**Module 2: Interpretability in Machine Learning**

**Topic 1. Introduction to Machine Learning**

**Resource link -** https://monkeylearn.com/machine-learning/#:~:text=Machine%20learning%20(ML)%20is%20a,to%20make%20their%20own%20predictions

The term machine learning was first coined in the 1950s when Artificial Intelligence pioneer Arthur Samuel built the first self-learning system for playing checkers. He noticed that the more the system played, the better it performed.

Fueled by advances in statistics and computer science, as well as better datasets and the growth of neural networks, machine learning has truly taken off in recent years.

**What Is Machine Learning?**

Machine learning (ML) is a branch of artificial intelligence (AI) that enables computers to “self-learn” from training data and improve over time, without being explicitly programmed. Machine learning algorithms are able to detect patterns in data and learn from them, in order to make their own predictions. In short, machine learning algorithms and models learn through experience.

In traditional programming, a computer engineer writes a series of directions that instruct a computer how to transform input data into a desired output. Instructions are mostly based on an IF-THEN structure: when certain conditions are met, the program executes a specific action.

Machine learning, on the other hand, is an automated process that enables machines to solve problems with little or no human input, and take actions based on past observations.

**Types of Machine Learning**

1. **Supervised Learning**

Supervised learning algorithms and supervised learning models make predictions based on labeled training data. Each training sample includes an input and a desired output. A supervised learning algorithm analyzes this sample data and makes an inference – basically, an educated guess when determining the labels for unseen data.

This is the most common and popular approach to machine learning. It’s “supervised” because these models need to be fed manually tagged sample data to learn from. Data is labeled to tell the machine what patterns (similar words and images, data categories, etc.) it should be looking for and recognize connections with.

**Classification in supervised machine learning**

There are a number of classification algorithms used in supervised learning, with Support Vector Machines (SVM) and Naive Bayes among the most common.

In classification tasks, the output value is a category with a finite number of options. For example, with this free pre-trained sentiment analysis model, you can automatically classify data as positive, negative, or neutral.

**Regression in supervised machine learning**

In regression tasks, the expected result is a continuous number. This model is used to predict quantities, such as the probability an event will happen, meaning the output may have any number value within a certain range. Predicting the value of a property in a specific neighborhood or the spread of COVID19 in a particular region are examples of regression problems.

1. **Unsupervised Learning**

Unsupervised learning algorithms uncover insights and relationships in unlabeled data. In this case, models are fed input data but the desired outcomes are unknown, so they have to make inferences based on circumstantial evidence, without any guidance or training. The models are not trained with the “right answer,” so they must find patterns on their own.

One of the most common types of unsupervised learning is clustering, which consists of grouping similar data. This method is mostly used for exploratory analysis and can help you detect hidden patterns or trends.

For example, the marketing team of an e-commerce company could use clustering to improve customer segmentation. Given a set of income and spending data, a machine learning model can identify groups of customers with similar behaviors.

1. **Semi Supervised learning**

In semi-supervised learning, training data is split into two. A small amount of labeled data and a larger set of unlabeled data.

In this case, the model uses labeled data as an input to make inferences about the unlabeled data, providing more accurate results than regular supervised-learning models.

This approach is gaining popularity, especially for tasks involving large datasets such as image classification. Semi-supervised learning doesn’t require a large number of labeled data, so it’s faster to set up, more cost-effective than supervised learning methods, and ideal for businesses that receive huge amounts of data.

1. **Reinforcement learning**

Reinforcement learning (RL) is concerned with how a software agent (or computer program) ought to act in a situation to maximize the reward. In short, reinforced machine learning models attempt to determine the best possible path they should take in a given situation. They do this through trial and error. Since there is no training data, machines learn from their own mistakes and choose the actions that lead to the best solution or maximum reward.

This machine learning method is mostly used in robotics and gaming. Video games demonstrate a clear relationship between actions and results, and can measure success by keeping score. Therefore, they’re a great way to improve reinforcement learning algorithms.

**Topic 2: Definitions of Interpretability and Examples**

Resource link - https://christophm.github.io/interpretable-ml-book/interpretability.html

**What is Interpretability?**

Interpretability is the degree to which a human can consistently predict the model’s result. The higher the interpretability of a machine learning model, the easier it is for someone to comprehend why certain decisions or predictions have been made. A model is better interpretable than another model if its decisions are easier for a human to comprehend than decisions from the other model.

**Importance of Interpretability**

Let us dive deeper into the reasons why interpretability is so important. When it comes to predictive modeling, you have to make a trade-off: Do you just want to know what is predicted? For example, the probability that a customer will churn or how effective some drug will be for a patient. Or do you want to know why the prediction was made and possibly pay for the interpretability with a drop in predictive performance? In some cases, you do not care why a decision was made, it is enough to know that the predictive performance on a test dataset was good.

**When we do not need interpretability.**

The following scenarios illustrate when we do not need or even do not want interpretability of machine learning models.

Interpretability is not required if the model has no significant impact. Imagine someone named Mike working on a machine learning side project to predict where his friends will go for their next holidays based on Facebook data. Mike just likes to surprise his friends with educated guesses where they will be going on holidays. There is no real problem if the model is wrong (at worst just a little embarrassment for Mike), nor is there a problem if Mike cannot explain the output of his model. It is perfectly fine not to have interpretability in this case. The situation would change if Mike started building a business around these holiday destination predictions. If the model is wrong, the business could lose money, or the model may work worse for some people because of learned racial bias. As soon as the model has a significant impact, be it financial or social, interpretability becomes relevant.

**Evaluation of Interpretability**

There is no real consensus about what interpretability is in machine learning. Nor is it clear how to measure it. But there is some initial research on this and an attempt to formulate some approaches for evaluation, as described in the following section.

Doshi-Velez and Kim (2017) propose three main levels for the evaluation of interpretability:

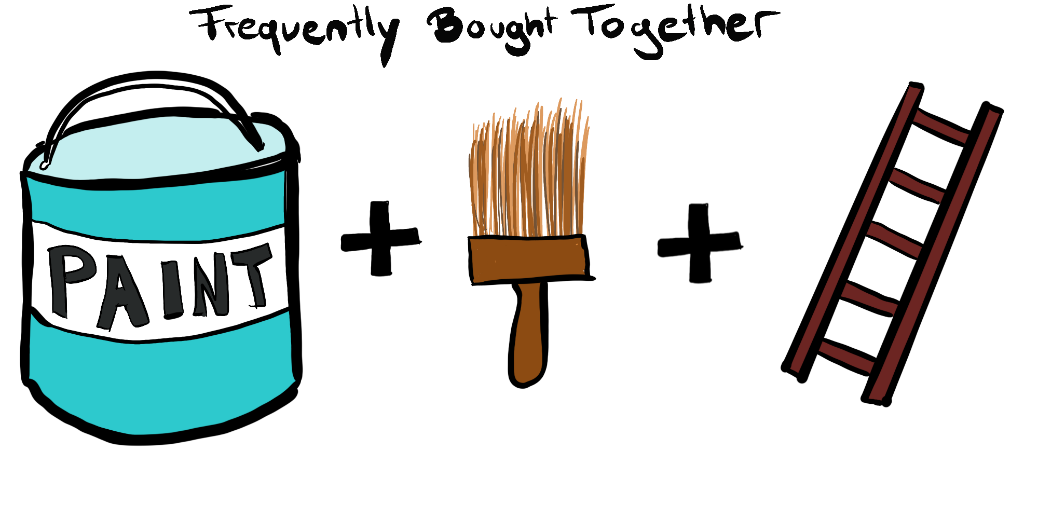
**Application level evaluation (real task):** Put the explanation into the product and have it tested by the end user. Imagine fracture detection software with a machine learning component that locates and marks fractures in X-rays. At the application level, radiologists would test the fracture detection software directly to evaluate the model. This requires a good experimental setup and an understanding of how to assess quality. A good baseline for this is always how good a human would be at explaining the same decision.

**Human level evaluation (simple task)** is a simplified application level evaluation. The difference is that these experiments are not carried out with the domain experts, but with laypersons. This makes experiments cheaper (especially if the domain experts are radiologists) and it is easier to find more testers. An example would be to show a user different explanations and the user would choose the best one.

**Function level evaluation (proxy task)** does not require humans. This works best when the class of model used has already been evaluated by someone else in a human level evaluation. For example, it might be known that the end users understand decision trees. In this case, a proxy for explanation quality may be the depth of the tree. Shorter trees would get a better explainability score. It would make sense to add the constraint that the predictive performance of the tree remains good and does not decrease too much compared to a larger tree.

**Example of Interpretability**

Another example is algorithmic product recommendation. Personally, I always think about why certain products or movies have been algorithmically recommended to me. Often it is quite clear: Advertising follows me on the Internet because I recently bought a washing machine, and I know that in the next days I will be followed by advertisements for washing machines. Yes, it makes sense to suggest gloves if I already have a winter hat in my shopping cart. The algorithm recommends this movie, because users who liked other movies I liked also enjoyed the recommended movie. Increasingly, Internet companies are adding explanations to their recommendations. A good example are product recommendations, which are based on frequently purchased product combinations:



In many scientific disciplines there is a change from qualitative to quantitative methods (e.g. sociology, psychology), and also towards machine learning (biology, genomics). The goal of science is to gain knowledge, but many problems are solved with big datasets and black box machine learning models. The model itself becomes the source of knowledge instead of the data. Interpretability makes it possible to extract this additional knowledge captured by the model.

**Topic 3: Social and Commercial Motivations for Machine Learning**

Resource link - https://machinelearningknowledge.ai/machine-learning-examples/

People who are exploring machine learning as a field to shift their careers or people who are just curious about why there is such a buzz of machine learning all over places often have one burning question in their mind – what all possible things can one achieve with machine learning. Well, the short answer is – that the possibility is endless and one’s creativity is the only limit.

Machine learning is influencing our day-to-day life and perhaps many people are not even aware of this. The key reason for this penetration is Google. Google is an integral part of one’s life and people are knowingly or unknowingly getting driven by machine learning. Let us have a look at a few examples –

1. Google search engine now processes over 40,000 searches per second on average and billions of searches daily and yet throws relevant search results within seconds. These numbers are mind-boggling in themselves.
2. Did you ever freak out at how google gives such accurate autocomplete prompts in the search boxes as if it is reading our minds about search context?
3. How the hell, google maps predict which route will be faster and shows such an accurate ETA of reaching the destination.
4. Ever noticed that spam folder in your Gmail and did you ever open it to see so many spam mails indeed.
5. And the wonderful google translator, which is becoming better and better with time.
6. Youtube suggestions, which keep people hooked to youtube for endless hours.

**Machine Learning in other industrial uses**

Now that we have made you realize how machine learning is influencing your regular life so closely, let us now understand some other popular use cases for which industries are adopting machine learning across the world.

1. Banks are using machine learning to detect frauds, pass loans, and maintain investment portfolios.
2. Hospital s are using machine learning to detect and diagnose diseases with more accuracy than actual doctors.
3. Companies are running ad campaigns targeted to specific customer segments using machine learning
4. Companies are now managing their supply chain and inventory control using machine learning.
5. Enterprise chatbots are now the next big things, with companies adopting them to interface with customers
6. Online retailers are building their recommendation systems with more accuracy than before using the latest advancements in machine learning.
7. Self Driving Cars are the latest fascination that has caught the imaginations of companies like Google, Uber, and Tesla who are investing heavily into this future vision using next-generation machine learning technology.
8. With the rising industry interest in Blockchain, Web 3.0, and Metaverse, machine learning technologies are going to play a huge role there also.

**Topic 4: Machine Learning Interpretability Taxonomy for Applied Practitioners**

Resource link - https://christophm.github.io/interpretable-ml-book/taxonomy-of-interpretability-methods.html

Methods for machine learning interpretability can be classified according to various criteria.

Intrinsic or post hoc? This criterion distinguishes whether interpretability is achieved by restricting the complexity of the machine learning model (intrinsic) or by applying methods that analyse the model after training (post hoc). Intrinsic interpretability refers to machine learning models that are considered interpretable due to their simple structure, such as short decision trees or sparse linear models. Post hoc interpretability refers to the application of interpretation methods after model training. Permutation feature importance is, for example, a post hoc interpretation method. Post hoc methods can also be applied to intrinsically interpretable models. For example, permutation feature importance can be computed for decision trees. The organization of the chapters in this book is determined by the distinction between intrinsically interpretable models and post hoc (and model-agnostic) interpretation methods.

Result of the interpretation method the various interpretation methods can be roughly differentiated according to their results.

Feature summary statistic: Many interpretation methods provide summary statistics for each feature. Some methods return a single number per feature, such as feature importance, or a more complex result, such as the pairwise feature interaction strengths, which consist of a number for each feature pair.

Feature summary visualization: Most of the feature summary statistics can also be visualized. Some feature summaries are actually only meaningful if they are visualized, and a table would be a wrong choice. The partial dependence of a feature is such a case. Partial dependence plots are curves that show a feature and the average predicted outcome. The best way to present partial dependences is to draw the curve instead of printing the coordinates.

Model internals (e.g., learned weights): The interpretation of intrinsically interpretable models falls into this category. Examples are the weights in linear models or the learned tree structure (the features and thresholds used for the splits) of decision trees. The lines are blurred between model internals and feature summary statistic in, for example, linear models, because the weights are both model internals and summary statistics for the features at the same time. Another method that outputs model internals is the visualization of feature detectors learned in convolutional neural networks. Interpretability methods that output model internals are model-specific (see next criterion).

**Topic 5: Common Interpretability Techniques**

Resource Link - https://neptune.ai/blog/ml-model-interpretation-tools

Interpretation is literally defined as explaining or showing your own understanding of something. When you create an ML model, which is nothing but an algorithm that can learn patterns, it might feel like a black box to other project stakeholders. Sometimes even to you. Which is why we have model interpretation tools.

**What is Model Interpretation?**

In general, an ML model has to obtain predictions, and use those predictions and eventual insights to solve a range of problems. Already, we can ask a couple of follow-up questions:

How trustworthy are these predictions?

Are they reliable enough to make big decisions?

Model Interpretation redirects your focus from ‘what was the conclusion?’ to ‘why was this conclusion reached?’. You can get an understanding of the model’s decision-making process, i.e. what exactly drives the model to classify a data point correctly or incorrectly.

**Why is Model Interpretation important?**

Consider an example of a husky versus wolf (dog breed) classifier, in which a few huskies were misclassified as wolves. Using interpretable machine learning, you might find that these misclassifications mainly happened because of snow in the image, which the classifier was using as a feature to predict wolves.

It’s a simple example, but already you can see why Model Interpretation is important. It helps your model in at least a few aspects:

Fairness – An interpretable model used by a company to decide raises and promotions can tell you exactly why any particular person was, or wasn’t offered a promotion.

Reliability – Small changes in input won’t lead to a domino effect and alter the output drastically.

Causality – Only causal relationships are useful for decision making.

Trust – It’s easier for all project stakeholders, especially on the non-technical side, to trust a model that can be explained in layman’s terms.

**How to interpret an ML model?**

Machine Learning models vary in degrees of complexity and performance. One size doesn’t fit them all. As a result, there are different ways to interpret them. Primarily, these methods can be categorized as:

**Model-specific / Model-agnostic**

Model-specific methods are specific to certain models, they depend on the inner machinery of a model to make certain conclusions. These methods may include the interpretation of coefficient weights in Generalized Linear Models (GLMs), or weights and biases in the case of Neural Networks.

Model-agnostic methods can be used on any model. They’re generally applied post-training. They usually work by analyzing the relationship between feature input-output pairs and don’t have access to the model’s internal mechanics such as weights or assumptions.

**Local / Global scope**

The local scope covers only an individual prediction, capturing the reasons behind only the specified prediction.

The global scope extends beyond an individual data point and covers the model’s general behavior.

Let’s create a model to interpret. We’ll do a short walkthrough of the model creation steps, and then we’ll focus on different model-agnostic tools and frameworks to interpret the created model, rather than solve the actual problem.

**Topic 6: Machine Learning Modelling from Healthcare Data, Benefits and Challenges**

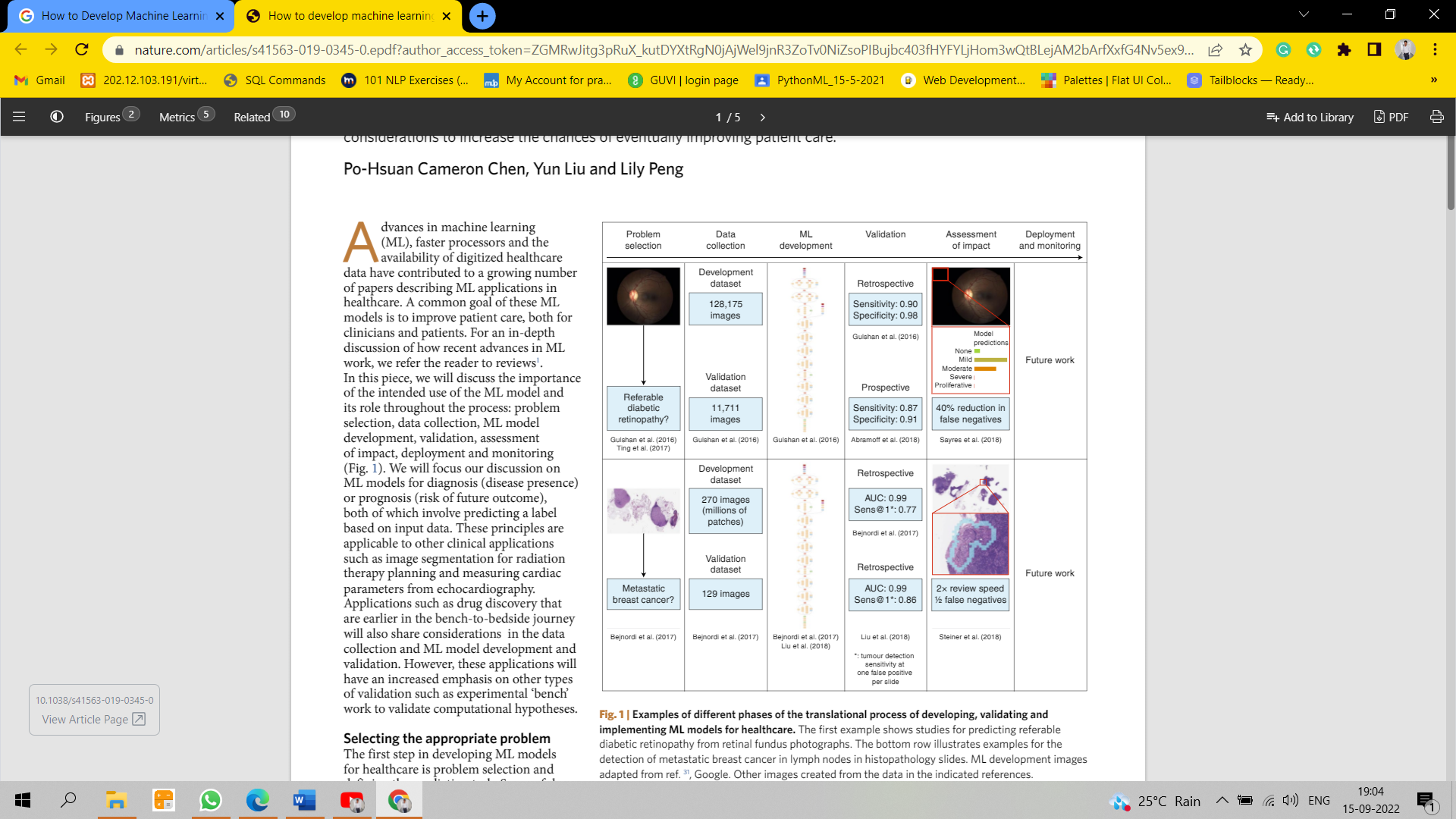
Reference link - https://www.nature.com/articles/s41563-019-0345-0.epdf?author\_access\_token=ZGMRwJitg3pRuX\_kutDYXtRgN0jAjWel9jnR3ZoTv0NiZsoPIBujbc403fHYFYLjHom3wQtBLejAM2bArfXxfG4Nv5ex9ozdDOtcUou5Ws9AIQrx\_iwUOmisAGRQcQp613cGPa7yADfIhNS-Txy5vA%3D%3D

Advances in machine learning (ML), faster processors and the availability of digitized healthcare data have contributed to a growing number of papers describing ML applications in healthcare. A common goal of these ML models is to improve patient care, both for clinicians and patients.

ML models for diagnosis (disease presence) or prognosis (risk of future outcome), both of which involve predicting a label based on input data. These principles are applicable to other clinical applications such as image segmentation for radiation therapy planning and measuring cardiac parameters from echocardiography. Applications such as drug discovery that are earlier in the bench-to-bedside journey will also share considerations in the data collection and ML model development and validation. However, these applications will have an increased emphasis on other types of validation such as experimental ‘bench’ work to validate computational hypotheses.

**Selecting the appropriate problem**

The first step in developing ML models for healthcare is problem selection and defining the prediction task. Successful ML models should be expected to make a meaningful impact in patient care by providing actionable insights. For example, an ML model can be used to predict factors that are used in clinical treatment guidelines.



**Developing models**

The process of model development for eventual clinical implementation has several main considerations that influence model architecture design: data modality and volume, model interpretability, model inference time, and balancing model overfitting and underfitting. A wide range of data modalities exist in healthcare, such as 2D images, 3D volumes, waveforms, laboratory measurements and text. For state-of-the-art performance, the ML model should be appropriate for the data modality, for example using convolutional neural networks for images and recurrent neural networks for sequences of waveforms, text, measurements. With each model type, the ‘complexity’ of the model (for example, as measured by the number of parameters) should also be appropriate given the dataset size. For example, a 100-layer network may overfit a classification model trained using a dataset of only 100 images.