

* פתרון בעיות חיפוש (ניתן לפתור הרבה בעיות בעזרת ייצוגם כבעיית חיפוש):
  + חיפוש בעץ
  + חיפוש בגרף(closed\_list עם states שביקרנו)
  + אלגוריתמי חיפוש סיסטמטיים:
    - uninformed search
      * BFS (tree search)
      * BFS (graph search)
      * DFS (tree search)
      * DFS-L: Depth-Limited-Search (ניסיון ראשון)
      * **IDS: Iterative-Deepening-DFS (ניסיון שני)**
      * BestFS (graph search)
        + **Uniform-Cost Search: f=g**
    - informed search
      * BestFS (graph search)
        + **Greedy BestFS: f=h**
        + **A\* search: f=g+h**

optimally-efficient

* + - * **IDA\*: Iterative Deepening A\***
  + Completeness (שלמות)
  + Optimality (אופטימליות)
  + פונקציות היוריסטיות:
    - Admissible/Underestimate heuristic function(קבילה)
    - Consistent/Monotone heuristic function(עקבית/מונוטונית)
    - Dominates(שולטת על/מיודעת מ)
  + אלגוריתמי חיפוש לוקאליים(local search):
    - דוגמאות ליישום בעולם האמיתי
    - Hill Climbing search
      * Simple
      * Stochastic
      * First Choice
      * Random-Restart
      * Simulated annealing
  + בעיות סיפוק אילוצים (CSP)
    - דוגמאות
      * Cryptarithmetic puzzle
      * 4 queens problem
      * map coloring problem
        + Binary CSP & Binary Constraint graph
    - פתרון CSP בעזרת חיפוש סיסטמטי:
      * back tracking
        + שיפור הסדר

Min. Remaining Values (MRV)

Least Constraining Value (LCV)

* + - * + שיפור הסינון

Forward Checking

Arc consistency

* + - פתרון CSP בעזרת חיפוש לוקאלי - Hill climbing with Simulated annealing.
  + משחקים מול יריבכ
    - סוגי המשחקים שנרצה לפתור בעזרת חיפוש
    - game tree, strategy tree
    - **Minimax Search**
      * Resource bound minimax search
        + horizon effect
      * **Alpha-Beta pruning**
* פתרון בעיות תכנון(Planning)
  + STRIPS
    - האלגוריתם
    - not optimal: Sussman anomaly
    - not complete: can't swap registers' value (due to sequential solving, not holistic)
* למידה
  + מהי למידת מכונה?
  + למה נלמד?
  + מה נלמד?
  + מה הייצוג של המודל?
  + מה הקלט/הפידבק שממנו אנחנו לומדים?
    - Supervised learning
      * Regression
      * Classification
        + train set(including validation set), test set
      * **K-Nearest-Neighboors(KNN)**
    - Unsupervised learning
      * **Clustering**
        + hard vs soft clustering
        + **K-Means clustering**

Elbow method for finding K(num of clusters)

* + - * + Hierarchial Clustering

**Agglomerative (bottom-up)**

הגדרת קירבה בין clusters:   
single-link,complete-link,centroid,average-link

Divisive (top-down)

* + - Semi-supervised learning
    - Reinforcement learning
  + הכללה(Inductive) או היסק(Deductive)?
    - הכללה
      * overfitting
      * Ockham Razor
  + קטגוריזציה בעזרת עצי החלטה DTL
    - אלג' בניית עץ ID3 ממשפחת TDIDT(Top Down Induction of Decision Trees)
      * Choose\_attribute בעזרת הInformation gain הכי גבוה(חישוב אנטרופיות)
    - המרת ערכים רציפים
    - טיפול בערכים חסרים(3 שיטות)
    - גיזום(Pruning) עצי החלטה כדי להחליש השפעת רעש(להמנע מoverfitting)
      * גיזום מוקדם- בזמן היצירה נהפוך צומת עם מס' דוג' קטן מידי לעלה. התיוג לפי רוב הדוג'.
      * גיזום מאוחר- לאחר היצירה נגזום כל תת עץ אפשרי ונשווה למקורי. נתחיל לגזום מלמטה.
  + הערכת טיב היפותזה מסויימת ע"י פרופורציית הטעויות:
    - על test set
    - על כל המידע בעזרת k-fold cross validation (חלוקה לk, אימון k-1. הכל k פעמים, וניקח את הממוצע).
    - Stratified k-fold cross validation(דומה אבל שומר על יחס התיוגים בכל fold)
  + overfitting שנגרם ממודל מסובך מידי(עץ גדול מידי) לעומת underfitting מעץ קטן מידי
  + רגולריזציה
    - הקטנת Cost
    - פונקציות loss
  + Linear Regression
    - Gradient Descent(נזוז כנגד כיוון הנגזרת של כל משקולת)
      * - learning rateα
      * רגולריזציה
        + ע"י L1
        + ע"י L2
        + Linear classification with logistic(Sigmoid) regression
      * בloss של כמה דוגמאות מתחשבים כדי לחשב לאן ללכת בצעד הבא?
        + Batch gradient Descent
        + Stochastic Gradient Descent (SGD)
        + Mini batches gradient descent
* מערכות המלצה
  + משימות
    - Top-k (recommendations) prediction
    - Rating prediction
    - התמודדות עם שיקולים נוספים
  + מבנה מערכת המלצה: Candidate generation, Scoring, Re-ranking
  + מה זה סט המלצות טובות ?
  + מהי מערכת המלצות טובה ?
  + קלט מהמשתמש:
    - Explicit
      * user/item bias
      * תפיסה עצמית לעומת התנהגות בפועל
      * private-social dissonance
    - Implicit
  + Clickbaits
  + מדידת ביצועי מערכת המלצה
    - Offline analysis
      * מטריקות(מדידת טיב החיזוי) לRating prediction:
        + MAE(Mean Absolute Error)
        + RMSE(Root Mean Squared Error)
      * מטריקות(מדידת טיב החיזוי) לTop-k prediction:
        + List-relevance metrics

Precision@k (P@K)

Recall@k (R@K)

* + - * + Rank-aware metrics

MRR(Mean Reciprocal Rank)

ARHR(Average Reciprocal Hit Rank)

* + - User surveys
    - Online analysis: A/B testing
  + סוגי מערכות המלצה
    - Non personalized
    - Collaborative-filtering
      * Used-based (User-to-User similarity)
      * Item-based (Item-to-Item similarity)
      * מדדי דימיון: cosine similarity...
      * Matrix factorization
      * user/product cold start
    - Content based
    - Hybrid/Ensemble
* Naïve Bayes
  + Quantifying Uncertainty (כימות אי ודאות)
    - expectedUtility
  + Bayes' rule
  + חישוב הנוסחא המלאה לסיווג בייס
    - prior and posterior probability
    - Maximum a posteriori hypothesis (CMAP: מה התיוג בהינתן התצפית-נחשב בעזרת argmax..)
    - Maximum Likelihood hypothesis (CNB: כנ"ל, אם ההס' לכל תיוג שווה
    - Conditional Independence Assumption
    - חישוב p(xi | cj)
    - (laplace) smoothing
    - נוסחא מלאה
* Probabilistic Belief State
  + נוסחאות.. הסתברות מותנית, ב"ת...
  + עדכון הbelief state בהינתן עדות(טבלת הfull joint probability table)
  + Bayesian Network כפתרון יותר קומפקטי מטבלת full joint probability table.
    - בטבלה צריך בסה"כ 2 בחזקת מס' המשתנים(פחות 1 כי אפשר להסיק בעזרת משלים ל1), ברשת צריך בטבלה הקטנה של כל אחד(CPT) 2 בחזקת מס' האבות(בדר"כ פחות מטבלת full join).
    - אלג' בניית Bayesian Network בהינתן אוסף משתנים(עדיף סיבה לפני תוצאה)
* Markov Decision Processes
  + Markov באופן כללי: תהליכים שבהם התוצאה של המצב הבא לא תלויה בעבר ובעתיד אלא רק בהווה.
  + הגדרת MDP
  + policy
  + Expectimax tree with chance nodes for each action
  + Discounting with γ (decay factor)
  + dealing with infinite reward routes
  + חישוב V(מרובעים) וQ(משולשים)
    - רקורסיבית: Bellman Equations
    - איטרטיבית: Value Iteration
      * policy extraction from V\*­(or from Q\*)
    - Policy Iteration
* תורת המשחקים
  + סוגי משחקים: סכום 0, סימטרי, שיתופי.
  + אסטרטגיות דומיננטיות(חזק/חלש מאוד): למחוק שורות/עמודות
  + אסטרטגיות טהורות: נאש