PRUEBA DE EVALUACIÓN: MICROSOFT MALWARE PREDICTION

Autor: Carlos Cabañó Muñoz

Preparación I: Importación de librerías ¶

In [1]:

```
#Librerías básicas
import numpy as np      #algebra lineal
import pandas as pd     #data processing
pd.set_option("display.max_rows", 200)
pd.options.display.float_format = '{:,.2f}'.format   #cambio del formato de decimales

#Librerías para plotting
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
plt.style.use("ggplot")

#Librerías para modelización
from sklearn import model_selection
from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.tree import export_graphviz
import graphviz
import xgboost as xgb
from xgboost import XGBClassifier
from sklearn import metrics
from sklearn.ensemble import RandomForestClassifier

# Otros
RANDOM_STATE=42
```

Preparación II: Importación de datos

In [2]:

```
data_dir='https://www.dropbox.com/s/sxl5bpi2620p496/sample_mmp.csv'
```

In [3]:

```
df=pd.read_csv("sample_mmp.csv", sep=",")
```

Machine learning checklist 1: Business Understanding

Dado un dataframe, predecir la probabilidad de que una máquina con Sistema Operativo Windows se vea infectada por algún tipo de malware con base en las características de cada máquina. Cada fila es una máquina única con el identificador MachineIdentifier. El target es la columna HasDetections.

Machine Learning Checklist 2: Data Understanding

MLC 2.1_Análisis univariable:

In [4]:

```
# Analizamos cada variable con la función info(). Vemos que tiene un total de 500 000 r
egistros y 83 columnas, tanto numéricas como categóricas.
df.info(verbose=False)
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 500000 entries, 8427007 to 4295573
Columns: 83 entries, MachineIdentifier to HasDetections
dtypes: float64(36), int64(17), object(30)
memory usage: 320.4+ MB
```

MLC 2.2_Visualización directa de los datos

In [5]:

```
# Primero transponemos los datos para facilitar la visualización de las columnas:

df.head().T
```

Out[5]:

		8427007
MachineIdentifier	f1cd864e97bae82bdf96523e1a539121	fd5ba
ProductName	win8defender	
EngineVersion	1.1.15100.1	
AppVersion	4.18.1807.18075	
AvSigVersion	1.273.1234.0	
IsBeta	0	
RtpStateBitfield	7.00	
IsSxsPassiveMode	0	
DefaultBrowsersIdentifier	NaN	
AVProductStatesIdentifier	53,447.00	
AVProductsInstalled	1.00	
AVProductsEnabled	1.00	
HasTpm	1	
CountryIdentifier	8	
CityIdentifier	85,219.00	
OrganizationIdentifier	NaN	
GeoNameIdentifier	205.00	
LocaleEnglishNameIdentifier	172	
Platform	windows10	
Processor	x64	
OsVer	10.0.0.0	
OsBuild	17134	
OsSuite	256	
OsPlatformSubRelease	rs4	
OsBuildLab	17134.1.amd64fre.rs4_release.180410-1804	17134.
SkuEdition	Pro	
IsProtected	1.00	
AutoSampleOptIn	0	
PuaMode	NaN	
SMode	0.00	
IeVerIdentifier	137.00	
SmartScreen	RequireAdmin	
Firewall	1.00	
UacLuaenable	1.00	
Census_MDC2FormFactor	Desktop	
Census_DeviceFamily	Windows.Desktop	

	8427007
Census_OEMNameIdentifier	1,443.00
Census_OEMModelIdentifier	275,891.00
Census_ProcessorCoreCount	4.00
Census_ProcessorManufacturerIdentifier	5.00
Census_ProcessorModelIdentifier	2,273.00
Census_ProcessorClass	NaN
Census_PrimaryDiskTotalCapacity	953,869.00
Census_PrimaryDiskTypeName	HDD
Census_SystemVolumeTotalCapacity	952,838.00
Census_HasOpticalDiskDrive	0
Census_TotalPhysicalRAM	8,192.00
Census_ChassisTypeName	AllinOne
Census_InternalPrimaryDiagonalDisplaySizeInInches	23.00
Census_InternalPrimaryDisplayResolutionHorizontal	1,920.00
Census_InternalPrimaryDisplayResolutionVertical	1,080.00
Census_PowerPlatformRoleName	Desktop
Census_InternalBatteryType	NaN
Census_InternalBatteryNumberOfCharges	4,294,967,295.00
Census_OSVersion	10.0.17134.165
Census_OSArchitecture	amd64
Census_OSBranch	rs4_release
Census_OSBuildNumber	17134
Census_OSBuildRevision	165
Census_OSEdition	Professional
Census_OSSkuName	PROFESSIONAL
Census_OSInstallTypeName	UUPUpgrade
Census_OSInstallLanguageIdentifier	27.00
Census_OSUILocaleIdentifier	120
Census_OSWUAutoUpdateOptionsName	FullAuto
Census_IsPortableOperatingSystem	0
Census_GenuineStateName	IS_GENUINE
Census_ActivationChannel	OEM:DM
Census_IsFlightingInternal	NaN
Census_IsFlightsDisabled	0.00
Census_FlightRing	Retail
Census_ThresholdOptIn	NaN
Census_FirmwareManufacturerIdentifier	355.00
Census_FirmwareVersionIdentifier	19,951.00
Census_IsSecureBootEnabled	0

	8427007
Census_IsWIMBootEnabled	NaN
Census_IsVirtualDevice	0.00
Census_IsTouchEnabled	0
Census_IsPenCapable	0
Census_IsAlwaysOnAlwaysConnectedCapable	0.00
Wdft_IsGamer	0.00
Wdft_RegionIdentifier	11.00
HasDetections	1

MLC 2.3_Atributos disponibles

In [6]:

```
# Esta vez, incluimos el parámetro verbose=True para ver todos los atributos del DF.
df.info(verbose=True)
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 500000 entries, 8427007 to 4295573
Data columns (total 83 columns):
 #   Column                                        Non-Null Count   D
type
---  ------                                        --------------   -
----
 0   MachineIdentifier                             500000 non-null  o
bject
 1   ProductName                                   500000 non-null  o
bject
 2   EngineVersion                                 500000 non-null  o
bject
 3   AppVersion                                    500000 non-null  o
bject
 4   AvSigVersion                                  500000 non-null  o
bject
 5   IsBeta                                        500000 non-null  i
nt64
 6   RtpStateBitfield                              498168 non-null  f
loat64
 7   IsSxsPassiveMode                              500000 non-null  i
nt64
 8   DefaultBrowsersIdentifier                     24061 non-null   f
loat64
 9   AVProductStatesIdentifier                     498062 non-null  f
loat64
 10  AVProductsInstalled                           498062 non-null  f
loat64
 11  AVProductsEnabled                             498062 non-null  f
loat64
 12  HasTpm                                        500000 non-null  i
nt64
 13  CountryIdentifier                             500000 non-null  i
nt64
 14  CityIdentifier                                481760 non-null  f
loat64
 15  OrganizationIdentifier                        345437 non-null  f
loat64
 16  GeoNameIdentifier                             499984 non-null  f
loat64
 17  LocaleEnglishNameIdentifier                   500000 non-null  i
nt64
 18  Platform                                      500000 non-null  o
bject
 19  Processor                                     500000 non-null  o
bject
 20  OsVer                                         500000 non-null  o
bject
 21  OsBuild                                       500000 non-null  i
nt64
 22  OsSuite                                       500000 non-null  i
nt64
 23  OsPlatformSubRelease                          500000 non-null  o
bject
 24  OsBuildLab                                    499999 non-null  o
bject
 25  SkuEdition                                    500000 non-null  o
bject
 26  IsProtected                                   498074 non-null  f
loat64
```

```
 27  AutoSampleOptIn                                   500000 non-null  i
nt64
 28  PuaMode                                           126 non-null     o
bject
 29  SMode                                             470152 non-null  f
loat64
 30  IeVerIdentifier                                   496791 non-null  f
loat64
 31  SmartScreen                                       321404 non-null  o
bject
 32  Firewall                                          494838 non-null  f
loat64
 33  UacLuaenable                                      499377 non-null  f
loat64
 34  Census_MDC2FormFactor                             500000 non-null  o
bject
 35  Census_DeviceFamily                               500000 non-null  o
bject
 36  Census_OEMNameIdentifier                          494619 non-null  f
loat64
 37  Census_OEMModelIdentifier                         494236 non-null  f
loat64
 38  Census_ProcessorCoreCount                         497653 non-null  f
loat64
 39  Census_ProcessorManufacturerIdentifier            497653 non-null  f
loat64
 40  Census_ProcessorModelIdentifier                   497651 non-null  f
loat64
 41  Census_ProcessorClass                             2082 non-null    o
bject
 42  Census_PrimaryDiskTotalCapacity                  497024 non-null  f
loat64
 43  Census_PrimaryDiskTypeName                        499291 non-null  o
bject
 44  Census_SystemVolumeTotalCapacity                  497024 non-null  f
loat64
 45  Census_HasOpticalDiskDrive                        500000 non-null  i
nt64
 46  Census_TotalPhysicalRAM                           495444 non-null  f
loat64
 47  Census_ChassisTypeName                            499963 non-null  o
bject
 48  Census_InternalPrimaryDiagonalDisplaySizeInInches  497346 non-null  f
loat64
 49  Census_InternalPrimaryDisplayResolutionHorizontal  497350 non-null  f
loat64
 50  Census_InternalPrimaryDisplayResolutionVertical    497350 non-null  f
loat64
 51  Census_PowerPlatformRoleName                      499998 non-null  o
bject
 52  Census_InternalBatteryType                        144397 non-null  o
bject
 53  Census_InternalBatteryNumberOfCharges             484962 non-null  f
loat64
 54  Census_OSVersion                                  500000 non-null  o
bject
 55  Census_OSArchitecture                             500000 non-null  o
bject
 56  Census_OSBranch                                   500000 non-null  o
bject
 57  Census_OSBuildNumber                              500000 non-null  i
```

```
nt64
 58  Census_OSBuildRevision                             500000 non-null  i
nt64
 59  Census_OSEdition                                   500000 non-null  o
bject
 60  Census_OSSkuName                                   500000 non-null  o
bject
 61  Census_OSInstallTypeName                           500000 non-null  o
bject
 62  Census_OSInstallLanguageIdentifier                 496668 non-null  f
loat64
 63  Census_OSUILocaleIdentifier                        500000 non-null  i
nt64
 64  Census_OSWUAutoUpdateOptionsName                   500000 non-null  o
bject
 65  Census_IsPortableOperatingSystem                   500000 non-null  i
nt64
 66  Census_GenuineStateName                            500000 non-null  o
bject
 67  Census_ActivationChannel                           500000 non-null  o
bject
 68  Census_IsFlightingInternal                         84775 non-null   f
loat64
 69  Census_IsFlightsDisabled                           491067 non-null  f
loat64
 70  Census_FlightRing                                  500000 non-null  o
bject
 71  Census_ThresholdOptIn                              181896 non-null  f
loat64
 72  Census_FirmwareManufacturerIdentifier              489651 non-null  f
loat64
 73  Census_FirmwareVersionIdentifier                   490939 non-null  f
loat64
 74  Census_IsSecureBootEnabled                         500000 non-null  i
nt64
 75  Census_IsWIMBootEnabled                            182334 non-null  f
loat64
 76  Census_IsVirtualDevice                             499099 non-null  f
loat64
 77  Census_IsTouchEnabled                              500000 non-null  i
nt64
 78  Census_IsPenCapable                                500000 non-null  i
nt64
 79  Census_IsAlwaysOnAlwaysConnectedCapable            495960 non-null  f
loat64
 80  Wdft_IsGamer                                       483050 non-null  f
loat64
 81  Wdft_RegionIdentifier                              483050 non-null  f
loat64
 82  HasDetections                                      500000 non-null  i
nt64
dtypes: float64(36), int64(17), object(30)
memory usage: 320.4+ MB
```

En este dataframe el campo Machine Identifier es un campo con valores únicos que podría servir como índice. Además, a simple vista podemos ver algunos campos que podrían ser claves como Platform, Product Namem o Firewall. Vamos a verlos:

MLC 2.4_Estadísticos descriptivos

In [7]:

```
# Vemos en primer lugar las métricas de los campos categóricos:

df.describe(include="object").T
```

Out[7]:

	count	unique	top
MachineIdentifier	500000	500000	76394f7b5c1764c1ca4552f51dbeb73f
ProductName	500000	3	win8defender
EngineVersion	500000	53	1.1.15200.1
AppVersion	500000	95	4.18.1807.18075
AvSigVersion	500000	6455	1.273.1420.0
Platform	500000	4	windows10
Processor	500000	3	x64
OsVer	500000	21	10.0.0.0
OsPlatformSubRelease	500000	9	rs4
OsBuildLab	499999	453	17134.1.amd64fre.rs4_release.180410-1804
SkuEdition	500000	8	Home
PuaMode	126	1	on
SmartScreen	321404	12	RequireAdmin
Census_MDC2FormFactor	500000	12	Notebook
Census_DeviceFamily	500000	3	Windows.Desktop
Census_ProcessorClass	2082	3	mid
Census_PrimaryDiskTypeName	499291	4	HDD
Census_ChassisTypeName	499963	34	Notebook
Census_PowerPlatformRoleName	499998	9	Mobile
Census_InternalBatteryType	144397	28	lion
Census_OSVersion	500000	305	10.0.17134.228
Census_OSArchitecture	500000	3	amd64
Census_OSBranch	500000	15	rs4_release
Census_OSEdition	500000	22	Core
Census_OSSkuName	500000	21	CORE
Census_OSInstallTypeName	500000	9	UUPUpgrade
Census_OSWUAutoUpdateOptionsName	500000	6	FullAuto
Census_GenuineStateName	500000	4	IS_GENUINE
Census_ActivationChannel	500000	6	Retail
Census_FlightRing	500000	8	Retail

Lo que nos da como resultado varios campos con nulos que habrá que rellenar y algunos campos con valores categóricos pero con pocos únicos, que podríamos codificar con un One Hot Encoding. En cambio, aquellos campos que tienen un gran número de únicos, veremos posteriormente si podemos agruparlos por frecuencia y codificarlos.

In [8]:

```
# A continuación, las métricas de los campos numéricos:

df.describe(include=np.number).T
```

Out[8]:

	count	mean	s
IsBeta	500,000.00	0.00	0.
RtpStateBitfield	498,168.00	6.85	1.
IsSxsPassiveMode	500,000.00	0.02	0.
DefaultBrowsersIdentifier	24,061.00	1,652.82	1,004.
AVProductStatesIdentifier	498,062.00	47,850.91	14,023.
AVProductsInstalled	498,062.00	1.33	0.
AVProductsEnabled	498,062.00	1.02	0.
HasTpm	500,000.00	0.99	0.
CountryIdentifier	500,000.00	108.04	63.
CityIdentifier	481,760.00	81,271.65	48,985.
OrganizationIdentifier	345,437.00	24.87	5.
GeoNameIdentifier	499,984.00	169.73	89.
LocaleEnglishNameIdentifier	500,000.00	122.61	69.
OsBuild	500,000.00	15,726.93	2,188.
OsSuite	500,000.00	574.72	248.
IsProtected	498,074.00	0.95	0.
AutoSampleOptIn	500,000.00	0.00	0.
SMode	470,152.00	0.00	0.
IeVerIdentifier	496,791.00	126.66	42.
Firewall	494,838.00	0.98	0.
UacLuaenable	499,377.00	13.73	8,995.
Census_OEMNameIdentifier	494,619.00	2,218.65	1,315.
Census_OEMModelIdentifier	494,236.00	239,128.05	72,048.
Census_ProcessorCoreCount	497,653.00	3.99	2.
Census_ProcessorManufacturerIdentifier	497,653.00	4.53	1.
Census_ProcessorModelIdentifier	497,651.00	2,370.99	842.
Census_PrimaryDiskTotalCapacity	497,024.00	514,043.32	370,446.
Census_SystemVolumeTotalCapacity	497,024.00	378,054.64	338,472.
Census_HasOpticalDiskDrive	500,000.00	0.08	0.
Census_TotalPhysicalRAM	495,444.00	6,129.23	4,964.
Census_InternalPrimaryDiagonalDisplaySizeInInches	497,346.00	16.69	5.
Census_InternalPrimaryDisplayResolutionHorizontal	497,350.00	1,548.30	368.
Census_InternalPrimaryDisplayResolutionVertical	497,350.00	898.24	214.
Census_InternalBatteryNumberOfCharges	484,962.00	1,125,600,150.21	1,888,768,455.
Census_OSBuildNumber	500,000.00	15,841.37	1,959.
Census_OSBuildRevision	500,000.00	967.22	2,920.
Census_OSInstallLanguageIdentifier	496,668.00	14.61	10.

	count	mean	s
Census_OSUILocaleIdentifier	500,000.00	60.45	45.
Census_IsPortableOperatingSystem	500,000.00	0.00	0.
Census_IsFlightingInternal	84,775.00	0.00	0.
Census_IsFlightsDisabled	491,067.00	0.00	0.
Census_ThresholdOptIn	181,896.00	0.00	0.
Census_FirmwareManufacturerIdentifier	489,651.00	402.68	221.
Census_FirmwareVersionIdentifier	490,939.00	33,030.99	21,220.
Census_IsSecureBootEnabled	500,000.00	0.49	0.
Census_IsWIMBootEnabled	182,334.00	0.00	0.
Census_IsVirtualDevice	499,099.00	0.01	0.
Census_IsTouchEnabled	500,000.00	0.13	0.
Census_IsPenCapable	500,000.00	0.04	0.
Census_IsAlwaysOnAlwaysConnectedCapable	495,960.00	0.06	0.
Wdft_IsGamer	483,050.00	0.28	0.
Wdft_RegionIdentifier	483,050.00	7.89	4.
HasDetections	500,000.00	0.50	0.

Podemos ver que la media del Target (campo HasDetections) es justo el 50%, es decir, la mitad de los valores será 0 (False) y la otra 1 (True), lo que tampoco nos indica mucho en una futura predicción.

MLC 2.5_Número de nulos

In [9]:

```
# Como el tratamiento de nulos va a ser diferente según el tipo de dato, en primer luga
r observamos los nulos de las variables categóricas:

df.select_dtypes(include="object").isnull().sum()
```

Out[9]:

```
MachineIdentifier                     0
ProductName                           0
EngineVersion                         0
AppVersion                            0
AvSigVersion                          0
Platform                              0
Processor                             0
OsVer                                 0
OsPlatformSubRelease                  0
OsBuildLab                            1
SkuEdition                            0
PuaMode                          499874
SmartScreen                      178596
Census_MDC2FormFactor                 0
Census_DeviceFamily                   0
Census_ProcessorClass            497918
Census_PrimaryDiskTypeName          709
Census_ChassisTypeName               37
Census_PowerPlatformRoleName          2
Census_InternalBatteryType       355603
Census_OSVersion                      0
Census_OSArchitecture                 0
Census_OSBranch                       0
Census_OSEdition                      0
Census_OSSkuName                      0
Census_OSInstallTypeName              0
Census_OSWUAutoUpdateOptionsName      0
Census_GenuineStateName               0
Census_ActivationChannel              0
Census_FlightRing                     0
dtype: int64
```

In [10]:

```
# Y a continuación, los de las variables numéricas:

df.select_dtypes(include=np.number).isnull().sum()
```

Out[10]:

```
IsBeta                                                 0
RtpStateBitfield                                    1832
IsSxsPassiveMode                                       0
DefaultBrowsersIdentifier                         475939
AVProductStatesIdentifier                           1938
AVProductsInstalled                                 1938
AVProductsEnabled                                   1938
HasTpm                                                 0
CountryIdentifier                                      0
CityIdentifier                                     18240
OrganizationIdentifier                            154563
GeoNameIdentifier                                     16
LocaleEnglishNameIdentifier                            0
OsBuild                                                0
OsSuite                                                0
IsProtected                                         1926
AutoSampleOptIn                                        0
SMode                                              29848
IeVerIdentifier                                     3209
Firewall                                            5162
UacLuaenable                                         623
Census_OEMNameIdentifier                            5381
Census_OEMModelIdentifier                           5764
Census_ProcessorCoreCount                           2347
Census_ProcessorManufacturerIdentifier              2347
Census_ProcessorModelIdentifier                     2349
Census_PrimaryDiskTotalCapacity                     2976
Census_SystemVolumeTotalCapacity                    2976
Census_HasOpticalDiskDrive                             0
Census_TotalPhysicalRAM                             4556
Census_InternalPrimaryDiagonalDisplaySizeInInches   2654
Census_InternalPrimaryDisplayResolutionHorizontal   2650
Census_InternalPrimaryDisplayResolutionVertical     2650
Census_InternalBatteryNumberOfCharges              15038
Census_OSBuildNumber                                   0
Census_OSBuildRevision                                 0
Census_OSInstallLanguageIdentifier                  3332
Census_OSUILocaleIdentifier                            0
Census_IsPortableOperatingSystem                       0
Census_IsFlightingInternal                        415225
Census_IsFlightsDisabled                            8933
Census_ThresholdOptIn                             318104
Census_FirmwareManufacturerIdentifier              10349
Census_FirmwareVersionIdentifier                    9061
Census_IsSecureBootEnabled                             0
Census_IsWIMBootEnabled                           317666
Census_IsVirtualDevice                               901
Census_IsTouchEnabled                                  0
Census_IsPenCapable                                    0
Census_IsAlwaysOnAlwaysConnectedCapable             4040
Wdft_IsGamer                                       16950
Wdft_RegionIdentifier                              16950
HasDetections                                          0
dtype: int64
```

A simple vista, podemos observar que campos que podrían ser claves como platform o product name no tienen nulos. Tampoco el Target.

MLC 2.6_Valores del target

In [11]:

```
# Analizamos los valores del target, en este caso el campo HasDetections

df["HasDetections"].value_counts()
```

Out[11]:

```
0    250047
1    249953
Name: HasDetections, dtype: int64
```

In [12]:

```
# El porcentaje de False, es decir, de No en el total es:

250047/(250047+249953)*100    # En este caso es el 22% del total
```

Out[12]:

```
50.00940000000001
```

El número de valores está muy igualado, tal y como las métricas ya nos indicaron.

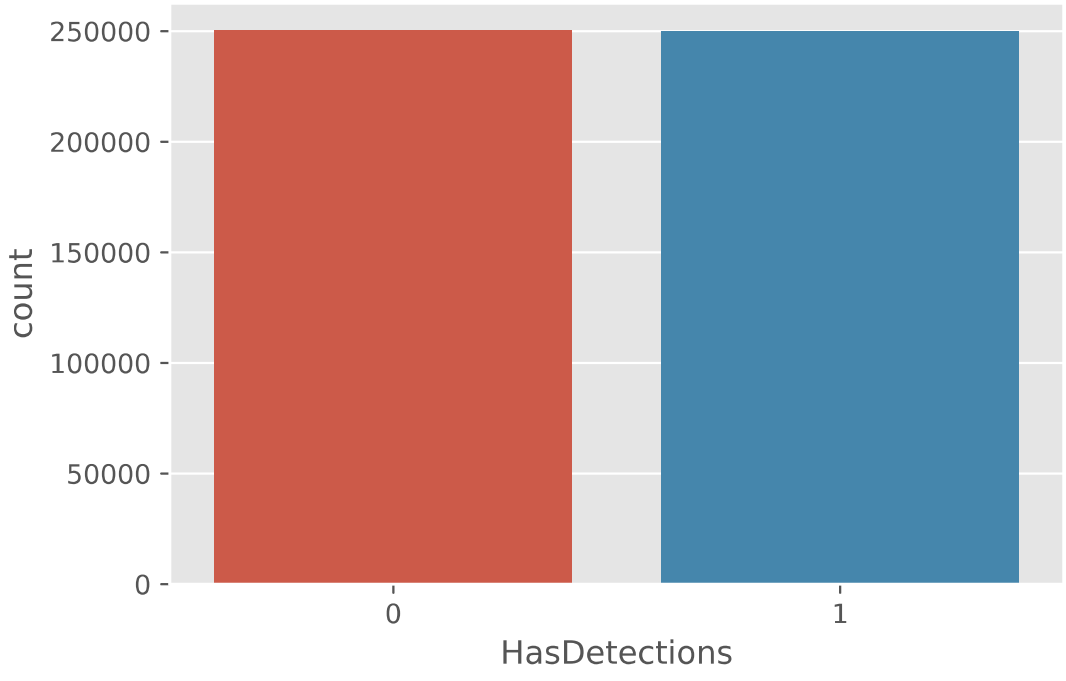
In [13]:

```
# Podemos visualizarlo con un countplot:

sns.countplot(data=df, x="HasDetections")
```

Out[13]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ea9475f7f0>
```



In [14]:

```
# Comprobamos el tipo de dato del campo target:

df["HasDetections"].dtype
```

Out[14]:

```
dtype('int64')
```

Como es un campo numérico, no hará falta que lo modifiquemos. No obstante, aunque sea un integer, en realidad lo que indica es una condición booleana: True (1) o False (0), es decir, que se detectó virus o no.

In [15]:

```
# Denominamos la variable Target para facilitar la escritura de código:

df["Target"]=df["HasDetections"]
```

In [16]:

```
# Eliminamos uno de los dos campos:

df.drop("HasDetections", axis=1, inplace=True)
```

In [17]:

```
df.head().T
```

Out[17]:

		8427007
MachineIdentifier	f1cd864e97bae82bdf96523e1a539121	fd5ba
ProductName	win8defender	
EngineVersion	1.1.15100.1	
AppVersion	4.18.1807.18075	
AvSigVersion	1.273.1234.0	
IsBeta	0	
RtpStateBitfield	7.00	
IsSxsPassiveMode	0	
DefaultBrowsersIdentifier	NaN	
AVProductStatesIdentifier	53,447.00	
AVProductsInstalled	1.00	
AVProductsEnabled	1.00	
HasTpm	1	
CountryIdentifier	8	
CityIdentifier	85,219.00	
OrganizationIdentifier	NaN	
GeoNameIdentifier	205.00	
LocaleEnglishNameIdentifier	172	
Platform	windows10	
Processor	x64	
OsVer	10.0.0.0	
OsBuild	17134	
OsSuite	256	
OsPlatformSubRelease	rs4	
OsBuildLab	17134.1.amd64fre.rs4_release.180410-1804	17134.
SkuEdition	Pro	
IsProtected	1.00	
AutoSampleOptIn	0	
PuaMode	NaN	
SMode	0.00	
IeVerIdentifier	137.00	
SmartScreen	RequireAdmin	
Firewall	1.00	
UacLuaenable	1.00	
Census_MDC2FormFactor	Desktop	
Census_DeviceFamily	Windows.Desktop	

	8427007
Census_OEMNameIdentifier	1,443.00
Census_OEMModelIdentifier	275,891.00
Census_ProcessorCoreCount	4.00
Census_ProcessorManufacturerIdentifier	5.00
Census_ProcessorModelIdentifier	2,273.00
Census_ProcessorClass	NaN
Census_PrimaryDiskTotalCapacity	953,869.00
Census_PrimaryDiskTypeName	HDD
Census_SystemVolumeTotalCapacity	952,838.00
Census_HasOpticalDiskDrive	0
Census_TotalPhysicalRAM	8,192.00
Census_ChassisTypeName	AllinOne
Census_InternalPrimaryDiagonalDisplaySizeInInches	23.00
Census_InternalPrimaryDisplayResolutionHorizontal	1,920.00
Census_InternalPrimaryDisplayResolutionVertical	1,080.00
Census_PowerPlatformRoleName	Desktop
Census_InternalBatteryType	NaN
Census_InternalBatteryNumberOfCharges	4,294,967,295.00
Census_OSVersion	10.0.17134.165
Census_OSArchitecture	amd64
Census_OSBranch	rs4_release
Census_OSBuildNumber	17134
Census_OSBuildRevision	165
Census_OSEdition	Professional
Census_OSSkuName	PROFESSIONAL
Census_OSInstallTypeName	UUPUpgrade
Census_OSInstallLanguageIdentifier	27.00
Census_OSUILocaleIdentifier	120
Census_OSWUAutoUpdateOptionsName	FullAuto
Census_IsPortableOperatingSystem	0
Census_GenuineStateName	IS_GENUINE
Census_ActivationChannel	OEM:DM
Census_IsFlightingInternal	NaN
Census_IsFlightsDisabled	0.00
Census_FlightRing	Retail
Census_ThresholdOptIn	NaN
Census_FirmwareManufacturerIdentifier	355.00
Census_FirmwareVersionIdentifier	19,951.00
Census_IsSecureBootEnabled	0

	8427007
Census_IsWIMBootEnabled	NaN
Census_IsVirtualDevice	0.00
Census_IsTouchEnabled	0
Census_IsPenCapable	0
Census_IsAlwaysOnAlwaysConnectedCapable	0.00
Wdft_IsGamer	0.00
Wdft_RegionIdentifier	11.00
Target	1

In [18]:

```
# Analizamos las métricas del target, especialmente count, mean y sum:

df["Target"].count()
```

Out[18]:

```
500000
```

In [19]:

```
df["Target"].mean()
```

Out[19]:

```
0.499906
```

In [20]:

```
df["Target"].sum()
```

Out[20]:

```
249953
```

Con el análisis de la media, vemos que, al igual que nos ha indicado el countplot, la probabilidad es del 50% aproximadamente.

VISUALIZACIONES

MLC 2.9_Correlación de valores con el target

Con el análisis de las correlaciones, podemos ver qué variables hacen más posible el target.

In [21]:

```
# Primero podemos hacer un pivot table para ver la relación de algunos campos clave con el target:

df.pivot_table(index=["Platform", "ProductName", "Firewall"], values="Target", aggfunc=[len, np.mean, np.sum])
```

Out[21]:

			len	mean	sum
			Target	Target	Target
Platform	ProductName	Firewall			
windows10	mse	1.00	14	0.29	4
	win8defender	0.00	10144	0.49	5011
		1.00	472278	0.50	236216
windows2016	win8defender	0.00	236	0.38	90
		1.00	526	0.35	182
windows7	mse	0.00	45	0.33	15
		1.00	793	0.37	293
windows8	mse	1.00	1	0.00	0
	win8defender	0.00	342	0.56	192
		1.00	10459	0.52	5411

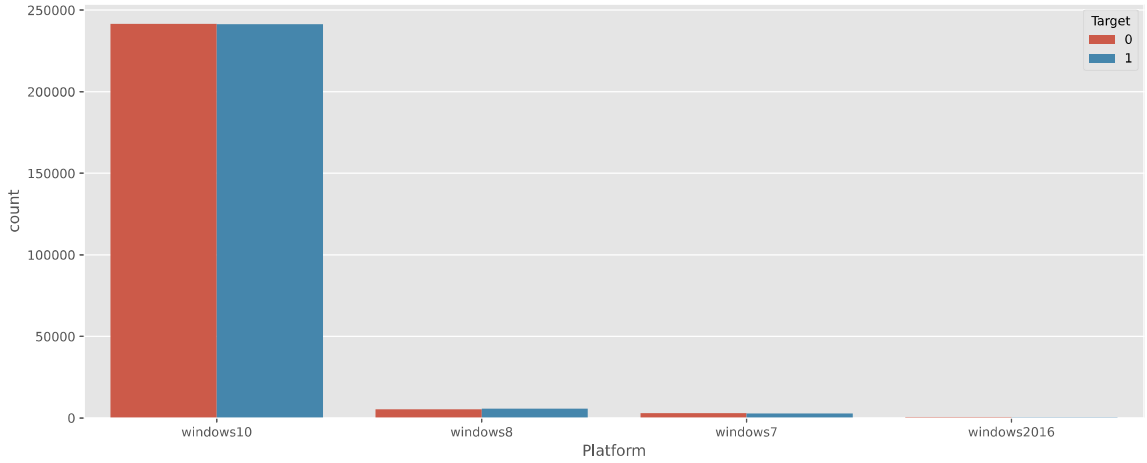
Observamos un dato interesante: el uso del antivirus MSE reduce ligeramente la probabilidad del Target, aunque esto puede deberse a la escasez de muestras. Y vemos que con win8defender, no influye mucho tener Firewall o no. Con MSE baja un poco la probabilidad de padecer un virus. Vamos a visualizar la relación de estos dos campos con el target:

In [22]:

```
plt.figure(figsize=[15,6])
sns.countplot(data=df, x="Platform", hue="Target")
```

Out[22]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ea932c3b50>
```

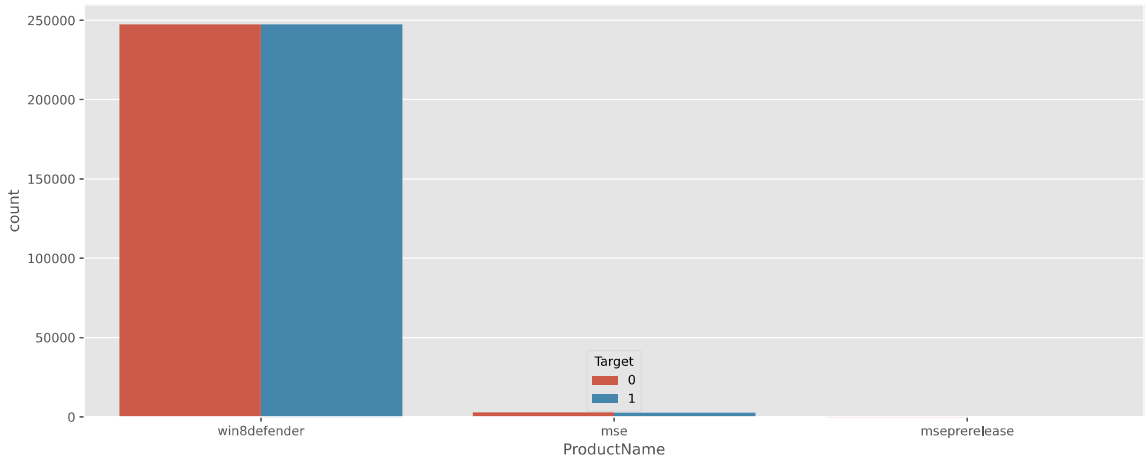


In [23]:

```
plt.figure(figsize=[15,6])
sns.countplot(data=df, x="ProductName", hue="Target")
```

Out[23]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ea9483a7c0>
```



Todos los sistemas operativos del dataframe son Windows con varias versiones y varios tipos de antivirus, aunque el win8defender es el más utilizado con diferencia. Podemos observar que con windows10, es más probable el Target con win8defender que con mse.

In [24]:

```
# Vamos a cambiar el índice por el campo MachineIdentifier. Como este campo es único pa
ra cada registro, podemos usarlo como índice:

df.set_index("MachineIdentifier", inplace=True)
```

Vamos a visualizar las variables en gráficas. Como tenemos un gran número de variables, vamos a definir dos funciones que nos faciliten el trabajo. Una para variables numéricas y la otra para categóricas:

In [25]:

```
# Utilizamos la siguiente función para las variables categóricas:

def Visualizacion (df_x, campo_x, campo_y):
    pt=df.pivot_table(index=campo_x, values=campo_y, aggfunc=[len, np.sum, np.mean]).so
rt_values(by=[campo_x], ascending = False)
    valor=df_x[campo_x].value_counts(dropna=False, normalize=True)*100
    plt.figure(figsize=[15,6])
    sns.countplot(data=df_x, x=campo_x, hue=campo_y)
    print("Resultados Pivot Table: ", campo_x, "\n", pt, "\n")
    print("Resultados Value Counts: ", campo_x, "\n", valor, "\n")
    plt.show()
    "\n"
    return df
```

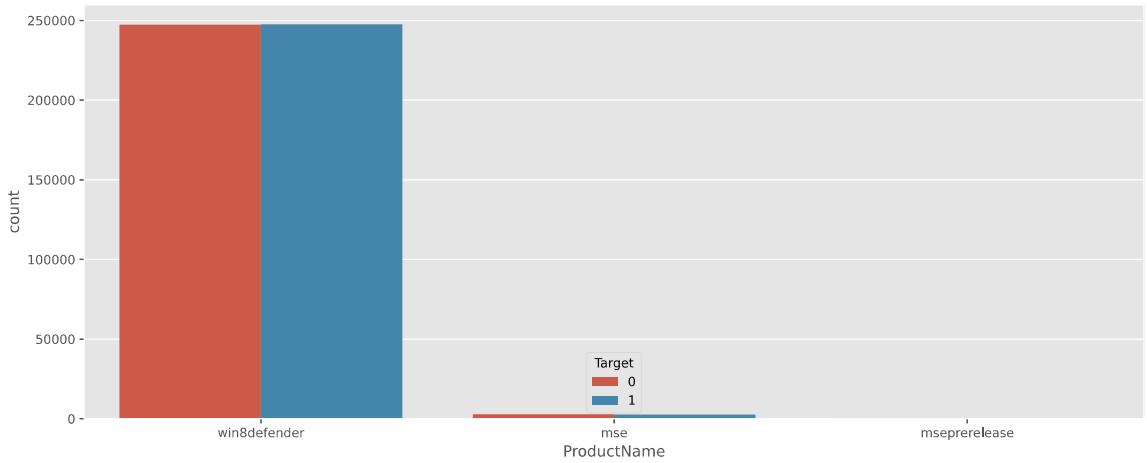
Aplicamos la función Visualizacion a todas las variables categóricas:

In [26]:

```
df=Visualizacion(df, "ProductName", "Target")
```

```
Resultados Pivot Table:  ProductName
                    len     sum   mean
                 Target  Target Target
ProductName
win8defender   494604  247367   0.50
mseprerelease       1       0   0.00
mse              5395    2586   0.48

Resultados Value Counts:  ProductName
 win8defender    98.92
mse              1.08
mseprerelease    0.00
Name: ProductName, dtype: float64
```



El 98% de los registros tiene el mismo valor, por lo que esta variable no influirá en el árbol de decisión.

In [27]:

```
df=Visualizacion(df, "EngineVersion", "Target")
```

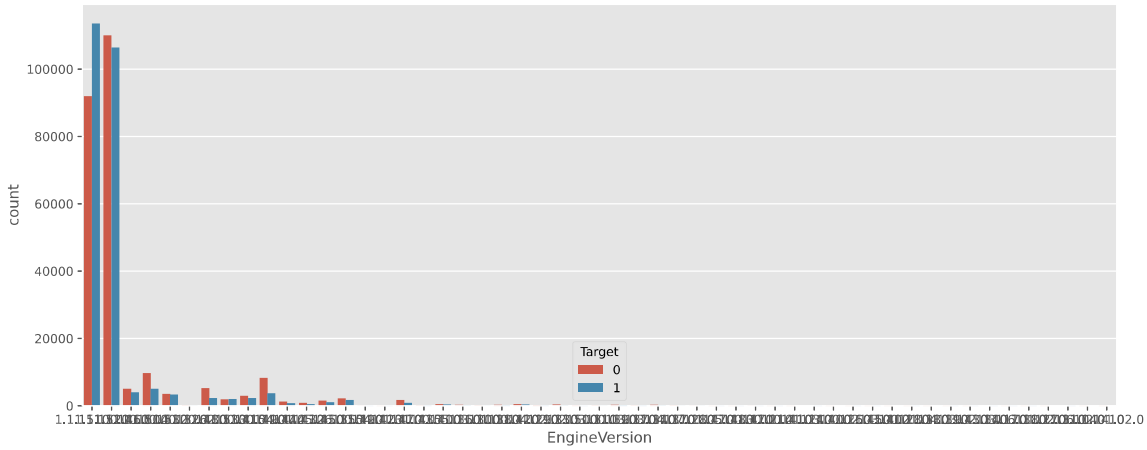
Resultados Pivot Table: EngineVersion

EngineVersion	len	sum	mean
	Target	Target	Target
1.1.15300.6	6769	3287	0.49
1.1.15300.5	3883	2025	0.52
1.1.15200.1	216491	106453	0.49
1.1.15100.1	205494	113557	0.55
1.1.15000.2	14752	5041	0.34
1.1.15000.1	138	33	0.24
1.1.14901.4	11984	3670	0.31
1.1.14901.3	71	18	0.25
1.1.14800.3	7548	2306	0.31
1.1.14800.1	51	13	0.25
1.1.14700.5	2593	855	0.33
1.1.14700.4	65	24	0.37
1.1.14700.3	61	21	0.34
1.1.14600.4	9005	3986	0.44
1.1.14500.5	2591	1032	0.40
1.1.14500.2	19	5	0.26
1.1.14405.2	1972	775	0.39
1.1.14306.0	1329	510	0.38
1.1.14305.0	270	114	0.42
1.1.14303.0	10	4	0.40
1.1.14202.0	840	357	0.42
1.1.14201.0	11	5	0.45
1.1.14104.0	5240	2298	0.44
1.1.14103.0	5	0	0.00
1.1.14102.0	2	0	0.00
1.1.14003.0	804	344	0.43
1.1.14002.0	2	0	0.00
1.1.14001.0	2	0	0.00
1.1.13903.0	515	208	0.40
1.1.13902.0	4	1	0.25
1.1.13804.0	513	194	0.38
1.1.13803.0	3	0	0.00
1.1.13802.0	4	3	0.75
1.1.13704.0	255	99	0.39
1.1.13701.0	255	86	0.34
1.1.13601.0	420	140	0.33
1.1.13504.0	3876	1706	0.44
1.1.13407.0	479	174	0.36
1.1.13406.0	1	1	1.00
1.1.13303.0	516	169	0.33
1.1.13202.0	245	85	0.35
1.1.13103.0	235	84	0.36
1.1.13102.0	1	1	1.00
1.1.13000.0	220	81	0.37
1.1.12902.0	321	128	0.40
1.1.12805.0	115	51	0.44
1.1.12804.0	4	1	0.25
1.1.12706.0	1	0	0.00
1.1.12603.0	3	1	0.33
1.1.12400.0	1	1	1.00
1.1.12101.0	7	4	0.57
1.1.11701.0	3	2	0.67
1.1.10401.0	1	0	0.00

Resultados Value Counts: EngineVersion

1.1.15200.1	43.30
1.1.15100.1	41.10

```
1.1.15000.2    2.95
1.1.14901.4    2.40
1.1.14600.4    1.80
1.1.14800.3    1.51
1.1.15300.6    1.35
1.1.14104.0    1.05
1.1.15300.5    0.78
1.1.13504.0    0.78
1.1.14700.5    0.52
1.1.14500.5    0.52
1.1.14405.2    0.39
1.1.14306.0    0.27
1.1.14202.0    0.17
1.1.14003.0    0.16
1.1.13303.0    0.10
1.1.13903.0    0.10
1.1.13804.0    0.10
1.1.13407.0    0.10
1.1.13601.0    0.08
1.1.12902.0    0.06
1.1.14305.0    0.05
1.1.13701.0    0.05
1.1.13704.0    0.05
1.1.13202.0    0.05
1.1.13103.0    0.05
1.1.13000.0    0.04
1.1.15000.1    0.03
1.1.12805.0    0.02
1.1.14901.3    0.01
1.1.14700.4    0.01
1.1.14700.3    0.01
1.1.14800.1    0.01
1.1.14500.2    0.00
1.1.14201.0    0.00
1.1.14303.0    0.00
1.1.12101.0    0.00
1.1.14103.0    0.00
1.1.12804.0    0.00
1.1.13802.0    0.00
1.1.13902.0    0.00
1.1.13803.0    0.00
1.1.12603.0    0.00
1.1.11701.0    0.00
1.1.14001.0    0.00
1.1.14002.0    0.00
1.1.14102.0    0.00
1.1.12400.0    0.00
1.1.13102.0    0.00
1.1.10401.0    0.00
1.1.12706.0    0.00
1.1.13406.0    0.00
Name: EngineVersion, dtype: float64
```



En este campo vemos que hay muchos valores únicos, pero solo dos acaparan la mayoría. Vamos a agruparlos posteriormente según los primeros dígitos.

In [28]:

```
df=Visualizacion(df, "AppVersion", "Target")
```

Resultados Pivot Table: AppVersion

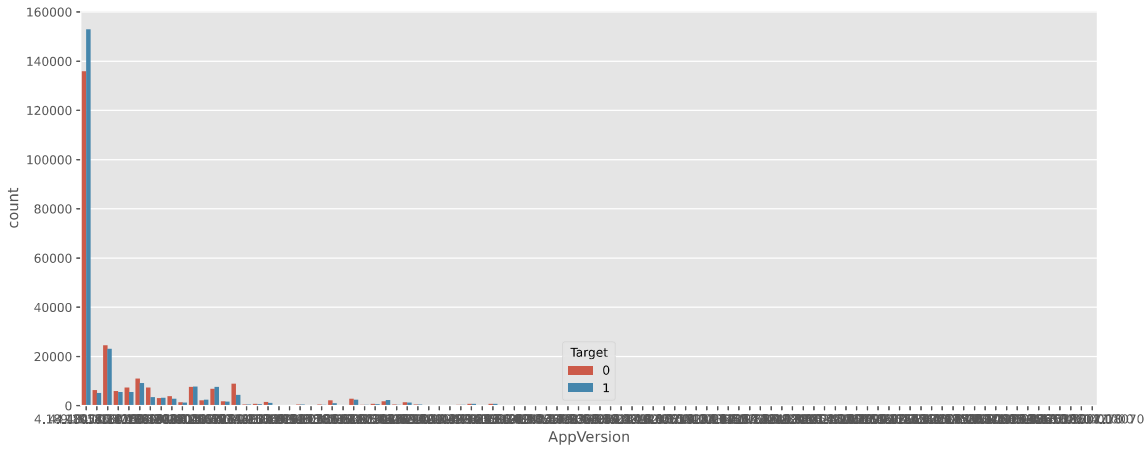
	len	sum	mean
	Target	Target	Target
AppVersion			
4.9.218.0	334	149	0.45
4.9.10586.965	266	99	0.37
4.9.10586.962	262	108	0.41
4.9.10586.916	311	128	0.41
4.9.10586.873	228	96	0.42
4.9.10586.839	248	112	0.45
4.9.10586.672	1198	536	0.45
4.9.10586.589	761	330	0.43
4.9.10586.494	1383	703	0.51
4.9.10586.456	1	1	1.00
4.9.10586.1177	2	1	0.50
4.9.10586.1106	11432	5091	0.45
4.9.10586.1045	617	264	0.43
4.9.10586.0	6213	3131	0.50
4.8.207.0	204	92	0.45
4.8.204.0	96	40	0.42
4.8.10240.17946	262	166	0.63
4.8.10240.17918	25	15	0.60
4.8.10240.17914	84	63	0.75
4.8.10240.17889	103	64	0.62
4.8.10240.17861	33	20	0.61
4.8.10240.17797	29	15	0.52
4.8.10240.17770	5	3	0.60
4.8.10240.17609	17	8	0.47
4.8.10240.17533	3	2	0.67
4.8.10240.17446	3	1	0.33
4.8.10240.17443	11385	5520	0.48
4.8.10240.17394	74	37	0.50
4.8.10240.17354	48	21	0.44
4.8.10240.17319	53	24	0.45
4.8.10240.17202	94	41	0.44
4.8.10240.17184	34	15	0.44
4.8.10240.17146	50	14	0.28
4.8.10240.17113	13	4	0.31
4.8.10240.17071	77	25	0.32
4.8.10240.16384	2617	1248	0.48
4.7.205.0	46	23	0.50
4.6.305.0	44	22	0.50
4.5.218.0	116	57	0.49
4.5.216.0	10	4	0.40
4.4.304.0	27	17	0.63
4.18.1809.2	733	358	0.49
4.18.1807.20063	39	19	0.49
4.18.1807.18075	288809	152973	0.53
4.18.1807.18072	12	7	0.58
4.18.1807.18070	1	1	1.00
4.18.1806.20033	1	1	1.00
4.18.1806.20021	7	2	0.29
4.18.1806.18062	47641	23108	0.49
4.17.17686.1003	5	0	0.00
4.17.17685.20082	4	3	0.75
4.17.17682.1000	4	0	0.00
4.17.17677.1000	5	1	0.20
4.17.17672.1000	4	1	0.25
4.16.17656.18052	13185	4310	0.33
4.16.17656.18051	4	1	0.25
4.15.17666.1000	2	0	0.00

```
4.15.17661.1001        4       1   0.25
4.15.17655.1000        1       0   0.00
4.15.17650.1001        1       1   1.00
4.14.17639.18041   10670    3372   0.32
4.14.17613.18039    2991     922   0.31
4.14.17613.18038      66      15   0.23
4.13.17639.1000        1       0   0.00
4.13.17618.1000        1       1   1.00
4.13.17604.1000        1       1   1.00
4.13.17134.319        97      64   0.66
4.13.17134.228     12729    5465   0.43
4.13.17134.191      1182     553   0.47
4.13.17134.112       651     309   0.47
4.13.17134.1       14414    7579   0.53
4.12.17007.18022    6470    2698   0.42
4.12.17007.18011    3392    1642   0.48
4.12.17007.17123     775     375   0.48
4.12.17007.17121       2       0   0.00
4.12.16299.15      20197    9207   0.46
4.11.15063.994         5       3   0.60
4.11.15063.447      5100    2389   0.47
4.11.15063.1155     2477    1057   0.43
4.11.15063.1154        2       1   0.50
4.11.15063.0        3874    2191   0.57
4.10.209.0         15292    7771   0.51
4.10.205.0            49      24   0.49
4.10.14393.953       403     190   0.47
4.10.14393.726        16      11   0.69
4.10.14393.2457        3       1   0.33
4.10.14393.2273       54      35   0.65
4.10.14393.2248       14       4   0.29
4.10.14393.1794     2425    1157   0.48
4.10.14393.1613      760     360   0.47
4.10.14393.1593      536     278   0.52
4.10.14393.1532        7       4   0.57
4.10.14393.1198     1271     653   0.51
4.10.14393.1066      404     192   0.48
4.10.14393.0        4399    2332   0.53

Resultados Value Counts:  AppVersion
 4.18.1807.18075    57.76
4.18.1806.18062     9.53
4.12.16299.15       4.04
4.10.209.0          3.06
4.13.17134.1        2.88
4.16.17656.18052    2.64
4.13.17134.228      2.55
4.9.10586.1106      2.29
4.8.10240.17443     2.28
4.14.17639.18041    2.13
4.12.17007.18022    1.29
4.9.10586.0         1.24
4.11.15063.447      1.02
4.10.14393.0        0.88
4.11.15063.0        0.77
4.12.17007.18011    0.68
4.14.17613.18039    0.60
4.8.10240.16384     0.52
4.11.15063.1155     0.50
4.10.14393.1794     0.48
4.9.10586.494       0.28
```

```
4.10.14393.1198     0.25
4.9.10586.672       0.24
4.13.17134.191      0.24
4.12.17007.17123    0.15
4.9.10586.589       0.15
4.10.14393.1613     0.15
4.18.1809.2         0.15
4.13.17134.112      0.13
4.9.10586.1045      0.12
4.10.14393.1593     0.11
4.10.14393.1066     0.08
4.10.14393.953      0.08
4.9.218.0           0.07
4.9.10586.916       0.06
4.9.10586.965       0.05
4.9.10586.962       0.05
4.8.10240.17946     0.05
4.9.10586.839       0.05
4.9.10586.873       0.05
4.8.207.0           0.04
4.5.218.0           0.02
4.8.10240.17889     0.02
4.13.17134.319      0.02
4.8.204.0           0.02
4.8.10240.17202     0.02
4.8.10240.17914     0.02
4.8.10240.17071     0.02
4.8.10240.17394     0.01
4.14.17613.18038    0.01
4.10.14393.2273     0.01
4.8.10240.17319     0.01
4.8.10240.17146     0.01
4.10.205.0          0.01
4.8.10240.17354     0.01
4.7.205.0           0.01
4.6.305.0           0.01
4.18.1807.20063     0.01
4.8.10240.17184     0.01
4.8.10240.17861     0.01
4.8.10240.17797     0.01
4.4.304.0           0.01
4.8.10240.17918     0.01
4.8.10240.17609     0.00
4.10.14393.726      0.00
4.10.14393.2248     0.00
4.8.10240.17113     0.00
4.18.1807.18072     0.00
4.5.216.0           0.00
4.18.1806.20021     0.00
4.10.14393.1532     0.00
4.11.15063.994      0.00
4.17.17686.1003     0.00
4.8.10240.17770     0.00
4.17.17677.1000     0.00
4.17.17682.1000     0.00
4.15.17661.1001     0.00
4.16.17656.18051    0.00
4.17.17672.1000     0.00
4.17.17685.20082    0.00
4.8.10240.17446     0.00
4.10.14393.2457     0.00
```

```
4.8.10240.17533     0.00
4.11.15063.1154     0.00
4.9.10586.1177      0.00
4.15.17666.1000     0.00
4.12.17007.17121    0.00
4.15.17655.1000     0.00
4.18.1807.18070     0.00
4.13.17618.1000     0.00
4.15.17650.1001     0.00
4.13.17604.1000     0.00
4.9.10586.456       0.00
4.18.1806.20033     0.00
4.13.17639.1000     0.00
Name: AppVersion, dtype: float64
```



Al igual que en la variable anterior, vemos que la mayoría de valores se acumulan en unas pocas versiones. Posteriormente los agruparemos para quedarnos solo con los más frecuentes.

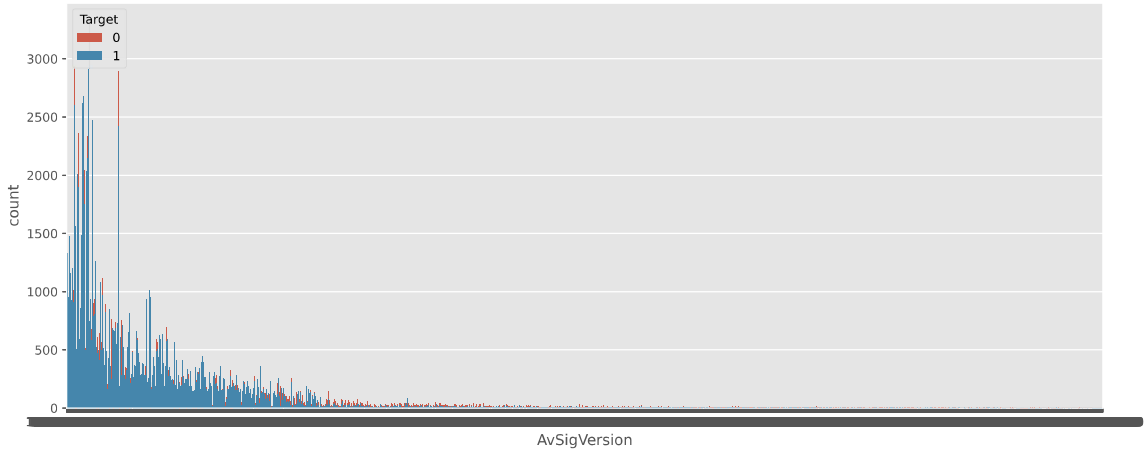
In [29]:

```
df=Visualizacion(df, "AvSigVersion", "Target")
```

```
Resultados Pivot Table:  AvSigVersion
                len    sum   mean
             Target Target Target
AvSigVersion
1.277.67.0        7      1   0.14
1.277.64.0       60     27   0.45
1.277.62.0       80     35   0.44
1.277.58.0       76     25   0.33
1.277.51.0      411    170   0.41
...             ...    ...    ...
1.207.2950.0      6      3   0.50
1.207.1891.0      1      1   1.00
1.199.1615.0      3      2   0.67
1.169.55.0        1      0   0.00
0.0.0.0           4      0   0.00

[6455 rows x 3 columns]

Resultados Value Counts:  AvSigVersion
 1.273.1420.0   1.15
1.263.48.0     1.11
1.275.1140.0   1.06
1.275.727.0    1.04
1.273.371.0    0.96
               ...
1.229.1357.0   0.00
1.237.969.0    0.00
1.249.1023.0   0.00
1.235.3087.0   0.00
1.241.660.0    0.00
Name: AvSigVersion, Length: 6455, dtype: float64
```



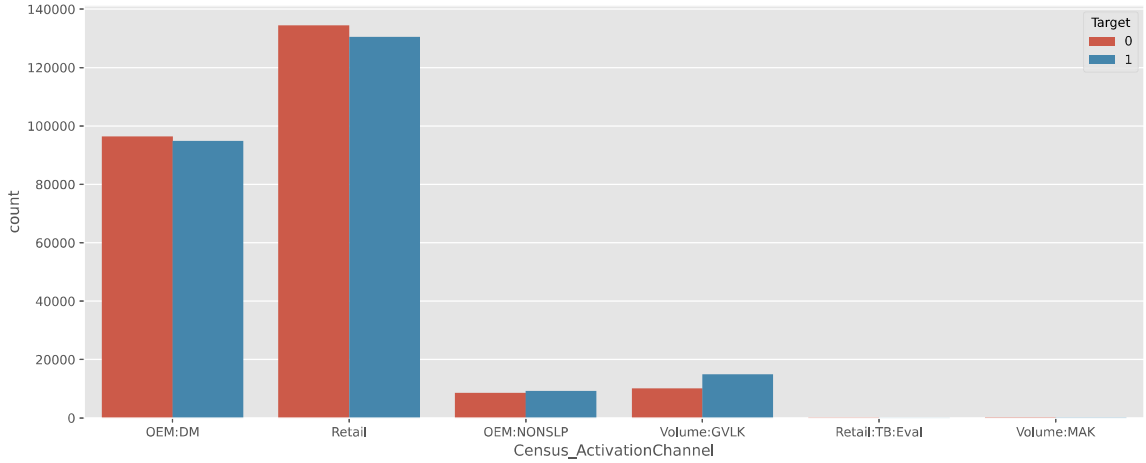
Con esta variable ocurre lo mismo que en las anteriores: muchos valores únicos, pero las frecuencias se reparten entre los primeros. Los agruparemos posteriormente. Se da una enorme diferencia entre algunas versiones en la prevalencia del target.

In [30]:

```
df=Visualizacion(df, "Census_ActivationChannel", "Target")
```

```
Resultados Pivot Table:  Census_ActivationChannel
                              len     sum   mean
                           Target  Target Target
Census_ActivationChannel
Volume:MAK                    468     212   0.45
Volume:GVLK                 25109   14963   0.60
Retail:TB:Eval                198      52   0.26
Retail                     264932  130532   0.49
OEM:NONSLP                  17943    9314   0.52
OEM:DM                     191350   94880   0.50

Resultados Value Counts:  Census_ActivationChannel
 Retail            52.99
OEM:DM            38.27
Volume:GVLK        5.02
OEM:NONSLP         3.59
Volume:MAK         0.09
Retail:TB:Eval     0.04
Name: Census_ActivationChannel, dtype: float64
```



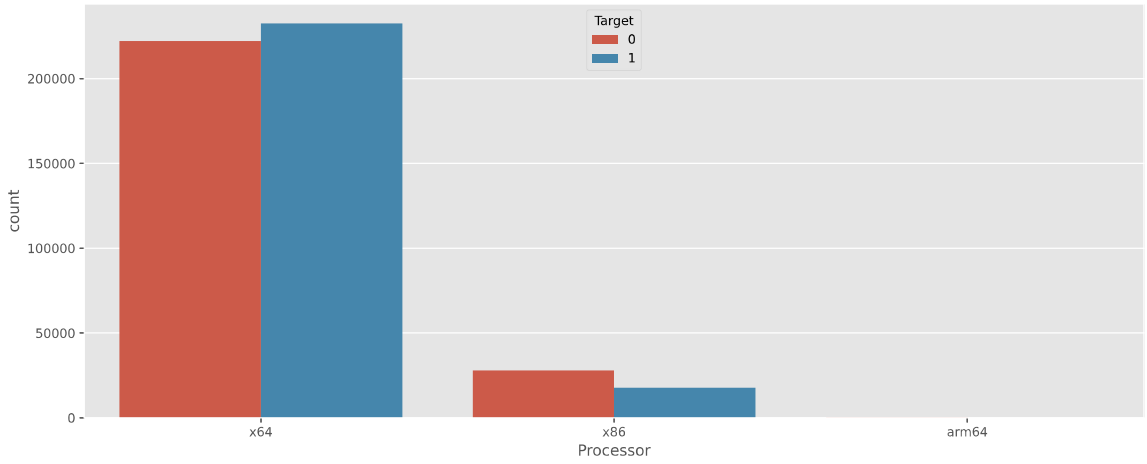
Pocos valores únicos y frecuencias respecto del target muy repartidas entre los dos primeros registros. Posteriormente haremos un OHE para codificar esta variable.

In [31]:

```
df=Visualizacion(df, "Processor", "Target")
```

```
Resultados Pivot Table:  Processor
              len     sum   mean
           Target  Target Target
Processor
x86         45563   17632   0.39
x64        454423  232321   0.51
arm64          14       0   0.00

Resultados Value Counts:  Processor
 x64     90.88
x86      9.11
arm64    0.00
Name: Processor, dtype: float64
```



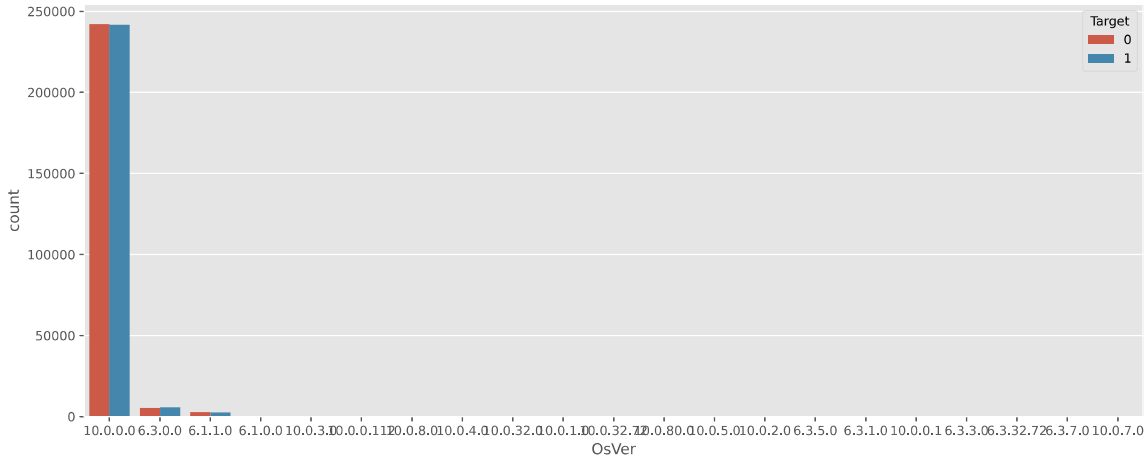
Al igual que en el anterior, pocos valores únicos y frecuencias respecto del target muy concentradas en un único registro (90,88%). Posteriormente haremos un OHE para codificar esta variable.

In [32]:

```
df=Visualizacion(df, "OsVer", "Target")
```

```
Resultados Pivot Table:  OsVer
               len     sum   mean
            Target  Target Target
OsVer
6.3.7.0          1       1   1.00
6.3.5.0          1       0   0.00
6.3.32.72        1       0   0.00
6.3.3.0          2       0   0.00
6.3.1.0          2       1   0.50
6.3.0.0      10818    5612   0.52
6.1.1.0       5281    2552   0.48
6.1.0.0         33      10   0.30
10.0.80.0        1       0   0.00
10.0.8.0         1       1   1.00
10.0.7.0         1       1   1.00
10.0.5.0         1       1   1.00
10.0.4.0         1       0   0.00
10.0.32.72       2       0   0.00
10.0.32.0        1       1   1.00
10.0.3.0        12       9   0.75
10.0.2.0         1       1   1.00
10.0.1.0         7       5   0.71
10.0.0.112       1       1   1.00
10.0.0.1         2       2   1.00
10.0.0.0    483830  241755   0.50

Resultados Value Counts:  OsVer
 10.0.0.0     96.77
6.3.0.0       2.16
6.1.1.0       1.06
6.1.0.0       0.01
10.0.3.0      0.00
10.0.1.0      0.00
6.3.1.0       0.00
10.0.32.72    0.00
10.0.0.1      0.00
6.3.3.0       0.00
10.0.5.0      0.00
10.0.80.0     0.00
10.0.4.0      0.00
6.3.32.72     0.00
10.0.2.0      0.00
10.0.8.0      0.00
6.3.7.0       0.00
10.0.7.0      0.00
6.3.5.0       0.00
10.0.0.112    0.00
10.0.32.0     0.00
Name: OsVer, dtype: float64
```



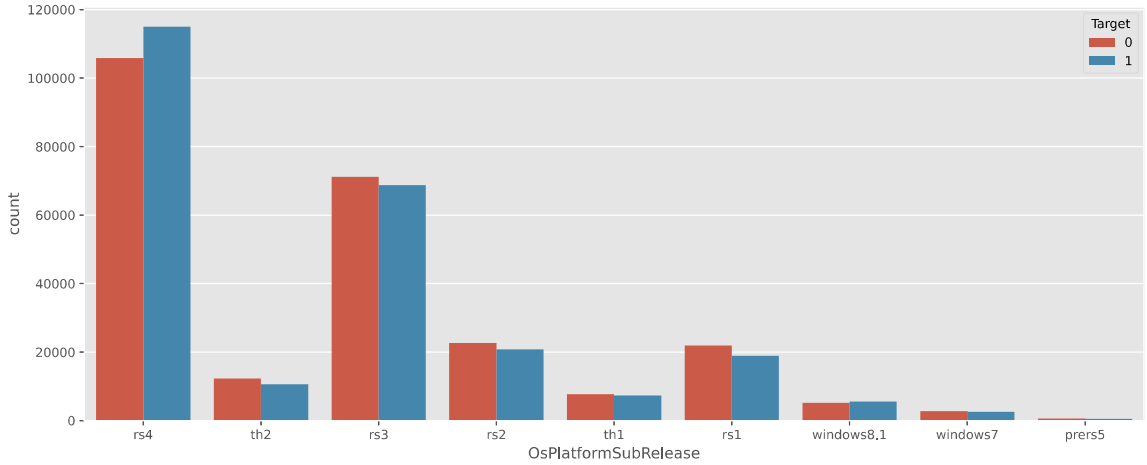
Más del 96% de las frecuencias se concentran en un único valor, por lo que no será una variable que influya en el árbol de decisión. Haremos una agrupación de valores más frecuentes y descartaremos el resto.

In [33]:

```
df=Visualizacion(df, "OsPlatformSubRelease", "Target")
```

```
Resultados Pivot Table:  OsPlatformSubRelease
                         len     sum   mean
                      Target  Target Target
OsPlatformSubRelease
windows8.1             10825    5614   0.52
windows7                5314    2562   0.48
th2                    22955   10608   0.46
th1                    15014    7309   0.49
rs4                   220779  114996   0.52
rs3                   139901   68735   0.49
rs2                    43352   20778   0.48
rs1                    40717   18848   0.46
prers5                  1143     503   0.44

Resultados Value Counts:  OsPlatformSubRelease
 rs4          44.16
rs3          27.98
rs2           8.67
rs1           8.14
th2           4.59
th1           3.00
windows8.1    2.17
windows7      1.06
prers5        0.23
Name: OsPlatformSubRelease, dtype: float64
```



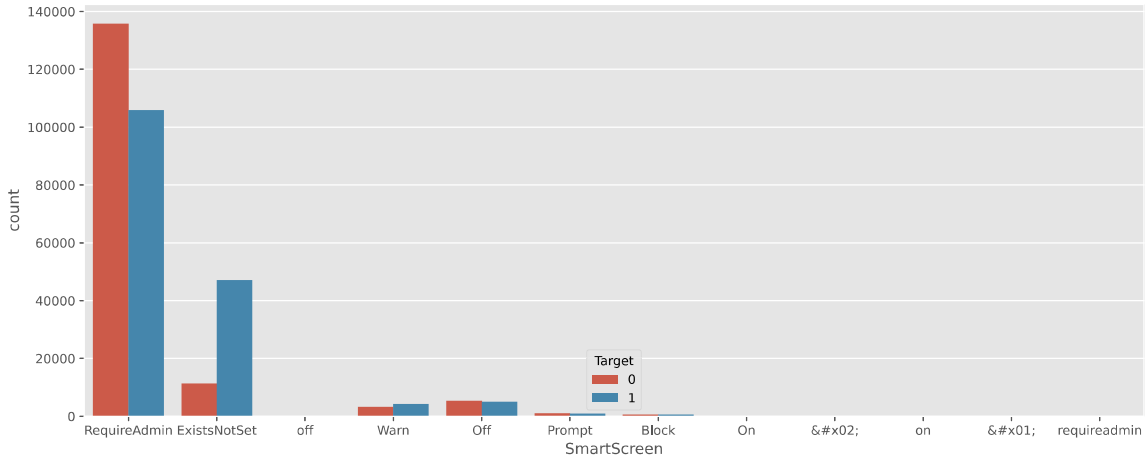
Pocos valores únicos y frecuencias respecto del target esta vez algo más repartidas. Posteriormente haremos un OHE para codificar esta variable.

In [34]:

```
df=Visualizacion(df, "SmartScreen", "Target")
```

```
Resultados Pivot Table:  SmartScreen
                 len     sum   mean
              Target  Target Target
SmartScreen
requireadmin       1       0   0.00
on                 8       4   0.50
off               75      43   0.57
Warn            7530    4306   0.57
RequireAdmin  241594  105890   0.44
Prompt          1950     928   0.48
On                53      36   0.68
Off            10388    5055   0.49
ExistsNotSet   58497   47115   0.81
Block           1274     640   0.50
&#x02;            20      11   0.55
&#x01;            14       6   0.43

Resultados Value Counts:  SmartScreen
 RequireAdmin   48.32
NaN            35.72
ExistsNotSet   11.70
Off             2.08
Warn            1.51
Prompt          0.39
Block           0.25
off             0.01
On              0.01
&#x02;          0.00
&#x01;          0.00
on              0.00
requireadmin    0.00
Name: SmartScreen, dtype: float64
```



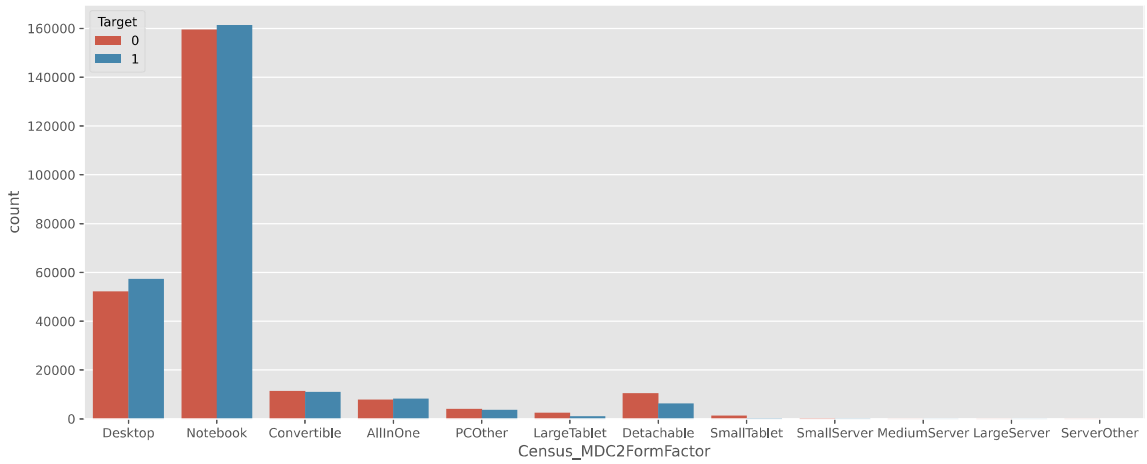
En esta variable vemos pocos valores únicos y con frecuencia concentrada en el primero. No obstante, como la probabilidad Sí/no del target es más dispareja, sí que puede ser una variable importante en nuestro árbol de decisión. Haremos un OHE

In [35]:

```
df=Visualizacion(df, "Census_MDC2FormFactor", "Target")
```

```
Resultados Pivot Table:  Census_MDC2FormFactor
                          len     sum   mean
                       Target  Target Target
Census_MDC2FormFactor
SmallTablet              1797     360   0.20
SmallServer               496     191   0.39
ServerOther                 2       0   0.00
PCOther                  7800    3715   0.48
Notebook               320948  161411   0.50
MediumServer              192      62   0.32
LargeTablet              3645    1138   0.31
LargeServer                50      10   0.20
Detachable              16802    6351   0.38
Desktop                109527   57306   0.52
Convertible             22369   11025   0.49
AllInOne                16372    8384   0.51

Resultados Value Counts:  Census_MDC2FormFactor
 Notebook       64.19
Desktop        21.91
Convertible     4.47
Detachable      3.36
AllInOne        3.27
PCOther         1.56
LargeTablet     0.73
SmallTablet     0.36
SmallServer     0.10
MediumServer    0.04
LargeServer     0.01
ServerOther     0.00
Name: Census_MDC2FormFactor, dtype: float64
```



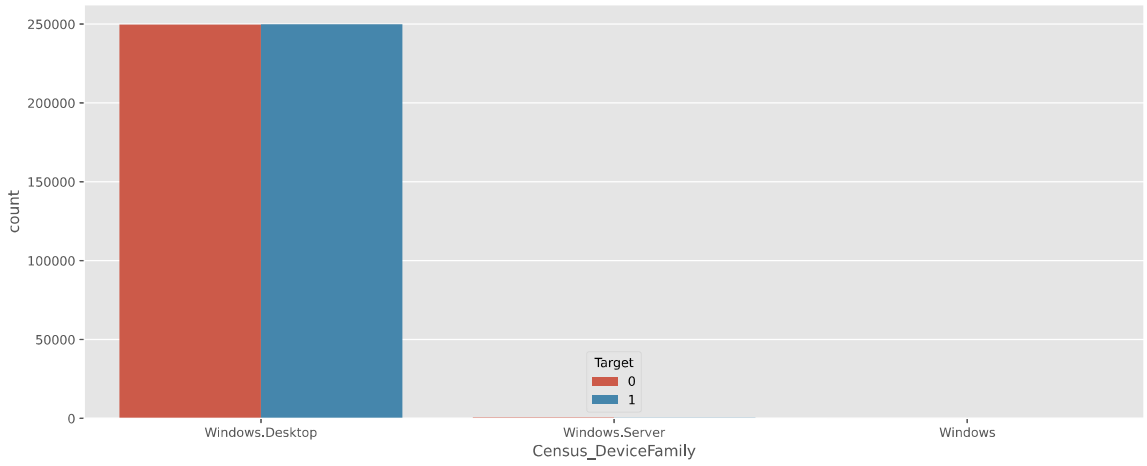
Frecuencias concentradas en un par de valores, pero esta vez la probabilidad del Target está repartida igualitariamente. Haremos un OHE.

In [36]:

```
df=Visualizacion(df, "Census_DeviceFamily", "Target")
```

```
Resultados Pivot Table:  Census_DeviceFamily
                         len     sum   mean
                      Target  Target Target
Census_DeviceFamily
Windows.Server           816     302   0.37
Windows.Desktop       499183  249651   0.50
Windows                    1       0   0.00

Resultados Value Counts:  Census_DeviceFamily
 Windows.Desktop   99.84
Windows.Server     0.16
Windows            0.00
Name: Census_DeviceFamily, dtype: float64
```



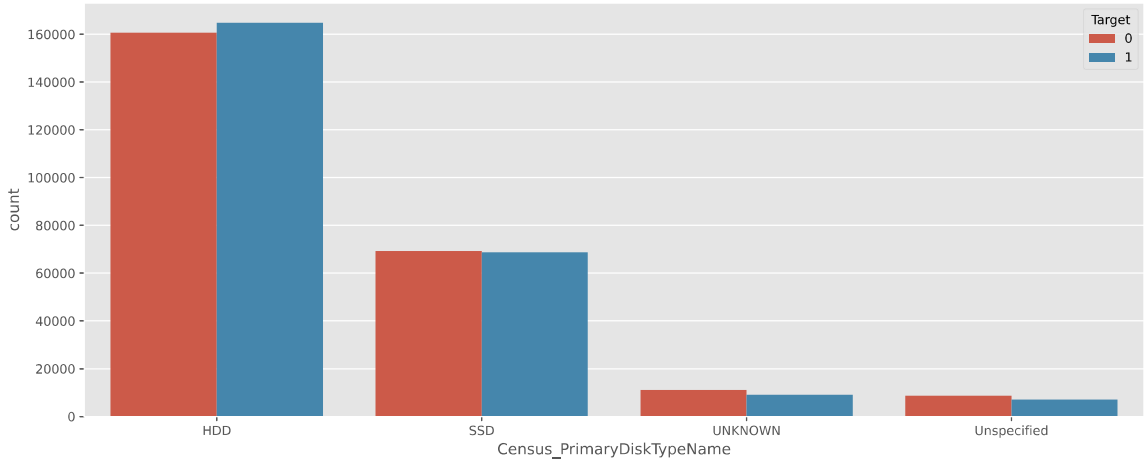
Con la frecuencia concentrada en el primer registro y con la probabilidad del target repartida un 50%, esta variable no nos servirá de mucho.

In [37]:

```
df=Visualizacion(df, "Census_PrimaryDiskTypeName", "Target")
```

```
Resultados Pivot Table:  Census_PrimaryDiskTypeName
                                len     sum   mean
                             Target  Target Target
Census_PrimaryDiskTypeName
Unspecified                   15624    7000   0.45
UNKNOWN                       20083    9019   0.45
SSD                          138155   68821   0.50
HDD                          325429  164762   0.51

Resultados Value Counts:  Census_PrimaryDiskTypeName
 HDD           65.09
SSD           27.63
UNKNOWN        4.02
Unspecified    3.12
NaN            0.14
Name: Census_PrimaryDiskTypeName, dtype: float64
```



Frecuencias repartidas entre los dos primeros valores y poca diferencia entre sí y no del target. No nos dirá mucho. Haremos un OHE.

In [38]:

```
df=Visualizacion(df, "Census_ChassisTypeName", "Target")
```

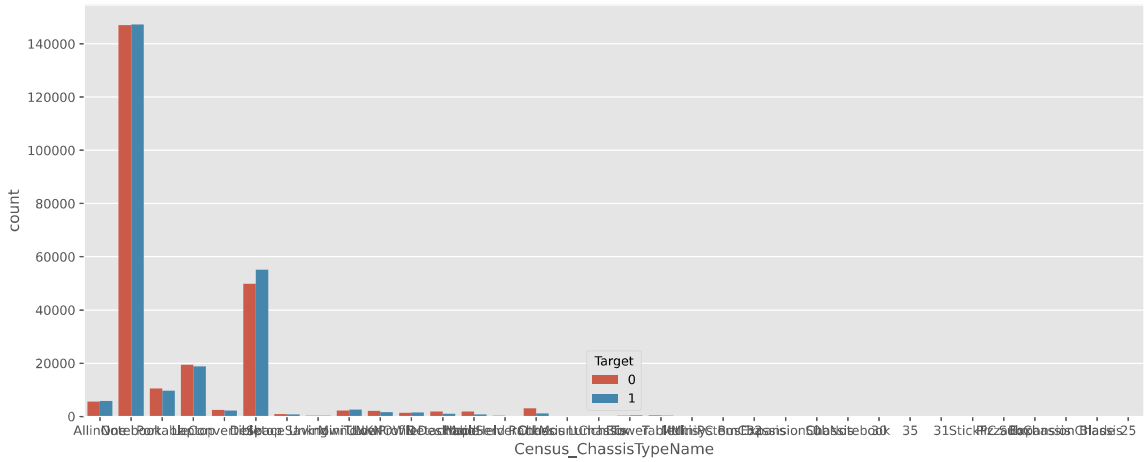
Resultados Pivot Table: Census_ChassisTypeName

Census_ChassisTypeName	len	sum	mean
	Target	Target	Target
Unknown	575	247	0.43
UNKNOWN	3695	1636	0.44
Tower	692	362	0.52
Tablet	730	263	0.36
SubNotebook	47	17	0.36
SubChassis	2	1	0.50
StickPC	7	0	0.00
SpaceSaving	1689	807	0.48
RackMountChassis	189	62	0.33
Portable	20181	9696	0.48
PizzaBox	3	1	0.33
Other	4215	1151	0.27
Notebook	294232	147270	0.50
MultisystemChassis	3	1	0.33
MiniTower	4849	2597	0.54
MiniPC	261	89	0.34
MainServerChassis	512	228	0.45
LunchBox	224	102	0.46
LowProfileDesktop	2878	1481	0.51
Laptop	38261	18864	0.49
HandHeld	2652	784	0.30
ExpansionChassis	1	0	0.00
Detachable	2930	1020	0.35
Desktop	104979	55150	0.53
Convertible	4685	2252	0.48
BusExpansionChassis	38	15	0.39
Blade	3	0	0.00
AllinOne	11407	5831	0.51
35	3	1	0.33
32	1	1	1.00
31	2	2	1.00
30	11	3	0.27
25	1	0	0.00
0	5	2	0.40

Resultados Value Counts: Census_ChassisTypeName

Notebook	58.85
Desktop	21.00
Laptop	7.65
Portable	4.04
AllinOne	2.28
MiniTower	0.97
Convertible	0.94
Other	0.84
UNKNOWN	0.74
Detachable	0.59
LowProfileDesktop	0.58
HandHeld	0.53
SpaceSaving	0.34
Tablet	0.15
Tower	0.14
Unknown	0.11
MainServerChassis	0.10
MiniPC	0.05
LunchBox	0.04
RackMountChassis	0.04
SubNotebook	0.01

```
BusExpansionChassis    0.01
NaN                    0.01
30                     0.00
StickPC                0.00
0                      0.00
35                     0.00
Blade                  0.00
MultisystemChassis     0.00
PizzaBox               0.00
31                     0.00
SubChassis             0.00
32                     0.00
25                     0.00
ExpansionChassis       0.00
Name: Census_ChassisTypeName, dtype: float64
```



Muchos valores únicos pero frecuencias concentradas en pocos valores, especialmente en Notebook.Haremos una agrupación (set_others) para descartar los menos frecuentes y quedarnos con los principales para posteriormente aplicar un OHE.

In [39]:

```
df=Visualizacion(df, "Census_PowerPlatformRoleName", "Target")
```

```
Resultados Pivot Table:  Census_PowerPlatformRoleName
                                len     sum   mean
                             Target  Target Target
Census_PowerPlatformRoleName
Workstation                    6235    3198   0.51
UNKNOWN                        1172     571   0.49
Slate                         27475   10082   0.37
SOHOServer                     2062    1024   0.50
PerformanceServer                 4       3   0.75
Mobile                       346378  173910   0.50
EnterpriseServer                406     177   0.44
Desktop                      116054   60942   0.53
AppliancePC                     212      46   0.22

Resultados Value Counts:  Census_PowerPlatformRoleName
 Mobile              69.28
Desktop              23.21
Slate                 5.50
Workstation           1.25
SOHOServer            0.41
UNKNOWN               0.23
EnterpriseServer      0.08
AppliancePC           0.04
PerformanceServer     0.00
NaN                   0.00
Name: Census_PowerPlatformRoleName, dtype: float64
```