

Telecomunicaciones : identificar operadores ineficientes

Proyecto Final



Presented By : Samuel Luis Juan



Problema: Ineficiencia de Operadores

El servicio de telefonía virtual CallMeMaybe está desarrollando una función para identificar a los operadores menos eficaces. Un operador se considera ineficaz si::

- Tiene muchas llamadas entrantes perdidas (internas y externas).
- Presenta tiempos de espera prolongados para las llamadas entrantes.
- Realiza pocas llamadas salientes, si se espera que haga este tipo de llamadas.

Proseso de análisis de datos

Paso 1

Se abre el archivo de datos, se revisa la información general y se preparan los datos renombrando las columnas y verificando los tipos de datos y valores ausentes para corregirlos si es necesario.

Paso 2

Identificar operadores ineficaces mediante el análisis de sus métricas de desempeño. Primero, evaluamos la Tasa de Llamadas Perdidas (LCR) y el Tiempo Promedio de Espera (ACD), destacando a los operadores con altos tiempos de espera. También analizamos la Tasa de Llamadas Internas (ICR), la Eficiencia de Llamadas Salientes (OCE) y la Tasa de Llamadas Exitosas (CSR). Con estos datos, generamos métricas clave (KPIs) para clasificar la ineficiencia y medir el desempeño de cada operador en términos de eficiencia..

Paso 3

Se realizan pruebas estadísticas para comparar los operadores eficaces e ineficaces en cuanto a tiempos de espera, porcentaje de llamadas perdidas y cantidad de llamadas salientes. Se verifica si las diferencias son estadísticamente significativas y se documentan las conclusiones.

Técnicas y tecnología utilizada



Pandas y NumPy: Se usaron para leer y estructurar datos en DataFrames, facilitando la limpieza y el análisis, como el cálculo del porcentaje de llamadas perdidas y tiempos de espera. NumPy proporcionó funciones matemáticas y estadísticas adicionales (Exercise 2: Python for Data Analysis with Google Colab, Curso: Análisis de datos en Python con NumPy y Pandas).

Matplotlib y Seaborn: Se utilizaron para crear gráficos de barras y líneas que representaran visualmente las métricas clave, mejorando la claridad y estética de los resultados (Data Visualization in Python with Matplotlib and Seaborn - Colab, Getting started with Seaborn using Google Colab notebook).

Scipy: Se empleó para realizar análisis estadísticos, como el cálculo del percentil 75 para identificar operadores con tiempos de espera excesivos y realizar pruebas estadísticas para validar las hipótesis (Análisis de datos con Python (Pandas, Numpy, Matplotlib, Seaborn)).

Google Colab: Proporcionó un entorno basado en la nube para ejecutar código y compartir resultados en tiempo real, sin necesidad de configuraciones locales, facilitando la colaboración (Getting started with Seaborn using Google Colab notebook).

Técnicas matemáticas utilizadas

Estadística Descriptiva:

- Media: Se utilizó para calcular el tiempo promedio de espera para las llamadas entrantes por operador (Average Call Duration - ACD).
- Percentiles: Se utilizaron los percentiles (en este caso, el percentil 95) para definir un umbral para el tiempo promedio de espera y identificar a los operadores que superaban ese umbral.
- Sumas y conteos: Se utilizaron para calcular la cantidad total de llamadas, llamadas perdidas, llamadas internas, llamadas salientes, etc., por operador.

2. Porcentajes y Razones:

- Porcentaje de llamadas perdidas (Lost Call Rate - LCR): Se calculó dividiendo el número de llamadas perdidas por el número total de llamadas entrantes, multiplicado por 100.
- Tasa de llamadas internas (Internal Call Rate - ICR): Se calculó dividiendo el número de llamadas internas por el número total de llamadas, multiplicado por 100.
- Eficiencia de llamadas salientes (Outbound Call Efficiency - OCE): Se calculó dividiendo el número de llamadas salientes exitosas por el número total de llamadas salientes, multiplicado por 100.
- Porcentaje de llamadas exitosas (Call Success Rate - CSR): Se calculó dividiendo el número de llamadas exitosas (no perdidas) por el número total de llamadas, multiplicado por 100.

3. Agregación y Agrupación:

- Se utilizaron funciones de agregación como `groupby()` para agrupar los datos por operador y calcular las métricas correspondientes para cada operador.

4. Ponderación y Normalización:

- Ponderación: Se asignaron pesos a cada métrica (LCR, ACD, ICR, OCE, CSR) para reflejar su importancia en la calificación general de la eficiencia del operador.
- Normalización: Se normalizaron las métricas para que tuvieran un rango comparable y se pudieran combinar en una única calificación. Se utilizaron diferentes métodos de normalización, como dividir por el valor máximo o por 100.

5. Visualización de datos: * Se utilizaron gráficos de dispersión para visualizar la relación entre las métricas y la calificación de los operadores. Esto ayudó a identificar patrones y tendencias en los datos.

Espero que esta información te sea útil. Avísame si tienes alguna otra pregunta.

Formulas

Estadística Descriptiva:

- **Media (ACD):**

$$\text{Media} = \frac{\sum_{i=1}^n X_i}{n}$$

- **Percentil (en este caso, el percentil 95):**

P_{95} = Valor en el percentil 95 de los datos ordenados

- **Sumas y Conteos:**

Total de llamadas = $\sum (\text{llamadas totales})$

Porcentajes y Razones:

- **Porcentaje de llamadas perdidas (LCR):**

$$LCR = \left(\frac{\text{Llamadas Perdidas}}{\text{Llamadas Totales Entrantes}} \right) \times 100$$

- **Tasa de llamadas internas (ICR):**

$$ICR = \left(\frac{\text{Llamadas Internas}}{\text{Llamadas Totales}} \right) \times 100$$

- **Eficiencia de llamadas salientes (OCE):**

$$OCE = \left(\frac{\text{Llamadas Salientes Exitosas}}{\text{Llamadas Salientes Totales}} \right) \times 100$$

- **Porcentaje de llamadas exitosas (CSR):**

$$CSR = \left(\frac{\text{Llamadas Exitosas}}{\text{Llamadas Totales}} \right) \times 100$$

Agregación y Agrupación:

- **Ponderación:**

Métrica ponderada = $\sum (\text{Métrica} \times \text{Peso})$

- **Normalización:**

Métrica normalizada = $\frac{\text{Métrica}}{\text{Valor máximo}} \quad o \quad \frac{\text{Métrica}}{100}$



Hitos

- **Llamadas perdidas (LCR):** Algunos operadores presentan un porcentaje de llamadas perdidas superior al promedio, lo que afecta la experiencia del cliente.
 - **Tiempos de espera (AWT):** Se identificaron operadores con tiempos de espera superiores al percentil 95, generando frustración y posibles pérdidas de negocio.
 - **Llamadas internas (ICR):** Variaciones en la tasa de llamadas internas sugieren problemas de comunicación o procesos inefficientes.
 - **Eficiencia en llamadas salientes (OCE):** Operadores con baja eficiencia requieren mejorar habilidades, estrategias o gestión del tiempo.
 - **Éxito de llamadas (CSR):** Un bajo porcentaje de llamadas exitosas impacta el servicio y la satisfacción del cliente.
 - **Sistema de calificación:** Se creó una métrica para evaluar a los operadores y priorizar mejoras.
- 

Desempeño de Operadores en Métricas Clave de Eficiencia de Call Center

Los 10 mas altos

	operator_id	calification
306	966686	0.993651
363	907986	0.952683
249	940610	0.932564
522	924932	0.917244
55	907952	0.908198
361	907970	0.900888
269	958454	0.898715
237	888536	0.890921
358	907440	0.887763
325	887278	0.883200

Los 10 mas bajos

	operator_id	calification
1	-1	0.293718
519	892798	0.462898
161	906070	0.480409
187	921318	0.492721
0	948756	0.499260
322	947642	0.505101
507	917680	0.507893
491	919536	0.511089
496	919190	0.512129
414	900458	0.517480

Son losn prieros 10 de cada uno
en el Jupiter se podra ver a todos los que sobrepasan el percentil

Análisis del tiempo de espera promedio

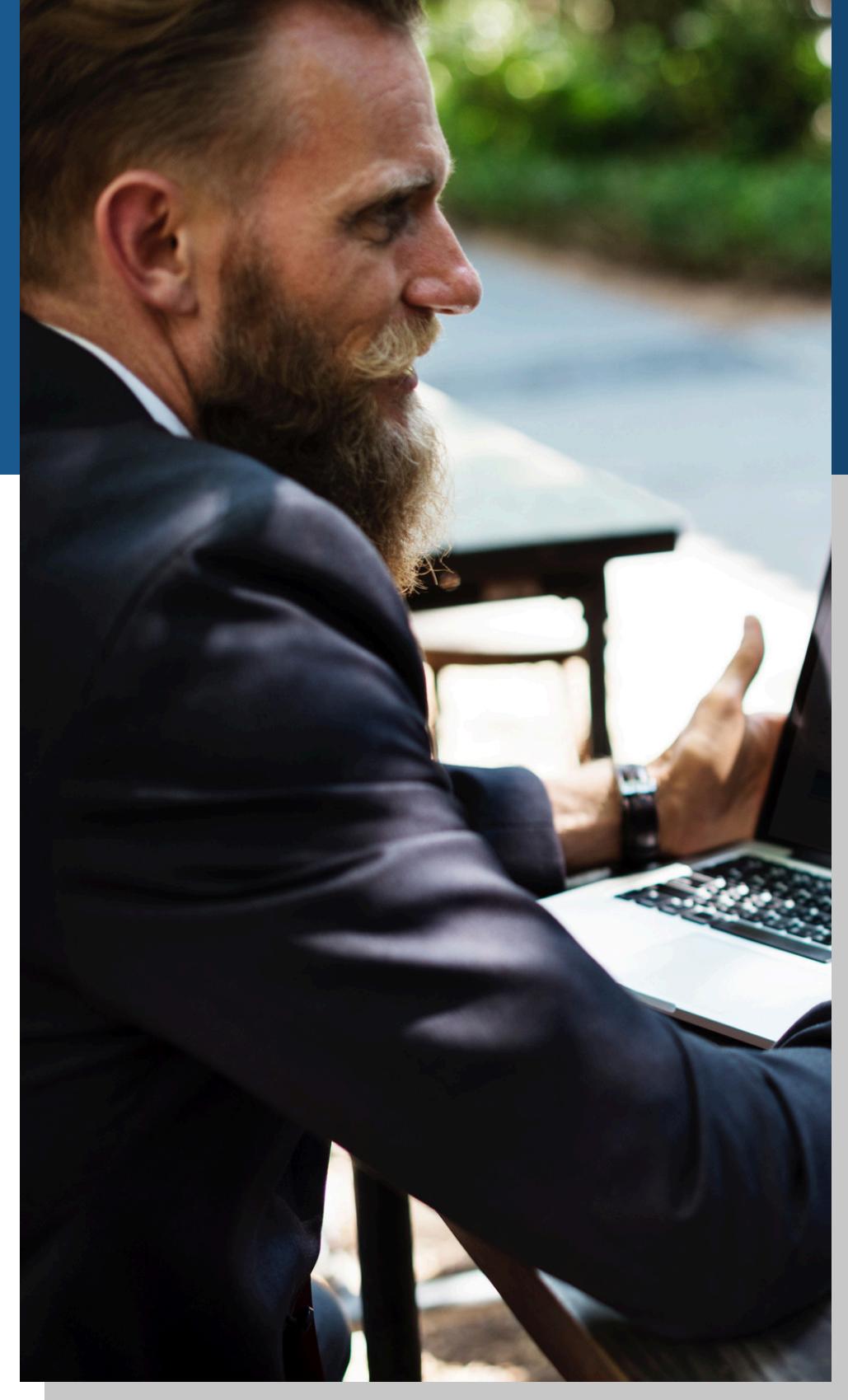
Percentil 95 del tiempo promedio de espera: 1126.974270270269
Operadores que superan el umbral del percentil 95: 55

operator_id	waiting_time	above_threshold	
552	925922	5907.443038	True
30	885890	5727.133574	True
584	929428	5325.342857	True
29	885876	4816.241379	True
318	908640	3456.461538	True
583	929426	3316.558824	True
844	945278	2858.479167	True
449	919364	2841.465116	True
848	945286	2185.774648	True
871	945904	2128.657895	True

Son losn prieros 10 de cada uno

Análisis con Dashboard

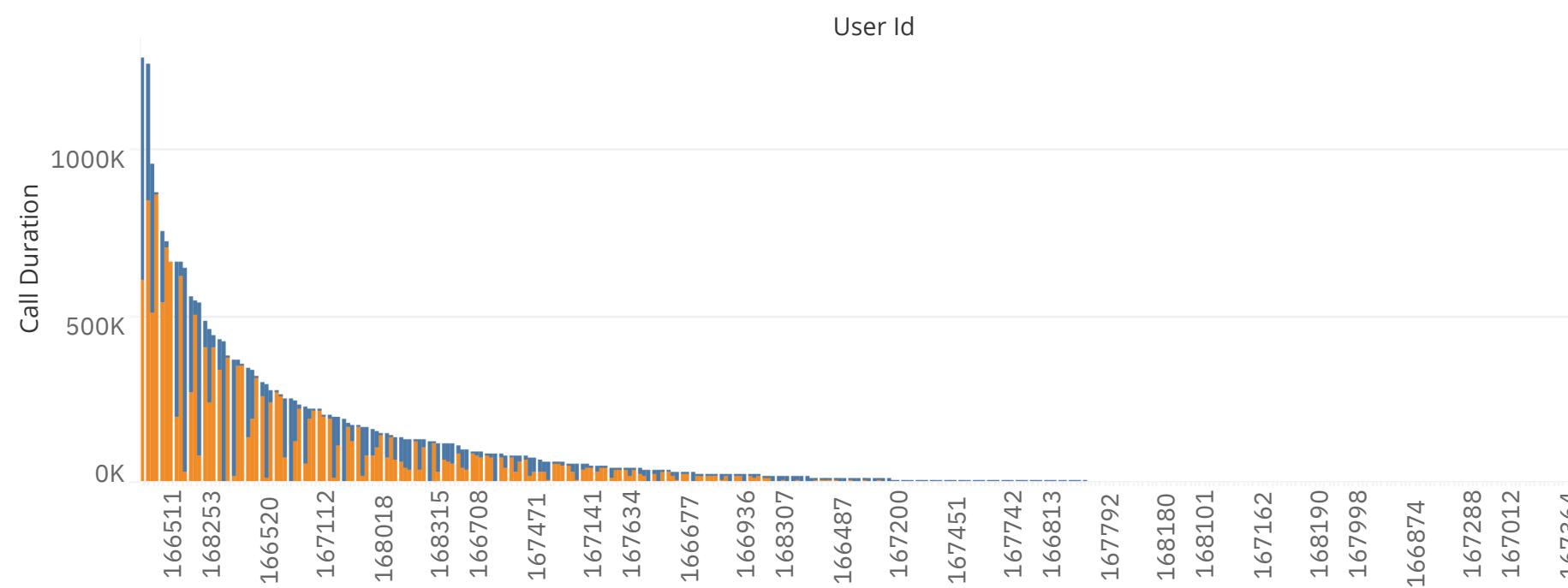
El dashboard proporcionará una representación visual intuitiva de los datos, lo que permitirá una toma de decisiones más rápida y fundamentada. Además, la interactividad nos dará la flexibilidad de explorar diferentes filtros y métricas para un análisis más detallado, facilitando la identificación de patrones y anomalías en el comportamiento de los operadores.



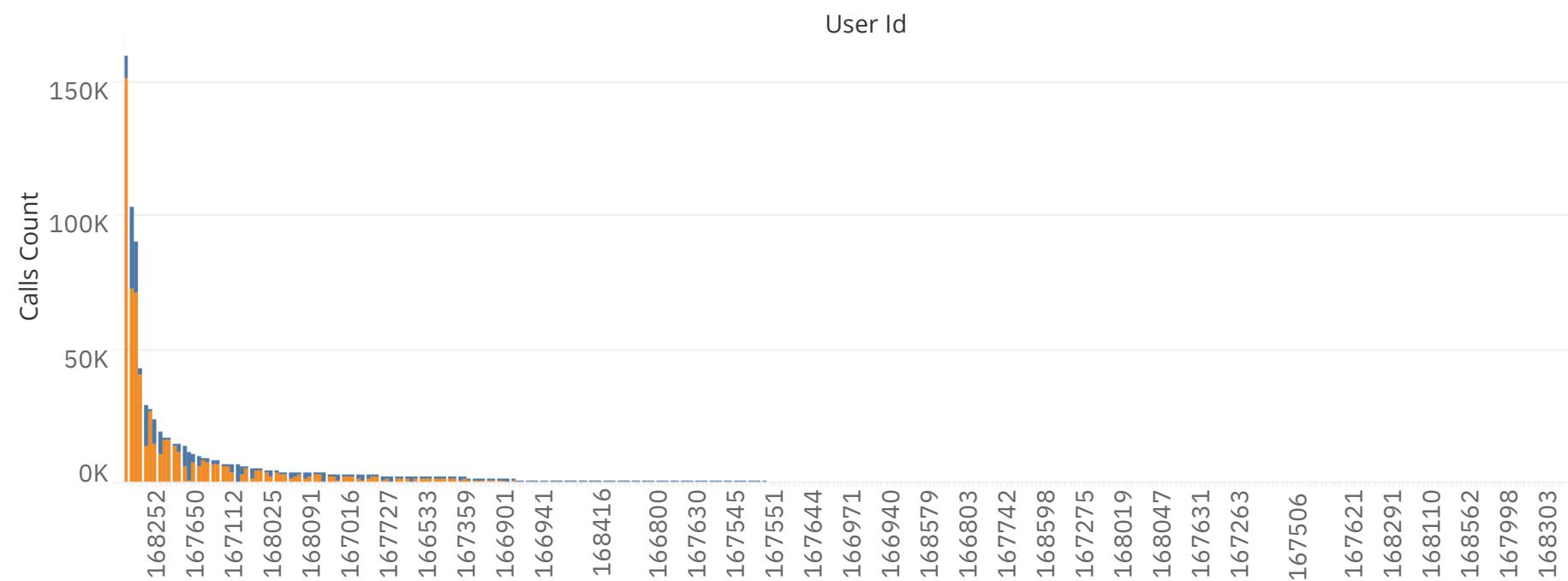
Telecomunicaciones: identificar operadores ineficaces

Direction
in out

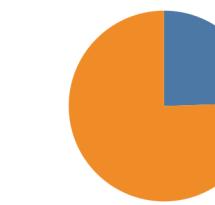
Duración de la llamada



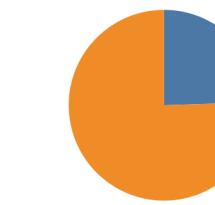
Número de llamadas por día



Llamadas internas y externas



Llamadas internas y externas de llamadas por dia



Link del Dashboard en mi Tableau

<https://public.tableau.com/app/profile/samuel.luis.juan.guerrero/viz/Telecomunicacionesidentificaroperadoresineficas/Dashboard>

Analisis del Dashboard

El análisis de este dashboard tiene como objetivo identificar áreas de mejora en la gestión de llamadas y el desempeño de los operadores. A partir de las métricas clave, se identificaron las siguientes oportunidades:

1. Optimización del tiempo de llamada:

- Algunos operadores presentan tiempos de llamada significativamente mayores al promedio, lo cual podría implicar una oportunidad para implementar estrategias de resolución más rápidas.

2. Equilibrio de cargas de trabajo:

- La distribución del número de llamadas por día sugiere posibles desbalances en la asignación de tareas, especialmente en días pico. Una planificación más eficiente podría mejorar la experiencia tanto para operadores como para clientes.

3. Mejora en la gestión de llamadas internas:

- La proporción de llamadas internas y externas señala áreas donde los operadores podrían requerir más autonomía o formación para resolver casos sin depender de comunicación interna.

Bibliografías

- Baumer, B. S., & Tredennick, A. T. (2020). Data Science for Social Good: Leveraging Big Data for Social Impact (1st ed.). Springer. <https://doi.org/10.1007/978-3-030-22768-5>
- Brownlee, J. (2020). Machine Learning Mastery with Python: Understand Your Data, Create Accurate Models, and Work Projects End-to-End. Machine Learning Mastery. <https://machinelearningmastery.com/machine-learning-mastery-with-python/>
- D. W. & K. H. (2021). Data Analysis with Python and Pandas: An Introduction. Wiley. <https://www.wiley.com/en-us/Data+Analysis+with+Python+and+Pandas>
- Le, T. (2018). Exploratory Data Analysis in Python with Pandas, Seaborn, and Matplotlib. Springer. <https://doi.org/10.1007/978-1-4842-4315-9>
- Plotly Technologies Inc. (2021). Plotly Python Graphing Library. <https://plotly.com/python/>
- Waskom, M. L., & Seaborn Developers (2020). Seaborn: Statistical Data Visualization. <https://seaborn.pydata.org/>
- Zhang, Z. (2020). Python for Data Science Handbook: Essential Tools for Working with Data. O'Reilly Media. <https://www.oreilly.com/library/view/python-for-data/9781491912058/>



Samuel Luis Juan Guerrero

Data Analyst