

Représentation de courbes en dimension finie

Benjamin Auder

CEA - UPMC

17 mars 2010

Thèse depuis 02/2008

Directeur de thèse : Gérard Biau (UPMC)

Encadrant CEA : Bertrand Iooss (EDF)

Contexte industriel

Cadre : durée de vie des cuves.

→ Diverses séquences d'accidents envisagées.

But : majorer la probabilité de chaque accident répertorié.

Contexte industriel

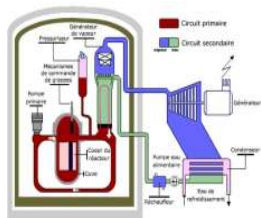
Cadre : durée de vie des cuves.

→ Diverses séquences d'accidents envisagées.

But : majorer la probabilité de chaque accident répertorié.

Méthodologie

Modélisation



Contexte industriel

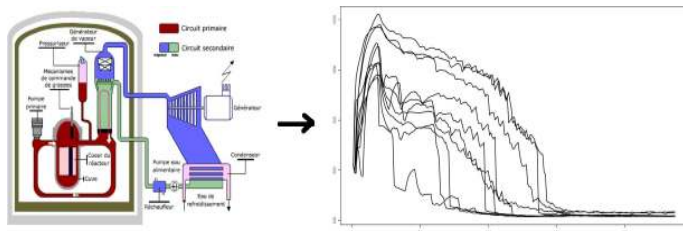
Cadre : durée de vie des cuves.

→ Diverses séquences d'accidents envisagées.

But : majorer la probabilité de chaque accident répertorié.

Méthodologie

Modélisation → Simulation



Contexte industriel

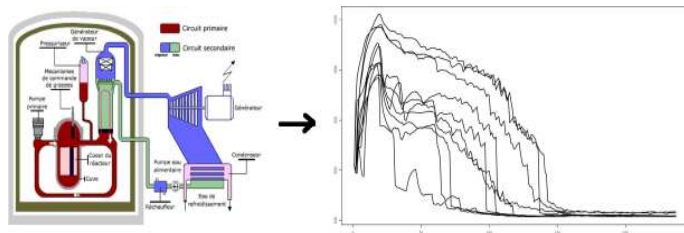
Cadre : durée de vie des cuves.

→ Diverses séquences d'accidents envisagées.

But : majorer la probabilité de chaque accident répertorié.

Méthodologie

Modélisation → Simulation → Calculs.



→ Analyse de sensibilité, propagation d'incertitudes ..etc.

Contexte industriel

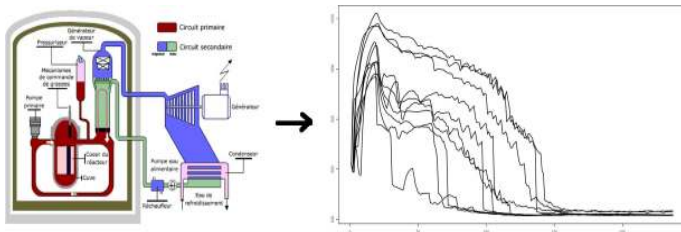
Cadre : durée de vie des cuves.

→ Diverses séquences d'accidents envisagées.

But : majorer la probabilité de chaque accident répertorié.

Méthodologie

Modélisation → **Simulation** → Calculs.



→ Analyse de sensibilité, propagation d'incertitudes ..etc.

Améliorer la phase simulation, pour effectuer des calculs plus fiables

Au CEA - DER/SESI/LSMR ...

Code thermo-hydraulique CATHARE **coûteux** en temps

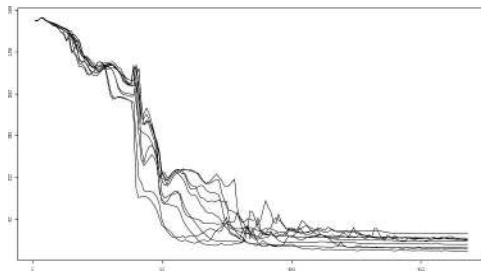


FIG.: Transitoires de température.

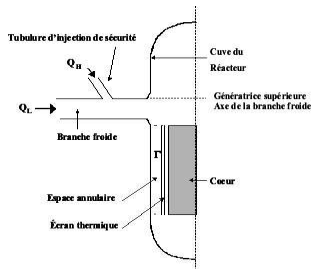


FIG.: Zone modélisée

Au CEA - DER/SESI/LSMR ...

Code thermo-hydraulique CATHARE coûteux en temps

- "boîte noire" ;

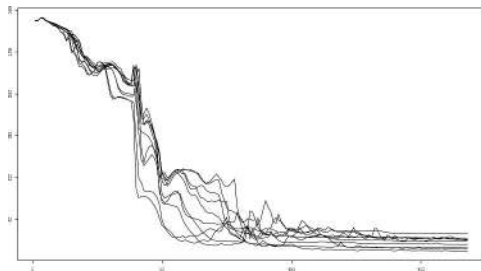


FIG.: Transitoires de température.

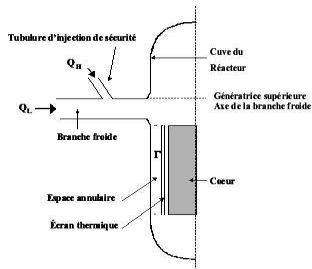


FIG.: Zone modélisée

Au CEA - DER/SESI/LSMR ...

Code thermo-hydraulique CATHARE coûteux en temps

- "boîte noire" ;
- paramètres d'entrée incertains.

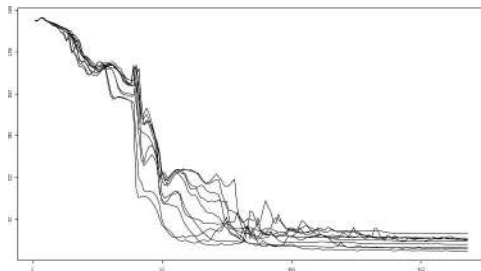


FIG.: Transitoires de température.

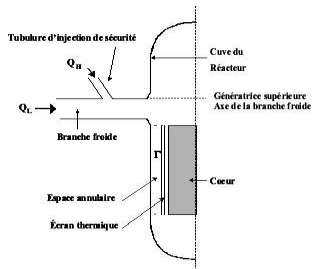


FIG.: Zone modélisée

Au CEA - DER/SESI/LSMR ...

Code thermo-hydraulique CATHARE coûteux en temps

- "boîte noire" ;
- paramètres d'entrée incertains.

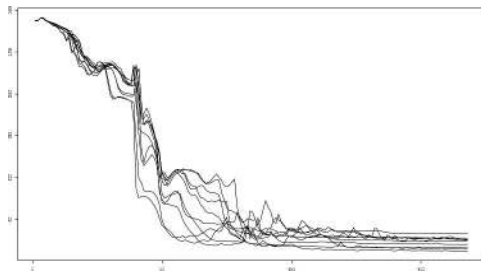


FIG.: Transitoires de température.

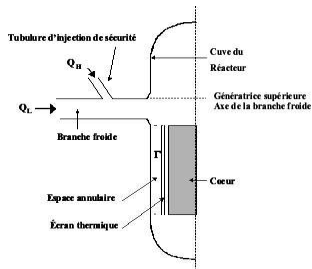


FIG.: Zone modélisée

"Accélérer" l'exécution du code CATHARE

Solution ?

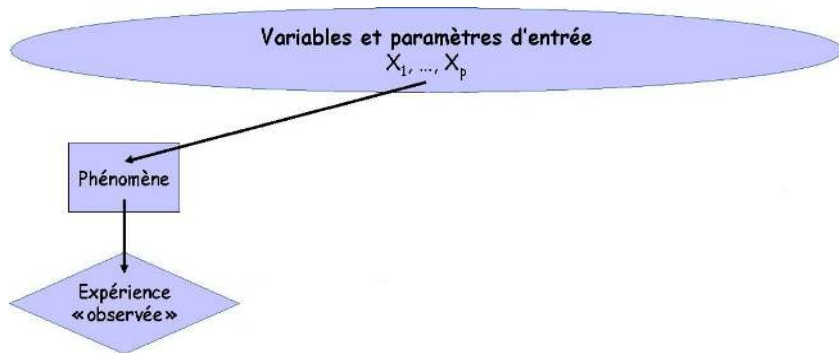


FIG.: Point de départ : phénomène physique.

marges d'incertitudes sur le phénomène

Solution ?

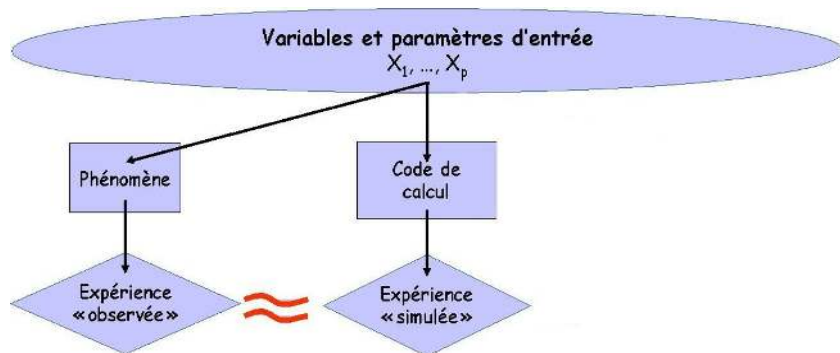


FIG.: Code de calcul = simulation du phénomène.

nécessitent marges d'incertitudes sur le phénomène
beaucoup de résultats de code

Solution : métamodèle

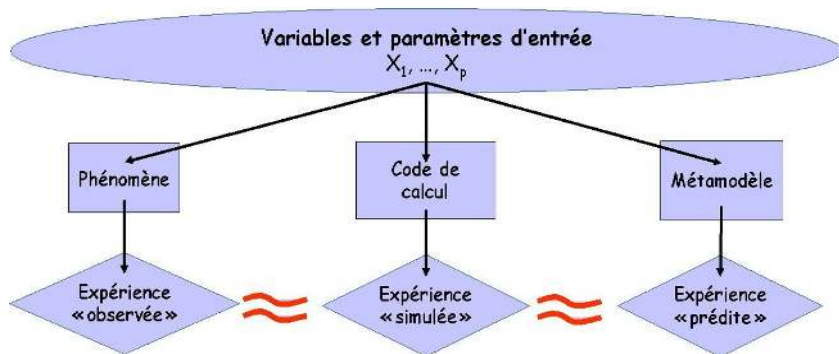


FIG.: Métamodèle = accélération des simulations.

nécessitent marges d'incertitudes sur le phénomène
obtenus avec beaucoup de résultats de code
un métamodèle (= modèle du code).

Solution : métamodèle

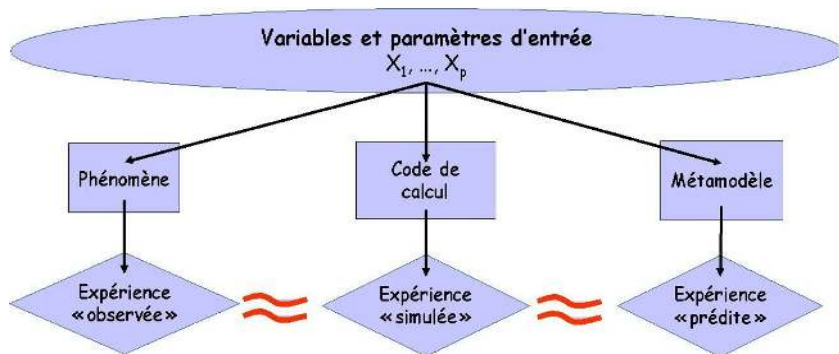


FIG.: Métamodèle = accélération des simulations.

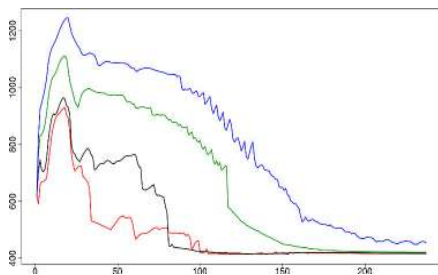
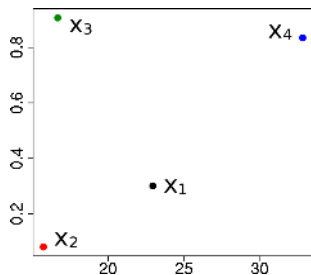
nécessitent marges d'incertitudes sur le phénomène
obtenus avec beaucoup de résultats de code
un métamodèle (= modèle du code).

Construire un modèle du code CATHARE

Résultats attendus

n couples (x_i, y_i) connus :

- Entrées $x_i \in \mathbb{R}^p =$ état initial du système physique ;
- Sorties $y_i \in \mathcal{C}([a, b], \mathbb{R}) =$ évolution des paramètres.



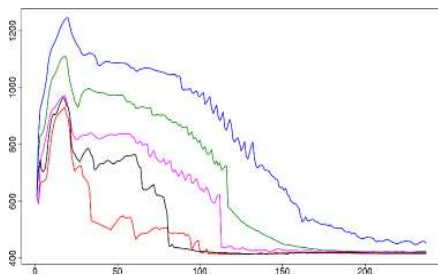
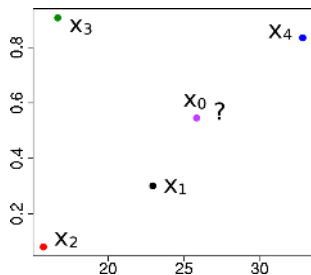
Résultats attendus

n couples (x_i, y_i) connus :

- Entrées $x_i \in \mathbb{R}^p =$ état initial du système physique ;
- Sorties $y_i \in \mathcal{C}([a, b], \mathbb{R}) =$ évolution des paramètres.

Objectif = **prédiction** de données fonctionnelles :

$$y^{\text{new}} \simeq \varphi(x^{\text{new}}).$$



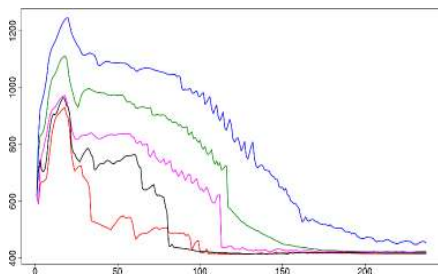
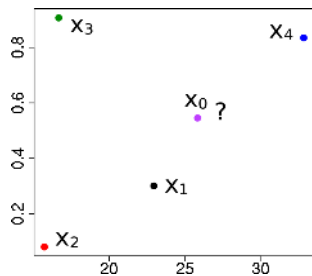
Résultats attendus

n couples (x_i, y_i) connus :

- Entrées $x_i \in \mathbb{R}^p =$ état initial du système physique ;
- Sorties $y_i \in \mathcal{C}([a, b], \mathbb{R}) =$ évolution des paramètres.

Objectif = **prédiction** de données fonctionnelles :

$$y^{\text{new}} \simeq \varphi(x^{\text{new}}).$$



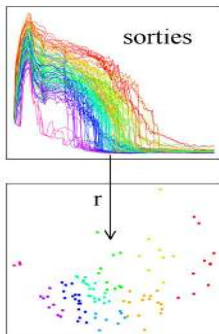
Apprentissage statistique "régression" $\mathbb{R}^p \rightarrow \mathcal{C}([a, b], \mathbb{R})$

Comment se ramener au cas "simple" $y_i \in \mathbb{R}^d$?

Comment se ramener au cas "simple" $y_i \in \mathbb{R}^d$?

① réduction de la dimension :

$r : \mathcal{C}([a, b], \mathbb{R}) \rightarrow \mathbb{R}^d$ (représentation) ;



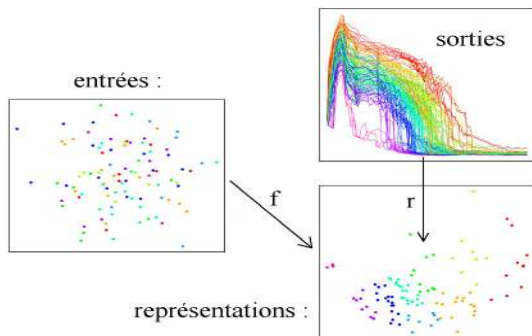
Comment se ramener au cas "simple" $y_i \in \mathbb{R}^d$?

① réduction de la dimension :

$r : \mathcal{C}([a, b], \mathbb{R}) \rightarrow \mathbb{R}^d$ (représentation) ;

② apprentissage statistique classique :

$f : \mathbb{R}^p$ (entrées) $\rightarrow \mathbb{R}^d$ (sorties réduites) ;



Comment se ramener au cas "simple" $y_i \in \mathbb{R}^d$?

① réduction de la dimension :

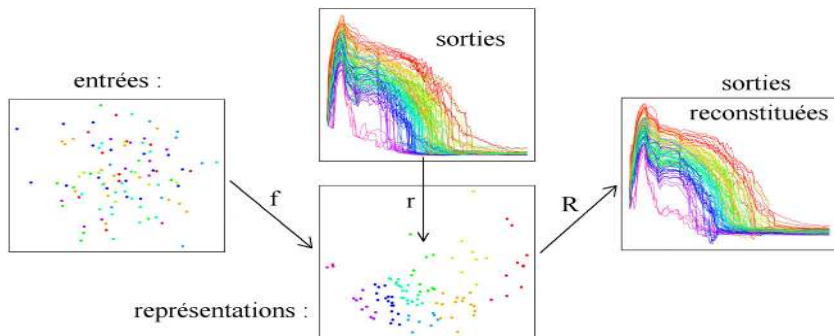
$r : \mathcal{C}([a, b], \mathbb{R}) \rightarrow \mathbb{R}^d$ (représentation) ;

② apprentissage statistique classique :

$f : \mathbb{R}^p$ (entrées) $\rightarrow \mathbb{R}^d$ (sorties réduites) ;

③ paramétrage de la variété :

$R : \mathbb{R}^d \rightarrow \mathcal{C}([a, b], \mathbb{R})$ (reconstruction).



1 Exemple introductif

2 Dimension réduite "pas à pas"

- Local PCA Manifold Learning (Zhan et al., 2008)
- Riemannian Manifold Learning (T. Lin & H. Zha, 2006)
- "Local Riemannian Manifold Learning"

3 Applications

1 Exemple introductif

2 Dimension réduite "pas à pas"

- Local PCA Manifold Learning (Zhan et al., 2008)
- Riemannian Manifold Learning (T. Lin & H. Zha, 2006)
- "Local Riemannian Manifold Learning"

3 Applications

Données brutes

	[.1]	[.2]	[.3]	[.4]	[.5]	[.6]
[1,]	0.5891564	1.577159	-0.675	2.183015	1.968054	1.1105867
[2,]	0.7788008	1.664494	-0.258	2.635077	1.280696	2.8141909
[3,]	0.6537698	1.197966	-0.425	2.486964	1.510319	1.8996824
[4,]	0.5627049	1.391838	-0.575	2.314497	1.722600	1.3368171
[5,]	0.6872893	1.158993	-0.375	2.535796	1.42348	2.8258229
[6,]	0.7595721	1.678558	-0.275	2.618038	1.311992	2.8167830
[7,]	0.6376782	1.724460	-0.458	2.460763	1.544910	1.7151039
[8,]	0.5229458	1.525771	-0.658	2.216842	1.931594	1.1825144
[9,]	0.6865307	1.204925	-0.500	2.405079	1.615146	1.5472631
[10,]	0.9512294	1.862503	-0.858	2.714887	1.851249	6.6648135
[11,]	0.5769498	1.353238	-0.558	2.345561	1.886554	1.4823176
[12,]	0.5352614	1.477904	-0.625	2.250074	1.795166	1.2173085
[13,]	0.8887080	1.822755	-0.158	2.687929	1.161182	3.3819186
[14,]	0.9277435	1.885641	-0.875	2.710651	1.377808	5.7996872
[15,]	0.4168620	2.158338	-0.875	1.898372	2.154467	6.7707595
[16,]	0.4865697	2.247908	-0.988	1.861923	2.188742	6.7352086
[17,]	0.8394570	1.631999	-0.175	2.677088	1.190181	3.8126013
[18,]	0.7846881	1.138319	-0.358	2.558376	1.409025	2.1504059
[19,]	0.8187308	1.648811	-0.288	2.664634	1.219779	3.3834914
[20,]	0.6783200	1.173511	-0.488	2.511954	1.476122	1.9128022
[21,]	0.4965853	1.632316	-0.708	2.148655	1.904497	1.9612462
[22,]	0.9848374	1.818950	-0.188	2.704736	1.104987	4.9916472
[23,]	0.7985162	1.851928	-0.225	2.650621	1.249954	3.8406655
[24,]	0.4493290	1.896481	-0.888	2.007132	2.349009	6.8852389
[25,]	0.7225274	1.111405	-0.325	2.579642	1.376176	2.2888031
[26,]	0.9753099	1.889625	-0.825	2.717433	1.825312	7.9996666
[27,]	0.3670794	2.710282	-1.000	1.716526	2.319777	6.6933518
[28,]	0.5488116	1.433329	-0.688	2.282647	1.758819	1.2752877
[29,]	0.7488182	1.894174	-0.388	2.599545	1.343825	2.4429793
[30,]	0.4274149	2.659576	-0.858	1.934766	2.119712	6.8075374
[31,]	0.3867410	2.465760	-0.958	1.789047	2.255599	6.6673863
[32,]	0.4687038	1.823258	-0.775	2.043064	2.313181	6.9264261
[33,]	0.3771924	2.587326	-0.975	1.752719	2.288054	6.5349357
[34,]	0.5915554	1.317354	-0.525	2.375776	1.650722	1.4722768
[35,]	0.4723666	1.755855	-0.758	2.078588	1.977115	6.9693713
[36,]	0.8824969	1.815748	-0.125	2.697155	1.132780	4.4315410
[37,]	0.4382350	1.975112	-0.825	1.971031	2.3884539	6.8456556
[38,]	0.6218951	1.253106	-0.475	2.433468	1.579863	1.5279430
[39,]	0.4043246	1.691516	-0.725	2.113826	1.940860	1.9142469
[40,]	0.3965314	2.352844	-0.925	1.825465	2.222473	6.7807821

→ rien de très clair ...

a priori 6 dimensions

Données brutes

	[.1]	[.2]	[.3]	[.4]	[.5]	[.6]
[1.]	0.5891564	1.577159	-0.675	2.183015	1.968054	1.1105867
[2.]	0.7788008	1.664494	-0.258	2.635077	1.280696	2.8141909
[3.]	0.6537698	1.197966	-0.425	2.486964	1.510319	1.8996824
[4.]	0.5627049	1.391838	-0.575	2.314497	1.722600	1.3368171
[5.]	0.6872893	1.158993	-0.375	2.535796	1.42348	2.8258229
[6.]	0.7595721	1.678558	-0.275	2.618038	1.311992	2.8167830
[7.]	0.6376782	1.724460	-0.458	2.460763	1.544910	1.7151039
[8.]	0.5229458	1.525771	-0.658	2.216842	1.931594	1.1825144
[9.]	0.6865307	1.204925	-0.500	2.405079	1.615146	1.5472631
[10.]	0.9512294	1.862503	-0.858	2.714887	1.851249	6.6648135
[11.]	0.5769498	1.353238	-0.558	2.345561	1.886554	1.4823176
[12.]	0.5352614	1.477904	-0.625	2.250074	1.795166	1.2173085
[13.]	0.8887080	1.822755	-0.158	2.687929	1.161182	3.3819186
[14.]	0.9277435	1.885641	-0.875	2.710651	1.377808	5.7996872
[15.]	0.4168620	2.158338	-0.875	1.898372	2.154467	6.7707595
[16.]	0.4865697	2.247908	-0.988	1.861923	2.188742	6.7352086
[17.]	0.8394570	1.631999	-0.175	2.677088	1.190181	3.8126043
[18.]	0.7846881	1.138319	-0.358	2.558376	1.409025	2.1504059
[19.]	0.8187308	1.648811	-0.288	2.664634	1.219779	3.3834914
[20.]	0.6783200	1.173511	-0.488	2.511954	1.476122	1.9128022
[21.]	0.4965853	1.632316	-0.708	2.148655	1.904497	1.9612462
[22.]	0.9848374	1.818950	-0.188	2.704736	1.104987	4.9916472
[23.]	0.7985162	1.851928	-0.225	2.650621	1.249954	3.8406655
[24.]	0.4493290	1.896481	-0.888	2.007132	2.849009	6.8852389
[25.]	0.7225274	1.111405	-0.325	2.579642	1.376176	2.2886031
[26.]	0.9753099	1.889625	-0.825	2.717433	1.825312	7.3996666
[27.]	0.3678794	2.710282	-1.000	1.716526	2.319777	6.6933518
[28.]	0.5488116	1.433329	-0.688	2.282647	1.758819	1.2752877
[29.]	0.7488182	1.894174	-0.388	2.599545	1.343825	2.4429793
[30.]	0.4274149	2.659576	-0.858	1.934766	2.119712	6.8075374
[31.]	0.3867410	2.465760	-0.958	1.789047	2.255599	6.6673863
[32.]	0.4687038	1.823258	-0.775	2.043064	2.813181	6.9264261
[33.]	0.3771924	2.587326	-0.975	1.752719	2.288054	6.5349357
[34.]	0.5915554	1.317354	-0.525	2.375776	1.650722	1.4722768
[35.]	0.4723666	1.755855	-0.758	2.078588	1.977115	6.9693713
[36.]	0.8824969	1.815748	-0.125	2.697155	1.132780	4.4315410
[37.]	0.4382350	1.975112	-0.825	1.971031	2.3884539	6.8456556
[38.]	0.6218951	1.253106	-0.475	2.433468	1.579863	1.5279430
[39.]	0.4043246	1.691516	-0.725	2.113826	1.940860	1.9142469
[40.]	0.3965314	2.352844	-0.925	1.825465	2.222473	6.7807821

→ rien de très clair ...

a priori 6 dimensions

représentation en
dimension réduite ?

1^{ere} approche : ACP



FIG.: Décomposition ACP à une composante : insuffisant

1^{ere} approche : ACP

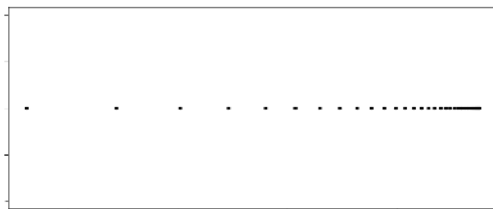


FIG.: Décomposition ACP à une composante : insuffisant



FIG.: Décomposition ACP à deux composantes : "OK"

1^{ere} approche : ACP



FIG.: Décomposition ACP à une composante : insuffisant



FIG.: Décomposition ACP à deux composantes : "OK"

"Donc" deux dimensions ?

Révélation

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]
[1,]	0.3678794	2.718282	-1.000	1.716526	2.319777	0.6033518
[2,]	0.3771924	2.587326	-0.975	1.752719	2.288054	0.6349357
[3,]	0.3867410	2.465768	-0.950	1.789947	2.255599	0.6673863
[4,]	0.3965314	2.352844	-0.925	1.825465	2.222473	0.7007821
[5,]	0.4065697	2.247988	-0.900	1.861323	2.188742	0.7352086
[6,]	0.4168520	2.150338	-0.875	1.898372	2.154467	0.7707595
[7,]	0.4274149	2.059576	-0.850	1.934760	2.119712	0.8075374
[8,]	0.4382350	1.975112	-0.825	1.971331	2.084539	0.8456556
[9,]	0.4493290	1.896481	-0.800	2.007132	2.049889	0.8852389
[10,]	0.4607938	1.823258	-0.775	2.043304	2.013181	0.9264261
[11,]	0.4723666	1.755655	-0.750	2.078588	1.977115	0.9693713
[12,]	0.4843246	1.691516	-0.725	2.113826	1.940868	1.0142469
[13,]	0.4965853	1.632316	-0.700	2.148655	1.904497	1.0612462
[14,]	0.5091564	1.577159	-0.675	2.183315	1.868654	1.1105867
[15,]	0.5220458	1.525771	-0.650	2.216842	1.831594	1.1625144
[16,]	0.5352614	1.477984	-0.625	2.250374	1.795166	1.2173085
[17,]	0.54880116	1.433329	-0.600	2.282647	1.758819	1.2752877
[18,]	0.5627949	1.391838	-0.575	2.314497	1.722688	1.3368171
[19,]	0.5769498	1.353238	-0.550	2.345561	1.686554	1.4023176
[20,]	0.5915554	1.317354	-0.525	2.375776	1.650722	1.4722768
[21,]	0.6065307	1.284825	-0.500	2.405379	1.615146	1.5472631
[22,]	0.6218951	1.253186	-0.475	2.433498	1.579863	1.6279438
[23,]	0.6376282	1.224468	-0.450	2.460703	1.544918	1.7151039
[24,]	0.6537598	1.197966	-0.425	2.486904	1.510319	1.8096824
[25,]	0.6702290	1.173511	-0.400	2.511954	1.476122	1.9128022
[26,]	0.6872893	1.150993	-0.375	2.535796	1.442348	2.0258229
[27,]	0.7048981	1.130319	-0.350	2.558376	1.408925	2.1504059
[28,]	0.7229274	1.111485	-0.325	2.579542	1.376176	2.2886031
[29,]	0.7414812	1.094174	-0.300	2.599545	1.343825	2.4429793
[30,]	0.7595721	1.078558	-0.275	2.618338	1.311992	2.6167836
[31,]	0.7782998	1.064494	-0.250	2.635977	1.280696	2.8141969
[32,]	0.7985162	1.051928	-0.225	2.652621	1.249954	3.0406655
[33,]	0.8187398	1.040811	-0.200	2.668534	1.219779	3.3034914
[34,]	0.8394570	1.031099	-0.175	2.677980	1.190184	3.6126043
[35,]	0.8607980	1.022755	-0.150	2.687929	1.161182	3.9819186
[36,]	0.8824969	1.015748	-0.125	2.697155	1.132788	4.4315418
[37,]	0.9048374	1.010650	-0.100	2.704736	1.104987	4.9916472
[38,]	0.9277435	1.006641	-0.075	2.711851	1.077888	5.7096872
[39,]	0.9512294	1.002503	-0.050	2.718487	1.051249	6.6648135
[40,]	0.9753999	1.000625	-0.025	2.724733	1.025312	7.9996666

réordonnancement

⇒ une **structure** émerge

Révélation

	[,1]	[,2]	[,3]	[,4]	[,6]	
[1,]	0.3678794	2.718282	-1.000	1.716526	2.319777	0.6033518
[2,]	0.3771924	2.587326	-0.975	1.752719	2.288054	0.6349357
[3,]	0.3867410	2.465768	-0.950	1.789947	2.255599	0.6673863
[4,]	0.3965314	2.352844	-0.925	1.825465	2.222473	0.7007821
[5,]	0.4065697	2.247988	-0.900	1.861323	2.188742	0.7352086
[6,]	0.4168620	2.150338	-0.875	1.898372	2.154467	0.7707595
[7,]	0.4274149	2.059576	-0.850	1.934760	2.119712	0.8075374
[8,]	0.4382350	1.975112	-0.825	1.971931	2.084539	0.8456556
[9,]	0.4493290	1.896481	-0.800	2.007132	2.049889	0.8852389
[10,]	0.4607938	1.823258	-0.775	2.043304	2.013181	0.9264261
[11,]	0.4723666	1.755655	-0.750	2.078588	1.971115	0.9693713
[12,]	0.4843246	1.691516	-0.725	2.113826	1.940868	1.0142469
[13,]	0.4965853	1.632316	-0.700	2.148855	1.904497	1.0612462
[14,]	0.5091564	1.577159	-0.675	2.183315	1.868654	1.1105867
[15,]	0.5220458	1.525771	-0.650	2.216842	1.831594	1.1625144
[16,]	0.5352614	1.477984	-0.625	2.250874	1.795166	1.2173085
[17,]	0.54880116	1.433329	-0.600	2.282647	1.758019	1.2752877
[18,]	0.5627949	1.391838	-0.575	2.314497	1.722688	1.3368171
[19,]	0.5769498	1.353238	-0.550	2.345561	1.688654	1.4023176
[20,]	0.5915554	1.317354	-0.525	2.375776	1.656722	1.4722768
[21,]	0.6065397	1.284625	-0.500	2.405979	1.615146	1.5477631
[22,]	0.6218951	1.253186	-0.475	2.433498	1.570863	1.6279428
[23,]	0.6376282	1.224468	-0.450	2.460703	1.544918	1.7151039
[24,]	0.6537698	1.197966	-0.425	2.486904	1.516319	1.8096824
[25,]	0.6703290	1.173511	-0.400	2.511954	1.476122	1.9128022
[26,]	0.6872893	1.150993	-0.375	2.535796	1.442348	2.0258229
[27,]	0.7046881	1.130319	-0.350	2.558376	1.408925	2.1504059
[28,]	0.7225274	1.111485	-0.325	2.579642	1.376176	2.2886031
[29,]	0.7408182	1.094174	-0.300	2.599545	1.343825	2.4429793
[30,]	0.75955721	1.078558	-0.275	2.618938	1.311992	2.6167836
[31,]	0.7788098	1.064494	-0.250	2.635977	1.280696	2.8141969
[32,]	0.7985162	1.051928	-0.225	2.650621	1.249954	3.0406655
[33,]	0.8187398	1.040811	-0.200	2.664534	1.219779	3.3034914
[34,]	0.8394570	1.031099	-0.175	2.677980	1.190184	3.6126043
[35,]	0.8607980	1.022755	-0.150	2.687929	1.161182	3.9819186
[36,]	0.8824969	1.015748	-0.125	2.697155	1.132788	4.4315416
[37,]	0.9048374	1.010650	-0.100	2.704736	1.104987	4.9916472
[38,]	0.9277435	1.006564	-0.075	2.710651	1.077888	5.7096872
[39,]	0.9512294	1.002503	-0.050	2.714987	1.051249	6.6648135
[40,]	0.9753999	1.000625	-0.025	2.717433	1.025312	7.9996666

réordonnancement

⇒ une **structure** émerge

fonction génératrice :

$$t \mapsto (e^{-t}, e^{t^2}, -t, e^{\cos t}, e^{\sin t}, \frac{1}{\tan t + 0.1}),$$

aux points

$$t = 1/40, 2/40, \dots, 1.$$

En une dimension



FIG.: Réduction de dimension non linéaire : une composante suffit.

En une dimension

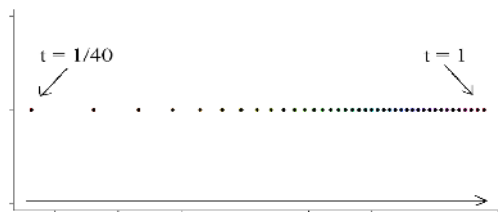


FIG.: Réduction de dimension non linéaire : une composante suffit.

Hypothèse

Les courbes en sortie ont une structure de **variété**

...éventuellement non linéaire

⇒ *décomposition sur une base = sous-optimal.*

1 Exemple introductif

2 Dimension réduite "pas à pas"

- Local PCA Manifold Learning (Zhan et al., 2008)
- Riemannian Manifold Learning (T. Lin & H. Zha, 2006)
- "Local Riemannian Manifold Learning"

3 Applications

Vers une représentation "optimale"

= Transformer l'ensemble des sorties du code $\mathcal{Y} \subset \mathcal{C}([a, b], \mathbb{R})$:

$$r : \mathcal{C}([a, b], \mathbb{R}) \rightarrow \mathbb{R}^d, \quad \text{avec } d \text{ le plus petit possible}$$

$$y_i \mapsto z_i$$

Vers une représentation "optimale"

= Transformer l'ensemble des sorties du code $\mathcal{Y} \subset \mathcal{C}([a, b], \mathbb{R})$:

$$r : \mathcal{C}([a, b], \mathbb{R}) \rightarrow \mathbb{R}^d, \quad \text{avec } d \text{ le plus petit possible}$$
$$y_i \mapsto z_i$$

..sous contraintes :

- conservation des voisinages ;
- conservation des distances.

Vers une représentation "optimale"

= Transformer l'ensemble des sorties du code $\mathcal{Y} \subset \mathcal{C}([a, b], \mathbb{R})$:

$$r : \mathcal{C}([a, b], \mathbb{R}) \rightarrow \mathbb{R}^d, \quad \text{avec } d \text{ le plus petit possible}$$
$$y_i \mapsto z_i$$

..sous contraintes :

- conservation des voisinages ;
- conservation des distances.

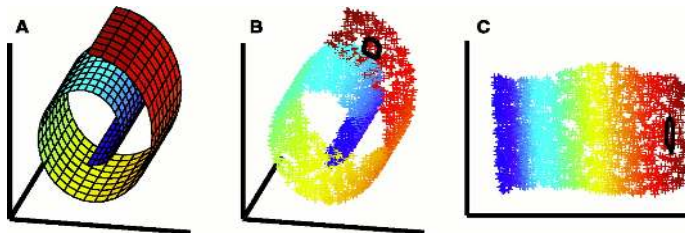


FIG.: carte 2D du jeu de données swissroll

Étapes

distances euclidiennes \leftarrow *distances géodésiques*.

Étapes

distances euclidiennes \leftarrow *distances géodésiques*.

- 1 Géométrie locale : graphe des k plus proches voisins.

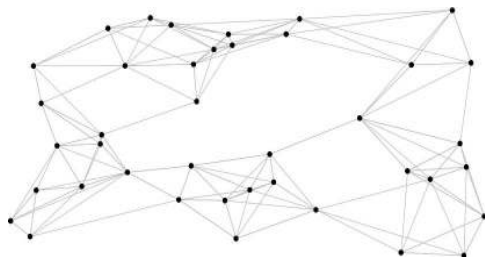


FIG.: Représentation des données dans un graphe ;
sommets = y_i , arêtes = $\|y_i - y_j\|_2$.

Étapes

distances euclidiennes \leftarrow *distances géodésiques*.

- 1 Géométrie locale : graphe des k plus proches voisins.
- 2 Estimation de la dimension : basée sur $\mathbb{P}(Y \in B(y, r)) \propto r^d$.

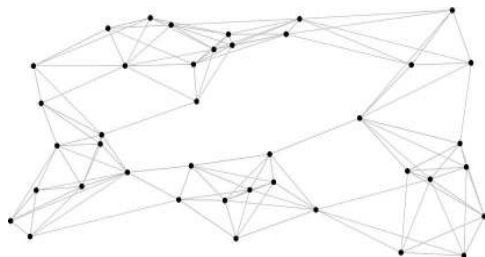


FIG.: Représentation des données dans un graphe ;
sommets = y_i , arêtes = $\|y_i - y_j\|_2$.

Étapes

distances euclidiennes \leftarrow *distances géodésiques*.

- 1 Géométrie locale : graphe des k plus proches voisins.
- 2 Estimation de la dimension : basée sur $\mathbb{P}(Y \in B(y, r)) \propto r^d$.
- 3 Représentation en dimension d selon les contraintes.

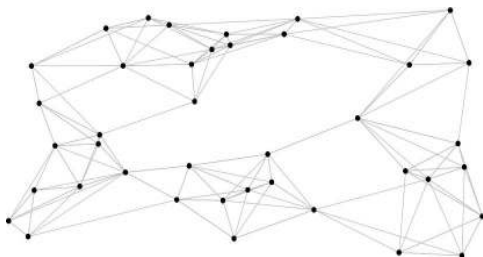


FIG.: Représentation des données dans un graphe ;
sommets = y_i , arêtes = $\|y_i - y_j\|_2$.

1 Exemple introductif

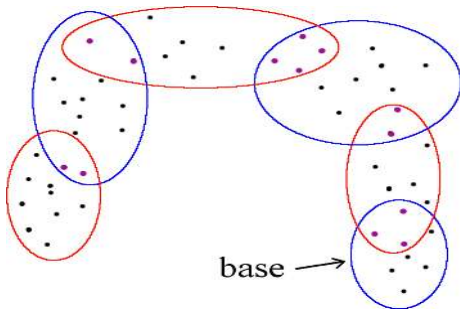
2 Dimension réduite "pas à pas"

- Local PCA Manifold Learning (Zhan et al., 2008)
- Riemannian Manifold Learning (T. Lin & H. Zha, 2006)
- "Local Riemannian Manifold Learning"

3 Applications

Description

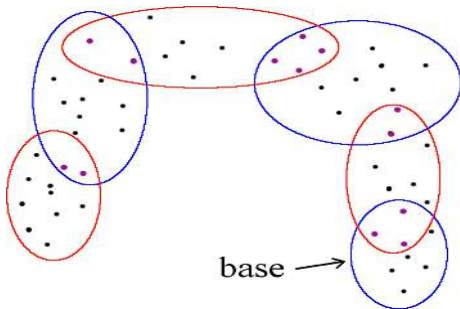
"Traversal
Sequence of
Local
Neighborhoods" :



- 1 trouver une suite $C_1, \dots, C_m \subset \{1, \dots, n\}$ vérifiant :
 - ▶ les C_i sont des voisinages du type k -ppv ;
 - ▶ $\cup_{i=1..m} C_i$ recouvre toutes les données ;
 - ▶ $\forall i > 1, \# (\cup_{j < i} C_j \cap C_i) \simeq \alpha \# C_i (\alpha \in]0, 1[)$;

Description

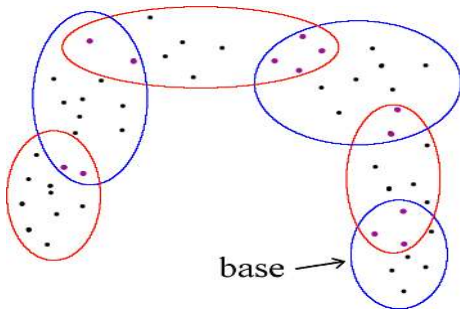
"Traversal
Sequence of
Local
Neighborhoods" :



- 1 trouver une suite $C_1, \dots, C_m \subset \{1, \dots, n\}$ vérifiant :
 - ▶ les C_i sont des voisinages du type k -ppv ;
 - ▶ $\cup_{i=1..m} C_i$ recouvre toutes les données ;
 - ▶ $\forall i > 1, \# (\cup_{j < i} C_j \cap C_i) \simeq \alpha \# C_i (\alpha \in]0, 1[)$;
- 2 ACP locale sur chaque $C_j \rightarrow$ coordonnées locales ;

Description

"Traversal
Sequence of
Local
Neighborhoods" :



- 1 trouver une suite $C_1, \dots, C_m \subset \{1, \dots, n\}$ vérifiant :
 - ▶ les C_i sont des voisinages du type k -ppv ;
 - ▶ $\cup_{i=1..m} C_i$ recouvre toutes les données ;
 - ▶ $\forall i > 1, \# (\cup_{j < i} C_j \cap C_i) \simeq \alpha \# C_i (\alpha \in]0, 1[)$;
- 2 ACP locale sur chaque $C_j \rightarrow$ coordonnées locales ;
- 3 *transformations affines* \rightarrow *coordonnées globales*.

Estimation des coordonnées globales

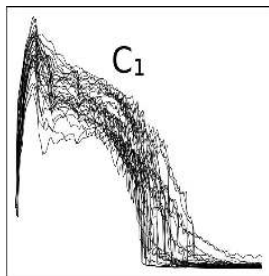


FIG.: données

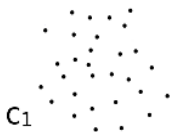


FIG.: coord. locales

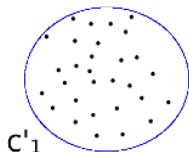


FIG.: coord. globales

- 1 ACP sur $C_1 \rightarrow$ coord. locales c_1 ;
on pose $c'_1 = c_1$ (coord. globales) ;

Estimation des coordonnées globales

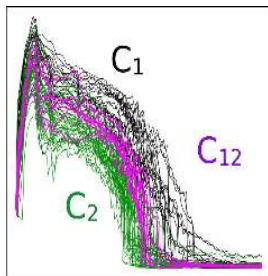


FIG.: données

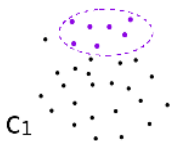


FIG.: coord. locales

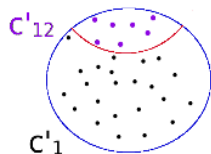


FIG.: coord. globales

- 1 ACP sur $C_1 \rightarrow$ coord. locales c_1 ;
on pose $c'_1 = c_1$ (coord. globales) ;
- 2 c'_1 contient les coord. globales de $C_1 \cap C_2$, notées c'_{12} ;

Estimation des coordonnées globales

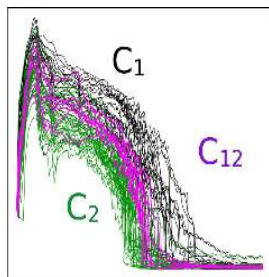


FIG.: données

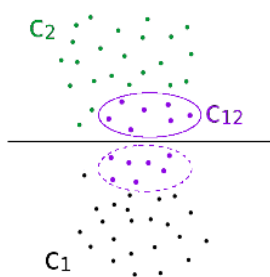


FIG.: coord. locales

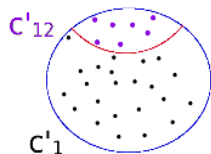


FIG.: coord. globales

- 1 ACP sur $C_1 \rightarrow$ coord. locales c_1 ;
on pose $c'_1 = c_1$ (coord. globales) ;
- 2 c'_1 contient les coord. globales de $C_1 \cap C_2$, notées c'_{12} ;
- 3 ACP sur $C_2 \rightarrow$ coord. locales c_2 ;

Estimation des coordonnées globales

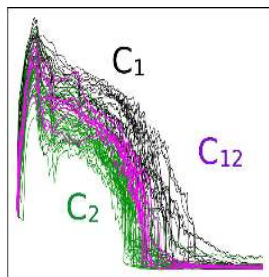


FIG.: données

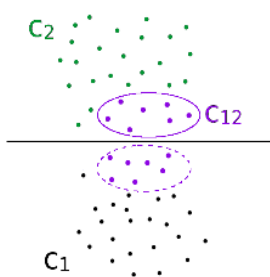


FIG.: coord. locales

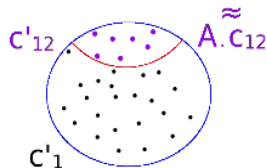


FIG.: coord. globales

- 1 ACP sur $C_1 \rightarrow$ coord. locales c_1 ;
on pose $c'_1 = c_1$ (coord. globales) ;
- 2 c'_1 contient les coord. globales de $C_1 \cap C_2$, notées c'_{12} ;
- 3 ACP sur $C_2 \rightarrow$ coord. locales c_2 ;
- 4 recherche d'une transformation affine A telle que $Ac_{12} \simeq c'_{12}$;

Estimation des coordonnées globales

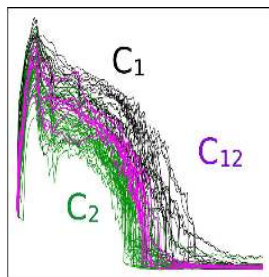


FIG.: données

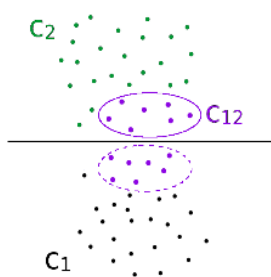


FIG.: coord. locales

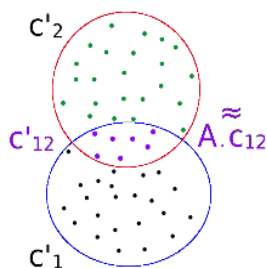


FIG.: coord. globales

- 1 ACP sur $C_1 \rightarrow$ coord. locales c_1 ;
on pose $c'_1 = c_1$ (coord. globales) ;
- 2 c'_1 contient les coord. globales de $C_1 \cap C_2$, notées c'_{12} ;
- 3 ACP sur $C_2 \rightarrow$ coord. locales c_2 ;
- 4 recherche d'une transformation affine A telle que $A c_{12} \simeq c'_{12}$;
- 5 application de A sur les points de $C_2 \setminus C_1 \rightarrow c'_2$;
retour en 2. avec $c'_2 \dots$ etc.

1 Exemple introductif

2 Dimension réduite "pas à pas"

- Local PCA Manifold Learning (Zhan et al., 2008)
- Riemannian Manifold Learning (T. Lin & H. Zha, 2006)
- "Local Riemannian Manifold Learning"

3 Applications

Étapes locales

RML \simeq conservation des angles et des distances géodésiques.

- 1 choisir une courbe origine y_0 parmi les y_i , (p.ex. la moyenne);

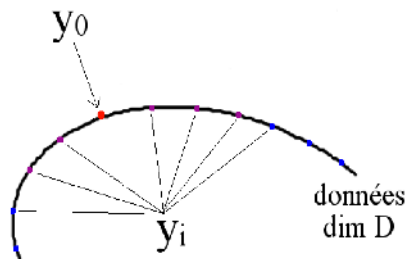


FIG.: Courbe origine y_0

Étapes locales

RML \simeq conservation des angles et des distances géodésiques.

- 1 choisir une courbe origine y_0 parmi les y_i , (p.ex. la moyenne);
- 2 déterminer une base locale $Q_0 = (e_1, \dots, e_d)$ de l'espace tangent en y_0 (ACP sur les points du voisinage);

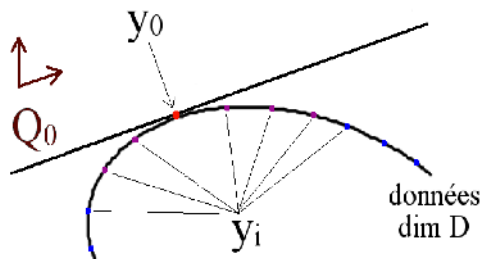


FIG.: Plan tangent en y_0 + base locale Q_0

Étapes locales

RML \simeq conservation des angles et des distances géodésiques.

- 1 choisir une courbe origine y_0 parmi les y_i , (p.ex. la moyenne);
- 2 déterminer une base locale $Q_0 = (e_1, \dots, e_d)$ de l'espace tangent en y_0 (ACP sur les points du voisinage);
- 3 calculer les coordonnées z_i des courbes y_i "proches" de y_0 en projetant sur la base Q_0 ,

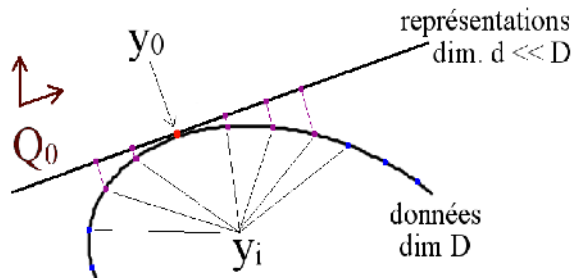


FIG.: Coordonnées z_i sur le plan tangent

Étapes locales

RML \simeq conservation des angles et des distances géodésiques.

- 1 choisir une courbe origine y_0 parmi les y_i , (p.ex. la moyenne);
- 2 déterminer une base locale $Q_0 = (e_1, \dots, e_d)$ de l'espace tangent en y_0 (ACP sur les points du voisinage);
- 3 calculer les coordonnées z_i des courbes y_i "proches" de y_0 en projetant sur la base Q_0 ,
- 4 puis en normalisant pour vérifier $\|y - y_0\| = \|x - x_0\|$.

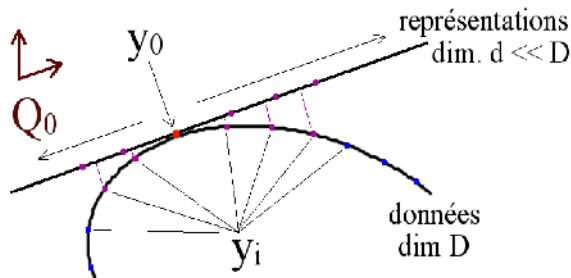


FIG.: Normalisation des coordonnées z_i

"Loin" de y_0

Étape 4 : pour y éloigné de y_0 ,

- y_p = prédécesseur de y sur un plus court chemin issu de y_0
- y_{i_1}, \dots, y_{i_d} = voisins de y_p déjà traités (parcours en largeur)

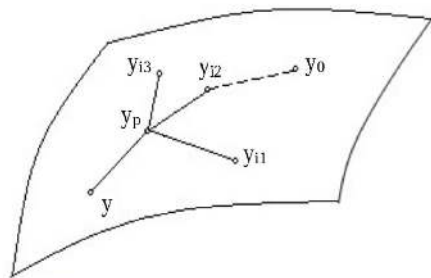


FIG.: Données y_i en dim. D

"Loin" de y_0

Étape 4 : pour y éloigné de y_0 ,

- y_p = prédécesseur de y sur un plus court chemin issu de y_0
- y_{i_1}, \dots, y_{i_d} = voisins de y_p déjà traités (parcours en largeur)

$z = r(y)$ calculé

- en conservant au mieux les angles : $\widehat{zz_p z_{i_j}} \simeq \widehat{yy_p y_{i_j}}$;
- sous la contrainte $\|y - y_p\| = \|z - z_p\|$.

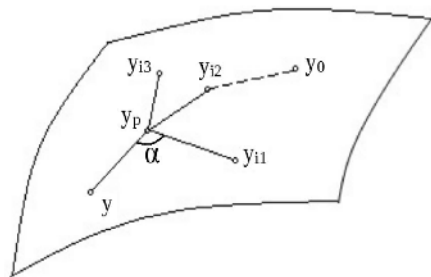


FIG.: Données y_i en dim. D

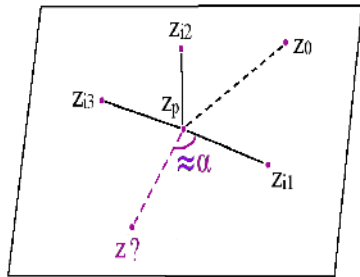


FIG.: $z_i = r(y_i)$ en dim. $d \ll D$

1 Exemple introductif

2 Dimension réduite "pas à pas"

- Local PCA Manifold Learning (Zhan et al., 2008)
- Riemannian Manifold Learning (T. Lin & H. Zha, 2006)
- "Local Riemannian Manifold Learning"

3 Applications

Fusion

RML : combien de projections sur le plan tangent ?

- Pas assez : imprécisions accumulées.
- Trop : simplification excessive de la surface.

Fusion

RML : combien de projections sur le plan tangent ?

- Pas assez : imprécisions accumulées.
- Trop : simplification excessive de la surface.

LPcaML : qualité de l'approximation ACP locale ?

Choix délicat du nombre de voisins . . .

Fusion

RML : combien de projections sur le plan tangent ?

- Pas assez : imprécisions accumulées.
- Trop : simplification excessive de la surface.

LPcaML : qualité de l'approximation ACP locale ?

Choix délicat du nombre de voisins . . .

"Nouvel" algorithme

LPcaML, en remplaçant les ACP par RML.

Fusion

RML : combien de projections sur le plan tangent ?

- Pas assez : imprécisions accumulées.
- Trop : simplification excessive de la surface.

LPcaML : qualité de l'approximation ACP locale ?

Choix délicat du nombre de voisins . . .

"Nouvel" algorithme

LPcaML, en remplaçant les ACP par RML.

- Nombre de voisins initial minimal.
- Meilleures approximations locales.

Exemples

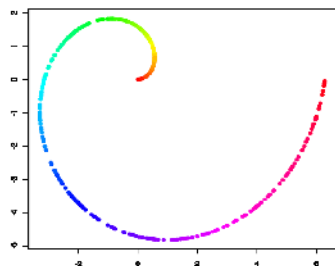


FIG.: Swissroll, 400 points 3D

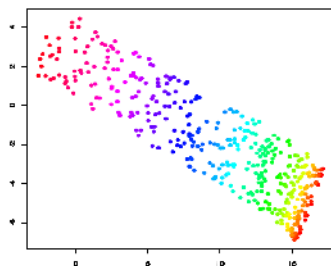


FIG.: Représentation LpcaML

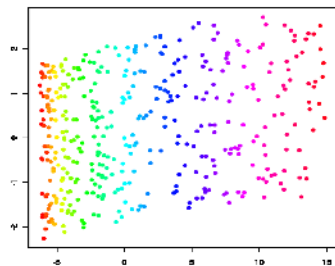


FIG.: Représentation RML

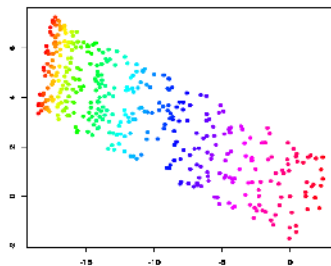


FIG.: Rep. LpcaML + RML

1 Exemple introductif

2 Dimension réduite "pas à pas"

- Local PCA Manifold Learning (Zhan et al., 2008)
- Riemannian Manifold Learning (T. Lin & H. Zha, 2006)
- "Local Riemannian Manifold Learning"

3 Applications

Test I - température "facile"

100 simulations,
4 dimensions en entrée,
168 points de discrétisation.
dimension estimée : 4

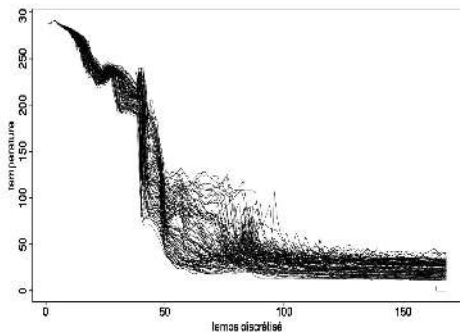


FIG.: Les 100 sorties du code

Test I - température "facile"

100 simulations,
4 dimensions en entrée,
168 points de discrétisation.
dimension estimée : 4

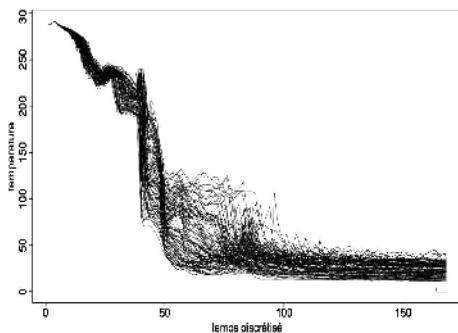


FIG.: Les 100 sorties du code

	ACP	LPcaML	RML	isomap
$d = 5$	6.6	5.4	6.4	6.1
$d = 7$	5.2	4.9	5.8	5.7
$d = 9$	4.5	4.6	5.5	5.5

TAB.: Erreur moyenne L_2 en fonction de d .

Courbes résultats ($d = 4$)

Courbes en noir, \pm écart-type en bleu.

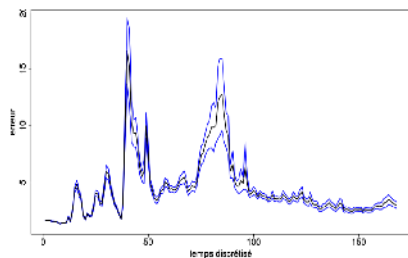


FIG.: ACP

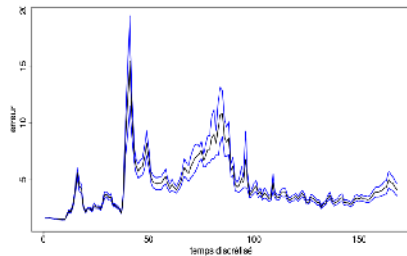


FIG.: LpcaML

Courbes résultats ($d = 4$)

Courbes en noir, \pm écart-type en bleu.

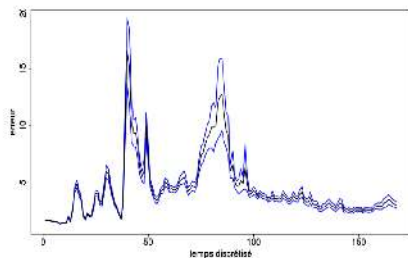


FIG.: ACP

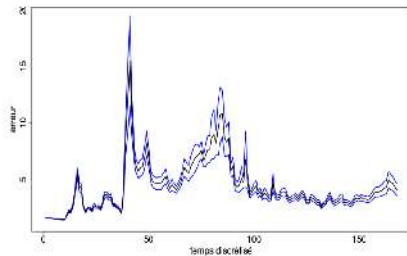


FIG.: LpcaML

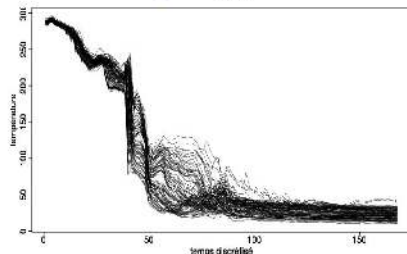
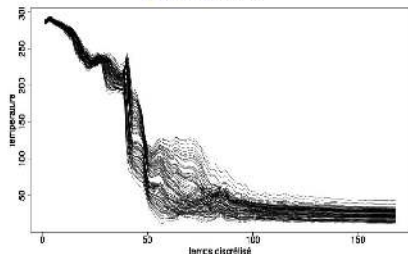


FIG.: Courbes reconstruites; ACP à g., LpcaML à d.

Test II - température "difficile"

600 simulations,
11 dimensions en entrée,
1039 points de discrétisation.
dimension estimée : 7

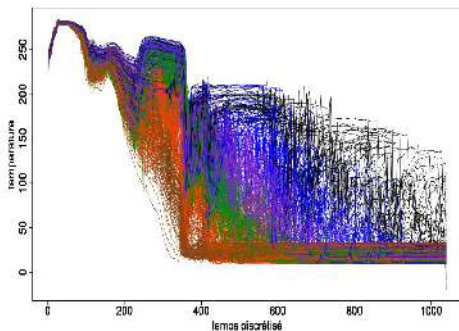


FIG.: Les 600 sorties du code

Test II - température "difficile"

600 simulations,
11 dimensions en entrée,
1039 points de discrétisation.
dimension estimée : 7

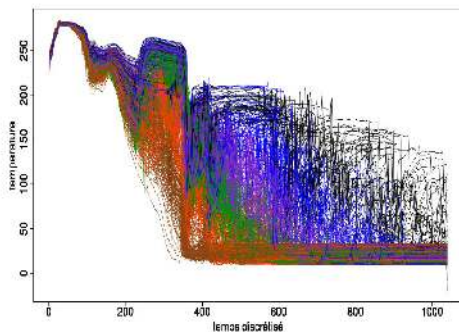


FIG.: Les 600 sorties du code

	ACP	LPcaML	RML	isomap
$d = 5$	12.5	13.3	14.4	14.5
$d = 7$	11.1	12.7	13.7	14.0
$d = 9$	10.1	11.8	13.0	13.9

TAB.: Erreur moyenne L_2 en fonction de d .

Courbes d'erreur pour $d = 7$

Courbes en noir, \pm écart-type en bleu.

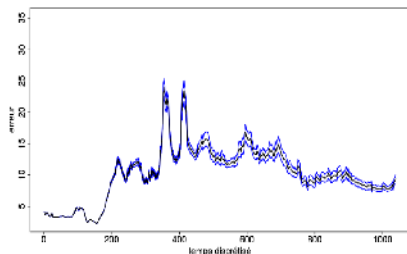


FIG.: ACP

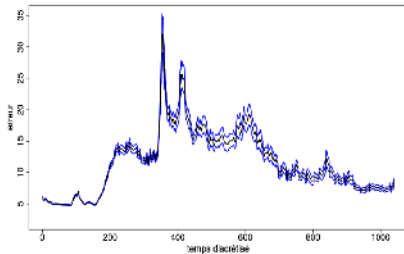


FIG.: LpcaML

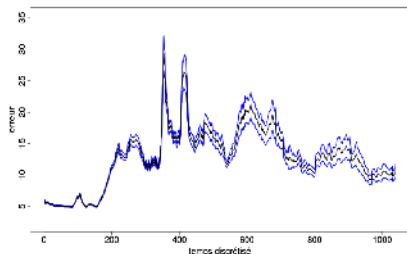


FIG.: RML

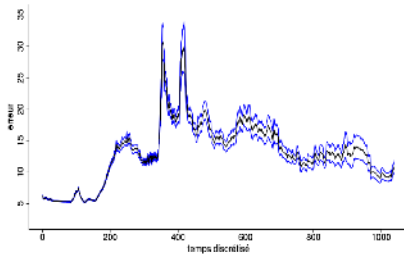


FIG.: isomap

Courbes reconstruites ($d = 7$)

Noir : vraie courbe ; bleu : LPcaML ; rouge : ACP.

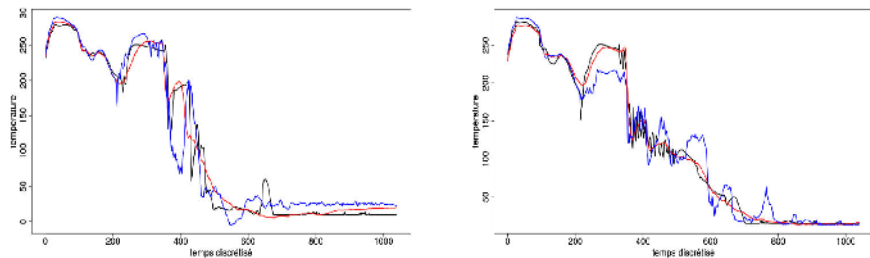


FIG.: Courbes mieux approximées par ACP

Courbes reconstruites ($d = 7$)

Noir : vraie courbe; bleu : LPcaML; rouge : ACP.

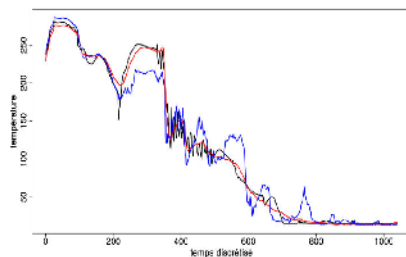
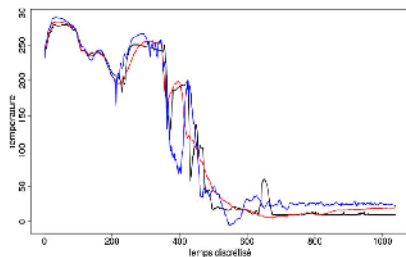


FIG.: Courbes mieux approximées par ACP

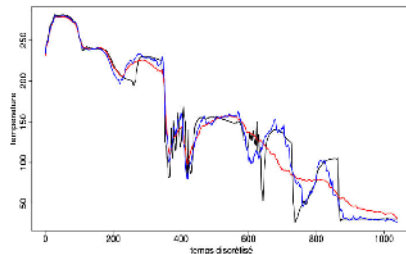
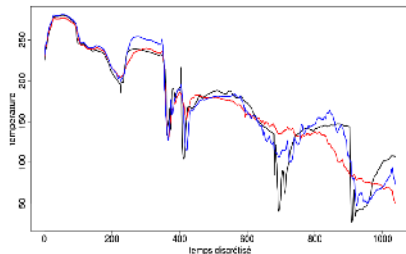


FIG.: Courbes mieux approximées avec LPcaML

Conclusion et perspectives

Réduction non linéaire parfois mieux adaptée (quand ?)

Conclusion et perspectives

Réduction non linéaire parfois mieux adaptée (quand ?)

→ Principaux paramètres à optimiser (automatiquement ?)

- nombre de plus proches voisins ;
- fraction de recouvrement des voisinages ;

Conclusion et perspectives

Réduction non linéaire parfois mieux adaptée (quand ?)

→ Principaux paramètres à optimiser (automatiquement ?)

- nombre de plus proches voisins ;
- fraction de recouvrement des voisinages ;

Fonction de reconstruction R à adapter aux algorithmes.

Conclusion et perspectives

Réduction non linéaire parfois mieux adaptée (quand ?)

→ Principaux paramètres à optimiser (automatiquement ?)

- nombre de plus proches voisins ;
- fraction de recouvrement des voisinages ;

Fonction de reconstruction R à adapter aux algorithmes.

Méthode alternative à explorer

Courbes, surfaces principales "fonctionnelles"

Exemple de surface
principale en $2D$:

