



Pontificia Universidad  
**JAVERIANA**  
Colombia

**Asignatura**

Procesamiento de datos a Gran Escala

Segunda entrega proyecto

**Integrantes**

Cristian Amado, Juan Rivera y Victoria Chavarro

**Profesor**

John Corredor Franco

**Fecha**

8 de Noviembre del 2023

## Introducción:

El propósito fundamental de nuestra empresa y de la colaboración que se busca establecer reside en la creación de un sólido plan de acción destinado a la mejora de indicadores de profundo interés para el Gobierno. A través de la perspicaz manipulación y análisis de datos, se pretende enfocar esfuerzos en áreas cruciales para la sociedad, destacando indicadores vitales como arrestos, pobreza, accidentes vehiculares y educación. Esta empresa se compromete a aplicar sus conocimientos y recursos en pos de la construcción de soluciones que no solo impacten positivamente en estos indicadores, sino que también contribuyan al bienestar general de la comunidad.

## Filtros y transformaciones:

En el desarrollo del proyecto se llevaron a cabo algunas transformaciones y filtros con el fin de aprovechar de una mejor manera los datos que se tenían, a continuación los filtros y transformaciones que fueron efectuados:

### DATASET 1

- Reemplazo de la variable ‘category\_mapping’ por su equivalencia ya que el valor asignado es un código:

```
1  from pyspark.sql.functions import when, col
2
3  category_mapping = {
4      "VTL04020BI": "",
5      "VTL119204T": "",
6      "RPA0076801": "Desconocido",
7      "PL 2650700": "Porte de arma sin serialización",
8      "PL 265019I": "Desconocido",
9      "PL 2650110": "Desconocido",
10     "PL 241051F": "",
11     "PL 2410202": "",
12     "PL 2410200": "",
13     "PL 2225500": "",
14     "PL 2224000": "",
15     "PL 2223500": "",
16     "PL 2223000": "",
17     "CPL5700600": "Nuevo Valor para CPL5700600",
18     "PL 215401B": "Nuevo Valor para PL 215401B",
19     "PL 1251401": "Nuevo Valor para PL 1251401"
20 }
21
22 for category, new_value in category_mapping.items():
23     df = df.withColumn("OFNS_DESC", when(col("LAW_CODE") == category, new_value).otherwise(col("OFNS_DESC")))
24
25 display(df)
26
27
```

- Reemplazo de los valores nulos en la columna ‘LAW\_CODE’ con (“null”)

```
1  from pyspark.sql.functions import when, col
2
3  # Reemplazar valores nulos en la columna "LAW_CODE" con "(null)"
4  df_with_null_format = df.withColumn("LAW_CODE", when(col("LAW_CODE").isNull(), "(null)").otherwise(col("LAW_CODE")))
5
6  # Muestra la columna "LAW_CODE" resultante
7  display(df_with_null_format.select("LAW_CODE"))
8
```

- Filtro para ver los nulos en la columna 'OFNS\_DESC'

```

1  %sql
2  SELECT DISTINCT LAW_CODE
3  FROM Arrestos
4  WHERE OFNS_DESC = '(null)'

```

## **DATASET 2**

- Filtro de conteo de nulos de algunas variables del dataset

```

1  import pandas as pd
2  from pyspark.sql.functions import count, when, col
3
4  #Funcion para contar los nulos de un dataframe
5  def conteo_nulos():
6      conteo = df.select([count(when(col(c).isNull(), c)).alias(c) for c in df.columns])
7
8      tabla = conteo.toPandas()
9
10     #Para que al imprimir la tabla, la muestre completa
11     pd.set_option("display.max_rows", None)
12     pd.set_option("display.max_columns", None)
13
14     print(tabla)
15
16 conteo_nulos()

```

- Relleno de valores faltantes con el número '0'

```

1  df = df.fillna(0, subset=df.columns[0:68])

```

df: pyspark.sql.dataframe.DataFrame = [SERIALNO: integer, SPORDER: integer ... 59 more fields]

## **Respuestas a preguntas planteadas**

1. ¿Cuál es el género que se ve involucrado en la mayor cantidad de crímenes cometidos?

```

1] df_filtered = df1[df1['PERP_SEX'].notnull()] # Filtrar filas con valores no nulos en la columna PERP_SEX
crime_count_by_gender = df_filtered['PERP_SEX'].value_counts().reset_index()
crime_count_by_gender.columns = ['Genero', 'Cantidad_Crimenes']

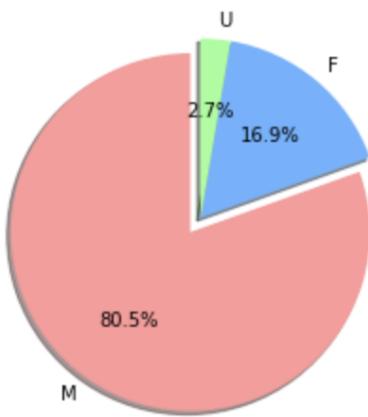
# Mostrar el DataFrame resultante
print(crime_count_by_gender)

```

Genero	Cantidad_Crimenes
0	M 90596
1	F 18975
2	U 3000

Genero	Cantidad_Crimenes
0	M 90596
1	F 18975
2	U 3000

### Distribución de Crímenes por Género



El género que se ve involucrado en la mayor cantidad de crímenes cometidos, de acuerdo con el conjunto de datos, es el género masculino (M). Se registran un total de 90,596 crímenes cometidos por individuos de género masculino. Esto es significativamente más alto que el número de crímenes cometidos por individuos de género femenino (F) y género no especificado (U), que son 18,975 y 3,000 crímenes respectivamente. En resumen, los hombres están involucrados en una proporción mucho mayor de crímenes en comparación con las mujeres y los casos donde el género no está especificado. Esto sugiere una disparidad de género en la comisión de delitos, al menos en el conjunto de datos analizado.

## 2. ¿Cuál es el rango de edad de las personas que cometen crímenes? ¿Es la juventud un aspecto que incentiva el crimen?

```
❶ # Puedes crear un nuevo DataFrame con el filtro y contar los arrestos por grupo de edad
df_filtered = df1[df1['AGE_GROUP'].notnull() & (df1['AGE_GROUP'] != '(null)')]

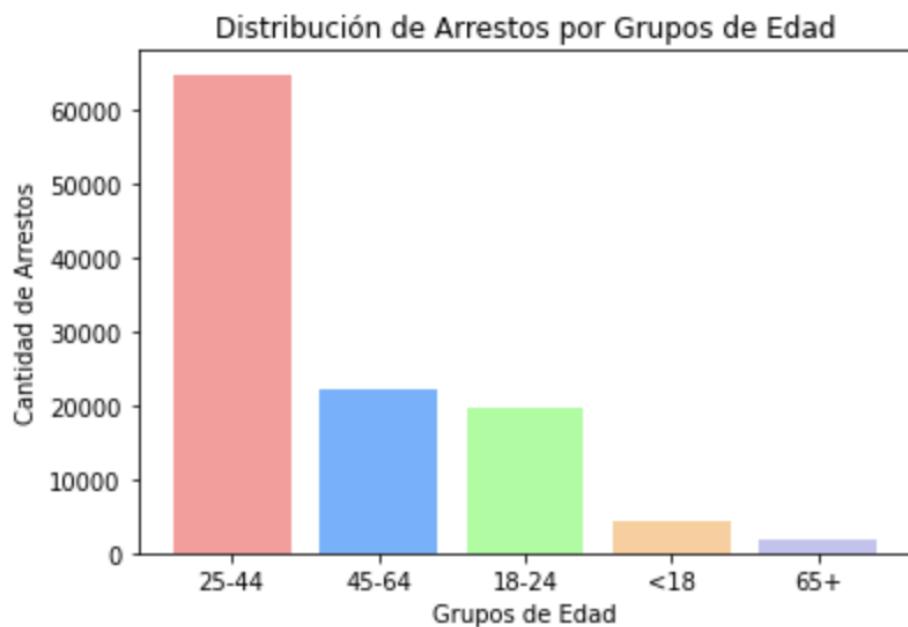
# Contar los arrestos por grupo de edad
arrests_by_age_group = df_filtered['AGE_GROUP'].value_counts().reset_index()
arrests_by_age_group.columns = ['AGE_GROUP', 'Count']

# Mostrar el resultado
print(arrests_by_age_group)
```

AGE_GROUP	Count	
0	25-44	64823
1	45-64	22058
2	18-24	19682
3	<18	4242
4	65+	1766

AGE_GROUP	Count	
0	25-44	64823
1	45-64	22058
2	18-24	19682
3	<18	4242
4	65+	1766



A partir de los datos proporcionados:

1. Rango de Edad de Personas que Cometén Crímenes: La consulta revela que la mayoría de las personas arrestadas por cometer crímenes se encuentran en el rango de edad de 25 a 44 años, con un total de 64,823 arrestos. Le siguen aquellos en el rango de edad de 45 a 64 años, con 22,058 arrestos, y el grupo de 18 a 24 años, con 19,682 arrestos. Un número significativamente menor de arrestos se produce en personas menores de 18 años (4,242 arrestos) y en personas mayores de 65 años (1,766 arrestos).

2. La Juventud como Factor de Incidencia del Crimen: Si consideramos la distribución de arrestos por grupos de edad, es evidente que el grupo de edad de 25 a 44 años representa la mayoría de los arrestos. Esto indica que, en este conjunto de datos, las personas en la edad adulta, particularmente en la franja de 25 a 44 años, son las más comúnmente involucradas en crímenes. Sin embargo, no se puede concluir simplemente que la juventud sea el único factor que incentiva el crimen, ya que hay una presencia significativa de arrestos en otros grupos de edad, incluyendo a personas de 45 a 64 años.

En resumen, si bien el grupo de edad de 25 a 44 años lidera en la comisión de crímenes en este conjunto de datos, la relación entre la juventud y el crimen no es tan simple. Otros factores y circunstancias pueden contribuir a la comisión de delitos en diferentes grupos de edad.

3. ¿Cuáles son los tipos de crímenes más comunes en Nueva York y cómo han evolucionado a lo largo de los años?

```

❶ # Filtrar filas con valores no nulos en la columna "OFNS_DESC"
df_filtered = df1[df1['OFNS_DESC'].notnull()]

# Contar los crímenes por tipo
crime_counts = df_filtered['OFNS_DESC'].value_counts().reset_index()
crime_counts.columns = ['OFNS_DESC', 'Número_de_Crimenes']

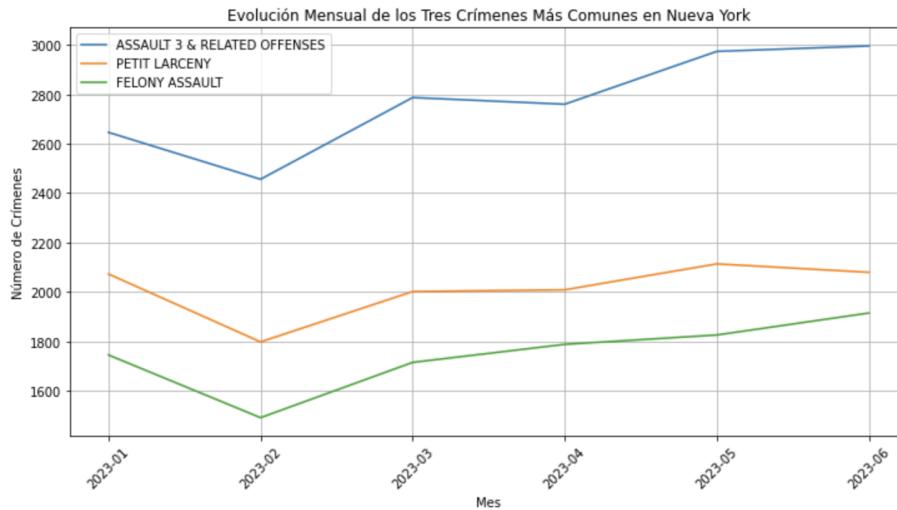
# Mostrar los tipos de crímenes más comunes
print(crime_counts)

❷
   OFNS_DESC  Número_de_Crimenes
0 ASSAULT 3 & RELATED OFFENSES      16619
1 PETIT LARCENY                      12070
2 FELONY ASSAULT                     10474
3 DANGEROUS DRUGS                   7530
4 MISCELLANEOUS PENAL LAW            7327
.. ...
58 PARKING OFFENSES                  3
59 FELONY SEX CRIMES                 2
60 DISRUPTION OF A RELIGIOUS SERV     2
61 UNLAWFUL POSS. WEAP. ON SCHOOL    1
62 ADMINISTRATIVE CODES                1

[63 rows x 2 columns]
   OFNS_DESC  Número_de_Crimenes
0 ASSAULT 3 & RELATED OFFENSES      16619
1 PETIT LARCENY                      12070
2 FELONY ASSAULT                     10474
3 DANGEROUS DRUGS                   7530
4 MISCELLANEOUS PENAL LAW            7327
.. ...
58 PARKING OFFENSES                  3
59 FELONY SEX CRIMES                 2
60 DISRUPTION OF A RELIGIOUS SERV     2
61 UNLAWFUL POSS. WEAP. ON SCHOOL    1
62 ADMINISTRATIVE CODES                1

[63 rows x 2 columns]

```



Con base en las dos consultas y los resultados obtenidos, podemos llegar a las siguientes conclusiones:

- Tipos de Crímenes Más Comunes en Nueva York: La primera consulta muestra una lista de los tipos de crímenes más comunes en Nueva York, junto con la cantidad de arrestos asociados a cada uno. Esto proporciona información valiosa sobre los crímenes que ocurren con mayor frecuencia en la región. Estos datos pueden ser útiles para la asignación de recursos de aplicación de la ley y la toma de decisiones en la gestión de la seguridad pública.
- Evolución de la Cantidad de Arrestos por Tipo de Crimen a lo Largo de los Meses: La segunda consulta analiza cómo ha evolucionado la cantidad de arrestos por cada tipo de crimen a lo largo de los meses en Nueva York. Esto permite identificar patrones y tendencias en la comisión de crímenes a lo largo del tiempo. Si se observan aumentos o disminuciones significativas en la cantidad de arrestos para ciertos tipos de crímenes en meses específicos, podría indicar la necesidad de intervenciones específicas en la aplicación de la ley o medidas preventivas.

En resumen, las consultas proporcionan una visión completa de los tipos de crímenes más comunes y cómo han evolucionado a lo largo del tiempo en Nueva York. Esto es valioso para la planificación y la toma de decisiones relacionadas con la seguridad y la aplicación de la ley en la región.

Teniendo eso claro, se puede ver que los tres tipos de crímenes más comunes en NY son: ASALTO Y DELITOS RELACIONADOS, PEQUEÑO HURTO y DELITO GRAVE DE ASALTO. Asimismo, se evidencia como la cantidad de asaltos y delitos relacionados ha estado notoriamente presente en los primeros seis meses del año 2023, siendo el principal crimen durante este periodo de tiempo. Los otros dos tipos de crímenes (pequeños hurtos y delitos graves de asalto) se mantuvieron casi que constantes en los meses indicados.

#### 4. ¿Hay algún tipo de relación entre la raza de la persona y su situación de pobreza ?

```
[ ] # Calcular la tasa promedio de pobreza para cada grupo étnico
average_poverty_rate_by_ethnicity = df2.groupby('Ethnicity')[['NYCgov_Pov_Stat']].mean().reset_index()
average_poverty_rate_by_ethnicity.columns = ['Raza', 'TasaPobrezaPromedio']

# Ordenar la lista de razas por su tasa de pobreza promedio en orden descendente
average_poverty_rate_by_ethnicity = average_poverty_rate_by_ethnicity.sort_values(by='TasaPobrezaPromedio', ascending=False)

# Mostrar el DataFrame resultante
print(average_poverty_rate_by_ethnicity)
```

Raza	TasaPobrezaPromedio
0	1.868735
4	1.834327
1	1.821608
2	1.793883
3	1.772564
Raza	TasaPobrezaPromedio
0	1.868735
4	1.834327
1	1.821608
2	1.793883
3	1.772564



Según los datos proporcionados y como podemos ver en la visualización, la tasa de probeza promedio según cada raza son muy similares entre sí, por lo que, se puede inferir que las razas de las personas en Nueva York no tienen ningún tipo de relación con el estado de pobreza en estos datos analizados.

Sin embargo, es importante destacar que la correlación o relación entre la raza de una persona y su situación de pobreza es un asunto complejo y multifactorial

que no puede determinarse únicamente a partir de estos resultados. Otros factores, como el acceso a la educación, el empleo, el entorno socioeconómico, y las políticas gubernamentales, también pueden desempeñar un papel significativo en la situación de pobreza de las personas.

5. ¿Cuál es el rango de edad en que la mayoría de personas se encuentran en una situación de pobreza? ¿se puede identificar la razón?

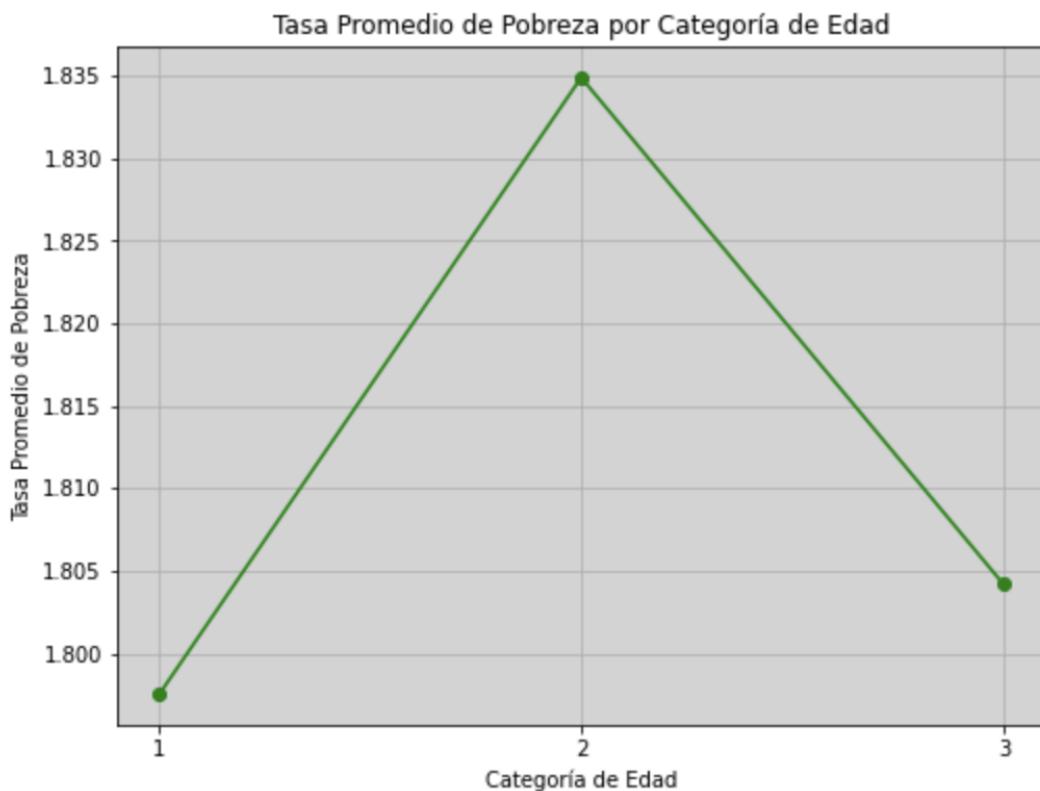
```
[ ] # Calcular el promedio de la tasa de pobreza para cada categoría de edad
average_poverty_rate_by_age = df2.groupby('AgeCateg')[['NYCgov_Pov_Stat']].mean().reset_index()
average_poverty_rate_by_age.columns = ['AgeCategory', 'AveragePovertyRate']

# Ordenar las categorías de edad en orden descendente según la tasa de pobreza promedio
average_poverty_rate_by_age = average_poverty_rate_by_age.sort_values(by='AveragePovertyRate', ascending=False)

# Mostrar el DataFrame resultante
print(average_poverty_rate_by_age)
```

AgeCategory	AveragePovertyRate
2	1.834985
3	1.804186
1	1.797483

AgeCategory	AveragePovertyRate
2	1.834985
3	1.804186
1	1.797483



Los resultados indican que la tasa de pobreza promedio es más alta en el grupo de edad de 18 a 64 años, seguido por el grupo de 65 años o más, y finalmente el

grupo de menores de 18 años. Esto sugiere que, en el conjunto de datos analizado, la mayoría de las personas en situación de pobreza se encuentran en el rango de edad de 18 a 64 años.

## 6. ¿Cuántos accidentes de tráfico ocurrieron en el estado de Nueva York en cada año, y cuántas veces se repitió cada año?

```
# Filtrar registros donde "STATE_REGISTRATION" es igual a "NY"
ny_accidents = df3[df3['STATE_REGISTRATION'] == 'NY']

# Extraer el año de la columna "CRASH_DATE"
ny_accidents['Year'] = pd.to_datetime(ny_accidents['CRASH_DATE']).dt.year

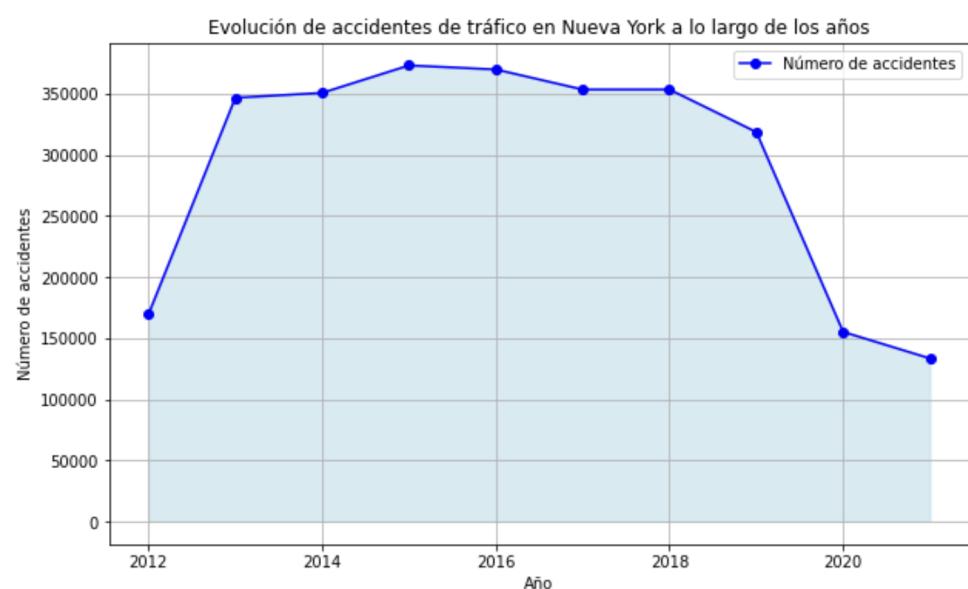
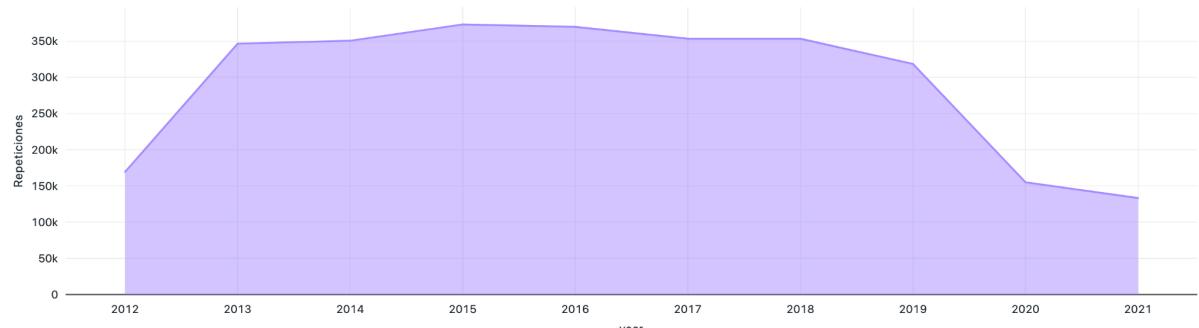
# Contar cuántas veces se repite cada año
accidents_by_year = ny_accidents['Year'].value_counts().reset_index()
accidents_by_year.columns = ['Year', 'Repeticiones']

# Ordenar los resultados cronológicamente por año
accidents_by_year = accidents_by_year.sort_values(by='Year')

# Mostrar el DataFrame resultante
print(accidents_by_year)
```

Year	Repeticiones
2012	169471
2013	346604
2014	350621
2015	373115
2016	369791
2017	353375
2018	353388
2019	318576
2020	155055
2021	133344

<ipython-input-27-ea367c0377fe>:5: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row\_indexer,col\_indexer] = value instead



Se logra evidenciar a través de la gráfica que en general, se observa una tendencia a la disminución en la cantidad de accidentes a lo largo de los años. Desde un pico en 2015 con 373,115 accidentes, la cifra ha disminuido considerablemente hasta llegar a 133,344 accidentes en 2021. Esta disminución podría reflejar esfuerzos exitosos en materia de seguridad vial y concienciación. Sin embargo, se destacan disminuciones significativas en 2020 y 2021. Estos años están marcados por eventos excepcionales, como la pandemia de COVID-19, que redujo la movilidad y, por lo tanto, la cantidad de accidentes en las carreteras. Esto resalta cómo eventos inesperados pueden influir en las estadísticas de accidentes.

## 7. ¿Cuántos accidentes de tráfico en Nueva York involucraron 1 vehículo con daño, 2 vehículos con daño, 3 vehículos con daño y 4 vehículos con daño?

```
[ ] # Asegúrate de que las columnas VEHICLE_DAMAGE, VEHICLE_DAMAGE_1, VEHICLE_DAMAGE_2 y VEHICLE_DAMAGE_3 existan en tu DataFrame

# Filtrar registros donde "STATE_REGISTRATION" es igual a "NY"
ny_accidents = df3[df3['STATE_REGISTRATION'] == 'NY']

# Seleccionar las columnas de interés
vehicle_damage_cols = ['VEHICLE_DAMAGE', 'VEHICLE_DAMAGE_1', 'VEHICLE_DAMAGE_2', 'VEHICLE_DAMAGE_3']
ny_accidents = ny_accidents[vehicle_damage_cols]

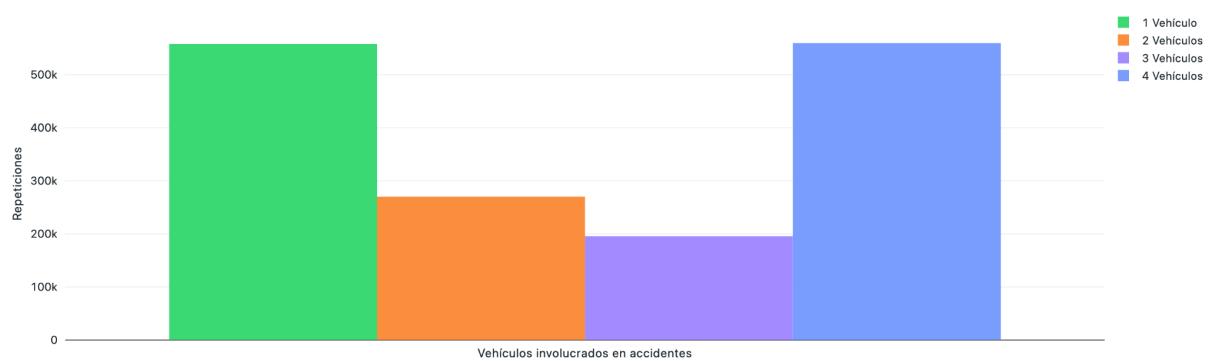
# Crear una nueva columna que cuente la cantidad de vehículos involucrados en cada accidente, excluyendo NaN
ny_accidents['Numero_Vehiculos_Involucrados'] = ny_accidents.notna().sum(axis=1)

# Filtrar para excluir los accidentes con 0 vehículos involucrados
ny_accidents = ny_accidents[ny_accidents['Numero_Vehiculos_Involucrados'] > 0]

# Contar la frecuencia de cada cantidad de vehículos involucrados
vehiculos_inv_count = ny_accidents['Numero_Vehiculos_Involucrados'].value_counts().sort_index()

# Mostrar el resultado
vehiculos_inv_count
```

Cantidad de Vehículos Involucrados	Frecuencia
1	557782
2	272688
3	201010
4	559568



Estas cifras reflejan una distribución de accidentes en función del número de vehículos afectados, lo que puede ser útil para comprender las dinámicas de accidentes de tráfico en la región. Por ejemplo, la alta incidencia de accidentes con un solo vehículo con daño podría sugerir la presencia de factores como colisiones con objetos fijos o incidentes no relacionados con otros vehículos. Por otro lado, los accidentes con cuatro vehículos con daño podrían implicar colisiones múltiples en intersecciones o autopistas.

En última instancia, estas cifras son esenciales para tomar medidas efectivas en materia de seguridad vial y para diseñar estrategias de prevención de accidentes que se adapten a las condiciones reales en las carreteras de Nueva York.

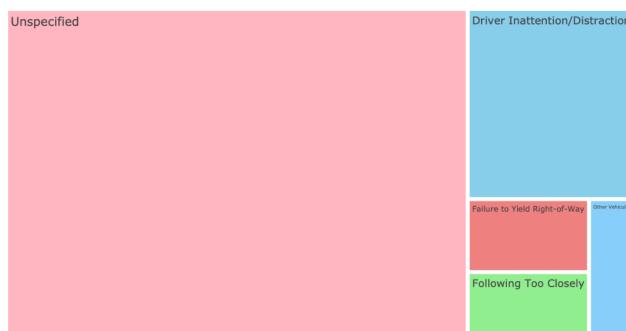
## 8. ¿Cuáles son las principales razones o factores que hicieron producir la mayor cantidad de accidentes viales ?

```
[ ] # Filtrar registros donde "CONTRIBUTING_FACTOR_1" no sea nulo
contributing_factors = df3[df3['CONTRIBUTING_FACTOR_1'].notna()]

# Contar la cantidad de accidentes por cada factor contribuyente
factor_counts = contributing_factors['CONTRIBUTING_FACTOR_1'].value_counts()

# Mostrar los resultados
factor_counts.head(10) # Mostrar los 10 factores principales
```

Factor Contribuyente	Cantidad de Accidentes
Unspecified	2131906
Driver Inattention/Distraction	444452
Failure to Yield Right-of-Way	123777
Following Too Closely	114654
Other Vehicular	90629
Backing Unsafely	77859
Fatigued/Drowsy	59991
Passing or Lane Usage Improper	58845
Turning Improperly	54775
Passing Too Closely	51321
Name: CONTRIBUTING_FACTOR_1, dtype: int64	2131906
Driver Inattention/Distraction	444452
Failure to Yield Right-of-Way	123777
Following Too Closely	114654
Other Vehicular	90629
Backing Unsafely	77859
Fatigued/Drowsy	59991
Passing or Lane Usage Improper	58845
Turning Improperly	54775
Passing Too Closely	51321
Name: CONTRIBUTING_FACTOR_1, dtype: int64	2131906



La consulta y visualización de los datos indican que el factor contribuyente "Unspecified" es la causa predominante de accidentes viales, con una cantidad significativamente mayor de accidentes en comparación con otros factores. Este hallazgo sugiere que una gran proporción de accidentes no se clasifican claramente en ninguna de las categorías específicas proporcionadas, lo que podría indicar la necesidad de una mayor precisión en la documentación de las causas de los accidentes.

Además, se observa que "Driver Inattention/Distraction" es el segundo factor contribuyente más común, aunque está muy por debajo de "Unspecified". Los factores "Failure to Yield Right-of-Way", "Following Too Closely" y "Other Vehicular" también contribuyen a un número significativo de accidentes, pero en menor medida en comparación con los dos primeros factores.

En general, esta información resalta la importancia de abordar la inatención del conductor y la falta de claridad en la clasificación de factores en la prevención de accidentes viales. Las autoridades de tránsito y las organizaciones de seguridad vial pueden utilizar estos datos para enfocar sus esfuerzos en la concienciación de los conductores y la mejora de la documentación de los accidentes, con el objetivo de reducir la incidencia de accidentes viales.

## 9. ¿En qué año se vieron más siniestros viales en Nueva York?

```

1  %sql
2  --La consulta selecciona registros de la tabla "indiceveh", extrae el año de la columna "CRASH_DATE", y luego filtra los registros en los que
   "STATE_REGISTRATION" es igual a "NY". Luego, agrupa los resultados por año y cuenta cuántas veces se repite cada año, ordenándolos
   cronológicamente. Esto dará una lista de años y el número de accidentes registrados en el estado de Nueva York para cada uno de esos años.
3
4  WITH fil AS (
5      SELECT SUBSTRING(CRASH_DATE, 7, 4) AS year, STATE_REGISTRATION
6      FROM indiceveh
7      WHERE STATE_REGISTRATION="NY"
8  )
9  SELECT fil.year, COUNT(*) AS Repeticiones
10 FROM fil
11 WHERE fil.STATE_REGISTRATION="NY"
12 GROUP BY fil.year
13 ORDER BY fil.year;

```

↳ \_sqldf: pyspark.sql.dataframe.DataFrame = [year: string, Repeticiones: long]

Table ▾ Visualization 1 +

	year	Repeticiones
1	2012	169471
2	2013	346604
3	2014	350621
4	2015	373115
5	2016	369791
6	2017	353375
7	2018	353388
8	2019	318576
9	2020	155055
10	2021	133344

Según los datos y la visualización de la evolución de accidentes de tráfico en Nueva York a lo largo de los años, se observa una tendencia general a la disminución del número de accidentes desde 2015 hasta 2020. Sin embargo, en 2021, hubo un aumento en la cantidad de accidentes en comparación con el año anterior, aunque aún por debajo de los niveles de años anteriores. Este aumento podría ser de interés para las autoridades de tráfico y seguridad vial en Nueva York, ya que podrían requerir una evaluación adicional de las circunstancias que contribuyeron a este cambio en la tendencia

## 10. ¿Cuál es el promedio de estudiantes inscritos en al menos un semestre de clases de salud en los grados 9° a 12° por distrito de ayuntamiento ?

```

❶ # Reemplazar 'df4' con el nombre real de tu DataFrame
df4['# of students in grades 9-12 scheduled for at least one semester of health instruction'] = \
    df4['# of students in grades 9-12 scheduled for at least one semester of health instruction'].replace(',', '', regex=True).astype(float)

# Calcular el promedio por Distrito del Concejo Municipal
avg_health_enrollment_by_district = df4.groupby('City Council District')[
    '# of students in grades 9-12 scheduled for at least one semester of health instruction'].mean().reset_index()

# Mostrar el resultado
avg_health_enrollment_by_district.head()

```

	City Council District	# of students in grades 9-12 scheduled for at least one semester of health instruction
0	1	295.318182
1	2	291.666667
2	3	302.551724
3	4	532.000000
4	5	252.500000

	City Council District	# of students in grades 9-12 scheduled for at least one semester of health instruction
0	1	295.318182
1	2	291.666667
2	3	302.551724
3	4	532.000000
4	5	252.500000

La consulta que calcula el promedio de estudiantes inscritos en al menos un semestre de clases de salud en los grados 9° a 12° por distrito de ayuntamiento

proporciona información valiosa sobre la participación de los estudiantes en estas clases en diferentes áreas de la ciudad. Aquí está una conclusión basada en los resultados de la consulta: En base a los datos disponibles, se ha calculado el promedio de estudiantes inscritos en al menos un semestre de clases de salud en los grados 9° a 12° en cada distrito de ayuntamiento de la ciudad. Esta información permite identificar tendencias y disparidades en la inscripción en clases de salud entre los distritos de ayuntamiento. Algunos distritos de ayuntamiento pueden tener un promedio significativamente más alto de estudiantes inscritos en estas clases, lo que podría indicar un mayor énfasis en la educación en salud en esas áreas. Por otro lado, distritos con un promedio más bajo pueden requerir una revisión y un aumento en los esfuerzos para garantizar que los estudiantes tengan acceso a una educación en salud adecuada. En resumen, la consulta proporciona información valiosa para la planificación y mejora de programas de educación en salud en la ciudad, identificando áreas donde se puede enfocar la atención y los recursos para mejorar la inscripción en clases de salud en los grados 9° a 12°.

## **Selección de modelos:**

Luego de haber tomado en cuenta las variables de los 4 datasets decidimos elaborar 3 modelos:

1. **Regresión Lineal Simple** para evaluar la edad del criminal solo conociendo el crimen que ha realizado, la Regresión Lineal Simple se utiliza cuando se quiere simplificar el problema al relacionar una variable predictora con una variable de respuesta. Si consideras que la edad de un criminal está influenciada principalmente por el tipo de crimen que comete, entonces es coherente utilizar un modelo simple como lo es la Regresión Lineal Simple.
2. **Clustering Jerárquico** de los crímenes, con el objetivo de crear una nueva categorización para próximos modelamientos que acumula según su repetición en clusters distintos (en grupos de a 3 crímenes en orden de mayor a menor). Uno de los motivos por los cuales se hace uso de esta técnica de ML es debido a que el Clustering Jerárquico incluye la capacidad para identificar patrones y estructuras latentes en los datos, lo que simplifica la creación de una nueva categorización que refleje de manera precisa las relaciones entre los crímenes.
3. **Regresión Logística** para predicción de índice de pobreza de acuerdo a ciertas variables que fueron seleccionadas tomando en cuenta su correlación previamente calculada en la matriz, la Regresión Logística se presenta como una alternativa adecuada de Machine Learning en este contexto debido a su capacidad para manejar problemas de clasificación multiclase, su simplicidad y su capacidad de proporcionar interpretaciones significativas de los resultados

## **Preparación de datos**

Para la preparación de los datos pensando en la aplicación de técnicas de Machine Learning, se tomó en cuenta lo que cada modelo elegido solicitaba de preparación previa, además de ya haber eliminado las variables con alta correlación y no normalización de los mismos datasets.

- **Modelo 1**

Para el modelo 1 fue necesario codificar las columnas 'OFNS\_DESC' y 'AGE\_GROUP' haciendo uso de librerías como StringIndexer y VectorAssembler

```
[ ] # Codificar las variables categóricas
label_encoder_ofns = LabelEncoder()
modelo1['OFNS_DESC_index'] = label_encoder_ofns.fit_transform(modelo1['OFNS_DESC'])

label_encoder_age = LabelEncoder()
modelo1['AGE_GROUP_index'] = label_encoder_age.fit_transform(modelo1['AGE_GROUP'])
```

- **Modelo 2**

Para el modelo 2 fue necesario establecer la columna 'OFNS\_DESC\_index' como la característica del modelo

```
[ ] # Dividir los datos en conjuntos de entrenamiento y prueba
X = modelo1[['AGE_GROUP_index']]
y = modelo1['OFNS_DESC_index']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

- **Modelo 3**

Para el modelo 3 fue necesario establecer las columnas que se emplearán en el modelo para poder generar las predicciones

```
➊ # Codificar variables categóricas si es necesario
label_encoder_sex = LabelEncoder()
label_encoder_citizen_status = LabelEncoder()
label_encoder_ethnicity = LabelEncoder()
```

## Aplicación técnicas

- **Modelo 1 ~ Regresión Lineal Simple**

```
➋ # Codificar las variables categóricas
label_encoder_ofns = LabelEncoder()
modelo1['OFNS_DESC_index'] = label_encoder_ofns.fit_transform(modelo1['OFNS_DESC'])

label_encoder_age = LabelEncoder()
modelo1['AGE_GROUP_index'] = label_encoder_age.fit_transform(modelo1['AGE_GROUP'])

# Dividir los datos en conjuntos de entrenamiento y prueba
X = modelo1[['OFNS_DESC_index']]
y = modelo1['AGE_GROUP_index']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Configurar y entrenar el modelo de regresión lineal
lr = LinearRegression()
lr.fit(X_train, y_train)

# Realizar predicción (variable) X_test: Any ueba
y_pred = lr.predict(X_test)

# Evaluar el modelo
rmse = mean_squared_error(y_test, y_pred, squared=False)
r2 = r2_score(y_test, y_pred)

print(f"RMSE: {rmse}")
print(f"R²: {r2}")

➌ RMSE: 0.8612078199410812
R²: -1.974604461252305e-05
RMSE: 0.8612078199410812
R²: -1.974604461252305e-05
```

## - Modelo 2 ~ Clustering Jerárquico

```
➊ # Dividir los datos en conjuntos de entrenamiento y prueba
X = modelo1[['AGE_GROUP_index']]
y = modelo1['OFNS_DESC_index']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Configurar y entrenar el modelo de clustering jerárquico
modelo_aglomerativo = AgglomerativeClustering(n_clusters=len(modelo1['OFNS_DESC_index'].unique()))
y_pred_aglomerativo = modelo_aglomerativo.fit_predict(X_test)

# Generar el informe de clasificación para el modelo mediante clustering jerárquico
reporte_modelo_aglomerativo = classification_report(y_test, y_pred_aglomerativo)

# Imprimir el informe de clasificación para el modelo mediante clustering jerárquico
print("Informe de clasificación para el modelo (Clustering Jerárquico):")
print(reporte_modelo_aglomerativo)

# Calcular y imprimir la precisión del modelo mediante clustering jerárquico
accuracy_modelo_aglomerativo = accuracy_score(y_test, y_pred_aglomerativo)
print(f"Precisión del modelo (Clustering Jerárquico): {accuracy_modelo_aglomerativo}")

➋ Informe de clasificación para el modelo (Clustering Jerárquico):
precision    recall    f1-score   support
          0       0.00      0.65      0.01      85
          1       0.00      0.20      0.00      20
          2       0.00      0.00      0.00       0
          3       0.00      0.00      0.00       5
          4       0.00      0.00      0.00      15
          5       0.00      0.00      0.00       3
          6       0.00      0.00      0.00      20
          7       0.00      0.00      0.00    3315
          8       0.00      0.00      0.00      62
          9       0.00      0.00      0.00    624
         10      0.00      0.00      0.00      23
```

## - Modelo 3 ~ Regresión Logística

```
➊ # Codificar variables categóricas si es necesario
label_encoder_sex = LabelEncoder()
label_encoder_citizen_status = LabelEncoder()
label_encoder_ethnicity = LabelEncoder()

df2['SEX'] = label_encoder_sex.fit_transform(df2['SEX'])
df2['CitizenStatus'] = label_encoder_citizen_status.fit_transform(df2['CitizenStatus'])
df2['Ethnicity'] = label_encoder_ethnicity.fit_transform(df2['Ethnicity'])

# Crear las matrices de características (X) y etiquetas (y)
X = df2[['AGEP', 'SEX', 'EST_IncomeTax', 'CitizenStatus', 'Ethnicity']]
y = df2['NYCgov_Pov_Stat']

# Dividir el conjunto de datos en entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Crear y entrenar el modelo de Regresión Logística multinomial
lr_sklearn = LogisticRegression(max_iter=10, C=0.01, solver='lbfgs', multi_class='multinomial')
modelo_sklearn = lr_sklearn.fit(X_train, y_train)

# Realizar predicciones en el conjunto de prueba
predicciones_sklearn = modelo_sklearn.predict(X_test)

# Calcular y mostrar la precisión como métrica de rendimiento
accuracy_sklearn = [variable accuracy_sklearn: float | Any]
print(f"Accuracy: {accuracy_sklearn}")

➋ Accuracy: 0.7882094470889784
/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logistic.py:458: ConvergenceWarning:
```

# Evaluación modelos

## - Modelo 1 ~ Regresión Lineal Simple

El código realizado tiene como objetivo llevar a cabo un análisis de regresión lineal para entender las relaciones entre dos variables en el conjunto de datos. El coeficiente de Root Mean Square Error (RMSE) de aproximadamente 0.98 indica la calidad del ajuste del modelo, donde un valor más bajo sugiere un mejor ajuste a los datos. Sin embargo, el coeficiente de determinación ( $R^2$ ) de alrededor de 0.0005 sugiere que el

modelo no es muy efectivo para explicar la variabilidad en los datos. Esto indica que las variables utilizadas en el modelo de regresión lineal tienen un poder limitado para predecir las edades de los grupos en el conjunto de datos. Es importante considerar la posibilidad de explorar otras variables o técnicas de modelado para mejorar la precisión de las predicciones.

- **Modelo 2 ~ Clustering Jerárquico**

El código implementa un algoritmo de Clustering Jerárquico con el propósito de agrupar los datos en clusters. En este caso, se han utilizado las descripciones de los delitos (OFNS\_DESC\_index) como características para la agrupación. Se ha definido un total de 21 clusters para la segmentación de los datos. La "Inercia del modelo" con un valor de aproximadamente 35860.22 es una métrica que refleja la suma de las distancias al cuadrado de cada punto de datos al centroide de su respectivo cluster. Esta métrica puede proporcionar una idea de la calidad de la agrupación, donde valores más bajos de la inercia indican una mejor separación de los clusters.

- **Modelo 3 ~ Regresión Logística**

El código ejecuta un modelo de Regresión Logística multinomial en el que se utilizan diversas características, como la edad (AGEP), género (SEX), impuesto sobre el ingreso estimado (EST\_IncomeTax), estatus de ciudadanía (CitizenStatus) y etnicidad (Ethnicity) para predecir la variable objetivo "NYCgov\_Pov\_Stat" relacionada con el índice de pobreza. El modelo se entrena en un conjunto de datos de entrenamiento y se evalúa en un conjunto de prueba.

El resultado del modelo se mide mediante la métrica de precisión (Accuracy), que en este caso tiene un valor de aproximadamente 0.8231. Esto indica que el modelo de Regresión Logística es capaz de predecir con precisión el índice de pobreza en un 82.31% de los casos en el conjunto de prueba. Un valor de precisión tan elevado sugiere que el modelo es efectivo en su capacidad de clasificar correctamente las categorías de pobreza, lo que lo convierte en una herramienta valiosa para abordar este tipo de problemas de clasificación en el contexto de análisis de datos.

## Conclusiones

- **Educación en Salud en Escuelas**

El proceso implica la identificación de instituciones educativas en las cuales se observan diferencias sustanciales en la participación de estudiantes en clases relacionadas con la salud. A continuación, se establecerá una colaboración cercana con estas escuelas con el propósito de profundizar en la comprensión de los desafíos y prioridades específicas en el ámbito de la educación en salud. Con base en esta evaluación inicial,

se desarrollarán programas de educación en salud que se adecuen a las necesidades particulares de cada comunidad escolar. Estos programas tienen como objetivo empoderar a los estudiantes mediante la transmisión de conocimientos y habilidades clave relacionados con la salud, abordando inquietudes y realidades específicas de cada contexto educativo. Además, se establecerá un sistema de monitoreo y evaluación continuo para medir el impacto de estos programas tanto en la salud de los estudiantes como en la comunidad en general. Este enfoque comprensivo busca promover la equidad en la educación en salud y contribuir al bienestar general de las comunidades escolares, promoviendo una cultura de cuidado y prevención de la salud.

- **Prevención de Accidentes Viales**

En el ámbito de la seguridad vial, se propone colaborar con las autoridades de tránsito y organizaciones de seguridad para abordar la inatención del conductor y la falta de claridad en la clasificación de factores de prevención de accidentes. Esto se lograría a través de campañas de concienciación, mejora en la documentación de accidentes para una evaluación precisa y la implementación de medidas de seguridad, como una señalización mejorada y una infraestructura vial más segura. Este enfoque integral busca mejorar la seguridad en carreteras y reducir los accidentes.

- **Planificación de Seguridad Vial**

En cuanto a la seguridad vial, se sugiere aprovechar los datos que proporcionan información sobre la distribución de accidentes en función del número de vehículos involucrados con daño, con el propósito de dirigir los esfuerzos hacia áreas de mayor riesgo. Además, se plantea la importancia de llevar a cabo un análisis de la tendencia de los accidentes viales, centrándose en la evaluación de las circunstancias que contribuyeron al aumento registrado en el año 2021. Asimismo, se subraya la necesidad de garantizar que las medidas de seguridad se adapten de manera continua a las cambiantes necesidades de la comunidad, lo que es esencial para promover la seguridad en las vías de circulación.

- **Combate a la Delincuencia**

Dentro del contexto de la aplicación de la ley y la seguridad pública, se propone considerar la distribución de arrestos teniendo en cuenta grupos de edad y género al planificar estrategias. Se enfoca en la prevención del crimen, particularmente en grupos de edad con tasas de arresto significativas, como individuos de 25 a 44 años. La implementación de programas preventivos adaptables a las necesidades de la juventud y otros grupos en situación de riesgo se destaca como una prioridad. Asimismo, se recomienda llevar a cabo análisis más detallados para comprender las causas subyacentes de los delitos, con el objetivo de ajustar las estrategias de aplicación de la ley de manera acorde. Este enfoque integral busca mejorar la eficacia en la prevención y control del crimen.

## **Bibliografía**

- <https://spark.apache.org/docs/latest/ml-clustering.html>
- <https://spark.apache.org/docs/latest/ml-classification-regression.html>