Texto

Descripción generada automáticamente

Carmona Bartolome Aldo Armando

1AVI

Introducción a Ciencia de Datos

Práctica No. 3: Aprendizaje automático

Repo del codigo: [Link](https://github.com/waldos09/ICD_1)

Introducción

El Machine learning, o aprendizaje automático, es una forma de la IA que permite a un sistema aprender de los datos en lugar de aprender mediante la programación explícita. Sin embargo, machine learning no es un proceso sencillo. Conforme el algoritmo ingiere datos de entrenamiento, es posible producir modelos más precisos basados en datos. Un modelo de machine learning es la salida de información que se genera cuando entrena su algoritmo de machine learning con datos. Después del entrenamiento, al proporcionar un modelo con una entrada, se le dará una salida. Por ejemplo, un algoritmo predictivo creará un modelo predictivo.

Existen 3 diferentes tipos de algoritmos para la clasificación autónoma de objetos o datos:

* Clasificación supervisada: disponemos de un conjunto de datos. Construímos un modelo en la fase de entrenamiento (training) utilizando dichas etiquetas, que nos dicen si una imagen está clasificada correcta o incorrectamente por el modelo. Una vez construído el modelo podemos utilizarlo para clasificar nuevos datos que, en esta fase, ya no necesitan etiqueta para su clasificación, aunque sí la necesitan para evaluar el porcentaje de objetos bien clasificados.
* Clasificación no supervisada: los datos no tienen etiquetas (o no queremos utilizarlas) y estos se clasifican a partir de su estructura interna (propiedades, características).
* Clasificación semisupervisada: algunos datos de entrenamiento tienen etiquetas, pero no todos. Este último caso es muy típico en clasificación de imágenes, donde es habitual disponer de muchas imágenes mayormente no etiquetadas.

El siguiente programa utiliza un CSV (data1.csv) para la obtención de datos previamente instanciados en el archivo acerca de 50 ofertas de trabajo para los Científicos de datos, en este caso los datos incluidos fueron el salario y los años de experiencia necesarios para ser contratado . El programa utiliza pandas para el manejo de los datos, puesto que esto puede dar una mayor organización si se tienen muchos datos en un solo archivo y por ultimo se uso matplotlib.pyplot para el ploteo de las graficas para demotrar las predicciones que el algoritmo usado “K-means” arroja.

Desarrollo

K-means es un algoritmo de clasificación no supervisada (clusterización) que agrupa objetos en k grupos basándose en sus características. El agrupamiento se realiza minimizando la suma de distancias entre cada objeto y el centroide de su grupo o cluster. Se suele usar la distancia cuadrática. El algoritmo utiliza los siguientes pasos:

1. Inicialización: una vez escogido el número de grupos, k, se establecen k centroides en el espacio de los datos, por ejemplo, escogiéndolos aleatoriamente.
2. Asignación objetos a los centroides: cada objeto de los datos es asignado a su centroide más cercano.
3. Actualización centroides: se actualiza la posición del centroide de cada grupo tomando como nuevo centroide la posición del promedio de los objetos pertenecientes a dicho grupo.
4. Se repiten los pasos 2 y 3 hasta que los centroides no se mueven, o se mueven por debajo de una distancia umbral en cada paso.

El algoritmo k-means resuelve un problema de optimización, siendo la función a optimizar (minimizar) la suma de las distancias cuadráticas de cada objeto al centroide de su cluster.

Imagen que contiene objeto, reloj

Descripción generada automáticamente

# -\*- coding: utf-8 -\*-

"""

Created on Tue Nov  9 16:16:28 2021

@author: waldo

"""

import pandas as pd

import matplotlib.pyplot as plot

import sklearn.cluster as skl

#datos del csv

dataset = pd.read\_csv('data1.csv')

x = dataset['Cientifico de Datos']

y = dataset['Experiencia']

#Inicializacion del cluster

cluster = skl.KMeans(n\_clusters=5)

#entrena el algoritmo con los datos

cluster.fit(dataset)

#obtencion de los centroides

cnt = cluster.cluster\_centers\_

#obtiene una lista con etiquetas de los datos

etiquetas = cluster.predict(dataset)

#agrega al frame de datos una nueva columna para las etiquetas

dataset['etiquetas'] = etiquetas

#tabla de colores

colores = ['red','orange','green','pink','blue']

colores\_datos=[]

colores\_cnt = []

for row in etiquetas:

    colores\_datos.append(colores[row])

for i in range(len(cnt)):

    colores\_cnt.append(colores[i])

#Grafica de dispersión

plot.scatter(x, y, c=colores\_datos, marker='o', s=40)

plot.scatter(cnt[:,0], cnt[:,1], c=colores\_cnt, marker='\*', s=200)

plot.xlabel('Salario')

plot.ylabel('Experiencia')

plot.title('K-Means Clustering Salarios vs Experiencia (Pedagogos)')

plot.grid(color='gray', linestyle='--', linewidth = 0.5)

plot.show()

Ejecución

Primeramente, tomemos en cuenta los datos de entrada para el calculo de las estadísticas solicitadas para los salarios de los Científicos de Datos mostrando los datasets referentes a estos. Tomaremos como valor independiente (X) los salarios de estas ofertas y como valor dependiente (Y) tomaremos la cantidad de años que requiere el empleado para obtener el trabajo.

|  |  |
| --- | --- |
| X = Salarios | Y = Experiencia |
|  |  |

Finalmente se muestra la grafica generada con ayuda del algoritmo seleccionado

Gráfico, Gráfico de dispersión

Descripción generada automáticamente

Conclusión

Como se puede observan en la grafica se puede deducir de forma evidente que dependiendo del salario que esta dispuesto a pagar el empleador, la cantidad de años de experiencia necesarios va aumentar o disminuir tomando como regla que entre mayor sea la cantidad dispuesta a pagar por el trabajo, mayor será la cantidad de años requerida, siendo esta cantidad menor a la cantidad promedio pagada por 3 años de experiencia. Con esto se puede inferir que ser bien pagado (ms beneficio para el empleado) ronda entre los primeros 3 años de experiencia ya que después de estos la suma que se pretende pagar es “justa” de acuerdo a tus habilidades en el campo.

El aprendizaje automático es de suma vitalidad para el futuro de la industria y en general de la humanidad ya que con ayuda de datos previamente obtenidos podemos tener una pequeña “vista hacia el futuro”, esto siempre tomando en cuenta que métodos se usaron para predecir los próximos datos y por supuesto la integridad y cantidad de los datos que se están tomando como referencia.

Referencias de Apoyo

IBM. (s. f.). *Machine Learning*. machine learnling. Recuperado 9 de noviembre de 2021, de https://www.ibm.com/mx-es/analytics/machine-learning

*kmeans*. (s. f.). K-means. Recuperado 9 de noviembre de 2021, de https://www.unioviedo.es/compnum/laboratorios\_py/kmeans/kmeans.html