ΕΠ08 Αναγνώριση Προτύπων – Μηχανική Μάθηση

2η Εργασία: Μέθοδοι μείωσης διαστάσεων

Τύπος εργασίας: ***Ατομική***

Ημερομηνία παράδοσης: ***Τετάρτη 19/05/2021, 23:55 (Δεν θα δοθεί παράταση)***

Τρόπος παράδοσης: ***Αποκλειστικά μέσω του eclass***

Σύνολο βαθμών: **100** (20% του τελικού βαθμού του μαθήματος)

Σε αυτή την εργασία καλείστε να υλοποιήσετε σε Python αλγορίθμους μηχανικής μάθησης και να ερμηνεύσετε τα αποτελέσματά τους σε πραγματικά δεδομένα (εικόνες) και συνθετικά δεδομένα.

Η εργασία είναι **ατομική** και αποτελείται από δύο ερωτήματα. Συνιστάται ιδιαίτερα, να αφιερώσετε χρόνο ώστε να κατανοήσετε τον θεμελιώδη λογισμό και τη λογική πίσω από τα ερωτήματα της εργασίας και να αποφύγετε την αναζήτηση έτοιμων λύσεων στο διαδίκτυο. Αν ωστόσο συμβουλευτείτε ή/και χρησιμοποιήσετε οποιοδήποτε υλικό ή/και κώδικα που είναι διαθέσιμος στο διαδίκτυο, πρέπει να αναφέρεται σωστά τη πηγή ή/και το σύνδεσμο στην ιστοσελίδα που αντλήσατε πληροφορίες. Σε κάθε περίπτωση, η αντιγραφή τμήματος ή του συνόλου της εργασίας δεν είναι αποδεκτή και στη περίπτωση που διαπιστωθεί αντιγραφή θα μηδενιστούν στο μάθημα όλα τα εμπλεκόμενα μέρη.

Θα πρέπει να υποβάλετε **ένα μόνο αρχείο Notebook IPython (Jupiter notebook) μέσω του εργαλείου εργασίες του eclass**, ακολουθώντας την εξής σύβαση ονομασίας για το αρχείο σας: *Επώνυμο_ΑριθμόςΜητρώου.ipynb*

Τόσο ο κώδικας Python όσο και οι απαντήσεις σας στις αναλυτικές/αριθμητικές ερωτήσεις πρέπει να είναι ενσωματωμένα στο ίδιο IPython notebook. Οι μαθηματικές πράξεις μπορούν να ενσωματωθούν στο IPython notebook είτε χρησιμοποιώντας LaTeX σημειογραφία είτε ως εικόνες (π.χ. φωτογραφία χειρόγραφου). Μπορείτε να χρησιμοποιήσετε κελιά επικεφαλίδας για να οργανώσετε περαιτέρω το έγγραφό σας. **Σημαντικό:** Το IPython notebook που θα παραδώσετε θα πρέπει βεβαιωθείτε ότι ανοίγει και να εκτελείται στο google colab.

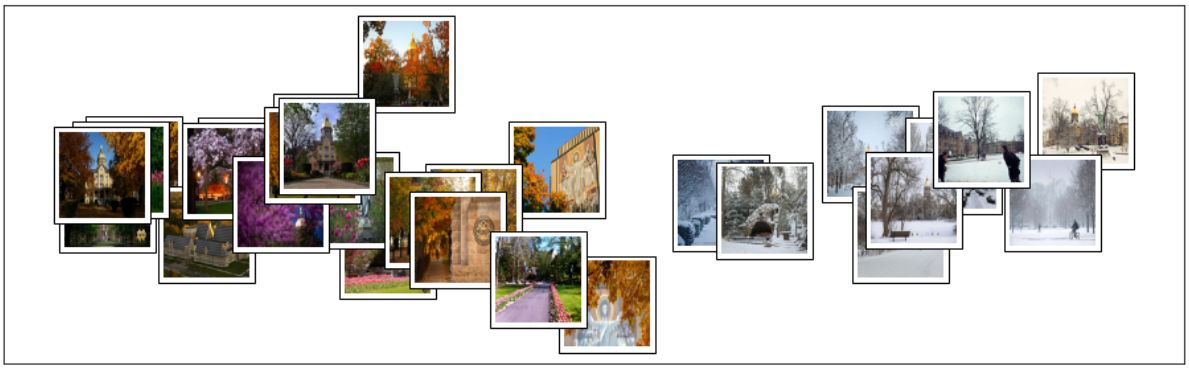
[Ερώτημα 1: Προ-επεξεργασία, μείωση διαστάσεων, και οπτικοποίηση και ταξινόμηση εικόνων] (50 βαθμοί)

Δεδομένα: Το σύνολο δεδομένων αποτελείται από 30 έγχρωμες RGB εικόνες που καταγράφουν τοπία κατά την άνοιξη (spring), το φθινόπωρο (fall) και το χειμώνα (winter) (10 εικόνες για κάθε εποχή). Το πρώτο γράμμα στο όνομα του αρχείου της κάθε εικόνας προσδιορίζει την εποχή κατά την οποία καταγράφηκε η εικόνα, π.χ. η εικόνα F1.jpg καταγράφηκε το φθινόπωρο (fall) ενώ η εικόνα W10.jpg καταγράφηκε το χειμώνα (winter). Συνεπώς η ονοματολογία των αρχείων καθορίζει πλήρως την κατηγορία στην οποία ανήκει κάθε εικόνα. Οι εικόνες αποτελούνται από διαφορετικά σε πλήθος εικονοστοιχεία (pixels). Κάθε pixel αποτελείται από τρεις τιμές χρώματος που κυμαίνονται μεταξύ 0 και 255 και καθορίζουν την ένταση φωτεινότητας του κόκκινου, του πράσινου και του μπλε αντίστοιχα για κάθε στοιχείο της εικόνας (pixel). Τα δεδομένα είναι διαθέσιμα στο αρχείο images.zip στο eclass.

Ζητούμενα:

I. Να γράψετε μία συνάρτηση loadImages(path) η οποία παίρνει ως είσοδο το path στο οποίο βρίσκεται ο φάκελος των εικόνων π.χ. loadImages("C:/images"), διαβάζει τις εικόνες, τις μετατρέπει σε διάσταση 100 x 100 pixels και επιστέφει έναν πίνακα δεδομένων 30 στηλών, όπου κάθε εικόνα αναπαρίσταται ως διάνυσμα στήλη. Η συνάρτηση επιστέφει επίσης τις κατηγορίες (labels) στις οποίες ανήκουν οι διαφορετικές εικόνες κωδικοποιημένες με ακεραίους (π.χ. 0 για φωτογραφίες που καταγράφηκαν το χειμώνα, 1 για τις φωτογραφίες που καταγράφηκαν το φθινόπωρο και 2 για αυτές που καταγράφηκαν την άνοιξη).

II. Να γράψετε μία συνάρτηση PCA_ImageSpaceVisualization(X) η οποία παίρνει ως είσοδο τον πίνακα δεδομένων, υπολογίζει τις δύο πρώτες κύριες συνιστώσες (principal components) των δεδομένων και προβάλει τα δεδομένα στις δύο πρώτες κύριες συνιστώσες. Η συνάρτηση επιστέφει ένα plot στο οποίο εμφανίζονται οι εικόνες στο δυσδιάστατο χώρο που προκύπτει από τη προβολή των δεδομένων στις δύο πρώτες κύριες συνιστώσες. Το plot αναμένεται να είναι της μορφής:



III. Τι σημαίνει όταν εικόνες βρίσκονται κοντά σε αυτό το χώρο δύο διαστάσεων που απεικονίζεται στο παραπάνω plot; Τι σημαίνει όταν εικόνες απέχουν πολύ; Μπορούμε να γενικεύσουμε αυτά τα συμπεράσματα για τον αρχικό χώρο των εικόνων ο οποίος είναι πολύ μεγάλης διάστασης;

IV. Οι εικόνες που αντιστοιχούν σε μία από τις εποχές τείνουν να ομαδοποιούνται πιο κοντά από ότι οι υπόλοιπες; Γιατί συμβαίνει αυτό;

V. Να συγκρίνετε την ακρίβεια ταξινόμησης (classification accuracy) του ταξινομητή πλησιέστερου γείτονα (1-NN) και του ταξινομητή τριών πλησιέστερων γειτόνων (3-NN) στο πρόβλημα της αναγνώρισης της εποχής κατά την οποία καταγράφηκε μια εικόνα. Με άλλα λόγια να συγκρίνετε την επίδοση (ως προς την ακρίβεια ταξινόμησης) των παραπάνω ταξινομητών στην ταξινόμηση των δεδομένων εικόνων στις κατηγορίες χειμώνας, άνοιξη και φθινόπωρο.

Καλείστε να αντιμετωπίσετε το πρόβλημα ταξινόμησης χρησιμοποιώντας 1) τις αρχικές μεγάλης διάστασης εικόνες σε μορφή διανύσματος και 2) χαρακτηριστικά χαμηλής διάστασης που θα εξάγετε μέσω της PCA.

V.1 Χρησιμοποιείστε 5-fold cross validation και αναφέρετε τη μέση ακρίβεια ταξινόμησης για τους δύο ταξινομητές τόσο για τα αρχικά δεδομένα μεγάλης διάστασης όσο και για τα χαρακτηριστικά χαμηλής διάστασης που εξάγετε με τη χρήση PCA .
V.2 Πώς θα προσδιορίσετε τη διάσταση των χαρακτηριστικών που θα εξάγετε μέσω της PCA;

Σημείωση: *Για το ερώτημα V μπορείτε να χρησιμοποιήσετε υλοποιήσεις του ταξινομητή πλησιέστερου γείτονα και της διαδικασία cross validation από τη βιβλιοθήκη scikit learn.*

[Ερώτημα 2 - Κανονικοποιημένη μη-αρνητική παραγοντοποίηση πινάκων] (50 βαθμοί)

Έστω το παρακάτω πρόβλημα βελτιστοποίησης για την κανονικοποιημένη μη-αρνητική παραγοντοποίηση πινάκων (regularized non-negative matrix factorization -regNMF):

$$\min_{\mathbf{A},\mathbf{B}} \|\mathbf{X}-\mathbf{AB}\|_F^2 + \lambda\|\mathbf{A}\|_F^2 \text{ s.t. } \mathbf{A}\geq\mathbf{0}, \mathbf{B}\geq\mathbf{0}$$

Το πρόβλημα προφανώς δεν έχει λύση σε κλειστή μορφή και συνεπώς πρέπει να λυθεί επαναληπτικά.

Για το σκοπό αυτό,

I. Να εξάγετε αναλυτικά βήμα προς βήμα με το χέρι, τους πολλαπλασιαστικούς κανόνες ανανέωσης (multiplicative update rules) των μεταβλητών **A** και **B**, παραθέτοντας όλα τα ενδιάμεσα βήματα.

II. Να υλοποιήσετε στη συνάρτηση RegNMF(X,k,lambda,epsilon) έναν επαναληπτικό αλγόριθμο για την επίλυση του παραπάνω προβλήματος βελτιστοποίησης χρησιμοποιώντας τους πολλαπλασιαστικούς κανόνες ανανέωσης του βρήκατε. Η συνάρτηση παίρνει ως είσοδο έναν μη-αρνητικό πίνακα **X** διαστάσεων d x N, το πλήθος των συνιστωσών k, και τη τιμή της παραμέτρου κανονικοποίησης λ (lambda) και το κατώφλι τερματισμού ε (epsilon) και επιστέφει τους μη αρνητικούς πίνακες **A** διάστασης d x k και **B** διάστασης k x N. Για να διαπιστώσουμε εάν συγκλίνει σε λύση ένας επαναληπτικός αλγόριθμος, συνήθως, παρακολουθούμε το σφάλμα ανακατασκευής

$\frac{\|\mathbf{X}-\mathbf{A}[t]\mathbf{B}[t]\|_F^2}{\|\mathbf{X}\|_F^2}$ σε κάθε επανάληψη και εάν η μεταβολή του ανάμεσα σε δύο διαδοχικές επαναλήψεις είναι μικρότερη από ένα κατώφλι ε, δηλαδή εάν

$$\frac{\|\mathbf{X}-\mathbf{A}[t-1]\mathbf{B}[t-1]\|_F^2 - \|\mathbf{X}-\mathbf{A}[t]\mathbf{B}[t]\|_F^2}{\|\mathbf{X}\|_F^2} \leq \epsilon$$

τερματίζουμε τον αλγόριθμο και λέμε ότι αλγόριθμος έχει συγκλίνει σε λύση. Το t στη παραπάνω σχέση συμβολίζει το δείκτη επανάληψης.

Καλείστε να μελετήσετε την σύγκλιση του αλγορίθμου χρησιμοποιώντας συνθετικά δεδομένα. Για το σκοπό αυτό, να κατασκευάστε ένα πίνακα **X** διάστασης 500 x 500

με μη αρνητικές τιμές ως γινόμενο δύο τυχαίων πινάκων **X** = **WH**, με τον **W** να έχει διαστάσεις 500 x 10 και τον **H** να έχει διαστάσεις 10 x 500. Για τη δημιουργία των τυχαίων πινάκων **W** και **H** μπορείτε να χρησιμοποιήσετε τη συνάρτηση rand της numpy. Για να βεβαιωθείτε ότι τα στοιχεία του πίνακα είναι μη αρνητικά μπορείτε να εφαρμόσετε ένα τελεστή απόλυτης τιμής σε κάθε στοιχείο του τυχαίου πίνακα που παρήγαγε η συνάρτηση rand. Χρησιμοποιώντας σταθερή τιμή για την παράμετρο λ=0.5 να μελετήστε τη συμπεριφορά του αλγορίθμου regNMF ως προς το πλήθος των επαναλήψεων που απαιτούνται για να συγκλίνει εάν k= 5, 10, 50 και ε = 0.1, 0.01, 0.001. Ποια είναι τα συμπεράσματά σας ως προς τη συμπεριφορά του αλγορίθμου για διαφορετικές τιμές του k και epsilon (ε);