Chapter ML:I (continued)

I. Introduction

- Examples of Learning Tasks
- Specification of Learning Tasks
- Elements of Machine Learning

- Notation Overview
- Classification Approaches Overview
Elements of Machine Learning

(1) Model Formation: Real World → Model World

Related questions:
- From what kind of experience should be learned?
- Which level of fidelity is sufficient to solve a certain task?
Elements of Machine Learning

(2) Design Choices in the Machine Learning Stack: LMS

- Optimization approach
- Optimization objective
  - Loss function [ + Regularization ]
- Model function $\sim$ Hypothesis space

Task

Data
Elements of Machine Learning

(2) Design Choices in the Machine Learning Stack: LMS (continued)

- **Optimization approach**
  - **Optimization objective**
    - Loss function: $[ + \text{Regularization} ]$
  - **Model function $\rightarrow$ Hypothesis space**

- **Task**
  - **Data**

- Binary classification

$$D = \{(x_1, c_1), \ldots, (x_n, c_n)\} \subseteq X \times \{-1, 1\}$$
(2) Design Choices in the Machine Learning Stack: LMS (continued)

- Optimization approach
  - Optimization objective
    - Loss function \[ + \text{ Regularization} \]
  - Model function \( \sim \) Hypothesis space

- Task
- Data

- Hypothesis space: \( w \in \mathbb{R}^{p+1} \)
- Linear model: \( y(x) = w_0 + \sum_{i=1}^{p} w_i x_i \)

Binary classification

\[ D = \{(x_1, c_1), \ldots, (x_n, c_n)\} \subseteq X \times \{-1, 1\} \]
(2) Design Choices in the Machine Learning Stack: LMS (continued)

- **Optimization approach**
  - **Optimization objective**
    - Loss function: $\ell_2(c, y(x)) = (c - y(x))^2$, $(x, c) \in D$
    - Hypothesis space: $w \in \mathbb{R}^{p+1}$
    - Linear model: $y(x) = w_0 + \sum_{i=1}^{p} w_i x_i$

- **Task**
  - Binary classification

- **Data**
  - $D = \{(x_1, c_1), \ldots, (x_n, c_n)\} \subseteq X \times \{-1, 1\}$
Elements of Machine Learning
(2) Design Choices in the Machine Learning Stack: LMS (continued)

- Optimization approach
- Optimization objective
  - Loss function \[ + \text{ Regularization} \]
- Model function \( \leadsto \) Hypothesis space

- Task
- Data

Stochastic gradient descent (SGD)
- Objective: minimize squared loss (RSS)
- Regularization: none
- Loss: \( l_2(c, y(x)) = (c - y(x))^2 \), \((x, c) \in D\)
- Hypothesis space: \( w \in \mathbb{R}^{p+1} \)
- Linear model: \( y(x) = w_0 + \sum_{i=1}^{p} w_i x_i \)

Binary classification
\[ D = \{(x_1, c_1), \ldots, (x_n, c_n)\} \subseteq X \times \{-1, 1\} \]
Related questions:

- What are useful classes of model functions?
- What are methods to fit (= learn) model functions?
- What are measures to assess the goodness of fit?
- How does (label) noise affect the learning process?
- How does the example number affect the learning process?
- How to deal with extreme class imbalance?
The feature space is an inner product space.

- An inner product space (also called pre-Hilbert space) is a vector space with an operation called “inner product”.
- Example: Euclidean vector space equipped with the dot product.
- Enables algorithms such as gradient descent and support vector machines.
The feature space is an inner product space.

- An inner product space (also called pre-Hilbert space) is a vector space with an operation called “inner product”.
- Example: Euclidean vector space equipped with the dot product.
- Enables algorithms such as gradient descent and support vector machines.

The feature space is a $\sigma$-algebra.

- A $\sigma$-algebra on a set $\Omega$ is a collection of subsets of $\Omega$ that includes $\Omega$ itself, is closed under complement, and is closed under countable unions.
- Enables probability spaces and statistical learning, such as naive Bayes.
The feature space is an inner product space.

- An inner product space (also called pre-Hilbert space) is a vector space with an operation called “inner product”.
- Example: Euclidean vector space equipped with the dot product.
- Enables algorithms such as gradient descent and support vector machines.

The feature space is a $\sigma$-algebra.

- A $\sigma$-algebra on a set $\Omega$ is a collection of subsets of $\Omega$ that includes $\Omega$ itself, is closed under complement, and is closed under countable unions.
- Enables probability spaces and statistical learning, such as naive Bayes.

The feature space is a finite set of vectors with nominal dimensions.

- Requires concept learning via set splitting as done by decision trees.
Remarks:

- The aforementioned examples of feature spaces are not meant to be complete. But, they illustrate a broad range of structures underlying the example sets we want to learn from.

- The structure of a feature space constrains the applicable learning algorithm. Usually, this structure is inherently determined by the application domain and cannot be chosen.
(4) Discriminative versus Generative Approach to Classification

- Discriminative classifiers (models) learn a boundary between classes.
- Generative classifiers exploit the distributions underlying the classes.
Discriminative versus Generative Approach to Classification (continued)

- Discriminative classifiers (models) learn a boundary between classes.
- Generative classifiers exploit the distributions underlying the classes.

Discriminative classification rule

\[ \sim \text{discriminative} \]

\[ \sim \text{classification rule} \]

Generative class membership probability

\[ \sim \text{generative} \]

\[ \sim \text{class membership probability} \]
Discriminative versus Generative Approach to Classification (continued)

- Discriminative classifiers (models) learn a boundary between classes.
- Generative classifiers exploit the distributions underlying the classes.

**Diagrams:**

- **Discriminative** classification rule:
  - A straight line separating data points of different classes.
  - Symbol `x` represents a new data point to classify.

- **Generative** class membership probability:
  - Ellipses representing probability distributions for each class.
  - Symbol `x` represents a new data point to classify.
When classifying a new example $x$, then

1. discriminative classifiers apply a decision rule that was learned via minimizing the misclassification rate given training examples $D$, while

2. generative classifiers maximize the probability of the combined event $p(x, y)$, or, similarly, the posterior probability $p(y \mid x)$, $y \in \{\emptyset, \oplus\}$.

The LMS algorithm computes “only” a decision boundary, i.e., it constructs a discriminative classifier. A Bayes classifier is an example for a generative model.

Yoav Freund provides an excellent video illustrating the pros and cons of discriminative and generative models respectively. [YouTube]

Discriminative models may be further differentiated in models that also determine the posterior class probabilities $p(y \mid x)$ (without computing the joint probabilities $p(x, y)$) and those that do not. In the latter case, only a so-called “discriminant function” is computed.
Frequentist:

- There is a hidden but unique mechanism that generated the data $D$.
- Consider a model for this mechanism, such as a family of distributions or a model function, parameterized by $\theta$, $\theta$, $w$, or similar. The considered parameter values (or vectors) form the hypothesis space $H$.
- Select for the unknown parameter (vector) that element from $H$ such that the observed data $D$ becomes most probable. The chosen element (our hypothesis), $h_{ML}$, is called maximum likelihood hypothesis.
Frequentist:

- There is a hidden but unique mechanism that generated the data $D$.
- Consider a model for this mechanism, such as a family of distributions or a model function, parameterized by $\theta$, $\theta$, $w$, or similar. The considered parameter values (or vectors) form the hypothesis space $H$.
- Select for the unknown parameter (vector) that element from $H$ such that the observed data $D$ becomes most probable. The chosen element (our hypothesis), $h_{ML}$, is called maximum likelihood hypothesis.

$$
\begin{align*}
\theta^*, \theta^* \text{ or } w^* \sim D, \quad h_{ML} &= \arg\max_{h \in H} p(D; h)
\end{align*}
$$
Frequentist:

- There is a hidden but \textbf{unique} mechanism that generated the data $D$.
- Consider a model for this mechanism, such as a family of distributions or a model function, parameterized by $\theta$, $\theta$, $w$, or similar. The considered parameter values (or vectors) form the hypothesis space $H$.
- Select for the unknown parameter (vector) that element from $H$ such that the observed data $D$ becomes most probable. The chosen element (our hypothesis), $h_{\text{ML}}$, is called \textit{maximum likelihood hypothesis}.

$$
\theta^*, \theta^* \text{ or } w^* \sim D, \quad h_{\text{ML}} = \underset{h \in H}{\text{argmax}} \ p(D; h)
$$

$$
\theta_{\text{ML}} = \underset{\theta \in [0;1]}{\text{argmax}} \ p(D; \theta) = \underset{\theta \in [0;1]}{\text{argmax}} \ \binom{n}{k} \cdot \theta^k \cdot (1 - \theta)^{n-k}
$$
Remarks:

- Likelihood is the hypothetical probability that an event that has already occurred (here: a coin flip experiment parameterized by $\theta$) would yield a specific outcome (here: a sequence $D$ of heads and tails).

  The concept differs from that of a probability in that a probability refers to the occurrence of future events, while a likelihood refers to past events with known outcomes. I.e., $p(D)$ is called likelihood since we reason about a past coin flip experiment. [Mathworld]

- By definition, the unknown parameter value (vector) of the data generation mechanism, $\theta^*$, $\theta^*$, $w^*$, etc., is considered as unique and has some value from $H$.

  This means that $\theta$, $\theta$, or $h$ in the argmax-expression is not the realization of a random variable or random vector—which would come along with a distribution and an expected value—but an exogenous parameter (vector), which we vary to find the maximum of $p(D; \theta)$, $p(D; \theta)$, $p(D; w)$, or, in general, $p(D; h)$.

  The fact that $h$ is a given, unique parameter (though it needs to be searched) and not a random variable is reflected by the notation, which uses a »;« instead of a »|« in $p()$.

- In the experiment of flipping a coin, we assume a Laplace experiment and apply the binomial distribution, $B(n, p)$, with exactly $k$ successes in $n$ independent Bernoulli trials.

- A general method for finding the maximum likelihood estimate of the parameters of an underlying distribution from a given data set $D$ (even if the data is incomplete) is the Expectation-Maximization (EM) algorithm. [Bilmes 1998]
Subjectivist:

- There is a hidden but ambiguous mechanism that generated the data $D$.
- As before, consider a model for this mechanism. In addition, we have beliefs (subjective prior probabilities) $p(h)$ for all elements in the hypothesis space $H$.
- Select the most probable hypothesis $h_{\text{MAP}} \in H$ by weighting the likelihoods $p(D | h), h \in H$, with the priors. $h_{\text{MAP}}$ is called maximum posterior hypothesis.
Subjectivist:

- There is a hidden but ambiguous mechanism that generated the data $D$.
- As before, consider a model for this mechanism. In addition, we have beliefs (subjective prior probabilities) $p(h)$ for all elements in the hypothesis space $H$.
- Select the most probable hypothesis $h_{\text{MAP}} \in H$ by weighting the likelihoods $p(D \mid h), h \in H$, with the priors. $h_{\text{MAP}}$ is called maximum posterior hypothesis.

Belief/Prior 1: $P(\Theta = 0.5) = 0.95$

Belief/Prior 2: $P(\Theta = 0.75) = 0.05$
Subjectivist:

- There is a hidden but ambiguous mechanism that generated the data $D$.
- As before, consider a model for this mechanism. In addition, we have beliefs (subjective prior probabilities) $p(h)$ for all elements in the hypothesis space $H$.
- Select the most probable hypothesis $h_{\text{MAP}} \in H$ by weighting the likelihoods $p(D \mid h), h \in H$, with the priors. $h_{\text{MAP}}$ is called maximum posterior hypothesis.

Belief/Prior 1: $P(\Theta=0.5) = 0.95$

Belief/Prior 2: $P(\Theta=0.75) = 0.05$

$\theta_1 + D \rightarrow p(D \mid \theta_1)$

$\theta_2 + D \rightarrow p(D \mid \theta_2)$
Subjectivist:

- There is a hidden but ambiguous mechanism that generated the data $D$.
- As before, consider a model for this mechanism. In addition, we have beliefs (subjective prior probabilities) $p(h)$ for all elements in the hypothesis space $H$.
- Select the most probable hypothesis $h_{\text{MAP}} \in H$ by weighting the likelihoods $p(D \mid h), h \in H$, with the priors. $h_{\text{MAP}}$ is called maximum posterior hypothesis.

\[
\begin{align*}
\text{Belief/Prior 1: } \ P(\Theta = 0.5) &= 0.95 \\
\theta_1 + D &\rightarrow p(D \mid \theta_1) \\
\text{Belief/Prior 2: } P(\Theta = 0.75) &= 0.05 \\
\theta_2 + D &\rightarrow p(D \mid \theta_2)
\end{align*}
\]

\[
\theta_{\text{MAP}} = \arg\max_{\theta \in \{\theta_1, \theta_2\}} p(\theta \mid D) = \arg\max_{\theta \in \{\theta_1, \theta_2\}} \frac{p(D \mid \theta) \cdot p(\theta)}{p(D)}
\]
By definition, the elements in $H$ (here: $\theta_1, \theta_2$) are considered as realizations of the random variable $\Theta$. There is (subjective) prior knowledge about the distribution of $\Theta$. Here, $\Theta$ models the parameter $p$ of the binomial distribution and defines the success probability for each trial.

- Belief in $\theta_1$ ($\Theta=0.5$): With probability 0.95 the coin is fair, i.e., sides are equally likely.
- Belief in $\theta_2$ ($\Theta=0.75$): With probability 0.05 the odds of preferring one side is 3:1.

We compute for each element in $H$ the likelihood of the observed data $D$, i.e., $p(D \mid \theta_1)$ and $p(D \mid \theta_2)$ under the binomial distribution. We then compute the respective values for $p(\theta_1 \mid D)$ and $p(\theta_2 \mid D)$ with Bayes’s rule, and finally select $\theta_{MAP}$.

The fact that $h$ is the realization of a random variable (and not an exogenous parameter) is reflected by the notation, which uses a »|« in $p()$ (and not a »;«).

- The subjectivist paradigm is powerful, if we want to consider knowledge about $H$ that we cannot get from $D$ by maximizing the likelihood. The subjectivist paradigm is necessary, if we have no data $D$ to optimize, e.g., if we reason about “one time events”. If all hypotheses are equally likely (a uniform prior), ML optimization and MAP optimization are equivalent.

If the prior probabilities (here: $p(\theta_1), p(\theta_2)$) are estimated from $D$ as well, we still apply the Bayes calculation rule for a “MAP hypothesis”. However, we are not subjective anymore but follow the frequentist paradigm.

- The subjectivist paradigm is also called Bayesian interpretation of probability. It enables by design the integration of prior knowledge or human expertise about alternative mechanisms one of which generated $D$. [Wikipedia: Bayesian interpretation, probability interpretations]