

Supplementary Materials: One-Stage Fair Multi-View Spectral Clustering

Anonymous Authors

1 APPENDIX A: DETAILED PSEUDO-CODE OF FMSC

Algorithm 1 shows the detailed algorithm of our proposed FMSC.

Algorithm 1 Fair Multi-View Spectral Clustering

Input: Multi-view data $\mathcal{X} = \{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \dots, \mathbf{X}^{(v)}\}$, protected groups $\mathcal{G}_1, \dots, \mathcal{G}_T$, fairness hyper-parameter λ .

- 1: Construct the Laplacian matrix $\mathbf{L}^{(i)}$ for each view. Construct the one-hot protected group matrix \mathbf{P} .
- 2: Initialize $\alpha_i = \frac{1}{v}$ and \mathbf{Y} by minimizing $\text{Tr}\left(\mathbf{Y}^T \left(\sum_{i=1}^v \alpha_i^2 \mathbf{L}^{(i)}\right) \mathbf{Y} (\mathbf{Y}^T \mathbf{Y})^{-1}\right)$.
- 3: **repeat**
- 4: // Update \mathbf{Y}
- 5: **repeat**
- 6: Compute $\mathbf{Y}_{\cdot k}^T \mathbf{B} \mathbf{Y}_{\cdot k}$ and $\mathbf{Y}_{\cdot k}^T \mathbf{Y}_{\cdot k}$, for $k = 1, \dots, c$.
- 7: **for** $i = 1$ to n **do**
- 8: Let m be the location of 1 in the i -th row of current \mathbf{Y} .
- 9: Calculate $\mathcal{L}(\mathbf{Y}^{(s)})$ by Eq.(16) or Eq.(17), for $s = 1, \dots, c$.
- 10: Obtain $s^* = \text{argmin} \mathcal{L}(\mathbf{Y}^{(s)})$.
- 11: **if** $s^* \neq m$ **then**
- 12: Update $\mathbf{Y}_{\cdot m}^T \mathbf{B} \mathbf{Y}_{\cdot m}$, $\mathbf{Y}_{\cdot m}^T \mathbf{Y}_{\cdot m}$, $\mathbf{Y}_{\cdot s^*}^T \mathbf{B} \mathbf{Y}_{\cdot s^*}$, and $\mathbf{Y}_{\cdot s^*}^T \mathbf{Y}_{\cdot s^*}$ by Eqs.(18) and (19).
- 13: Set $\mathbf{Y}(i, m) = 0$, $\mathbf{Y}(i, s^*) = 1$.
- 14: **end if**
- 15: **end for**
- 16: **until** Converges
- 17: // Update α
- 18: Update α by Eq.(21).
- 19: **until** Converges
- Output:** The final partition matrix \mathbf{Y} .

2 APPENDIX B: CONVERGENCE CURVES

Figure 1 shows the convergence curves of our method on all data sets. The results show that our method converges very fast, which often converges within 10 iterations.

3 APPENDIX C: RUNNING TIME COMPARISON

Figure 2 shows the running time of all multi-view clustering methods on all data sets. We report the logarithm of the time (in seconds) for better comparison. From Figure 2, we can see that our method is faster than many state-of-the-art methods, which well demonstrates the efficiency of our proposed method.

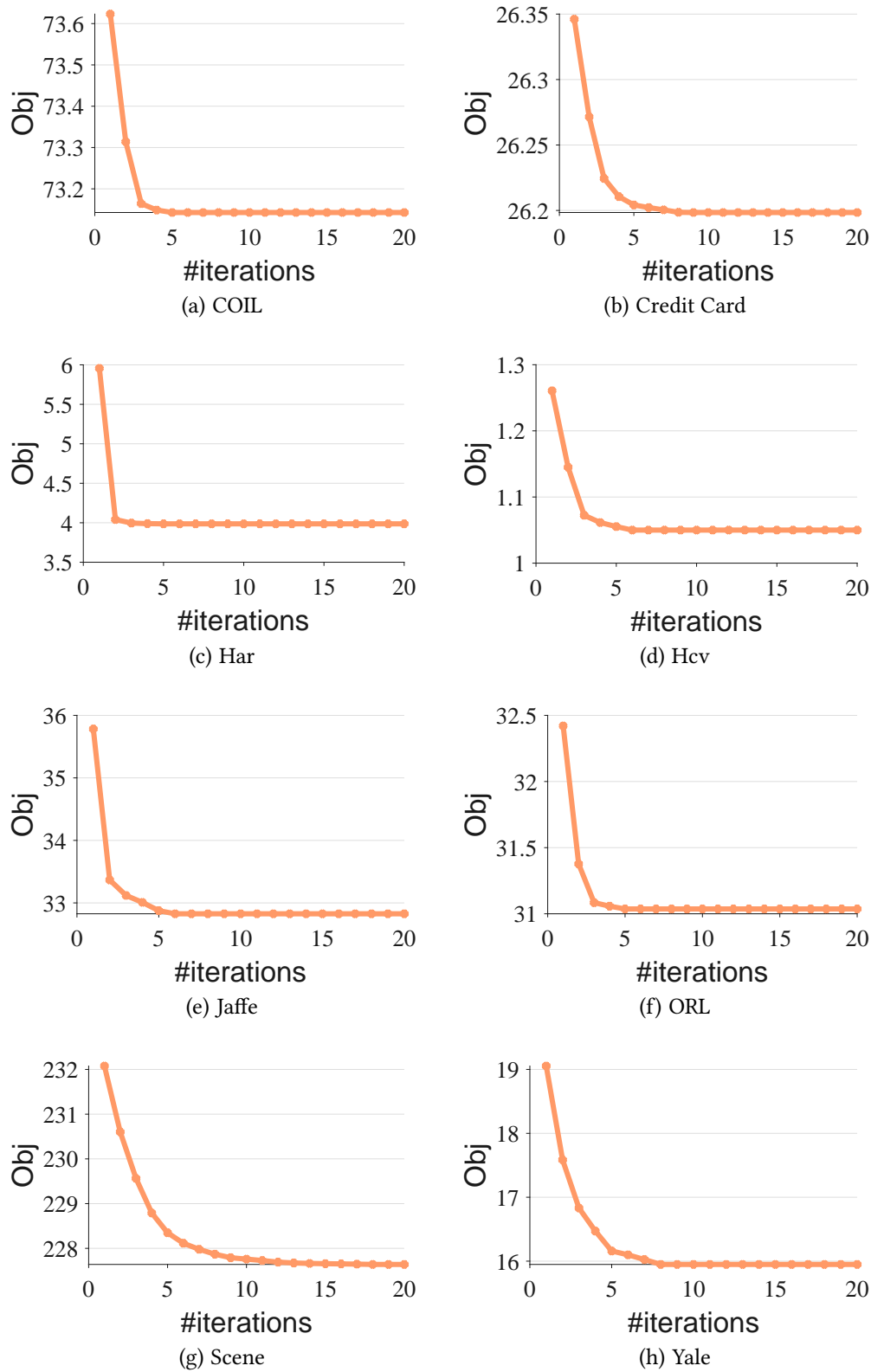


Figure 1: Convergence curves on all data sets

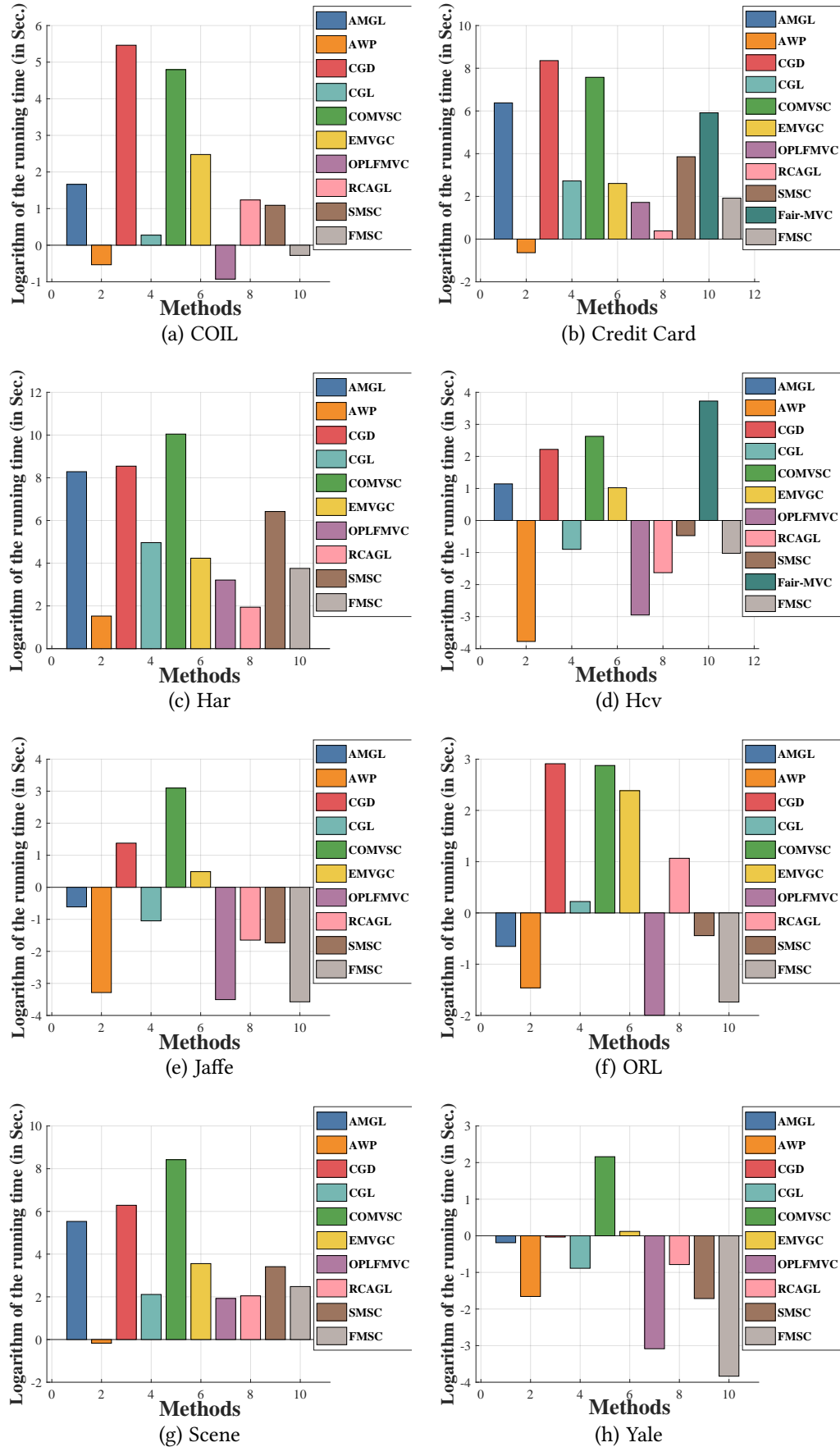


Figure 2: Running time of all multi-view clustering methods on all data sets