Dynamically Persistent Remote Inference of Nuclear Facility Activity: Challenges and Approach

Lee Burke, Jackson Chin, Dan Rosa de Jesús, Carlos Gonzalez Rivera, Jereme Haack, Romarie Morales Rosado

Pacific Northwest National Laboratory, Richland, WA, USA

Abstract

Monitoring and characterization of nuclear facilities is an essential activity of nuclear nonproliferation, materials control, and safeguards. Such inferences are best supported by plentiful, persistent, close-range sensors operating under control of the inference system, but those resources are not always available for real-world nonproliferation problems. We identify challenges and associated mitigation strategies related to integrating few, non-persistent, remote, third-party sensors into an autonomous monitoring and inference system. We present promising results from applying a prototype ML/AI system employing the proposed strategies to two testbed facilities and discuss ongoing efforts to improve knowledge management and update, uncertainty quantification, and ML/AI model interpretation and explanation.

1 Introduction

The Persistent DyNAMICS (Dynamic Nuclear Activity Monitoring through Intelligent Coordinated Sensing) project aims to provide persistent awareness of activities involving nuclear fuel cycle production and/or materials. The project engages multiple interdisciplinary teams toward this goal, architecting a system capable of intelligently coordinated collection and interpretation of information at several sites representing different steps in the nuclear fuel cycle. This work describes progress from one of the interpretation teams: efforts to make sense of information collected in the challenging data environment inherent in the Persistent DyNAMICS use cases.

The remainder of this paper presents an overview of that use case and what makes it so challenging, introduces our approach to addressing the challenges, shows results applying that approach to two testbed facilities, and concludes with a discussion of future work.

2 Dynamically Persistent Remote Inference of Nuclear Facility Activity

The task of the interpretation team is, in part, to answer high-level questions about activities at a target facility. We identify four categories of user questions which require different approaches. We go on to identify two primary challenges to achieving that goal: a dearth of representative training data and characteristics of collected data.

2.1 User Questions

First, we make distinctions between activity hypotheses; given a set of predicted possible processes, which are running? Answering this question relies heavily on defining what those possible processes could be; nuclear activities are typically complex, and delineating between processes is not trivial, especially when they overlap in time, location, or resources.

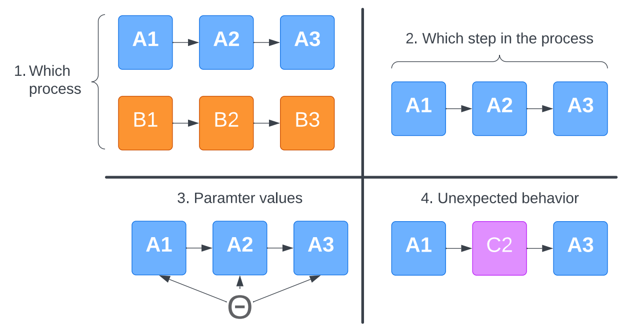

Figure 1. Graphical depiction of the four categories of user questions.

Second, assuming at least partial knowledge of the process, we identify in which step or temporal state of the process the facility is operating. Answering this second question does not always require answering the first; many processes share common states such as distinguishing between "operating" and "not operating". Indeed, at the reactor testbed site, we answer the process hypothesis question by first determining how long the reactor was in the "irradiating" state.

Third, we estimate real- or ordinal-valued characteristics or parameters of the process, such as material throughput. This has proven very challenging in the constraints of our use case and therefore has been demonstrated only in simulation.

Finally, we detect anomalous behavior of the target. We posit two broad categories of anomalies: behavior or activity at the site that is outside of normal operations (such as a reactor "scram") and behavior that an operator at the site might consider normal but that we did not predict as a likely occurrence. The first sort of anomaly is difficult to foresee, and we try only to build resilience and uncertainty quantification into the inference system. On the other hand, anomalies of the second sort require manual follow-up: ideally, we would learn from the occurrence and update our predictions of likely behavior. We refer to this predict-observe-learn cycle as "knowledge update". Knowledge update is an important target of ongoing research.

2.2 Challenge: Dearth of Representative Training Data

Real-world processes in the nuclear fuel cycle are complex and contingent on technological, political, economic, and other factors, driving large variation between and within processes, even at the same facility. Such variation limits how representative historical data gathered at a site is of future activities at that site, let alone diverse sites around the world. Automated inference systems such as machine learning (ML) rely on historical data to learn indicators and patterns of target activities that will transfer to future data. Without representative training data to learn from, indicators must be predicted manually, opening the system to new and different issues of bias and limited knowledge.

Moreover, the predicted indicators must be encoded in a way understandable to the automated inference systems. Persistent DyNAMICS has pursued two strategies to accomplish this: dynamic Bayesian networks, which encode *a priori* knowledge as a set of variables and their conditional dependencies over time, and a synthetic training data approach, which uses simulators to mitigate directly the dearth of representative training data. We focus on the latter strategy.

2.3 Challenge: Data Characteristics

Characterizing a dynamic process from remote data collection often requires distributed, multi-modal sensor networks. Unifying heterogeneous information into a common framework for interpretation requires source-specific expertise, and exfiltrating high-dimensional streaming data commands high bandwidth capacity. Edge processing provides an attractive solution to both concerns: processing data using resources co-located with a sensor avoids the overhead of data transmission and empowers expert sensor operators to process their own data.

Edge sensing assemblies deployed in such a network are often subject to physical, bandwidth or other constraints that cause sparse, irregular, and asynchronous reporting; that is, observations may be infrequent, hold to no consistent pattern, and sensors are not necessarily observing the same thing at the same time. In addition, third-party sensors may be employed which makes re-configuration in response to evolving information slow or impossible.

Prevented from observing everything all the time, Persistent DyNAMICS coordinates data collection in an event-driven manner to observe the right things at the right time. In the process of non-persistent but overlapping monitoring, sensors independently report on events of interest. In response, the coordination system automatically "tasks" sensors to collect information to confirm the event and better understand it.

The combination of edge processing and intelligent coordination has been demonstrated successfully to observe events of interest [Burke, et al., 2022]. However, this data collection approach does not fully ameliorate challenges interpreting the data; observations more likely to contain relevant information are nonetheless sparse, irregular, and asynchronous.

3 Approach

To address the dearth of representative training data, we predict likely behavior using *a priori* knowledge of facility operations and design. Leveraging subject-matter expert (SME) understanding of physical constraints, best practices, and prior observation of a site, we synthesize a high-level model of the target process. We encode this high-level model using the system dynamics model interchange language XMILE [Everlein & Chichakly, 2013] to describe an activity and its indicators as a set of stocks and flows which evolve over time according to fixed rules, possibly with stochastic elements. This variation over time represents process states (User Question Type 2). Variations on the XMILE structure constitute different activity hypotheses (User Question Type 1), and the XMILE representation can also encode parameters (User Question Type 3).

System dynamics modeling software which can read XMILE specifications, pySD [Martin-Martinez, et al. 2022], is used to simulate many possible realizations of the evolution of those stocks and flows. We refer to this first simulation step as the "Process Simulator". Those simulations are randomly sampled and transformed as if from sensors observing the indicators, what we call the "Sensed Information Simulator". The Sensed Information Simulator requires detailed design information about the sensor array, including how different behaviors will be encoded by the edge sensing assemblies (ESAs) and performance characteristics. The simulated data are used to train automated inference systems for later application to real, collected data.

Both real data and those simulated to approximate real data possess the challenging characteristics identified above. To address those issues, a multi-step pipeline is applied to transform asynchronous, irregular time series of heterogeneous sensor data into real-valued, fixed-length vectors. Before processing can begin, raw ESA output is transformed into a tabular format that is stored and processed using the Pandas Python package [Pandas Development Team, 2023]. Processing begins as follows. First, sub-sets of the data are extracted from time windows, e.g., the past 24 hours or the past week. This step could help the model focus on different time scales of activity. Next, features are extracted from each of these sub-sets using Tsfresh [Christ, et al., 2018]. The features could include statistics like mean and standard deviation, transformations like Fourier or Laplace coefficients, and trends like autoregression coefficients. Finally, general-purpose machine learning (ML) algorithms such as those from Sci-Kit Learn [Pedregosa, et al., 2011] and XGBoost [Chen & Guestrin, 2016] are applied to the feature representations of the data.

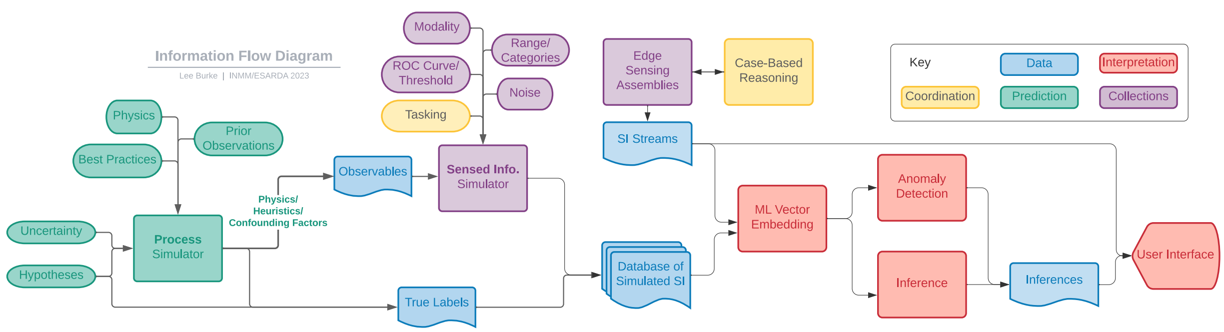

Figure 2.Information flow diagram for the proposed interpretation approach, including the Process Simulator, Sensed Information Simulator, ML Vector Embedding, Edge Sensing Assemblies, and other components.

These approaches allow for the modeling of the temporal processes associated with nuclear fuel cycle activities while bringing to bear tools from the broader field of ML, such as plug-and-play pipeline components, domain shift detection, anomaly detection, and explainability.

4 Results

We present results applying the proposed approaches at two testbed sites: a reactor and a mill. Data from the reactor site were collected first, and the analysis is accordingly more focused and more mature. Research at the mill site is ongoing.

4.1 Reactor

As was previously reported [Burke, et al., 2022], in 2021 Persistent DyNAMICS completed a multi-year deployment at a research reactor testbed. Two user questions were asked:

1. What kind of product isotope was the reactor producing during this irradiation cycle? Specifically, is the evidence consistent with the irradiation of a short-lived medical isotope (like Mo-99) or a longer-term isotope like Pu-238?
2. What state is the reactor in—maintenance (irradiating) or not irradiating?

Three visible-spectrum imaging cameras and four thermal-spectrum imaging cameras monitored for steam plumes over the reactor cooling tower while two vibration sensors monitored for water flowing through the cooling system. ESAs transformed raw data into binary (presence or absence of a plume) or ordinal (minimal, low, or high water flow) information.

Subject matter experts (SMEs) provided predictions for indicators of reactor state: the presence of steam plumes and elevated levels of water flow. Thos e predictions were fleshed out into simulations that took into account variations in timing, sensor sensitivity and specificity, and other factors. A large set of synthetic training data was generated, and a classifier trained to predict from the outputs of the remote sensing array whether the reactor was operating (User Question Type 2).

The SMEs also predicted that the type of product isotope would dictate how long the reactor would operate continuously. Accordingly, inferences of reactor operation over time were rolled-up using a simple rule-based algorithm to determine what type of product isotope was being produced (User Question Type 1).

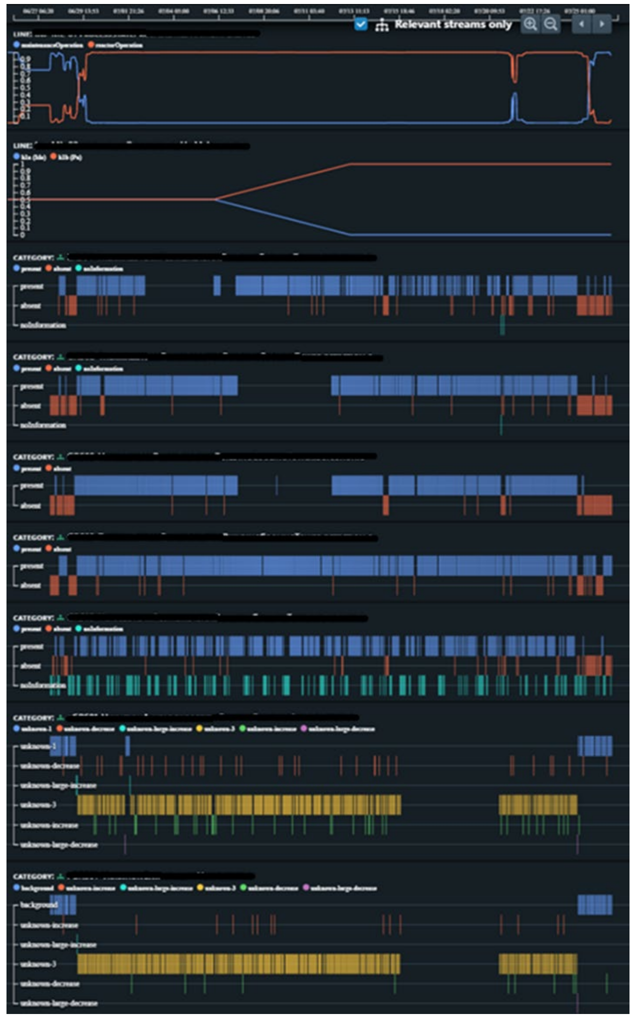

Figure 3. Screenshot of Persistent DyNAMICS live dashboard showing interpretation results and ESA data at the reactor testbed.

Table 1. The interpretation system at the reactor testbed site over four reactor cycles is evaluated against ground-truth. Metrics shown are precision, recall, and their harmonic mean, F_1-score, considering reactor "on" as the positive class.

	Precision	Recall	F_1-Score
Cycle 1	85.1	99.0	91.6
Cycle 2	95.7	96.6	96.2
Cycle 3	91.0	99.5	95.0
Cycle 4	65.4	99.8	97.6

Four reactor cycles were observed, all producing long-lived isotopes. In all four cycles, the interpretation system was able to identify the correct type of product isotope, and to infer the correct reactor state most of the time. The last two cycles were observed in real time. Quantitative metrics for the latter question are shown in Table 1. An example of edge-processed sensor data, process state inferences, and hypothesis inference are shown in Figure 3.

4.2 Mill

In 2022, Persistent DyNAMICS began a new deployment at a commercial mill that, among other things, produces powdered natural uranium. At the request of the site operator, we present only general information about activities at the mill. The primary user question posed was:

1. What state or states is the mill in, among a set of 10 steps?

Although progress has been made toward the ability to answer that question in real time, results presented here were produced after the fact with some benefit of hindsight, what we call "Playback".

4.2.1 Process State A

SMEs predicted two indicators of state A, persistently elevated heat at a specific location at the mill, and an effluent detectable with spectral sensing that is released occasionally. An ESA was built for each indicator. The machine learning pipeline described in Section 3 was configured to infer state A from the SME-predicted indicators. Figure 4 shows ESA outputs (events and observations each from a thermal and a spectral sensor), outputs from the ML model, and ground truth for state A. The ground truth identifies days in which state A was active over a period of several weeks; the thermal indicator was predicted to persist throughout the period and the effluent indicator was predicted to occur occasionally throughout the period and to disperse quickly, leading to many "absent" observations despite recent "present" detections indicating activity related to state A.

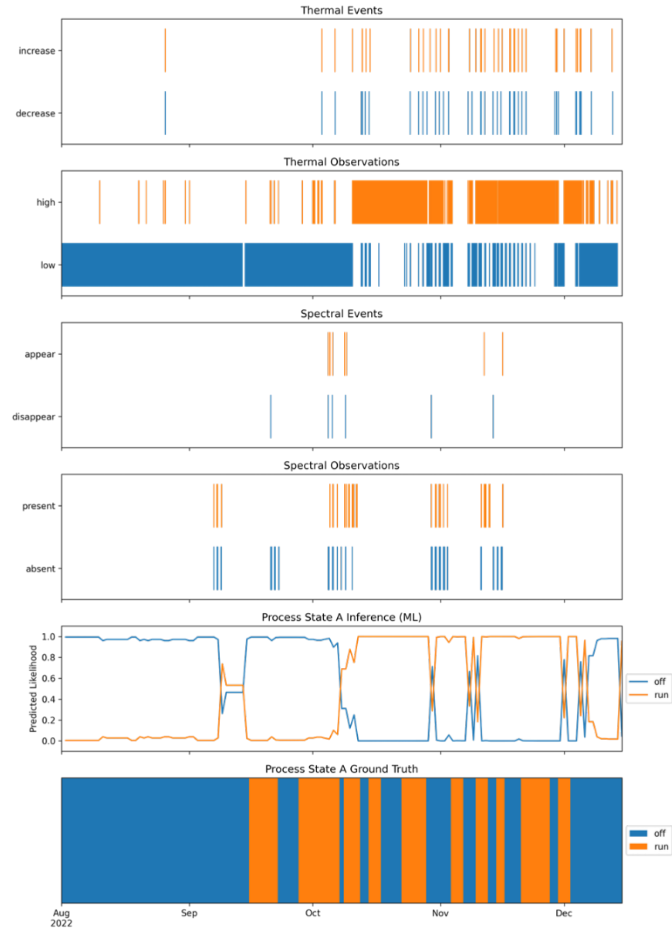

Figure 4. Playback ESA data, interpretation results, and ground truth for State A at the mill testbed.

Visually comparing the sensor outputs to the ground truth, several limitations are apparent. The thermal observation stream does not show elevated temperature until several weeks after when the ground truth would suggest it should; later analysis showed that the sensor was misconfigured until about the middle of October. The spectral observation stream does not report persistently. Finally, the event streams were intended mostly to serve the coordination element of the sensor array and are of limited utility to inference of the user question. Nonetheless, the ML pipeline performed in line with expectations, achieving the metrics presented in Table 2. It would be unfair to expect high-quality, daily inference given the indicators and sensors available.

Table 2. The interpretation system for process state A at the mill testbed is evaluated against ground truth. Metrics shown are precision recall, and their harmonic mean, F_1-score, considering state A "on" as the positive class.

Precision	Recall	F_1-Score
46.8	60.4	52.7

4.2.2 Process State B

Inference of process state B proceeded differently. SMEs predicted visible evidence of material movement at specific locations at the mill site which would indicate that state B was underway. Two ESAs were built for this indicator, a visible spectrum camera that would pinpoint when material was being moved and a synthetic aperture radar (SAR) system that would measure changes in the amount of material at the source and destination. Again, the machine learning pipeline described in Section 3 was configured to infer state B from the SME-predicted indicators.

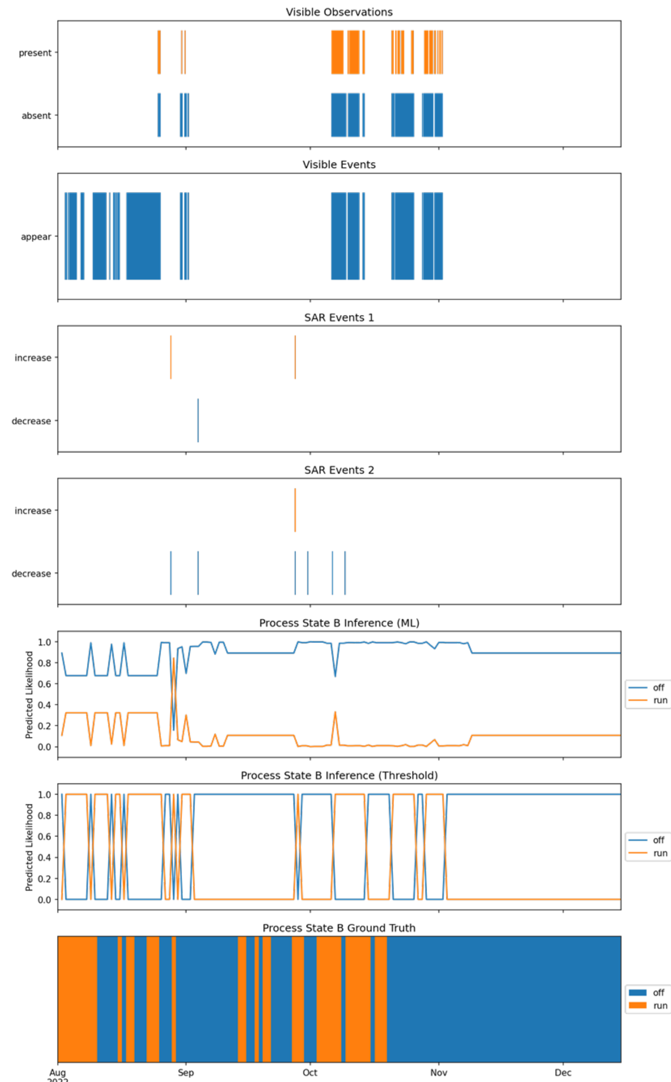

Figure 5. Playback ESA data, interpretation results from both machine learning and the threshold models, and ground truth for State B at the mill testbed.

However, inspection of the results showed that during a period when we know state B occurred several times, the machine learning approach only inferred a single positive instance. Visually comparing sensor outputs with expected behavior of the indicator, we realized that the coordination strategy for these sensors and fleeting nature of the indicator preempted meaningful negative messages. That is, all ESA data related to state B should be interpreted as indicative of state B being "on", and the lack of data as indicative of state B being "off". In response, we designed a simple threshold algorithm that predicts state B whenever three or more messages were received from any of the four data streams. This threshold model performed significantly better than the machine learning model. Figure 5 shows ESA outputs, outputs from the ML model and the threshold model, and ground truth for state B. Table 3 presents performance metrics for the two inference models.

Table 3. The interpretation systems, machine learning- and threshold-based, for process state B at the mill testbed are evaluated against ground truth. Metrics shown are precision recall, and their harmonic mean, F_1-score, considering state B "on" as the positive class.

	Precision	Recall	F_1-Score
Machine Learning	100	2.6	5.0
Threshold	48.9	56.4	52.4

5 Conclusion

This work presents progress from an interpretation team on the Persistent DyNAMICS project. We characterize the challenges to interpretation inherent in the Persistent DyNAMICS data environment and present our approach to overcoming those challenges. Results from a reactor testbed site are recapitulated, and new, preliminary results from a mill testbed site are presented.

Ongoing research focuses on improving the lifecycle of the proposed interpretation tools. Anomaly detection techniques to detect when the *a priori* knowledge our tools rely on falls short of being correct, current, and complete—improving analyses like the one performed on state B at the mill—are the subject of another paper at this conference. Uncertainty quantification improvements have been hampered by the difficult of simulating realistic uncertainty information produced by edge sensing assemblies; those efforts continue. Finally, and perhaps most importantly, automated inference systems can aid human review of high-dimensional and high-volume data but, in applications as important as nuclear nonproliferation, such systems must be understood as augmenting human intelligence, not replacing it. Techniques for improved human-machine teaming like visualization and improved machine learning explainability remain active areas of research.

6 Acknowledgement

We are grateful for the support of the National Nuclear Security Administration, Office of Defense Nuclear Nonproliferation towards this research under the Persistent DyNAMICS venture. A special thank you to the Office Program Manager Christopher Ramos for all their wisdom and guidance to make this presentation and venture a success.

7 References

Burke, Lee, Jackson Chin, Jereme Haack, Siddharth Manay, Lucas Parker, Norma Pawley, Romarie Morales Rosado, J.R. Powers-Luhn, and Dan Rosa de Jesús. "Persistent DyNAMICS." In *Proceedings of the INMM Annual Meeting*. Virtual, 2022.

Chen, Tianqi, and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–94. San Francisco California USA: ACM, 2016. https://doi.org/10.1145/2939672.2939785.

Christ, Maximilian, Nils Braun, Julius Neuffer, and Andreas W. Kempa-Liehr. "Time Series FeatuRe Extraction on Basis of Scalable Hypothesis Tests (Tsfresh – A Python Package)." *Neurocomputing* 307 (2018): 72–77. https://doi.org/10.1016/j.neucom.2018.03.067.

Eberlein, Robert L., and Karim J. Chichakly. “XMILE: A New Standard for System Dynamics.” *System Dynamics Review* 29, no. 3 (2013): 188–95. https://doi.org/10.1002/sdr.1504.

Martin-Martinez, Eneko, Roger Samsó, James Houghton, and Jordi Solé. “PySD: System Dynamics Modeling in Python.” *Journal of Open Source Software* 7, no. 78 (October 17, 2022): 4329. https://doi.org/10.21105/joss.04329.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, et al. “Scikit-Learn: Machine Learning in Python.” *Journal of Machine Learning Research* 12, no. 85 (2011): 2825–30. http://jmlr.org/papers/v12/pedregosa11a.html.

The Pandas Development Team. *Pandas-Dev/Pandas: Pandas* (version v2.0.0). Zenodo, 2023. https://doi.org/10.5281/ZENODO.3509134.