Fast and Memory-Efficient Significant Pattern Mining via Permutation Testing (KDD2015)

Felipe Linares-López, Mahito Sugiyama, Laetitia Papaxanthos, Karsten Borgwardt (ETH Zürich)

杉山 麹人 (大阪大学，さきがけ研究者)
Summary

• Computing \( p \)-values in (supervised) pattern mining
  – Itemsets, subgraphs, ...
  – Significant pattern mining

• Challenge: How to correct for multiple testing?
  – Control the false positive rate of resulting patterns
  – Number of patterns are massive (more than billions!)

• We propose a new method “Westfall-Young light”
  – Empirically estimate the null distribution of pattern frequencies in each class via permutations
  – Embed “permutation + \( p \)-value computation” into pattern mining
## Itemset Mining (GWAS)

<table>
<thead>
<tr>
<th>Case</th>
<th>Items (SNPs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample 1:</td>
<td>0 0 1 1 0 0 1 1 1 0 0 0 0 1 1 0 0 1 1 1 0</td>
</tr>
<tr>
<td>Sample 2:</td>
<td>1 1 0 1 1 0 1 1 1 0 0 0 0 0 1 0 1 0 1 0 0 0</td>
</tr>
<tr>
<td>Sample 3:</td>
<td>1 0 1 1 0 0 1 1 1 0 0 0 0 1 1 0 0 0 0 0 1</td>
</tr>
<tr>
<td>Sample 4:</td>
<td>1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 1 1</td>
</tr>
<tr>
<td>Sample 5:</td>
<td>1 1 0 1 1 0 1 1 1 0 0 1 0 1 0 1 0 1 0 0 0 0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Control</th>
<th>Items (SNPs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample 6:</td>
<td>0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 0 1 1 1 0 0 0</td>
</tr>
<tr>
<td>Sample 7:</td>
<td>0 1 0 1 1 0 1 1 0 0 0 0 0 1 0 0 0 1 0 1 0 0</td>
</tr>
<tr>
<td>Sample 8:</td>
<td>1 0 1 1 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0 0 0</td>
</tr>
<tr>
<td>Sample 9:</td>
<td>1 1 0 0 1 0 0 1 0 1 0 1 0 1 0 0 0 1 0 1 0 1</td>
</tr>
</tbody>
</table>

3/24
## Itemset Mining (GWAS)

<table>
<thead>
<tr>
<th>Case</th>
<th>Sample 1:</th>
<th>Sample 2:</th>
<th>Sample 3:</th>
<th>Sample 4:</th>
<th>Sample 5:</th>
<th>Sample 6:</th>
<th>Sample 7:</th>
<th>Sample 8:</th>
<th>Sample 9:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0 0 1 1 0 0</td>
<td>1 1 0 1 1 0</td>
<td>1 0 1 1 0 0</td>
<td>1 1 0 1 1 0</td>
<td>1 1 0 1 1 0</td>
<td>0 0 1 1 0 0</td>
<td>0 1 0 1 1 0</td>
<td>1 0 1 1 0 0</td>
<td>1 1 0 0 1 0</td>
</tr>
<tr>
<td></td>
<td><strong>1 1</strong></td>
<td><strong>1 1</strong></td>
<td><strong>1 1</strong></td>
<td><strong>1 1</strong></td>
<td><strong>1 1</strong></td>
<td><strong>1 1</strong></td>
<td><strong>1 1</strong></td>
<td><strong>1 1</strong></td>
<td><strong>1 1</strong></td>
</tr>
<tr>
<td></td>
<td>0 0 0 0 1 1</td>
<td>0 0 0 0 1 1</td>
<td>0 0 0 0 1 1</td>
<td>0 0 0 0 1 1</td>
<td>0 0 0 0 1 1</td>
<td>0 0 0 0 1 1</td>
<td>0 0 0 0 1 1</td>
<td>0 0 0 0 1 1</td>
<td>0 0 0 0 1 1</td>
</tr>
<tr>
<td>Control</td>
<td>Sample 6:</td>
<td>Sample 7:</td>
<td>Sample 8:</td>
<td>Sample 9:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 0 1 1 0 0</td>
<td>0 1 0 1 1 0</td>
<td>1 0 1 1 0 0</td>
<td>1 1 0 0 1 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 1 1 0 0 0</td>
<td>1 1 0 0 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 1 1 0 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 1 1 0 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 1 1 0 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 1 1 0 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 1 1 0 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 1 1 0 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 1 1 0 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td>0 1 0 1 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Subgraph Mining (Drag Discovery)

Active

Inactive
Subgraph Mining (Drag Discovery)

Active

Inactive

6/24
Timeline

(Itemset) (Speed-up)
LAMP [Terada et al. PNAS 2013] ▶ LAMP ver.2 [Minato et al. ECML 2014]
(Subgraph, Speed-up)
Testability on subgraphs [Sugiyama et al. SDM 2015]

(Speed-up + Permutation test + Itemset + Subgraph)
Westfall-Young light [Llinares-López et al. KDD 2015]
(Interval)
FAIS (FAIS-WY) [Llinares-López et al. ISMB 2015]

FastWY [Terada et al. BIBM 2013] (Permutation test)
## Itemset Mining (GWAS)

<table>
<thead>
<tr>
<th>Case</th>
<th>Sample 1:</th>
<th>Sample 2:</th>
<th>Sample 3:</th>
<th>Sample 4:</th>
<th>Sample 5:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0 0 1 1 0 0 1 1 1 0 0 0 0 1 1 0 0 1 1 1 0</td>
<td>1 1 0 1 1 0 1 1 1 0 0 0 0 0 1 0 1 0 1 0 0</td>
<td>1 0 1 1 0 0 1 1 1 0 0 0 1 1 0 0 0 0 0 0 1</td>
<td>1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 0 1 0 0 1 1</td>
<td>1 1 0 1 1 0 1 1 1 0 0 0 1 0 1 0 1 0 0 0 0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Control</th>
<th>Sample 6:</th>
<th>Sample 7:</th>
<th>Sample 8:</th>
<th>Sample 9:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0 0 1 1 0 0 0 1 1 0 0 0 1 0 1 1 1 0 0 0 0</td>
<td>0 1 0 1 1 0 1 1 0 0 0 0 1 0 0 0 1 0 1 0 0</td>
<td>1 0 1 1 0 0 1 0 1 0 1 0 0 0 0 1 0 1 0 0 0 0</td>
<td>1 1 0 0 1 0 0 1 0 1 0 1 0 0 0 1 0 1 0 1 0 1</td>
</tr>
</tbody>
</table>

8/24
Testing the Independence of Pattern

- Given two sets of transactions $\mathcal{X}, \mathcal{X'}$
  - $|\mathcal{X}| = n$, $|\mathcal{X'}| = n'$ ($n \leq n'$)

- The *p*-value of each pattern (itemset) $H$ is determined by the Fisher’s exact test
  - $x = |\{ X \in \mathcal{X} \mid H \subseteq X \}|$

<table>
<thead>
<tr>
<th></th>
<th>Occ.</th>
<th>Non-occ.</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{X}$</td>
<td>$x$</td>
<td>$n - x$</td>
<td>$n$</td>
</tr>
<tr>
<td>$\mathcal{X'}$</td>
<td>$x'$</td>
<td>$n' - x'$</td>
<td>$n'$</td>
</tr>
<tr>
<td>Total</td>
<td>$x + x'$</td>
<td>$(n - x) + (n' - x')$</td>
<td>$n + n'$</td>
</tr>
</tbody>
</table>

\[
\begin{align*}
\text{Occ.} & \quad \text{Non-occ.} & \quad \text{Total} \\
\mathcal{X} & \quad x & \quad n - x & \quad n \\
\mathcal{X}' & \quad x' & \quad n' - x' & \quad n' \\
\text{Total} & \quad x + x' & \quad (n - x) + (n' - x') & \quad n + n' \\
\end{align*}
\]
Fisher’s Exact Test

- The probability \( q(x) \) of obtaining \( x \) and \( x' \) is given by the hypergeometric distribution:

\[
q(x) = \binom{n}{x} \binom{n'}{x'} / \binom{n + n'}{x + x'}
\]

<table>
<thead>
<tr>
<th>Occ.</th>
<th>Non-occ.</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x )</td>
<td>( n - x )</td>
<td>( n )</td>
</tr>
<tr>
<td>( x' )</td>
<td>( n' - x' )</td>
<td>( n' )</td>
</tr>
<tr>
<td>( x + x' )</td>
<td>( (n - x) + (n' - x') )</td>
<td>( n + n' )</td>
</tr>
</tbody>
</table>

\[ p\text{-value} \leq = \max\{0, x + x' - n'\} \leq = \min\{x + x', n\} \]
Multiple Testing Correction

- In each test, \((p\text{-value} < \alpha)\) ⇒ statistically significant

- If we test \(m\) patterns, \(am\) subgraphs are false positives
  - \(\alpha\): Significance level (predetermined by the user)

- Example in itemset mining:
  - There are 100000 items
  - Number of combinations are \(2^{100000}\)
  - Set significance level \(\alpha = 0.01\)
  - Number of false positives: \(0.01 \cdot 2^{100000} = 10^{30101}\)

- FWER: Probability of having more than one false positives among all patterns
  - \(FWER = 1 - (1 - \alpha)^m\) if patterns are independent
Controlling the FWER

- FWER = Pr(FP > 0)
  - FP: Number of false positives

- To achieve FWER = $\alpha$, change the significance level for each test from $\alpha$ to $\delta$
  - $\delta$: corrected significance level
  - $\delta \leq \alpha$

- Objective is to optimize (maximize) $\delta$:
  - $\delta^* = \arg\max_{\delta} \text{FWER}(\delta)$  s.t. $\text{FWER}(\delta) \leq \alpha$
  - FWER(\delta): FWER at corrected significance level $\delta$
    - Cannot be evaluated in closed form
    - Bonferroni correction is popular: $\delta_{\text{Bon}}^* = \alpha / m$
Westfall-Young Permutation

1. Randomly permute class labels

2. Compute $p$-values for all patterns using the permuted class labels

3. Find the minimum $p$-value $p_{\min}$ among them
   - $\text{FP} > 0 \iff p_{\min} < \delta$
     - FP: Number of false positives

4. Repeat steps 1 to 3 $h$ times and obtain $p_{\min}^1, p_{\min}^2, \ldots, p_{\min}^h$
   - $\text{FWER}(\delta) \approx \{ i : p_{\min}^i \leq \delta \} / h$

5. $\delta^*$ is the $\alpha$-quantile of $p_{\min}^1, p_{\min}^2, \ldots, p_{\min}^h$
Pattern Mining

Transaction data

<table>
<thead>
<tr>
<th>ID</th>
<th>1 2 3 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>2</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>3</td>
<td>0 1 0 1</td>
</tr>
<tr>
<td>4</td>
<td>0 1 1 1</td>
</tr>
</tbody>
</table>

Task:
Find all patterns (sets of features) whose support ≥ 2

Apriori principle:
For a pattern $H$ with support $\sigma$, none of its superset’s support > $\sigma$
“Westfall-Young light”

- Precompute \( h \) permuted labels; \( \sigma \leftarrow 1; p^i_{\text{min}} \leftarrow 1 \)

- **Westfall-Young light** does the following whenever a miner (like LCM) finds a new frequent pattern \( H \):
  - **for** \( i \leftarrow 1 \) **to** \( h \) **do:**
    - \( p^i \leftarrow \) the \( p \)-value of \( H \) for \( i \)th permutation
    - \( p^i_{\text{min}} \leftarrow \min\{p^i_{\text{min}}, p^i\} \)
  - FWER \( \leftarrow |\{ i : p^i_{\text{min}} \leq \Psi(\sigma) \}| / h \)
    // \( \Psi(\sigma) \) is the min. achievable \( p \)-value at \( \sigma \)
  - **while** FWER > \( \alpha \) **do:**
    - \( \sigma \leftarrow \sigma + 1 \)    // \( \sigma \) is the **minimum support**
    - FWER \( \leftarrow |\{ i : p^i_{\text{min}} \leq \Psi(\sigma) \}| / h \)
  - Go children of \( H \)
Minimum Achievable $p$-value

- $\Psi(\sigma)$ is the minimum achievable $p$-value of a pattern $H$ when its support $\sigma = |\{ X \in \mathcal{X} \cup \mathcal{X}' \mid H \subseteq X \}|$

- $\Psi(\sigma) = \min\{ p(x) \mid x_{\min} \leq x \leq x_{\max} \}$
  - $x_{\min} = \max\{0, \sigma - n'\}$, $x_{\max} = \min\{\sigma, n\}$

<table>
<thead>
<tr>
<th></th>
<th>Occ.</th>
<th>Non-occ.</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{X}$</td>
<td>$x$</td>
<td>$n - x$</td>
<td>$n$</td>
</tr>
<tr>
<td>$\mathcal{X}'$</td>
<td>$x'$</td>
<td>$n' - x'$</td>
<td>$n'$</td>
</tr>
<tr>
<td>Total</td>
<td>$\sigma$</td>
<td>$(n - x) + (n' - x')$</td>
<td>$n + n'$</td>
</tr>
</tbody>
</table>

Probability $q(X) = \max\{0, f(H) - n'\}$

$X_{\text{min}} = \min\{f(H), n\}$

Minimum achievable $p$-value
Experiments

• Compare runtime and memory usage of FastWY and Westfall-Young light
  – We reimplemented FastWY in C (x1000 speedup, x10 less memory compared to the Python version)

• Datasets:
  – 20 itemset mining datasets (LCM v3 used as a miner)
  – 12 graph mining datasets (Gaston used as a miner)

• All experiments run on a single 2.5 GHz Intel Xeon CPU with 256 GB of memory
Runtime in Itemset Mining

Execution time (s)

FastWY
Westfall-Young light
Runtime in Subgraph Mining

Execution time (s)

- PTC(MR)
- PTC(FR)
- PTC(MM)
- PTC(FM)
- MUTAG
- ENZYMES
- D&D
- NCI1
- NCI41
- NCI109
- NCI167
- NCI220

- FastWY
- Westfall-Young light
Peak Memory in Subgraph Mining

Memory usage (MB)

FastWY
Westfall-Young light
Gaston

Memory usage (MB)
FWER in Itemset Mining

![Graphs showing FWER and FWER$_{LAMP}$ for BmsWebview and T40I10D100K datasets.](image-url)
FWER in Subgraph Mining

ENZYMES

NCI220

FWER vs. Number of permutations
Conclusion

• Westfall-Young light

• The area of significant pattern mining is emerging
  – Find statistically significant combinatorial patterns while controlling false positive rate

• Pattern mining, a classical yet central topic in data mining, can be enriched by introducing statistical assessment
  – Can be applied in scientific fields such as biology