- dueling bandit
  - Copeland Bandit
    - motivation
      - a Condorcet winner may not exist
        - Condorcet winner: the arm beat all other arms with probability of 0.5
        - Copeland winner: the arm beat the most other arms
    - SAVAGE
      - motivation: solve the N dimensional parametric decision problem with high confidence
      - problem definition
        - parametric decision problem: find the optimal arm choice; with minimum amount of samples
        - (ε, δ)-PAC: epsilon-approximation (error rate); probability of at least 1-delta (confidence bound); terminate with T samples
      - Assumption
        - Condorcet assumption: there is a unique arm that is superior to the others; does not require the arms to be totally ordered
      - Algorithm
        - idea: work by reducing progressively a box-shaped confidence set H until a single decision remains in f(H).; eliminate the arms loss with high probability to another arm until leave with Condorcet winner
        - Condition of "Accept": Only one arm (Condorcet) left in feasible set; other replacement, e.g. bound of sub of r.v.
        - IndepTest: choose the preference: bound of confidence interval ([a,b]); exist copeland winner
        - Parameters Elimination by Sampling
        - Regret: outperforms IF and BTM for moderate K: O(K^2 log T); finite horizon; infinite horizon; regret analysis: Theorm 1
      - Application
        - Copeland bandit
          - Coperland score: Copeland winner; number of one-to-one winning
        - Borda bandit
          - mean performance against others
          - Advantage: stability (the utility is linear); reduces the dimension of the problem to only K parameters
          - Disadvantage: Borda winner may not be the Condorcet winner
          - SAVAGE Sampling/Borda: a Borda relaxation of the COndorcet feasible set
      - Reference: Generic Exploration and K-armed Voting Bandits, 2013
    - CCB, RCB
      - idea: Upper confidence bound
      - CCB
        - motivation: design for small number of arms Copeland winner problem
        - principle: optimism followed by pessimism
          - optimistic copeland winner: have confidence to beat other arms
          - pessimistic opponent: have confidence to beat potential winner
        - Algorithm: Remove non-Copeland winners; Add Copeland winners
        - Regret
      - RCB
        - motivation: design for large-scale arms Coepland winner problem
        - Algorithm: Approximate Copeland Solver; KL-best arm-elimination indetification
        - Regret
      - Reference: Copeland Dueling Bandits, 2015
    - RMED
      - idea: Minimum Empirical Divergence
      - theory: empirical divergence I(t): defines the minimum number of comparisons to prove i is inferior to j; likehood function of condorcet winner: exp(-I(t))
      - Algorithm
        - subroutine
          - RMED1: set the exploration phase as 1
          - RMED2: set the exploration phase as 1; Unlike RMED1, RMED2 keeps drawing pairs of arms i, j at least α log log t times (Line 10 in Algorithm 1
          - RMED2FH: conducts an "α log log T" exploration in the initial phase
      - regret
      - Reference: Regret Lower Bound and Optimal Algorithm, 2015
    - DTS
      - motivation: Extends the Thompson Sampling in relative setting to apply dueling bandit; TS achieves lower regret in practive and more robust than other algorithms
      - Assumption: Copeland winner
      - Algorithm
        - D-TS
          - idea: selects both the first and the second candidates according to Thompson Sampling; sample from the parameters representing the number of winning: B_{ij}
          - elimination
            - RUCB-based elimination: choose the first arm based on upper confidence bound
            - RLCB-based elimination: choose the second arm only from the uncertain pair (with high lower confidence bound)
            - objective: avoid falling in suboptimal comparisons
        - D-TS+: compare with D-TS
          - change the selection of the first arms
          - minimize the regret of comparison
      - regret: O(K log T + K^2 log log T)
      - Reference: double-thompson-sampling-for-dueling-bandits-Paper, 2016
  - Dependent arm
    - motivation: solve the problem of dueling bandit arm with dependent arms
    - Gaussian process
      - use covariance between arms to model the dependency
      - posterior inference updates the mean reward estimates for all arms and update the dependency
    - Self-Sparring
      - idea
        - reduces the multi-dueling bandits problem to a conventional MAB bandit
        - simultaneously dueling multiple arms as well as modeling dependencies between arms using a kernel
        - Assumpation: approximate linearity: using linear function to estimate the preference based on utility u
      - Algorithm
        - Ind-Self-Sparring
          - independent case
          - SBM: TS
        - Kernel-Self-Sparring
          - dependent case
          - use Gaussian processes GP(μ(b), k(b, b')). to model the preference function f(b)
          - apply Bayesian update using (i_j (t), r_{jk}) to obtain (μ_t, σ_t)
      - Regret: O(K log T /Δ)
      - Reference: Multi-dueling Bandits with Dependent Arms, 2017
    - CoSpar
      - the user can suggest improvements in the form of coactive feedback
      - Reference: Preference-Based Learning for Exoskeleton Gait Optimization, 2020
  - Condorcet winner
    - IF
      - motivation: solve the problem where absolute rewards have no natural scale or are difficult to measure
      - Assumption
        - ordered arms
        - strong stochastic transitivity (SST)
        - stochastic triangle inequality (STI)
        - model: logisitic (Bradley-Terry model); Gaussian model
      - Algorithm
        - IF1
          - property: Condorcet winner; SST; finite horizon
          - idea: combine estimate \hat{P} with confidence interval, update the optimality
        - IF2
          - property: Condorcet winner; SST; finite horizon
          - core: IF1 + pruning
      - Regret
        - logarithmic round
        - regret analysis: confidence intervals; regret per match; random walk
      - Lower Bound: regret of dueling bandit algorithm has d lower bound
      - Reference: The K-armed Dueling bandits problem, 2009
    - BTM
      - motivation: extend the Dueling Bandits Problem to a relaxed setting where preference magnitudes can violate transitivity
      - Assumption: relaxed stochastic transitivity (RST); stochasitic triangle inequality (STI)
      - Algorithm
        - BTM
          - setting: PAC: confident near-optimal bandit; online
          - property: Condorcet winner; ordered arms; RST: drawback: require explicit value of \gamma; mistake-free
          - horizon: PAC: infinite; online: finite
          - idea: compare the fewest recored comparison bandit with the mean bandit (random sample)
          - confidence interval: PAC; online — different parameters
          - Another perspective: Borda score
            - theorem: First, the Borda score of the Condorcet winner is always greater than or equal to 0.5; the Condorcet winner remains optimal after removing other bandits
            - method: Keep eliminating Borda losers, eventually the Condorcet winner
      - Regret
        - logarithmic round
        - regret analysis: mistake-free; number of removed comparisons; incurred regret of comparison: RST
      - Reference: Beat the Mean Bandit, 2011
    - UBDB
      - motivation: reducing the Dueling Bandits problem to the conventional (stochastic) Multi-Armed Bandits problem
      - UBDB problem
        - reward (not observed): average utility; choice-based utility
        - linear link function: observed choice: left alternative; right alternative
        - regret: accumlative gap with optimlity
        - partial monitoring games: objective: choose at each round an action from some finite set of actions, and receive a reward based on some unknown function chosen by an oblivious process
      - SBM
        - reset
        - advance: choose the arm with some strategy, observe
        - feedback: update based on feedback
      - Assumption
        - linear link function
        - procedure is based on SBM
        - Large of Structured: Large or possibly infinite set of arms X
        - Unstructured: the elements of X typically have no structure
      - idea: view the dueling bandit as the dueling of two arms with different strategy (left and right)
      - Algorithm
        - Structured: Doubler: left: random choice; right: SBM
        - Unstructured: MultiSBM: left: arm of last round; right: SBM
        - Sparring: left: SBM; right: SBM
      - Regret
        - Doubler: when SBM is UCB
        - MultiSBM
        - Sparring: a constant times the regret of the two SBMs
        - Lower Bound: UBDB problem has the same lower boudn of the classic MAB problem
        - Setting
          - UBDB setting: constant utility of arms; UBDB has a lower bound of √KT matching with MAB
          - Adversial Setting: arbitrary utilities of arms; Sparring can obtain the same regret bound of SBM with adversial setting
      - Reference: Reducing Dueling Bandits to Cardinal Bandits, 2014
    - RUCB
      - motivation: Extends the Upper Confidence Bound algorithm in relative setting to apply dueling bandit
      - standard setting
        - finite-horizon setting: fixed number of iteration; goal: maxmize the accuracy of returning the best arm in T
        - horizonless setting: unfixed T or infinite; goal: minimize the cunulative regret over time
        - transition: horizonless to finite: stop at T; finite to horizonless: doubling trick
      - Assumption: Condorcet winner; horizonless or finite
      - Algorithm
        - RUCB
          - idea: Apply UCB in relative setting: upper bound
          - two kinds of comparison: optimistically: u_{cj}; pessimistically: u_{jc}; objective: avoid auto-comparisons until is great certainty of optimality
          - Major Improvement: achieves high-probability regret bounds while making minimal assumptions other than assuming a Condorcet winner.
          - Causes: avoids arm elimination
        - MergeRUCB: idea: partition the K arms into small batches and compares arms within each batch
      - Regret
        - RUCB: O(K^2 + K log T)
        - MergeRUCB: O(K log T); Regret analysis: MergeRUCB does not require global pairwise comparisons between all pairs of arms than RUCB
      - Reference: Relative Upper Confidence Bound for the K-Armed Dueling Bandit Problem, 2014