Data Science para Economía y Negocios

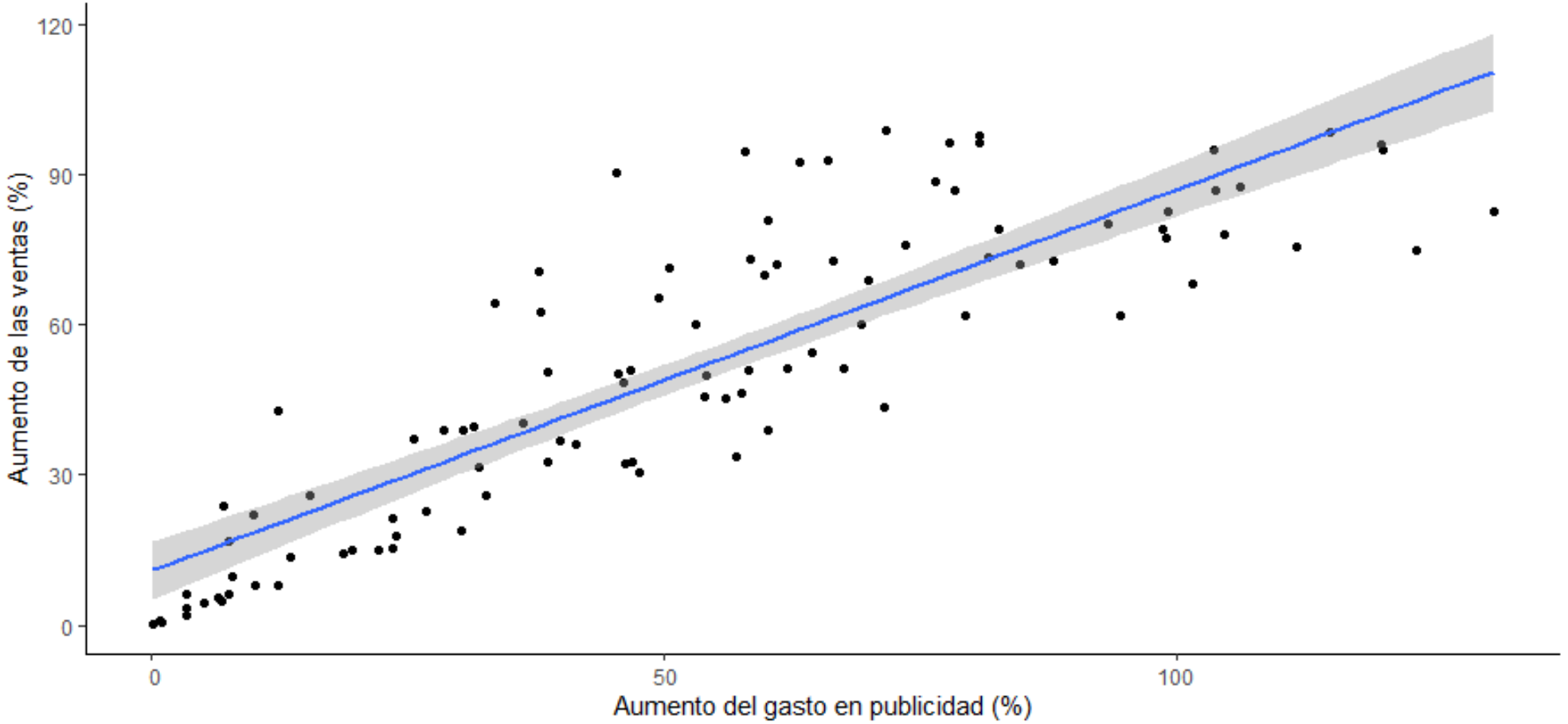
VALIDACIÓN CRUZADA

JAVIER FERNÁNDEZ

Motivación

- Uno de los objetivos de DS es predecir alguna variable de interés.
- Dado lo que vimos en la clase pasada, podemos utilizar regresión lineal como una herramienta de predicción.
- Pensemos el siguiente modelo lineal que intenta explicar el crecimiento de las ventas.

Los datos


120
90
60
30
0
Aumento de las ventas (%)
0
50
100
Aumento del gasto en publicidad (%)

Regresión

MODEL INFO:
Observations: 100
Dependent Variable: gventas
Type: OLS linear regression

MODEL FIT:
$F(1,98) = 274.31$, $p = 0.00$
$R^2 = 0.74$
Adj. $R^2 = 0.73$

Standard errors: OLS

	Est.	S.E.	t val.	p
(Intercept)	0.11	0.03	3.79	0.00
ggastopub	0.76	0.05	16.56	0.00

Evaluar Predicciones

Raíz del error cuadrático medio:

$$\mathrm{RMSD} = \sqrt{\frac{\sum_{t=1}^{T} (\hat{y}_t - y_t)^2}{T}}.$$

Media absoluta del error:

Evaluar Predicciones

Raíz del error cuadrático medio:

$$\mathrm{RMSD} = \sqrt{\frac{\sum_{t=1}^{T} (\hat{y}_t - y_t)^2}{T}}.$$

Media absoluta del error:

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |\hat{y}_t - y_t|}{n}.$$

Para nuestro caso

Raíz del error cuadrático medio:

```
> RMSE(data[,pred],data[,gventas])
[1] 0.1535141
```

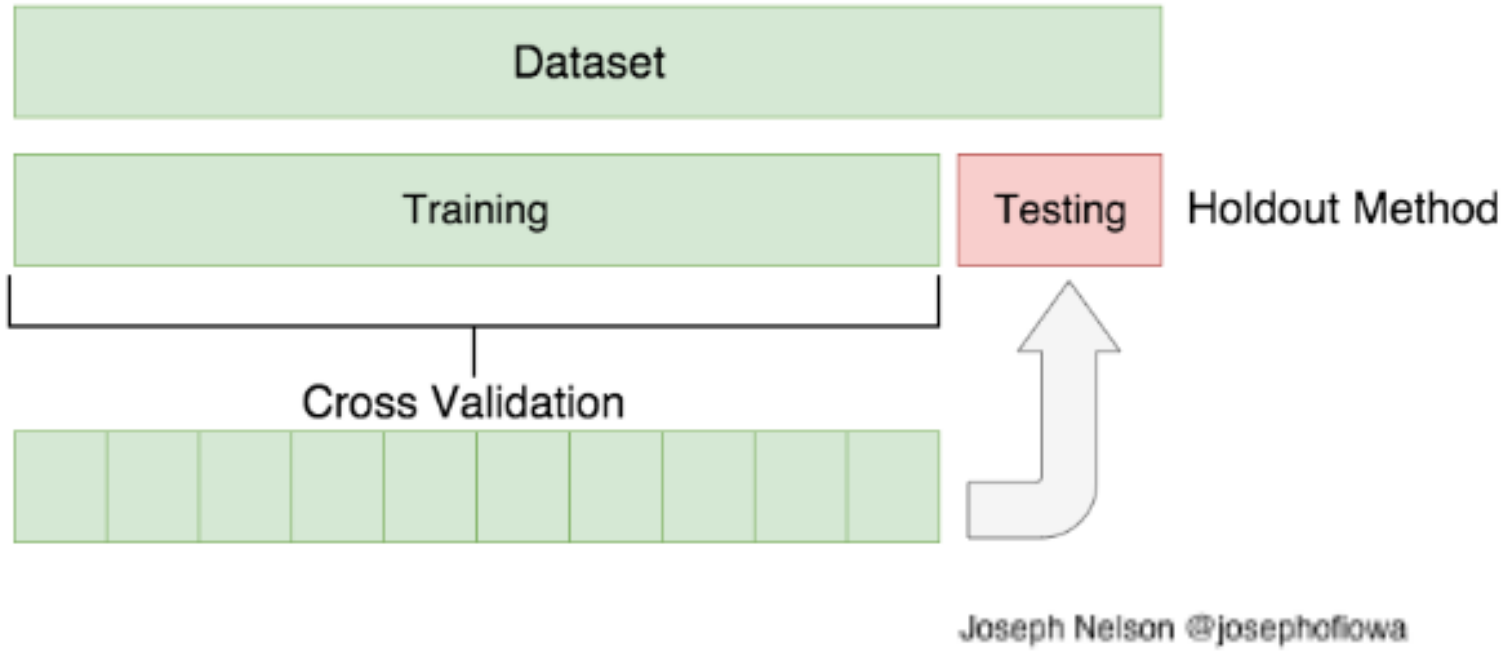
Media absoluta del error:

```
> MAE(data[,pred],data[,gventas])
[1] 0.1231316
```

En la realidad…

- Predecir los valores “fuera de muestra” es siempre un desafío ya que:
  - Las predicciones tienden a ser bastante diferentes a lo que pasa en la muestra.
  - Esto es lógico, ya que se generó la predicción a partir de la muestra.

Validación Cruzada



Validación Cruzada

- Revisaremos dos métodos:
  - Leave one out (LOO).
  - K-folds.
- Pero primero es fundamental introducir una noción básica de los procesos iterativos (control flow).

Procesos Iterativos (Control flow)

- Al trabajar en DS, muchas veces necesitaremos repetir algún proceso: un cálculo, la creación de alguna variable o generar una base consolidada a partir de bases más pequeñas.
- La generación de este proceso en general va a estar condicionada a algún parámetro o característica de los objetos que estemos trabajando.
- Automatizar estos procesos nos permite realizar esto de forma más eficiente que realizando de forma manual. A veces nos permite realizar tareas que de forma manual serían imposible (Ej: LOOCV)

Procesos Iterativos

- En general, la estructura de las funciones será similar:

```
función (condición){acción}
función (condición){acción 1 ; acción 2}
función (condición){acción 1
                    acción 2}
```

- La(s) acción(es) se realiza(n) solo si se cumple la condición(es).
- El cómo se realiza la acción, esta dado por el tipo de función.

IF

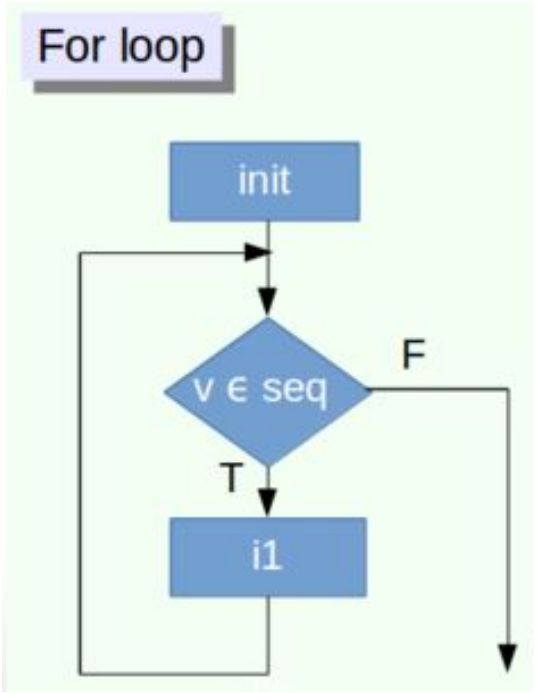
Sí se cumple la condición se realiza una acción.

```
if(condición){acción}
```

FOR

- Para cada elemento del conjunto se realizará la acción que le precede.
- La “condición” es ser parte del conjunto.

```
for(elemento en un conjunto){acción}
```



Validación Cruzada

- La lógica detrás de la validación cruzada es poder testear el “como sería” nuestro modelo con datos que no están en la muestra.
- En esencia, ambos métodos que revisaremos aquí, tienen la misma estructura:
  - Dividir la muestra disponible en N partes.
  - Utilizar N-1 submuestras para correr el modelo (entrenamiento) y la restante para probar que tan bien predice nuestro modelo (test).
  - Realizar este proceso iterativamente para cada una de las N partes, es decir, cada N debe ser usado una vez para predecir.
- La utilidad de este método es que nos permite comparar modelos en términos predictivos (RMSE y MAE).

Validación Cruzada: Leave One Out

- La estructura de este método es la siguiente:
  - Dividir la muestra disponible en 2 partes, una tiene N-1 observaciones y la otra una observación.
  - Utilizar la muestra con N-1 observaciones para correr el modelo (entrenamiento) y la observación restante para probar que tan bien predice nuestro modelo (test).
  - Realizar este proceso iterativamente para cada una de las N observaciones de la base de datos.
- Para bases de datos muy grandes, este método tiene un costo computacional altísimo.

Validación Cruzada: K-folds

- La estructura de este método es la siguiente:
  - Dividir la muestra disponible aleatoriamente en k submuestras, cada una de las k submuestras debe tener similar número de observaciones.
  - Utilizar k-1 submuestras para correr el modelo (entrenamiento) y la submuestra restante para probar que tan bien predice nuestro modelo (test).
  - Realizar este proceso iterativamente para cada una de las k submuestras.
- En general, la literatura recomienda usar k=5 o k=10.

Lab 10

- En primer lugar, programaremos nuestra propia validación cruzada(CV) con las herramientas vistas en el curso más, lo que revisamos esta clase de procesos iterativos.
- Posteriormente utilizaremos las bondades del paquete "**caret**", para realizar CV. Las funciones a utilizar son las siguientes:
  - **trainControl():** Nos permite elegir el tipo de validación cruzada y utilizarlo como argumento de la función **train()**. Argumentos:
    - method: método de CV a utilizar. Para kfolds method="cv" y para LOO method="LOOCV".
    - number: En los métodos k folds y relacionados, nos permite elegir el número de submuestras (valor de k)
  - **train():** Nos permite realizar CV, dadas las opciones que seteamos en el objeto creado a partir de la función **trainControl()**. Arg:
    - formula: modelo (misma sintaxis que para regresión lineal)
    - data= base de datos
    - method= tipo de modelo a evaluar, para regresión lineal method="lm".
    - trControl= se inserta el objeto creado con **trainControl()**.