Scalable and consistent embedding of probability measures into Hilbert spaces via measure quantization

Erell Gachon
University of Bordeaux

Scalable and consistent embedding of probability measures into Hilbert spaces via measure quantization

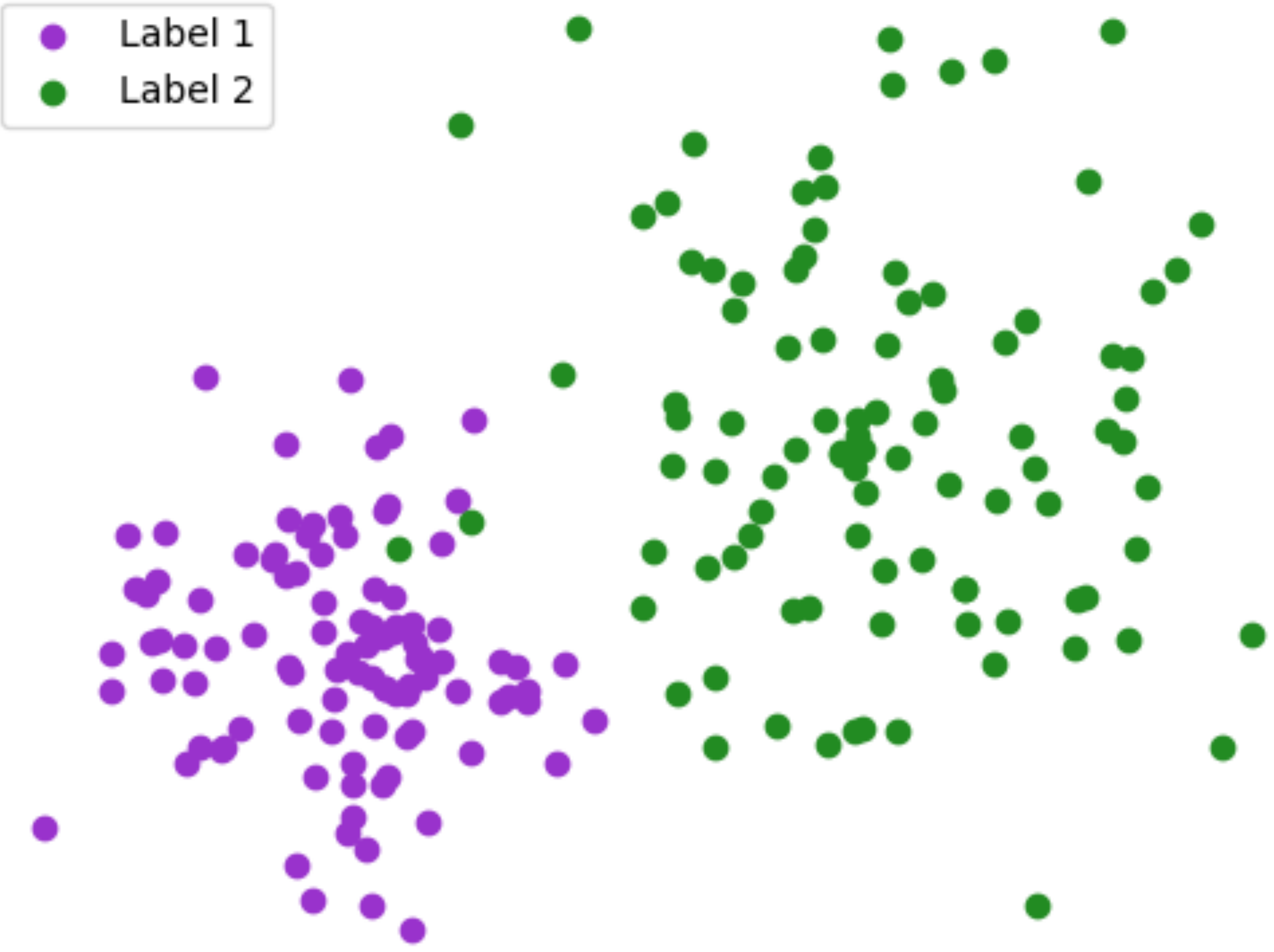
Erell Gachon

University of Bordeaux

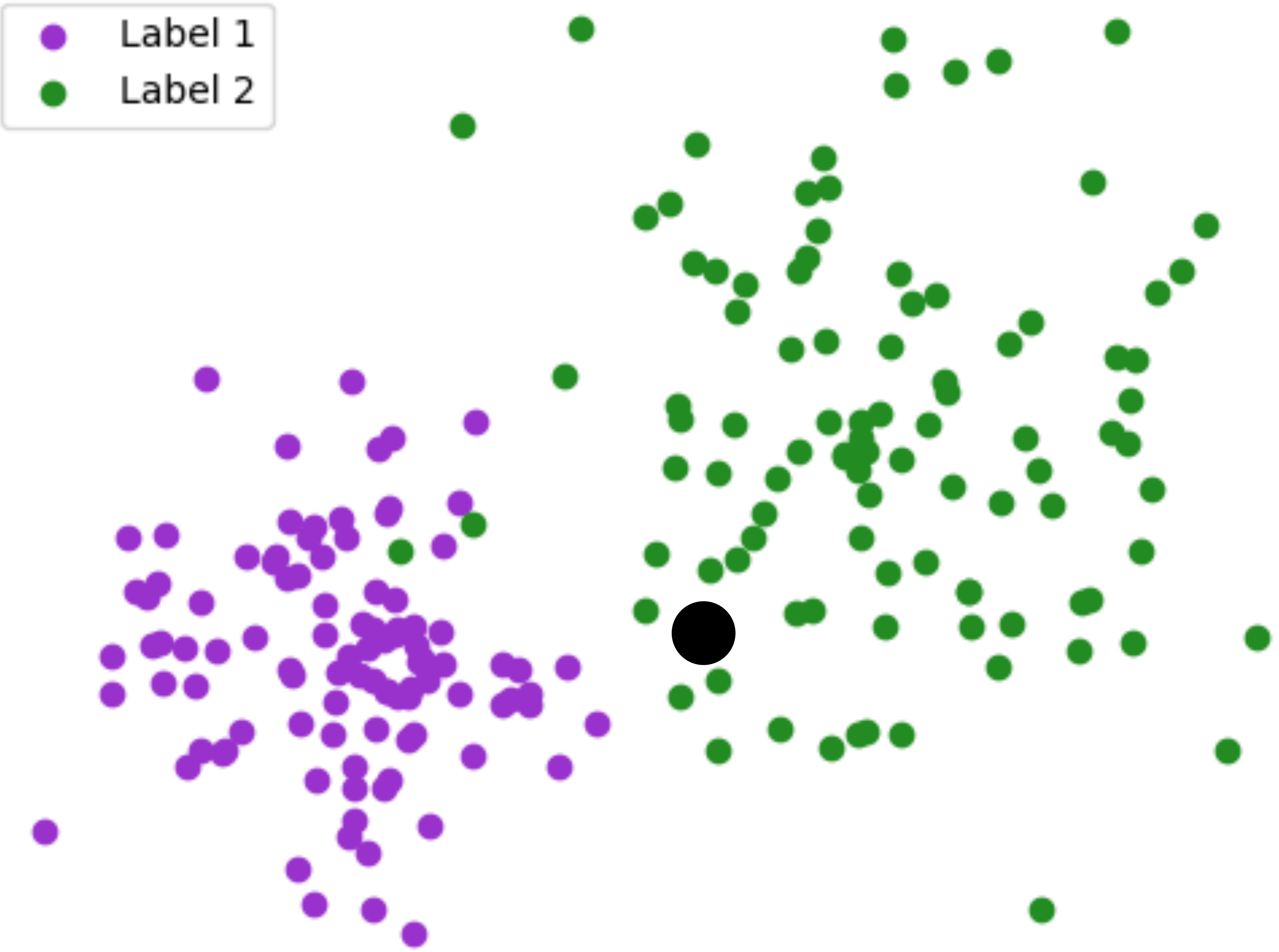
Elsa Cazelles

Jérémie Bigot

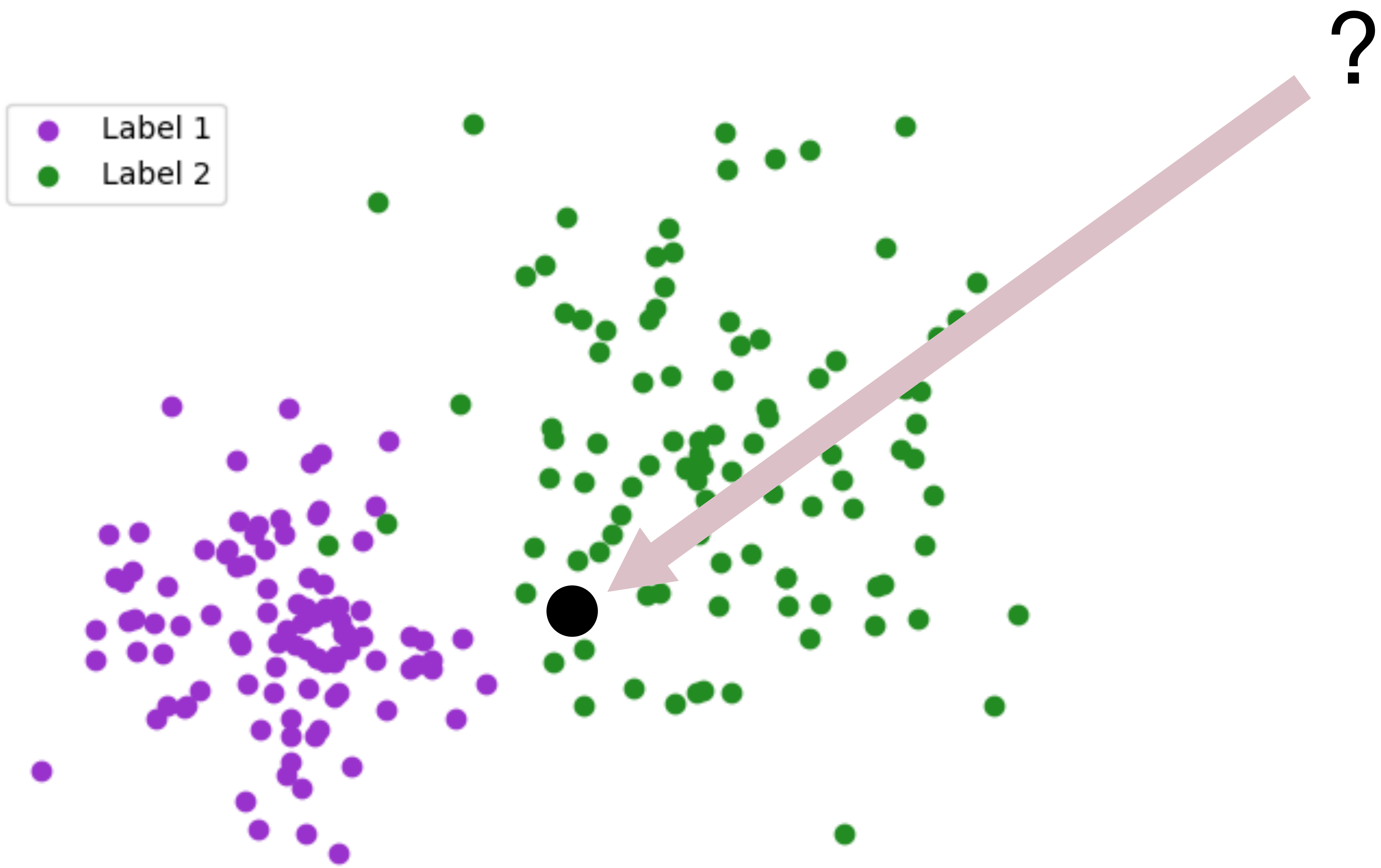
Statistical learning in a standard context



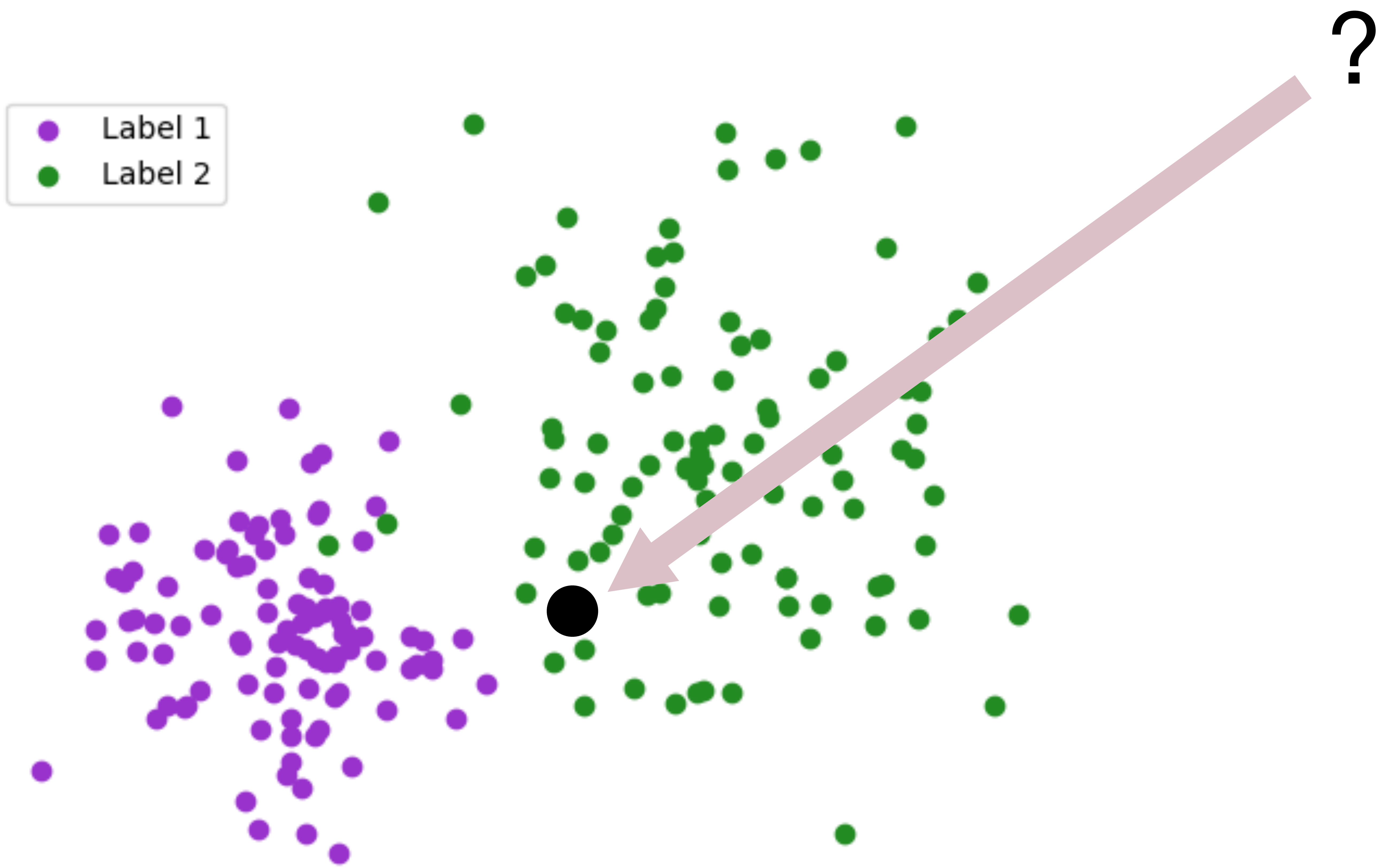
Statistical learning in a standard context



Statistical learning in a standard context

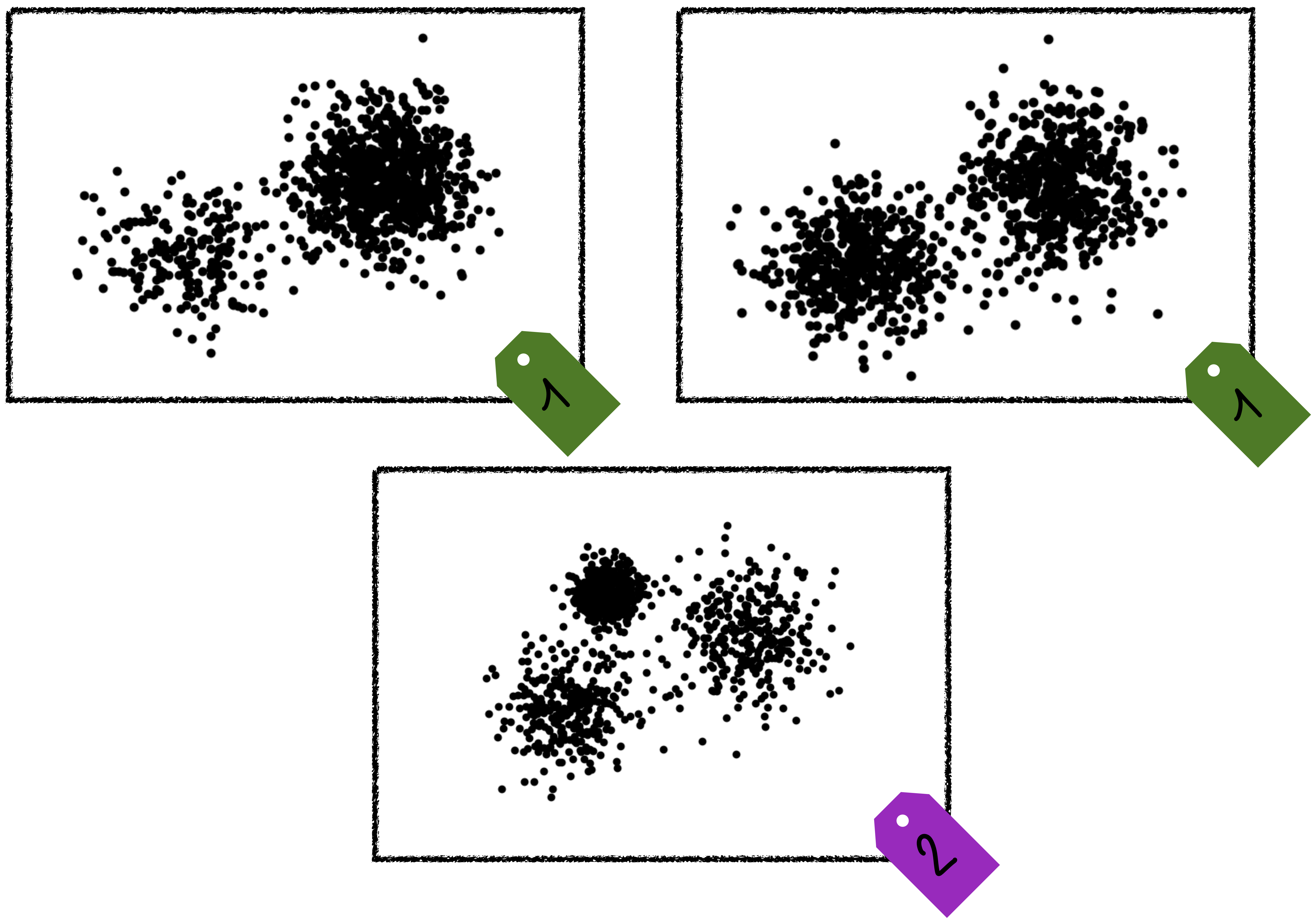


Statistical learning in a standard context

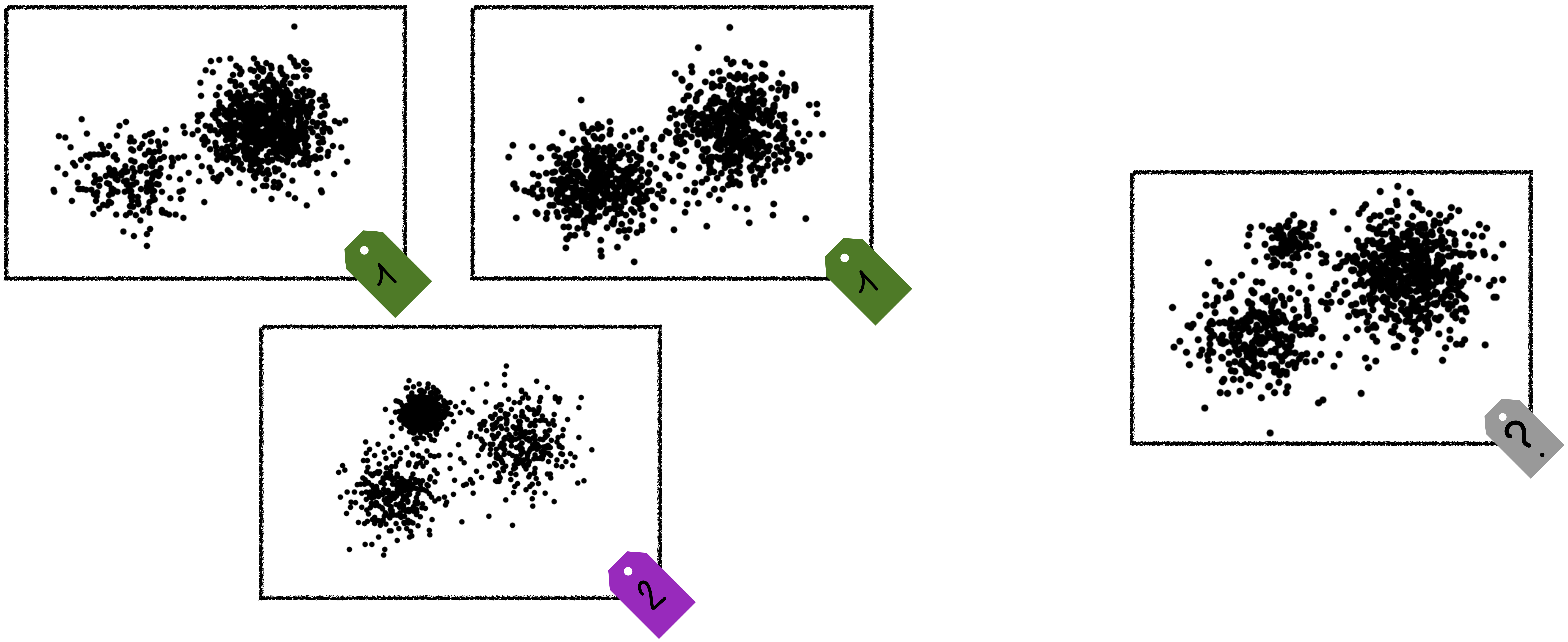


Data : N points in $\mathbb{R}^d$

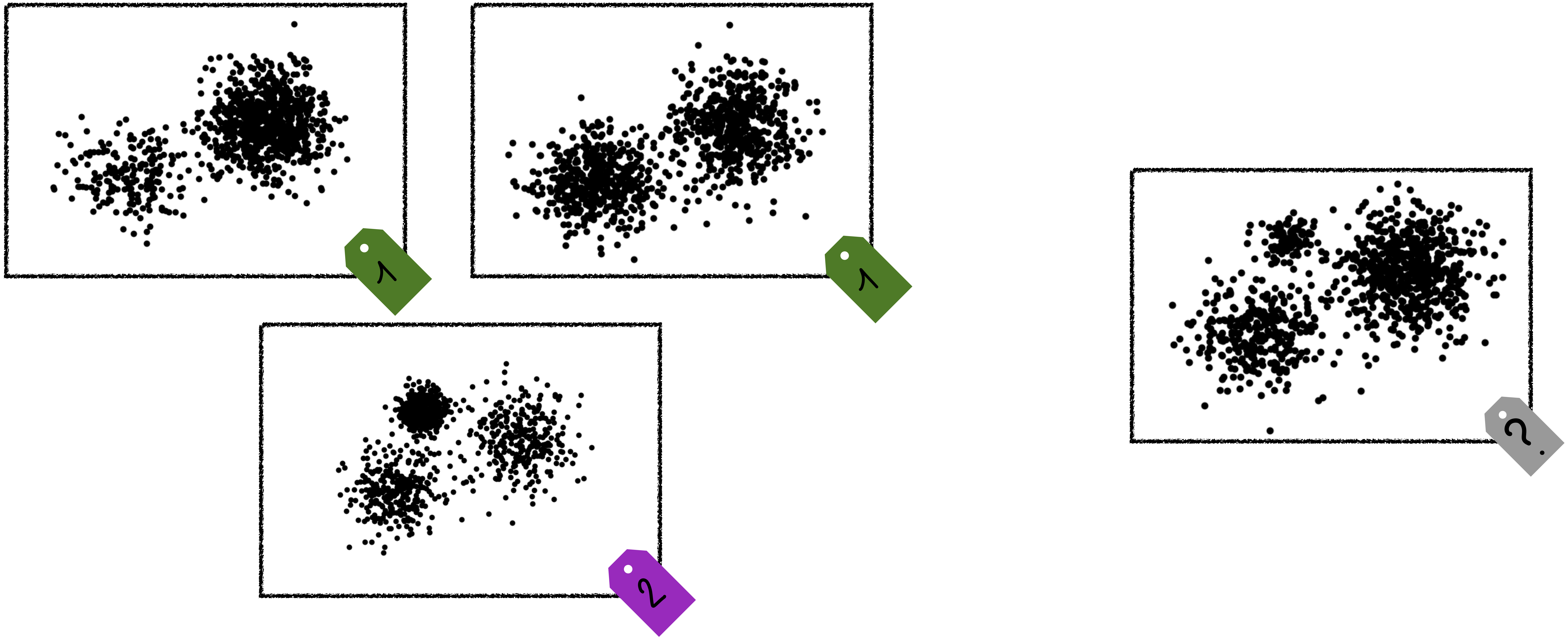
Our context



Our context



Our context



Data : N probability measures $(\mu^{(i)})_{i=1}^{N}$ supported on a compact set $\mathscr{X}$ of $\mathbb{R}^d$

Motivations : flow cytometry

Motivations : flow cytometry

Flow cytometry dataset : point cloud of m points (cells) in dimension d

	FS INT	FS TOF	SS PEAK	SS INT	SS TOF	CD34 FITC	CD13 PE	CD38 ECD	CD7 PerCP C5.5	CD33 PC7	CD56 APC	FL7 INT	CD117 APC750	HLA-DR PB	CD45 KO
0	410821	97	27712	191362	102	1.094111	1.915423	2.720374	1.267352	2.848508	1.364131	0.790964	1.011507	3.013860	2.795701
1	229951	105	25360	192800	116	1.039640	2.108282	1.574634	1.687131	2.489072	2.276289	0.945842	0.464528	1.699834	2.333237
2	497906	102	38080	288257	112	1.147993	1.618946	2.727084	1.148997	3.003344	2.652774	1.061185	1.446943	1.654458	2.974949
3	553610	103	27696	206529	113	1.017119	1.524717	2.843736	1.244978	2.789574	1.888895	1.438322	0.590620	3.350601	2.556790
4	393809	92	25424	177654	105	0.988503	1.316766	2.589928	1.231925	2.773919	2.955182	0.891658	0.868261	1.901177	2.723740
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
443405	595994	93	54208	377015	105	1.077221	1.056086	2.810727	1.852034	2.787875	3.024863	0.705716	1.319084	3.304823	2.782335
443406	531817	95	29920	225140	113	1.280923	1.447887	2.536496	1.455544	3.276666	2.071142	0.432120	0.919439	2.812525	3.595473
443407	486870	92	31616	226768	107	1.233555	1.482677	2.505156	1.663138	3.141107	1.802799	0.909684	1.140449	2.229564	3.265084
443408	256153	97	29136	219459	110	1.258310	2.019732	1.256600	1.281009	2.354184	2.501133	0.495965	1.030419	2.111731	2.895691
443409	508035	104	21056	156636	108	1.002701	0.739097	3.336932	1.105585	2.828619	0.993095	0.885740	1.463521	3.388605	2.673081

Example of flow cytometry measurements from a sample of a patient diagnosed with acute myeloid leukemia.

Motivations : flow cytometry

Flow cytometry dataset : point cloud of m points (cells) in dimension d

$m \approx 10^5$ $d \approx 10$

	FS INT	FS TOF	SS PEAK	SS INT	SS TOF	CD34 FITC	CD13 PE	CD38 ECD	CD7 PerCP C5.5	CD33 PC7	CD56 APC	FL7 INT	CD117 APC750	HLA-DR PB	CD45 KO
0	410821	97	27712	191362	102	1.094111	1.915423	2.720374	1.267352	2.848508	1.364131	0.790964	1.011507	3.013860	2.795701
1	229951	105	25360	192800	116	1.039640	2.108282	1.574634	1.687131	2.489072	2.276289	0.945842	0.464528	1.699834	2.333237
2	497906	102	38080	288257	112	1.147993	1.618946	2.727084	1.148997	3.003344	2.652774	1.061185	1.446943	1.654458	2.974949
3	553610	103	27696	206529	113	1.017119	1.524717	2.843736	1.244978	2.789574	1.888895	1.438322	0.590620	3.350601	2.556790
4	393809	92	25424	177654	105	0.988503	1.316766	2.589928	1.231925	2.773919	2.955182	0.891658	0.868261	1.901177	2.723740
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
443405	595994	93	54208	377015	105	1.077221	1.056086	2.810727	1.852034	2.787875	3.024863	0.705716	1.319084	3.304823	2.782335
443406	531817	95	29920	225140	113	1.280923	1.447887	2.536496	1.455544	3.276666	2.071142	0.432120	0.919439	2.812525	3.595473
443407	486870	92	31616	226768	107	1.233555	1.482677	2.505156	1.663138	3.141107	1.802799	0.909684	1.140449	2.229564	3.265084
443408	256153	97	29136	219459	110	1.258310	2.019732	1.256600	1.281009	2.354184	2.501133	0.495965	1.030419	2.111731	2.895691
443409	508035	104	21056	156636	108	1.002701	0.739097	3.336932	1.105585	2.828619	0.993095	0.885740	1.463521	3.388605	2.673081

Example of flow cytometry measurements from a sample of a patient diagnosed with acute myeloid leukemia.

Motivations : flow cytometry

Flow cytometry dataset : point cloud of m points (cells) in dimension d

$m \approx 10^5$ $\qquad\qquad$ $d \approx 10$

What kind of information do we want to learn ?

	FS INT	FS TOF	SS PEAK	SS INT	SS TOF	CD34 FITC	CD13 PE	CD38 ECD	CD7 PerCP C5.5	CD33 PC7	CD56 APC	FL7 INT	CD117 APC750	HLA-DR PB	CD45 KO
0	410821	97	27712	191362	102	1.094111	1.915423	2.720374	1.267352	2.848508	1.364131	0.790964	1.011507	3.013860	2.795701
1	229951	105	25360	192800	116	1.039640	2.108282	1.574634	1.687131	2.489072	2.276289	0.945842	0.464528	1.699834	2.333237
2	497906	102	38080	288257	112	1.147993	1.618946	2.727084	1.148997	3.003344	2.652774	1.061185	1.446943	1.654458	2.974949
3	553610	103	27696	206529	113	1.017119	1.524717	2.843736	1.244978	2.789574	1.888895	1.438322	0.590620	3.350601	2.556790
4	393809	92	25424	177654	105	0.988503	1.316766	2.589928	1.231925	2.773919	2.955182	0.891658	0.868261	1.901177	2.723740
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
443405	595994	93	54208	377015	105	1.077221	1.056086	2.810727	1.852034	2.787875	3.024863	0.705716	1.319084	3.304823	2.782335
443406	531817	95	29920	225140	113	1.280923	1.447887	2.536496	1.455544	3.276666	2.071142	0.432120	0.919439	2.812525	3.595473
443407	486870	92	31616	226768	107	1.233555	1.482677	2.505156	1.663138	3.141107	1.802799	0.909684	1.140449	2.229564	3.265084
443408	256153	97	29136	219459	110	1.258310	2.019732	1.256600	1.281009	2.354184	2.501133	0.495965	1.030419	2.111731	2.895691
443409	508035	104	21056	156636	108	1.002701	0.739097	3.336932	1.105585	2.828619	0.993095	0.885740	1.463521	3.388605	2.673081

Example of flow cytometry measurements from a sample of a patient diagnosed with acute myeloid leukemia.

Motivations : flow cytometry

Flow cytometry dataset : point cloud of m points (cells) in dimension d

$m \approx 10^5$ $\quad\quad\quad$ $d \approx 10$

What kind of information do we want to learn ?

	FS INT	FS TOF	SS PEAK	SS INT	SS TOF	CD34 FITC	CD13 PE	CD38 ECD	CD7 PerCP C5.5	CD33 PC7	CD56 APC	FL7 INT	CD117 APC750	HLA-DR PB	CD45 KO
0	410821	97	27712	191362	102	1.094111	1.915423	2.720374	1.267352	2.848508	1.364131	0.790964	1.011507	3.013860	2.795701
1	229951	105	25360	192800	116	1.039640	2.108282	1.574634	1.687131	2.489072	2.276289	0.945842	0.464528	1.699834	2.333237
2	497906	102	38080	288257	112	1.147993	1.618946	2.727084	1.148997	3.003344	2.652774	1.061185	1.446943	1.654458	2.974949
3	553610	103	27696	206529	113	1.017119	1.524717	2.843736	1.244978	2.789574	1.888895	1.438322	0.590620	3.350601	2.556790
4	393809	92	25424	177654	105	0.988503	1.316766	2.589928	1.231925	2.773919	2.955182	0.891658	0.868261	1.901177	2.723740
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
443405	595994	93	54208	377015	105	1.077221	1.056086	2.810727	1.852034	2.787875	3.024863	0.705716	1.319084	3.304823	2.782335
443406	531817	95	29920	225140	113	1.280923	1.447887	2.536496	1.455544	3.276666	2.071142	0.432120	0.919439	2.812525	3.595473
443407	486870	92	31616	226768	107	1.233555	1.482677	2.505156	1.663138	3.141107	1.802799	0.909684	1.140449	2.229564	3.265084
443408	256153	97	29136	219459	110	1.258310	2.019732	1.256600	1.281009	2.354184	2.501133	0.495965	1.030419	2.111731	2.895691
443409	508035	104	21056	156636	108	1.002701	0.739097	3.336932	1.105585	2.828619	0.993095	0.885740	1.463521	3.388605	2.673081

Example of flow cytometry measurements from a sample of a patient diagnosed with acute myeloid leukemia.

Classification

Does this patient have cancer ?

Motivations : flow cytometry

Flow cytometry dataset : point cloud of m points (cells) in dimension d

$m \approx 10^5$ $\qquad$ $d \approx 10$

What kind of information do we want to learn ?

	FS INT	FS TOF	SS PEAK	SS INT	SS TOF	CD34 FITC	CD13 PE	CD38 ECD	CD7 PerCP C5.5	CD33 PC7	CD56 APC	FL7 INT	CD117 APC750	HLA-DR PB	CD45 KO
0	410821	97	27712	191362	102	1.094111	1.915423	2.720374	1.267352	2.848508	1.364131	0.790964	1.011507	3.013860	2.795701
1	229951	105	25360	192800	116	1.039640	2.108282	1.574634	1.687131	2.489072	2.276289	0.945842	0.464528	1.699834	2.333237
2	497906	102	38080	288257	112	1.147993	1.618946	2.727084	1.148997	3.003344	2.652774	1.061185	1.446943	1.654458	2.974949
3	553610	103	27696	206529	113	1.017119	1.524717	2.843736	1.244978	2.789574	1.888895	1.438322	0.590620	3.350601	2.556790
4	393809	92	25424	177654	105	0.988503	1.316766	2.589928	1.231925	2.773919	2.955182	0.891658	0.868261	1.901177	2.723740
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
443405	595994	93	54208	377015	105	1.077221	1.056086	2.810727	1.852034	2.787875	3.024863	0.705716	1.319084	3.304823	2.782335
443406	531817	95	29920	225140	113	1.280923	1.447887	2.536496	1.455544	3.276666	2.071142	0.432120	0.919439	2.812525	3.595473
443407	486870	92	31616	226768	107	1.233555	1.482677	2.505156	1.663138	3.141107	1.802799	0.909684	1.140449	2.229564	3.265084
443408	256153	97	29136	219459	110	1.258310	2.019732	1.256600	1.281009	2.354184	2.501133	0.495965	1.030419	2.111731	2.895691
443409	508035	104	21056	156636	108	1.002701	0.739097	3.336932	1.105585	2.828619	0.993095	0.885740	1.463521	3.388605	2.673081

Example of flow cytometry measurements from a sample of a patient diagnosed with acute myeloid leukemia.

Classification

Does this patient have cancer ?

Regression

How many sick cells are in this sample ?

Motivations : flow cytometry

Flow cytometry dataset : point cloud of m points (cells) in dimension d

$m \approx 10^5$ $\qquad$ $d \approx 10$

What kind of information do we want to learn ?

	FS INT	FS TOF	SS PEAK	SS INT	SS TOF	CD34 FITC	CD13 PE	CD38 ECD	CD7 PerCP C5.5	CD33 PC7	CD56 APC	FL7 INT	CD117 APC750	HLA-DR PB	CD45 KO
0	410821	97	27712	191362	102	1.094111	1.915423	2.720374	1.267352	2.848508	1.364131	0.790964	1.011507	3.013860	2.795701
1	229951	105	25360	192800	116	1.039640	2.108282	1.574634	1.687131	2.489072	2.276289	0.945842	0.464528	1.699834	2.333237
2	497906	102	38080	288257	112	1.147993	1.618946	2.727084	1.148997	3.003344	2.652774	1.061185	1.446943	1.654458	2.974949
3	553610	103	27696	206529	113	1.017119	1.524717	2.843736	1.244978	2.789574	1.888895	1.438322	0.590620	3.350601	2.556790
4	393809	92	25424	177654	105	0.988503	1.316766	2.589928	1.231925	2.773919	2.955182	0.891658	0.868261	1.901177	2.723740
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
443405	595994	93	54208	377015	105	1.077221	1.056086	2.810727	1.852034	2.787875	3.024863	0.705716	1.319084	3.304823	2.782335
443406	531817	95	29920	225140	113	1.280923	1.447887	2.536496	1.455544	3.276666	2.071142	0.432120	0.919439	2.812525	3.595473
443407	486870	92	31616	226768	107	1.233555	1.482677	2.505156	1.663138	3.141107	1.802799	0.909684	1.140449	2.229564	3.265084
443408	256153	97	29136	219459	110	1.258310	2.019732	1.256600	1.281009	2.354184	2.501133	0.495965	1.030419	2.111731	2.895691
443409	508035	104	21056	156636	108	1.002701	0.739097	3.336932	1.105585	2.828619	0.993095	0.885740	1.463521	3.388605	2.673081

Example of flow cytometry measurements from a sample of a patient diagnosed with acute myeloid leukemia.

Classification

Does this patient have cancer ?

Regression

How many sick cells are in this sample ?

Data : N probability measures $(\mu^{(i)})_{i=1}^{N}$ *with large support* on a compact $\mathcal{X}$ of $\mathbb{R}^d$

How to perform statistical learning on probability measures ?

How to perform statistical learning on probability measures ?

$\mathscr{P}(\mathscr{X})$ is the set of probability measures on $\mathscr{X}$, endowed with the 2-Wasserstein distance W_2.

How to perform statistical learning on probability measures ?

$\mathscr{P}(\mathscr{X})$ is the set of probability measures on $\mathscr{X}$, endowed with the 2-Wasserstein distance W_2.

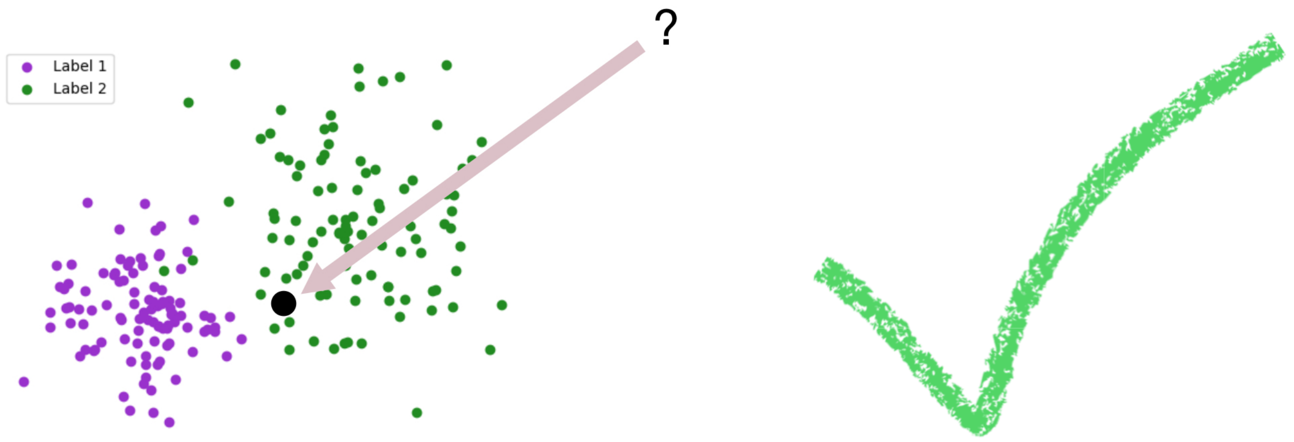
Classical machine learning algorithms are designed to handle:

How to perform statistical learning on probability measures ?

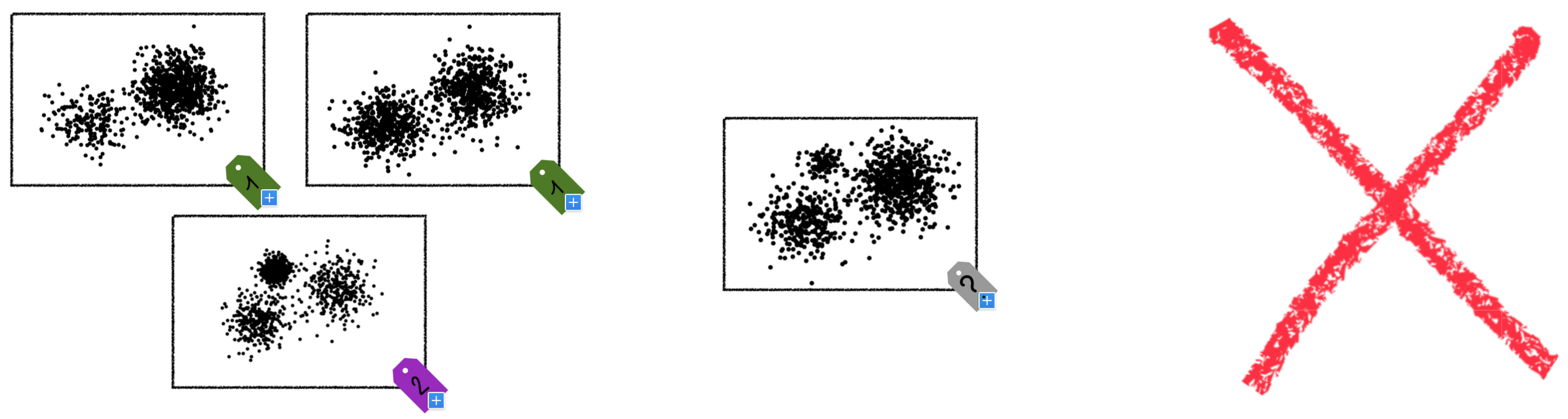
$\mathscr{P}(\mathscr{X})$ is the set of probability measures on $\mathscr{X}$, endowed with the 2-Wasserstein distance W_2.

Classical machine learning algorithms are designed to handle:

N sample points



N distributions

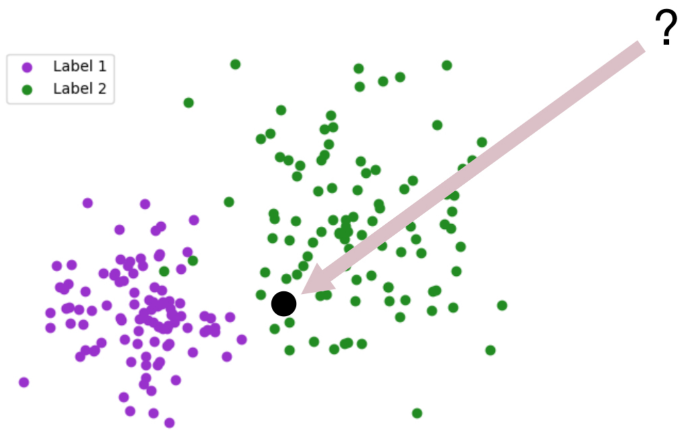


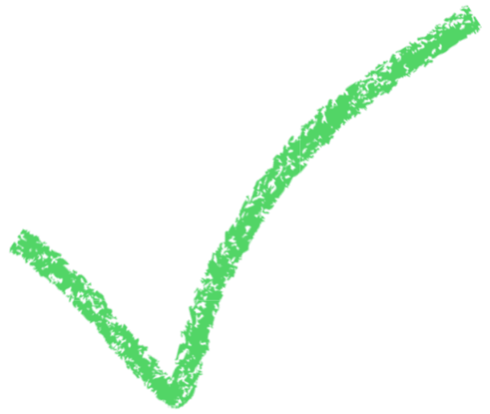
How to perform statistical learning on probability measures ?

$\mathscr{P}(\mathscr{X})$ is the set of probability measures on $\mathscr{X}$, endowed with the 2-Wasserstein distance W_2.

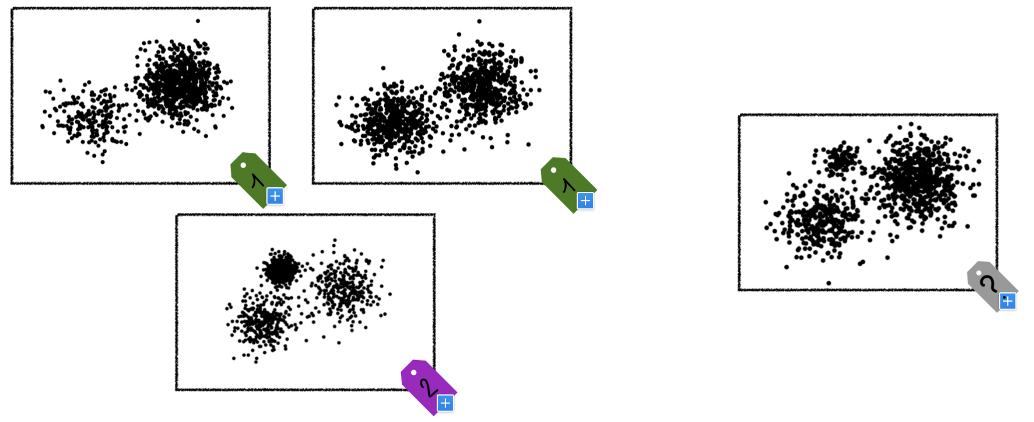
Classical machine learning algorithms are designed to handle:

N sample points





N distributions



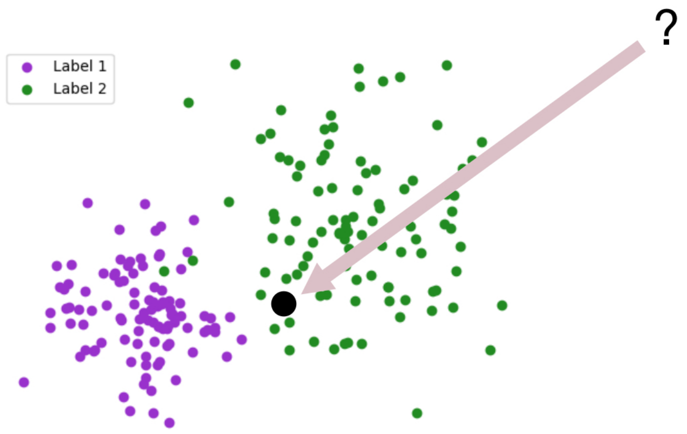
The key : inner-product!

How to perform statistical learning on probability measures ?

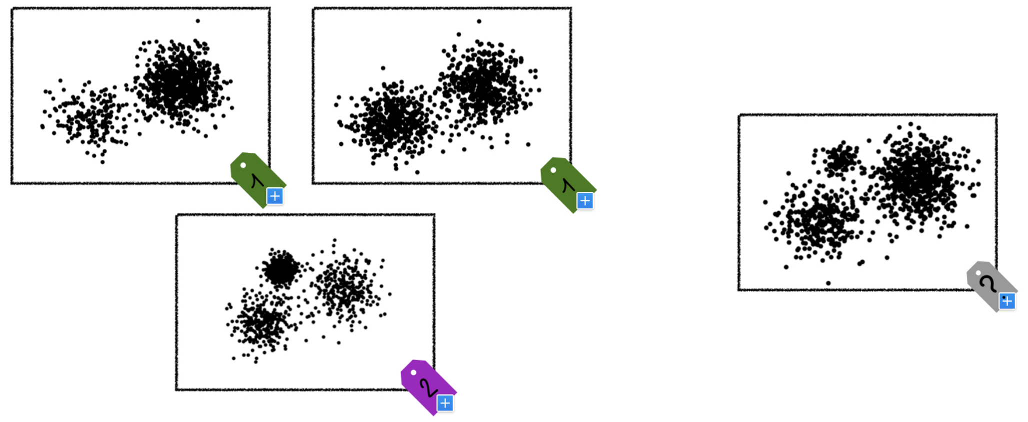
$\mathscr{P}(\mathscr{X})$ is the set of probability measures on $\mathscr{X}$, endowed with the 2-Wasserstein distance W_2.

Classical machine learning algorithms are designed to handle:

N sample points



N distributions



The key : inner-product!

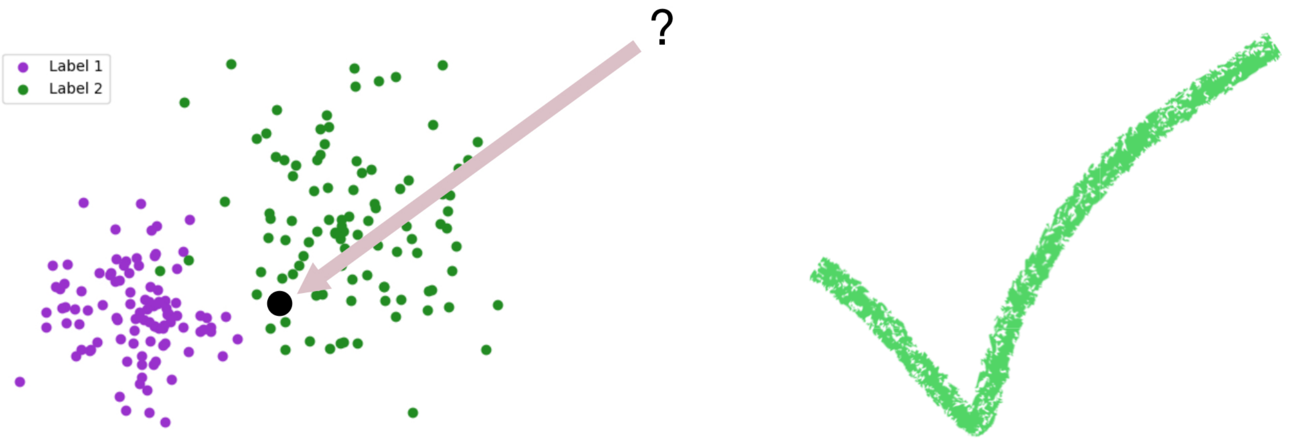
$(\mathscr{P}(\mathscr{X}), W_2)$ not a Hilbert space

How to perform statistical learning on probability measures ?

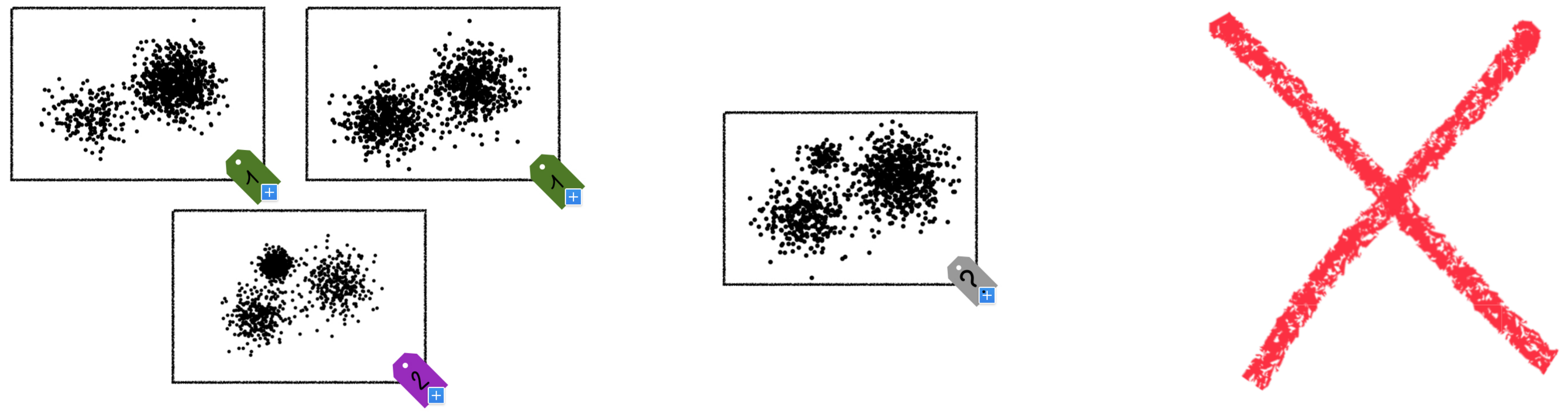
$\mathscr{P}(\mathscr{X})$ is the set of probability measures on $\mathscr{X}$, endowed with the 2-Wasserstein distance W_2.

Classical machine learning algorithms are designed to handle:

N sample points



N distributions



The key : inner-product!

$(\mathscr{P}(\mathscr{X}), W_2)$ not a Hilbert space

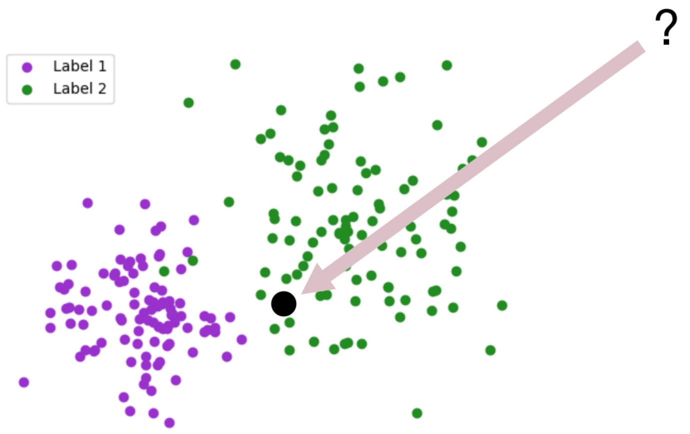
Solution: Embedding the probability measures into a Hilbert space using…

How to perform statistical learning on probability measures ?

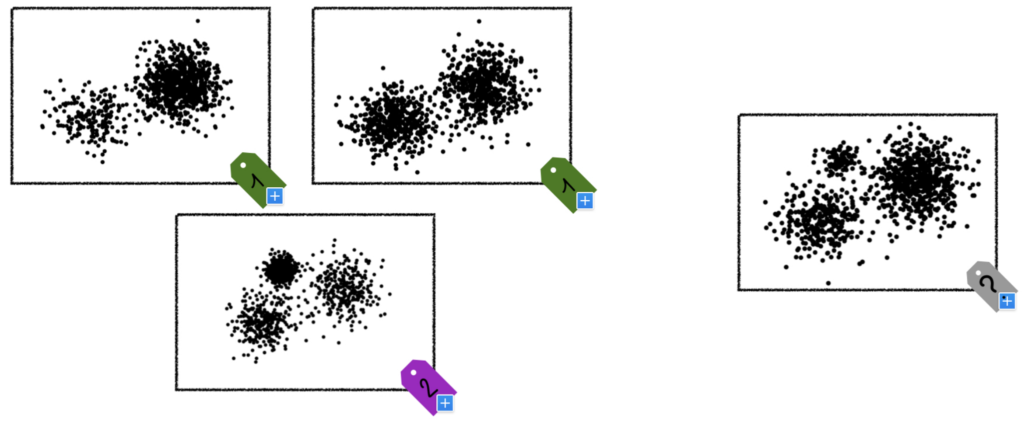
$\mathscr{P}(\mathscr{X})$ is the set of probability measures on $\mathscr{X}$, endowed with the 2-Wasserstein distance W_2.

Classical machine learning algorithms are designed to handle:

N sample points



N distributions



The key : inner-product!

$(\mathscr{P}(\mathscr{X}), W_2)$ not a Hilbert space

Solution: Embedding the probability measures into a Hilbert space using…

Linearized optimal transport

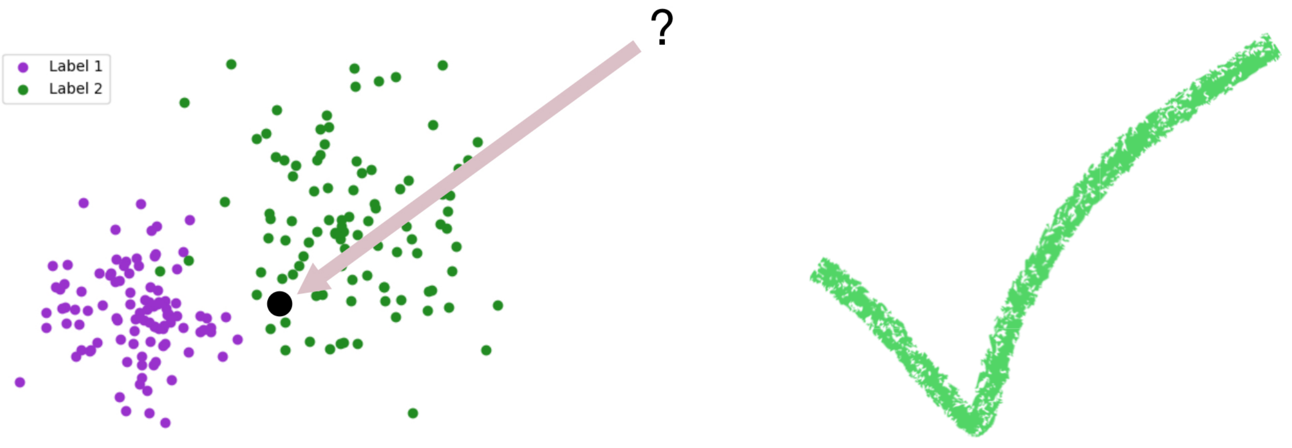
Kernel Mean Embedding

How to perform statistical learning on probability measures ?

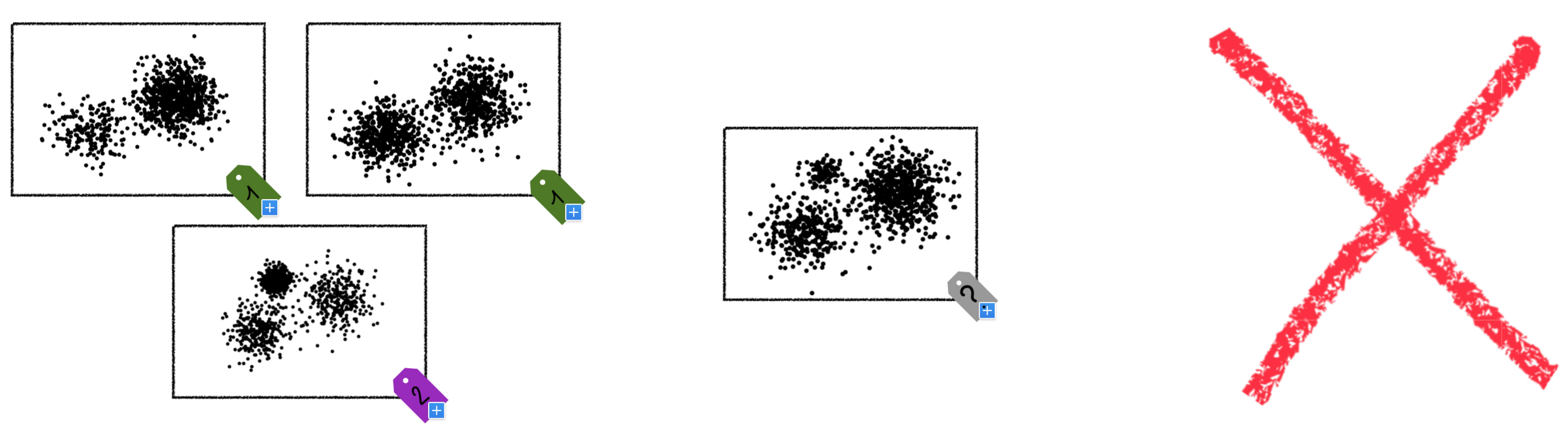
$\mathscr{P}(\mathscr{X})$ is the set of probability measures on $\mathscr{X}$, endowed with the 2-Wasserstein distance W_2.

Classical machine learning algorithms are designed to handle:

N sample points



N distributions



The key : inner-product!

$(\mathscr{P}(\mathscr{X}), W_2)$ not a Hilbert space

Solution: Embedding the probability measures into a Hilbert space using…

Linearized optimal transport

Kernel Mean Embedding

Too costly when $m > 10^4$!

How to perfo… al learning on pr… es ?

$\mathscr{P}(\mathscr{X})$ is the set of probability mea…

Classical machine le…

…istributions

(…

Sol…

Linearized … edding

Solution : reduce the support size of the $\mu^{(i)}$ with quantization

Quantization

Quantization

Definition. A K-points quantization of a probability measure μ aims at solving

$$\min_{a \in \Sigma_K, X \in (\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^{K} a_k \delta_{X_k}\Big)$$

Quantization

Definition. A K-points quantization of a probability measure μ aims at solving

$$\min_{a\in\Sigma_K, X\in(\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^{K} a_k\delta_{X_k}\Big)$$

For $X=(X_1,\cdots,X_K)\in(\mathbb{R}^d)^K$, minimiser a^* of

$$\min_{a\in\Sigma_K} W_2^2\Big(\mu, \sum_{k=1}^{K} a_k\delta_{X_k}\Big) \text{ verifies } a_k^* = \mu(V_{X_k})$$

$$V_{X_k} = \{y\in\mathbb{R}^d \mid \forall l\neq k, \|X_k-y\|^2 \leq \|X_l-y\|^2\}$$

Voronoi cell centered at X_k.

Quantization

Definition. A K-points quantization of a probability measure μ aims at solving

$$\min_{a\in\Sigma_K, X\in(\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^K a_k\delta_{X_k}\Big)$$

For $X = (X_1, \cdots, X_K) \in (\mathbb{R}^d)^K$, minimiser a^* of

$$\min_{a\in\Sigma_K} W_2^2\Big(\mu, \sum_{k=1}^K a_k\delta_{X_k}\Big) \text{ verifies } a_k^* = \mu(V_{X_k})$$

$$V_{X_k} = \{y \in \mathbb{R}^d \mid \forall l \neq k, \|X_k - y\|^2 \leq \|X_l - y\|^2\}$$

Voronoi cell centered at X_k.

$$\min_{a\in\Sigma_K, X\in(\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^K a_k\delta_{X_k}\Big) = \min_{X\in(\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^K \mu(V_{X_k})\delta_{X_k}\Big)$$

Quantization

Definition. A K-points quantization of a probability measure μ aims at solving

$$\min_{a\in\Sigma_K, X\in(\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^K a_k \delta_{X_k}\Big)$$

For $X = (X_1, \cdots, X_K) \in (\mathbb{R}^d)^K$, minimiser a^* of

$$\min_{a\in\Sigma_K} W_2^2\Big(\mu, \sum_{k=1}^K a_k \delta_{X_k}\Big) \text{ verifies } a_k^* = \mu(V_{X_k})$$

$$V_{X_k} = \{y \in \mathbb{R}^d \mid \forall l \neq k, \|X_k - y\|^2 \leq \|X_l - y\|^2\}$$

Voronoi cell centered at X_k.

$$\min_{a\in\Sigma_K, X\in(\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^K a_k \delta_{X_k}\Big) = \min_{X\in(\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^K \mu(V_{X_k}) \delta_{X_k}\Big)$$

Quantization

Definition. A K-points quantization of a probability measure μ aims at solving

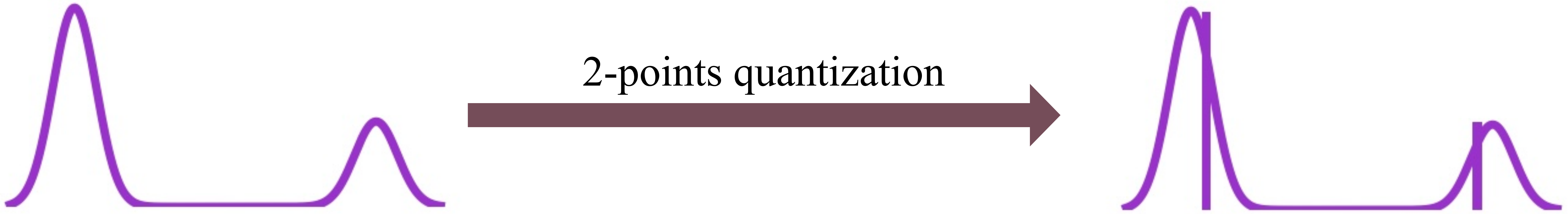
$$\min_{a\in\Sigma_K, X\in(\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^{K} a_k \delta_{X_k}\Big)$$

For $X = (X_1, \cdots, X_K) \in (\mathbb{R}^d)^K$, minimiser a^* of $\min_{a\in\Sigma_K} W_2^2\Big(\mu, \sum_{k=1}^{K} a_k \delta_{X_k}\Big)$ verifies $a_k^* = \mu(V_{X_k})$

$V_{X_k} = \{y \in \mathbb{R}^d \mid \forall l \neq k, \|X_k - y\|^2 \leq \|X_l - y\|^2\}$

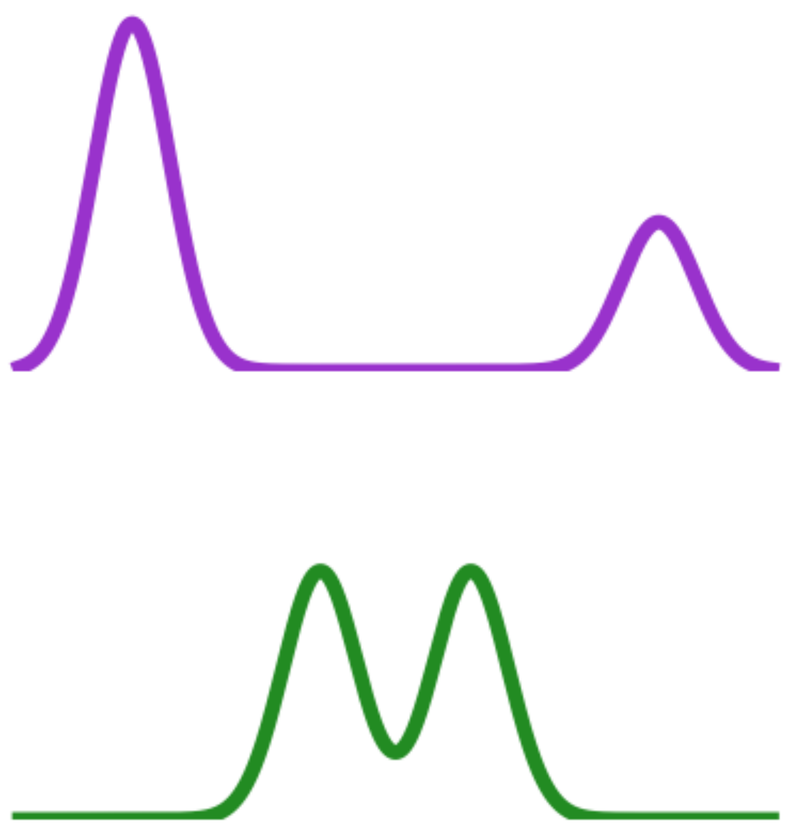
Voronoi cell centered at X_k.

$$\min_{a\in\Sigma_K, X\in(\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^{K} a_k \delta_{X_k}\Big) = \min_{X\in(\mathbb{R}^d)^K} W_2^2\Big(\mu, \sum_{k=1}^{K} \mu(V_{X_k}) \delta_{X_k}\Big)$$

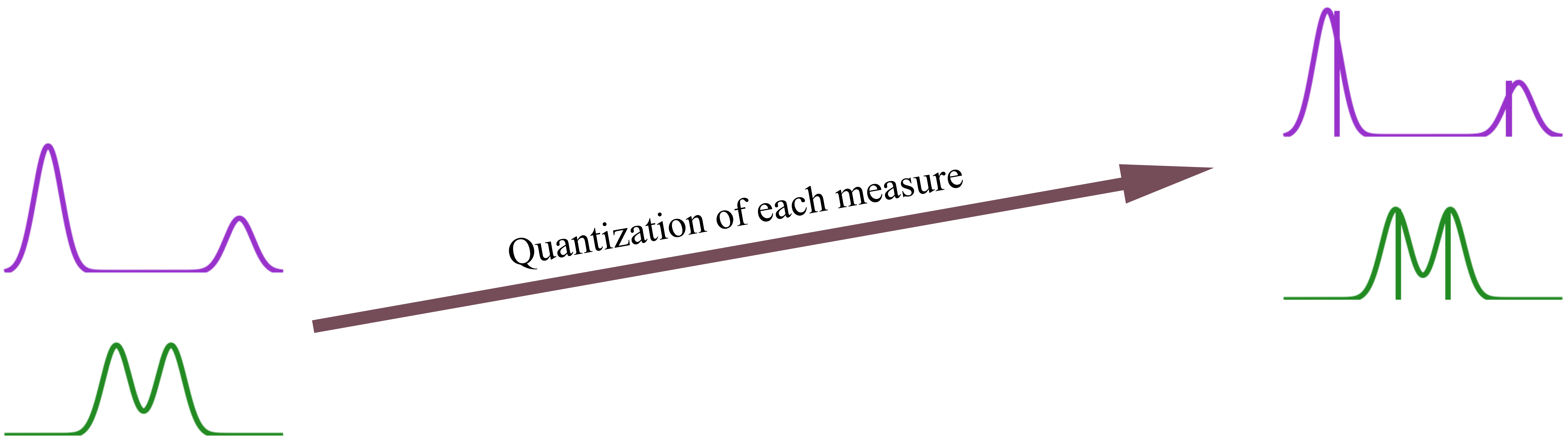


Quantization of each measure vs mean-measure quantization

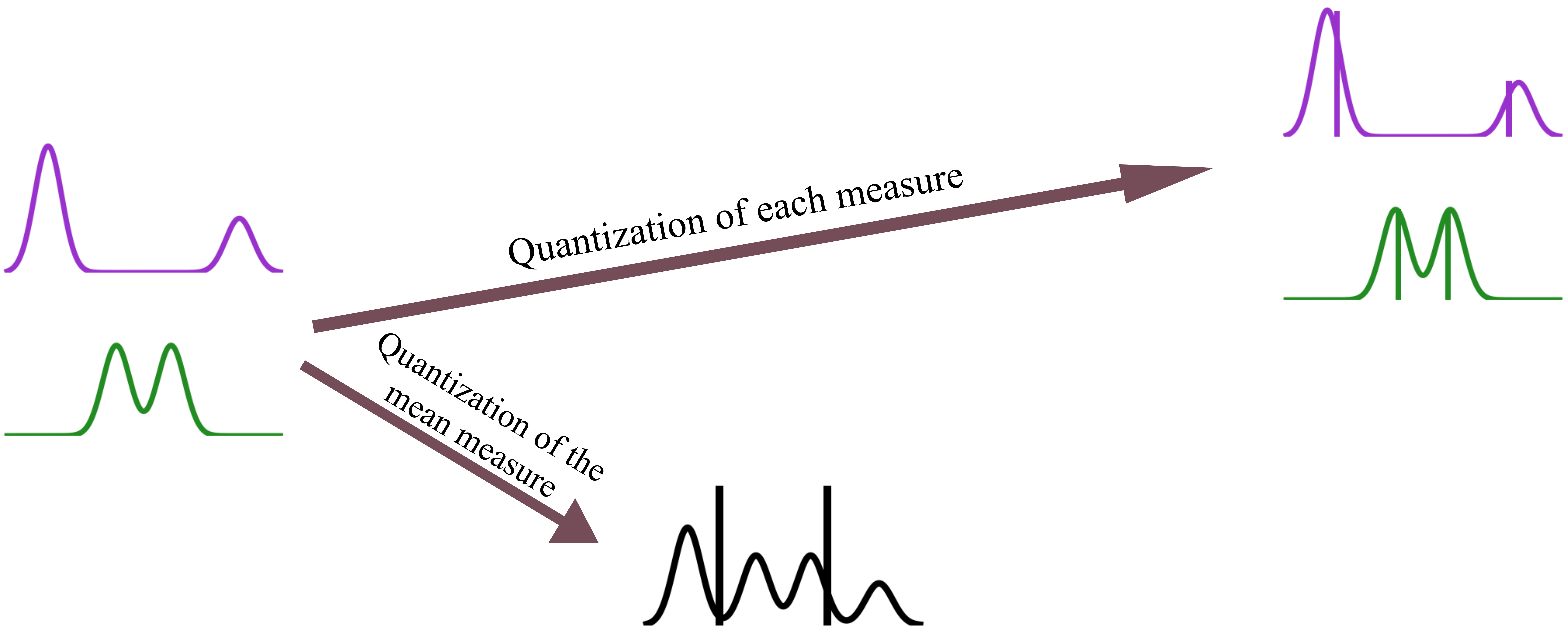
Quantization of each measure vs mean-measure quantization



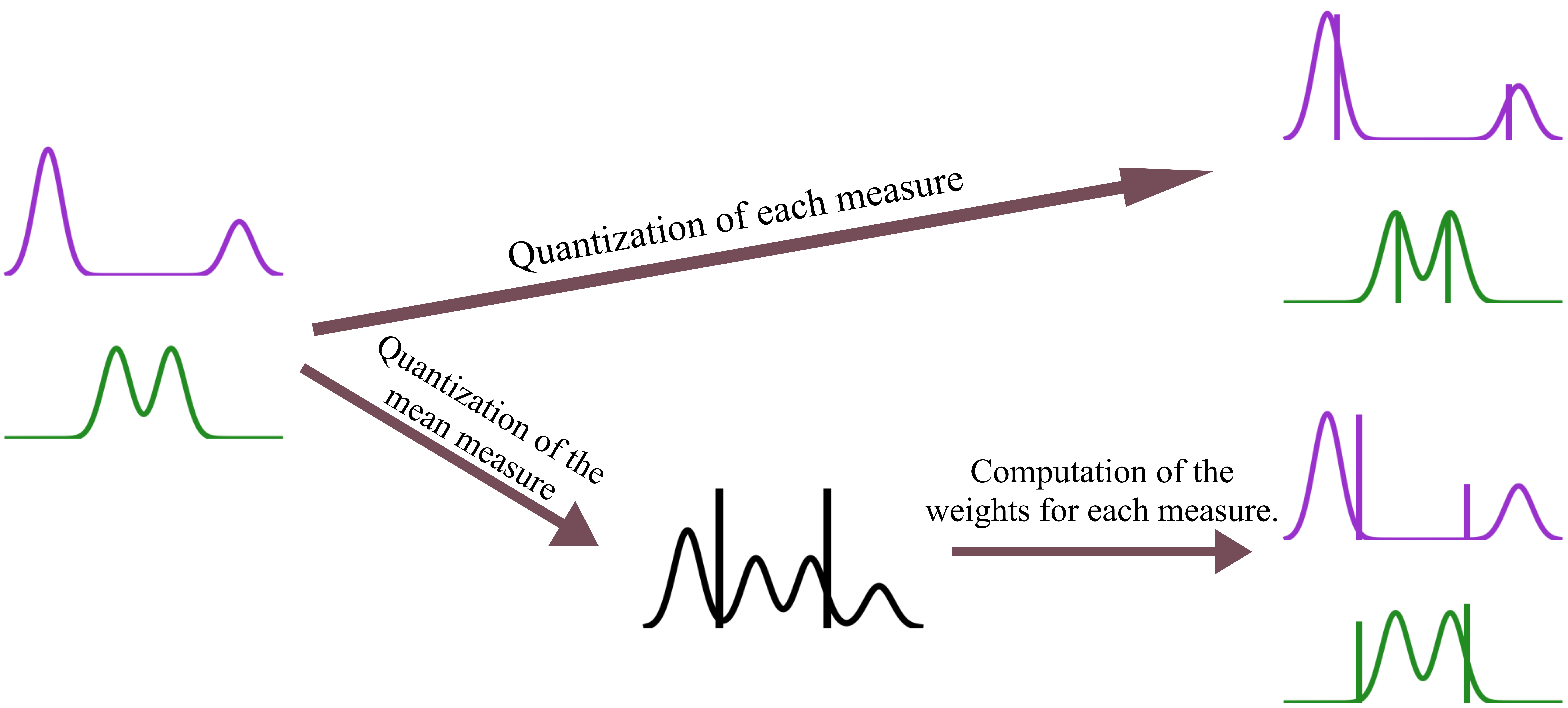
Quantization of each measure vs mean-measure quantization



Quantization of each measure vs mean-measure quantization



Quantization of each measure vs mean-measure quantization



Mean-measure quantization

Mean-measure quantization

Idea : solving

$$\min_{a \in (\Sigma_K)^N, X \in (\mathbb{R}^d)^K} \frac{1}{N} \sum_{i=1}^{N} W_2^2\Big(\sum_{k=1}^{K} a_k^{(i)} \delta_{X_k}, \mu^{(i)} \Big)$$

Mean-measure quantization

Idea : solving

$$\min_{a\in(\Sigma_K)^N, X\in(\mathbb{R}^d)^K} \frac{1}{N}\sum_{i=1}^{N} W_2^2\Big(\sum_{k=1}^{K} a_k^{(i)}\delta_{X_k}, \mu^{(i)}\Big)$$

$\Rightarrow$one common support, N weight vectors

Mean-measure quantization

Idea : solving

$$\min_{a\in(\Sigma_K)^N, X\in(\mathbb{R}^d)^K} \frac{1}{N}\sum_{i=1}^{N} W_2^2\Big(\sum_{k=1}^{K} a_k^{(i)}\delta_{X_k}, \mu^{(i)}\Big)$$

$\Rightarrow$one common support, N weight vectors

Proposition (Gachon et al., 2024). Let $\bar{\mu} = \frac{1}{N}\sum_{i=1}^{N}\mu^{(i)}$ be the mean measure. Then,

$$\min_{a\in(\Sigma_K)^N, X\in(\mathbb{R}^d)^K} \frac{1}{N}\sum_{i=1}^{N} W_2^2\Big(\sum_{k=1}^{K} a_k^{(i)}\delta_{X_k}, \mu^{(i)}\Big) = \min_{X\in(\mathbb{R}^d)^K} W_2^2\Big(\sum_{k=1}^{K} \bar{\mu}(V_{X_k})\delta_{X_k}, \bar{\mu}\Big)$$

Furthermore, minimizers a^* and X^* verify $\quad a_k^{(i)} = \mu^{(i)}(V_{X_k})$

Mean-measure quantization

Idea : solving

$$\min_{a\in(\Sigma_K)^N, X\in(\mathbb{R}^d)^K} \frac{1}{N}\sum_{i=1}^{N} W_2^2\Big(\sum_{k=1}^{K} a_k^{(i)}\delta_{X_k}, \mu^{(i)}\Big)$$

$\Rightarrow$one common support, N weight vectors

Proposition (Gachon et al., 2024). Let $\bar{\mu} = \frac{1}{N}\sum_{i=1}^{N}\mu^{(i)}$ be the mean measure. Then,

$$\min_{a\in(\Sigma_K)^N, X\in(\mathbb{R}^d)^K} \frac{1}{N}\sum_{i=1}^{N} W_2^2\Big(\sum_{k=1}^{K} a_k^{(i)}\delta_{X_k}, \mu^{(i)}\Big) = \min_{X\in(\mathbb{R}^d)^K} W_2^2\Big(\sum_{k=1}^{K} \bar{\mu}(V_{X_k})\delta_{X_k}, \bar{\mu}\Big)$$

Furthermore, minimizers a^* and X^* verify $a_k^{(i)} = \mu^{(i)}(V_{X_k})$

Method (Mean-measure quantization).

Compute the mean-measure $\bar{\mu} = \frac{1}{N}\sum_{i=1}^{N}\mu^{(i)}$

and solve the K-points quantization problem. Let X be the corresponding minimizer.

We define the quantized measures as

$$\nu^{(i)} = \sum_{k=1}^{K}\mu^{(i)}(V_{X_k})\delta_{X_k}$$

$\Rightarrow$ 1 support X, N weights $a^{(i)}$

Main result

Main result

Raw measures

$$(\mu^{(i)})_{i=1}^N$$

$$\downarrow$$

$$\mathbb{P}^N = \frac{1}{N} \sum_{i=1}^N \delta_{\mu^{(i)}}$$

Main result

Raw measures

$$(\mu^{(i)})_{i=1}^N$$

$$\mathbb{P}^N = \frac{1}{N}\sum_{i=1}^N \delta_{\mu^{(i)}}$$

Quantized measures

$$(\nu_K^{(i)})_{i=1}^N$$

$$\mathbb{P}_K^N = \frac{1}{N}\sum_{i=1}^N \delta_{\nu_K^{(i)}}$$

Main result

Raw measures

$$(\mu^{(i)})_{i=1}^N$$

$$\mathbb{P}^N = \frac{1}{N}\sum_{i=1}^N \delta_{\mu^{(i)}}$$

Quantized measures

$$(\nu_K^{(i)})_{i=1}^N$$

$$\mathbb{P}_K^N = \frac{1}{N}\sum_{i=1}^N \delta_{\nu_K^{(i)}}$$

$\Rightarrow$ need for a metric on $\mathscr{P}(\mathscr{P}(\mathscr{X}))$

Main result

Raw measures

$$(\mu^{(i)})_{i=1}^N$$

$$\mathbb{P}^N = \frac{1}{N}\sum_{i=1}^N \delta_{\mu^{(i)}}$$

Quantized measures

$$(\nu_K^{(i)})_{i=1}^N$$

$$\mathbb{P}_K^N = \frac{1}{N}\sum_{i=1}^N \delta_{\nu_K^{(i)}}$$

$\Rightarrow$ need for a metric on $\mathscr{P}(\mathscr{P}(\mathscr{X}))$

$$\mathscr{W}_2^2(\mathbb{P}, \mathbb{Q}) = \min_{\gamma \in \Gamma(\mathbb{P}, \mathbb{Q})} \int_{\mathscr{P}(\mathscr{X}) \times \mathscr{P}(\mathscr{X})} W_2^2(\mu, \nu) \mathrm{d}\gamma(\mu, \nu)$$

where $\Gamma(\mathbb{P}, \mathbb{Q})$ is the set of probability distributions on $\mathscr{P}(\mathscr{X}) \times \mathscr{P}(\mathscr{X})$ with respective marginals $\mathbb{P}$ and $\mathbb{Q}$.

Main result

Raw measures

$$(\mu^{(i)})_{i=1}^N$$

$$\mathbb{P}^N = \frac{1}{N}\sum_{i=1}^N \delta_{\mu^{(i)}}$$

Quantized measures

$$(\nu_K^{(i)})_{i=1}^N$$

$$\mathbb{P}_K^N = \frac{1}{N}\sum_{i=1}^N \delta_{\nu_K^{(i)}}$$

$\Rightarrow$ need for a metric on $\mathscr{P}(\mathscr{P}(\mathscr{X}))$

$$\mathscr{W}_2^2(\mathbb{P}, \mathbb{Q}) = \min_{\gamma \in \Gamma(\mathbb{P},\mathbb{Q})} \int_{\mathscr{P}(\mathscr{X})\times\mathscr{P}(\mathscr{X})} W_2^2(\mu, \nu) \mathrm{d}\gamma(\mu, \nu)$$

where $\Gamma(\mathbb{P}, \mathbb{Q})$ is the set of probability distributions on $\mathscr{P}(\mathscr{X}) \times \mathscr{P}(\mathscr{X})$ with respective marginals $\mathbb{P}$ and $\mathbb{Q}$.

Proposition (Gachon et al., 2025).

$$\mathscr{W}_2^2(\mathbb{P}_K^N, \mathbb{P}^N) = O(K^{-2/d}) \xrightarrow{K\to+\infty} 0$$

Convergence of the Gram matrices

Convergence of the Gram matrices

As mentioned, some algorithms entirely rely on inner-products between data points, usually through the diagonalization of the Gram matrix of the pairwise inner-products.

Convergence of the Gram matrices

As mentioned, some algorithms entirely rely on inner-products between data points, usually through the diagonalization of the Gram matrix of the pairwise inner-products.

Do we have convergence of the Gram matrices of the quantized embedded measures towards the Gram matrix of the raw embedded measures ?

Convergence of the Gram matrices

As mentioned, some algorithms entirely rely on inner-products between data points, usually through the diagonalization of the Gram matrix of the pairwise inner-products.

Do we have convergence of the Gram matrices of the quantized embedded measures towards the Gram matrix of the raw embedded measures ?

For a given embedding ϕ of probability measures into a Hilbert space $\mathscr{H}$ equipped with the inner-product $\langle \cdot , \cdot \rangle_{\mathscr{H}}$, we construct the following Gram matrices:

$$(G^{\phi}_{\mu})_{ij} = \langle \phi(\mu^{(i)}), \phi(\mu^{(j)}) \rangle_{\mathscr{H}} \qquad (G^{\phi}_{\nu_K})_{ij} = \langle \phi(\nu_K^{(i)}), \phi(\nu_K^{(j)}) \rangle_{\mathscr{H}}$$

Gram matrix of the embedded raw measures

Gram matrix of the embedded quantized measures

Convergence of the Gram matrices

As mentioned, some algorithms entirely rely on inner-products between data points, usually through the diagonalization of the Gram matrix of the pairwise inner-products.

Do we have convergence of the Gram matrices of the quantized embedded measures towards the Gram matrix of the raw embedded measures ?

For a given embedding ϕ of probability measures into a Hilbert space $\mathscr{H}$ equipped with the inner-product $\langle \cdot , \cdot \rangle_{\mathscr{H}}$, we construct the following Gram matrices:

$$(G^{\phi}_{\mu})_{ij} = \langle \phi(\mu^{(i)}), \phi(\mu^{(j)}) \rangle_{\mathscr{H}} \qquad (G^{\phi}_{\nu_K})_{ij} = \langle \phi(\nu_K^{(i)}), \phi(\nu_K^{(j)}) \rangle_{\mathscr{H}}$$

Gram matrix of the embedded raw measures

Gram matrix of the embedded quantized measures

Proposition (Gachon et al., 2025, Informal). With ϕ = LOT or ϕ = KME, then:

$$\|G^{\phi}_{\mu} - G^{\phi}_{\nu_K}\|_F^2 \xrightarrow{K \to \infty} 0$$

Numerical experiments

Numerical experiments

Flow cytometry datasets from two healthcare centers (Marburg and Dresden) and of different nature (peripheral blood, heathy bone marrow and leukemic bone marrow)

Numerical experiments

Flow cytometry datasets from two healthcare centers (Marburg and Dresden) and of different nature (peripheral blood, heathy bone marrow and leukemic bone marrow)

⇒ we can perform supervised clustering by predicting either the healthcare center or the nature of the sample

Numerical experiments

Flow cytometry datasets from two healthcare centers (Marburg and Dresden) and of different nature (peripheral blood, heathy bone marrow and leukemic bone marrow)

⇒ we can perform supervised clustering by predicting either the healthcare center or the nature of the sample

$$N = 108$$
$$10^5 \leq m \leq 10^6$$
$$d = 10$$

Numerical experiments

Flow cytometry datasets from two healthcare centers (Marburg and Dresden) and of different nature (peripheral blood, heathy bone marrow and leukemic bone marrow)

⇒ we can perform supervised clustering by predicting either the healthcare center or the nature of the sample

$$N = 108$$
$$10^5 \leq m \leq 10^6$$
$$d = 10$$

1. Compute the quantized measures
2. Embed the measures with chosen embedding
3. Perform a PCA in the corresponding space

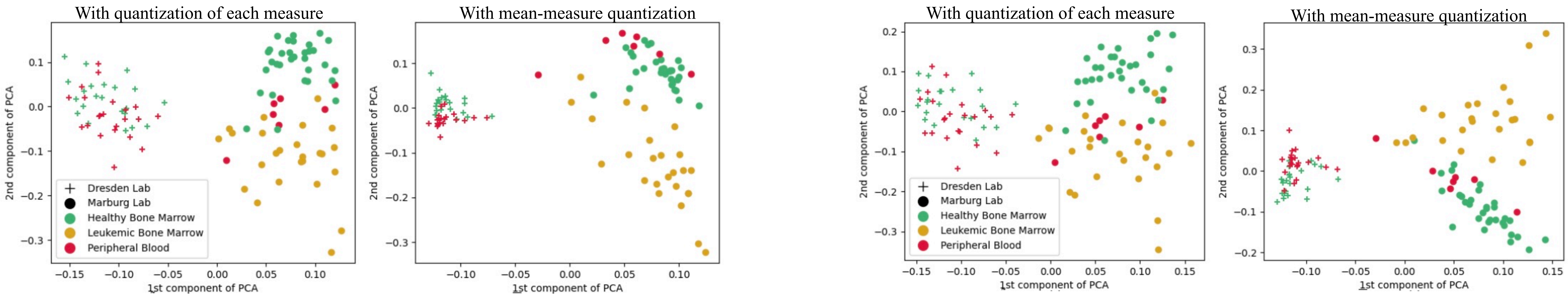
Numerical experiments

Flow cytometry datasets from two healthcare centers (Marburg and Dresden) and of different nature (peripheral blood, heathy bone marrow and leukemic bone marrow)

$\Rightarrow$ we can perform supervised clustering by predicting either the healthcare center or the nature of the sample

$$N = 108$$
$$10^5 \leq m \leq 10^6$$
$$d = 10$$

1. Compute the quantized measures
2. Embed the measures with chosen embedding
3. Perform a PCA in the corresponding space



Projection of the flow cytometry datasets on the first components of PCA after embeddings LOT (left) and KME (right) on the quantized measures with $K = 16$.

Numerical experiments

	$\overline{K}$-LOT/$\tilde{K}$-LOT			$\overline{K}$-KME/$\tilde{K}$-KME			KME with RFF			
K	Accuracy (lab)	Accuracy (type)	Time (s)	Accuracy (lab)	Accuracy (type)	Time (s)	s	Accuracy (lab)	Accuracy (type)	Time (s)
16	100/100	85/81	23/103	100/100	83/69	15/96	16	73	44	4524
32	100/100	94/81	25/166	100/100	83/69	34/174	32	75	44	4701
64	100/100	94/81	30/281	100/100	85/69	105/358	64	83	52	5035
128	100/100	88/81	32/555	100/100	77/71	387/909	128	92	44	5676

Classification accuracies and execution times for LDA after 10-component PCA. $\tilde{K}$ stands for the method of quantization of each measure and $\bar{K}$ designs the method with mean-measure quantization. RFF (Random Fourier Features) is an approximation of KME.

Conclusion

Conclusion

Adapting classical machine learning algorithms to handle probability measures not straightforward

Conclusion

Adapting classical machine learning algorithms to handle probability measures not straightforward

Popular approach: Hilbert space embeddings (LOT, KME…) but come with computational burden when dealing with large support size measures

Conclusion

Adapting classical machine learning algorithms to handle probability measures not straightforward

Popular approach: Hilbert space embeddings (LOT, KME…) but come with computational burden when dealing with large support size measures

Our approach: Reduce the support size with two methods based on quantization

Conclusion

Adapting classical machine learning algorithms to handle probability measures not straightforward

Popular approach: Hilbert space embeddings (LOT, KME…) but come with computational burden when dealing with large support size measures

Our approach: Reduce the support size with two methods based on quantization
Results show that both methods approximate the input probability measures at the asymptotic rate of $O(K^{-2/d})$

Conclusion

Adapting classical machine learning algorithms to handle probability measures not straightforward

Popular approach: Hilbert space embeddings (LOT, KME…) but come with computational burden when dealing with large support size measures

Our approach: Reduce the support size with two methods based on quantization

Results show that both methods approximate the input probability measures at the asymptotic rate of $O(K^{-2/d})$

the good performance of machine learning methods on the embedded quantized measures

Conclusion

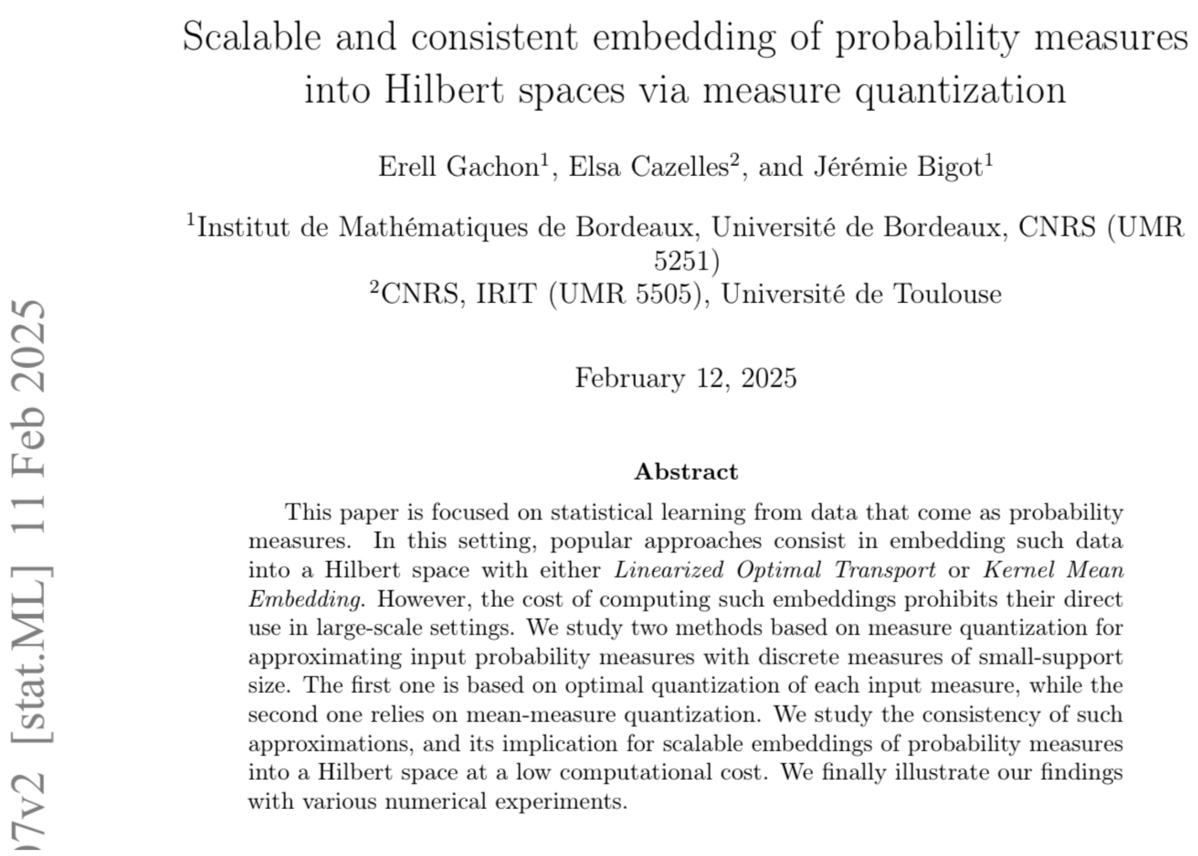
Adapting classical machine learning algorithms to handle probability measures not straightforward

Popular approach: Hilbert space embeddings (LOT, KME…) but come with computational burden when dealing with large support size measures

Our approach: Reduce the support size with two methods based on quantization

Results show that both methods approximate the input probability measures at the asymptotic rate of $O(K^{-2/d})$

the good performance of machine learning methods on the embedded quantized measures



Scalable and consistent embedding of probability measures into Hilbert spaces via measure quantization

Erell Gachon[1], Elsa Cazelles[2], and Jérémie Bigot[1]

[1]Institut de Mathématiques de Bordeaux, Université de Bordeaux, CNRS (UMR 5251)

[2]CNRS, IRIT (UMR 5505), Université de Toulouse

February 12, 2025

)7v2 [stat.ML] 11 Feb 2025

Abstract

This paper is focused on statistical learning from data that come as probability measures. In this setting, popular approaches consist in embedding such data into a Hilbert space with either *Linearized Optimal Transport* or *Kernel Mean Embedding*. However, the cost of computing such embeddings prohibits their direct use in large-scale settings. We study two methods based on measure quantization for approximating input probability measures with discrete measures of small-support size. The first one is based on optimal quantization of each input measure, while the second one relies on mean-measure quantization. We study the consistency of such approximations, and its implication for scalable embeddings of probability measures into a Hilbert space at a low computational cost. We finally illustrate our findings with various numerical experiments.

Thank you for your attention!