



# Proyecto BEDU

## Google Ads / Lavado de Muebles





# Introducción

## Nuestro enfoque

Somos una **agencia** de **publicidad digital** orientada a performance buscando siempre reducir los costos por conversión de nuestros clientes principalmente en las plataformas de **Google** y **Facebook**

## Problemática

En estos años se han detectado variaciones en los CPC's (costo por clic) de la cuenta que impactan directamente en los CPL (costo por lead) que estamos presentando al cliente, lo cual afecta los resultados de la cuenta.

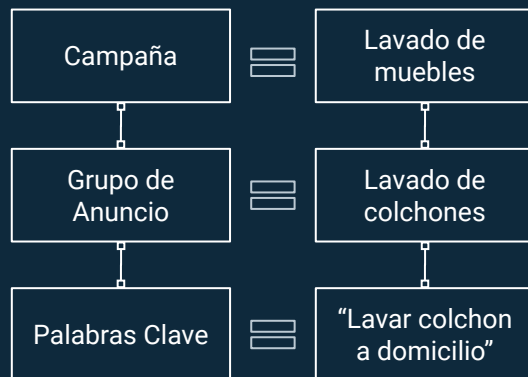


# Contexto

Los datos de nuestro proyecto están basados en una cuenta de google ads de un cliente que su giro principal es el **Lavado de Muebles** en la **CDMX**

Es importante destacar el funcionamiento de una campaña publicitaria para su entendimiento:

## Google Ads



- ◇ Moneda
- ◇ Día de la semana
- ◇ Día
- ◇ **Clics**
- ◇ Impresiones
- ◇ CTR
- ◇ **CPC promedio**
- ◇ Costo
- ◇ % Impresiones
- ◇ **Conversiones**
- ◇ Costo / Conversión



# Enfoque

El enfoque que queremos tener en este **proyecto** está basado en la necesidad que se tiene para poder tener alertas mucho más precisas respecto a las diferentes variaciones de CPC de las campañas, buscando encontrar un mayor equilibrio en las mismas y anticiparnos a posibles factores externos que nos puedan afectar:

- **Ingreso de nueva competencia**
- **Competidores actuales con aumento de presupuesto**
- **Temporalidades relacionadas al servicio ofrecido**




*“Y como conclusión poder tomar mejores decisiones en cuanto a la inversión y ajustes en las campañas.”*



# Datos a utilizar

Nuestro conjunto de datos es de fácil manipulación debido a que las plataformas ya nos lo proporcionan limpio, en este caso incluye las siguientes características:

- ◆ Palabras Clave
- ◆ Tipo de concordancia
- ◆ Nombre de la campaña
- ◆ Grupo de Anuncios
- ◆ CPC Máximo

| Exclude adult ideas  Add filter |                       |             |                     |                             |                              |  SAVE |  REMOVE |
|--|-----------------------|-------------|---------------------|-----------------------------|------------------------------|--|--|
| <input type="checkbox"/> Keyword (by relevance)  | Avg. monthly searches | Competition | Ad impression share | Top of page bid (low range) | Top of page bid (high range) |  |  |
| <input type="checkbox"/> moz directory listings  | 10 – 100              | Low         | —                   | £2.59                       | £7.98                        |  |  |
| <input type="checkbox"/> moz customer service  | 10 – 100              | Low         | —                   | £0.94                       | £4.03                        |  |  |
| <input type="checkbox"/> moz pricing   | 1K – 10K              | Low         | —                   | £1.85                       | £6.14                        |  |  |
| <input type="checkbox"/> moz pro pricing   | 100 – 1K              | Low         | —                   | £3.51                       | £9.21                        |  |  |
| <input type="checkbox"/> moz pro free trial  | 10 – 100              | Low         | —                   | £5.63                       | £13.69                       |  |  |
| <input type="checkbox"/> moz analytics free  | 10 – 100              | Low         | —                   | £1.09                       | £6.71                        |  |  |
| <input type="checkbox"/> moz pro trial   | 10 – 100              | Low         | —                   | £2.48                       | £11.96                       |  |  |



# Objetivos y Tareas

- ◇ Cargar los datos en un modelo de base de datos relacional.
- ◇ Realizar un análisis exploratorio de dato con el fin de obtener estadísticas sobre los datos obtenidos.
- ◇ Conectar la base de datos con Python usando la herramienta Colab para realizar labores de análisis y predicción.
- ◇ Con base en el campo que nos permite determinar si el CPC es atípico (mayor / inferior), evaluar los distintos modelos de regresión con el fin de encontrar, predecir y alertar este tipo de movimientos.

# Carga y preprocesamiento de datos

Se optó por usar el Gestor de Bases de datos MySQL y la plataforma Colab para poder generar nuestro procesamiento de la data, en este caso se utilizó el siguiente conjunto de datos:

◇ Germany-Clean\_Keywords\_20210101-20210630.csv

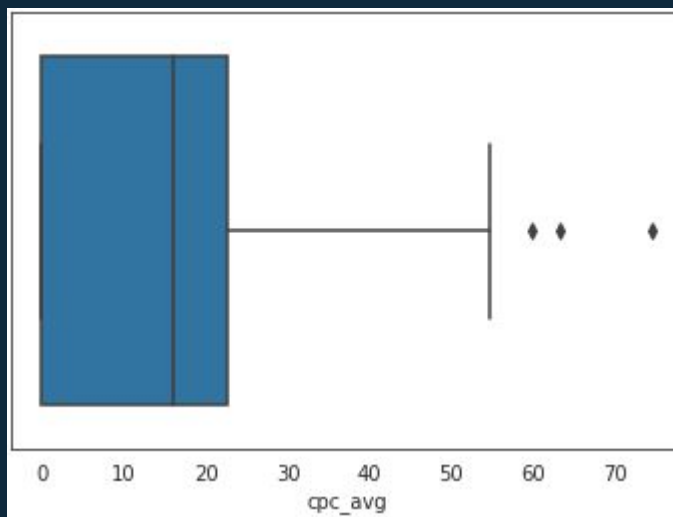
Previo a la carga de datos, se revisó su estructura y no se encontraron problemas mayores, con la excepción de los nombres de las columnas, los cuales se renombraron para un mejor manejo de los mismos. Para fines del proyecto se generó una nueva vista (df\_keywords).

```
[ ] df_keywords.columns = ['keyword', 'match', 'campaign', 'ad_groups', 'cpc_max', 'currency', 'day', 'date', 'clicks', 'impressions', 'ctr', 'cpc_avg',  
    'cost', 'top_abs_percent', 'top_percent', 'conversions', 'conv_post_impression', 'cpl',  
    'conv_percent', 'cpm_prom']  
df_keywords.columns  
  
Index(['keyword', 'match', 'campaign', 'ad_groups', 'cpc_max', 'currency',  
      'day', 'date', 'clicks', 'impressions', 'ctr', 'cpc_avg', 'cost',  
      'top_abs_percent', 'top_percent', 'conversions', 'conv_post_impression',  
      'cpl', 'conv_percent', 'cpm_prom'],  
      dtype='object')  
  
[ ] df_keywords.head()
```

# Análisis Exploratorio de Datos

Se dividió el Análisis Exploratorio en los siguientes pasos:

- ◇ Modificación del dtypes para los diferentes datos
- ◇ Generación de Bloxplot → buscando representar gráficamente nuestra serie de datos numéricos (CPC) a través de sus cuartiles



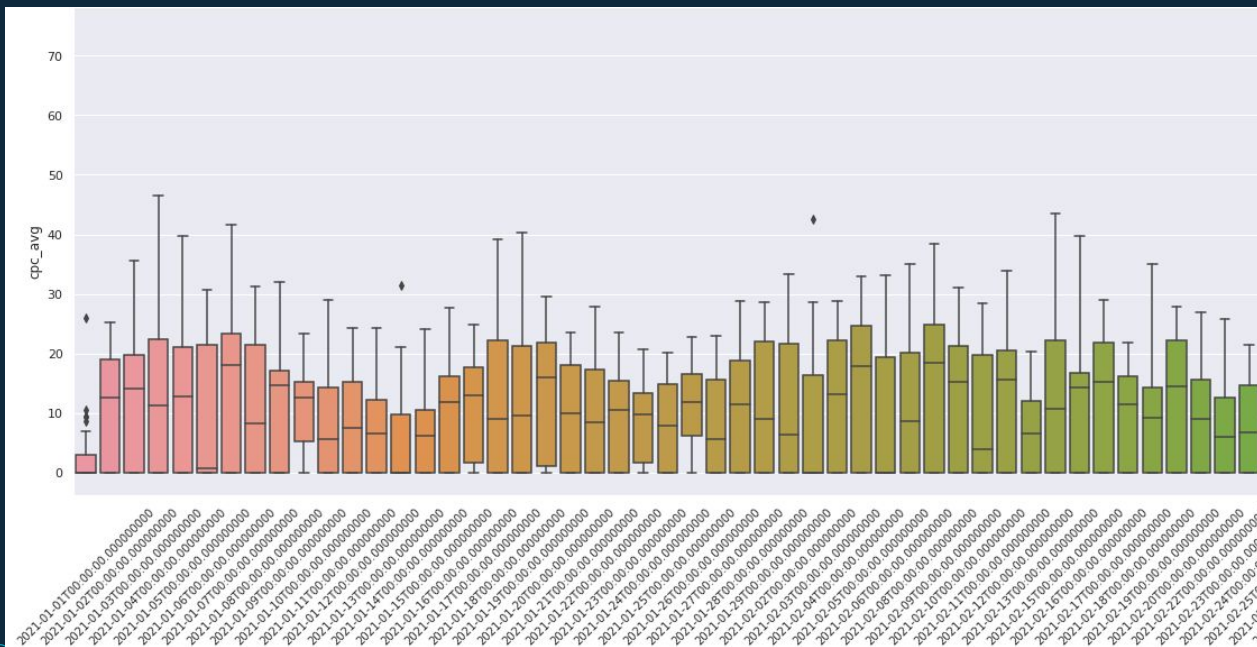
```
count    2980.000000
mean      15.033470
std       11.515071
min        0.000000
25%        0.000000
50%       16.045000
75%       22.690000
max       74.440000
Name: cpc_avg, dtype: float64
```





- ◆ Nuestro objetivo principal es el comportamiento del costo por clic (CPC) durante un periodo de tiempo determinado, por lo cual se implementa el siguiente código para poder visualizarlo correctamente:

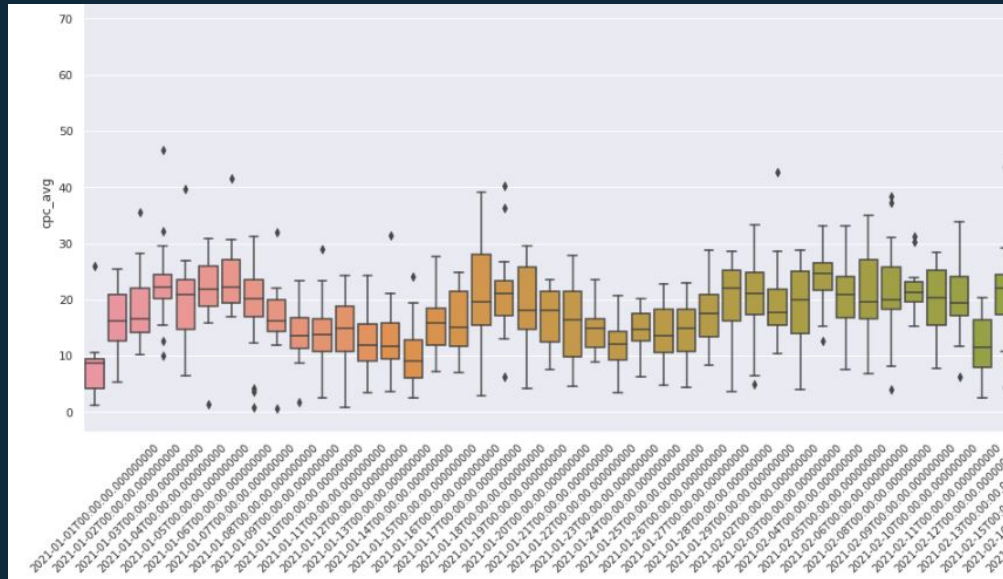
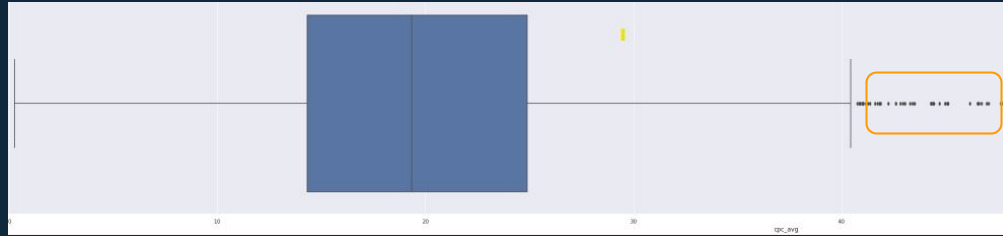
```
sns.set(rc={'figure.figsize':(60,8)})  
key = sns.boxplot(data=df_keywords.sort_values(by='date',ascending=True), x='date', y='cpc_avg');  
key.set_xticklabels(key.get_xticklabels(),rotation = 45, size = 10)
```



Se puede observar en la gráfica que en la mayoría de los datos comienzan desde el valor “0”, lo cual no nos permitirá un análisis correcto de los mismos.



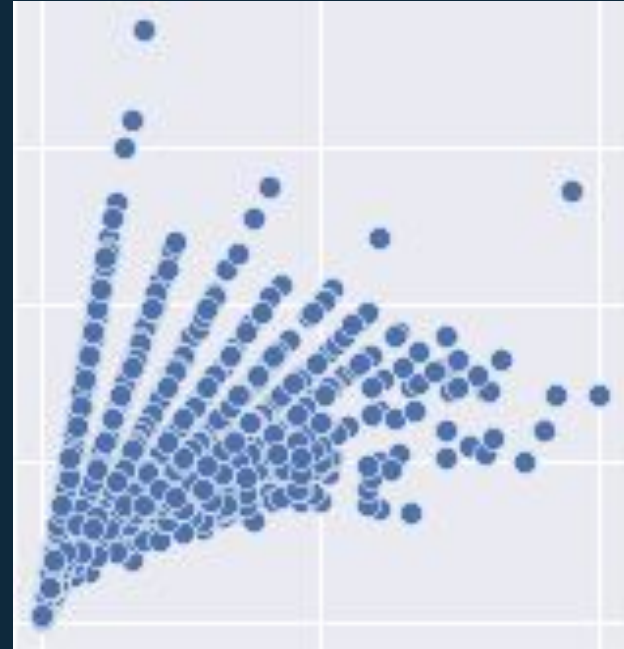
- ◇ Con base en lo anterior se elimina todo valor (CPC) igual a “0”, lo cual no afecta nuestro modelo al no tener interacción por parte de un usuario final, y se procede a realizar de nuevo la gráfica: Bloxplot y Comportamiento diario del CPC.



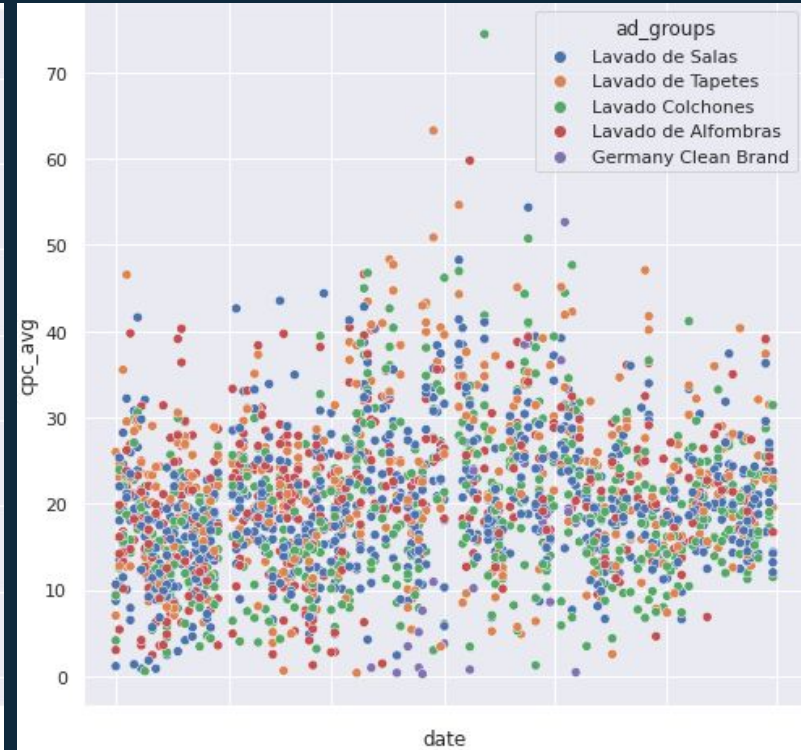
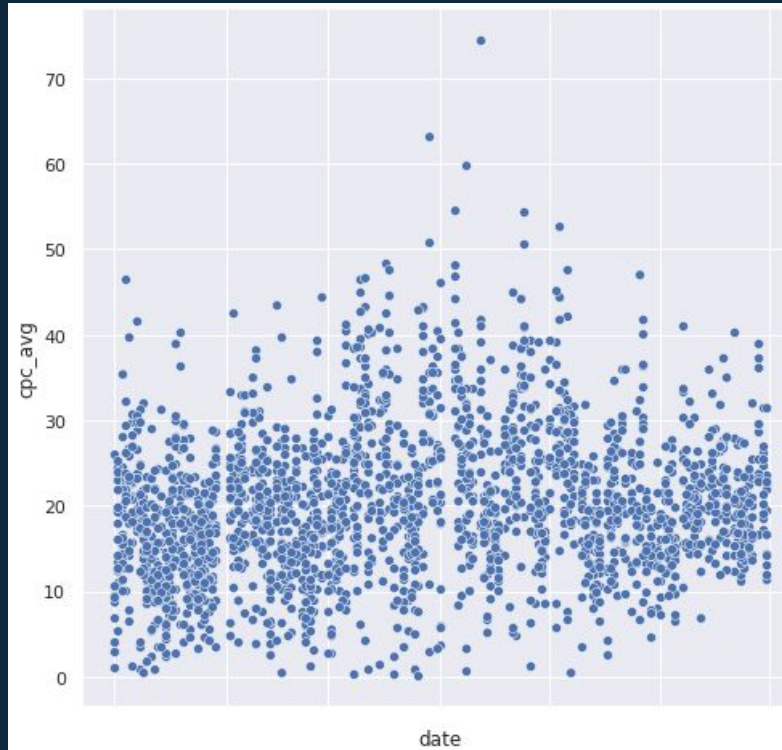
- ◇ Se generó un pairplot buscando conocer a detalle una tendencia en los datos sin embargo, en cuanto al CPC vs Costo no es muy detallada

- ◇ Uno de los puntos importantes a mencionar (tomando en cuenta la estructura de una campaña), es que puede existir más de un grupo de anuncio dentro de ella, por lo cual si se busca tener un análisis más exhaustivo es necesario hacer la división de estas; por lo cual se procede a generar:

- Rendimiento por campaña
- Rendimiento por grupo de anuncios



- ◇ De igual forma se generó un **Diagrama de Dispersión** buscando visualizar los elementos de 2 coordenadas considerando CPC y Date, en este primer diagrama es importante considerar que estamos evaluando la campaña general, pero de igual forma se generó otro considerando grupos de anuncios.



# Caso de estudio

## Lavado de Salas

- ◇ En este punto nuestra división se realizó tomando en cuenta los grupos de anuncios de las campañas, para un análisis mucho más preciso.

```
#Cantidad de grupos de anuncios
keywords_non_0['ad_groups'].unique()
```

```
array(['Lavado Colchones', 'Lavado de Alfombras', 'Lavado de Salas',  
      'Lavado de Tapetes', 'Germany Clean Brand'], dtype=object)
```

```
#Separar por grupos de anuncios
```

```
keywords_Lavado_de_Salas = keywords_non_0[keywords_non_0.ad_groups == "Lavado de Salas"]  
keywords_Lavado_de_Tapetes = keywords_non_0[keywords_non_0.ad_groups == "Lavado de Tapetes"]  
keywords_Lavado_Colchones = keywords_non_0[keywords_non_0.ad_groups == "Lavado Colchones"]  
keywords_Lavado_de_Alfombras = keywords_non_0[keywords_non_0.ad_groups == "Lavado de Alfombras"]  
keywords_Germany_Clean_Brand = keywords_non_0[keywords_non_0.ad_groups == "Germany Clean Brand"]
```

```
#Ordenar los DF
```

```
keywords_Lavado_de_Salas = keywords_Lavado_de_Salas.sort_values(by='date',ascending=True)  
keywords_Lavado_de_Tapetes = keywords_Lavado_de_Tapetes.sort_values(by='date',ascending=True)  
keywords_Lavado_Colchones = keywords_Lavado_Colchones.sort_values(by='date',ascending=True)  
keywords_Lavado_de_Alfombras = keywords_Lavado_de_Alfombras.sort_values(by='date',ascending=True)  
keywords_Germany_Clean_Brand = keywords_Germany_Clean_Brand.sort_values(by='date',ascending=True)
```

# Caso de estudio

## Lavado de Salas

- ◇ Tomando en cuenta lo anterior, se tomó como caso de estudio el Grupo de Anuncios de “**Lavado de Salas**” y pasamos a describirlo para encontrar los valores promedio y atípicos

```
keywords_Lavado_de_Salas['cpc_avg'].describe()
```

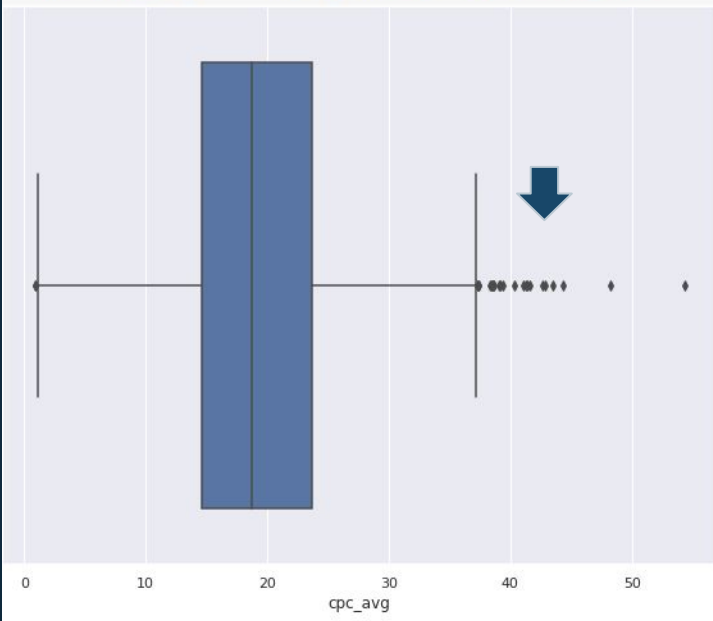
|       |            |
|-------|------------|
| count | 737.000000 |
| mean  | 19.672619  |
| std   | 7.737799   |
| min   | 0.870000   |
| 25%   | 14.610000  |
| 50%   | 18.770000  |
| 75%   | 23.660000  |
| max   | 54.340000  |

Name: cpc\_avg, dtype: float64

# Caso de estudio

## Lavado de Salas

```
sns.set(rc={'figure.figsize':(10,8)})  
sns.boxplot(keywords_Lavado_de_Salas['cpc_avg'])
```



- ◇ Gráficamente con el boxplot podemos observar que los valores de los **CPC** en un rango están dentro de los **\$14** a los **\$23** y también alcanzamos a notar bastantes valores atípicos, que según nuestra descripción previa nos dice que hubo clics de hasta **\$54**





# Caso de estudio

## Lavado de salas

- ◇ En una serie de tiempo podemos ver una tendencia cíclica en el comportamiento de las **CPC** de las palabras claves con algunos picos atípicos, pero en general podemos apreciar como dependiendo de la temporada el CPC sube o baja habiendo días extraordinarios donde la variación es mucho mayor.

```
sns.set(rc={'figure.figsize':(60,8)})  
  
bp = sns.boxplot(data=keywords_Lavado_de_Salas.sort_values(by='date',ascending=True), x='date', y='cpc_avg');  
bp.set_xticklabels(bp.get_xticklabels(),rotation = 45, size = 10)
```





# Caso de estudio

## Lavado de salas

- Considerando que tenemos valores muy atípicos por arriba del percentil 75 se genera un DF diferente para poderlos evaluar

```
keywords_Lavado_de_Salas_high = keywords_Lavado_de_Salas[keywords_Lavado_de_Salas.cpc_avg > np.percentile(keywords_Lavado_de_Salas.cpc_avg, 75)]
```

```
keywords_Lavado_de_Salas_high['cpc_avg'].describe()
```

```
count    184.000000
mean      30.157554
std        5.378148
min       23.700000
25%       25.932500
50%       28.695000
75%       32.852500
max       54.340000
Name: cpc_avg, dtype: float64
```

```
keywords_Lavado_de_Salas_high['cost'].sum()
```

```
23125.07
```

```
keywords_Lavado_de_Salas_high['conversions'].sum()
```

```
190.3
```

```
keywords_Lavado_de_Salas_high['cost'].sum() / keywords_Lavado_de_Salas_high['conversions'].sum()
```

```
121.5190225959012
```



# Caso de estudio

## Lavado de salas

- ◇ Se revisó de igual forma la parte de los datos promedio buscando evaluar estos resultados vs la parte High

```
keywords_Lavado_de_Salas_promedio = keywords_Lavado_de_Salas[keywords_Lavado_de_Salas.cpc_avg < np.percentile(keywords_Lavado_de_Salas.cpc_avg, 75)]
```

```
keywords_Lavado_de_Salas_promedio['cost'].sum()
```

```
28846.72
```

```
keywords_Lavado_de_Salas_promedio['conversions'].sum()
```

```
314.93
```

```
keywords_Lavado_de_Salas_promedio['cost'].sum() / keywords_Lavado_de_Salas_promedio['conversions'].sum()
```

```
91.59724383196266
```





# Conclusiones

Con base en los distintos trabajos de análisis y definición de modelos se dan las siguientes conclusiones:

- ◆ En el ámbito del marketing digital analizar estos datos y poder apreciarlos graficados nos ayuda a tomar decisiones más acertadas con base en los históricos de nuestras campañas, el poder conocer el comportamiento típico y atípico nos ayuda a predecir el CPC normal y poder manipular de forma más eficiente los recursos asignados por el cliente.



# Trabajo a futuro

- ◇ Replicar el modelo generado para las demás cuentas de la agencia, considerando las diferentes líneas de negocio y los diferentes CPC que se tienen en cada una de ellas.
- ◇ Así como también la futura posibilidad de poder generar una conexión API entre los datos de nuestros clientes para una evaluación automática que nos permita ser más eficientes día a día.





GRACIAS!

