

Pràctica 2: Etiquetatge

Part 3: Millores i Avaluació

Intel·ligència Artificial

Departament de Ciències de la Computació
Universitat Autònoma de Barcelona

1 Introducció

En aquesta Part 3 de la pràctica continuem treballant en resoldre el problema d'etiquetatge d'imatges, combinant els mètodes de Kmeans d'etiquetatge de color i KNN d'etiquetatge de forma per a obtenir un resultat similar al que es veu a la figura 1.

En aquesta part de la pràctica resoldrem el problema d'etiquetar automàticament imatges per tipus de peça. Farem servir 8 tipus de peces de roba.

1. K-means (**k_means**): Mètode de classificació no supervisada que es farà servir per trobar els colors predominants d'una imatge.
2. K-NN (**k_nn**): Mètode de classificació supervisada que es farà servir per assignar les etiquetes de tipus de roba.



Figure 1: Objectius de la pràctica

2 Fitxers necessaris

Farem servir els mateixos fitxers que ja varem descarregar per fer la Part 1 i 2, però ens centrarem sobretot en el fitxer `My_labeling.py`. Com que el K-NN és un algorisme d'aprenentatge supervisat necessitarem les imatges que hi ha al directori `Images`, i els conjunts d'aprenentatge i de test que hi tenim:

1. **Images:** Carpeta que conté les bases de dades amb les imatges que utilitzarem.
Dins d'aquesta carpeta trobareu:
 - (a) **Test:** Conjunt d'imatges que farem servir com a conjunt de test.
 - (b) **Train:** Conjunt d'imatges que utilitzarem com a conjunt d'entrenament per a la classificació de formes.
 - (c) **gt.json:** Arxiu amb la informació del *Ground-Truth* de les imatges.
2. **test:** Carpeta amb fitxers per a fer els tests (En aquesta part 3 ja no els farem servir).
3. **codi:** A l'arrel del fitxer comprimit trobareu tot el codi necessari i els fitxers que haureu d'omplir (`My_labeling.py`). Els fitxers necessaris per treballar són:
 - (a) `utils.py`: Conté una sèrie de funcions que us poden ser útils principalment per a la part del KMeans.
 - (b) `TestCases_kmeans.py`: Arxiu amb el qual podreu comprovar si les funcions que programeu en el fitxer `Kmeans.py` donen el resultat esperat.
 - (c) `TestCases_knn.py`: Arxiu amb el qual podreu comprovar si les funcions que programeu en el fitxer `KNN.py` donen el resultat esperat.
 - (d) `Kmeans.py`: Arxiu on ja hi haureu programat les funcions necessàries per a implementar Kmeans per classificar el COLOR.
 - (e) `KNN.py`: Arxiu on haureu de programar les funcions necessàries per a implementar KNN per classificar el NOM de la peça de roba.
 - (f) **`utils_data.py`:** Conté una sèrie de funcions que us poden ser útils en aquesta última part.
 - (g) **`My_labeling.py`:** Arxiu on combinareu els dos mètodes d'etiquetatge i les vostres millores per a obtenir el nom finals de les imatges.

3 Funcions útils (`utils_data.py`)

El fitxer `utils_data.py` que us donem programat, conté un seguit de funcions que us serviran per fer l'anàlisi dels vostres resultats.

`read_dataset`: Funció que carrega les imatges de `train` i `test`, i el *Ground-Truth* de color i de forma. Aquesta funció ja es crida a l'inici del fitxer `My_labeling`.

visualize_retrieval: Funció que rep com a entrada el conjunt ordenat d'imatges que han estat ordenades com a resultat d'una cerca concreta i el número d'imatges que volem mostrar. La funció fa una visualització d'aquestes imatges. Com a paràmetres addicionals també pot rebre:

1. **info:** Llista de strings que es mostraran sobre cada una de les imatges donades pel primer paràmetre (per exemple: ['blue jeans', 'blue flip-flops', 'blue jeans']).
2. **ok:** Llista d'elements de tipus **true** o **false**. Cada element fa referència a la imatge corresponent del primer paràmetre. El valor és **true** si la imatge predicció i el *Ground-Truth* coincideixen, retorna **false** en altre cas.
3. **title:** String que la funció posarà com a títol global a la figura.
4. **query:** Imatge respecte a la que hem cercat les imatges més semblants (només és útil en el cas que s'hagi implementat aquesta funció).

Plot3DCloud: Funció que rep com a entrada la classe **Kmeans** i retorna la visualització del núvol de punts que conté la imatge dins de l'espai RGB. Cada punt apareix del color RGB corresponent al seu centroide més proper.

visualize_k_means: Funció que rep com a entrada la classe **Kmeans** i el tamany de la imatge original i retorna la visualització da imatge origina utilitzant només els colors dels centroides que hem obtingut.

4 Què s'ha de programar?

En aquesta tercera part de la pràctica programareu experiments per avaluar l'eficàcia dels vostres mètodes de classificació Kmeans i KNN, així com millores per a aquests.

Les funcions a programar es dividiran en els següents apartats:

1. **Funcions d'anàlisi qualitatiu:** Funcions que ens permeten avaluar d'una manera visual els nostres classificadors. Com a mínim n'haureu d'implementar una.
2. **Funcions d'anàlisi quantitatiu:** Funcions que ens permeten avaluar de manera numèrica els nostres classificadors. Com a mínim n'haureu d'implementar una.
3. **Millores als mètodes de classificació:** Funcions que avaluaran com variant diferents paràmetres es poden obtenir millors o pitjors resultats. Com a mínim haureu d'implementar tantes millores com el número de membres que té el grup.

A continuació us oferim alguns exemples de funcions que podeu implementar per a realitzar els anàlisis i les millores. Recordeu que podeu proposar i implementar altres funcions per cada un dels apartats, les quals seran valuades positivament.

4.1 Funcions d'anàlisi qualitatiu

L'objectiu d'aquest anàlisi és avaluar de manera qualitativa la qualitat de les etiquetes assignades amb els nostres algorismes. Això ho farem a partir de fer cerques a partir de preguntes concretes, i podrem veure com varien els resultats de les cerques en funció dels paràmetres que fem servir en els nostres algorismes a l'hora d'etiquetar les imatges. Així doncs, serà interessant visualitzar com canviarien aquestes cerques.

Retrieval_by_color: Funció que rep com a entrada una llista d'imatges, les etiquetes que hem obtingut en aplicar l'algorisme Kmeans a aquestes imatges i la pregunta que fem per a una cerca concreta (això és, un string o una llista d'strings amb els colors que volem buscar). Retorna totes les imatges que contenen les etiquetes de la pregunta que fem. Aquesta funció pot ser millorada afegint un parametre d'entrada que contingui el percentatge de cada color que conte la imatge, i retorni les imatges ordenades.

Retrieval_by_shape: Funció que rep com a entrada una llista d'imatges, les etiquetes que hem obtingut en aplicar l'algorisme Kmeans a aquestes imatges i la pregunta que fem per a una cerca concreta (això és, un string la forma de roba que volem buscar). Retorna totes les imatges que contenen l'etiqueta de la pregunta que fem. Aquesta funció pot ser millorada afegint un parametre d'entrada que contingui el percentatge de K-neighbors amb l'etiqueta que busquem i retorni les imatges ordenades.

Retrieval_combined: funció que rep com a entrada una llista d'imatges, les etiquetes de forma i les de color, una pregunta de forma i una pregunta de color. Retorna les imatges que coincideixen amb les dos preguntes, per exemple: Red Flip Flops. Com en les funcions anteriors, aquesta funció pot ser millorada introduint les dades de percentatge de color i forma de les etiquetes.

4.2 Funcions d'anàlisi quantitatiu

Kmean_statistics: Funció que rep com a entrada la classe Kmeans amb un conjunt d'imatges i un valor de K que representa la màxima que volem analitzar, Kmax. Per cada valor des de K=2 fins a K=Kmax executarà la funció fit i calcularà la WCD, el nombre d'iteracions/temps que ha necessitat per convergir, etc. Finalment farà una visualització amb aquestes dades.

Get_shape_accuracy: Funció que rep com a entrada les etiquetes que hem obtingut en aplicar el KNN i el *Ground-Truth* d'aquestes. Retorna el percentatge d'etiquetes correctes.

Get_color_accuracy: Funció que rep com a entrada les etiquetes que hem obtingut en aplicar el kmeans i el *Ground-Truth* d'aquestes. Retorna el percentatge d'etiquetes correctes. Cal tenir en compte que per a cada imatge podem tenir més d'una etiqueta, per tant heu de pensar com puntuareu si la predicció i el *Ground-Truth* coincideixen parcialment. A la classe de teoria us varen donar algunes idees per mesurar la similitud entre aquests conjunts.

4.3 Millores als mètodes de classificació

Inicializations de Kmeans: Dins de la funció `init_centroids` de la classe `Kmeans`, heu d'implementar com a mínim dos mètodes d'inicialització diferents al "first".

Diferents heurístiques per BestK: Durant la Part 1 d'aquesta pràctica vam programar l'heurística `WithinClassDistance` per a mesurar quina era la millor K, però hi ha altres heurístiques com la *inter-class distance*, *inter/intra-class variance* o el coeficient de Fisher, els quals podeu implementar. Heu d'implementar com a mínim dos heurístiques diferents.

Find_BestK: Durant la Part 1 d'aquesta pràctica vam decidir que per a trobar la millor K utilitzaríem la fórmula següent, posant un llindar al 20%:

$$\%DEC_k = 100 \frac{WCD_k}{WCD_{k-1}} \quad (1)$$

En aquesta Part 3 us demanem que proveu d'obtenir millors resultats jugant amb el valor del llindar, o utilitzant altres equacions per determinar quina es la millor K.

Features for KNN: Durant la Part 2 d'aquesta pràctica vam utilitzar com a característiques per a classificar les imatges cada un dels píxels d'aquestes amb el seu nivell de gris. Aquest espai de característiques es podria millorar treballant amb els colors dels píxels (RGB), i per fer-ho més eficient reduint el tamany de les imatges, o potser ho podeu millorar creant els vostres propis espais de característiques (per exemple: valor mitjà dels píxels, variança dels píxels, valor dels píxels superiors vs valor dels píxels inferiors).

5 Informe i presentació

Per a l'avaluació d'aquesta última part de la pràctica, haureu d'elaborar un informe i fer una presentació explicant els resultats que heu posat en aquest informe.

L'informe haurà de contenir les següents parts:

1. **Portada:** Ha de contenir el títol, l'assignatura, els noms i els NIUs dels participants.
2. **Introducció a la pràctica:** Explicació breu sobre el problema que la pràctica plantejava, l'algorisme Kmeans, i el KNN.
3. **Mètodes d'anàlisi implementats:** Explicació dels mètodes d'anàlisi que heu implementat per valorar com funciona el vostre Kmeans i KNN. Heu d'explicar com funcionen els mètodes qualitatius i quantitativs que heu programat. Després heu de mostrar els resultats que obteniu sobre el programa base de Kmeans i KNN. Podeu posar totes les imatges, gràfiques o taules que facin falta per explicar tots els experiments ben justificats que heu fet.
4. **Millores sobre Kmeans i KNN:** Per a cada una de les millores o paràmetres que heu provat de modificar:

- (a) Explicació de quin paràmetre esteu modificant.
 - (b) Resultats que obtenim quan modifiqueu aquest paràmetre o quan utilitzeu la millora proposada. (Utilitzant els mètodes d'anàlisi)
 - (c) Conclusió sobre els resultats. Comparació amb els resultats obtinguts sense millora i explicació de perquè hem obtingut un resultat millor/pitjor o semblant.
5. **Conclusió global:** Explicació sobre els coneixements adquirits durant la pràctica, possibles aplicacions del que heu après i opinió personal sobre la pràctica.

La presentació, seguirà la mateixa estructura que el vostre informe i durarà 15 minuts (10 minuts de presentació i 5 minuts de preguntes).

6 Entrega de la Part 3

Per a l'avaluació d'aquesta ultima part, haureu de punar al Moodle un zip amb els arxius següents:

1. Codi (Kmeans.py, KNN.py i My_labeling.py).
2. Informe, en format PDF
3. Presentació, en format PDF o .ppt.