Data-Centric Engineering (2025), 6: e21
doi:10.1017/dce.2025.3

RESEARCH ARTICLE

Predictive monitoring of asset populations in a rotating plant under operational uncertainty: a transfer learning approach

Ziad Ghauch[1], Andrew Young[2], Blair David Brown[2], Bruce Stephen[2], Graeme West[2], Stephen McArthur[2] and Andrew Duncan[1,3]

[1]Data Centric Engineering, Alan Turing Institute, London, UK
[2]Electronic and Electrical Engineering, University of Strathclyde, Scotland, UK
[3]Department of Mathematics, Imperial College London, London, UK
Corresponding author: Andrew Duncan; Email: a.duncan@imperial.ac.uk

Received: 22 March 2024; **Revised:** 11 November 2024; **Accepted:** 26 November 2024

Keywords: change point detection; condition monitoring; domain adaptation; power station; rotating plant; statistical discrepancies; transfer learning; vibration

Abstract

Processing and extracting actionable information, such as fault or anomaly indicators originating from vibration telemetry, is both challenging and critical for an accurate assessment of mechanical system health and subsequent predictive maintenance. In the setting of predictive maintenance for populations of similar assets, the knowledge gained from any single asset should be leveraged to provide improved predictions across the entire population. In this paper, a novel approach to population-level health monitoring is presented adopting a transfer learning approach. The new methodology is applied to monitor multiple rotating plant assets in a power generation scenario. The focus is on the detection of statistical anomalies as a means of identifying deviations from the typical operating regime from a time series of telemetry data. This is a challenging task because the machine is observed under different operating regimes. The proposed methodology can effectively transfer information across different assets, automatically identifying segments with common statistical characteristics and using them to enrich the training of the local supervised learning models. The proposed solution leads to a substantial reduction in mean square error relative to a baseline model.

Impact Statement

Predictive maintenance analytics can provide operators and decision makers with crucial insight to support lifetime extension decisions. Machine learning models have been shown to be effective tools for predictive condition monitoring; however, their quality is hampered by the scarcity of data, particularly in unusual operating regimes. This work demonstrates a method by which a machine learning algorithm can leverage sensor information from other similar assets through an approach called transfer learning, thus enriching the training dataset and ultimately increasing the effectiveness of the predictive model.

1. Introduction

For power plant generation, condition monitoring sensors can provide operators and decision makers with real-time indicators of asset health, thus playing a fundamental role in supporting lifetime extension decisions (Coble et al., 2015). It is standard practice for engineers to manually analyze condition

© The Author(s), 2025. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (http://creativecommons.org/licenses/by/4.0), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

monitoring data using predefined diagnostic techniques (Young et al., 2022); however, this is a time-consuming operation requiring significant specialized expertise (Deng et al., 2020). The application of data-driven machine learning methods to asset monitoring data can deliver prognostic and diagnostic outcomes that are comparable to what engineers achieve but in a fraction of time (Deng et al., 2020). However, most of these data-driven techniques are based on supervised learning methods, which require large volumes of labeled training data to produce an accurate result (Surucu et al., 2023).

This paper demonstrates how predictive health monitoring for an individual asset within a population can be made more effective through the transfer of information from similar assets that have been used under similar operational conditions. This enables our data-driven model to generalize far beyond the set of operating regimes to which the asset has been subjected. The efficacy of the proposed methodology is demonstrated in a case study application using real condition monitoring data from a rotating plant asset in a power generation scenario.

The methodology proposed is based on partitioning time series into a series of segments using a change-point detection approach. Each segment captures the behavior of the asset under a single operational regime and is assumed to be statistically stationary. Segments that represent the same operating characteristics are identified by statistical tests for the equality of statistical distributions based on some notion of statistical divergence. Statistically similar segments are combined into an enriched training dataset on which a deep neural network model can be trained. The weights of the resulting deep neural network are transferred to the target domain, whereby any time series data in the target domain is used to further refine the model. This approach is shown to produce an order of magnitude reduction in mean square error with respect to the baseline control model.

The key contribution of this paper is the application of transfer learning to train predictive learners across a heterogeneous set of time series. This methodology has the ability to significantly improve the performance of individual predictive learners while remaining robust to missing, noisy, and erroneous data.

The paper is organized as follows. In Section 2, the research methodology is presented that includes notation, time series partitioning through change-point detection, domain similarity and subset pairs selection, two-sample hypothesis testing, and statistical distances as a measure of mismatch in probabilistic distributions. A real-world application problem related to telemetry monitoring of a roto-dynamic pump within a power plant is presented in Section 3.2. A discussion of the results obtained and conclusions is provided in Section 5.

1.1. Transfer learning and domain adaptation

Traditional predictive modeling approaches assume that training and prediction data have very similar distributions. In practice, this means that models are trained on a set of data with certain characteristics (the source domain) and are expected to generalize well on new scenarios which have similar characteristics (the target domain). This assumption can be a limitation when the data-generating process shifts over time, due to different operating conditions, and there is only a little training data available for each operating regime. To address this limitation, approaches like domain adaptation and transfer learning aim to adapt models trained on a source domain to perform well on a different, but related, domain (target), especially when the two domains do not share the same distribution.

Both transfer learning and domain adaptation are aimed at maximizing the available training data while mitigating mismatch in distributions between the source and target domains. In practical settings, this corresponds to training a predictive learner on the source domain and performing a subsequent evaluation on a target domain under limited to no availability of labeled data. Transfer learning and domain adaptation methods differ in one central aspect—domain adaptation approaches are centered on modifying at least one source domain in such a way as to minimize the mismatch in probabilistic distributions between the source and target domains, in transfer learning the source domain is unaltered (Weiss et al., 2016).

There are two general approaches to transfer learning. In *parameter-based transfer learning*, a model is trained in the source domain and then the associated model parameters are adapted to the target domain

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

(Chattopadhyay et al., 2012; Duan et al., 2012b; Tommasi et al., 2010; Yao and Doretto, 2010). In *feature-based transfer learning*, the goal is to optimally transfer a selection of features from the source to the target domain (Daumé, 2007; Duan et al., 2012a; Glorot et al., 2011; Long et al., 2014; Pan et al., 2011). Daumé (2007) provided a simple approach of combining the feature spaces of the source and target domains and using the combined augmented feature space to train any predictive model. Maximum Mean Discrepancy (MMD) is a commonly used method to learn a good feature mapping between the source and target domains, for example, Pan et al. (2011). The idea of leveraging statistical discrepancies will be central to our approach.

Transfer learning has been extensively utilized in fields like computer vision and natural language processing. Recently, there has been a significant push to apply these techniques to sequentially structured data, including time series (Gupta et al., 2020; He et al., 2020; Ismail Fawaz et al., 2018; Laptev et al., 2018; Matias et al., 2021; Sagheer et al., 2021; Ye and Dai, 2018, 2021; Zellinger et al., 2020). For example, Ismail Fawaz et al. (2018) used transfer learning to improve a classification model by leveraging a public dataset archive as the source domain. This study highlighted that transfer learning could either improve or worsen the performance of model predictions, which depends on the public datasets selected for pretraining the network model in the source domain. Laptev et al. (2018) proposed a transfer learning approach for time series prediction using a Convolutional Neural Network (CNN) model, demonstrating superior performance over the baseline deep learning models, especially in small to medium-sized datasets. He et al. (2020) adopted a multi-source transfer learning strategy for financial time series forecasting, introducing various ensemble methods for information transfer across multiple domains and considering various predictive models. In another context, Matias et al. (2021) explored optimal time series segmentation and pattern identification, employing an Hourglass deep CNN model coupled with transfer learning.

A naive application of transfer learning across two domains will not always lead to better performance. If the source and target tasks are structurally different, the predictive performance of a model may be negatively impacted, a phenomenon known as *negative transfer.* The level of predictive performance degradation associated with transfer learning is dependent on the level of mismatch between the source and target domains. With a mismatch in feature space distributions between the source and target domains, negative transfer occurs when a transfer learning approach from the source carries information that does not essentially generalize to the target domain. In a practical setting, it is possible to identify negative transfer by comparing against a baseline model which has been trained exclusively on the target domain. Within the scope of this paper, the effects of negative transfer are mitigated by carefully selecting the source data which is best aligned with the target domain, and filtering out instances where the source assets behave in a structurally different manner to the target asset.

2. Proposed methodology

2.1. Overview

The proposed methodology can be outlined in the following steps:

1. **Identification of distinct operational states**: Initially, to tackle the non-stationarity arising from different operational regimes in the time series data, a change-point detection algorithm is employed to partition the source and target time series into segments, each consisting of sensor data arising from a single operational regime.
2. **Selection of transfer candidates**: Next, a subset selection algorithm is applied with the objective of finding pairs of segments across the source and target domains that are indicative of similar operational regimes. This involves filtering pairs which exceed a degree of dissimilarity, ensuring that only relevant data is transferred, to minimize negative transfer.
3. **Training the deep neural network**: In the third phase, focus is placed on the pairs that exhibit the least dissimilarity. These matched pairs are amalgamated into a unified training dataset. A deep

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

neural network model is then trained on this consolidated dataset, optimizing it to capture the underlying patterns across these operational regimes.

4. **Transferring model weights to the target domain**: The final step involves transferring the weights of the deep neural network, which was trained in the previous stage, to the target domain. This allows for the fine-tuning of the model using time series data specific to the target domain, thereby aligning the model more closely with the characteristics of the target environment.

The entire process is delineated in Algorithm 1.

2.2. Transfer problem formulation and notation

Consider a source dataset $\mathcal{D}_S$ comprising two components:

1. the time series $(x_{S1},\ldots,x_{Sn})$ of p operational variables and
2. the time series of associated vibration sensor measurements $(y_{S1},\ldots,y_{Sn})$.

It is assumed that there exists a function $f_S:\mathbb{R}^p\times\mathcal{I}\to\mathbb{R}^q$ such that

$$y_{Si}=f_S(x_{Si},r_{Si})+\xi_i,$$

where ξ_i are independent identically distributed samples from a mean zero distribution and r_i denotes the unobserved operational state indicator, assumed to take values in $\{1,\ldots,R\}$, where R is the number of operational states. The source task involves learning the function f_S based on the observations in $\mathcal{D}_S$. Let $\mathcal{D}_T$ be a similarly defined target dataset, and suppose the target task involves learning the function $f_T:\mathbb{R}^p\times\mathcal{I}\to\mathbb{R}^q$ in an analogous manner. A transfer learning approach is leveraged to enhance the approximation accuracy of the function f_T by judiciously pooling information from $\mathcal{D}_S$ and $\mathcal{D}_T$ in a strategic fashion.

2.3. Detecting changes in operational state

If the operational state indicator r_{Si} was known, then pooling data from source to target domains would be a relatively straightforward process. Without this variable, the transfer process becomes significantly more challenging. This section focuses on an approach to mitigate this issue, specifically the use of change-point detection to identify structural shifts, and partition the time series into a set of sub-series, each of which is assumed to lie in a single operational regime.

Generally, methods for detecting change points fall into two broad categories: offline (Truong et al., 2020) versus online (Aminikhanghahi and Cook, 2017) schemes. In offline settings, analysis is done after the full time series data is available, while in online approaches anomalies and sudden changes are detected as data is collected. This work focuses on offline detection methods, leveraging all data from the source domain, whereby it is assumed that at the time of training a predictive learner on the target task, any data in the corresponding source domain is fully available.

Offline change-point detection methods seek to partition a time series into segments, such that the measurements within each partition are statistically homogeneous (Truong et al., 2020). The boundary points between partitions correspond to times at which a significant shift in statistical behavior is observed, known as change points. Given a time series $\mathbf{x}=x_1,\ldots,x_N$, a candidate partition $\mathbf{c}=(c_1,\ldots,c_K)$ of change points satisfies $1=c_1<c_2<\cdots<c_K=N$. The optimal partition of change points is found by finding the partition $\mathbf{c}$ which minimizes the loss function of the form

$$\mathcal{L}(\mathbf{x},\mathbf{c}):=\sum_{i=1}^{K-1} l\left(\left\{x_j : c_i\le x_j<c_{i+1}\right\}\right)+f(|\mathbf{c}|), \tag{1}$$

where l is a local loss function which measures the degree of statistical homogeneity within each segment and f is a penalization term to encourage small numbers of change-points, mitigating overfitting. A minimization of equation 1 leads to a partition $\mathbf{c}^\star$ of the dataset. The local loss function will typically

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

arise a log-likelihood of a model for the stationary behavior between change points. The stationary behavior can be described by a Gaussian distribution. The minimization problem (1) can be solved through direct minimization or (2) by reformulating it as a dynamical programming problem. In this work, the PELT (Killick et al., 2012) algorithm is used, which is an $O(n)$ offline change-point detection method which is an approximate dynamical programming approach, which leverages pruning to reduce the computational cost.

2.4. Identifying transfer candidates

The change-point partitioning scheme is applied above to both the source and target time series. Although each segment will correspond to a single distinct regime, the process does not identify the specific operating state. To achieve data transfer from source to target, a way of identifying source and target segments that represent a similar operating state is needed. The assumption that such pairs would be statistically similar and therefore can be identified through some notion of statistical distance or discrepancy is adopted. Let u_{Si} be a segment from the partitioned source dataset, comprising of both the operational and monitoring variables, that is,

$$u_{Si} = \begin{pmatrix} x_{S_{[c_i, c_{i+1}]}} \\ y_{S_{[c_i, c_{i+1}]}} \end{pmatrix},$$

for some change-point c_i, and let u_{Tj} be defined analogously. The pair (u_{Si}, u_{Tj}) is considered a valid candidate for transfer if the statistical distance between the samples (treated as independent) in u_{Si} and u_{Tj} is less than some threshold. The challenge with this approach is that the number of samples within each segment can vary significantly, so there is no obvious choice of threshold which would be appropriate for all possible pairs. This problem is addressed by introducing a statistical testing framework, whereby a two-sample test is performed for each pair, and rejection takes place at a pre-determined significance level. More specifically, assume that the columns of u_{Si} and u_{Tj} are distributed according to the probability distributions p_{Si} and p_{Tj}, respectively, both defined on $\mathbb{R}^p \times \mathbb{R}^q$. Then, the test hypothesis

$$H_{ij} : d(p_{Si}, p_{Tj}) = 0,$$

is tested based on the sample approximation of the discrepancy $\widehat{d}(u_{Si}, u_{Sj})$. For a general discrepancy d, the null distribution will not have a closed form, and so a permutation resampling is used to estimate it empirically, rejecting the hypothesis if the test statistic $\widehat{d}(u_{Si}, u_{Sj})$ lies outside the $\alpha/(K_T K_S)$-quantile of the null distribution, where $\alpha \in [0,1]$ and K_S and K_T are the number of source and target segments, respectively. The family-wise significance level α determines the sensitivity of the test, with the test being more likely to reject pairs (u_{Si}, u_{Tj}) as α is decreased. Scaling with $K_S K_T$ is a Bonferroni correction, applied to maintain a family error rate of α on all tests. Performing these tests on all possible source–target pairs results in the construction of a set Φ of m pairs for which the joint operational monitoring distribution is statistically similar.

The choice of discrepancy will have a strong influence on which source–target pairs are selected in this similarity assessment. As it was not a priori clear which discrepancy would yield the best predictive performance after the transfer, a range of different statistical distances (and their estimators) are considered and empirically assessed. In this work, Total Variation distance, Bhattacharyya distance (Bhattacharyya, 1943), Kullback–Leibler (Kullback and Leiber, 1951) and Jensen–-Shannon divergences (Endres and Schindelin, 2003), Renyi divergences (Rényi, 1960), Energy distance (Székely, 2002), Wasserstein distance (Vaserstein 1969), and MMD (Gretton et al., 2012) are considered. Definitions and further details can be found in the Supplementary Information.

2.5. Early stopping

A direct implementation of the above matching process would require performing $P = K_S \times K_T$ two-sample tests, each of which requires substantial computational effort due to the resampling process.

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

To reduce the cost of this step, an adaptive matching process is performed, where pairs are prioritized over others for testing. Let $z_1, \ldots, z_P$ represent the pairs P to be compared, where each z_k is of the form (u_{Si}, u_{Tj}) for some $1 \le i \le K_S$ and $1 \le j \le K_T$. The pairs are sorted in ascending order in discrepancy $\widehat{d}(u_{Si}, u_{Tj})$. Starting from z_1 each pair is considered sequentially, applying the two-sample test described in Section 2.4. At each step, the relative increase in the discrepancy from the previous step is computed. If it exceeds a threshold β (set at 10%), the transfer process is prematurely terminated. Although this potentially misses good transfer candidate pairs which score a high discrepancy due to noise, the termination process can drastically reduce the computational burden of this selection process. The sensitivity can be adjusted by increasing β. At the end of this process, all source terms selected by this scheme are combined into a training set Φ_S.

2.6. Predictive modeling

Previous studies (Deng et al., 2020; Idowu et al., 2022; Liao and Ahn, 2016) have demonstrated that deep-learning-based approaches can capture complex relationships between data-streams, as is typical of sensor data arising from engineering assets. In this case study, a basic feedforward deep multilayer perceptron (MLP) model is adopted. Although this is only a basic architecture, it performs sufficiently well for this demonstration. A detailed exploration of the capabilities of different deep learning architectures in this setting will be left for future work. A model is initially trained on the dataset Φ_S generated during the previous step. It is subsequently fine-tuned to the target domain, by freezing the lower hidden layers of the model and retraining the remaining layers on the target dataset. This is motivated by the fact that it is expected that the source and target domains behave quite similarly at a coarse level (captured by the lower layers of the network), in which higher layers are focused on finer-scale structure. Later, a parametric study is presented to explore this in more detail.

Algorithm 1 Overview of transfer learning methodology

Input:
source domain $\mathcal{D}_S \leftarrow ((x_{Si}, y_{Si}); 1 \le i \le n)$.
target domain $\mathcal{D}_T \leftarrow \left(\left(x_{Tj}, y_{Tj}\right); 1 \le j \le m\right)$
$\alpha \leftarrow$ significance level of two-sample test
Stage 1: Source and target partitioning
$\mathbf{c}_S \leftarrow PELT(\mathcal{D}_S)$,
$\mathbf{c}_T \leftarrow PELT(\mathcal{D}_T)$
$K_S \leftarrow |\mathbf{c}_S|$
$K_T \leftarrow |\mathbf{c}_T|$.
$(u_{S1}, \ldots, u_{SK_S}) \leftarrow$ partition of $\mathcal{D}_S$ into segments.
$(u_{T1}, \ldots, u_{TK_T}) \leftarrow$ partition of $\mathcal{D}_T$ into segments.
Stage 2: Transfer candidate selection
$P \leftarrow K_S K_T$.
$\Phi = \{\}$.
$\mathbf{z} = (z_1, \ldots, z_P) \leftarrow \left(\left(u_{Si}, u_{Tj}\right); 1 \le i \le K_S, 1 \le j \le K_T\right)$
Sort $\mathbf{z}$ in order of increasing discrepancy.
for u_s, u_t in $\mathbf{z}$: ▷ *two-sample test* with significance α
$p_{d_r} \leftarrow$ evaluate two-sample test p-value
if $p_{d_r} \le \alpha$:
$\Phi = \Phi \cup \{u_s\}$
Stage 3: Create optimized training set
$S_\Phi \leftarrow$ concatenate subsets in Φ to a training set
$M \leftarrow$ define predictive model
$M_{S_\Phi} \leftarrow$ train model M on source dataS_Φ

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

Stage 4: Transfer weights to target domain
$\mathbf{W}_{S_\Phi} \leftarrow$ extract parameters of predictive model M_{S_Φ}
$M_{S'_\Phi} \leftarrow$ transfer model parameters $\mathbf{W}_{S_\Phi}$ to target domain
$M_{S'_\Phi T} \leftarrow$ retrain $M_{S'_\Phi}$ on target data $\left\{ \left(x_{Tj}, y_{Tj} \right) \right\}_{j=1}^{m}$
Return: $M_{S'_\Phi T}$ predictive model after transfer learning

3. Method demonstration: Power plant case study

A nuclear power plant is a complex system of systems made up of interdependent, interconnected subsystems such as the reactor core and primary and secondary cooling systems. Each of these subsystems comprises different components; the latter are integral to the safe operation of the plant. The integrity of the entire nuclear power plant facility could be compromised if any of these components fails. A thorough understanding of each component of the nuclear power plant is required in order for the plant to operate safely. To do this, in the context of condition monitoring and risk mitigation, having a real-time warning system is crucial. A real-time warning system provides plant operators with support for decision-making in complex environments (Mumaw et al., 2000).

Condition monitoring is used primarily to differentiate between normal and abnormal states. During the operation of a nuclear power plant, it remains in a normal state most of the time. The occurrence of an abnormal state is infrequent and brief, resulting in data from these periods being sparse and of low density. Anomalies are essentially characterized by a sparse distribution and low density. Prevalent methods for anomaly monitoring include distance-based, density-based, neighboring-based, and model-based.

Prediction methods in nuclear power plants consist of physical or data-based approaches. Physical modeling is commonly used at different stages within the nuclear sector. On the other hand, for data-based methods, following the remarkable achievements of neural networks in various industrial domains, extensive research has been undertaken to employ neural networks in nuclear power plants. Applications include passive system response surface modeling Solanki et al. (2020), evaluating the reliability of instrumentation and control cables Santhosh et al. (2018), monitoring pipe thinning (Chae et al., 2020), detecting component cracks (Chen and Jahanshahi, 2018), and diagnosing accidents and abnormal states (Kim et al., 2020).

3.1. Rotating plant case study

In the context of asset management and condition monitoring within power generation environments, sensor-derived data provides decision makers with timely insights into the health of their assets. The availability of such health metrics allows those responsible for asset operations to consider extending the lifespan of these assets, as suggested by Coble et al. (2015). This case study, which employs the methodology suggested, utilizes time series data from a civil nuclear power plant. The plant design features a secondary loop in which turbines generate power and water pumps facilitate the circulation of the working fluid, as outlined by Dragunov et al. (2015). An illustrative diagram of the power plant is presented in Figure 1. Sensors are strategically placed at various points on the pump casings to track vibration levels, a critical factor in monitoring pump health. The deployment of these sensors, which are relatively expensive, is intentionally focused on areas of the pump that are expected to be prone to different failure modes (Coble et al., 2015). Keeping in mind industry standards, these vibration sensors are designed to detect a frequency range from 0.2 times the lowest rotational frequency to approximately 3.5 times the highest frequency, according to the guidelines outlined in the study by ISO (2002).

In this paper, the focus is placed on a specific case study of rotating plant assets, specifically main boiler feed pumps, operating within a power generation scenario. The system under consideration consists of two identical feed pumps denoted U_1 and U_2, which are operated independently on a rotational basis. The state of the system is monitored through four vibration sensors, denoted $\left\{ S^{(i)} \right\}_{i=1}^{4}$, in addition to five other

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

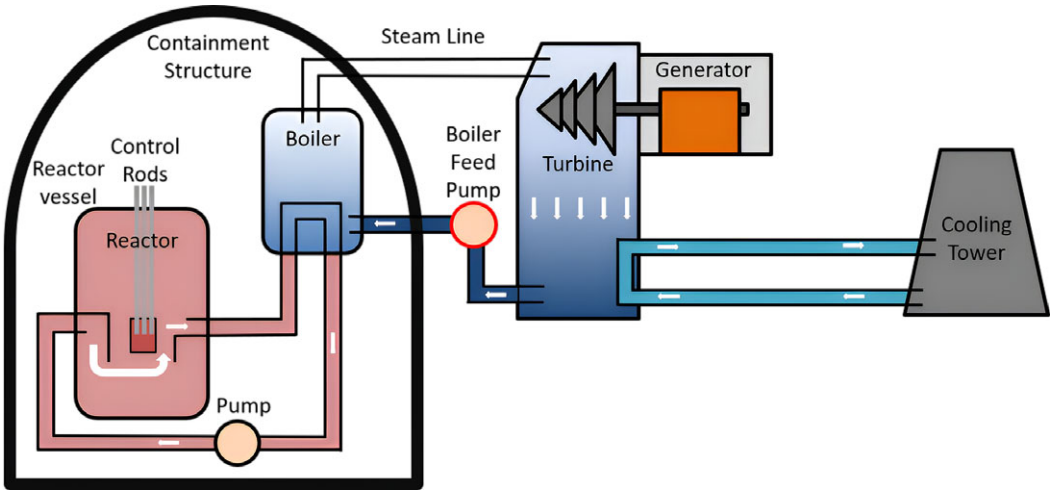

Figure 1. *Schematic of the nuclear power plant.*

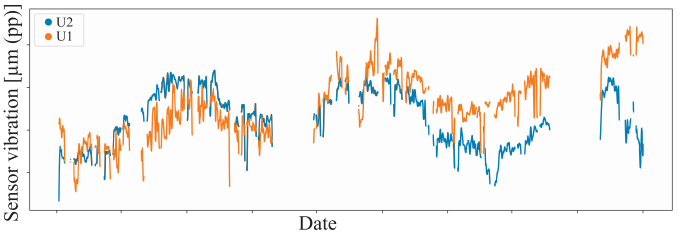

Figure 2. *Time series for sensor s_1 for primary (U_1) and corresponding secondary pump (U_2).*

sensor measurements directly related to pump operation: flow Q, velocity v, feed water inlet pressure P_{in}, feed water outlet pressure P_{out}, and power station load l_p. The vibration sensors are installed at four pump casing locations along the drive chain shaft: turbine, gearbox, pressure stage drive end, and pressure stage nondrive end. Vibration measurements across sensors capture any anomalies in the pump, including mechanical looseness, rotational shaft distortion, and machine unbalance. Sensor vibration measurements and the corresponding operational variables were recorded at 10-min intervals.

Our objective is to learn a predictive relationship between the operational features (Q, v, P_{in}, and P_{out}) and the four vibration features. Given that the two pumps are operating in rotation, it is expected that they encounter similar operational profiles, and so exhibit common performance characteristics. Discrepancies in behavior will arise from wear, as well as past maintenance and part replacement. Figure 2 presents the time series for a single vibration sensor for both the primary pump (U_1) and the corresponding secondary pump (U_2). The objective is to apply transfer learning to leverage the information from U_1 to improve the performance of a predictive model for the vibration profile of U_2. Note that axis values have been anonymized for sensitivity reasons.

Each pump operates in three distinct configurations: (*1*) normal state when the pump is online and operating at normal capacity, (*2*) offline state when the pump is turned off for maintenance and rehabilitation, and (*3*) in refueling mode where the pump operational speed is reduced (but still operational) while fueling is performed (Costello et al., 2017). The key challenge related to this case study is that there is no explicit indicator of operational state within the dataset. This is a common challenge in prognostic health monitoring, where human-driven decisions and associated events are not captured within the telemetry data used to provide decision support (Costello, 2019; Zio, 2022). Despite the lack of an explicit state variable, the regime changes manifest in both the operational and vibration

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

measurements quite clearly, with periods of statistically stationary behavior punctuated by sudden shifts. In particular, the evolution of operational states is tracked through structural shifts in (*1*) the mean of the vibration measurements shifting between different states and (*2*) the variance of a given time series which characterize the oscillations. As will be described in detail in later sections, change-point detection methods are leveraged to automatically partition the telemetry data into distinct regimes.

3.2. Pre-processing of the dataset

The dataset consists of a multivariate time series with five operational variables and four vibration monitoring variables. The operational variables are quite strongly correlated (see Figure 3), which motivates the use of dimension reduction on these features. Principal component analysis is performed, jointly on both the source and target datasets. Approximately 88% of the variance of the original input feature space can be essentially captured by the first principal component. Hence, in subsequent analysis, only the first principal component is used to represent operational features.

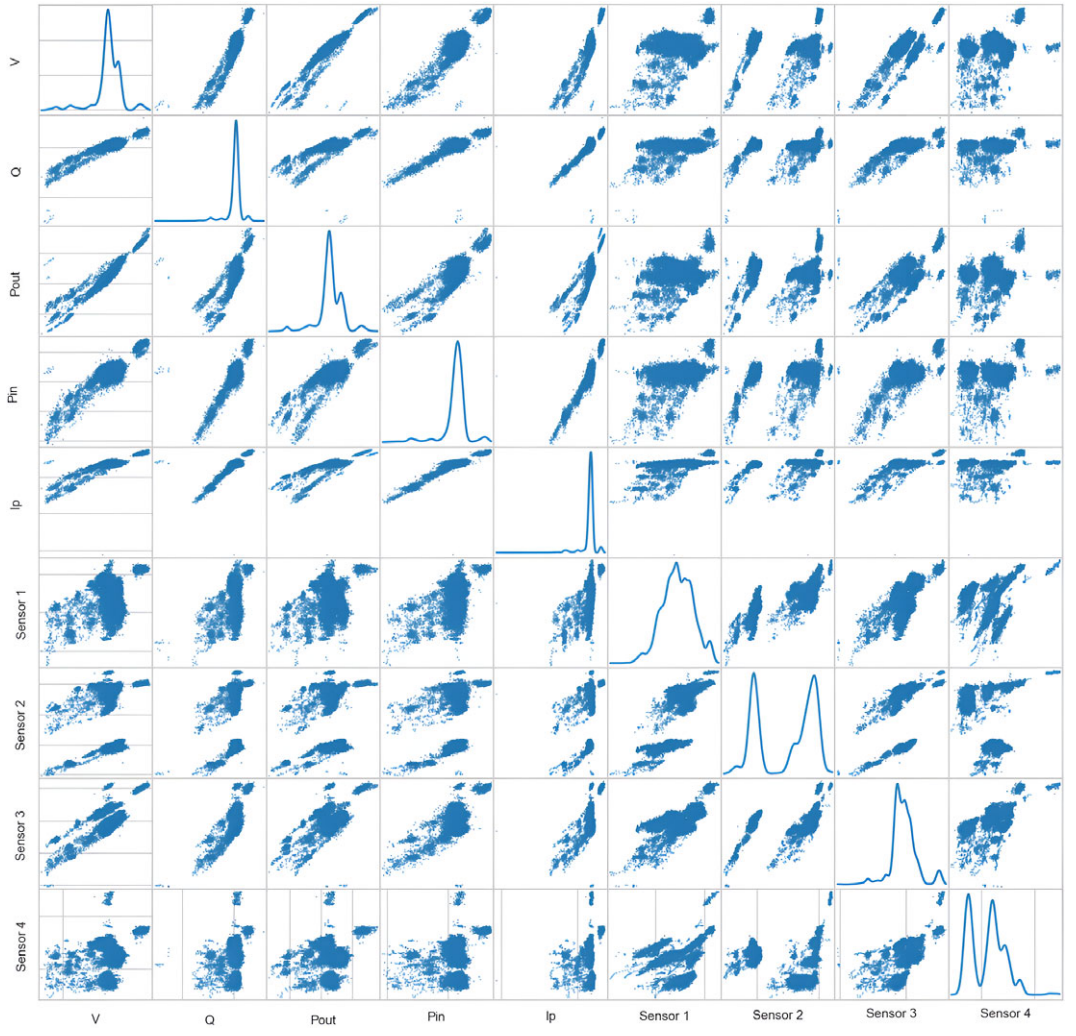

Figure 3. *Correlation matrix of inputs (operational variables) and outputs (sensor vibration) for the nuclear feed pump dataset.*

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

The raw time series vibration data is filtered in order to smooth the data and remove outliers to prepare the data for the subsequent transfer methodology. This was done following the Savitzky–Golay (SG) approach (Savitzky and Golay, 1964) as this approach was developed specifically for data related to "continuous physical experiments." Figure 4 shows example raw sensor data along with the best SG filter. The best SG filter parameters in terms of window size and fitting polynomial order were selected following a parametric study to best fit the multivariate time series.

4. Results

4.1. Implementation details

When applying the PELT change-point detection algorithm to the source (U_1) time series, the scheme identified $K_S = 95$ segments. Figure 5(a) demonstrates the inferred partition of the source dataset.

After partitioning the source and target time series, an evaluation of pairwise discrepancies is performed on every source–target pair. An illustration of the source–target distributions corresponding to sensor S_1 for the first eight source and target segments is shown in Figure 6. As expected, most pairs exhibit significant differences between the source and target distributions, which can be observed visually.

A parametric study of statistical divergences was performed. Figure 7 shows a raster of the statistical divergence between all source–target pairs for the first vibration sensor, with brighter (yellow) colors denoting areas of greater dissimilarity (thus not likely candidates for transfer). In contrast, darker areas (blue) denote pairs where a transfer is more appropriate. A shared characteristic pattern between the Total Variation distance, Energy distance, Wasserstein metric and MMD distance is observed.

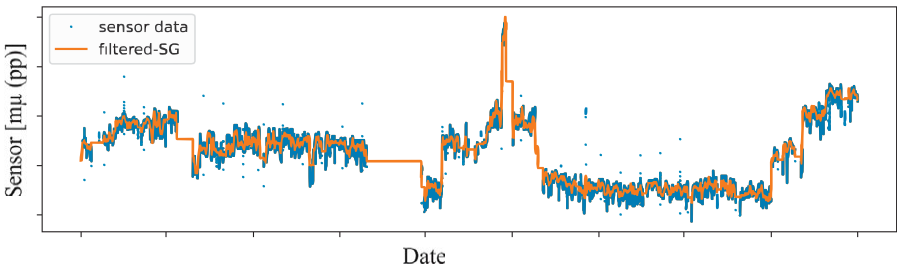

Figure 4. *Filtering of raw time series from vibration sensor telemetry following the SG method.*

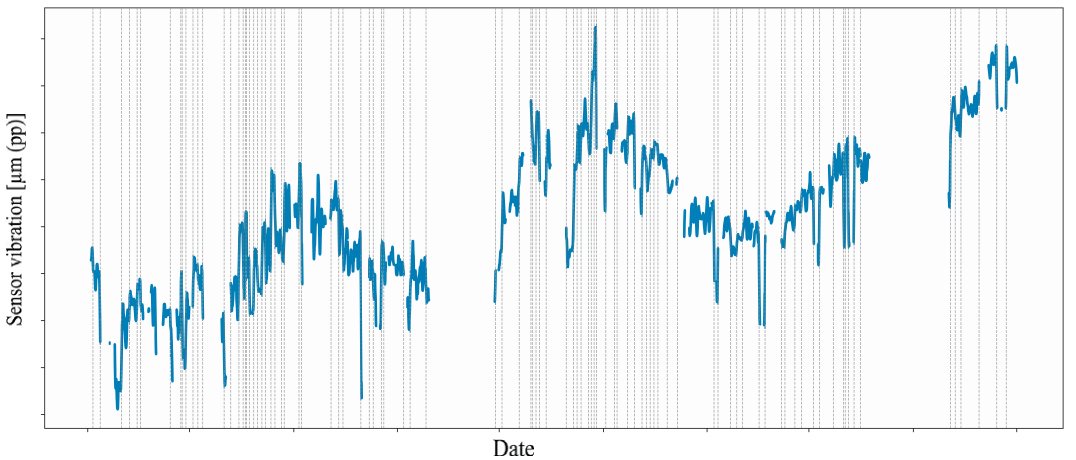

Figure 5. *Partitioning of the source time series data based on a change-point detection scheme for sensor U_1.*

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

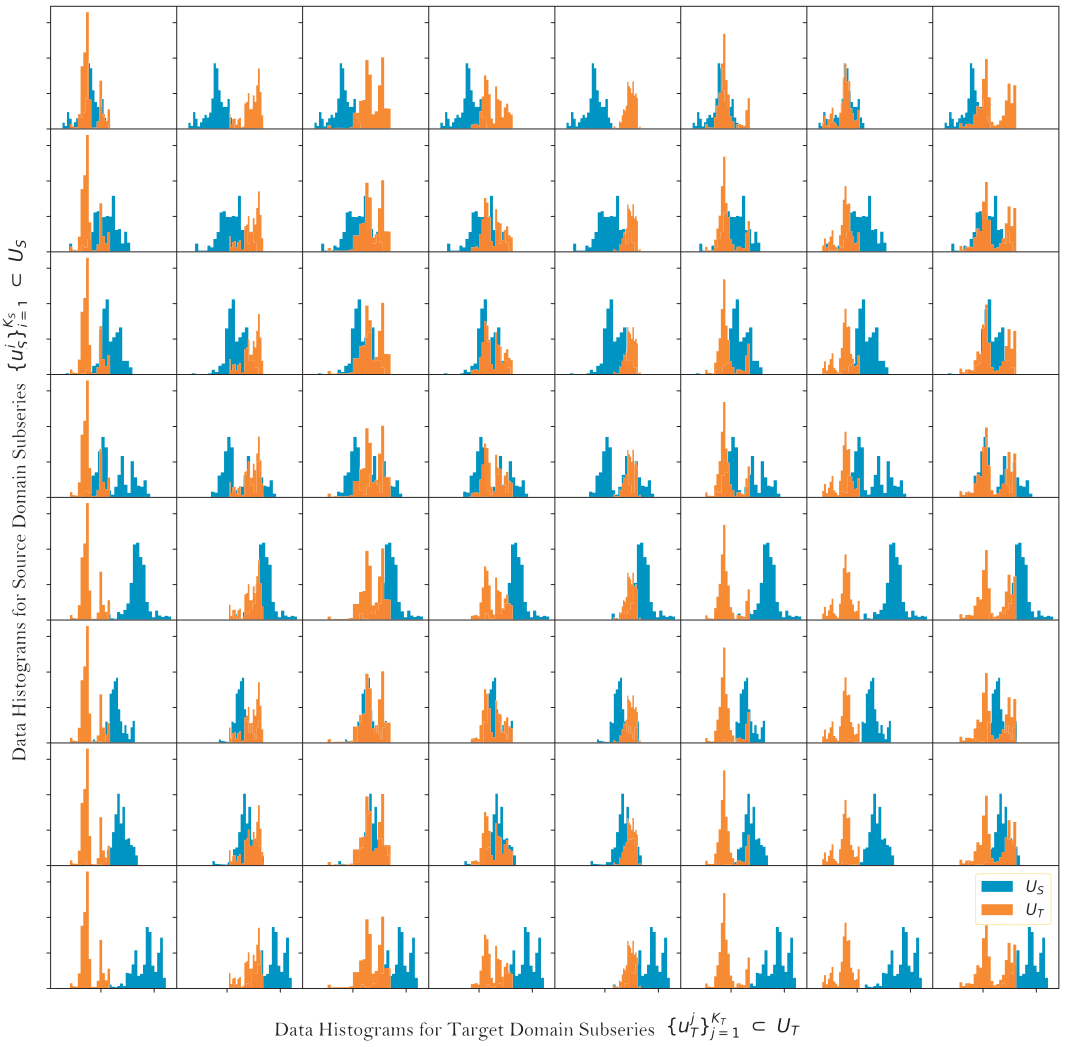

Figure 6. *Pairwise comparison of segment distributions between source and target domains.*

A quantification and comparison of the potential of the different statistical distances considered to effectively transfer information from the source to the target domain was carried out. Table 1 shows the MAE associated with vibration predictions using the transfer learning methodology for different statistical distances in an example test set for three random segments on the target domain, denoted by $u_{T,15}$, $u_{T,50}$, and $u_{T,92}$. Let M_T denote the control model, which consists of a deep neural network model with randomly initialized weights, trained over the target domain without transfer learning. Since M_T does not involve any information transfer from the source to the target domain, identical error levels are observed for all statistical distances considered. The corresponding model with transfer learning is indicated by $M_{S'_\Phi T}$. The following observations can be made: (*1*) in our setting, Bhattacharyya, Kullback–Leibler, Jensen–Shannon, and Rényi divergences demonstrated poor transfer learning potential; (*2*) strong performance in terms of positive transfer learning potential observed for Total Variation, Energy, Wasserstein, and MMD measures. In fact, not only is positive learning identified, but there is an approximately 19–27% reduction in error. This result highlights the potential for positive transfer in the proposed methodology.

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

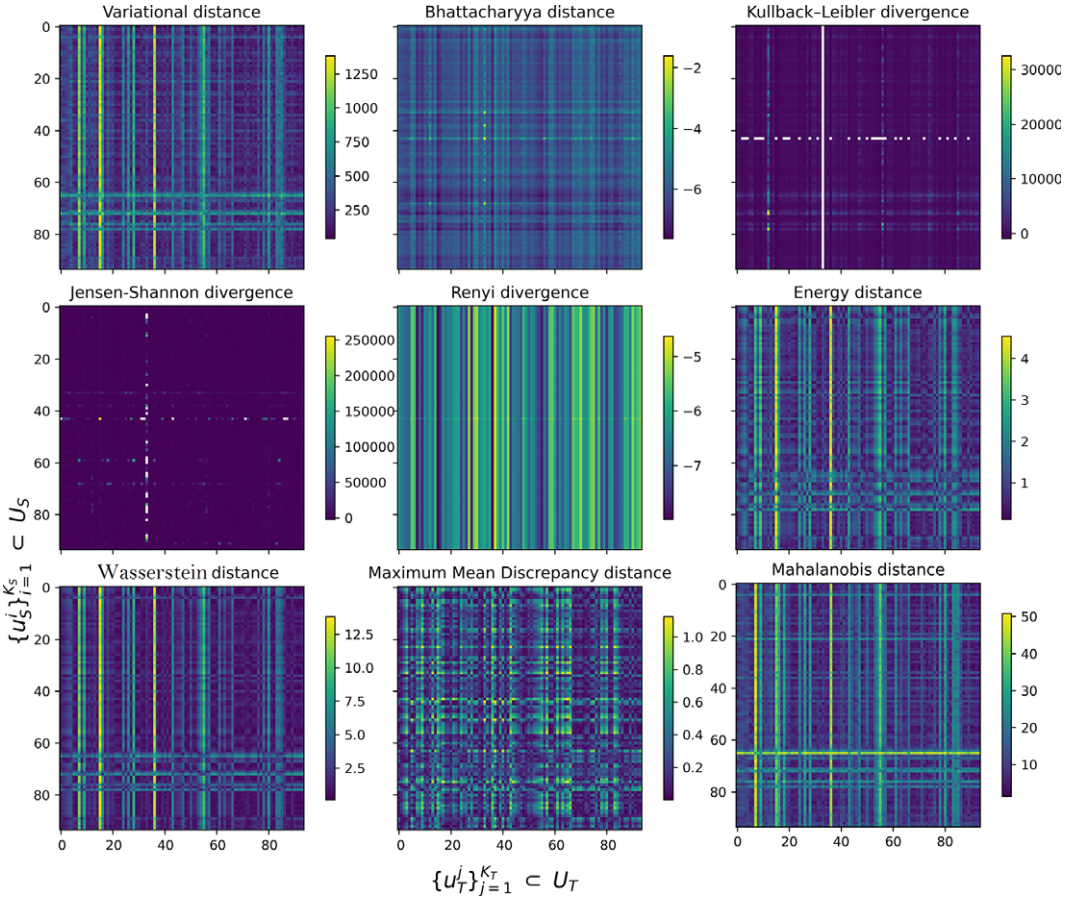

Figure 7. *Heatmap representing statistical distances as measures of dissimilarity between source and target pairs. Darker (blue) areas denote pairs where transfer is more appropriate.*

Table 1. *Similarity metrics' MAE comparisons for control model and model utilizing transfer learning for three example source–target pairs*

	$u_{T,15}$			$u_{T,50}$			$u_{T,92}$		
Distance	M_T	$M_{S'_\Phi T}$	% ch	M_T	$M_{S'_\Phi T}$	% ch	M_T	$M_{S'_\Phi T}$	% ch
Total Variation	0.0219	0.0162	−26.14	0.0417	0.0284	−32.02	0.226	0.1823	−19.36
Bhattacharyya	0.0219	0.0162	−25.97	0.0417	0.0384	−8.01	0.226	0.2414	6.8
Kullback–Leibler	0.0219	0.0158	−27.67	0.0417	0.0384	−7.98	0.226	0.2414	6.8
Jensen–Shannon	0.0219	0.0223	1.89	0.0417	0.0385	−7.74	0.226	0.2414	6.8
Rényi	0.0219	0.0158	−27.8	0.0417	0.0384	−8.01	0.226	0.2414	6.8
Energy	0.0219	0.0162	−26.14	0.0417	0.0284	−32.02	0.226	0.1823	−19.36
Wassestein	0.0219	0.0162	−26.14	0.0417	0.0284	−32.02	0.226	0.1823	−19.36
MMD	0.0219	0.0162	−26.14	0.0417	0.0284	−32.02	0.226	0.1823	−19.36

We also note that among the discrepancies considered, MMD has the advantage of being a robust metric on probability measures (Briol et al., 2019), which means that it is not overly sensitive to the presence of outliers in the distributions. This is highly beneficial in this context where telemetry data can

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

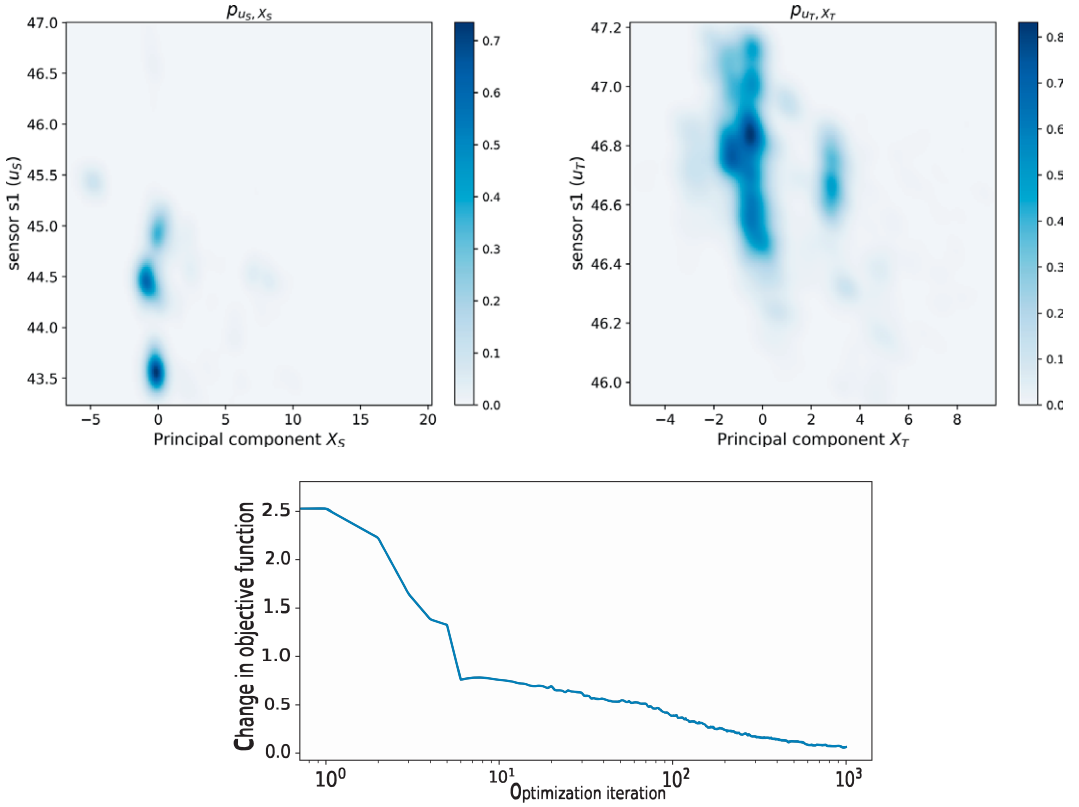

Figure 8. *Transfer of matching segments (a) joint density over $\mathcal{D}_S$, (b) joint density over $\mathcal{D}_T$, and (c) relative change in the discrepancy.*

often be polluted with spurious values which would other result in a rejection of similarity between two candidate distributions. Based on the above results, the MMD is adopted as the measure to identify source–target pairs. Two-sample MMD-based tests were performed using permutation resampling, with a total of 2000 permutations for each test. The null hypothesis (i.e. that the source–target distributions are equal) was rejected with a significance level $\alpha = 95\%$.

Figure 8 shows the convergence results for the transfer learning problem, which provides a measure of the weight of each selected matching subset pair in both the source and target domains, starting with the most important pairs with the highest weight (that is, the lowest statistical dissimilarity measure of MMD) and proceeding further with source–target pairs of lower weights, until the stopping criterion is met. Figure 8 shows the weight change during iterations, with a cutoff of 1% adopted to stop the pair matching algorithm.

The specific architecture parameters and model training hyperparameters were tuned using a Bayesian optimization scheme, using a look-back window of size 20 (10-min) increments and a forecasting horizon of size 10 (10 min). The following hyperparameter values were adopted: the number of epochs of {20,500}, the mini-batch size of {8,16,32,64,128,256}, and a learning rate in the range [0.0,0.01], using the Stochastic Gradient Descent algorithm. The optimal hyperparameters are listed in Table 2.

4.2. Sensitivity analysis to network parameters

4.2.1. Hidden layers

To understand the effect of the architecture in the context of transfer learning, a series of parametric studies are conducted. Figure 9 shows the convergence results in terms of MAE for different numbers of hidden

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

Table 2. *Optimal model hyperparameters*

Parameter	Optimal value
`batch_size`	32
`learning_rate`	0.0075
`num_hidden_layers`	50
`num_units`	32
`activation_function`	ReLU
`lookback_window`	200 min
`forecast_window`	100 min
`optimization`	Stochastic gradient descent
`distance_metric`	MMD

layers following a deep MLP network architectures. Figure 9 also shows the percentage change in MAE of the model with transfer learning compared to the control case, whereby a negative percentage change indicates a decrease in the error of the model with transfer learning compared to the control case. On the vertical axis of Figure 9, different data scarcity levels over the target domain are considered by artificially imposing scarcity on the target data. For example, for a target data scarcity level of 0.3, only 30% of the target data is considered, while completely disregarding the remaining target time series data. The following observations can be made: (*1*) the transfer learning accuracy heavily depends on the deep network architecture, whereby at higher number of hidden layers, only negative transfer is observed, with the optimal number of hidden layers bound at around 25; (*2*) at sub-optimal number of hidden layers (i.e. above 25) the model with transfer learning performs worse than the control model at all target data scarcity levels, suggesting that negative transfer is occurring at higher network depths (i.e. beyond 25 hidden layers); (*3*) below the threshold of 25 hidden layers, the model with transfer learning performs better than the control model essentially across most target data scarcity levels, suggesting positive transfer is taking place; (*4*) moreover, the expected pattern of drop in error with higher data availability on the target domain only manifests itself for higher number of hidden layers; and (*5*) finally, similar results are observed between MSE and MAE measures.

4.2.2. Transferable layers

The specific manner in which the weights in the trained model are transferred from the source domain model to the target is explored. Figure 10 shows the results of the transfer learning error in terms of MAE for different transferable layers, for a constant number of hidden layers set at 25. Error results are also obtained for a range of data scarcity levels imposed on the target domain. Based on Figure 10, the following observations can be put forward: (*1*) for a fixed number of hidden layers set at 25, transfer of higher level hidden layers reduces the magnitude of positive transfer; a possible explanation is that lower-level hidden layers carry information on low-level data structure, while higher-level hidden layers carry information on high-level data structure; (*2*) best performance observed when the information carried from the source to the target domain focuses on transfer of low-level data structure as captured by lower-level hidden layers, suggesting that the source and target domain data might share low-level data structure as opposed to higher-level structure; (*3*) for a fixed number of hidden layers set at 25, at the optimal number of transferred layers at 3/25, transfer learning models tremendously outperform control cases, highlighting that positive transfer learning occurs in such settings; (*4*) similarly, the expected drop in error with more target data availability as shown on the vertical axis only manifests itself for higher number of transferable layers (i.e. 23/25); and (*5*) similar performance is observed in terms of MAE and MSE metrics.

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

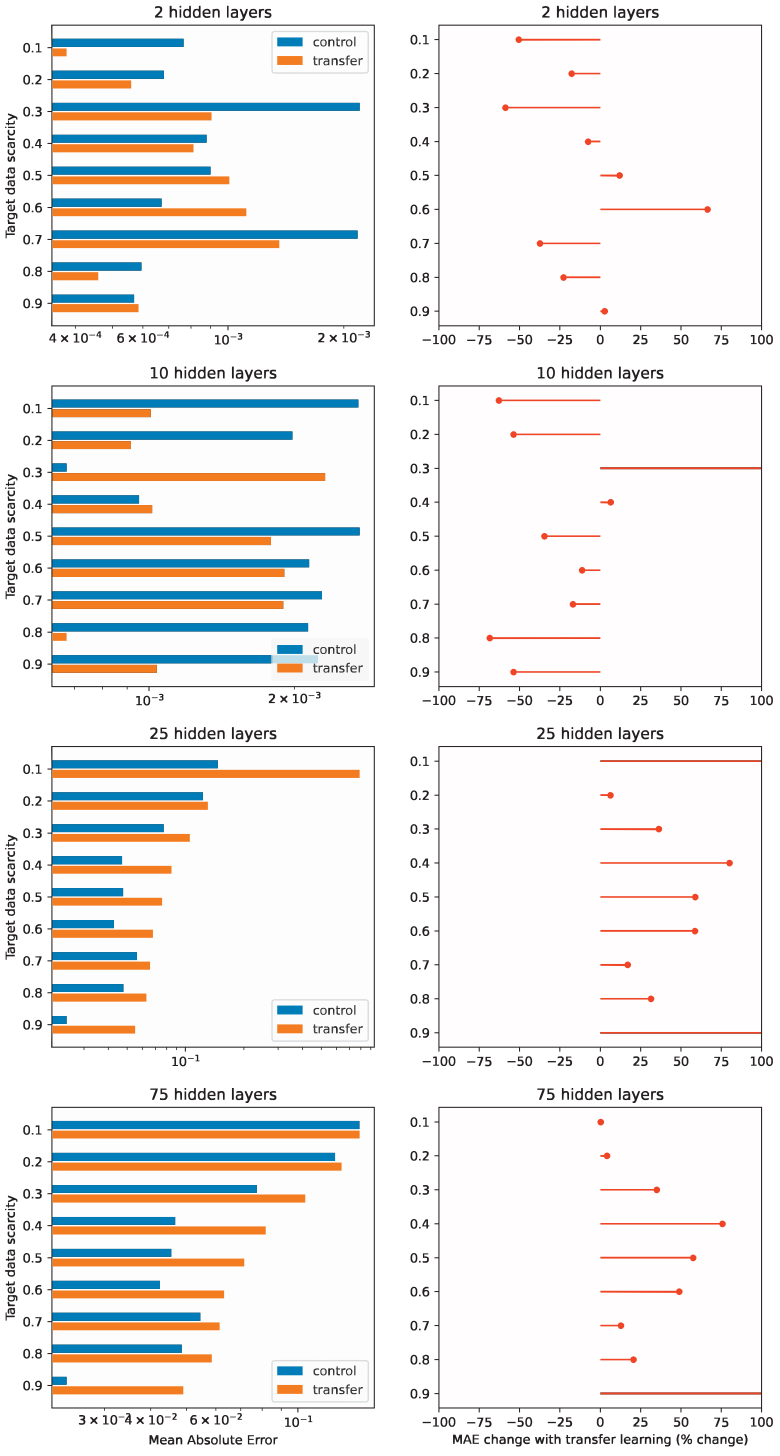

Figure 9. *Transfer learning error (MAE) for different network architectures (number of hidden layers).*

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

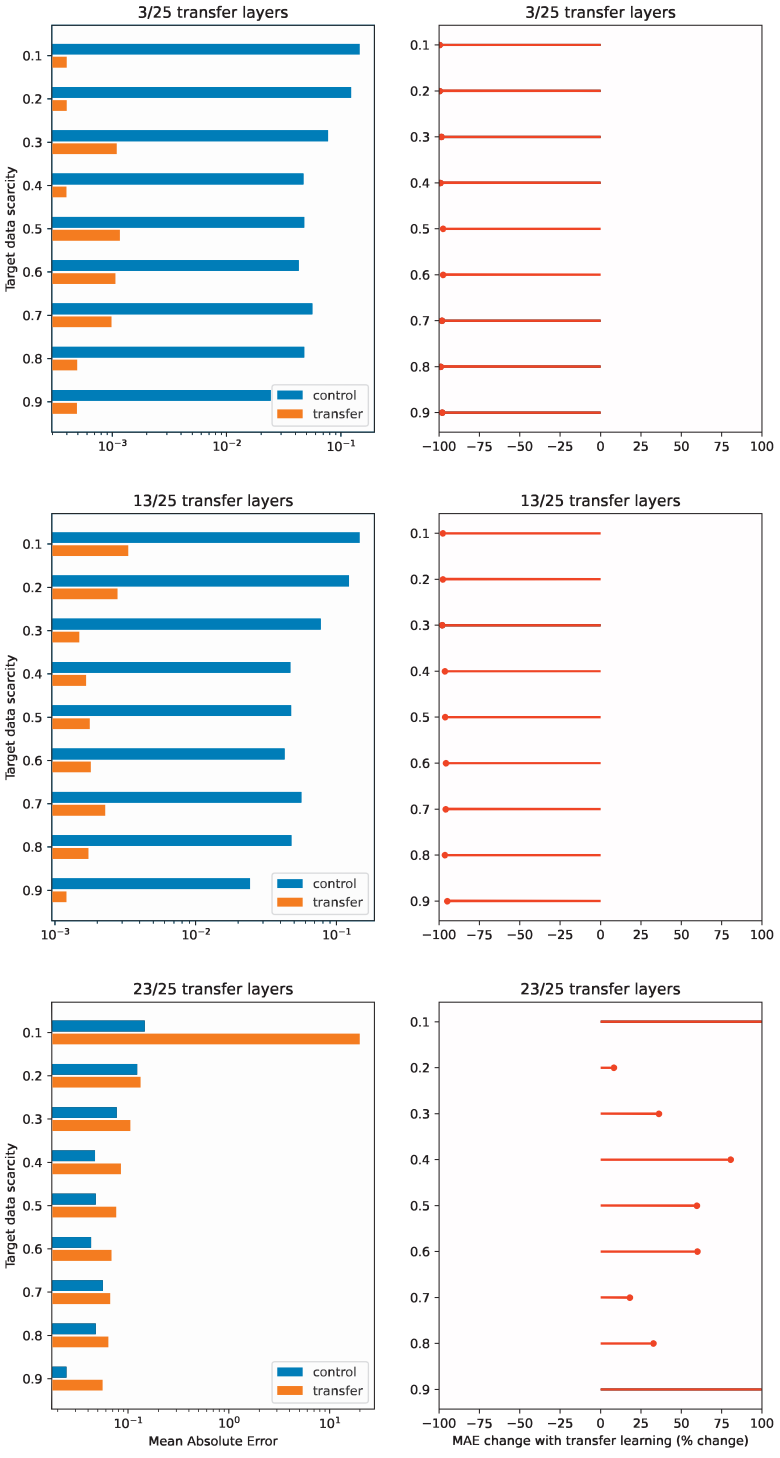

Figure 10. *Transfer learning error (MAE) for different transferable layers.*

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

5. Conclusions

This work addresses the issues associated with applying transfer learning in a real engineering scenario. The time series data commonly arising from a complex engineering asset is challenging to analyze and model due to the inherent non-stationarity caused by the presence of different operating states, which might not be explicitly characterized in the data.

The methodology proposed in this work seeks to leverage any additional information that can be obtained from adjacent assets running under similar operating conditions through a careful transfer learning process. The efficacy of the proposed methodology is demonstrated in real condition monitoring data which have been obtained from a boiler feed pump in a civil nuclear power station, where two similar pumps with differing data characteristics form the basis of the transfer learning problem.

The results obtained provide a positive indication that transfer learning can be applied to this challenging condition monitoring setting, as long as specific considerations are taken into account when developing and applying the transfer learning methodology. When conducted correctly, an engineer can expect to achieve a substantial improvement in the predictive accuracy of associated condition monitoring models. The results also demonstrate the need for careful consideration of the statistical discrepancy used to identify similar source and target distributions. We note that the robustness properties of MMD play a crucial role in enabling effective positive transfer from source to target. Some limitations of the work contained herein have been identified and could be the focus of future work: (*1*) for the MMD statistic, which was the statistical distance used to generate the bulk of the results presented herein, the hyper-parameters of the kernel function were not optimized; MMD hyper-parameter tuning is an active research area and falls beyond the scope of the current work; moreover, (*2*) the transfer learning methodology proposed herein was only tested on two pumps that operate jointly within the same power station—future work could further generalize the proposed approach to transfer additional information across more assets from a fleet of power stations.

Supplementary material. The supplementary material for this article can be found at http://doi.org/10.1017/dce.2025.3.

Data availability statement. For any queries related to the data used and analysed in this study, please contact the corresponding author.

Acknowledgments. The authors acknowledge the Lloyd's Register Foundation and The Alan Turing Institute Data-Centric Engineering Programme for supporting this work.

Author contribution. Conceptualization: Z.G. and A.Y.; Data curation: A.Y.; Formal analysis: Z.G. and A.Y.; Investigation: Z.G. and A.Y.; Methodology: Z.G., A.Y., B.B., and B.S.; Software: Z.G. and A.Y.; Supervision: A.D., S.M., and B.S.; Visualization: Z.G.; Writing—original draft: Z.G. and A.Y; Writing, review and editing: B.B., B.S., S.M., and A.D.; all authors approved the final submitted draft.

Funding statement. This work was also supported by the Engineering and Physical Sciences Research Council under grant EP/R004889/1 "Delivering Enhanced Through-Life Nuclear Asset Management.".

Competing interest. None

Ethical standard. The research meets all ethical guidelines, including adherence to the legal requirements of the study country.

References

Aminikhanghahi S and Cook D (2017) A survey of methods for time series change point detection. *Knowledge and Information Systems 51*, 339–367.

Bhattacharyya A (1943) On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society 35*, 99–109.

Briol F-X, **Barp A**, **Duncan AB and Girolami M** (2019) Statistical inference for generative models with maximum mean discrepancy. *arXiv preprint arXiv:1906.05944*.

Chae YH, **Kim SG**, **Kim H**, **Kim JT, and Seong PH** (2020) A methodology for diagnosing fac induced pipe thinning using accelerometers and deep learning models. *Annals of Nuclear Energy 143*, 107501.

Chattopadhyay R, **Sun Q**, **Fan W**, **Davidson I**, **Panchanathan S and Ye J** (2012) Multisource domain adaptation and its application to early detection of fatigue. *ACM Transactions on Knowledge Discovery from Data 6*(4), 1–26.

Chen F-C and Jahanshahi MR (2018) Nb-CNN: Deep learning-based crack detection using convolutional neural network and naïve bayes data fusion. *IEEE Transactions on Industrial Electronics 65*(5), 4392–4400.

Coble J, **Ramuhalli P**, **Bond L**, **Hines J and Upadhyaya B** (2015) A review of prognostics and health management applications in nuclear power plants. *International Journal of Prognostics and Health Management 6*, 1–22.

Costello JJA (2019) *Machine learning techniques for the health monitoring of rotating machinery in nuclear power plants.* PhD thesis, University of Strathclyde.

Costello JJA, **West GM and McArthur SDJ** (2017) Machine learning model for event-based prognostics in gas circulator condition monitoring. *IEEE Transactions on Reliability 66*(4), 1048–1057.

Daumé IIIH (2007) Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Prague, Czech Republic: Association for Computational Linguistics, pp. 256–263.

Deng C, **Ji X**, **Rainey C**, **Zhang J and Lu W** (2020) Integrating machine learning with human knowledge. *iScience 23*(11), 101656.

Dragunov A, **Saltanov E**, **Pioro I**, **Kirillov P and Duffey R** (2015) Power cycles of generation iii and iii+ nuclear power plants. *Journal of Nuclear Engineering and Radiation Science 1*(2), 021006.

Duan L, **Tsang IW and Xu D** (2012a) Domain transfer multiple kernel learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence 34*(3), 465–479.

Duan L, **Xu D and Chang S-F** (2012b) Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1338–1345.

Endres D and Schindelin J (2003) A new metric for probability distributions. *IEEE Transactions on Information Theory 49*(7), 1858–1860.

Glorot X, **Bordes A and Bengio Y** (2011) Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th International Conference on International Conference on Machine Learning (ICML'11)*, pp. 513–520.

Gretton A, **Borgwardt KM**, **Rasch MJ**, **Schölkopf B and Smola A** (2012) A kernel two-sample test. *Journal of Machine Learning Research 13*(25), 723–773.

Gupta P, **Malhotra P**, **Narwariya J**, **Vig L and Shroff GM** (2020) Transfer learning for clinical time series analysis using deep neural networks. *Journal of Healthcare Informatics Research 4*, 112–137.

He Q, **Pang PC-I and Si YW** (2020) Multi-source transfer learning with ensemble for financial time series forecasting, pp. 227–233.

Idowu S, **Strüber D and Berger T** (2022) Asset management in machine learning: State-of-research and state-of-practice. *ACM Computing Surveys 55*(7), 1–35.

Ismail Fawaz H, **Forestier G**, **Weber J**, **Idoumghar L and Muller P-A** (2018) Transfer learning for time series classification. In *IEEE International Conference on Big Data*, pp. 1367–1376.

ISO (2002) Condition monitoring and diagnostics of machines — vibration condition monitoring — part 1: general procedures. *International Organization for Standardization 13373*, 1–51.

Killick R, **Fearnhead P and Eckley I** (2012) Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association 107*, 1590–1598.

Kim JM, **Lee G**, **Lee C and Lee SJ** (2020) Abnormality diagnosis model for nuclear power plants using two-stage gated recurrent units. *Nuclear Engineering and Technology 52*(9), 2009–2016.

Kullback S and Leibler RA (1951) On Information and Sufficiency. *The Annals of Mathematical Statistics 22*(1), 79–86.

Laptev NP, **Yu J and Rajagopal R** (2018) Reconstruction and regression loss for time-series transfer learning. In *Proceedingsof the Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD) and the 4th Workshop on the Mining and LEarning from Time Series (MiLeTS)*. London, UK vol. *20*, pp. 1–8.

Liao L and Ahn H-i (2016) Combining deep learning and survival analysis for asset health management. *International Journal of Prognostics and Health Management 521*, 436–444.

Long M, **Wang J**, **Ding G**, **Pan SJ and Yu PS** (2014) Adaptation regularization: A general framework for transfer learning. *IEEE Transactions on Knowledge and Data Engineering 26*(5), 1076–1089.

Matias P, **Folgado D**, **Gamboa H and Carreiro A** (2021) Time series segmentation using neural networks with cross-domain transfer learning. *Electronics 10*(15), 1805.

Mumaw RJ, **Roth EMJVK and Burns CM** (2000) There is more to monitoring a nuclear power plant than meets the eye. *Human Factors 42*, 36–55.

Pan SJ, **Tsang IW**, **Kwok JT and Yang Q** (2011) Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks 22*(2), 199–210.

Rényi A (1960) On measures of information and entropy. Jerzy, Neyman, In *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics, and Probability*. Berkeley: University of California Press, p. 547–561.

Sagheer A, **Hamdoun H and Youness H** (2021) Deep lstm-based transfer learning approach for coherent forecasts in hierarchical time series. *Sensors 21*(13), 4379.

Santhosh T, **Gopika V**, **Ghosh A and Fernandes B** (2018) An approach for reliability prediction of instrumentation and control cables by artificial neural networks and weibull theory for probabilistic safety assessment of npps. *Reliability Engineering and System Safety 170*, 31–44.

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press

Savitzky A and Golay MJE (1964) Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, *36*(8), 1627–1639.

Solanki R, **Kulkarni HD**, **Singh S**, **Varde P and Verma A** (2020) Reliability assessment of passive systems using artificial neural network based response surface methodology. *Annals of Nuclear Energy 144*, 107487.

Surucu O, **Gadsden SA and Yawney J** (2023) Condition monitoring using machine learning: A review of theory, applications, and recent advances. *Expert Systems with Applications 221*, 119738.

Székely GJ (2002) E-statistics: The energy of statistical samples. *Technical Report BGSU No 02-16.*

Tommasi T, **Orabona F and Caputo B** (2010) Safety in numbers: Learning categories from few examples with multi model knowledge transfer, pp. 3081–3088.

Truong C, **Oudre L and Vayatis N** (2020) Selective review of offline change point detection methods. *Signal Processing 167*, 107299.

Vaserstein LN (1969) Markov processes over denumerable products of spaces, describing large systems of automata. *Problemy Peredachi Informatsii 5*(3), 64–72.

Weiss KR, **Khoshgoftaar TM and Wang D** (2016) A survey of transfer learning. *Journal of Big Data 3*, 1–40.

Yao Y and Doretto G (2010) Boosting for transfer learning with multiple sources. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1855–1862.

Ye R and Dai Q (2018) A novel transfer learning framework for time series forecasting. *Knowledge-based Systems 156*, 74–99.

Ye R and Dai Q (2021) Implementing transfer learning across different datasets for time series forecasting. *Pattern Recognition 109*, 107617.

Young A, **West G**, **Brown B**, **Stephen B**, **Duncan A**, **Michie C and McArthur SD** (2022) Parameterisation of domain knowledge for rapid and iterative prototyping of knowledge-based systems. *Expert Systems with Applications 208*, 118169.

Zellinger W, **Grubinger T**, **Zwick M**, **Lughofer E**, **Schöner H**, **Natschläger T and Saminger-Platz S** (2020) Multi-source transfer learning of time series in cyclical manufacturing. *Journal of Intelligent Manufacturing 31*, 777–787.

Zio E (2022) Prognostics and health management (phm): Where are we and where do we (need to) go in theory and practice. *Reliability Engineering & System Safety 218*, 108119.

Cite this article: Ghauch Z, Young A, Brown BD, Stephen B, West G, McArthur S and Duncan A (2025). Predictive monitoring of asset populations in a rotating plant under operational uncertainty: a transfer learning approach. *Data-Centric Engineering*, 6, e21. doi:10.1017/dce.2025.3

https://doi.org/10.1017/dce.2025.3 Published online by Cambridge University Press