Iniciativa en Ciencia de Decisiones Solución a la Prueba - 2024-1

Jesús González Godoy, PhD | 23-02-2024

DESCRIPCIÓN

Nos interesa obtener y analizar información acerca del Consumo doméstico de energía de transporte por modo y tipo de combustible.

Particularmente, la proporción de tipo de combustible (Gasoline, Diesel fuel, Liquified petroleum gas, Natural gas, Electricity, etc.) por modo de transporte:

- 1. Light vehicles: Cars, Light trucks, Motorcycles
- 2. Buses: Transit, Intercity, School
- 3. Medium/heavy trucks
- 4. Air: General aviation, Domestic air carriers, international air carriers
- 5. Water: Freight, Recreational
- 6. Pipeline
- 7. Rail: Freight (Class I), Passenger, Passenger Transit, Passenger Commuter, Passenger Intercity

Necesitamos conocer esta información de 2008 a 2019 según la información contenida en el Transportation Energy Data Book (*TEDB*) de la edición 28 a la edición 40.

ENTREGABLES

- Repositorio: Un repositorio en GitHub o GitLab para los programas de descarga, procesamiento y análisis de la información y salidas de los programas. con una estructura de directorios intuitiva y ordenada. Todos los programas deber ser replicables. El README debe ser autocontenido y tener la claridad suficiente para poder replicar los resultados. (10 puntos)
- Descarga de la información: Un programa que descargue la información directamente de la página donde se alojan los recursos (Estos recursos se encuentran en archivos Excel o PDF).
 (30 puntos)
- 3. **Preprocesamiento de la información**: Un programa que obtenga la información de los recursos descargados y genere las razones de tipo de combustible por modo de transporte de 2008 a 2019. La salida de este programa son dos archivos csv, el primero contendrá la información en formato long y el segundo en formato wide. **(30 puntos)**
- 4. Visualización y análisis de la información: Un programa que genere las visualizaciones para analizar las salidas de la tarea anterior. Analizar los cambios registrados en el consumo de combustible por cada tipo de transporte. Particularmente analiza los cambios en las proporciones de consumo de gasolina y dísel. Este análisis, debe estar incluido en algún lugar del repositorio. (30 puntos)
- 5. **Extra**: Utilizar la información correspondiente en Motor Vehicle Registrations by State and Vehicle Type de la edición 40 de *TEDB* para estimar el consumo de combustible por estado, y realizar algunas visualizaciones. **(20 puntos)**

SOLUCIÓN

Repositorio

Con el fin de almacenar, administrar y compartir los recursos que conforman esta solución, se creo un repositorio en GitHub.

https://github.com/JesusGonzalezMX/prueba 2024 1 solucion.git

Este repositorio tiene la siguiente estructura:



En el folder ".devcontainer" se encuentran las configuraciones necesarias para hacer de este proyecto reproducible, consistente y fácil de compartir, de esto se hablara mas adelante.

En el folder data están los recursos descargados y los archivos generados al procesar estos recursos, en formato long y wide.

Finalmente en el folder scripts, como su nombre lo indica esta el código que da solución a los requerimientos del problema.

Cabe destacar que existe un documento de requerimientos "requirements.txt" que será usado para crear el contenedor con todo lo necesario para correr los scripts de la solución.

Entorno de desarrollo

Para tener una solución exitosa se usó un "Developer container" o "Dev Container", una herramienta que asegura consistencia reproducibilidad y facilita la colaboración. Los Dev containers son una forma de usar Docker para crear ambientes de desarrollo contenidos en un solo espacio de trabajo organizado y colaborativo, con el cual tenemos las siguientes ventajas:

- Consistencia: Estos contenedores aseguran que todos los desarrolladores trabajan con el mismo ambiente sin importar sus configuraciones locales, reduciendo problemas de compatibilidad.
- Aislamiento: Estos contenedores están encapsulados y aislados del sistema principal y de otros contenedores, por lo que los cambios hechos en el contenedor de un proyecto no afectan a los

demás proyectos, es decir que se pueden tener diferentes bibliotecas, aplicaciones y herramientas en cada contenedor.

• Reproducibilidad: Se facilita reproducir el ambiente de trabajo en diferentes máquinas y tener distintas etapas de desarrollo.

Este contenedor es muy fácil e intuitivo de usar desde visual studio code, pues basta con clonar el repositorio allí y automáticamente, los únicos requisitos además de tener visual studio code, es tener instalado Docker y la extensión "Dev containers" para visual studio.

Descarga de la información.

En el menú de la página proporcionada hay un enlace llamado Data y otro llamado Archive.



En Data podemos descargar los datos solicitados como pdf y xls.

Table 2.07	Domestic Consumption of Transportation Energy by Mode and Fuel Type, 2019	xlsx	pdf
Figure 2.06	Domestic Consumption of Transportation Energy by Mode and Fuel Type, 2019	xlsx	pdf

Siendo la figura una grafica y un resumen de la tabla proporcionada, la tabla de 2019 con la información solicitada se muestra a modo de ejemplo a continuación.

		D: 1	Liquefied		D :1 1	N		
	Gasoline	Diesel fuel	petroleum	Jet fuel	Residual fuel oil	Natural	Electricityb	Total
HICHWAY			gas			gas		
HIGHWAY	15,278.3	6,237.2	76.4	-	-	26.7	14.2	21,632.8
Light vehicles	14,640.3	454.5	55.4	-	-	-	14.0	15,164.2
Cars	5,852.8	33.7					12.2	5,898.7
Light trucksd	8,731.5	420.9	55.4				1.8	9,209.5
Motorcycles	56.0							56.0
Buses	10.4	186.3	0.2	-	-	26.7	0.2	223.8
Transit	1.7	60.6	0.2			26.7	0.2	89.5
Intercity		38.9						38.9
School	8.7	86.7						95.4
Medium/heavy trucks	627.6	5,596.4	20.8	-	-	-	-	6,244.8
Class 3-6 trucks	577.4	783.5	20.6					1,381.5
Class 7-8 trucks	50.2	4,812.9	0.2					4,863.3
NONHIGHWAY	194.4	812.5	-	2,347.4	530.5	973.5	109.2	4,967.6
Air	22.6	_	_	2,347.4	-	-	-	2,370.0
General aviation	22.6			204.6				227.1
Domestic air carriers				1,693.1				1,693.1
International air carrierse				449.8				449.8
Water	171.9	309.5	_	_	530.5		_	1,011.9
Freight		268.1			530.5			798.0
Recreational	171.9	41.4						213.3
Pipeline	-	_	_	-	-	973.5	84.6	1,058.1
Rail	_	502.9	_	_	_	_	24.6	527.5
Freight (Class I)		480.4						480.4
Passenger		22.5					24.6	47.
Transit							16.9	16.9
Commuter		14.4					6.1	20.5
Intercity		8.1					1.7	9.8
TOTAL HWY &								
NONHWY	15,472.7	7,049.7	76.4	2,347.4	530.5	1,000.2	123.4	26,600.4

Sin embargo, para los otros años solo se encuentran las versiones anteriores en pdf del documento completo en el menú archive antes mencionado.

2009	Transportation Energy Data Book: Edition 28	ORNL, 4 MB
2010	Transportation Energy Data Book: Edition 29	ORNL, 4 MB
2011	Transportation Energy Data Book: Edition 30	ORNL, 7 MB
2012	Transportation Energy Data Book: Edition 31	ORNL, 6 MB
2013	Transportation Energy Data Book: Edition 32	ORNL, 6 MB
2014	Transportation Energy Data Book: Edition 33	ORNL, 5 MB
2015	Transportation Energy Data Book: Edition 34	ORNL, 15 MB
2016	Transportation Energy Data Book: Edition 35	ORNL, 6 MB
2017	Transportation Energy Data Book: Edition 36	ORNL, 10 MB
2019	Transportation Energy Data Book: Edition 37	ORNL, 13 MB
2020	Transportation Energy Data Book: Edition 38	ORNL, 15 MB
2021	Transportation Energy Data Book: Edition 39	ORNL, 15 MB

Por lo tanto, se creó un script "prueba_2024_1_solucion/scripts/descarga.py" en Python, que corre dentro del contenedor y descarga estos archivos de manera automática y los acomoda dentro del folder "data/raw_data" como archivos pdf.

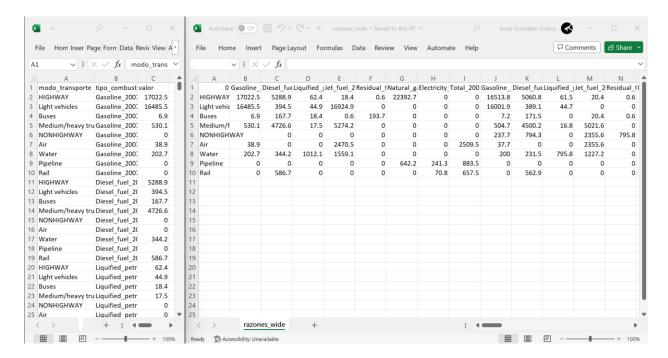
Preprocesamiento de la información

Se creo un script "prueba_2024_1_solucion/scripts/procesamiento.py", que busca en los archivos pdf descargados anteriormente, los (TEDB) de la edición 28 a la edición 40 uno por uno la pagina en la cual esta contenida la tabla que contiene la información del Consumo doméstico de energía de transporte por modo y tipo de combustible, y se realiza un procesamiento del texto encontrado para así obtener los csv de estas tablas, cabe aclarar que se tomaron en cuenta solo los 7 modo de transporte mencionados en la descripción del problema sin ir más allá en los transportes mencionados según cada modo.

Estas tablas se guardan como archivos csv que describen cada año en "prueba_2024_1_solucion/data/Procesamiento_salidas/datos_anuales" como un extra por si se necesitan según los análisis deseados.

azones_2007.csv	azones_2008.csv	azones_2009.csv
azones_2010.csv	razones_2011.csv	azones_2012.csv
razones_2013.csv	razones_2014.csv	azones_2015.csv
razones_2016.csv	razones_2017.csv	azones_2018.csv
razones_2019.csv		

Además se usa esta información para generar los dos archivos csv de salida requeridos almacenados en "prueba_2024_1_solucion/data/Procesamiento_salidas/", el primero en formato long y el segundo en formato wide.



Visualización y análisis de la información

Para la visualización de la información se usa una herramienta que permite visualizar los resultados en el navegador web, de manera local y remota. Las ventajas que presentan estas visualizaciones además de que son mas bonitas, es que son interactivas con el usuario y permiten ser desplegadas de manera fácil y sencilla en internet si así se requiere, ya sea para desarrollar una aplicación o hacer un demo o presentación.

La herramienta usada es streamlit y el código esta en "prueba_2024_1_solucion/scripts/visualizacion.py".

```
vscode →/workspaces/prueba_2024_1_solucion (main) $ streamlit run scripts/visualizacion.py

Collecting usage statistics. To deactivate, set browser.gatherUsageStats to False.

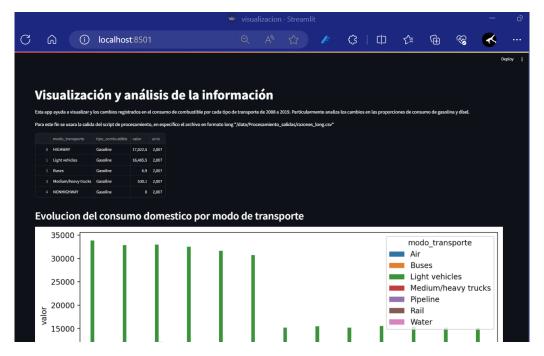
You can now view your Streamlit app in your browser.

Network URL: http://172.17.0.2:8501

External URL: http://201.141.19.6:8501
```

El script se corre de manera sumamente sencilla como se muestra en la imagen anterior y automáticamente podemos ver el resultado en el navegador.

Como nota importante esta que el contenedor usado es muy sencillo dado el tiempo, por lo que esas direcciones que aparecen en la terminal no podrán ser usadas, pero se puede ver la visualización en localhost sin ningún problema como se muestra en la siguiente imagen.



Además streamlit permite guardar estas visualizaciones para poder analizarlas sin tener que correr el script, estas visualizaciones están en "prueba_2024_1_solucion/docs/Solucion a la prueba – documentacion.pdf".

Nota final

En esta solución se mostro el uso de herramientas avanzadas que se usan en la industria pero que pueden ser usadas también para la investigación potenciando así los proyectos, colaboraciones y beneficios obtenidos. Se mostro el uso de los dev containers para la reproducibilidad, lo fácil que es colaborar y compartir el trabajo usando un repositorio y docker, además de una forma interesante de generar las visualizaciones, las cuales pueden ser interactivas, fáciles de compartir incluso de manera remota, exportar para integrarlas en la documentación y muy bonitas.

Además se usaron herramientas comunes en el análisis de datos, como Python, pandas, seaborn, VS Code, GitHub, etc.

Se trabajo de manera rápida por el tiempo pero dando la sensación de un trabajo completo, bien realizado, profesional y escalable.

El punto extra quedo fuera de la solución por el poco tiempo.