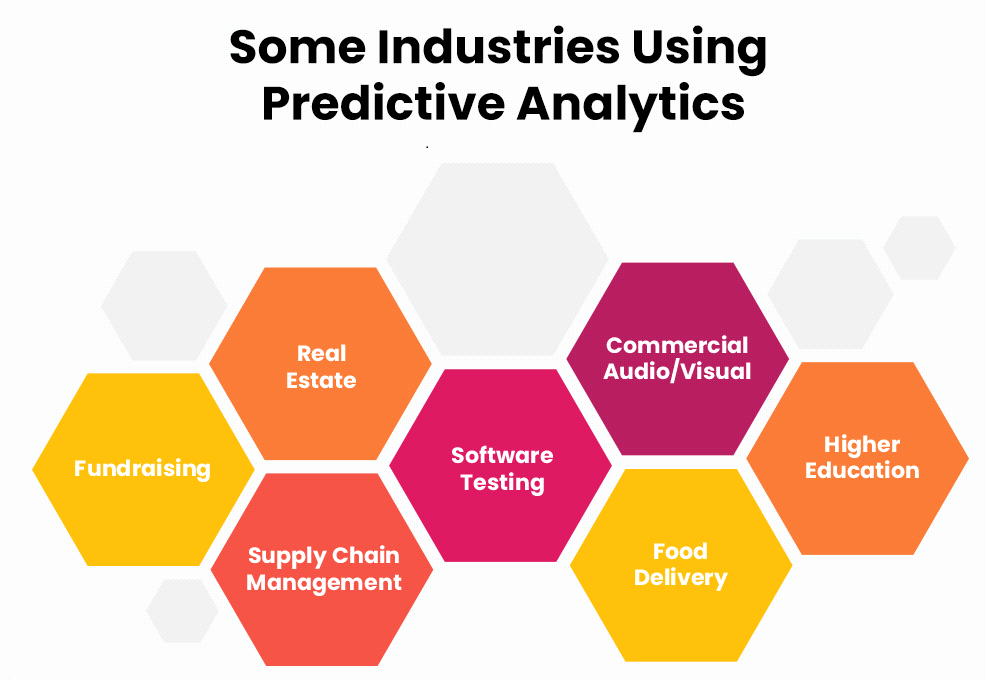
MACHINE LEARNING MODEL DEPLOYMENT WITH IBM CLOUD

WATSON STUDIO

INTRODUCTION :

**The term predictive analytics refers to the use of statistics and modeling techniques to make predictions about future outcomes and performance. Predictive analytics looks at current and historical data patterns to determine if those patterns are likely to emerge again. This allows businesses and investors to adjust where they use their resources to take advantage of possible future events. Predictive analysis can also be used to improve operational efficiencies and reduce risk.**

****

**WHAT IS DATA PREPROCESSING IN MACHINE LEARNING :**

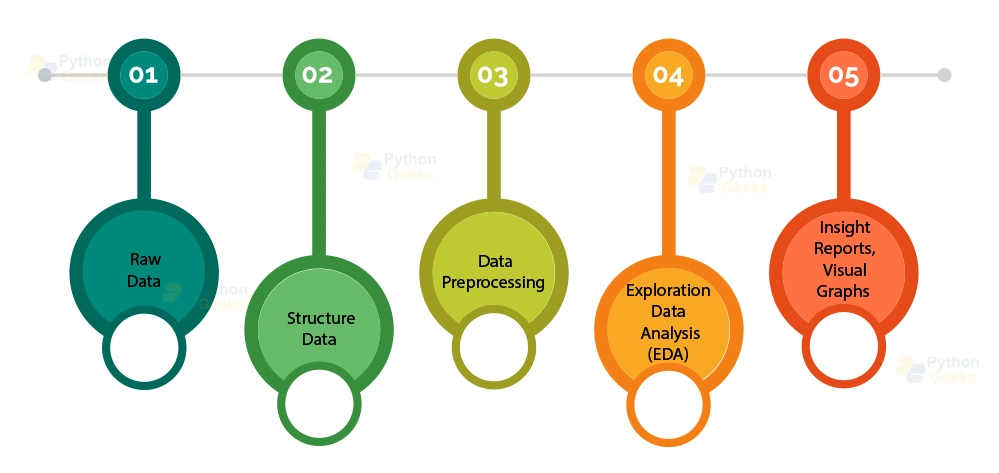
**Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.**

**It involves:**

**Analyzing, filtering, transforming, and encoding data**

**Cleaning and organizing data**

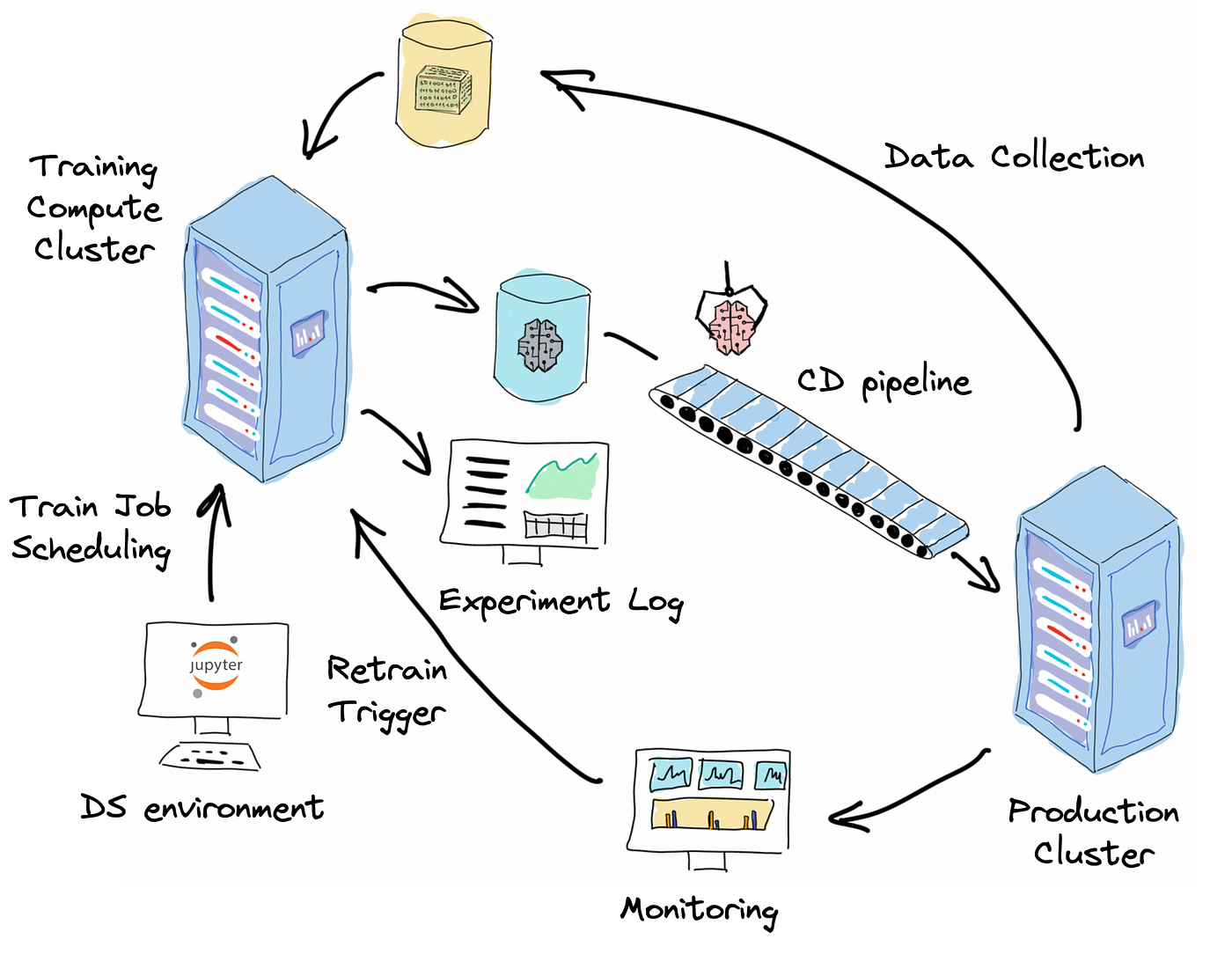
**Making data easier for a machine to parse**

****

**WHY DO WE NEED DATA PREPROCESSING :**

**A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data preprocessing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.**

**BUILD A MACHINE LEARNING MODEL USING IBM CLOUD WATSON STUDIO**

****

**A machine learning (ML) model is a program that uses a mathematical formula to make predictions about future events. It is trained on a set of data and then used to make predictions about new data.**

**There are three main types of ML models:**

**Descriptive: To help understand what happened in the past**

**Prescriptive: To automate business decisions and processes based on data**

**Predictive: To predict future business scenarios**

**Step 1: Define the Predictive Use Case**

**Start by defining your predictive use case. In this example, we'll use a customer churn prediction use case. Customer churn prediction involves identifying customers who are likely to leave a service or product, such as a subscription or a telecom service.**

**Step 2: Select a Customer Churn Prediction Dataset**

**we need a dataset to train and evaluate your machine learning model. we can find customer churn prediction datasets on various platforms, including Kaggle.** *Download a suitable dataset in a format like CSV.*

**Step 3:** Set Up IBM Cloud Watson Studio

Sign in or create an IBM Cloud account if we don't have one.Once logged in, access IBM Watson Studio through the cloud console.

**Step 4: Create a New Project**

**In Watson Studio, create a new project for our machine learning task. This project will help us organize your work.**

**Step 5: Import the Dataset**

**Inside the project:**

**Click "Add to project" and choose "Data."**

**Select "Load" and then "browse" to upload our customer churn dataset.**

**Once uploaded, add the data asset to our project.**

**Step 6: Preprocess the Data**

**we can preprocess the data using various tools and libraries within Watson Studio. Here's a Python code example :**

**import pandas as pd**

**from sklearn.model\_selection import train\_test\_split**

*# Load the dataset*

**data = pd.read\_csv('your\_churn\_dataset.csv')**

*# Data preprocessing*

*# Handle missing values, encode categorical variables, etc.*

*# Example: data = data.dropna()*

*# Example: data = pd.get\_dummies(data, columns=['categorical\_feature'])*

*# Split the data into training and testing sets*

**X = data.drop(columns=['target\_column'])**

**y = data['target\_column']**

**X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)**

**Step 7: Feature Selection**

**we can use various feature selection techniques available in Watson Studio or libraries like scikit-learn. Here's a simple example:**

**from sklearn.feature\_selection import SelectKBest, f\_classif**

*# Select the top K features*

**k\_best = SelectKBest(score\_func=f\_classif, k=5)**

**X\_train\_selected = k\_best.fit\_transform(X\_train, y\_train)**

**X\_test\_selected = k\_best.transform(X\_test)**

**Step 8: Train the Machine Learning Model**

**we can choose from various machine learning algorithms (e.g., Decision Trees, Random Forest, Logistic Regression). Here's an example using scikit-learn:**

**from sklearn.ensemble import RandomForestClassifier**

*# Create and train the model*

**model = RandomForestClassifier(n\_estimators=100, random\_state=42)**

**model.fit(X\_train\_selected, y\_train)**

**Step 9: Evaluate the Model**

**Evaluate the model's performance using metrics like accuracy, precision, recall, and F1-score. You can do this using code and tools provided in Watson Studio.**

**Step 10: Deploy the Model**

**If the model performs well, you can deploy it within IBM Cloud for real-time or batch predictions.**

**These are the general steps involved in building a machine learning model using IBM Cloud Watson Studio.**

**OVERALL GENERAL PROGRAM :**

**import pandas as pd**

**from sklearn.model\_selection import train\_test\_split**

**from sklearn.preprocessing import StandardScaler, LabelEncoder**

**from sklearn.compose import ColumnTransformer**

**from sklearn.pipeline import Pipeline**

*# Load the dataset*

**data = pd.read\_csv("customer\_data.csv")**

*# Data Cleaning and Preprocessing*

*# Handle missing values (if any)*

**data = data.dropna()**

*# Encoding categorical variables*

**label\_encoder = LabelEncoder()**

**data['contract'] = label\_encoder.fit\_transform(data['contract'])**

*# Example: Convert contract types to numeric values*

*# Split the dataset into features (X) and target variable (y)*

**X = data.drop(columns=['customer\_id', 'churn'])**

**y = data['churn']**

*# Split data into training and testing sets*

**X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)**

*# Feature Scaling (if needed)*

**scaler = StandardScaler()**

**X\_train[['age', 'monthly\_charges', 'total\_charges', 'tenure']] = scaler.fit\_transform(X\_train[['age', 'monthly\_charges', 'total\_charges', 'tenure'])**

**X\_test[['age', 'monthly\_charges', 'total\_charges', 'tenure']] = scaler.transform(X\_test[['age', 'monthly\_charges', 'total\_charges', 'tenure'])**

*# You can apply other preprocessing steps like one-hot encoding for categorical variables if necessary*

*# Create a machine learning pipeline*

*# (You would replace the placeholders with your actual preprocessing and modeling steps)*

**pipeline = Pipeline([**

**('preprocessing', ColumnTransformer(transformers=[**

**('num', StandardScaler(), ['age', 'monthly\_charges', 'total\_charges', 'tenure']),**

**# Add more transformers for other features if needed**

**])),**

**('model', YourMachineLearningModel()) # Replace with your machine learning model**

**])**

*# Train the model*

**pipeline.fit(X\_train, y\_train)**

***# Evaluate the model***

**accuracy = pipeline.score(X\_test, y\_test)**

**print(f"Accuracy: {accuracy}")**

*# Make predictions*

**predictions = pipeline.predict(X\_test)**

**CONCLUSION :**

**in this doccument submitted ,i had include the definition of predictive use analysis and data preprocessing . i had also clarify what is need of data preprocessing ,followed by introduction . further we see how to login and build an machine learning model in IBM cloud watson studio .after that we see the samle general progrram written in python .**

**doccument submitted by ,**

**N.Gokulkumar .**