COL866: Foundations of Data Science

Ragesh Jaiswal, IITD

Machine Learning

- One of the main tasks in Machine Learning is classification.
  - The goal is to learn a rule for labeling data (given a few labeled examples).
- The data comes from an instance space $\mathcal{X}$ and typically $\mathcal{X} = \mathbb{R}^d$ or $\mathcal{X} = \{0, 1\}^d$.
- So, a data item is typically described by a d-dimensional feature vector.
  - For example in spam classification, the features could be the presence (or absence) of certain words.
- For performing the learning task, the learning algorithm is given a set S of *training examples* that are items from $\mathcal{X}$ along with their correct classification.
- The main idea is generalization. That is, use one set of data to perform well on new data that the learning algorithm has not seen.

- One of the main tasks in Machine Learning is classification.
    - The goal is to *learn* a rule for labeling data (given a few labeled examples).
- The data comes from an instance space $\mathcal{X}$ and typically $\mathcal{X} = \mathbb{R}^d$ or $\mathcal{X} = \{0, 1\}^d$.
- So, a data item is typically described by a d-dimensional feature vector.
    - For example in spam classification, the features could be the presence (or absence) of certain words.
- For performing the learning task, the learning algorithm is given a set S of *training examples* that are items from $\mathcal{X}$ along with their correct classification.
- The main idea is generalization. That is, use one set of data to perform well on new data that the learning algorithm has not seen.
- The hope is that if the training data is representative of what the future data will look like, then we can try learning some simple rules that work for the training data and perhaps that will work well for the future data.

- Let us now try to formalize the ideas in the previous slide with respect to binary classification.
- Future data being representative of the training set:
  - There is a distribution D over the instance space $\mathcal{X}$.
  - Training set S consists of points drawn independently at random from D.
  - The new points are also drawn from D.
- A target concept w.r.t binary classification is simply a subset of $c^\star \subseteq \mathcal{X}$ denoting the positive data items of the classification task.
- The learning algorithm's goal is to produce a a set $h \subseteq \mathcal{X}$ called hypothesis that is close to $c^\star$ w.r.t. distribution D.
- The true error of hypothesis h is defined as $err_D(h) = \mathbf{Pr}[h \Delta c^\star]$, where Δ denotes symmetric difference and the probability is over the distribution D.
- The goal is to produce a hypothesis h with low true error.

- Let us now try to formalize the ideas in the previous slide with respect to binary classification.
- Future data being representative of the training set:
  - There is a distribution D over the instance space $\mathcal{X}$.
  - Training set S consists of points drawn independently at random from D.
  - The new points are also drawn from D.
- A target concept w.r.t binary classification is simply a subset of $c^\star \subseteq \mathcal{X}$ denoting the positive data items of the classification task.
- The learning algorithm's goal is to produce a a set $h \subseteq \mathcal{X}$ called hypothesis that is close to $c^\star$ w.r.t. distribution D.
- The true error of hypothesis h is defined as $err_D(h) = \mathbf{Pr}[h \Delta c^\star]$, where Δ denotes symmetric difference and the probability is over the distribution D.
- The goal is to produce a hypothesis h with low true error.
- The training error (or empirical error) of a hypothesis h is defined as $err_S(h) = \frac{|S \cap (h \Delta c^\star)|}{|S|}$.

- Let us now try to formalize the ideas in the previous slide with respect to binary classification.
- Future data being representative of the training set:
  - There is a distribution D over the instance space $\mathcal{X}$.
  - Training set S consists of points drawn independently at random from D.
  - The new points are also drawn from D.
- A target concept w.r.t binary classification is simply a subset of $c^\star \subseteq \mathcal{X}$ denoting the positive data items of the classification task.
- The learning algorithm's goal is to produce a a set $h \subseteq \mathcal{X}$ called hypothesis that is close to $c^\star$ w.r.t. distribution D.
- The true error of hypothesis h is defined as $err_D(h) = \mathbf{Pr}[h \Delta c^\star]$, where Δ denotes symmetric difference and the probability is over the distribution D.
- The goal is to produce a hypothesis h with low true error.
- The training error (or empirical error) of a hypothesis h is defined as $err_S(h) = \frac{|S \cap (h \Delta c^\star)|}{|S|}$.
- Question: Is it possible that the true error of a hypothesis is large but the training error is small?

- Let us now try to formalize the ideas in the previous slide with respect to binary classification.
- Future data being representative of the training set:
  - There is a distribution D over the instance space $\mathcal{X}$.
  - Training set S consists of points drawn independently at random from D.
  - The new points are also drawn from D.
- A target concept w.r.t binary classification is simply a subset of $c^{\star} \subseteq \mathcal{X}$ denoting the positive data items of the classification task.
- The learning algorithm's goal is to produce a a set $h \subseteq \mathcal{X}$ called hypothesis that is close to $c^{\star}$ w.r.t. distribution D.
- The true error of hypothesis h is defined as $err_D(h) = \mathbf{Pr}[h \Delta c^{\star}]$, where Δ denotes symmetric difference and the probability is over the distribution D.
- The goal is to produce a hypothesis h with low true error.
- The training error (or empirical error) of a hypothesis h is defined as $err_S(h) = \frac{|S \cap (h \Delta c^{\star})|}{|S|}$.
- Question: Is it possible that the true error of a hypothesis is large but the training error is small? Unlikely if S is sufficiently large

- Future data being representative of the training set:
  - There is a distribution D over the instance space $\mathcal{X}$.
  - Training set S consists of points drawn independently at random from D.
  - The new points are also drawn from D.
- A target concept w.r.t binary classification is simply a subset of $c^\star \subseteq \mathcal{X}$ denoting the positive data items of the classification task.
- The learning algorithm's goal is to produce a a set $h \subseteq \mathcal{X}$ called hypothesis that is close to $c^\star$ w.r.t. distribution D.
- The true error of hypothesis h is defined as $err_D(h) = \mathbf{Pr}[h \Delta c^\star]$, where Δ denotes symmetric difference and the probability is over the distribution D.
- The goal is to produce a hypothesis h with low true error.
- The training error (or empirical error) of a hypothesis h is defined as $err_S(h) = \frac{|S \cap (h \Delta c^\star)|}{|S|}$.
- Question: Is it possible that the true error of a hypothesis is large but the training error is small? Unlikely if S is sufficiently large
- Im many learning scenarios, a hypothesis is not an arbitrary subset of $\mathcal{X}$ but constrained to be a member of a hypothesis class (also called concept class) denoted by $\mathcal{H}$.
  - Consider example $\mathcal{X} = \{(-1,-1), (-1,1), (1,-1), (1,1)\}$ and $\mathcal{H}$ consists of all subsets that can be formed using a *linear separator*. What is $|\mathcal{H}|$?

- Future data being representative of the training set:
  - There is a distribution D over the instance space $\mathcal{X}$.
  - Training set S consists of points drawn independently at random from D.
  - The new points are also drawn from D.
- A target concept w.r.t binary classification is simply a subset of $c^\star \subseteq \mathcal{X}$ denoting the positive data items of the classification task.
- The learning algorithm's goal is to produce a a set $h \subseteq \mathcal{X}$ called hypothesis that is close to $c^\star$ w.r.t. distribution D.
- The true error of hypothesis h is defined as $err_D(h) = \mathbf{Pr}[h \Delta c^\star]$, where Δ denotes symmetric difference and the probability is over the distribution D.
- The goal is to produce a hypothesis h with low true error.
- The training error (or empirical error) of a hypothesis h is defined as $err_S(h) = \frac{|S \cap (h \Delta c^\star)|}{|S|}$.
- Question: Is it possible that the true error of a hypothesis is large but the training error is small? Unlikely if S is sufficiently large
- Im many learning scenarios, a hypothesis is not an arbitrary subset of $\mathcal{X}$ but constrained to be a member of a hypothesis class (also called concept class) denoted by $\mathcal{H}$.
- We would like to argue that for all $h \in \mathcal{H}$ the probability that there is a large gap between true error and training error is small.
  - Question: How large should S be the above to be true?

Theorem

Let $\mathcal{H}$ be a hypothesis class and let $\varepsilon, \delta > 0$. If a training set S of size

$$n \geq \frac{1}{\varepsilon}(\ln|\mathcal{H}| + \ln 1/\delta),$$

is drawn from distribution D, then with probability at least $(1-\delta)$ every $h \in \mathcal{H}$ with true error $err_D(h) \geq \varepsilon$ has training error $err_S(h) > 0$. Equivalently, with probability at least $(1-\delta)$, every $h \in \mathcal{H}$ with training error 0 has true error at most ε.

Theorem

Let $\mathcal{H}$ be a hypothesis class and let $\varepsilon, \delta > 0$. If a training set S of size

$$n \geq \frac{1}{\varepsilon}(\ln |\mathcal{H}| + \ln 1/\delta),$$

is drawn from distribution D, then with probability at least $(1-\delta)$ every $h \in \mathcal{H}$ with true error $err_D(h) \geq \varepsilon$ has training error $err_S(h) > 0$. Equivalently, with probability at least $(1-\delta)$, every $h \in \mathcal{H}$ with training error 0 has true error at most ε.

- The above result is called the PAC-learning guarantee since it states that if we can find an $h \in \mathcal{H}$ consistent with the sample, then this h is *Probably Approximately Correct*.
- What if we manage to find a hypothesis with small disagreement on the sample? Can we say that the hypothesis will have small true error?

Theorem

Let $\mathcal{H}$ be a hypothesis class and let $\varepsilon, \delta > 0$. If a training set S of size

$$n \geq \frac{1}{\varepsilon}(\ln |\mathcal{H}| + \ln (1/\delta)),$$

is drawn from distribution D, then with probability at least $(1 - \delta)$ every $h \in \mathcal{H}$ with true error $err_D(h) \geq \varepsilon$ has training error $err_S(h) > 0$. Equivalently, with probability at least $(1 - \delta)$, every $h \in \mathcal{H}$ with training error 0 has true error at most ε.

Theorem (Uniform convergence)

Let $\mathcal{H}$ be a hypothesis class and let $\varepsilon, \delta > 0$. If a training set S of size

$$n \geq \frac{1}{2\varepsilon^2}(\ln |\mathcal{H}| + \ln (2/\delta)),$$

is drawn from distribution D, then with probability at least $(1 - \delta)$ every $h \in \mathcal{H}$ satisfies $|err_D(h) - err_S(h)| \leq \varepsilon$.

Theorem

Let $\mathcal{H}$ be a hypothesis class and let $\varepsilon, \delta > 0$. If a training set S of size

$$n \geq \frac{1}{\varepsilon}(\ln |\mathcal{H}| + \ln (1/\delta)),$$

is drawn from distribution D, then with probability at least $(1 - \delta)$ every $h \in \mathcal{H}$ with true error $err_D(h) \geq \varepsilon$ has training error $err_S(h) > 0$. Equivalently, with probability at least $(1 - \delta)$, every $h \in \mathcal{H}$ with training error 0 has true error at most ε.

Theorem (Uniform convergence)

Let $\mathcal{H}$ be a hypothesis class and let $\varepsilon, \delta > 0$. If a training set S of size

$$n \geq \frac{1}{2\varepsilon^2}(\ln |\mathcal{H}| + \ln (2/\delta)),$$

is drawn from distribution D, then with probability at least $(1 - \delta)$ every $h \in \mathcal{H}$ satisfies $|err_D(h) - err_S(h)| \leq \varepsilon$.

- The above theorem essentially means that conditioned on S being sufficiently large, good performance on S will translate to good performance on D.

Theorem (Uniform convergence)

Let $\mathcal{H}$ be a hypothesis class and let $\varepsilon, \delta > 0$. If a training set S of size

$$n \geq \frac{1}{2\varepsilon^2}(\ln |\mathcal{H}| + \ln (2/\delta)),$$

is drawn from distribution D, then with probability at least $(1 - \delta)$ every $h \in \mathcal{H}$ satisfies $|err_D(h) - err_S(h)| \leq \varepsilon$.

- The above theorem follows from the following tail inequality.

Theorem (Chernoff-Hoeffding bound)

Let $x_1, ..., x_n$ be independent $\{0, 1\}$ random variables such that $\forall i, \mathbf{Pr}[x_i = 1] = p$. Let $s = \sum_{i=1}^{n} x_i$. For any $0 \leq \alpha \leq 1$,

$$\mathbf{Pr}[s/n > p + \alpha] \leq e^{-2n\alpha^2} \quad and \quad \mathbf{Pr}[s/n < p - \alpha] \leq e^{-2n\alpha^2}.$$

- Let us do a case study of *Learning Disjunctions*.
- Consider a binary classification context where the instance space $\mathcal{X} = \{0, 1\}^d$.
- Suppose we believe that the target concept is a disjunction over a subset of features. For example, $c^\star = \{x : x_1 \vee x_{10} \vee x_{50}\}$.
- What is the size of the concept class $\mathcal{H}$?

- Let us do a case study of *Learning Disjunctions*.
- Consider a binary classification context where the instance space $\mathcal{X} = \{0, 1\}^d$.
- Suppose we believe that the target concept is a disjunction over a subset of features. For example, $c^\star = \{x : x_1 \vee x_{10} \vee x_{50}\}$.
- What is the size of the concept class $\mathcal{H}$? $|\mathcal{H}| = 2^d$
- So, if the sample size $|S| = \frac{1}{\varepsilon}(d \ln 2 + \ln(1/\delta))$ then good performance on the training set generalizes to the instance space.
- Question: Suppose the target concept is indeed a disjunction, then given any training set S is there an algorithm that can at least output a disjunction consistent with S.

- Occam's razor: William of Occam around 1320AD stated that one should prefer simpler explanations over more complicated ones.

- Occam's razor: William of Occam around 1320AD stated that one should prefer simpler explanations over more complicated ones.
- What do we mean by a rule being simple?
- Different people may have different description languages for describing rules.
- How many rules can be described using fewer than b bits?

- Occam's razor: William of Occam around 1320AD stated that one should prefer simpler explanations over more complicated ones.
- What do we mean by a rule being simple?
- Different people may have different description languages for describing rules.
- How many rules can be described using fewer than b bits? $< 2^b$

Theorem (Occam's razor)

Fix any description language, and consider a training sample S drawn from distribution D. With probability at least $(1-\delta)$ any rule h consistent with S that can be described in this language using fewer than b bits will have $err_D(h) \leq \varepsilon$ for $|S| = \frac{1}{\varepsilon}(b \ln 2 + \ln(1/\delta))$. Equivalently, with probability at least $(1-\delta)$ all rules that can be described in fewer than b bits will have $err_D(h) \leq \frac{b \ln(2) + \ln(1/\delta)}{|S|}$.

Theorem (Occam's razor)

Fix any description language, and consider a training sample S drawn from distribution D. With probability at least $(1 - \delta)$ any rule h consistent with S that can be described in this language using fewer than b bits will have $err_D(h) \leq \varepsilon$ for $|S| = \frac{1}{\varepsilon}(b \ln 2 + \ln (1/\delta))$. Equivalently, with probability at least $(1 - \delta)$ all rules that can be described in fewer than b bits will have $err_D(h) \leq \frac{b \ln (2) + \ln (1/\delta)}{|S|}$.

- The theorem is valid irrespective of the description language.
- It does not say that complicated rules are bad.
- It suggests that Occam's rule is a good policy since simple rules are unlikely to fool us since there are not too many of them.

End