Summary

- We address a recommender system setting without prior rating data and constantly-changing items (e.g. news articles).
- Previous work (Cesa-Bianchi'13) show that, in addition to features, sharing information across users improves performance. But previous algorithms are not scalable. Other previous approaches cluster nodes (Gentile'14), but lose personalization.
- We show how to scale to large graphs by making a connection to Gaussian Markov Random Fields (GMRFs).
- We also prove regret bounds and give a heuristic to learn the graph on the fly.

Gang of Bandits model (Cesa-Bianchi'13)

- **Input:**
  - Recommender system (RS) with no past rating data or user meta-data.
  - Each item \( j \) can be described by a set of features \( \mathbf{x}_j \).
  - RS has an associated network (with Laplacian \( L \)).
- **Aim:** Use contextual bandits to trade-off exploration (learn users’ preferences) and exploitation (recommend items which the user likes) and the associated network to share information between users and improve recommendations.
- **Assumptions:**
  - Linear generative model: \( r_j = \langle w, \mathbf{x}_j \rangle + \theta_j \).
  - Homophily: Users connected in the network have similar preferences.
- **Objective:** Minimize regret \( R(T) = \sum_{t=1}^{T} \max_{j \in \mathcal{C}_t} \langle \mathbf{x}_j, \mathbf{x}_t \rangle - \langle \mathbf{w}_t, \mathbf{x}_t \rangle \).
- **Mean estimation:** \( \mathbf{w}_t = \argmin_{\mathbf{w}} \frac{1}{n} \sum_{t=1}^{T} \sum_{d \in \mathcal{D}, \mathbf{x}_d} \langle \mathbf{x}_d, \mathbf{x}_t \rangle^2 + \lambda \mathbf{w}^T L \mathbf{w} \).

Scaling up Gang of Bandits

- **Basic Idea:** Mean estimation is equivalent to MAP estimation in a GMRF.
- Likelihood: \( r_j \sim N\left(\langle \mathbf{w}, \mathbf{x}_j \rangle, \sigma^2\right) \).
- Prior: \( \mathbf{w} \sim N(0, (L \otimes L^{-1})) \).
- Posterior: \( N\left(\hat{\mathbf{w}}, \Sigma^{-1}\right) \) such that \( \hat{\mathbf{w}} = \frac{1}{\lambda} \Sigma^{-1} \mathbf{b} \), with \( \mathbf{b} = \mathbf{X}^T \mathbf{r} \).
- Solve linear system using conjugate gradient in \( O(nt^2 + d \cdot \text{nnz}(L)) \) time and \( O(dt^2 + \text{nnz}(L)) \) space.

Algorithms

- **UCB**
  - Algorithm: Pick \( j_t = \argmax_{j \in \mathcal{C}_t} \left\langle \mathbf{w}_t, \mathbf{x}_j \right\rangle + \sqrt{\left(2 \ln n \right) / \left(2n \right)} \).
  - Regret: \( R(T) = O\left(\sqrt{\left(\ln n\right) T} \right) \).
- **Epoch Greedy**
  - Algorithm: Explicitly separate exploration and exploitation rounds. Exploration - Pick a random item. Exploitation - Pick \( j_t = \argmax_{j \in \mathcal{C}_t} \left\langle \mathbf{w}_t, \mathbf{x}_j \right\rangle \).
  - Regret: \( R(T) = O\left(\sqrt{\left(\ln n\right) T^2} \right) \).
- **Large Scale Thompson Sampling**
  - Algorithm: Obtain sample from posterior i.e. \( \hat{\mathbf{w}}_t \sim N(\mathbf{w}_t, \Sigma^{-1}) \). Pick \( j_t = \argmax_{j \in \mathcal{C}_t} \left\langle \hat{\mathbf{w}}_t, \mathbf{x}_j \right\rangle \).
  - Regret: \( R(T) = O\left(\frac{\sqrt{T} \ln n}{\sqrt{p}} \right) \).
- **Naive Sampling**
  - Using Cholesky factorization. Requires \( O(n^2d^2) \) computation.
- **Proposed Sampling**
  - To obtain unbiased sample from a GMRF (Papandreou’10), solve \( \mathbf{H}_t \mathbf{w}_t = (L \otimes L) \mathbf{w}_0 + \mathbf{X}^T \mathbf{r} \).
  - Same computational complexity as MAP estimation.

Experiments

- **Graph Based:** G-EG, GOBLIN (Cesa-Bianchi’13), GOBLIN++ (scalable GOBLIN), G-TS
- **Baselines:** No sharing: EG-IND, LINUCB-IND, TS-IND; No personalization: LINUCB-SIN; Clustering: CLUB
- **Scalability:**
  - If the graph is not available, write a joint minimization w.r.t \( \mathbf{w} \) and precision matrix \( V_t \).
  - Variants: L-EG: Learning starting from empty graph; U-EG: Updating starting from given graph

Learning the graph on the fly

- **If the graph is not available, write a joint minimization w.r.t \( \mathbf{w} \) and precision matrix \( V_t \).
  - Variants: L-EG: Learning starting from empty graph; U-EG: Updating starting from given graph

Future Work

- Tighten the regret bound for Thompson Sampling and prove regret bounds for the learning the graph variant.