SCIENTIFIC TEST & ANALYSIS TECHNIQUES
CENTER OF EXCELLENCE

Machine Learning Model Monitoring

November 2024

Brittany Fischer, Ctr
Isaac Dobes, Ctr

To develop and apply independent, tailored Scientific Test & Analysis Techniques solutions to Test and Evaluation that deliver insight to inform better decisions.

About this Publication:
This work was conducted by the Scientific Test & Analysis Techniques Center of Excellence under contract FA8075-18-D-0002, Task FA8075-21-F-0074.

For more information:
Visit, www.AFIT.edu/STAT
Email, AFIT.ENS.STATCOE@us.af.mil
Call, 937-255-3636 x4736

Technical Reviewers:
Corinne Stafford
Wayne Adams

Scientific Test & Analysis Techniques Center of Excellence
2950 Hobson Way
Wright-Patterson Air Force Base, Ohio

Version: 1, FY25

Abstract

Artificial intelligence (AI) has the potential to improve the ability and function of military systems and operations and is critical to maintaining a competitive edge within the DOD. AI is the ability of machines to perform tasks that would typically require human intelligence. It usually involves machine learning (ML), which is a subset of AI. ML models are developed to recognize patterns in data or to make predictions. Well-trained machine learning models can be useful once deployed, but their useful lifespan is typically very short. The model's predictive performance will inevitably degrade over time. This degradation of model performance over time can be mitigated by model retraining, but this requires determining when and how to retrain the model effectively. Methods for monitoring and retraining are still evolving, but these activities are required to ensure the model continues to make accurate predictions after deployment.

Keywords: model retraining, model monitoring, drift, machine learning

Table of Contents

Introduction

The 2023 National Defense Science and Technology Strategy states that to maintain a competitive edge, the Department of Defense (DOD) must leverage critical emerging technologies, such as artificial intelligence (AI) which has the potential to improve the ability and function of nearly all systems and operations. As such, AI is being embedded into weapons and other military systems, cybersecurity, logistics, healthcare, combat simulation, and threat monitoring (Abell, 2020).

At a high level, AI is the ability of machines to perform tasks that would typically require human intelligence. It usually involves machine learning (ML), which is a subset of AI. ML models are developed to recognize patterns in data or to make predictions. The model learns on appropriate training data, where a learning algorithm is selected that dictates how the model will learn from the training data. An ML model can be retrained on new data so that it continues to learn over time.

Machine learning models are deployed with the expectation that they will perform as well as they did during model evaluation. This assumes that future data will be similar to the data used during training. However, the distributions of features (inputs) and responses (targets) will inevitably change over time. When the distributions deviate from the training set, the machine learning model will need to be retrained. Another indication for model retraining is when model performance has degraded. Monitoring metrics over time is a direct way to track model performance when performance metrics are available. Knowing when to retrain a model requires monitoring data distributions and metrics over time after deployment. Therefore, model deployment should be regarded as a continuous process that includes model retraining so the model continues to make accurate predictions. This paper gives background on the degradation of model performance over time, defines different approaches to monitoring ML models, introduces the current methods being used to detect changes in data distributions, and discusses how model retraining can be used to mitigate effects on predictive performance.

Background

Well-trained machine learning models can be useful once deployed, but their useful lifespan is typically very short. The model's predictive performance will degrade over some period of time from changes in the environment that violate the model's assumptions. A classic example would be when there is seasonality in the data, like increases in flight demand during holidays. Another example would be the performance of a model that predicts house prices when the COVID-19 pandemic caused the significant and unexpected increase. This degradation of model performance over time is known as drift. Drift can be mitigated by model retraining, but this requires determining when and how to retrain the model effectively.

There are two primary methods available for detecting or inferring when drift occurs: monitoring model performance and monitoring changes in data. Monitoring a model performance metric over time, as shown in Figure 1, allows direct quantification of the amount of degradation. When the metric falls below a certain threshold it would trigger the need to retrain the model. Once retrained, the model performance improves as it adapts to the current data to provide more accurate outputs.

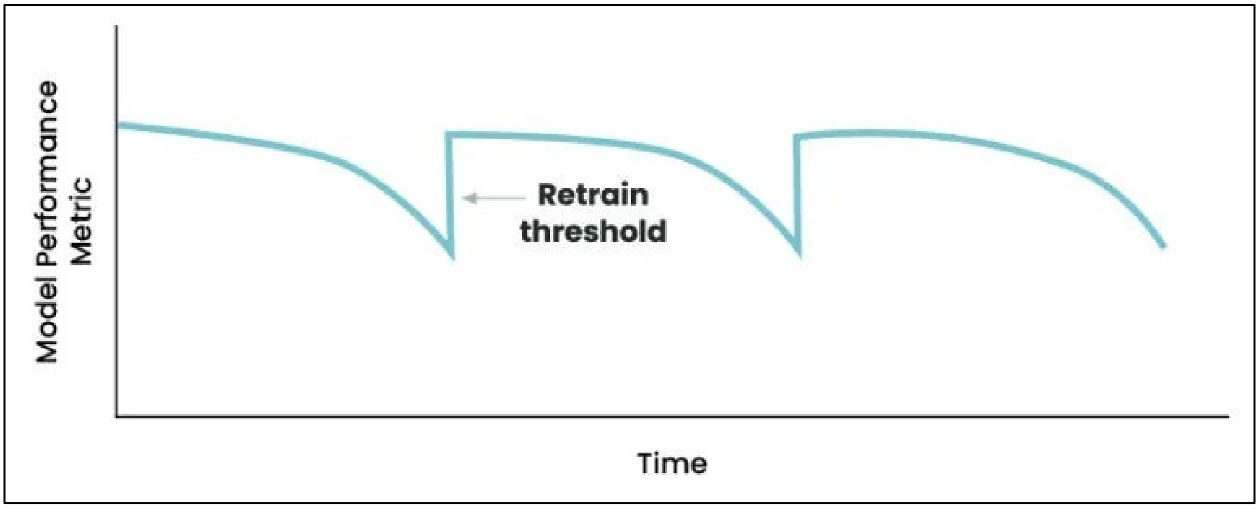


Figure 1
Trigger-Based Retraining at a Given Threshold

Note. Figure/adapted from Diliberto, L. (2021, May 6). *When Should You Retrain Machine Learning Models?*. phData. https://www.phdata.io/blog/when-to-retrain-machine-learning-models/

Alternatively, a substantial change in the distribution of targets or input data would imply model drift. Even without checking model performance, models typically degrade when data changes. This is because the model reacts to the input data it receives—changes to those inputs will most likely influence performance. Comparing the distributions of current target or input data to the distributions of data used in training should be done periodically. As an example, Figure 2 shows a deviation between a training distribution and a current distribution of target variable data. When a difference between these distributions is found, the model should be retrained with the new representative data.

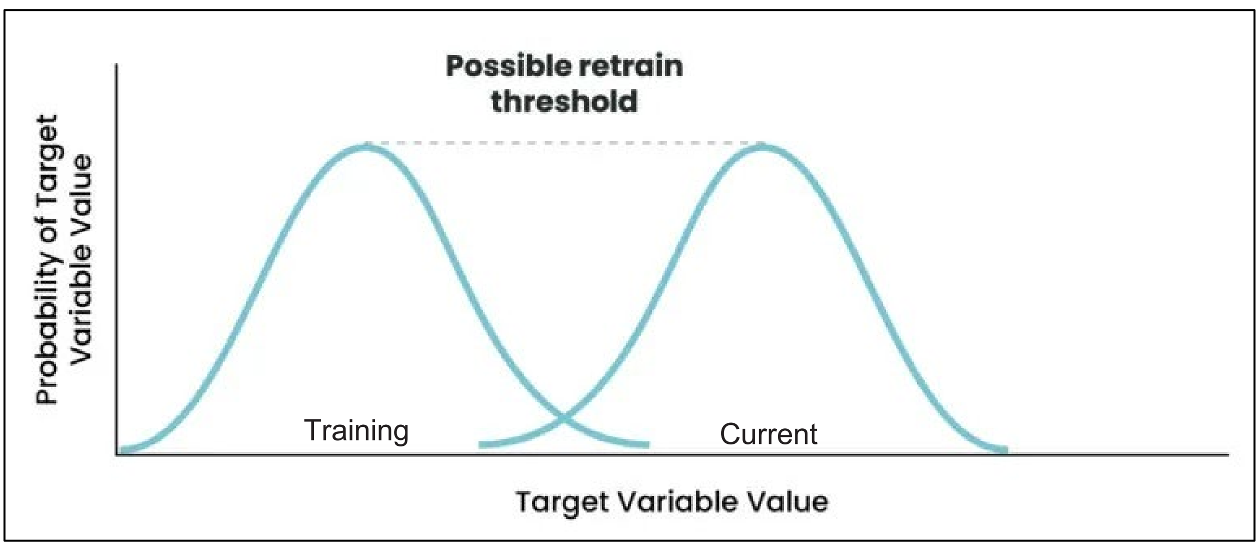


Figure 2
Current Target Variable Distribution vs. Training Target Variable Distribution

Note. Figure/adapted from Diliberto, L. (2021, May 6). *When Should You Retrain Machine*

Learning Models?. phData. https://www.phdata.io/blog/when-to-retrain-machine-learning-models/

Methods for model performance monitoring and detecting deviations in data distributions are discussed in the next section.

Methods

Model Performance Monitoring

Monitoring a performance metric is the most direct way to track model performance. When applicable, trigger-based monitoring can be straightforward to implement. However, this technique is dependent on being able to access predictions generated by the model as well as the actual truth. In other words, it requires labeled datasets, which are used in supervised ML algorithms.

There are four main types of ML algorithms: supervised, unsupervised, semi-supervised, and reinforcement. Supervised models are trained using labeled datasets to be able to make future predictions based on patterns found during training. Unsupervised learning algorithms do not use labeled data. Instead, these algorithms are used to group a dataset based on similarities, differences, and patterns among inputs. Semi-supervised learning uses a combination of labeled and unlabeled datasets to train its algorithms, and therefore has characteristics of both supervised and unsupervised ML. Reinforcement learning is based on feedback and lacks labeled data—it seeks to learn from experiences only. Monitoring model performance will only apply to supervised learning, since it explicitly uses labeled data.

The primary objective of supervised learning is to map the input variable with the output variable. Supervised learning can be further categorized as classification or regression based on the type of output variable. The model is called a classifier when it predicts a categorical response (e.g., threat/not threat), and a regressor when it predicts a continuous response (e.g., signal strength). Different metrics may be suitable for different models based on the type of model being either classification or regression.

Classification Models

Binary classification models have a positive result (e.g., hit, detected, threat) and a negative result (e.g., miss, not detected, not threat). Multiclass classification models can also be considered in binary terms (class of interest or not class of interest). Accuracy is a simple and straightforward metric for evaluating model goodness, and is defined as the ratio of correct classifications to the total number of classifications for a dataset. Another common metric used to evaluate binary classifiers is the F-score—a measure of predictive performance. It is a function of precision and recall, which measure a model's ability to correctly classify positives.

Regression Models

Additional metrics exist to assess numeric prediction. A common metric used to evaluate numeric prediction is R-squared (R^2), which is the proportion of the variation in a dependent variable (output or response) which can be explained by independent variables (inputs or factors). RMSE, the square root of the average of squared errors, is another common metric used to evaluate numeric predictions. See Stafford and Pai (2024) for more information on metrics for both classification and numeric prediction and their applicability.

Considerations

When applicable, monitoring performance metrics is straightforward, but it is not without challenges. It is important to choose an appropriate metric based on the type of model and overall objective. It is also important to define an appropriate threshold that triggers retraining. Too large a threshold can result in unnecessary model retraining and too small a threshold can result in not retraining enough. Another consideration is whether there is sufficient data to retrain the model. Monitoring may detect sizable degradation, but if the available data does not represent the current environment, retraining can be unproductive.

Deviations in Data Distributions

When performance metrics are not available, or even if they are, monitoring the distributions of target and input data can indicate if retraining is needed. Monitoring data is important for maintaining model performance. Changes in the distribution of data is referred to as data drift. One method used to detect data drift is comparing statistical measures of the current data distribution with the training data distribution. These might include the mean, variance, or correlation between the data distributions.

More formal statistical techniques are also regularly used to determine if there is a difference between the current data and training data. Knowing that the current data is similar to data used for training could lend some level of trust to a model's predictability. A few of the commonly used techniques are described in this section. These methods are all for comparing distributions to determine whether the datasets are drawn from the same population. Although these techniques are being used to detect data drift, it should be noted that they do not directly account for any time dependence of data collection, instead treating data as coming from two separate groups. For a deployed model, data is continuously being fed into the model and trends over time may occur that are obscured when simply grouping the data. This implies the need for multivariate, time streaming metrics to measure data drift. Further discussion on these challenges is included at the end of the section on Method Selection.

Kolmogorov-Smirnov Test

The Kolmogorov-Smirnov (KS) test uses a hypothesis test to determine if the training data and current data are from the same population. It is a nonparametric statistical test, meaning that it does not make any assumptions about the underlying distribution. The KS statistic measures the maximum difference between the empirical cumulative distribution functions of the training data and current data. When that difference is statistically significant, model retraining is recommended. It is important to note that the KS test is sensitive to the size of the dataset. It can detect very small deviations in larger datasets. This sensitivity may be beneficial if it's important to detect small changes but would otherwise result in an increased number of false positives—an increased rate of incorrect indications of drift. More details on applying the two-sample Kolmogorov-Smirnov test can be found at The MathWorks, Inc. (n.d.).

Kullback-Leibler Divergence

The Kullback-Leibler (KL) divergence can be used to measure the difference between the current data and the training data. This method measures how a reference probability distribution is different from a second probability distribution. The KL divergence, or relative entropy, calculation is non-negative and can only be equal to 0 when the two probability distributions are identical. Therefore, the higher the score, the greater the difference between the two distributions. In ML, the relative entropy of the current distribution with respect to the training distribution is referred to as the information gained if the current data would be used instead of the training data. Another interpretation is the amount of information lost when the training data distribution is used to approximate the current data distribution.

Jensen-Shannon Divergence

The Jensen-Shannon (JS) divergence is another method to detect differences between distributions that is based on the KL divergence. There are two main differences between these metrics: symmetry and range of values. The JS divergence is symmetric, while the KL is not. This means that the KL divergence will have different values if the training data distribution and current data distribution are swapped in its equation. The KL divergence is nonnegative ($KL \geq 0$) and the JS divergence can be normalized so that it is always bounded: $0 \leq JS \leq 1$ (Lin, 1991). In both cases, a value of 0 occurs when the distributions are identical. More details on both the KL divergence and JS divergence can be found in Nielsen (2019) or Dhinakaran (2023).

Population Stability Index

The Population Stability Index (PSI) is another symmetric measure that quantifies the difference between one probability distribution and a reference probability distribution. The PSI produces a value ≥ 0 that indicates the relative size of drift. A larger value would mean more separation between distributions. The PSI is a model monitoring metric that has been popular in financial applications. It is a measure of the stability between two samples. The two distributions are compared based on a set of cut-off points that form bins or intervals. A typical interpretation of the PSI is no population change when $PSI < 0.1$. Yurdakul (2018) provides a comprehensive discussion on the importance of selecting an appropriate binning strategy and its impact on the PSI value.

Wasserstein Distance

The Wasserstein distance is a measure of dissimilarity between two probability distributions. It is a measure of how much effort it takes to turn one distribution into another. The Wasserstein distance is also known as Earth Mover distance because it can be interpreted as the minimum energy cost of transforming a pile of dirt in the shape of one probability distribution to the shape of another probability distribution (Weng, 2017). The cost is the product of the amount of dirt moved and the distance moved.

Method Selection

Some resources have compared methods on their ability to detect drift. The five approaches introduced above are summarized in Figure 3. This section discusses how the different methods perform when drift is introduced. Considerations and challenges on using these techniques are also included.

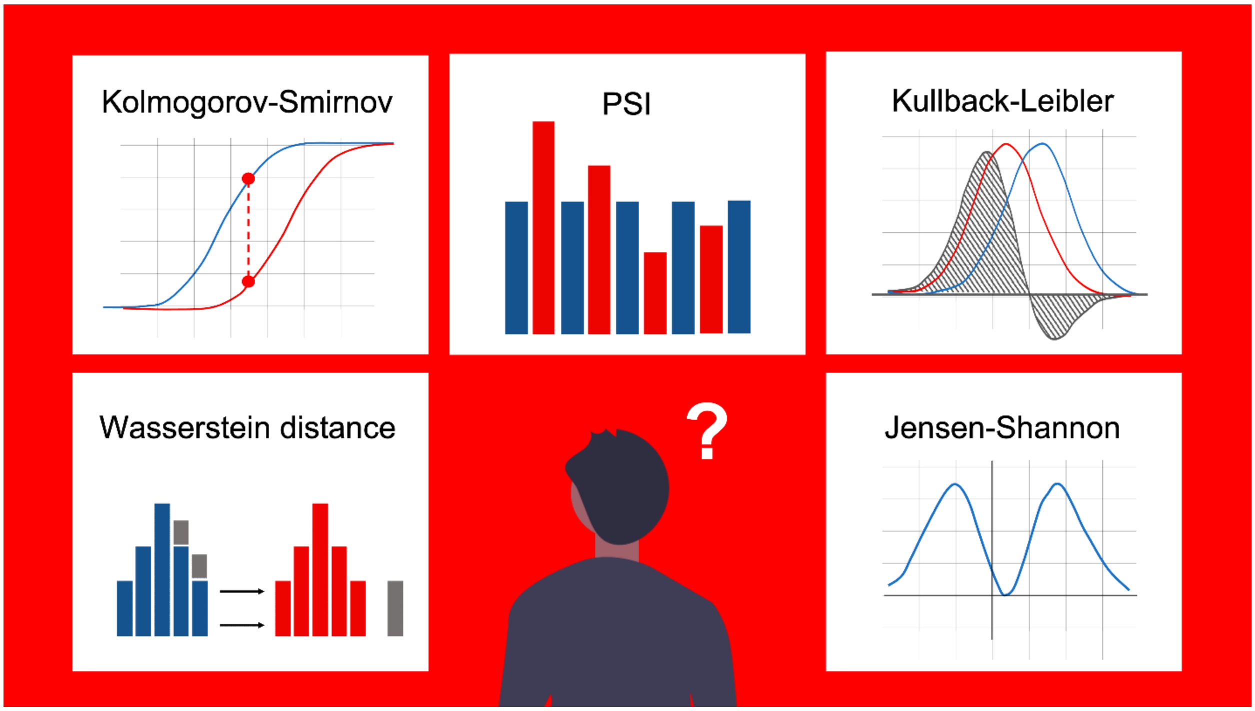


Figure 3

ML Monitoring: Which test is the best?

Note. Figure/adapted from Filippova, O., & Maliugina, D. (2022, June 20). *Which test is the best? We compared 5 methods to detect data drift on large datasets*. Evidently AI - Open-Source ML Monitoring and Observability. https://www.evidentlyai.com/blog/data-drift-detection-large-datasets

Weberman (2024) considers how the PSI compares to the JS divergence and Wasserstein distance for two normal distributions with varying means and standard deviations. Some of the outputs are displayed in Figure 4, which shows how each method responds as the mean of the monitored data set moves away from the mean of the reference data set while the variance is held constant—the reference data was sampled from $N(0, 1)$. The author shows that while the JS is able to detect a small drift, its value eventually plateaus making it less suitable for quantifying the drift. Compared to JS, Wasserstein is less sensitive to a small drift but continues to increase linearly as the drift increases. This makes it a good choice to determine how much drift is in the data. Further results show that the PSI is less sensitive to a small drift than the JS distance and less capable for quantifying drift intensity than Wasserstein since it does not increase linearly with drift.

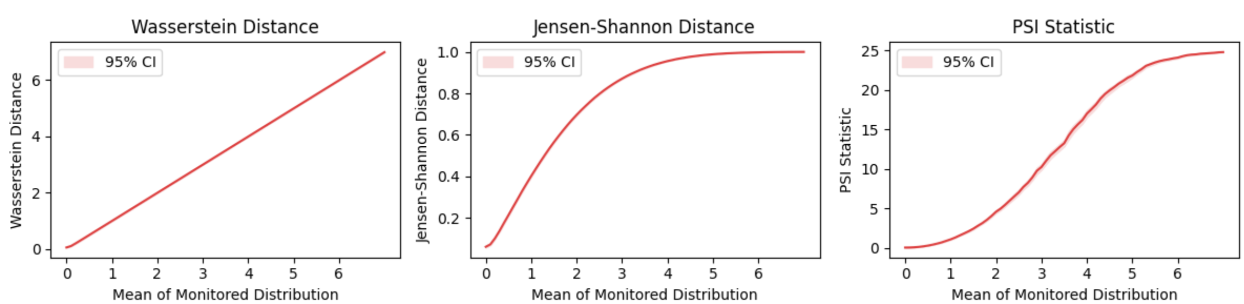
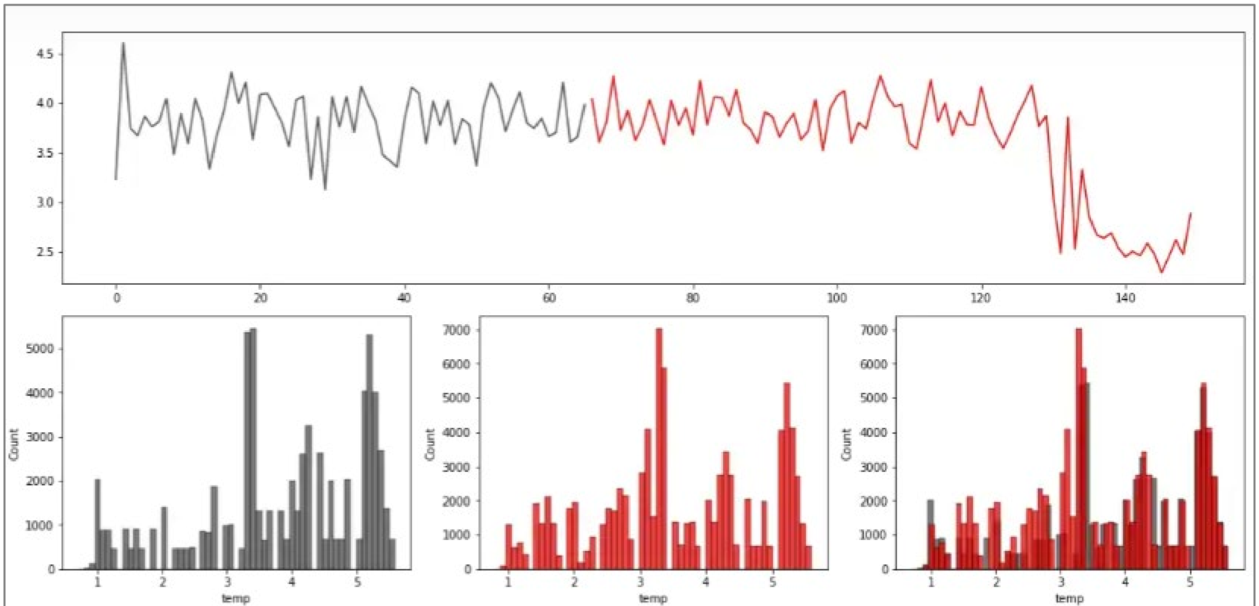


Figure 4
Univariate Methods Results for Distributions with Varying Means

Note. Figure/adapted from Weberman, M. (2024, June 3). *Population Stability Index (PSI): A Comprehensive Overview*. NannyML Blog. https://www.nannyml.com/blog/population-stability-index-psi

Filippova and Maliugina (2022) consider the different methods to detect data drift on large datasets. The authors intend to build intuition on how detection results relate to the different visual changes in data distributions. Examples are included that compare the performance of the five methods: KS, PSI, KL, JS, and Wasserstein. Their findings show that all methods are in agreement with a big enough change in a large dataset. Relatively minor changes, however, are only detected by the KS test. There are varying results among methods when a change in the trend of data was considered. One example of their output is provided in Figure 5. The top graph shows the data plotted over time or index. The two distributions are colored gray and red. In this example, the KS ($\text{p-value} < 0.05$), PSI (≥ 0.1), and JS (≥ 0.1) detected drift while the KL (< 0.1) and Wasserstein (< 0.1) did not, as given in the results table.



	stat_test	drift_score	is_drifted
0	K-S p_value	0.0	True
1	PSI	0.14468	True
2	Kullback-Leibler divergence	0.068238	False
3	Jensen-Shannon distance	0.133564	True
4	Wasserstein distance (normed)	0.069508	False

Figure 5
Data Drift Example: A Change in the Trend

Note. Figure/adapted from Filippova, O., & Maliugina, D. (2022, June 20). *Which test is the best? We compared 5 methods to detect data drift on large datasets*. Evidently AI - Open-Source ML Monitoring and Observability. https://www.evidentlyai.com/blog/data-drift-detection-large-datasets

Things to consider when choosing a monitoring method include the size of drift to detect for the ML model and the sample size to be tested. The impact of the model performance drop will influence the size of drift to detect. The cost of degradation in performance helps determine if it is more important to reduce falsely detecting drift or possibly missing when drift has occurred. Understanding the risks of false-positive and false-negative drift detection rates can help to set thresholds for detection.

Challenges

All of these current techniques assume that the distribution of the current data is not a function of time. In other words, these methods are for stationary, or batch processing, data, and do not measure patterns over time. An important comment on the example in Figure 5 that the authors acknowledge is that the outcomes would vary again if different samples of data were used (i.e. different time windows of data). The examples that consider a change in trend is why something more similar to statistical process control (SPC) should be used for monitoring data drift when the data is continuously streamed to the model over time. SPC is a technique used to monitor and evaluate the quality of a process. If streaming data is aggregated such that the timestamps are removed, then it's likely the most important component of drift has been removed. Another potential problem with the methods currently being used is that they compare one variable at a time, but these systems are often not single variable. Therefore, these systems need multivariate, time streaming metrics to measure drift. SPC is a possible solution, and its application to machine learning drift detection is an ongoing area of study (see Zamzmi et al., 2024).

Model Retraining

Model retraining can be used to mitigate effects on a model's predictive performance. The two main approaches to model retraining are trigger-based and periodic. Trigger-based retraining is dynamic and involves retraining when the model's performance drops below predetermined performance thresholds or when data drift is detected. Periodic retraining is a fixed approach where the model is retrained at specified time intervals, such as weekly or monthly.

When available, monitoring a performance metric is the most direct way to track model performance. In this case, trigger-based retraining can be implemented. When performance metrics are not available, or even if they are, detecting drift in the distributions of target and input data can indicate the need to retrain a machine learning model. Drift detection could be set up for periodic retraining or trigger-based retraining potentially through the use of drift detection algorithms. Both trigger-based and periodic approaches to model retraining still require the user to determine a threshold for when it is necessary to retrain. Choosing a threshold should be based on the cost of degradation in performance to maintain a reasonable number of false-positives or false-negatives. In this case, a false-positive means retraining when it might not be necessary, and a false-negative means missing drift when it has occurred.

When drift is detected through performance monitoring or changes in distributions, a good first step is to consider if there has been a change in the data quality (e.g., how the data is collected). If the data collection and processing is acceptable, then retraining the model can mitigate the drift. Determining the new training data depends on the ML problem. If a data stream is dynamic, then the model should be retrained frequently, replacing older data with the new data as it becomes available. If an old dataset is simply not representative of the new environment, then it is better to replace the entire dataset. However, it is important to ensure that sufficient data is available for retraining, otherwise there may be little to no effect on model performance. Some other considerations for retraining include the cost to retrain and the impact—this relates to deciding how to manage prioritizing false-positives and false negative rates.

Discussion and Conclusions

Monitoring a model performance metric over time allows direct quantification of drift. A change in the distribution of output or input data would imply model drift. Directly monitoring model performance, when possible, and monitoring changes to data distributions are crucial to maintaining model performance beyond deployment. Many methods used in practice to detect changes to input or output data, such as the methods described in the previous sections, are based on detecting differences in distributions. However, the results of these methods are dependent on the sample distributions used. Furthermore, none of these methods for comparing distributions are designed to measure patterns or trends in data over time. Monitoring data continuously over time requires something more similar to statistical process control. Methods to detect differences between distributions ignore the information available from the time data.

Different machine learning models require different maintenance plans. However, drift is inevitable and, therefore, model retraining should be a consideration for ML model maintenance plans. Model retraining only involves changing the training data set. The frequency of retraining a model and how to determine the new training set depends on the modeling problem. Trigger-based and periodic retraining approaches are available. Determining thresholds or the periodicity should be based on the performance cost and how quickly the distributions are changing. Another consideration is how much new data is available. Even if drift is detected, retraining with insufficient data may not be very useful if the new training set doesn't represent the new model environment.

ML models are deployed with the expectation that they will perform as well as they did during model evaluation. The reality is that model performance inevitably will degrade over time. Therefore, the development of ML models cannot end after deployment. Although there seems to be consensus on this belief, methods for monitoring and retraining are still evolving. Still, having some sort of maintenance plan for an ML model beyond deployment will be crucial to its success. ML, and other emerging technologies, have the potential to improve the ability and function of systems and operations and are critical to maintaining a competitive edge within the DOD. The development and deployment of ML models should be regarded as a continuous process that includes model retraining so the model continues to perform as it was intended.

References

Abell, N. (2020, October 2). *7 Key Military Applications of Machine Learning*. Medium. https://medium.com/@nqabell89/7-key-military-applications-of-machine-learning-9818dfa2ea86

Dhinakaran, A. (2023, March 2). *How to Understand and Use the Jensen-Shannon Divergence*. Medium. https://towardsdatascience.com/how-to-understand-and-use-jensen-shannon-divergence-b10e11b03fd6

Diliberto, L. (2021, May 6). *When Should You Retrain Machine Learning Models?*. phData. https://www.phdata.io/blog/when-to-retrain-machine-learning-models/

Filippova, O., & Maliugina, D. (2022, June 20). *Which test is the best? We compared 5 methods to detect data drift on large datasets*. Evidently AI - Open-Source ML Monitoring and Observability. https://www.evidentlyai.com/blog/data-drift-detection-large-datasets

Lin, J. (1991). Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, *37*(1), 145–151. https://doi.org/10.1109/18.61115

Nielsen, F. (2019). On the Jensen–Shannon Symmetrization of Distances Relying on Abstract Means. *Entropy*, *21*(5), 485. https://doi.org/10.3390/e21050485

Stafford, C. and Pai, R. (2024). *Cross-Validation for Machine Learning Models*. Best Practice. Scientific Test & Analysis Techniques Center of Excellence.

The MathWorks, Inc. (n.d.). *Two-sample Kolmogorov-Smirnov Test*. MathWorks. https://www.mathworks.com/help/stats/kstest2.html

Weberman, M. (2024, June 3). *Population Stability Index (PSI): A Comprehensive Overview*. NannyML Blog. https://www.nannyml.com/blog/population-stability-index-psi

Weng, L. (2017, August 20). *From GAN to WGAN*. Lil'Log. https://lilianweng.github.io/posts/2017-08-20-gan/#what-is-wasserstein-distance

Yurdakul, B. (2018). *Statistical Properties of Population Stability Index* (dissertation). 3208. https://scholarworks.wmich.edu/dissertations/3208

Zamzmi, G., Venkatesh, K., Nelson, B., Prathapan, S., Yi, P. H., Sahiner, B., & Delfino, J. G. (2024, February 12). *Out-of-Distribution Detection and Data Drift Monitoring using Statistical Process Control*. arXiv.org. https://arxiv.org/abs/2402.08088