Feature Selection, Clustering, and Prototype Placement for Turbulence Data Sets

Matthew Barone*, Jaideep Ray† and Stefan Domino‡
Sandia National Laboratories§, Albuquerque, NM 87185

This paper explores automated approaches for the analysis and categorization of turbulent flow data, as a means of assessing the quality of a turbulence dataset used for constructing data-driven turbulence closures. Single point statistics from several high-fidelity turbulent flow simulation data sets are differentiated into groups using a Gaussian mixture model clustering algorithm. Candidate features are proposed, and a feature selection algorithm is applied to the data in a sequential fashion, flow by flow, to identify a good feature set and an optimal number of clusters for each dataset. Clusters are first identified for plane channel flows, producing results that agree with existing theory and empirical observations. Further clusters are then identified in an incremental fashion for flow over a wavy-walled channel, flow over a bump in a channel, and flow past a square cylinder. Some clusters are closely identified with the anisotropy state of the turbulence, whereas others can be connected to physical phenomena, such as boundary layer separation and free shear layers. Exemplar points from the clusters, or prototypes, are then identified using a prototype selection method. These exemplars effectively summarize the dataset using a greatly reduced collection of data points. The clusters and their prototypes are used to assess the quality of a training dataset constructed by simply pooling the four flows. We enumerate the dataset's shortcomings and state the limits of generalizability of any data-driven closure trained on it.

*Principal Member of the Technical Staff, Aerosciences Department, AIAA Associate Fellow, mbarone@sandia.gov
†Distinguished Member of the Technical Staff, Extreme Scale Data Science and Analytics Department, jairay@sandia.gov
‡Principal Member of the Technical Staff, Computational Thermal and Fluid Mechanics Department, spdomin@sandia.gov
§Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the 'views of the U.S. Department of Energy or the United States Government.

Nomenclature

b	Prototype quality measure
$b_{ij}, \mathbf{b}$	Reynolds stress anisotropy tensor
C_1, C_2, C_3	Barycentric map limiting state coordinates
d_M	Mahalanobis distance
f_x	Feature set, with case label x
g_i	Coefficient in Pope tensor basis expansion
$\bar{i}$	Impurity
k	Number of clusters
m	Misclassification rate
N_{proto}	Number of prototypes
$\mathcal{P}$	Turbulent production
Re_τ	Channel flow Reynolds number based on friction velocity and channel half-width
$\overline{S}_{ij}, \mathbf{S}$	Mean strain rate tensor
$\mathbf{T}_i$	Pope tensor invariant
u	Fraction of uncovered examples
u^+	Mean velocity normalized by friction velocity
$\mathbf{W}$	Mean rotation rate tensor
x_B, y_B	Barycentric map Cartesian coordinates
y^+	Distance from wall normalized by viscous length scale
δ	Prototype algorithm sphere radius
ϵ	Turbulent dissipation
$\eta_1, \eta_2, \ldots, \eta_5$	Scalar invariants of Reynolds stress, strain, and rotation rate tensor combinations
θ, ϕ, ζ	Angles defining relative tensor or vector coordinate system orientations
$\kappa_1, \kappa_2, \kappa_3$	Eigenvalues of the Reynolds stress anisotropy tensor
$\lambda_1, \lambda_2, \ldots, \lambda_5$	Pope scalar invariants

ξ	Data set example
$\tau_{ij} \equiv \overline{u_i' u_j'}$	Reynolds stress tensor
χ	Prototype penalty parameter
$\overline{\omega}_i$	Mean vorticity vector

I. Introduction

Recently, there has been significant research activity in development of fluid turbulence models using machine learning approaches. The models are typically trained using high-fidelity simulation data, from either direct numerical simulation (DNS) or sufficiently resolved large-eddy simulation (LES). Being data-driven, the models can only be trusted to reproduce the turbulent states and dynamics present in the training data (TD) and therefore it becomes imperative to identify those states in the TD if the model is to generalize (i.e., be used to simulate flows other than those in the TD) in a controlled fashion. Further, the TD should contain approximately equal proportions of examples of diverse turbulence physics, lest the model trained on it be biased against the dynamics poorly represented in the TD (see, *e.g.*, Ref. [1]). In this paper, we develop methods that could be used to gauge these qualities of the TD.

Machine-learned turbulence models have been constructed for closure terms in LES [2] and Reynolds-Averaged Navier-Stokes (RANS) models [3]. They have also been studied as a way of post-processing experimental measurements e.g., using convolutional neural nets to map particle images (from particle image velocimetry experiments) to the resultant velocity fields, with the velocity field training data generated via DNS [4]. Machine-learned closure models generally predict some turbulence quantity of interest, given some set of input variables or, in the machine learning parlance, *features*. For the RANS equations, the model may predict either the Reynolds stress [3], or a perturbation to a modeled Reynolds stress [5]. Feature selection is a well-studied topic in machine learning. In the context of regression models using supervised learning, the aim of feature selection is to identify a set of inputs to a model that leads to optimal outputs. In turbulence modeling studies to date, features for machine-learned models are often "hand-selected," using domain knowledge to guide the selection. It would be preferable to select features in a principled way, guided by theory, expertise and structures present in turbulence data sets, and this paper takes the first steps in identifying what such a procedure might be.

Another context for feature selection is clustering (see Ref. [6], Chp. 14) of unlabeled data using unsupervised learning approaches. Unsupervised learning is the branch of machine learning that seeks to find structure in a data set without recourse to labeling of data records (also called *examples*) and, thus, without the injection of exogenous information by an analyst. In this context, we seek a set of features that effectively divides turbulence data into distinct clusters. This clustering is performed with the aid of a metric (or *distance function*) defined in terms of flow-features. We

wish the clusters to conform, as much as possible, to our human understanding of turbulence, which requires a judicious selection of flow-features, and definition of the metric. This type of construct can be useful in turbulence modeling for the identification of canonical turbulent states. Moreover, in the analysis of turbulence models, this construct could be used to extract data points that conform to well-studied turbulent states, so that we can verify that the behavior of a data-driven turbulence model reproduces known relationships. Thus, unsupervised learning methods provide tools for automated identification of a data point (or example) as a member of a particular cluster of points with similar characteristics of the turbulent state. These tools can be useful for determining the completeness of a training data set; that is, whether the training data are sufficiently rich to produce models that are predictive for a given application. Further, if a data-driven turbulence model is evaluated at a *test* example that does not "belong" to a previously identified cluster, it is an indication that we may be extrapolating to a new region of physics not covered by the TD.

Note that since clustering is performed based on a metric, the segregation of examples into subsets may not be clear-cut, and it is quite possible to find outliers that could plausibly be assigned to other clusters. This can considerably complicate the calibration or validation of turbulence models, as the training/validation data would be inconsistent with the turbulence/physics being studied. Thus we seek a subset of examples that avoid outliers, i.e., are *representative* of the cluster and can be tested, via theory, if the clusters map to canonical turbulent flows. We call these representative examples *prototypes*, or *exemplars*, and describe a method to identify them from a clustered data set.

Unsupervised learning has previously been applied to unsteady vortical and turbulent fluid flow in a number of contexts. Cluster-based analysis of instantaneous flowfield snapshot data has been used to define a representative set of states to serve as a basis for a reduced order model [7, 8], to identify coherent structures or other key spatial features in the flow [9, 10], and to identify important nonlinear dynamical behaviors [11]. Features other than field data have been used to develop cluster-based analysis as well; for example, Nair *et al.* [12] used force coefficient features derived from LES of flow past an airfoil, and clustered into distinct dynamical regions, to derive optimal flow control laws for improved aerodynamic performance. Several studies have utilized unsupervised learning to partition turbulent flow data into regions with distinct physical behavior. Ser-Giacomi *et al.* [13] created flow networks from geophysical flow data, then used a network community detection algorithm to partition the flowfields into coherent, well-mixed regions with little fluid interchange between regions. Ali *et al.* [14] applied the k-means clustering algorithm to large eddy simulation data for wind farms, to extract physically significant flow regions based on the Reynolds stress anisotropy state. Callaham *et al.* [15] have developed the concept of data-driven balance models to partition regions of a flow (or other physical system) based on the local "dominant balance" at play. Each term in the governing equation at each location in space provides a data record with information on the relative importance of the terms; these data are clustered using a Gaussian Mixture Model, then a sparse Principal Components Analysis extracts the dominant balance of terms from each cluster. They have shown how this automated, data-driven procedure is able to correctly identify known asymptotic states in turbulent flows.

In the present study, we target the problem of clustering of turbulent flow statistical data sets, followed by the selection of examples that could serve as prototypes for clusters (called *prototype placement*). In Sec. II, we review the numerical algorithms used to assemble a feature-space and perform clustering. As our first task, in Sec. III, we formulate a set of features that could be used for clustering purposes, as well as the theoretical rationale for doing so. Thereafter, in Sec. IV, we apply clustering and feature selection algorithms to turbulent flow data to explore the feasibility and effectiveness of these algorithms. The candidate features are comprised of statistics that are typically available from DNS or LES data sets; most of the quantities are derived from invariants of the Reynolds-averaged strain rate, rotation rate, and turbulent stress tensors. We begin with a study of turbulent channel flow, as it is well characterized and has already been classified into distinct regions by theoretical analysis and empirical observations. We apply a feature selection algorithm that wraps around a Gaussian mixture model (GMM) clustering algorithm to identify a feature set that effectively clusters the data, and that can be reconciled with our existing domain knowledge of the regions of turbulent channel flow. We then consider several other two-dimensional turbulent flows, re-applying the clustering algorithm to these flows in sequential fashion to find new clusters that the channel flow data did not identify.

In our second task, in Sec. V, we formulate prototype placement as a set-cover problem [16] and test the efficacy of a solution algorithm for its applicability to turbulent data sets. We define figures of merit that measure the quality of a set of prototypes, allowing us to trade-off simplicity of representation (i.e., a small number of prototypes) against the fidelity of representation.

Finally, in Sec. VI, we illustrate some practical ramifications of our clustering study, *viz.*, how flow simulation data sets could be assessed for inclusion in a training data set for data-driven turbulence models. We also illustrate how prototypes can be used to study and characterize the turbulence in individual clusters, and thus "label" them by the type of physics they contain. Together, they are used to assess the quality of a TD that could be assembled by simply pooling the flow data sets used in this paper.

Fig. 1 summarizes the process followed in this paper. In Sec. III andIV we develop the principles for partitioning a simulation data set into clusters with similar types of turbulent processes. In doing so, we also find flows that contain much the same physical (turbulent) processes that have been encountered in the flows studied before. In Sec. V we develop the principles of summarizing simulation data sets with a handful of prototypes, which are then used to interpret the physical contents of the training data set (TD) and identify the missing physics (Sec. VI). The missing physics, in turn, qualifies any data-driven model trained on the TD.

Turbulence data sets are large and unwieldy, often containing distinct regions with different turbulent processes. This work begins to lay a foundation for semi-automated isolation and classification of turbulent flow states, which is the first contribution of this paper. We also show how prototypes, selected from clusters, can be used to identify the turbulent processes that exist in a turbulence data set and therefore could be learned from it by a data-driven model. This is the second contribution of our paper. Together, the two advances assist in the assembly of well-balanced training

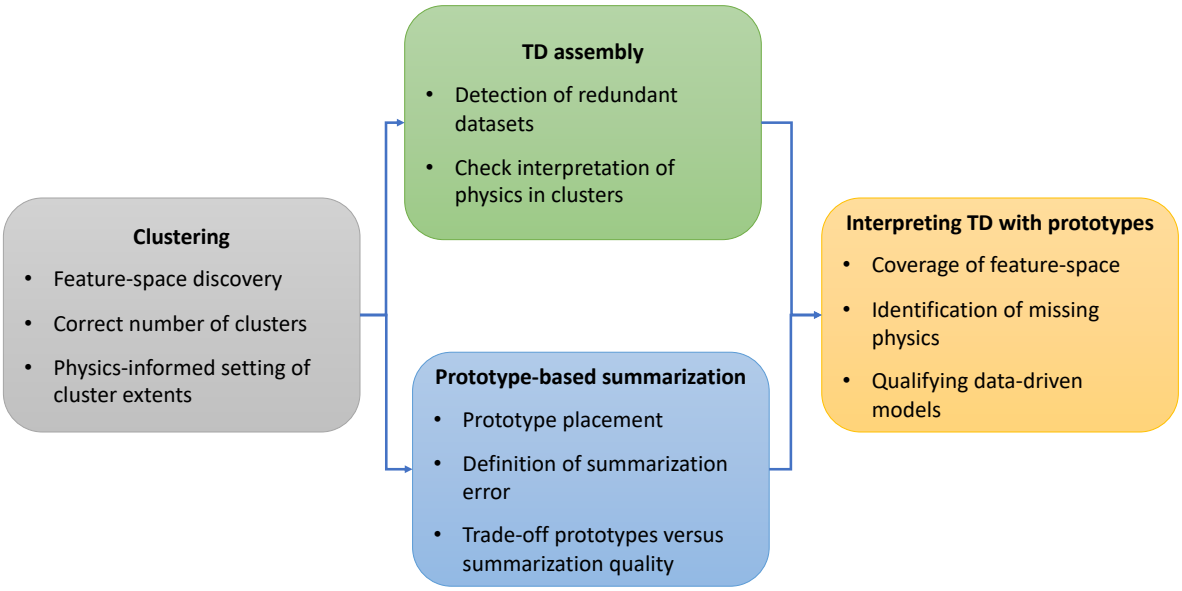

Fig. 1 Schematic of the process followed in this paper. We start with clustering (left) which allows us to identify redundant flow data sets (upper half). The lower half concerns itself with rendering the flow data sets interpretable (via prototype selection) which helps us qualify the training data (TD) and models trained on it.

data sets, which are necessary for learning accurate and generalizable data-driven turbulence closures.

II. Clustering and Feature Selection Algorithms

The aim of the present work is to identify, from a set of candidate flow state features, a subset of features that successfully clusters turbulent flow states. While many metrics have been devised to compare the performance of clustering algorithms, defining success in a clustering application can be challenging. Here, we loosely define success as automated partitioning of the data into flow-field regions that reconcile with our human understanding of turbulent flow physics. We will rely heavily on the plane channel flow case to select parameters for the clustering and feature selection algorithms since, for this flow, where the single-point statistics vary only in a single coordinate direction, we have a good concept of how the flow should be divided into regions based on empirically supported theory.

To perform this unsupervised learning task, we use the Feature Subset Selection (FSS) algorithm described in Dy and Brodley [17]. The FSS algorithm is an example of a wrapper approach for unsupervised learning, where a search for optimal feature subsets is wrapped around a clustering algorithm. There are three tasks involved in the wrapper approach: the feature search, the clustering algorithm, and the feature subset evaluation [17].

For the search, we use a sequential forward search that starts with a small number of features (or no features) and adds one feature at a time. The feature that is added maximizes some chosen (scalar) evaluation criterion. This leads to

a maximum complexity of the search that is $O(d^2)$ for d features, whereas an exhaustive search would need to evaluate 2^d possible feature subsets. The sequential search stops when adding more features no longer improves the evaluation criterion.

The clustering algorithm used here is the soft Gaussian mixture model algorithm [18]. The GMM algorithm approximates the probability distribution function of the data using a finite mixture of multi-variate Gaussian distributions. The parameters of the Gaussian mixture are calculated by a maximum likelihood fit, using the Expectation-Maximization algorithm [18]. We used the Matlab function `fitgmdist()` for this calculation. The k-means++ algorithm [19] is used to seed the initial Gaussian component means. The output of the k-means++ algorithm is not completely deterministic and the problem of finding local optima is mitigated by replicating the fitting calculation forty times, and selecting the best fit from the replicates.

A data point can belong to more than one cluster, with a probability assigned to each point/cluster pair. The number of clusters, k, is an input parameter for the GMM algorithm. For each candidate feature subset considered, a sweep over number of clusters from 2 to k_{max} is performed in order to identify an optimal number of clusters. In this work we set $k_{max} = 9$. The metric used to evaluate the suitability of the number of clusters is Bayes' Information Criterion (BIC), also known as the Schwarz Information Criterion (SIC) [20]; we seek the fit that minimizes the BIC. For our turbulence data sets, we observed that the BIC tended to favor large numbers of clusters (ten or more). Its value often decreased rapidly with k initially, then more gradually as k became larger. We adopted a selection criterion that the BIC for $k + 1$ clusters, denoted $BIC(k + 1)$, must be less than $BIC(k) - 2\sigma_k$, where σ_k is the standard deviation of BIC values over the ensemble of forty GMM trials using k clusters. In this way, we demand that the benefit of adding an additional cluster be clear relative to the variation in results over the trials for one less cluster.

We follow Dy and Brodley [17] and use the scatter-separability metric as the feature subset evaluation criterion. This metric is larger (more optimal) when the distances between samples within a cluster are small (low scatter) and when the cluster means are far apart (separated). The metric is also invariant to linear transformations of the features. Cross-projection is used to remove the bias of the scatter-separability metric towards larger feature sets, when comparing two feature sets of different size [17].

All clustering calculations were performed in serial, in Matlab. The computational cost of the sequential search algorithm scales with d^2, so for large candidate feature vectors the search could become costly. For the relatively modest number of features considered here, computations were not onerous, with the feature selection taking less than one hour in each case. More efficient search algorithms, in addition to parallelization of the search and/or clustering operations, could be pursued for larger problem sizes.

III. Candidate Features for Turbulent Flow

We must begin with some candidate features that are readily available, satisfy Galilean and rotational invariance, and preferably play some role in our current understanding of turbulence. In RANS, we typically have available the mean flow velocity, pressure, temperature, and density fields, the spatial gradients of these quantities, as well as the Reynolds stress as provided by the turbulence model. Ignoring temperature and density variations for the moment, we thus have statistical quantities in the form of two symmetric tensors: the mean strain rate tensor $\overline{S_{ij}}$ and Reynolds stress tensor τ_{ij}; and three vectors: the mean velocity vector, the vorticity vector (which can be represented using the rotation rate tensor), and the pressure gradient vector. The velocity vector and pressure gradient vector depend on the particular reference frame, *i.e.* they are not Galilean invariant, and so are not used to derive features. Invariant features can be constructed from the other quantities, as summarized in the following. A more complete description of the features is contained in Barone *et al*[21]. Other features that have not been used here may prove useful for classification, should they be available in both training data sets as well as accessible in a RANS model. For example, intermittency factor, a measure of the level of transition from a laminar to turbulent state [22], could be a useful feature when the turbulence model includes a transition model.

Barycentric Coordinates

The degree and type of anisotropy present in the turbulent stress is described by the barycentric map [23]. The normalized Reynolds stress anisotropy tensor is $b_{ij} = \frac{\overline{u_i' u_j'}}{2\overline{k}} - \frac{1}{3}\delta_{ij}$. The eigenvalues of the anisotropy tensor, κ_i, are first computed and ordered according to $\kappa_1 \geq \kappa_2 \geq \kappa_3$. The eigenvalues are then transformed to two coordinates within an equilateral triangle via a linear mapping:

$$C_1 = \kappa_1 - \kappa_2, \quad C_2 = 2\,(\kappa_2 - \kappa_3), \quad C_3 = 3\kappa_3 + 1 \tag{1}$$

$$x_B = C_1 x_1 + C_2 x_2 + C_3 x_3 \tag{2}$$

$$y_B = C_1 y_1 + C_2 y_2 + C_3 y_3 \tag{3}$$

Here, (x_1, y_1) are the two-dimensional coordinates of the "one-component" vertex of the triangle, (x_2, y_2) are the coordinates of the "two-component" vertex, and (x_3, y_3) are the coordinates of the "three-component" vertex. The limiting state coordinates C_1, C_2, C_3 are the weights of each of these three limiting anisotropy states associated with the triangle vertices.

Typically, the barycentric map is plotted on an equilateral triangle with vertices at, for example, $(0, 0)$, $(1, 0)$, and $(0, \sqrt{3}/2)$. The vertices correspond to physical anisotropy states of turbulence. The right vertex corresponds to

one-component turbulence, where turbulent fluctuations act only in one direction. This state is approximately achieved in the buffer layer of turbulent channel flow. The left vertex corresponds to the two-component axisymmetric limit, where turbulent fluctuations are active in two directions, but not in the third. The top vertex corresponds to the homogeneous limit, where turbulent fluctuations are omnidirectional. Other physical situations can be described by lines in the barycentric map, such as axisymmetric contraction, axisymmetric expansion, and plane strain [23].

Angles Between Tensors and Vectors

The strain rate and Reynolds stress tensors are symmetric and, therefore, can be diagonalized to reveal a set of principal axes, defined by the set of orthonormal eigenvectors of the tensor. Thus, each tensor defines a coordinate system, and one can describe their relative orientation by describing the angular coordinates of one tensor in the other tensor's coordinate system. Three angles are required for this purpose. The same can be done to describe the orientation of a vector relative to a tensor, for which two angles are required. These angles can be useful features for classifying a turbulent flow state because they are coordinate-system invariant (at least for the quantities involving velocity gradients), and they are naturally scaled, $O(1)$ quantities.

This development follows the eigensystem ordering conventions of Tao *et al.* [24], described in the present context in [21]. The relative orientation of the coordinate systems implied by two tensors is described by a collection of three angles: θ, ϕ, and ζ. Likewise, the orientation between a vector and the coordinate system implied by one of the tensors can be described by two angular coordinates: θ and ϕ. Given the tensors $\overline{S}_{ij}$ and τ_{ij}, and mean vorticity vector $\overline{\omega}_i$, we can calculate seven angles that can be used to describe the alignment of the tensor-tensor and tensor-vector eigensystems.

Some of these angles can be assigned a physical interpretation. For example, the production term in the enstrophy evolution equation can be written in terms of the alignment of the principal strain directions with the vorticity. The intermediate strain eigenvector is often preferentially aligned with the vorticity, although the degree of alignment can vary depending on instantaneous turbulent structure [25]. Presumably, the alignment of the mean strain intermediate eigenvector and mean vorticity also vary depending on the location within an inhomogeneous turbulent flow. Other angles contain information relevant for turbulence modeling. For example, the commonly invoked Boussinesq approximation assumes alignment of the turbulent stress with the local strain. Angles between the principal stress and strain directions give a measure of the degree to which this assumption is violated. We hypothesize that these angles may also help differentiate between different turbulent flow states.

Scalar Invariants

A general polynomial expression relating the Reynolds stress anisotropy tensor to the mean rate of strain and rate of rotation tensors is given in Pope [26]. This ten-term expression is*

$$\mathbf{b} = \sum_{i=1}^{10} g_i(\lambda_1, \lambda_2, \ldots, \lambda_5)\, \mathbf{T}_i. \tag{4}$$

The coefficients of this expansion, g_i, are functions of the following scalar invariants of the strain rate tensor, $\mathbf{S}$, and rotation rate tensor, $\mathbf{W}$.

$$\lambda_1 = \{\mathbf{S}^2\}, \quad \lambda_2 = \{\mathbf{W}^2\}, \quad \lambda_3 = \{\mathbf{S}^3\}, \quad \lambda_4 = \{\mathbf{W}^2\mathbf{S}\}, \quad \lambda_5 = \{\mathbf{W}^2\mathbf{S}^2\} \tag{5}$$

Truncated forms of Eq. 4 can be used, which contain a subset of terms. For example, Schmitt and Hirsch [27] used a four-term expansion, retaining only terms $\mathbf{T}_i$ that are linear or quadratic in $\mathbf{S}$ and $\mathbf{W}$. This enabled them to solve for the coefficients in Eq. 4 as algebraic expressions involving both the set of invariants λ_k, as well as additional invariants that involve the anisotropy tensor:

$$\eta_1 = \{\mathbf{b}^2\}, \quad \eta_2 = \{\mathbf{bS}\}, \quad \eta_3 = \{\mathbf{bSW}\}, \quad \eta_4 = \{\mathbf{bS}^2\}, \quad \eta_5 = \{\mathbf{bW}^2\} \tag{6}$$

The set of five invariants, $\lambda_k, k = 1, \ldots, 5$, in addition to the five invariants, $\eta_k, k = 1, \ldots, 5$, together provide a set of ten scalar quantities that contain information on the local stress-strain relationship in a turbulent flow. The λ_k invariants only describe the mean velocity gradient tensor and, as such, do not directly contain any information about statistics of the turbulence itself. The η_k invariants are formed from combinations of the strain, rotation, and anisotropy tensors, and thus contain information about both the mean velocity gradients as well as the turbulent stress. We do not attach any particular physical significance to any of these invariants, except inasmuch as they describe the local stress-strain relationship in a manner that respects coordinate-system invariance.

In their dimensional form, these scalar invariants can vary over orders of magnitude from invariant to invariant within the same flow, or for the same invariant across different flows with varying length and time scales. They are much more useful as descriptors of a turbulent flow state when they are cast in non-dimensional form. We use the local mean turbulence kinetic energy and mean turbulent dissipation to nondimensionalize these features. For our DNS data sets, we did not have the dissipation available for all the data, so we used a linear eddy viscosity relation to estimate dissipation [21].

*Here we employ matrix notation, representing *e.g.* the two-dimensional tensor S_{ij} using the matrix symbol $\mathbf{S}$.

Additional Candidate Feature

Other ad-hoc features may be useful for classification of turbulent states. The ratio of turbulent production to turbulent dissipation rate, $\mathcal{P}/\epsilon$ is often used to characterize turbulent flows. It is approximately equal to unity in the logarithmic layer of an equilibrium turbulent boundary layer, for example, and takes on other values in different regions of turbulent flows.

Summary of Candidate Features

There are a total of twenty-one candidate features: the three limiting state barycentric map coordinates, seven tensor-tensor and tensor-vector angles, ten scalar invariants, and the ratio of turbulent production to dissipation rate. We consider only two-dimensional turbulent flows in the present study, with a homogeneous third dimension. In this case, only one of the angles is non-trivial - $\theta_{S-\tau}$ - and there are then only 15 candidate features.

We note that the choice and scaling of candidate features is arbitrary. We are guided by physical principles, such as choice of invariant features that are non-dimensionalized by the local turbulence time scale, where appropriate, in order that the features generalize across many flow of interest. The scatter separability metric for determining suitability of features for clustering has the important property that it is invariant to arbitrary linear scaling of any of the features, such that the feature selection process itself is not sensitive to such scaling. However, the collection of selected features remains a heuristic. Norms in the feature space do not possess a physical significance and, as such, we have no guarantees that other features or other scalings may not perform better for developing a balanced training data set for turbulence models. We must rely on performance of the clustering on data sets with known partitioning (on theoretical grounds), such as the channel flow described in detail in the next section, to make such assessments. The ultimate test, not covered in the scope of this paper, will be to demonstrate that a turbulence model trained on a data set balanced using the present approach performs better than a naively constructed training data set.

IV. Clustering Results

The present study analyzes simulation data from four two-dimensional turbulent flows: plane channel flow at five Reynolds numbers [28], a wavy-walled channel at a bulk Reynolds number of 6850 [29], a bump in a channel [30] at $Re_\tau \approx 600$, and a square cylinder in cross-flow at Reynolds number of 21,400. Figure 2 shows mean stream-wise velocity fields for each of the latter three flows; these plots also illustrate the sampling of the fields, with one symbol plotted for each sampled data point.

We initially attempted to pool all of the training data from the four flow cases into one data set and performed feature selection and clustering on the pooled data. These initial experiments did not result in clusters that we could interpret in a meaningful way, so we took the approach of applying the classification in a flow-by-flow manner, as described in the following sections.

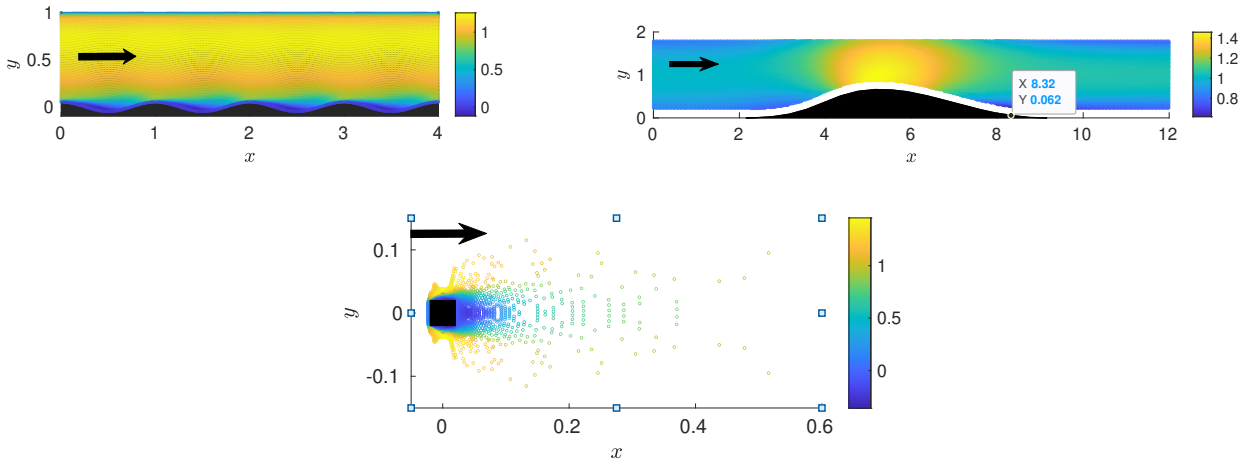

Fig. 2 Mean stream-wise velocity fields, normalized by a reference velocity, at sampled data points for the wavy wall, bump, and square cylinder flows. Arrows indicate bulk flow direction.

Plane Channel Flow

We first investigate the performance of clustering and feature selection on a single flow: plane turbulent channel flow. Channel flow is one of the most well-studied and well-characterized turbulent flows, and therefore serves as a good case with which to judge the machine learning techniques. In other words, the clustering results should be consistent with our existing understanding of this flow. If they are not, then it is likely not worth pursuing the application of these clustering techniques to more complex flow situations.

We apply the clustering algorithm to DNS data for plane channel flow at five different Reynolds numbers: $Re_\tau = 180, 550, 1000, 2000$ and 5200 (the data sets are described in [28]). For clustering and feature selection, we operate on a single channel flow data set that includes the data from all five Reynolds numbers. All of the proposed candidate features are available in this data set, including the dissipation rate for calculation of the ratio of production to dissipation and for non-dimensionalization of the velocity gradients. We make the choice to populate the feature set with an initial set of features consisting of the barycentric coordinates, either in Cartesian form (x_B, y_B) or limiting state form (C_1, C_2, C_3). We first apply the GMM algorithm to cluster the data based only on the barycentric coordinates; the results are shown in Figure 3. The results are presented as mean velocity profiles, with individual data points colored by the most likely cluster for each point. The known regions of turbulent channel flow are easily identified from the mean velocity profiles: the viscous sublayer ($y+ \lessapprox 5$), the buffer layer ($5 \lessapprox y+ \lessapprox 30$), the logarithmic layer ($30 \lessapprox y+,\ y/H \lessapprox 0.15$), and the outer layer ($y/H \gtrapprox 0.15$). Accordingly, we choose initially to set the number of clusters equal to four. The barycentric coordinate feature sets allows for reasonable clustering of the data into four regions that approximately resemble the known regions of turbulent channel flow. However, there is a lack of sharp distinction between the viscous sublayer region and the buffer layer, and the Cartesian coordinates result in a viscous sublayer that ends slightly early, whereas the limiting state coordinates result in a viscous sublayer that extends too far

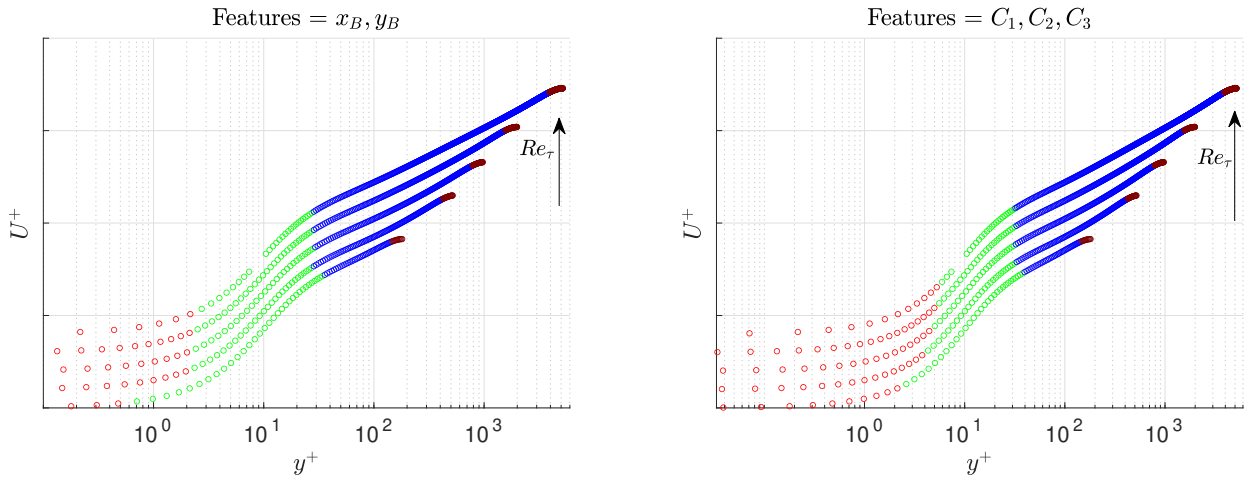

Fig. 3 Clustering results for turbulent channel flow. Velocity profiles are vertically offset from one another for clarity. Left: Cartesian barycentric coordinate feature set (x_B, y_B). Right: Limiting state barycentric coordinate feature set (C_1, C_2, C_3).

away from the wall.

We then ran the FSS algorithm to augment the feature set with further useful features, and to automatically find the optimal number of clusters. We found that we first needed to filter the data to remove anomalous outlier points. Certain tensor angle features were especially problematic. For example, the mean strain rate tensor becomes very small (theoretically zero) at the channel centerline, causing angles between the mean strain rate and Reynolds stress tensors to become ill-defined. We found that removing data points that had features which lay greater than eight standard deviations away from the mean value was sufficient. This resulted in only one percent of the data being eliminated. The optimal feature set identified by the algorithm includes four new features and five clusters; the complete optimal feature set, denoted f_{ch}, is: $f_{ch} = \{C_1, C_2, C_3, \eta_1, \lambda_1, \eta_4, \eta_3\}$. The resulting clusters are shown in Figure 4. The clustering is remarkably consistent with our prior knowledge of turbulent channel flow regions. Note that no explicit information on distance from the wall, or the wall shear stress typically used for inner scaling, was provided to the algorithm. The boundaries of the regions are in approximately the correct locations, and the boundaries are "sharp," with little overlap between the clusters.

Upon initial examination, the selection of five clusters seems to be inconsistent with the conventional categorization of four regions of channel flow. However, the buffer region is in actuality simply a transition region between the viscous sublayer and the logarithmic layer, and we do not have much in the way of theory to suggest that the buffer region turbulence has uniform characteristics throughout. The clustering algorithm has split the buffer region into two regions (cluster ID's 2 and 3), with the boundary at $y^+ \approx 25$. The algorithm has also placed the boundary between the viscous sublayer (cluster ID 5) and the buffer regon at $y^+ \approx 2.5$, which seems to contradict the usual demarcation of $y^+ \approx 5$. Figure 4 shows, however, that the theoretical curve $u^+ = y^+$ for the mean velocity profile in the viscous sublayer is

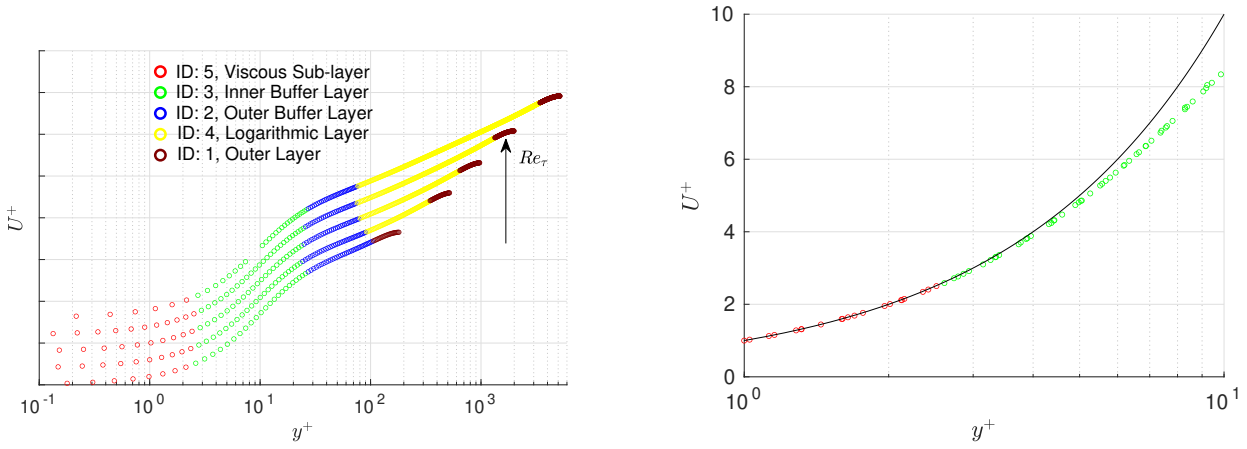

Fig. 4 Clustering results for turbulent channel flow with feature set including the limiting state barycentric coordinates and the invariants $\eta_1, \lambda_1, \eta_4, \eta_3$. On the left, velocity profiles are vertically offset from one another for clarity. For reference, the theoretical profile for the viscous sublayer is shown as a solid black line.

accurate to within three percent at $y^+ = 5$, but strictly speaking begins to depart from the DNS closer to the wall. At $y^+ = 2.5$, the error relative to the theoretical profile is close to one percent; thus, the cluster boundary near $y^+ = 2.5$ is not unreasonable. We are re-assured that there is a distinct cluster (ID 4) associated with the inertial, or logarithmic layer, of the flow. Interestingly, the lowest Reynolds number case, at $Re_\tau = 180$, does not contain any points in this cluster. This is consistent with previous observations that channel flow does not exhibit a clear logarithmic layer at this low Reynolds number, while we also note that entirely precise identification of a logarithmic layer has remained somewhat elusive for all DNS results to date. The cluster with ID 1 groups the outer layer points, despite the distribution of these points across a broad range of y^+ values.

Overall, the clustering produced by the FSS algorithm is consistent with our existing knowledge of channel flow turbulence, lending some confidence that this technique can group data points by a physical state that is, in some way, meaningful.

Wavy Wall

The next flow we consider is the flow in a plane channel with a wavy bottom wall. Again, as with all the flow cases considered in this paper, the geometry is two-dimensional with a homogeneous lateral dimension. We wish to completely utilize the clusters already obtained with the channel flow data, so we first assign clusters to the wavy wall data points that are well-classified by the channel flow clusters. We use the Mahalanobis distance d_M (see Ref. [6], Chp. 12) to determine the distance from each wavy wall data point to each of the channel flow cluster means. For the channel flow data, the vast majority of the data points satisfied $d_M < 25$, so we applied this threshold to the wavy wall data. The procedure followed was: 1) calculate the Mahalanobis distance from each wavy wall data point to each channel flow cluster, d_M^k; 2) determine the nearest channel flow cluster based on the minimum distance $d_{M_{min}}^k \equiv \min_k d_M^k$; 3)

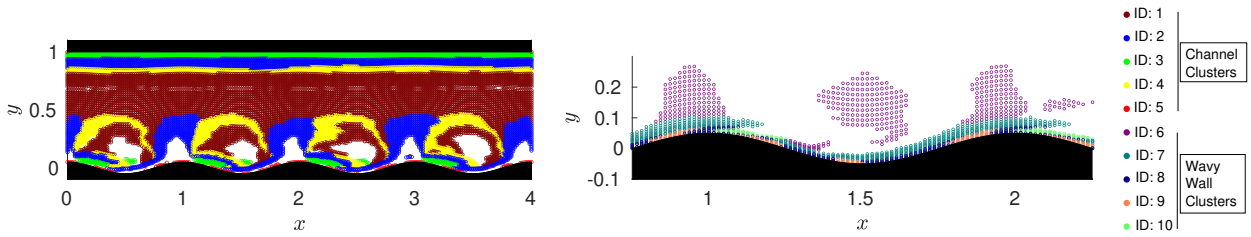

Fig. 5 Left: Wavy wall data that is well-classified by the channel flow clusters, colored by cluster. Right: Wavy wall clusters, showing only one periodic segment of the domain.

if the minimum distance satisfies $d^k_{M_{min}} < 25$, then the data point is assigned to the k^{th} channel cluster; 4) apply the FSS algorithm to the remaining unclassified wavy wall data points to identify new wavy wall clusters; 5) group the remaining wavy wall data points using the new clusters.

Figure 5 shows the wavy wall data points that are well-classified by the channel flow clusters. Virtually the entire top half of the domain is well-classified by the channel flow clusters, as well as many points near, but not immediately adjacent to, the wavy wall surface. Many of the points associated with the region of flow separation downstream of the hump apexes are classified by channel cluster 3, which is associated with near-wall buffer layer points. This is due primarily to both regions being characterized by one-component anisotropy (observed also in Emory and Iaccarino [31]).

The FSS algorithm was run on the remaining data, again beginning with an initial feature set $\{C_1, C_2, C_3\}$, resulting in an optimal clustering with five clusters and an optimal feature set $f_{ww} = \{C_1, C_2, C_3, \eta_1\}$. The clusters for one periodic segment of the wavy wall domain are shown in Figure 5. There are two near-wall clusters (cluster ID's 8 and 9) which alternately appear along the wall surface direction. Interestingly, these do not correspond to distinct regions based on the sign of the stream-wise pressure gradient. Two other clusters (cluster ID's 6 and 7) group points lying just outside this near-wall region, while a fifth cluster (ID 10) classifies a small region of near-wall flow just downstream of the hump apex.

Further visualizations reveal connections between the spatial distributions of the features and the clusters. For the wavy wall flow, the barycentric map coordinates play an important role. The primacy of the barycentric coordinates in determining the clusters for the wavy wall data is demonstrated in Figure 6. Here, the points are colored by position within the barycentric triangle, following Emory and Iaccarino [31]. The clusters from Figure 5 correspond closely to the position within the barycentric map. This is not surprising since only one additional feature, η_1, is employed in the wavy wall clustering. The two near-wall clusters that alternately appear along the wavy surface are largely determined by the stream-wise and span-wise normal stress components $\overline{u'u'}$ and $\overline{w'w'}$, respectively. These two components are much larger than $\overline{v'v'}$. In cluster 9, which appears just upstream of the apex, on the lee side of the wave, and in the trough of the wave, $\overline{u'u'} \approx \overline{w'w'}$, and the anisotropy state is close to the two-component limiting state (lower left corner of the barycentric map). Cluster 8 appears at the apex, on the lee side of the wave downstream of the inflection point,

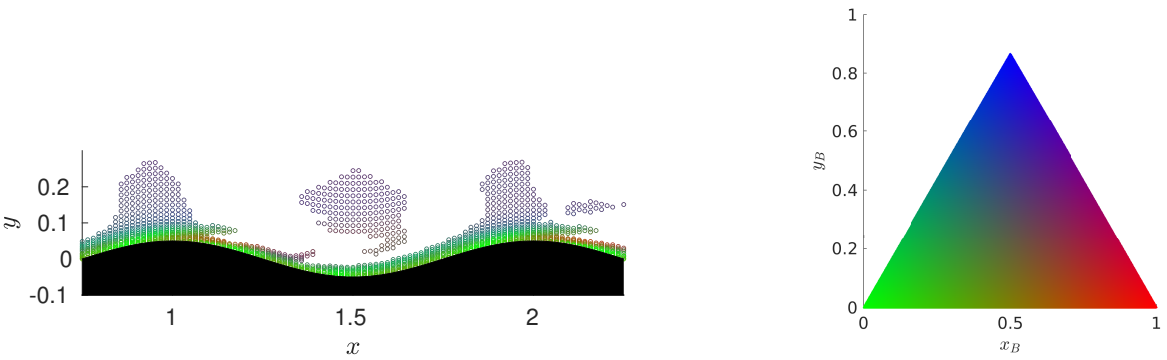

Fig. 6 Wavy wall clustered points, colored by position within the barycentric map.

and from the re-attachment point up to the inflection point on the windward side of the wave. This cluster is associated with an anisotropy state between the one-component and two-component limiting states, where two components of the stress are active, but one is larger than the other. For the first two regions associated with this cluster, $\overline{u'u'} > \overline{w'w'}$. For the latter location (downstream of re-attachment), the anisotropy state is similar but now $\overline{w'w'} > \overline{u'u'}$ (see also [32]).

Contours of η_1 are shown in Figure 7. It appears that η_1 is useful in differentiating cluster 10 from Figure 5, which describes points just downstream of the apex of the wavy surface, where the boundary layer has separated. The invariant η_1 is a scalar measure of the degree of anisotropy. Its relatively large value in this separation shear layer cluster reflects the dominance of the streamwise normal stress component $\overline{u'u'}$. Likewise, cluster 7, which lies just above the two near-wall clusters described previously, is associated with a lower value of η_1 indicating a relatively low degree of anisotropy, and a position within the barycentric map closer to the three-component, isotropic, limiting state.

In summary, the data points from the wavy wall data set that are not already well-classified by the channel flow data, correspond largely to the region of the flow-field close to the wavy wall. It is observed (but not explicitly shown here) that even points in the near-wavy-wall region with similar anisotropy state as one of the channel clusters, have remaining features from $f_{ch} - \eta_1, \lambda_1, \eta_4, \eta_4$ – that differ from the near-wall channel flow clusters, explaining why these points are not classified by the channel clusters. A different optimal feature set, f_{ww}, is found to classify these points which is, interestingly, a subset of the optimal feature set for channel flow. This can be explained partly by the observations, noted above, that the new near-wavy-wall clusters are largely differentiated by the anisotropy state of the turbulence, which is described by the barycentric map coordinate features C_1, C_2, C_3. The additional feature in f_{ww}, η_1, appears useful for distinguishing the separation shear layer from surrounding flow regions.

Bump in Channel

The third flow considered is a two-dimensional bump in a channel. The available data set for this flow did not include data near solid surfaces; the minimum distance from the closest point to a wall was about twenty percent of the channel half-width. The channel flow clusters were able to classify 46.3 % of the bump-in-channel data points, while the wavy

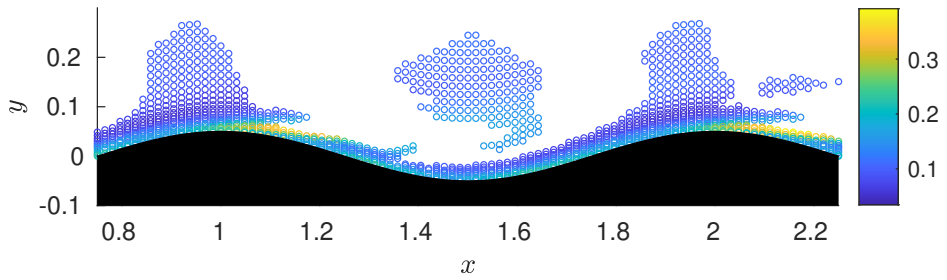

Fig. 7 Contours of scalar invariant η_1 for the wavy wall clustered points.

wall clusters classified a further 49.8 %, leaving only 3.9 % of the data points left to be clustered. It is re-assuring that the previous clusters were good fits for much of the bump flow-field. The remaining points were run through the FSS algorithm, with an additional six clusters found as optimal, and the best feature set was $f_b = \{C_1, C_2, C_3, \cos\theta_{S-\tau}, \lambda_3\}$. It is notable that the bump feature set is the first set to include the stress-strain tensor angle. For brevity, we do not show the bump clusters here (see Supplementary Online Material for more details).

Square Cylinder

The fourth flow considered is a square cylinder in cross-flow at a Reynolds number of 21,000 based on cylinder width[†]. Despite the significantly different flow topology relative to the previous three cases, a large number of points are well-classified by the existing clusters identified from those cases. In fact, the Mahalanobis distance threshold used for the square cylinder case was lowered from 25 to 15 to allow for an adequate number of data points to identify new clusters. With this threshold, 15 % of the square cylinder points were classified by channel flow clusters, while 63 % were classified by the wavy wall points. These points are visualized in Figure 8. Of interest is the large number of points in the wake region that are well-classified by wavy wall cluster 6. This cluster was associated with wavy-wall points between the near-wavy-wall and outer regions, see Figure 5. These wake points have a similar anisotropy state to the identified region of the wavy wall flow, but otherwise a physical interpretation of this cluster is not clear. Only three points were classified by the bump clusters. The remaining points were grouped into three clusters by the FSS algorithm, with optimal feature set $f = \{C_1, C_2, C_3, \theta_{S-\tau}\}$. Although the optimal feature set for the square cylinder clusters is a subset of that for the bump clusters, the bump clusters occupy a different region of feature space than the square cylinder clusters.

The square cylinder clusters are shown in Figure 9. Cluster 19 is comprised of points mainly located around the forward corners of the cylinder and in the early separtion shear layers, a high strain-rate region where the flow accelerates around the corner; note that the flow in this region has likely not fully transitioned to turbulence. Its position in the barycentric map is close to the one-component limiting state. The two other clusters (ID's 17 and 18) are comprised of points that mainly lie within the mean position of the shear layers above and below the cylinder, although with some

[†]Results obtained from S. Arunajatesan, personal communication

points also in the near-wake region. Points in these two clusters lie along the edge of the barycentric map connecting the one-component and three-component limiting states. Note that Pope has reported that DNS results for the central region of a turbulent mixing layer show similar anisotropy states [33]. These two clusters are differentiated by the feature describing the angle between the strain rate and Reynolds stress principal directions. This is somewhat of an artifact of how the principal direction is defined (the eigenvector associated with the most extensive, or positive, eigenvalue). This angle can change suddenly when the ordering of stress eigenvalues changes, which is the case here. This may motivate a search for a different feature describing alignment of stress and strain that is less sensitive to perturbations in those tensors.

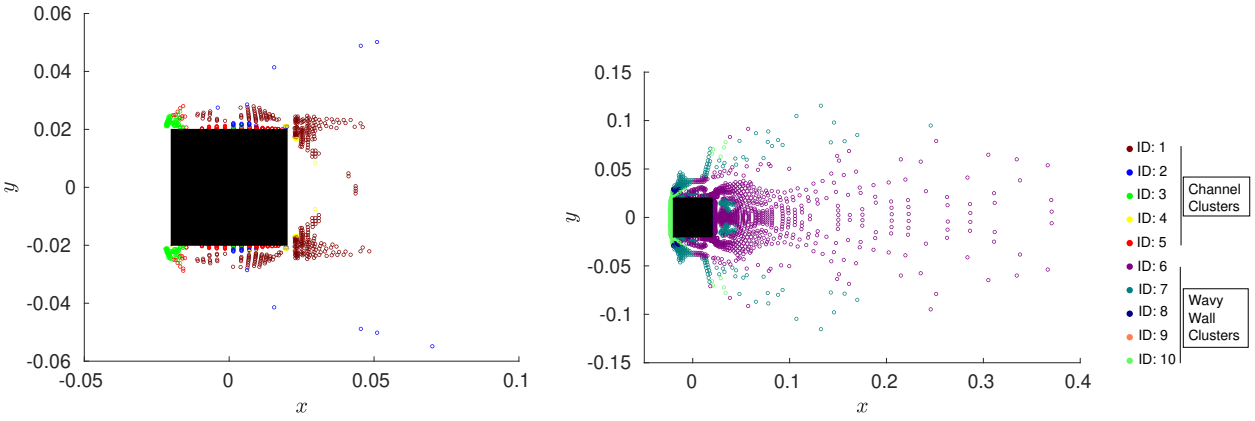

Fig. 8 Left: Square cylinder data that is well-classified by the channel flow clusters, colored by cluster. Right: Square cylinder in channel data that is well-classified by the wavy wall clusters, colored by cluster.

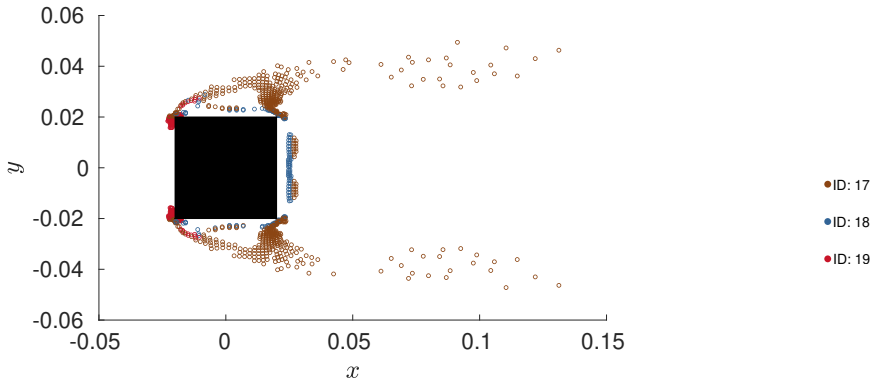

Fig. 9 Square cylinder data clusters.

V. Prototype Placement

An example in a data set, ξ, is a single data record, consisting of an n-dimensional vector of features. Prototype placement (also known as selection of exemplars) is the selection of a subset of examples from a data set that can adequately summarize it. It yields a tractably small data set that can be used to interpret and understand a larger data set.

Prototype placement can be performed in an unlabeled data set or in a data set where the examples have been "colored", i.e., where they have been labeled/categorized via clustering, as in our case (see Sec. IV). Below, we describe some prototype placement approaches and apply them to our turbulence data sets, to extract a subset. The distribution of the prototypes provides an approximate measure of the variation of the features in space.

Concepts similar to prototype placement have arisen in fluid dynamics research. The use of cluster-based and network-community-based reduced-order models (ROMs) for flow control has been explored [8, 12, 34]. Here, one models the flow discretely as a set of vortices arising, for example, from the disaggregation of coherent structures in turbulent flows. For computational ease and speed, the vortices are collated into clusters and some aerodynamic variable e.g., drag, is modeled by representing each cluster in a simplified manner. If the simplified model for a cluster has to be associated to a point in the feature-space, the cluster-centroid is generally employed for the purpose. This is similar to a prototype conceptually, but the details are different. A prototype would ensure that the location in feature-space would correspond to one of the members of the cluster, but there are only a handful of scenarios where this would be helpful (see Sec. VII). Prototypes would also quantify how well the chosen locations summarize the clusters, but it is unclear how this information would be used to improve the ROMs. Thus prototype placement, though somewhat allied to the construction of network-community ROMs, has not been pursued in that field.

Sensor placement is another field where prototype-like concepts have been explored [35]. Here, the aim is to reconstruct spatiotemporal fields e.g., a velocity field downstream of an object in a flow, by measuring it at a few sensor locations. These studies employ CFD (computational fluid dynamics) solutions of the fields on a grid, and have to be approximated with the help of orthogonal bases (obtained via Proper Orthogonal Decomposition) and noisy measurements in a few grid-cells. QR-decomposition with column pivoting has been used to isolate the most influential grid-points for constraining the reconstructed fields [35], with impressive results. However, the objective in sensor placement is quite different from prototype placement. Whereas sensor placement seeks to reconstruct the target field as accurately as possible (while minimizing the number of sensors), in prototype placement, the field has to be represented only so well that reconstruction errors of a feature vector does not change its class label (i.e., the cluster ID). This, of course, is irrelevant in sensor placement, and so that body of work has ignored prototypes. On the other hand, it is difficult to conceptualize how prototype selection could be cast as a sensor placement problem. To start with, it is difficult to formulate a QR decomposition with a categorical variable (the class label), though it would certainly require a classifier to map an imperfectly reconstructed feature vector into its class label, when optimizing sensor locations. In addition, proper orthogonal decomposition (which is based on singular value decomposition) scales as $N_{examples} \times d^2$ (where $N_{examples}$ is the number of feature vectors and d is the length of the feature vector) which can get large for DNS data sets. Thus sensor-placement concepts are not quite the same as those in prototype placement in labeled data.

A. Prototype Placement

There are three different prototype placement scenarios, each with its own set of algorithmic solutions, *viz.* when features are unlabeled but can be projected to a low-dimensional space, when they are unlabeled and intrinsically high-dimensional and lastly, when they are high-dimensional but can be labeled (or "colored"). Methods exist to place prototypes in unlabeled data, for both intrinsically low-dimensional [36–42], and high-dimensional feature spaces [43, 44]. Our problem, however, involves placing prototypes among examples that have been labeled by the cluster IDs.

Ref. [45] describes an algorithm by which prototypes can be placed in a colored data set and is the method used in our study. The method is meant for problems where (1) ξ can be represented by a point in a high-dimensional continuous feature-space and (2) the ξ can be "colored" or labeled by their cluster ID (signifying a particular type of turbulence that they represent). One starts with the assumption that all ξ could potentially be prototypes. One grows spheres, of radius δ, around all ξ, with the aim of collecting a subset to serve as prototypes for class l. The prototypes are the smallest subset of ξ that provide maximum coverage (cover ξ of class l) and minimum impurity (coverage of ξ of class other than l). Prototypes are assembled in a greedy manner. The example ξ_i that provides the best coverage and impurity is first added to the set of prototypes, and removed from ξ, along with the ξ (of class l) that it covers. The process then repeats to find the subsequent prototypes. A running estimate of the *quality* of a set of prototypes is maintained, computed using the coverage, the impurity and a penalty $N_{proto} \times \chi$, which is proportional to the number of prototypes N_{proto}. The algorithm stops when the greedy search can no longer find another prototype (of any class) that would maximize the quality of the set cover. The two user-defined inputs, δ and χ, control the N_{proto} (number of prototypes) that are placed and the quality of the cover/summarization that the prototypes achieve. Small δ usually lead to good coverage and small impurity, but very large penalties, and thus may not lead to a high-quality set of prototypes. A set of prototypes can incur three types of errors which ultimately define the quality of the coverage:

1) *Uncovered features:* A set of N_{proto} prototypes, with spheres of radius δ can leave a fraction u of examples uncovered. Optimal prototype selection will minimize u.
2) *Impure spheres:* A sphere of radius δ, centered on example ξ_i, of class l, could cover examples of a different class, leading to its "impurity" $\bar{i}$ (expressed as a ratio of examples from classes other than l that the sphere covers). As impurity is undesirable, the sum $b = u + \bar{i}$ is a measure of the quality of the selected prototypes. An overlap of clusters/classes will cause impurity, as will an excessively large δ.
3) *Misclassification rate:* A set of prototypes, along with δ, can serve as a nearest-neighbor classifier of the data, and forms a second measure of the quality of a set of prototypes. The performance of this classifier can be quantified as its misclassification or error rate m.

B. Prototype Placement Procedure

The process of placing prototypes essentially reduces to determining a value for (δ, χ) that delivers a desired quality and level of summarization (as quantified by (b, m, N_{proto})) of a data set. The user specifies a desired quality level (b^*, m^*) and one searches for the "optimal" (δ^*, χ^*) that can deliver it. Upper and lower bounds are specified in the $\delta - \chi$ space and we draw 200 samples of (δ, χ) from it in a space-filling manner. The search for (δ^*, χ^*) is conducted via seven-fold cross-validation. Computations are performed in the `R` statistical framework using the package `protoclass` [46] for prototype placement and `randtoolbox` [47] for random sampling.

The first step in prototype placement involves balancing the data set, since the sizes of the clusters vary widely. Equal numbers of examples of each class are drawn from the data set to constitute a new data set for the search. The features are centered (each component of ξ has its mean subtracted from it) and scaled (each component of ξ is divided by the standard deviation, so that the values vary, approximately, between -3 and 3). This data set is then randomly divided into seven folds. We iterate through the 200 (δ, χ) samples. For a given (δ, χ)-pair, we designate one of the folds as the "testing" fold whereas the rest are the "learning" folds. Prototypes are placed, using `protoclass`, into the "learning" folds and (b, N_{proto}) is computed from the placement. The prototypes, with their δ, are next used as a nearest-neighbor classifier to classify the data in the "testing" fold to compute m. We iterate through the seven folds to compute seven different (b, N_{proto}, m) values; these are averaged to serve as the performance figures for the (δ, χ)-pair. 200 such performance figures-of-merit are computed, and (b, m) plotted as a function of N_{proto}. The (δ, χ)-pair that yields (b, m) closest to (b^*, m^*) is designated the "optimal" (δ^*, χ^*) result.

The final prototype placement is performed by configuring `protoclass` with (δ^*, χ^*) and applying to the full data set; $(u^*, \bar{i}^*)$ are computed as a measure of the quality of final set of prototypes.

C. Channel Flow Results

As a first step, we apply prototype selection to the plane channel flow results in Sec. IV for $\mathrm{Re}_\tau = 180, 550, 1000, 2000$ and 5200. Here we set $\xi = f_{ch} == \{C_1, C_2, C_3, \eta_1, \lambda_1, \eta_4, \eta_3\}$, the optimal feature set for clustering of channel data, and perform prototype placement one Re_τ at a time. Figure 10 (left) shows the computation of (b, m, N_{proto}) via 7-fold cross-validation (CV), as described in Sec. V.B, performed for the $\mathrm{Re}_\tau = 1000$ data set. We plot the "bad coverage" error b and the misclassification error m as a function of N_{proto}, the number of prototypes chosen, as we iterate through the (δ, χ) samples. The horizontal line shows the desired values (b^*, m^*) leading to a (δ^*, χ^*) that results in $N_{proto} = 16$. Similar analyses were run for all the other channel-flow cases, to calculate data set-specific (δ^*, χ^*) configurations. Since the prototype selection is performed one Re_τ at a time, the (b^*, m^*) are slightly different, and the number of prototypes selected varies substantially (see Table 1)

In Figure 10 (right), we plot the mean velocity profiles from the channel flow cases, colored by the cluster ID. The open symbols denote the prototypes, placed for all the velocity profiles. As is clear, the number of prototypes changes

from flow to flow, and their locations on the velocity profile are not the same. In Table 1 we tabulate the cluster sizes and the number of prototypes chosen in each cluster, as a measure of the summarization achieved by the prototypes. $(u^*, \bar{i}^*)$ are also stated, as a measure of the quality of the summarization by the prototypes. We see that geometrically large regions/clusters with many grid points need not necessarily have many prototypes. This is because spheres are grown (when computing prototypes) after ξ has been centered and scaled i.e., the spheres, mapped back to physical space are highly skewed and irregular. In addition, the quality of the summarization by the prototypes is quite variable, e.g., some sections of the boundary layer have many prototypes while others have only a few (see Table 1). The summarization is also seen to become more efficient with Re_τ. This could be due, in part, to the improved separation of the log layer from the inner and outer layers as the Reynolds number increases. For each case with a log layer present, a single prototype is placed, consistent with the self-similar nature of the turbulence in this region.

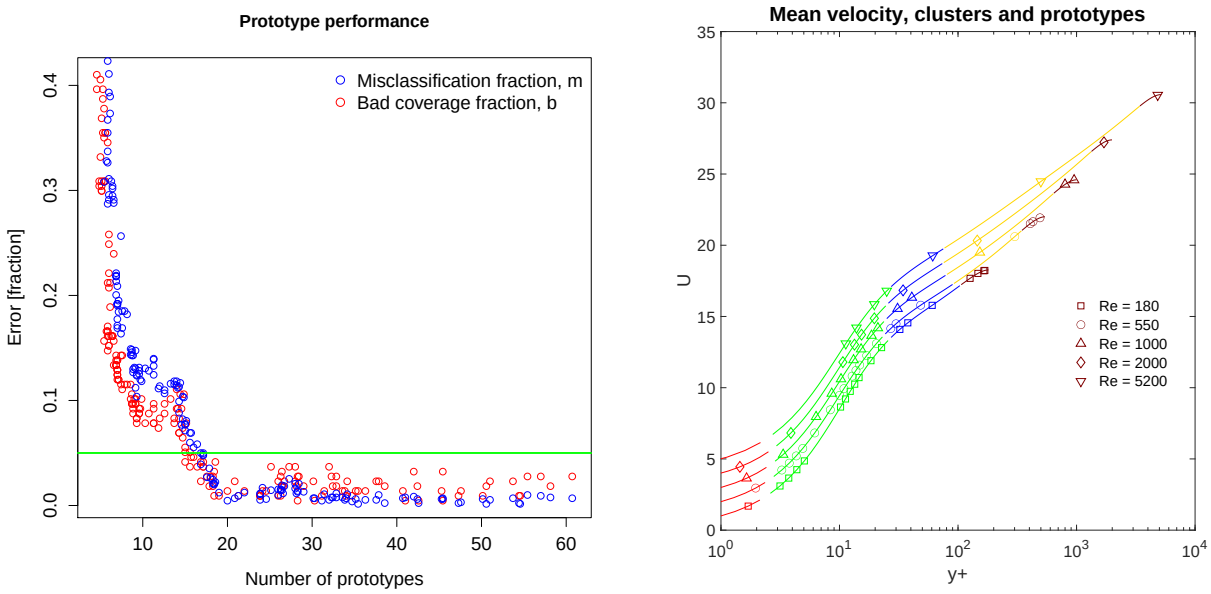

Fig. 10 Left: (b, m) plotted against N_{proto} for plane channel flow $\mathrm{Re}_\tau = 1000$ case as we search through 200 (δ, χ) samples for those close to the desired value (plotted in green). Right: Plots of the mean velocity profile, colored by their cluster, for $\mathrm{Re}_\tau = 180, 550, 1000, 2000$ and 5200. Prototypes are plotted with open symbols and have the same color as the cluster they summarize. For $\mathrm{Re}_\tau = 5200$ the prototype in the viscous sublayer (red) points) is positioned at $y_+ < 1$ and is not visible in the plot. The mapping between cluster colors and their IDs are in Table 1. Note: The velocity profiles have been shifted vertically to make them legible.

D. Complex Flow Results

We now apply the placement of prototypes to somewhat more complex flows, *viz.*, flow over a wavy wall. The feature set $\xi = f_{ww} = \{C_1, C_2, C_3, \eta_1\}$ is used for placement of prototypes. Figure 11 (left) plots the cluster sizes and the number of prototypes placed in each cluster. We see that there are 10 clusters whose sizes vary over a factor of 10. In the same figure, we also plot the ratio of the number of prototypes and the cluster size, as a measure of the

Cluster	$\mathrm{Re}_\tau = 180$	$\mathrm{Re}_\tau = 550$	$\mathrm{Re}_\tau = 1000$	$\mathrm{Re}_\tau = 2000$	$\mathrm{Re}_\tau = 5200$
Outer layer (brown; ID 1)	26 (4)	45 (3)	59 (2)	82 (1)	174 (1)
Log-layer (yellow; ID 4)	0 (0)	73 (1)	129 (1)	232 (1)	525 (1)
Outer buffer layer (blue; ID 2)	35 (3)	35 (3)	29 (2)	28 (1)	33(1)
Inner buffer layer (green; ID 3)	23 (11)	24 (13)	24 (8)	26 (5)	28 (3)
Viscous sublayer (red; ID 5)	5 (3)	15 (3)	15 (3)	16 (3)	8 (1)
N_{proto}	21	23	16	11	7
$(u^*, \bar{i}^*)$	(0%, 0%)	(0.5%, 4.7%)	(0.4, 6.2%)	(1.3%, 5.5%)	(2%, 6%)

Table 1 Cluster sizes (number of examples) and number of prototypes (in parentheses) for the 5 plane channel flow problems considered in this study. All flows have been segregated into 5 clusters, whose sizes are tabulated; the figures in the parentheses are the number of prototypes placed in the cluster. The last row of the table provides the fraction of examples left uncovered by the prototypes and the average impurity of the spheres grown at the prototypes. Cluster colors and IDs are provided in line with Fig. 10.

summarization obtained by prototypes; it is clear that for all but the smallest cluster, prototypes form 5% or less of the cluster. The data set has 461 prototypes, with the fraction of uncovered examples $u = 20\%$ and the "impurity" or misclassification fraction $\bar{i} = 9.7\%$.

In Figure 11 (right) we plot the flow-over-a-wavy-wall data set colored by the cluster ID. The prototypes are plotted with symbols of the same color. Not surprisingly, most of the cluster and prototypes are found near the wall where high gradients of the turbulent statistics exist. The periodic structure of the wall causes periodicity of the blue and red clusters. These distinct (in physical space) clusters have the same color because they occupy a contiguous region in the ξ-space (they are similar, from a turbulent processes point of view). The splitting of a contiguous region in feature space, as seen in this figure, is one of the reasons for using prototypes (rather than cluster centers) to summarize a turbulent data set. Prototypes, being examples drawn from the data set, can be mapped between ξ-space and physical space trivially, which simplifies their fluid-dynamical interpretation. We see that the prototypes are *not* uniformly distributed in physical space, indicating severe contortions of the clusters as they mapped between physical and ξ-space where clustering and prototype-pacement is performed.

Prototype placement for the bump-in-channel and flow-around-a-square-cylinder data sets are illustrated in the Supplementary Online Materials, with similar characteristics already noted in the description of the wavy wall prototypes.

VI. Discussion

As alluded to in Sec. I, data-driven turbulence models can only learn the turbulent states and dynamics in their TD, making it imperative to be able to label or characterize a TD by the kind of turbulence physics it contains. It is also necessary to *balance* a TD i.e., ensure that the various turbulent processes are represented by approximately equal numbers of examples (or DNS/LES grid cells). Here we show how clustering and prototypes may help us achieve these

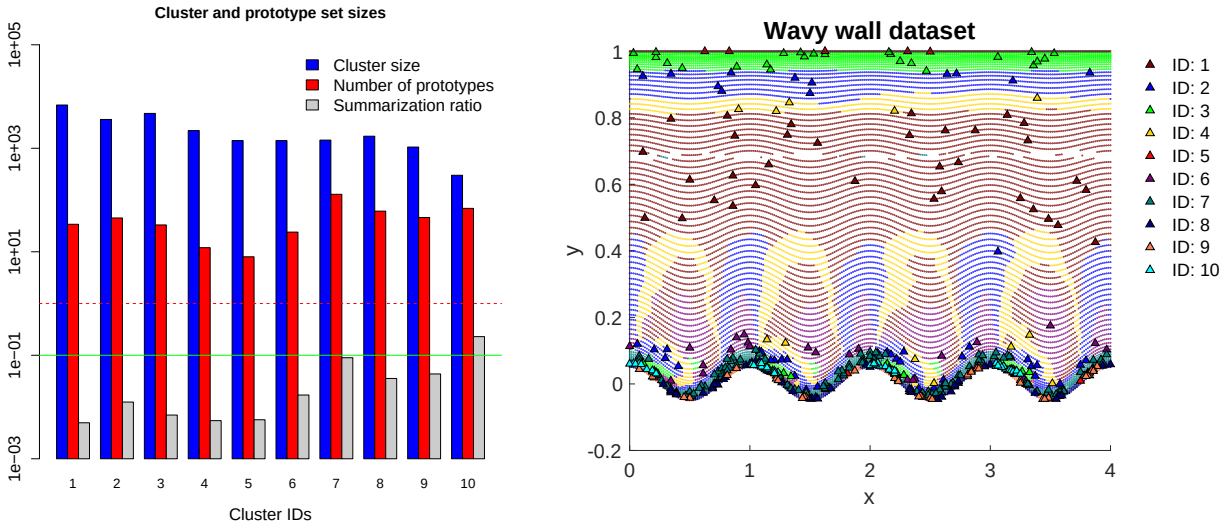

Fig. 11 Left: Comparison of the size of the 10 clusters in the wavy-wall data set and the prototypes placed in them. The prototypes, as a fraction of the cluster size, are also plotted ("summarization ratio"). The solid horizontal line corresponds to 0.1 whereas the dashed line corresponds to 1. Except for the smallest cluster (cluster ID 10), we obtain a reduction of 10x or more with the prototypes. Right: The flow-over-a-wavy-wall data set with points clustered by their cluster ID. Prototypes are plotted with symbols. The flow is from left to right. Cluster 10 has been colored cyan, instead of the original bright green, to distinguish it from Cluster 3 (colored green).

aims.

Balancing a TD: A useful TD should have a diversity of turbulence physics in it, i.e., pooling DNS/LES simulations with much the same physics will not lead to a generalizable turbulence model that is accurate in a diverse variety of flows. We use our clustering method to identify the types of turbulent processes in a data set (e.g., the channel flows) and then use previously learned physics to identify new ones in a previously unseen data set (e.g., the processes near the lower wall of the wavy wall data set). This incremental learning process revealed that only about 4% of the examples/grid cells in the "Bump in Channel" data set were new, making the bulk of the examples redundant. Thus the "Bump in Channel" data set is a poor candidate for inclusion in a TD, as it contributes little to diversity or information content. Note that without the ability to automatically group examples by their physics, it would be infeasible to detect the redundant nature of this data set. In addition, the ability to cluster can also allow us to check whether a TD obtained by pooling of DNS/LES results in a balanced TD. In our case, a naive pooling of our four data sets results in an extremely skewed distribution of examples (see Fig. 12), with the bulk of them being drawn from the near-wall boundary layer region. This is not entirely unexpected - DNS/LES grids are densely resolved in those regions. Thus our clustering method can be used to uncover the (lack of) diversity and imbalance in a TD.

Labeling a TD: While clustering analysis allowed us to identify the homogenous partitions in the TD, it does not allow us to characterize the type of physics in them, and thus discover the shortcomings of the TD assembled from a set

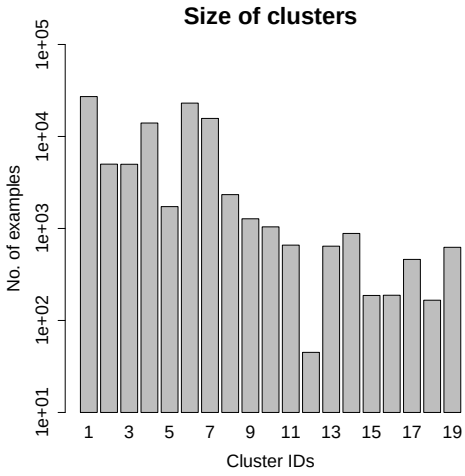

Fig. 12 Distribution of cluster sizes, assembled from our four flow data sets. They vary over almost three orders of magnitude.

of DNS/LES simulations. We show how prototypes assist in the process of TD characterization.

Difference in the anisotropy is a sufficient, though not necessary, condition for distinguishing two turbulence states. If a TD, mapped to the barycentric triangle, leaves parts of it uncovered, it identifies some of the turbulence physics that it does not contain. Since the barycentric coordinates are only a *subspace* of the clustering feature-space, it is quite possible that if all the examples of all clusters are plotted inside it, they will completely occlude some of the data. However, if a sparse, representative subset of the members of a cluster are plotted, the possibility of occlusion becomes small, allowing for easier illustration and explanation. However, this representative subset should uniformly cover all the clusters so that the (possibly sparsely populated) extremities of a cluster are well represented, as we seek to find the coverage of the barycentric triangle. Prototypes, which are selected via coverage arguments, are well-suited for the purpose and are used here.

In Fig. 13, left column, we plot prototypes from plane channel flow ($\mathrm{Re}_\tau = 2000$), the wavy wall flow and the square cylinder simulation, within the barycentric triangle. In the right column, we plot them inside the flow domain. Prototypes are colored by their cluster ID; therefore, many prototypes have the same color. The first row plots prototypes from channel flow, showing its coverage of the barycentric triangle (Fig. 13, top-left corner) and their position (Fig. 13, top-right corner) in the velocity profile. Their physical interpretations e.g., viscous layer etc. are evident. In the central row, we plot prototypes from the wavy wall data set. We see two groups of prototypes. A set of prototypes, in the center of the barycentric triangle (Fig. 13, middle row, left), occupies much the same locations as the prototypes from plane channel flow. These are drawn from the top of flow, where the flow is indeed similar to a plane channel flow (Fig. 13, middle row, right). The second set of prototypes (colors: violet, orange and brown ; cluster IDs: 7, 8 and 9 respectively) occupy the lower-left corner of the barycentric triangle and represent clusters with various combinations

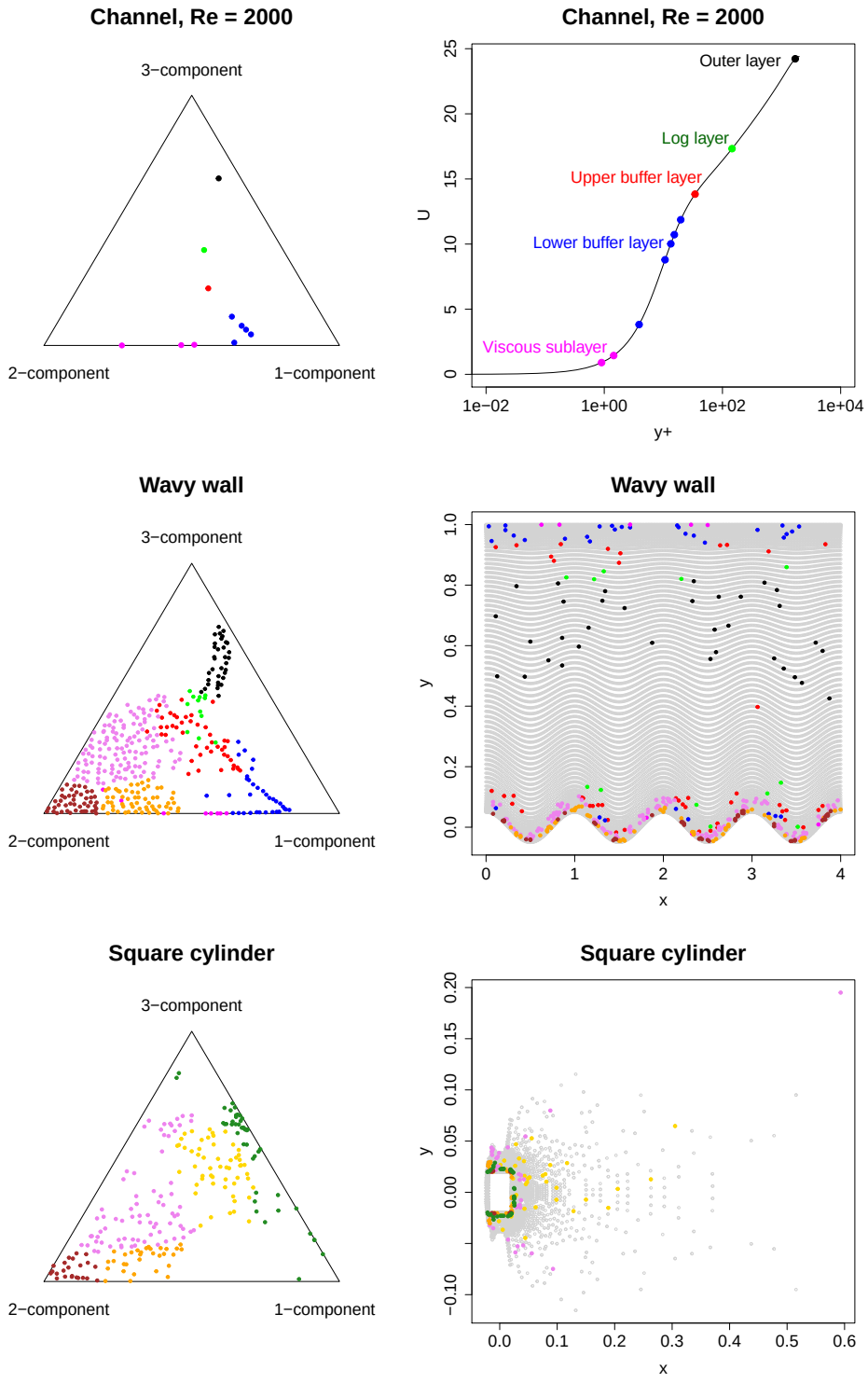

Fig. 13 Top row: Prototypes from the $\mathrm{Re}_\tau = 2000$ channel flow data set plotted inside the barycentric triangle (left) and on the velocity profile (right). Middle row: Prototypes from the wavy wall data set plotted inside the barycentric triangle (left) and inside the flow (right). Bottom row: Prototypes from the square cylinder data set plotted inside the barycentric triangle (left) and inside the flow (right). Note: This figure uses a new color scheme to constrast clusters adjacent in the barycentric triangle.

of one-and-two-component turbulence and two-and-three-component turbulence, with the two-component turbulence being the distinguishing characteristic. These are present near the lower wall. These clusters are characterized by approximately equal turbulent stresses ($\overline{u'u'} \approx \overline{w'w'}$) as well as clusters where one somewhat dominates over the other. Details are in Sec. IV. Thus the wavy wall data set serves the role of supplying examples of two-component turbulence to our TD.

In the bottom row, we plot prototypes from the square cylinder flow in the barycentric triangle (Fig. 13, bottom-left corner) and inside the flow (Fig. 13, bottom-right corner). It contributes a combination of two-component turbulence examples, in the lower left corner of the barycentric triangle, as well as examples (colors: khaki and dark green ; cluster IDs: 17 and 18 respectively) that occupy the boundary along 1-component and 3-component turbulence. These prototypes are drawn from the flow near the windward (or leading) corner where it accelerates around the square cylinder (1-component turbulence) and from the clusters corresponding to the shear layers on the top and bottom of the cylinder (barycentric triangle boundary joining 1- and 3-component turbulence). Details are in Sec. IV). Thus the square cylinder data set supplies examples that "fill" the right half of the barycentric triangle.

Compositing the prototypes in the barycentric triangles in the left column, we find there are no prototypes that occupy the 3-component corner of the barycentric triangle. Thus, apart from being very skewed, a naive TD assembled by simply pooling our four data sets would have some very common and elementary turbulent physics missing from it. This qualifies any data-driven turbulence closure trained with it. However, this shortcoming also identifies the type of simulation data sets that would improve our TD most and render the data-driven closure more generalizable. Note that the *quantity* of new simulation data required depends on the use-case, e.g., a deep neural net turbulence closure [3] will require far more data than one constructed with a random forest [48].

Noisy Data: We have performed a limited number of studies on the effect of noisy data on the clustering for channel flow. First, random noise was added to the training data features; the noise amplitude was specified as a fixed percentage of the standard deviation of each feature, and the noise was assumed to be uncorrelated across features. We found that the clustering results were somewhat sensitive to noise with magnitude of one percent or more. Second, we trained the Gaussian mixture model with clean data (no added noise), then tested it on the noisy data. We found that the clustering performed well with noise levels up to at least ten percent. The robustness of the classification to noise in the test data, conditional on *noiseless training data*, is similar to the results in Ref. [35]. Representative results for these noisy data scenarios are shown in the Online Supplementary Materials. These experiments suggest that care should be taken to reduce noise for training data, but that the resulting clustering model will be robust to significant noise in subsequent data that require classification. Note that our training data come from DNS, and as such do contain a small amount of statistical sampling error (typically one percent or less, although this can vary by feature and was not quantified for our training data sets). Caution should be taken when applying the present clustering algorithms and feature selection to more noisy training data, such as data from experimental measurements.

VII. Conclusions

In this paper we have explored the premise that turbulent states can be successfully classified, or categorized, using unsupervised machine learning techniques. The clustering and prototype-placement methods demonstrated in this paper serve a practical purpose - that of assessing the quality of a training data set used to construct data-driven turbulence closures. These data sets are generally assembled from multiple DNS/LES simulations' data and should contain the types of turbulent states/dynamics the closure is supposed to reproduce. Further, for a widely generalizable turbulence closure, the training data set should contain a diversity of turbulent physics, and considering DNS/LES data sets with redundant physics serves little purpose. We demonstrate how these desirable properties of a training data set might be investigated using the newly developed methods. Clustering analysis identifies partitions where the turbulent states/statistics are approximately homogeneous, as well as how abundantly they would be represented if the DNS/LES data sets were to be simply pooled together. In our case, we find that simply pooling the DNS/LES data sets considered here would lead to a training data set that is immensely skewed, with examples drawn from boundary layers dominating other forms of turbulent flows. In addition, we found that one of the DNS data sets considered here, channel with a bump, is largely redundant. Prototypes drawn from the clusters illustrate the space of turbulent states that is occupied by examples drawn from the training data, and more importantly, the parts that are not. Again, we find that simply pooling our DNS/LES data sets would provide very few examples of three-component turbulence e.g., isotropic turbulence, which would likely not be learned by any data-driven turbulence closure trained on it.

Apart from assessing the quality of a training data set, the methods developed here could have other uses. Prototypes could be used in network community-based ROMs of vortical and turbulent flows [8, 34] e.g., for modeling the aerodynamic forces on a pitching airfoil, caused by the Karman vortex street behind it. Such ROMs need collation of entities (in this case, small vortices) into larger structures or "communities" (for dimensionality reduction) which can them be approximated by a simpler model. Prototypes could be used to seed and grow communities via agglomerative clustering, as it would ensure that these communities are evenly spaced out and of similar sizes. Prototypes can also be used in the improved training of neural net turbulence models via transfer learning. Ref. [49, 50] describe how (convolutional) neutral net models for turbulent flows benefit from "pre-training". In Ref. [50], a previously trained neural net model could be generalized to a different Reynolds number by retraining with a fraction of the appropriate training data. In Ref. [49] a neural net was "pre-trained" using a smaller training data assembled by randomly sampling the full data set. Prototypes could perhaps be used to assemble the data set for "pre-training" in a rigorous fashion. They would ensure uniform representation of the various types of turbulence present, and the chances of omitting minority classes from "pre-training" data set would be reduced. Our own motivation is to use the prototypes identified here to facilitate the creation of explainable neural network turbulence models. The prototypes give a relatively small number of data points, which can be considered representative of a certain set of turbulent flow physics. We can probe the behavior of the neural network in the vicinity of the prototypes, and test whether its predictions are consistent with

desired model behavior for the identified physical situation. The clustering method developed here could also be used to identify when a data-driven closure, trained on a data set, is used in an extrapolatory fashion. Again we are considering the case where a machine-learned turbulence model has been trained using a training data set, and then required to make a prediction at a new point. If the new point clearly belongs to one of the clusters identified in the training data, then the model will likely make a valid prediction. If the new point is not well-classified by one of the clusters in the training data, this would indicate the model may not give a valid prediction, and new training data are required. Further studies are certainly required to explore the full utility of the approach, using a greater variety of turbulence data and including three-dimensional flow-fields. We conclude by noting that the present approach is not necessarily limited to single-point statistical features but could, in principle, be applied to any invariant statistical quantities one may choose to define a turbulent state.

Acknowledgements

This work was supported by the Laboratory Directed Research and Development program at Sandia National Laboratories. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

References

[1] Bowyer, N. V. C. K. W., Hall, L. O., and Kegelmeyer, W. P., "SMOTE: Synthetic Minority Over-sampling Technique," *J. Artificial Intelligence Research*, Vol. 16, 2002, pp. 321–357. https://doi.org/10.1613/jair.953.

[2] Beck, A., Flad, D., and Munz, C.-D., "Deep neural networks for data-driven LES closure models," *Journal of Computational Physics*, Vol. 398, 2019, p. 108910. https://doi.org/10.1016/j.jcp.2019.108910, URL https://www.sciencedirect.com/science/article/pii/S0021999119306151.

[3] Ling, J., Kurzawski, A., and Templeton, J., "Reynolds averaged turbulence modelling using deep neural networks with embedded invariance," *J. Fluid Mech.*, Vol. 807, 2016, pp. 155–166. https://doi.org/10.1017/jfm.2016.615.

[4] Cai, S., Zhou, S., Xu, C., and Gao, Q., "Dense motion estimation of particle images via a convolutional neural network," *Experiments in Fluids*, Vol. 60, No. 73, 2019. 10.1007/s00348-019-2717-2.

[5] Wang, J.-X., Wu, J.-L., and Xiao, H., "Physics-informed machine learning approach for reconstructing Reynolds stress modeling discrepancies based on DNS data," *Phys. Rev. Fluids*, Vol. 2, 2017. https://doi.org/10.1103/PhysRevFluids.2.034603.

[6] Hastie, T., Tibshirani, R., and Friedman, J., *The Elements of Statistical Learning*, 2nd ed., Springer, 233 Sprint Street, New York, NY, 2008.

[7] Kaiser, E., Noack, B. R., Cordier, L., Spohn, A., Segond, M., Abel, M., Daviller, G., Östh, J., Krajnović, S., and Niven, R. K., "Cluster-based reduced-order modelling of a mixing layer," *J. Fluid Mech.*, Vol. 754, 2014, pp. 365–414. https://doi.org/10.1017/jfm.2014.355.

[8] Gopalakrishnan Meena, M., Nair, A. G., and Taira, K., "Network community-based model reduction for vortical flows," *Physical Review E*, Vol. 97, 2018. https://doi.org/10.1103/PhysRevE.97.063103.

[9] Hadjighasem, A., Karrasch, D., Teramoto, H., and Haller, G., "Spectral-clustering approach to Lagrangian vortex detection," *Physical Review E*, Vol. 93, 2016. https://doi.org/10.1103/PhysRevE.93.063107.

[10] Baker, L., Frankel, A., Mani, A., and Coletti, F., "Coherent clusters of intertial particles in homogeneous turbulence," *J. Fluid Mech.*, Vol. 833, 2017, pp. 364–398. https://doi.org/10.1017/jfm.2017.700.

[11] Murayama, S., Kinugawa, J., Tokuda, I. T., and Gotoda, H., "Characterization and detection of thermoacoustic combusion oscillations based on statistical complexity and complex-network theory," *Physical Review E*, Vol. 97, 2018. https://doi.org/10.1103/PhysRevE.97.022223.

[12] Nair, A. G., Yeh, C.-A., Kaiser, E., Noack, B. R., Brunton, S. L., and Taira, K., "Cluster-based feedback control of turbulent post-stall separated flows," *J. Fluid Mech.*, Vol. 875, 2019, pp. 345–375. https://doi.org/10.1017/jfm.2019.469.

[13] Ser-Giacomi, E., Rossi, V., López, C., and Hernández-García, E., "Flow networks: A characterization of geophysical fluid transport," *Chaos*, Vol. 25, 2015. https://doi.org/10.1063/1.4908231.

[14] Ali, N., Hamilton, N., Calaf, M., and Cal, R. B., "Classification of the Reynolds stress anisotropy tensor in very large thermally stratified wind farms using colormap segmentation," *J. Renewable Sustainable Energy*, Vol. 11, 2019. https://doi.org/10.1063/1.5113654.

[15] Callaham, J. L., Koch, J. V., Brunton, B. W., Kunz, J. N., and Brunton, S. L., "Learning dominant physical processes with data-driven balance models," *Nature Communications*, Vol. 12, 2021. https://doi.org/10.1038/s41467-021-21331-z.

[16] Wikipedia contributors, "Set cover problem — Wikipedia, The Free Encyclopedia," https://en.wikipedia.org/w/index.php?title=Set_cover_problem&oldid=977797135, 2020. [Online; accessed 31-October-2020].

[17] Dy, J. G., and Brodley, C. E., "Feature Selection for Unsupervised Learning," *J. of Machine Learning Research*, Vol. 5, 2004, pp. 845–889.

[18] McLachlan, G. J., and Peel, D., *Finite Mixture Models*, John Wiley & Sons, Inc., Hoboken, NJ, 2000.

[19] Arthur, D., and Vassilvitskii, S., "K-means++: The Advantages of Careful Seeding," *SODA '07: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2007, p. 1027–1035.

[20] Schwarzl, G. E., "Estimating the dimension of a model," *Annals Statistics*, Vol. 6, 1978, pp. 461–464. https://doi.org/10.1214/aos/1176344136.

[21] Barone, M. F., Ray, J., and Domino, S., "Feature selection, clustering, and prototype placement for turbulence data sets," SciTech 2021, AIAA 2021-1750, 2021. https://doi.org/10.2514/6.2021-1750.

[22] Zhang, D. H., Chew, Y. T., and Winoto, S. H., "Investigation of intermittency measurement methods for transitional boundary layer flows," *Experimental Thermal and Fluid Science*, Vol. 12, No. 4, 1996, pp. 433–443. https://doi.org/10.1016/0894-1777(95)00133-6.

[23] Banerjee, S., Krahl, R., Durst, F., and Zenger, C., "Presentation of anisotropy properties of turbulence, invariants versus eigenvalue approaches," *J. of Turbulence*, Vol. 8, No. 32, 2007. https://doi.org/10.1080/14685240701506896.

[24] Tao, B., Katz, J., and Meneveau, C., "Statistical geometry of subgrid-scale stresses determined from holographic particle image velocimetry measurements," *J. Fluid Mech.*, Vol. 457, 2002, pp. 35–78. https://doi.org/10.1017/S0022112001007443.

[25] Buchner, A.-J., Lozano-Durán, A., Kitsios, V., Atkinson, C., and Soria, J., "Local topology via the invariants of the velocity gradient tensor within vortex clusters and intense Reynolds stress structures in turbulent channel flow," *2nd Multiflow Summer School on Turbulence, J. of Physics: Conference Series*, Vol. 708, 2016. https://doi.org/10.1088/1742-6596/708/1/012005.

[26] Pope, S. B., "A more general effective-viscosity hypothesis," *J. Fluid Mech.*, Vol. 72, No. 2, 1975, pp. 331–340. https://doi.org/10.1017/S0022112075003382.

[27] Schmitt, F., and Hirsch, C., "Experimental study of the constitutive equation for an axisymmetric complex turbulent flow," *Z. Angew. Math. Mech.*, Vol. 80, 2000, pp. 815–825. https://doi.org/10.1002/1521-4001(200011)80:11/12<815::AID-ZAMM815>3.0.CO;2-H.

[28] Lee, M. K., and Moser, R. D., "Direct numerical simulation of turbulent channel flow up to $Re_\tau \approx 5200$," *J. Fluid Mech.*, Vol. 774, 2015, pp. 395–415. https://doi.org/10.1017/jfm.2015.268.

[29] Gorlé, C., Emory, M., Larsson, J., and Iaccarino, G., "Epistemic uncertainty quantification for RANS modeling of the flow over a wavy wall," Center for Turbulence Research Annual Research Briefs, 2012.

[30] Marquillie, M., Ehrenstein, U., and Laval, J. P., "Instability of streaks in wall turbulence with adverse pressure gradient," *J. Fluid Mech.*, Vol. 681, 2011, pp. 205–240. https://doi.org/10.1017/jfm.2011.193.

[31] Emory, M., and Iaccarino, G., "Visualizing turbulence anisotropy in the spatial domain with componentality contours," Center for Turbulence Research Annual Research Briefs, 2014.

[32] Yuan, J., Mishra, A. A., Brereton, G., Iaccarino, G., and Vartdal, M., "Single-point structure tensors in turbulent channel flows with smooth and wavy walls," *Phys. Fluids*, Vol. 31, 2019. https://doi.org/10.1063/1.5130629.

[33] Pope, S. B., *Turbulent Flows*, Cambridge University Press, 2000.

[34] Meena, M. G., and Taira, K., "Identifying vortical network connectors for turbulent flow modification," *Journal of Fluid Mechanics*, Vol. 915, 2021, p. A10. 10.1017/jfm.2021.35.

[35] Manohar, K., Brunton, B. W., Kutz, J. N., and Brunton, S. L., "Data-Driven Sparse Sensor Placement for Reconstruction: Demonstrating the Benefits of Exploiting Known Patterns," *IEEE Control Systems Magazine*, Vol. 38, No. 3, 2018, pp. 63–86. 10.1109/MCS.2018.2810460.

[36] Elhamifar, E., Sapiro, G., and Vidal, R., "See all by looking at a few: Sparse modeling for finding representative objects," *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1600–1607. https://doi.org/10.1109/CVPR.2012.6247852.

[37] Esser, E., Moller, M., Osher, S., Sapiro, G., and Xin, J., "A Convex Model for Nonnegative Matrix Factorization and Dimensionality Reduction on Physical Space," *IEEE Transactions on Image Processing*, Vol. 21, No. 7, 2012, pp. 3239–3252. https://doi.org/10.1109/TIP.2012.2190081.

[38] Tropp, J. A., "Column Subset Selection, Matrix Factorization, and Eigenvalue Optimization," *Proceedings of the 2009 Annual ACM-SIAM Symposium on Discrete Algorithms*, 2009, pp. 978–986. https://doi.org/10.1137/1.9781611973068.106, URL https://epubs.siam.org/doi/abs/10.1137/1.9781611973068.106.

[39] Boutsidis, C., Mahoney, M. W., and Drineas, P., "An Improved Approximation Algorithm for the Column Subset Selection Problem," *Proceedings of the 2009 Annual ACM-SIAM Symposium on Discrete Algorithms*, 2009, pp. 968–977. https://doi.org/10.1137/1.9781611973068.105, URL https://epubs.siam.org/doi/abs/10.1137/1.9781611973068.105.

[40] Chan, T. F., "Rank revealing QR factorizations," *Linear Algebra and its Applications*, Vol. 88-89, 1987, pp. 67 – 82. https://doi.org/10.1016/0024-3795(87)90103-0, URL http://www.sciencedirect.com/science/article/pii/0024379587901030.

[41] Balzano, L., Nowak, R., and Bajwa, W., "Column subset selection with missing data," *NIPS Workshop Low-Rank Methods Large-Scale Mach. Learn.*, 2010. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.295.9188&rep=rep1&type=pdf.

[42] Bien, J., Xu, Y., and Mahoney, M. W., "CUR from a sparse optimization viewpoint," *Proc. Adv. Neural Inf. Process. Syst.*, 2010. URL http://papers.nips.cc/paper/3890-cur-from-a-sparse-optimization-viewpoint.

[43] Frey, B. J., and Dueck, D., "Clustering by Passing Messages Between Data Points," *Science*, Vol. 315, No. 5814, 2007, pp. 972–976. https://doi.org/10.1126/science.1136800, URL https://science.sciencemag.org/content/315/5814/972.

[44] Elhamifar, E., Sapiro, G., and Sastry, S. S., "Dissimilarity-Based Sparse Subset Selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 38, No. 11, 2016, pp. 2182–2197. https://doi.org/10.1109/TPAMI.2015.2511748.

[45] Bien, J., and Tibshirani, R., "Prototype selection for interpretable classification," *Annals of Applied Statistics*, Vol. 5, No. 4, 2011, pp. 2403–2424. https://doi.org/10.1214/11-AOAS495, URL https://doi.org/10.1214/11-AOAS495.

[46] Bien, J., and Tibshirani, R., *protoclass: Interpretable classification with prototypes*, 2013. URL https://CRAN.R-project.org/package=protoclass, r package version 1.0.

[47] Christophe, D., and Petr, S., *randtoolbox: Generating and Testing Random Numbers*, 2019. R package version 1.30.0.

[48] Milani, P. M., Ling, J., and Eaton, J. K., "Physical Interpretation of Machine Learning Models Applied to Film Cooling Flows," *Journal of Turbomachinery*, Vol. 141, No. 1, 2018. https://doi.org/10.1115/1.4041291, 011004.

[49] Morimoto, M., Fukami, K., Zhang, K., and Fukagata, K., "Generalization techniques of neural networks for fluid flow estimation," , 2020.

[50] Guastoni, L., Güemes, A., Ianiro, A., Discetti, S., Schlatter, P., Azizpour, H., and Vinuesa, R., "Convolutional-network models to predict wall-bounded turbulence from wall quantities," , 2020.