Results Model Adult Set – 2° approach

The train and test datasets are quite well distributed, as it is possible to see in summary table below:

Train						
Numeric Columns						
idx	age	fnlwgt	education-num	capital-gain	capital-loss	hours-per-week
nulls	0.00	0.00	0.00	0.00	0.00	0.00
count	32,561.00	32,561.00	32,561.00	32,561.00	32,561.00	32,561.00
mean	38.58	189,778.37	10.08	1,077.65	87.30	40.44
std	13.64	105,549.98	2.57	7,385.29	402.96	12.35
min	17.00	12,285.00	1.00	0.00	0.00	1.00
25%	28.00	117,827.00	9.00	0.00	0.00	40.00
50%	37.00	178,356.00	10.00	0.00	0.00	40.00
75%	48.00	237,051.00	12.00	0.00	0.00	45.00
max	90.00	1,484,705.00	16.00	99,999.00	4,356.00	99.00

String Columns									
idx	workclass	education	marital-status	occupation	relationship	race	sex	native-country	Target
nulls	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
count	30725	32561	32561	30718	32561	32561	32561	31978	32561
unique	8	16	7	14	6	5	2	41	2
top	PRIVATE	HS-GRAD	MARRIED-CIV-SPOUSE	PROF-SPECIALTY	HUSBAND	WHITE	MALE	UNITED-STATES	<=50K
freq	22696	10501	14976	4140	13193	27816	21790	29170	24720

Test						
Numeric Columns						
idx	age	fnlwgt	education-num	capital-gain	capital-loss	hours-per-week
nulls	0.00	0.00	0.00	0.00	0.00	0.00
count	16,281.00	16,281.00	16,281.00	16,281.00	16,281.00	16,281.00
mean	38.77	189,435.68	10.07	1,081.91	87.90	40.39
std	13.85	105,714.91	2.57	7,583.94	403.11	12.48
min	17.00	13,492.00	1.00	0.00	0.00	1.00
25%	28.00	116,736.00	9.00	0.00	0.00	40.00
50%	37.00	177,831.00	10.00	0.00	0.00	40.00
75%	48.00	238,384.00	12.00	0.00	0.00	45.00
max	90.00	1,490,400.00	16.00	99,999.00	3,770.00	99.00

String Columns									
idx	workclass	education	marital-status	occupation	relationship	race	sex	native-country	Target
nulls	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
count	15318	16281	16281	15315	16281	16281	16281	16007	16281
unique	8	16	7	14	6	5	2	40	2
top	PRIVATE	HS-GRAD	MARRIED-CIV-SPOUSE	PROF-SPECIALTY	HUSBAND	WHITE	MALE	UNITED-STATES	<=50K
freq	11210	5283	7403	2032	6523	13946	10860	14662	12435

Data Cleansing steps were:

1. Encoding the labels of target variable (>50K and <=50K). To achieve that, it was necessary to normalize the strings in all columns, including the target, to get rid of punctuations and any special character.

2. There are duplicates inside both data sets, however as it belongs to US Census, it is possible, even with a small odd, to be two or more individuals with the same features. Those duplicates have been left in data sets.

3. Null values (or "?" as in original data set) have been found in columns:
   a. *workclass* (1836 entries for Train data set and 963 entries for Test data set)
   b. *occupation* (1843 entries for Train data set and 966 entries for Test data set)
   c. *native-country* (583 entries for Train data set and 274 entries for Test data set)

   These nulls have been deleted from both data sets to decrease the cardinality and help to improve performance. Since they only represent 5% of the entire dataset, it has not missed much information with this deletion.

4. Some aggregations have been made in columns *workclass*, *education* and *marital-status*. These aggregations had the goal to reduce the cardinality, grouping minorities and similar classes present in each column.

5. To remove collinearity existent in features (it is possible to see that also in Correlations/Association matrix), it was chosen to remove the columns *fnlwgt*, *education-num*, *occupation* and *relationship* over to *education*, *workclass* and *marital-status*.

6. The target variable is imbalanced (approximately 24% of train and test set belongs to “>50K” class). This imbalance is not extreme, so it hasn’t been necessary any complex treatment to deal with it. Although some metrics like precision, accuracy and recall could be distorted.

7. For preprocess, has been applied two types since one of the models is Naïve Bayes, therefore using the same steps from the other models will consequently increase the error in Naïve Bayes due to his assumption of independence between features, so:
   a. Naïve Bayes:
      i. Dicretize continuous variables, such as *age* and *hours-per-week* into 6 and 7 size variable bins, respectively, making the categorical then applying and Order Encoding of ascending type.

      ii. For *education*, which has an implicitly order underlying, It has been made the same type of Order Enconding, applying low values to more basic degrees, and higher values for advanced degrees.

      iii. For the rest of features, which doesn’t hold an ordering, it was applied a standard Target enconding, which chooses the value of the category as the mean of target feature for each category.

   b. Boosting (LightGBM) and Logistic Regression:
      i. First have been applied One-Hot encoder algorithm to encode each class of each column in a new column it 1 or 0, indicating if each entry belongs to the class or not;

      ii. Then the *MinMax scaler* was applied to numerical features, preventing any problem of different scales.

8. Also, the columns *capital-gain and capital-loss* were removed, since they are mostly composed of zeros.

After these preprocessing steps, the train set have been modeled in three different algorithms (CategoricalNB, LogisticRegression and LightGBM) using cross-validation, with 3 folds, to tune the hyperparameters and then tested on test set.

After fitting, tunning and verifying each predictor, it was found the best threshold in Receiver Operating Curve, then the metrics were also measured with that threshold up to date.

Those algorithms were implemented in Python 3.7, in Google Colab platform, using python libraries as pandas, numpy, and scikit learn. The results are shown below:

	Categorical Naïve Bayes			Logistic Regression			Boost - LightGBM		
idx	Train	Teste (threshold=0.5)	Test (best threshold)	Train	Teste (threshold=0.5)	Test (best threshold)	Train	Teste (threshold=0.5)	Test (best threshold)
ROC	0,66	0,8	0,8	0,87	0,87	0,87	0,89	0,88	0,88
accuracy	0,79	0,79	0,77	0,82	0,82	0,78	0,83	0,82	0,78
Precision	0,62	0,62	0,53	0,68	0,67	0,53	0,7	0,68	0,53
Recall	0,41	0,39	0,56	0,5	0,49	0,83	0,55	0,53	0,85

If necessary, the code for 1° and 2° approach is in:

https://colab.research.google.com/drive/1QRv8eSxl5iIBdGRzdpZRl3rur87Bgi5L?usp=sharing