Proceedings of the ASME 2022 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference
IDETC/CIE2022
August 14-17, 2022, St. Louis, Missouri

DETC2022-90068

EFFECT OF OPTIMAL GEOMETRIES AND PERFORMANCE PARAMETERS ON AIRFOIL LATENT SPACE DIMENSION

Alec Van Slooten*
Dept. of Mechanical Engineering
University of Maryland
College Park, MD 20742
Email: vansloot@umd.edu

Mark Fuge
Dept. of Mechanical Engineering
University of Maryland
College Park, MD 20742
Email: fuge@umd.edu

ABSTRACT

Although learning low-dimensional airfoil manifolds can facilitate aerodynamic optimizations, the properties of these latent spaces are not well understood. This paper investigates airfoil manifolds to provide greater insight into the effects of optimized geometry and data set features on latent spaces. Specifically, we investigate if optimized geometries occupy lower dimensional manifolds than non-optimized geometries. We also examine the effect of including target optimization conditions as data set features for a range of latent space sizes. We explore these areas using the UIUC airfoil database and a subset of these airfoils optimized with CBGAN and CEBGAN models. Lower dimensional airfoil manifolds are learned using both autoencoders and principal component analysis (PCA) models. The performance of these models are also compared to each other in ranges of training sample sizes and latent dimension size using mean squared error (MSE) between the original testing samples and the reprojected data constructed from the models as a metric. The results of this study suggest that optimized geometry does not always lie in a lower dimensional latent space as the two data sets were observed to have similar intrinsic dimensionalities. This study further demonstrates that including input parameters used in airfoil coordinate generation as data set features does not necessarily decrease the latent space dimensionality.

*Corresponding Author

1 INTRODUCTION

Airfoil optimization is a established field that is commonly explored in the design of turbine blades, While much progress has been made in the past in this field, numerical optimizations are still time consuming. These optimization techniques need to be executed repeatedly to find an optimal design resulting in a process often constrained by the run time [1]. The dimensionality reduction (DR) type of unsupervised machine learning can be leveraged to help solve this problem. DR creates a smaller subspace from the original data wherein little of the explained variance is lost. Optimizations within this space typically converge much faster. Once finished, the results can be reprojected from the latent space back into the real space. This workflow is becoming increasingly common in many areas of optimization and has been used recently in everything from aerodynamic design optimization [2,3] to molecular drug design [4] and optimization of robotic motion [5].

Although this methodology has already been established, exploring this reduced subspace may still be computationally expensive. The cost of sampling the design space of high dimensional models often increases exponentially with dimensionality [6]. This phenomena, dubbed the curse of dimensionality [7], illustrates the importance of minimizing the subspace dimensionality to reduce the computational time necessary for design space exploration. To streamline this reduction, it is critical to understand the properties of low-dimensional design manifolds. With respect to airfoils, the effects of using optimal geometry and including performance parameters the geometry was optimized for are of particular interest. Were the use of optimal geometry or performance parameters to result in lower dimensional manifolds, they could be leveraged to expedite aerodynamic shape optimization. These properties are studied using PCA and autoencoder models as investigative tools to learn generalities about the manifold. The key contributions of this paper are three-fold.

First, we compared the performance of PCA and autoencoders on an optimized airfoil data set to provide insight into each model's behavior with respect to the amount of training data used and the latent space size. Performance was measured by reconstruction testing MSE values using 1-20 dimensions and 0.5% to 98% training data. Figs. 1, 2, 3, and 4 show that the autoencoders perform better in low dimensional spaces but PCA has greater stability with extremely low or high fractions of training samples.

Second, we tested the hypothesis that optimal airfoil geometries live on a lower dimensional space than non-optimized geometries. We found, surprisingly, that this is not necessarily true and show in Fig. 4 that UIUC airfoil and optimized airfoil data sets have similar intrinsic dimension.

Third, we studied the effect of including the properties that the airfoils were optimized for (target lift coefficient, Mach number, angle of attack, and Reynolds number) as features on the intrinsic dimensionality of the latent space. We found in Fig. 5 that appending these features to the optimized airfoil data set do not result in a significant change in dimensionality.

2 BACKGROUND AND RELATED WORK

The related work to the key contributions of this paper fall into roughly six categories: (1) dimensionality reduction, (2) airfoil decomposition, (3) optimized and UIUC airfoil databases, (4) dimensionality and estimation methods, (5) latent space of optimized geometry data sets, and (6) impact of feature selection on latent spaces.

2.1 Dimensionality Reduction

Dimensionality reduction is the process of deriving a set of degrees of freedom which can be used to reproduce most of the variability of a data set [8]. There are numerous methods of DR which are classified by their linearity. The most popular linear methods are PCA and metric Multi-Dimensional Scaling [9]. Commonly used nonlinear models include autoencoders, isomaps, locally-linearly embedding (LLE), non-metric Multi-Dimensional Scaling and kernal PCA [9, 10]. Although each modeling method comes with its advantages, PCA and autoencoders were selected as the DR models to keep the scope of this investigation sufficiently narrow.

PCA is a fast and computationally inexpensive method of dimensionality reduction that is used as a baseline in this paper. PCA is a basic linear model commonly used to perform a change of basis on a given data set [11]. Although the transformation is linear, many principal components can be neglected depending on the percentage of explained variance they account for. The explained variance ratio is the percentage of variance attributed by each component and is a typical metric used when determining how many principal components are necessary [12]. Besides speed and simplicity, PCA also has the advantage of axes ordered based on their representational power. This makes it easier to interpret and identify the component with the greatest explained variance [13].

An autoencoder is a neural network commonly used in unsupervised learning. They are constructed of an encoder to learn a representation for a set of data, and a decoder to reproject that representation back into the original design space. Autoencoders are more computationally expensive processes that are better able to model nonlinear data and data with recurring patterns [14, 15]. Nonlinear deep autoencoders typically reduce dimensionality better than linear methods such as PCA [16, 17]. In fact, PCA is a special case of an autoencoder using squared loss and identity activation functions [18]. Additional layers and nonlinear activation functions allow autoencoders to model more complex and nonlinear functions. Our work uses both PCA and autoencoder models to reduce data set dimensionality.

2.2 Airfoil Decomposition

Previous works in airfoil decomposition have generally used proper orthogonal decomposition (POD). This method is synonymous with PCA and Karhunen–Loève expansion [19]. POD has been successfully implemented for airfoil design optimization, reducing computational cost by approximately three orders of magnitude for a two-dimensional invicid flow calculation [20]. It has also been applied to micro air vehicle wing CFD simulation data [21] and demonstrated to be effective for reconstructing flowfields from incomplete aerodynamic data sets [22].

More recently, models such as generative adversarial networks and autoencoders have been used in airfoil decomposition. A modified generative adversarial network dubbed Bézier-GAN has been applied to learn an interpretable low-dimensional space that encodes major shape variation of aerodynamic designs [23]. Autoencoders were applied to unsteady flows around a two-dimensional airfoil [24], as well as inverse design of airfoil shapes [25]. These deep learning techniques aim to improve upon the limitations of linear methods and model complex and nonlinear data with greater accuracy.

2.3 Aeronautical Databases

To test the effects of geometry and additional parameters on the latent space for each model, non-optimized and optimized databases are necessary. The UIUC airfoil database was selected as the non-optimized database for its content and previous use in dimensionality reduction studies. The intrinsic complexity and dimensionality of the design space has been measured via shape reconstruction error, pairwise distance preservation, and captured semantic attributes. Using MLE estimator, TwoNN estimator, and the Geo MLE estimator, the dimensionality of the UIUC data set was found to be between 2 and 6 with a greater probability around 4 dimensions [26]. More recently, the UIUC airfoil database was used to visualize the effect of latent dimension on airfoil manifold learning [27].

The optimized airfoil data set was previously generated using the open-source PDE analysis toolset SU2 to perform gradient-based shape optimization on airfoils from the UIUC database [28]. This data set was originally used to evaluate the ability of CBGAN and CEBGAN generative models to perform an inverse design of two-dimensional airfoils. [28]. To date, dimensionality reduction has yet to be performed on this optimized airfoil data set. Since it was constructed from optimizing UIUC database airfoils it provides a means to investigate the impact of geometry optimization on the latent space when modeled in conjunction with the UIUC airfoil database.

2.4 Airfoil Training Size and Dimensionality Estimation

Having selected the models and established the airfoil databases, determining model latent space size is the next essential step. The dimensionality of the latent space is a crucial model hyperparameter that has been shown to greatly impact the autoencoder model accuracy [29]. In previous studies on the low complexity MNIST data set, best results were achieved when the number of hidden layer nodes in the autoencoder is set around the intrinsic dimensionality of the data [16]. Although this specific trend was not observed in a greater complexity data set in the same study, it is clear that latent space size is an important modeling parameter. Intrinsic dimension also has a close correlation with the number of samples needed for learning [30]. The effects of latent space dimensionality and sample size are investigated in this paper.

A common method to estimate data set dimensionality is to perform Maximum Likelihood Estimation (MLE). This method has been applied to a range of real and simulated datasets, demonstrating its performance over other dimensionality estimators [31]. The efficacy of MLE was confirmed on the UIUC airfoil database as estimates were found to agree with brute-force hyperparameter tuning [27]. Estimators such as MLE rely on independent and identically distributed data as well as smooth and locally uniform data density. While MLE has been previously applied to the UIUC database, it is not clear that these assumptions were met. Since the optimized airfoil data set runs into this same issue, intrinsic dimensionality is determined based off model performance.

2.5 Effects of Optimized Geometry in Latent Spaces

While this investigation is specifically interested in the effect that these models have on uncovering the intrinsic dimension of non-optimized and optimized geometries, to the best of our knowledge we were not able to find prior research addressing this question. The closest field of research is in geometric parameterization for dimensionality reduction where largely studies have centered around automotive evolutionary shape optimization. Rios et al. explored the efficiency of the bottleneck layer of a point cloud autoencoder as a geometric representation for optimization [32]. In later work, aerodynamic drag and lift of five car shapes were optimized within latent spaces of PCA, K-PCA, and autoencoder models to identify model advantages in shape optimization. Autoencoder representations were shown to be more sensitive to changes in latent variables [33]. While this is not directly related, aerodynamic optimizations of vehicles share many similarities with airfoils. These types of techniques have been helpful in investigating the latent space of various optimization generations. From intuition, optimized geometry should generally be able to be represented in fewer dimensions than an entire database of non-optimized geometries.

2.6 Feature Selection and Latent Spaces

Using all of the features of a data set is not always necessary to create an efficient and accurate model. In fact, several studies have found that using more features than necessary may increase the computational load and thereby slow down the learning process, while yielding similar results as those obtained with a much smaller feature subset [34–36]. For these reasons, a common goal of feature selection is to use as few features as possible. Although it is generally best to follow this rule of thumb, features having a direct impact on the latent space representation may be useful points to consider.

Disentanglement and consistency are shared properties favorable latent spaces [37, 38]. One such measure of consistency within this basis is Latent Space Consistency, which uses the Pearson coefficient as a metric of how the data changes along any basis of the latent space [38]. While we do not apply these metrics since the autoencoder latent space does not specifically refer to the parameters; we focus on using the MSE between testing data and reprojected samples to evaluate the performance of each latent space dimension.

3 METHODS

In contrast to the related work, our contributions focus on the effects of optimized geometry and training models using performance parameters as additional features on the latent space dimensionality. To evaluate these contributions, we performed three steps: (1) preprocessing of the airfoil data sets, (2) construction of the dimension reduction models, and (3) model hyperparameter tuning.

Preprocessing Preprocessing of the data was necessary to ensure all features were equally valued. The parameters of Reynold's number, Mach number, angle of attack, and target lift coefficient for each individual sample were added to the optimized airfoil data set to create a new third set. A data frame was then created for each set, where all values were scaled between 0 and 1 using the MinMaxScaler from scikit-learn [39].

Dimension Reduction After scaling the data sets, two main trials were conducted to evaluate our models. The first trial compared our models to better understand the effects of using PCA and autoencoders on variable training fractions. The second trial holds this training fraction constant and evaluates model performance with respect to latent space size.

Trial 1 iterated through fractions of training data using the optimized airfoil data set and fixed latent space dimensions. Sample sizes ranged from 5-1025 airfoils to get a full range of sizes while saving close to 2% of the data for testing. The lower bound was selected to be 5 since the number of components must be less than or equal to than the number of samples. For each sample size, 25 training and testing splits were conducted so that reconstruction error could be averaged and a confidence interval could be determined. This was done in sub-trials with latent space sizes of 3, 4, and 5 for PCA and autoencoder models.

Trial 2 iterated through latent space sizes of 1-20 while holding the training testing split constant at 80%-20%. For each latent space size, 25 training and testing splits were conducted. This cross validation was performed so reconstruction error could be averaged over many training and testing splits, and to generate confidence intervals for PCA and autoencoder models. This trial was conducted with three datasets: UIUC airfoil, optimized airfoil, and the optimized airfoil data set with target optimization conditions.

Model Tuning After creating these models, optimization of their performance is necessary to ensure a fair comparison between PCA and autoencoders. This is accomplished via hyperparameter tuning. One such hyperparameter that requires fine tuning is an autoencoder's learning rate. This sets the step size of the optimizer at each iteration while it is minimizing loss. Since its value directly affects the training of the model, it also impacts the model accuracy. If this parameter is too small, the model will likely get trapped in local minimum. If this parameter is too large, its step size will be too big resulting in passing over the global minimum.

To mitigate the effects of learning rate on the model accuracy, this value needs to be optimized. From human fine tuning, the optimal learning rate was found to be approximately 1.0×10^{-4}. Additionally, ReduceLROnPlateau, a PyTorch dynamic scheduler, was utilized to further optimize the learning rate [40]. This scheduler works by reducing the value of the learning rate every time that learning stagnates and the loss remains the same for a given number of epochs. Unfortunately, after tweaking the parameters of this function, the results were no better than the constant learning rate, so this constant value was used throughout experimentation. An additional sensitivity study suggested that further optimization of learning rate would not result in significant performance enhancement with the data and model architecture used.

Model architecture may additionally impact autoencoder accuracy. The encoder was simplified to have one fully connected layer followed by a hidden layer with 1-20 nodes depending on the trial. The decoder is identically opposite to the encoder, expanding from the hidden layer to the fully connected layer. The number of nodes in the hidden layer were variable to account for many latent space sizes. The number of features in the data set were used as the number of nodes in the fully connected layers. This value was typically 384, however it was increased to 388 in the trial where additional airfoil properties were included as features. Reconstruction error was calculated by taking the MSE of encoder input and decoder output. We conducted a brief study by examining the testing MSE of autoencoders with additional hidden layers, finding that extra hidden layers do not result in significant performance benefits on the optimized airfoil data set. However, the inclusion of additional hidden layers resulted in increased computational expense. For these reasons only the simple autoencoder with one hidden layer was used throughout the study.

4 RESULTS

This section first demonstrates the model performance on the optimized airfoil data set. We then highlight the effects on the latent space of using non-optimized airfoils and finally exhibit the effects on the latent space of adding the specific input conditions the airfoils were optimized for as features of the data set.

4.1 Model Performance on Optimized Data set

Figure 1 displays the averaged mean testing MSE for each sample size along with bootstrapped empirical 95% confidence intervals over mean testing MSE values. Both PCA and autoencoder (AE) models are evaluated with latent space sizes of 3, 4, and 5 dimensions. In low dimensions, the autoencoder models have a significantly more narrow confidence interval and perform better than PCA on the majority of the training data fractions. With larger latent space sizes, extreme training sample sizes begin to result in high MSE values. As latent size increases, the mean testing confidence interval for PCA narrows considerably until it is similar to that of autoencoders at five dimensions. Please also note that it is somewhat deceptive to evaluate the con-

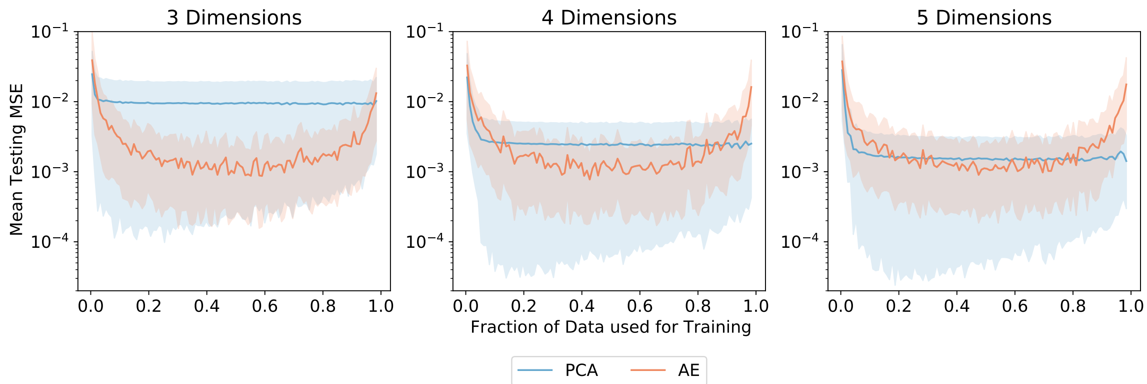

FIGURE 1. EFFECT OF TRAINING SAMPLE SIZE ON MEAN PCA AND AUTOENCODER TESTING MSE VALUES

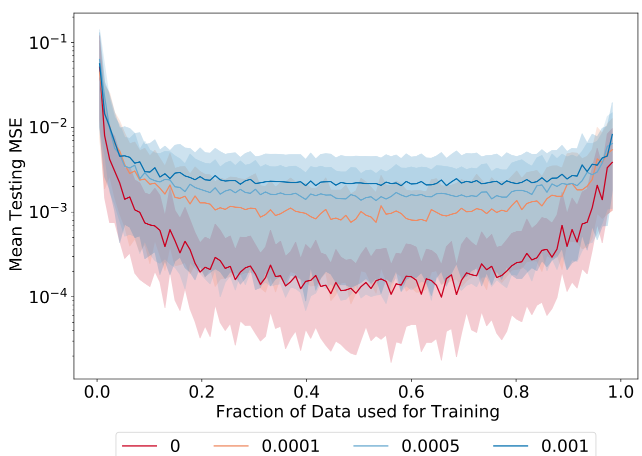

FIGURE 2. EFFECT OF WEIGHT DECAY ON MEAN TEST MSE AND TRAINING SIZE

fidence interval in a log-based scale. In this case the upper bound is substantially more telling of the confidence interval than the lower bound.

Secondly, the mean testing MSE of the autoencoder models increase considerably with high training sample fractions and greater latent dimensions. The confidence interval is much wider in this area for the autoencoder than for PCA. This may have occurred simply because the testing data size was small or because the autoencoder models became overfit with large sampling sizes. On the other hand, PCA is more stable with extreme training sample sizes and there was no systematic increase in testing MSE in the high training sample size regiment.

To address the concerns of overfitting, L2 regularization was applied and the previous trial was repeated for several weight decay parameters. Figure 2 shows that increasing weight decay results in higher test MSE values. From these results, there does not appear to be much overfitting as an increase in the weight decay parameter (as shown below the plot) typically corresponds to an increase in testing MSE. Nevertheless, there is still significant variance in testing MSE between cross validation trials that could potentially obfuscate overfitting.

Figure 3 displays histograms of the reconstruction error for each testing sample averaged over 25 trials for latent space sizes of 3, 4, and 5 dimensions. This suggests that in low dimensional spaces, autoencoders perform significantly better than PCA on this data set. Figure 4 shows the full story where autoencoder models start off with lower reconstruction error with low latent space dimensionality but are outperformed in higher dimensions by PCA. We interpret this as the autoencoder is likely getting stuck in local optima, a common issue for this model type.

4.2 Optimized Geometry vs Non-Optimized Geometry

We had originally hypothesized that the optimized airfoil coordinates would likely lie on a lower dimensional design manifold. Although this study did not confirm our hypothesis, it did show us some of the differences between these two datasets. For one, from Fig. 4 it is clear that the optimized airfoil data set is highly nonlinear. Testing MSE for PCA was substantially higher with this data set than that of the UIUC airfoil database. It took an additional two dimensions for the PCA models to reach the same accuracy as the autoencoder models for the optimized airfoil dataset. On the UIUC data set, autoencoder and PCA models performed similarly in lower dimensions.

The bootstrapped empirical 95% confidence interval over mean testing MSE is given by transparent shading for PCA models and darker shading for autoencoder models. This confidence interval is significantly wider for the optimized airfoil data set than the UIUC set. This implies that that both PCA and autoencoder models were struggling to model this data more so than the UIUC airfoil database. The optimized airfoil coordinates do not appear to lie on a lower dimensional design manifold. Of course, further studies are necessary to confirm this given that not all modeling methods and activation functions were tested.

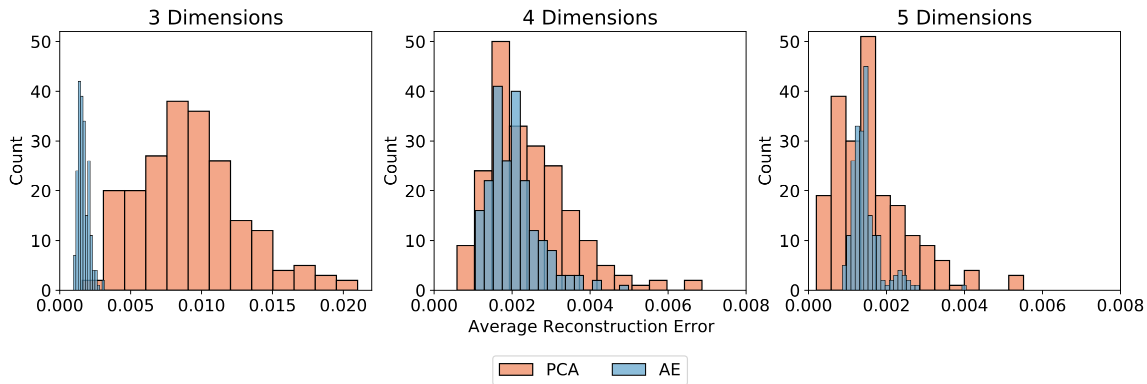

FIGURE 3. HISTOGRAM OF AVERAGE TESTING RECONSTRUCTION ERROR FOR VARIOUS LATENT SPACE SIZES USING OPTIMIZED DATA SET

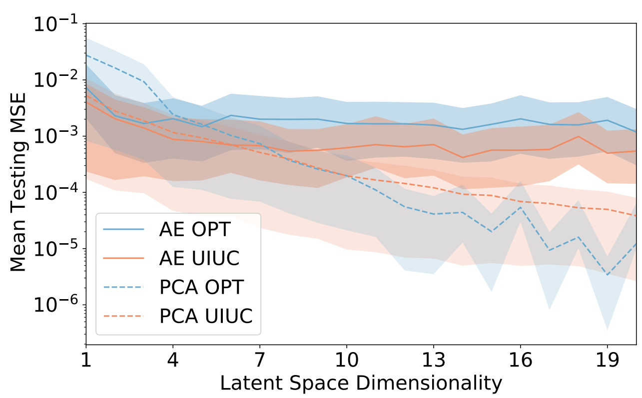

FIGURE 4. MEAN PCA AND AUTOENCODER TESTING MSE VALUES VS. DIMENSION ON OPTIMIZED AND UIUC AIRFOIL DATASETS

4.3 Including Parameters

The input parameters of Reynold's number, Mach number, angle of attack, and coefficient of lift were introduced as data set features. The optimized airfoil coordinates were generated using these conditions. Since the original features are dependent on these four inputs, we examine their effect on the latent space dimensionality in Fig. 5.

There are no significant differences in performance and dimensionality between the data sets. The greatest contrast is between the selected models. Once again, the autoencoders perform considerably better than PCA in lower dimensional latent spaces until five dimensions.

After adding the input parameters, the autoencoders took over four times as long to run as they did on the original optimized airfoil data set. This computational cost is great for insignificant performance benefits. Input parameters were found to be not only unnecessary for model fitting, but to hinder it. As discussed in Section 2.6, previous work has found larger feature sizes to take longer to run. We attribute the increased run time to increased feature size and to the model selection. Extensions of this work using various autoencoder architectures and activation functions may yield different results.

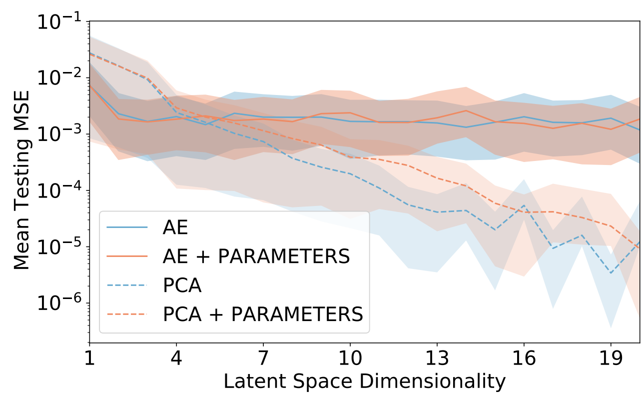

FIGURE 5. EFFECT OF INPUT PARAMETERS ON MEAN PCA AND AUTOENCODER TESTING MSE VALUES VS. DIMENSION

5 DISCUSSION

This section first discusses how geometry optimality and performance parameters affect reconstruction and latent space dimensionality. We then digress to an overview of the limitations of this study such as autoencoder architecture, activation functions, and the issue of autoencoders getting stuck in local optima.

How Does Geometry Optimality Affect Latent Space Dimensionality? Using various methods, the UIUC data set intrinsic dimensionality was found to be between 2 and 6 with a greater probability around 4 dimensions [26]. Using PCA and autoencoders, we found that these methods had similar performance around 4-6 dimensions for the UIUC and the optimized airfoil data set. While our data suggest that optimized geometry does not necessarily lie in a lower dimensional latent space than the original geometry, the model variety used was not comprehensive. Alternative generative models such as variational autoencoders or generative adversarial networks may exhibit different behavior.

How Does Including Performance Parameters Affect Reconstruction? We found that including input parameters as data set features from which the airfoil coordinates were generated does not necessarily decrease the latent space dimensionality. Since additional features only increase the computational cost, this finding suggests that there is no benefit to including additional features in the data set on which the rest of the airfoil geometry is dependent on. It also could be the case that the range of physical parameters over which the geometries were optimized, such as the range of Reynolds numbers, Mach numbers, angles of attack and lift coefficients, may not have been helpful to the dimensionality reduction. Extensions of this work into broad aerodynamic regimes or alternative flow models may alter our findings.

How Might the Autoencoder Architecture Alter Our Results? Autoencoder architecture may additionally impact reprojection accuracy. For uniformity, all autoencoder models were given the same architecture in the results shown. A separate study was conducted to validate that this architecture was appropriate. Deeper neural networks from 2 to 10 hidden layers were additionally investigated to confirm that there was no significant performance increase for various architectures using the optimized airfoil data set. It was found that shallow networks were sufficient and that increasing the number of hidden layers actually results in worse performance for deep models. Moreover, as the network depth grew, so did the training time. For this reason and for uniformity, the original model with one hidden layer was used for all experiments. Autoencoder architecture was also validated by repeating some of the trials after removing the nonlinear activation functions. Since the loss metric was already MSE, this newly-constructed autoencoder represents the same linear transformation as PCA, resulting in the same solution as in the PCA trial.

How Might the Activation Functions Alter Our Results? The ReLU activation function was used throughout this study because it is fast, does not typically have the problem of vanishing gradients, and it generates sparse representations. We also assumed that the since PCA performs worse on the optimized airfoils than on the UIUC airfoils that the optimized data set is more nonlinear. While we did make this assumption, the curvature of the manifold was not thoroughly studied. We plan to investigate this phenomenon further and assign explicit manifold curvature metrics. Methodologies to better investigate the manifold curvature would likely need to rely on other, more differentiable activation functions.

Additional activation functions were used in combination with the architecture study. Sigmoid, Tanh, ELU, and CELU were tested in models ranging from 1-10 hidden layers. Sigmoid and Tanh were expected to have poor performance as they can often have vanishing gradients. This was observed when Sigmoid activation was used. On the other hand, Tanh, ELU, CELU, all performed reasonably similar to ReLU throughout all architecture sizes. Further study of these activation functions is necessary for future curvature investigations.

How Might Autoencoders Be Forced to Global Optima? Throughout this study, one of the major issues observed was autoencoders losing their way with local optima and not converging to global minima. A method to solve the problem is to learn their optimal representations using gradient-based optimizers. This method was recently developed and applied to linear autoencoders [41]. Gradient-based approaches and regularization methods show promise in learning the optimal representation of principal components. We plan to conduct further research in applying these methods to nonlinear autoencoders.

6 CONCLUSION

In this paper we investigated three characteristics of airfoil manifolds. First, the impact of latent dimension size and sample size on the optimized airfoil data set reconstruction error was explored. We found that autoencoders perform better in low dimensional spaces but PCA is more stable with extremely low or high fractions of training samples. Second, we investigated the intrinsic dimensionality between the UIUC airfoil database and a subset of these airfoils optimized using CBGAN and CEBGAN models. From this study we determined that optimized geometries do not necessarily live in lower dimensional manifolds. Finally, we investigated the effect of including target optimization conditions as data set features on reconstruction error over a swath of latent space sizes. We found that adding features that all data is dependent on does not necessarily reduce data set dimensionality.

Airfoil manifolds remain a complex area of study with numerous factors impacting latent space dimension and reprojection accuracy. Our approach provides several experiments to help understand the implicit effect of data and model selection on the latent space. Our findings suggest that performing DR on optimized airfoil geometries and including dependent features in the data set do not benefit the latent space dimensionality or reprojection error. With our contributions, we hope to shed some light on various characteristics of airfoil manifolds and to lay groundwork for future studies in this field.

ACKNOWLEDGMENT

This research was supported in part by funding from the U.S. Department of Energy's Advanced Research Projects Agency-Energy (ARPA-E) DIFFERENTIATE funding opportunity through award DE-AR0001201.

REFERENCES

[1] Vanderplaats, G. N., 1979. Approximation Concepts for Numerical Airfoil Optimization. Tech. rep., NASA Technical Paper 1370, 03.

[2] Viswanath, A., Forrester, A. I., and Keane, A., 2011. "Dimension reduction for aerodynamic design optimization". *AIAA journal,* ***49***(6), pp. 1256–1266.

[3] Yasong, Q., Junqiang, B., Nan, L., and Chen, W., 2018. "Global aerodynamic design optimization based on data dimensionality reduction". *Chinese Journal of Aeronautics,* ***31***(4), pp. 643–659.

[4] Gómez-Bombarelli, R., Duvenaud, D., Hernández-Lobato, J. M., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A., 2016. "Automatic chemical design using a data-driven continuous representation of molecules". *CoRR,* ***abs/1610.02415***.

[5] Delgado-Guerrero, J. A., Colomé, A., and Torras, C., 2020. "Sample-efficient robot motion learning using gaussian process latent variable models". In 2020 IEEE International Conference on Robotics and Automation (ICRA), IEEE, pp. 314–320.

[6] Regier, J. C., and Stark, P. B., 2015. "Mini-minimax uncertainty quantification for emulators". *SIAM/ASA Journal on Uncertainty Quantification,* ***3***(1), pp. 686–708.

[7] Bellman, R., 1957. "Dynamic programming". *Press Princeton, New Jersey*.

[8] Ghodsi, A., 2006. "Dimensionality reduction a short tutorial". *Department of Statistics and Actuarial Science, Univ. of Waterloo, Ontario, Canada,* ***37***(38), p. 2006.

[9] Izenman, A. J., 2012. "Introduction to manifold learning". *Wiley Interdisciplinary Reviews: Computational Statistics,* ***4***(5), pp. 439–446.

[10] Van Der Maaten, L., Postma, E., Van den Herik, J., et al., 2009. "Dimensionality reduction: A comparative review". *J Mach Learn Res,* ***10***(66-71), p. 13.

[11] Shlens, J., 2014. "A tutorial on principal component analysis". *arXiv preprint arXiv:1404.1100*.

[12] Holland, S. M., 2008. "Principal components analysis (pca)". *Department of Geology, University of Georgia, Athens, GA*, pp. 30602–2501.

[13] Ladjal, S., Newson, A., and Pham, C.-H., 2019. "A pca-like autoencoder". *arXiv preprint arXiv:1904.01277*.

[14] Bank, D., Koenigstein, N., and Giryes, R., 2020. "Autoencoders". *arXiv preprint arXiv:2003.05991*.

[15] Fournier, Q., and Aloise, D., 2019. "Empirical comparison between autoencoders and traditional dimensionality reduction methods". In 2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), IEEE, pp. 211–214.

[16] Wang, Y., Yao, H., and Zhao, S., 2016. "Auto-encoder based dimensionality reduction". *Neurocomputing,* ***184***, pp. 232–242.

[17] Wang, W., Huang, Y., Wang, Y., and Wang, L., 2014. "Generalized autoencoder: A neural network framework for dimensionality reduction". In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops.

[18] Plaut, E., 2018. "From principal subspaces to principal components with linear autoencoders". *arXiv preprint arXiv:1804.10253*.

[19] Berkooz, G., Holmes, P., and Lumley, J. L., 1993. "The proper orthogonal decomposition in the analysis of turbulent flows". *Annual review of fluid mechanics,* ***25***(1), pp. 539–575.

[20] LeGresley, P., and Alonso, J., 2000. "Airfoil design optimization using reduced order models based on proper orthogonal decomposition". In Fluids 2000 conference and exhibit, p. 2545.

[21] Liu, D., and Tang, L., 2006. Proper orthogonal decomposition (pod)/response surface methodology (rsm) methodology for low reynolds number aerodynamics on micro aerial vehicle (mav). Tech. rep., ZONA TECHNOLOGY INC SCOTTSDALE AZ.

[22] Bui-Thanh, T., Damodaran, M., and Willcox, K., 2004. "Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition". *AIAA journal,* ***42***(8), pp. 1505–1516.

[23] Chen, W., Chiu, K., and Fuge, M. D., 2020. "Airfoil design parameterization and optimization using bézier generative adversarial networks". *AIAA journal,* ***58***(11), pp. 4723–4735.

[24] Omata, N., and Shirayama, S., 2019. "A novel method of low-dimensional representation for temporal behavior of flow fields using deep autoencoder". *Aip Advances,* ***9***(1), p. 015006.

[25] Yonekura, K., Wada, K., and Suzuki, K., 2021. "Generating various airfoil shapes with required lift coefficient using conditional variational autoencoders". *arXiv preprint arXiv:2106.09901*.

[26] Chen, W., Fuge, M., and Chazan, N., 2017. "Design manifolds capture the intrinsic complexity and dimension of design spaces". *Journal of Mechanical Design,* ***139***(5), pp. 051102–051102–10.

[27] Chen, Q., Pope, P., and Fuge, M., 2022. "Learning airfoil manifolds with optimal transport". In AIAA SCITECH 2022 Forum, p. 2352.

[28] Chen, Q., Wang, J., Pope, P., Chen, W., and Fuge, M., 2021.

"Inverse design of two-dimensional airfoils using conditional generative models and surrogate log-likelihoods". *Journal of Mechanical Design,* ***144***(2), p. 021712.

[29] Chen, N., Karl, M., and Van Der Smagt, P., 2016. "Dynamic movement primitives in latent space of time-dependent variational autoencoders". In 2016 IEEE-RAS 16th international conference on humanoid robots (Humanoids), IEEE, pp. 629–636.

[30] Pope, P., Zhu, C., Abdelkader, A., Goldblum, M., and Goldstein, T., 2021. "The intrinsic dimension of images and its impact on learning". *arXiv preprint arXiv:2104.08894*.

[31] Levina, E., and Bickel, P. J., 2005. "Maximum likelihood estimation of intrinsic dimension". In Advances in neural information processing systems, pp. 777–784.

[32] Rios, T., Sendhoff, B., Menzel, S., Bäck, T., and van Stein, B., 2019. "On the efficiency of a point cloud autoencoder as a geometric representation for shape optimization". In 2019 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, pp. 791–798.

[33] Rios, T., van Stein, B., Wollstadt, P., Bäck, T., Sendhoff, B., and Menzel, S., 2021. "Exploiting local geometric features in vehicle design optimization with 3d point cloud autoencoders". In 2021 IEEE Congress on Evolutionary Computation (CEC), IEEE, pp. 514–521.

[34] Mladenić, D., 2005. "Feature selection for dimensionality reduction". In International Statistical and Optimization Perspectives Workshop" Subspace, Latent Structure and Feature Selection", Springer, pp. 84–102.

[35] Jiang, X., 2011. "Linear subspace learning-based dimensionality reduction". *IEEE Signal Processing Magazine,* ***28***(2), pp. 16–26.

[36] Battiti, R., 1994. "Using mutual information for selecting features in supervised neural net learning". *IEEE Transactions on Neural Networks,* ***5***(4), pp. 537–550.

[37] Kim, H., and Mnih, A., 2018. "Disentangling by factorising". In International Conference on Machine Learning, PMLR, pp. 2649–2658.

[38] Chen, W., and Fuge, M., 2019. "Synthesizing designs with inter-part dependencies using hierarchical generative adversarial networks". *Journal of Mechanical Design,* ***141***(11), p. 111403.

[39] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E., 2011. "Scikit-learn: Machine learning in Python". *Journal of Machine Learning Research,* ***12***, pp. 2825–2830.

[40] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S., 2019. "Pytorch: An imperative style, high-performance deep learning library". In *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds. Curran Associates, Inc., pp. 8024–8035.

[41] Bao, X., Lucas, J., Sachdeva, S., and Grosse, R. B., 2020. "Regularized linear autoencoders recover the principal components, eventually". *Advances in Neural Information Processing Systems,* ***33***, pp. 6971–6981.