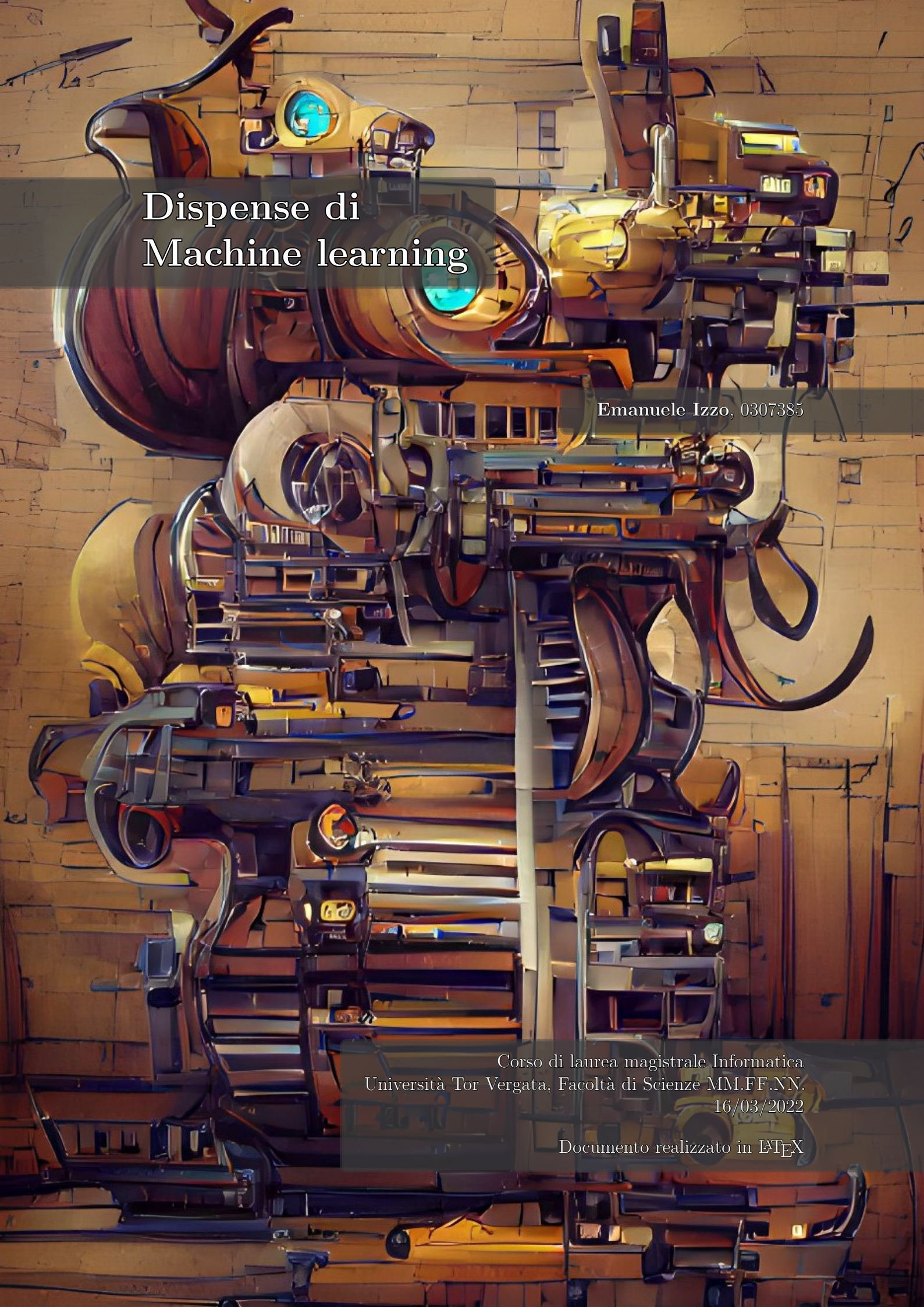

Dispense di Machine learning

Emanuele Izzo, 0307385

Corso di laurea magistrale Informatica
Università Tor Vergata, Facoltà di Scienze MM.FF.NN.
16/03/2022

Documento realizzato in LaTeX

Indice

Capitolo 1

Fondamenti del machine learning

Intuitivamente, l'obiettivo principale del **machine learning** è quello di cercare di apprendere **caratteristiche comuni** (o **patterns**) sulla base di **esempi** (ossia il **training set**), in modo tale da ricavare un **modello** che ci consenta di fare **predizioni**. Per predizione si intende l'inferenza di informazioni da dei dati in put, senza però che venga definito in maniera formale come estrapolare queste informazioni (dato che, se fosse possibile, verrebbe definito direttamente un algoritmo). Un training set è composto da n oggetti, ed è rappresentato come un insieme di vettori $x_1, x_2, ..., x_n \in \mathbb{D}^d$ (per qualche dominio $\mathbb{D}$ e una costante $d > 0$). Insieme agli n oggetti, il training set è provvisto da un insieme di valori $t_1, t_2, ..., t_n$, detti valori **target**, che indicano le informazioni *reali* che siamo interessati a predirre. Come già accennato, il training set è utilizzato per derivare un **modello di previsione** in grado di stimare il più precisamente possibile i target reali, il quale dipenderà fortemente dal training set a disposizione. Lo scopo finale è quello di realizzare un modello in grado di predirre il target con le migliori prestazioni possibili, sia in termini di *efficenza* che di *correttezza*.

1.1 Tipologie di problemi

1.1.1 Apprendimento supervisionato

Nell'**apprendimento supervisionato** vogliamo predirre, dati i valori di un set (**features**) di un oggetto x, il valore sconosciuto associato ad x detto **target**. È importante distinguere due casi quando definiamo il target:

- Se il target è in $\mathbb{R}$, allora siamo interessati ad una **regressione**.
- Se il target è in un insieme $\{1, 2, ..., K\}$, allora siamo interessati ad una **classificazione**.

L'approccio generico si basa sul definire (attraverso l'apprendimento da un insieme di esempi) un **modello** della relazione tra i valori delle feature e quelli del target. Il training set (X, t) include un vettore delle feature $x_i = \{x_{i_1}, x_{i_2}, ..., x_{i_d}\}$ e il corrispettivo target t_i per ogni oggetto i. Il modello può essere di due tipi:

- Può essere una funzione $y()$, la quale per ogni oggetto x ritorna un valore $y(x)$ come stima di t
- Può essere una distribuzione di probabilità che associa ad ogni possibile valore $\bar{y}$ nel dominio target la corrispettiva probabilità $P(y = \bar{y}|x)$.

1.1.2 Apprendimento non supervisionato

Nell'**apprendimento non supervisionato** vogliamo estrarre, data una collezione di oggetti (**dataset**) $X := \{x_1, x_2, ..., x_n\}$ senza target associati, alcune informazioni sintetiche, come:

- Un sottoinsieme di oggetti simili (*cluseting*).
- La distribuzione degli oggetti nel loro dominio (*density estimation*).
- La proiezione, più informativa possibile, degli oggetti in sottospazi dimensionali inferiori, cioè la loro caratterizzazione per mezzo di un insieme più piccolo di features (*feature selection*, *feature extraction*).

Un **modello** adatto, delle sole feature dei dati, viene di solito definito e applicato anche nel caso dell'apprendimento non supervisionato.

1.1.3 Apprendimento per rinforzo

Nell'**apprendimento per rinforzo** vogliamo identificare, in un dato framework, una sequenza di azioni da eseguire in modo da massimizzare un certo profitto. Rispetto all'apprendimento supervisionato, non vengono forniti esempi, ma è disponibile un ambiente il quale ritorna un profitto in base all'esecuzioni di una qualsiasi azione.

1.2 I framework del machine learning

1.2.1 Domain set

Il **domain set** $\mathcal{X}$ è un set di oggetti che vogliamo etichettare. Ogni oggetto è modellato come un vettore di **features**. Il numero di features è la **dimensionalità** del problema

1.2.2 Label set

Il **label set** $\mathcal{Y}$ è il set di tutti i possibili valori delle etichette associate agli oggett in $\mathcal{X}$. Bisogna notare che:

- Se $\mathcal{Y}$ è continuo, allora stiamo lavorando ad un problema di **regressione**.
- Se $\mathcal{Y}$ è discreto, allora stiamo lavorando ad un problema di **classificazione**.

1.2.3 Training set

Il **training set** $\mathcal{T}$ è un set di coppie oggetti-etichette definito nel seguente modo: $\mathcal{T} := \{((x_1, t_1), (x_2, t_2), ..., (x_n, t_n))\}$. Di solito si denota con X la matrice degli oggetti (**matrice delle feature**):

$$\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix}$$

E con t il vettore delle etichette (**vettore target**):

$$\begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_n \end{pmatrix}$$

1.2.4 Learner

Il **learner** (un algoritmo A) è richiesto per restituire, dato un certo training set $\mathcal{T}$, una **regola di predizione** (*classificazione*, *regressione*) $A(\mathcal{T}) = h : \mathcal{X} \mapsto \mathcal{Y}$

1.2.5 Modello di generazione del training set

Si assume che ogni oggetto del training set $\mathcal{T}$ è campionato dal dominio $\mathcal{X}$ in acordo ad una distribuzione $\mathcal{D}_1$. Perciò, pero ogni elemento $x \in \mathcal{X}$, avremo che $p_{\mathcal{D}_1}(x)$ è la probabilità che x sia presente nell'insieme $\mathcal{T}$. Analogamente assumiamo anche che i rispettivi vvalori target degli elementi in $\mathcal{T}$ sono campionati in accordo ad una distribuzione $\mathcal{D}_2$ *condizionata* sugli elementi di $mathcalX$. Perciò, per ogni $t \in \mathcal{Y}$ avremo che $p_{\mathcal{D}_2}(t|x)$ è la probabilità che osservando il valore di target di x nel training set $\mathcal{T}$, esso sia pari a t.

1.2.6 Predizione del rischio

Dato un qualsiasi elemento $x \in \mathcal{T}$:

- L'**errore** di un predittore h deriva dalla comparazione della sua predizione $h(x)$ e la corretta etichetta target y.
- La **loss** è la comparazione effettuata aplicando una **funzione di loss** predefinita:

$$L : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$$

- Il **rischio di una predizione** permette di definire l'errore di una predizione $\hat{y}$ applicando la funzione di loss:

$$\mathcal{R}(\hat{y}, y) = \mathcal{L}(h(x), y)$$

  Nel caso generale quando viene assunta una relazione probabilistica $p_{\mathcal{D}_2}(\hat{y}|x)$ tra le etichette e i target, questa corrisponde nel caso della regressione a:

$$\mathcal{R}(\hat{y}, y) = \mathbb{E}_{\mathcal{D}_2}[L(\hat{y}, y)] = \int_{\mathcal{Y}} L(\hat{y}, y) \cdot p_{\mathcal{D}_2}(y|x) dy$$

  O, nel caso della classificazione a:

$$\mathcal{R}(\hat{y}, y) = \mathbb{E}_{\mathcal{D}_2}[L(\hat{y}, y)] = \sum_{\mathcal{Y}} L(\hat{y}, y) \cdot p_{\mathcal{D}_2}(y|x)$$

  In questo framework, la predizione ottimale è quella che minimizza il rischio:

$$y^*(x) = \text{argmin}_{\hat{Y}}\, \mathcal{R}(\hat{y}, x) = \text{argmin}_{\hat{Y}}\, \mathbb{E}_{\mathcal{D}_2}[L(\hat{y}, y)]$$

Ossia:

$$y^*(x) = \text{argmin}_{\hat{Y}}\, \mathbb{E}_{\mathcal{D}_2}[L(\hat{y}, f(x))] \qquad \text{nel caso più semplice}$$

$$y^*(x) = \text{argmin}_{\hat{Y}} \int_{\mathcal{Y}} L(\hat{y}, y) \cdot -p_{\mathcal{D}_2} p(y|x) dy \qquad \text{nel caso generale}$$

Nel caso generale, questo è indicato come **stimatore di Bayes**. Noitare però che questo approccio non può essere applicato siccome siccome sia la funzione f e la distribuzione $\mathcal{D}_2$ di $p(y|x)$ sono assunte come sconosciute.

1.2.7 Valutazione del learner

Poiché $\mathcal{D}_1$ e $\mathcal{D}_2$ (o f) sono sconosciute, il rischio può essere stimato dai dati disponibili (ossia il training set $\mathcal{T}$); ciò può essere fatto tramite il **rischio empirico**, che viene ottenuto tramite una media artimentica rispetto alle sole informazioni di $\mathcal{T}$:

$$\bar{\mathcal{R}}_{\mathcal{T}}(h) = \frac{1}{|\mathcal{T}|} \sum_{(x,t)\in\mathcal{T}} L(h(x), t)$$

L'approccio fondamentale nel machine learning è derivare un predittore h che minimizza (almeno approssimativamente) il rischio empirico calcolato tramite il training set disponibile. Un problema di apprendimento è allora ridotto ad un problema di minimizzazione in un qualche spazio funzionale $\mathcal{H}$, l'insieme di tutti i possibili predittori h:

$$h^* = \text{argmin}_{h\in\mathcal{H}}\, \bar{\mathcal{R}}_{\mathcal{T}}(h)$$

Qui, $\mathcal{H}$ è l'insieme di **ipotesi** o **bias induttivi**.

1.2.8 Hypotesis set

L'**hypotesis set** è l'insieme $\mathcal{H}$ formato da tutte le possibili ipotesi considerate per la ricerca del miglior predittore h^*. La scelta dell'insieme di ipotesi è un problema importante nel machine learning:

- Qual è l'effetto della struttura e grandezza di $\mathcal{H}$?
- Come definire $\mathcal{H}$ in modo da rendere fattibile il calcolo di h^*?

Osserviamo che non possiamo scegliere $\mathcal{H} = \mathcal{Y}^{\mathcal{X}}$ come l'insieme di tutte le possibili funzioni $f : \mathcal{X} \to \mathcal{Y}$, in quanto esso crece in grandezza come $|\mathcal{Y}|^{|\mathcal{X}|}$. Ciò che si fa generalmente è scegliere la classe $\mathcal{H}$ sulla base di *conoscenze pregresse* che il learner possiede rispetto al task. A volte, invece, si sceglie la classe $\mathcal{H}$ in maniera *sperimentale/empirica*, proseguendo per tentativi.

Problema con $\mathcal{H}$ grande

Supponiamo di avere larga libertà nel scegliere $\mathcal{H}$. Supponiamo di voler fare una *classificazione binaria* con training set $\mathcal{T} = (X, t)$ e con una funzione di loss *binaria*:

$$\forall y \in \mathcal{Y}, L(y, t) = \begin{cases} 0 & \text{se } y = t \\ 1 & \text{se } y \neq t \end{cases}$$

Assumiamo inoltre che la distribuzione del target sia *uniforme*, ovvero che le due classi nella popolazione hanno più o meno la stessa dimensione:

$$\forall x \in \mathcal{X}, P(t_x = 1|x) = P(t_x = 0|x) = \frac{1}{2}$$

Un classificatore h banale è il seguente:

$$\forall x \in \mathcal{X}, h(x) = \begin{cases} 1 & \text{se } x = x_1 \in X, t_i = 1 \\ 0 & \text{altrimenti} \end{cases}$$

Oero h risponde correttamente per tutte le x che appartengono al training set, 0 in qualsiasi altro caso. Dal punto di ista del training set $\mathcal{T}$ il predittore h è ottimo (in quanto risponde correttamente a tutti i alori di X), e infatti il rischio empirico risultante è 0. Quando invece applichiamo h su un campione casuale della popolazione, h sbaglierà praticamente tutti gli oggetti con etichetta 1 che non sono in $\mathcal{T}$. Dato che abbiamo detto che le due classi sono equamente distribuite nella popolazione, avremo che il rischio di h è circa $\cong \frac{1}{2}$. Quando il predittore h dipende troppo dal training set, ovvero è molto preciso per $\mathcal{T}$ ma si comporta male per la popolazione $\mathcal{X}$ in generale, si parla del fenomeno dell'**overfitting**. Il problema che ci ha indotto a trovare una funzione h^* ottima per $\mathcal{T}$ ma pessima per $\mathcal{X}$ è stato appunto dare *troppa libertà* alla caratterizzazione di $\mathcal{H}$. Infatti se lo spazio delle ipotesi $\mathcal{H}$ è *troppo vasto*, allora il rischio di trovare una funzione h^* ottima empiricamente ma che globalmente non va bene è alto. Viceversa, però, potrebbe accadere una situazione inversa: quando il predittore h è poco preciso anche per $\mathcal{T}$ si parla del fenomeno dell'**underfitting**. Perciò possiamo fare le seguenti osservazioni rispetto all'insieme delle ipotesi $\mathcal{H}$:

- Se $\mathcal{H}$ è troppo vasto (o *complesso*), potremmo andare in **overfitting**, ovvero potremmo trovare un h^* *troppo specifica* per $\mathcal{T}$, ma poco precisa rispetto a $\mathcal{H}$ in generale.
- Se $\mathcal{H}$ è troppo ristretto (o *semplice*), potremmo andare in **underfitting**, ovvero potrebbero non esistere funzioni h^* vadano mediamente bene né per $\mathcal{T}$ né per $\mathcal{X}$.

Bias e varianza

Il rischio associato ad h^*, ossia al predittore che minimizza il rischio empirico, può essere decomposto in come $\mathcal{R}(h^*) = \epsilon_B + \epsilon_V$, dove:

- ϵ_B, chiamato **bias**, è il rischio minimo ottenibile da una qualsiasi $h \in \mathcal{H}$: questo valore è determinato dai bias induttivi, ed è indipendente dal training set

Capitolo 2

Apprendimento probabilistico

Come fatto prima, assumiamo che il dataset osservato sia stato derivato tramite un campionamento randomico:

- $\mathcal{X}$ in accordo alla distribuzione di probabilità $p_{\mathcal{D}_1}(x)$ (di solito è usata una distribuzione uniforme).
- $\mathcal{Y}$ in accordo distribuzione condizionata $p_{\mathcal{D}_2}(y|x)$.

Vorremmo poi considerare una classe di possibili distribuzioni condizionate $\mathcal{P}$ e selezionare (inferire) la "miglior" distribuzione condizionale $p^* \in \mathcal{P}$ tramite la conoscenza disponibile (ossia il dataset) in accordo ad una certa misura q. Dato un qualsiasi nuovo oggetto x, si applica $p^*(y|x)$ per assegnare una probabilità a ciascuno dei possibili valori del corrispettivo target, e tramite l'applicazione di una **strategia di decisione** indipendente a $p^*(y|x)$ si restituisce una specifica predizione $h(x)$. Per definire la classe delle possibili probabilità condizionate $p(y|x)$ si usa solitamente un approccio parametrico: la distribuzione è definita tramite un struttura arbitraria e un set di parametri. La misura della qualità $q(p, \mathcal{F})$ della distribuzione (dato il dataset $\mathcal{T} = (X, t)$) dipende da come il dataset è stato generato usando il campionamento casuale tramite $mathcal D_1$ (usualmente uniforme) e $\mathcal{D}_2$ potrebbe essere simile al dataset $\mathcal{T}$ disponibile. Invece di trovare la distribuzione migliore $p^* \in \mathcal{P}$ e usarla per predirre le probabilità target come $p^*(y|x)$ per ogni elemento x, possiamo considerare per ogni possibile distribuzione condizionale $p \in \mathcal{P}$ la sua qualità $q(p, \mathcal{T})$, comporre tutte le distribuzioni condizionali $p(y|x)$, ognuna pesata dalla sua qualità $q(p, \mathcal{T})$ (per esempio tramite una media ponderata) e applicare la distribuzione risultante.