BIG DATA ANALYSIS
13/06/2024

Nome:	Cognome:
Matricola:	

Parte 1	
Parte 2	
Totale	

Regole:

1. E' vietato comunicare con altri durante la prova.
2. Al termine della prova, lo studente rinomina il notebook con il proprio nome e cognome e lo manda via email alla docente: federica.rollo@unimore.it, oggetto: BDA: 13-6-2024.
3. I risultati sono pubblicati entro il giorno 21/6/2024.

Note:

Durata della prova: 2 ore. Il file csv che si trova al link https://bit.ly/3xoah2F

Parte 0: Il Dataset

Il dataset (preso da kaggle -- https://www.kaggle.com/datasets/iabhishekofficial/mobile-price-classification?select=train.csv) contiene dati relativi ad alcuni cellulari. Il dataset contiene diverse feature descritte di seguito, la variabile da predire è "price_range":

- battery_power: potenza della batteria (mAh)
- blue: ha il bluetooth oppure no (boolean)
- clock_speed: velocità del microprocessore
- dual_sim: ha il supporto dual sim oppure no (boolean)
- fc: mega pixels della telecamera frontale
- four_g: ha il 4G oppure no (boolean)
- int_memory: nemoria interna (GB)
- m_dep: Mobile Depth (cm)
- mobile_wt: peso
- n_cores: numero di core del processore
- pc: mega pixels della fotocamera esterna
- px_height: Pixel Resolution Height
- px_width: Pixel Resolution Width
- ram: RAM (MB)
- sc_h: altezza schermo (cm)
- sc_w: larghezza schermo (cm)
- talk_time: durata massima della batteria
- three_g: ha il 3G oppure no (boolean)
- touch_screen: ha il touch screen oppure no (boolean)
- wifi: ha il wifi oppure no (boolean)
- price_range: colonna target con valori 0 (costo basso), 1 (costo medio), 2 (costo alto), 3 (costo molto alto)

Parte 1: Analisi (10 punti)

1. Quante sono le istanze contenute nel dataset? _____ Il dataset è completo (cioè per ogni istanza tutti i valori di ogni attributo sono sempre correttamente specificati – non esistono "missing values")? _____ Il dataset è bilanciato per quanto riguarda la classe da predire? _____ (punti 1).

2. Considerare la feature `battery_power` e rappresentare con un istogramma la distribuzione dei valori. Raggruppare poi i valori secondo questa suddivisione: gruppo1 501-800; gruppo2 801-1200; gruppo3 1201-1600; gruppo4 1601-1998, visualizzare la distribuzione nei gruppi. Indicare per ogni gruppo il numero di istanze per ogni range di prezzo. (punti 3)

3. Considerare solo i cellulari che hanno il 4G e una RAM superiore (>) a 2 GB (2048 MB). Rappresentare in uno scatterplot i valori di ram (ascisse) e memoria interna (ordinate). Colorare i punti nel grafico in base al valore della colonna `price_range`. Usare i nomi dei due attributi come etichette dell'asse delle ascisse e dell'asse delle ordinate. (punti 4)

4. Realizzare una tabella pivot in cui rappresentare il numero di cellulari per ogni range di prezzo (variabile sulle colonne) considerando sulle righe le variabili `touch_screen` e `int_memory` (suddivisa in 5 gruppi). (punti 2)

Parte 2: Trasformazione e Predizione (20 punti)

1. Si vuole predire il valore di range di prezzo sulla base degli attributi presenti nel dataset. Ricaricare il dataset originale, eliminare eventuali attributi inutili (giustificare la scelta), eliminare le eventuali istanze che contengono valori nulli, e dividere il dataset in modo che 3/4 degli elementi siano contenuti in un nuovo dataset "train" e 1/4 nel dataset "test" preservando le proporzioni delle classi nella colonna target.

Allenare il train con il modello DecisionTree e valutare l'accuracy ottenuta calcolata sia sul dataset train sia sul dataset test. Confrontare i risultati ottenuti con quelli ottenuti con una predizione basata sul modello KNeighborsClassifier. Effettuare alcune considerazioni sui risultati ottenuti, tenendo in considerazione anche l'analisi della confusion matrix e la predizione effettuata da un dummy classifier a scelta. (punti 4)

2. Confrontare l'accuratezza ottenuta nel punto precedente con l'accuratezza che si ottiene con una 10 Fold cross validation. (punti 1)

3. Trovare i parametri migliori del classificatore DecisionTree. Agire sui parametri criterion e min_samples_leaf. Verificare se l'accuratezza che si ottiene con la nuova configurazione supera quella ottenuta con i parametri di default al punto 1. (punti 3)

4. Studiare la correlazione tra tutte le feature del dataset, creare un dataframe che contiene, oltre alla colonna target, le 5 feature più correlate (positivamente) al target. Ripetere la classificazione sul nuovo dataset e verificare se l'accuratezza ottenuta con DecisionTree e KNeighborsClassifier migliora (punti 3).

5. Creare una pipeline in cui gli attributi `int_memory, ram` e `talk_time` sono scalati in modo che abbiano media 0 e varianza 1, gli attributi `mobile_wt` e `battery_power` sono discretizzati in 5 intervalli, e tutti gli altri attributi sono lasciati invariati. La pipeline deve applicare il modello DecisionTree con i parametri migliori trovati al punto 2. Valutare l'accuratezza della classificazione. (punti 3)

6. Aggiungere alla pipeline del punto 5 la funzione SelectKBest (https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html?highlight=selectkbest#sklearn.feature_selection.SelectKBest). Utilizzare la funzione di

gridSearchCV per selezionare il K migliore e anche il numero migliore di bin in cui discretizzare i valori di `mobile_wt` e `battery_power` (scegliere a piacere alcuni valori) (punti 3).

7. Creare una nuova pipeline che applica la decomposizione TruncatedSVD (simile alla PCA - https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html) al dataset iniziale e aggiunge le componenti ottenute alla pipeline del punto 5. Valutare il valore migliore per il numero di componenti di TruncatedSVD tra 2, 4 e 6. (punti 3).