

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ



ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ ΓΙΑ ΑΛΓΟΡΙΘΜΙΚΑ ΠΡΟΒΛΗΜΑΤΑ

3η Εργασία:

Διανυσματική αναπαράσταση εικόνας σε χώρο χαμηλότερης διάστασης με χρήση νευρωνικού δικτύου αυτοκωδικοποίησης. Αναζήτηση και συσταδοποίηση των εικόνων στον νέο χώρο και σύγκριση με προσεγγιστική και εξαντλητική αναζήτηση και συσταδοποίηση στον αρχικό χώρο

ΒΕΚΡΑΚΗ ΑΘΗΝΑ

1115201400020

ΠΟΛΥΧΡΟΝΑΚΗΣ ΣΑΒΒΑΣ

1115201200150

ΠΕΡΙΕΧΟΜΕΝΑ

Python files :

- [reduce.py](#) : Μετατρέπει τα datasets στη νέα διάσταση
- [autoencoder.py](#) : Υλοποίηση του autoencoder μαζί με τα στρώματα συμπίεσης και αποσυμπίεσης
- [autoencoder_functions.py](#) : Συναρτήσεις για τον autoencoder
- [classification.py](#) : Κατηγοριοποίηση εικόνων σύμφωνα με τα labels τους
- [classification_functions.py](#) : Συναρτήσεις για το classification
- [search.py](#) : Υλοποίηση της μετρικής Earth Mover's Distance (EMD)

C++ files :

- [search.cpp](#) : Υπολογίζει και συγκρίνει Brute Force και LSH στον αρχικό και στο νέο χώρο
- [cluster.cpp](#) : Συσταδοποίηση στον αρχικό και στο νέο χώρο. Σύγκριση των silhouette και των objective functions
- Αρχεία με βοηθητικές συναρτήσεις:

.cpp		.h
help_functions.cpp		help_functions.h
calculations.cpp		calculations.h
calculations_lsh.cpp		calculations_lsh.h
calculations_cluster.cpp		calculations_cluster.h

Αρχεία αποτελεσμάτων :

- [classification_results](#) : Αποτελέσματα του classification στο νέο χώρο
- [results_search.txt](#) : Αποτελέσματα Brute Force και LSH για το Β ερώτημα
- [results_cluster.txt](#) : Αποτελέσματα Clustering για το Δ ερώτημα
- [results_emd.txt](#) : Αποτελέσματα Earth Mover's Distance για το Γ ερώτημα

Datasets : [train-images-idx3-ubyte](#), [train-labels-idx1-ubyte](#), [t10k-images-idx3-ubyte](#), [t10k-labels-idx1-ubyte](#)

Datasets στο νέο χώρο : [train-images-idx1-ushort](#), [t10k-images-idx1-ushort](#)

Αρχείο [cluster.conf](#), το μοντέλο [reduce_model.h5](#) που παράγει ο autoencoder και [Makefile](#).

ΓΕΝΙΚΑ

Α ερώτημα : Το `autoencoder.py` παράγει το μοντέλο `reduce_model.h5` μαζί με τα στρώματα συμπίεσης και αποσυμπίεσης (“bottleneck”) , το οποίο στη συνέχεια χρησιμοποιείται από το `reduce.py` για να παράγει τα datasets στη νέα διάσταση.

Β ερώτημα : Εύρεση του κοντινότερου γείτονα των εικόνων του συνόλου αναζήτησης στον νέο διανυσματικό χώρο με τη μέθοδο Brute Force και στον αρχικό χώρο με Brute Force και LSH. Σύγκριση των αποτελεσμάτων ως προς τον χρόνο αναζήτησης και το κλάσμα προσέγγισης στον αρχικό χώρο.

Γ ερώτημα : Σπάσιμο της κάθε εικόνας σε 4x4 clusters μεγέθους 7x7 και εύρεση του signature που αποτελείται από το κεντροειδές και το βάρος του κάθε cluster. Εύρεση των 10 κοντινότερων γειτόνων κάθε εικόνας του query file με EMD και Manhattan και σύγκριση της ορθότητας των αλγορίθμων.

Δ ερώτημα : Συσταδοποίηση των εικόνων στο νέο και στον αρχικό χώρο. Σύγκριση των αποτελεσμάτων ως προς το silhouette και το objective function αυτών, καθώς και με το αποτέλεσμα του `classification.py`.

ΟΔΗΓΙΕΣ ΧΡΗΣΗΣ

- Για το A :

`$python3 autoencoder.py -d <dataset>` , όπου:

`-d` : η επόμενη παράμετρος είναι το dataset

Αν ο χρήστης δε δώσει παραμέτρους, τότε χρησιμοποιούνται default τιμές.

Ως default τιμή ορίζεται:

`<dataset>` : train-images-idx3-ubyte

Στο τέλος κάθε πειράματος:

Επιλογή 1: Επανάληψη του πειράματος με διαφορετικές τιμές υπερπαραμέτρων.

Επιλογή 2: Γραφικές παραστάσεις

Επιλογή 0: Ως προς το πλήθος των layers

Επιλογή 1: Ως προς το μέγεθος των φίλτρων

Επιλογή 2: Ως προς τον αριθμό φίλτρων στο πρώτο στρώμα

Επιλογή 3: Ως προς τον αριθμό των εποχών

Επιλογή 4: Ως προς το batch size

Επιλογή 5: Ως προς το latent dimension

Επιλογή 3: Αποθήκευση μοντέλου με filename που ζητάει από το χρήστη

\$python3 reduce.py -d <dataset> -q <queryset> -od <output_dataset_file>
-oq <output_query_file> , όπου:

-d	:	η επόμενη παράμετρος είναι το dataset στον αρχικό χώρο
-q	:	η επόμενη παράμετρος είναι το query file στον αρχικό χώρο
-od	:	η επόμενη παράμετρος είναι το output του dataset στο νέο χώρο
-oq	:	η επόμενη παράμετρος είναι το output του query file στο νέο χώρο

Αν ο χρήστης δε δώσει παραμέτρους, τότε χρησιμοποιούνται default τιμές.

Ως default τιμή ορίζεται:

<dataset>	:	train-images-idx3-ubyte
<queryset>	:	t10k-images-idx3-ubyte
<output_dataset_file>	:	train-images-idx1-ushort
<output_query_file>	:	t10k-images-idx1-ushort

- Για το B :

\$./search -d <input file original space> -i <input file new space> -q <query file original space> -s <query file new space> -k <int> -L <int> -o <output file>

όπου :	-d	:	η επόμενη παράμετρος είναι το dataset στον αρχικό χώρο
	-i	:	η επόμενη παράμετρος είναι το dataset στο νέο χώρο
	-q	:	η επόμενη παράμετρος είναι το query file στον αρχικό χώρο
	-s	:	η επόμενη παράμετρος είναι query file στο νέο χώρο
	-k	:	η επόμενη παράμετρος είναι το πλήθος των συναρτήσεων h για το LSH
	-L	:	η επόμενη παράμετρος είναι ο αριθμός των HashTables
	-o	:	η επόμενη παράμετρος είναι το output file

Αν ο χρήστης δε δώσει παραμέτρους, τότε χρησιμοποιούνται default τιμές.

Ως default τιμές ορίζονται :

<input file original space>	:	train-images-idx3-ubyte
<input file new space>	:	train-images-idx1-ushort
<query file original space>	:	t10k-images-idx3-ubyte
<query file new space>	:	t10k-images-idx1-ushort
k	:	4
L	:	3
<output file>	:	results_search.txt

- Για το Γ :

\$python3 search.py -d <input file original space> -q <query file original space> -l1 <labels of input dataset> -l2 <labels of query dataset> -o <output file> , όπου:

-d	:	η επόμενη παράμετρος είναι το dataset στον αρχικό χώρο
-q	:	η επόμενη παράμετρος είναι το query file στον αρχικό χώρο
-l1	:	η επόμενη παράμετρος είναι τα labels του input
-l2	:	η επόμενη παράμετρος είναι τα labels του query
-o	:	η επόμενη παράμετρος είναι το output file

Αν ο χρήστης δε δώσει παραμέτρους, τότε χρησιμοποιούνται default τιμές.

Ως default τιμές ορίζονται :

<input file original space>	:	train-images-idx3-ubyte
<query file original space>	:	t10k-images-idx3-ubyte
<labels of input dataset>	:	train-labels-idx1-ubyte
<labels of query dataset>	:	t10k-labels-idx1-ubyte
<output file>	:	results_emd.txt

- Για το Δ:

```
$ ./cluster -d <input file original space> -i <input file new space>  
-n <classes from NN as clusters file> -c <configuration file> -o <output file>
```

όπου :	-d	:	η επόμενη παράμετρος είναι το dataset στον αρχικό χώρο
	-i	:	η επόμενη παράμετρος είναι το dataset στο νέο χώρο
	-n	:	η επόμενη παράμετρος είναι το dataset που προκύπτει από το classification στο νέο χώρο
	-c	:	η επόμενη παράμετρος είναι το αρχείο ρύθμισης παραμέτρων
	-o	:	η επόμενη παράμετρος είναι το output file

Αν ο χρήστης δε δώσει παραμέτρους, τότε χρησιμοποιούνται default τιμές.

Ως default τιμές ορίζονται :

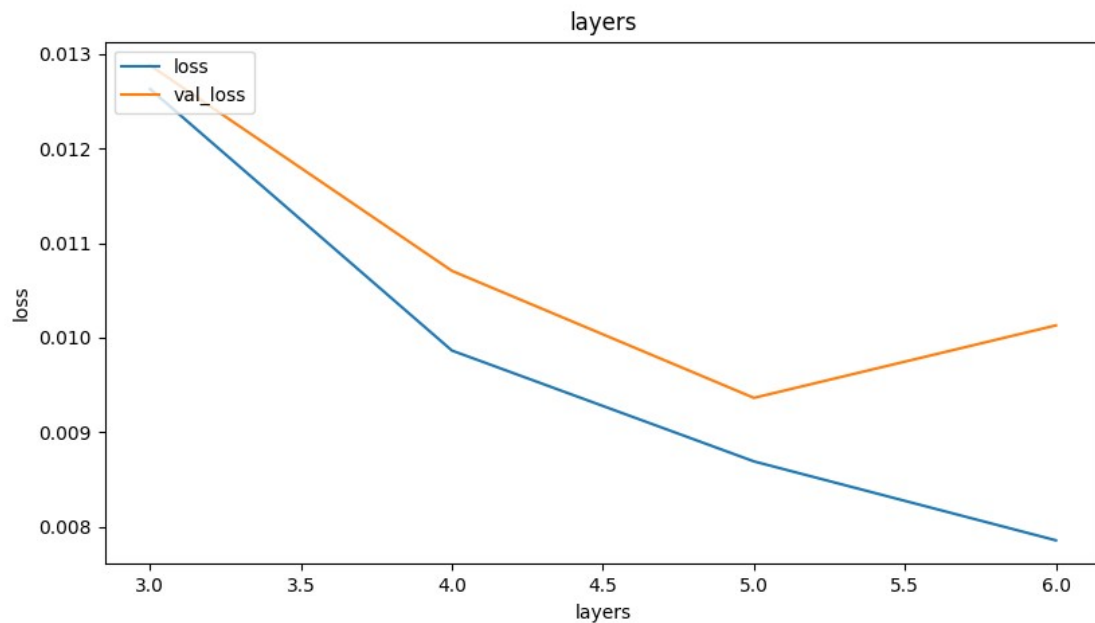
<input file original space>	:	train-images-idx3-ubyte
<input file new space>	:	train-images-idx1-ushort
<classes from NN as clusters file>	:	classification_results
<configuration file>	:	cluster.conf
<output file>	:	results_cluster.txt

ΔΟΚΙΜΕΣ – ΠΕΙΡΑΜΑΤΑ

AUTOENCODER με στρώματα συμπίεσης και αποσυμπίεσης (“bottleneck”)

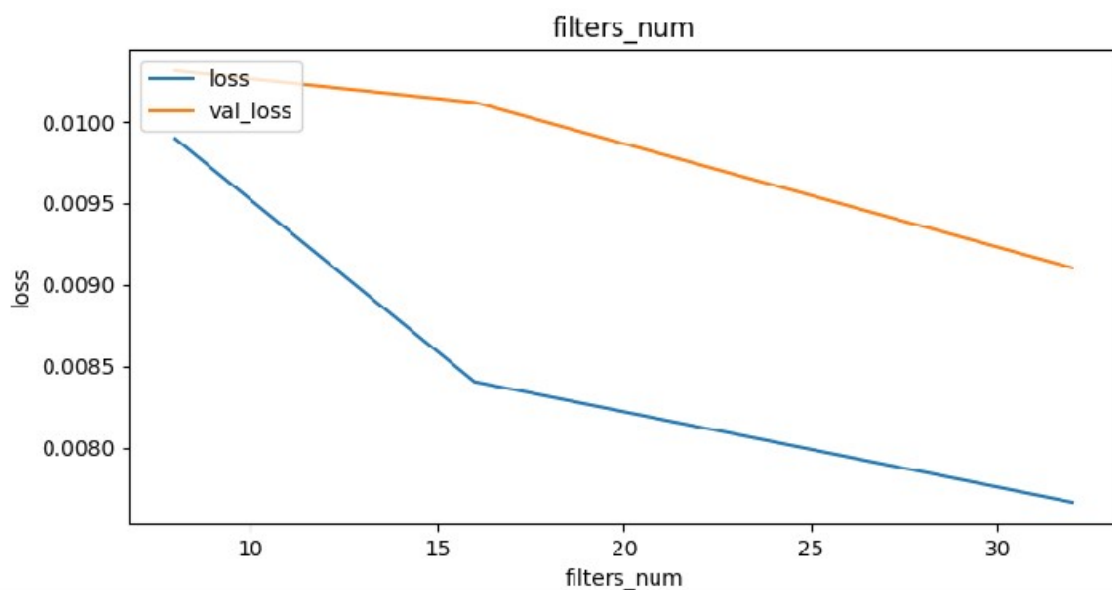
Δοκιμές για αριθμό layers:

(κρατώντας σταθερές τιμές epochs = 20, batch_size = 64, filters_size = 3, filters_num = 16, latent_dim = 10)



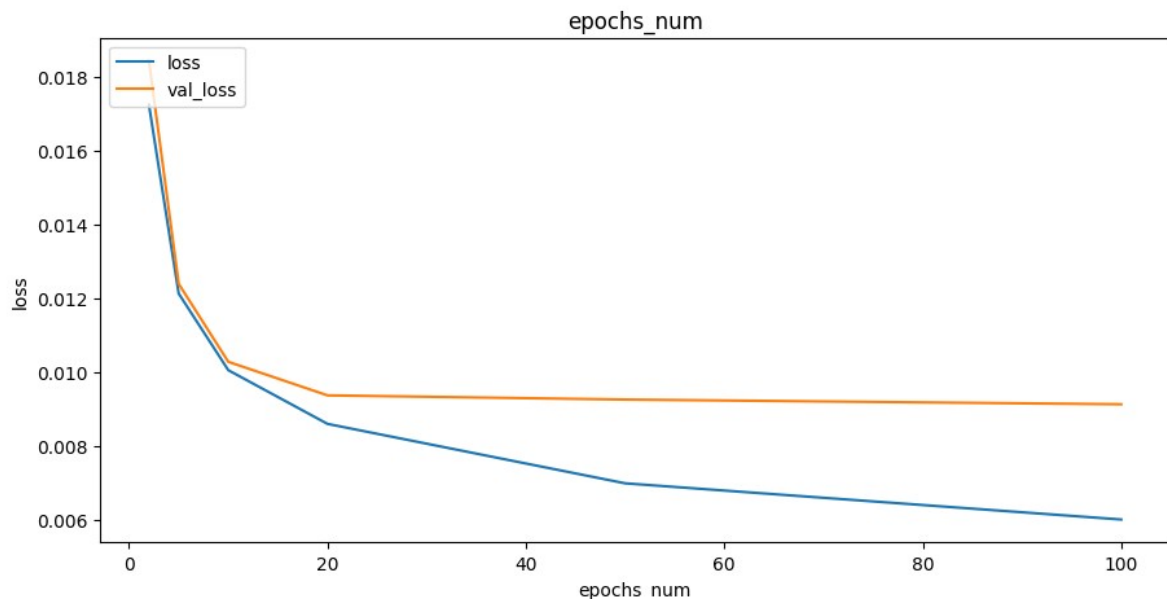
Δοκιμές για αριθμό φίλτρων στο πρώτο layer:

(κρατώντας σταθερές τιμές layers = 5, epochs = 20, batch_size = 64, filters_size = 3, latent_dim = 10)



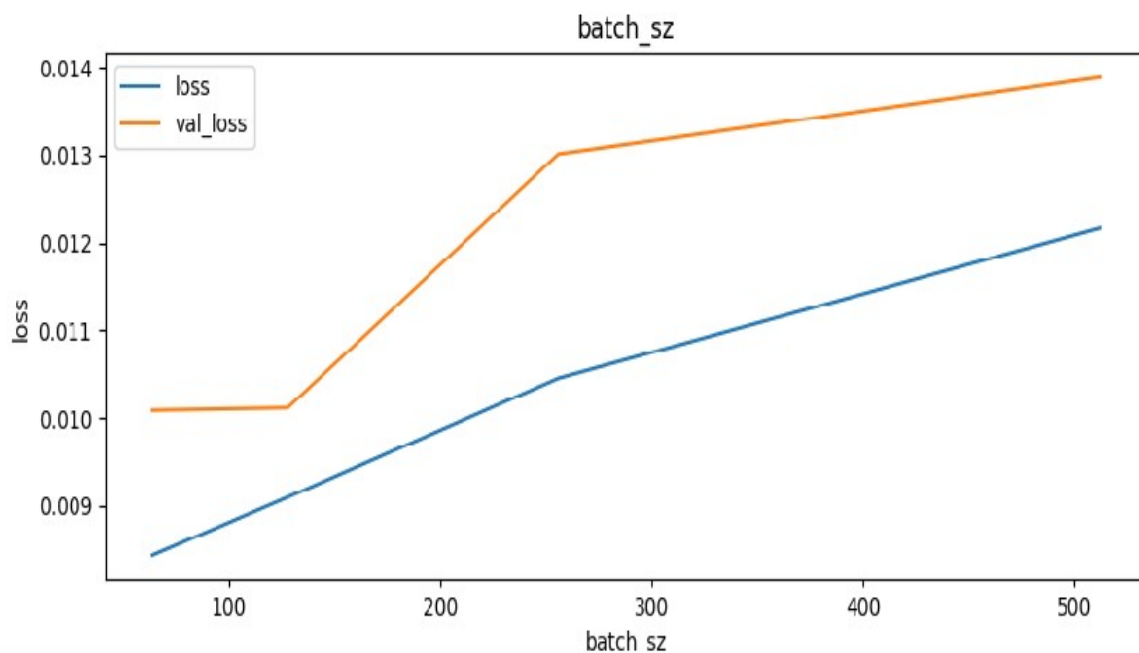
Δοκιμές για αριθμό epochs:

(κρατώντας σταθερές τιμές layers = 5, batch_size = 64, filters_size = 3, filters_num = 16, latent_dim = 10)



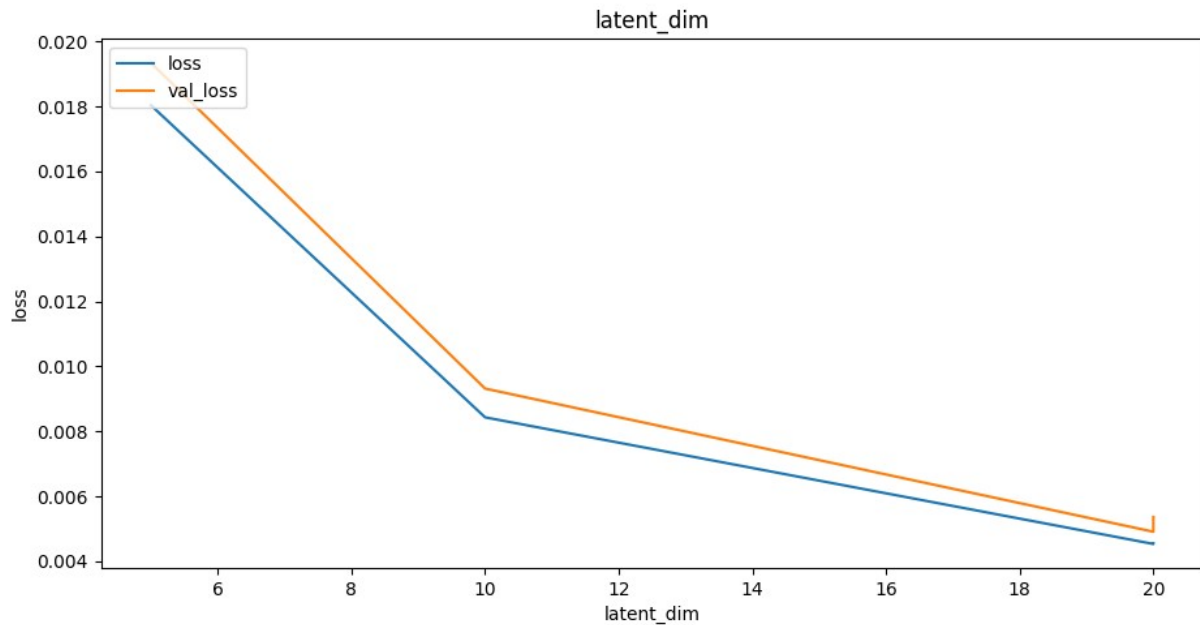
Δοκιμές για batch size:

(κρατώντας σταθερές τιμές layers = 5, epochs = 20, filters_size = 3, filters_num = 16, latent_dim = 10)



Δοκιμές για latent_dim :

(κρατώντας σταθερές τιμές layers = 5, epochs = 20, batch_size = 64, filters_size = 3, filters_num = 16)



ΕΠΙΛΟΓΕΣ ΥΛΟΠΟΙΗΣΗΣ

Το Γ ερώτημα υλοποιήθηκε εξ'ολοκλήρου σε Python και για την υλοποίησή του επιλέχθηκε η εικόνα να σπάει σε 4x4 Clusters μεγέθους 7x7. Προτιμήθηκε η χρήση της βιβλιοθήκης PULP για την επίλυση του προβλήματος γραμμικού προγραμματισμού. Για τα αποτελέσματα, λόγω καθυστέρησης του προγράμματος, χρησιμοποιήθηκαν οι πρώτες 10000 εικόνες του input file και οι πρώτες 100 του query file.

ΠΗΓΕΣ

- <https://www.coin-or.org/PuLP/pulp.html>
- <https://wihoho.github.io/2013/08/18/EMD-Python.html>
- <https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/rubner-jcviu-00.pdf>
- https://en.wikipedia.org/wiki/Earth_mover%27s_distance