Introducción al modelado de mezcla de marketing en Python

¿Qué inversión publicitaria está impulsando realmente sus ventas?

Dr. Robert Kübler 22 de septiembre · 8 min de lectura★

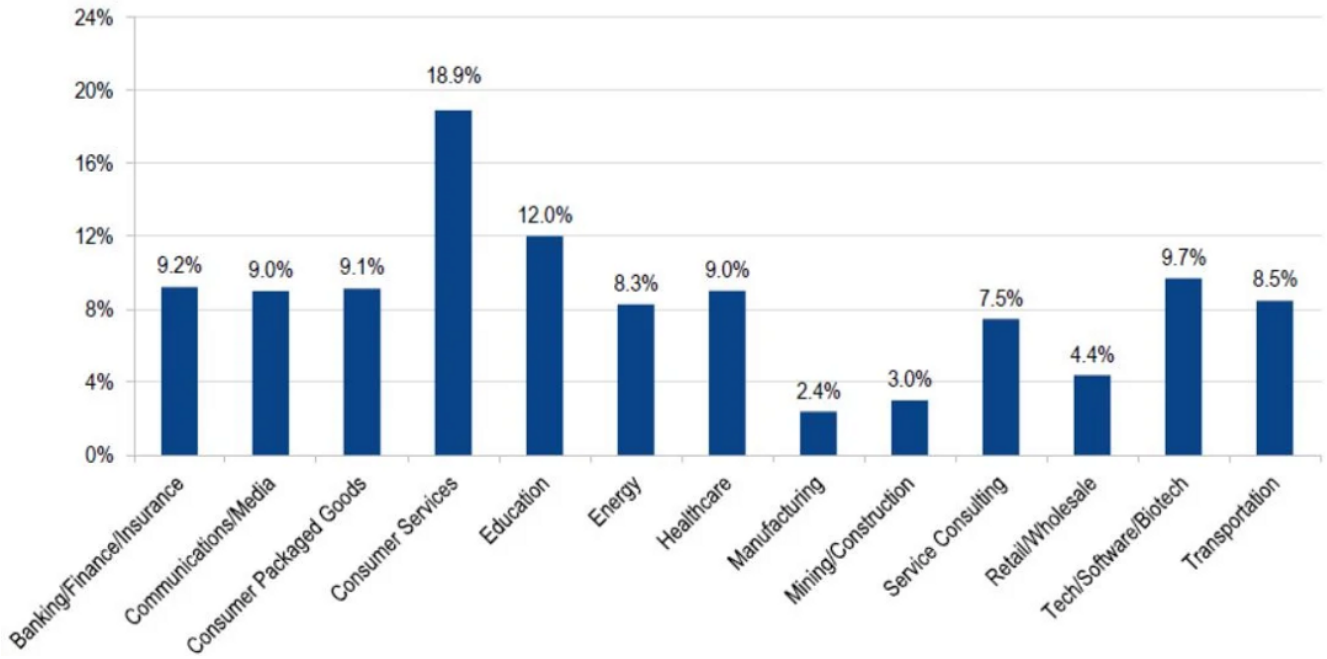
Foto de Tim Johnson en Unsplash

Introducción a la publicidad

Para mantener una empresa en funcionamiento, gastar dinero en publicidad es fundamental; este es el caso independientemente de si la empresa es pequeña o ya está establecida. Y la cantidad de gastos publicitarios en la industria es enorme:

Figure 3.9. Marketing spending as percent of company revenues by industry

Gasto en marketing como porcentaje de los ingresos de la empresa por sector



Fuente: https://www.webstrategiesinc.com/blog/how-much-budget-for-online-marketing-in-2014 , (artículo actualizado en 2020)

Estos volúmenes hacen necesario gastar sabiamente cada dólar publicitario. Sin embargo, es más fácil decirlo que hacerlo, o como lo expresaron el magnate minorista estadounidense **John Wanamaker** o el industrial británico **Lord Leverhulme** hace unos cien años:

> “La mitad del dinero que gasto en publicidad se desperdicia; el problema es que no sé qué mitad ".

Se podría pensar que esto i s menos de un problema hoy en día, pero curiosamente, todavía persiste. Afortunadamente, estamos en la posición de tener acceso a una gran cantidad de datos y computadoras poderosas para cambiar este estado de cosas a través de análisis avanzados, como el **Modelado de atribución** o el **Modelado de mezcla de marketing** . En este artículo, nos centraremos en este último.

Un ejemplo de conjunto de datos y modelado simple

Imagínese ahora que **usted** es responsable por el presupuesto de marketing de alguna empresa establecida. Para aumentar las ventas, reproduce publicidad en tres *canales*

publicitarios diferentes *:*

- **TV** ,
- **radio** y
- **banners web** .

Los datos

En cada semana, decides gastar alguna cantidad de dinero en cada canal, o no. Además, puedes observar el número de ventas cada semana. Los datos recopilados durante 200 semanas podrían verse así:

Date	TV	Radio	Banners	Sales
2018-01-07	13528.10	0.00	0.00	9779.80
2018-01-14	0.00	5349.65	2218.93	13245.19
2018-01-21	0.00	4235.86	2046.96	12022.66
2018-01-28	0.00	3562.21	0.00	8846.95
2018-02-04	0.00	0.00	2187.29	9797.07
...	...	...	...	...
2021-10-03	0.00	0.00	1691.68	9030.17
2021-10-10	11543.58	4615.35	2518.88	15904.11
2021-10-17	0.00	4556.16	1919.19	12839.29
2021-10-24	0.00	0.00	1707.65	9063.45
2021-10-31	0.00	0.00	1863.31	7250.21

Imagen del autor.

Todos los números de la tabla están en la moneda de su elección, usaré € a partir de ahora. **Puede obtener el archivo anterior aquí.**

En el pequeño adelanto anterior, podemos ver que hay muchas semanas sin publicidad en televisión (71%), y también algunas sin publicidad en la radio

(54%). Los banners web solo están desactivados en alrededor del 24% de las observaciones, lo que lo convierte en el canal más utilizado.

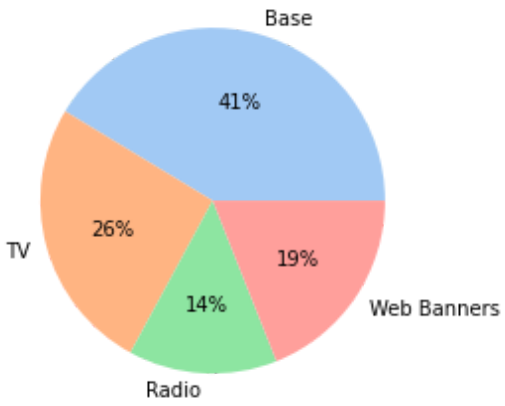
Sin embargo, cuando hacemos gastos en televisión, tienden a ser más altos que los gastos de radio, que a su vez son más altos que los gastos de publicidad en banners web. Además, hay rebajas todo el tiempo.

Ahora, antes de empezar a modelar, aclaremos primero el objetivo.

La meta

Al final, queremos poder responder preguntas como

> ¿Qué porcentaje de las ventas de 15.904,11 € en la semana que finalizó en 2021-10-10 (consulte la tabla anterior) se generó mediante publicidad televisiva? ¿Y cuánto por radio y banners web? ¿Y cuál es la línea de base, es decir, la cantidad de ventas que hubiéramos tenido sin publicidad?



Un resultado potencial. Imagen del autor.

Si nuestro modelo puede hacer esto, también podemos usarlo para calcular el ROI y optimizar el gasto, que es lo que las empresas quieren en última instancia. Teniendo este objetivo en mente, podemos restringirnos a modelos **aditivos** , es decir, modelos de la forma

Ventas = f (TV) + g (Radio) + h (Banners) + Base

porque nos permiten descomponer las ventas fácilmente. Las ventas son solo la suma de alguna función que solo depende de los gastos de TV, otra función que solo depende de los gastos de radio, otra función que solo depende de los gastos de banner web y una línea de base (constante).

Los modelos como **los bosques aleatorios** , el **aumento de gradiente** o **las redes neuronales** (simples y simples) **no** son **adecuados** aquí, ya que no podemos obtener tal descomposición de ellos.

> **Nota:** Claro, hay valores de Shapley que *tipo de* hacer lo que queremos, pero a menudo las contribuciones de acuerdo con los valores de Shapley son negativos, lo cual es algo razonable que la gente de marketing no quieren oír.

Un candidato que nos da una contribución aditiva para cada canal es un viejo amigo: ¡la regresión lineal, el representante más simple de los modelos aditivos!

Puede encontrar el artículo de seguimiento con un modelo más elaborado aquí:

https://towardsdatascience.com/an-upgraded-marketing-mix-modeling-in-python-5ebb3bddc1b6

Modelado mediante regresión lineal

Después de almacenar los datos de arriba en la variable `data` , hacemos lo siguiente:

```
desde sklearn.linear_model importar LinearRegression
desde sklearn.model_selection importar cross_val_score,
TimeSeriesSplit
importar pandas como pd

data = pd.read_csv (
    '
https://raw.githubusercontent.com/Garve/datasets/4576d323bf2b66c906d5
130d686245ad205505cf/mmm.csv ',
    parse_dates = ['Fecha'],
```

```
    index_col = 'Fecha'
)

X = data.drop (columnas = ['Ventas'])
y = datos ['Ventas']

lr = LinearRegression ()
print (cross_val_score (lr, X, y, cv = TimeSeriesSplit ()))

# Salida: [0.69594303 0.69302285 0.66850729 0.78807363 0.73512387]
```

> **Nota:** No usamos la validación cruzada estándar *k*- veces aquí porque estamos tratando con datos de series de tiempo. `TimeSeriesSplit` es algo más razonable, y puede leer más al respecto aquí .

Bueno, esto ya parece razonable, aunque podría ser mejor. Pero este modelo nos permite dividir las ventas como quisiéramos, ya que la fórmula es simplemente

$$\text{Ventas} = 0.36 * \text{TV} + 0.49 * \text{Radio} + 1.23 * \text{Banners} + 6678.40$$

Podemos obtener los coeficientes y la intersección mediante un simple

```
lr.fit (X, y) # reajuste el modelo con el conjunto de datos completo

print ('Coeficientes:', lr.coef_)
print ('Intercepción:', lr.intercepción_)

# Salida:
# Coeficientes: [0.35968382 0.48833246 1.2159193]
# Intercepción: 6678.396933606161
```

Desglose de las ventas

Para ilustrar el cálculo de las contribuciones, consideremos una sola semana:

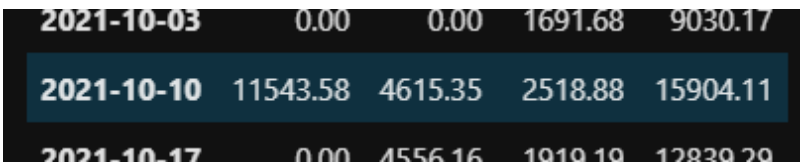


Imagen del autor.

Ingresemos los números y veamos qué obtenemos:

```
imprimir (lr.predict (([[11543.58, 4615.35, 2518.88]])))

# Salida: [16147.01594158]
```

Esta no es exactamente la respuesta verdadera de 15904.11 de la tabla anterior, pero sigamos con ella por ahora. Ahora podemos ver que la contribución (no ajustada) de la televisión es

$$coef\_TV * gastar\_TV = 0.36 * 11543.58 = 4155.69,$$

y para los otros canales en consecuencia. Ahora, las contribuciones se suman a la predicción del modelo 16147.0159, que no es el verdadero objetivo de 15904.11, así que multipliquemos las contribuciones y la línea de base por un factor de corrección de correct_factor = 15904.11 / 16147.0159 ≈ 0.985 y todo está bien. Obtenemos

$$contribución\_TV = factor\_de\_corrección * 4155.69 = 4089.57.$$

También obtenemos

- contrib_radio = 2219.92 y
- contribución_banners = 3016.68
- base = 6577,93.

Sumando todo, se obtiene la etiqueta observada, como la queremos:

$$4089,57 + 2219,93 + 3016,68 + 6577,93 = 15904,11.$$

Podemos generar una buena gráfica de contribución para todas las observaciones como esta:

```
pesos = pd.Series (
    lr.coef_,
    index = X.columns
)

base = lr.intercept_

unadj_contributions = X.mul (pesos) .assign (Base = base)
adj_contributions = (unadj_contributions
                     .div (unadj_contributions.sum (eje = 1), eje =
0)
                     .mul (y, eje = 0)
                    ) # contiene todas las contribuciones para cada
día

ax = (adj_contributions [['Base', 'Banners', 'Radio', 'TV']]
      .plot.area (
          figsize = (16, 10),
          linewidth = 1,
          title = 'Predicted Sales and Breakdown',
          ylabel = 'Ventas',
          xlabel = 'Fecha')
     )

tiradores, etiquetas = ax.get_legend_handles_labels ()
ax.legend (
    tiradores [:: - 1], etiquetas [:: - 1],
    título = 'Canales', loc = "centro izquierda",
    bbox_to_anchor = (1.01, 0.5)
)
```

La salida es:

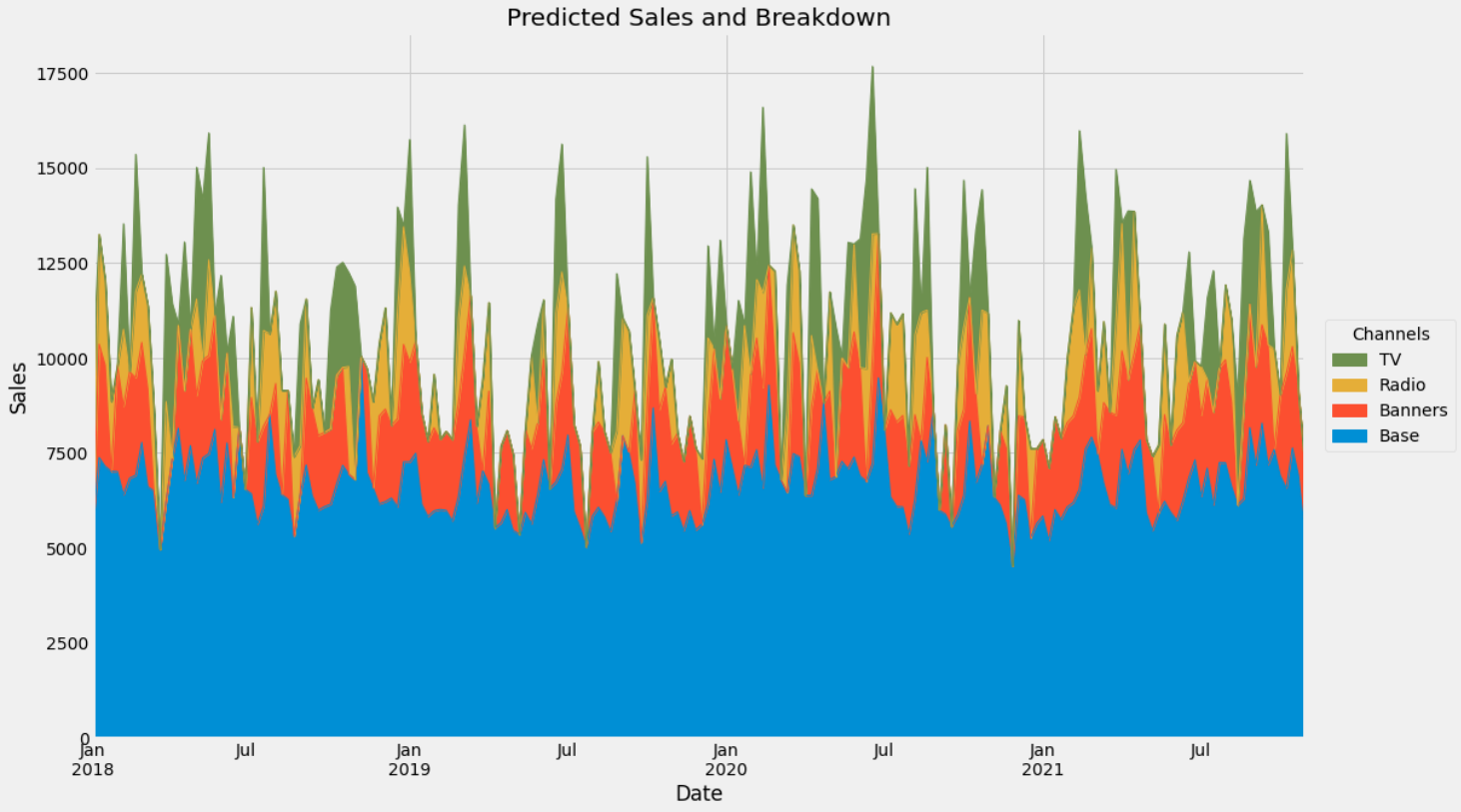


Imagen del autor.

Podemos ver (o calcular) que la línea de base es de alrededor de 6500 ventas diarias, los banners y la radio contribuyen en promedio alrededor de 2500 cuando están activos y la televisión alrededor de 3500 cuando está activo. ¡Bonito!

Calcular el retorno de la inversión (ROI)

Ahora podemos determinar qué canal fue el mejor en términos de ROI, un número que mide la eficiencia. La fórmula es simple:

channel_ROI = Ventas del canal / gasto del canal

Con los fragmentos de código de arriba, ya tenemos todos los datos que necesitamos para el cálculo. Puede calcular el ROI de TV de la siguiente manera:

```
sales_from_tv = adj_contributions ['TV']. sum ()
gastar_en_tv = datos ['TV']. sum ()

tv_roi = sales_from_tv / gastando_en_tv

# tv_roi es alrededor de 0.36
```

Simple como eso. Un ROI menor que 1 significa que el canal tuvo un desempeño deficiente. Para el ROI de TV podemos decir:

> Por cada euro que gastamos en televisión, recuperamos 36 centavos.

Ese es un tipo de trato que no queremos hacer con demasiada frecuencia si queremos que la empresa sobreviva. Por otro lado, los banners tienen un ROI de 1.21 que es mucho mejor, parece que este canal funcionó bastante bien en el período de tiempo que consideramos.

Problemas con este enfoque simple

Si bien el enfoque anterior parece razonable, tiene ciertas deficiencias que debemos abordar:

1. El rendimiento podría ser mejor. A veces, no hay nada que podamos hacer sobre el mal rendimiento porque depende mucho de los datos, pero deberíamos hacer todo lo posible de todos modos.
2. Aún más severo: el modelo en sí definitivamente **no** refleja la realidad. De acuerdo con la fórmula lineal, podemos impulsar las ventas tanto como queramos gastando cada vez más dinero en publicidad. Dado que los banners tienen un coeficiente alto de 1,23, por cada 1 € que gastamos en este canal generamos 1,23 € de ventas adicionales. Repita por dinero infinito, ¡los clientes odiarán este truco!
3. La optimización también se vuelve trivial y poco realista. Para maximizar las ventas, ahora pondríamos todo el dinero en el canal de banners web porque tiene el coeficiente más alto. Abandonaríamos la publicidad en televisión y radio por completo, lo que puede no ser lo correcto si la empresa quiere mantener su conocimiento entre las personas.

Resolveremos todos estos problemas en el artículo de seguimiento, ¡estad atentos!

Resumen y Outlook

Hemos visto que las empresas utilizan una parte importante de sus ingresos en publicidad para animar a los clientes a comprar sus productos. Sin embargo, utilizar

el presupuesto de marketing de manera eficiente no es fácil, tanto hace cien años como hoy. No es tan fácil averiguar cuánto influyó en las ventas un determinado gasto en publicidad de televisión y, por tanto, si valió la pena el gasto en publicidad y cómo optimizarlo para la próxima vez.

Para resolver este problema, creamos un pequeño modelo de mezcla de marketing que nos permitió desglosar las ventas observadas en acciones: TV, radio, banner y base share. Estas contribuciones de canal nos permiten calcular los ROI que nos permitieron ver el rendimiento de cada canal.

Sin embargo, este modelo es demasiado simple para capturar la realidad, lo que crea muchos problemas. Pero aprenderemos a eludirlos haciendo que el modelo sea un poco más complejo, pero aún interpretable.