

# <Ottimizzatore Energetico per Condomini>

Sviluppato da:

- D'Ingeo Antonio, matricola: 798436, [a.dingeo2@studenti.uniba.it](mailto:a.dingeo2@studenti.uniba.it)

<<https://github.com/antonio19-tech/OttimizzatoreEnergiaCondominio.git>>

AA 2025-26

## Introduzione

Il progetto affronta il problema dell'**ottimizzazione energetica di un condominio**, con particolare riferimento alla gestione del riscaldamento in presenza di **incertezza sull'occupazione, variabilità ambientale e vincoli economici**.

L'obiettivo è decidere **quando e quanto attivare il riscaldamento** nei diversi appartamenti, bilanciando:

- comfort degli occupanti,
- consumo energetico,
- rischio decisionale,
- budget globale disponibile.

Il sistema è progettato come un **Knowledge-Based System (KBS)** modulare, che integra tecniche di **apprendimento automatico supervisionato, ragionamento sotto incertezza e ottimizzazione matematica**.

## Sommario

Il sistema integra tre moduli principali, ciascuno collegato a specifiche parti del programma del corso:

1. **Apprendimento automatico probabilistico**  
(reti neurali, softmax, classificazione probabilistica)
2. **Ragionamento sotto incertezza e utilità attesa**  
(teoria delle decisioni)
3. **Decisione ottima tramite Programmazione Lineare**  
(ottimizzazione vincolata)

Il flusso è il seguente:

**Dati sensoriali → Rete neurale → Distribuzione di probabilità → Utilità attesa → Programmazione Lineare → Piano energetico ottimo**

## Elenco argomenti di interesse

- Apprendimento Automatico Supervisionato (sezione: Reti Neurali / Machine Learning)
- Incertezza e Utilità Attesa (sezione: Ragionamento sotto incertezza / Decisione razionale)
- Programmazione Lineare (sezione: Ottimizzazione e Ricerca Operativa)

## Sezione 1 - Apprendimento Automatico

### Sommario

L'apprendimento automatico è utilizzato per **stimare probabilisticamente lo stato di occupazione** di ciascun appartamento.

Lo stato è modellato come una variabile discreta con tre valori:

- Away
- Home
- Sleep

L'output del modello non è una decisione deterministica, ma una **distribuzione di probabilità**  $P(\text{stato} | \text{evidenze})$ .

### Strumenti utilizzati

- Rete neurale feed-forward implementata in C
- Funzione di attivazione ReLU
- Softmax sull'output
- Loss: Cross-Entropy

Il modello implementa un **classificatore probabilistico**

### Decisioni di Progetto

- Input: 7 feature normalizzate  
(ora, temperatura esterna, luci, movimento, consumo, prezzo, temperatura interna)
- Hidden layer: 16 neuroni
- Output: 3 neuroni (Away, Home, Sleep)
- Learning rate: 0.01
- Regolarizzazione L2: 0.001
- Training supervisionato su dataset caricato da file CSV

La scelta della softmax consente di ottenere **probabilità interpretabili**, necessarie per il modulo successivo.

## Valutazione

La valutazione del modulo di apprendimento automatico non è stata effettuata tramite metriche di classificazione deterministiche (accuracy, precision, recall), in quanto l'obiettivo del sistema non è l'assegnazione rigida di uno stato, ma la stima di una **distribuzione di probabilità sugli stati di occupazione**.

Durante la fase di addestramento è stata utilizzata la **cross-entropy loss**, metrica standard per classificatori probabilistici con funzione softmax, che misura la distanza tra la distribuzione stimata dal modello e quella reale associata ai dati supervisionati.

In fase di validazione, l'apprendimento è stato valutato in modo **qualitativo e funzionale**, analizzando la coerenza delle probabilità prodotte rispetto agli scenari osservati e al loro utilizzo nei moduli decisionali successivi.

Paragrafi che richiamino (non spieghino, se standard) le metriche adottate  
+ tabelle sui risultati e loro discussione

```
---- ANALISI AGENTE INTELLIGENTE ----

Appartamento 1:
ORA[19:00] T_EXT[6°] T_INT[16°] LUCI[0.8] MOVIMENTO[0.3]->
P(Away): 0.00 | P(Home): 1.00 | P(Sleep): 0.00 | EU Totale: 1.699

Appartamento 2:
ORA[19:00] T_EXT[6°] T_INT[17°] LUCI[0.7] MOVIMENTO[0.5]->
P(Away): 0.00 | P(Home): 1.00 | P(Sleep): 0.00 | EU Totale: 1.699

Appartamento 3:
ORA[19:00] T_EXT[5°] T_INT[16°] LUCI[0.0] MOVIMENTO[0.0]->
P(Away): 0.84 | P(Home): 0.00 | P(Sleep): 0.16 | EU Totale: 0.032

Appartamento 4:
ORA[19:00] T_EXT[5°] T_INT[18°] LUCI[0.5] MOVIMENTO[0.6]->
P(Away): 0.00 | P(Home): 0.84 | P(Sleep): 0.16 | EU Totale: 1.466
```

## Discussione dei risultati

I risultati mostrano che la rete neurale apprende relazioni coerenti tra le evidenze ambientali e lo stato di occupazione:

- in presenza di **luci accese, movimento rilevato e temperature interne basse**, la probabilità associata allo stato *Home* risulta dominante;
- in assenza di segnali di presenza, lo stato *Away* viene correttamente favorito;
- lo stato *Sleep* emerge come stato intermedio in condizioni di attività ridotta, tipiche di fasce serali o notturne.

La produzione di distribuzioni di probabilità ben calibrate consente al sistema di propagare l'incertezza verso il modulo di **utilità attesa**, rendendo l'apprendimento automatico funzionale al processo decisionale basato su utilità attesa.

## Sezione 2 - Incertezza e Utilità Attesa

### Sommario

Il modulo `Incertezza` implementa il **ragionamento decisionale sotto incertezza**, utilizzando il principio dell'**utilità attesa**:

$$EU = \sum_s P(s) \cdot U(s)$$

dove:

- $P(s)$  è fornita dalla rete neurale e rappresenta la **distribuzione di probabilità sugli stati di occupazione** (`Away`, `Home`, `Sleep`), stimata dalla rete neurale a partire dalle evidenze ambientali (ora, temperatura esterna, luci, movimento, consumo, prezzo, temperatura interna), interpretabile come probabilità a posteriori dello stato di occupazione.  
Essa esprime il grado di incertezza del sistema rispetto allo stato reale dell'appartamento.
- $U(s)$  è una funzione di utilità definita dal dominio **applicativo**, che associa a ciascuno stato di occupazione un valore numerico proporzionale al beneficio o alla penalità derivante dall'attivazione del riscaldamento, in funzione delle condizioni interne ed esterne.

### Strumenti utilizzati

- Funzioni di utilità definite manualmente sulla base del dominio applicativo;
- Modulo C dedicato alla gestione dell'incertezza (`Incertezza.c` / `Incertezza.h`);
- Principi di **teoria delle decisioni**, in particolare il criterio dell'utilità attesa.

### Decisioni di Progetto

La funzione di utilità considera:

- stato di occupazione,
- temperatura interna,
- temperatura esterna.

Esempi di modellazione dell'utilità:

- penalità per riscaldamento durante `Sleep` se troppo caldo,
- bonus per comfort in `Home` con temperatura bassa,
- incentivo al risparmio in `Away`.

Questo consente un comportamento **contestuale e spiegabile**.

## Valutazione

```
--- ANALISI AGENTE INTELLIGENTE ---  
  
Appartamento 1:  
ORA[19:00] T_EXT[6°] T_INT[16°] LUCI[0.8] MOVIMENTO[0.3]->  
P(Away): 0.00 | P(Home): 1.00 | P(Sleep): 0.00 | EU Totale: 1.699  
  
Appartamento 2:  
ORA[19:00] T_EXT[6°] T_INT[17°] LUCI[0.7] MOVIMENTO[0.5]->  
P(Away): 0.00 | P(Home): 1.00 | P(Sleep): 0.00 | EU Totale: 1.699  
  
Appartamento 3:  
ORA[19:00] T_EXT[5°] T_INT[16°] LUCI[0.0] MOVIMENTO[0.0]->  
P(Away): 0.84 | P(Home): 0.00 | P(Sleep): 0.16 | EU Totale: 0.032  
  
Appartamento 4:  
ORA[19:00] T_EXT[5°] T_INT[18°] LUCI[0.5] MOVIMENTO[0.6]->  
P(Away): 0.00 | P(Home): 0.84 | P(Sleep): 0.16 | EU Totale: 1.466
```

Il valore di **utilità attesa**, calcolato a partire dalle probabilità apprese dalla rete, rappresenta il principale indicatore della qualità dell'apprendimento, in quanto:

- aumenta quando il riscaldamento è realmente utile,
- diventa negativo in situazioni inefficienti o non occupate,
- funge da **ponte semantico** tra la fase di apprendimento e la fase di decisione ottimizzata.

In questo modo, l'apprendimento non viene valutato in modo isolato, ma in relazione alla sua capacità di **supportare decisioni energeticamente ed economicamente razionali**.

L'utilità attesa costituisce l'input principale per il modulo di programmazione lineare, consentendo di tradurre l'incertezza in decisioni ottimizzate.

## Sezione 3 - Programmazione Lineare

### Sommario

La fase finale del sistema è formulata come un **problema di Programmazione Lineare (PL)**, con l'obiettivo di determinare il **livello ottimale di riscaldamento** per ciascun appartamento, tenendo conto contemporaneamente di comfort, costi energetici e rischio decisionale.

La Programmazione Lineare consente di trasformare le informazioni probabilistiche e semantiche prodotte dai moduli precedenti in **decisioni quantitative ottimali**, soggette a vincoli globali.

## Strumenti utilizzati

- **GLPK (GNU Linear Programming Kit)** per la modellazione e risoluzione del problema
- **Modello di Programmazione Lineare continua**
- **Metodo del simplex** per la risoluzione del problema di ottimizzazione

L'utilizzo di GLPK permette una risoluzione efficiente e riproducibile del problema, mantenendo il modello esplicito e interpretabile.

## Decisioni di Progetto

Funzione obiettivo:

$$\max \sum_i x(i) \cdot (P_{\text{occ}}(i) \cdot EU(i) - \text{price}(i))$$

dove:

- $P_{\text{occ}}(i)$  è la probabilità di occupazione (Home + Sleep) stimata dalla rete neurale;
- $EU(i)$  è il valore di utilità attesa calcolato dal modulo di incertezza;
- $\text{price}(i)$  rappresenta il costo dell'energia nello slot considerato.

Questa formulazione consente di **bilanciare comfort e costo**, favorendo il riscaldamento solo quando il beneficio atteso supera il costo energetico.

Vincoli:

- budget energetico totale:

$$\sum_i x(i) \cdot \text{price}(i) \leq \text{budget}$$

- rischio massimo basato sulle probabilità di `Away`:

$$\sum_i x(i) \cdot \text{risk_coeff}(i) \leq \text{risk\_max}$$

- dominio sulle variabili decisionali:

$$0 \leq x_i \leq 1$$

Il vincolo di rischio impedisce di allocare energia in situazioni ad alta incertezza di occupazione, rendendo il comportamento del sistema più prudente.

## Valutazione

La Programmazione Lineare rappresenta il **livello decisionale finale** del sistema, nel quale:

- l'apprendimento automatico fornisce informazioni probabilistiche,
- il modulo di incertezza traduce tali informazioni in valore semantico,
- la PL produce una **decisione ottimizzata, esplicabile e vincolata**.

Questo approccio consente di passare da dati e probabilità a **azioni razionali**, rendendo il sistema completo dal punto di vista dell'intelligenza artificiale decisionale.

#### --- ANALISI AGENTE INTELLIGENTE ---

##### Appartamento 1:

ORA[19:00] T\_EXT[6°] T\_INT[16°] LUCI[0.8] MOVIMENTO[0.3] ->  
P(Away): 0.00 | P(Home): 1.00 | P(Sleep): 0.00 | EU Totale: 1.699

##### Appartamento 2:

ORA[19:00] T\_EXT[6°] T\_INT[17°] LUCI[0.7] MOVIMENTO[0.5] ->  
P(Away): 0.00 | P(Home): 1.00 | P(Sleep): 0.00 | EU Totale: 1.699

##### Appartamento 3:

ORA[19:00] T\_EXT[5°] T\_INT[16°] LUCI[0.0] MOVIMENTO[0.0] ->  
P(Away): 0.84 | P(Home): 0.00 | P(Sleep): 0.16 | EU Totale: 0.032

##### Appartamento 4:

ORA[19:00] T\_EXT[5°] T\_INT[18°] LUCI[0.5] MOVIMENTO[0.6] ->  
P(Away): 0.00 | P(Home): 0.84 | P(Sleep): 0.16 | EU Totale: 1.466

#### --- PIANO ENERGETICO OTTIMALE ---

Appartamento 1 -> Potenza 100.0%

Appartamento 2 -> Potenza 100.0%

Appartamento 3 -> Potenza 0.0%

Appartamento 4 -> Potenza 51.1%

I risultati ottenuti mostrano un comportamento coerente con le aspettative del dominio:

- gli appartamenti con **alta probabilità di occupazione** ricevono potenza elevata;
- le situazioni con **assenza di occupazione o bassa utilità attesa** vengono correttamente penalizzate;
- il sistema **evita sprechi energetici** in presenza di temperature interne già elevate;
- tutti i vincoli globali di budget e rischio risultano rispettati.

Il piano energetico risultante riflette un compromesso ottimale tra comfort, costo e incertezza, dimostrando l'efficacia dell'approccio integrato.

## Conclusioni

Il progetto dimostra come **apprendimento automatico probabilistico, ragionamento decisionale sotto incertezza e ottimizzazione matematica** possano essere integrati in modo coerente all'interno di un unico **Knowledge-Based System (KBS)** per il supporto alle decisioni energetiche.

L'architettura proposta separa chiaramente le tre macroaree funzionali:

- apprendimento dei modelli probabilistici a partire dai dati,
- valutazione delle alternative tramite utilità attesa,

- selezione delle decisioni ottimali tramite programmazione lineare.

Questa separazione consente al sistema di mantenere un'elevata **chiarezza concettuale**, oltre a facilitare la comprensione del ruolo di ciascun modulo nel processo decisionale complessivo.

#### Punti di forza

- **Modularità:** ogni componente (rete neurale, modulo di incertezza, ottimizzatore) è indipendente e sostituibile;
- **Interpretabilità:** l'uso di probabilità esplicite e funzioni di utilità rende il comportamento del sistema spiegabile;
- **Coerenza con i modelli teorici del corso:** il progetto applica in modo concreto i concetti di apprendimento supervisionato (ICON7–8), utilità attesa (ICON9) e decisione ottimizzata (ICON3).

#### Possibili estensioni

- introduzione di **reti bayesiane esplicite** per modellare in modo dichiarativo le dipendenze causali;
- estensione verso **apprendimento online**, per adattare il sistema a cambiamenti nel comportamento degli utenti;
- utilizzo di **modelli dinamici temporali** (es. Markoviani) per gestire sequenze di decisioni nel tempo.

Nel complesso, il sistema realizzato rappresenta un esempio concreto di **integrazione tra intelligenza artificiale e metodi di ottimizzazione**, mostrando come l'incertezza possa essere trattata in modo razionale e operativo all'interno di un processo decisionale automatico.

#### Riferimenti Bibliografici

- [1] Russell, Norvig – *Artificial Intelligence: A Modern Approach*
- [2] Slide del corso – ICON3, ICON7, ICON8, ICON9
- [3] GLPK Documentation