Corso di Bioinformatica

Machine Learning: apprendimento, generalizzazione e stima dell'errore di generalizzazione

Giorgio Valentini

DI – Università degli Studi di Milano

Metodi di machine learning

- I metodi di apprendimento automatico (Machine Learning - ML) consentono di effettuare inferenze e predizioni da un insieme limitato di dati disponibili
- Gli algoritmi di ML “apprendono” una funzione predittiva utilizzando una serie di esempi (dati) estratti secondo una determinata distribuzione di probabilità dall’ “Universo” dei dati.

Apprendimento induttivo

- Insieme finito di oggetti: $S = \{x_1, x_2, \ldots, x_n\}, S \subset U$
  - **Obiettivo**: apprendere una proprietà *F* di *U* (universo dei dati) dall'analisi di *S* tramite un algoritmo di apprendimento *A.*
  - **Es. 1**: Dato un insieme *S* sottoinsieme di *U* (universo delle proteine) predire per un *x* di *U* se *x* ha struttura secondaria α o β, utilizzando un algoritmo *A(S).*
  - **Es. 2**: Dato un insieme *S* sottoinsieme di *U* (universo dei pazienti) predire per un *x* di *U* se è sano o malato, utilizzando un algoritmo *A(S).*

Principali tipologie di problemi di ML

1. Problemi supervisionati:
   Dato *X* sottoinsieme di *U* (insieme di oggetti) e *l(X)* (etichette), predire F(*x*) per ogni *x* appartenente a *U*.
   L'algoritmo *A* utilizza *X* per costruire un predittore *f* che approssimi *F* (non nota a priori).
2. Problemi non supervisionati:
   Dati *X* (insieme di oggetti) con *l(X)* ignota, predire direttamente F*(x).*
3. Problemi semisupervisionati:
   Dato X=X'UX'' tale che l(X') e' nota e l(X'') e' ignota, predire F(*x*)

Principali problemi supervisionati

1. Problemi di classificazione: *F(x)* è una funzione a valori discreti
2. Problemi di regressione: *F(x)* è una funzione a valori continui
3. Problemi di stima della densità di probabilità *F(x)*

Apprendimento supervisionato

Apprendimento da dati "etichettati": ciascun campione viene etichettato (ad es: normale o malato)

Supervisionato in quanto un "supervisore" assegna le etichette ai campioni da apprendere: cioè la learning machine è addestrata tramite un insieme di dati etichettati (**x**,*t*)

La learning machine impara ad associare un determinato campione **x** ad una classe *t*

L' obiettivo della learning machine consiste nell' *assegnare un' etichetta corretta a campioni la cui classe di appartenenza non è nota a priori* (ad es: deve essere in grado di predire sulla base dei dati di espressione genica se un paziente sia sano o malato)

Obiettivo dell' apprendimento supervisionato

Consiste nel predire correttamente campioni la cui classe di appartenenza non è nota priori (*generalizzazione*).

La generalizzazione dipende da:

- Accuratezza del classificatore sul training set
- Complessità della funzione computata dal classificatore
- Dimensione training set

In caso i dati siano caratterizzati da ridotta cardinalità e/o elevata dimensionalità, può sorgere il problema di *overfitting* (sovraadattamento)

Generalizzazione

L viene addestrata su un *training set* $D \subset U$.

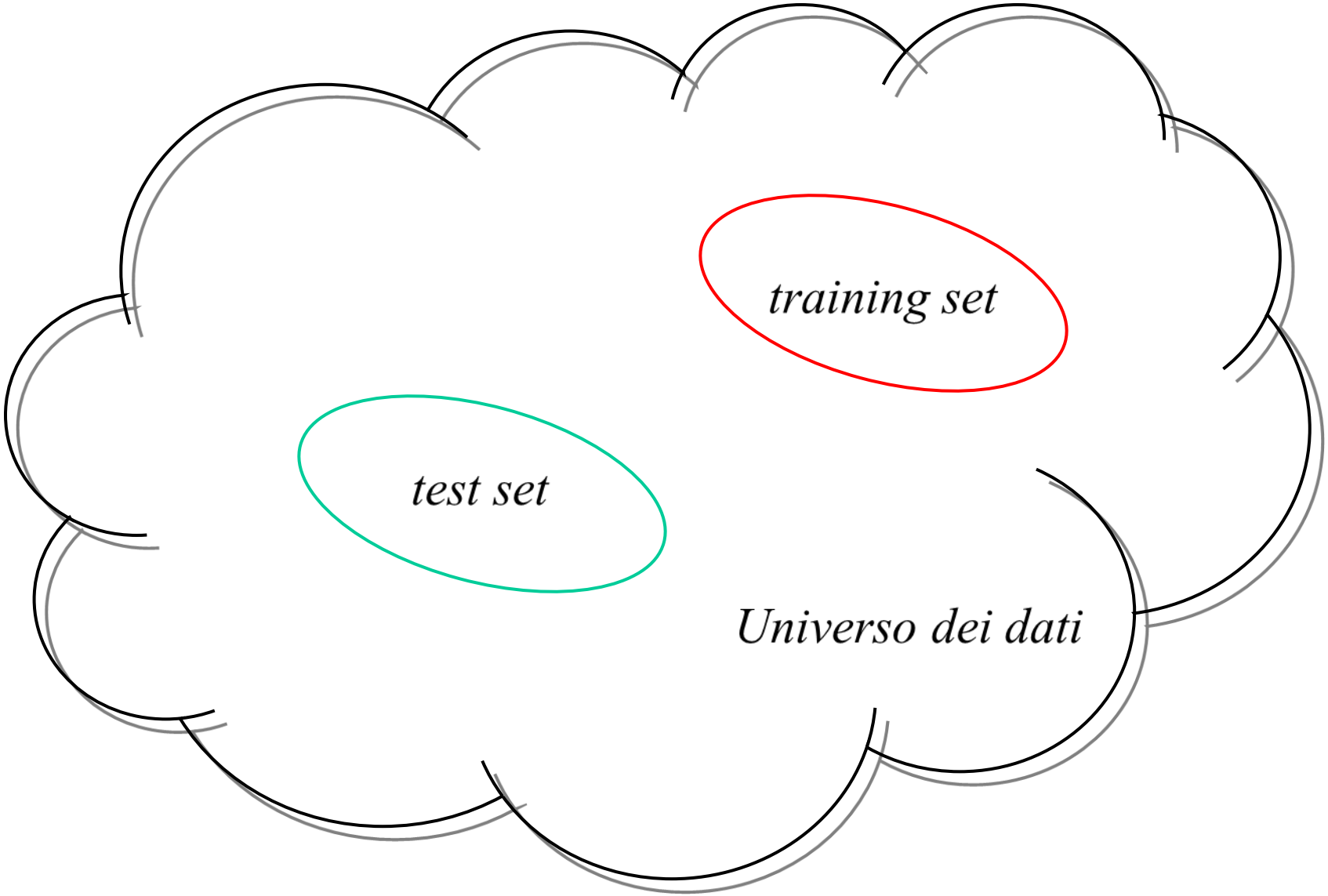
La learning machine *L* è utile se può fare delle previsioni sull' Universo non noto *U* dei dati:

vogliamo cioè che *generalizzi* correttamente su dati che non conosce.

A questo fine *L* deve prevedere correttamente non tanto i dati *D* su cui è stata addestrata, ma i dati $(\mathbf{x},t) \in U$, $(\mathbf{x},t) \notin D$ che non conosce.

Poichè di solito non si conosce a priori *U* o equivalentemente la distribuzione di probabilità congiunta $P_U(\mathbf{x},t)$, le capacità di generalizzazione di *L* vengono valutate rispetto ad un test set *T* separato da *D*, cioè tale che $T \subset U$ e $T \cap D = \emptyset$.

Universo dei dati e campioni



Rischio atteso e rischio empirico

L` apprendimento di una funzione non nota $f : R^d \rightarrow C$ avviene tramite un algoritmo *L* che genera un insieme di funzioni *g* che approssimano *f* utilizzando solo un training set $D = \{(\mathbf{x}_i, t_i)\}_{i=1}^{n}$ distribuito secondo un distribuzione di probabilità non nota *P*(**x**,t) :

$$g : \Re^d \times \Omega \rightarrow C$$

Ω rappresenta un insieme di parametri della learning machine (ad es., l' insieme dei pesi delle unità di calcolo di una rete neurale).

• Obiettivo dell' apprendimento non è minimizzare il rischio empirico $R_{emp}(\omega)$:

$$R_{emp}(\omega) = \frac{1}{n}\sum_{i=1}^{n} Loss\big(g(\mathbf{x}_i, \omega), t_i\big)$$

bensì il rischio atteso R(ω):

$$R(\omega) = \iint Loss\big(g(\mathbf{x}, \omega), t\big)\, p(\mathbf{x}, t)\, d\mathbf{x}\, dt$$

A parte le difficoltà matematiche della minimizzazione del funzionale R(ω), quasi sempre la funzione di densità di probabilità congiunta non è nota ...

Stima del rischio atteso

Il rischio empirico non sempre converge al rischio atteso.

La *Teoria Statistica dell' Apprendimento di Vapnik* ha mostrato che un limite superiore al rischio atteso può essere scomposto in due componenti:

$$R(\omega) \leq R_{emp}(\omega) + \Phi(h/m)$$

dove il primo termine dipende dal rischio empirico, mentre l' intervallo di confidenza Φ dipende principalmente dal rapporto fra la complessità h della learning machine e la cardinalità m del training set disponibile.

• Per valutare le capacità di generalizzazione delle learning machine è necessario stimare il rischio atteso e non semplicemente il rischio empirico.

• Il problema è: come stimare il rischio atteso ?

Due approcci principali alla stima del rischio atteso

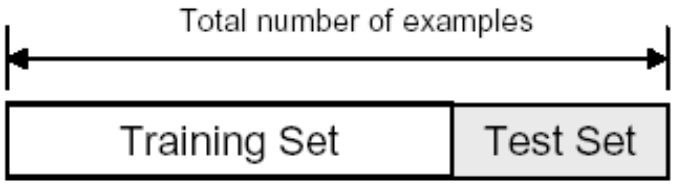
1. Stima teorica dei limiti superiori al rischio atteso (basati sull' errore empirico e sulla stima della complessità della learning machine)
2. Stima sperimentale (basata sul campionamento dei dati disponibili)

Metodi di stima sperimentale dell' errore di generalizzazione

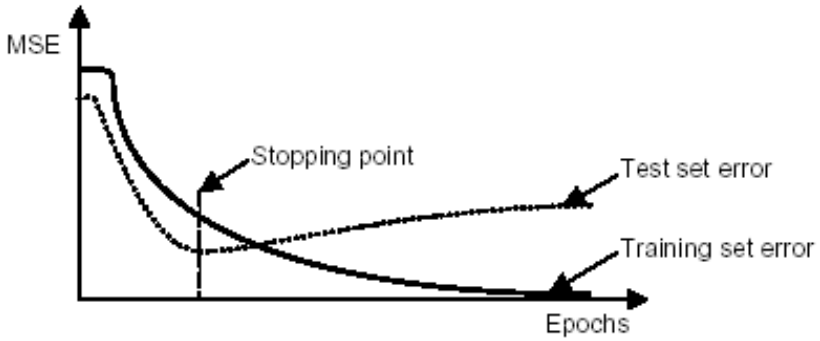
- Holdout
  - Suddivisione dei dati in *training* e *test* set (tipicamente 2/3 ed 1/3)
- Holdout multiplo
  - Holdout ripetuto n volte
- Cross validation
  - Partizione dei dati in k sottoinsiemi disgiunti (fold)
  - k-fold: training con k-1 fold, test sul rimanente; il processo è ripetuto k volte utilizzando ognimvolta come test set un fold differente.
  - Leave-one-out: k = numero dei campioni disponibili
- Bootstrap
  - Campionamento con rimpiazzo
- Metodi out-of-bag
  - Training sui campioni estratti tramite bootstrap e testing sui rimanenti campioni non selezionati. Il processo è ripetuto n volte.

Metodo di hold-out (1)

- **Split dataset into two groups**
  - Training set: used to train the classifier
  - Test set: used to estimate the error rate of the trained classifier



- **A typical application the holdout method is determining a stopping point for the back propagation error**

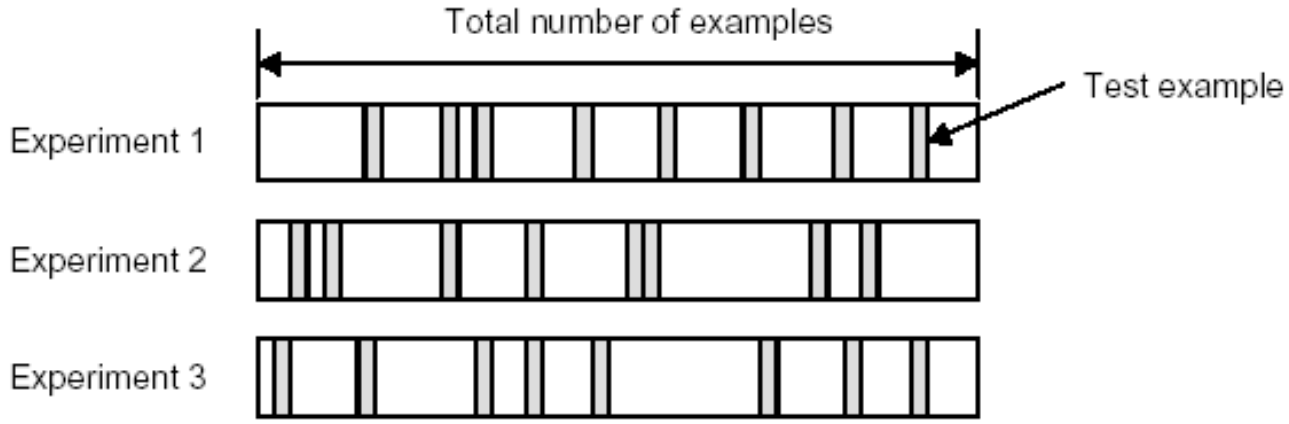


Metodo di hold-out (2)

- **The holdout method has two basic drawbacks**
  - In problems where we have a sparse dataset we may not be able to afford the “luxury” of setting aside a portion of the dataset for testing
  - Since it is a single train-and-test experiment, the holdout estimate of error rate will be misleading if we happen to get an “unfortunate” split
- **The limitations of the holdout can be overcome with a family of resampling methods at the expense of higher computational cost**
  - Cross Validation
    - Random Subsampling
    - K-Fold Cross-Validation
    - Leave-one-out Cross-Validation
  - Bootstrap

Holdout ripetuto

- **Random Subsampling performs K data splits of the entire dataset**
  - Each data split randomly selects a (fixed) number of examples without replacement
  - For each data split we retrain the classifier from scratch with the training examples and then estimate E_i with the test examples

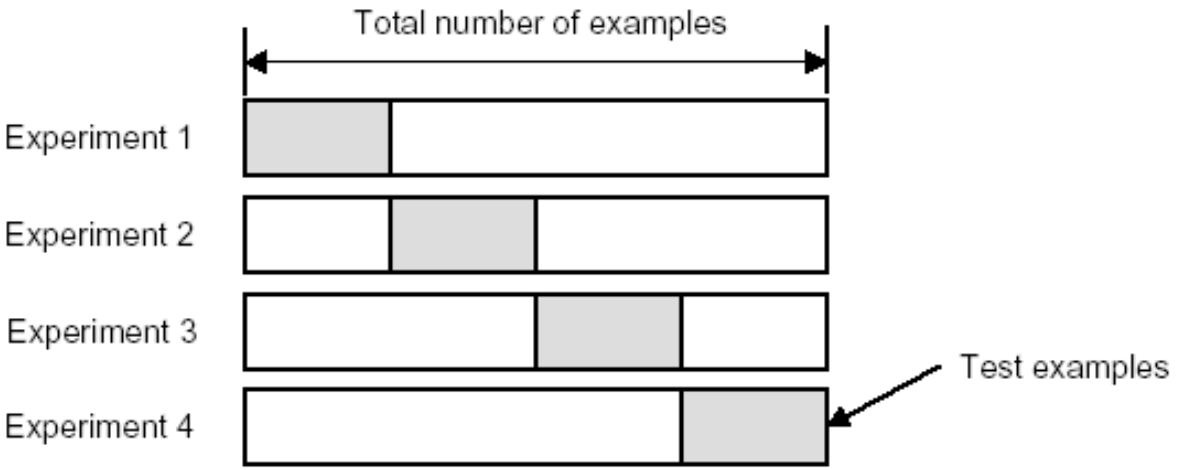


- **The true error estimate is obtained as the average of the separate estimates E_i**
  - This estimate is significantly better than the holdout estimate

$$E = \frac{1}{K}\sum_{i=1}^{K} E_i$$

K-fold cross validation

- **Create a K-fold partition of the the dataset**
  - For each of K experiments, use K-1 folds for training and a different fold for testing
  - This procedure is illustrated in the following diagram for K=4

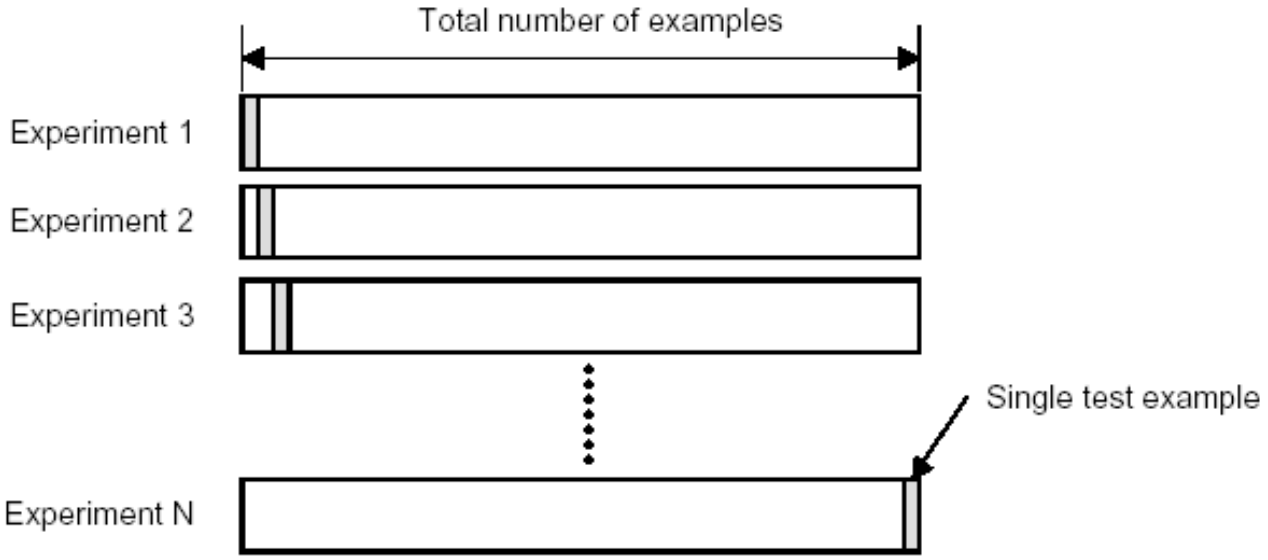


- **K-Fold Cross validation is similar to Random Subsampling**
  - The advantage of K-Fold Cross validation is that all the examples in the dataset are eventually used for both training and testing
- **As before, the true error is estimated as the average error rate on test examples**

$$E = \frac{1}{K}\sum_{i=1}^{K} E_i$$

Leave-one-out

- **Leave-one-out is the degenerate case of K-Fold Cross Validation, where K is chosen as the total number of examples**
  - For a dataset with N examples, perform N experiments
  - For each experiment use N-1 examples for training and the remaining example for testing



- **As usual, the true error is estimated as the average error rate on test examples**

$$E = \frac{1}{N}\sum_{i=1}^{N} E_i$$

Sintesi della lezione

- Tipologie di metodi di apprendimento automatico (Machine Learning - ML)
- Apprendimento e generalizzazione
- Rischio empirico e rischio atteso
- Stima sperimentale del rischio atteso (errore di generalizzazione)