Application of machine learning methods to pathogen safety evaluation in biological manufacturing processes

Shyam Panjwani1 | Ivan Cui1 | Konstantinos Spetsieris1 | Michal Mleczko1 | Wensheng Wang2 | June X. Zou2 | Mohammad Anwaruzzaman2 | Shawn Liu2 | Roger Canales3 | Oliver Hesse4

1Engineering & Technology, Bayer Pharmaceuticals, Berkeley, California
2Pharma Biotech - Pathogen Safety, Bayer Pharmaceuticals, Berkeley, California
3Bayer Business Services - Pharma R&D ITO, Bayer Pharmaceuticals, Berkeley, California
4Pharma Biotech - Lab Automation & Data Management, Bayer Pharmaceuticals, Berkeley, California

Abstract
The production of recombinant therapeutic proteins from animal or human cell lines entails the risk of endogenous viral contamination from cell substrates and adventitious agents from raw materials and environment. One of the approaches to control such potential viral contamination is to ensure the manufacturing process can adequately clear the potential viral contaminants. Viral clearance for production of human monoclonal antibodies is achieved by dedicated unit operations, such as low pH inactivation, viral filtration, and chromatographic separation. The process development of each viral clearance step for a new antibody production requires significant effort and resources invested in wet laboratory experiments for process characterization studies. Machine learning methods have the potential to help streamline the development and optimization of viral clearance unit operations for new therapeutic antibodies. The current work focuses on evaluating the usefulness of machine learning methods for process understanding and predictive modeling for viral clearance via a case study on low pH viral inactivation.

KEYWORDS
biological manufacturing process, low pH viral inactivation, machine learning, monoclonal antibody, pathogen safety

1 | INTRODUCTION

The majority of recombinant therapeutic proteins are manufactured by large-scale culture of animal or human derived host cells that are genetically engineered to express the genes of interest. Typical host cells include the baby hamster kidney (BHK-21), Chinese hamster ovary (CHO), mouse myeloma (NS0), and some potential human cell lines that is, HEK293. These cells contain endogenous retrovirus-like particles (ERLVPs) coding sequences (provirus like) intrinsically integrated in their chromosomes. ERLVPs are spontaneously produced in cell cultures but they are typically noninfectious and are defective due to lacking the entire coding sequence.1 Nevertheless, the capacity of the manufacturing process to clear these ERLVPs should be determined using a specific model virus. A more significant risk of contamination is imposed by the introduction of adventitious agents into the bioprocess via cell substrates, raw materials and mechanical,
environmental, personnel and process-related factors. Therefore, demonstration of effective viral removal and/or inactivation by the manufacturing process using scale-down test systems with specific and nonspecific model viruses are crucial to ensure pathogen safety of biologicals.  

Specifically, recombinant human monoclonal antibody (rhumAb) therapeutics are produced using CHO cell culture in upstream process and purified through a series of steps (unit operations) in downstream process. A typical downstream purification process is shown in Figure 1. The initial harvested cell culture fluid (HCCF) is first loaded onto a Protein A column, then the captured rhumAb is eluted with a solution at low pH after thorough washes with equilibration and high salt wash buffers. Subsequently, the eluate is adjusted to low pH ranging from 3.7 to 3.9 as most of our antibody molecules are not very stable at pH buffers. Subsequently, the eluate is then held for a duration time no less than 2 h to inactivate enveloped viruses and achieve a value of \( \log_{10} \) reduction factor (LRF) \( \geq 5.0 \) that is close to the value of LRF achieved at a pH \( \leq 3.6 \) for a 30 min treatment. The eluate post low pH viral inactivation is then neutralized and further polished by anion or cation exchange column or membrane adsorber chromatographic steps to remove impurities and potential viruses or model viruses. The product intermediate will be further filtered through viral filter to remove potential viruses.

The capacity and robustness of the viral clearance by a manufacturing process must be validated by viral clearance studies using a scale-down model. The results of viral clearance studies are provided a viral safety assurance of a biological product made using the process and can be accepted by the health authorities to gain approval for clinical development and/or commercial approval. The validation of viral clearance studies should be performed in compliant with GLP (good laboratory practices) guidance using scale-down test system representing large scale cGMP manufacturing process, so that the viral clearance results, expressed as a \( \log_{10} \) virus reduction factor (LRF), obtained from the scale-down test systems can be extrapolated to the viral clearance by the cGMP process. Such studies require significant effort and resources invested in wet laboratory experiments for process characterization to determine the appropriate conditions for effective viral clearance. Therefore, the use of machine learning (ML) methods has the potential to help streamline the development of pathogen safety processes for new therapeutic molecules, using existing and validated results of historical records. For example, establishing the relationship between LRF and the viral clearance process parameter space, could speed up development and optimization for a new therapeutic protein or antibody product at the initial stage of drug development. Internal and external viral clearance experimental data could be compiled into a process development database. Therefore, it is valuable to support the establishment of a platform for viral clearance in order to minimize the development effort at the very early stage. Specifically, in the early stage of process development, scientists need to make decisions regarding the experimental design. Typically, this requires review of past experiment results and initial wet lab experiments, which may take weeks to complete. Thus, the use of machine learning models trained with historical data can substantially expedite early phase development as model-based insights can be generated within hours. This is aligned with the several regulatory guidelines encouraging the adoption of mathematical models in order to improve the effectiveness and efficiency of pharmaceutical process development.

Some attempts have been made in the past to analyze such data with traditional statistical methods. The current work focuses on evaluating the usefulness of machine learning methods to support the development of processes for viral clearance of biological therapeutic molecules. Specifically, a case study is presented for low pH viral inactivation unit operation. First, the utility of unsupervised ML for contextualization and process understanding of experimental results is discussed. Additionally, several supervised ML algorithms are evaluated regarding their predictive ability for viral clearance as well their interpretability.

In the subsequent sections, first, the experimental set-up for viral inactivation, data collection, data preprocessing, and advanced machine learning methods will be presented. Second, data-driven models will be developed using various machine learning methods and evaluated in terms of their usefulness for better process understanding and viral clearance prediction. Finally, the outcome of this evaluation will be summarized and the need for industrialization of such models will be briefly discussed.

## 2 | MATERIALS AND METHODS

### 2.1 | Experimental set-up

The viral clearance data for low pH viral inactivation used in this paper was generated at Bayer pathogen safety bio-safety level-2 laboratory using scale-down viral clearance test systems and representative model viruses. The test systems were proportionally scaled down from the respective cGMP unit operations of various purification processes. The model viruses used for viral clearance studies were xenotropic murine leukemia virus (X-MuLV) and pseudorabies virus (PRV) as specific model viruses to represent the EVRLPs harboring within CHO cells.

Performance parameters for each scale-down process step are identical to those of the corresponding cGMP large scale unit operation. Buffers for reagents were either obtained directly from Bayer manufacturing facility or prepared in Pathogen Safety laboratory according to the buffer specifications.

The model virus was spiked into the pH-adjusted load material prior to start of each viral inactivation experimental run. Fractions of the spiked load samples were collected at 0, 30, 60, 120, and 240 min after an incubation at 18 °C and were subjected to virus titration by a TCID<sub>50</sub> infectivity assay to determine the amount of infectious virus remained in each sample after the low pH inactivation. Viral clearance result, or \( \log_{10} \) reduction factor (LRF), was calculated as the logarithm of the ratio of the total virus amount in the virus-spiked load over the

![FIGURE 1 Schematic of purification process](image-url)
total virus amount in the postprocess material (i.e., the sample after the low pH treatment). A mock spike run (no virus spike) was performed to evaluate in-process samples for potential toxicity and interference to virus titration methods.

The input and output parameters for each viral clearance study vary according to the process and the unit operation. The focus of this work is low pH viral inactivation unit operation. For this unit operation, the input variables are pH, Temperature, Spike ratio, Load Protein Concentration, and Load virus titer. The output is the final viral clearance result (i.e., LRF).

2.2 | Data collection and preprocessing

As presented in Table 1, the dataset comprises experimental data from Bayer’s pathogen safety laboratory and publicly available data.

Preprocessing is performed to ensure that the source data is brought into a consistent format amenable to data analysis. Data preprocessing entails steps such as mapping, pivoting, table joining, table row and column filtering, data annotation. Preprocessing is conducted through the development of workflows in KNIME Analytics Platform software.

2.3 | Machine learning algorithms

Machine learning is a collection of algorithms for data analysis that have the ability of finding patterns and getting insights from the data they are exposed to without being explicitly programmed with expected relationships. Specifically, these algorithms can be used to detect patterns and relationships in data. Some applications of these methods are clustering (detection of groupings), classification (determining group/class membership), and regression (determining relationships between inputs and continuous outputs).

The machine learning methods can be categorized into unsupervised and supervised. Unsupervised machine learning models are used to obtain an overview of the underlying data without a priori information labeling or mapping them to a target or output value. These algorithms are used to find structure or patterns in the data. Clustering is an application for unsupervised ML algorithms. In contrast to the unsupervised machine learning methods, supervised methods aim at determining a functional relationship in labeled data. Partial least squares (PLS) regression is an example of supervised ML algorithm. In the subsections below, machine learning algorithms used in this work will be briefly discussed.

2.3.1 | Principal component analysis

Principal component analysis (PCA) is an unsupervised machine learning method used to reduce the dimensionality of datasets in which collinear relationships are present. The working principle of PCA is summarizing the original data by defining new, orthogonal, latent variables called principal components (PCs). These PCs or score vectors (t) are linear combinations of the original variables (x), which can be written in matrix form as follows:

\[ T = XP \]  

where \( T = [t_1, t_2, ..., t_r]^T \) is score matrix with \( t_i \in \mathbb{R}^1 \) denoting a score vector for original variable \( x_i \in \mathbb{R}^m \), \( X = [x_1, x_2, ..., x_n]^T \) is original variables matrix with dimension \( n \times m \), and \( P = [p_1, p_2, ..., p_r] \) is loading matrix with \( p_i \in \mathbb{R}^m \) denoting a loading vector.

The reduction of dimensionality from \( m \) to \( r \) is performed in such a way that the variance explained by \( r \) PCs can be maximized. Refer to chapter 3 of Reference 12 for more details.

2.3.2 | Orthogonal PLSs discriminant analysis

Orthogonal partial least squares discriminant analysis (OPLS-DA) is a classification machine learning method. OPLS-DA is a variant of partial least squares discriminant analysis (PLS-DA) and standard PLS regression.

Similar to PLS and PLS-DA, OPLS-DA is conducted not on the original variables available in a dataset, but on fewer, orthogonal ones, called latent variables. These are linear combinations of the original variables. Briefly, the working principle of OPLS-DA algorithm is as follows:

1. First, the input matrix \( X \) is decomposed into predictive and orthogonal components to output \( Y \).

\[ X = T_P P_0^T + T_o P_0^T + E \]  

where \( T_P \) and \( T_o \) denote predictive and orthogonal scores, respectively; \( P_P \) and \( P_o \) denote predictive and orthogonal loadings, respectively and \( E \) denotes residuals \( X \) matrix.

2. Then, classification model is developed using the categorical output \( Y \) and transformed input matrix \( X \)

\[ X_P = X - T_o P_0^T \]  

Since \( Y \) is a categorical variable, it does not fit into the PLS modeling framework. Therefore, the categorical variable \( Y \) must be encoded. In case of binary (two class) classification, class 1 and 2 can be encoded as “0” and “1,” respectively. Then, a regression model (Equation (4)) is developed between the encoded output \( Y_e \) and transformed input matrix \( X \)

\[ Y_e = X_P B + F \]
where $B$ is the model coefficient matrix; $F$ is the residuals matrix.

3. Since, the regression model (Equation (4)) output $Y_e = X_eB$ is a continuous variable. $Y_e$ must be converted back to categorical variable in order to fulfill the objective of classification. This is achieved by calculating the posterior probabilities of belonging to either class. An observation is assigned to the class with the highest probability. For the binary classification, an observation is simply assigned to the class with probability greater than .5.

Refer to literature 13 for more detailed discussion about OPLS discriminant analysis.

The model quality is assessed with cross-validation as well as external datasets, if available.

### 2.3.3 Logistic regression

Logistic regression (LR) is a supervised machine learning method used for $K$ class ($K = 2$ is binary classification) classification. The classification model is defined in terms of $K - 1$ logits or log-odds:

$$\log \frac{Pr(C = i|X = x)}{Pr(C = K|X = x)} = \beta_0 + \beta_i^T x = \beta_0 + [\beta_1, \beta_2, \ldots, \beta_{K-1}]^T x_i = 1, 2, \ldots, K-1$$

(5)

where $\theta = \{\beta_0, \beta_1, \ldots, \beta_{K-1}\}$ is the set of model coefficients and $Pr(C = i|X = x)$ is the probability of belonging to $i$th class given $x$ is the vector of predictor variables.

$\beta_0 + \beta_1 x = 0$ can also be seen as a hyperplane separating $i$th class from $K$th class. If $\beta_0 + \beta_1 x > 0$, the probability of belonging to $i$th class will be higher than the probability of belonging to $K$th class. Since, the probabilities $Pr(C = i|X = x)$ for $i = 1, 2, \ldots, K$ sum to one, the class membership probability can be calculated using Equation (6).

$$Pr(C = i|X = x) = \frac{\exp(\beta_0 + \beta_i^T x)}{1 + \sum_{j=1}^{K-1} \exp(\beta_0 + \beta_j^T x)}$$

$$Pr(C = K|X = x) = \frac{1}{1 + \sum_{j=1}^{K-1} \exp(\beta_0 + \beta_j^T x)}$$

(6)

The LR model can be interpreted in terms of model coefficients set. The magnitude and sign (positive or negative) of coefficient $\beta_i$ provide respectively the extent and direction (positive or negative) of correlation between predictor variable $x_i$ and log-odds of belonging to $i$th class.

Refer to chapter 4 of reference book 18 for more details.

### 2.3.4 Support vector machine

Support vector machine (SVM) is a supervised machine learning method primarily used for classification. The objective of this method is to find a hyperplane that creates the biggest margin between the two classes. The hyperplane coefficients ($\beta_0, \beta_i$) are determined numerically (Figure 2).

Refer to chapter 12 of reference book 18 for more details.

#### 2.3.5 Decision tree

Unlike other methods described so far, decision tree (DT) is a non-parametric method. A DT performs classification by finding split(s) in predictor variable(s) to maximize the accuracy of the classification model.

As the name suggests, DT model has a tree-like structure with nodes and branches. As shown in Figure 3, each node (except terminal nodes) represents a split on a predictor variable. The hierarchy of nodes represents the ranking of important predictor variables for classification.

From the model interpretation perspective, DT with fewer nodes are very useful in providing insights about the problem at hand. However, results from the very deep trees are not easy to analyze.

DT is also a greedy method and it tends to easily over fit the training data. A number of methods have been proposed to avoid the problem of overfitting. Random forest (RF) has turned out to be one of the best methods to avoid the overfitting problem. Refer to chapter 9 of reference book 18 for more details.

#### 2.3.6 Random forest

RF is a supervised machine learning method primarily used for classification. RF is an ensemble of DTs, which is another supervised method for classification.

![Figure 2](image-url)

**FIGURE 2** For perfect class separation, there should not be any point in the band of width $2M$. The red colored data points in the blue region and the blue colored data points in the red region are considered incorrectly classified. In case, a perfect separation is not possible, margin $M$ is maximized by constraining the total distance of incorrectly classified data points from their respective margins.
As mentioned earlier, RF is a collection of many DTs. For classification, RF method makes a class prediction by means of voting. For a data point \((x, y)\) with input \(x\) and output class \(y \in \{1, 2, \ldots, K\}\), let \(\hat{y}_{\text{DT}}(x)\) be the class prediction of the \(b\)th DT of RF model, then the class prediction of the RF model

\[
\hat{y}_{\text{RF}}(x) = \text{majority vote} \left\{ \hat{y}_{\text{DT}}(x) \right\}_1^N
\]

where \(N\) is the total number of DTs in RF model.

In terms of model accuracy, RF is a better method than DT. However, due to ensemble nature, RF loses model interpretability. In case of a DT (with a smaller number of nodes), it is easy to visualize how predictor variables are contributing to the class membership. The same visualization becomes difficult due to large number (order of 100) of DTs used by RF method. Refer to chapter 15 of reference book 18 for more details.

\section*{RESULTS AND DISCUSSION}

As mentioned in the introduction, the current article focuses on assessing the usefulness of machine learning methods for pathogen safety in process development of biological therapeutics. In this section, the results for a case study centering on the low pH viral inactivation unit operation are presented and discussed. First, the application and utility of unsupervised ML for process understanding of experimental results is demonstrated. Subsequently, the implementation of several supervised ML algorithms for viral clearance prediction is presented along with an assessment of each algorithms’ predictive ability and interpretability.

\subsection*{Pattern recognition and process understanding}

Process development scientists need to contextualize experimental results in order to maximize insights. This can be facilitated by comparing and contrasting new experimental results against historical data. Typically, in a univariate approach, multiple univariate overlay plots are generated for each parameter of interest. Although this approach can generate some insights, it comes with certain downsides. First, it is laborious and time-consuming to create and review multiple univariate plots. Second, it is limited as it does not explicitly account for the relationships between process parameters, which is of great interest to development scientists. An alternative to univariate approach is PCA. PCA is an unsupervised machine learning technique (see Section 2.3.1 for more details) that provides an effective and efficient way of contextualizing experimental data. Specifically, PCA enables scientists to quickly obtain an overview of their data by plotting score vectors in a reduced dimensional space.

As shown in Figure 4, a score plot can be used to identify groupings among experiments as well as atypical results. By color-coding the observations based on available meta data (e.g., product name),
additional patterns could emerge. Moreover, PCA facilitates the identification of: (a) process parameters that drive groupings of the experimental results in the score plot, (b) relationship between process parameters. For example, differences between experimental results for low pH viral inactivation can be easily identified in Figure 4 and linked to original process parameters via a contribution plot in Figure 5. Also, the loading plot in Figure 6 can reveal the relationships between the underlying process parameters and assist scientists to confirm known relationships or identify new ones. For instance, it is observed that inactivation time is positively correlated to pH.

### 3.2 Predictive modeling

In addition to pattern recognition and process understanding, there is a need for development scientists to be able to determine the
outcome of experimental studies for a range of process parameters. To this end, supervised machine learning can be leveraged to model the relationship between process parameters and viral clearance. Specifically, for low pH viral inactivation unit operation, which is the focus of this case study, a classification modeling approach was proposed to enable viral clearance prediction. To formulate such classification problem, two categories were defined based on the inactivation time:

1. Fast inactivation (complete inactivation within certain time limit)
2. Slow inactivation (incomplete inactivation within certain time limit). The inactivation time is determined as the first time point where the virus titer drops below the assay limit of detection. This study assesses the predictive ability and interpretability of multiple machine learning algorithms. Specifically, the following algorithms were evaluated:

- OPLS-DA
- LR
- SVM
- DT
- RF

The working principle and mathematical formulation of OPLS-DA, LR, SVM, DT, and RF machine learning algorithms used for classification are presented in Section 2.3.

### 3.2.1 Evaluation of predictive ability

The predictive ability of each classification model was evaluated using the overall model accuracy and individual class accuracy metrics calculated with cross-validation and the entire training dataset as shown in Table 3. The CV overall accuracy and class accuracy were calculated as averages of their respective values over the total number of CV groups.

Based on the results for CV overall accuracy shown in Table 3, SVM and RF have the lowest (0.74) and highest (0.94) predictive ability, respectively. OPLS-DA with a CV overall accuracy of 0.89 outperforms LR, SVM, and DT with respective CV accuracies of 0.86, 0.74, and 0.83. The same conclusion can be drawn with regard to the accuracy for class “Slow.” RF is the only machine learning algorithm, among the ones evaluated in this work, which performed better than OPLS-DA. Although OPLS-DA

![Confusion matrix for: (a) OPLS-DA, (b) LR, (c) SVM, (d) DT, and (e) RF machine learning algorithms](image)

![Predictive loadings for OPLS-DA classification model](image)

| TABLE 2 Model performance summary for OPLS-DA, LR, SVM, DT, and RF |
|-----------------|-----------------|-----------------|-----------------|-----------------|------------------|
|                 | CV overall accuracy | Overall accuracy | CV accuracy of class slow | Accuracy of class slow | Test accuracy of class slow | Overall test accuracy |
| OPLS-DA         | 0.89             | 0.89             | 0.92             | 0.93             | 1.00              | 1.00              |
| LR              | 0.86             | 0.86             | 0.86             | 0.87             | 0.00              | 0.11              |
| SVM             | 0.74             | 0.89             | 0.67             | 0.93             | 0.00              | 0.11              |
| DT              | 0.83             | 1.00             | 0.86             | 1.00             | 0.00              | 0.11              |
| RF              | 0.94             | 0.94             | 0.92             | 0.93             | 0.87              | 0.89              |

$R^2(\text{adj}) = 0.59$
has lower CV overall accuracy relative to RF, it still performed equally well for predicting class “Slow” with a CV class accuracy value of 0.92.

The overall accuracy of 1 clearly indicates the tendency of DT model to overfit. This is a condition where the model memorizes specific patterns in the training dataset without being able to generalize them well to new data, as reflected in the lower overall accuracy score for cross-validation and low test accuracy score. Overfitting was addressed by using an ensemble of DT’s in the RF ML algorithm that resulted in superior predictive performance as already shown.

The confusion matrices were generated for all five ML algorithms, based on the entire training dataset, as shown in Figure 7. The confusion matrices were used to evaluate the performance of a classification model through the number of correctly and incorrectly predicted observations per class. For instance, there was only 1 incorrectly classified observation for each class for RF algorithm. However, for DT, there were none misclassified results for “Slow” and “Fast” classes which was earlier discussed as indicative of model overfitting. Thus, the use of cross-validation in conjunction with the confusion matrix can detect cases of model over-fitting enabling an effective assessment of model performance.

In addition to CV-based assessment, an external dataset consisting of 9 data points (8 “Slow” class and 1 “Fast” class) was used to evaluate model performance, as shown below in Table 2. Despite the small size of the external dataset, the model’s predictive performance was assessed for the sake of completeness.

| Method               | Metric                          | Sign*: of correlation (±) | Variable importance |
|----------------------|--------------------------------|---------------------------|---------------------|
| OPLS-DA              | Sign (Loading)                 | VIP                       |                     |
| Logistic regression  | Sign (Model coefficient)       | [Model coefficient]       |                     |
| Support vector machine | Sign (Hyperplane coefficient) | | [Hyperplane coefficient] |
| Decision tree        | N/A                            | Count of split occurrences |                     |
| Random forest        | N/A                            | Count of split occurrences |                     |
Although LR, SVM, and DT demonstrated comparable performance on training and CV scores, their external testing scores suffered noticeably. This could be due to their sensitivities to class imbalance ratio and dataset size. Additional factors for poor performance of LR, SVM, and DT could be the relationships among parameters that were not observed in the training dataset. This hypothesis is based on preliminary analysis (results not shown here). Additional work is required to obtain a comprehensive understanding of the poor performance.

In contrast, both RF and OPLS-DA performed consistently better than the rest of algorithms on both training and testing datasets. However, due to small size of testing dataset, additional data should be collected to verify this finding.

### 3.2.2 Evaluation of model interpretability

Each machine learning method provides different ways to interpret the modeling outcomes. In case of linear methods, a model can be interpreted in terms of direction and magnitude of the correlation between inputs and output. This is best demonstrated with the loading plot of the OPLS-DA model, shown in Figure 8. For example, a positive loading for \( \text{pH} \) means that the higher the value of \( \text{pH} \), the slower the inactivation.

Nonlinear models may not be interpreted in the same way as linear ones. Tree-based classification models maximize their accuracies by finding split(s) in predictor variable(s) where the Gini index is minimized (see Section 2.3.5). As shown in Figure 9, one of the DTs from the RF classification model identifies that Initial Virus Titer and \( \text{pH} \) are two most important variables in achieving high model accuracy.

The comparison across both linear and nonlinear models can be facilitated by variable importance, which can serve as a common concept for estimating the contribution of an input variable to output. A single common metric cannot be used for variable importance estimation across machine learning models due to inherent differences in the machine learning algorithms. Table 3 presents a list of metrics used for different machine learning methods to interpret their respective modeling outcomes.

Table 4 presents the summary of input variables’ correlation with the model output. For OPLS-DA and LR, the sign of the correlations matches SME (subject matter expert) feedback on all input parameters except Initial Protein Concentration. The SVM-based model failed to provide correct sign of the correlation for \( \text{pH} \).

It should be noted that the relationship between Initial protein concentration with LRF is different from reported in. This warrants further investigation, given that the range of inactivation times used in the model training was wider (10–240 min) relative to 30 min inactivation time in Reference 10.

Table 5 presents the summary of input variables’ importance for different machine learning methods. For OPLS-DA, VIP score was used for variable importance estimation. A variable with VIP score close to or greater than 1 is generally considered important in a given model. While a variable with VIP score significantly less than 1 is generally considered less important. Within the OPLS-DA model, all input parameters have VIP scores greater than 1 except for Spike Ratio, which has a VIP score of 0.28.

For tree-based models, variable importance for a specific model input is estimated by counting the occurrences of splits corresponding to it. Variable importance estimation based on split count, may underestimate the predictive contribution of a model input in the presence of multi-collinearity among input variables. For both, Random Forest and DT models, Initial Virus Titer ranked the highest and Spike Ratio ranked the lowest.

Although the variable importance magnitude should not be compared across models, the overall relative variable importance ranking within each

### Table 4: Sign of the input variable’s correlation with the model output

| Input variable          | OPLS-DA | Logistic regression | Support vector machine | Literature findings\(^9,10\) |
|-------------------------|---------|---------------------|------------------------|-----------------------------|
| pH                      | +       | +                   | –                      | +                           |
| Temperature             | –       | –                   | –                      | –                           |
| Spike Ratio             | +       | +                   | +                      | +                           |
| Initial Protein Conc.   | +       | +                   | –                      | No effect                   |
| Initial Virus Titer     | –       | –                   | –                      | –                           |

### Table 5: Variable importance for different machine learning algorithms

|                | pH   | Temperature | Spike ratio | Initial protein conc. | Initial virus titer |
|----------------|------|-------------|-------------|-----------------------|--------------------|
| OPLS-DA        | 1.07 | 1.13        | 0.28        | 1.16                  | 1.07               |
| Logistic reg.  | 0.13 | 0.12        | 0.05        | 0.29                  | 0.63               |
| Support vector | 0.17 | 0.44        | 0.06        | 0.09                  | 0.69               |
| Decision tree  | 0.09 | 0.17        | 0.00        | 0.00                  | 0.74               |
| Random forest  | 0.17 | 0.02        | 0.00        | 0.38                  | 0.43               |
model can still be assessed. Overall, Spike Ratio was ranked the lowest and Initial Virus Titer was ranked the highest in terms of variable importance across algorithms. It is worth noting that the OPLS-DA method, which takes into account the multi-collinearity in the input data, produced similar variable importance values across input variables except for Spike Ratio. However, other methods not accounting for multi-collinearity, produced a range of variable importance values. Further research is required to investigate whether other methods truly underestimate the variable importance in the presence of multi-collinearity.

4 | INDUSTRIALIZATION OF MODELS

An IT framework for industrialization of the viral safety models is a prerequisite for fully realizing the benefits of the machine learning models presented in this work. Specifically, a cloud-based environment was established using Amazon Web Services (AWS) to allow for a more scalable infrastructure as more data is accumulated or more computationally intensive operations become necessary. This AWS infrastructure hosts various pipelines for data storage, preprocessing, and machine learning model execution, with a web-based graphical user interface for displaying model outputs and diagnostics. Details of the design and implementation of the cloud infrastructure that enables model industrialization may be discussed in a subsequent article.

5 | CONCLUSION AND FUTURE WORK

The development and optimization of viral clearance unit operations is critical to ensure the manufacture of safe and efficacious therapeutic biologics such as human monoclonal antibodies. The development of viral clearance unit operations requires substantial amount of effort in wet lab experiments for process characterization. The current work focused on demonstrating the benefits of leveraging machine learning methods to help streamline the drug process development.

Specifically, a case study for low pH viral inactivation was presented. It was shown that the use of unsupervised machine learning techniques, such as PCA can increase the effectiveness and efficiency of contextualizing experimental results. By using PCA-based models and metrics, process development scientists can easily obtain an overview of multidimensional data for new experiments in the context of historical data. Thus, the ease of identifying patterns and relationships between process parameters can greatly streamline process understanding from the data. Additionally, supervised machine learning techniques were evaluated for the viral clearance prediction. The pathogen safety scientists’ question regarding adequate viral clearance, given certain experimental conditions, was formulated into a binary classification problem.

Various classification algorithms were evaluated with respect to their predictive ability and interpretability. These machine learning algorithms provided insights about the effect of process parameters on viral clearance. While RF algorithm and OPLS-DA provided promising results for viral clearance prediction, other machine learning algorithms were limited in terms of model performance.

Overall, it has been shown that machine learning algorithms can be helpful in streamlining process development for therapeutic biologics. The cloud-based infrastructure established for viral clearance studies will enable the full realization of machine learning models’ utility, while facilitating model improvement with additional data in the future.

ACKNOWLEDGMENTS

The authors would like to thank Martin Poggel and Klaus Kaiser, Thomas Daszkowski, Felix Oehme, Teit Agger, Andres Friese, and Scott Probst, for their guidance, support, and funding for this innovative project.

CONFLICT OF INTEREST

A provisional patent application has already been filed to protect the intellectual property.

DATA AVAILABILITY STATEMENT

The complete dataset cannot be shared due to confidentiality.

ORCID

Shyam Panjwani https://orcid.org/0000-0001-7611-8425

REFERENCES

1. Schwantes A, Specht R, Chen Q. Proceedings of the 2017 viral clearance symposium, session 4: submission strategies. PDA J Pharm Sci Technol. 2018;72(5):498-510.
2. Galbraith D. ICH Q5A. ICH Quality Guidelines 2017. 311-335. https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118971147.ch10
3. ASTM E2888-12, Standard Practice for Process for Inactivation of Rodent Retrovirus by pH. West Conshohocken, PA2012.
4. PAT – a framework for innovative pharmaceutical development, manufacturing, and quality assurance In: FDA, ed2004.
5. Q8(R2) pharmaceutical development. In: FDA, ed2009.
6. Proposed regulatory framework for modifications to artificial intelligence/machine learning (AI/ML)-based software as a medical device (SaMD)—Discussion Paper and Request for Feedback. In: FDA, ed2019.
7. Banton D, Vacante D, Bulthuis B, Goldstein J, Wineburg M, Schreffler J. The use of Bayesian hierarchical logistic regression in the development of a modular viral inactivation claim. PDA J Pharm Sci Technol. 2019;73(6):552-561.
8. Bronson K, Krejci S, Lee K, Hamilton E, Stein K, Xu Y. Bracketed generic inactivation of rodent retroviruses by low pH treatment for monoclonal antibodies and recombinant proteins. Biotechnol Bioeng. 2003;82(3):321-329.
9. Chinniah S, Hinckley P, Connell-Crowley L. Characterization of operating parameters for XMuLV inactivation by low pH treatment. Biotechnol Prog. 2016;32(1):89-97.
10. Mattila J, Clark M, Liu S, et al. Retrospective evaluation of low-pH viral inactivation and viral filtration data from a multiple company collaboration. PDA J Pharm Sci Technol. 2016;70(3):293-299.
11. Berthold MR, Cebron N, Dill F, et al. KNIME - the Konstanz information miner: version 2.0 and beyond. SIGKDD Explor Newsl. 2009;11(1):26-31.
12. Basilevsky AT. Statistical Factor Analysis and Related Methods: Theory and Applications; New York, NY, John Wiley & Sons 2009:97-181.
13. Bylesjö M, Rantalainen M, Cloarec O, Nicholson JK, Holmes E, Trygg J. OPLS discriminant analysis: combining the strengths of PLS-DA and SIMCA classification. J Chemometr. 2006;20(8–10):341-351.
14. Panjwani S, McDaniel J, Nikolaou M. Improvement of zonal isolation in horizontal shale gas wells: a data-driven model-based approach. J Nat Gas Sci Eng. 2017;47:101-113.

15. Pérez NF, Ferré J, Boqué R. Calculation of the reliability of classification in discriminant partial least-squares binary classification. Chemom Intel Lab Syst. 2009;95(2):122-128.

16. Wold S, Ruhe A, Wold H, Dunn WJ. The collinearity problem in linear regression. The partial least squares (PLS) approach to generalized inverses. SIAM J Sci Stat Comp. 1984;5(3):735-743.

17. Wold S, Sjöström M, Eriksson L. PLS-regression: a basic tool of chemometrics. Chem Int Lab Syst. 2001;58(2):109-130.

18. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning, Data Mining, Inference, and Prediction. 2nd ed. Springer: Springer-Verlag New York; 2008.

19. Bramer M. Avoiding overfitting of decision trees. Principles of Data Mining. London: Springer London; 2013:121-136.

20. Ting K.M. (2017) Confusion Matrix. In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning and Data Mining. Springer, Boston, MA. https://doi.org/10.1007/978-1-4899-7687-1_50

21. Farrés M, Platikanov S, Tsakovski S, Tauler R. Comparison of the variable importance in projection (VIP) and of the selectivity ratio (SR) methods for variable selection and interpretation. J Chemometr. 2015;29(10):528-536.

How to cite this article: Panjwani S, Cui I, Spetsieris K, et al. Application of machine learning methods to pathogen safety evaluation in biological manufacturing processes. Biotechnol Progress. 2021;37:e3135. https://doi.org/10.1002/btpr.3135