Learning Functional Maps for Nonlinear Dynamical Systems

Anand Srinivasan
Massachusetts Institute of Technology
asrini@mit.edu

Abstract

The Koopman Operator is an operator-theoretic framework for describing the evolution of nonlinear dynamical systems in an infinite-dimensional linear space, lifting state-space dynamics to observable functions of the state. Of interest to theoreticians and practitioners is identifying features of the underlying system, such as invariant sets, by the study of maps in the function space. In this paper, we propose an algorithm for learning an empirical Koopman operator, or functional map, directly from data. Previous methods such as Extended Dynamic Mode Decomposition rely on the manual specification of a basis for the Koopman operator; we use kernel methods to automatically lift a low-dimensional state space into an infinite-dimensional function space. We demonstrate how to achieve space efficiency while doing so using random Fourier features. Finally, we experimentally test our functional map predictor in systems with non-unit spectra, such as the Duffing oscillator and FitzHugh-Nagumo model of neural dynamics.

1 Introduction: Koopman theory of nonlinear dynamics

1.1 Motivation: predicting system evolution

Predicting the evolution of real-world systems is an important problem across the natural sciences, from forecasting the weather to predicting economic outcomes to creating stable mechanical devices such as robots and aircraft. From a statistical learning theory perspective, two major hurdles in the design of predictive models are (a) the inability to observe system states directly, and (b) inherent nonlinearities which make model selection difficult.

Consider a discrete-time dynamical system as a sequence x generated by

$$x \mapsto F(x), \quad F : X \to X$$

where F is an invertible, nonlinear function over state space X. The *inverse problem* of learning F from (dependent; non-i.i.d.) samples $\{(x_i, x_{i+1})\}$ is called *system identification* and is of interest in the control and prediction of dynamical systems. We phrase the problem of learning F as an empirical risk minimization:

$$\hat{F}_* = \min_{F \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \ell(F(x_i), x_{i+1})$$

Assuming underlying linear F, optimal SVD-based solutions exist for finding it, such as the Eigensystem Realization Algorithm [1]. The theory for nonlinear systems is generally restricted to analysis around equilibria, involving linearization techniques which are intrinsically locally predictive. The problem for nonlinear systems is to design the hypothesis space $\mathcal{F}$ and find a parsimonious predictor F within it, the classical problem of machine learning.

Preprint. Under review.

1.2 Linear models for nonlinear systems

As early as the 1930's, work by B.O. Koopman and von Neumann [2] suggested that nonlinear systems could be approached using *linear* maps, by studying the effect of F via *functions* on X. The so-called Koopman Operator is a map $\mathcal{K}$ between function spaces that commutes with F:

$$\begin{tikzcd}
G(X) \arrow[r, "\mathcal{K}"] & G(X) \\
X \arrow[u, "g"] \arrow[r, "F"] & X \arrow[u, "g"]
\end{tikzcd}$$

where the lifting functions $g : X \to \mathbb{C}$ are called *observables* which yield instantaneous descriptions (think: sensors) of the system. This is a dynamical system over function spaces of X, which is useful for our analysis because it is now linear:

$$\mathcal{K}(ag_1 + bg_2) = (ag_1 + bg_2) \circ F = ag_1 \circ F + bg_2 \circ F = a\mathcal{K}g_1 + b\mathcal{K}g_2 \quad \forall a, b \in \mathbb{R}$$

What's more interesting is, given the functional map $\mathcal{K}$, we can completely recover the dynamics F. Adapting an argument from [3], let a simple one-to-one map from X into $G(X)$ be given by the indicator functions:

$$g_i(x) = \mathbb{I}\{x = x_i\}$$

Then, for any x, $F(x)$ can be recovered as

$$F(x) := \arg\max_x \, (\mathcal{K}g)(x)$$

Thus, to learn F, it suffices to solve an ERM over a *linear* hypothesis space:

$$\hat{\mathcal{K}}_* = \min_{\mathcal{K} \in \{T : G(X) \to G(X)\}} \frac{1}{n} \sum_{i=1}^{n} \ell_G((g_1, .., g_d), (\mathcal{K}g_1, .., \mathcal{K}g_d)) \tag{1}$$

This is useful as we can now use linear algebraic notions like basis, span, and invariance in our design of the hypothesis space. Note, however, that we have paid a cost in dimension: for smooth observables g with finite support, by the Stone-Weierstrass theorem, they are approximable with polynomials of arbitrary degree, resulting in G being infinite-dimensional. In section (2.1), we will discuss how to address this problem with kernel methods.

1.3 Related work

For the special case that $g(x) = x$ (which is essentially the restriction of $\mathcal{K}$ to the identity function), solving the above ERM can be done exactly via a simple pseudoinverse problem. Let $x_i \in \mathbb{R}^d$ be a sample at a given instant, and $X_i^j = [g(x_i)...g(x_j)]$ be a column matrix of such samples. Then, the Koopman operator maps over state space:

$$\mathcal{K}X_1^{t-1} = X_2^t \implies \mathcal{K} = X_2^t(X_1^{t-1})^\dagger$$

This approach is termed *Dynamic Mode Decomposition* [4], and is popular in the fluid dynamics community, due to the ability to extract interpretable *eigenfunctions* of $\mathcal{K}$ via the singular value decomposition. These often correspond to real physical phenomena.

However, DMD, and variants such as Extended-DMD [5], suffer from the design problem of having to specify a (fixed) observable function basis. A further complexity is that this chosen basis may not be an invariant subspace of $\mathcal{K}$ for that system, imbuing the model with high bias by design.

Such algorithms can be called *point-wise* because, as in classical supervised learning, the objective is to learn a point-to-point mapping of samples in some given vector space.

On the other hand, learning the functional map directly (as opposed to a point-wise correspondence) is less common in machine learning, but is an up-and-coming technique in the field of computational geometry for problems of *shape matching*. We note a surprising similarity between the goal of predicting a smooth dynamical system and matching shapes on manifolds under near-isometric transformations (such as an animated character). Such methods are called *functional maps* [3], and we take inspiration from this approach in this paper.

2 Methods

2.1 Kernelized observable functions

We established that a state-space predictor can be found by solving an ERM for the function-space operator $\mathcal{K}$. As the hypothesis space is a linear map over an infinite-dimensional function space, we approximate it with a finite-dimensional operator over a subspace $G(X)$ to achieve tractability. This allows us to express $\mathcal{K}$ as a finite-dimensional matrix mapping coefficients in a basis $\{\phi\}$ with span $G'(X) \subset G(X)$. We can think of these ϕ as the "atoms" of our dynamical system representation.

Choosing the correct ϕ is data-dependent and is much like a *dictionary learning* or autoencoder problem. For example, in fluid flows, ϕ may be a pointwise pressure or vorticity measurement; in neural dynamics, it could be membrane potential at an axon. In Section (2.3) we present our specific choice of ϕ, and in Section (2.4) we discuss the generalization error due to mis-specified basis. For now, assume a basis $\Phi(x)$ is given as follows :

$$\Phi(x) = [\phi_1(x) \cdots \phi_d(x)]^T \quad \left| \begin{array}{l} 1.\ \phi_i : X \to \mathbb{R} \\ 2.\ \exists\, c \in \mathbb{R}^d \text{ s.t. } c^T\Phi(x) = x \end{array} \right.$$

Condition (1) is simply that the basis functions are functionals of the state space. Condition (2) is what ensures recoverability; meaning, after computing a functional map $\mathcal{K}$, there is an observable g we can construct whose output will be the next state. (To see this, $\mathcal{K}g(x) = g(F(x))$, thus $\mathcal{K}$ maps identity $g(x) = x$ to itself, if it exists in the span). This generalizes the indicator trick from section [3] which we first used to show un-lifting from the function space was possible.

Since any observable $\in \text{Span}(\Phi)$ can be expressed as a vector of coefficients $c \in \mathbb{R}^d$, we can reformulate the ERM in terms of $\mathcal{K}$ as a map between two coefficient matrices. Let (X_i, X_{i+1}) be a sample, and $C_i, C_{i+1} \in \mathbb{R}^{k\times d}$ be their respective identity-observable coefficients. Let $\ell(X,Y)$ for matrices X, Y be the squared $\ell_{(2,2)}$ norm of their difference. Then,

$$\mathcal{K}_* = \min_{\mathcal{K}\in\mathbb{R}^{k\times k}} \frac{1}{n}\sum_{i=1}^{n} ||\mathcal{K}C_i - C_{i+1}||_F^2 + \lambda||\mathcal{K}||$$

Now, consider some $g'(x)$ not in the span of $\Phi(x)$. The minimizer $\mathcal{K}_*$ is not guaranteed to correctly map any such g'. How can we increase the power of our basis without adding extra coordinates?

Since each $g(x) \in \text{Span}(\{\phi\})$ has a one-to-one correspondence with its coefficient vector c, consider lifting g, as c, via some function φ to an arbitrary-dimensional feature space. Ignoring the regularization term for clarity,

$$\begin{aligned}\mathcal{K}_* &= \min_{\mathcal{K}} \frac{1}{n}\sum_{i=1}^{n} ||\mathcal{K}\varphi(C_i) - \varphi(C_{i+1})||_F^2 \\ &= \min_{\mathcal{K}} \frac{1}{n}\sum_{i=1}^{n} \text{Tr}(\mathcal{K}^T\mathcal{K}\varphi(C_i)^T\varphi(C_i)) - 2\,\text{Tr}(\mathcal{K}\varphi(C_i)^T\varphi(C_{i+1})) + \text{Tr}(\varphi(C_{i+1})^T\varphi(C_{i+1}))\end{aligned}$$

It can be shown (not proven here) that any positive-definite matrix K (not confused with $\mathcal{K}$) defines an inner product over a feature space induced by some φ. This is the *kernel trick* and we can use this to rewrite our expression in terms of a kernel function $K(x,z) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$:

$$\mathcal{K}_* = \min_{\mathcal{K}} \frac{1}{n}\sum_{i=1}^{n} \text{Tr}(\mathcal{K}^T\mathcal{K}K(C_i,C_i)) - 2\,\text{Tr}(\mathcal{K}K(C_i,C_{i+1})) + \text{Tr}(K(C_{i+1},C_{i+1})) \tag{2}$$

For matrix-valued arguments X, Y, $K(X,Y)$ will act as an inner product matrix instantiated on training data. We discuss in Section (2.2) the efficiency of this approach. Since $\mathcal{K}$ is finite-dimensional and $\ell^2_{(2,2)}$ is convex, we can use gradient methods to find the minimizer:

$$\nabla_{\mathcal{K}}L = \frac{1}{n}\sum_{i=1}^{n} \mathcal{K}(K(C_i,C_i)^T + K(C_i,C_i)) - 2K(C_i,C_{i+1})^T \tag{3}$$

$$\mathcal{K}_{i+1} = \mathcal{K}_i - \gamma\nabla_{\mathcal{K}}L \tag{4}$$

for learning rate $\gamma > 0$. Finally, given $\mathcal{K}_*$, our state-space predictor is

$$x_{i+1} = (\varphi^{-1}(\mathcal{K}_* \varphi(c_i)))^T \Phi(x_i) \tag{5}$$

where c_i is prepared according to $c_i = \min_c ||c^T \Phi(x_i) - x_i||^2$, so that $\varphi^{-1} \circ \mathcal{K} \varphi$ maps the identity observable over x_i to identity over x_{i+1}.

2.2 Kernel approximation via random Fourier features

In the previous section, we discussed how to mitigate the inadequacies of a hand-designed basis via kernelization. However, the storage cost of a kernel matrix $K(.,.)$ scales in the number of training examples as $O(n^3)$. This becomes quickly prohibitive for dynamics simulations, where we may have virtually infinite-length trajectories spanning many initial conditions. Note that whether we pre-compute K or evaluate it on-the fly simply trades space for time, and does not improve things.

One idea is to approximate the implicit feature map $\varphi(.)$ with a low-dimensional explicit map $z(.)$:

$$K(x, y) = \varphi(x)^T \varphi(y) \approx z(x)^T z(y)$$

We reconstruct an argument from [6]. Let $K(x, y)$ be a *shift-invariant* kernel, so that $K(x, y) = k(x-y)$ for some $x, y \in \mathbb{R}^m$. Several popular kernels such as the Gaussian, $k(x, y) = \exp(-\gamma||x - y||^2)$, satisfy this property. The Fourier inversion theorem states that we can recover $k(x - y = \Delta)$ as

$$k(\Delta) = \int_{\mathbb{R}} \int_{\mathbb{R}^d} e^{2\pi i(\Delta - \delta)\omega} k(\delta)\, d\delta\, d\omega$$

We proceed by constructing an approximation of $k(\Delta)$ via *random* Fourier features. Up to a scaling constant, the Fourier transform $(\mathcal{F}k)(\omega)$ is a probability distribution by positive-definiteness of $K(.,.)$, due to Bochner's theorem [7]. Let us sample frequencies $\omega \in \mathbb{R}^d \sim p(\omega)$ from the Fourier transform and offsets b uniformly from $[0, 2\pi]$, and define the approximate feature map z as:

$$z(x)_i := \sqrt{2}\, \mathrm{Re}\left[e^{\omega_i^T x + b_i}\right] = \sqrt{2} \cos(\omega_i^T x + b_i) \tag{6}$$

Let $z(x) = [z(x)_1 \cdots z(x)_d]$. Since z_ω is a bounded random variable in $[a, b]$, Hoeffding's inequality yields

$$\Pr\left(|z(x)^T z(y) - K(x, y)| \geq \epsilon\right) \leq 2 \exp\left(-\frac{2d\epsilon^2}{(b - a)^2}\right) = 2 \exp\left(-\frac{d\epsilon^2}{4}\right)$$

This guarantees the convergence of $z(x)$ to implicit map $\varphi(x)$ exponentially in d.

2.3 Wavelet energy functions as a natural choice of basis

The fundamental design choice of any algorithm involving the computation of functional maps, such as Extended-DMD [5] or segment matching [3] is choice of basis $\Phi(x) = \{\phi_i(x)\}$, which is hypothesized to be an invariant subspace of the Koopman operator for the system. The empirical Koopman $\mathcal{K}$ is kept faithful to the underlying dynamics F via the *identity* observable $c^T \Phi(x) = x$, as mentioned in Section 2.1.

Let $x(j)$ be a measurement function indexed by discrete channels $j_1 \cdots j_m$ (thus giving us *samples* $x \in \mathbb{R}^m$). The basis selection problem can be re-phrased as choosing an invertible map $\mathcal{F}$ between the function spaces $x(j) \in \mathbb{R}$ and $\phi(x) \in \mathbb{R}$. If we restrict this map to be linear, then the Schwartz kernel theorem [8] states that every such map has an associated integral transform: $(\mathcal{F}x)(\cdot) = \int_j x(j) k(\cdot, j)$ for some *kernel function* k. If the kernel is shift-invariant, then the Fourier expansion $k(\omega, j) = e^{ij\omega}$ is a natural choice, just as we used for our Fourier random features.

However, achieving good spatial reconstruction suggests that k should be able to localize features in both space *and* frequency, and thus should *not* be shift-invariant. The *wavelet* kernel is a natural choice for this case. A wavelet is a function $\Psi(j, \theta, s) \in L^2(\mathbb{R})$ which takes two parameters, *phase* θ and *scale* s which capture position and spread respectively. If we let θ take values only in $\{j\}$ (a fine assumption for a discrete-space system), then a simple way to define an observable basis with the reconstruction property is to take a wavelet per-dimension:

$$\phi_{s,j}(x) := \langle \Psi_{s,j}, x \rangle_k = \sum_{k=1}^{m} x(k)\Psi_{s,j}(k) \tag{7}$$

$$\Phi_s(x) = [\phi_{s,1}(x) \cdots \phi_{s,j}] \tag{8}$$

If the spatial channels of x are real-valued, and Ψ has tight scale, the individual channels of Φ_s will correspond approximately to the energies of x, and the identity observable $C\Phi(x) = x$ will be easy to find.

We also note that wavelet-based descriptor functions enjoy empirical success in functional maps for shape analysis problems [9].

2.4 Analysis: stability with respect to basis mis-specification

Let us consider the case where the true Koopman operator $\mathcal{K}_{\text{True}}$ for some system is invariant on some subspace $G'(X) \subset G(X)$ spanned by $\{\phi(x)\}$, and we have run our ERM with one of these basis functions left out. How biased of an estimator is our in-sample loss?

Let $L(\mathcal{K}_*)$ be a random variable denoting the loss of the ERM minimizer. Leaving out one basis function during learning is equivalent to fixing a random dimension of the canonical basis coefficients C_i to 0; let us denote the minimizer in this case by $\mathcal{K}_*^{-d} := \min_{\mathcal{K}} \hat{L}[C_i^{-d}](\mathcal{K})$. By assumption of the domain of $\mathcal{K}_{\text{True}}$, it follows that $\mathbb{E}L(\mathcal{K}_*^{-d}) \geq \mathbb{E}\hat{L}(\mathcal{K}_*^{-d})$. Then our in-sample estimator has bias

$$\mathbb{E}L(\mathcal{K}_*^{-d}) - \mathbb{E}\hat{L}(\mathcal{K}_*^{-d}) = \mathbb{E}L(\mathcal{K}_*^{-d}) - \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\ell(\mathcal{K}_* C_i^{-d}, C_{i+1})$$

To analyze the second term, we make the intuitive observation that the in-sample loss is more biased if our sample coefficients C_i fail to contain the dth dimension. In dynamics terms, our sampling procedure measured the system only in a particular sub-regime, such as a basin of attraction.

We can upper bound the above by considering the most extreme version of the above, $\text{Span}(C_i^{-d}) < \text{Span}(C_i)$. If the samples X are bounded, since wavelets are square integrable, $|c_{ij}|$ will have some bound α. We conjecture, without rigor, that the above is bounded by

$$\leq \alpha^2 \int_C \Pr(\text{Span}(C_i^{-d}) < \text{Span}(C_i))\, dC$$

given some density function over the sample coefficients. We will not analyze this further, but point the interested reader to the algebraic properties of random matrices [10].

3 Experiments

3.1 LFM algorithm

Data. Arrange samples $x(t) \in \mathbb{R}^n$ of a discrete-time dynamical system into blocks of size b, that is, $X_t^{t+b} := [x(t), ..., x(t+b)]$, letting $b = n$ to obtain square samples. Since ERM assumes i.i.d. samples and dynamical systems exhibit autocorrelation (time-dependence), we use a block bootstrap [11] sampling procedure.

Preprocessing. Choose a mother wavelet Ψ and set of scales s, and prepare function basis $\Phi(x)$ according to (8). For each sample X_t^{t+b}, pre-compute identity observable coefficients according to $c_t = \min_c ||c^T\Phi(x_t) - x_t||^2$. We now have a sequence of canonical coefficients $C_1 \cdots C_{T-b} \simeq X_1^b \cdots X_{T-b}^T$.

Learning. Choose a feature kernel $K(.,.)$ and compute its random Fourier features up to dimension d per (6). Instantiate the Koopman operator $\mathcal{K} \sim \mathcal{N}(0,1)^{d\times d}$. Minimize the ERM (2) with choice of learning rate γ by computing the gradient (3) and the update $\Delta\mathcal{K}$ (4); we use Nesterov-accelerated GD. Terminate when $\Delta\mathcal{K} < \epsilon$, for some value of ϵ.

Prediction. For any new sample $x(t)$, prepare its canonical coefficients and compute $x(t+1)$ according to (5).

Experimental parameters. We use a Gaussian kernel $K(x, y) = \exp(-\gamma||x - y||^2)$, with no. Fourier features: 50, 100, 200, 400. We use an observable function basis consisting of orthonormal cubic spline wavelets [9], with dimension m (the number of spatial channels per sample). Terminate with $||\Delta\mathcal{K}|| \leq 1e - 3$.

3.2 Results

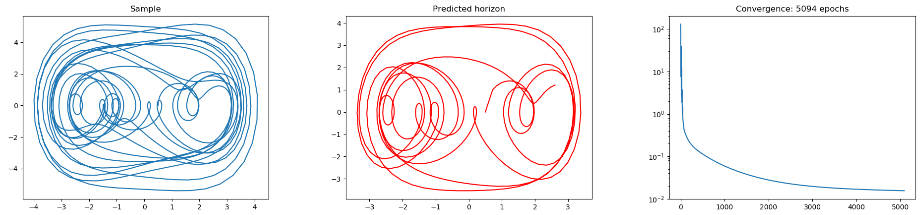
For brevity, we show predictions only for the best LFM variant. Each predicted horizon is calculated using a set of initial conditions, randomly sampled throughout the timeline.

3.2.1 Duffing oscillator

The Duffing oscillator is a non-linear elastic oscillator whose general governing equation is:

$$\ddot{x} + \dot{x} + \beta x + \alpha x^3 = \gamma \cos \omega t$$

We write this as a two-dimensional system of first-order equations, and use the parameters $\alpha = 0.5$, $\beta = 1/16, \gamma = 0.1$, $\omega = 2.0$. It exhibits double-well characteristics, as well as chaos for different values of the elasticity parameter.



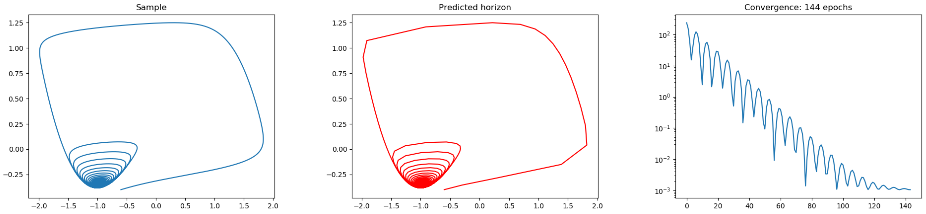
We truncate unbounded predictions, and this results in the partial trajectory reconstruction seen above.

3.2.2 FitzHugh-Nagumo Model

The FitzHugh-Nagumo model is a simplified version of the Hodgkin-Huxley model of neural excitation, with the following governing equations:

$$\dot{v} = v - v^3/3 - w + I_{ext}$$
$$\dot{w} = 0.08(v + 0.7 - 0.8w)$$

We ran a relatively simple experiment using $v_0 = -0.7$, $w_0 = -0.5$, and $I_{ext} = 0.8$. This immediately excites the neuron, which subsequently relaxes into and never leaves its refractory state. Nevertheless, we hope to study dynamics of neural populations in the future with the predictor for an individual in hand.



4 Discussion

To conclude, we formulated a learning functional maps (LFM) algorithm which is able to learn an empirical predictor over function spaces for a nonlinear dynamical system. In the future, we hope to perform more in-depth comparisons to existing system identification algorithms, such as extended-Dynamic mode decomposition, which we were not able to perform due to time constraints. Furthermore, we would like to investigate the effect of different regularization constraints; this was a common topic in the computational geometry literature in particular, where e.g. commutativity with the Laplace-Beltrami operator tremendously improved the recovery of underlying point-to-point maps.

References

[1] Juang, J.-N.; Pappa, R. S. (1985). "An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction". Journal of Guidance, Control, and Dynamics. 8 (5): 620–627. doi:10.2514/3.20031.

[2] B. O. Koopman and J. von Neumann, Dynamical systems of continuous spectra, Proceedings of the National Academy of Sciences of the United States of America 18 (1932), no. 3 255.

[3] Ovsjanikov M., Ben-Chen M., Solomon J., Butscher A., Guibas L.: Functional maps: a flexible representation of maps between shapes. TOG 31, 4 (2012), 30.

[4] P.J. Schmid. "Dynamic mode decomposition of numerical and experimental data." Journal of Fluid Mechanics 656.1 (2010): 5–28.

[5] Williams, Matthew O., Ioannis G. Kevrekidis, and Clarence W. Rowley. “A Data–Driven Approximation of the Koopman Operator: Extending Dynamic Mode Decomposition.” Journal of Nonlinear Science 25.6 (2015): 1307–1346.

[6] Rahimi, A., & Recht, B. (2008). Random features for large-scale kernel machines. In Advances in neural information processing systems (pp. 1177-1184).

[7] W. Rudin. Fourier Analysis on Groups. Wiley Classics Library. Wiley-Interscience, New York, reprint edition, 1994.

[8] Gask, H. (1961). A Proof of the Schwartz’ Kernel Theorem. Mathematica Scandinavica, 8(2), 327-332. Retrieved from www.jstor.org/stable/24488947

[9] Li, C., Ben Hamza, A. A multiresolution descriptor for deformable 3D shape retrieval. Vis Comput 29, 513–524 (2013) doi:10.1007/s00371-013-0815-3

[10] Edelman, A., & Rao, N. R. (2005). Random matrix theory. Acta Numerica, 14, 233-297.

[11] Shalizi, C. (2013). Advanced data analysis from an elementary point of view. 495-521.

[12] Petersen, K. B., Pedersen, M. S. (2008). The matrix cookbook. Technical University of Denmark, 7(15), 510.