|  |  |  |
| --- | --- | --- |
| **Mapa de ruta de caso práctico - Preguntar** | | |
| **Preguntas orientativas** |  |  |
| * ¿Qué tema estás explorando? |  |  |
| * ¿Cuál es el problema que intentas resolver? |  |  |
| * ¿Qué métricas utilizarás para medir tus datos a fin de lograr | tu | objetivo? |
| * ¿Quiénes son los interesados? |  |  |
| * ¿Cuál es tu audiencia? |  |  |
| * ¿Cómo pueden ayudar tus conocimientos a que el cliente tome decisiones? |  |  |
| **Tareas clave** |  |  |
| 1. Identificar la tarea empresarial |  |  |
| 1. Determinar los interesados clave 2. Elegir un conjunto de datos 3. Establecer métricas |  |  |
| **Entregable**  Una declaración clara de la tarea empresarial que elegiste investigar | | |

**Resolución de Caso Practico: “Análisis de Accidentes en Ductos de EEUU y su relación con las Importaciones de Crudo, para el periodo 2010-2017”**

Certificación “Google Data Analyst” – Coursera

***Desarrollado por: Nicolas Gabriel Gimenez***

***Fecha: 12/05/2025***

**Escenario y Definición del Problema**

Eres un analista de datos junior que trabaja en una consultora de ingeniería mecánica con orientación al mantenimiento industrial. En este caso, la Cámara de Comercio de Transportadores de Petróleo y Gas de Estados Unidos ha solicitado un estudio sobre los eventos y accidentes que han ocurrido a lo largo de los años en activos físicos de transporte. El objetivo es cruzar la información de las importaciones de petróleo para poder determinar:

* Número total de accidentes por año y mes
* Volumen derramado o afectado
* Tipo y causa principal de los accidentes
* Volumen de crudo importado en cada período
* Países de origen del crudo importado

En este análisis se busca explorar la relación entre los accidentes en cañerías del sector energético en Estados Unidos y los volúmenes de importación de crudo a lo largo de los últimos años. El objetivo es identificar posibles patrones o correlaciones que ayuden a comprender cómo se comportan los incidentes logísticos dentro de la cadena de suministro energética y ayudar a las empresas que pertenecen a la cámara, a tomar decisiones basadas en datos.

Los interesados clave en este análisis incluyen:

* Empresas de transporte y distribución de hidrocarburos
* Compañías aseguradoras
* Comunidades cercanas a las infraestructuras afectadas
* Organismos de control

La audiencia principal son los analistas de datos, ingenieros de procesos y responsables de toma de decisiones en empresas del sector energético, así como funcionarios públicos relacionados con la infraestructura y la seguridad industrial.

El problema principal que se intenta abordar es la frecuencia y gravedad de los accidentes en cañerías, especialmente en contextos de alta demanda o grandes volúmenes de importación, lo cual podría afectar la eficiencia operativa, la seguridad ambiental y el abastecimiento energético nacional.

Los conocimientos obtenidos a través de este estudio pueden ayudar a los interesados a:

* Tomar decisiones preventivas basadas en evidencia
* Optimizar la planificación de importaciones y rutas logísticas
* Implementar medidas de mitigación en zonas de alto riesgo
* Priorizar inversiones en mantenimiento y control de calidad

|  |
| --- |
| **Mapa de ruta de caso práctico - Preparar** |
| **Preguntas orientativas**   * ¿Dónde se ubican tus datos? * ¿Cómo están organizados los datos? * ¿Hay problemas con el sesgo o la credibilidad de estos datos? [¿Tus datos son confiables, originales, integrales, actuales y citados (ROCCC)?](https://www.coursera.org/learn/data-preparation/lecture/lHirM/what-is-bad-data) * ¿Cómo estás abordando la autorización, la privacidad, la seguridad y la accesibilidad? * ¿Cómo verificaste la integridad de los datos? * ¿De qué manera te ayuda a responder tu pregunta? * ¿Existe algún problema con los datos? |
| **Leyenda**   * Descarga los datos y almacénalos adecuadamente. * Identifica cómo están organizados. * Ordena y filtra los datos. * Determina la credibilidad de los datos. |
| **Entregable**  Una descripción de todas las fuentes de datos utilizadas |

***Almacenamiento***

Los datos se ubicarán en Google Drive en la siguiente dirección:

Mi unidad\Data Analyst – Google\Coursera – 8. Proyecto Final\Datos\_Crudos

En dicha dirección, se dividió la carpeta en dos:

* Datos\_Analisis, compuesta por dos archivos .csv que son archivos base para cargar en la herramienta de procesamiento:
  + “imp\_crudo.csv”: Contine datos de las importaciones de crudo en los Estados Unidos.
  + “accidentes.csv”: Contiene información de los accidentes en oleoductos de los Estados Unidos
* Datos\_BackUp, compuesta por dos archivos .csv tal cual fueron descargados de la fuente de estos (lo único que se realizó fue cambiarles el nombre ya que ambos tenían el mismo de fuente).
  + “data.csv” -> “Importaciones de Crudo – Estados Unidos.zip”
  + “database.csv” -> “Accidentes en Cañerias de Crudo.zip”

***Fuente de los Datos***

Los datos fueron obtenidos de la plataforma Kaggle y son de uso público. Los datos son íntegros ya que ambos provienen de organismos estatales de Estados Unidos:

* En el caso de “imp\_crudo.csv”, los datos fueron recopilados por U.S Energy Information Administration. Poseen licencia “U.S. Government Works”. Para la comunidad Kaggle, tiene nota en usabilidad de 10,0.
* Por otro lado, para el caso de “accidentes.csv”, los datos fueron recopilados y publicados por el “DOT's Pipeline and Hazardous Materials Safety Administration”. Poseen licencia de dominio público y el dueño es U.S Transportation Department. Para la comunidad Kaggle, tiene nota en usabilidad de 8,2.

***Seguridad y Privacidad de los Datos***

Los conjuntos de datos analizados no contienen información personal ni identificable, por lo que no se presentan riesgos asociados a privacidad o confidencialidad.

***Organización de los Datos***

* “imp\_crudo.csv”
  + Este conjunto de datos proporciona información detallada sobre las importaciones de petróleo crudo de EE. UU. por mes para cada año desde 2009 hasta 2024. Los datos incluyen el país de origen, el puerto de entrada de EE. UU., el nombre de la compañía petrolera, el tipo de petróleo crudo y el volumen importado (en miles de barriles).
* “accidentes.csv”
  + Esta base de datos incluye un registro de cada fuga o derrame de petróleo informado a la Administración de Seguridad de Oleoductos y Materiales Peligrosos desde 2010. Estos registros incluyen la fecha y hora del incidente, el operador y el oleoducto, la causa del incidente, el tipo de líquido peligroso y la cantidad perdida, las lesiones y muertes, y los costos asociados.

***Problemas Detectados en los Datos***

Se puede observar que en el caso de “accidentes.csv”, los datos desde 2017 no están actualizados. Se investigó en los archivos oficiales de “U.S. Department of Transportation – Pipeline and Hazardous Materials Safety Administration” y se observa que, si bien los datos son existentes, no tienen un arreglo igual al que se está tratando. Para el alcance de este análisis, se analizarán los datos de importaciones y de accidentes en oleoductos para el periodo 2010-2017 y se dejará una vía abierta de investigación para actualizar esta base de datos y extender el análisis al periodo 2017-2024.

|  |
| --- |
| **Mapa de ruta de caso práctico - Procesar** |
| **Preguntas orientativas**   * ¿Qué herramientas eliges y por qué? * ¿Has garantizado la integridad de los datos? * ¿Qué pasos seguiste para garantizar que tus datos están limpios? * ¿Cómo puedes verificar que tus datos están limpios y listos para analizar? * ¿Documentaste tu proceso de limpieza para poder revisar y compartir estos resultados? |
| **Tareas clave**   1. Verifica si hay errores en los datos. 2. Elige tus herramientas. 3. Transforma los datos para que puedas trabajar con ellos eficazmente. 4. Documenta el proceso de limpieza. |
| **Entregable**  Documentación de todas las limpiezas y manipulaciones de datos |

**Procesamiento de los Datos**

Para el procesamiento de datos se utilizó el lenguaje R y el entorno RStudio. Se utilizaron los paquetes tidyverse, lubridate y ggplot2.

Se verificó la integridad de los datos revisando los tipos de variables, detectando y tratando valores nulos (NA), y realizando sumas y análisis descriptivos para validar que los datos numéricos no estuvieran mal formateados (por ejemplo, costos con texto o separadores inválidos). Se usó summary(), str() y unique() para esta tarea.

Para garantizar que los datos están limpios, se realizaron las siguientes tareas:

* Conversión de columnas a numérico cuando daban error por tipo.
* Filtrado de valores con NA en columnas clave (Operator Name, All Costs).
* Transformación de datos de formato wide a long con pivot\_longer() para análisis por tipo de costo.
* Cálculo de variables auxiliares (total\_costos, etc.).
* Filtrado por período válido: 2010–2017, justificando la decisión en función de la calidad de datos.

Además, se realizaron conteos, resúmenes estadísticos (summary()), y filtrados de prueba (filter() + nrow()) para confirmar que los datos no contienen errores estructurales. También se generaron primeros gráficos (barras por año) que mostraron resultados coherentes con lo esperado.

El proceso de limpieza y análisis se detalla a continuación.

|  |
| --- |
| **Mapa de ruta de caso práctico - Analizar** |
| **Preguntas orientativas**   * ¿Cómo deberías organizar tus datos para realizar un análisis? * ¿Tus datos tienen el formato correcto? * ¿Qué sorpresas descubriste en los datos? * ¿Qué tendencias o relaciones encontraste en los datos? * ¿Cómo te ayudarán estos conocimientos para responder a tus preguntas empresariales? |
| **Tareas clave**   1. Consolida tus datos para que sean útiles y accesibles. 2. Organiza y formatea tus datos. 3. Realiza cálculos. 4. Identifica tendencias y relaciones. |
| **Entregable**  Un resumen de tu análisis |

|  |
| --- |
| **Mapa de ruta de caso práctico - Compartir** |
| **Preguntas orientativas**   * ¿Pudiste responder estas preguntas empresariales? * ¿Qué historia cuentan tus datos? * ¿De qué manera tus hallazgos se relacionan con tu pregunta original? * ¿Cuál es tu audiencia? ¿Cuál es la mejor manera de comunicarte con ella? * ¿La visualización de datos puede ayudarte a compartir tus hallazgos? * ¿Tu representación es accesible para tu público? |
| **Tareas clave**   1. Determina la mejor manera de compartir tus hallazgos. 2. Crea visualizaciones de datos efectivas. 3. Presenta tus hallazgos. 4. Garantiza que tu trabajo sea accesible. |
| **Entregable**  Visualizaciones de respaldo y hallazgos clave |

Descripcion Analisis - Accidentes en Oleoductos EEUU e Importaciones de Crudo

Gimenez Nicolas

2025-05-12

#Analisis de Accidentes en Oleoductos de EEUU y su relacion con las Importaciones Anuales de Crudo El analisis se realizará en dos etapas. En primer lugar, se analizará el archivo “accidentes.csv”, que tiene alojados una serie de datos referidos a reportes de accidentes en oleoductos de Estados Unidos. En segundo lugar, se analizará el archivo “imp\_crudo.csv” que contiene informacion historica anualizada respecto a los volumenes de crudo importados en el ultimo tiempo de Estados Unidos

## Analisis de Accidentes en Oleoductos de EEUU

En primer lugar, se carga el ambiente de trabajo en el cual se desarrollará el analisis.

install.packages("tidyverse")

## Installing package into '/cloud/lib/x86\_64-pc-linux-gnu-library/4.4'  
## (as 'lib' is unspecified)

Cargamos la librerìa

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.5.2 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.4 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.4   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

Acto seguido, se carga el archivo “accidentes.csv” a un dataframe que llamaremos accidentes\_df:

accidentes\_df<- read.csv("accidentes.csv")

Hacemos una vista general de los primeros 10 datos que contiene:

head(accidentes\_df,n=10L)

## Report.Number Supplemental.Number Accident.Year Accident.Date.Time  
## 1 20100016 17305 2010 2010-01-01 07:15:00  
## 2 20100254 17331 2010 2010-01-04 08:30:00  
## 3 20100038 17747 2010 2010-01-05 10:30:00  
## 4 20100260 18574 2010 2010-01-06 19:30:00  
## 5 20100030 16276 2010 2010-01-07 13:00:00  
## 6 20100021 17161 2010 2010-01-08 23:38:00  
## 7 20110036 18052 2010 2010-01-09 00:15:00  
## 8 20100255 18584 2010 2010-01-09 01:12:00  
## 9 20100261 18050 2010 2010-01-10 19:46:00  
## 10 20100024 18390 2010 2010-01-11 14:30:00  
## Operator.ID Operator.Name Pipeline.Facility.Name  
## 1 32109 ONEOK NGL PIPELINE LP KINDER MORGAN JCT  
## 2 15786 PORTLAND PIPELINE CORP 24-INCH MAIN LINE  
## 3 20160 PETROLOGISTICS OLEFINS, LLC   
## 4 11169 ENBRIDGE ENERGY, LIMITED PARTNERSHIP SUPERIOR TERMINAL  
## 5 300 PLAINS PIPELINE, L.P. RED RIVER EAST  
## 6 11169 ENBRIDGE ENERGY, LIMITED PARTNERSHIP   
## 7 26041 KINDER MORGAN LIQUID TERMINALS, LLC   
## 8 12624 MOBIL CORP HULL STATION  
## 9 26041 KINDER MORGAN LIQUID TERMINALS, LLC   
## 10 31684 CONOCOPHILLIPS TANK 1501  
## Pipeline.Location Pipeline.Type  
## 1 ONSHORE ABOVEGROUND  
## 2 ONSHORE ABOVEGROUND  
## 3 ONSHORE ABOVEGROUND  
## 4 ONSHORE UNDERGROUND  
## 5 ONSHORE UNDERGROUND  
## 6 ONSHORE UNDERGROUND  
## 7 ONSHORE TANK  
## 8 ONSHORE ABOVEGROUND  
## 9 ONSHORE ABOVEGROUND  
## 10 ONSHORE TANK  
## Liquid.Type  
## 1 HVL OR OTHER FLAMMABLE OR TOXIC FLUID, GAS  
## 2 CRUDE OIL  
## 3 HVL OR OTHER FLAMMABLE OR TOXIC FLUID, GAS  
## 4 CRUDE OIL  
## 5 CRUDE OIL  
## 6 CRUDE OIL  
## 7 REFINED AND/OR PETROLEUM PRODUCT (NON-HVL), LIQUID  
## 8 HVL OR OTHER FLAMMABLE OR TOXIC FLUID, GAS  
## 9 REFINED AND/OR PETROLEUM PRODUCT (NON-HVL), LIQUID  
## 10 REFINED AND/OR PETROLEUM PRODUCT (NON-HVL), LIQUID  
## Liquid.Subtype Liquid.Name  
## 1 LPG (LIQUEFIED PETROLEUM GAS) / NGL (NATURAL GAS LIQUID)   
## 2   
## 3 OTHER HVL ETHANE  
## 4   
## 5   
## 6   
## 7 GASOLINE (NON-ETHANOL)   
## 8 LPG (LIQUEFIED PETROLEUM GAS) / NGL (NATURAL GAS LIQUID)   
## 9 DIESEL, FUEL OIL, KEROSENE, JET FUEL   
## 10 GASOLINE (NON-ETHANOL)   
## Accident.City Accident.County Accident.State Accident.Latitude  
## 1 MCPHERSON MCPHERSON KS 38.67070  
## 2 RAYMOND CUMBERLAND ME 43.94028  
## 3 SULPHER CALCASIEU LA 30.18240  
## 4 SUPERIOR DOUGLAS WI 46.68930  
## 5 SHERMAN GRAYSON TX 33.58266  
## 6 NECHE PEMBINA ND 48.99555  
## 7 GALENA PARK HARRIS TX 29.43050  
## 8 HULL LIBERTY TX 30.08533  
## 9 TX 29.43050  
## 10 PASADENA HARRIS TX 29.71478  
## Accident.Longitude Cause.Category  
## 1 -97.78123 INCORRECT OPERATION  
## 2 -70.49336 MATERIAL/WELD/EQUIP FAILURE  
## 3 -93.35240 MATERIAL/WELD/EQUIP FAILURE  
## 4 -92.06120 NATURAL FORCE DAMAGE  
## 5 -96.64881 EXCAVATION DAMAGE  
## 6 -97.52554 MATERIAL/WELD/EQUIP FAILURE  
## 7 -95.12010 MATERIAL/WELD/EQUIP FAILURE  
## 8 -94.38050 NATURAL FORCE DAMAGE  
## 9 -95.12010 MATERIAL/WELD/EQUIP FAILURE  
## 10 -95.17611 ALL OTHER CAUSES  
## Cause.Subcategory Unintentional.Release..Barrels.  
## 1 PIPELINE/EQUIPMENT OVERPRESSURED 21.00  
## 2 PUMP OR PUMP-RELATED EQUIPMENT 0.12  
## 3 DEFECTIVE OR LOOSE TUBING/FITTING 2.00  
## 4 TEMPERATURE 0.48  
## 5 THIRD PARTY EXCAVATION DAMAGE 700.00  
## 6 MANUFACTURING-RELATED 3784.00  
## 7 ENVIRONMENTAL CRACKING-RELATED 35.00  
## 8 TEMPERATURE 0.24  
## 9 OTHER EQUIPMENT FAILURE 0.40  
## 10 MISCELLANEOUS 0.48  
## Intentional.Release..Barrels. Liquid.Recovery..Barrels. Net.Loss..Barrels.  
## 1 0.1 0.00 21.00  
## 2 0.0 0.12 0.00  
## 3 0.0 0.00 2.00  
## 4 0.0 0.48 0.00  
## 5 NA 698.00 2.00  
## 6 0.0 1547.00 2237.00  
## 7 0.0 30.00 5.00  
## 8 0.0 0.00 0.24  
## 9 0.0 0.40 0.00  
## 10 0.0 0.48 0.00  
## Liquid.Ignition Liquid.Explosion Pipeline.Shutdown Shutdown.Date.Time  
## 1 NO NO NO   
## 2 NO NO   
## 3 NO NO   
## 4 NO NO   
## 5 NO NO NO   
## 6 NO NO YES 1/8/2010 23:41  
## 7 NO NO NO   
## 8 NO NO   
## 9 NO NO   
## 10 NO NO NO   
## Restart.Date.Time Public.Evacuations Operator.Employee.Injuries  
## 1 NA NA  
## 2 NA NA  
## 3 NA NA  
## 4 NA NA  
## 5 NA NA  
## 6 1/13/2010 9:17 0 NA  
## 7 NA NA  
## 8 NA NA  
## 9 NA NA  
## 10 NA NA  
## Operator.Contractor.Injuries Emergency.Responder.Injuries Other.Injuries  
## 1 NA NA NA  
## 2 NA NA NA  
## 3 NA NA NA  
## 4 NA NA NA  
## 5 NA NA NA  
## 6 NA NA NA  
## 7 NA NA NA  
## 8 NA NA NA  
## 9 NA NA NA  
## 10 NA NA NA  
## Public.Injuries All.Injuries Operator.Employee.Fatalities  
## 1 NA NA NA  
## 2 NA NA NA  
## 3 NA NA NA  
## 4 NA NA NA  
## 5 NA NA NA  
## 6 NA NA NA  
## 7 NA NA NA  
## 8 NA NA NA  
## 9 NA NA NA  
## 10 NA NA NA  
## Operator.Contractor.Fatalities Emergency.Responder.Fatalities  
## 1 NA NA  
## 2 NA NA  
## 3 NA NA  
## 4 NA NA  
## 5 NA NA  
## 6 NA NA  
## 7 NA NA  
## 8 NA NA  
## 9 NA NA  
## 10 NA NA  
## Other.Fatalities Public.Fatalities All.Fatalities Property.Damage.Costs  
## 1 NA NA NA 110  
## 2 NA NA NA 4000  
## 3 NA NA NA 0  
## 4 NA NA NA 200  
## 5 NA NA NA 20000  
## 6 NA NA NA 76940  
## 7 NA NA NA 0  
## 8 NA NA NA 400  
## 9 NA NA NA 0  
## 10 NA NA NA 0  
## Lost.Commodity.Costs Public.Private.Property.Damage.Costs  
## 1 1517 0  
## 2 8 0  
## 3 200 0  
## 4 40 0  
## 5 150 0  
## 6 167775 150000  
## 7 400 0  
## 8 13 0  
## 9 336 0  
## 10 50 0  
## Emergency.Response.Costs Environmental.Remediation.Costs Other.Costs  
## 1 0 0e+00 0  
## 2 0 0e+00 0  
## 3 0 0e+00 0  
## 4 11300 0e+00 0  
## 5 7500 2e+03 0  
## 6 1800000 2e+06 0  
## 7 0 7e+04 0  
## 8 0 0e+00 0  
## 9 0 4e+04 0  
## 10 10000 1e+04 10000  
## All.Costs Pressure..psi. Temperature...C. Flow.Rate..barrels.day.  
## 1 1627 902 68 116820  
## 2 4008 887 75 100263  
## 3 200 1057 63 102955  
## 4 11540 1076 72 93707  
## 5 29650 848 66 65658  
## 6 4194715 987 55 99150  
## 7 70400 850 42 109092  
## 8 413 966 57 50627  
## 9 40336 1064 65 108148  
## 10 30050 1063 74 75920  
## Vibration.Level..mm.s. Corrosion.Level..mm.year. Last.Maintenance.Date  
## 1 1.3 0.5 2022-02-19  
## 2 2.4 0.3 2024-03-05  
## 3 2.9 0.3 2023-08-13  
## 4 2.2 0.3 2024-02-09  
## 5 1.8 0.4 2023-10-08  
## 6 1.9 1.2 2022-02-26  
## 7 2.2 0.5 2023-05-23  
## 8 2.9 1.0 2022-03-15  
## 9 1.5 0.6 2023-12-22  
## 10 1.5 1.1 2025-01-14  
## Авария  
## 1 1  
## 2 1  
## 3 1  
## 4 1  
## 5 1  
## 6 1  
## 7 1  
## 8 1  
## 9 1  
## 10 1

Exploramos un poco mas las columnas y sus datos

colnames(accidentes\_df)

## [1] "Report.Number"   
## [2] "Supplemental.Number"   
## [3] "Accident.Year"   
## [4] "Accident.Date.Time"   
## [5] "Operator.ID"   
## [6] "Operator.Name"   
## [7] "Pipeline.Facility.Name"   
## [8] "Pipeline.Location"   
## [9] "Pipeline.Type"   
## [10] "Liquid.Type"   
## [11] "Liquid.Subtype"   
## [12] "Liquid.Name"   
## [13] "Accident.City"   
## [14] "Accident.County"   
## [15] "Accident.State"   
## [16] "Accident.Latitude"   
## [17] "Accident.Longitude"   
## [18] "Cause.Category"   
## [19] "Cause.Subcategory"   
## [20] "Unintentional.Release..Barrels."   
## [21] "Intentional.Release..Barrels."   
## [22] "Liquid.Recovery..Barrels."   
## [23] "Net.Loss..Barrels."   
## [24] "Liquid.Ignition"   
## [25] "Liquid.Explosion"   
## [26] "Pipeline.Shutdown"   
## [27] "Shutdown.Date.Time"   
## [28] "Restart.Date.Time"   
## [29] "Public.Evacuations"   
## [30] "Operator.Employee.Injuries"   
## [31] "Operator.Contractor.Injuries"   
## [32] "Emergency.Responder.Injuries"   
## [33] "Other.Injuries"   
## [34] "Public.Injuries"   
## [35] "All.Injuries"   
## [36] "Operator.Employee.Fatalities"   
## [37] "Operator.Contractor.Fatalities"   
## [38] "Emergency.Responder.Fatalities"   
## [39] "Other.Fatalities"   
## [40] "Public.Fatalities"   
## [41] "All.Fatalities"   
## [42] "Property.Damage.Costs"   
## [43] "Lost.Commodity.Costs"   
## [44] "Public.Private.Property.Damage.Costs"  
## [45] "Emergency.Response.Costs"   
## [46] "Environmental.Remediation.Costs"   
## [47] "Other.Costs"   
## [48] "All.Costs"   
## [49] "Pressure..psi."   
## [50] "Temperature...C."   
## [51] "Flow.Rate..barrels.day."   
## [52] "Vibration.Level..mm.s."   
## [53] "Corrosion.Level..mm.year."   
## [54] "Last.Maintenance.Date"   
## [55] "Авария"

#str(accidentes\_df)

Viendo los datos, los nombres de las columnas, podemos hacer un analisis conceptual para dirigir el desarrollo hacia los objetivos del analisis: • Número total de accidentes por año y mes • Volumen derramado o afectado • Tipo y causa principal de los accidentes • Volumen de crudo importado en cada período

Se observa que existen varios conjuntos de datos interesantes para abordar en torno a un accidente en ductos de EEUU: - Hay una parte de los datos que habla sobre los costos totales y por clasificacion. -Luego, se observa un conjunto que habla de las fatalidades y lesionados relacionados a los eventos. - Tambien hay informacion relevante respecto a los derrames. - Finalmente, se tratará los operadores de ductos con mayor cantidad de eventos.

Ahora bien, explorando los datos disponibles, podemos observar que hay una gran cantidad de datos en NA (non-available).Sobre todo en lo que refiere a lesionados y fatalidades. Esto es llamativo.

Pero avanzamos, y determinamos los costos totales, para entender cuanto dinero se gastó en accidentes de ductos. Para esto primero, realizamos un pivot a long, considerando los items referidos a costos, es decir todos aquellos que tienen la palabra “cost”:

accidentes\_largo <- accidentes\_df %>%  
 select(`Accident.Year`,  
 `Property.Damage.Costs`,  
 `Lost.Commodity.Costs`,  
 `Public.Private.Property.Damage.Costs`,  
 `Emergency.Response.Costs`,  
 `Environmental.Remediation.Costs`,  
 `Other.Costs`) %>%  
 pivot\_longer(  
 cols = -`Accident.Year`,  
 names\_to = "Tipo\_Costo",  
 values\_to = "Valor\_Costo"  
 )

La tabla se vería así:

head(accidentes\_largo, n=10L)

## # A tibble: 10 × 3  
## Accident.Year Tipo\_Costo Valor\_Costo  
## <int> <chr> <dbl>  
## 1 2010 Property.Damage.Costs 110  
## 2 2010 Lost.Commodity.Costs 1517  
## 3 2010 Public.Private.Property.Damage.Costs 0  
## 4 2010 Emergency.Response.Costs 0  
## 5 2010 Environmental.Remediation.Costs 0  
## 6 2010 Other.Costs 0  
## 7 2010 Property.Damage.Costs 4000  
## 8 2010 Lost.Commodity.Costs 8  
## 9 2010 Public.Private.Property.Damage.Costs 0  
## 10 2010 Emergency.Response.Costs 0

Ahora bien, agrupamos y sumamos por tipo de costo. Los ordenamos de manera descendente:

resumen\_costos <- accidentes\_largo %>%  
 group\_by(`Accident.Year`, Tipo\_Costo) %>%  
 summarise(Total = sum(Valor\_Costo, na.rm = TRUE)) %>%  
 arrange(`Accident.Year`, desc(Total))

## `summarise()` has grouped output by 'Accident.Year'. You can override using the  
## `.groups` argument.

head(resumen\_costos)

## # A tibble: 6 × 3  
## # Groups: Accident.Year [1]  
## Accident.Year Tipo\_Costo Total  
## <int> <chr> <dbl>  
## 1 2010 Environmental.Remediation.Costs 710715040  
## 2 2010 Emergency.Response.Costs 249808092  
## 3 2010 Property.Damage.Costs 54535020  
## 4 2010 Public.Private.Property.Damage.Costs 37517397  
## 5 2010 Other.Costs 17827193  
## 6 2010 Lost.Commodity.Costs 4791248

Finalmente, agrupamos por año:

resumen\_costos\_año <- accidentes\_df %>%  
 group\_by(`Accident.Year`) %>%  
 summarise(Total = sum(`All.Costs`, na.rm = TRUE))  
print(resumen\_costos\_año)

## # A tibble: 13 × 2  
## Accident.Year Total  
## <int> <int>  
## 1 2010 1075193990  
## 2 2011 273526547  
## 3 2012 145247426  
## 4 2013 278525540  
## 5 2014 131684524  
## 6 2015 253696042  
## 7 2016 173161626  
## 8 2017 87228  
## 9 2018 0  
## 10 2019 0  
## 11 2020 0  
## 12 2021 0  
## 13 2022 0

Si lo graficamos por año en un grafico de barras se observa:

ggplot(resumen\_costos\_año, aes(x = `Accident.Year`, y = Total)) +  
 geom\_col(fill = "blue") +  
 labs(  
 title = "Costos Totales por Año",  
 x = "Año del Accidente",  
 y = "Costo Total (USD)"  
 ) +  
 theme\_bw()

Gráfico, Histograma

El contenido generado por IA puede ser incorrecto.

Aqui hay algo interesante. No hay datos para los años 2017 en adelante. Mas adelante, exploramos esto. Ahora, graficamos costos en un grafico de barras compuesto donde se observa el desglose de los costos segun su clasificacion.

ggplot(resumen\_costos, aes(x = `Accident.Year`, y = Total, fill = Tipo\_Costo)) +  
 geom\_col(position = "stack") +  
 labs(  
 title = "Costos por Tipo y Año de Accidente",  
 x = "Año del Accidente",  
 y = "Costo Total (USD)",  
 fill = "Tipo de Costo"  
 ) +  
 theme\_minimal()

Gráfico, Gráfico de barras

El contenido generado por IA puede ser incorrecto. Sin dudas en el 2010, hubieron muchos costos asociados a la remediacion ambiental y para los años siguientes, el costo por respuesta ante emergencia siempre fue superior al resto de los costos, salvo por los años 2012 y 2014.

Ahora bien, nuevamente, los costos desde 2017 hasta la actualidad son 0. La pregunta es, ¿Hubieron accidentes? Bueno, analicemos si hubieron daños ambientales, fatalidades, etc.

accidentes\_df %>%  
 filter(`Accident.Year` >= 2017) %>%  
 count(`Accident.Year`)

## Accident.Year n  
## 1 2017 82  
## 2 2018 79  
## 3 2019 81  
## 4 2020 75  
## 5 2021 84  
## 6 2022 73

accidentes\_df %>%  
 filter(`Accident.Year` >= 2017) %>%  
 summarise(Derrame = sum(`Net.Loss..Barrels.`, na.rm = TRUE))

## Derrame  
## 1 0

accidentes\_df %>%  
 filter(`Accident.Year` >= 2017) %>%  
 summarise(Fatalidades = sum(`All.Fatalities`, na.rm = TRUE))

## Fatalidades  
## 1 0

accidentes\_df %>%  
 filter(`Accident.Year` >= 2017) %>%  
 summarise(Lesionados = sum(`All.Injuries`, na.rm = TRUE))

## Lesionados  
## 1 0

Aqui hay un hallazgo interesante. Hay casos reportados entre 2017 y 2022 pero: fatalidades, lesionados, derrames y costos son 0. Por lo tanto, este periodo de tiempo no es util para nuestro analisis.Trabajaremos con los datos entre 2010 y 2016.

Avanzamos un poco mas. Observar que hay una columna que indica cada operador y el caso reportado.¿Cuantos casos tuvo cada operador?

q\_casos\_operadores <- accidentes\_df %>%  
 filter(`Accident.Year` <= 2017 )%>%  
 group\_by(`Operator.Name`) %>%  
 summarise(Cantidad\_Casos= n())%>%  
 arrange(desc(Cantidad\_Casos))  
  
head(q\_casos\_operadores, n=10L)

## # A tibble: 10 × 2  
## Operator.Name Cantidad\_Casos  
## <chr> <int>  
## 1 "" 608  
## 2 "ENTERPRISE CRUDE PIPELINE LLC" 195  
## 3 "SUNOCO PIPELINE L.P." 180  
## 4 "PLAINS PIPELINE, L.P." 156  
## 5 "ENTERPRISE PRODUCTS OPERATING LLC" 155  
## 6 "MAGELLAN PIPELINE COMPANY, LP" 140  
## 7 "COLONIAL PIPELINE CO" 136  
## 8 "BUCKEYE PARTNERS, LP" 115  
## 9 "KINDER MORGAN LIQUID TERMINALS, LLC" 87  
## 10 "MARATHON PIPE LINE LLC" 87

Aqui detectamos que hay casos que no tienen el operador reportado (està vacìo, por lo que lo hacemos visible). Esto es un problema si tienen costos asociados. Por lo que determinamos que impacto tiene en los costos, ya que es una gran cantidad:

accidentes\_df <- accidentes\_df %>%  
 mutate(`Operator.Name` = na\_if(`Operator.Name`, ""))

Determinamos el impacto:

accidentes\_df %>%  
 group\_by(is.na(`Operator.Name`)) %>%  
 summarise(Total\_Costos = sum(`All.Costs`, na.rm = TRUE),  
 Casos = n())

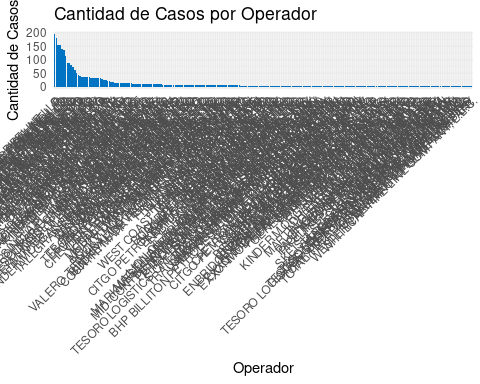
## # A tibble: 2 × 3  
## `is.na(Operator.Name)` Total\_Costos Casos  
## <lgl> <dbl> <int>  
## 1 FALSE 2331122923 2795  
## 2 TRUE 0 1000

Se identifica que aproximadamente un 26% de los registros (1000 de 3795) no informan el nombre del operador responsable del accidente. Sin embargo, al analizar la columna All Costs, se observó que el 100% de los costos económicos están concentrados en los casos donde sí se reporta el operador. En consecuencia, se decidió excluir los registros con Operator Name = NA del análisis económico, ya que no aportan información relevante sobre impacto financiero.

q\_casos\_operadores2 <- accidentes\_df %>%  
 filter(`Accident.Year` <= 2017 & !is.na(`Operator.Name`))%>%  
 group\_by(`Operator.Name`) %>%  
 summarise(Cantidad\_Casos= n())%>%  
 arrange(desc(Cantidad\_Casos))

Si graficamos los casos por operador obtenemos:

ggplot(q\_casos\_operadores2, aes(x = reorder(`Operator.Name`, -Cantidad\_Casos), y = Cantidad\_Casos)) +  
 geom\_bar(stat = "identity", fill = "#0073C2FF") +  
 labs(  
 title = "Cantidad de Casos por Operador",  
 x = "Operador",  
 y = "Cantidad de Casos"  
 ) +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))

 Sin dudas, utilizar esta sentencia nos lleva a una grafica que no es clara, por lo que se aplicará el principio de pareto para determinar el 20% de operadores que concentrar el 80 % de los casos y se los listarà y graficarà Primero, determinamos el total de casos:

total\_casos <- sum(q\_casos\_operadores2$Cantidad\_Casos)

Luego, determinamos la lista aplicando el Principio de Pareto

casos\_pareto <- q\_casos\_operadores2 %>%  
 arrange(desc(Cantidad\_Casos)) %>%  
 mutate(  
 Porcentaje = Cantidad\_Casos / total\_casos,  
 Porcentaje\_Acumulado = cumsum(Porcentaje)  
 )

Determinamos el top\_operadores

top\_operadores <- casos\_pareto %>%  
 filter(Porcentaje\_Acumulado <= 0.8)

Finalmente, lo graficamos:

ggplot(top\_operadores, aes(x = reorder(`Operator.Name`, -Cantidad\_Casos), y = Cantidad\_Casos)) +  
 geom\_bar(stat = "identity", fill = "#E69F00") +  
 labs(  
 title = "Operadores con el 80% de los Casos",  
 x = "Operador",  
 y = "Cantidad de Casos"  
 ) +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))

Gráfico

El contenido generado por IA puede ser incorrecto. Estamos hablando de que 42 operadores concentran el 80% de los casos. El top ten es:

top\_10\_casos <- top\_operadores %>%  
 slice\_max(order\_by = Cantidad\_Casos, n = 10)

Graficamos el top 10 de operadores con mayor cantidad de casos reportados:

ggplot(top\_10\_casos, aes(x = reorder(`Operator.Name`, -Cantidad\_Casos), y = Cantidad\_Casos)) +  
 geom\_bar(stat = "identity", fill = "red") +  
 labs(  
 title = "Top 10 Cantidad de Casos Operadores",  
 x = "Operador",  
 y = "Cantidad de Casos"  
 ) +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))

Gráfico, Gráfico de barras

El contenido generado por IA puede ser incorrecto.

Ahora determinamos los costos por operador:

costos\_por\_operador <- accidentes\_df %>%  
 filter(`Accident.Year` <= 2017 & !is.na(`Operator.Name`)) %>%  
 group\_by(`Operator.Name`) %>%  
 summarise(Total\_Costos = sum(`All.Costs`, na.rm = TRUE)) %>%  
 arrange(desc(Total\_Costos))

Determinamos el top 10 de operadores con mayores costos por accidentes en ductos:

top\_10\_costos <- costos\_por\_operador %>%  
 slice\_max(order\_by = Total\_Costos, n = 10)

Graficamos lo anterior

ggplot(top\_10\_costos, aes(x = reorder(`Operator.Name`, -Total\_Costos), y = Total\_Costos)) +  
 geom\_bar(stat = "identity", fill = "yellow") +  
 labs(  
 title = "Top 10 Mayores Costos Operadores",  
 x = "Operador",  
 y = "Total Costos"  
 ) +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))

Imagen que contiene Gráfico

El contenido generado por IA puede ser incorrecto.

Ahora queremos saber si coincide el top 10 de casos con el top 10 de costos y lo graficamos con un grafico de burbujas:

interseccion\_top\_10 <- inner\_join(top\_10\_casos, top\_10\_costos, by = "Operator.Name")

ggplot(interseccion\_top\_10, aes(x = Cantidad\_Casos, y = Total\_Costos, size = Total\_Costos, label = `Operator.Name`)) +  
 geom\_point(alpha = 0.7, color = "green") +  
 geom\_text(vjust = -1, size = 3) +  
 scale\_size\_continuous(range = c(4, 12)) +  
 labs(  
 title = "Top Operadores por Casos y Costos",  
 x = "Cantidad de Casos",  
 y = "Costos Totales (USD)",  
 size = "Costo Total"  
 ) +  
 theme\_minimal()

Gráfico, Gráfico de dispersión, Gráfico de burbujas

El contenido generado por IA puede ser incorrecto.

Probamos con otra forma de cruzar las tablas, ya que no el grafico propuesto no brinda mucha info: Determinamos otro cruce de tablas y lo graficamos:

top\_10\_combinado <- inner\_join(top\_10\_casos, costos\_por\_operador, by = "Operator.Name")

ggplot(top\_10\_combinado, aes(x = Cantidad\_Casos, y = Total\_Costos, size = Total\_Costos, label = `Operator.Name`)) +  
 geom\_point(alpha = 0.7, color = "magenta") +  
 geom\_text(vjust = -1, size = 3) +  
 scale\_size\_continuous(range = c(4, 12)) +  
 labs(  
 title = "Top Operadores por Casos y Costos",  
 x = "Cantidad de Casos",  
 y = "Costos Totales (USD)",  
 size = "Costo Total"  
 ) +  
 theme\_minimal()

Gráfico, Gráfico de burbujas

El contenido generado por IA puede ser incorrecto. Acomodamos el grafico dado que las etiquetas se superponen y no lo hacen claro

install.packages("ggrepel")

## Installing package into '/cloud/lib/x86\_64-pc-linux-gnu-library/4.4'  
## (as 'lib' is unspecified)

library(ggrepel)

Y volvemos a graficar…

ggplot(top\_10\_combinado, aes(x = Cantidad\_Casos, y = Total\_Costos, size = Total\_Costos, label = `Operator.Name`)) +  
 geom\_point(alpha = 0.7, color = "magenta") +  
 geom\_text\_repel(size = 3) +  
 scale\_size\_continuous(range = c(4, 12)) +  
 labs(  
 title = "Top Operadores por Casos y Costos",  
 x = "Cantidad de Casos",  
 y = "Costos Totales (USD)",  
 size = "Costo Total"  
 )

Gráfico

El contenido generado por IA puede ser incorrecto.

theme\_minimal()

## List of 136  
## $ line :List of 6  
## ..$ colour : chr "black"  
## ..$ linewidth : num 0.5  
## ..$ linetype : num 1  
## ..$ lineend : chr "butt"  
## ..$ arrow : logi FALSE  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_line" "element"  
## $ rect :List of 5  
## ..$ fill : chr "white"  
## ..$ colour : chr "black"  
## ..$ linewidth : num 0.5  
## ..$ linetype : num 1  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_rect" "element"  
## $ text :List of 11  
## ..$ family : chr ""  
## ..$ face : chr "plain"  
## ..$ colour : chr "black"  
## ..$ size : num 11  
## ..$ hjust : num 0.5  
## ..$ vjust : num 0.5  
## ..$ angle : num 0  
## ..$ lineheight : num 0.9  
## ..$ margin : 'margin' num [1:4] 0points 0points 0points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : logi FALSE  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ title : NULL  
## $ aspect.ratio : NULL  
## $ axis.title : NULL  
## $ axis.title.x :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 1  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 2.75points 0points 0points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.title.x.top :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 0  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 0points 2.75points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.title.x.bottom : NULL  
## $ axis.title.y :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 1  
## ..$ angle : num 90  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 2.75points 0points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.title.y.left : NULL  
## $ axis.title.y.right :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 1  
## ..$ angle : num -90  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 0points 0points 2.75points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : chr "grey30"  
## ..$ size : 'rel' num 0.8  
## ..$ hjust : NULL  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : NULL  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text.x :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 1  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 2.2points 0points 0points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text.x.top :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 0  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 0points 2.2points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text.x.bottom : NULL  
## $ axis.text.y :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : num 1  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 2.2points 0points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text.y.left : NULL  
## $ axis.text.y.right :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : num 0  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 0points 0points 2.2points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text.theta : NULL  
## $ axis.text.r :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : num 0.5  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 2.2points 0points 2.2points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.ticks : list()  
## ..- attr(\*, "class")= chr [1:2] "element\_blank" "element"  
## $ axis.ticks.x : NULL  
## $ axis.ticks.x.top : NULL  
## $ axis.ticks.x.bottom : NULL  
## $ axis.ticks.y : NULL  
## $ axis.ticks.y.left : NULL  
## $ axis.ticks.y.right : NULL  
## $ axis.ticks.theta : NULL  
## $ axis.ticks.r : NULL  
## $ axis.minor.ticks.x.top : NULL  
## $ axis.minor.ticks.x.bottom : NULL  
## $ axis.minor.ticks.y.left : NULL  
## $ axis.minor.ticks.y.right : NULL  
## $ axis.minor.ticks.theta : NULL  
## $ axis.minor.ticks.r : NULL  
## $ axis.ticks.length : 'simpleUnit' num 2.75points  
## ..- attr(\*, "unit")= int 8  
## $ axis.ticks.length.x : NULL  
## $ axis.ticks.length.x.top : NULL  
## $ axis.ticks.length.x.bottom : NULL  
## $ axis.ticks.length.y : NULL  
## $ axis.ticks.length.y.left : NULL  
## $ axis.ticks.length.y.right : NULL  
## $ axis.ticks.length.theta : NULL  
## $ axis.ticks.length.r : NULL  
## $ axis.minor.ticks.length : 'rel' num 0.75  
## $ axis.minor.ticks.length.x : NULL  
## $ axis.minor.ticks.length.x.top : NULL  
## $ axis.minor.ticks.length.x.bottom: NULL  
## $ axis.minor.ticks.length.y : NULL  
## $ axis.minor.ticks.length.y.left : NULL  
## $ axis.minor.ticks.length.y.right : NULL  
## $ axis.minor.ticks.length.theta : NULL  
## $ axis.minor.ticks.length.r : NULL  
## $ axis.line : list()  
## ..- attr(\*, "class")= chr [1:2] "element\_blank" "element"  
## $ axis.line.x : NULL  
## $ axis.line.x.top : NULL  
## $ axis.line.x.bottom : NULL  
## $ axis.line.y : NULL  
## $ axis.line.y.left : NULL  
## $ axis.line.y.right : NULL  
## $ axis.line.theta : NULL  
## $ axis.line.r : NULL  
## $ legend.background : list()  
## ..- attr(\*, "class")= chr [1:2] "element\_blank" "element"  
## $ legend.margin : 'margin' num [1:4] 5.5points 5.5points 5.5points 5.5points  
## ..- attr(\*, "unit")= int 8  
## $ legend.spacing : 'simpleUnit' num 11points  
## ..- attr(\*, "unit")= int 8  
## $ legend.spacing.x : NULL  
## $ legend.spacing.y : NULL  
## $ legend.key : list()  
## ..- attr(\*, "class")= chr [1:2] "element\_blank" "element"  
## $ legend.key.size : 'simpleUnit' num 1.2lines  
## ..- attr(\*, "unit")= int 3  
## $ legend.key.height : NULL  
## $ legend.key.width : NULL  
## $ legend.key.spacing : 'simpleUnit' num 5.5points  
## ..- attr(\*, "unit")= int 8  
## $ legend.key.spacing.x : NULL  
## $ legend.key.spacing.y : NULL  
## $ legend.frame : NULL  
## $ legend.ticks : NULL  
## $ legend.ticks.length : 'rel' num 0.2  
## $ legend.axis.line : NULL  
## $ legend.text :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : 'rel' num 0.8  
## ..$ hjust : NULL  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : NULL  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ legend.text.position : NULL  
## $ legend.title :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : num 0  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : NULL  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ legend.title.position : NULL  
## $ legend.position : chr "right"  
## $ legend.position.inside : NULL  
## $ legend.direction : NULL  
## $ legend.byrow : NULL  
## $ legend.justification : chr "center"  
## $ legend.justification.top : NULL  
## $ legend.justification.bottom : NULL  
## $ legend.justification.left : NULL  
## $ legend.justification.right : NULL  
## $ legend.justification.inside : NULL  
## $ legend.location : NULL  
## $ legend.box : NULL  
## $ legend.box.just : NULL  
## $ legend.box.margin : 'margin' num [1:4] 0cm 0cm 0cm 0cm  
## ..- attr(\*, "unit")= int 1  
## $ legend.box.background : list()  
## ..- attr(\*, "class")= chr [1:2] "element\_blank" "element"  
## $ legend.box.spacing : 'simpleUnit' num 11points  
## ..- attr(\*, "unit")= int 8  
## [list output truncated]  
## - attr(\*, "class")= chr [1:2] "theme" "gg"  
## - attr(\*, "complete")= logi TRUE  
## - attr(\*, "validate")= logi TRUE

Ahora, realizamos el analisis de fatalidades y lesiones totales por año

fatalidades\_lesionados\_anuales <- accidentes\_df %>%  
 filter(`Accident.Year` <= 2017) %>%  
 group\_by(`Accident.Year`) %>%  
 summarise(  
 Fatalidades = sum(`All.Fatalities`, na.rm = TRUE),  
 Lesionados = sum(`All.Injuries`, na.rm = TRUE)  
 )

Si lo vemos como una tabla:

print(fatalidades\_lesionados\_anuales)

## # A tibble: 8 × 3  
## Accident.Year Fatalidades Lesionados  
## <int> <dbl> <dbl>  
## 1 2010 1 3  
## 2 2011 1 2  
## 3 2012 3 4  
## 4 2013 1 6  
## 5 2014 0 0  
## 6 2015 1 0  
## 7 2016 3 5  
## 8 2017 0 0

Desglosamos por tipo de afectado y lo graficamos.

afectados\_tipo <- accidentes\_df %>%  
 filter(`Accident.Year` <= 2017) %>%  
 summarise(  
 Empleados = sum(`Operator.Employee.Fatalities`, na.rm = TRUE),  
 Contratistas = sum(`Operator.Contractor.Fatalities`, na.rm = TRUE),  
 Rescatistas = sum(`Emergency.Responder.Fatalities`, na.rm = TRUE),  
 Publico = sum(`Public.Fatalities`, na.rm = TRUE),  
 Otros = sum(`Other.Fatalities`, na.rm = TRUE)  
 ) %>%  
 pivot\_longer(everything(), names\_to = "Tipo", values\_to = "Cantidad")

ggplot(afectados\_tipo, aes(x = "", y = Cantidad, fill = Tipo)) +  
 geom\_bar(stat = "identity", width = 1) +  
 coord\_polar("y") +  
 labs(  
 title = "Distribución de Fatalidades por Tipo de Afectado",  
 x = NULL,  
 y = NULL,  
 fill = "Tipo de Afectado"  
 ) +  
 theme\_void()

Gráfico, Gráfico circular

El contenido generado por IA puede ser incorrecto. Lo resonante del grafico anterior es que hubieron fatalidades.

Veamos ahora el top 10 de operadores relacionados a fatalidades y lesionados:

impacto\_humano\_por\_operador <- accidentes\_df %>%  
 filter(`Accident.Year` <= 2017, !is.na(`Operator.Name`)) %>%  
 group\_by(`Operator.Name`) %>%  
 summarise(  
 Total\_Fatalidades = sum(`All.Fatalities`, na.rm = TRUE),  
 Total\_Lesionados = sum(`All.Injuries`, na.rm = TRUE),  
 Total\_Afectados = Total\_Fatalidades + Total\_Lesionados  
 ) %>%  
 arrange(desc(Total\_Afectados))

Determinamos el top 10 y lo graficamos.

top\_10\_afectados <- impacto\_humano\_por\_operador %>%  
 slice\_max(order\_by = Total\_Afectados, n = 10)

ggplot(top\_10\_afectados, aes(x = reorder(`Operator.Name`, -Total\_Afectados), y = Total\_Afectados)) +  
 geom\_bar(stat = "identity", fill = "#E41A1C") +  
 labs(  
 title = "Top 10 Operadores con Mayor Cantidad de Afectados (Fatalidades + Lesionados)",  
 x = "Operador",  
 y = "Cantidad de Afectados"  
 ) +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))

Gráfico

El contenido generado por IA puede ser incorrecto.

Pero queremos destacar entre fatalidades y lesiones y verlo en un grafico:

top\_10\_afectados\_2 <- top\_10\_afectados %>%  
 select(`Operator.Name`, Total\_Fatalidades, Total\_Lesionados) %>%  
 pivot\_longer(  
 cols = c(Total\_Fatalidades, Total\_Lesionados),  
 names\_to = "Tipo",  
 values\_to = "Cantidad"  
 ) %>%  
 mutate(Tipo = recode(Tipo,  
 "Total\_Fatalidades" = "Fatalidades",  
 "Total\_Lesionados" = "Lesionados"))

ggplot(top\_10\_afectados\_2, aes(x = reorder(`Operator.Name`, -Cantidad), y = Cantidad, fill = Tipo)) +  
 geom\_bar(stat = "identity") +  
 labs(  
 title = "Top 10 Operadores con Mayor Cantidad de Afectados",  
 x = "Operador",  
 y = "Cantidad de Personas",  
 fill = "Tipo de Afectado"  
 ) +  
 scale\_fill\_manual(values = c("Fatalidades" = "#D73027", "Lesionados" = "#4575B4")) +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))

Gráfico, Gráfico en cascada

El contenido generado por IA puede ser incorrecto.

El gráfico anterior muestra el Top 10 de operadores que concentraron la mayor cantidad de personas afectadas por accidentes en oleoductos entre 2010 y 2017, considerando tanto fatalidades como lesionados.

Se observa que, si bien la mayoría de los casos corresponden a personas lesionadas, algunos operadores presentan una proporción significativa de fatalidades dentro de su total de afectados. Este desglose permite diferenciar entre operadores que tuvieron numerosos eventos con consecuencias moderadas y aquellos que, aunque quizás con menos eventos, generaron un impacto más grave.

Este análisis es clave para enfocar estrategias regulatorias y de fiscalización sobre aquellos operadores que representan un mayor riesgo para la seguridad de las personas, más allá del volumen total de incidentes.

Evaluamos el daño ambiental generado:

derrames <- accidentes\_df %>%  
 filter(`Accident.Year` <= 2017) %>%  
 group\_by(`Accident.Year`) %>%  
 summarise(Volumen\_Derramado = sum(`Net.Loss..Barrels.`, na.rm = TRUE))

Graficamos el volumen derramado a lo largo del periodo:

ggplot(derrames, aes(x = `Accident.Year`, y = Volumen\_Derramado)) +  
 geom\_line(color = "#1B9E77", size = 1.2) +  
 labs(  
 title = "Volumen de Crudo Derramado por Año",  
 x = "Año del Accidente",  
 y = "Barriles Perdidos"  
 ) +  
 theme\_minimal()

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
## ℹ Please use `linewidth` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

Gráfico, Gráfico de líneas

El contenido generado por IA puede ser incorrecto.

Determinamos los costos por daño ambiental y lo graficamos:

costos\_ambientales\_anuales <- accidentes\_df %>%  
 filter(`Accident.Year` <= 2017) %>%  
 group\_by(`Accident.Year`) %>%  
 summarise(Costos\_Remediacion = sum(`Environmental.Remediation.Costs`, na.rm = TRUE))

ggplot(costos\_ambientales\_anuales, aes(x = `Accident.Year`, y = Costos\_Remediacion)) +  
 geom\_line(color = "#D95F02", size = 1.2) +  
 labs(  
 title = "Costos de Remediación Ambiental por Año",  
 x = "Año",  
 y = "Costos (USD)"  
 ) +  
 theme\_minimal()

Gráfico, Gráfico de líneas

El contenido generado por IA puede ser incorrecto.

Relacion entre costos y volumenes derramados y su grafico correspondiente.

relacion\_derrames\_costos <- accidentes\_df %>%  
 filter(`Accident.Year` <= 2017) %>%  
 group\_by(`Accident.Year`) %>%  
 summarise(  
 Volumen = sum(`Net.Loss..Barrels.`, na.rm = TRUE),  
 Costo = sum(`Environmental.Remediation.Costs`, na.rm = TRUE)  
 )

ggplot(relacion\_derrames\_costos, aes(x = Volumen, y = Costo)) +  
 geom\_point(size = 3, color = "#7570B3") +  
 geom\_smooth(method = "lm", se = FALSE, linetype = "dashed") +  
 labs(  
 title = "Relación entre Volumen Derramado y Costo de Remediación",  
 x = "Barriles Perdidos",  
 y = "Costos Ambientales (USD)"  
 ) +  
 theme\_minimal()

## `geom\_smooth()` using formula = 'y ~ x'

Gráfico, Gráfico de dispersión

El contenido generado por IA puede ser incorrecto.

Determinamos el top 10 de operadores con mayores derrames:

top\_10\_ambiental <- accidentes\_df %>%  
 filter(`Accident.Year` <= 2017, !is.na(`Operator.Name`)) %>%  
 group\_by(`Operator.Name`) %>%  
 summarise(  
 Volumen\_Derramado = sum(`Net.Loss..Barrels.`, na.rm = TRUE),  
 Costo\_Ambiental = sum(`Environmental.Remediation.Costs`, na.rm = TRUE)  
 ) %>%  
 mutate(Impacto\_Ambiental = Volumen\_Derramado + Costo\_Ambiental) %>%  
 arrange(desc(Impacto\_Ambiental)) %>%  
 slice\_max(order\_by = Impacto\_Ambiental, n = 10)

Finalmente, armamos un tablero general para identificar los operadores top 10 segun cada tipo de daño:

top\_10\_casos <- top\_10\_casos %>%  
 mutate(Metrica = "Cantidad de Casos") %>%  
 select(`Operator.Name`, Valor = Cantidad\_Casos, Metrica)  
  
top\_10\_costos <- top\_10\_costos %>%  
 mutate(Metrica = "Costos Totales") %>%  
 select(`Operator.Name`, Valor = Total\_Costos, Metrica)  
  
top\_10\_afectados\_simple <- top\_10\_afectados %>%  
 mutate(Metrica = "Personas Afectadas") %>%  
 select(`Operator.Name`, Valor = Total\_Afectados, Metrica)  
  
top\_10\_ambiental\_simple <- top\_10\_ambiental %>%  
 mutate(Metrica = "Daño Ambiental") %>%  
 select(`Operator.Name`, Valor = Impacto\_Ambiental, Metrica)

Los unimos todos y los graficamos con un facet wrap

top10\_general <- bind\_rows(  
 top\_10\_casos,  
 top\_10\_costos,  
 top\_10\_afectados\_simple,  
 top\_10\_ambiental\_simple  
)

ggplot(top10\_general, aes(x = reorder(`Operator.Name`, Valor), y = Valor, fill = Metrica)) +  
 geom\_bar(stat = "identity", show.legend = FALSE) +  
 facet\_wrap(~ Metrica, scales = "free", ncol = 2) +  
 coord\_flip() +  
 labs(  
 title = "Top 10 de Operadores según Distintos Tipos de Impacto",  
 x = "Operador",  
 y = NULL  
 ) +  
 theme\_minimal()

Imagen de la pantalla de un celular de un mensaje en letras blancas

El contenido generado por IA puede ser incorrecto.

## Analisis de Importaciones de Crudo de Estados Unidos

Descargamos el archivo “imp\_crudo.csv”, desplegamos su vision general, nombres de columnas y tipos de datos:

importaciones\_df <- read\_csv("imp\_crudo.csv")

## Rows: 483053 Columns: 8  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (6): month, originName, originTypeName, destinationName, destinationType...  
## dbl (2): year, quantity  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

head(importaciones\_df)

## # A tibble: 6 × 8  
## year month originName originTypeName destinationName destinationTypeName  
## <dbl> <chr> <chr> <chr> <chr> <chr>   
## 1 2009 01 Belize Country EXXONMOBIL REFINING… Refinery   
## 2 2009 01 Belize Country FLINT HILLS RESOURC… Refinery   
## 3 2009 01 Algeria Country SHELL OIL PRODUCTS … Refinery   
## 4 2009 01 Algeria Country OIL TANKING PL INC … Refinery   
## 5 2009 01 Algeria Country UNKNOWN PROCESSOR-T… Refinery   
## 6 2009 01 Algeria Country VALERO REFINING CO … Refinery   
## # ℹ 2 more variables: gradeName <chr>, quantity <dbl>

colnames(importaciones\_df)

## [1] "year" "month" "originName"   
## [4] "originTypeName" "destinationName" "destinationTypeName"  
## [7] "gradeName" "quantity"

#str(importaciones\_df)

Determinamos la evolucion de importaciones anuales. Para esto, considerando los hallazgos del analisis de los accidentes en ductos, solamente nos centramos en el periodo 2010 a 2017.

importaciones\_anuales <- importaciones\_df %>%  
 filter(`year`>=2010 & `year`<=2017) %>%  
 group\_by(year) %>%  
 summarise(Volumen\_Total = sum(quantity, na.rm = TRUE))

Graficamos lo mencionado:

ggplot(importaciones\_anuales, aes(x = year, y = Volumen\_Total)) +  
 geom\_line(size = 1.2, color = "#1F78B4") +  
 labs(  
 title = "Volumen Total de Importaciones de Crudo por Año",  
 x = "Año",  
 y = "Volumen (barriles)"  
 ) +  
 theme\_minimal()

Gráfico, Gráfico de líneas

El contenido generado por IA puede ser incorrecto. Determinamos el top 10 de los paises de origen, en ese periodo y lo graficamos:

top\_paises <- importaciones\_df %>%  
 filter(`year`>=2010 & `year`<=2017) %>%  
 group\_by(originName) %>%  
 summarise(Volumen = sum(quantity, na.rm = TRUE)) %>%  
 arrange(desc(Volumen)) %>%  
 slice\_max(order\_by = Volumen, n = 10)

ggplot(top\_paises, aes(x = reorder(originName, Volumen), y = Volumen)) +  
 geom\_bar(stat = "identity", fill = "#33A02C") +  
 coord\_flip() +  
 labs(  
 title = "Top 10 Países Exportadores de Crudo a EE.UU.",  
 x = "País de Origen",  
 y = "Volumen Total (barriles)"  
 ) +  
 theme\_minimal()

Gráfico

El contenido generado por IA puede ser incorrecto.

En las sentencias estamos ignorando los NA, pero suman volumen?

importaciones\_df %>%  
 count(is.na(originName))

## # A tibble: 1 × 2  
## `is.na(originName)` n  
## <lgl> <int>  
## 1 FALSE 483053

importaciones\_df %>%  
 filter(originName=NA)%>%  
 summarise(Volumen = sum(quantity, na.rm = TRUE))

## # A tibble: 1 × 1  
## Volumen  
## <dbl>  
## 1 0

La respuesta es no, no suman volumen, por lo tanto se pueden obviar.

Top 10 destinos

top\_destinos <- importaciones\_df %>%  
 group\_by(destinationName) %>%  
 summarise(Volumen = sum(quantity, na.rm = TRUE)) %>%  
 arrange(desc(Volumen)) %>%  
 slice\_max(order\_by = Volumen, n = 10)  
print(top\_destinos)

## # A tibble: 10 × 2  
## destinationName Volumen  
## <chr> <dbl>  
## 1 United States 167359914  
## 2 PADD3 (Gulf Coast) 136162252  
## 3 PADD2 (Midwest) 96349956  
## 4 Texas 85113976  
## 5 PADD5 (West Coast) 51074916  
## 6 Illinois 46313580  
## 7 Louisiana 38831660  
## 8 California 36754240  
## 9 PADD1 (East Coast) 34435100  
## 10 Chicago, IL 17415804

Top 10 crudos mas importados

tipos\_crudo <- importaciones\_df %>%  
 group\_by(gradeName) %>%  
 summarise(Volumen = sum(quantity, na.rm = TRUE)) %>%  
 arrange(desc(Volumen)) %>%  
 slice\_max(order\_by = Volumen, n = 10)  
print(tipos\_crudo)

## # A tibble: 5 × 2  
## gradeName Volumen  
## <chr> <dbl>  
## 1 Heavy Sour 598600688  
## 2 Medium 372162574  
## 3 Light Sweet 99507912  
## 4 Light Sour 55276984  
## 5 Heavy Sweet 45971240

|  |
| --- |
| **Mapa de ruta de caso práctico - Actuar** |
| **Preguntas orientativas**   * ¿Cuál es tu conclusión en función de tu análisis? * ¿Cómo podrían tu equipo y tu empresa aplicar tus conclusiones? * ¿Qué próximos pasos tú o los interesados podrían adoptar en función de tus hallazgos? * ¿Existen datos adicionales que podrías utilizar para ampliar tus hallazgos? |
| **Tareas clave**   * Crea tu portfolio. * Agrega tu caso práctico. * Practica presentando tu caso práctico a un amigo o familiar. |
| **Entregable**  Tus conclusiones de alto nivel más importantes basadas en tu análisis |

* ¿Cuál es tu conclusión en función de tu análisis?  
  El análisis reveló patrones consistentes en los accidentes en oleoductos en EE.UU. entre 2010 y 2017. Se identificó que un pequeño grupo de operadores concentra la mayoría de los casos, los mayores costos económicos, las fatalidades y los daños ambientales. Además, los registros sin operador informado no presentan costos reportados, por lo que fueron excluidos del análisis económico. Por otro lado, las importaciones de crudo muestran una dependencia marcada de ciertos países y tipos de crudo.
* ¿Cómo podrían tu equipo y tu empresa aplicar tus conclusiones?  
  Estas conclusiones permiten focalizar acciones correctivas y preventivas. Una empresa del sector podría usar estos hallazgos para identificar mejores prácticas, fortalecer políticas de mantenimiento, y anticipar riesgos asociados al tipo de crudo transportado o a la antigüedad de los equipos. También puede servir de insumo para benchmarking interno o auditorías regulatorias.
* ¿Qué próximos pasos tú o los interesados podrían adoptar en función de tus hallazgos?  
  Se recomienda investigar en profundidad los casos de mayor impacto humano y económico, cruzar esta información con prácticas de mantenimiento (tiempo desde último mantenimiento, por ejemplo), e implementar monitoreo específico sobre los operadores más críticos. También sería valioso abrir un canal de intercambio de datos con operadores para mejorar la trazabilidad de incidentes.
* ¿Existen datos adicionales que podrías utilizar para ampliar tus hallazgos?  
  Sí. Se podrían incorporar datos de monitoreo satelital de trazas, reportes de inspección internos, presión/temperatura del sistema en tiempo real y datos desde 2018 en adelante, que actualmente no están disponibles con la misma estructura. También sería útil contar con información sobre las políticas de mantenimiento específicas por operador.