Distributionally Robust Learning of Sum-Product Networks

Rohith Peddi[1] **Vibhav Gogate**[1]

[1]The University of Texas at Dallas

Abstract

Sum-Product networks (SPNs) are generative probabilistic models that use a deep architecture comprised of alternating layers of sum and product nodes to compactly represent a high-dimensional joint probability distribution. In this paper, we consider the problem of learning robust SPNs from the lens of distributionally robust optimization (DRO) under the Wasserstein metric. We show that SPNs learned by maximizing likelihood exhibit poor performance when data is subject to noise/corruptions. To address this issue, we construct probabilistic uncertainty sets and leverage the tractability of SPNs to efficiently learn distributionally robust SPNs. We show our proposed approach's efficacy on a collection of benchmark datasets.

1 INTRODUCTION

Tractable probabilistic models (TPMs) that offer guarantees on efficient computation of probabilistic inference queries encompasses representations like Sum-Product Networks (SPNs) [12], Arithmetic circuits (ACS), Cutset Networks (CNets) [14], Probabilistic Sentential Decision Diagrams (PSDDs) [6]. The robustness of TPMs to corruptions induced by measurement errors, adversaries, or noise is an under-explored area. This paper explores approaches for robust learning of Sum-Product Networks (SPNs) when data is subject to corruptions/distribution shifts from the lens of distributionally robust optimization(DRO). DRO approach assumes the true data generating distribution is unknown and lies in an uncertainty set of probability distributions constructed based on the observed empirical distribution. In this approach, we hedge against the uncertainty in the observed data distribution by taking a worst-case approach.

Contributions. We note that estimating the worst-case distribution exactly in an uncertainty set constructed based on the optimal transport discrepancy measure is computationally infeasible. We propose two approaches to efficiently approximate worst-case distribution and learn distributionally robust SPNs. Empirically, we evaluate the proposed approaches on benchmark Twenty datasets for density estimation task on SPNs.

- In our first approach, we propose constructing a restricted uncertainty set by randomly sampling $K-$distributions from the original uncertainty set. Our experiments observed that the robustness of learned models increased initially with the uncertainty set's size but plateaued after a point.
- Our second approach proposes a fast gradient-based method to approximate the worst-case distribution in the uncertainty set.

2 BACKGROUND

Notation We use upper case letters (X) for datasets, lower case letters (x) for individual samples and upper case calligraphic letters ($\mathcal{P}$) for distributions. We consider dataset X as an ensemble of n individual samples $x_i \in \mathscr{X}$. In this paper, we concentrate on binary datasets for the sake of simplicity but our methods can be easily extended to discrete datasets. The space $\mathscr{X}$ consists of d-dimensional $0/1$ vectors i,e $\mathscr{X} = \{0, 1\}^d$. We denote probability mass function parameterized by θ at x using $f_\theta(x)$. We denote corruptions of individual samples x_i using Δx_i where Δx_i is a d-dimensional $0/1$ vector (or mask), 1 indicates that the particular dimension is corrupted and 0 indicates that it is not. Given a dataset X, we denote an ensemble of corruptions/distribution shifts (or masks) by ΔX, where each $x_i \in X$ is associated with a mask $\Delta x_i \in \Delta X$, namely $\Delta X = [\Delta x_1, \Delta x_2, \ldots, \Delta x_n]^T$. We denote an XOR operation over two binary vectors using $\oplus$. We denote the corrupted dataset by $X \oplus \Delta X = [x_1 \oplus \Delta x_1, \ldots, x_n \oplus \Delta x_n]^T$.

Note that in the remainder of the paper, unless explicitly stated, whenever we denote a distribution by $\mathcal{Q}_X$ we imply

Accepted for the 5th Workshop on Tractable Probabilistic Modeling at UAI (TPM 2022).

that $\mathcal{Q}_X = \frac{1}{n}\sum_{i=1}^{n} \delta_{x_i}$ where $X = [x_1, x_2, \ldots, x_n]^T$.

Sum-Product Networks [12] are rooted directed acyclic graphs with variables as terminal nodes, weighted sums (convex combinations) and products of probability distributions as non-terminal nodes. SPNs use a deep architecture comprised of alternating layers of sum and product nodes and can be roughly described as deep mixture models [9]. They compactly represent high dimensional joint probability distributions while ensuring numerous inference queries can be answered in time and space that scales polynomially with the model's size. Precisely, SPNs admit linear time computation of exact likelihood and marginal probability distributions over a small subset of variables given evidence. We can learn the parameters θ of a given SPN generatively by maximizing the log-likelihood function or equivalently minimizing the cross-entropy $H(\mathcal{P}_X, f_\theta(x))$ where $\mathcal{P}_X$ denotes the empirical distribution.

$$\max_\theta \sum_{i=1}^{n} \log\left(f_\theta\left(x_i\right)\right) \equiv \min_\theta \left(- \mathop{\mathbb{E}}_{x \sim \mathcal{P}_X} [\log f_\theta(x)] \right) \quad (1)$$

For SPNs, the objective in Eq. (1) is non-convex, and we reach a local minimum when iterative algorithms such as gradient descent or their stochastic versions are employed. In SPNs, gradient of each parameter θ_i can be computed in polynomial time (cf. [9], [3]).

Distributionally Robust Optimization (DRO) as a learning paradigm has recently received significant traction due to its connections with regularization, generalization and robustness. DRO treats data uncertainty in a probabilistic way and finds the optimal solution for a learning problem under a probabilistic uncertainty set constructed from the observed data and characterized by the distributional knowledge about the true data-generating distribution. Uncertainty sets used in the DRO literature can be broadly categorized into two types; moment-based and discrepancy-based [13]. Moment-based uncertainty sets encompass distributions that satisfy specific properties on moments and discrepancy based uncertainty sets encompass distributions close to a nominal distribution under some discrepancy measure. Typical choices for discrepancy measures include optimal transport discrepancy, $\phi-$divergences and total variation distance.

Related work on DRO [17] showed that distributionally robust adversaries are stronger than point-wise adversaries and proposed an approach based on block coordinate descent to estimate worst-case distribution. [15] proposed Principled Adversarial Training to combat adversarial examples in deep learning using DRO. [4] proposed Adversarial Graphical Models for structured prediction problems using DRO under Moment Matching constraints. [5] proposed Distributionally Robust Bayesian Optimization under Maximum Mean Discrepancy based uncertainty sets to study robustness to distribution shifts. For many simpler tasks such as linear regression, [1], performing DRO is equivalent to Empirical Risk Minimization (ERM) with a specific regularization term.

Robust Maximum Likelihood Estimators [8] proposed a systematic approach to learn robust estimators under the assumption that we observe corrupted samples x_i^{obs} and the true samples x_i^{true} generated from true data generating distribution $\mathcal{P}_{X^{\text{true}}}$ remain unobserved. They estimate parameters θ by maximizing the log-likelihood function of true samples x_i^{true} (see eq.2) or equivalently minimizing the cross entropy $H(\mathcal{P}_{X^{\text{true}}}, f_\theta(x))$

$$\min_\theta \left(- \mathop{\mathbb{E}}_{x^{\text{true}} \sim \mathcal{P}_{X^{\text{true}}}} [\log f_\theta(x^{\text{true}})] \right) \quad (2)$$

3 PROBLEM FORMULATION

3.1 DISTRIBUTIONALLY ROBUST ESTIMATORS

We seek to estimate parameters θ in the presence of a stronger adversary that is not limited to corrupting individual samples but can corrupt the data distribution by moving it within an uncertainty set $\mathscr{U}$. In a real-world setting, we are oblivious to these corruptions and assume that the true unobserved distribution $\mathcal{P}_{X^{\text{true}}}$ lies in an uncertainty set $\mathscr{U}$ constructed based on the observed distribution $\mathcal{P}_{X^{\text{obs}}}$. Therefore, we estimate parameters θ by minimizing the cross-entropy between $f_\theta(x)$ and the worst-case distribution in the uncertainty set $\mathscr{U}$. Formally, the robust parameter estimation task under the assumption that the structure of the SPN is fixed and provided as an input is given by

$$\min_\theta \max_{\mathcal{Q} \in \mathscr{U}} \left(- \mathop{\mathbb{E}}_{x \sim \mathcal{Q}} [\log f_\theta(x)] \right) \quad (3)$$

Construction of uncertainty set $\mathscr{U}$ In the above optimization problem, the choice of uncertainty set describes our knowledge about the corruptions/distribution shifts. Thus we learn estimators hedging against distributions encompassed by the uncertainty set. So learning DRO estimators using an uncertainty set $\mathscr{U}$ constructed based on the observed distribution $\mathcal{P}_{X^{\text{obs}}}$ not only hedges against adversarially corrupted distributions but also aim toward generalization. This paper focuses on uncertainty sets constructed using the Optimal Transport discrepancy measure.

Remark 1. *Standard maximum likelihood estimation is a special case of DRO where the uncertainty set is a singleton with only observed distribution $\mathcal{P}_{X^{obs}}$ as its element.*

Optimal Transport Discrepancy Measure Let $\mathcal{P}_X = \sum_{i=1}^m p_i \delta_{x_i},\ \mathcal{Q}_Y = \sum_{j=1}^n q_i \delta_{y_j}$ be two discrete probability distributions where $\sum_{i=1}^m p_i = \sum_{j=1}^n q_j = 1$ and $\forall ij\ \ x_i, y_j \in \mathscr{X}$. Then, optimal transport distance or $t-$Wasserstein distance [16] between $\mathcal{P}_X, \mathcal{Q}_Y$ using a non-negative cost metric $c : \mathscr{X} \times \mathscr{X} \to \mathbb{R}^+$ is given by

$$\begin{aligned} &\mathrm{OT}(\mathcal{P}_X, \mathcal{Q}_Y, c) = [W_{t,c}(\mathcal{P}_X, \mathcal{Q}_Y)]^t \\ &= \begin{cases} \min_{\pi \in \mathbb{R}^{m\times n}} & \sum_{ij} \pi_{ij} \times c^t(x_i, y_j) \\ \text{s.t} & \sum_i \pi_{ij} = q_j \ \forall j \in \{1,\dots,n\} \\ & \sum_j \pi_{ij} = p_i \ \forall i \in \{1,\dots,m\} \\ & \pi_{ij} \geq 0 \ \forall i,j \end{cases} \end{aligned} \tag{4}$$

Here, π_{ij} denotes probability mass moved from x_i to y_j. We use L_1 distance as our cost metric i,e $c(x_i, y_j) = \|x_i - y_j\|_1$ and 1-Wasserstein distance $W_c(\mathcal{P}_X, \mathcal{Q}_Y)$ in all our formulations present in subsequent sections.

3.2 WASSERSTEIN DISTRIBUTIONALLY ROBUST ESTIMATORS

The robust parameter estimation task under the assumption that the true distribution $\mathcal{P}_{X^{\text{true}}}$ lies in the wasserstein uncertainty set $\mathscr{U}$ constructed based on the observed distribution $\mathcal{P}_{X^{\text{obs}}}$ can be expressed as

$$\begin{aligned} &\min_{\theta} \max_{\mathcal{Q} \in \mathscr{U}} \mathop{\mathbb{E}}_{x \sim Q} [-\log f_\theta(x)] \\ &\text{s.t } \mathscr{U} = \{\mathcal{Q} : W_c(\mathcal{Q}, \mathcal{P}_{X^{\text{obs}}}) \leq \epsilon\} \end{aligned} \tag{5}$$

In 5, the inner maximization problem requires estimation of the worst-case distribution $\mathcal{Q}^* \in \mathscr{U}$. Without any further knowledge about the worst-case distribution or additional assumptions on the wasserstein uncertainty set the inner maximization problem remains a 2^d- dimensional constrained optimization problem. In an attempt to make the inner maximization problem computationally tractable we impose constraints on the uncertainty set $\mathscr{U}$ thereby restricting the number of distributions contained in $\mathscr{U}$. Specifically, we construct a restricted uncertainty set $\mathscr{U}_r$ that encapsulates all the distributions $\mathcal{Q}_X$ which satisfy $W_c(\mathcal{Q}_X, \mathcal{P}_{X^{\text{obs}}}) \leq \epsilon$ where $X = X^{\text{obs}} \oplus \Delta X$. We note that as the size of the X^{obs} increases, $\mathscr{U}_r \to \mathscr{U}$. The transformed objective after using the restricted uncertainty set $\mathscr{U}_r$ can be compactly expressed as

$$\begin{aligned} &\min_{\theta} \max_{\mathcal{Q}_X \in \mathscr{U}_r} \mathop{\mathbb{E}}_{x \sim \mathcal{Q}_X} [-\log f_\theta(x)] \\ &\mathscr{U}_r = \left\{ \mathcal{Q}_X \;\middle|\; \begin{array}{l} X = X^{\text{obs}} \oplus \Delta X \text{ and } \|\Delta X\|_1 \leq \rho, \\ \mathcal{Q}_X = \mathcal{Q}_{X^{\text{obs}} \oplus \Delta X} = \frac{1}{n} \sum_i^n \delta_{x_i^{\text{obs}} \oplus \Delta x_i}, \\ W_c(\mathcal{Q}_X, \mathcal{P}_{X^{\text{obs}}}) \leq \epsilon \end{array} \right\} \end{aligned} \tag{6}$$

In other words, for every perturbed dataset X obtained by jointly perturbing the observed dataset X^{obs} with ΔX under a budget constraint given by $\|\Delta X\|_1 \leq \rho$, we include the distribution $\mathcal{Q}_X$ in the uncertainty set $\mathscr{U}_r$. For min-max problems such as (6), [2] has shown that we always reach local optimal solutions when the inner maximization problem is solved optimally. Formally, we can show that:

Proposition 1. *[2] Let $h(x) = -\log f_\theta(x)$ and*

$$\mathcal{Q}_X^*(\theta) = \mathop{\arg\max}_{\mathcal{Q}_X \in \mathscr{U}_r} \mathop{\mathbb{E}}_{x \sim \mathcal{Q}_X} [h(x)]$$

then

$$\nabla_\theta \max_{\mathcal{Q}_X \in \mathscr{U}_r} \mathop{\mathbb{E}}_{x \sim \mathcal{Q}_X} [h(x)] \Big|_{\theta = \theta_t} = \nabla_\theta \mathop{\mathbb{E}}_{x^* \sim \mathcal{Q}_X^*(\theta_t)} [h(x^*)] \Big|_{\theta = \theta_t} \tag{7}$$

To put it in perspective, if we can find a solution $\mathcal{Q}_X^*(\theta_t)$ to the inner maximization problem, then the gradient of the objective at $\theta = \theta_t$ equals the gradient of the cross-entropy $H(\mathcal{Q}_X^*(\theta_t), f_\theta(x))$. In SPNs, this gradient can be computed efficiently in time that scales linearly with the size of the model [9].

4 APPROACH

Efficient estimation of robust estimators depends on the efficiency and practicality of the algorithm used to find the solution to the inner maximization problem. Considering X a $d-$dimensional binary dataset of size n i,e $|X| = n$, the size of the uncertainty set $\mathscr{U}_r$ remains exponential with $O(2^{n \times d})$ distributions contained in it. We propose two approaches for efficient estimation of the inner maximization problem; sampling-based and gradient-based.

Lemma 4.1. *Let X be any dataset of size n obtained by jointly perturbing X^{obs} with ΔX where $\|\Delta X\|_1 \leq n \times \epsilon$ then $W_c(\mathcal{Q}_X, \mathcal{P}_{X^{obs}}) \leq \epsilon$*

4.1 SAMPLING METHOD

In this approach, we construct a new uncertainty set $\mathscr{U}_{rs}$ by randomly sampling K probability distributions from the uncertainty set $\mathscr{U}_r$. Since every distribution in the uncertainty set $\mathscr{U}_r$ is determined by a perturbed dataset X, randomly sampling K distributions is equivalent to randomly sampling K perturbed datasets $\{X_1, \dots, X_k, \dots, X_K\}$ from $\{X : X = X^{\text{obs}} \oplus \Delta X \text{ and } \|\Delta X\| \leq n \times \epsilon\}$. The uncertainty set $\mathscr{U}_{rs}$ thus constructed as described contains K distributions given by $\{\mathcal{Q}_{X_1}, \dots, \mathcal{Q}_{X_k}, \dots, \mathcal{Q}_{X_K}\}$ and by lemma 4.1 every distribution $\mathcal{Q}_{X_k} \in \mathscr{U}_r$.

The description above results in an algorithm (see Alg 2 in Appendix) where we iteratively solve the inner maximization problem to obtain $\mathcal{Q}_X^* \in \mathscr{U}_{rs}$ and use the distribution $\mathcal{Q}_X^*$ to update parameters of the model.

Table 1: Generative performance: Test set log-likelihood scores of models having latent variables. $\epsilon \in \{1, 2, 3\}$: wasserstein distance thresholds. SPN: SPN trained original training data, SPN−k: SPN trained using sampling based DRO, SPN−g: SPN trained using fast gradient method. $\mathcal{T}$: original test data, $\mathcal{T}_a$: adversarially perturbed $\mathcal{T}$ by SPN, $\mathcal{T}_r$: randomly perturbed $\mathcal{T}$ by SPN.

DATASET	ϵ	$\mathcal{T}$			$\mathcal{T}_a$			$\mathcal{T}_r$		
		SPN	SPN−k	SPN−g	SPN	SPN−k	SPN−g	SPN	SPN−k	SPN−g
plants	1	-14.94	-15.98	-15.49	-21.51	-20.33	-19.53	-21.26	-20.12	-19.48
	3		-16.67	-16.69	-33.84	-26.82	-26.14	-28.36	-24.45	-24.32
	5		-16.89	-18.35	-44.48	-32.69	-31.05	-34.25	-28.37	-28.04
Avg.		-14.94	-16.51	-16.84	-33.28	-26.61	-25.57	-27.96	-24.31	-23.95
netflix	1	-57.61	-57.55	-57.74	-61.0	-60.33	-59.75	-60.44	-59.96	-59.62
	3		-57.53	-58.37	-66.49	-64.69	-63.0	-61.73	-60.93	-60.84
	5		-57.59	-58.84	-70.84	-68.45	-65.6	-62.69	-61.94	-61.78
Avg.		-57.61	-57.56	-58.32	-66.11	-64.49	-62.78	-61.62	-60.94	-60.75
dna	1	-99.51	-98.91	-98.52	-101.21	-100.55	-100.08	-101.15	-100.54	-100.04
	3		-99.58	-99.59	-104.5	-104.5	-103.63	-102.28	-102.36	-102.08
	5		-99.35	-99.63	-107.56	-107.08	-106.15	-103.5	-103.18	-103.13
Avg.		-99.51	-99.28	-99.25	-104.42	-104.04	-103.29	-102.31	-102.03	-101.75
each-movie	1	-58.03	-57.12	-57.08	-64.55	-62.81	-62.75	-64.46	-62.93	-62.75
	3		-57.25	-56.56	-77.43	-72.83	-72.05	-73.04	-70.64	-69.81
	5		-59.45	-58.74	-90.04	-84.57	-82.45	-81.85	-80.2	-78.8
Avg.		-58.03	-57.94	-57.46	-77.34	-73.4	-72.42	-73.12	-71.26	-70.45
bbc	1	-275.22	-275.11	-275.15	-278.59	-278.37	-278.34	-278.52	-278.36	-278.34
	3		-269.57	-275.33	-285.25	-279.78	-284.45	-283.06	-277.57	-282.76
	5		-275.28	-270.66	-291.84	-291.17	-286.29	-287.63	-287.07	-282.89
Avg.		-275.22	-273.32	-273.71	-285.23	-283.11	-283.03	-283.07	-281.0	-281.33

4.2 FAST GRADIENT METHOD

Inspired by the work of [18], we compute gradients with respect to inputs by back-propagating through the learning phase of the SPN and approximately estimate the maximizer $\mathcal{Q}_X^* \in \mathscr{U}_r$. Specifically, we find perturbed dataset $X = X^{\text{obs}} \oplus \Delta X^*$ by estimating the joint perturbation ΔX^* using the gradient information.

Consider X^{obs} as a $d-$dimensional binary dataset of size n represented as $n \times d$ matrix. At iteration t, for every index (u, v) in X^{obs}, we estimate gradient of the objective with respect to the entry x_{uv}^{obs} given by $G_{x_{uv}^{\text{obs}}}^t = \nabla_{x_{uv}^{\text{obs}}}^t (\mathbb{E}_{x \sim \mathcal{Q}_X}[-\log f_\theta(x)])$ and define a score function $S : n \times d \to \mathbb{R}$ given by $S(u, v) = G_{x_{uv}^{\text{obs}}}^t \times (-2 \times x_{uv}^{\text{obs}} + 1)$. The score function can be understood as a tool to flip the sign of the gradient because whenever an entry $x_{uv} = 1$ we shall have a gradient for a change in the negative direction (i,e changing entry x_{uv} from a 1 to 0). After we estimate the score function, we greedily pick $n \times \epsilon$ indices (u, v) with highest scores one at a time maintaining the number of indices picked in each row of the dataset ≤ 5. We construct ΔX^* a $n \times d$ matrix with 1 at all the entries picked above and 0 otherwise and obtain $X = X^{\text{obs}} \oplus \Delta X^*$

The above discussion yields an algorithm (see Alg 3 in Appendix) where we iteratively estimate approximate minimizer $\mathcal{Q}_X^* \in \mathscr{U}_r$ and use the distribution $\mathcal{Q}_X^*$ to update parameters.

5 EXPERIMENTS

In this section, we evaluated the impact of our proposed parameter estimation method on both the generative 1 and predictive performance of SPNs [12] as well as their robustness to adversarial attacks and random noise. Our experiments on SPNs were performed using two open-source implementations: EiNETs [11] and RAT-SPNs [10]. For each dataset, we learned three types of SPNs : 1) SPN learned by minimizing cross entropy with the empirical distribution, 2) SPN−k learned by sampling-based DRO approach and finally 3) SPN−g learned by gradient-based DRO approach. Note that the structure of all SPNs is learned from the original training data. The three SPNs differ from each other in how the parameters are learned; in other words, the structure is constant across all models. We evaluated our method on 20 benchmark datasets that have been used in several experimental evaluations of SPNs [7].

Acknowledgements

This work was supported in part by the DARPA Explainable Artificial Intelligence (XAI) Program under contract number N66001-17-2-4032, by the DARPA Perceptually-enabled Task Guidance (PTG) Program under contract number HR00112220005 and by the National Science Foundation CAREER award IIS-1652835.

References

[1] Ruidi Chen and Ioannis Ch. Paschalidis. Distributionally robust learning, 2021. URL https://arxiv.org/abs/2108.08993.

[2] John M. Danskin. The theory of max-min, with applications. *SIAM Journal on Applied Mathematics*, 14(4):641–664, 1966. doi: 10.1137/0114053. URL https://doi.org/10.1137/0114053.

[3] Professor Adnan Darwiche. *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press, USA, 1st edition, 2009. ISBN 0521884381.

[4] Rizal Fathony, Ashkan Rezaei, Mohammad Ali Bashiri, Xinhua Zhang, and Brian D. Ziebart. Distributionally robust graphical models. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 8354–8365, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/79a3308b13cd31f096d8a4a34f96b66b-Abstract.html.

[5] Johannes Kirschner, Ilija Bogunovic, Stefanie Jegelka, and Andreas Krause. Distributionally robust bayesian optimization. In Silvia Chiappa and Roberto Calandra, editors, *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pages 2174–2184. PMLR, 2020. URL http://proceedings.mlr.press/v108/kirschner20a.html.

[6] Doga Kisa, Guy Van den Broeck, Arthur Choi, and Adnan Darwiche. Probabilistic sentential decision diagrams. In Chitta Baral, Giuseppe De Giacomo, and Thomas Eiter, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, KR 2014, Vienna, Austria, July 20-24, 2014*. AAAI Press, 2014. URL http://www.aaai.org/ocs/index.php/KR/KR14/paper/view/8005.

[7] Daniel Lowd and Jesse Davis. Learning markov network structure with decision trees. In Geoffrey I. Webb, Bing Liu, Chengqi Zhang, Dimitrios Gunopulos, and Xindong Wu, editors, *ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010*, pages 334–343. IEEE Computer Society, 2010. doi: 10.1109/ICDM.2010.128.

[8] Rohith Peddi, Tahrima Rahman, and Vibhav Giridhar Gogate. Robust learning of tractable probabilistic models. In *The 38th Conference on Uncertainty in Artificial Intelligence*, 2022. URL https://openreview.net/forum?id=Sg-UEuUj5xq.

[9] Robert Peharz, Robert Gens, Franz Pernkopf, and Pedro M. Domingos. On the latent variable interpretation in sum-product networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(10):2030–2044, 2017. doi: 10.1109/TPAMI.2016.2618381. URL https://doi.org/10.1109/TPAMI.2016.2618381.

[10] Robert Peharz, Antonio Vergari, Karl Stelzner, Alejandro Molina, Martin Trapp, Xiaoting Shao, Kristian Kersting, and Zoubin Ghahramani. Random sum-product networks: A simple and effective approach to probabilistic deep learning. In Amir Globerson and Ricardo Silva, editors, *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2019, Tel Aviv, Israel, July 22-25, 2019*, volume 115 of *Proceedings of Machine Learning Research*, pages 334–344. AUAI Press, 2019. URL http://proceedings.mlr.press/v115/peharz20a.html.

[11] Robert Peharz, Steven Lang, Antonio Vergari, Karl Stelzner, Alejandro Molina, Martin Trapp, Guy Van den Broeck, Kristian Kersting, and Zoubin Ghahramani. Einsum networks: Fast and scalable learning of tractable probabilistic circuits. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 7563–7574. PMLR, 2020. URL http://proceedings.mlr.press/v119/peharz20a.html.

[12] Hoifung Poon and Pedro M. Domingos. Sum-product networks: A new deep architecture. In Fábio Gagliardi Cozman and Avi Pfeffer, editors, *UAI 2011, Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence, Barcelona, Spain, July 14-17, 2011*, pages 337–346. AUAI Press, 2011. URL https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1&smnu=2&article_id=2194&proceeding_id=27.

[13] Hamed Rahimian and Sanjay Mehrotra. Distributionally robust optimization: A review, 2019. URL https://arxiv.org/abs/1908.05659.

[14] Tahrima Rahman, Prasanna Kothalkar, and Vibhav Gogate. Cutset networks: A simple, tractable, and scalable approach for improving the accuracy of chow-liu trees. In Toon Calders, Floriana Esposito, Eyke Hüllermeier, and Rosa Meo, editors, *Machine*

Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part II, volume 8725 of *Lecture Notes in Computer Science*, pages 630–645. Springer, 2014. doi: 10.1007/978-3-662-44851-9_40. URL `https://doi.org/10.1007/978-3-662-44851-9_40`.

[15] Aman Sinha, Hongseok Namkoong, and John C. Duchi. Certifying some distributional robustness with principled adversarial training. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL `https://openreview.net/forum?id=Hk6kPgZA-`.

[16] Justin M. Solomon. Optimal transport on discrete domains. *ArXiv*, abs/1801.07745, 2018.

[17] Matthew Staib and Stefanie Jegelka. Distributionally robust deep learning as a generalization of adversarial training. 2017.

[18] Daniel Zügner and Stephan Günnemann. Adversarial attacks on graph neural networks via meta learning. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL `https://openreview.net/forum?id=Bylnx209YX`.

A PROOF OF LEMMA 4.1

Proof: Any feasible π and optimal π^* in 4 satisfy

$$\sum_{ij} \pi^*_{ij} c(i,j) \leq \sum_{ij} \pi_{ij} c(i,j) \tag{8}$$

To prove $W_c(\mathcal{Q}_X, \mathcal{P}_{X^{\text{obs}}}) \leq \epsilon$ it suffices to show that there exists a feasible π that satisfies

$$\sum_{ij} \pi_{ij} c(i,j) \leq \epsilon$$

Given, X, X^{obs} are two $d-$dimensional datasets of size n and we obtain X by jointly perturbing X^{obs} with ΔX such that $\|\Delta X\|_1 \leq n \times \epsilon$

Define π s.t

$$\pi_{ij} = \begin{cases} \frac{1}{n} & i = j \\ 0 & otherwise \end{cases}$$

then

$$\begin{aligned} \sum_{ij} \pi_{ij} c(i,j) &= \frac{1}{n} \sum_{i=1}^{n} c(i,i) = \frac{1}{n}(\|\Delta X\|_1) \\ &\leq \frac{1}{n}(n \times \epsilon) = \epsilon \end{aligned} \tag{9}$$

From 8 and 9 we have

$$\sum_{ij} \pi^*_{ij} c(i,j) \leq \sum_{ij} \pi_{ij} c(i,j) \leq \epsilon \tag{10}$$

Thus we can say $W_c(\mathcal{Q}_X, \mathcal{P}_{X^{\text{obs}}}) \leq \epsilon$

B SAMPLING METHOD

Algorithm 1: Sampling a random distribution

```
Input: $d-$ dimensional binary dataset $X^{\text{obs}}$ of size $n$,
       Wasserstein threshold $\epsilon$
Output: A distribution $\mathcal{Q}_X$
begin
    // Maximum number of flips
    $M = \epsilon \times n$
    // Indices to be flipped
    Pick $M$ random indices $(u, v)$ in $X^{\text{obs}}$ such that
      number of chosen indices in each row $u$ is $\leq 5$
    $\Delta X_{uv} = \begin{cases} 1 & \text{(u,v)} \in \text{picked indices} \\ 0 & \text{otherwise} \end{cases}$
    $X = X^{\text{obs}} \oplus \Delta X$
    return $\mathcal{Q}_X$
end
```

Algorithm 2: $K-$ Samples DRO

```
Input: $d-$ dimensional binary dataset $X^{\text{obs}}$ of size $n$,
       an SPN structure having parameters $\theta$,
       Wasserstein threshold $\epsilon$, Size of the uncertainty
       set $\mathscr{U}_{rs}$ given by $K$
Output: An assignment to $\theta$
begin
    Randomly initialize all $\theta_i \in \theta$
    repeat
        Randomly sample $K$ distributions in
          $\epsilon-$wasserstein ball (see Alg. 1)
        // Solve Inner Maximization
        For SPN defined by current parameters $\theta$, find
          the distribution $\mathcal{Q}_X^*$ in the uncertainty set $\mathscr{U}_{rs}$
          that has maximum cross entropy score
        // Outer Minimization
        for m steps do
            Use one step of stochastic gradient descent
              to update parameters $\theta$ using $\mathcal{Q}_X^*$
        end
    until convergence;
    return $\theta$
end
```

C FAST GRADIENT METHOD

Algorithm 3: Gradient based DRO

```
Input: $d-$ dimensional binary dataset $X^{\text{obs}}$ of size $n$,
       an SPN structure having parameters $\theta$,
       Wasserstein threshold $\epsilon$
Output: An assignment to $\theta$
begin
    Randomly initialize all $\theta_i \in \theta$
    $\mathcal{Q}_X = \mathcal{Q}_{X^{\text{obs}}}$
    $t = 1$
    repeat
        // Gradient of the objective
           with respect to inputs
        $G^t_{X^{\text{obs}}} = \nabla^t_{X^{\text{obs}}}(\mathbb{E}_{x \sim \mathcal{Q}_X}[-\log f_\theta(x)])$
        // Score of the inputs
        $S = G^t_{X^{\text{obs}}} \cdot (-2 \cdot X^{\text{obs}} + 1)$
        // Maximum perturbations
        $M = n \times \epsilon$
        // Indices to be perturbed
        Greedily pick $M$ indices $(u, v)$ in $S$ with
          maximum score $S_{uv}$ such that for each row $u$
          the number of chosen indices is $\leq 5$
        $\Delta X^*_{uv} = \begin{cases} 1 & \text{(u,v)} \in \text{picked indices} \\ 0 & \text{otherwise} \end{cases}$
        $X = X^{\text{obs}} \oplus \Delta X^*$
        // Outer Minimization
        for k steps do
            Use one step of stochastic gradient descent
              to update parameters $\theta$ using $\mathcal{Q}_X$
        end
        $t = t + 1$
    until convergence;
    return $\theta$
end
```

D EXPERIMENTAL DETAILS

For RAT-SPNs, we used the following structural parameters for all datasets: depth $D = 3$, number of replicas $R = 50$, number of sum nodes $C = 10$, number of input distributions $I = 10$. We used stochastic gradient descent a learning rate of $1e$-2 in estimation of parameters that minimize the cross entropy. All our experiments for SPNs were performed on machine equipped with a NVIDIA A40 GPU.

We experimented with three values, $\{1,\ 3,\ 5\}$, for the wasserstein-ball of radius ϵ. Models of types (2) and (3) were learnt on uncertainty sets $\mathcal{U}_{rs}$, $\mathcal{U}_r$ respectively with varying size of the $\epsilon-$ball. These sets govern the number of distributions we hedge against.

For each dataset and each ϵ, we generated two additional test sets. The first test set, which we call fully adversarial test set, denoted by $\mathcal{T}_a$ was generated by a point-wise adversary from $\mathcal{T}$ as follows. We begin with an empty $\mathcal{T}_a$. Then, for each test example in $\mathcal{T}$, we use *greedy local search* to find a neighbor of the example that is at most ϵ hamming distance away and has the largest negative log-likelihood score w.r.t. either the SPN or CN and add it to $\mathcal{T}_a$. The second test set which we call randomly perturbed test set, denoted by $\mathcal{T}_r$, was generated from $\mathcal{T}$ as follows. We begin with an empty $\mathcal{T}_r$. Then, for each test example in $\mathcal{T}$, we select a neighbor from 100 *randomly generated neighbors* such that each neighbor is at most ϵ hamming distance away from the example and the selected neighbor has the largest negative log-likelihood score w.r.t. either the SPN, and add it to $\mathcal{T}_r$.

We evaluate both the generative and predictive performances of all three types of models under various corruption scenarios as described above.

D.1 ROBUST GENERATIVE PERFORMANCE

To evaluate the generative performance and robustness of the learned models, we compare their log-likelihood scores on three different test sets described above ($\mathcal{T}$,$\mathcal{T}_r$,$\mathcal{T}_a$) for $\epsilon = \{1, 3, 5\}$. Scores on the set $\mathcal{T}$ indicate the model's *goodness-of-fit* to the empirical distribution and larger scores imply a better fit. On the other hand, scores on the sets $\mathcal{T}_a$ and $\mathcal{T}_r$ are representative of a model's robustness to adversarial and random perturbations. Higher scores imply that the model is resilient to small perturbations to the samples in $\mathcal{T}$.

We observe that although SPNs have slightly higher scores on $\mathcal{T}$ as compared to their robust counterparts {SPN−k, SPN−g }'s, they have significantly lower scores on the corrupted sets $\mathcal{T}_a$ and $\mathcal{T}_r$. SPNs trained using our proposed approaches consistently exhibit superior robust test-set log-likelihood scores as compared with standard SPNs.

Impact of increasing ϵ: We observe that as we increase ϵ, the performance of both SPN−k and SPN−g degrades on the original test set $\mathcal{T}$. On the adversarial and random test sets, namely on $\mathcal{T}_a$ and $\mathcal{T}_r$ respectively, we observe that increasing ϵ significantly degrades the performance of SPNs which are trained on the original training set. For instance, there is several orders of magnitude difference between the log-likelihood scores on $\mathcal{T}_a$ (and $\mathcal{T}_r$) for $h = 5$ and $h = 1$. On the other hand, as compared with SPNs, the rate of decrease in log-likelihoods (as we increase ϵ) is much smaller for SPN−k and SPN−g.

D.2 ROBUST PREDICTIVE PERFORMANCE

We used conditional log-likelihood (CLL) scores to evaluate the predictive performance. Given query variables q and evidence variables e, the CLL score of a data point x equals $\log f(x^q|x^e)$. We compare the average CLL scores of all models on $\mathcal{T}$, $\mathcal{T}_a$ and $\mathcal{T}_r$. We randomly selected different percentages of variables as query variables and set the remaining variables as evidence variables.

We observe a similar trend: {SPN−k, SPN−g } have better CLL scores compared to SPN respectively on $\mathcal{T}_a$ and $\mathcal{T}_r$. These results demonstrate that our proposed method yields robust predictions.

Table 2: Generative performance: Test set log-likelihood scores of SPN trained using $\epsilon \in \{1, 2, 3\}$: `SPN`: SPN trained using maximum likelihood estimation on original training data, `SPN−k`: SPN trained using sampling-based DRO, `SPN−g`: SPN trained using DRO with fast gradient method. $\mathcal{T}$: original test data, $\mathcal{T}_a$: adversarially perturbed $\mathcal{T}$ by `SPN`, $\mathcal{T}_r$: randomly perturbed $\mathcal{T}$ by `SPN`.

DATASET	ϵ	$\mathcal{T}$			$\mathcal{T}_a$			$\mathcal{T}_r$		
		`SPN`	`SPN−k`	`SPN−g`	`SPN`	`SPN−k`	`SPN−g`	`SPN`	`SPN−k`	`SPN−g`
nltcs	1	-6.29	-6.47	-6.47	-9.9	-9.22	-8.76	-9.88	-9.08	-8.72
	3		-7.03	-7.54	-15.77	-12.12	-11.37	-12.59	-10.45	-10.27
	5		-7.01	-8.29	-18.78	-13.62	-12.21	-14.4	-11.49	-10.78
Avg.		-6.29	-6.84	-7.43	-14.82	-11.65	-10.78	-12.29	-10.34	-9.92
msnbc	1	-6.15	-6.68	-6.96	-11.84	-9.66	-8.89	-11.32	-9.43	-8.86
	3		-7.0	-7.61	-19.43	-12.61	-11.25	-14.52	-10.66	-10.2
	5		-6.93	-8.34	-21.28	-14.5	-12.45	-16.42	-11.67	-10.96
Avg.		-6.15	-6.87	-7.64	-17.52	-12.26	-10.86	-14.09	-10.59	-10.01
kdd-2k	1	-2.17	-2.63	-2.5	-9.3	-7.04	-6.84	-9.2	-7.02	-6.83
	3		-3.69	-3.31	-23.13	-14.0	-13.78	-18.74	-13.16	-13.2
	5		-3.9	-4.23	-33.91	-20.42	-19.37	-25.33	-18.55	-18.09
Avg.		-2.17	-3.41	-3.35	-22.11	-13.82	-13.33	-17.76	-12.91	-12.71
plants	1	-14.94	-15.98	-15.49	-21.51	-20.33	-19.53	-21.26	-20.12	-19.48
	3		-16.67	-16.69	-33.84	-26.82	-26.14	-28.36	-24.45	-24.32
	5		-16.89	-18.35	-44.48	-32.69	-31.05	-34.25	-28.37	-28.04
Avg.		-14.94	-16.51	-16.84	-33.28	-26.61	-25.57	-27.96	-24.31	-23.95
audio	1	-40.58	-40.65	-40.67	-44.47	-43.99	-43.38	-44.07	-43.74	-43.31
	3		-40.69	-41.11	-50.98	-49.32	-47.89	-47.01	-46.28	-45.9
	5		-40.77	-41.45	-56.21	-53.93	-51.75	-49.85	-48.8	-48.24
Avg.		-40.58	-40.7	-41.08	-50.55	-49.08	-47.67	-46.98	-46.27	-45.82
jester	1	-53.3	-53.31	-53.49	-56.0	-55.76	-55.54	-55.73	-55.53	-55.42
	3		-53.31	-53.6	-60.64	-59.74	-59.07	-57.25	-56.94	-56.75
	5		-53.3	-53.74	-64.6	-63.24	-62.2	-58.69	-58.21	-58.14
Avg.		-53.3	-53.31	-53.61	-60.41	-59.58	-58.94	-57.22	-56.89	-56.77
netflix	1	-57.61	-57.55	-57.74	-61.0	-60.33	-59.75	-60.44	-59.96	-59.62
	3		-57.53	-58.37	-66.49	-64.69	-63.0	-61.73	-60.93	-60.84
	5		-57.59	-58.84	-70.84	-68.45	-65.6	-62.69	-61.94	-61.78
Avg.		-57.61	-57.56	-58.32	-66.11	-64.49	-62.78	-61.62	-60.94	-60.75
accidents	1	-42.93	-41.43	-43.47	-47.68	-45.95	-46.51	-47.44	-45.34	-46.4
	3		-41.66	-43.4	-56.58	-52.97	-50.22	-50.4	-47.97	-48.28
	5		-41.5	-44.45	-64.31	-58.87	-55.88	-53.5	-50.71	-51.74
Avg.		-42.93	-41.53	-43.77	-56.19	-52.6	-50.87	-50.45	-48.01	-48.81
retail	1	-11.43	-11.57	-11.56	-16.24	-15.93	-15.73	-16.23	-15.91	-15.75
	3		-12.1	-12.01	-25.85	-23.59	-23.22	-24.26	-22.55	-22.55
	5		-12.24	-12.64	-35.43	-30.91	-29.67	-32.09	-28.86	-28.36
Avg.		-11.43	-11.97	-12.07	-25.84	-23.48	-22.87	-24.19	-22.44	-22.22
pumsb-star	1	-42.16	-41.32	-40.35	-47.59	-45.35	-45.01	-47.52	-45.41	-45.05
	3		-42.35	-43.05	-58.3	-52.97	-53.78	-53.11	-50.8	-51.02
	5		-41.5	-44.15	-68.87	-61.23	-59.71	-58.52	-55.31	-55.43
Avg.		-42.16	-41.72	-42.52	-58.25	-53.18	-52.83	-53.05	-50.51	-50.5
dna	1	-99.51	-98.91	-98.52	-101.21	-100.55	-100.08	-101.15	-100.54	-100.04
	3		-99.58	-99.59	-104.5	-104.5	-103.63	-102.28	-102.36	-102.08
	5		-99.35	-99.63	-107.56	-107.08	-106.15	-103.5	-103.18	-103.13
Avg.		-99.51	-99.28	-99.25	-104.42	-104.04	-103.29	-102.31	-102.03	-101.75
kosarek	1	-11.4	-11.59	-11.67	-17.54	-17.0	-16.77	-17.49	-16.9	-16.75
	3		-12.33	-12.43	-29.58	-25.62	-25.3	-27.28	-24.56	-24.48
	5		-12.49	-13.02	-41.06	-34.05	-33.02	-36.23	-31.85	-31.35

Avg.		-11.4	-12.14	-12.37	-29.39	-25.56	-25.03	-27.0	-24.44	-24.19
msweb	1	-11.39	-11.73	-11.74	-18.9	-17.7	-17.34	-18.86	-17.64	-17.35
	3		-12.54	-12.5	-33.84	-27.63	-27.03	-30.68	-26.39	-26.3
	5		-12.74	-13.32	-48.76	-37.08	-35.36	-42.21	-34.76	-34.18
Avg.		-11.39	-12.34	-12.52	-33.83	-27.47	-26.58	-30.58	-26.26	-25.94
book	1	-35.7	-35.61	-35.8	-41.12	-40.7	-40.44	-40.96	-40.69	-40.45
	3		-36.06	-36.01	-51.57	-50.29	-49.26	-49.73	-48.95	-48.59
	5		-36.09	-36.46	-61.8	-59.48	-57.52	-58.32	-56.84	-56.2
Avg.		-35.7	-35.92	-36.09	-51.5	-50.16	-49.07	-49.67	-48.83	-48.41
each-movie	1	-58.03	-57.12	-57.08	-64.55	-62.81	-62.75	-64.46	-62.93	-62.75
	3		-57.25	-56.56	-77.43	-72.83	-72.05	-73.04	-70.64	-69.81
	5		-59.45	-58.74	-90.04	-84.57	-82.45	-81.85	-80.2	-78.8
Avg.		-58.03	-57.94	-57.46	-77.34	-73.4	-72.42	-73.12	-71.26	-70.45
web-kb	1	-161.51	-163.85	-162.38	-167.09	-168.14	-166.74	-166.67	-168.19	-166.74
	3		-165.41	-162.64	-177.82	-176.9	-174.91	-172.73	-174.84	-172.54
	5		-162.82	-162.0	-188.17	-186.55	-181.69	-178.55	-179.07	-177.37
Avg.		-161.51	-164.03	-162.34	-177.69	-177.2	-174.45	-172.65	-174.03	-172.22
reuters-52	1	-97.75	-96.04	-97.09	-104.28	-102.27	-102.34	-103.72	-101.91	-102.27
	3		-96.56	-97.73	-116.63	-113.68	-112.16	-111.44	-109.88	-110.15
	5		-97.35	-96.63	-128.42	-124.91	-119.87	-118.98	-118.02	-116.1
Avg.		-97.75	-96.65	-97.15	-116.44	-113.62	-111.46	-111.38	-109.94	-109.51
20ng	1	-156.86	-155.8	-155.64	-162.03	-161.12	-160.09	-161.64	-160.96	-160.03
	3		-156.06	-156.25	-171.86	-171.01	-168.65	-167.81	-167.2	-166.45
	5		-156.37	-156.4	-181.37	-180.11	-176.32	-173.81	-173.2	-172.53
Avg.		-156.86	-156.08	-156.1	-171.75	-170.75	-168.35	-167.75	-167.12	-166.34
bbc	1	-275.22	-275.11	-275.15	-278.59	-278.37	-278.34	-278.52	-278.36	-278.34
	3		-269.57	-275.33	-285.25	-279.78	-284.45	-283.06	-277.57	-282.76
	5		-275.28	-270.66	-291.84	-291.17	-286.29	-287.63	-287.07	-282.89
Avg.		-275.22	-273.32	-273.71	-285.23	-283.11	-283.03	-283.07	-281.0	-281.33
ad	1	-63.97	-64.17	-63.96	-69.59	-69.51	-69.12	-69.5	-69.52	-69.16
	3		-64.31	-64.38	-80.8	-80.92	-79.41	-78.94	-79.27	-78.87
	5		-64.41	-64.68	-91.93	-91.5	-89.23	-88.86	-88.55	-88.1
Avg.		-63.97	-64.3	-64.34	-80.77	-80.64	-79.25	-79.1	-79.11	-78.71

Table 3: Predictive performance: Conditional log-likelihood scores given 20% evidence for models having latent variables (SPNs).$\epsilon \in \{1, 2, 3\}$: SPN: SPN trained using maximum likelihood estimation on original training data, SPN−k: SPN trained using sampling-based DRO, SPN−g: SPN trained using DRO with fast gradient method. $\mathcal{T}$: original test data, $\mathcal{T}_a$: adversarially perturbed $\mathcal{T}$ by SPN, $\mathcal{T}_r$: randomly perturbed $\mathcal{T}$ by SPN.

DATASET	ϵ	$\mathcal{T}$			$\mathcal{T}_a$			$\mathcal{T}_r$		
		SPN	SPN−k	SPN−g	SPN	SPN−k	SPN−g	SPN	SPN−k	SPN−g
nltcs	1	-5.07	-5.21	-5.23	-8.58	-7.93	-7.44	-8.58	-7.76	-7.42
	3		-5.68	-6.15	-13.61	-10.44	-9.54	-10.53	-8.78	-8.41
	5		-5.64	-6.75	-15.53	-10.96	-9.93	-12.16	-9.54	-8.85
Avg.		-5.07	-5.51	-6.04	-12.57	-9.78	-8.97	-10.42	-8.69	-8.23
msnbc	1	-4.39	-4.91	-5.18	-10.07	-7.89	-7.11	-9.55	-7.66	-7.08
	3		-5.22	-5.83	-17.67	-10.83	-9.47	-12.62	-8.76	-8.3
	5		-5.14	-6.53	-19.51	-12.7	-10.65	-14.4	-9.71	-8.99
Avg.		-4.39	-5.09	-5.85	-15.75	-10.47	-9.08	-12.19	-8.71	-8.12
kdd-2k	1	-2.04	-2.4	-2.26	-2.28	-2.45	-2.29	-2.57	-2.96	-2.68
	3		-3.24	-2.87	-10.01	-3.95	-3.14	-11.11	-8.74	-8.56
	5		-3.4	-3.6	-20.47	-10.45	-7.43	-16.58	-13.49	-12.98
Avg.		-2.04	-3.01	-2.91	-10.92	-5.62	-4.29	-10.09	-8.4	-8.07
plants	1	-10.87	-11.85	-11.34	-14.1	-13.67	-13.14	-15.61	-15.32	-14.13
	3		-12.43	-12.41	-23.34	-18.49	-18.04	-21.1	-18.71	-18.73
	5		-12.63	-13.92	-31.88	-22.86	-21.93	-26.59	-21.99	-21.57
Avg.		-10.87	-12.3	-12.56	-23.11	-18.34	-17.7	-21.1	-18.67	-18.14
audio	1	-31.95	-32.01	-32.03	-35.56	-35.12	-34.49	-35.17	-34.76	-34.33
	3		-32.04	-32.42	-41.14	-39.73	-38.24	-37.44	-36.83	-36.43
	5		-32.1	-32.71	-45.67	-43.57	-41.5	-39.86	-38.9	-38.47
Avg.		-31.95	-32.05	-32.39	-40.79	-39.47	-38.08	-37.49	-36.83	-36.41
jester	1	-40.94	-40.96	-41.14	-43.35	-43.08	-42.95	-43.22	-43.02	-42.9
	3		-40.95	-41.24	-47.85	-46.94	-46.36	-44.55	-44.26	-44.13
	5		-40.95	-41.38	-51.6	-50.22	-49.3	-45.88	-45.42	-45.38
Avg.		-40.94	-40.95	-41.25	-47.6	-46.75	-46.2	-44.55	-44.23	-44.14
netflix	1	-45.14	-45.07	-45.25	-48.41	-47.74	-47.16	-47.79	-47.33	-46.98
	3		-45.05	-45.83	-53.48	-51.79	-50.09	-48.94	-48.16	-48.03
	5		-45.1	-46.26	-57.44	-55.14	-52.4	-49.77	-49.07	-48.86
Avg.		-45.14	-45.07	-45.78	-53.11	-51.56	-49.88	-48.83	-48.19	-47.96
accidents	1	-33.46	-32.0	-33.87	-36.49	-35.45	-35.75	-36.15	-34.63	-35.32
	3		-32.31	-33.7	-38.04	-35.79	-35.56	-38.63	-36.61	-36.82
	5		-32.09	-34.67	-42.13	-36.98	-38.26	-40.35	-38.92	-40.22
Avg.		-33.46	-32.13	-34.08	-38.89	-36.07	-36.52	-38.38	-36.72	-37.45
retail	1	-5.94	-6.07	-6.07	-10.75	-10.43	-10.24	-10.74	-10.41	-10.25
	3		-6.55	-6.51	-20.36	-18.05	-17.72	-17.6	-15.94	-15.91
	5		-6.68	-7.11	-29.93	-25.35	-24.14	-24.32	-21.34	-20.75
Avg.		-5.94	-6.43	-6.56	-20.35	-17.94	-17.37	-17.55	-15.9	-15.64
pumsb-star	1	-31.6	-31.03	-30.68	-35.24	-33.73	-33.51	-36.0	-34.24	-34.51
	3		-31.71	-32.27	-44.24	-39.84	-40.51	-40.35	-38.45	-38.55
	5		-31.66	-34.08	-53.53	-45.3	-46.55	-45.08	-42.3	-43.21
Avg.		-31.6	-31.47	-32.34	-44.34	-39.62	-40.19	-40.48	-38.33	-38.76
dna	1	-79.43	-78.89	-78.51	-81.07	-80.52	-80.0	-80.98	-80.46	-79.95
	3		-79.48	-79.5	-84.18	-84.15	-83.33	-81.92	-82.01	-81.71
	5		-79.29	-79.53	-87.2	-86.74	-85.8	-82.86	-82.61	-82.53
Avg.		-79.43	-79.22	-79.18	-84.15	-83.8	-83.04	-81.92	-81.69	-81.4
kosarek	1	-4.71	-4.91	-4.94	-10.85	-10.3	-10.04	-10.79	-10.21	-10.03
	3		-5.59	-5.74	-22.89	-18.93	-18.61	-19.27	-16.7	-16.57
	5		-5.77	-6.24	-34.35	-27.24	-26.23	-26.83	-22.85	-22.32

Avg.		-4.71	-5.42	-5.64	-22.7	-18.82	-18.29	-18.96	-16.59	-16.31
msweb	1	-3.42	-3.74	-3.76	-10.93	-9.71	-9.36	-9.52	-9.65	-8.24
	3		-4.47	-4.48	-18.41	-19.57	-14.16	-19.85	-17.0	-16.03
	5		-4.66	-5.26	-33.33	-29.0	-22.92	-29.97	-24.09	-22.6
Avg.		-3.42	-4.29	-4.5	-20.89	-19.43	-15.48	-19.78	-16.91	-15.62
book	1	-27.52	-27.43	-27.61	-32.85	-32.36	-32.15	-32.63	-32.4	-32.13
	3		-27.8	-27.77	-42.92	-41.62	-40.56	-39.59	-38.99	-38.62
	5		-27.83	-28.14	-52.74	-50.5	-48.35	-46.64	-45.4	-44.59
Avg.		-27.52	-27.69	-27.84	-42.84	-41.49	-40.35	-39.62	-38.93	-38.45
each-movie	1	-41.28	-40.51	-40.46	-47.62	-45.85	-45.98	-47.53	-45.86	-45.75
	3		-40.53	-40.39	-60.24	-54.73	-55.54	-54.47	-52.21	-52.03
	5		-42.0	-41.89	-72.53	-64.9	-65.04	-61.76	-59.61	-59.03
Avg.		-41.28	-41.01	-40.91	-60.13	-55.16	-55.52	-54.59	-52.56	-52.27
web-kb	1	-129.01	-131.0	-129.77	-133.95	-134.32	-133.64	-133.36	-134.75	-133.4
	3		-132.35	-130.02	-143.22	-141.86	-140.66	-138.35	-140.0	-138.46
	5		-130.04	-129.3	-151.22	-148.66	-145.46	-143.03	-143.09	-141.68
Avg.		-129.01	-131.13	-129.7	-142.8	-141.61	-139.92	-138.25	-139.28	-137.85
reuters-52	1	-77.56	-76.11	-77.05	-83.77	-82.32	-82.03	-82.69	-81.6	-81.5
	3		-76.45	-77.69	-95.45	-93.24	-91.25	-89.01	-87.98	-87.96
	5		-77.19	-76.65	-104.28	-103.61	-96.56	-95.03	-94.49	-92.47
Avg.		-77.56	-76.58	-77.13	-94.5	-93.06	-89.95	-88.91	-88.02	-87.31
20ng	1	-124.48	-123.54	-123.45	-129.54	-127.59	-127.79	-128.86	-127.9	-127.49
	3		-123.73	-123.92	-138.88	-135.7	-135.83	-133.8	-133.05	-132.57
	5		-124.0	-124.13	-148.05	-144.19	-143.23	-138.61	-137.81	-137.5
Avg.		-124.48	-123.76	-123.83	-138.82	-135.83	-135.62	-133.76	-132.92	-132.52
bbc	1	-182.02	-181.85	-181.78	-185.39	-185.08	-184.97	-185.32	-185.1	-184.97
	3		-177.71	-182.04	-192.05	-187.86	-191.17	-189.4	-185.18	-189.04
	5		-181.8	-178.58	-198.64	-197.69	-194.21	-193.54	-192.72	-189.84
Avg.		-182.02	-180.45	-180.8	-192.03	-190.21	-190.12	-189.42	-187.67	-187.95
ad	1	-50.57	-50.73	-50.57	-56.19	-56.09	-55.72	-55.4	-55.97	-54.82
	3		-50.85	-50.91	-67.39	-67.46	-65.94	-63.2	-63.35	-62.89
	5		-50.93	-51.15	-78.53	-77.92	-75.7	-70.24	-71.02	-70.43
Avg.		-50.57	-50.84	-50.88	-67.37	-67.16	-65.79	-62.95	-63.45	-62.71

Table 4: Predictive performance: Conditional log-likelihood scores given 50% evidence for models having latent variables (SPNs).$\epsilon \in \{1, 2, 3\}$: `SPN`: SPN trained using maximum likelihood estimation on original training data, `SPN−k`: SPN trained using sampling-based DRO, `SPN−g`: SPN trained using DRO with fast gradient method.$\mathcal{T}$: original test data, $\mathcal{T}_a$: adversarially perturbed $\mathcal{T}$ by `SPN`, $\mathcal{T}_r$: randomly perturbed $\mathcal{T}$ by `SPN`.

DATASET	ϵ	$\mathcal{T}$			$\mathcal{T}_a$			$\mathcal{T}_r$		
		SPN	SPN−k	SPN−g	SPN	SPN−k	SPN−g	SPN	SPN−k	SPN−g
nltcs	1	-2.77	-2.86	-2.88	-5.94	-5.55	-4.87	-5.9	-5.31	-4.73
	3		-3.19	-3.53	-9.63	-7.62	-6.22	-7.08	-5.67	-5.22
	5		-3.14	-3.92	-10.8	-7.62	-6.52	-8.24	-6.13	-5.53
Avg.		-2.77	-3.06	-3.44	-8.79	-6.93	-5.87	-7.07	-5.7	-5.16
msnbc	1	-2.31	-2.77	-2.99	-7.99	-5.74	-4.92	-7.31	-5.41	-4.84
	3		-2.99	-3.48	-14.94	-8.56	-6.85	-9.57	-6.0	-5.48
	5		-2.88	-4.01	-14.35	-9.82	-7.21	-10.8	-6.59	-5.87
Avg.		-2.31	-2.88	-3.49	-12.43	-8.04	-6.33	-9.23	-6.0	-5.4
kdd-2k	1	-1.08	-1.3	-1.22	-1.15	-1.35	-1.25	-1.38	-1.86	-1.64
	3		-1.81	-1.62	-1.91	-2.39	-1.87	-6.97	-5.52	-5.34
	5		-1.91	-2.08	-9.16	-8.8	-5.89	-10.15	-8.7	-8.13
Avg.		-1.08	-1.67	-1.64	-4.07	-4.18	-3.0	-6.17	-5.36	-5.04
plants	1	-6.62	-7.36	-6.93	-8.44	-8.67	-7.92	-9.93	-9.05	-8.59
	3		-7.79	-7.75	-15.39	-11.64	-11.69	-13.26	-11.47	-11.9
	5		-7.92	-8.84	-21.45	-14.7	-14.4	-17.03	-13.92	-13.58
Avg.		-6.62	-7.69	-7.84	-15.09	-11.67	-11.34	-13.41	-11.48	-11.36
audio	1	-19.53	-19.55	-19.57	-21.51	-21.27	-20.84	-21.34	-21.05	-20.8
	3		-19.58	-19.82	-25.02	-24.01	-23.09	-22.52	-22.37	-22.11
	5		-19.59	-20.0	-27.68	-26.43	-25.04	-24.3	-23.74	-23.58
Avg.		-19.53	-19.57	-19.8	-24.74	-23.9	-22.99	-22.72	-22.39	-22.16
jester	1	-25.79	-25.81	-25.91	-27.45	-27.05	-27.1	-27.21	-27.13	-27.01
	3		-25.79	-25.97	-30.16	-29.3	-29.07	-28.01	-27.77	-27.65
	5		-25.8	-26.03	-32.65	-31.09	-30.93	-28.84	-28.51	-28.38
Avg.		-25.79	-25.8	-25.97	-30.09	-29.15	-29.03	-28.02	-27.8	-27.68
netflix	1	-28.55	-28.47	-28.55	-29.72	-29.4	-29.05	-29.96	-29.49	-29.42
	3		-28.46	-28.89	-32.08	-31.02	-30.39	-30.26	-30.02	-29.91
	5		-28.47	-29.19	-34.3	-32.71	-31.65	-31.11	-30.53	-30.52
Avg.		-28.55	-28.47	-28.88	-32.03	-31.04	-30.36	-30.44	-30.01	-29.95
accidents	1	-16.2	-15.83	-16.3	-17.21	-16.34	-16.96	-18.11	-17.46	-17.46
	3		-15.87	-16.76	-18.17	-16.49	-17.56	-20.07	-18.85	-19.08
	5		-15.51	-17.33	-22.24	-17.53	-19.66	-21.38	-20.59	-21.66
Avg.		-16.2	-15.74	-16.8	-19.21	-16.79	-18.06	-19.85	-18.97	-19.4
retail	1	-3.24	-3.33	-3.36	-8.05	-7.7	-7.53	-8.04	-7.68	-7.55
	3		-3.66	-3.69	-17.66	-15.16	-14.9	-12.63	-11.17	-10.99
	5		-3.74	-4.1	-27.23	-22.41	-21.13	-17.03	-14.49	-14.03
Avg.		-3.24	-3.58	-3.72	-17.65	-15.09	-14.52	-12.57	-11.11	-10.86
pumsb-star	1	-19.83	-19.59	-19.29	-20.18	-20.11	-19.58	-21.19	-21.11	-20.1
	3		-20.19	-20.21	-22.92	-23.47	-22.2	-24.11	-23.77	-23.01
	5		-19.65	-20.69	-25.14	-24.91	-23.63	-26.65	-25.93	-25.74
Avg.		-19.83	-19.81	-20.06	-22.75	-22.83	-21.8	-23.98	-23.6	-22.95
dna	1	-50.39	-50.26	-49.85	-50.72	-50.51	-50.13	-50.97	-50.66	-50.31
	3		-50.4	-50.31	-51.95	-51.99	-51.58	-51.49	-51.48	-51.3
	5		-50.38	-50.32	-53.84	-53.45	-53.14	-51.95	-51.88	-51.75
Avg.		-50.39	-50.35	-50.16	-52.17	-51.98	-51.62	-51.47	-51.34	-51.12
kosarek	1	-2.47	-2.61	-2.63	-8.61	-4.9	-7.73	-8.02	-6.76	-7.28
	3		-3.04	-3.16	-20.64	-13.81	-16.03	-13.6	-11.22	-11.38
	5		-3.17	-3.49	-32.09	-22.22	-23.47	-18.55	-15.33	-14.92

Avg.		-2.47	-2.94	-3.09	-20.45	-13.64	-15.74	-13.39	-11.1	-11.19
msweb	1	-0.9	-1.14	-1.19	-8.41	-7.11	-6.79	-6.99	-7.05	-5.67
	3		-1.67	-1.75	-15.88	-16.76	-11.42	-13.98	-11.57	-10.48
	5		-1.79	-2.31	-30.79	-26.14	-19.95	-20.72	-16.09	-14.52
Avg.		-0.9	-1.53	-1.75	-18.36	-16.67	-12.72	-13.9	-11.57	-10.22
book	1	-16.75	-16.69	-16.81	-21.94	-21.61	-21.25	-20.72	-21.14	-20.24
	3		-16.95	-16.94	-31.21	-30.58	-28.91	-25.12	-25.17	-24.54
	5		-16.97	-17.22	-36.97	-38.87	-33.26	-29.46	-29.22	-27.91
Avg.		-16.75	-16.87	-16.99	-30.04	-30.35	-27.81	-25.1	-25.18	-24.23
each-movie	1	-24.76	-24.29	-24.29	-27.46	-28.88	-26.57	-28.6	-28.59	-27.6
	3		-24.39	-24.15	-35.13	-34.99	-32.52	-32.47	-32.26	-31.21
	5		-25.05	-25.19	-43.01	-42.86	-38.65	-37.54	-37.02	-35.65
Avg.		-24.76	-24.58	-24.54	-35.2	-35.58	-32.58	-32.87	-32.62	-31.49
web-kb	1	-77.8	-78.86	-78.18	-81.69	-81.91	-81.33	-81.0	-81.67	-80.84
	3		-79.7	-78.36	-89.3	-87.52	-87.25	-84.2	-84.87	-84.2
	5		-78.37	-77.97	-95.98	-93.04	-91.51	-87.08	-87.01	-86.24
Avg.		-77.8	-78.98	-78.17	-88.99	-87.49	-86.7	-84.09	-84.52	-83.76
reuters-52	1	-47.89	-46.75	-47.52	-53.53	-49.3	-52.04	-50.96	-49.72	-50.27
	3		-47.09	-47.91	-60.44	-55.32	-57.25	-54.91	-53.65	-54.26
	5		-47.63	-47.25	-66.5	-63.85	-60.93	-58.57	-58.1	-56.9
Avg.		-47.89	-47.16	-47.56	-60.16	-56.16	-56.74	-54.81	-53.82	-53.81
20ng	1	-77.69	-77.07	-76.97	-81.75	-77.85	-80.54	-80.48	-79.29	-79.55
	3		-77.19	-77.29	-86.79	-82.96	-84.73	-83.59	-82.52	-82.66
	5		-77.37	-77.44	-91.9	-88.18	-88.69	-86.54	-85.65	-85.75
Avg.		-77.69	-77.21	-77.23	-86.81	-83.0	-84.65	-83.54	-82.49	-82.65
bbc	1	-93.79	-93.74	-93.68	-97.16	-96.97	-96.85	-97.1	-96.99	-96.86
	3		-91.74	-93.88	-103.82	-101.92	-102.99	-99.93	-98.1	-99.58
	5		-93.68	-92.46	-110.4	-109.53	-108.09	-102.93	-102.29	-101.24
Avg.		-93.79	-93.05	-93.34	-103.79	-102.81	-102.64	-99.99	-99.13	-99.23
ad	1	-31.82	-31.93	-31.83	-37.44	-37.31	-36.99	-35.77	-36.48	-35.23
	3		-32.01	-32.05	-43.07	-48.62	-42.13	-40.87	-40.65	-40.51
	5		-32.05	-32.2	-54.19	-53.76	-51.83	-44.0	-45.08	-44.55
Avg.		-31.82	-32.0	-32.03	-44.9	-46.56	-43.65	-40.21	-40.74	-40.1

Table 5: Predictive performance: Conditional log-likelihood scores given 80% evidence for models having latent variables (SPNs). $\epsilon \in \{1, 2, 3\}$: SPN: SPN trained using maximum likelihood estimation on original training data, SPN−k: SPN trained using sampling-based DRO, SPN−g: SPN trained using DRO with fast gradient method.$\mathcal{T}$: original test data, $\mathcal{T}_a$: adversarially perturbed $\mathcal{T}$ by SPN, $\mathcal{T}_r$: randomly perturbed $\mathcal{T}$ by SPN.

DATASET	ϵ	$\mathcal{T}$			$\mathcal{T}_a$			$\mathcal{T}_r$		
		SPN	SPN−k	SPN−g	SPN	SPN−k	SPN−g	SPN	SPN−k	SPN−g
nltcs	1	-1.17	-1.24	-1.25	-3.98	-2.53	-2.98	-4.01	-2.59	-2.67
	3		-1.43	-1.65	-5.72	-4.03	-3.38	-3.68	-2.86	-2.66
	5		-1.4	-1.85	-5.43	-3.79	-3.41	-4.67	-3.36	-2.82
Avg.		-1.17	-1.36	-1.58	-5.04	-3.45	-3.26	-4.12	-2.94	-2.72
msnbc	1	-0.66	-0.93	-1.06	-5.17	-3.54	-2.56	-4.6	-3.07	-2.5
	3		-1.06	-1.39	-9.78	-5.3	-3.68	-5.54	-3.13	-2.61
	5		-0.96	-1.67	-7.87	-5.38	-3.7	-6.31	-3.26	-2.7
Avg.		-0.66	-0.98	-1.37	-7.61	-4.74	-3.31	-5.48	-3.15	-2.6
kdd-2k	1	-0.38	-0.48	-0.44	-0.39	-0.48	-0.44	-0.4	-0.48	-0.45
	3		-0.69	-0.61	-0.52	-0.7	-0.63	-2.49	-2.03	-1.99
	5		-0.73	-0.81	-2.3	-1.2	-1.02	-3.71	-3.15	-3.08
Avg.		-0.38	-0.63	-0.62	-1.07	-0.79	-0.7	-2.2	-1.89	-1.84
plants	1	-2.76	-3.05	-2.86	-3.6	-3.33	-3.35	-3.55	-3.6	-3.32
	3		-3.23	-3.28	-6.04	-4.73	-4.74	-5.36	-4.58	-4.83
	5		-3.4	-3.66	-8.3	-6.24	-5.54	-6.7	-5.57	-5.54
Avg.		-2.76	-3.23	-3.27	-5.98	-4.77	-4.54	-5.2	-4.58	-4.56
audio	1	-7.88	-7.88	-7.89	-8.96	-8.9	-8.54	-8.67	-8.45	-8.39
	3		-7.91	-7.99	-10.04	-9.76	-9.23	-9.19	-9.24	-8.88
	5		-7.93	-8.06	-11.26	-10.73	-10.09	-9.82	-9.67	-9.5
Avg.		-7.88	-7.91	-7.98	-10.09	-9.8	-9.29	-9.23	-9.12	-8.92
jester	1	-10.53	-10.51	-10.55	-10.84	-10.78	-10.74	-10.88	-10.88	-10.83
	3		-10.51	-10.56	-11.69	-11.4	-11.29	-11.26	-11.1	-11.08
	5		-10.5	-10.58	-12.62	-12.25	-11.95	-11.6	-11.39	-11.38
Avg.		-10.53	-10.51	-10.56	-11.72	-11.48	-11.33	-11.25	-11.12	-11.1
netflix	1	-11.35	-11.32	-11.34	-11.94	-11.78	-11.61	-11.99	-11.78	-11.83
	3		-11.32	-11.51	-13.38	-12.74	-12.42	-12.17	-12.08	-12.0
	5		-11.31	-11.62	-14.5	-13.64	-13.02	-12.54	-12.2	-12.23
Avg.		-11.35	-11.32	-11.49	-13.27	-12.72	-12.35	-12.23	-12.02	-12.02
accidents	1	-4.9	-4.84	-5.05	-5.81	-5.23	-5.68	-6.41	-5.27	-6.0
	3		-5.16	-5.33	-6.47	-5.59	-6.12	-7.59	-6.31	-6.67
	5		-4.84	-5.56	-9.02	-5.91	-7.71	-7.87	-7.03	-8.01
Avg.		-4.9	-4.95	-5.31	-7.1	-5.58	-6.5	-7.29	-6.2	-6.89
retail	1	-1.18	-1.21	-1.24	-5.99	-5.58	-5.41	-5.98	-5.15	-5.42
	3		-1.35	-1.38	-15.6	-12.19	-12.59	-7.79	-6.45	-6.63
	5		-1.39	-1.55	-25.17	-16.41	-18.59	-9.57	-7.63	-7.6
Avg.		-1.18	-1.32	-1.39	-15.59	-11.39	-12.2	-7.78	-6.41	-6.55
pumsb-star	1	-5.54	-5.51	-5.41	-5.55	-5.51	-5.41	-5.91	-6.05	-5.48
	3		-5.62	-5.8	-5.59	-5.99	-5.8	-7.34	-7.14	-6.98
	5		-5.62	-6.0	-6.34	-6.66	-6.41	-8.3	-8.33	-8.18
Avg.		-5.54	-5.58	-5.74	-5.83	-6.05	-5.87	-7.18	-7.17	-6.88
dna	1	-20.31	-20.29	-20.21	-20.33	-20.3	-20.22	-20.33	-20.3	-20.22
	3		-20.34	-20.28	-20.37	-20.38	-20.31	-20.55	-20.58	-20.49
	5		-20.31	-20.29	-20.42	-20.39	-20.33	-20.77	-20.73	-20.75
Avg.		-20.31	-20.31	-20.26	-20.37	-20.36	-20.29	-20.55	-20.54	-20.49
kosarek	1	-0.87	-0.94	-0.95	-6.38	-1.92	-5.46	-4.71	-3.08	-4.15
	3		-1.12	-1.17	-13.02	-6.41	-9.61	-6.88	-5.04	-5.48
	5		-1.16	-1.32	-22.24	-11.0	-15.39	-8.67	-7.06	-6.65

Avg.		-0.87	-1.07	-1.15	-13.88	-6.44	-10.15	-6.75	-5.06	-5.43
msweb	1	-0.19	-0.31	-0.35	-7.7	-6.28	-5.95	-5.86	-5.88	-4.47
	3		-0.54	-0.6	-15.17	-15.62	-10.26	-8.76	-7.28	-6.3
	5		-0.59	-0.83	-30.08	-20.11	-18.48	-11.64	-9.05	-7.57
Avg.		-0.19	-0.48	-0.59	-17.65	-14.0	-11.56	-8.75	-7.4	-6.11
book	1	-7.06	-7.02	-7.08	-7.55	-11.59	-7.49	-9.55	-9.49	-9.0
	3		-7.14	-7.13	-16.3	-12.36	-14.87	-11.12	-11.09	-11.02
	5		-7.13	-7.22	-20.84	-19.43	-18.19	-12.76	-12.56	-12.05
Avg.		-7.06	-7.1	-7.14	-14.9	-14.46	-13.52	-11.14	-11.05	-10.69
each-movie	1	-8.62	-8.51	-8.47	-8.99	-11.33	-8.74	-10.36	-10.64	-9.94
	3		-8.57	-8.43	-13.39	-13.62	-12.22	-12.3	-12.12	-11.66
	5		-8.8	-8.77	-17.64	-16.45	-15.27	-14.67	-13.93	-12.96
Avg.		-8.62	-8.63	-8.56	-13.34	-13.8	-12.08	-12.44	-12.23	-11.52
web-kb	1	-29.25	-29.49	-29.34	-31.4	-31.51	-31.25	-30.59	-30.79	-30.54
	3		-29.83	-29.39	-34.73	-33.61	-33.8	-32.06	-32.18	-31.97
	5		-29.42	-29.29	-38.03	-35.83	-35.93	-33.2	-33.03	-32.95
Avg.		-29.25	-29.58	-29.34	-34.72	-33.65	-33.66	-31.95	-32.0	-31.82
reuters-52	1	-18.33	-17.75	-18.24	-21.04	-17.91	-20.78	-19.64	-18.46	-19.74
	3		-17.9	-18.38	-24.93	-18.93	-23.96	-21.44	-20.4	-21.4
	5		-18.26	-17.95	-29.26	-20.31	-26.59	-22.96	-22.16	-21.94
Avg.		-18.33	-17.97	-18.19	-25.08	-19.05	-23.78	-21.35	-20.34	-21.03
20ng	1	-31.76	-31.48	-31.49	-35.29	-31.71	-34.67	-33.0	-32.39	-32.67
	3		-31.54	-31.61	-36.26	-34.81	-35.29	-34.33	-33.69	-33.72
	5		-31.6	-31.71	-38.43	-37.83	-36.94	-35.38	-34.98	-35.17
Avg.		-31.76	-31.54	-31.6	-36.66	-34.78	-35.63	-34.24	-33.69	-33.85
bbc	1	-32.4	-32.39	-32.39	-35.76	-35.67	-35.57	-35.7	-35.61	-35.57
	3		-31.62	-32.52	-42.4	-41.95	-41.62	-36.86	-36.37	-36.72
	5		-32.39	-32.18	-48.96	-48.54	-47.75	-38.33	-37.92	-37.68
Avg.		-32.4	-32.13	-32.36	-42.37	-42.05	-41.65	-36.96	-36.63	-36.66
ad	1	-13.26	-13.3	-13.26	-13.27	-13.3	-13.27	-14.57	-15.48	-14.02
	3		-13.33	-13.35	-18.83	-13.37	-18.36	-16.14	-16.58	-17.03
	5		-13.35	-13.41	-18.83	-18.81	-18.27	-18.53	-18.8	-17.94
Avg.		-13.26	-13.33	-13.34	-16.98	-15.16	-16.63	-16.41	-16.95	-16.33