Clasificador de ganancias a partir de variables socioeconómicas

Francisco Eduardo Tagliavini

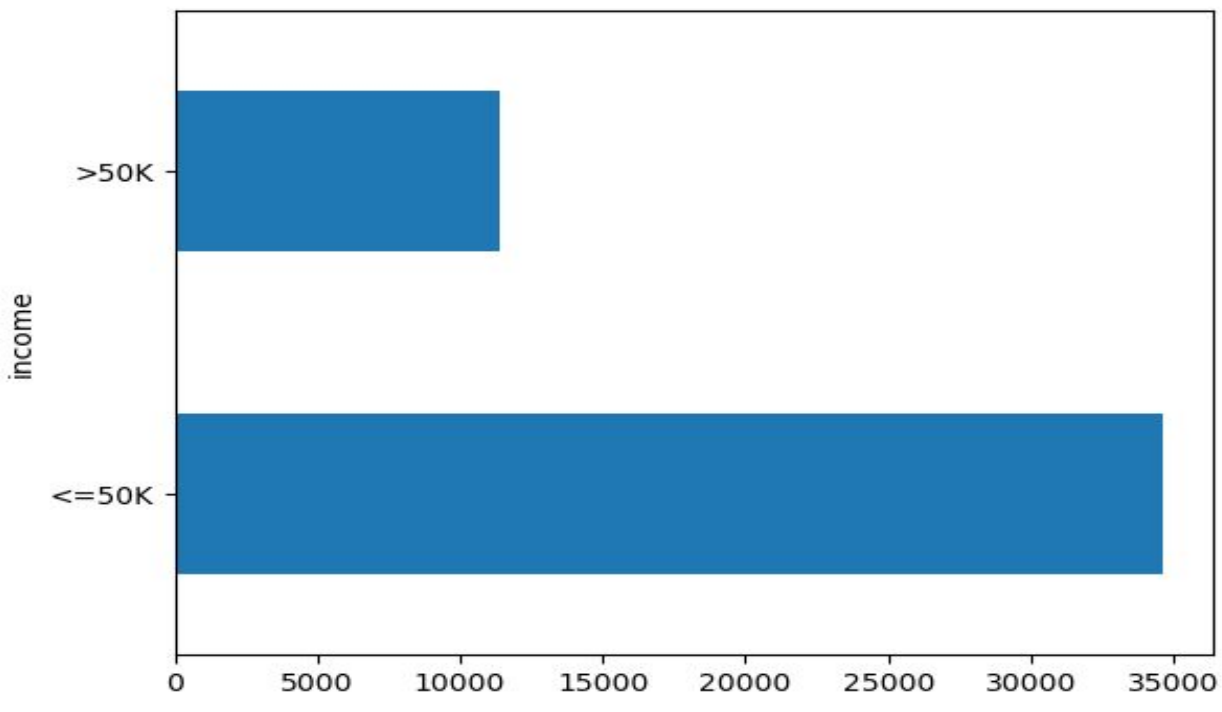
Diccionario de categorías iniciales del datsets

age	74
workclass	9
fnlwgt	28523
education	16
marital-status	7
occupation	15
relationship	6
race	5
sex	2
hours-per-week	96
native-country	42
income	4

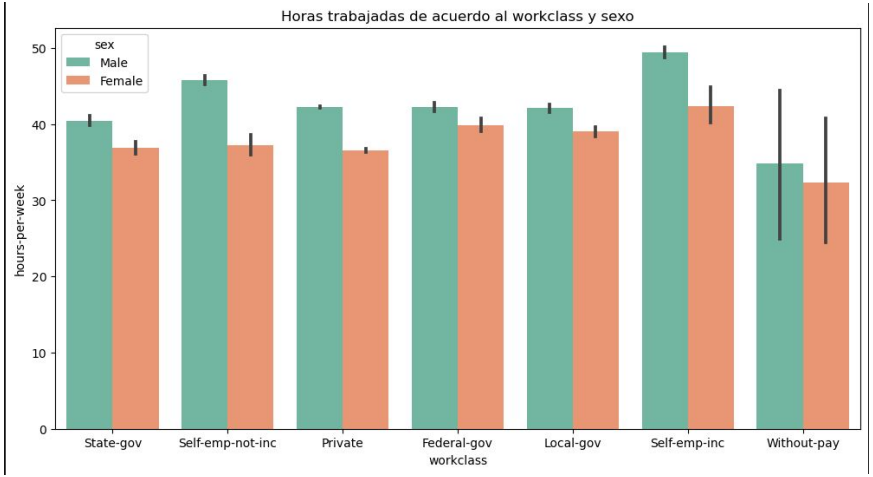
Diccionario de categorías del dataset final

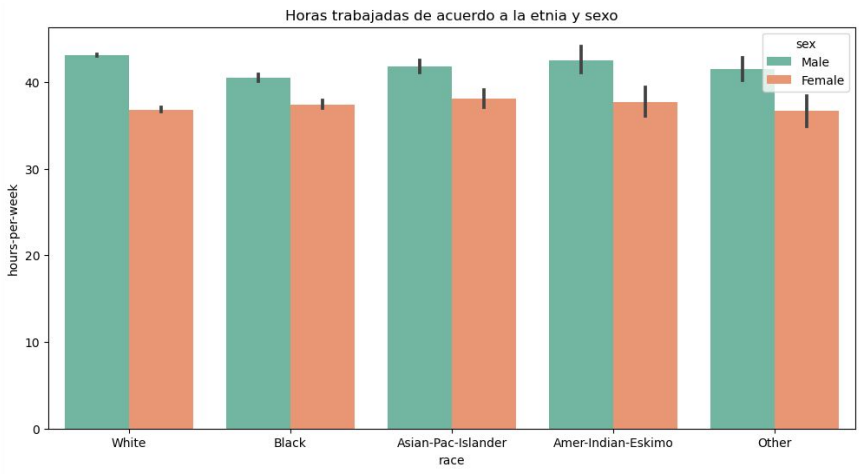
encoder_workclass	7
encoder_education	16
encoder_marital-status	5
encoder_occupation	14
encoder_relationship	6
encoder_race	5
encoder_sex	2
encoder_native-country	42
encoder_income	2
encoder_encoder_age	9
encoder_encoder_hpw	9

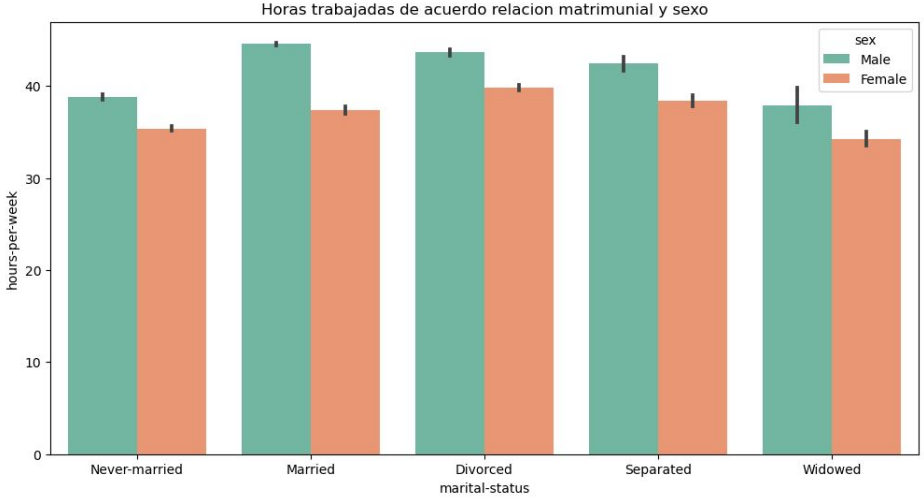
Nuestra variable target

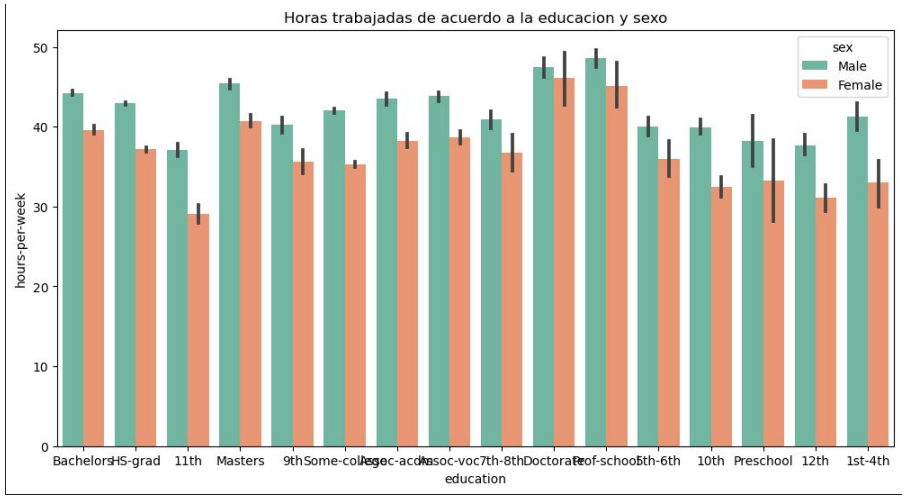


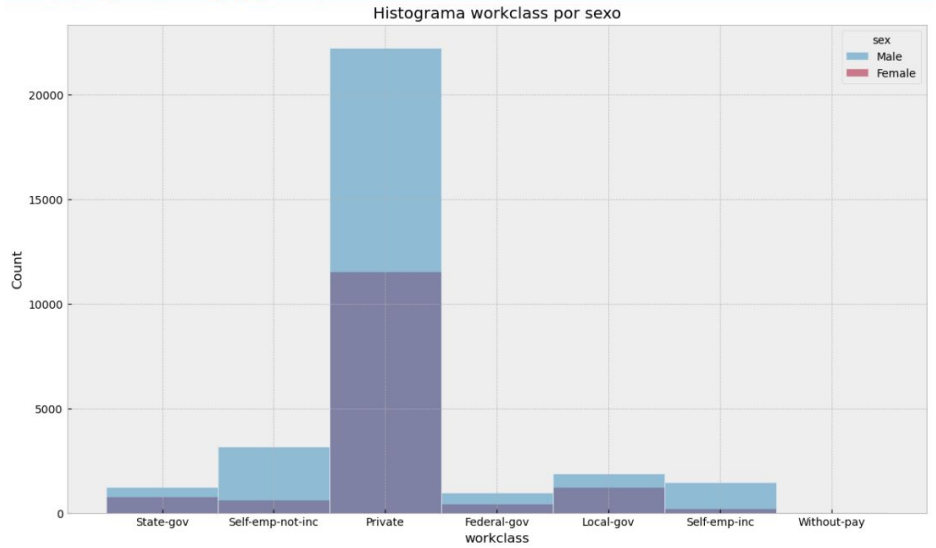
Algunas aproximaciones





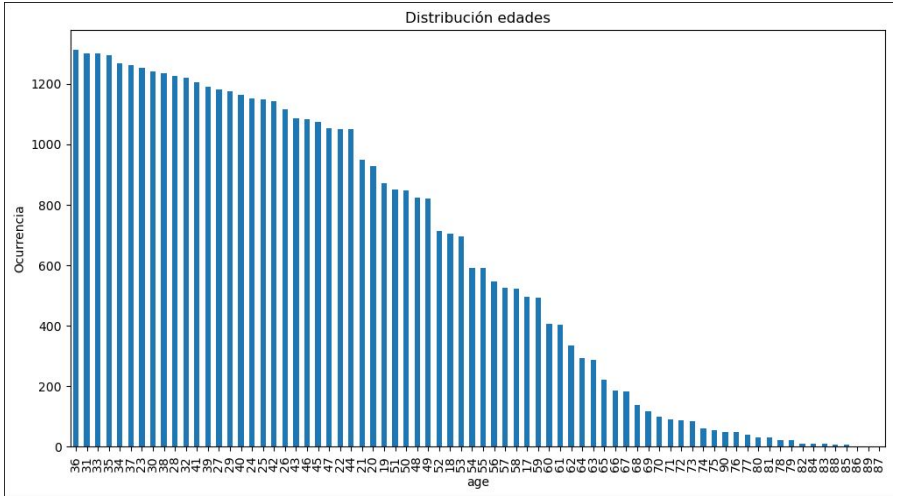

Horas trabajadas de acuerdo relacion matrimunial y sexo
sex
Male
Female
hours-per-week
0
10
20
30
40
Never-married
Married
Divorced
Separated
Widowed
marital-status

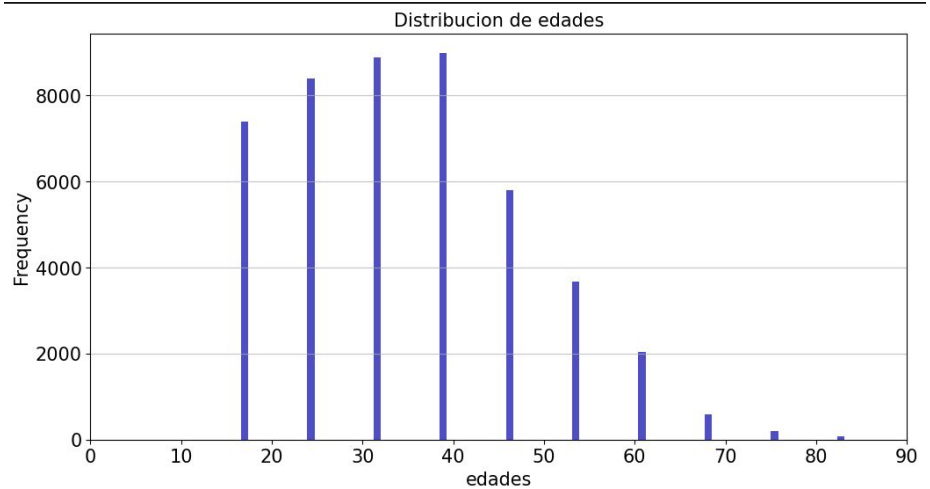

Horas trabajadas de acuerdo a la educacion y sexo
sex
Male
Female
hours-per-week
0
10
20
30
40
50
Bachelors
HS-grad
11th
Masters
9th
Some-college
Assoc-acdm
Assoc-voc
7th-8th
Doctorate
Prof-school
5th-6th
10th
Preschool
12th
1st-4th
education


Histograma workclass por sexo
sex
Male
Female
20000
15000
10000
5000
0
Count
State-gov
Self-emp-not-inc
Private
Federal-gov
Local-gov
Self-emp-inc
Without-pay
workclass

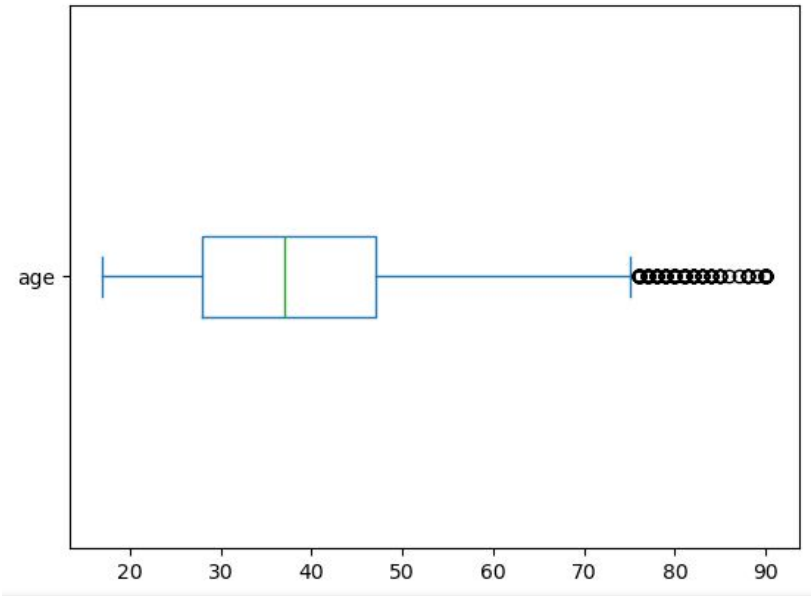

Distribución por sexo en el dataset
Male
Female
Male
Female

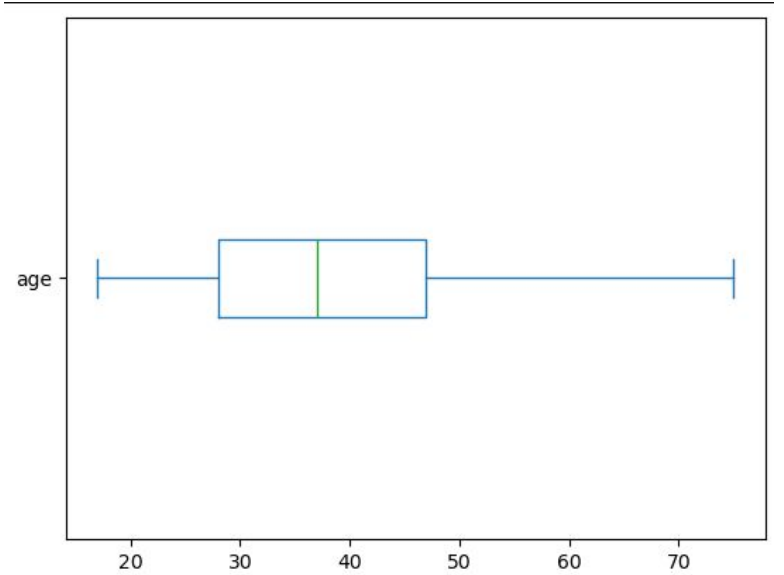
Mejoras en el campo de edad



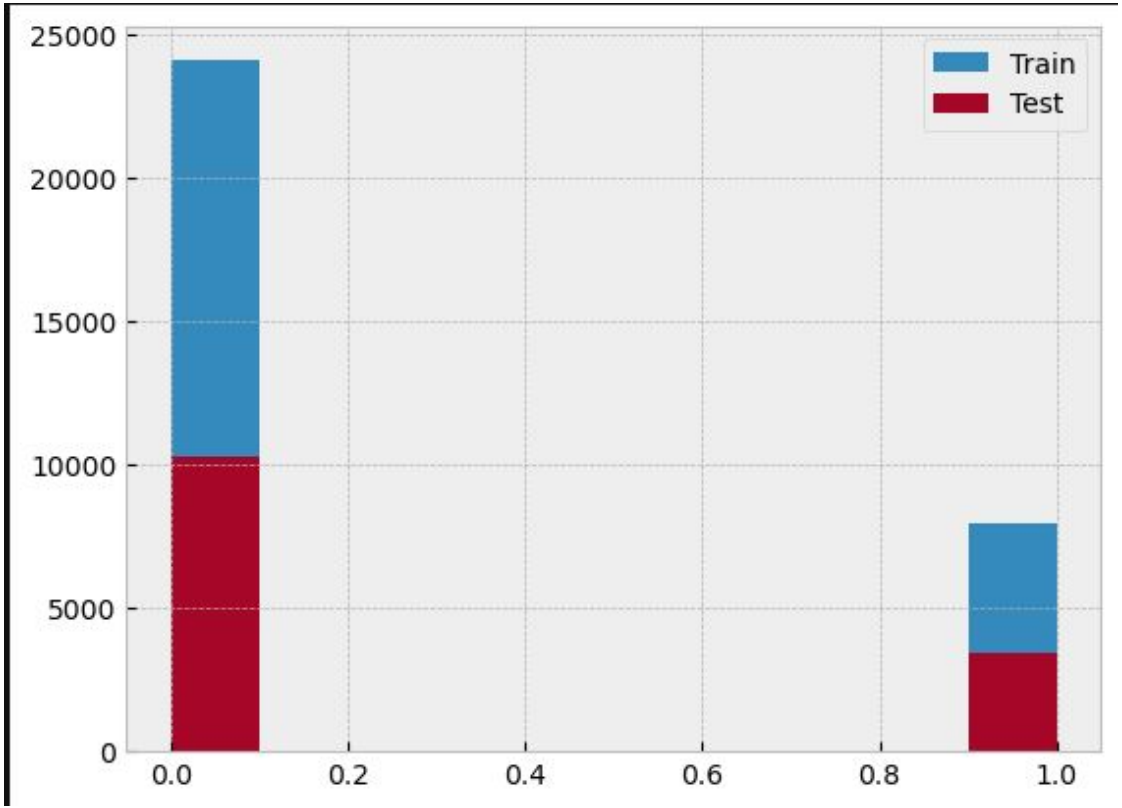


Mejoras en el campo edad

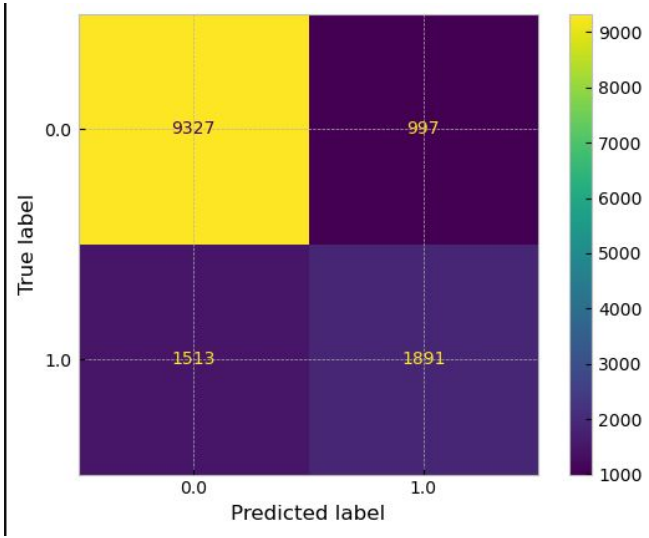




Muestras del modelo Test vs. Train



Random forest



	feature	importance
1	encoder_education	0.163833
8	encoder_encoder_age	0.155769
4	encoder_relationship	0.151174
3	encoder_occupation	0.149526
9	encoder_encoder_hpw	0.110338
2	encoder_marital-status	0.105351
0	encoder_workclass	0.079716
7	encoder_native-country	0.036401
5	encoder_race	0.029248
6	encoder_sex	0.018644

```
Modelo Random Forest Classifier accuracy score criterio "gini": 0.8172
```

Modelo de Random Forest

stratifiedkfold

split: 5

random_state=1

n_estimators=30

criterion="gini"

max_depth=4

```
Iteracion:  1  Accuracy: 0.796875
Iteracion:  2  Accuracy: 0.7934877622377622
Iteracion:  3  Accuracy: 0.8014641608391608
Iteracion:  4  Accuracy: 0.7965472027972028
Iteracion:  5  Accuracy: 0.8046115178668998
```

Modelo de Random Forest

random_state=11

n_jobs=6

Con iteración de parámetros

```
% de aciertos sobre el set de evaluación: 0.8161421911421911
```

```
# Definir la grilla de los parametros, cada combinación es un modelo adicional
param_grid = {'n_estimators': [4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048],
              'max_features': [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]}
```

```
GS_RF.best_params_
```

```
{'max_features': 0.1, 'n_estimators': 512}
```

```
% de aciertos sobre el set de evaluación: 0.8161421911421911
```

```python
# Que un algoritmo busque los mejores parámetros dentro del rango
from scipy.stats import uniform, randint

param_dist = {"n_estimators": randint(4, 2048),
              "max_features": uniform(0, 1)}

# y el número de iteraciones, dependiendo de esto se consume más tiempo

iteraciones = 10
```

```python
RS_RF.best_params_
```

```
{'max_features': 0.3944636253118162, 'n_estimators': 1963}
```

```
% de aciertos sobre el set de evaluación: 0.81745337995338
```

Super vector machine

Lineal:

```
% de aciertos sobre el set de evaluación: 0.752039627039627
```

Poly:

```
% de aciertos sobre el set de evaluación: 0.7961101398601399
```

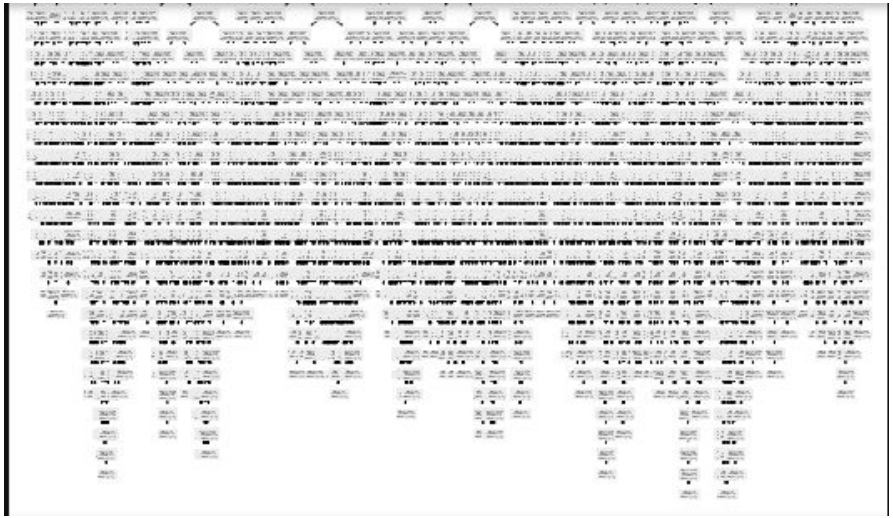
rfb:

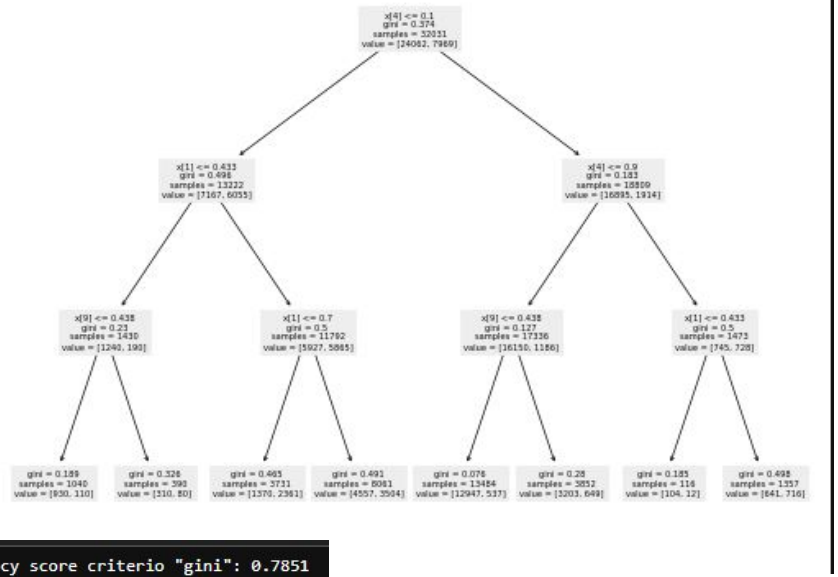
```
% de aciertos sobre el set de evaluación: 0.8049970862470862
```

Sigmoid:

```
% de aciertos sobre el set de evaluación: 0.6873543123543123
```

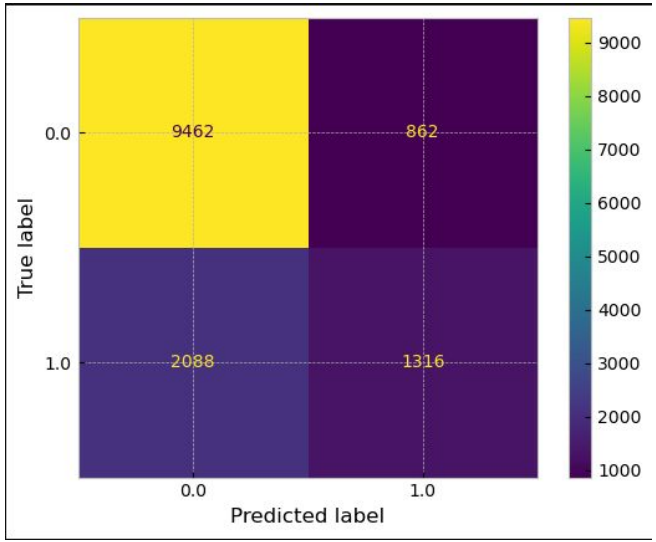
Árbol de decisión





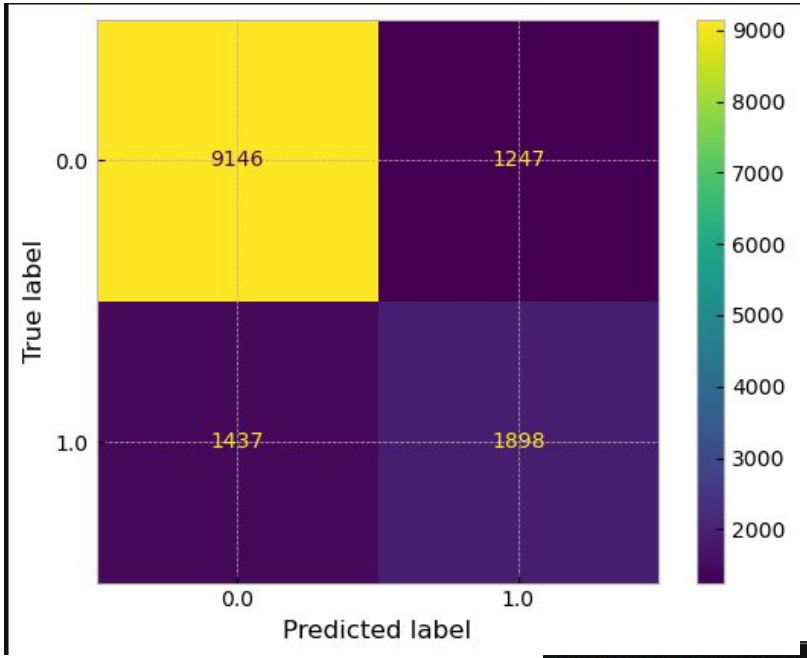

```
Modelo Decision Tree Classifier accuracy score criterio "gini": 0.7851
```

Árbol de decisión



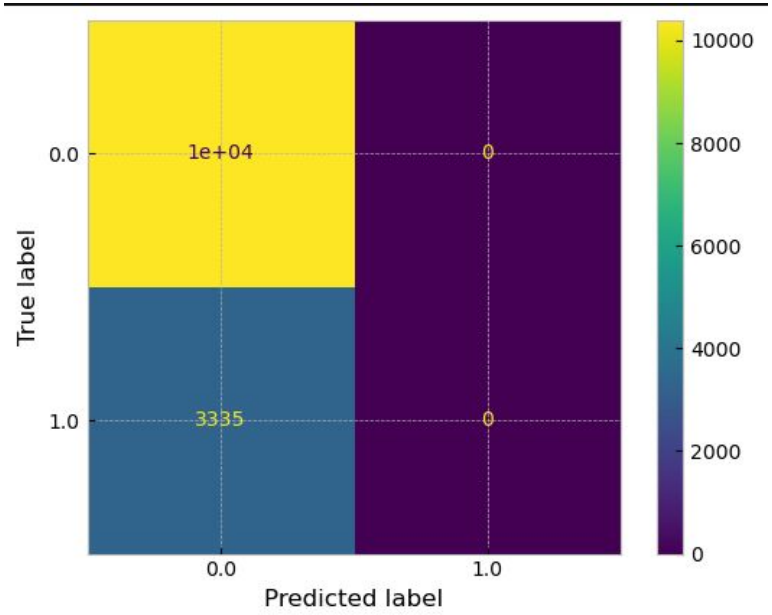
	feature	importance
4	encoder_relationship	0.290222
1	encoder_education	0.180197
8	encoder_encoder_age	0.126108
3	encoder_occupation	0.113122
9	encoder_encoder_hpw	0.102455
0	encoder_workclass	0.077193
7	encoder_native-country	0.041707
5	encoder_race	0.035536
2	encoder_marital-status	0.023592
6	encoder_sex	0.009868

Vecinos cercanos




```
Modelo vecinos cercanos Classifier accuracy score: 0.8045
```

Super Vector Machine




```
Modelo Super vector machine Classifier accuracy score: 0.7571
```

Resultados

Se probaron varios modelos.

Se setearon rangos de parámetros

Se probaron algoritmos para la determinación de parámetros de los modelos

Se llegó a un modelos que superó el 80% de precisión