PharmaSUG China 2023 - Paper AA-118

Application of machine learning in pharmeceutical manufacturing

Kaiping Yang, BeiGene

ABSTRACT

This article introduces the application of machine learning in pharmaceutical manufacturing, especially using near-infrared spectroscopy to monitor the quality attributes of products in real time. It also explains how to deploy machine learning web apps using Streamlit and RStudio Connect, two popular tools for data science and visualization. The article consists of four sections:

The first section gives a brief overview of machine learning, its problem setting, and its main categories including supervised learning and unsupervised learning. It also introduces some common frameworks for developing machine learning algorithms, such as Scikit-learn, TensorFlow and PyTorch.

The second section focuses on the spectroscopic multivariate calibration models, which are the basic models of process analytical technology (PAT) for near-infrared spectroscopy. It compares traditional testing of analyte property to near-infrared spectroscopy which can be non-destructive, non-invasive, and fast. It also mentions some techniques to deal with non-linear light scattering effect, such as kernel methods and neural networks.

The third section presents Streamlit, a Python library for creatingand sharing web apps for machine learning and data science. It demonstrates a Streamlit app that can do real-time quality monitoring with near-infrared spectroscopy. It also covers some common issues and how to fix them when deploying Streamlit apps with Streamlit Cloud or RStudio Connect.

The fourth section shows how to use RStudio Connect to deploy Streamlit apps which can share data and machine learning insights. It gives process and solutions for deploying Streamlit apps with rsconnect-python. It also explains why machine learning apps can be more efficient and cost-effective.

In a word, this article demonstrates how machine learning can be applied to improve the pharmaceutical manufacturing process and achieve real-time quality monitoring. It also highlights the challenges and solutions for deploying machine learning web apps in a secure and compliant way.

INTRODUCTION

The application of machine learning in pharmaceutical manufacturing has revolutionized the industry, enabling real-time quality monitoring and enhancing product attributes. One powerful tool in this field is near-infrared spectroscopy, which offers non-destructive, non-invasive, and fast analysis capabilities. By harnessing the potential of machine learning algorithms, pharmaceutical companies can optimize their manufacturing processes and achieve unprecedented levels of efficiency and quality control.

In this article, we will explore the various aspects of machine learning in pharmaceutical manufacturing, with a specific focus on the utilization of near-infrared spectroscopy for real-time quality monitoring. We will delve into the fundamental concepts of machine learning, including supervised and unsupervised learning, and discuss the common frameworks used for developing machine learning algorithms, such as TensorFlow and PyTorch.

Furthermore, we will introduce two popular tools, Streamlit and RStudio Connect, which play a crucial role in creating and sharing web applications for machine learning and data science in the pharmaceutical industry. These tools provide a seamless platform for deploying machine learning models and visualizing data, enabling pharmaceutical professionals to gain valuable insights and make informed decisions.

Throughout this article, we will address the challenges faced in deploying machine learning web apps in a secure and compliant manner. We will also highlight the solutions and best practices for overcoming these challenges, ensuring that the application of machine learning in pharmaceutical manufacturing adheres to industry standards and regulations.

By the end of this article, you will have a comprehensive understanding of how machine learning can be effectively applied to improve the pharmaceutical manufacturing process and achieve real-time quality monitoring. Whether you are a researcher, data scientist, or pharmaceutical professional, this article will provide valuable insights into the potential of machine learning in transforming the industry.

Let us embark on this exciting journey into the world of machine learning in pharmaceutical manufacturing and discover the immense possibilities it holds for enhancing efficiency, quality, and innovation.

WHAT IS MACHINE LEARNING

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy. Machine learning is an important component of the growing field of data science. Through the use of statistical methods, algorithms are trained to make classifications or predictions, and to uncover key insights in data mining projects. Machine learning algorithms are typically created using frameworks that accelerate solution development, such as Scikit-learn, TensorFlow and PyTorch.

Machine learning is a pervasive and powerful form of artificial intelligence that is changing every industry. It has applications in various domains, such as image recognition, speech recognition, traffic prediction, product recommendation, self-driving cars, and many more.

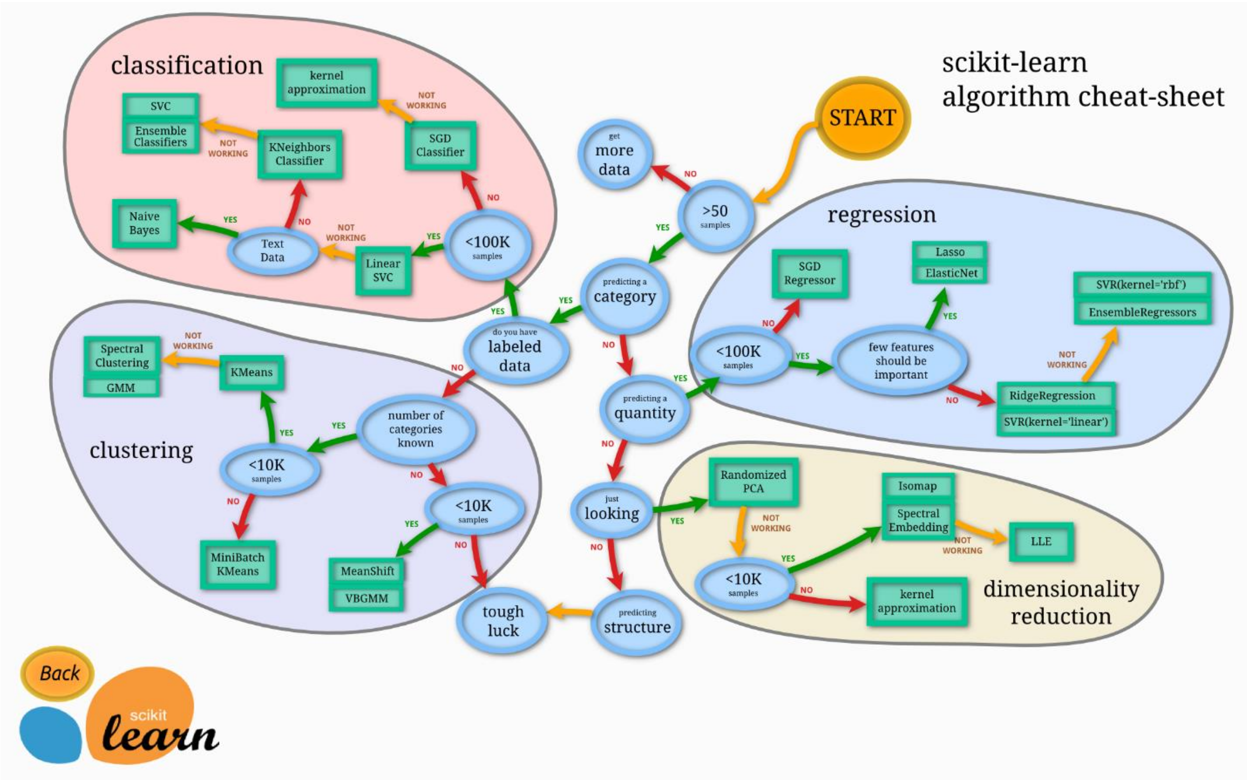
MACHINE LEARNING: THE PROBLEM SETTING

In general, a learning problem considers a set of n samples of data and then tries to predict properties of unknown data. Learning problems fall into a few categories:

- **Supervised learning**, in which the data comes with additional attributes that we want to predict. This problem can be either:
  - **Classification**: samples belong to two or more classes and we want to learn from already labeled data how to predict the class of unlabeled data.
  - **Regression**: if the desired output consists of one or more continuous variables, then the task is called regression.
- **Unsupervised learning**, in which the training data consists of a set of input vectors x without any corresponding target values. The goal in such problems may be to discover groups of similar examples within the data, where it is called clustering, or to determine the distribution of data within the input space, known as density estimation, or to project the data from a high-dimensional space down to two or three dimensions for the purpose of visualization.

MACHINE LEARNING FLOW CHART

The following flow chart shows how to choose a machine learning algorithm for a particular problem based on various criteria such as the size and type of data, the desired output, and the complexity of the model.



TRAINING SET AND TESTING SET

A common practice in machine learning is to evaluate an algorithm by splitting a data set into two:

- **Training set**, on which we learn some properties;
- **Testing set**, on which we test the learned properties.

The training set is used to fit the machine learning model, while the testing set is used to assess how well the model generalizes to unseen data. The goal is to avoid overfitting, which occurs when a model performs well on the training set but poorly on the testing set. This means that the model has learned specific patterns from the training data that do not apply to new data.

One way to avoid overfitting is to use cross-validation, which is a technique that splits the data into multiple subsets and uses each subset as both training and testing sets. This way, we can estimate how well our model will perform on average on new data.

LEARNING AND PREDICTING

Once we have chosen a machine learning algorithm and trained it on some data, we can use it to make predictions on new data. For example, if we have a classification problem, we can use our model to predict the class label of a new input vector x by passing it as an argument to a predict function:

```
y_pred = model.predict(x)
```

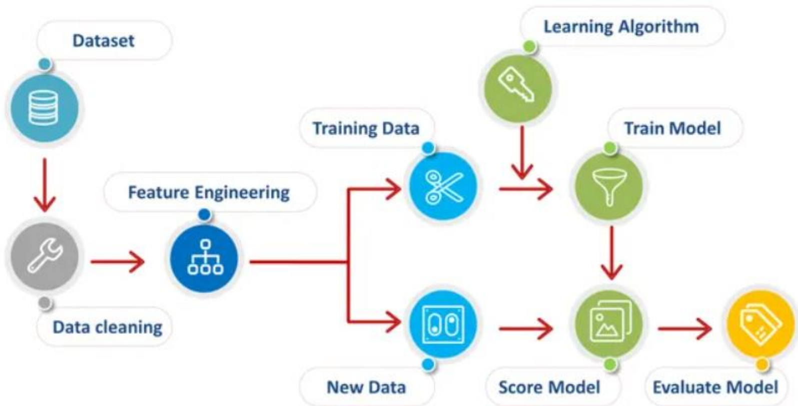
The output y_pred will be an array containing the predicted class labels for each sample in x.

Similarly, if we have a regression problem, we can use our model to predict the output value for a new input vector x by passing it as an argument to a predict function:

```
y_pred = model.predict(x)
```

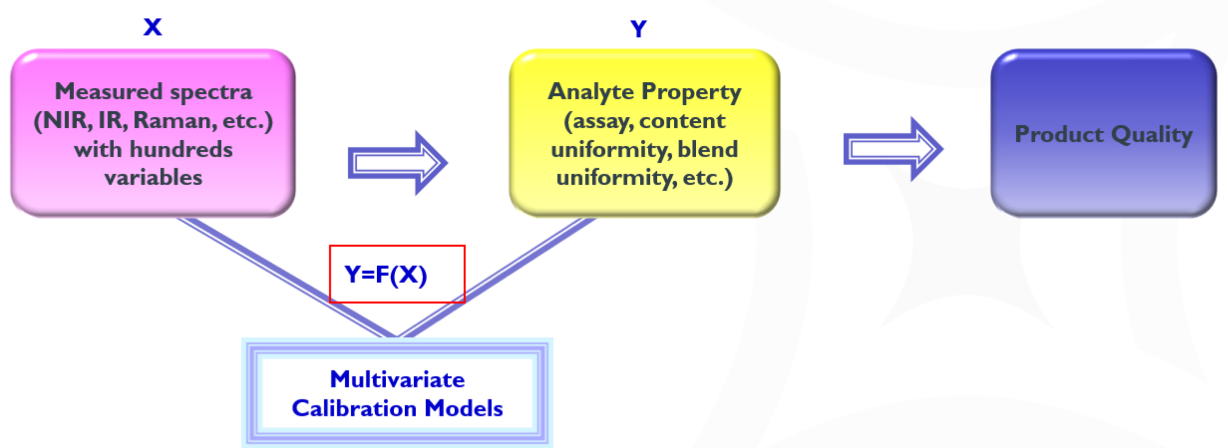
The output y_pred will be an array containing the predicted output values for each sample in x.

We can also measure how well our model performs on new data by comparing the predicted values with the true values using various metrics such as accuracy, precision, recall, f1-score, mean squared error, etc.

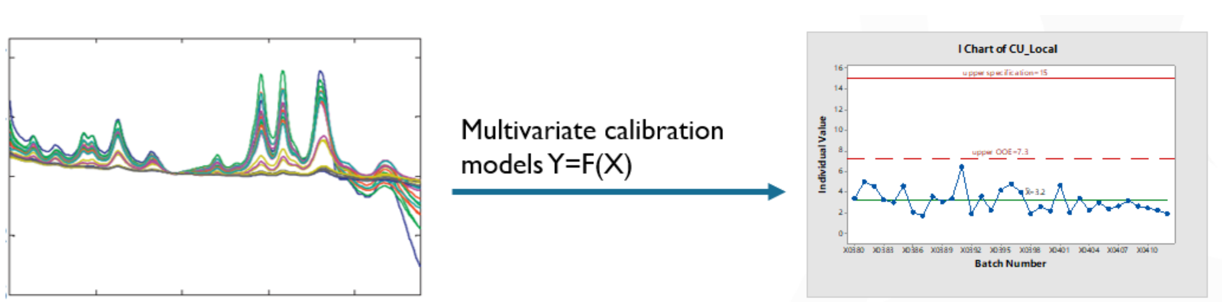


SPECTROSCOPIC MULTIVARIATE CALIBRATION MODELS

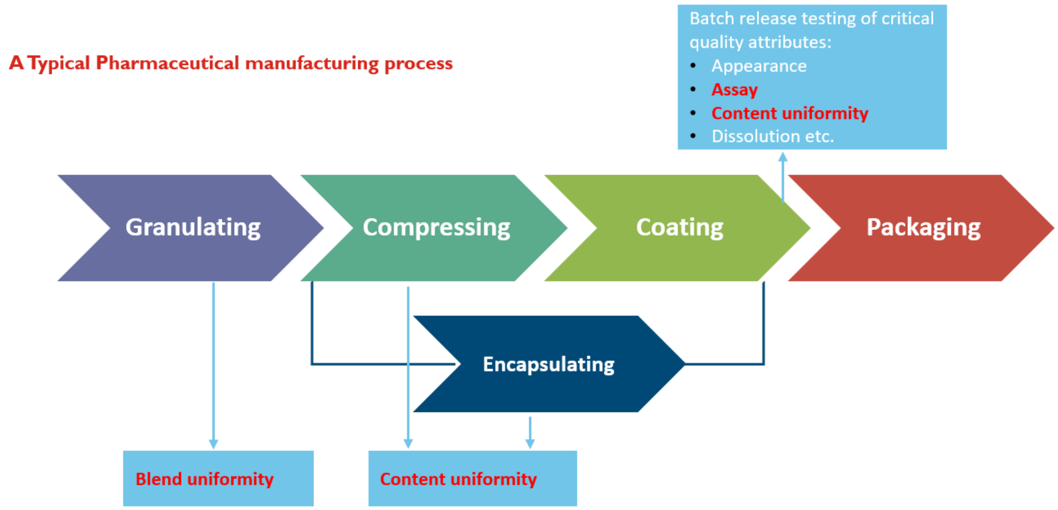
Spectroscopic Multivariate Calibration Model is a technique that uses spectroscopic data, such as near-infrared (NIR) spectra, to predict the concentration or quality of analytes in samples, such as soils, tablets, or products. It is based on the assumption that there is a linear or nonlinear relationship between the spectral features and the analyte property of interest.



Spectroscopic Multivariate Calibration Model is the basic model of process analytical technology (PAT) which can be used for on-line monitoring in pharmaceutical manufacturing process.



Traditionally, the testing of Analyte Property (assay, content uniformity, blend uniformity, etc.) is conducted in laboratory after the manufacturing process. And the testing procedure is complicated and time consuming.



Compared with traditional off-line testing system (such as HPLC), spectroscopic multivariate calibration model has the following advantages which can make on-line monitoring come true:

- Non-destructive and non-invasive.
- No sample preparation.
- Fast measurement.

It is usually expected to detect the abnormal quality issue during manufacturing process but not in batch release testing after manufacturing production.

OFF-LINE AND ON-LINE TESTING

Off-line testing refers to the testing of quality attributes after manufacturing process. It usually involves sampling, sample preparation, analysis, and data processing. Off-line testing can provide accurate and reliable results, but it is also costly, time-consuming, and prone to human errors.

On-line testing refers to the testing of quality attributes during manufacturing process in real time. It usually involves installing sensors or probes in the process equipment, collecting spectral data, and applying multivariate calibration models to predict the analyte property. On-line testing can provide timely and continuous feedback, but it also faces challenges such as matrix effect, noise, drift, and interference.

MATRIX EFFECT

Matrix effect is a phenomenon that causes variations in spectral data due to differences in physical or chemical properties of samples, such as particle size, moisture content, temperature, pressure, etc. Matrix effect can affect the accuracy and robustness of multivariate calibration models, especially for complex samples such as soils or tablets.

To deal with matrix effect, various techniques have been proposed, such as:

- Data preprocessing: applying mathematical transformations or corrections to spectral data to reduce noise, baseline drift, scatter effects, etc.
- Data splitting: dividing the data into calibration set and validation set according to some criteria such as sample type, concentration range, etc.
- Data augmentation: adding artificial samples or spectra to the data set to increase its diversity and representativeness.

- Data selection: choosing a subset of spectral variables or regions that are most relevant or informative for the analyte property prediction.
- Data fusion: combining spectral data from different sources or modalities to enhance the information content and reduce redundancy.

TECHNIQUES TO DEAL WITH NON-LINEAR LIGHT SCATTERING EFFECT

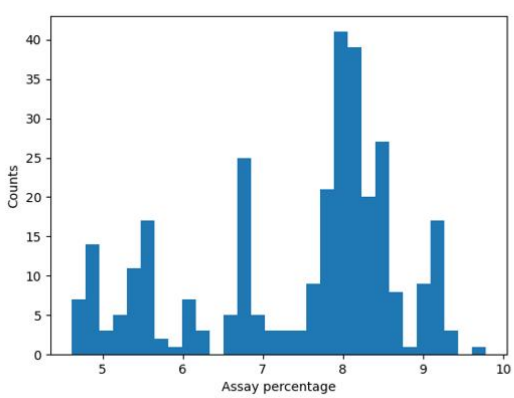
Non-linear light scattering effect is another phenomenon that causes distortions in spectral data due to multiple scattering events within heterogeneous samples. Non-linear light scattering effect can result in non-linear relationships between spectral features and analyte property, which cannot be captured by linear multivariate calibration models.

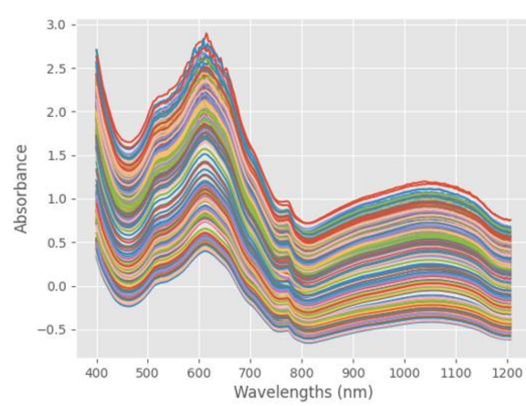
To deal with non-linear light scattering effect, various techniques have been proposed, such as:

- Pre-processing:
    - First and second derivatives (D1 and D2): applying numerical differentiation to spectral data to remove baseline shifts and enhance spectral resolution
    - Standard normal variate (SNV): applying mean centering and variance scaling to spectral data to remove multiplicative effects
    - Extended multiplicative signal correction (EMSC): applying a linear regression model to spectral data to correct for additive and multiplicative effects
    - Extended inverted signal correction (EISC): applying a non-linear regression model to spectral data to correct for additive and multiplicative effects
    - Optical path length estimation and correction (OPLEC): applying a physical model to estimate and correct for optical path length variations in spectral data
- Non-linear calibration:
    - Artificial neural network (ANN): applying a non-linear model that uses multiple layers of neurons to learn complex non-linear relationships
    - Gaussian process regression (GPR): applying a non-linear model that uses a probabilistic framework based on kernel functions
    - Least-square support vector machine (LS-SVM): applying a non-linear model that uses kernel functions to map spectral data into a high-dimensional feature space

DATASETS: TABLET

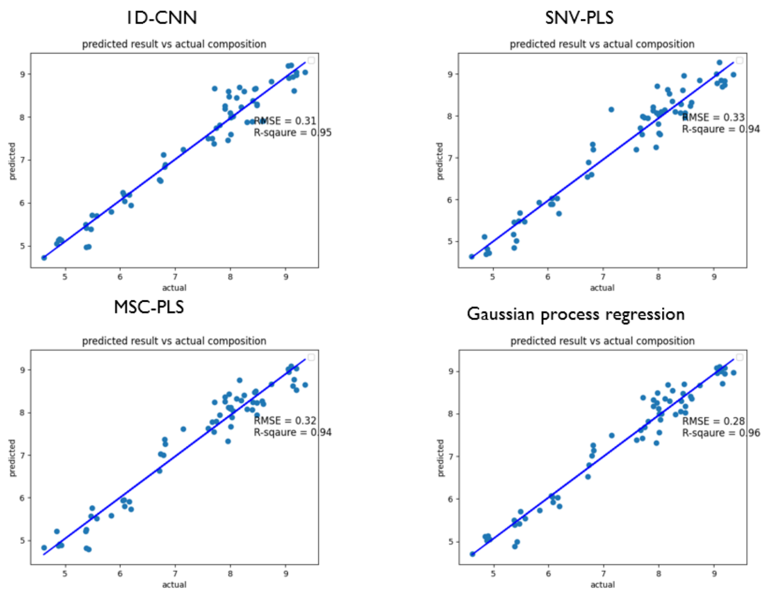
A public Near-Infrared spectroscopy dataset containing 310 samples with wavelength from 400 nm to 1200 nm. The samples are tablets with different concentrations of active pharmaceutical ingredient (API). The goal is to build a multivariate calibration model to predict the API concentration from NIR spectra.





RESULT COMPARISON

The following figure shows the result comparison of different multivariate calibration models applied to the tablet and product datasets.



The models are:

- ID-CNN: a nonlinear model that uses convolutional neural network (CNN) to extract spectral features and identify the sample type
- SNV-PLS: a linear model that uses standard normal variate (SNV) to preprocess spectral data and partial least squares regression (PLSR) to predict the analyte property
- MSC-PLS: a linear model that uses multiplicative scatter correction (MSC) to preprocess spectral data and partial least squares regression (PLSR) to predict the analyte property
- MSC-SVR: a nonlinear model that uses multiplicative scatter correction (MSC) to preprocess spectral data and support vector regression (SVR) to predict the analyte property

The performance metrics are:

- Root mean squared error of prediction (RMSEP): a measure of how well the model predicts new data
- Coefficient of determination (R2): a measure of how much variation in analyte property is explained by the model

From the figure, we can see that:

- All models have relatively low RMSEP values, indicating that they can predict the analyte property with high accuracy
- All models have relatively high R2 values, indicating that they can explain most of the variation in analyte property
- MSC-SVR has the lowest RMSEP value and the highest R2 value, indicating that it is the best model among the four
- ID-CNN has the second lowest RMSEP value and the second highest R2 value, indicating that it is a good model but slightly worse than MSC-SVR

- SNV-PLS and MSC-PLS have similar RMSEP values and R2 values, indicating that they are comparable models but worse than ID-CNN and MSC-SVR

STREAMLIT: A PYTHON LIBRARY FOR ML WEB APPS

WHAT IS STREAMLIT

Streamlit is an open-source Python library that makes it easy to create and share web apps for machine learning and data science. Streamlit allows you to display descriptive text and model outputs, visualize data and model performance, and modify model inputs through the UI using sidebars.

Streamlit is designed to be simple, fast, and interactive. You can write your app in pure Python, without any HTML, CSS, or JavaScript. You can also use Streamlit components to extend your app with custom widgets or integrations with other frameworks.

Streamlit has a powerful caching mechanism that automatically optimizes your app performance. You can also deploy your app to Streamlit Cloud, a free hosting service that lets you share your app with anyone.

WHY WE NEED STREAMLIT?

Streamlit is a great tool for data scientists and machine learning engineers who want to showcase their work and share their insights with others. Streamlit can help you:

- Build interactive web apps for data exploration, analysis, visualization, and communication
- Prototype and iterate on your machine learning models quickly and easily
- Collaborate with your team and get feedback from your stakeholders
- Educate and inspire your audience with engaging and informative content

For example, suppose you have a task to perform real-time quality monitoring using near-infrared spectroscopy. You have developed a machine learning framework using Python to process the spectral data and predict the quality attributes. You want to show your results in a web app that allows users to interact with the data and the model.

With Streamlit, you can build such an app in a few lines of code, without worrying about the web development details. You can also deploy your app to Streamlit Cloud and share it with anyone who has access to the internet.

STREAMLIT CLOUD

Streamlit Cloud is a service that allows you to deploy, manage, and share your Streamlit apps from Streamlit — all for free. Streamlit Cloud has three tiers: Community, Teams, and Enterprise.

- Community: A free tier for anyone who wants to share their apps publicly. You can deploy up to three apps per account, with unlimited viewers and collaborators.
- Teams: A paid tier for teams who want to share their apps privately within their organization or with specific people. You can deploy unlimited apps per account, with access control and authentication features.
- Enterprise: A custom tier for large organizations who need advanced features such as single sign-on (SSO), dedicated resources, custom domains, etc.

To use Streamlit Cloud, you need to have a GitHub account or SSO login. You also need to upload your app code to a GitHub repository. Then you can connect your repository to Streamlit Cloud and deploy your app in minutes.

STREAMLIT APP DEMO

To demonstrate how Streamlit works, here is an example of a Streamlit app that performs sentiment analysis on movie reviews using natural language processing (NLP). The app allows users to enter a movie review text and see the predicted sentiment (positive or negative) and the confidence score. The app also displays some statistics on the dataset and the model performance.

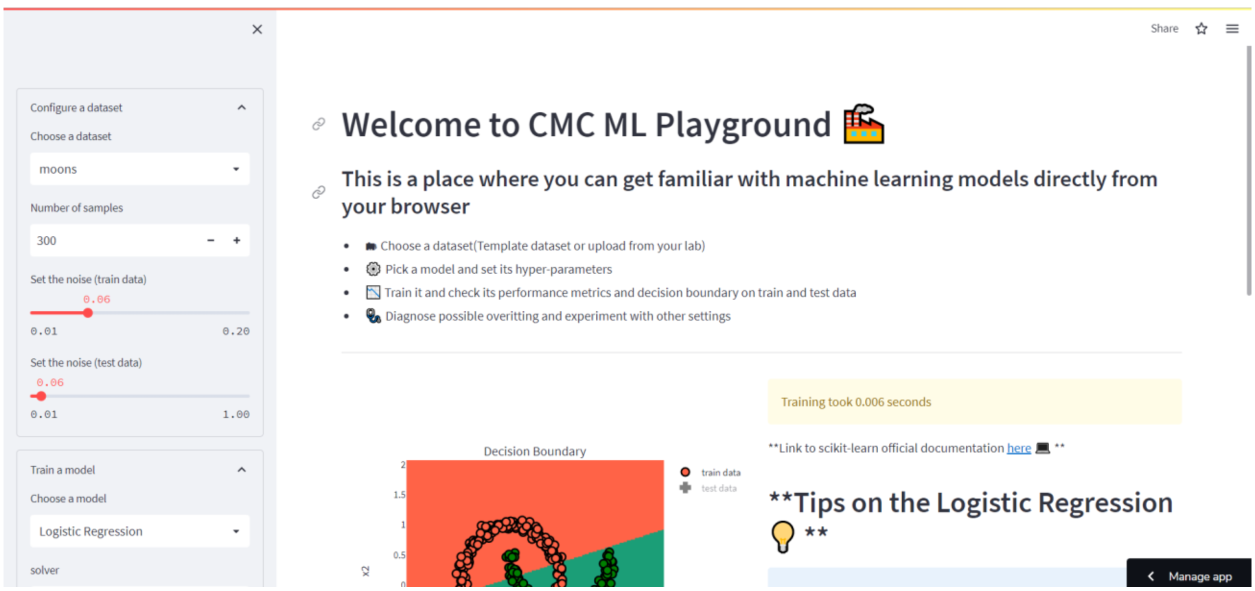
The app is deployed on Streamlit Cloud using the Community tier. You can access the app from this link:

https://kaipingyang-mlplayground-app-dm43sk.streamlitapp.com/

You can also view the source code of the app from this GitHub repository:

https://github.com/kaipingyang/MLPlayground

Here is a screenshot of the app:

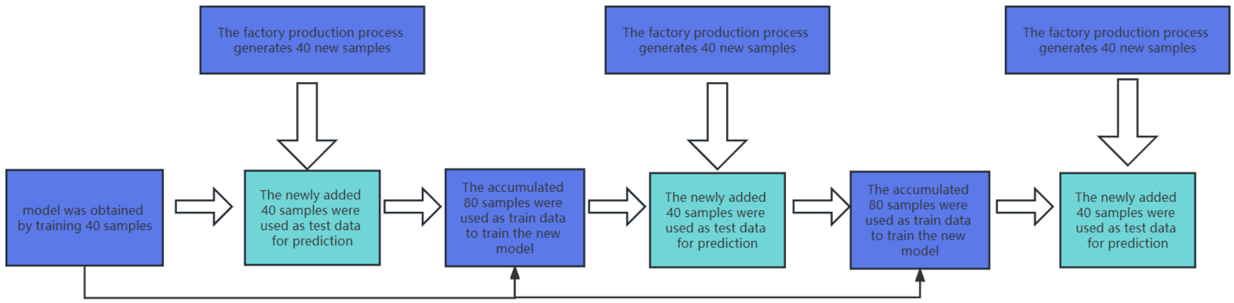


FUTURE MODEL MODIFICATION FOR APP

If you want to modify your machine learning model for your Streamlit app, you can do so by editing your Python code and pushing it to your GitHub repository. Streamlit Cloud will automatically detect the changes and update your app accordingly.

You can also use Streamlit widgets to allow users to adjust the model parameters or select different models from the app UI.

We can use the NIR spectral data obtained for the first time as a training set to fit the IR spectral data obtained for the second time by the machine learning model, and then add the data of the second time to the training set to fit the data predicted by the machine learning model for the third time. By analogy, we can improve the accuracy of predictions with each training and prediction. This process can be done in the background in the form of a configuration file without requiring the user to do too much on the UI. This will be a direction for future model revision, and the specific process is shown in the figure below.



CHALLENGE

Deploying the Streamlit app requires a GitHub account or SSO login, but our company only has a GitLab account. This means that we cannot use Streamlit Cloud to host our app, unless we create a GitHub account or SSO login.

One possible solution is to use Streamlit for Teams, which is a paid tier that allows us to use our own GitLab account to deploy our app. However, this may incur additional costs and require approval from our management.

Another possible solution is to use Streamlit self-hosted, which is a way to run our app on our own server or cloud platform. This may require more technical skills and resources, but it also gives us more control and flexibility over our app deployment.

RSTUDIO CONNECT: EASILY SHARE YOUR INSIGHTS

You worked hard to find key insights. Now you need to get those insights off your desktop and in the hands of decision-makers. RStudio Connect makes this so very easy. Move beyond spreadsheets and screenshots, and augment your BI tools for sharing insights.

RStudio Connect is a standalone publishing platform for the work your teams create in R and Python. You can share Shiny applications, R Markdown reports, dashboards, plots, models, APIs, Jupyter Notebooks, interactive Python content, and more.

RStudio Connect allows you to:

- Deploy your work with a single click or command
- Manage access and permissions for your content
- Schedule updates and send email notifications
- Monitor usage and performance metrics
- Integrate with your existing infrastructure and tools

RStudio Connect supports various deployment methods, such as rsconnect-python, git, or manual upload. You can also use RStudio Connect Server API to automate and customize your deployments.

DEPLOYING STREAMLIT USING RSTUDIO CONNECT

Streamlit is a Python-based library that makes it easy to create and share web apps for machine learning and data science. Streamlit allows you to display descriptive text and model outputs, visualize data and model performance, and modify model inputs through the UI using sidebars.

Streamlit is compatible with RStudio Connect, which means you can deploy your Streamlit apps to RStudio Connect and share them with your audience. You can also leverage the features of RStudio Connect, such as authentication, access control, scheduling, email alerts, etc.

The following describes how to deploy Streamlit in RStudio Connect:

- Install Streamlit and rsconnect-python on your local machine
- Create a Streamlit app using Python code
- Deploy your app using rsconnect-python command-line tool
- Access your app from RStudio Connect web interface

DEPLOYING

Streamlit apps can be deployed with the rsconnect-python package. For Streamlit apps, the entrypoint is the name of the Python file containing your app. For example, if your app's source file is named main.py, use:

```
rsconnect deploy streamlit -n <saved server name> --entrypoint main.py
 ...
```

You can also specify other options, such as:

- --server: The URL of the server where you want to deploy your app
- --api-key: The API key for authentication with the server
- --name: The name of your app on the server
- --title: The title of your app on the server
- --extra-files: A list of additional files or directories that are required by your app

For more details on rsconnect-python options, see:

https://github.com/rstudio/rsconnect-python

CONTACT IT OF RSTUDIO

If you encounter any problems or have any questions when deploying your Streamlit app using RStudio Connect, you can contact IT of Rstudio for support. You can reach them by:

Step 1: Upload your code to GitLab and share with IT

To deploy your Streamlit app using RStudio Connect, you need to upload your code to a GitLab repository and share it with IT of Rstudio. You can do this by:

- Creating a new project on GitLab or using an existing one
- Adding your Streamlit app source file and any other required files or directories to the project
- Committing and pushing your changes to the remote repository
- Inviting IT of Rstudio as a collaborator to your project with the appropriate permissions

For more details on how to use GitLab, see:

https://docs.gitlab.com/ee/user/index.html

Step 2: IT will deploy the Streamlit App through Rstudio Connect

Once you have shared your code with IT of Rstudio, they will deploy your Streamlit app through RStudio Connect. They will do this by:

- Installing Streamlit and rsconnect-python on the server where RStudio Connect is running
- Cloning your GitLab repository to the server
- Deploying your app using rsconnect-python command-line tool
- Configuring your app settings and permissions on RStudio Connect web interface

You will receive an email notification when your app is deployed. You can then access your app from the URL provided in the email.

CONCLUSION

In conclusion, the application of machine learning in pharmaceutical manufacturing has opened up new avenues for optimizing manufacturing processes and achieving real-time quality monitoring. Near-infrared spectroscopy, in particular, has emerged as a powerful tool for non-destructive and non-invasive analysis of product attributes.

Through the using of machine learning algorithms, pharmaceutical professionals can gain valuable insights into their manufacturing processes and make informed decisions to enhance efficiency and quality control. The deployment of machine learning web apps using Streamlit and RStudio Connect has further facilitated the visualization and sharing of data, enabling collaboration and innovation in the industry.

However, the deployment of machine learning in pharmaceutical manufacturing also poses challenges in terms of security and compliance. It is crucial to adhere to industry standards and regulations to ensure the safety and efficacy of pharmaceutical products.

In this article, we have explored the fundamental concepts of machine learning, the common frameworks used for developing machine learning algorithms, and the role of Streamlit and RStudio Connect in creating and sharing web applications for machine learning and data science. We have also highlighted the challenges and solutions for deploying machine learning web apps in a secure and compliant manner.

Overall, the application of machine learning in pharmaceutical manufacturing holds immense potential for enhancing efficiency, quality, and innovation in the industry. By staying up-to-date with the latest developments and adhering to industry standards and regulations, pharmaceutical professionals can harness the power of machine learning to transform the industry and improve patient outcomes.

REFERENCES

[1] What is Machine Learning? | IBM.

https://www.ibm.com/topics/machine-learning

[2] Applications of Machine Learning - Javatpoint.

https://www.javatpoint.com/applications-of-machine-learning

[3] Machine learning, explained | MIT Sloan.

https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained

[4] Machine Learning Allows Calibration Models to Predict Trace ... - Nature.
https://www.nature.com/articles/s41598-019-47751-y

[5] Multivariate Calibration in SIMCA of Spectroscopic Data - Sartorius.
https://www.sartorius.com/download/545668/simca-appnote3-spectroscopydata-en-b-00061-sartorius-data.pdf

[6] Wavelength Selection for Simultaneous Spectroscopic Analysis ....
https://pubs.acs.org/doi/10.1021/ac951142s

[7] Streamlit Tutorial: A Beginner's Guide to Python Streamlit - Built In.

https://builtin.com/machine-learning/streamlit-tutorial

[8] Streamlit • A faster way to build and share data apps.

https://streamlit.io/

[9] Advanced features - Streamlit Docs.

https://docs.streamlit.io/library/advanced-features

[10] Streamlit Web App | Build Web Applications using Streamlit.
https://www.analyticsvidhya.com/blog/2021/06/build-web-app-instantly-for-machine-learning-using-streamlit/

[11] Getting Started with RStudio Connect – Posit Support.

https://support.posit.co/hc/en-us/articles/115000960428-Getting-Started-with-RStudio-Connect

[12] RStudio Connect - RStudio.

https://www.rstudio.com/categories/rstudio-connect/

[13] What is RStudio Connect? - RStudio - Posit.

https://cloud.rstudio.com/resources/product-videos/what-is-rstudio-connect/

[14] What is the difference between rstudio-connect and rstudio-pro.
https://community.rstudio.com/t/what-is-the-difference-between-rstudio-connect-and-rstudio-pro/43949

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Kaiping Yang
Enterprise: BeiGene
Phone: 13588214900
E-mail: kaiping.yang@beigene.com