Other Columns",{{"acusado",
"tipo_vehiculo"}})
in
    #"Renamed Columns2"
```

tipo_vehiculo: dataset que contiene la tipificación de los vehículos intervinientes en los siniestros

```
let
    Source = Table.Combine({tipo_vehiculo_victima, tipo_vehiculo_acusado}),
    #"Removed Duplicates" = Table.Distinct(Source),
    #"Removed Top Rows" = Table.Skip(#"Removed Duplicates",1),
    #"Added Index" = Table.AddIndexColumn(#"Removed Top Rows", "Index", 1, 1, Int64.Type),
    #"Renamed Columns" = Table.RenameColumns(#"Added Index",{{"Index", "id_tipo_vehiculo"}}),
    #"Reordered Columns" = Table.ReorderColumns(#"Renamed Columns",{"id_tipo_vehiculo",
"tipo_vehiculo"})
in
    #"Reordered Columns"
```

victimas-ppal-normalizada: dataset principal de victimas, contiene todas las modificaciones necesarias para la obtencion de datos

```

let
    Source = Excel.Workbook(File.Contents("D:\coderhouse\data analytics flex\ds-lesiones-v3 -
Juan Pablo Ludueña.xlsx"), null, true),
    Tabla2_Table = Source[{Item="Tabla2",Kind="Table"}][Data],
    #"Changed Type" = Table.TransformColumnTypes(Tabla2_Table,{{"ID hecho", type text},
{"AAA", Int64.Type}, {"MM", Int64.Type}, {"DD", Int64.Type}, {"FECHA ", type date},
{"VEHICULO_VICTIMA", type text}, {"SEXO", type text}, {"EDAD_VICTIMA", type any}, {"GRAVEDAD",
type text}}),
    #"Merged Queries" = Table.NestedJoin(#"Changed Type", {"VEHICULO_VICTIMA"}, tipo_vehiculo,
{"tipo_vehiculo"}, "tipo_vehiculo", JoinKind.LeftOuter),
    #"Expanded tipo_vehiculo" = Table.ExpandTableColumn(#"Merged Queries", "tipo_vehiculo",
{"id_tipo_vehiculo"}, {"tipo_vehiculo.id_tipo_vehiculo"}),
    #"Renamed Columns" = Table.RenameColumns(#"Expanded
tipo_vehiculo",{{"tipo_vehiculo.id_tipo_vehiculo", "id_tipo_vehiculo"}}),
    #"Removed Columns" = Table.RemoveColumns(#"Renamed Columns",{"AAA", "MM", "DD", "FECHA ",
"GRAVEDAD"}),
    #"Added Index" = Table.AddIndexColumn(#"Removed Columns", "Index", 1, 1, Int64.Type),
    #"Renamed Columns1" = Table.RenameColumns(#"Added Index",{{"Index", "id_victima"}}),
    #"Reordered Columns" = Table.ReorderColumns(#"Renamed Columns1",{"id_victima", "ID hecho",
"VEHICULO_VICTIMA", "SEXO", "EDAD_VICTIMA", "id_tipo_vehiculo"}),
    #"Changed Type1" = Table.TransformColumnTypes(#"Reordered Columns",{{"EDAD_VICTIMA", type
number}}),
    #"Replaced Value" = Table.ReplaceValue(#"Changed
Type1", "SD", null, Replacer.ReplaceValue, {"SEXO"}),
    #"Replaced Value1" = Table.ReplaceValue(#"Replaced
Value", "sd", null, Replacer.ReplaceValue, {"VEHICULO_VICTIMA"}),
    #"Replaced Errors" = Table.ReplaceErrorValues(#"Replaced Value1", {{"EDAD_VICTIMA",
null}}),
    #"Filtered Rows" = Table.SelectRows(#"Replaced Errors", each true),
    #"Replaced Value2" = Table.ReplaceValue(#"Filtered
Rows", "sd", null, Replacer.ReplaceValue, {"SEXO"}),
    #"Changed Type2" = Table.TransformColumnTypes(#"Replaced Value2",{{"EDAD_VICTIMA",
Int64.Type}}),
    #"Replaced Value3" = Table.ReplaceValue(#"Changed Type2", null, "No
Definido", Replacer.ReplaceValue, {"SEXO"})
in
    #"Replaced Value3"

```

Tablas auxiliares:

- tabla_medidas

```
cantidad_hechos =  
// cantidad de hechos registrados, si es nulo devolvemos 0  
var cant = COUNT('hechos-ppal-normalizada'[id_tipo_lesion])  
RETURN IF(ISBLANK(cant),0,cant)  
  
cantidad_hechos_fatales =  
// cantidad de hechos que tuvieron victimas fatales, si el valor es  
nulo devolvemos 0  
var cant =  
CALCULATE(  
    COUNT('hechos-ppal-normalizada'[id_tipo_lesion]),  
    UPPER('tipo-lesion'[gravedad]) = "FATAL"  
)  
RETURN IF(ISBLANK(cant),0,cant)  
  
cantidad_hechos_graves =  
// cantidad de accidentes graves registrados, si el valor es nulo  
devolvemos 0  
var cant =  
CALCULATE(  
    COUNT('hechos-ppal-normalizada'[id_tipo_lesion]),  
    UPPER('tipo-lesion'[gravedad]) = "GRAVE"  
)  
RETURN IF(ISBLANK(cant),0,cant)  
  
cantidad_hechos_leves =  
// cantidad de accidentes leves registrados, si el valor es nulo  
devolvemos 0  
var cant =  
CALCULATE(  
    COUNT('hechos-ppal-normalizada'[id_tipo_lesion]),  
    UPPER('tipo-lesion'[gravedad]) = "LEVE"  
)  
RETURN IF(ISBLANK(cant),0,cant)  
  
cantidad_personas_en_siniestros =  
// cantidad total de victimas registradas en todos los siniestros  
COUNT('victimas-ppal-normalizada'[id_victima])
```

```
cantidad_siniestros_automovil =
// cantidad de siniestros en los que participo un vehiculo de tipo
automovil
CALCULATE(
    [cantidad_hechos],
    'hechos-ppal-normalizada'[id_tipo_vehiculo_victima] = 1
)

cantidad_tipo_vehiculo_acusado =
// cantidad de tipos de vehiculos utilizados en los siniestros (del
acusado)
COUNT('hechos-ppal-normalizada'[id_tipo_vehiculo_acusado])

cantidad_tipo_vehiculo_victima =
// cantidad por tipo de vehiculo de la victima
COUNT('hechos-ppal-normalizada'[id_tipo_vehiculo_victima])

cantidad_total_victimas =
// cantidad total de victimas involucradas en siniestros
COUNT('victimas-ppal-normalizada'[id_victima])

cantidad_victimas_mayores_edad =
// cantidad de victimas mayores de edad (mayores o iguales a 18 años)
CALCULATE(
    COUNT('victimas-ppal-normalizada'[id_victima]),
    'victimas-ppal-normalizada'[EDAD_VICTIMA] >= 18
)

cantidad_victimas_menores_edad =
// cantidad de victimas menores de edad (menos de 18 años)
CALCULATE(
    COUNT('victimas-ppal-normalizada'[id_victima]),
    'victimas-ppal-normalizada'[EDAD_VICTIMA] < 18
)

cantidad_victimas_sexo_femenino =
// cantidad de victimas femeninas
CALCULATE(
    COUNT('victimas-ppal-normalizada'[SEXO]),
    UPPER('victimas-ppal-normalizada'[SEXO]) = "MUJER"
)
```

```
cantidad_victimas_sexo_masculino =  
// cantidad de victimas masculinas  
CALCULATE(  
    COUNT('victimas-ppal-normalizada'[SEXO]),  
    UPPER('victimas-ppal-normalizada'[SEXO]) = "VARON"  
)  
  
cantidad_victimas_sexo_no_definido =  
// cantidad de victimas con sexo no definido  
CALCULATE(  
    COUNT('victimas-ppal-normalizada'[SEXO]),  
    UPPER('victimas-ppal-normalizada'[SEXO]) = "NO DEFINIDO"  
)  
  
promedio_accidentes_por_dia =  
AVERAGE(total_siniestros_por_fecha[total_siniestros])  
  
promedio_edad_victimas =  
// calcula el promedio de edad de las victimas  
CALCULATE(  
    AVERAGE('victimas-ppal-normalizada'[EDAD_VICTIMA]),  
    'victimas-ppal-normalizada'[EDAD_VICTIMA] <> BLANK()  
)
```

- **tabla_calendario**

```
tabla_calendario =  
CALENDAR(  
    MIN('hechos-ppal-normalizada'[fecha]),  
    MAX('hechos-ppal-normalizada'[fecha])  
)
```

- **cantidad_hechos_fatales_por_comuna**

//tabla que contiene la cantidad de accidentes fatales sumariados por comuna

```
SUMMARIZE(  
    'hechos-ppal-normalizada',  
    comunas[ID],  
    "total_de_fatalidades",  
    CALCULATE(  
        COUNTROWS(  
            'hechos-ppal-normalizada'  
        ),  
        'hechos-ppal-normalizada'[id_tipo_lesion] = 3  
    )  
)
```

- **comuna_con_mas_fatalidades**

// registro de la comuna con mas fatalidades registradas

```
TOPN(  
    1,  
    'cantidad_hechos_fatales_por_comuna',  
    'cantidad_hechos_fatales_por_comuna'[total_de_fatalidades],  
    DESC  
)
```

- **fecha_mas_siniestros_registrados**

// registro de fechas con mayor cantidad de siniestros

```
TOPN(  
    1,  
    total_siniestros_por_fecha,  
    total_siniestros_por_fecha[total_siniestros]  
)
```

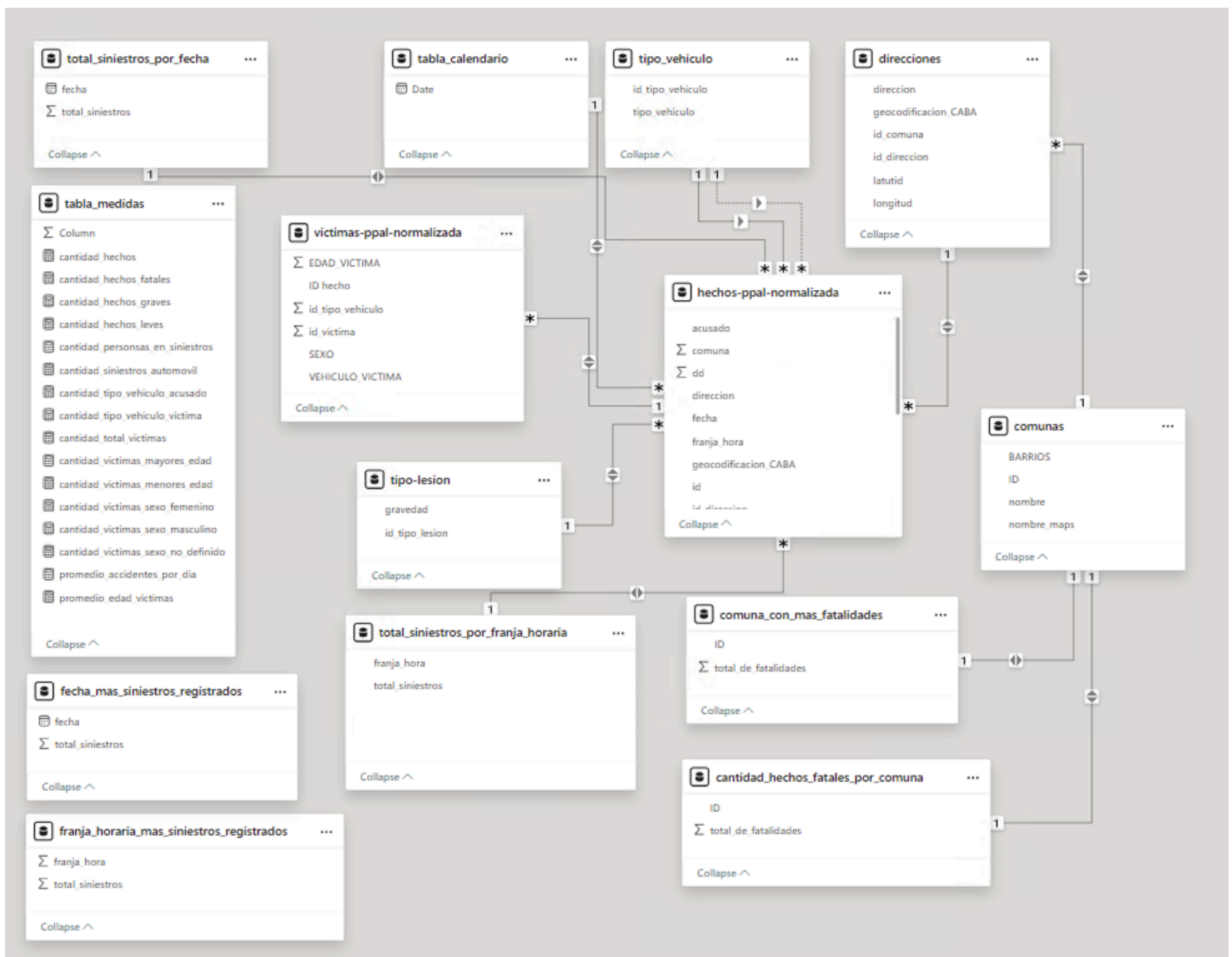
- **franja_horaria_mas_siniestros_registrados** =

```
// tabla que contiene la cantidad de siniestros sumados por franja
horaria
TOPN(
    1,
    total_siniestros_por_franja_horaria,
    total_siniestros_por_franja_horaria[total_siniestros]
)
```
- **total_siniestros_por_fecha** =

```
// tabla que contiene el total de siniestros agrupados por fecha
SUMMARIZE(
    'hechos-ppal-normalizada',
    'hechos-ppal-normalizada'[fecha],
    "total_siniestros",
    COUNTROWS('hechos-ppal-normalizada')
)
```
- **total_siniestros_por_franja_horaria** =

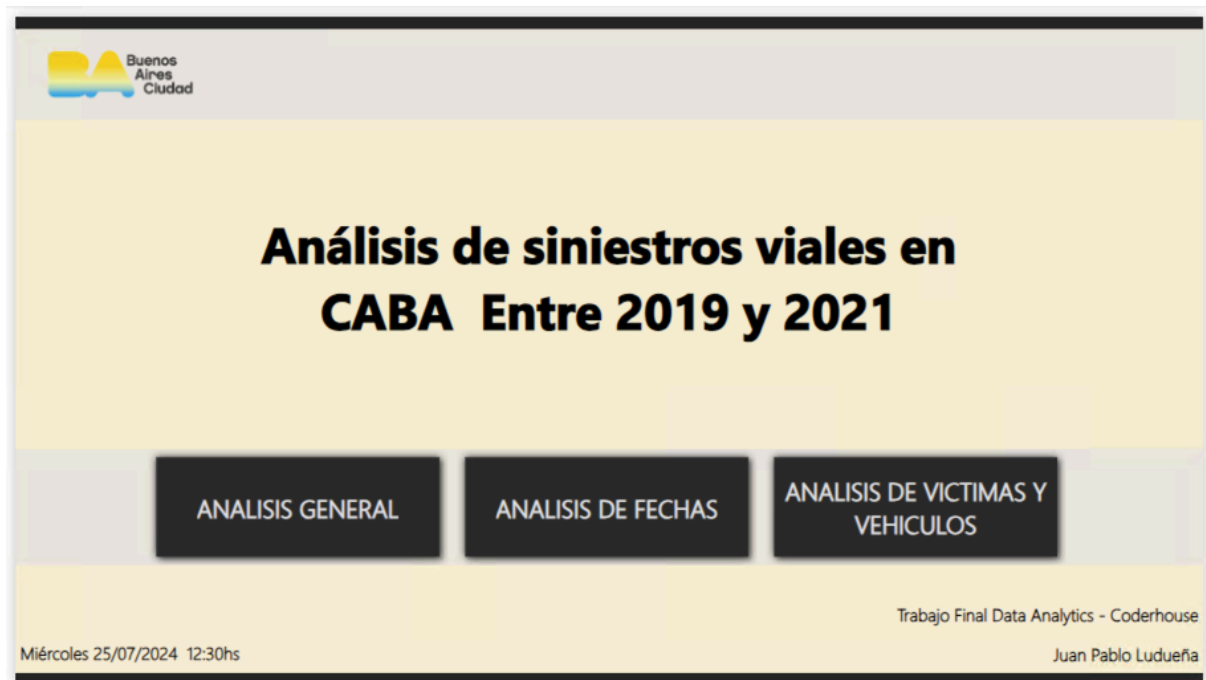
```
// tabla que contiene el total de siniestro agrupado por franja horaria
SUMMARIZE(
    'hechos-ppal-normalizada',
    'hechos-ppal-normalizada'[franja_hora],
    "total_siniestros",
    COUNTROWS('hechos-ppal-normalizada')
)
```

Modelo de datos en Power Bi



Pantallas/Solapas del dashboard

- **Portada**

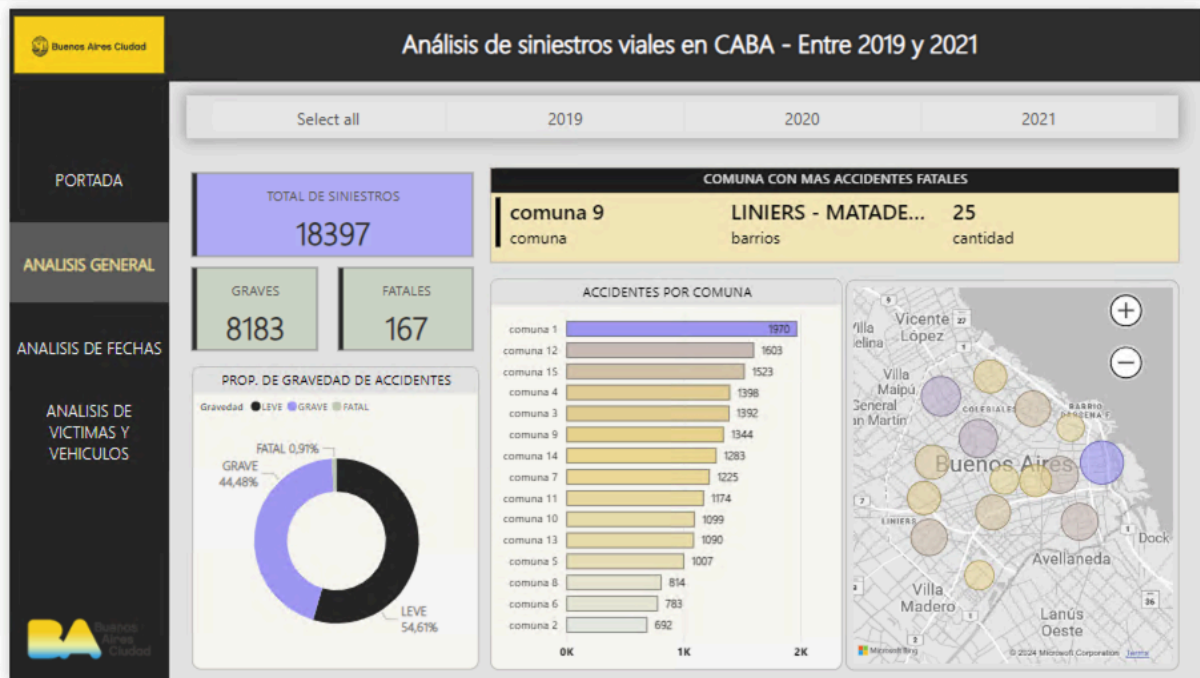


- **Análisis General**

Aquí se muestra un análisis general, detallando cantidades de siniestros y comunas en las que más ocurrencias se registraron junto con la gravedad de cada siniestro.

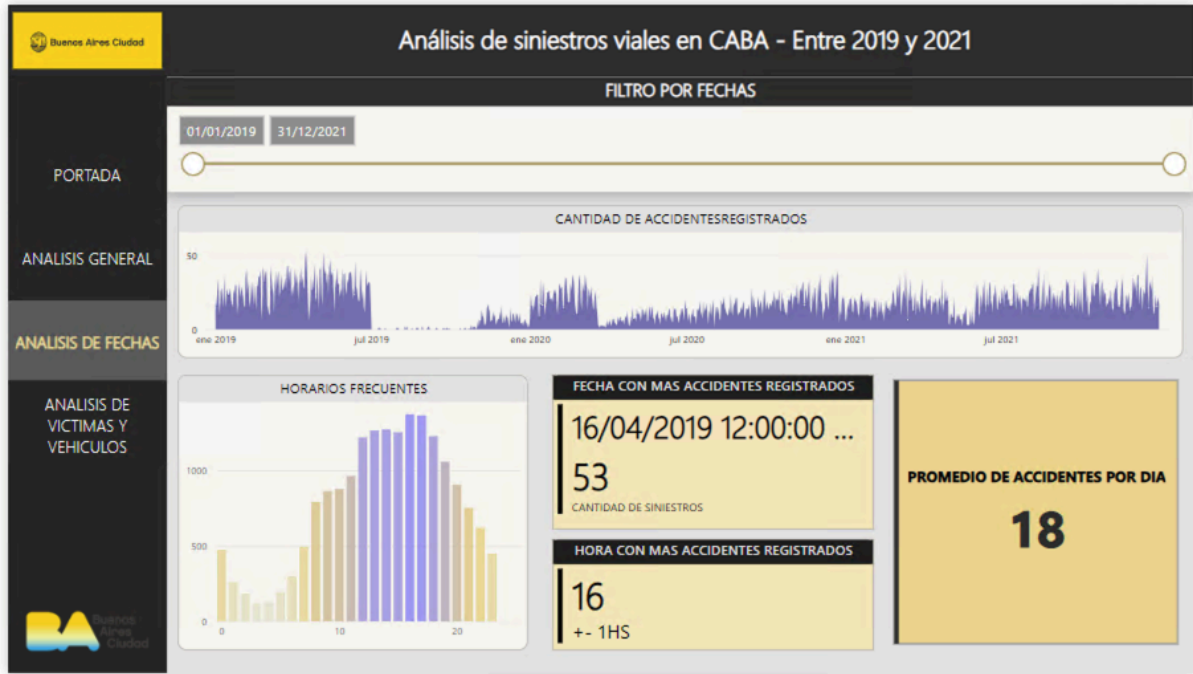
Se puede ver un indicador de cuál es la comuna que más accidentes tiene

Se puede filtrar por años si es necesario.



- Análisis de fechas

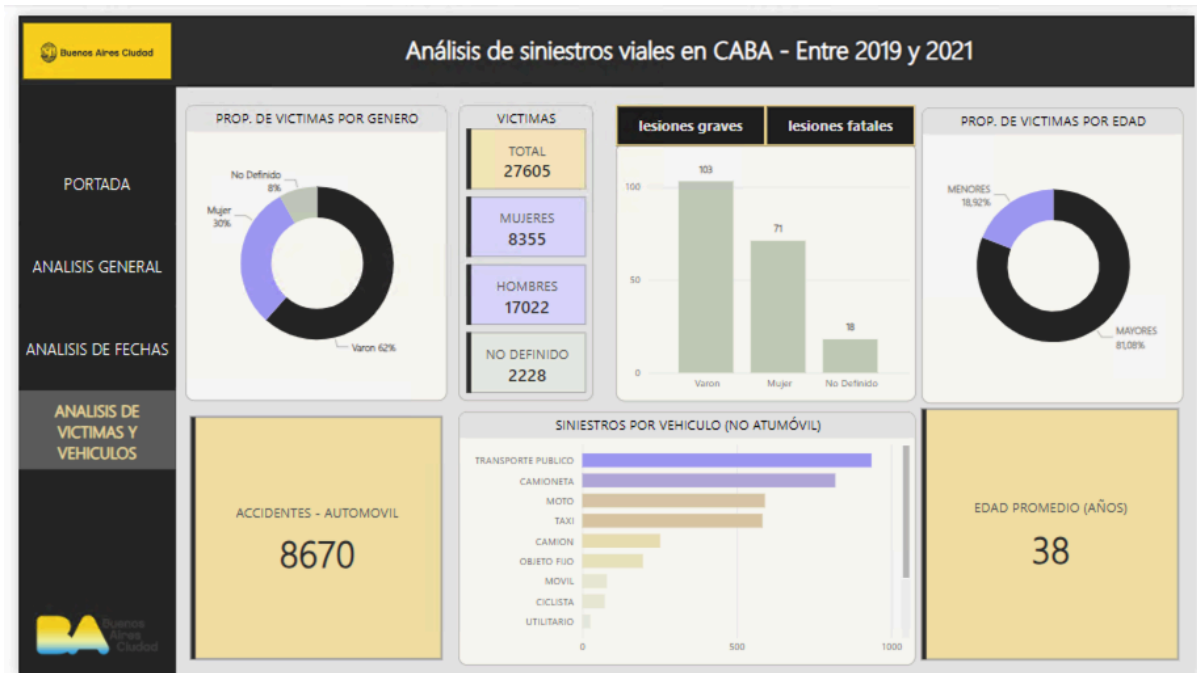
En ésta solapa encontramos el detalle de momentos del año y del día en los que ocurren la mayor cantidad de accidentes.



- **Análisis de víctimas y vehículos**

Aquí encontramos detalle de las personas que fueron víctimas de los accidentes registrados. Podemos diferenciar por sexo y edad.

Además se pueden ver los tipos de vehículos que participaron en los siniestros.



Futuras líneas

Para complementar el análisis presentado y tener un estudio completo de los siniestros en CABA sería deseable que se pueda contar con más información relacionada a los mismos como por ej:

- hechos de siniestros actualizados al corriente año
- investigar el motivo por el cual se produce el siniestro y englobar en tipos/subtipos de motivos para un mejor análisis
- saber si las personas involucradas en los siniestros cumplieron con las normas de tránsito correspondiente al tipo de vehículo que conducía
- tener la información de la VTV de los vehículos involucrados