AeroML: Generative Modeling of Airfoil Geometries with Desirable Aerodynamic Performance

MAE 551/451: Applied Machine Learning for Mechanical and Aerospace Engineers

Fall 2025

Group 4: Dan Hlevca, Ryan Yeager, Trevor Dyhrkopp, Vedant Bhat and Vishavjit Singh Khinda

Problem Statement

Modelling complex aerodynamic data using conventional simulation methods poses a significant challenge for high-throughput processes. Multidimensional problems involving solutions to the Navier-Stokes equation require highly specialized tools and substantial computational resources. Running these simulations for large meshes demands even more time and resources, and remains exceptionally slow. Despite their accuracy, these methods are too inefficient for high-throughput generation of new aerodynamic data.

With the advent of more sophisticated and powerful machine learning techniques, ML models present themselves as a powerful tool for applications like these. Uses include regression for predicting properties based upon input features, to the generation of new systems with desired properties. We utilize an autoencoder in tandem with an inverse neural network (INN) to generate new airfoil geometries from a list of desired aerodynamic parameters. The trained model shows promising results by generating new airfoil geometries that conform to desired performance metrics.

Approach

To obtain an ML model, a dataset must first be acquired. Our goal was to work with and generate new airfoils; we needed accurate aerodynamic data on a wide range of airfoil geometries and boundary conditions so that the model could learn a relationship between the aerodynamic behavior and airfoil contour. For this reason, the *airfRANS* dataset developed by Bonnet et al. (2022) was picked. This dataset consists of numerical simulations solving the incompressible steady-state Reynolds-Averaged Navier-Stokes equations over 2D airfoils in the subsonic regime with different angles of attack (AoA) (Bonnet et al., 2022). It comprises 1,000 simulations of the NACA 4 and 5-digit series airfoils with Reynolds numbers between 2 to 6×10^6 and AoA ranging from 5° to 15°. Each simulation in this dataset includes the following information: coordinates, global inlet velocities, distance to airfoil, normals, velocity, pressure per unit density, and turbulent kinematic viscosity. A subset of the entire dataset, the ‘scarce’ dataset was chosen, with 200 simulations in the training split. While this dataset significantly reduces the number of simulations available for training, the training times are much shorter.

The next step in the process involves choosing and setting up a model. We decided to utilize an autoencoder for generative modeling, which was then coupled with an inverse neural network. Both models were designed to work in tandem with each other; the autoencoder (decoder portion) was responsible for the generation of new airfoil geometries, while the INN handles converting aerodynamic properties into the latent space representation required as input to the decoder. The autoencoder has an input dimension of 18, a hidden dimension of 10, and a latent space dimension of six. The INN has an input dimension of 1,007, accounting for the scalar aerodynamic data (C_d, C_L, C_M, etc.; total of 7) and pressure distribution across the airfoil (total of 1,000). The first two hidden-layer dimensions were set to size 1,024, then the third was stepped down to 512, and finally to an output dimension of 6, the same as the latent space dimension of the decoder.

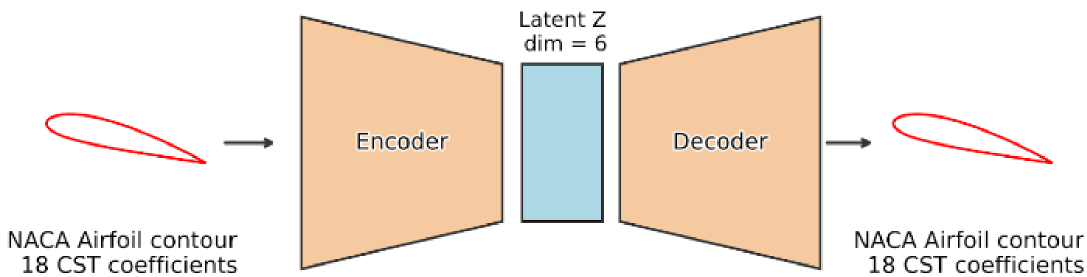

Figure 1: Schematic diagram of the autoencoder used to get the latent space representation of the NACA airfoil's profile contour.

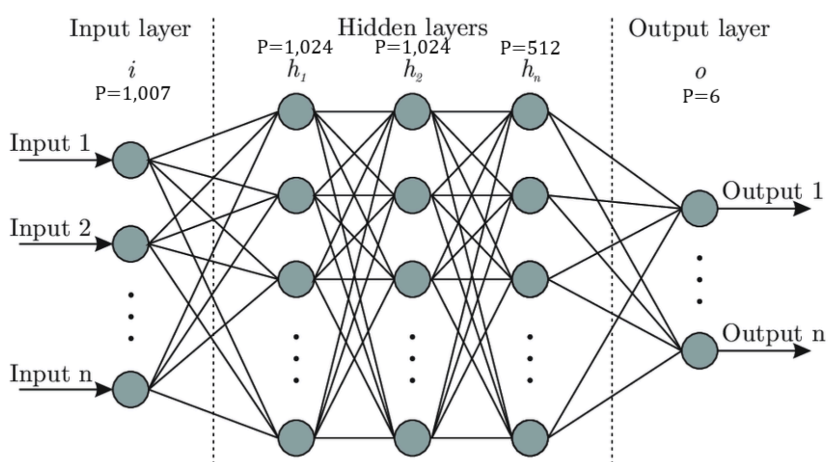

Figure 2: Schematic diagram of the inverse neural network used to map aerodynamic parameters to the autoencoder's latent space.

To prepare the dataset for training, a preprocessing step was necessary. Each airfoil geometry in the airfRANS dataset simulations does not sample an identical number of points. This results in distinct airfoil geometries having a different number of points sampled. This was alleviated by using Class Shape Transformations (CST). CST is a method that can be used to analytically represent surface coordinates of aerospace structures. This method was used here to convert the inhomogeneous simulation dataset sizes to a standardized representation containing 18 CST coefficients; all of the 200 training samples were thus converted to their CST equivalents and used for training the autoencoder. To train the INN, a similar challenge existed. A set of global parameters was used to describe the desirable aerodynamic properties of the airfoil. These features include the total drag and lift force on the airfoil; the drag, lift, and moment coefficients; the inlet velocity; and the angle of attack. The pressure acting on the airfoil is also an important metric that needs to be included to accurately describe the aerodynamic behavior. As stated above, since the simulations do not sample a homogeneous number of points, different airfoil geometries had different numbers of points with pressure values. This necessitated another preprocessing step. For pressure, having the mesh on the airfoil meant a better description of the behavior, and thus, it was not reduced using CST. Rather, all the pressure arrays were reshaped to have 1,000 data points per simulation. This meant interpolating for geometries with fewer simulation observations and dropping values for those with more simulation results. All values listed above as input can be obtained from either of three ways: directly from the dataset, by calling the class methods of the dataset's Simulation class, or from the simulation file names. At the end of the preprocessing steps, the inputs to both models were homogenized: the autoencoder accepted an input of 18 CST coefficients, and the inverse neural network accepted an input of the 7 global aerodynamic parameters combined with the pressure acting on the airfoil at 1,000 points.

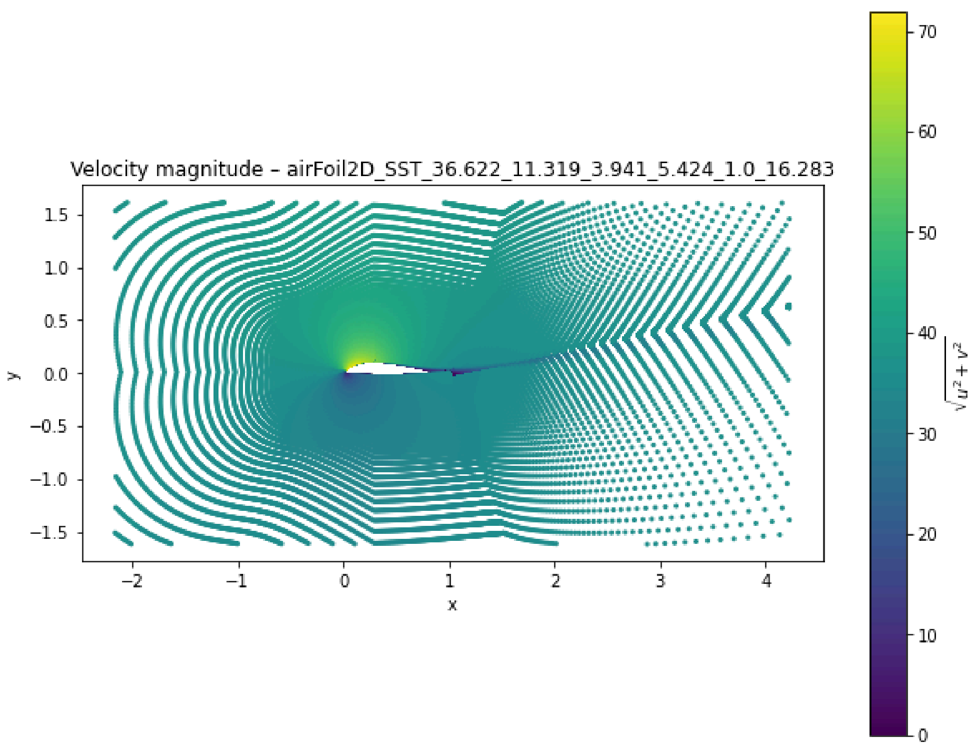

Figure 3: Velocity magnitude around the airfoil. The mesh is markedly denser near the airfoil surface.

The autoencoder and INN were then trained for 500 and 300 epochs, respectively. Both models utilized MSE loss and the Adam optimizer algorithm, with a learning rate of 1×10^{-3}. The inverse neural network was additionally given an L2 penalty of 1×10^{-4} to prevent the model from overfitting due to the small size of the training dataset.

Results

Before discussing results from the machine learning models, it was necessary to validate the CST approach taken to homogenize the inputs to the autoencoder.

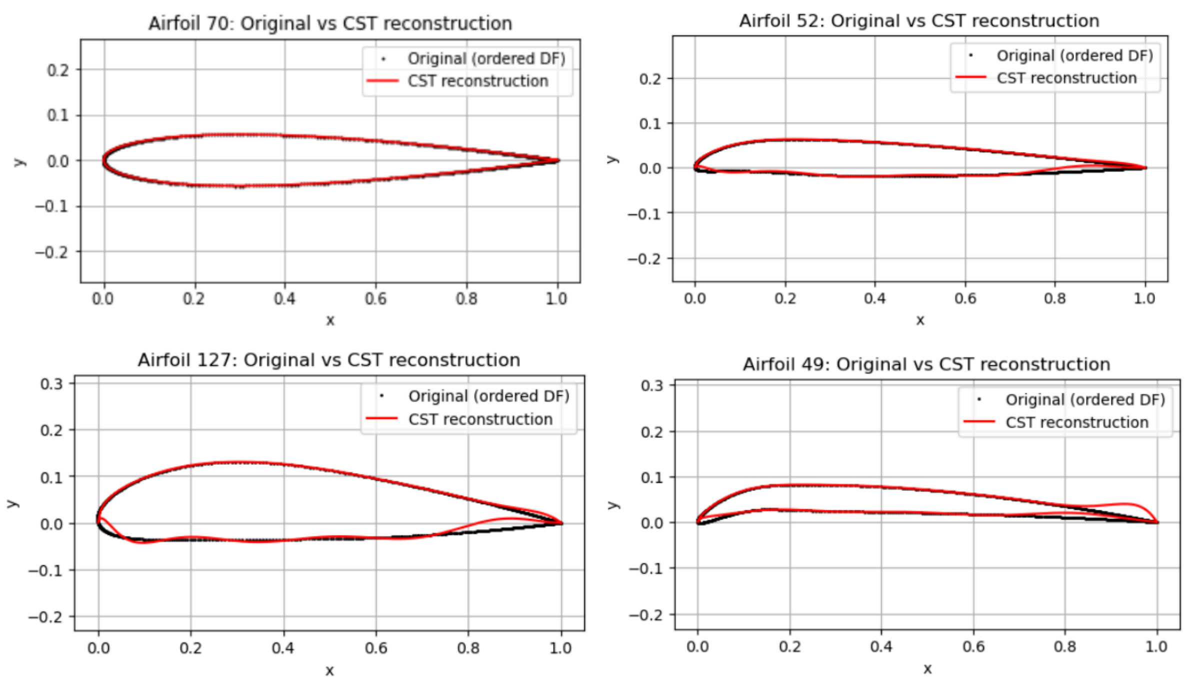

Figure 4: Comparison of an airfoil's geometry, as described by datapoints and a CST reconstruction.

Figure 4 shows the original geometry and a CST reconstruction of one of the airfoils in the dataset. As is evident, the CST reconstruction matches the original airfoil perfectly; however, the same cannot be said for all airfoil geometries. As shown by Figure 5, CST was unable to regenerate some airfoil geometries accurately. After comparing CST reconstructions of 200 airfoils with their originals, the autoencoder correctly matched the original airfoil's shape in 75% observations, another 20% of samples showing an extremely good match, with the final 5% of samples displaying a poor reconstruction. As seen in Figure 4, Airfoil #70 shows a correct reconstruction, #52 is what we're describing as an 'extremely good' match, and airfoils #49 and #127 are a bad match. A discussion on the inconsistent reconstructions can be found in the upcoming section.

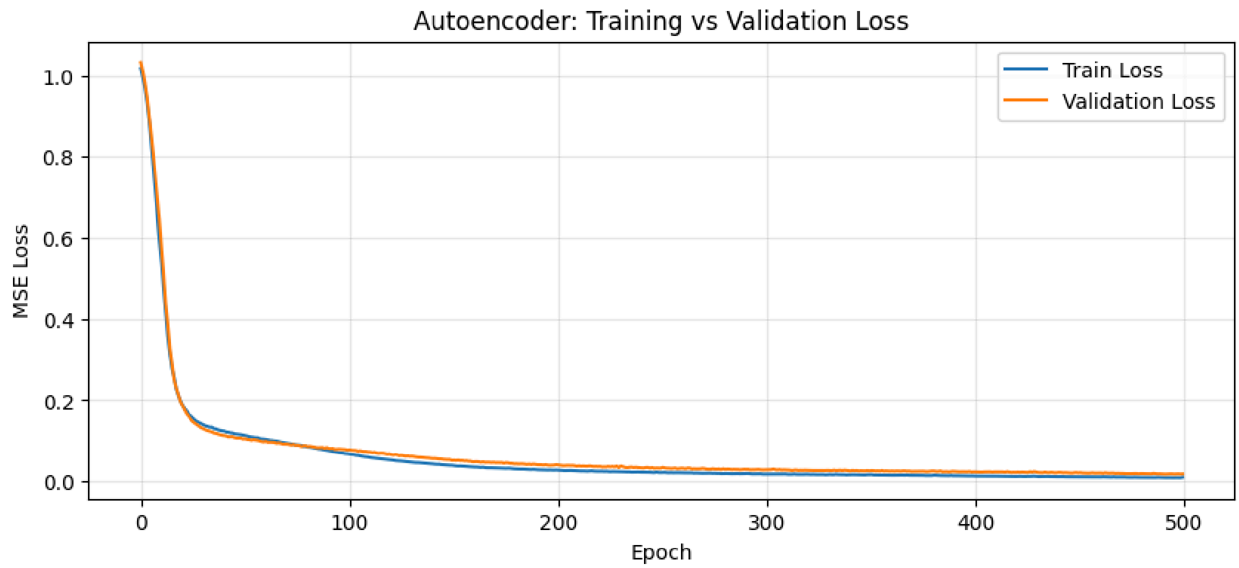

Figure 5: Training and validation loss curves for the autoencoder.

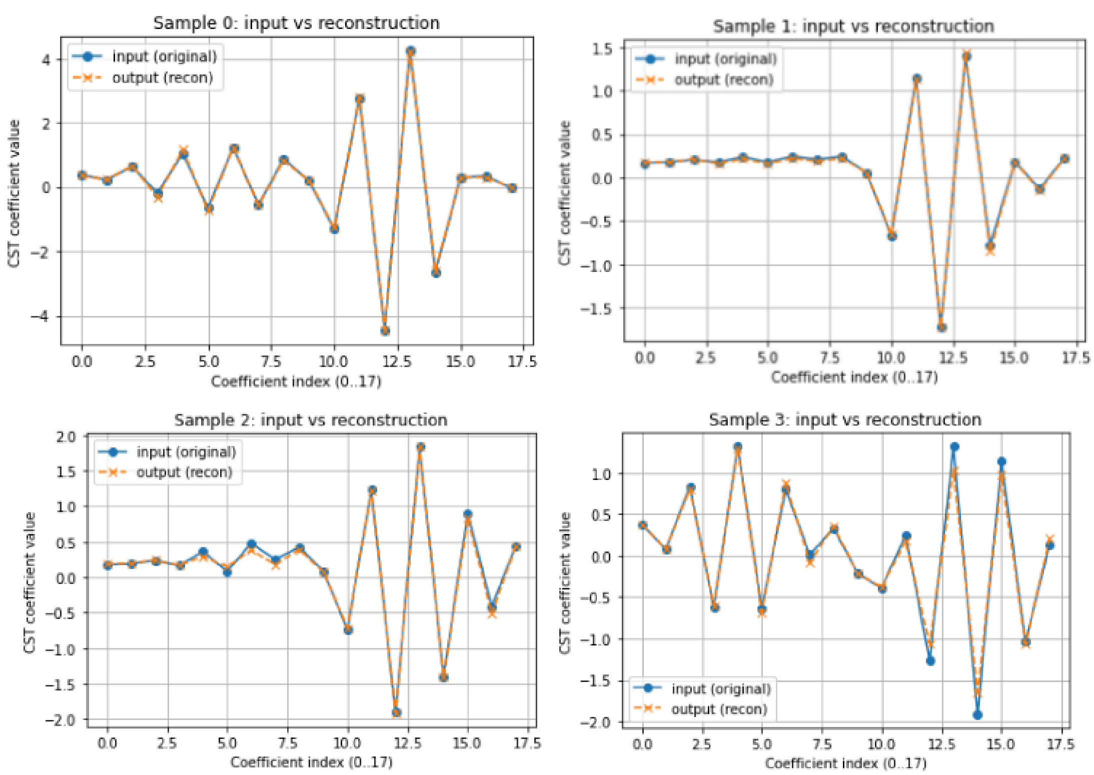

Figure 6: Validating the autoencoder, referencing the 18 input CST coefficients with the decoded (reconstructed) 18 CST coefficients.

Figure 5 shows the loss curves for the autoencoder. The final training and validation losses at the end of 500 epochs were observed to be 0.009951 and 0.01785, respectively. The trained autoencoder was then validated for its ability to reconstruct the CST coefficients from the training dataset. Figure 6 shows a comparison of the original and autoencoder-predicted CST coefficients. Excellent agreement was observed between the original CST coefficients and the autoencoder's predictions.

After training the autoencoder, the inverse neural network was trained to map aerodynamic parameters to the autoencoder's latent space. At the end of 300 epochs, a final training loss of 0.02541 was observed. The trained inverse neural network showed a final validation loss of 0.3106.

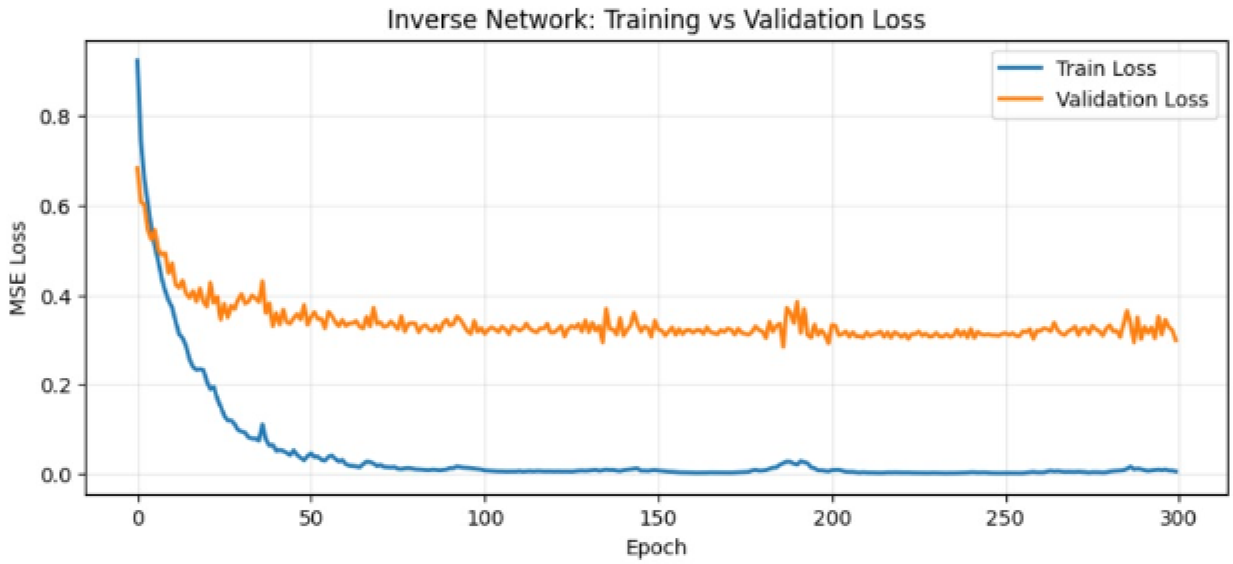

Figure 7: Training and validation curve for the inverse neural network.

With both components trained, an ideal airfoil was generated using the models. This was done by setting the inputs to the inverse neural network equal to the idealized global parameters within the dataset. The criteria for choosing the ideal parameters based on the observations in the dataset are discussed in depth later on. On passing these values through both models, the airfoil shown in Figure 8 was generated.

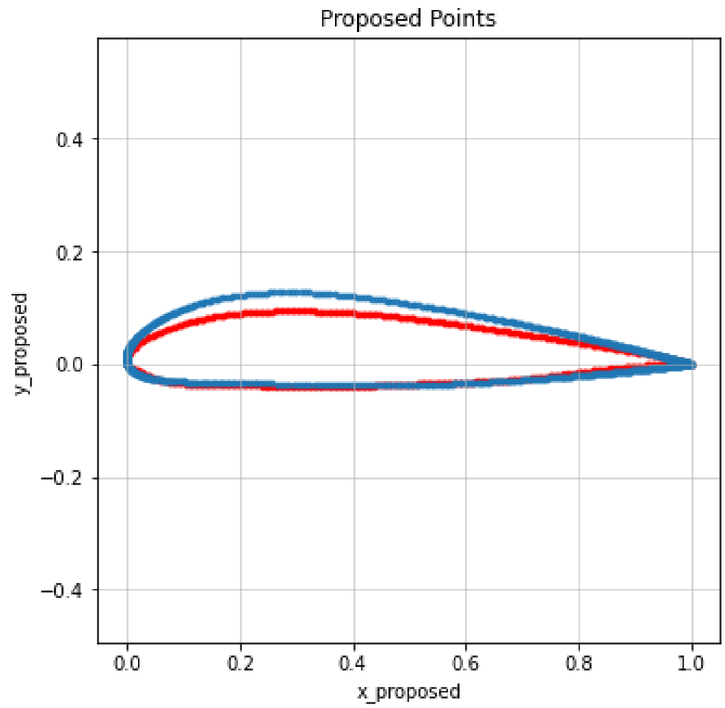

Figure 8: Proposed airfoil geometry from the idealized aerodynamic parameters and pressure distribution. The red curve shows the proposed airfoil, and the blue curve shows a standard NACA airfoil.

An important caveat of using this approach is that the idealized aerodynamic parameters need to be chosen carefully. Assuming a well-trained model, it needs to be ensured that the ideal point lies within the three standard deviations of the training manifold within the latent space. Failure to ensure this leads to the generated airfoils being unrealistic. Principal component analysis (PCA) provides a way to ensure that the ideal point lies within the training manifold in the latest space. PCA can be performed on the latent space of the autoencoder to obtain the manifold of the airfoil geometries the model was trained on. Figure 9 below shows the PCA results for the airfoil geometry proposed above in Figure 8.

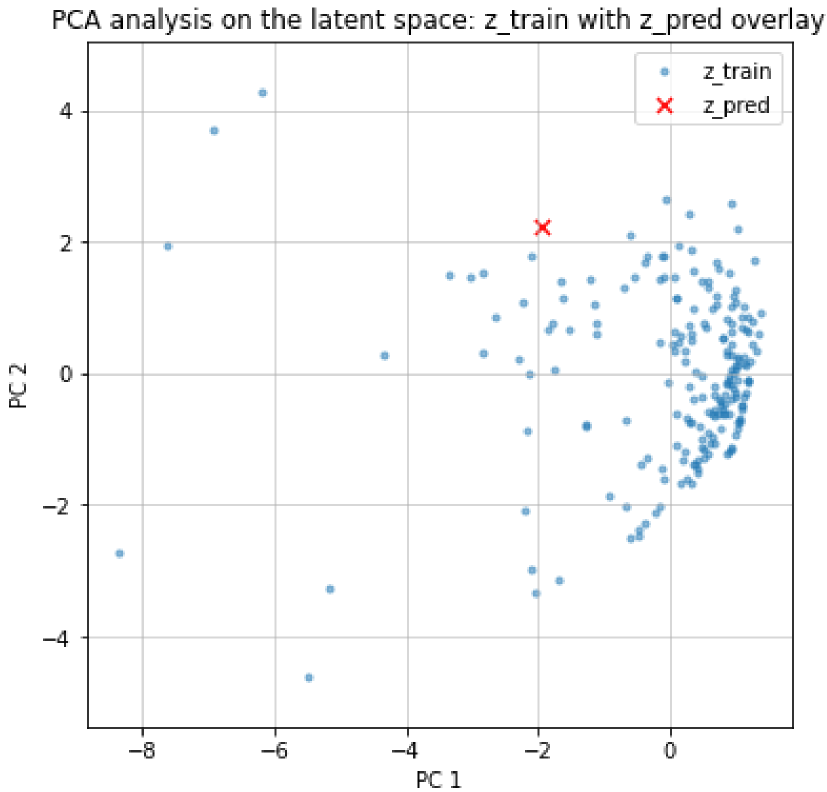

Figure 9: PCA of the latent space of the autoencoder and the predicted point (this is to check if the prediction lies on the same manifold).

Discussion

The choice of the smaller 'scarce' dataset was also influenced by the results delivered by this regime in the paper by Bonnet and coworkers. They found this regime performed better than other regimes for force coefficient predictions and showed the smallest MSE loss. To calculate these coefficients, an integration is performed over the airfoil's surface; the lower number of observations in the 'scarce' dataset led to smaller loss accumulation over the entire dataset. However, given the smaller size of the dataset compared to the full dataset, the authors argued that MSE loss is not a good proxy for gauging the accuracy of the force coefficients (Bonnet et al., 2022). This contradictory behavior of the 'scarce' dataset warrants further investigation using different loss functions and hyperparameters to determine its usability in scenarios identical to the one in this project.

Regarding the CST reconstructions, an interesting observation is that the majority of the inaccuracies occurred on the bottom surface of the airfoil, while the top surface showed vastly better agreement with the original airfoils in terms of max height and concavity. Since CST is an extremely complex field and its calibration requires specialized knowledge and procedures, given the scope of this project, we will refrain from a detailed discussion on this.

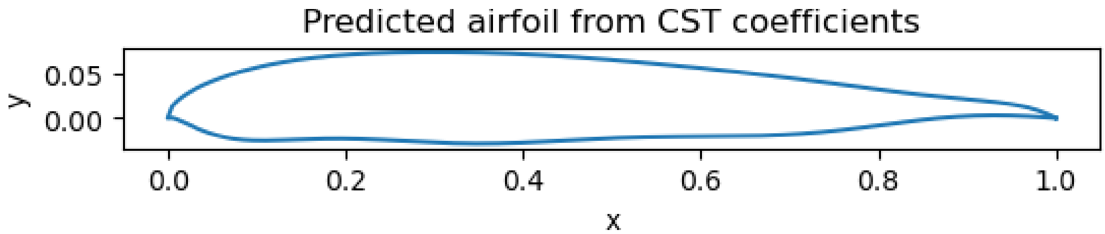

Figure 10: A CST reconstruction showcasing the differences in upper and lower airfoil surface generation.

The validation loss on the trained inverse neural network was not able to converge with any variation in the number of hidden layers, hidden neurons, or the learning rate. The only changes available for the INN to improve its validation loss are a larger dataset and longer training. However, training for longer will likely result in an overfitted model, given the already small training loss. Thus, utilizing a larger dataset remains the only viable option here. Given the time and scope of this project, this was not tested here, but should be tested on future iterations nonetheless.

The criteria for choosing ideal values for the inputs to the inverse neural network are as follows: minimum total drag and maximum lift force on the airfoil; minimum drag coefficient, maximum lift coefficient, and moment coefficient closest to -0.05; minimum inlet velocity; and AoA closest to 5°. The INN also requires the pressure distribution along the airfoil as an input; however, we could not find a reliable method of either choosing or generating this input data. Therefore, the pressure distribution from the first observation in the dataset was utilized.

Additionally, an important caveat of using this approach is that the idealized aerodynamic parameters need to be chosen carefully. Assuming a well-trained model, it needs to be ensured that the ideal point lies within the three standard deviations of the training manifold within the latent space. Failure to ensure this leads to the generated airfoils being unrealistic. Some examples of the unrealistic airfoils we obtained are shown below in Figure 11.

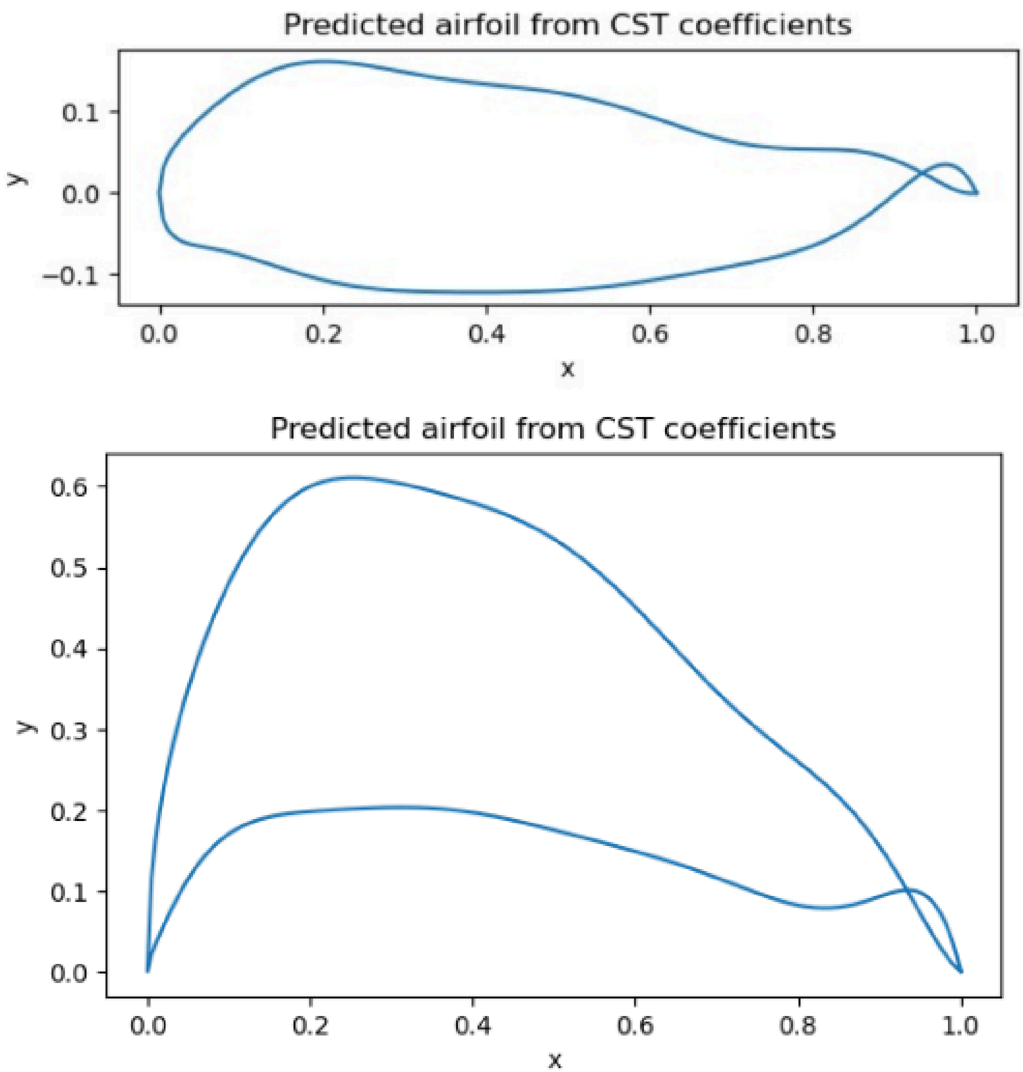

Figure 11: Unrealistic airfoils generated by the autoencoder.

Contribution Claim

- Dan Hlevca: 20% - Code
- Ryan Yeager: 20% - Presentation
- Trevor Dyhrkopp: 20% - Report
- Vedant Bhat: 20% - Report
- Vishavjit Singh Khinda: 20% - Code

References

Bonnet, F., Mazari, J., Cinnella, P., & Gallinari, P. (2022). Airfrans: High fidelity computational fluid dynamics dataset for approximating reynolds-averaged navier–stokes solutions. *Advances in Neural Information Processing Systems*, *35*, 23463-23478.