|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| import pandas as pd  import numpy as np  import matplotlib.pyplot as plt  %matplotlib inline  plt.rcParams['figure.figsize'] = (16, 9)  plt.style.use('ggplot')  from sklearn.decomposition import PCA  from sklearn.preprocessing import StandardScaler    #cargamos los datos de entrada  dataframe = pd.read\_csv(r"comprar\_alquilar.csv")  print(dataframe.tail(10))    #normalizamos los datos  scaler=StandardScaler()  df = dataframe.drop(['comprar'], axis=1) # quito la variable dependiente "Y"  scaler.fit(df) # calculo la media para poder hacer la transformacion  X\_scaled=scaler.transform(df)# Ahora si, escalo los datos y los normalizo    ============================================================================================================  #Instanciamos objeto PCA y aplicamos  pca=PCA(n\_components=9) # Otra opción es pca = PCA(.85)  pca.fit(X\_scaled) # obtener los componentes principales  X\_pca=pca.transform(X\_scaled) # convertimos nuestros datos con las nuevas dimensiones de PCA    print("shape of X\_pca", X\_pca.shape)  expl = pca.explained\_variance\_ratio\_  print(expl)  print('suma:',sum(expl[0:5])) #0.9  #Vemos que con 5 componentes tenemos algo mas del 85% de varianza explicada     |  |  | | --- | --- | | #graficamos el acumulado de varianza  #explicada en las nuevas dimensiones  plt.plot(np.cumsum(pca.explained\_variance\_ratio\_))  plt.xlabel('number of components')  plt.ylabel('cumulative explained variance')  plt.show()  En esta gráfica de variabilidad explicada acumulada, vemos que tomando los primeros 5 componentes llegamos al 90% |  | | #graficamos en 2 Dimensiones, tomando  #los 2 primeros componentes principales  Xax=X\_pca[:,0]  Yax=X\_pca[:,1]  labels=dataframe['comprar'].values  cdict={0:'red',1:'green'}  labl={0:'Alquilar',1:'Comprar'}  marker={0:'\*',1:'o'}  alpha={0:.3, 1:.5}  fig,ax=plt.subplots(figsize=(7,5))  fig.patch.set\_facecolor('white')  for l in np.unique(labels):      ix=np.where(labels==l)      ax.scatter(Xax[ix],Yax[ix],c=cdict[l],label=labl[l],s=40,  marker=marker[l],alpha=alpha[l])    plt.xlabel("First Principal Component",fontsize=14)  plt.ylabel("Second Principal Component",fontsize=14)  plt.legend()  plt.show() | [Chart, scatter chart  Description automatically generated](https://www.aprendemachinelearning.com/wp-content/uploads/2018/10/grafica-2componentes.png)  Al reducir de 9 a 2 dimensiones nos da la idea de donde visualizar las predicciones de comprar o alquilar | |