reto

January 22, 2024

0.0.1 1. Importa las librerías

```python
import pandas as pd
import numpy as np
from sklearn.preprocessing import MinMaxScaler
from sklearn.preprocessing import OneHotEncoder
from sklearn.decomposition import PCA
pd.set_option('display.max_columns', None)
```

0.0.2 2. Lee el archivo CSV

```python
EmpleadosAttrition = pd.read_csv("empleados.csv")
EmpleadosAttrition.head(3)
```

[21]:

	Age	BusinessTravel	Department	DistanceFromHome	Education	\
0	50	Travel_Rarely	Research & Development	1 km	2	
1	36	Travel_Rarely	Research & Development	6 km	2	
2	21	Travel_Rarely	Sales	7 km	1	

	EducationField	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	\
0	Medical	1	997	4	
1	Medical	1	178	2	
2	Marketing	1	1780	2	

	Gender	JobInvolvement	JobLevel	JobRole	JobSatisfaction	\
0	Male	3	4	Research Director	4	
1	Male	3	2	Manufacturing Director	2	
2	Male	3	1	Sales Representative	2	

	MaritalStatus	MonthlyIncome	NumCompaniesWorked	HiringDate	Over18	\
0	Divorced	17399	9	06/06/2013	Y	
1	Divorced	4941	6	12/25/2015	Y	
2	Single	2679	1	2/14/2017	Y	

	OverTime	PercentSalaryHike	PerformanceRating	RelationshipSatisfaction	\
0	No	22	4	3	
1	No	20	4	4	
2	No	13	3	2	

```
       StandardHours  TotalWorkingYears  TrainingTimesLastYear  WorkLifeBalance  \
    0             80                 32                      1                2
    1             80                  7                      0                3
    2             80                  1                      3                3

       YearsInCurrentRole  YearsSinceLastPromotion  Attrition
    0                   4                        1         No
    1                   2                        0         No
    2                   0                        1        Yes
```

0.0.3 3. Elimina las columnas

```python
[22]: EmpleadosAttrition = EmpleadosAttrition.drop(['EmployeeCount',
      'EmployeeNumber', 'Over18', 'StandardHours'], axis = 1)
      #EmpleadosAttrition.head(3)
```

0.0.4 4. 5. 6. Analiza la información proporcionada

```python
[23]: #pd.DatetimeIndex(EmpleadosAttrition['HiringDate']) ## Error no existe fecha '2/
      30/2012'
      EmpleadosAttrition['Year'] = pd.DatetimeIndex(pd.
      to_datetime(EmpleadosAttrition["HiringDate"], format="%m/%d/%Y",
      errors="coerce")).year
      EmpleadosAttrition['Year'].fillna(0, inplace=True)
      EmpleadosAttrition['Year'] = EmpleadosAttrition['Year'].astype(int)


      EmpleadosAttrition['YearsAtCompany'] = 2018 - EmpleadosAttrition['Year']

      # Elimina Year
      EmpleadosAttrition = EmpleadosAttrition.drop(['Year'], axis = 1)

      #EmpleadosAttrition.head(5)
```

0.0.5 7. 8. 9. Renombra la variable DistanceFromHome a DistanceFromHome_km

```python
[24]: EmpleadosAttrition.rename(columns = {'DistanceFromHome':'DistanceFromHome_km'},
      inplace = True)
      EmpleadosAttrition['DistanceFromHome'] =
      EmpleadosAttrition['DistanceFromHome_km'].str.replace("km", "").astype(int)
```

0.0.6 10. Eliminar columnas Year, HiringDate y DistanceFromHome_km¶

```python
[25]: EmpleadosAttrition = EmpleadosAttrition.drop(['HiringDate',
      'DistanceFromHome_km'], axis = 1)
```

0.0.7 11. Nuevo frame

```
[26]: SueldoPromedioDepto = EmpleadosAttrition[['Department', 'MonthlyIncome']]
      SueldoPromedioDepto = SueldoPromedioDepto.groupby('Department').
      ↪agg({"MonthlyIncome": "mean"}).rename(columns={'MonthlyIncome':␣
      ↪'SueldoPromedio'}).reset_index()
      SueldoPromedioDepto
```

```
[26]:             Department  SueldoPromedio
      0       Human Resources     6239.888889
      1  Research & Development   6804.149813
      2                 Sales     7188.250000
```

0.0.8 12. Escalado

```
[27]: scaler = MinMaxScaler()
      EmpleadosAttrition['MonthlyIncome'] = scaler.
      ↪fit_transform(EmpleadosAttrition[['MonthlyIncome']])
      #EmpleadosAttrition.head()
```

0.0.9 13. Variables categóricas

```
[28]: columns = ['OverTime', 'BusinessTravel', 'Department', 'EducationField',␣
      ↪'Gender', 'JobRole', 'MaritalStatus', 'Attrition']
      EmpleadosAttrition = pd.get_dummies(EmpleadosAttrition, columns = columns,␣
      ↪drop_first = True, dtype=int)
      EmpleadosAttrition.head(3)
```

```
[28]:    Age  Education  EnvironmentSatisfaction  JobInvolvement  JobLevel  \
      0   50          2                        4               3         4
      1   36          2                        2               3         2
      2   21          1                        2               3         1

         JobSatisfaction  MonthlyIncome  NumCompaniesWorked  PercentSalaryHike  \
      0                4       0.864269                   9                 22
      1                2       0.207340                   6                 20
      2                2       0.088062                   1                 13

         PerformanceRating  RelationshipSatisfaction  TotalWorkingYears  \
      0                  4                         3                 32
      1                  4                         4                  7
      2                  3                         2                  1

         TrainingTimesLastYear  WorkLifeBalance  YearsInCurrentRole  \
      0                      1                2                   4
      1                      0                3                   2
      2                      3                3                   0
```

```
     YearsSinceLastPromotion  YearsAtCompany  DistanceFromHome  OverTime_Yes  \
0                          1               5                 1             0
1                          0               3                 6             0
2                          1               1                 7             0

     BusinessTravel_Travel_Frequently  BusinessTravel_Travel_Rarely  \
0                                   0                             1
1                                   0                             1
2                                   0                             1

     Department_Research & Development  Department_Sales  \
0                                    1                 0
1                                    1                 0
2                                    0                 1

     EducationField_Life Sciences  EducationField_Marketing  \
0                                0                         0
1                                0                         0
2                                0                         1

     EducationField_Medical  EducationField_Other  \
0                         1                     0
1                         1                     0
2                         0                     0

     EducationField_Technical Degree  Gender_Male  JobRole_Human Resources  \
0                                  0            1                         0
1                                  0            1                         0
2                                  0            1                         0

     JobRole_Laboratory Technician  JobRole_Manager  \
0                                0                0
1                                0                0
2                                0                0

     JobRole_Manufacturing Director  JobRole_Research Director  \
0                                 0                          1
1                                 1                          0
2                                 0                          0

     JobRole_Research Scientist  JobRole_Sales Executive  \
0                             0                        0
1                             0                        0
2                             0                        0

     JobRole_Sales Representative  MaritalStatus_Married  MaritalStatus_Single  \
```

	0	0	0
0	0	0	0
1	0	0	0
2	1	0	1

	Attrition_Yes
0	0
1	0
2	1

0.0.10 14. 15. Correlacion

```python
[29]: # Calcular la correlación lineal entre todas las columnas
      corr = EmpleadosAttrition.corr()['Attrition_Yes']

      # Filtrar las columnas con una correlación inferior a 0.1
      cols_to_drop = [col for col in EmpleadosAttrition.columns if corr[col] < 0.1]

      # Eliminar las columnas
      EmpleadosAttritionFinal = EmpleadosAttrition.drop(columns=cols_to_drop, axis =
       ↪1)
```

0.0.11 16. EmpleadosAttritionPCA

```python
[30]: variablesNum = EmpleadosAttritionFinal[
      ['OverTime_Yes', 'EducationField_Technical Degree', 'JobRole_Laboratory
       ↪Technician', 'JobRole_Sales Representative',
       'MaritalStatus_Single', 'Attrition_Yes']]
      pca = PCA(6)
      pca.fit(EmpleadosAttritionFinal)
      nuevasFeat = pca.transform(variablesNum)
```

0.0.12 17. Agrega el mínimo

```python
[31]: EmpleadosAttritionPCA = EmpleadosAttritionFinal.assign(C0=(nuevasFeat[:,0]))
      EmpleadosAttritionPCA = EmpleadosAttritionPCA.assign(C1=(nuevasFeat[:,1]))
      EmpleadosAttritionPCA = EmpleadosAttritionPCA.assign(C2=(nuevasFeat[:,2]))
      EmpleadosAttritionPCA = EmpleadosAttritionPCA.assign(C3=(nuevasFeat[:,3]))
      EmpleadosAttritionPCA
```

```
[31]:      OverTime_Yes  EducationField_Technical Degree  \
      0                0                                0
      1                0                                0
      2                0                                0
      3                0                                0
      4                1                                0
      ..             ...                              ...
      395              1                                0
```

```
396              1                              0
397              1                              0
398              0                              0
399              0                              0

     JobRole_Laboratory Technician  JobRole_Sales Representative  \
0                                0                             0
1                                0                             0
2                                0                             1
3                                0                             0
4                                0                             0
..                             ...                           ...
395                              1                             0
396                              0                             0
397                              0                             0
398                              1                             0
399                              0                             0

     MaritalStatus_Single  Attrition_Yes        C0        C1        C2  \
0                       0              0         0 -0.418658  0.025340 -0.139366
1                       0              0         0 -0.418658  0.025340 -0.139366
2                       1              1         1  0.732381 -0.770272  0.132097
3                       1              0         0  0.129888 -0.755010 -0.129133
4                       0              1         1  0.748708  0.716183 -0.077530
..                    ...            ...       ...       ...       ...       ...
395                     0              1         1  0.747378  0.841807  0.844788
396                     0              1         1  0.748708  0.716183 -0.077530
397                     0              0         0  0.232371  0.616774 -0.377940
398                     0              0         0 -0.419988  0.150963  0.782952
399                     0              0         0 -0.418658  0.025340 -0.139366

           C3
0    0.082370
1    0.082370
2    0.846814
3   -0.202455
4    0.413427
..        ...
395  0.103794
396  0.413427
397 -0.302280
398 -0.227263
399  0.082370

[400 rows x 10 columns]
```

0.0.13 18. Guarda

```
[34]: EmpleadosAttritionFinal.to_csv("EmpleadosAttritionFinal.csv", index=False)
```