Proper Orthogonal Decomposition
Model Reduction

Benjamin Kurt Miller

14.11.2018

1 Introduction

This paper investigates a method, known as the *Proper Orthogonal Decomposition* (written later as POD), in which an optimal basis decomposition is found for a set of experimental or numerically simulated data. Optimality implies that this basis is the most efficient way to capture the dominant components of an infinitely dimensional process using finitely many modes. Before that, some background...

Despite the simplicity of the Navier-Stokes equations (NSE), low-dimensional modelling is called for due to the complexity of the resulting fluid motion. In an incompressible, Newtonian fluid the NSE may be written in terms of a rescaled velocity ($\mathbf{u}$), pressure (p), density (ρ), and Reynolds number (Re based on a suitable macroscopic length scale) as

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{1}{\rho} \nabla p + \frac{1}{Re} \nabla^2 \mathbf{u}; \text{ with } \nabla \cdot \mathbf{u} = 0. \tag{1}$$

You'll notice that the NSE have infinitely many degrees of freedom, as they are partial differential equations. It is stated without proof that the Fourier transform of the NSE indicates strong dynamic coupling between Fourier modes, implying long distance interactions. [1]. In isotropic and homogenous turbulence, flow structures have a range of length scales, suggesting many modes are necessary to model their behavior. [2]. In contrast, flows with low Reynolds number or in constrained geometries have a set of basic *coherent structures* which appear, disappear, then reappear again. [3] The POD is an unbiased technique for finding the modes which can be suitably combined to form these coherent structures.

Our goal is to use the POD to provide a set of spatial basis functions, forming a low dimensional subspace, onto which we can project our governing equations. The POD, in addition to providing an optimal basis, orders each mode according to the descending content of some observable. When we expand in modes which map position to streamwise velocity, this observable is the kinetic energy contained in that mode. Given a descending energetic ordering of spatial basis functions, we can project our system onto a finite, truncated set of highly energetic POD basis functions. Using this reduced basis, we aim to reconstruct the coherent structures by dynamically mixing these POD modes.

In this paper, we will introduce the optimal basis for a general Hilbert space $\mathcal{H}$. The abstract definition will be useful for briefly touching on an example of Principal Component Analysis, a method which produces an optimal basis for a finite dimensional cloud of data points. Principal Component Analysis (or PCA) helps ground the notion of an optimal basis visually since the basis optimially captures the variance of the cloud of data. Next we show how to find the POD modes given velocity data from numerical simulation or experiment. The computational difficulty of the problem depends on the arrangement of the velocity data. In one case we optimize for many grid points compared to time steps, in the other the situation is reversed. Finally, we mention the importance of symmetry when finding the POD modes.

It is worth mentioning the sources from which this work is derived. The document is a synthesis of ideas from "Low-dimensional modelling of turbulence using the proper orthogonal decomposition: A tutorial" by Troy Smith, et. al. [1] and the chapter "Proper Orthogonal Decomposition" from the book *Turbulence, coherent structures, dynamical systems and symmetry* by Philip Holmes, et. al. [4].

2 Proper Orthogonal Decomposition

In this section, we define the POD generally because it is suited for further discussion across domains. We define the POD in a Hilbert space $\mathcal{H}$, with an inner product $(\cdot,\cdot)$. Let $u, \varphi \in \mathcal{H}$ where u represents a typical data element from an ensemble $\{u_{(k)}\}$ and φ is chosen to be an optimal basis element. For example, in turbulence u is a snapshot of the streamwise velocity for the system and φ is a particular mode. Optimality is when φ is chosen such that the average error between u and the projection onto the basis φ is minimized:

$$\min_{\varphi\in\mathcal{H}} \left\langle ||u - \frac{(u,\varphi)}{||\varphi||^2}\varphi|| \right\rangle \tag{2}$$

where $||\cdot||$ is the induced norm $||f|| = (f,f)^{1/2}$ and $\langle\cdot\rangle$ means ensemble average. Depending on the problem, ensemble average might be the time average or an average across several experiments. Minimizing the average error for this projection is equivalent to maximizing the average projection of u onto φ,

$$\max_{\varphi\in\mathcal{H}} \frac{\langle|(u,\varphi)^2|\rangle}{||\varphi||^2}, \tag{3}$$

where $|\cdot|$ implies the absolute value. Now, equation (3) provides a solution for just a single function, but we want a family of functions for the desired basis. I.e. we must use the calculus of variations. We impose the constraint of $||\varphi||^2 = 1$. Recall that extrema are found when the functional derivative vanishes for all variations, i.e. $\frac{d}{d\delta}J[\varphi+\delta\psi]|_{\delta=0} = 0, \delta\in\mathbb{R}$. The functional for this problem is

$$J[\varphi] = \langle|(u,\varphi)^2|\rangle - \lambda(||\varphi||^2 - 1). \tag{4}$$

From equation (3): recall that $(f,g)^* = (g,f)$,

$$\begin{aligned}
&\frac{d}{d\delta}J[\varphi+\delta\psi]\,|_{\delta=0} \\
&\quad= \frac{d}{d\delta}\left[\langle(u,\varphi+\delta\psi)(\varphi+\delta\psi,u)\rangle - \lambda(\varphi+\delta\psi,\varphi+\delta\psi)\right]|_{\delta=0} \\
&\quad= \frac{d}{d\delta}\left[\,\langle(u,\varphi)(\varphi,u) + (u,\varphi)(\delta\psi,u) + (u,\delta\psi)(\varphi,u) + (u,\delta\psi)(\delta\psi,u)\rangle\right. \\
&\qquad\qquad \left. - \lambda\left\{(\varphi,\varphi) + (\varphi,\delta\psi) + (\delta\psi,\varphi) + (\delta\psi,\delta\psi)\right\}\,\right]|_{\delta=0} \\
&\quad= \langle(u,\varphi)(\psi,u) + (u,\psi)(\varphi,u)\rangle - \lambda\left\{(\varphi,\psi) + (\psi,\varphi)\right\} \\
&\quad= 2\mathrm{Re}\left[\langle(u,\psi)(\varphi,u)\rangle - \lambda(\varphi,\psi)\right] = 0.
\end{aligned}$$

The final few steps require interchanging the order of the ensemble average and the inner product as well as the definition of a new linear operator $\mathcal{R}(\cdot) \coloneqq \langle(\cdot,u)u\rangle$. Now we have

$$\begin{aligned}
\langle(u,\psi)(\varphi,u)\rangle - \lambda(\varphi,\psi) &= \langle((\varphi,u)u,\psi)\rangle - \lambda(\varphi,\psi) &(5)\\
&= (\langle(\varphi,u)u\rangle,\psi) - \lambda(\varphi,\psi) &(6)\\
&= (\mathcal{R}\varphi - \lambda\phi,\psi) = 0. &(7)
\end{aligned}$$

Finally, since ψ was an arbitrary variation, we are left with the eigenvalue problem

$$\mathcal{R}\varphi = \lambda\varphi. \tag{8}$$

The optimal basis, defined above, is the set of eigenfunctions $\left\{\varphi^{(n)}\right\}$ of the operator $\mathcal{R}$ which is defined based on the empirical data u from data ensemble $\{u_{(k)}\}$. The eigenfunctions are called the POD modes.

3 Principal Component Analysis

Proper Orthogonal Decomposition in Finite Dimensional Spaces

Principal Component Analysis (PCA) is a special case of the general method defined above which we are calling the POD. Specifically, let $\mathcal{H} = \mathbb{R}^N$ and use the standard inner product $(\mathbf{x}, \mathbf{y}) = \mathbf{y}^T\mathbf{x}$. PCA applies to cases in which the data is a collection of M vectors $\mathbf{u}_{(k)} \in \mathbb{R}^N$, in other words each data point is a collection of N observations. In contrast to the POD which delivers optimal modes, PCA delivers optimal axes, or principal axes, for the cloud of data points. It can also be thought of as fitting an n-dimensional ellipsoid to the data.

In particular, PCA is commonly used in exploratory data analysis and dimension reduction. Since the principal axes are ordered according to the variance along that axis it is often appropriate to truncate axes which have low variance without significant loss of information. In exploratory data analysis this is thought of as viewing the most informative axes of the data. The purpose of this section of the document is to ground the POD modes into something clearer and more visual. Since PCA and the POD are really the same process just applied to finite dimensional samples and infinite dimensional partial differential equations respectively, we can learn a lot of intuition from understanding PCA. After showing the mathematics, we briefly present an example with samples from a multivariate Gaussian distribution and from images of faces.

It is considered standard practice to first make the data *mean free* by subtracting the mean from each observation $\tilde{\mathbf{u}}_{(k)}$,

$$\mathbf{u}_{(k)} = \tilde{\mathbf{u}}_{(k)} - \frac{1}{M}\sum_{k=1}^{M} \tilde{\mathbf{u}}_{(k)}. \tag{9}$$

Depending on the situation, some practitioners also whiten the data by normalizing each axes' variance to one. In this finite dimensional space our operator $\mathcal{R}$ becomes

$$\mathcal{R}\mathbf{x} = \frac{1}{M}\sum_{k=1}^{M} \mathbf{u}_{(k)}^T\mathbf{x}\mathbf{u}_{(k)} \tag{10}$$

$$\mathcal{R} = \frac{1}{M}\sum_{k=1}^{M} \mathbf{u}_{(k)}\mathbf{u}_{(k)}^T \tag{11}$$

$$\mathcal{R}_{ij} = \frac{1}{M}\sum_{k=1}^{M} u_{(k),i}u_{(k),j} \tag{12}$$

where i, j are indices. Notice that in this case $\mathcal{R}$ is a symmetric $N \times N$ correlation matrix $\left\langle \mathbf{u}\mathbf{u}^T \right\rangle$. Equation (8) becomes a regular eigenvalue problem in $\mathbb{R}^N$. The calculation of these eigenvalues is usually done using the singular-value decomposition on $\mathbf{X} = \left[\mathbf{u}_{(1)} \ ... \ \mathbf{u}_{(M)}\right]$ because it can be done in fewer calculations than determining the correlation matrix $\left\langle \mathbf{u}\mathbf{u}^T \right\rangle$ then doing an eigenvalue decomposition on $\left\langle \mathbf{u}\mathbf{u}^T \right\rangle$.

3.1 PCA Examples

Now we present two simple examples of PCA in action. In figure 1 on the left side, a contour plot of the probability density function of the distribution is plotted. On the right side, 2000 samples were drawn from a multivariate Gaussian distribution with mean, $\boldsymbol{\mu} = [0, 0]^T$, and correlation matrix,

$$\mathbf{u}\mathbf{u}^T = \begin{bmatrix} 1.25 & 0.875 \\ 0.875 & 1.0625 \end{bmatrix}. \tag{13}$$

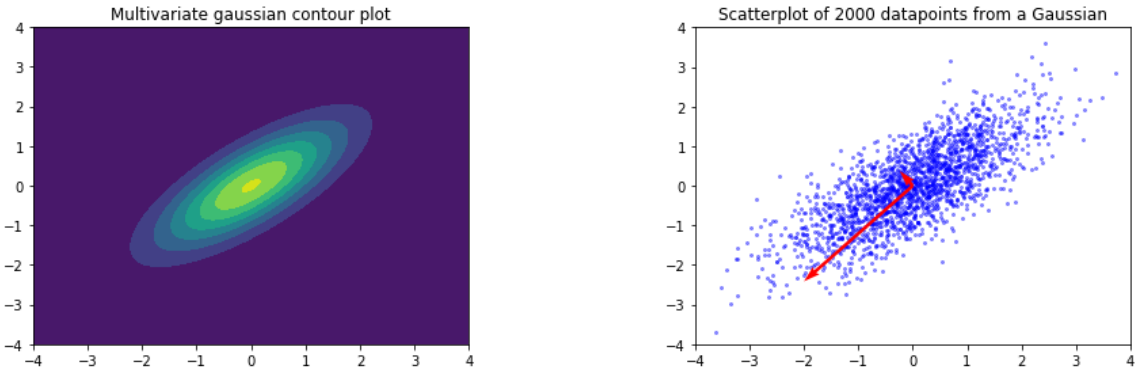

Figure 1: A contour plot of a multivariate Gaussian's probability density function is plotted in two dimensions in the left panel. 2000 samples were drawn from the same distribution and plotted on the right, along with the two principal component vectors. The length of the vectors were scaled up for visibility but the magnitude ratio is accurate. It's important to remember that PCA has no access to the distribution itself, only the samples drawn from it.

The red arrows represent the PCA axes with the relative variance proportional to their relative lengths.

The next example is from the field of computer vision, specifically, facial recognition. The set of grayscale, 100×100 pixel images of human faces exists in $\mathbb{R}^{10000}$; however, the manifold in which most samples lie is significantly lower in dimension. As you can see in figure 2, only 16 PCA eigenvectors already recreate a face fairly accurately. Images from [5].

Figure 2: On the left, samples of grayscale, 100×100 pixel images of human faces. On the right are the first sixteen eigenvectors generated using PCA successively added together to recreate a face. [5].

4 Using the POD modes in Turbulence

In the problems with constrained geometries, we often consider the scalar streamwise velocity $u(x)$ to produce the POD modes for the recreation of coherent structures. In this domain, let $\mathcal{H} = L^2([0,1])$ with inner product

$$(f, g) = \int_0^1 f(x)g^*(x)dx. \tag{14}$$

For an ensemble of scalar-valued functions $u(x)$, our eigenvalue equation 8 becomes

$$\mathcal{R}\varphi(x) = \int_0^1 \langle u(x)u^*(x')\rangle \, \varphi(x')dx' = \lambda\varphi(x); \tag{15}$$

the kernel of this integral equation is the averaged *autocorrelation function* $R(x, x') = \langle u(x)u^*(x')\rangle$. We will discuss the autocorrelation function again briefly in section 5 because the behavior of the autocorrelation function can indicate that certain symmetries exist in our problem.

We will not discuss the expansion further, but simply take it as assumed that $\mathcal{R}$ is "well behaved" enough for it to generate the POD modes in a sensible way. In particular, the POD modes are orthogonal with respect to the inner product (14) and optimal in the sense of capturing, on average, the greatest possible fraction of the total kinetic energy for a projection onto a given number of modes. By considering a something other than velocity for $u(x)$, we could compute different POD modes which optimally represent a different quantity than the kinetic energy. If you are curious about a more rigorous treatment of $\mathcal{R}$ consider looking into [4].

Although we are working in the space $\mathcal{H} = L^2([0, 1])$, in practice, one solves (15) by transforming it into a matrix eigenvalue problem through an appropriate discretization, then solving that problem using a linear algebra software package. We present two methods for calculating the POD modes, the direct method and the method of snapshots. Each method is better suited to a certain relative number of grid points, n_g, and observations or time-snapshots, N_T, in the data ensemble. Although the direct method subsection is more or less a quote from [1], we included it because it is the core step for determining the POD modes. In the method of snapshots subsection directly following it, we present the purpose and main result without an explicit derivation. By avoiding a quote, the author demonstrates comprehension and acknowledges that the reader can simply consult [1, 4] if necessary.

4.1 Discretization: Direct method

We explicitly write our ensemble average as a time average of the N_T snapshots and interchange the sum and integral, we rewrite (15) as

$$\frac{1}{N_T}\sum_{k=1}^{N_T} u_{(k)}(x) \int_{\Omega_x} u^*_{(k)}(x')\varphi(x')dx' = \lambda\varphi(x). \tag{16}$$

We approximate the integral over x' using either the trapezoidal rule or Simpson's rule. In both cases, we can express the integral as

$$\int_{\Omega_x} u^*_{(k)}(x')\varphi(x')dx' = \sum_{i=0}^{n_x} \omega_i u^*_{(k)}(x_i)\varphi(x_i) = \hat{\mathbf{u}}^*_{(k)}\hat{\boldsymbol{\varphi}} \tag{17}$$

where

$$\hat{\mathbf{u}}^*_{(k)} = \begin{bmatrix} \sqrt{\omega_1}u^*_{(k)}(x_1) \\ \sqrt{\omega_2}u^*_{(k)}(x_2) \\ \cdots \\ \sqrt{\omega_{n_x}}u^*_{(k)}(x_{n_x}) \end{bmatrix}, \hat{\boldsymbol{\varphi}} = \begin{bmatrix} \sqrt{\omega_1}\varphi(x_1) \\ \sqrt{\omega_2}\varphi(x_2) \\ \cdots \\ \sqrt{\omega_{n_x}}\varphi(x_{n_x}) \end{bmatrix}, \tag{18}$$

and ω_i are the weight functions for the particular quadrature method used. With these definitions we may write (16) as

$$\frac{1}{N_T}\sum_{k=1}^{N_T} u_{(k)}(x)\hat{\mathbf{u}}^*_{(k)}\hat{\boldsymbol{\varphi}} = \lambda\varphi(x). \tag{19}$$

In particular, this equation is satisfied at each of the n_x grid points, x_j:

$$\frac{1}{N_T}\sum_{k=1}^{N_T} u_{(k)}(x_j)\hat{\mathbf{u}}^*_{(k)}\hat{\boldsymbol{\varphi}} = \lambda\varphi(x_j) \text{ for } j = 1, ..., n_x. \tag{20}$$

Multiplying (20) by $\sqrt{\omega_j}$ for each $j = 1, ..., n_x$, we may write the resulting set of equations as a single matrix-vector equation:

$$\frac{1}{N_T}\sum_{k=1}^{N_T} \mathbf{u}_{(k)}\hat{\mathbf{u}}^*_{(k)}\hat{\boldsymbol{\varphi}} \coloneqq \tilde{\mathbf{A}}\hat{\boldsymbol{\varphi}} = \lambda\hat{\boldsymbol{\varphi}}. \tag{21}$$

This, the integral equation (16) becomes a symmetric (in general, Hermitian) eigenvalue problem for the $n_x \times n_x$ matrix $\tilde{\mathbf{A}}$. It is necessary to multiply the components of the resulting eigenvector by $1/\sqrt{\omega_j}$ to get the POD modes $\left\{\varphi^{(n)}\right\}$, but this requires very little computational effort. The modes may be normalized to ensure that they are orthonormal. Depending on whether the trapezoidal or Simpson's rule is used, the POD modes found in this way have either $\mathcal{O}(\Delta x^2)$ or $\mathcal{O}(\Delta x^4)$ error, respectively.

4.2 Discretization: Method of snapshots

The purpose of the method of snapshots is to recast the problem by finding the eigenvalues of an $N_T \times N_T$ matrix instead of an $n_g \times n_g$ one. For this reason, the method is best suited for problems where the number of grid points is much greater than the number of time steps $n_g >> N_T$, i.e. three dimensional domains over relatively short simulation periods. This situation is rather common because in three dimensions the number of grid points scales like $\mathcal{O}(n^3)$, yielding many grid points prima facia and also increasing the complexity of each time step calculation.

The method of snapshots begins by letting

$$c_i = \int_{\Omega_x} u^*_{(i)}(x')\varphi(x')dx'. \tag{22}$$

We can rewrite (16) as

$$\frac{1}{N_t}\sum_{j=1}^{N_T} c_j u_{(j)}(x) = \lambda\varphi(x), \tag{23}$$

which implies that we express the eigenfunctions as a linear combination of the time steps or "snapshots." Next we multiply each side by $u^*_{(i)}$ and integrate to get

$$\frac{1}{N_t}\sum_{j=1}^{N_T} c_j \int_{\Omega_x} u^*_{(i)}(x)u_{(j)}(x)dx = \lambda\int_{\Omega_x} u^*_{(i)}(x)\varphi(x)dx. \tag{24}$$

If we assume that $u^*_{(i)}$ and $u_{(j)}(x)$ are linearly independent and we define

$$a_{ij} = \int_{\Omega_x} u^*_{(i)}(x)u_{(j)}(x)dx. \tag{25}$$

Then we have the $N_T \times N_T$ eigenvalue problem in matrix form of

$$\begin{bmatrix} a_{11} & ... & a_{1N_T} \\ ... & & ... \\ a_{N_T} & ... & a_{N_T N_T} \end{bmatrix} \begin{bmatrix} c_1 \\ ... \\ c_{N_T} \end{bmatrix} = \lambda \begin{bmatrix} c_1 \\ ... \\ c_{N_T} \end{bmatrix}. \tag{26}$$

In which the eigenvectors $\boldsymbol{c}^n = [c_1^n ... c_{N_T}^n]^T$ with eigenvalues $\lambda^{(n)}$. To reconstruct the n-th eigenfunction of the original problem, write

$$\varphi^{(n)}(x) = \frac{1}{\lambda^{(n)N_T}} \sum_{j=1}^{N_T} c_j^n u_{(j)}(x). \tag{27}$$

For more details and further calculation steps please refer to [1, 4].

5 Symmetry Considerations

Symmetry can enter a fluid system through inherent geometry or assumptions and simplifications. Either way, due to round-off, discretization, systematic, random, or other errors these symmetries are not automatically preserved in the POD. Leaving these symmetries untreated can affect the degeneracy of bifurcation points and even lead to multiple solution branches. For example, suppose that

$$\dot{\boldsymbol{a}} = f(\boldsymbol{a}; \mu), \tag{28}$$

with $\mathbf{a} \in \mathbb{R}^n$ and $\mu \in \mathbb{R}^m$ represent dependent variables and system parameters, respectively. Let G describe a linear group acting on the dependent variables. We say that if

$$f(\gamma \mathbf{a}; \mu) = \gamma f(\boldsymbol{a}; \mu) \tag{29}$$

for all $\gamma \in G$, then (28) is *equivariant* with respect to the group G. This is equivalent to the statement, if $\mathbf{a}(t)$ is a solution to (28) then so is $\gamma \mathbf{a}(t), \forall \in G$. Assuming commutation of γ and the time derivative as well as equivariance, we can see

$$\dot{\boldsymbol{a}} = f(\boldsymbol{a}; \mu) \tag{30}$$
$$\gamma \dot{\boldsymbol{a}} = \gamma f(\boldsymbol{a}; \mu) \tag{31}$$
$$\gamma \dot{\boldsymbol{a}} = f(\gamma \boldsymbol{a}; \mu) \tag{32}$$
$$\frac{d(\gamma \mathbf{a})}{dt} = f(\gamma \boldsymbol{a}; \mu). \tag{33}$$

Therefore $\mathbf{a}$ and $\gamma \mathbf{a}$ are both solutions of f. That example was merely to highlight the importance of considering symmetry when working with the POD. Now, let's consider a specific, common example of symmetry in fluid dynamics called homogeneity. When the averaged two point correlation $R(x, x') = R(x - x')$, i.e. R only depends on the difference of the two coordinates, it is called homogeneous or translation invariant. Homogeneity occurs in spatially unbounded systems as well as systems with period boundary conditions. In either case, the Fourier modes correspond exactly to the POD modes. Consider the series

$$R(x - x') = \sum c_k e^{2\pi i k(x - x')}. \tag{34}$$

We simply solve 15 by substituting the unique representation

$$R(x, x') = \sum c_k e^{2\pi i k x} e^{2\pi i k x'}, \tag{35}$$

implying that $\{e^{2\pi i k x}\}$ are exactly the eigenfunctions with eigenvalues c_k. Going in the other direction, if the eigenfunctions are the Fourier modes, we are allowed to define R using (35) which implies (34). In other words, homogeneity completely determines the form of the empirical eigenfunctions. Nevertheless, we use the data to determine the Fourier spectrum according to the POD.

6 Conclusion

In this paper we introduced the very basics of turbulence and the NSE as a motivation for studying the POD. We discussed value of the POD in that it delivers a basis which, with early truncation, is most efficient at capturing some value of interest. In the case of turbulence where u denotes streamwise velocity, that value of interest was the kinetic energy. We proved the optimality of the POD modes in a general Hilbert space and defined them using an eigenvalue problem. Next we applied the general POD to a finite dimensional space where it is known as PCA. Here we included an original example as well as another from computer vision. After that, we discussed the POD in the world of fluid dynamics in which we showed multiple discretization methods. Finally we mentioned that symmetry must be considered explicitly and gave an example where the form of the POD modes is determined by said symmetry.

One final remark, both [1, 4] make a point to mention that the POD modes are not exactly a complete basis of $L^2(\Omega)$ because they do not include the kernel of the operator $\mathcal{R}$. We simply note that this is not a necessary consideration since we are interested in physical modes which are included in the POD. For further discussion we refer you to the sources previously mentioned.

References

[1] Troy R. Smith, Jeff Moehils, and Philip Holmes. Low-dimensional modelling of turbulence using the proper orthogonal decomposition: A tutorial. *Nonlinear Dynamics*, 41(1):275–307, Aug 2005.

[2] H. Tennekes and John L. Lumley. *A First Course in Turbulence*. MIT Press, Cambridge, MA, 2014.

[3] Garry L. Brown and Anatol Roshko. On density effects and large structure in turbulent mixing layers. *Journal of Fluid Mechanics*, 64:775–816, 1974.

[4] Philip Holmes, John L Lumley, Gahl Berkooz, and Clarence W Rowley. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge university press, 2012.

[5] Ashish Rana. Journey from principle component analysis to autoencoders. `https://medium.com/analytics-vidhya/journey-from-principle-component-analysis-to-autoencoders-e60d066f191a`. Accessed: 13.11.2018.