A PROOFS OF STABILITY THEOREMS

Definition of diagram distances. Recall (see Section 1.3) that persistence diagrams are generally represented as multisets of points (i.e. points counted with multiplicity) supported on the upper half plane $\Omega = \{(b, d) \in \mathbb{R}^2, d > b\}$. Let $\mu = \{x_1, \ldots, x_n\}$ and $\nu = \{y_1, \ldots, y_m\}$ be two such diagrams and $s \geq 1$ be a parameter. Note in particular that $n \neq m$ in general. Let $\Delta = \{(t, t), t \in \mathbb{R}\}$ denote the diagonal, and let $\Pi(\mu, \nu)$ denote the set of all bijections between $\mu \cup \Delta$ and $\nu \cup \Delta$. Then, the s-diagram distance between μ and ν is defined as:

$$d_s(\mu, \nu) = \inf_{\pi \in \Pi(\mu,\nu)} \left(\sum_{x \in \mu \cup \Delta} \|x - s(x)\|^p \right)^{\frac{1}{p}}. \quad (6)$$

In particular, if $s = \infty$, we recover the bottleneck distance defined as:

$$\mathrm{d}_B(\mu, \nu) = \inf_{\pi \in \Pi(\mu,\nu)} \sup_{x \in \mu \cup \Delta} \|x - s(x)\|. \quad (7)$$

Proof of Theorem 2.2 The proof directly follows from the following two theorems. This first one, proved in (Hu et al., 2014), is a consequence of classical arguments from matrix perturbation theory.

Theorem A.1 ((Hu et al., 2014), Theorem 1)**.** *Let $t \geq 0$ and let L_w be the Laplacian matrix of a graph G with n vertices. Let $\lambda_1 < \cdots < \lambda_k$, $k \leq n$ be the distinct eigenvalues of L_w and denote by $\delta > 0$ the smallest distance between two distinct eigenvalues: $\delta = \min_{j=1,\cdots,k-1} |\lambda_{j+1} - \lambda_j|$. Let G' be another graph with n vertices and Laplacian matrix $\tilde{L}_w = L_w + W$ with $\|W\| < \delta$, where $\|W\|$ denotes the Frobenius norm of W. Then, if $k = n$, there exists a constant $C_0(G, t) > 0$ such that for any vertex $v \in G$,*

$$|\mathrm{hks}_{G,t}(v) - \mathrm{hks}_{G',t}(v)| \leqslant C_0(G, t)\|W\|;$$

if $k < n$, there exists two constants $C_1(G, t), C_2(G, t) > 0$ such that for any vertex $v \in G$,

$$|\mathrm{hks}_{G,t}(v) - \mathrm{hks}_{G',t}(v)| \leqslant C_1(G, t)\frac{\|W\|}{\delta - \|W\|} + C_2(G, t)\|W\|$$

In particular, if $\|W\| < \frac{\delta}{2}$, there exists a constant $C(G, t) > 0$—notice that δ also depends on G—such that in the two above cases,

$$|\mathrm{hks}_{G,t}(v) - \mathrm{hks}_{G',t}(v)| \leqslant C(G, t)\|W\|.$$

Theorem 2.2 then immediately follows from the second following theorem, which is a special case of general stability results for persistence diagrams.

Theorem A.2 ((Chazal et al., 2016; Cohen-Steiner et al., 2009))**.** *Let $G = (V, E)$ be a graph and $f, g : V \to \mathbb{R}$ be two functions defined on its vertices. Then:*

$$\mathrm{d}_B(\mathrm{Dg}(G, f), \mathrm{Dg}(G, g)) \leqslant \|f - g\|_\infty, \quad (8)$$

where d_B stands for the so-called bottleneck distance *between persistence diagrams and $\|f - g\|_\infty = \sup_{v \in G} |f(v) - g(v)|$. Moreover, this inequality is also satisfied for each of the subtypes $\mathrm{Ord}_0, \mathrm{Rel}_1, \mathrm{Ext}_0^+$ and Ext_1^- individually.*

Proof of Theorem 2.3 Fix a graph $G = (V, E)$. With the same notations as in Section 2.2, recall that the eigenvalues of the normalized graph Laplacian satisfy $0 \leq \lambda_1 \leq \cdots \leq \lambda_n \leq 2$, and the corresponding eigenvectors $\{\psi_1, \ldots, \psi_n\}$ define an orthonormal family. In particular, $t \mapsto \exp(-t\lambda_k)$ is 2-Lipschitz continuous for $t > 0$. Let t, t' be two positive diffusion parameters. We have, for any $v \in V$:

$$\left| \sum_{k=1}^{n} (\exp(-t\lambda_k) - \exp(-t'\lambda_k))\psi_k(v)^2 \right| \leqslant 2 \cdot |t' - t| \underbrace{\sum_{k=1}^{n} \psi_k(v)^2}_{=1}.$$

Thus in particular,

$$\sup_{v \in V} |\mathrm{hks}_{G,t}(v) - \mathrm{hks}_{G,t'}(v)| \leqslant 2|t - t'|.$$

As in the previous proof, we conclude using the stability of persistence diagrams w.r.t. the bottleneck distance (see Thm. A.2).

B DATASETS DESCRIPTION

Tables 3 and 4 summarize key information of each dataset for both our experiments. We also provide in Figure 4 an illustration of the orbits we generated in Section 3.2.

C COMPLEMENTARY EXPERIMENTAL RESULTS

C.1 Weight learning

Figure 5 provides an illustration of the weight grid w learned after training on the `MUTAG` dataset. Roughly speaking, activated cells highlight the areas of the plane where the presence of points was discriminating in the classification process. These learned grids thus emphasize the points of the persistence diagrams that matter w.r.t. learning task.

Dataset	Nb of orbit observed	Number of classes	Number of points per orbit
ORBIT5K	5,000	5	1,000
ORBIT100K	100,000	5	1,000

Table 3: Description of the two orbits dataset we generated. The five classes correspond to the five parameter choices for $r \in \{2.5, 3.5, 4.0, 4.1, 4.3\}$. In both ORBIT5K and ORBIT100K, classes are balanced.

Dataset	Nb graphs	Nb classes	Av. nodes	Av. Edges	Av. β_0	Av. β_1
REDDIT5K	5,000	5	508.5	594.9	3.71	90.1
REDDIT12K	12,000	11	391.4	456.9	2.8	68.29
COLLAB	5,000	3	74.5	2457.5	1.0	2383.7
IMDB-B	1,000	2	19.77	96.53	1.0	77.76
IMDB-M	1,500	3	13.00	65.94	1.0	53.93
COX2	467	2	41.22	43.45	1.0	3.22
DHFR	756	2	42.43	44.54	1.0	3.12
MUTAG	188	2	17.93	19.79	1.0	2.86
PROTEINS	1,113	2	39.06	72.82	1.08	34.84
NCI1	4,110	2	29.87	32.30	1.19	3.62
NCI109	4,127	2	29.68	32.13	1.20	3.64

Table 4: Datasets description. β_0 (resp. β_1) stands for the 0th-Betti-number (resp. 1st), that is the number of connected components (resp. cycles) in a graph. In particular, an average $\beta_0 = 1.0$ means that all graph in the dataset are connected, and in this case $\beta_1 = \#\{\text{edges}\} - \#\{\text{nodes}\}$.

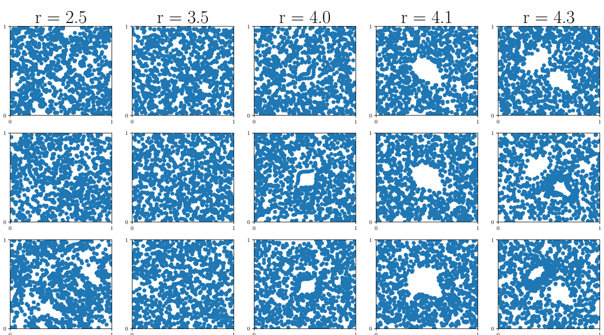

Figure 4: Some example of orbits generated by the different choices of r (three simulations are represented for the different values of r).

C.2 Selection of HKS diffusion parameter

As stated in Theorem 2.3, for a fixed graph G, the function $t \mapsto \mathrm{Dg}(G, t)$ is 2-Lipschitz continuous with respect to the bottleneck distance between persistence diagrams. Informally (see Supplementary Material, Section A for a formal definition), it means that the points of $\mathrm{Dg}(G, t)$ must move smoothly with respect to t. This is experimentally illustrated in Figure 7, where we plot the four diagrams built from a graph of the MUTAG dataset.

As mentioned in Section 2.2, the parameter t can also be treated as a trainable parameter that is optimized during the learning. In our experiment, however, it does not prove to be worth it. Indeed, our diagrams are not particularly sensitive to the choice of t, and thus fixing some t sampled in log-scale is enough. Figure 6 illustrates the evolution of parameter t over 40 epochs when trained on the MUTAG dataset (one epoch correspond to a stochastic gradient descent performed on the whole dataset). As one can see, parameter t converges quickly. More importantly, it remains almost constant when initialized at $t_0 = 10.0$, suggesting that this choice is a (locally) optimal one. Fortunately, this is the parameter we use in our experiment (see Table 5). On the other hand, each time t is updated (that is, at each epoch), one must recompute the diagrams for all the graphs in the training set, significantly increasing the running time of the algorithm.

C.3 Experimental settings

Input data was fed to the network with mini-batches of size 128. For each dataset, various parameters are given (extended persistence diagrams, neural network architecture, optimizers, etc.) that were used to obtain the scores from Table 2. In Table 5, we use the following shortcuts:

- Alpha_d: persistence diagrams obtained with Gudhi's d-dimensional AlphaComplex filtration.
- hks_t: extended persistence diagram obtained with HKS on the graph with parameter t.
- $\mathrm{prom}(k)$: preprocessing step selecting the k points that are the farthest away from the diagonal.
- PersLay channel $\mathrm{Im}(p,\ (a,\ b),\ q,\ \texttt{op})$ stands for a function ϕ obtained by using a Gaussian point

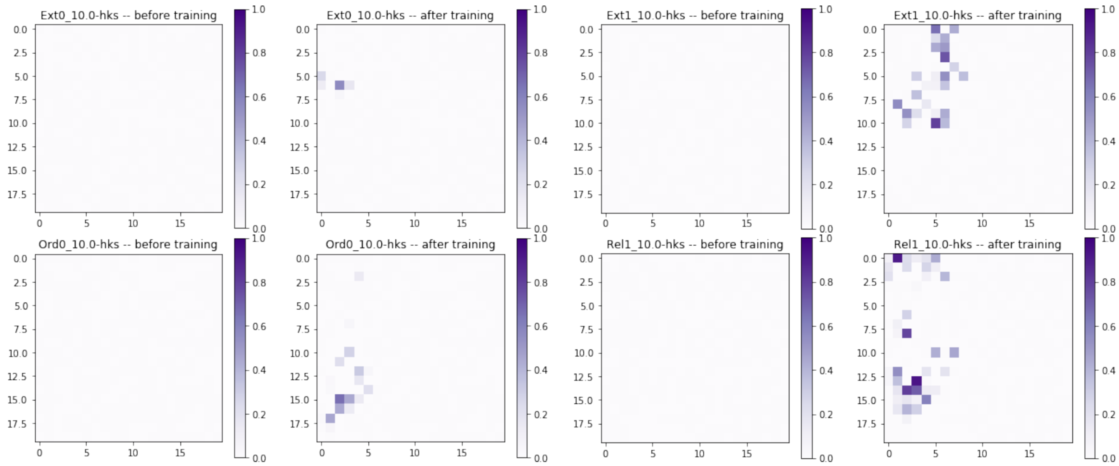

Figure 5: Weight function w when chosen to be a grid with size 20×20 before and after training (`MUTAG` dataset). Here, Ord0, Rel1, Ext0, and Ext1 denote the extended diagrams corresponding to downwards branches, upwards branches, connected components and loops respectively (cf Section 2.1).

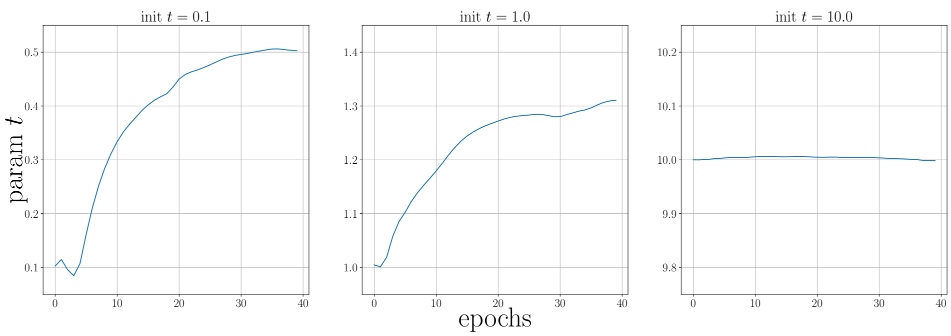

Figure 6: Evolution of HKS parameter t when considered as a trainable variable (i.e. differentiating $t \mapsto \mathrm{Dg}(G, t)$ for all G) across 40 epochs for three different initializations of t, namely 0.1, 1 and 10, on the `MUTAG` dataset.

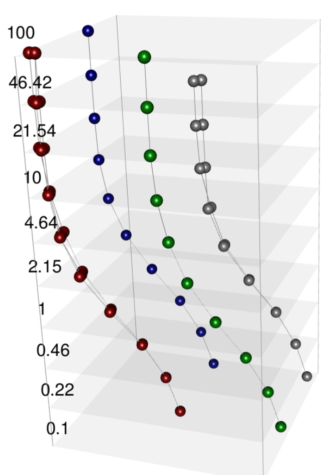

Figure 7: Evolution of $t \mapsto \mathrm{Dg}(G, t))$ for one graph from the `MUTAG` dataset ($t \in [0.1, 100]$, t in log-scale).

transformation ϕ_Γ sampled on $(p \times p)$ grid on the unit square followed by a convolution with a filters of size $b \times b$, for a weight function w optimized on a $(q \times q)$ grid and for an operation `op`.

- PersLay channel $\mathrm{Pm}(d_1,\ d_2,\ q,\ \texttt{op})$ stands for a function ϕ obtained by using a line point transformation ϕ_L with d_1 lines followed by a permutation equivariant function (Zaheer et al., 2017) in dimension d_2, for a weight function w optimized on a $(q \times q)$ grid and for an operation `op`.
- $\mathrm{adam}(\lambda, d, e)$ stands for the ADAM optimizer (Kingma & Ba, 2014) with learning rate λ, using an Exponential Moving Average[5] with decay rate d, and run during e epochs.

C.4 Hyper-parameters influence

As our approach mix our topological features and some standard graph features, we provide two ablations stud-

[5] `https://www.tensorflow.org/api_docs/python/tf/train/ExponentialMovingAverage`

Dataset	Func. used	PD preproc.	PersLay	Optim.
ORBIT5K	Alpha_0, Alpha_1	prom(500)	Pm(25,25,10,top-5)	adam(0.01, 0., 300)
ORBIT100K	Alpha_0, Alpha_1	prom(500)	Pm(25,25,10,top-5)	adam(0.01, 0., 300)
REDDIT5K	$\text{hks}_{1.0}$	prom(500)	Pm(25,25,10,sum)	adam(0.01, 0.99, 500)
REDDIT12K	$\text{hks}_{1.0}$	prom(500)	Pm(5,5,10,sum)	adam(0.01, 0.99, 1000)
COLLAB	$\text{hks}_{0.1}$, hks_{10}	prom(500)	Pm(5,5,10,sum)	adam(0.01, 0.9, 1000)
IMDB-B	$\text{hks}_{0.1}$, hks_{10}	prom(500)	Im(20,(10,2),20,sum)	adam(0.01, 0.9, 500)
IMDB-M	$\text{hks}_{0.1}$, hks_{10}	prom(500)	Im(10,(10,2),10,sum)	adam(0.01, 0.9, 500)
COX2	$\text{hks}_{0.1}$, hks_{10}	—	Im(20,(10,2),20,sum)	adam(0.01, 0.9, 500)
DHFR	$\text{hks}_{0.1}$, hks_{10}	—	Im(20,(10,2),20,sum)	adam(0.01, 0.9, 500)
MUTAG	hks_{10}	—	Im(20,(10,2),10,sum)	adam(0.01, 0.9, 100)
PROTEINS	hks_{10}	prom(500)	Im(15,(10,2),10,sum)	adam(0.01, 0.9, 70)
NCI1	$\text{hks}_{0.1}$, hks_{10}	—	Pm(25,25,10,sum)	adam(0.01, 0.9, 300)
NCI109	$\text{hks}_{0.1}$, hks_{10}	—	Pm(25,25,10,sum)	adam(0.01, 0.9, 300)

Table 5: Settings used to generate our experimental results.

		Grid size for trainable weights w(p)						**Point transformation ϕ**			**Perm op**	
		None	2×2	5×5	10×10	20×20	50×50	Gaussian	line	triangle	Sum	Max
MUTAG	Train/Test acc (%)	92.3/88.9	91.1/88.8	91.7/89.6	92.3/**89.9**	93.7/88.3	94.1/87.7	92.5/**89.7**	89.2/84.2	91.5/85.0	92.3/**89.5**	91.9/87.4
	Run time, CPU (s)	2.30	2.77	2.79	2.77	2.77	2.78	2.80	5.91	4.42	2.75	2.82
COLLAB	Train/Test acc (%)	76.5/75.3	78.6/75.8	79.0/76.2	80.0/**76.5**	83.5/73.9	94.0/71.3	79.7/75.3	79.9/**76.1**	79.4/74.7	80.0/**76.4**	78.8/75.0
	Run time, GPU (s)	26.0	40.4	43.5	43.8	44.1	45.6	45.8	54.0	61.4	44.3	48.1

Table 6: *Influence of hyper-parameters and ablation study.* When varying a single hyper-parameter (e.g. grid size), all the others (e.g. perm op) are fixed to the values described in Supplementary Material, Table 5. Accuracies and running times are averaged over 100 runs (i5-8350U 1.70GHz CPU for the small MUTAG dataset, P100 GPU for the large COLLAB one). Bold-blue font refers to the experimental setting used in Section 4.

ies. In Table 7, the column "Spectral" reports the test accuracies obtained by using only these additional features, while the column "PD alone" records the accuracies obtained with the extended and ordinary persistence diagrams alone. As ordinary persistence only encodes the connectivity properties of graphs, a gap in performance between extended and ordinary persistence can be interpreted as 1-dimensional features (i.e. loops) being informative for classification purpose. It also reports the standard deviations, that were omitted in 2 for the sake of clarity.

Similarly, we give in Table 6 the influence of the grid size that we choose as weight function w. In particular, we also perform an ablation study: grid size being `None` meaning that we enforce $w(p) = 1$ for all p. As expected, increasing the grid size improves train accuracy but leads to overfitting for too large values. However, this increase has only a small impact on running times whereas not using any grid significantly lowers it.

Finally, Figure 8 illustrates the variation of accuracy for both `MUTAG` and `COLLAB` datasets when varying the HKS parameter t used when generating the extended persistence diagrams. One can see that the accuracy reached on `MUTAG` does not depend on the choice of t, which could intuitively be explained by the small size of the graphs in this dataset, making the t parameter not very relevant. Experiments are performed on a single 10-fold, with 100 epochs. Parameters of PersLay are set to `Im(20,(),20)` for this experiment.

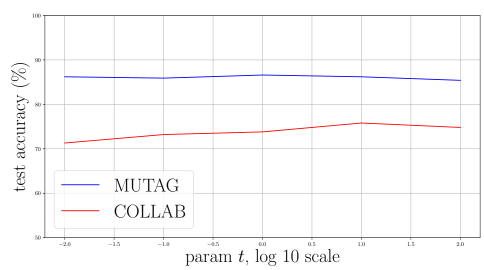

Figure 8: Variation of test accuracy for `MUTAG` and `COLLAB` dataset when varying HKS parameter t between 10^{-2} and 10^2 (log-10 scale).

	Spectral alone	PD alone		PersLay
		Extended	Ordinary	
REDDIT5K	49.7(±0.3)	55.0	52.5	55.6(±0.3)
REDDIT12K	39.7(±0.1)	44.2	40.1	47.7(±0.2)
COLLAB	67.8(±0.2)	71.6	69.2	76.4(±0.4)
IMDB-B	67.6(±0.6)	68.8	64.7	71.2(±0.7)
IMDB-M	44.5(±0.4)	48.2	42.0	48.8(±0.6)
COX2 *	78.2(±1.3)	81.5	79.0	80.9(±1.0)
DHFR *	69.5(±1.0)	78.2	71.8	80.3(±0.8)
MUTAG *	85.8(±1.3)	85.1	70.2	89.8(±0.9)
PROTEINS *	73.5(±0.3)	72.2	69.7	74.8(±0.3)
NCI1 *	65.3(±0.2)	72.3	68.9	73.5(±0.3)
NCI109 *	64.9(±0.2)	67.0	66.2	69.5(±0.3)

Table 7: Complementary report of experimental results.