Learning Dynamical Systems with Efficient Kernel Methods

Journées GAIA

Antoine Chatalic

24 novembre 2023

UniGe | DIBRIS | MaLGa MACHINE LEARNING GENOA CENTER | European Research Council Established by the European Commission


Ph.D.
Postdoc
Paris
Rennes
Genova
Dunkerque
Vlaanderen
Köln
Siegen
Thüringen
Southampton
Brighton
Exeter
Portsmouth
Plymouth
Calais
Lille
België / Belgique / Belgien
Bonn
Hessen
Wallonie
Koblenz
Hauts-de-France
Frankfurt am Main
Würzburg
Lëtzebuerg
Guernsey
Le Havre
Rouen
Kaiserslautern
Mannheim
Nürnberg
Plzeň
Jersey
Normandie
Reims
Metz
Saarbrücken
Karlsruhe
Heilbronn
Bayern
Grand Est
Baden-Württemberg
Brest
Troyes
Bretagne
Ulm
Augsburg
München
Le Mans
Centre-Val de Loire
Freiburg im Breisgau
Salzburg
Pays de la Loire
Bourgogne-Franche-Comté
Zürich
Tirol
France
Schweiz/Suisse/Svizzera/Svizra
Poitiers
Lausanne
Trentino-Alto Adige/Südtirol
Genève
La Rochelle
Limoges
Clermont-Ferrand
Annecy
Udine
Lombardia
Veneto
Golfe de Gascogne / Golfo de Vizcaya
Auvergne-Rhône-Alpes
Piemonte
Pavia
Verona
Venezia
Nouvelle-Aquitaine
Alessandria
Piacenza
Parma
Bologna
Città di San Marino
La Spezia
Avignon
Monaco
Firenze
Occitanie
Montpellier
Aix-en-Provence
Gijón / Xixón
Santander
Donostia / San Sebastián
A Coruña
Toscana
Asturias / Asturies
Bilbao
Marseille
Perugia
Grosseto
Galicia
Navarra - Nafarroa
Andorra
Perpignan

Large-scale machine learning with kernel methods

Goal in ML: learn (from data) a model that generalizes to new data samples.
Data $(x_i, y_i)_{1 \leq i \leq n}$ with n large $\rightsquigarrow$ good accuracy but slow algorithms;

Large-scale machine learning with kernel methods

Goal in ML: learn (from data) a model that generalizes to new data samples.
Data $(x_i, y_i)_{1\leq i\leq n}$ with n large $\rightsquigarrow$ good accuracy but slow algorithms;

My goal: compress (time/space) learning algorithms using randomized approximations.

Focus: kernel methods.

(César — Renault VL 06 — Photo: marcovdz)

Why compressing?

Example: kernel ridge regression (KRR)

$$f_{\mathrm{KRR}} := \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|_{\mathcal{H}}^2$$

Why compressing?

Example: kernel ridge regression (KRR)

$$f_{\mathrm{KRR}} := \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|_{\mathcal{H}}^2$$

Space of functions $f(x) = \langle u, \phi(x) \rangle$
RKHS with kernel $\kappa(x, y) = \langle \phi(x), \phi(y) \rangle$

Why compressing?

Example: kernel ridge regression (KRR)

Space of functions $f(x) = \langle u, \phi(x) \rangle$
RKHS with kernel $\kappa(x, y) = \langle \phi(x), \phi(y) \rangle$

$$f_{\mathrm{KRR}} := \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|_{\mathcal{H}}^2$$

$$= \sum_{i=1}^{n} w_i \phi(x_i) \quad \text{with} \quad w := (K_n + \lambda n I)^{-1} y$$

Space: $O(n^2)$, Time: $O(n^3)$.

($n = 10^6$ samples, 64 bit precision $\rightsquigarrow$ 8000 GB of RAM)

Sketching for ML: compression with no tradeoff

Important to keep in mind the **end goal** when compressing.
(Think of signal compression!)

Sketching for ML: compression with no tradeoff

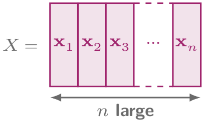
Important to keep in mind the **end goal** when compressing.
(Think of signal compression!)

The goal in statistical ML is to **generalize to new data**. For a sketched algorithm:

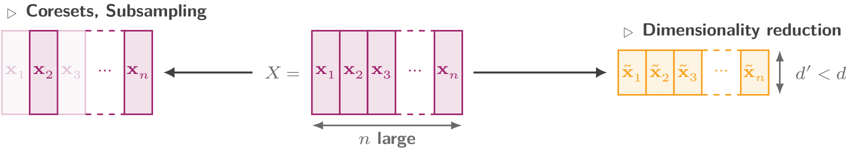
$$\text{Generalization error} = \text{Bias} + \underbrace{\text{Epistemic error}}_{\text{Decreases with } n} + \underbrace{\text{Approximation error}}_{\text{Can be tuned!}}$$

One can often compress **without any tradeoff.**

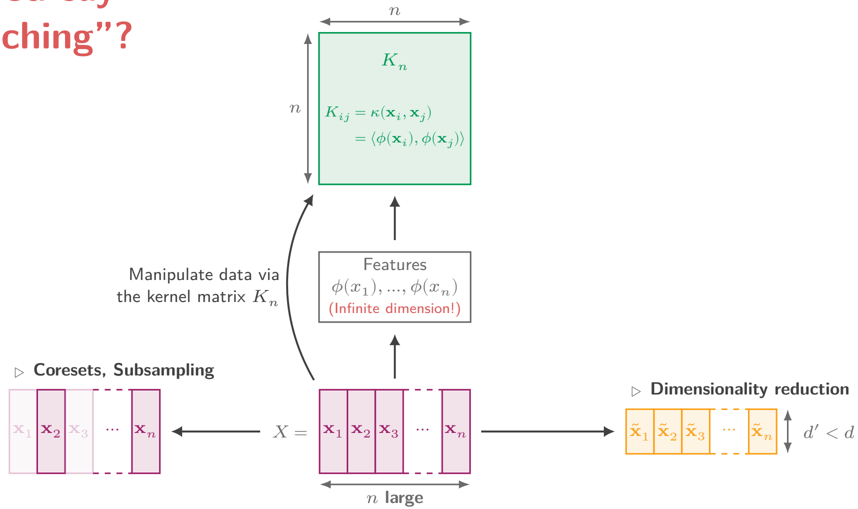
Did you say “sketching”?



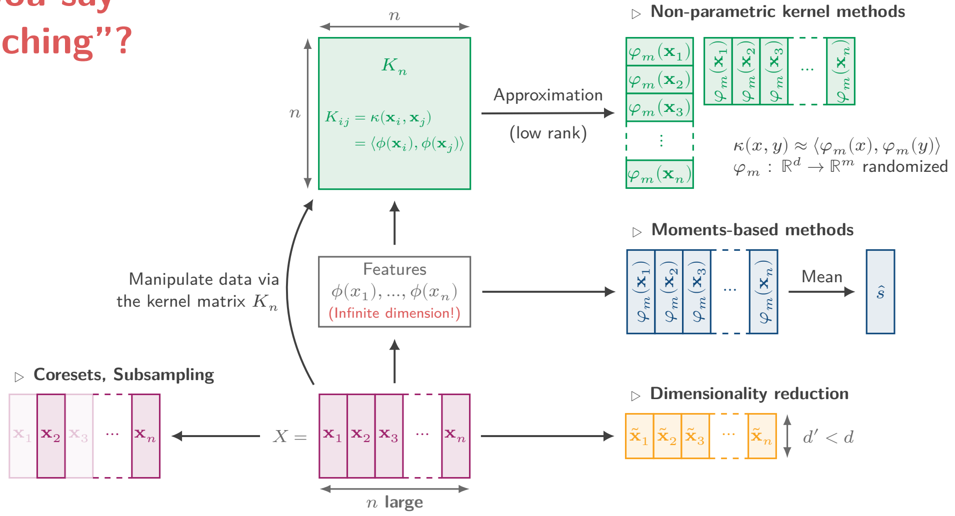
Did you say "sketching"?



Did you say "sketching"?



Did you say "sketching"?



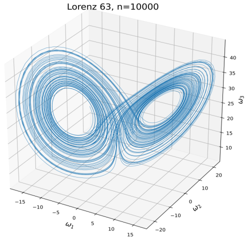
Learning Dynamical Systems

(Joint work with G. Meanti, V. Kostić, P. Novelli, M. Pontil, L. Rosasco)

Dynamical Systems

Discretized dynamical system with state x:

$$x_{t+1} = F(x_t)$$



Typically non-linear, stochastic.

Goals:

- forecasting;
- estimate the system (interpretability);
- control (then $x_{t+1} = F(x_t, u_t)$).

Linear Approximations to Dynamical Systems

Linear systems can be described by their spectral decomposition
$\rightsquigarrow$ efficient algorithms for estimation, prediction and control.

Problem: most encountered dynamical systems are non-linear.

Approach: choose a **non-linear** feature ϕ such that the dynamics of the lifted states $\phi(x_t)$ is approximately linear:

$$\phi(x_{t+1}) \approx A\phi(x_t)$$

Learning dynamical systems

Koopman operator:

$$(\mathcal{K}\varphi)(x) = \mathbf{E}[\varphi(F(x))], \quad \forall \varphi \in \mathcal{F}$$

- advances a measurement function (in $\mathcal{F}$) of the state forward in time;
- defined over an infinite-dimensional space of observable functions;
- linear operator.

Goal: approximate $\mathcal{K}$ ...

- for prediction;
- to compute an eigenfunctions/values $\rightsquigarrow$ this usually provides an interpretable decomposition of the dynamics (especially the principal modes).

Just another regression problem

Formalization: auto-regression problem using training pairs $(x_t, y_t = x_{t+1})$ and a feature map ϕ:

$$\hat{\mathcal{R}}(A) := \frac{1}{n}\sum_{i=1}^{n}\left\|\phi(x_{i+1}) - A\phi(x_i)\right\|^2$$

Intuitively, $A : \mathcal{H} \to \mathcal{H}$ approximates the restriction of $\mathcal{K}$ to the chosen RKHS $\mathcal{H}$.

Multiple regularizations possible. Minimizers depend on the covariance & cross-covariance.

Just another regression problem

Formalization: auto-regression problem using training pairs $(x_t, y_t = x_{t+1})$ and a feature map ϕ:

$$\hat{\mathcal{R}}(A) := \frac{1}{n}\sum_{i=1}^{n}\left\|\phi(x_{i+1}) - A\phi(x_i)\right\|^2$$

Intuitively, $A : \mathcal{H} \to \mathcal{H}$ approximates the restriction of $\mathcal{K}$ to the chosen RKHS $\mathcal{H}$.

Multiple regularizations possible. Minimizers depend on the covariance & cross-covariance.

Cost of minimizing $\hat{\mathcal{R}}$: $O(n^2)$ space / $\Theta(n^3)$ time.

The Nyström approximation

We "compress" using a subsample $\tilde{x}_1, ..., \tilde{x}_m$ of the data.

Multiple interpretations:

- Look for a minimizer of $\hat{\mathcal{R}}$ defined on $\mathcal{H}_m$ rather than $\mathcal{H}$, where

$$\mathcal{H}_m := \mathrm{span}(\phi(\tilde{x}_1), ..., \phi(\tilde{x}_m)).$$

- Approximate the $n \times n$ kernel matrix by a rank-m approximation.

$$\kappa(x, y) \approx \langle P_m\phi(x), P_m\phi(y)\rangle, \quad P_m \text{ orthogonal projector on } \mathcal{H}_m.$$

Intuition:

- Best rank-m approximation is costly (eigendecomposition).
- Few samples are enough to estimate the covariance principal subspaces.

Multiple estimators

We provide compressed variants for...

- **ridge** regression (KRR): min $\hat{\mathcal{R}}$ with Tikhonov regularization;
- **principal component** regression (PCR): least-squares after projection on top eigenfunctions of C;
- **reduced rank** regression (RRR): min $\hat{\mathcal{R}}$ under a hard rank constraint (more robust for eigenvalues than PCR [Kostic, 2023]).

Estimators computable in $\Theta(m^3 + m^2 n) = \Theta(m^2 n)$ time.
One can choose m to get optimal rates in $O(n^2)$ time.

Learning rates

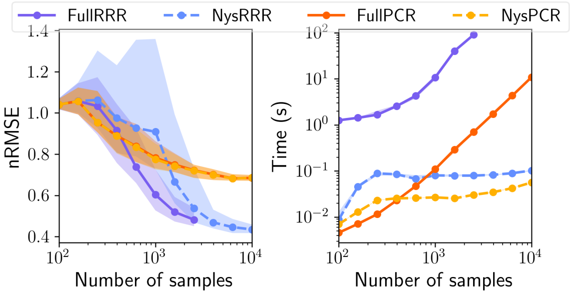
We consider a time-homogeneous Markov process with invariant density π.
Let ρ denote the distribution of (X_t, X_{t+1}).
We consider rates in operator norm, for i.i.d. data $(x_i, y_i)_{1\leq i\leq n}$.

Let $C = \mathbf{E}_\pi \phi(x) \otimes \phi(x)$, and $\beta \in (0, 1]$ such that $\lambda_i(C) \leq ci^{-1/\beta}$.

Contribution: We reach the optimal learning rates $O(n^{-1/(2(1+\beta))})$...
...while using a sketch size ranging from $m \approx \log(n)$ to $m \approx \sqrt{n}$.

[Meanti et al., 2023. ***Estimating Koopman Operators with Sketching to Provably Learn Large Scale Dynamical Systems***]

Experimental results (toy dataset)

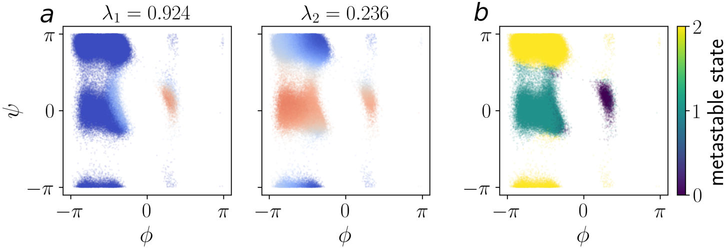


Lorenz '63 system (toy example). Setting: $m = 250$, n increasing.
Much faster estimators to reach a similar accuracy.

Experimental results (large-scale dataset)

Application in molecular dynamics.

- system = molecule structure (position of atoms, encoded by pairwise distances)
- the recovered top two eigenfunctions coincide with angles ψ, ϕ (known to capture relevant long-term dynamics).



Setting: $n \approx 450\,000$, $m = 10\,000$, RRR estimator. (Right = PCCA+ trained on the eigenfunctions).

Conclusion and perspectives

Challenges:

- Our rates are for i.i.d. data. Not realistic in practice. (First step: use results for mixing processes.)
- Analysis with refined hypotheses: source condition, misspecified setting...

Perspectives

- Generalization for control!

$$x_{t+1} = F(x_t, u_t)$$

(Image: Real Estate Japan)