tarea_4_nicolas_netz

July 11, 2023

1 Laboratorio de métodos aplicados: Tarea 4

Estudiante: Nicolás Netz

N°Mat: 2018458791

Correo: nnetz2018@udec.cl

Prof. Juan Carlos Caro

2 Importar los módulos necesarios y la data

En el presente notebook se utilizan distintas librerías para apoyar los cálculos y las visualizaciones. Además, se incluye una serie de funciones extra, para reutilizar código en distintas celdas.

3 Pregunta 1

Utilizando el set de datos *junaeb2.csv* realice una regresion para predecir la variable *imce* con regularizacion via Lasso con cross-validation. Muestre que sus resultados son robustos a la seleccion de hiperparametros y compute una metrica de calidad de ajuste del modelo.

Tabla 1: Vista general de los datos en junaeb2.csv

	sexo	edad	imce	vive_padre	vive_madre	sk1	sk2	sk3	sk4	sk5	…	\
1	0	76	0.71	0	1	1	1	1	1	1	…	
3	1	84	2.05	1	1	1	1	1	1	1	…	
4	0	86	1.05	1	1	1	1	1	1	1	…	
5	0	74	1.39	1	1	1	2	1	1	1	…	
6	1	91	2.75	1	1	1	1	1	2	2	…	

	sk9	sk10	sk11	sk12	sk13	act_fisica	area	educm	educp	madre_work
1	1	1	1	1	1	5.0	0	8.0	8	1
3	1	1	1	1	1	2.0	1	16.0	12	-1
4	1	1	1	1	1	1.0	1	17.0	15	0
5	1	1	1	1	1	4.0	0	8.0	8	-1
6	3	3	3	2	2	2.0	1	20.0	19	1

[5 rows x 23 columns]

[Text(0.5, 1.0, 'Figura 1: Distribución de variable IMCE')]

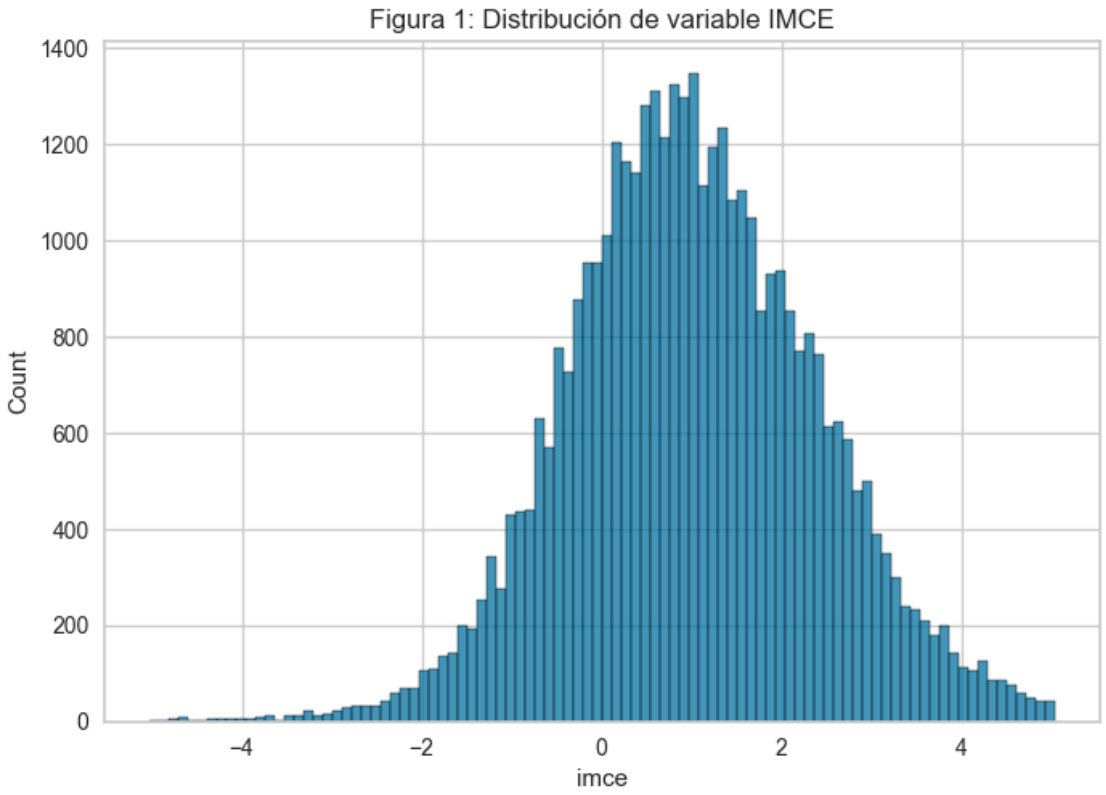


Tabla 2: Estadísticos descriptivos de las variables

	sexo	edad	vive_padre	vive_madre	sk1	\
count	39898.000000	39898.000000	39898.000000	39898.000000	39898.000000	
mean	0.552409	83.022006	0.719284	0.975713	1.111885	
std	0.497252	3.938669	0.450246	0.164488	0.385352	
min	0.000000	62.000000	0.000000	0.000000	1.000000	
25%	0.000000	81.000000	0.000000	1.000000	1.000000	
50%	1.000000	82.000000	1.000000	1.000000	1.000000	
75%	1.000000	84.000000	1.000000	1.000000	1.000000	
max	1.000000	107.000000	2.000000	2.000000	5.000000	

	sk2	sk3	sk4	sk5	sk6	\
count	39898.000000	39898.000000	39898.000000	39898.000000	39898.000000	
mean	1.391874	1.263271	1.256730	1.271793	1.491002	
std	0.653606	0.583646	0.578441	0.567844	0.739283	
min	1.000000	1.000000	1.000000	1.000000	1.000000	
25%	1.000000	1.000000	1.000000	1.000000	1.000000	
50%	1.000000	1.000000	1.000000	1.000000	1.000000	

```
75%        2.000000      1.000000      1.000000      1.000000      2.000000
max        5.000000      5.000000      5.000000      5.000000      5.000000

       …           sk9           sk10          sk11          sk12  \
count  …  39898.000000  39898.000000  39898.000000  39898.000000
mean   …      1.337862      1.877087      1.390596      1.502958
std    …      0.669773      0.950319      0.674868      0.800377
min    …      1.000000      1.000000      1.000000      1.000000
25%    …      1.000000      1.000000      1.000000      1.000000
50%    …      1.000000      2.000000      1.000000      1.000000
75%    …      2.000000      3.000000      2.000000      2.000000
max    …      5.000000      5.000000      5.000000      5.000000

                sk13    act_fisica          area         educm         educp  \
count  39898.000000  39898.000000  39898.000000  39898.000000  39898.000000
mean       1.708030      2.552409      0.911976     13.015916     12.947942
std        0.995917      1.069471      0.283334      3.365582      3.452305
min        1.000000      1.000000      0.000000      0.000000      0.000000
25%        1.000000      2.000000      1.000000     11.000000     11.000000
50%        1.000000      2.000000      1.000000     13.000000     13.000000
75%        2.000000      3.000000      1.000000     15.000000     14.000000
max        5.000000      5.000000      1.000000     22.000000     22.000000

          madre_work
count  39898.000000
mean       0.098150
std        0.941687
min       -1.000000
25%       -1.000000
50%        0.000000
75%        1.000000
max        1.000000

[8 rows x 22 columns]
```

A partir de la Tabla 2, se revisan los estadísticos descriptivos de los features del modelo. Se nota que se presentan 39898 observaciones en total. En la Tabla 2 se pueden ver las medias, las desviaciones estándar y los valores mínimos y máximos de las distintas variables que se usarán como features en la regresion con Lasso.

Se separa el dataset en dos partes, train y test, usando un tamaño de 20% para el dataset de prueba (test). A este dataset se le aplica una regresión Lasso con valor $\alpha = 0.1$ para controlar la intensidad de la regularización. A partir de este modelo, se obtiene que el RMSE (Root Mean Squared Error) es de 1.3756. Así, se obtiene un modelo de regresión con regularización Lasso que predice la variabl *imce*.

```
Root Mean Squared Error: 1.3756663784187488
```

Además de lo anterior, se aplica otra metodología, en la que se aplica Cross Validation. En este

caso la data se separa en 100 partes, 3 veces. Así, en este caso se aplica Lasso 100 veces. Dado este experimento, se obtiene un valor $RMSE$ promedio similar a lo obtenido en el experimento anterior, y se puede ver que el resultado es robusto, pues la desviación estándar de la métrica a lo largo de estos KFolds es baja. La métrica de calidad de ajuste en este caso entregó un resultado de $1.379 + -0.062$

```
Mean RMSE: 1.379 (0.062)
```

```
Tabla 3: Resultados GridSearchCV - Lasso
```

kernel	params	rank_test_score	mean_test_score	std_test_score
0.1_10000_0.0001	{'alpha': 0.1, 'max_iter': 10000, 'tol': 0.0001}	1	-0.000306	0.000281
0.5_10000_0.0001	{'alpha': 0.5, 'max_iter': 10000, 'tol': 0.0001}	1	-0.000306	0.000281
1_10000_0.0001	{'alpha': 1, 'max_iter': 10000, 'tol': 0.0001}	1	-0.000306	0.000281
5_10000_0.0001	{'alpha': 5, 'max_iter': 10000, 'tol': 0.0001}	1	-0.000306	0.000281
10_10000_0.0001	{'alpha': 10, 'max_iter': 10000, 'tol': 0.0001}	1	-0.000306	0.000281
100_10000_0.0001	{'alpha': 100, 'max_iter': 10000, 'tol': 0.0001}	1	-0.000306	0.000281

Finalmente, podemos ver en la Tabla 3 que el modelo es robusto a la selección de hiperparametros según la métrica de mean test score (y su respectiva desviación estándar), pues para distintos valores de alpha, se obtuvo en todos los casos el mismo valor.

4 Pregunta 2

Utilizando el set de datos *charls2.csv* realice una clasificacion de la variable *retired* usando Random Forest sobre las demas variables del dataset con cross-validation. Muestre que sus resultados son sensibles a la seleccion de hiperparametros y compute una metrica de calidad de ajuste del modelo.

```
Tabla 4: Datos charls.csv
```

	age	cesd	child	drinkly	female	hrsusu	hsize	intmonth	married	retage	retin	retired	schadj	urban	wealth
0	46	6.0	2	0.0	1	0.000000	4	7	1	24	1	0	0	0	-5800.0
1	48	0.0	2	1.0	0	4.143135	4	7	1						
2	56	6.0	1	0.0	1	0.000000	6	8	1						
3	59	6.0	1	1.0	0	0.000000	6	8	1						
4	47	4.0	1	1.0	0	3.806663	3	8	1						

```
1      22      1        0       4      0 -5800.0
2       0      0        0       0      0   350.0
3       0      0        0       0      0   350.0
4      11      1        0       4      0 -8100.0
```

```
GridSearchCV(cv=10, estimator=RandomForestClassifier(), n_jobs=-1,
             param_grid={'bootstrap': [True], 'max_depth': [4, 8, 16],
                         'max_features': [6, 8], 'min_samples_leaf': [3, 4, 5],
                         'min_samples_split': [8, 10, 12],
                         'n_estimators': [100, 200, 300]})
```

Tabla 5 - Resultados GridSearchCV RandomForestClassifier

```
                                                                  params  \
kernel
True_16_6_3_12_300  {'bootstrap': True, 'max_depth': 16, 'max_feat…
True_8_6_5_12_300   {'bootstrap': True, 'max_depth': 8, 'max_featu…
True_16_6_4_8_100   {'bootstrap': True, 'max_depth': 16, 'max_feat…
True_16_6_4_12_300  {'bootstrap': True, 'max_depth': 16, 'max_feat…
True_16_6_3_8_300   {'bootstrap': True, 'max_depth': 16, 'max_feat…
…                                                                 …
True_4_8_3_12_300   {'bootstrap': True, 'max_depth': 4, 'max_featu…
True_16_8_4_10_100  {'bootstrap': True, 'max_depth': 16, 'max_feat…
True_4_6_4_8_100    {'bootstrap': True, 'max_depth': 4, 'max_featu…
True_4_6_3_10_100   {'bootstrap': True, 'max_depth': 4, 'max_featu…
True_4_8_5_8_100    {'bootstrap': True, 'max_depth': 4, 'max_featu…

                    rank_test_score  mean_test_score  std_test_score
kernel
True_16_6_3_12_300                1         0.906557        0.010915
True_8_6_5_12_300                 2         0.906093        0.009508
True_16_6_4_8_100                 3         0.905475        0.011919
True_16_6_4_12_300                4         0.905475        0.012160
True_16_6_3_8_300                 5         0.905475        0.012905
…                               …                 …               …
True_4_8_3_12_300               158         0.902072        0.007203
True_16_8_4_10_100              159         0.902070        0.013538
True_4_6_4_8_100                160         0.901299        0.007697
True_4_6_3_10_100               161         0.901298        0.006393
True_4_8_5_8_100                162         0.900989        0.007867

[162 rows x 4 columns]
```

Se utilizaron los datos de *charls2.csv* presentados brevemente en la Tabla 4. Luego, se ejecutó un modelo de Random Forest Classifier sobre un GridSearchCV. Las features fueron estandarizadas usando StandardScaler y el target correspondió a "retired"

Para el GridSearchCV se revisaron los siguientes parámetros: * bootstrap: [True] * max_depth:

[4, 8, 16] * max_features: [6,8] * min_samples_leaf: [3,4,5] * min_samples_split: [8,10,12] * n_estimators: [100,200,300]

Así, se entrenaron 162 configuraciones de modelos sobre una Cross Validation de 10 dobleces, obteniendo las métricas de evaluación rankeadas en la Tabla 5. Se puede ver que el modelo es sensible a la selección de hiperparámetros, pues se tienen distintos valores en la columna mean test score, aunque se podría argumentar que en la grilla elegida no son tan significativamente diferentes, pues el mean test score no varía sino desde el orden de 10^{-4}. Otras métricas de calidad de ajuste se pueden aplicar. A continuación se presenta el valor de los parámetros y el Accuracy en Cross validation del mejor estimador obtenido por el GridSearch:

```
RandomForestClassifier(max_depth=16, max_features=6, min_samples_leaf=3,
                       min_samples_split=12, n_estimators=300)
Accuracy CV: 0.905629
```

5 Pregunta 3

Repita el analisis de la Pregunta 2 usando Stacking, con tres modelos (Random Forest, Gradient Boosting y SVM). Muestre que sus resultados son robustos a la seleccion de hiperparametros y compute una metrica de calidad de ajuste del modelo.

```
GridSearchCV(cv=10,
             estimator=StackingClassifier(estimators=[('rf',
RandomForestClassifier()),
                                                      ('boost',
GradientBoostingClassifier()),
                                                      ('svc', SVC())],
                                          final_estimator=LogisticRegression()),
             n_jobs=-1,
             param_grid={'boost__learning_rate': [0.02],
                         'boost__max_depth': [4],
                         'boost__n_estimators': [100, 150],
                         'boost__subsample': [0.9], 'rf__bootstrap': [True],
                         'rf__max_depth': [8], 'rf__max_features': [6],
                         'rf__min_samples_leaf': [4],
                         'rf__min_samples_split': [10],
                         'rf__n_estimators': [100], 'svc__C': [0.01],
                         'svc__gamma': [1], 'svc__kernel': ['rbf']})
```

Tabla 6 - Resultados GridSearchCV StackingClassifier

```
        params  \
kernel
0.02_4_150_0.9_True_8_6_4_10_100_0.01_1_rbf  {'boost__learning_rate': 0.02,
'boost__max_dep…
0.02_4_100_0.9_True_8_6_4_10_100_0.01_1_rbf  {'boost__learning_rate': 0.02,
'boost__max_dep…
```

```
                                                rank_test_score  mean_test_score  \
kernel
0.02_4_150_0.9_True_8_6_4_10_100_0.01_1_rbf                   1         0.905473
0.02_4_100_0.9_True_8_6_4_10_100_0.01_1_rbf                   2         0.903926

                                                std_test_score
kernel
0.02_4_150_0.9_True_8_6_4_10_100_0.01_1_rbf        0.013582
0.02_4_100_0.9_True_8_6_4_10_100_0.01_1_rbf        0.012401

The best estimator has:
StackingClassifier(estimators=[('rf',
                                RandomForestClassifier(max_depth=8,
                                                       max_features=6,
                                                       min_samples_leaf=4,
                                                       min_samples_split=10)),
                               ('boost',
                                GradientBoostingClassifier(learning_rate=0.02,
                                                           max_depth=4,
                                                           n_estimators=150,
                                                           subsample=0.9)),
                               ('svc', SVC(C=0.01, gamma=1))],
                   final_estimator=LogisticRegression())

Accuracy:  0.9127475247524752
Accuracy CV: 0.903619
```

En esta sección se plantea un modelo Stacking, donde se incluyen 3 modelos, RandomForest, Boosting y Support Vector Classifier. Se aplica una busqueda GridSearch sobre los parámetros de Boosting y SVC según: * 'boost___learning_rate': [0.02], * 'boost___subsample' : [0.9], * 'boost___n_estimators' : [100, 150], * 'boost___max_depth' : [4], * 'svc___C': [0.01], * 'svc___gamma': [1], * 'svc___kernel': ['rbf'],

Lo que equivale a solo 2 ajustes de modelo. Hubiese sido conveniente explorar el espacio con más valores, pero eran demasiados y generaba un cuello de botella en mi computador, no entregandome resultados en menos de 10 minutos. Según lo observado en la Tabla 6, se puede ver que al cambiar los hiperparámetros, cambian los scores de prueba, indicando que el modelo no es robusto a la selección de hiperparametros. Quizá es una baja varianza, pero habría que explorar mejor el espacio de parámetros posibles.

Se obtiene un accuracy de 0.909 y de 0.903 para el CV, indicando una buena performance (ajuste) del modelo

6 Pregunta 4

Utilizando la base de datos *enia.csv* realice un analisis de cluster usando k-means incluyendo todas las variables excepto *tamano* y *ID*. Muestre que sus resultados son sensibles a la seleccion de hiperparametros y compute una metrica de calidad de ajuste del modelo.

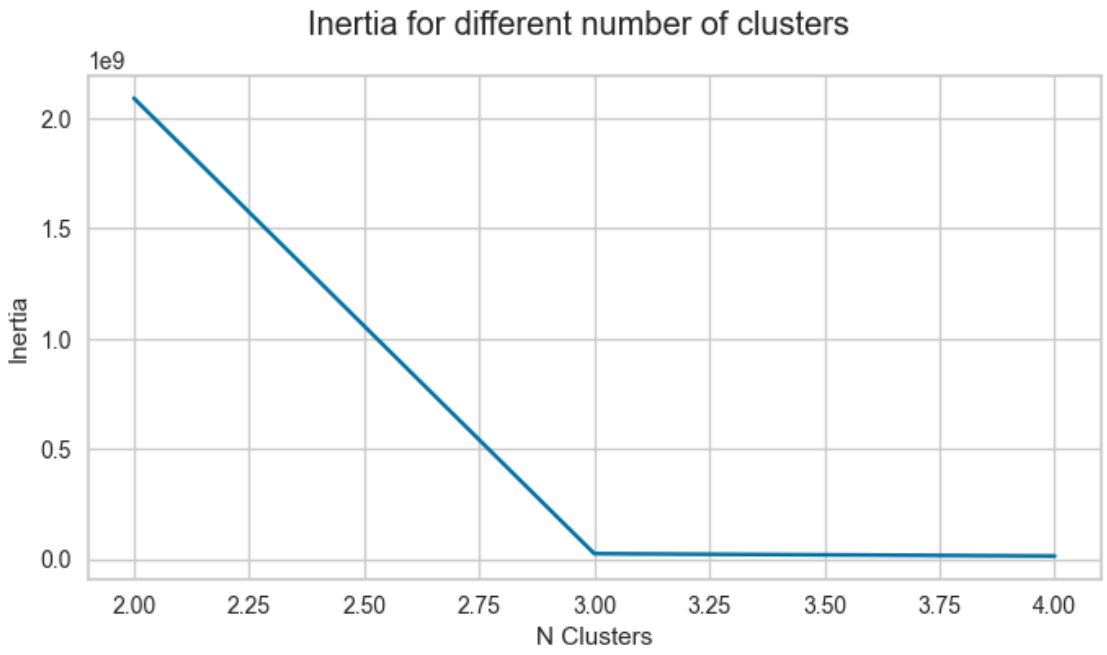
Tabla 7 - Datos enia.csv

```
         id  year  tamano     sales  age  foreign  export   workers  fomento  \
0  100003  2007       1  7.046558   22        1     1.0  3.486855        0
1  100003  2009       1  7.875563   24        1     1.0  3.504607        0
2  100003  2013       1  7.437399   23        1     1.0  4.691621        1
3  100003  2015       1  7.356472   30        1     1.0  4.682614        1
4  100003  2017       1  4.014772   32        1     1.0  4.611691        1

   iyd  impuestos  utilidades
0    1   1.231345    7.113892
1    0   8.762233   33.976108
2    1   0.001886    3.137265
3    0   0.411670    1.298413
4    0   0.000721    0.001949
```

```
/Users/nicolas/.virtualenvs/lab-maa/lib/python3.8/site-
packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of
`n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init`
explicitly to suppress the warning
  warnings.warn(
/Users/nicolas/.virtualenvs/lab-maa/lib/python3.8/site-
packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of
`n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init`
explicitly to suppress the warning
  warnings.warn(
/Users/nicolas/.virtualenvs/lab-maa/lib/python3.8/site-
packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of
`n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init`
explicitly to suppress the warning
  warnings.warn(
/var/folders/04/419yynhj4_qgs5fqm07823c00000gn/T/ipykernel_53962/598443739.py:12
: UserWarning: Matplotlib is currently using
module://matplotlib_inline.backend_inline, which is a non-GUI backend, so cannot
show the figure.
  fig.show()
/Users/nicolas/.virtualenvs/lab-maa/lib/python3.8/site-
packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of
`n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init`
explicitly to suppress the warning
  warnings.warn(
```

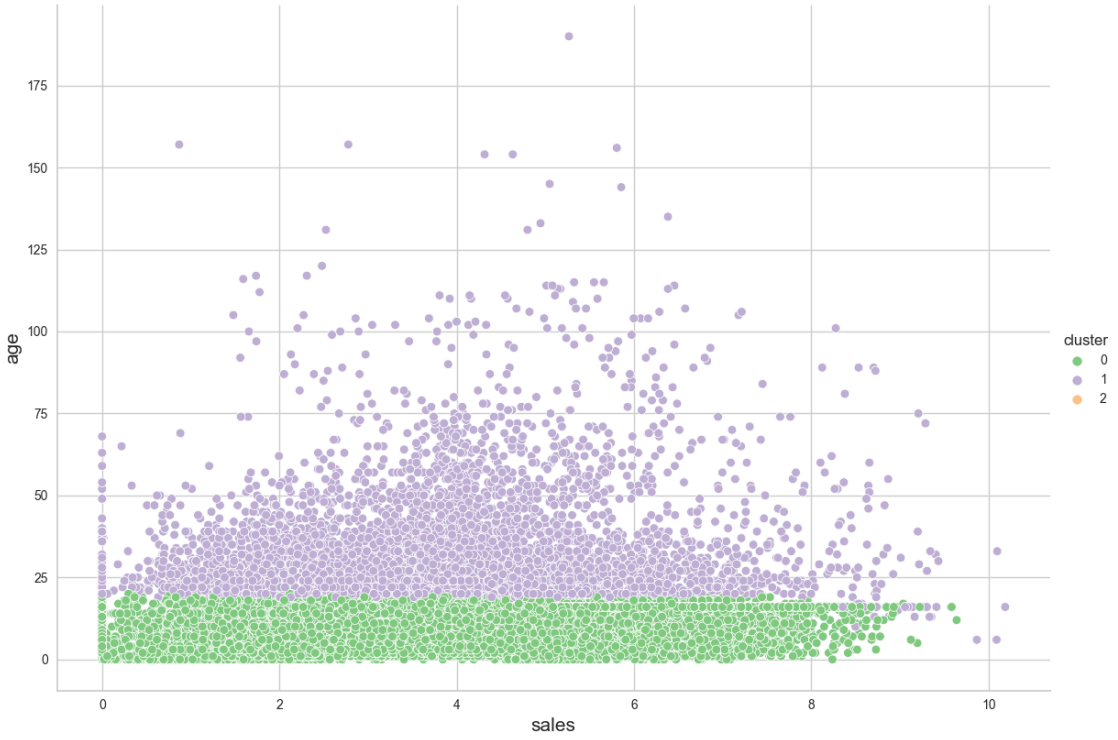
```
0    31281
1        1
2        1
dtype: int64
```




```
/Users/nicolas/.virtualenvs/lab-maa/lib/python3.8/site-
packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of
`n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init`
explicitly to suppress the warning
  warnings.warn(

 Text(35.61597189247652, 0.5, 'age')

<Figure size 1500x800 with 0 Axes>
```



Al ver la figura "Inertia for different number of clusters", notamos que el codo se produce abruptamente cuando se tienen 3 clusters. Así, en la siguiente etapa se identifica cómo los clusters están relacionados a la variable edad y trabajadores, donde se puede ver una relación lineal que: A medida que la empresa es más vieja, sin importar la cantidad de trabajadores, esta pertenece a otra categoría. Finalmente, se puede decir que el modelo es sensible a la selección de hiperparámetros, pues si se varía el número de clusters, por ejemplo a 1, análisis como el anterior no serían posibles, y también se tendría una falta de uso de lo interpretado a partir de la figura "Inertia for different number of clusters."