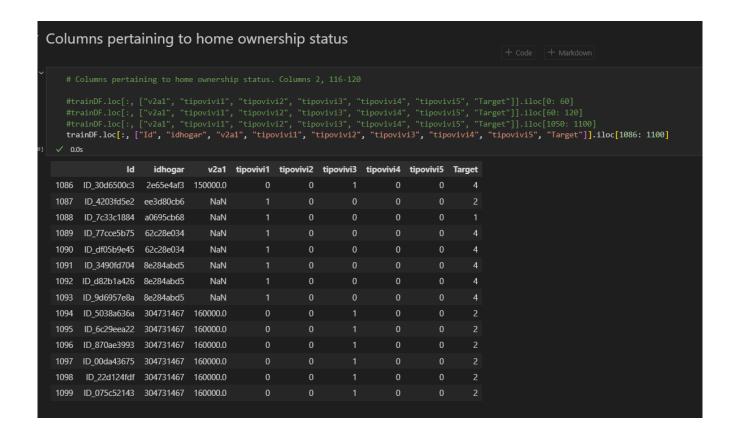
Income Qualification Screenshots

Here are some screenshots from my source code. Much of this can be found in the PDF file of the source code, but the outputs are still listed here.



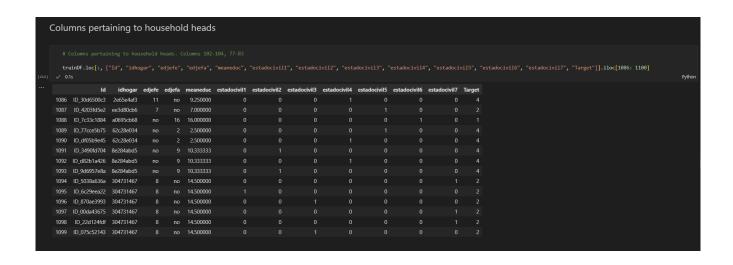
Columns pertaining to number of rooms and overcrowding #trainDF.loc[:, ["hacdor", "rooms", "hacapo", "bedrooms", "overcrowding", "Target"]].iloc[60: 120] #trainDF.loc[:, ["hacdor", "rooms", "hacapo", "bedrooms", "overcrowding", "Target"]].iloc[1050: 1100] trainDF.loc[:, ["Id", "idhogar", "hacdor", "rooms", "hacapo", "bedrooms", "overcrowding", "Target"]].iloc[1086: 1100] ld idhogar hacdor rooms hacapo bedrooms overcrowding Target 1086 ID 30d6500c3 2e65e4af3 2.500000 4 4 1087 ID 4203fd5e2 ee3d80cb6 1.000000 1088 ID_7c33c1884 a0695cb68 0.333333 ID_77cce5b75 62c28e034 1.000000 1090 ID_df05b9e45 62c28e034 1.000000 ID_3490fd704 8e284abd5 2.500000 1092 ID_d82b1a426 8e284abd5 2.500000 1093 ID 9d6957e8a 8e284abd5 2.500000 1094 ID_5038a636a 304731467 2.000000 1095 ID_6c29eea22 304731467 2.000000 1096 ID 870ae3993 304731467 2.000000 1097 ID_00da43675 304731467 2.000000 ID_22d124fdf 304731467 2.000000 1099 ID_075c52143 304731467 2.000000

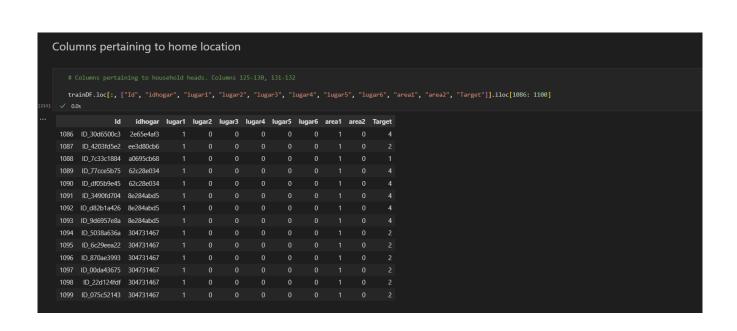


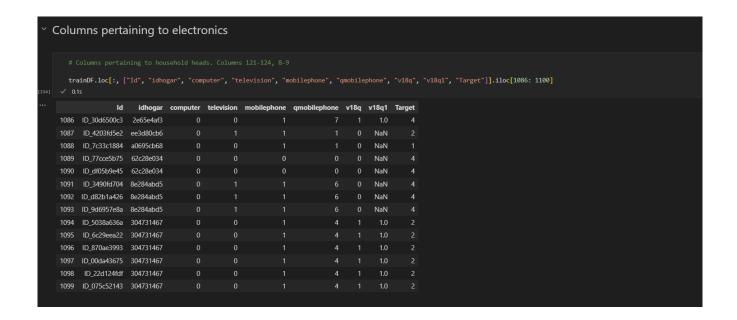
						67, 68-70,												
																		"eviv3", "Target"]].iloc[0: 60] "eviv3", "Target"]].iloc[1050: 1100]
																		viv1", "eviv2", "eviv3", "Target"]].iloc[1086: 110
	ld	idhogar	techozinc	techoentrepiso	techocane	techootro	cielorazo	epared1	epared2	epared3	etecho1	etecho2	etecho3	eviv1	eviv2	eviv3	Target	
1086	ID_30d6500c3	2e65e4af3		0														
1087	ID_4203fd5e2	ee3d80cb6																
1088	ID_7c33c1884	a0695cb68																
1089	ID_77cce5b75	62c28e034																
1090	ID_df05b9e45	62c28e034																
1091	ID_3490fd704	8e284abd5														0		
1092	ID_d82b1a426	8e284abd5		0												0		
1093	ID_9d6957e8a	8e284abd5														0		
1094	ID_5038a636a	304731467		0												0		
1095	ID_6c29eea22	304731467														0		
1096	ID_870ae3993	304731467		0												0		
1097	ID_00da43675	304731467														0		
1098	ID_22d124fdf	304731467		0												0		
1099	ID_075c52143	304731467														0		

													mbasu5", "elimbasu6", "Target"]].iloc[1086: 1100]
		146		 		-11				-Under out	-lihc	T	
1086	ID 30d6500c3		energcocinar i	energcocinar3		eiimbasu	enmbasuz (elimbasu4	elimbasub ()	elimbasuo 0		
1087	ID_4203fd5e2		0	0		1							
1088	ID_7c33c1884		0		0					0			
1089	ID_77cce5b75				0	1							
1090	ID_df05b9e45	62c28e034				1							
1091	ID_3490fd704	8e284abd5											
1092	ID_d82b1a426	8e284abd5				1							
1093	ID_9d6957e8a	8e284abd5											
1094	ID_5038a636a	304731467											
1095	ID_6c29eea22	304731467											
1096	ID_870ae3993	304731467				1							
1097	ID_00da43675	304731467											
1098	ID_22d124fdf	304731467				1							
1099	ID_075c52143	304731467) 0					









```
# We must fill all the NaN values in the dataset. First, determine which columns have NaN values.

for column in X.columns:
    if(X[column].isna().any() == True):
        print(column)

v 0.1s

v2a1
v18q1
rez_esc
meaneduc
SQBmeaned
```