

# Investigación aplicada utilizando datos públicos del Ministerio de Salud de Brasil: Análisis histórico del período de 2014 a 2022.

Lucas E. Zonfrilli<sup>1</sup>, Matheus S. Soares<sup>2</sup>, Me. Antonio R. Pepece Jr<sup>3</sup>, Me. Fábio E. Cardoso <sup>4</sup>

<sup>1</sup>Graduación en Gestión de Tecnologías de la Información Fatec de Assis, [lucas.zonfrilli@fatec.sp.gov.br](mailto:lucas.zonfrilli@fatec.sp.gov.br)

<sup>2</sup>Graduación en Gestión de Tecnologías de la Información Fatec de Assis, [matheus.soares29@fatec.sp.gov.br](mailto:matheus.soares29@fatec.sp.gov.br)

<sup>3</sup>Asesor Profesor Fatec Assis, [antonio.pepece@fatec.sp.gov.br](mailto:antonio.pepece@fatec.sp.gov.br)

<sup>4</sup> Codirector Profesor Fatec Assis, [fabio.eder6@fatec.sp.gov.br](mailto:fabio.eder6@fatec.sp.gov.br)

## RESUMEN

Este artículo muestra que es posible utilizar datos públicos que permitan el cálculo de modelos estadísticos, además de posibilitar el cálculo de modelos estadísticos. El Portal de Transparencia del gobierno brasileño es una herramienta útil para acceder y monitorear los gastos e ingresos del gobierno brasileño. El objetivo es proporcionar medios para acceder a datos públicos disponibles en las API oficiales del gobierno brasileño y realizar cálculos utilizando modelos estadísticos, como "Correlación lineal simple" y "Regresión lineal simple", utilizando el lenguaje de programación Python. Se describe el proceso de recolección de datos utilizando API oficiales del gobierno, extracción de datos relacionados con los gastos del Ministerio de Salud y considerando el número de camas de hospitalización disponibles en el sistema público de salud del 2014 al 2022. Se presentan los métodos estadísticos utilizados y cómo se asociará el lenguaje Python. Estos elementos y calcular modelos estadísticos para reproducir el resultado. Esta investigación hace que la información del pasado sea accesible y transparente para construir un futuro abundante, y brinda a las personas y empresas la capacidad de utilizar el lenguaje Python para obtener información de datos gubernamentales abiertos para calcular indicadores estadísticos para la toma de decisiones. El estudio utilizó datos públicos y una API bien documentada, superando algunas limitaciones técnicas. El objetivo se logró exitosamente y el código está disponible en GitHub, concluyendo que es posible extraer datos públicos y evaluar modelos estadísticos.

**Palabras clave:** API; lenguaje Python; Regresión lineal.

## 1. INTRODUCCIÓN

Actualmente vivimos una época en la que recibimos miles y miles de información desde redes sociales, medios digitales, TV, radio, entre otros medios. Saber analizar y separar la información verdadera de la información falsa, también conocida como "*fakenews* <sup>1</sup>", es complejo, especialmente cuando se trata de datos públicos relacionados con el gasto gubernamental en beneficio de la sociedad.

El Portal de Transparencia del Gobierno Federal <sup>2</sup> es una herramienta útil para que la población acceda y controle los gastos e ingresos del Gobierno brasileño. A

---

<sup>1</sup>Neologismo, utilizado para referirse a noticias fabricadas o falsas (HUTTNER, 2020).

<sup>2</sup><https://api.portaldatransparencia.gov.br/swagger-ui.html>

través de este portal es posible consultar gastos en diferentes áreas, economía, seguridad social, educación, seguridad, salud, entre otras. Además, existe la posibilidad de acceder a información detallada de cada gasto realizado. Así, a través de este portal, la población puede consultar dónde se aplicaron los recursos financieros puestos a disposición por el gobierno federal.

En este sentido, es necesario que cualquier ciudadano tenga acceso a los datos disponibles en el citado Portal, para que pueda consultar y analizar los gastos según las necesidades de la población. De esta manera, es posible señalar dónde hay necesidad de mejoras en los servicios públicos, además de identificar posibles desvíos de recursos y aplicaciones inapropiadas.

En este sentido, el artículo tiene como objetivo proporcionar medios para consultar datos públicos de las API (Interfaz de programación de aplicaciones) oficiales del Gobierno Federal de Brasil. Posteriormente realizar cálculos que utilizan modelos estadísticos como el uso de “Correlación Lineal Simple” y “Regresión Lineal Simple”, utilizando el lenguaje de programación Python.

Por tanto, será necesario contextualizar los portales de datos públicos disponibles, también conocidos como API (Application Programming Interface). Presentar los modelos estadísticos Correlación Lineal Simple y Regresión Lineal Simple. Conceptualizar el lenguaje de programación Python y presentar las bibliotecas de lenguajes que utilizaremos.

A continuación, se describe todo el proceso de recolección de datos a través de la API oficial del Gobierno Federal, en el Portal de Transparencia y Portal DataSUS; Se considera la extracción de datos relativos a gastos del Ministerio de Salud y, como parámetro temporal, el número de camas de hospitalización disponibles por el SUS de 2014 a 2022. De esta forma, luego de recolectar todos los datos, se aplica el modelo antes mencionado. matemático, junto con las bibliotecas Pandas, Matplotlib, Scikit-learn y Numpy.

Uno de los vacíos de investigación detectados es cómo brindar acceso a la información de manera más precisa y con mayor facilidad de interpretación. Compartir información sin análisis objetivo y de difícil verificación, así como evaluar el desempeño en función del gasto y no de la eficiencia, hace de este trabajo una alternativa viable.

En la era de la información, hacer que la información del pasado sea accesible y transparente, completa y clara es necesario para construir un futuro abundante.

Además, poner a disposición de personas y empresas la posibilidad de utilizar el lenguaje Python para obtener información de datos gubernamentales abiertos con el fin de calcular indicadores estadísticos para obtener información para la toma de decisiones.

## **2. MATERIALES Y MÉTODOS**

Para organizar y desarrollar los objetivos de este artículo, se presentará cómo ocurre la correlación y regresión lineal simple en el uso de datos estadísticos, la búsqueda de datos a través del Portal DataSUS y la API del Portal de Transparencia y cómo el lenguaje Python asociará estos ítems. y calcular modelos estadísticos y reproducir el resultado.

### **2.1 Métodos estadísticos**

### 2.1.1 Correlación lineal simple

Según Bussab y Morettin (2017), la correlación lineal es una función que determina la relación entre dos variables, se pueden representar mediante pares ordenados (X,Y), siendo X la variable exploratoria y Y la variable dependiente. Esta correlación tiene como objetivo indicar la fuerza de la asociación entre las variables X e Y.

Cuando se representan en un diagrama de dispersión, los datos se distribuyen alrededor de una línea imaginaria, por lo que se puede decir que visualmente los datos sugieren una correlación lineal.

Al calcular el coeficiente de correlación lineal se obtiene el grado de relación entre las variables, estos grados oscilan entre -1 y 1, y cuanto más cerca de -1 la correlación es más débil entre las variables, si es cercana o igual a 0 se muestra una correlación nula y cercana a 1, la relación se considera fuerte o perfecta si es igual a 1, como se ilustra en la figura 1.

Figura 1 - Fórmula utilizada para obtener el coeficiente (r).

$$r = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{\sqrt{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \sqrt{n \sum_{i=1}^n y_i^2 - (\sum_{i=1}^n y_i)^2}}$$

Si se confirma la correlación entre las variables, agudizando el experimento del científico, surge la necesidad de una función matemática que describa esta relación en una línea de tendencia recta, es decir, una ecuación de regresión.

### 2.1.2 Regresión lineal simple

Según estudios de HOFFMANN (2006), la regresión lineal simple es una técnica muy utilizada en estadística y econometría para realizar predicciones de una variable x a través de una variable y, es necesario conocer dos características importantes para realizar la recta de regresión, una es la coeficiente angular de la línea y el otro es el coeficiente lineal de la línea. La Figura 2 ilustra la fórmula para obtener la recta de regresión lineal.

Figura 2 – Fórmula para obtener regresión lineal simple.

$$\hat{y} = aX + b$$
$$a = \frac{n \cdot \sum x \cdot y - \sum x \cdot \sum y}{n \cdot \sum x^2 - (\sum x)^2}$$
$$b = \frac{\sum y}{n} - a \cdot \frac{\sum x}{n}$$

El coeficiente lineal de la línea está representado por α, b es el coeficiente angular y N es el tamaño de la muestra. La variable x funciona como dato predictivo, por lo que debe ser fácilmente accesible y la variable menos accesible. Según HOFFMANN (2006), es importante tener en cuenta que no todas las situaciones

quedan bien aproximadas mediante una ecuación lineal, por lo que es necesario calcular la correlación lineal y presentar los datos en un diagrama de dispersión para verificar que los valores de  $x$  e  $y$  tienen una predisposición lineal.

## 2.2 API (Interfaz de programación de aplicaciones)

API en su origen significa *Interfaz de Programación de Aplicaciones*, traduciendo al portugués significa Interfaz de Programación de Aplicaciones, y según Jacobson, Brail y Woods (2012), corresponde a un conjunto de software con la función de intermediar la comunicación de datos entre aplicaciones. A través de una API es posible que una o más aplicaciones se comuniquen de forma transparente y ágil para el usuario, como si de una sola aplicación se tratara.

Actividades del día a día como: publicar una foto en Instagram y también enviarla a Facebook, consultar su saldo bancario, pasar su tarjeta de crédito, son ejemplos del uso de una API. Su integración agrega interacción a nivel de datos donde se envían o reciben datos entre las aplicaciones involucradas.

Para una organización, una API es un punto de conexión entre ella y sus socios y clientes, compartiendo procesos comerciales, servicios y contenido; conectando equipos de trabajo internos con desarrolladores independientes, de forma fácil y segura (RICHARDSON Y RUBY, 2007).

El propósito de desarrollar una API puede ser que la utilice un grupo restringido de usuarios o puede desarrollarse pensando en el público en general. Así, las API pueden clasificarse como privadas o públicas (DOERRFELD, 2016), y corresponde a la organización evaluar y decidir el tipo de API que pretende utilizar, en función de su finalidad y objetivos.

Los dos principales estilos arquitectónicos que han dominado el mercado en los últimos años son: REST y SOAP. Sin embargo, es importante resaltar que el significado mismo de SOAP deja clara la intención de ser un protocolo mientras que “REST se define únicamente como un estilo arquitectónico” (FIELDING, 2014), lo que significa que no corresponde establecer una comparación directa entre ambos estilos.

Para llevar a cabo la comunicación entre diferentes aplicaciones y una API es necesario contar con estándares de comunicación y la definición del formato en el que se transmitirán los datos. Actualmente existen dos formatos muy utilizados para transportar y presentar datos a través de Internet: 1) formato JSON ( *Java Script Object Notation* ); y 2) formato XML ( *lenguaje de marcado extensible* ).

El formato JSON tiene la característica de estar disponible en un tamaño más pequeño que XML debido a la mayor cantidad de *etiquetas* (caracteres de formato especial) que se utilizan al escribir un archivo XML, y el formato JSON ha sido el formato de presentación de datos elegido en la mayoría de las API desarrolladas.

### 2.2.1 API del Portal de Transparencia

Según el Portal de Transparencia del Gobierno Federal, para utilizar la API de datos puesta a disposición por el Portal de Transparencia, es necesario registrar primero una dirección de correo electrónico en <http://portaldatransparencia.gov.br/api-de-dados/cadastrar-email>. Luego del registro se enviará un token al correo electrónico proporcionado, el cual deberá ser utilizado en las consultas realizadas a través de la API.

Se puede acceder a los datos de los servicios ofrecidos por la API a través de solicitudes HTTP a servicios REST. La documentación API completa está disponible en <http://api.portaldatransparencia.gov.br/>. En la página "API de datos - Ejemplos de uso" puede encontrar ejemplos de código fuente para consumir la API en varios lenguajes de programación, como Javascript, Java, PHP y .NET.

Es importante resaltar que el uso de la API de datos del Portal de Transparencia del Gobierno Federal es gratuito y de uso abierto, de conformidad con lo dispuesto en el Decreto N° 8.777, de 11 de mayo de 2016.

## 2.2.2 DATOS

Según el sitio DataSUS, Tabnet es un portal que proporciona información relevante para análisis objetivos de la situación de salud, toma de decisiones basadas en evidencia y desarrollo de programas de acción en salud.

El sitio web proporciona datos sobre mortalidad, morbilidad, acceso a servicios, calidad de la atención, condiciones de vida y factores ambientales, que se utilizan para construir indicadores de salud. Además, también se encuentra información sobre atención de salud a la población, registros de redes hospitalarias y ambulatorias, registros de establecimientos de salud, recursos financieros e información demográfica y socioeconómica.

Para consultar los datos, puede acceder al enlace: <https://datasus.saude.gov.br/informacoes-de-saude-tabnet/> y seleccionar la categoría que desee. En el presente estudio se seleccionará: Red de Atención, después CNES<sup>3</sup>- Recursos Físicos y finalmente seleccionar Hospital - Camas de Internación\*.

El sistema lo dirigirá al enlace: <http://tabnet.datasus.gov.br/cgi/defthtm.exe?cnes/cnv/leiintbr.def>. Donde se filtra la muestra de datos que serán analizados. Luego filtre por "Línea": Año/mes en competencia. en "Columna" no activa, "Contenido" Cantidad SUS y en "Periodo disponible" seleccione enero de 2014 hasta diciembre de 2022.

Finalmente, simplemente haga clic en el botón Mostrar y los datos se generarán y guardarán en .xlsx.

Para finalizar los ajustes para realizar la Correlación Lineal Simple y la Regresión Lineal Simple, se calculó el valor promedio anual, ya que los datos se dieron mensualmente.

## 2.3 Lenguaje Python

El lenguaje elegido para aplicar el experimento es Python, este se eligió porque es de muy alto nivel, tiene un tipado dinámico y fuerte, además de estar orientado a objetos. Otra ventaja es que Python es de código abierto, lo que significa que puede incorporarse a otros productos de forma gratuita. Según el Índice TIOBE, Python es hoy el lenguaje más utilizado en el mundo, representando el 14,83% de la *cuota de mercado*.

Según McKinney (2018), "en los últimos diez años, Python ha pasado de ser un lenguaje informático científico innovador, o de uso "bajo su propia responsabilidad", a uno de los lenguajes más importantes en ciencia de datos, el aprendizaje automático. ( *Machine Learning* ) y desarrollo de software en general, en

---

<sup>3</sup>Registro Nacional de Establecimientos de Salud de Brasil

el ámbito académico y en el mercado. ”, en este trabajo se utilizan bibliotecas de ciencia de datos disponibles para lograr el objetivo.

Para realizar operaciones matemáticas complejas entre *matrices* y funciones lineales se utilizará la biblioteca Numpy. También será utilizado como contenedor de datos para su manipulación durante el desarrollo del trabajo.

Gran parte de la popularidad de Python en la ciencia de datos se debe a la biblioteca Pandas, que trae una gama de herramientas para manipular, seleccionar y procesar información en alto rendimiento. Funciona con una estructura tabular (tablas) llamada Dataframe indexada por columnas y filas. Después de tratar y transformar los datos surge la necesidad de presentar estos datos de forma visual, para ello usaremos matplotlib. Es una biblioteca para trazar y visualizar datos en forma bidimensional, se usa ampliamente y su popularidad la ha convertido en la herramienta estándar para la visualización de datos.

### 3 RESULTADOS Y DISCUSIONES

Para una mejor organización del código, se separó en dos etapas en diferentes archivos denominados: “ETL” y “PLOTAGE”.

En la etapa ETL, el código se divide en tres partes: importar las bibliotecas necesarias, definir la función `get_variables()` y llamar a la función. En este contexto, el código se comenta línea por línea dentro de la función `get_variables()`, como se muestra en las figuras 3, 4 y 5.

Figura 3 – Importación de bibliotecas

```
ETL.py > get_variables
1  import requests
2  import pandas as pd
3  import json
4  import time
```

Fuente: Autores

Figura 4 - Definición de la función `get_variables()`.

```
6  def get_variables():
7      api_key = 'insira sua chave aqui'
8
9      dataframes = []
```

Fuente: Autores

La función `get_variables()` está definida y una clave API está definida en `api_key`. La variable de marcos de datos se inicializa en una lista vacía.

Figura 5 – Código de bucle para obtener datos.

```

11     for ano in range(2014, 2023):
12         url = f'https://api.portaldatransparencia.gov.br/api-de-dados/despesas/por-orgao?ano={ano}&orgao=36000&pagina=1'
13         print(url)
14         headers = {"chave-api-dados": api_key}
15
16         response = requests.get(url, headers=headers)
17
18         if response.status_code == 200:
19             # A requisição foi bem-sucedida
20             print(response.text)
21             data = json.loads(response.text)
22             df = pd.DataFrame(data)
23             dataframes.append(df)
24             time.sleep(120)
25             # Faça o que quiser com os dados da API
26         else:
27             # A requisição falhou
28             print(f"A requisição falhou com código de status {response.status_code}.")
29

```

Fuente: Autores

Se crea un bucle "for" para iterar desde 2014 hasta 2022 (años incluidos). Para cada iteración, se crea una URL utilizando el año actual y una variable de página de 1. La URL se imprime en la pantalla y se configuran los encabezados para la solicitud de API. La función request.get() se utiliza para realizar la llamada a la API y la respuesta se almacena como respuesta.

Si la respuesta tiene un status\_code de 200, se ejecuta el bloque de código dentro de if. La respuesta se imprime en la pantalla, la respuesta en formato JSON se carga en datos y luego se convierte en un DataFrame de pandas en df. df se agrega a la lista de marcos de datos y la ejecución del código se pausa durante 2 minutos usando time.sleep(120), cumpliendo con los requisitos de la API del Portal de Transparencia del Gobierno Federal.

Si la respuesta tiene un código de estado distinto de 200, el bloque de código dentro del else se ejecuta y se imprime un mensaje de error en la pantalla.

Figura 6 – Código para concatenar Dataframes.

```

31     df = pd.concat(dataframes, ignore_index=True)
32
33     x = df['pago'].str.replace('.', '').str.replace(',', '.').astype(float).values
34
35     leitons = pd.read_excel("Leitons.xlsx")
36     y = leitons['Quantidade_SUS'].values
37
38     return x, y
39

```

Fuente: Autores

Luego de la etapa ETL (extracción, almacenamiento y modelado de datos), comienza el cálculo estadístico de Correlación Lineal Simple y Regresión Lineal Simple, que se denomina PLOTTING.

En esta etapa, el código también se divide en tres bloques principales: importar bibliotecas, procesar datos y trazar el diagrama de dispersión con la línea de regresión lineal, como se ilustra en las figuras 7, 8 y 9.

Figura 7: Importación de bibliotecas sklearn, matplotlib y ETL

```

PLOTAGEM.py > ...
1     from sklearn.linear_model import LinearRegression
2     import matplotlib.pyplot as plt
3     import ETL

```

Fuente: Autores



Se importan las bibliotecas sklearn, matplotlib y ETL. El primero importa la clase LinearRegression de la biblioteca scikit-learn, que se usa para calcular la línea de regresión lineal, matplotlib se usa para trazar gráficos y ETL importa el módulo ETL, que contiene la función get\_variables() que se usará para obtener las variables x e y.

Figura 8: Llamada a la función “get\_variables()” del módulo ETL.

```
5 # Obtendo as variáveis x e y da função ETL.get_variables()
6 x, y = ETL.get_variables()
7
8 # Calculando a linha de regressão linear
9 reg = LinearRegression().fit(x.reshape(-1, 1), y)
10 r = reg.coef_[0]
```

Fuente: Autores

Se llama a la función get\_variables() del módulo ETL, la cual devuelve las variables x e y obtenidas de un proceso de Extracción, Transformación y Limpieza de datos (ETL);

Después de este procedimiento, el objeto LinearRegression de Scikit-learn se utiliza para calcular la línea de regresión lineal. Para ello, utiliza el método fit(), que recibe los valores de x e y y ajusta una línea de regresión lineal.

A continuación, se calcula el coeficiente de regresión lineal (r) a partir del objeto LinearRegression, que representa la pendiente de la línea de regresión.

Figura 9 – Cálculo de regresión lineal

```
12 # Plotando o gráfico de dispersão com a linha de regressão linear
13 plt.scatter(x, y)
14 plt.plot(x, reg.predict(x.reshape(-1, 1)), color='red')
15 plt.xlabel('x')
16 plt.ylabel('y')
17 plt.title(f"Gráfico de dispersão com linha de regressão linear (r={r:.2f})")
18 plt.show()
```

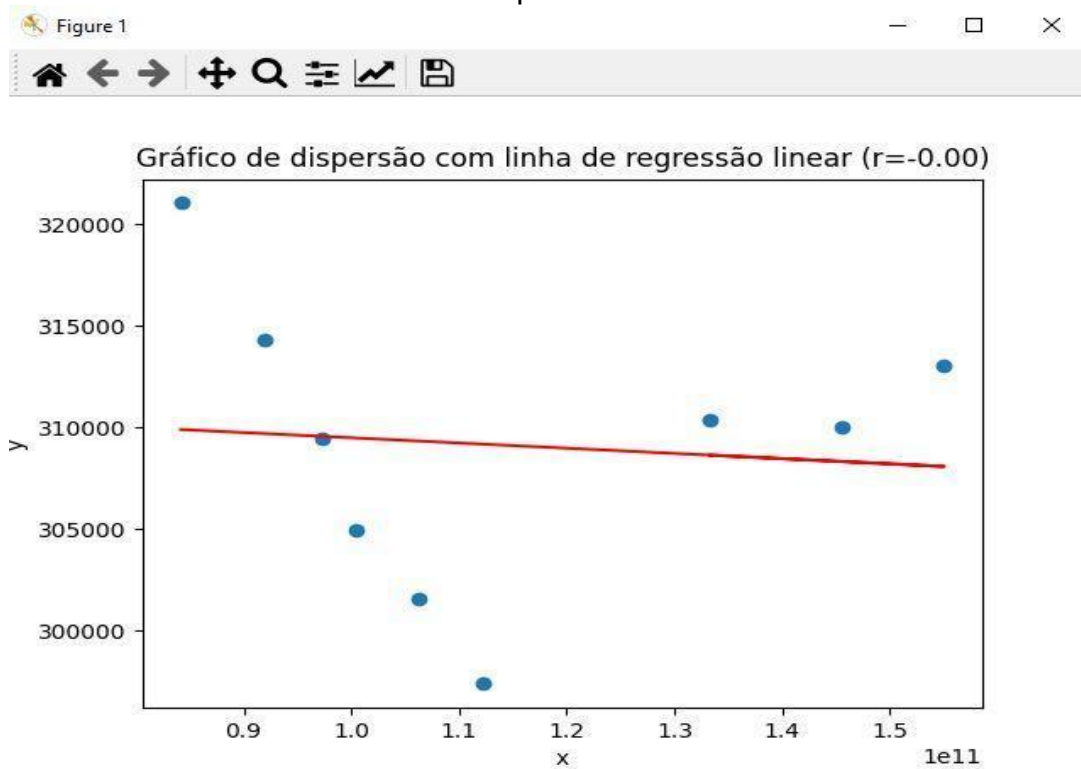
Fuente: Autores

Finalmente, el diagrama de dispersión se traza usando el método scatter() de la biblioteca matplotlib, pasando las variables x e y como parámetros, y la línea de regresión lineal se traza usando el método plot(), pasando los valores x e y como parámetros, junto con la línea de regresión calculada a partir del método predict() del objeto LinearRegression. Consecutivamente se añaden títulos de ejes y un título general para el gráfico.

Al llamar al método show() del objeto pyplot, el gráfico se muestra en la pantalla, como se muestra en la figura 10.



Figura 10 – Diagrama de dispersión



Fuente: Autores

En el gráfico anterior, el eje “y”, que representa la variable dependiente, es el número de camas de hospital disponibles. En el eje “x”, los Gastos totales del Ministerio de Salud de Brasil. Este gráfico muestra la dispersión de estos datos distribuidos entre 2014 y 2022 y calcula una línea de tendencia mediante el cálculo de Regresión Lineal Simple. Es posible evaluar el índice de correlación utilizando el indicador “r”, que se muestra después del título del gráfico.

#### 4 ANÁLISIS DE DATOS

El 26 de febrero de 2020, el Hospital Albert Einstein de São Paulo confirmó el primer caso de COVID-19 en Brasil. Desde entonces, muchos cambios estructurales que antes no eran tan importantes ahora se han convertido en una prioridad en la agenda del Ministerio de Salud de Brasil.

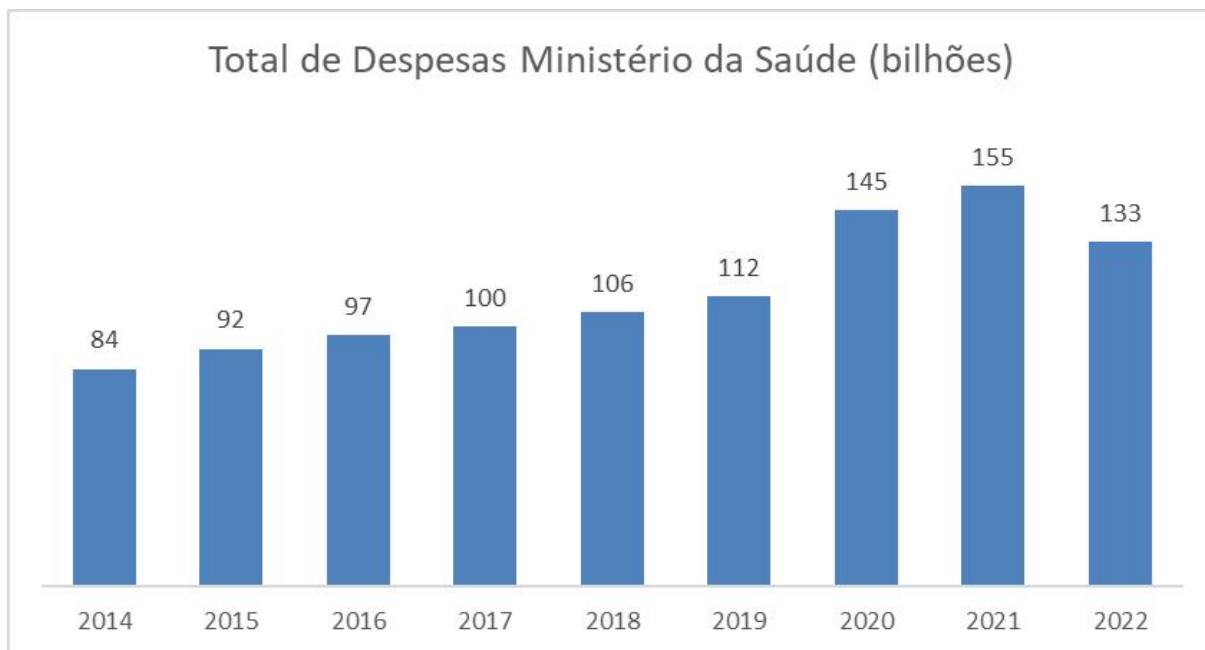
Por tanto, el análisis se realizará desde el período 2014 a 2022, dividido en dos períodos de tiempo, período prepandémico 2014 a 2019, y período pandémico 2020 a 2021.



Fuente: Autores

El pico de disponibilidad de camas ofrecidas por el Sistema Único de Salud (SUS) de Brasil se produjo en 2014, con más de 320 mil camas disponibles. A continuación se muestra una fuerte caída del 7,37% de 2014 a 2019, el período anterior al Covid19.

Impulsado por el Covid19 con 310 mil camas disponibles, en 2020 se registró un crecimiento del 4,26% respecto a 2019. Con el agravamiento de la pandemia, en 2021 continuó un ligero crecimiento, estabilizándose nuevamente en 2022.



Fuente: Autores

A diferencia del Total de Camas Disponibles, el valor de los Gastos del Ministerio de Salud del Gobierno brasileño tuvo un crecimiento gradual de 2014 a 2021, donde alcanzó su punto máximo con 155 mil millones de reales gastados por el Ministerio de Salud. En 2022, una caída de aproximadamente 14%, poco comparado con el crecimiento del 58,4% considerando 2014 y 2022.

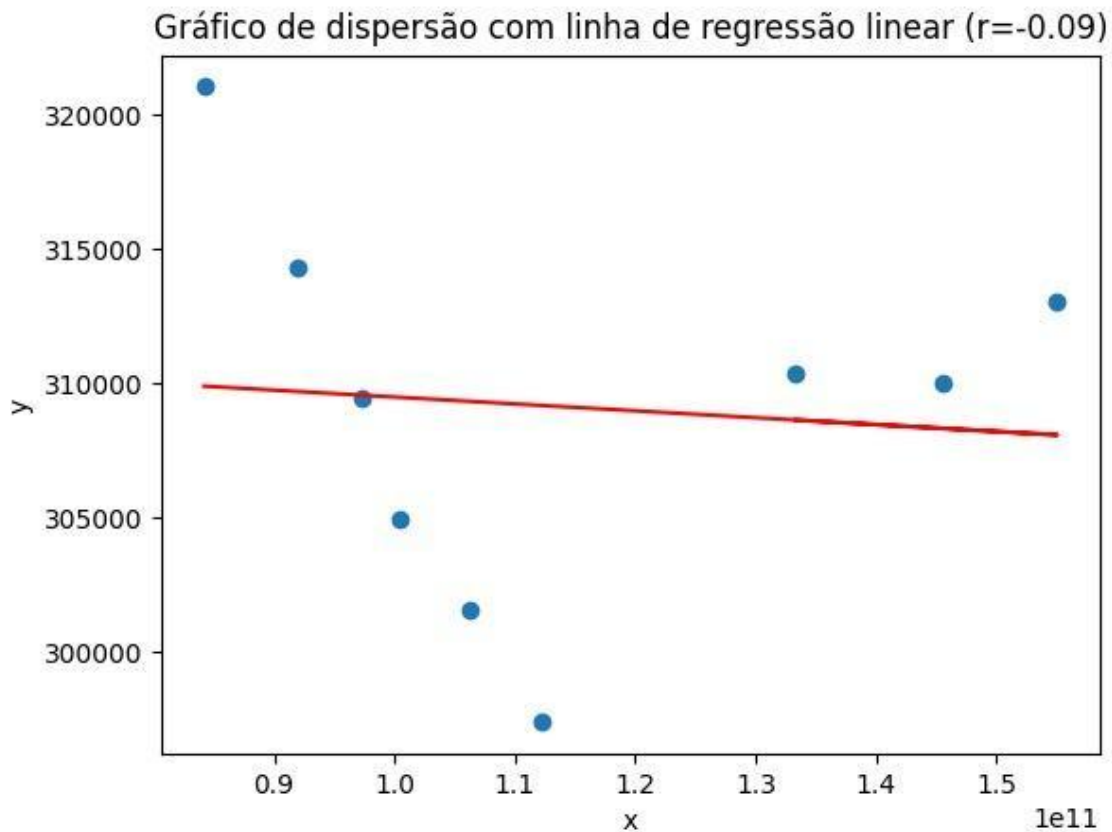


Fuente: Autores

En este análisis se calcula un índice, representado por la división de los Gastos Totales del Ministerio de Salud entre el Total de Camas Disponibles por el SUS, y de forma genérica se analiza el costo de cada habitación disponible. Es visible la tendencia ascendente hasta 2021, alcanzando el coste anual de 495 mil reales por

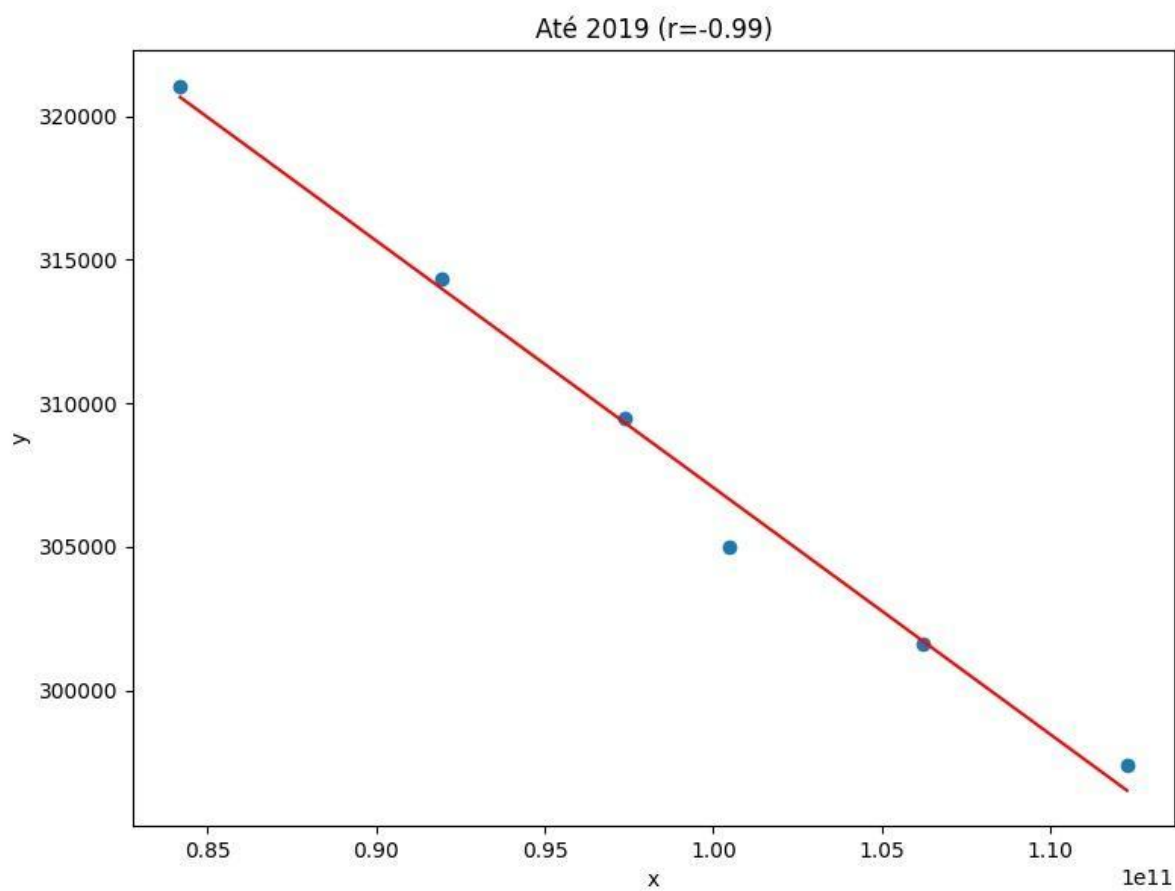
cama disponible. Al comparar 2014 con su pico de gasto en 2021, hubo un aumento de alrededor del 88,9% en el valor comparativo.

### Correlación y regresión lineal simple



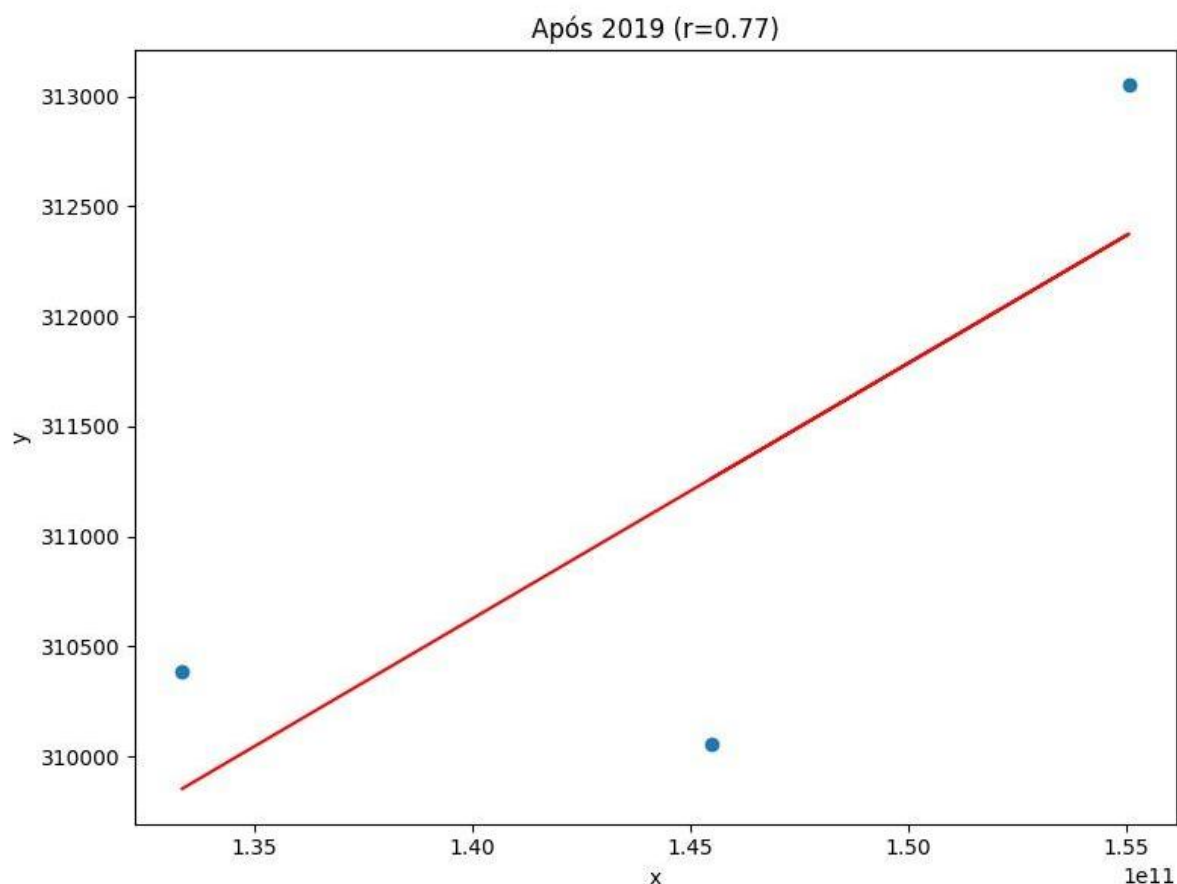
Fuente: Autores

Considerando los puntos de 2014 a 2019, se ve su fuerte tendencia a la baja, y a partir de 2020 hay una reversión del movimiento, donde se puede deducir el enfoque del Gobierno en la creación de más camas, campaña de hospitales para atender las demandas del COVID19. Debido a este factor de priorización, al analizar el período total la correlación es nula. Para mejorar el análisis y la experimentación, la regresión se realiza en dos períodos diferentes, prepandemia 2014 a 2019, y pandemia, 2020 a 2022, a continuación en los siguientes gráficos.



Fuente: Autores

Al analizar el periodo prepandemia (2014-2019) se obtiene una correlación negativa perfecta del 99%, mostrando una fuerte relación entre las variables. Lo sorprendente es que al mirar se nota que la correlación es negativa o inversamente proporcional. Es decir, cuanto mayor sea el monto gastado por el Ministerio de Salud, menor será el número de camas disponibles en el mismo período.



Fuente: Autores

Con el inicio de la pandemia en 2020 y su agravamiento en 2021 y 2022, se obtiene una inversión de la pendiente de la línea, manteniendo un fuerte grado de correlación entre las variables con alrededor del 77%, la línea toma una tendencia positiva, es decir, cómo a mayor valor de los Gastos del Ministerio de Salud, mayor será el número de Camas Disponibles.

Es posible concluir que durante el período prepandemia analizado (2014 a 2019) la provisión de camas por parte del SUS no era una prioridad para el Ministerio de Salud de Brasil, sin embargo este escenario cambia tras el descubrimiento y proliferación de la COVID19 en territorio brasileño. , impulsado por el aporte de capital destinado a Gastos del Ministerio de Salud combinado con una priorización de emergencia, se invierte la tendencia de la línea recta, haciéndola proporcionalmente positiva.

## 5 CONSIDERACIONES FINALES

La facilidad de acceso a los datos públicos fue esencial para realizar el estudio. El uso de datos públicos permitió desarrollar el trabajo sin necesidad de recursos económicos ni autorización de terceros, lo que hizo que el proyecto fuera viable y accesible.

La documentación de la API utilizada para acceder a datos públicos en este trabajo fue excelente. La información estaba bien organizada y las funcionalidades de la API estaban bien descritas, lo que permitió una fácil comprensión de cómo utilizar la herramienta.

Aunque enfrentamos algunas dificultades, como el límite de una solicitud por minuto y la restricción de información de solo un año a la vez, logramos superarlas insertando un tiempo de espera adecuado entre solicitudes. Estas limitaciones no impidieron el éxito del trabajo, que fue posible gracias a los métodos de ciencia de datos utilizados en el análisis de los datos recopilados.

Se logró con éxito el objetivo del trabajo. Utilizando el modelo estadístico de regresión lineal, fue posible calcular y reproducir los datos recopilados. El resultado obtenido confirmó la hipótesis inicial del estudio y demostró que los datos públicos se pueden utilizar de manera eficiente en el análisis de datos.

El código desarrollado para este trabajo está disponible en un repositorio en GitHub <sup>4</sup>. Este repositorio contiene todo el código utilizado en el análisis de datos, así como las bibliotecas utilizadas y los datos recopilados.

Por lo tanto, se concluye que el uso de datos públicos en proyectos de ciencia de datos es una estrategia viable y eficiente para la obtención de información valiosa, siempre y cuando se utilicen métodos y técnicas apropiadas para su extracción, modelado y reproducción.

## 6 REFERENCIAS

**DATOS** . Ministerio de Salud Disponible en: <<https://datasus.saude.gov.br/informacoes-de-saude-tabnet/>>. Consultado el: 14 de abril. 2023.

DOERRFELD, B., WOOD, C., ANTHONY, A, SALDOVAL, K. LAURET, A. **La economía API: disrupción y el negocio de las API** 1ed. API nórdicas , 2016.

FIELDING, R. y Reschke, J. **Protocolo de transferencia de hipertexto (http/1.1):** sintaxis y enrutamiento de mensajes. RFC 7230, Solicitudes de comentarios en Internet, Editor de RFC, 2014.

HOFFMANN, R. **Análisis de regresión: una introducción a la econometría** . São Paulo: Editora Hucitec, 2006.

HUTTNER, Luiz Ricardo Goulart. **¿Son noticias falsas?: cómo se utilizan elementos del periodismo para crear “noticias falsas”** . Tesis de maestría. Universidad Federal de Rio Grande do Sul, Porto Alegre, 2020.

JACOBSON, D., BRAIL, F., WOODS, D. **API: una guía estratégica**. 1 ed. Gravenstein Highway North, Sebastopol, Estados Unidos: O'Reilly, 2012.

MCKINNEY, Wes. **Python para análisis de datos: procesamiento de datos con Pandas, NumPy e IPython** . São Paulo: Novatec, 2018.

MORETTIN, Pedro Alberto; BUSSAB, Wilton O. **Estadística básica** . 9. ed. São Paulo: Saraiva, 2017. 468 p. ISBN 978-85-472-2023-5.

---

<sup>4</sup><https://github.com/MatheusSouzaSoares/USO-DE-DADOS-ABERTOS-APLICADOS-A-MODELOS-ESTAT-ISTICOS-UM-ESTUDO-DE-CASO-A-PARTIR-DO-DATASUS>



**PORTAL DE TRANSPARENCIA.** Gobierno federal. Disponible en: <<https://portaldatransparencia.gov.br/>>. Consultado el: 14 de abril. 2023

RICHARDSON, L., RUBY, S. **Servicios web RESTful**. 1 ed. Gravenstein Highway North, Sebastopol, EE.UU. O'Reilly, 2007.

**ÍNDICE DE TIOBE.** Software de Tlobe. Disponible en: <<https://www.tiobe.com/tiobe-index/>>. Consultado el: 05 de abril. 2023.

## **AGRADECIMIENTOS**

Nos gustaría expresar nuestro más sincero agradecimiento a todos los que contribuyeron al éxito de este artículo. En primer lugar, agradecemos a Dios por el regalo de la vida y por brindarnos la oportunidad de aprender y crecer durante todo el proceso. También queremos agradecer a nuestras familias por su apoyo y apoyo incondicional, quienes estuvieron a nuestro lado durante todo el viaje.

No podemos dejar de reconocer la importancia de nuestros docentes, quienes nos guiaron y motivaron en cada etapa del trabajo. Le agradecemos su dedicación, paciencia y conocimiento compartido. Sin el valioso aporte de cada uno de ellos, no habiéramos logrado nuestros objetivos.

Finalmente, nos gustaría agradecer a todos los que nos apoyaron y alentaron a lo largo del camino. Sabemos que este trabajo fue resultado de un esfuerzo conjunto y estamos inmensamente agradecidos de ser parte de tan increíble equipo. ¡Agradecido!