ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΙΟΝΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

Ιόνιο Πανεπιστήμιο

Τμήμα Πληροφορικής

Διαδραστικό Pipeline Ανάλυσης σε δεδομένα Μοριακής Βιολογίας

Συγγραφείς:
Χρήστος Σπυρίδων Καρύδης - inf2022076
Ευαγγελία Φώτη - inf2022224
Ευτυχία Φίλιου - Π2019202

Υπεύθυνος Καθηγητής:
Αριστείδης Βραχάτης

Η εργασία κατατέθηκε για το μάθημα:
Τεχνολογίες Λογισμικού 2025

Μάιος 2025

Περιεχόμενα

Περίληψη

Η παρούσα εργασία παρουσιάζει μια διαδραστική εφαρμογή ανάλυσης δεδομένων μοριακής βιολογίας (scRNA-seq), υλοποιημένη σε περιβάλλον Streamlit. Η εφαρμογή επιτρέπει την παραμετροποιημένη ανάλυση με ενσωματωμένες τεχνικές προεπεξεργασίας, PCA, UMAP, clustering, διόρθωση batch effect με Harmony, καθώς και ανάλυση διαφορικής γονιδιακής έκφρασης. Οι χρήστες μπορούν εύκολα να ανεβάζουν αρχεία τύπου '.h5ad', να ρυθμίζουν τις παραμέτρους, να εκτελούν ανάλυση και να εξάγουν αποτελέσματα και οπτικοποιήσεις. Η εφαρμογή συνοδεύεται από δυνατότητα dockerization για εύκολη μεταφορά και αναπαραγωγή σε κάθε σύστημα και από τα UML διαγράμματα (Class Diagram & Use Case Diagram) που περιγράφουν την αρχιτεκτονική και τη λειτουργικότητα της εφαρμογής.

Κεφάλαιο 1

Εισαγωγή

Η ανάλυση δεδομένων μονοκυτταρικής ακολουθίας RNA (single-cell RNA sequencing / scRNA-seq) αποτελεί σημαντική πρόοδο στη βιοπληροφορική, επιτρέποντας τη μελέτη της έκφρασης γονιδίων σε μεμονωμένα κύτταρα. Η ποικιλομορφία των κυτταρικών τύπων και η ανάγκη ανακάλυψης νέων κυτταρικών υποπληθυσμών απαιτεί την ύπαρξη διαδραστικών εργαλείων, τα οποία επιτρέπουν οπτικοποίηση και επεξεργασία των δεδομένων με εύχρηστο τρόπο. Στο πλαίσιο του μαθήματος Τεχνολογίες Λογισμικού (2025), αναπτύχθηκε μια εφαρμογή βασισμένη σε Streamlit, η οποία συνδυάζει τα στάδια ανάλυσης scRNA-seq σε ενιαίο γραφικό περιβάλλον.

Κεφάλαιο 2

Σχεδιασμός της Υλοποίησης

Η εφαρμογή οργανώνεται σε έξι βασικά tabs, τα οποία αντιστοιχούν στα στάδια της ανάλυσης:

- **Δεδομένα:** Ανέβασμα αρχείων τύπου '.h5ad' και προεπισκόπηση μεταδεδομένων και γονιδίων.
- **Προεπεξεργασία:** Φιλτράρισμα κυττάρων/γονιδίων, αφαίρεση MT-, ERCC γονιδίων, κανονικοποίηση, log1p, HVG επιλογή και scaling.
- **Ανάλυση:** PCA, clustering με Leiden, UMAP (2D/3D), και επιλογή χρήσης ή μη Harmony.
- **Γονιδιακή Ανάλυση:** Ανάληση marker genes και DEG (Differential Expression) με διάφορες οπτικοποιήσεις.
- **Εξαγωγή:** Κατεβάσματα preprocessed αρχείων, DEGs σε CSV/XLSX, Volcano plots, Heatmap, Dotplot, Violin και UMAP εικόνες.
- **Ομάδα:** Παρουσίαση μελών ομάδας και των ρόλων τους.

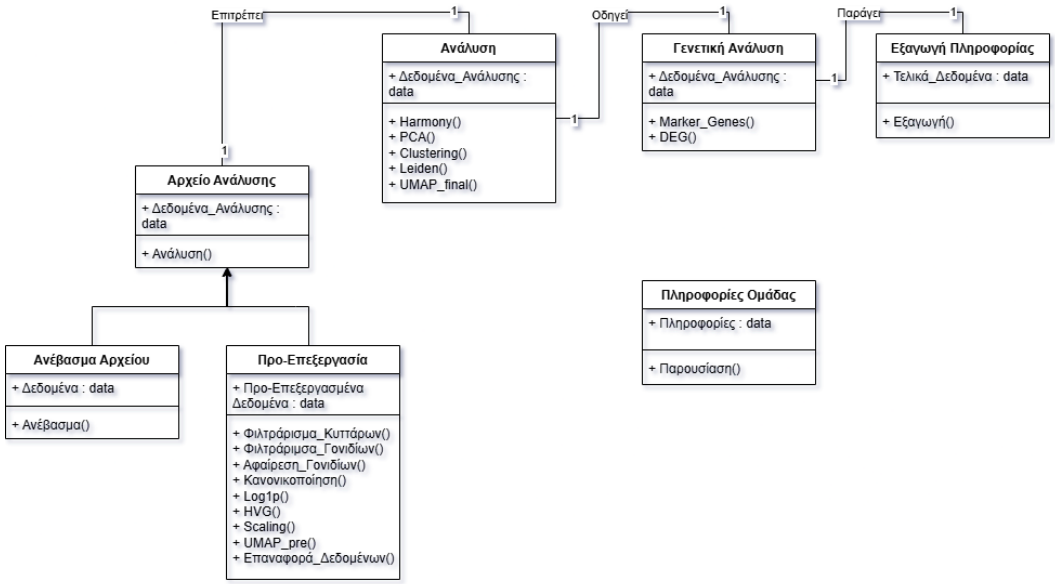
Η εφαρμογή βασίζεται στο Streamlit και αξιοποιεί 'session state' για μεταφορά δεδομένων μεταξύ των tabs, ενώ κάθε βήμα ελέγχεται με παραμετρικά sliders και επιλογές που ενημερώνουν δυναμικά την επόμενη ενέργεια.

Κεφάλαιο 3

UML Διαγράμματα

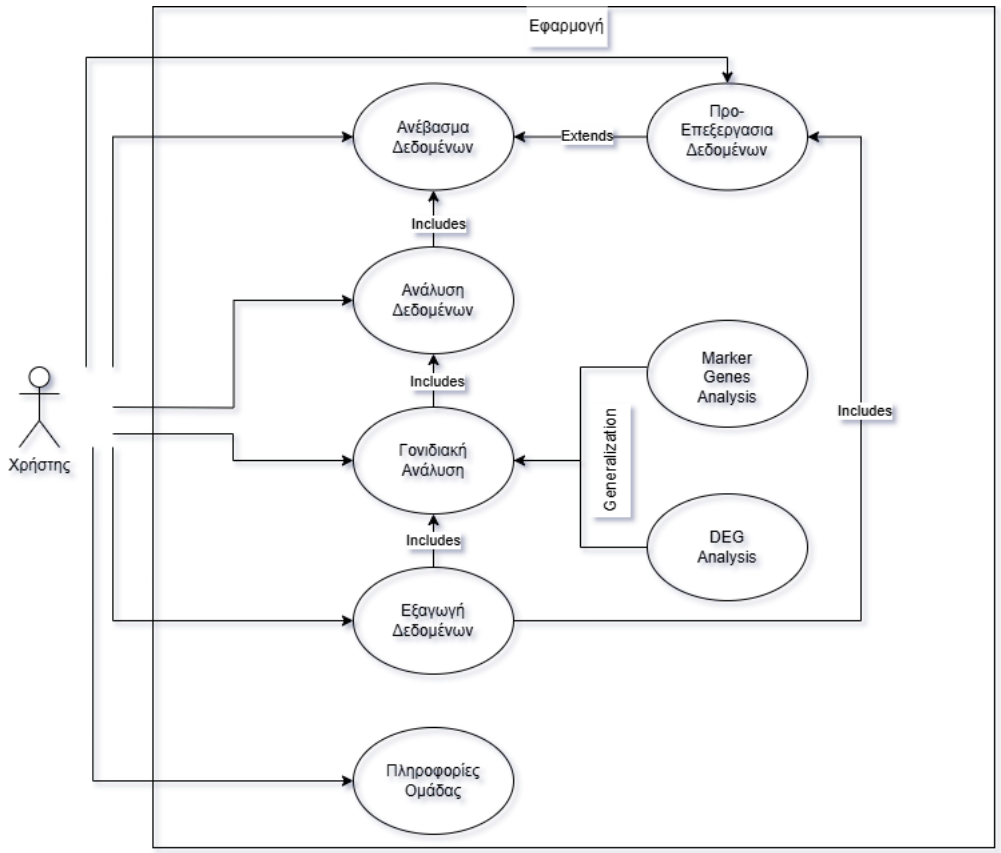
Στην παρούσα ενότητα παρουσιάζονται δύο βασικά διαγράμματα UML για την τεκμηρίωση της λειτουργικότητας και της αρχιτεκτονικής της εφαρμογής.

Το **Class Diagram** απεικονίζει τις κύριες κλάσεις και τις μεταξύ τους σχέσεις, με έμφαση στη ροή δεδομένων και τις κύριες μεθόδους.



Σχήμα 3.1: UML Class Diagram της εφαρμογής

Το **Use Case Diagram** περιγράφει τις ενέργειες που εκτελεί ο χρήστης μέσα στην εφαρμογή, καθώς και τις σχέσεις μεταξύ των διεργασιών (use cases) που υλοποιούνται μέσω των διαφορετικών tabs.



Σχήμα 3.2: Use Case Diagram της εφαρμογής

Κεφάλαιο 4

Ανάλυση της Υλοποίησης με Τεχνικές Λεπτομέρειες

Η εφαρμογή έχει υλοποιηθεί σε Python με χρήση των βιβλιοθηκών Streamlit, Scanpy, anndata, Plotly, Pandas, Matplotlib, NumPy, SciPy, seaborn και HarmonyPy. Όλη η ροή της εφαρμογής οργανώνεται στο αρχείο `main.py`, το οποίο διαχωρίζει τη λειτουργικότητα σε έξι διαδραστικά tabs.

0. Ανέβασμα Αρχείου

Ο χρήστης μπορεί να φορτώσει ένα αρχείο τύπου `.h5ad` μέσω του `file_uploader` του Streamlit. Εφόσον το αρχείο φορτωθεί επιτυχώς, γίνεται άμεση ανάγνωση με τη συνάρτηση `scanpy.read_h5ad()` και το αντικείμενο AnnData αποθηκεύεται σε `st.session_state`.

Παράλληλα εμφανίζεται συνοπτική προεπισκόπηση μετρικών (αρ. κυττάρων, αρ. γονιδίων), ενώ ο χρήστης έχει δυνατότητα να περιηγηθεί στις εγγραφές του `adata.obs` και `adata.var` ανά σελίδες.

1. Προεπεξεργασία

Περιλαμβάνει τον καθαρισμό των δεδομένων και τις παρακάτω λειτουργίες:

- `filter_cells`, `filter_genes` για φιλτράρισμα βάση τιμής που δίνεται από τον χρήστη.
- Αφαίρεση γονιδίων `MT-`, `ERCC` με χρήση προθέματος.
- Κανονικοποίηση με `normalize_total` και μετασχηματισμός `log1p`.
- Επιλογή γονιδίων υψηλής διακύμανσης με `highly_variable_genes`.
- Κανονικοποίηση τιμών μέσω `scale`.

2. PCA, Clustering, UMAP, Harmony

Ανάλυση κύριων συνιστωσών (clusters) και ομαδοποίηση:

- Υπολογισμός `PCA` με δυνατότητα επιλογής αριθμού συνιστωσών (clusters).
- Δημιουργία γράφου γειτνίασης και clustering με τον αλγόριθμο Leiden.
- Χρήση UMAP για 2D ή 3D προβολή.
- Προαιρετικά: διόρθωση batch effect με `harmony_integrate` από το scanpy.external.

3. Marker Genes

Ανίχνευση γονιδίων που διαφοροποιούν τα clusters:

- Μέθοδοι: `logreg`, `wilcoxon`, `t-test` μέσω `rank_genes_groups`.
- Οπτικοποιήσεις με dotplot, heatmap και violin.

4. DEG (Differential Expression)

Ανάλυση μεταξύ δύο ομάδων:

- Επιλογή ομάδας ενδιαφέροντος και ομάδας αναφοράς (group vs. reference).
- Χρήση της Wilcoxon για εξαγωγή διαφορικά εκφραζόμενων γονιδίων.
- Δημιουργία Volcano plot με χρωματική ένδειξη για UP, DOWN και NS (not significant) γονίδια.

5. Εξαγωγή Αποτελεσμάτων

Ο χρήστης μπορεί να:

- Κατεβάσει preprocessed δεδομένα σε .h5ad.
- Εξαγάγει πίνακες DEGs σε CSV/XLSX.
- Κατεβάσει εικόνες UMAP, Volcano, Heatmap, Dotplot, Violin.

Όλες οι ενέργειες και τα intermediate data αποθηκεύονται σε `st.session_state`, εξασφαλίζοντας διατήρηση κατάστασης μεταξύ των tabs και της αλληλεπίδρασης του χρήστη.

Κεφάλαιο 5

Οπτικοποιήσεις και Αποτελέσματα

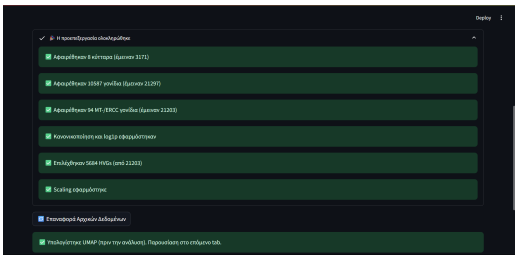
Η εφαρμογή προσφέρει δυναμικές οπτικοποιήσεις για όλα τα στάδια της ανάλυσης:

- UMAP πριν και μετά την ανάλυση, με χρωματισμούς κατά 'batch' ή 'celltype'.
- 3D προβολές των clusters με δυνατότητα περιστροφής.
- Αποτελέσματα προεπεξεργασίας σε μορφή ενημερωτικών καρτών.
- Volcano plot με logFC και p-value για DEG.
- Dotplot/Violin/Heatmap για marker genes.

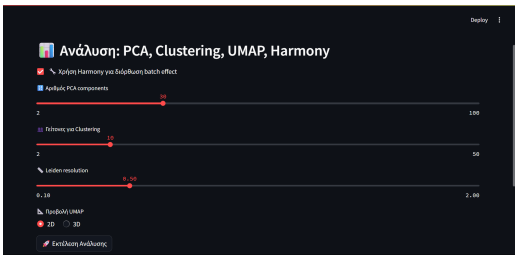
Όλες οι εικόνες είναι εξαγώγιμες ως PNG από την ίδια την εφαρμογή.

Ενδεικτικές Οπτικοποιήσεις

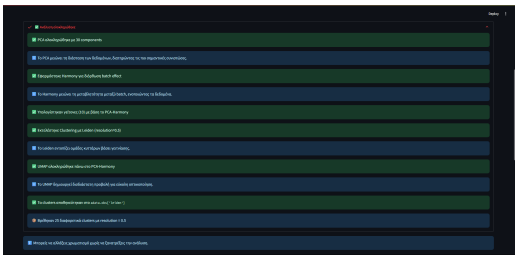
Οθόνη Επιλογών Προεπεξεργασίας



Αποτελέσματα Προεπεξεργασίας

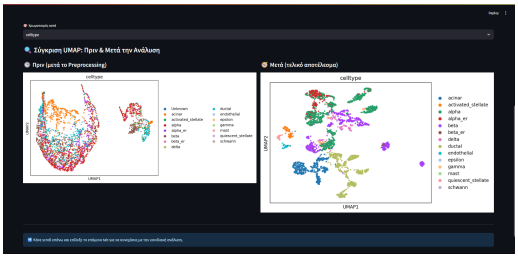


Παράμετροι Ανάλυσης



Αποτελέσματα PCA & Harmony

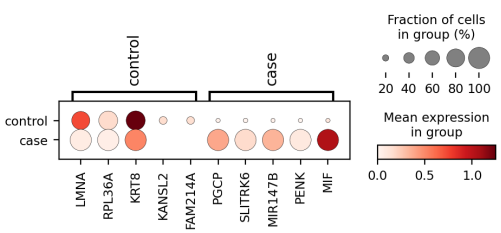
UMAP 2D πριν & μετά - Batch



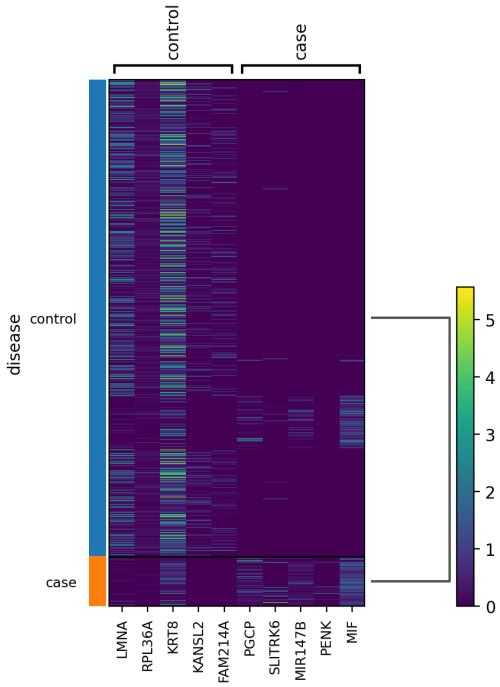
UMAP 2D πριν & μετά - Celltype

UMAP 3D πριν & μετά - Batch

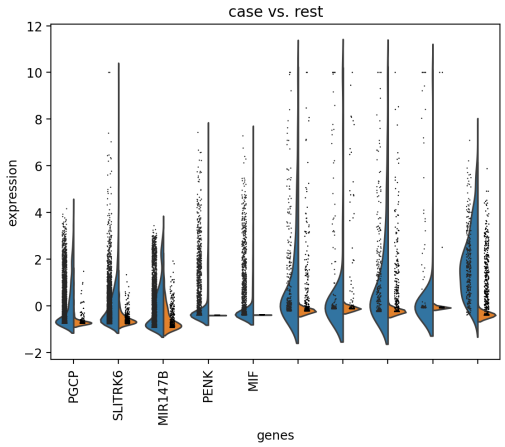
UMAP 3D πριν & μετά - Celltype



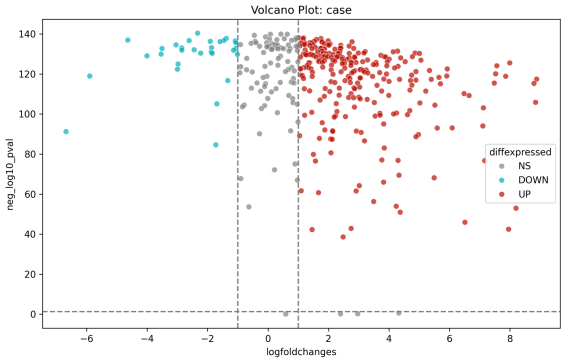
Dotplot Marker Genes



Heatmap Marker Genes



Violin Plot Marker Genes



Volcano Plot

Κεφάλαιο 6

Dockerization της Εφαρμογής

Η εφαρμογή έχει υλοποιηθεί ώστε να τρέχει πλήρως απομονωμένα μέσω Docker, διευκολύνοντας την εγκατάσταση και εκτέλεση σε οποιοδήποτε περιβάλλον.

Αρχείο Dockerfile

Το Dockerfile βασίζεται στην επίσημη εικόνα `python:3.11-slim` και περιλαμβάνει:

- Εγκατάσταση εξαρτήσεων από το αρχείο `requirements.txt`
- Αντιγραφή όλων των απαραίτητων αρχείων στον container
- Ορισμός entrypoint με την εντολή: `CMD ["streamlit", "run", "main.py"]`

Αρχείο requirements.txt

Το αρχείο `requirements.txt` περιλαμβάνει όλες τις βιβλιοθήκες που απαιτούνται για την εκτέλεση της εφαρμογής. Ενδεικτικά:

- streamlit, scanpy, anndata, matplotlib, numpy, pandas
- scipy, plotly, seaborn, harmonypy
- python-igraph, leidenalg, openpyxl, xlsxwriter

Εντολές Εκτέλεσης

Η δημιουργία και εκτέλεση του Docker container γίνεται με:

```
docker build -t scrna-app .
```

```
docker run -p 8501:8501 scrna-app
```

Αυτό καθιστά την εφαρμογή διαθέσιμη στη διεύθυνση: `http://localhost:8501`

Αρχείο .dockerignore

Το αρχείο `.dockerignore` περιλαμβάνει τα εξής:

- `.git/` – αποφυγή μεταφοράς git ιστορικού
- `.vscode/` – ρυθμίσεις editor
- `__pycache__/` – προερμηνευμένα αρχεία Python

Με αυτόν τον τρόπο μειώνεται το μέγεθος του Docker image και διασφαλίζεται καθαρό περιβάλλον εκτέλεσης.

Κεφάλαιο 7

Αποθετήριο στο GitHub

Ο πηγαίος κώδικας της εφαρμογής είναι διαθέσιμος στο GitHub, στο εξής αποθετήριο:

- **Link:** `https://github.com/chriskarydis/scRNA_seq_Pipeline`
- Περιέχει: ‘main.py‘, ‘Dockerfile‘, ‘requirements.txt‘, ‘.dockerignore‘, ‘README.md‘
- Το ‘README.md‘ περιλαμβάνει οδηγίες εγκατάστασης, τρέξιμο μέσω Docker, και παραδείγματα αρχείων.

Ο χρήστης μπορεί να κλωνοποιήσει το repository και να εκτελέσει την εφαρμογή είτε τοπικά είτε μέσω Docker σε περιβάλλον παραγωγής.