ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Τμήμα Πληροφορικής

Εικόνα που περιέχει ορθογώνιο παραλληλόγραμμο, σχεδίαση

Το περιεχόμενο που δημιουργείται από τεχνολογία AI ενδέχεται να είναι εσφαλμένο.

Εργασία Μαθήματος ***Αναλυτική Δεδομένων και Μηχανική Μάθηση***

|  |  |
| --- | --- |
| Αρ. Άσκησης - Τίτλος Άσκησης | ***Αναλυτική Δεδομένων και Μηχανική Μάθηση*** |
| Όνομα φοιτητή - Αρ. Μητρώου | Ραυτόπουλος Μάριος – ΜΠΚΕΔ24034 |
| Ημερομηνία παράδοσης | 15/03/25 |

# Εισαγωγή

Σκοπός της παρούσας εργασίας είναι η εφαρμογή τεχνικών Μηχανικής Μάθησης για την ανάλυση και μοντελοποίηση ενός πραγματικού dataset. Επιλέξαμε το MovieLens 100K Dataset, το οποίο περιέχει βαθμολογίες χρηστών σε κινηματογραφικές ταινίες.

Τα κύρια βήματα της εργασίας περιλαμβάνουν:

1. Προεπεξεργασία δεδομένων (30%): Καθαρισμός, μετασχηματισμοί, στατιστική ανάλυση, των δεδομένων με σκοπό να εξάγουμε αρχικές συσχετίσεις και συμπεράσματα των δεδομένων.
2. Συσταδοποίηση (Clustering) (30%): Εφαρμογή K-Means και DBSCAN, σύγκριση και ερμηνεία, ώστε να βρούμε διασυνδέσεις των δεδομένων που δεν φαίνονται με γυμνό μάτι.
3. Ταξινόμηση (Classification) (40%): Δημιουργία μοντέλων για πρόβλεψη, αξιολόγηση με κατάλληλους δείκτες.

Η τεχνική αναφορά που ακολουθεί περιγράφει την προσέγγισή μας σε κάθε στάδιο, τις παραμέτρους, τα αποτελέσματα και τη σχετική ερμηνεία.

# Περιγραφή του Συνόλου Δεδομένων

Το **MovieLens 100K** αποτελείται από:

* **ratings.csv**: πίνακας βαθμολογιών ταινιών (userId, movieId, rating, timestamp).
* **movies.csv**: πίνακας με τα μεταδεδομένα των ταινιών (movieId, title, genres).
* **tags.csv:** πίνακας με χαρακτηρισμούς ταινιών (userId, movieId, tag, timestamp)
* **links.csv:** πίνακας με τα links των ταινιών στα imdb, themoviedb (movieId, imdbId, tmdbId)

**Βασικά Στατιστικά**

* Αριθμός εγγραφών: ~100.000.
* Χρήστες: ~943.
* Ταινίες: ~1.682.

**3. Προεπεξεργασία Δεδομένων (Data Preprocessing)**

**3.1 Φόρτωση Δεδομένων**

1. Φορτώσαμε τα δεδομένα από τα csv σε pandas DataFrames.
2. Πραγματοποιήσαμε **συγχώνευση (merge)** στα δύο σύνολα με βάση το movieId, δημιουργώντας βοηθητικούς πίνακες, όπως ratings\_with\_genres.

**3.2 Καθαρισμός Δεδομένων**

* **Έλεγχος Κενών Τιμών**: Διαπιστώθηκαν [αναφέρατε αριθμό/ποσοστό] μη διαθέσιμες τιμές σε [συγκεκριμένες στήλες].
  + Επιλέξαμε **fillna(0)** / διαγραφή σειρών / άλλες μεθόδους για να αντιμετωπιστούν.
* **Ακραίες Τιμές (Outliers)**: Δεν υπήρχε ουσιαστικό ζήτημα outliers στις βαθμολογίες (που κυμαίνονται σε [1,5]).
* **Διαγραφή διπλότυπων εγγραφών**

**3.3 Μετασχηματισμοί**

* **One-Hot Encoding** για τις στήλες genres. Κάθε είδος (genre) μετατράπηκε σε dummy μεταβλητή (0/1).
* **Κωδικοποίηση Έτους** (προαιρετικά), αν το αρχικό dataset είχε πληροφορία για έτος κυκλοφορίας.
* **Κανονικοποίηση (Normalization)**: Επιλέξαμε MinMaxScaler για να μετατρέψουμε τα αριθμητικά πεδία στην κλίμακα [0,1].

**3.4 Στατιστική Ανάλυση**

* **Κατανομή Βαθμολογιών**: Παρατηρήθηκε ότι το 70% των βαθμολογιών συγκεντρώνεται μεταξύ 3 και 4 (σχήμα...).
* **Συχνότητα Ειδών (Genres)**: Τα δημοφιλέστερα είδη ήταν (π.χ.) *Drama*, *Comedy*, *Action*.
* **Διαγράμματα (Plots)**:
  + **Ιστόγραμμα** βαθμολογιών (Σχήμα 1).
  + **Διάγραμμα Ράβδων** για genres (Σχήμα 2).
  + **Correlation Heatmap** για συσχέτιση χαρακτηριστικών (Σχήμα 3).

*(Επισυνάψτε τα αντίστοιχα screenshots/διαγράμματα και σχολιάστε τα)*

**4. Συσταδοποίηση (Clustering)**

**4.1 Σκοπός & Σχεδιασμός**

Ο στόχος ήταν να ανακαλύψουμε **μη προφανή πρότυπα** στη συμπεριφορά των χρηστών ή στις ταινίες. Αποφασίσαμε να εστιάσουμε στα **χαρακτηριστικά είδους** και ενδεχομένως σε μεταβλητές όπως τον μέσο όρο βαθμολογίας ή τον αριθμό βαθμολογιών ανά ταινία.

* **Σύνολο Δεδομένων για Clustering**: Αφαιρέσαμε στήλες όπως userId, movieId· διατηρήσαμε τα one-hot features των genres.
* **Μέθοδος Κανονικοποίησης**: Χρήση MinMaxScaler.

**4.2 K-Means**

1. **Εύρεση Βέλτιστου K**:
   * Δοκιμάσαμε τιμές k∈{3,4,5,6,7,8,9,10}k \in \{3, 4, 5, 6, 7, 8, 9, 10\}k∈{3,4,5,6,7,8,9,10}.
   * Υπολογίσαμε *Inertia*, *Silhouette Score*, *Davies-Bouldin*, *Calinski-Harabasz*.
   * Επιλέχθηκε k=…k = \dotsk=… με κριτήριο τη μεγιστοποίηση του Silhouette Score και την παρατήρηση στο “Elbow” διάγραμμα.
2. **Τελική Εκτέλεση**:
   * Με k=…k = \dotsk=…, πραγματοποιήθηκε η ομαδοποίηση και προέκυψαν [π.χ. 5] συσταδοποιήσεις.
   * Ο **μέσος Silhouette Score** ήταν ~0.35, ένδειξη *μέτριας* διαχωριστικότητας.
3. **Ανάλυση Αποτελεσμάτων**:
   * Προβάλαμε τα δεδομένα σε 2 διαστάσεις μέσω **PCA** (Σχήμα 4). Οι συστάδες διαχωρίζονται … (σχολιασμός).
   * Εξετάσαμε τον αριθμό ταινιών σε κάθε συστάδα· παρατηρήθηκε ότι η συστάδα #2 είναι η μεγαλύτερη, κ.λπ.

**4.3 DBSCAN**

1. **Περιορισμός Διαστασιμότητας**: Χρήση PCA (π.χ. 10 συνιστώσες) για ταχύτερη εκτέλεση DBSCAN.
2. **Έρευνα Παραμέτρων ε\varepsilonε και min\_samples**:
   * Δοκιμάστηκαν ε∈{0.3,0.35,0.4,0.45,0.5}\varepsilon \in \{0.3, 0.35, 0.4, 0.45, 0.5\}ε∈{0.3,0.35,0.4,0.45,0.5} και min⁡\_samples∈{3,5}\min\\_samples \in \{3, 5\}min\_samples∈{3,5}.
   * Το καλύτερο Silhouette Score ήταν …, επιτυγχάνεται με ε=…\varepsilon = \dotsε=… και min⁡\_samples=…\min\\_samples = \dotsmin\_samples=….
3. **Τελική Εφαρμογή**:
   * Προέκυψαν [π.χ. 4] έγκυρες συστάδες και [π.χ. 200] outliers (ετικέτα -1).
   * Το Silhouette Score ήταν ~0.25, χαμηλότερο σε σχέση με το K-Means, ενδεχομένως λόγω πολλών outliers.

**4.4 Συμπέρασμα & Σύγκριση Τεχνικών**

* Το **K-Means** παρήγαγε πιο ομοιογενείς συστάδες (πιο υψηλό Silhouette).
* Το **DBSCAN** ανέδειξε ότι κάποιες ταινίες/χαρακτηριστικά δεν ταιριάζουν σε συγκεκριμένες συστάδες (outliers).
* Και οι δύο αλγόριθμοι επιβεβαιώνουν ότι υπάρχουν συγκεκριμένα μοτίβα, π.χ. μία συστάδα είναι κυρίως **Action/Adventure**, άλλη κυρίως **Comedy**, κ.λπ.

**5. Ταξινόμηση (Classification)**

**5.1 Ορισμός Στόχου (Label)**

Για τη δημιουργία μοντέλου ταξινόμησης, ορίσαμε τη δυαδική ετικέτα **is\_popular** για τις ταινίες.

* Κριτήριο: Αν ο αριθμός βαθμολογιών μίας ταινίας ξεπερνά τη διάμεσο του συνολικού δείγματος, τότε is\_popular = 1, αλλιώς 0.
* Εναλλακτικά κριτήρια θα μπορούσαν να βασιστούν στον μέσο όρο της βαθμολογίας, κ.λπ.

**5.2 Προετοιμασία Συνόλου Δεδομένων**

* **Χαρακτηριστικά (Features X)**: Τα κωδικοποιημένα genres + (προαιρετικά) έτος κυκλοφορίας ή άλλες αριθμητικές στήλες.
* **Στόχος (y)**: Δυαδική μεταβλητή is\_popular.
* Διαχωρίσαμε σε **train/test** (80%/20%) με stratify=y για διατήρηση της αναλογίας θετικών/αρνητικών δειγμάτων.
* Εφαρμόσαμε MinMaxScaler ή StandardScaler στα χαρακτηριστικά.

**5.3 Εκπαίδευση Μοντέλων**

Δημιουργήσαμε τρία μοντέλα:

1. **Random Forest Classifier**
2. **SVM (Support Vector Machine)**
3. **Neural Network (MLPClassifier)**

Για αξιολόγηση:

* **Accuracy** (Ακρίβεια)
* **Classification Report** (Precision, Recall, F1)
* **Confusion Matrix**
* **ROC-AUC Curve**
* **Precision-Recall Curve** (πιο σημαντική αν τα δεδομένα είναι μη ισορροπημένα)

**5.3.1 Random Forest**

* **Βασικές Παράμετροι**: n\_estimators=100, max\_depth=None κ.λπ.
* **Accuracy** στο test set: π.χ. 0.85.
* **F1 Score**: π.χ. 0.83.
* **Συμπέρασμα**: Καλή ικανότητα γενίκευσης, χαμηλή υπερεκπαίδευση.

**5.3.2 Support Vector Machine**

* Χρησιμοποιήσαμε SVC με default ρυθμίσεις (kernel=‘rbf’, C=1.0).
* **Accuracy**: π.χ. 0.80.
* **Παρατηρήσεις**: Ελαφρώς χαμηλότερη επίδοση από RF· πιθανή βελτίωση με GridSearch για παραμέτρους C, γ, κ.λπ.

**5.3.3 Neural Network (MLP)**

* **Αρχιτεκτονική**: 1 κρυφή στρώση με 100 νευρώνες, max\_iter=500.
* **Accuracy**: π.χ. 0.82.
* **ROC-AUC**: 0.85.
* **Σχόλιο**: Η επίδοση είναι συγκρίσιμη με τα άλλα μοντέλα· ίσως απαιτείται tuning (π.χ. διαφορετικό μέγεθος κρυφών επιπέδων).

**5.4 Τελική Σύγκριση Μοντέλων**

Παρακάτω π.χ. ένας συνοπτικός πίνακας (ενδεικτικά νούμερα):

| **Μοντέλο** | **Accuracy** | **F1 Score** | **ROC-AUC** |
| --- | --- | --- | --- |
| Random Forest | 0.85 | 0.83 | 0.88 |
| SVM (RBF) | 0.80 | 0.78 | 0.84 |
| MLP Classifier | 0.82 | 0.80 | 0.85 |

*(Εισάγετε αληθινά αποτελέσματα και συζητήστε τις διαφορές.)*

**6. Συμπεράσματα**

* **Data Preprocessing**: Η προκαταρκτική επεξεργασία βελτίωσε την ποιότητα των δεδομένων μας (handling missing values, καθαρισμός, μετασχηματισμοί). Τα διαγράμματα EDA υπογράμμισαν την κατανομή των βαθμολογιών και την επικράτηση ορισμένων ειδών.
* **Clustering**:
  + Το K-Means επέτρεψε σχετικά καθαρές συστάδες, ενώ το DBSCAN φανέρωσε outliers.
  + Η ποιοτική ερμηνεία έδειξε π.χ. μια συστάδα ταινιών περιπέτειας, μια κωμωδίας κ.λπ.
* **Classification**:
  + Ο Random Forest είχε την **καλύτερη** απόδοση, ενώ το SVM ήταν ελαφρώς χειρότερο στα default hyperparameters.
  + Μικρές διαφορές παρατηρήθηκαν στο precision/recall των δημοφιλών/μη δημοφιλών.

**Δυνατές Επεκτάσεις**

* Εφαρμογή πιο **εκτενούς hyperparameter tuning** για SVM/Neural Network.
* Λήψη **μεγαλύτερης έκδοσης** MovieLens (π.χ. 1M ή 20M) για πιο αντιπροσωπευτικά αποτελέσματα.
* Προσθήκη **χρονικής πληροφορίας** (timestamp) για να μελετηθούν αλλαγές στις προτιμήσεις των χρηστών στο χρόνο.

**7. Αναφορές**

1. MovieLens Dataset: [GroupLens.org](https://grouplens.org/datasets/movielens/latest/)
2. Scikit-Learn Documentation: <https://scikit-learn.org/>
3. Εγχειρίδια/σημειώσεις μαθήματος
4. Σχετική βιβλιογραφία (π.χ. Bishop, *Pattern Recognition & Machine Learning*)

**8. Παράρτημα: Κώδικας**

Στο συμπιεσμένο αρχείο *.zip* που επισυνάπτεται υποβάλλονται:

* **Τεχνική Αναφορά (PDF/DOCX)**: Το παρόν κείμενο με αναλυτικές περιγραφές.
* **Jupyter Notebook / Source Code**: Όλη η υλοποίηση σε Python (π.χ. MOVIES.ipynb).
* **Παραγόμενα Αρχεία (π.χ. plots)**: Διαγράμματα και πίνακες αξιολόγησης σε μορφή εικόνας.

*(Συμπεριλάβετε εδώ οτιδήποτε απαιτείται για την αναπαραγωγή των αποτελεσμάτων.)*

**Τέλος Αναφοράς**

**Σημείωση**:

* Φροντίστε να προσαρμόσετε τα **νούμερα**, τους **πίνακες** και τα **διαγράμματα** βάσει των πραγματικών αποτελεσμάτων που πήρατε.
* Ενσωματώστε **συγκεκριμένα plots** (Silhouette, elbow, confusion matrix κ.λπ.) και σχολιάστε τα συνοπτικά μέσα στο κείμενο.
* Διατηρήστε σε ξεχωριστά αρχεία τον κώδικα (π.χ. Jupyter Notebook) και την αναφορά σε PDF ή DOCX.
* Για πλήρη **ακαδημαϊκή** αρτιότητα, προσθέστε σύντομη θεωρητική εισαγωγή (π.χ. τι είναι K-Means, DBSCAN, ROC curve) αν το ζητά το μάθημα.

Το **νόημα** όλων αυτών των διαδικασιών (προεπεξεργασία, συσταδοποίηση, ταξινόμηση) είναι να μας **βοηθήσουν να κατανοήσουμε και να αξιοποιήσουμε** τα δεδομένα σε πολλαπλά επίπεδα:

1. **Προεπεξεργασία δεδομένων**:
   * *Τι επιδιώκουμε:* Να καθαρίσουμε και να μετασχηματίσουμε τα δεδομένα έτσι ώστε να είναι έγκυρα, συνεπή και κατάλληλα για τα επόμενα βήματα.
   * *Τι πετυχαίνουμε:*
     + Βελτιώνουμε την ποιότητα της πληροφορίας (π.χ. αντιμετώπιση κενών τιμών, ακραίων τιμών).
     + Διευκολύνουμε τις αλγοριθμικές μεθόδους (π.χ. η κανονικοποίηση βοηθά να μη «βαραίνουν» δυσανάλογα μεγάλες τιμές).
     + Με μια πρώτη στατιστική ανάλυση (EDA), αποκτούμε **έναν «χάρτη»** του dataset (συχνότητες, κατανομές), ώστε να δούμε αν κρύβονται παραφωνίες ή ενδιαφέροντα μοτίβα.
2. **Συσταδοποίηση (Clustering)**:
   * *Τι επιδιώκουμε:* Να βρούμε ομάδες (clusters) δεδομένων που να μοιράζονται κοινά χαρακτηριστικά, χωρίς να έχουμε προκαθορισμένες ετικέτες ή κατηγορίες.
   * *Τι πετυχαίνουμε:*
     + Ανακαλύπτουμε **μη προφανή πρότυπα** ή μοτίβα. Π.χ. ποιες ταινίες μοιάζουν μεταξύ τους ως προς τα είδη (genres) ή τα στατιστικά βαθμολογιών.
     + Μπορούμε να εντοπίσουμε «ακραίες» περιπτώσεις / outliers (ειδικά με DBSCAN).
     + Παίρνουμε μια **επισκόπηση της εσωτερικής δομής** των δεδομένων, που μας βοηθά να βγάλουμε επιχειρηματικά ή ερευνητικά συμπεράσματα.
3. **Ταξινόμηση (Classification)**:
   * *Τι επιδιώκουμε:* Να δημιουργήσουμε μοντέλα που μπορούν να προβλέψουν συγκεκριμένες κατηγορίες (π.χ. αν μια ταινία είναι «δημοφιλής» ή όχι) βάσει των χαρακτηριστικών της.
   * *Τι πετυχαίνουμε:*
     + Αποκτάμε **προβλεπτική ικανότητα**: όταν βλέπουμε ένα νέο δείγμα (μια καινούρια ταινία ή νέα δεδομένα), μπορούμε να εκτιμήσουμε την κατηγορία του.
     + Επιβεβαιώνουμε αν ορισμένα χαρακτηριστικά (genres, αριθμός κριτικών κ.λπ.) έχουν σχέση με την «επιτυχία» ή άλλη ιδιότητα που εξετάζουμε.
     + Αποτιμούμε την απόδοση διαφόρων αλγορίθμων (π.χ. Random Forest, SVM) και βλέπουμε ποιος ταιριάζει καλύτερα στην περίπτωσή μας.

**Τι «βλέπουμε» συνολικά**

* **Καθαρότερα δεδομένα** σημαίνει **καλύτερες αποφάσεις**. Όταν αντιμετωπίσουμε λάθη, κενά ή θόρυβο, τα μοντέλα μπορεί να πάρουν λανθασμένες εκτιμήσεις. Άρα, η προεπεξεργασία μάς σώζει από πολλά μελλοντικά προβλήματα.
* Με το **Clustering** μαθαίνουμε αν το dataset έχει **υπο-ομάδες με κοινά χαρακτηριστικά**. Στο MovieLens, π.χ., μπορεί να βρούμε ταινίες δράσης που λαμβάνουν παρόμοιες βαθμολογίες ή κοινό χρηστών με παρόμοιες προτιμήσεις.
* Με την **Classification** περνάμε από την εξερευνητική φάση (ποιοι μοιάζουν μεταξύ τους) στη **φάση πρόβλεψης**: μπορούμε να προβλέψουμε αν μια νέα ταινία (ή ένας νέος χρήστης) θα θεωρείται «δημοφιλής», «πετυχημένη» κ.λπ.
* Όλα αυτά **μας επιτρέπουν** να κτίσουμε αναλυτικές και προγνωστικές εφαρμογές — είτε για **συστάσεις** ταινιών, είτε για **επιχειρηματικές αποφάσεις**, είτε για **ερευνητικές ερωτήσεις** γύρω από τα δεδομένα.

Με απλά λόγια, η όλη διαδικασία **ενώνει τα κομμάτια**:

1. *Προεπεξεργασία*: Καθαρό Dataset.
2. *Συσταδοποίηση*: Κρυμμένα πρότυπα & μοτίβα.
3. *Ταξινόμηση*: Προγνωστική δύναμη & ενέργειες με βάση μοντέλα.

Στον πυρήνα της, η ανάλυση αυτή δείχνει **πώς «βλέπουμε» και αξιοποιούμε τη γνώση** που βρίσκεται κρυμμένη στα δεδομένα, η οποία μπορεί να μη φαινόταν απλώς με το «μάτι» ή με απλές μεθόδους.

**1. Τι μπορούμε να δημιουργήσουμε από αυτήν την ανάλυση;**

Ανάλογα με τα αποτελέσματά μας, θα μπορούσαμε να κατασκευάσουμε:

**🎯 1. Ένα Σύστημα Σύστασης Ταινιών (Recommendation System)**

📌 **Πώς λειτουργεί;**

* Χρησιμοποιούμε το **Clustering** για να βρούμε ομάδες ταινιών με παρόμοια χαρακτηριστικά.
* Όταν ένας χρήστης βαθμολογεί θετικά μια ταινία, του προτείνουμε άλλες από την ίδια ομάδα (cluster).
* Αντί για **απλό φιλτράρισμα** με βάση το genre (π.χ. "αν σου άρεσε μια ταινία δράσης, δες και αυτή"), **εκμεταλλευόμαστε τα patterns που ανακαλύψαμε με τα δεδομένα**.
* Θα μπορούσαμε να προσθέσουμε ακόμα πιο **προσωποποιημένες προτάσεις** με machine learning (όπως κάνουν το Netflix & το Spotify).

💡 **Τι προσφέρει;**  
✅ Προτάσεις με βάση **μοτίβα χρηστών** & όχι απλά το genre.  
✅ Δυνατότητα χρήσης **AI για αυτοματοποιημένες προτάσεις**.  
✅ Εξατομίκευση των προτιμήσεων του χρήστη.

**📊 2. Μοντέλο Πρόβλεψης Επιτυχίας μιας Νέας Ταινίας**

📌 **Πώς λειτουργεί;**

* Το μοντέλο **Classification** που φτιάξαμε μπορεί να χρησιμοποιηθεί για να **προβλέπει αν μια νέα ταινία θα είναι δημοφιλής ή όχι**.
* Με βάση τα χαρακτηριστικά της (είδος, αριθμός βαθμολογιών, μέσος όρος βαθμολογίας), μπορούμε να κάνουμε πρόβλεψη πριν καν κυκλοφορήσει!
* Οι παραγωγοί ταινιών μπορούν να αξιολογήσουν **αν μια ταινία έχει πιθανότητες να γίνει επιτυχία** πριν επενδύσουν σε αυτήν.

💡 **Τι προσφέρει;**  
✅ Δυνατότητα **πρόβλεψης της δημοτικότητας** μιας ταινίας πριν κυκλοφορήσει.  
✅ Χρήσιμο εργαλείο για **παραγωγούς & κινηματογραφικά στούντιο**.  
✅ Βοήθεια στη **λήψη αποφάσεων για το marketing μιας ταινίας**.

**📌 3. Ανάλυση Προτιμήσεων Κοινού για Στοχευμένο Marketing**

📌 **Πώς λειτουργεί;**

* Με την **ανάλυση clusters** μπορούμε να **διαχωρίσουμε το κοινό** σε ομάδες με διαφορετικά γούστα.
* Π.χ., αν εντοπίσουμε ότι υπάρχει **μια ομάδα χρηστών που λατρεύουν sci-fi & action** αλλά δεν βλέπουν καθόλου ρομαντικές ταινίες, οι κινηματογραφικές εταιρείες μπορούν να **στοχεύσουν διαφορετικά τις διαφημίσεις τους**.
* Μπορούμε να δημιουργήσουμε **προφίλ χρηστών** για πιο στοχευμένο περιεχόμενο.

💡 **Τι προσφέρει;**  
✅ **Εξατομικευμένο marketing** στις πλατφόρμες streaming & κινηματογραφικές εταιρείες.  
✅ Δυνατότητα **στοχοποίησης διαφημίσεων** ανάλογα με το κοινό.  
✅ Προσαρμογή των **trailers και των posters** για διαφορετικές ομάδες χρηστών.

**2. Πώς η ανάλυση μας οδηγεί σε αυτά τα αποτελέσματα;**

🔹 **Από την προεπεξεργασία** → Καθαρίζουμε τα δεδομένα για να έχουν νόημα οι αναλύσεις μας.  
🔹 **Από τη συσταδοποίηση (clustering)** → Καταλαβαίνουμε ποια ταινία μοιάζει με ποια άλλη & ποιοι χρήστες έχουν παρόμοιες προτιμήσεις.  
🔹 **Από την ταξινόμηση (classification)** → Δημιουργούμε ένα μοντέλο που **προβλέπει την επιτυχία μιας ταινίας** με βάση τα χαρακτηριστικά της.  
🔹 **Από τις μετρήσεις μοντέλων** → Επιβεβαιώνουμε αν οι μέθοδοι που χρησιμοποιούμε αποδίδουν και πώς μπορούμε να τις βελτιώσουμε.

**3. Πρακτική Χρησιμότητα της Εργασίας**

Αν αυτή η ανάλυση εφαρμοζόταν σε **μια πραγματική επιχείρηση**, όπως το Netflix, το IMDB ή μια εταιρεία παραγωγής ταινιών, θα μπορούσε να έχει **τεράστια εμπορική αξία**:

🎯 **Πλατφόρμες Streaming (Netflix, Disney+, Prime Video):**  
✅ **Βελτίωση του recommendation system** → Προτάσεις βασισμένες σε AI & συσταδοποίηση.  
✅ **Ανάλυση κοινών προτιμήσεων** → Προώθηση περιεχομένου ανά χρήστη.

🎯 **Κινηματογραφικές Εταιρείες (Warner Bros, Universal):**  
✅ **Πρόβλεψη αν μια ταινία θα είναι επιτυχία πριν κυκλοφορήσει**.  
✅ **Βελτίωση marketing στρατηγικής** με στοχευμένες διαφημίσεις.

🎯 **Διαφημιστικές Εταιρείες:**  
✅ **Ανάλυση προφίλ χρηστών** για στοχευμένες προωθήσεις ταινιών.  
✅ **Βελτιστοποίηση διαφημίσεων** με βάση προτιμήσεις θεατών.

**4. Συμπέρασμα**

🎯 Αυτό που κάνουμε **δεν είναι απλά μια ακαδημαϊκή ανάλυση** – είναι μια πραγματική εφαρμογή της **Μηχανικής Μάθησης** στον χώρο του κινηματογράφου!  
🎯 Τα αποτελέσματά μας θα μπορούσαν να χρησιμοποιηθούν **σε εμπορικές & επιστημονικές εφαρμογές** (streaming, παραγωγές, διαφήμιση).  
🎯 Το clustering μας δείχνει **τι μοτίβα κρύβονται** στις ταινίες και στους χρήστες, ενώ το classification βοηθά **να προβλέψουμε τι θα γίνει δημοφιλές**.

**➡ Έτσι, έχουμε χτίσει ένα βασικό θεμέλιο για ένα "AI-Powered" σύστημα προτάσεων, προβλέψεων και marketing στο χώρο του κινηματογράφου. 🚀**