YIDDISH

Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308

Highlighting the main differences between neural networks and linear stochastic estimates The cause of reversals in fluid flow

V Krishna Veni [1], I Kalpana[2],
Assistant Professor [1,2],
Mail Id : krishnaveni.v@visvodayata.ac.in, Kalpana.i@visvodayata.ac.in,
Department of Mathematics,
PBR VISVODAYA INSTITUTE OF TECHNOLOGY AND SCIENCE,KAVALI.

Abstract

Neural networks and linear stochastic estimation (LSE) are widely employed as effective methods for fuid-fow regressions. Two classical fuid-fow issues exist: (1) the state estimation in a turbulent channel fow from the wall features, and (2) the estimate of high-order proper orthogonal decomposition coefficients from low-order counterparts for a two-dimensional cylindrical fow. Within the framework of these two issues, we examine their underlying distinctions. In the first problem, LSE is contrasted with a multi-layer perceptron (MLP). In the channel fow example, a nonlinear model called the convolutional neural network (CNN) is employed to manage high-dimensional flows. Because of the nonlinear activation functions, the nonlinear NNs perform better than the linear techniques in both scenarios. Error-curve studies are also used to analyze the response of weights in models and the estimate error. Our study aims to visualize the error-curve resistance against noise while emphasizing the key distinctions between the covered tools for fuid-fow regressions. NNs have shown their enormous promise in data estimation, control, reduced-order modeling, and turbulence modeling at the fundamental level 3–5. Despite the present excitement for the use of NNs, we are forced to rely on linear theories for the time being because of their transparency and generalizability.

INTRODUCTION

If we can learn from the interaction between linear theories and neural networks, we can design more sophisticated NN-based methods for fuidfow analysis. The connection between NNs and linear approaches is of special significance here. Low-dimensionalization was achieved using an autoencoder (AE) in Milano and Koumoutsakos9's9 research. AE based on multi-layer perceptron (MLP) with linear activation functions is comparable to proper orthogonal decomposition (POD)10, according to Tey's research. Murata et al.11 studied the similarity between an AE based on a convolutional neural network (CNN) and a POD in more recent research. NNs' nonlinear activation functions were shown to have significant strength. Nair & Goza12 recently examined the MLP, Gappy POD, and linear stochastic estimation (LSE) for the POD coeffcient estimate from local sensor measurements of the laminar wake of a fat plate using local sensor data. LSE, MLP, and CNN are used to compare the abilities of linear methods and NNs by comparing two canonical fluid fow regression problems whose complexities are different; 1. estimation of high-order POD coeffcients from low-order counterparts of a fow around a cylinder13, and 2. state estimation from information on the wall in a turbulent channel fow14. By concentrating on the influence of biases in NNs, optimization approaches and resilience against noisy inputs, we seek crucial features for their estimates.

Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308

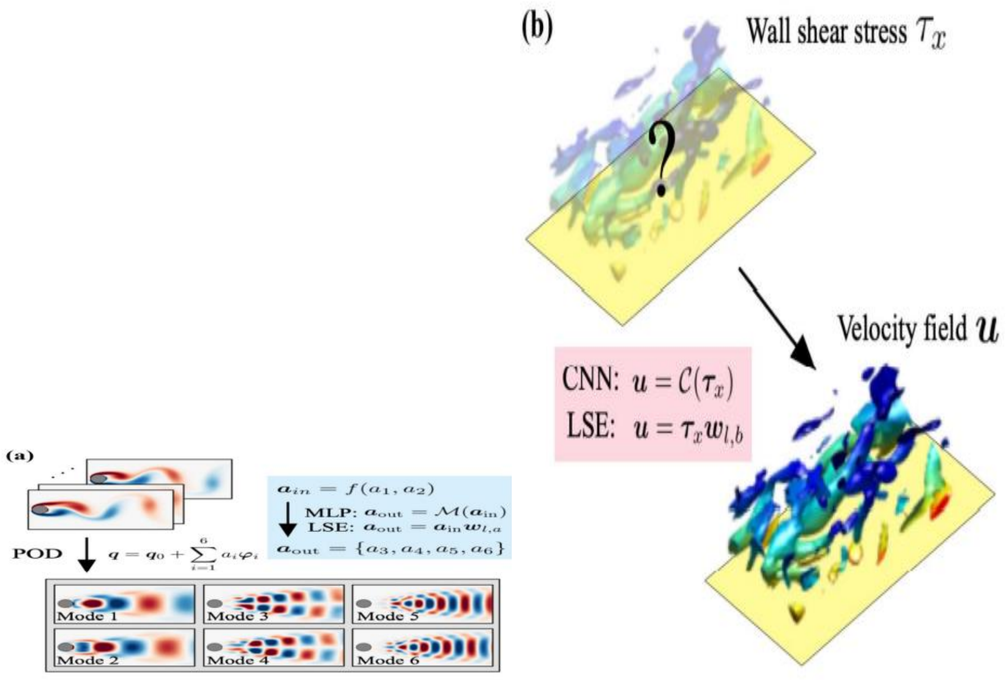

(a)
POD
$q = q_0 + \sum_{i=1}^{6} a_i \varphi_i$
$a_{in} = f(a_1, a_2)$
MLP: $a_{out} = \mathcal{M}(a_{in})$
LSE: $a_{out} = a_{in} w_{l,a}$
$a_{out} = \{a_3, a_4, a_5, a_6\}$
Mode 1
Mode 2
Mode 3
Mode 4
Mode 5
Mode 6
(b)
Wall shear stress τ_x
?
Velocity field u
CNN: $u = \mathcal{C}(\tau_x)$
LSE: $u = \tau_x w_{l,b}$

Figure 1 depicts the regressions used in this work for covered fuidfow regressions. Estimation of a flock of birds around a cylinder using POD coeffcients A nonlinear activation function is used once biases b have been introduced,

$$q_i^{(l)} = \phi\left(\sum_j W_{ij}^{(l)} q_j^{(l-1)} + b_j^{(l)}\right),$$

allowing for state estimation in a turbulent channel fow.

when the layer index l is used. Back-propagation18 is used to optimise the weights of all edges Wij to minimise a loss function. E. While the number of output nodes is four in the current MLP, there are four hidden units (3rd to 6th POD modes). Sect. 3.1 provides information on the various circumstances that are taken into account when determining the number of input nodes. It is possible to express the issue in terms of weights wm (representation of both W and b) within the MLP as follows: aout = a3, a4..., a5, a6, and wm represents both W and b.

$$\boldsymbol{w}_m = \mathrm{argmin}_{\boldsymbol{w}_m} ||\boldsymbol{a}_{\mathrm{out}} - \mathcal{M}(\boldsymbol{a}_{\mathrm{in}}; \boldsymbol{w}_m)||_2.$$

L2 error norm is the loss function we utilise. Notably, the current loss function does not take penalty terms like the Lasso and Ridge penalties into account due to the difficulty and expense of setting the hyperparameter for regularisation 17.

Convolutional neural network

Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308

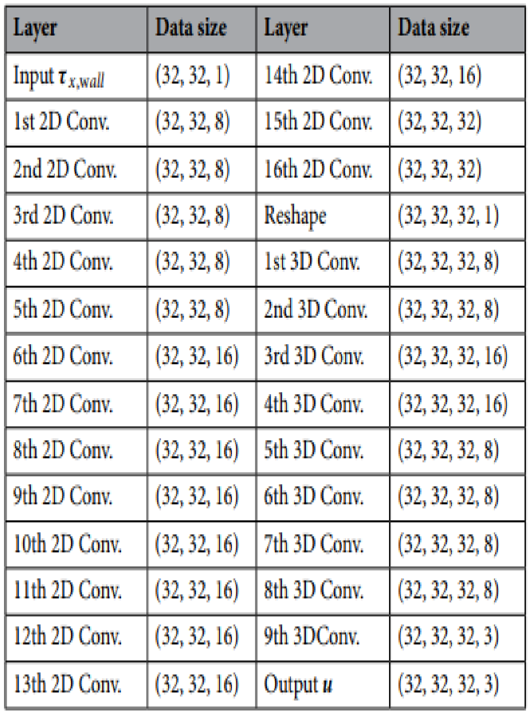
. Due to its completely linked nature, the number of edges in the MLP may expand when dealing with high-dimensional data, such as fuidfows. A convolutional neural network (CNN)19 has been largely acknowledged as a promising choice for dealing with fuidfow issues. 3,20. This work utilises a two-and three-dimensional CNN combination for state estimation. A flter operation is used to extract spatial information from input data in the convolutional layer, which is a basic process in CNNs.

$$q_{ijm}^{(l)} = \phi\left(b_m^{(l)} + \sum_{k=0}^{K-1} \sum_{p=0}^{H-1} \sum_{s=0}^{H-1} h_{pskm}^{(l)} q_{i+p-C\,j+s-C\,k}^{(l-1)} \right),$$

A convolution layer has K flters, and the bias is b (l) m = floor(H/2). For two- or three-dimensional fowfelds, this flter technique provides efficient data management. Similar to Eq., the three-dimensional convolution of the CNN is conducted in Eq (3). The font size H has been set to 5 in the current document. There are two types of activation functions used in the CNN: the fltering activation function and the MLP activation function (Eq. (1)). In order to minimise a loss function, back-propagation is used in the optimization of h filters. Three-dimensional turbulent state u is generated by combining two-dimensional and three-dimensional CNNs (details will be provided in Sect. 3.2). In order to solve the weight optimization issue for the CNN C, we may write it as

$$\boldsymbol{w}_c = \mathrm{argmin}_{\boldsymbol{w}_c} ||\boldsymbol{u} - \mathscr{C}\ (\boldsymbol{\tau}_{x,\mathrm{wall}}; \boldsymbol{w}_c)||_2.$$

The purpose of this research is to compare the LSE and NN models. To provide a fair comparison, it is best to use the same number of weights for LSE and NNs. Singular value decomposition (SVD) is used to align the number of weights in the CNNs and the LSE in order to do this.



Layer	Data size	Layer	Data size
Input $\tau_{x,wall}$	(32, 32, 1)	14th 2D Conv.	(32, 32, 16)
1st 2D Conv.	(32, 32, 8)	15th 2D Conv.	(32, 32, 32)
2nd 2D Conv.	(32, 32, 8)	16th 2D Conv.	(32, 32, 32)
3rd 2D Conv.	(32, 32, 8)	Reshape	(32, 32, 32, 1)
4th 2D Conv.	(32, 32, 8)	1st 3D Conv.	(32, 32, 32, 8)
5th 2D Conv.	(32, 32, 8)	2nd 3D Conv.	(32, 32, 32, 8)
6th 2D Conv.	(32, 32, 16)	3rd 3D Conv.	(32, 32, 32, 16)
7th 2D Conv.	(32, 32, 16)	4th 3D Conv.	(32, 32, 32, 16)
8th 2D Conv.	(32, 32, 16)	5th 3D Conv.	(32, 32, 32, 8)
9th 2D Conv.	(32, 32, 16)	6th 3D Conv.	(32, 32, 32, 8)
10th 2D Conv.	(32, 32, 16)	7th 3D Conv.	(32, 32, 32, 8)
11th 2D Conv.	(32, 32, 16)	8th 3D Conv.	(32, 32, 32, 8)
12th 2D Conv.	(32, 32, 16)	9th 3DConv.	(32, 32, 32, 3)
13th 2D Conv.	(32, 32, 16)	Output $\boldsymbol{u}$	(32, 32, 32, 3)

Table 1. Structure of 2D-3D CNN C for turbulent channel fow example.

In the next part, we'll go through the LSE's weight distribution. It is possible to describe the operation as wl = UVT, where U is a singular vector and VT is an output vector with the same rank as the input vector, and (nrank) is a diagonal matrix with singular values on the diagonal. According to our preliminary test, the number of LSE weights based on SVD reduction nwLSE,SVD is 197,632 (= ninputnrank = 1024 193), which is the current number of LSE weights. As a result of this SVD-based weight reduction, we now have the 197,000 weights needed to compute the CNN parameters. According to Table 1, the 196 608 weights in our CNN have the following parameters: Estimation using linear stochastic models. Linear stochastic estimation (LSE)14,21 is used as a comparison to the neural networks (NNs). Q = Pwl, where ndata represents the number of training snapshots, ninput represents the number of input attributes, and noutput represents the number of output attributes. In this study, we express target data Q wlRninputnoutput as a linear map with respect to input data P wlRndataninput, where Q = Pwl. The linear map wl may be created by minimization in a manner similar to that used for NN optimization.

$$\boldsymbol{w}_l = \mathrm{argmin}_{\boldsymbol{w}_l} \parallel \boldsymbol{Q} - \boldsymbol{w}_l \boldsymbol{P} \parallel_2 = (\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P})^{-1}\boldsymbol{P}^{\mathrm{T}}\boldsymbol{Q}.$$

For the sake of a fair comparison with NNs, penalization terms are not taken into account in the current loss function. It's important to note, however, that the LSE is analytically optimised by solving Equation (5), whereas the NNs are numerically optimised by back-propagation. Because of the difference in optimization methodologies, we may also compare NNs and LSEs. It is therefore possible to apply the optimal weights wl to test data.

Results

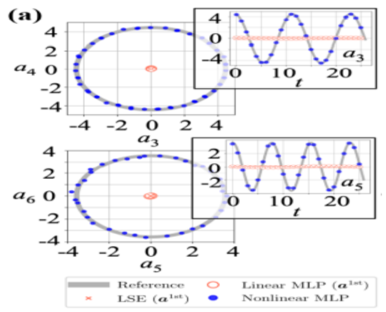
Example 1: The POD coefficient of a two-dimensional cylinder wake at ReD = 100. First, as shown in Fig. 1a, we aim at estimating the two-dimensional high-order POD coefcients of an initial cylinder wake at ReD = 100 from knowledge about its low-order counterparts, ain, such that F1 designates the model for this purpose. We next use this model to estimate aout = ain. LSE and MLP serve as model F1 in this experimentation. A two-dimensional direct numerical simulation is used to create flow snapshots (DNS). Incompressible Navier–Stokes equations are the governing equations.

$$\nabla \cdot \boldsymbol{u} = 0,\ \ \partial_t \boldsymbol{u} + \nabla \cdot (\boldsymbol{u}\boldsymbol{u}) = -\nabla p + Re_D^{-1}\nabla^2 \boldsymbol{u},$$

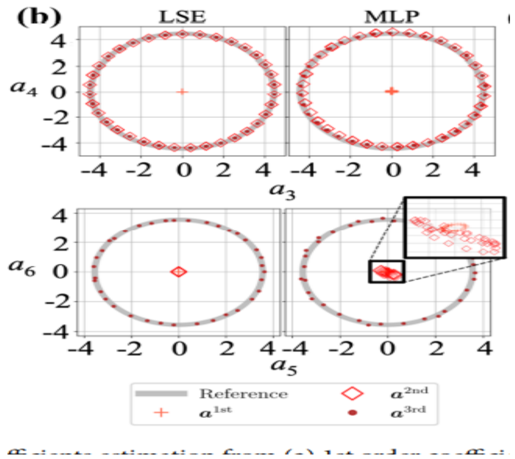
The pressure is represented by p, and the velocity vector by u. It is non-dimensionalized using the fuid density, the free-flow velocity, and the diameter of the cylinder. There are 25.6 and 20.0 units in the computational domain, respectively, and the cylinder's centre lies at (x, y) = (9, 0). Using an immersed boundary method22, the no-slip boundary condition is applied to the cylinder's surface, with a grid spacing of 0.025 and a time step of 2.5 10-3. The DNS uses (Nx, Ny) = (Nx, Ny) grid points (1024, 800). (Nx, Ny) = (Nx, Ny) = (Nx, Ny) for the POD, which extracts the vorticity field around the cylindrical surface (384, 192). For each POD mode, we take the average of the fow field's temporal average and divide it by one to get the POD basis. Our current MLP and LSE are trained using 5000 snapshots. We do not separate the MLP training data into training and validation for the sake of comparison with LSE. Additional 5000 images are taken into account throughout the analysis as well. Here, we compare the LSE with the linear MLP and the nonlinear MLP using the ReLU activation function23. TeReLU has a reputation for being a strong contender in the fight against disappearing gradients. Using just a1st with the LSE and linear MLP, we consider three patterns of input ain = F (a1, a2): a1st = A1, A2, A1A2, A2 1, A2 2, A2 1a2 2, A3 1, and

YIDDISH

Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308



A3 2 with the other two patterns.



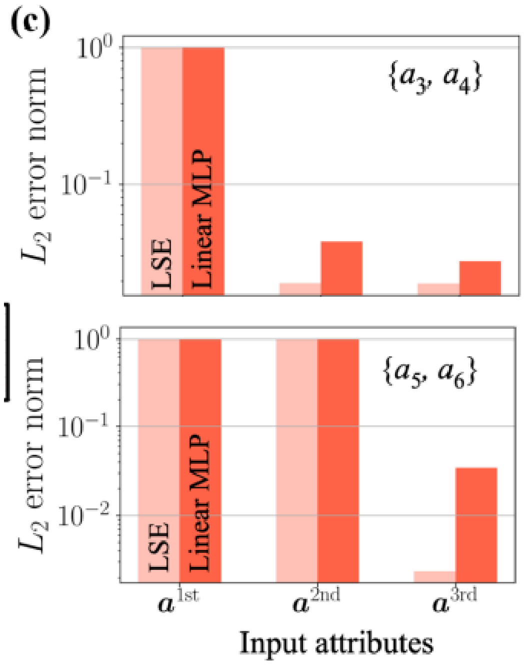


FIGURE 2: PREDICTION OF POD COEFFICIENTS FROM (A) FIRST ORDER COEFFICIENTS, (B) SECOND AND THIRD ORDER COEFFICIENTS, AND (C) THE LITTLE 2ND ORDER ERROR NORM,

YIDDISH

Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308

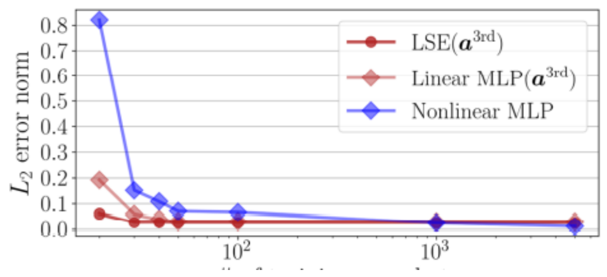


Figure 3. Dependence of the L2 error norm for the POD coefcient estimation on the number of training snapshots.

a nonlinear MLP In this study, we investigate if the nonlinear function in MLP works to capture such nonlinear interactions, and whether the lin ear models may also be used if the essential combinations of nonlinear terms are given as input. As illustrated in Fig. 2a, the first-order coefcients a1st = a1, a2 may be used to estimate aout = a3st, which is equal to F1(a1st). For both coefcient maps, the nonlinear MLP has a clear advantage over the LSE and the linear MLP. Linear MLP has an L2 error norm of 1.00; non-linear MLP has an L2 error norm of 0.0119; and LSE has an L2 error norm of 1.00. According to this, estimates may benefit from consideration of the nonlinear activation function. As seen in Figs. 2b and 2c, this nonlinearity may be restored even if we solely utilise linear techniques by providing the right inputs, ain=a2nd and a3rd. In order to get an accurate estimate of a3, a4, we can use input up to the second order term a2nd with the LSE and the linear MLP, but for a5, a6 we need the third order term a3rd. According to Loiseau et al.13's findings, this tendency is similar to the one seen here. As demonstrated in Fig. 2c, the LSE beats the linear MLP with high-order coefcient inputs. However, as we'll see later, there's a significant difference between the two in terms of noise resilience. The LSE and MLP are then compared in terms of the amount of training data that is readily available. Figure 3 shows the relationship between the L2 error and the amount of training snapshots. For the linear models, we use the third-order coeffcientsain = a3rd and the first-order coeffcients for the nonlinear MLP, as input for the models. When the number of training snapshots is minimal, Te LSE displays its advantage over the MLP models. Because the MLP has a greater degree of flexibility than the LSE, this is the case. In terms of noise resilience, the MLP clearly outperforms the LSE, revealing the linear MLP's inherent advantages over the LSE. Consider the Gaussian white noise denoted by the signal-to-noise (SNR), which is defined as the ratio of the variances of input data and background noise, 2 data/2 noise. In Fig. 4, the results for noisy inputs are summarised. We employ the LSE and the linear MLP with ain = a3rd as our linear models. As a point of reference,

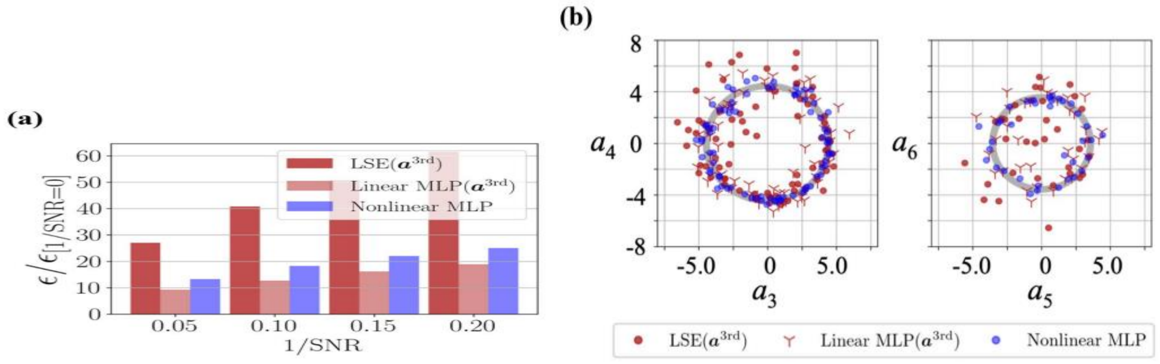


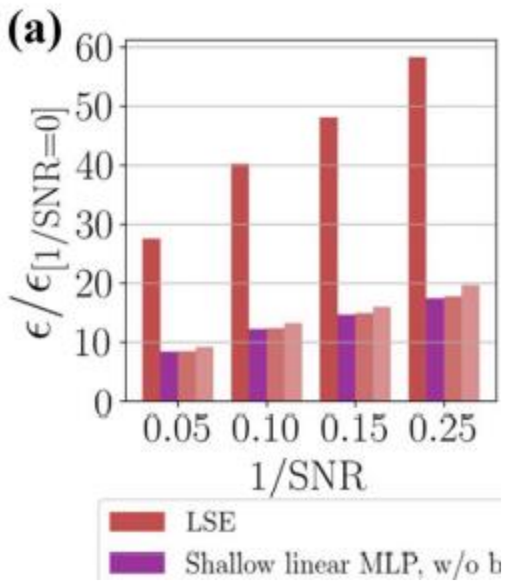
Figure 4: POD coefcients estimation is robust to noisy input. (a) The relationship between the L2 error norm /[1/SNR=0increasing]'s ratio and noise magnitude. For linear models, ain = a3rd, and for nonlinear MLP models, with ain = a1st, we get an aout with 1/SNR = 0.1. The nonlinear MLP is likewise monitored with ain = a1st. Compared to the covered MLPs, LSE's reaction is far more precise. Biases in MLPs (as described in Eq. (1)) and differences in optimization approaches are to blame for this problem, which may be summarised as follows: In this section, we'll look for the component that has the most impact on noise robustness. The LSE and three kinds of MLPs are taken into account for the study of the primary contribution as previously described. 1. LSE model: the same LSE model as used

Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308

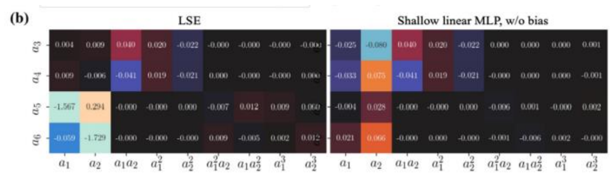
earlier. 2. LSE model: This is the same linear MLP model that was used before, but with a bias of M1. Third, the biases are removed from the linear MLP model M1 in order to examine their influence on the bias. When the number of weights is aligned with the LSE, it may be used to evaluate the differences between optimization strategies. Figure 5a shows the relationship between the L2 error norm and the noise level of each model. The covered MLP models have essentially little difference in terms of performance. These results imply that noise resilience is not much influenced by the bias or the number of layers. The shallow linear MLP is still more resilient than the LSE, despite the fact that the model structure is same. Figure 5b shows the weights in the LSE and the shallow linear MLP to illustrate this idea. Second-order term input weights are optimised to the same values as those for the first-order term input, although the first-order term input weights differ significantly. Tis is a result of a difference in the optimization techniques. Figure 5d depicts an error surface for the inputs of a1, 1 and a3 1 with the output of a5, as illustrated. The output a5 is one of the LSE's most sensitive components to noise, as seen in Fig. 5c. This is why this input-output combination was chosen. To understand why the optimal solutions differ, it may be necessary to look at the techniques employed to optimise them. The presence of noise significantly alters the error-surface form of the LSE, but only marginally alters the shape of the MLP. The MSSIM (mean structural similarity index measure)24 may be used to quantify this difference. If Er RMN, we apply this to the elevation of each error surface (i.e., without Er, and with Er, and with noise En, where M, N are the number of samples on each weight axis). MSSIM may be stated numerically as

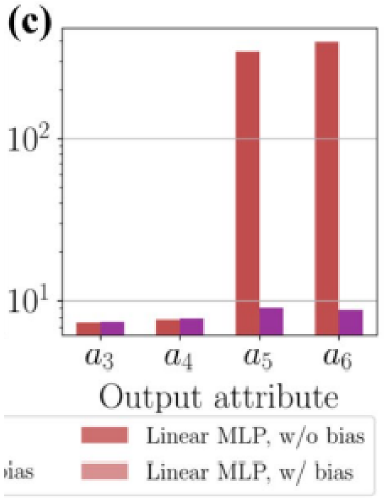
$$\mathrm{MSSIM}(E_r, E_n) = \frac{1}{M'N'}\sum_{i=1}^{M'}\sum_{j=1}^{N'}\mathrm{SSIM}(e_{r,ij}, e_{n,ij}), \quad \mathrm{SSIM}(e_r, e_n) = \frac{(2\mu_r\mu_n + C_1)(2\sigma_{rn} + C_2)}{(\mu_r^2 + \mu_n^2 + C_1)(\sigma_r^2 + \sigma_n^2 + C_2)}.$$

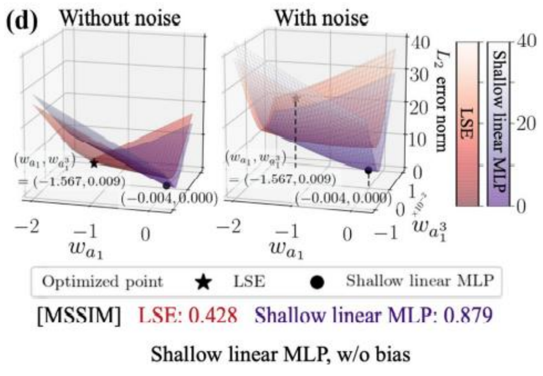
In order to compare two photos, Tis uses the mean and standard deviation of the two images to determine how similar they are. If you want to get MSSIM, you need to take the average across the whole picture of M′ and N′, where M′ is equal to M + m and N + n + 1, in order to calculate the SSIM in that tiny window of two photos. In accordance with Wang et al.24, we used C1, C2=0.16, 1.44 as the constant values in Eq. (7). Error surfaces are significantly deformed in LSE, as seen in Fig. 5d, where MSSIM for shallow linear MLPs (0.879) and LSE (0.428) are shown. The LSE's optimal point in Fig. 5d's error space is raised vertically as a result of the substantial distortion of the error surface. According to this, the LSE weights derived analytically ensure the global optimum solution over the training data, but this solution may not be optimal from the standpoint of noise resilience. Although it isn't an exact global best across the training data, the MLP weights are a numerical solution achieved using back-propagation, which makes it a noise-resistant solution. The shallow linear MLP (without bias) and LSE both benefit from adding SNR = 0.05 noisy data to the training data, thus we do so (as described in Fig. 6) to account for the LSE's properties. Fig. 6a illustrates this.



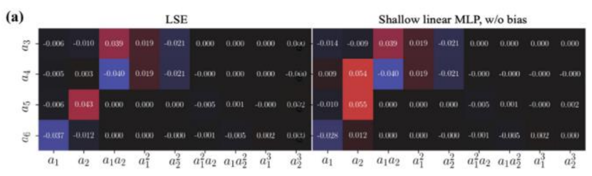
Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308





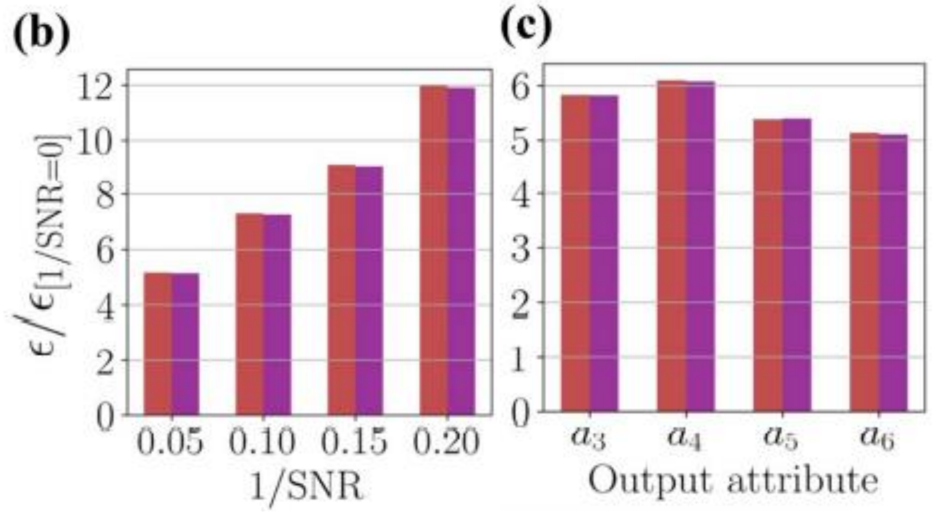


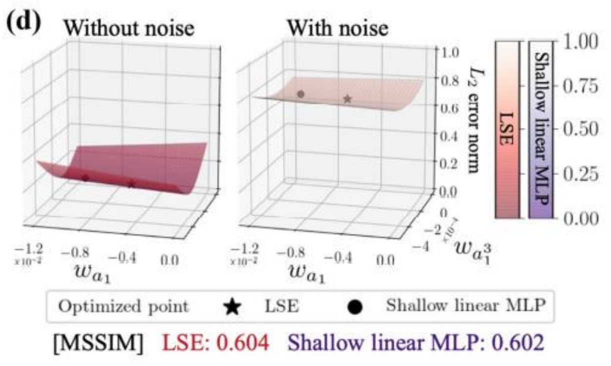
Comparison of LSE and MLP based on bias and optimization approach shown in Figure 5. To increase the L2 error norm from the initial error [1/SNR=0] without the noisy input, we may use the following strategy: (b) LSE and shallow linear MLP weight values. [1/SNR=0] and the output POD coefcients depend on the growing ratio [1/SNR=0] of L2 error norms. d) The error surface surrounding the optimised location is shown graphically.



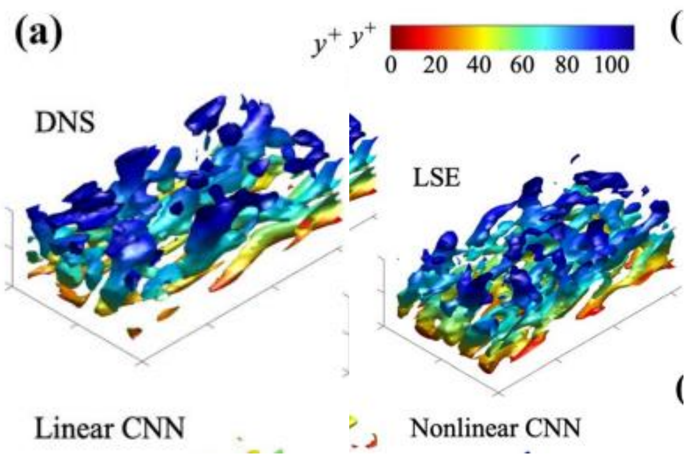
YIDDISH

Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308



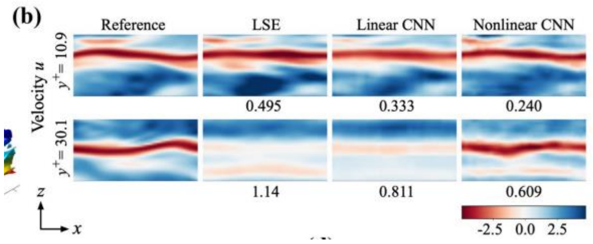


Using the noisy training data, we can see a comparison between the linear MLP and the LSE in Figure 6. (a) LSE and shallow linear MLP weight values. For example, instead of using the noisy input, we may boost the L2 error norm by a ratio of /[1/SNR=0]. [1/SNR=0] and the output POD coefcients depend on the growing ratio [1/SNR=0] of L2 error norms. d) The error surface surrounding the optimised location is shown graphically.



YIDDISH

Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308



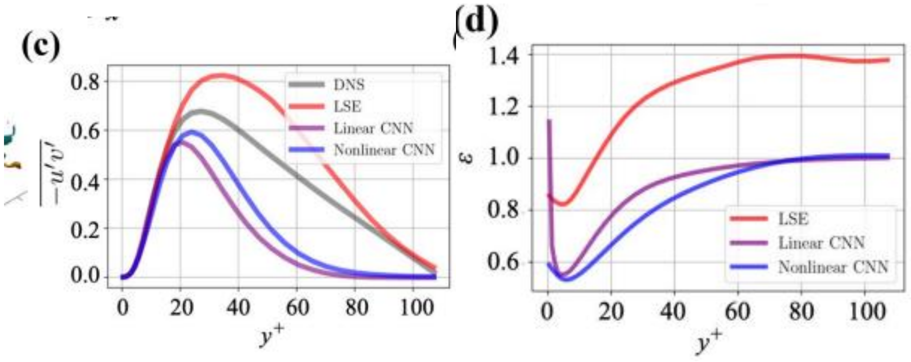


Figure 7 illustrates how streamwise wall-shear stress may be used to estimate turbulent channel flow. Q+ criteria isosurfaces (Q+ = 0.005). Assuming that (a) is the case, then (b) is the case as well. For each velocity attribute, the Te values beneath the contours indicate the L2 error norm for the corresponding Te. c) Reynolds shear stress u′ v′. (d) Ensemble L2 error norm dependence on y position for three velocity components.

While there was a difference in weight values in Figure 5, there is almost no difference here. Figs. 6b–d show comparable results for a variety of other studies. According to Tese, training data may be made more robust by include noise. As you can see in Figure 6b, the growing ratio of /[1/SNR=0] = 5 is not nearly huge since the initial error [1/SNR=0] is tiny ([1/SNR=0] = 4.49 x 102 for the LSE, whereas the shallow linear MLP has a little larger error of 4.52 x 102 x 102).

Example 2: velocity feld in a minimal turbulent channel fow at Reτ = 110. We next compare the LSE and NNs for a more difficult issue, which requires a more thorough analysis. According to the streamwise wall shear stress input x in the least turbulent channel fow at Re = 110, it is possible to estimate the velocity field (u) of the channel. Model F2 is an example of a model used in Example 2 and is denoted by the symbol F2. LSE and CNN are the models F2 that we employ. Training data is created by numerically solving the incompressible Navier–Stokes equations with a three-dimensional DNS.

$$\nabla \cdot \boldsymbol{u} = 0, \quad \partial_t \boldsymbol{u} + \nabla \cdot (\boldsymbol{u}\boldsymbol{u}) = -\nabla p + Re_\tau^{-1} \nabla^2 \boldsymbol{u},$$

where u and p denote the velocity and pressure velocities, respectively25,26. The channel half-width and the friction velocity u are non-dimensionalized. Lx, Ly and Lz are the computing domains with the number of grid points of (Nx, Ny and Nz) = 0.5. (32, 64, 32). There are two distinct grids: one is symmetrical in the x and z dimensions while the other is asymmetrical in the y direction. As the subscript + specifies the number of wall units, t+ = 0.0385. The models are trained using 10,000 pictures. We do not separate the training data into training and validation in order to get comparable results to LSE. In addition, we create 2700 new snapshots for the evaluation. Model-estimated flow channels are shown in Fig. 7a. The amount of weights inside the CNN and LSE are almost identical, as mentioned

Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308

before in Section 2.2. A CNN with a different filter operation from that of a fully-connected MLP is also taken into account to see whether the same results can be obtained as for the linear MLP in the cylinder example using a linear CNN (i.e., the same CNN structure, but with a linear activation function). The second invariant of the velocity gradient tensor Q+ of 0.005 is used to illustrate the estimated felds. However, it should be noted that the LSE feld only offers turbulent-like structure that is distinct from that seen in the DNS, whilst the DNS feld displays qualitatively comparable behaviour to that of the reference LSE data. Figure 7b shows the x z sectional streamwise velocity distributions to study this site. However, the nonlinear CNN is superior to both the linear CNN and LSE in terms of accuracy, which is surprising given the discovery of Q isosurfaces. At y+ = 30.1, the advantages of the nonlinear technique are most apparent. Fig. 7c also shows the time-averaged Reynolds shear stress u′ v′. Despite its high error level, the form of the LSE curve seems to be very consistent (albeit exaggerated). A decent reconstruction of LSE in Fig. 7a suggests that the LSE model's predicted fowfeld approximates the DNS data in a time-ensemble sense, even if it does not match for each instantaneous feld. In Fig. 7d, we examine the estimation's relationship to the y-coordinate. Due to the paucity of data in the vicinity of the wall, it is difficult to determine the velocity field away from the wall.14,27 of

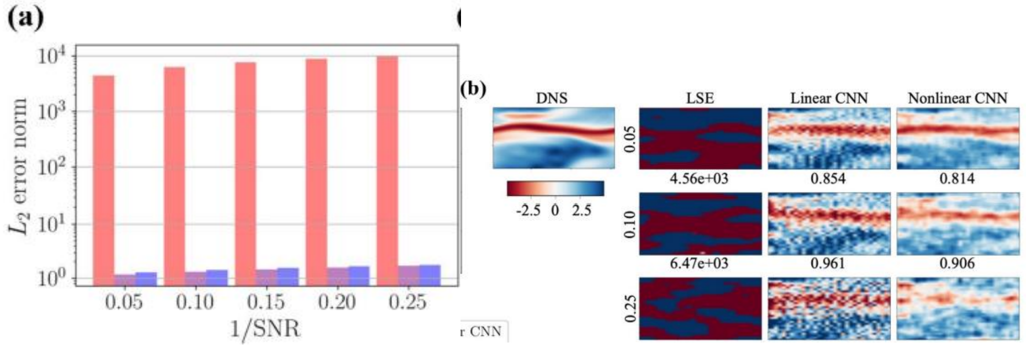


Figure 8 illustrates the robustness of the system in the case of turbulence. The (a) /SNR[1/SNR=0] on the noise magnitude. Assumed streamwise velocity fluctuation at y+ = 10.9 (b) Contours. L2 error norm across three velocity components is shown by the Te values shown below. Percentage of SNR (SNR) on the side of the contour is shown by the numbers

Conclusions

The standard fuidfow regression issues were used to study the fundamental differences between NNs and linear methods: A fow is studied in two ways: first, to determine its POD coefficients, then, to determine its state using wall measurements in a turbulent channel fow. SSE, MLP, and a convolutional neural network were evaluated against LSE and SSE with a multi-layer perceptron (CNN). Nonlinear function efficacy may be found in both regression tasks. As a result, we discovered that a linear model might be used as a substitute for a nonlinear model by providing an adequate mix of inputs. A similar discovery has been made in numerous prior studies28–30, thus we may assume that the combination of nonlinear activation functions and adequate inputs can boost the prediction capabilities of the model further. It was also found that linear NNs were more resistant to noise than the LSE, and that visualising the error surface helped to uncover why. Noise resilience may be attributed to the difference in optimization strategies, according to error surfaces. The learning process of nonlinear neural networks (NNs) may be unstable depending on the issue setting since it is based on a gradient approach. Despite this, we discovered that NNs have a number of advantages over linear neural networks. We may not be able to get to a tolerable weight valley, particularly if the issue is multimodal and training data is scarce. In this regard, the LSE may provide us a theoretically sound answer. These features may be combined for further enhancement of NN learning pipelines, such as transfer learning32,33 so that the NN may be launched from a reasonable solution and achieve sufficient noise resilience, as shown in our study. For this reason, we may include a loss function connected with physics-based presuppositions in order to minimise the weights used in both neural networks and lasso-style estimation (LSE). Incorporating a physical loss function with both systems might be a significant route toward practical use. 34–3

References

1. Brunton, S. L., Noack, B. R. &Koumoutsakos, P. Machine learning for fuid mechanics. Annu. Rev. Fluid Mech. 52, 477–508 (2020).

2. Fukami, K., Fukagata, K. & Taira, K. Assessment of supervised machine learning for fuidfows. Teor. Comp. Fluid Dyn. 34, 497–519 (2020).

3. Fukami, K., Fukagata, K. & Taira, K. Super-resolution reconstruction of turbulent fows with machine learning. J. Fluid Mech. 870, 106–120 (2019).

4. Fukami, K., Fukagata, K. & Taira, K. Machine-learning-based spatio-temporal super resolution reconstruction of turbulent fows. J. Fluid Mech. 909, A9. https://doi.org/10.1017/jfm.2020.948 (2021).

5. Fukami, K., Maulik, K., Ramachandra, N., Fukagata, K. & Taira, K. Global feld reconstruction from sparse sensors with Voronoi tessellation-assisted deep learning. Nat. Mach. Intell. 3, 945–951 (2021).

6. Brenner, M. P., Eldredge, J. D. & Freund, J. B. Perspective on machine learning for advancing fuid mechanics. Phys. Rev. Fluids 4, 100501. https://doi.org/10.1103/PhysRevFluids.4.100501 (2019).

7. Maulik, R., Fukami, K., Ramachandra, N., Fukagata, K. & Taira, K. Probabilistic neural networks for fuidfow surrogate modeling and data recovery. Phys. Rev. Fluids 5, 104401. https://doi.org/10.1103/PhysRevFluids.5.104401 (2020).

8. Duraisamy, K., Iaccarino, G. & Xiao, H. Turbulence modeling in the age of data. Annu. Rev. Fluid. Mech. 51, 357–377 (2019).

9. Milano, M. &Koumoutsakos, P. Neural network modeling for near wall turbulent fow. J. Comput. Phys. 182, 1–26 (2002).

10. Lumley, J. L. Te structure of inhomogeneous turbulent fows. In Yaglom, A. M. &Tatarski, V. I. (eds.) Atmospheric turbulence and radio wave propagation (Nauka, 1967).

11. Murata, T., Fukami, K. &Fukagata, K. Nonlinear mode decomposition with convolutional neural networks for fuid dynamics. J. Fluid Mech. 882, A13. https://doi.org/10.1017/jfm.2019.822 (2020).

12. Nair, N. J. &Goza, A. Leveraging reduced-order models for state estimation using deep learning. J. Fluid Mech. 897. https://doi. org/10.1017/jfm.2020.409 (2020).

13. Loiseau, J.-C., Brunton, S. L. & Noack, B. R. From the POD-Galerkin method to sparse manifold models.https://doi.org/10.13140/ RG.2.2.27965.31201 (2018).

Vol 12 Issue 02,April 2022
ISSN NO: 0364-4308

14. Suzuki, T. & Hasegawa, Y. Estimation of turbulent channel fow at Reτ = 100 based on the wall measurement using a simple sequential approach. J. Fluid Mech. 830, 760–796 (2006).

15. Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagation errors. Nature 322, 533–536 (1986).

16. Domingos, P. A few useful things to know about machine learning. Commun. ACM 55, 78–87 (2012).

17. Nakamura, T., &Fukagata, K. Robust training approach of neural networks for fuidfow state estimations. Preprint at arXiv:2112. 02751 (2021).

18. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. Preprint at arXiv:1412.6980 (2014).

19. LeCun, Y., Bottou, L., Bengio, Y. & Hafner, P. Gradient-based learning applied to document recognition. Proc. IEEE 86, 2278–2324 (1998).

20. Morimoto, M., Fukami, K., Zhang, K., Nair, A. G. &Fukagata, K. Convolutional neural networks for fuidfow analysis: Toward efective metamodeling and low-dimensionalization. Teor. Comp. Fluid Dyn. 35, 633–658 (2021).