POLITECNICO DI TORINO

Corso di Laurea in Ingegneria Gestionale

Tesi di Laurea Magistrale

Analisi della prestazione energetica di edifici con tecniche data-driven per generare mappe energetiche

Caso di studio: la città di Torino

Relatori:
prof. Tania Cerquitelli
prof. Alfonso Capozzoli

Candidato:
Andrea Nasso

Tutore Aziendale
Edison Spa
Ing. Silvia Casagrande

Marzo - Aprile 2019

Ringraziamenti

Innanzitutto, vorrei ringraziare la Prof.ssa Tania Cerquitelli per avermi dato la possibilità di svolgere questa tesi e per il supporto fornitomi durante la stesura. Ringrazio inoltre il Prof. Alfonso Capozzoli per le nozioni e gli utili consigli.

Un ringraziamento va anche all'Ing. Silvia Casagrande e a tutti i componenti delle Officine Edison per l'accoglienza e la disponibilità ricevuta.

Ringrazio la Dott.ssa Evelina Di Corso, il Dott. Daniele Mazzarelli e il Dott. Stefano Proto per la pazienza, la disponibilità e l'aiuto ricevuto durante questi mesi.

Un ringraziamento speciale va alla mia famiglia che mi ha sempre sostenuto nei momenti di difficoltà e che mi ha permesso di raggiungere questo traguardo. Ringrazio infine gli amici per l'affetto e il sostegno ricevuto in questi anni.

Sommario

L'Attestato di Prestazione Energetica (APE) è un documento che permette di stimare le *performance* energetiche di un edificio basandosi su caratteristiche termo-fisiche e geometriche. A causa del volume dei dati disponibili e dell'eterogeneità degli attributi, l'esplorazione di questa collezione di dati risulta impegnativa. Questa tesi propone un nuovo *framework*, sviluppato in *Python*, che consente di trovare automaticamente elementi di conoscenza nascosta all'interno dei dati e di renderli facilmente fruibili attraverso la generazione di mappe geolocalizzate. Diversi potrebbero essere gli *stakeholder* interessati, come i privati che desiderano individuare delle zone in cui è conveniente comprare o affittare casa, la pubblica amministrazione che vuole attuare degli interventi migliorativi in alcune aree della città e gli esperti di dominio che desiderano approfondire le determinanti della prestazione energetica di un edificio. Il *framework* include una fase di *Data Integration*, seguita dal *Data Cleaning* e dal *Data Preparation* che permettono di ripulire il *dataset* e selezionare in maniera automatica gli attributi rilevanti per l'analisi. L'estrazione della conoscenza viene attuata nella fase di *Data Mining*, attraverso l'utilizzo di algoritmi di *clustering*. La conoscenza viene poi presentata agli utenti finali attraverso delle regole estratte a partire da alberi di decisione e per mezzo di una visualizzazione interattiva su mappe geolocalizzate. La metodologia proposta è stata valutata su un *dataset* con dati *open* contenente APE reali, appartenenti alla Regione Piemonte. La fase sperimentale dimostra l'efficacia del *framework* nell'esplorazione di *dataset* contenenti grandi collezioni di attestati di prestazione energetica e nella visualizzazione intuitiva degli elementi di conoscenza estratti.

La presente tesi è sviluppata in 4 capitoli, di cui viene di seguito proposta una disamina.

- Nel **Capitolo 1** viene fatta un'introduzione al concetto di certificazione energetica, passando dall'analisi del quadro normativo di riferimento ed arrivando alla descrizione degli elementi caratterizzanti un attestato di prestazione energetica

- Nel **Capitolo 2** viene introdotto il *framework* progettato, proponendo una descrizione dei principali blocchi costituenti anche attraverso una spiegazione dei concetti teorici presenti
- Nel **Capitolo 3** vengono presentati il *setting* dei parametri e i risultati sperimentali ottenuti dall'applicazione del *framework* al *dataset* di riferimento, descrivendo gli *output* relativi alle tecniche di *data mining* e alle mappe geolocalizzate
- Nel **Capitolo 4** vengono tratte delle conclusioni sul lavoro svolto e suggeriti dei possibili sviluppi futuri

Indice

Elenco delle figure

Elenco delle tabelle

Capitolo 1

La certificazione energetica

La certificazione energetica è un documento che esprime, in maniera sintetica, la prestazione energetica di un edificio attraverso l'attribuzione di un'etichetta di classe; in essa sono contenute varie caratteristiche energetiche dell'edificio, sia di natura geometrica sia di natura termofisica. Il certificato energetico permette agli acquirenti e ai conduttori di ottenere informazioni esaustive riguardanti l'efficienza energetica dell'edificio, fornendo un'indicazione degli interventi più significativi ed economicamente convenienti per il miglioramento della prestazione energetica [1]. Guardando ai consumi globali di energia, circa il 40% è attribuibile agli edifici residenziali e commerciali; fattori come la crescita della popolazione, l'aumento della domanda di edifici residenziali e del tempo speso all'interno degli edifici faranno aumentare velocemente il consumo di energia e le emissioni di gas serra [2]. La certificazione energetica diventa dunque uno strumento molto importante per raggiungere l'obiettivo fissato dalla Commissione Europea di ridurre entro il 2020 il consumo di energia del 20%. Il potenziale risparmio di energia può essere raggiunto operando su ciascuno dei comparti principali ma si stima che circa il 50% del risparmio complessivo possa provenire dal settore civile [1].

1.1 Evoluzione del quadro normativo

Nel corso degli anni si sono susseguite una serie di normative di riferimento e direttive, a partire dalle *Leggi 373/1976* e *10/1991* fino ai più recenti *Decreti del 26 giugno 2015*. Con la *Legge 373/1976* vennero introdotti per la prima volta dei vincoli per la progettazione, installazione e manutenzione degli impianti di produzione del calore e indicazioni sull'isolamento termico degli edifici. La prima legge finalizzata a regolare le modalità progettuali e la gestione dell'edificio è però stata la *Legge 10/1991* (attuata con il *Dpr n. 414/1993*) con la quale venne introdotta la classificazione del territorio nazionale in funzione dei gradi

giorno (GG) e degli edifici per destinazione d'uso. Di fondamentale importanza all'interno del quadro normativo è stata la *Direttiva 2002/91/CE*, denominata anche EPBD (*Energy Performance Building Directive*), con la quale la Comunità Europea ha cercato di orientare i vari Stati membri sul tema delle prestazioni energetiche nell'edilizia, adempiendo agli obiettivi del Protocollo di Kyoto. In particolare, veniva espressa l'importanza della definizione di una metodologia per il calcolo delle prestazioni energetiche e l'imposizione del rispetto dei requisiti minimi di efficienza energetica per gli edifici di nuova costruzione o in via di ristrutturazione. È qui che si ritrova per la prima volta il concetto di Certificazione Energetica, sulla quale doveva essere espressa in maniera semplice la prestazione energetica dell'edificio ed eventuali interventi migliorativi, in modo da consentire ai cittadini di fare delle scelte consapevoli. L'Italia ha attuato la Direttiva Europea 2002/91/CE con il *Dlgs. 19 agosto 2005, n.192*, integrato poi con il *Dlgs 29 dicembre 2006 n.311*. Con il *Dpr n.59 del 2009* vennero definiti le metodologie di calcolo, requisiti minimi e criteri realativi alla climatizzazione invernale ed estiva, all'acqua calda sanitaria e all'illuminazione artificiale. Sono state inoltre definite le norme tecniche utilizzate per il calcolo, valide su tutto il territorio, come la UNI/TS 11300-1 e la UNI/TS 11300-2. Nello stesso anno, con il *Dm 26 giugno 2009*, la certificazione energetica eseguita da un soggetto indipendente divenne obbligatoria. In seguito con il *Dlgs. 28/2011* fu introdotto l'obbligo dal 1° gennaio 2012 di riportare sugli annunci di vendita l'indice di prestazione energetica e di renderlo disponibile assieme al contratto di compravendita o locazione. Successivamente con il *Decreto del 22 novembre 2012* venne abrogata la possibilità di effettuare un'autodichiarazione in classe G (la peggiore dal punto di vista delle performance), la quale era stata introdotta con il precedente *Dm 26 giugno 2009*. Con il *Dl 63/2013* (convertito dalla *Legge 90/2013*), l'Attestato di Certificazione Energetica (ACE) diventa APE (Attestato di Prestazione Energetica) e recepisce la *Direttiva Europea 2010/31/CE* (detta anche EPBD *Recast*), attuata poi attraverso i tre *Decreti del 26 giugno 2015*. Una disamina sull'evoluzione del quadro normativo è presente nella Tabella 1.1.

Anno	Norma
1976	Legge del 30/04/1976 n. 373 - Norme per il contenimento del consumo energetico per usi termici negli edifici
1991	Legge 9 gennaio 1991, n. 10 - Norme per l'attuazione del Piano energetico nazionale in materia di uso razionale dell'energia, di risparmio energetico e di sviluppo delle fonti rinnovabili di energia
1993	Dpr n.412/1993 - Regolamento recante norme per la progettazione, l'installazione, l'esercizio e la manutenzione degli impianti termici degli edifici ai fini del contenimento dei consumi di energia
2002	Direttiva 2002/91/CE del Parlamento Europeo e del Consiglio sul rendimento energetico nell'edilizia
2005-2006	Decreto legislativo 19 agosto 2005, n. 192 - Attuazione della Direttiva 2002/91/CE, integrato con il Decreto legislativo 29 dicembre 2006, n. 311
2009	Decreto del Presidente della Repubblica 2 aprile 2009, n. 59 - Regolamento di attuazione dell'art. 4 c. 1 lett. a) e b) del D.Lgs. 192/2005
2009	Decreto interministeriale 26 giugno 2009, Certificazione energetica degli edifici - Linee guida nazionali per la certificazione energetica degli edifici
2010	Direttiva 2010/31/UE del Parlamento Europeo e del Consiglio sulla prestazione energetica nell'edilizia
2011	Decreto Legislativo 3 marzo 2011, n. 28 - Attuazione della Direttiva 2009/28/CE sulla promozione dell'uso dell'energia da fonti rinnovabili, recante modifica e successiva abrogazione delle direttive 2001/77/CE e 2003/30/CE
2012	Decreto 22 novembre 2012 - Modifica del Decreto 26 giugno 2009
2013	Decreto - Legge 4 giugno 2013, n. 63 - Disposizioni urgenti per il recepimento della Direttiva 2010/31/UE, sulla prestazione energetica nell'edilizia
2013	Legge 3 agosto 2013, n. 90 - Conversione in legge, con modificazioni, del Decreto legge 4 giugno 2013, n. 63, recante disposizioni urgenti per il recepimento della Direttiva 2010/31/UE sulla prestazione energetica nell'edilizia
2015	Decreto interministeriale 26 giugno 2015 - Applicazione delle metodologie di calcolo delle prestazioni energetiche e definizione delle prescrizioni e dei requisiti minimi degli edifici
2015	Decreto interministeriale 26 giugno 2015 - Schemi e modalità di riferimento per la compilazione della relazione tecnica di progetto ai fini dell'applicazione delle prescrizioni e dei requisiti minimi di prestazione energetica negli edifici
2015	Decreto interministeriale 26 giugno 2015 - Adeguamento linee guida nazionali per la certificazione energetica degli edifici

Tabella 1.1: Evoluzione del quadro normativo [3]

1.2 ACE ed APE: le differenze

Come detto nella sezione precedente, con la Legge 90/2013 l'Attestato di Certificazione Energetica (ACE) diventa Attestato di Prestazione Energetica (APE), con cui vengono introdotte delle novità. Nella Figura 1.1 un esempio di APE viene riportato. Una delle differenze più evidenti è l'aumento del numero di classi energetiche, con la suddivisione della vecchia classe "A" (la più performante) in quattro sottoclassi, da A1 (più bassa) ad A4 (più alta), mentre le altre classi da "B" a "G" rimangono invariate. Un altro cambiamento evidente si ha nella modalità di calcolo che permette di assegnare una certa classe energetica ad un edificio, anche se la base di partenza rimane, per entrambi gli attestati, l'energia primaria non rinnovabile. Per l'ACE la costruzione dei range è fatta partendo da valori fissi differenziati per zona climatica di appartenenza e fattore forma, mentre per l'APE il range deriva dal confronto dell'energia primaria non rinnovabile dell'edificio in esame con quella dell'edificio di riferimento (maggiori dettagli sulle modalità di calcolo saranno esposti nella sezione riguardante la classe energetica). Novità dell'APE è l'introduzione all'interno del certificato dei consumi stimati espressi in termini di energia consegnata all'edificio, a seconda del vettore energetico, in modo da permettere un confronto con i consumi reali [4].

1.3 Certificazione energetica nella Regione Piemonte

Le disposizioni del Dlgs 192/2005 permettono alle regioni di deliberare in merito alle certificazioni energetiche. La Regione Piemonte ha introdotto l'obbligo dell'Attestato di Certificazione energetica per edifici di nuova costruzione, sottoposti a ristrutturazione o oggetto di compravendita o locazione, con la Legge regionale 28 maggio 2007 n.13 *"Disposizioni in meteria di rendimento energertico"* [6]. Per permettere la raccolta degli ACE, è stato realizzato un Sistema Informativo denominato SICEE (Sistema Informativo Certificazione Energetica Edifici), nel quale i certificatori energetici iscritti potevano caricare i certificati di prestazione energetica e i cittadini verificare la presenza di un certificatore regolarmente iscritto sul Sistema [7]. Con la delibera della Giunta Regionale del *21 settembre 2015 n.14-2119*, la Regione Piemonte introduce il SIPEE (Sistema Informativo per Prestazione Energetica degli Edifici) in sostituzione del SICEE. Per il controllo della qualità e regolarità dell'attestazione, l'Agenzia Regionale per la Protezione Ambientale (ARPA), si impegna a controllare almeno il 2% degli APE depositati sul SIPEE con ispezioni a campione [8]. Il dataset oggetto di analisi, comprende certificazioni energetiche rilasciate dal CSI-Piemonte in modalità *open*, dagli anni 2009 al 2014 (ACE) e dal 2016 al primo semestre del 2018 (APE).

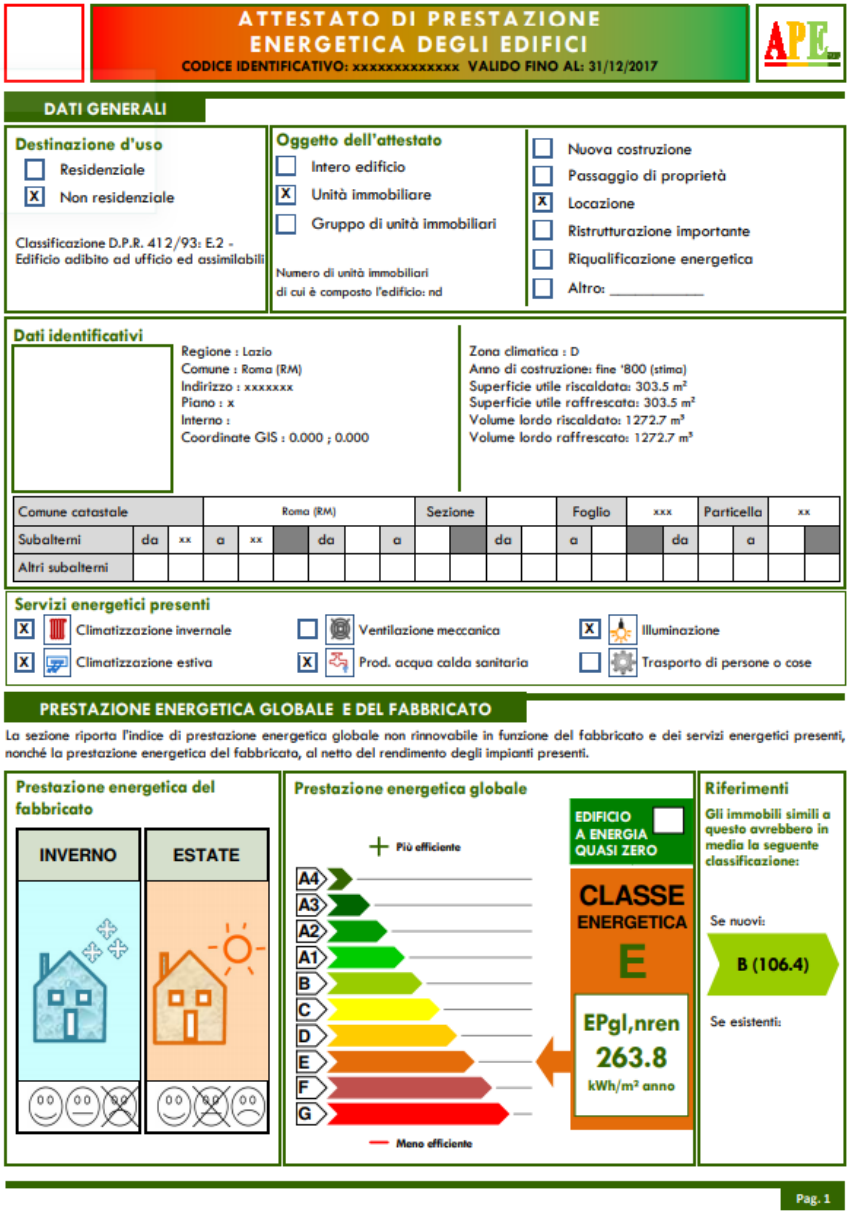

Figura 1.1: Fac simile di APE [5]

1.4 Descrizione degli attributi principali

Il *dataset* oggetto di analisi contiene sia dati provenienti dal vecchio SICEE, sia dati provenienti dal SIPEE. Salvo esplicita specifica, gli attributi di seguito analizzati si intendono

facenti parte (anche se con nomi leggermente diversi) in entrambi i dataset. Innanzitutto, è possibile suddividere gli attributi in varie sezioni:

- Dati catastali
- Dati tecnici generali dell'edificio
- Dati sui rendimenti e fabbisogni
- Dati sugli impianti

1.4.1 Dati catastali

In questa sezione vengono elencati gli attributi di natura catastale, che permettono principalmente di caratterizzare l'edifico dal punto di vista geografico.

- **Foglio**: espresso in numeri, rappresenta l'unità territoriale nella quale è catastalmente suddiviso ogni comune, spesso contenuto all'interno di una sezione
- **Particella**: anche noto come mappale (o numero di mappa), rappresenta nell'ambito di un foglio catastale una porzione di terreno o fabbricato
- **Subalterno**: consente l'identificazione della singola unità immobiliare all'interno della particella

La tripletta foglio, particella e subalterno consente di identificare in maniera univoca l'unità immobiliare all'interno del dataset; nel caso di duplicati è stata considerato il certificato con la data di caricamento sul portale più recente.

- **Comune**: il comune in cui si trova l'unità abitativa oggetto della certificazione
- **Indirizzo**: campo testuale libero in cui è specificato l'indirizzo dell'unità abitativa considerata
- **Numero Civico**: il numero civico dell'unità abitativa considerata
- **CAP**: Codice di Avviamento Postale del comune in esame
- **Latitudine**: rappresenta l'arco meridiano compreso tra l'equatore ed il punto considerato
- **Longitudine**: rappresenta l'arco di equatore compreso tra il meridiano di Greenwich ed il punto considerato

La coppia latitudine e longitudine permette la geolocalizzazione dell'edificio oggetto dell'attestato di prestazione energetica. Questi dati e il CAP non erano presenti all'interno dei vecchi certificati, ma sono stati generati prima dell'integrazione a partire dall'indirizzo e dal comune. Questo aspetto verrà poi approfondito nel successivo capitolo.

1.4.2 Dati tecnici generali

In questa sezione vengono elencati dati tecnici generali sull'edifico come l'epoca costruttiva o riguardanti le caratteristiche geometriche.

- **Anno di costruzione**: anno di costruzione dell'unità abitativa
- **Anno ultima ristrutturazione**: se presente, rappresenta l'anno dell'ultima ristrutturazione dell'edificio
- **Destinazione d'uso**: secondo la classificazione data dal *Dpr n. 412/1993* [9], permette di distinguere la categoria di appartenenza dell'edificio in esame; se un edificio è costituito da parti individuabili come appartenenti a categorie diverse, le stesse devono essere considerate in maniera separata. Le diverse categorie sono illustrate nella Tabella 1.2.

- **Superficie riscaldata** [m^2]: è la sola superficie coinvolta dalla climatizzazione invernale
- **Superficie utile** [m^2]: è data dall'unione delle superfici climatizzate dell'edificio, ovvero quelle riscaldate e raffrescate
- **Volume lordo riscaldato** (V) [m^3]: volume lordo delle parti di edificio coinvolte dalla climatizzazione invernale
- **Superficie disperdente totale** (S) [m^2]: superficie che delimita il volume climatizzato V rispetto all'esterno, al terreno, ad ambienti a diversa temperatura o ambienti non dotati di impianto di climatizzazione
- **Fattore forma** (S/V) [m^{-1}]: rapporto tra la superficie disperdente S e il volume lordo riscaldato V [10]. Questo parametro, anche se puramente geometrico, è molto importante in quanto più è estesa la superficie (in relazione al volume) maggiori sono le dispersioni termiche. Diversi tipi di edificio a parità di volume lordo riscaldato possono presentare diversi valori di fattore forma, ad esempio per una villetta un valore atteso sarebbe di circa 0.8 mentre per un edificio a torre circa 0.3 [11]

Destinazione d'uso	Descrizione
E1	Edifici adibiti a residenza e assimilabili
E1 (1)	abitazioni adibite a residenza con carattere continuativo, quali abitazioni civili e rurali, collegi,conventi, case di pena, caserme
E1 (2)	abitazioni adibite a residenza con occupazione saltuaria, quali case per vacanze, fine settimana e simili
E1 (3)	edifici adibiti ad albergo, pensione ed attività similari
E2	Edifici adibiti a uffici e assimilabili pubblici o privati, indipendenti o contigui a costruzioni adibite anche ad attività industriali o artigianali, purché siano da tali costruzioni scorporabili agli effetti dell'isolamento termico
E3	Edifici adibiti a ospedali, cliniche o case di cura e assimilabili ivi compresi quelli adibiti a ricovero o cura di minori o anziani nonché le strutture protette per l'assistenza ed il recupero dei tossicodipendenti e di altri soggetti affidati a servizi sociali pubblici
E4	Edifici adibiti ad attività ricreative o di culto e assimilabili
E4 (1)	quali cinema e teatri, sale di riunioni per congressi
E4 (2)	quali mostre, musei e biblioteche, luoghi di culto
E4 (3)	quali bar, ristoranti, sale da ballo
E5	Edifici adibiti ad attività commerciali e assimilabili quali negozi, magazzini di vendita all'ingrosso o al minuto, supermercati, esposizioni
E6	Edifici adibiti ad attività sportive
E6 (1)	piscine, saune e assimilabili
E6 (2)	palestre e assimilabili
E6 (3)	servizi di supporto alle attività sportive
E7	Edifici adibiti ad attività scolastiche a tutti i livelli e assimilabili
E8	Edifici adibiti ad attività industriali ed artigianali e assimilabili

Tabella 1.2: Destinazione d'uso stabilite nel *Decreto del Presidente della Repubblica 26 agosto 1993, n. 412*

- **Tipologia edificio**: indica se l'edificio in esame è ad esempio un appartamento, una villetta, in linea, a torre, ecc.

Prima di poter introdurre il concetto di trasmittanza delle superfici opache e trasparenti, è utile definire la trasmittanza termica. La qualità degli elementi che caratterizzano l'involucro di un edificio (come pareti e infissi) è molto importante e può avere un notevole

impatto sulle prestazioni energetiche. La trasmittanza termica (U) permette di quantificare la quantità di calore che passa da un corpo caldo ad un corpo freddo quando è sottoposto ad una differenza di temperatura. Secondo la UNI EN ISO 9646, la trasmittanza si definisce come il flusso di calore che attraversa una superficie unitaria sottoposta a differenza di temperatura pari ad un 1°C, tenendo in considerazione spessore, resistenza e conduttività termica dei vari strati costituenti l'involucro [12].

- **Trasmittanza media delle superfici opache** [W/m^{2}K]: trasmittanza termica media ponderata delle superfici opache che confinano con l'esterno, al netto dei ponti termici

- **Trasmittanza media delle superfici trasparenti** [W/m^{2}K]: trasmittanza termica media ponderata delle superfici trasparenti che confinano con l'esterno

- **Gradi giorno** (attributo disponibile nelle certificazioni ACE): è la somma, estesa a tutti i giorni di un periodo annuale convenzionale di riscaldamento, delle sole differenze positive giornaliere tra la temperatura dell'ambiente, convenzionalmente fissata a 20 °C, e la temperatura media esterna giornaliera; l'unità di misura utilizzata è il grado-giorno (GG) [9]. Il periodo si riferisce alla stagione termica per ciascuna zona climatica.

- **Zona climatica** (attributo disponibile nelle certificazioni ACE): il territorio nazionale è suddiviso in sei zone climatiche in funzione dei gradi-giorno, indipendentemente dalla ubicazione geografica (Vedi tabella 1.3)

Gradi Giorno	**Ore/giorno**	**Periodo**	**Zona**
GG $\leq$ 600	6	1/12 - 15/03	A
600 $<$ GG $\leq$ 900	8	1/12 - 31/03	B
900 $<$ GG $\leq$ 1400	10	15/11 - 31/03	C
1400 $<$ GG $\leq$ 2100	12	15/11 - 31/03	D
2100 $<$ GG $\leq$ 3000	14	15/10 - 15/04	E
GG $>$ 3000	-	-	F

Tabella 1.3: Gradi Giorno e zona climatica secondo il *Decreto del Presidente della Repubblica 26 agosto 1993, n. 412* [9]

1.4.3 Dati su rendimenti e fabbisogni

In questa sezione vengono elencati gli attributi relativi al rendimento dei vari sottosistemi e al fabbisogno di energia.

Nel mondo reale i sistemi di riscaldamento presentano delle perdite e, per questo motivo, l'energia fornita all'elemento radiante è maggiore rispetto a quella emanata verso l'ambiente. Per tenere conto di queste condizioni bisogna considerare il rendimento dei vari sottosistemi ovvero di produzione, di distribuzione, quello di di trasmissione e di regolazione. Di seguito vengono definiti i vari rendimenti che sono quelli medi stagionali:

- **Rendimento di generazione**: è il rapporto fra il calore utile prodotto dal generatore nella stagione di riscaldamento e l'energia fornita nello stesso periodo sotto forma di combustibile ed energia elettrica [13]
- **Rendimento di distribuzione**: è il rapporto tra il calore utile fornito ai terminali di emissione e il calore utile fornito al sistema di distribuzione del generatore [14]
- **Rendimento di regolazione**: è il rapporto tra l'energia richiesta per il riscaldamento degli ambienti con una regolazione teorica perfetta e l'energia richiesta per il riscaldamento degli ambienti con l'impianto di regolazione reale [14]
- **Rendimento di emissione**: è il rapporto tra l'energia richiesta per il riscaldamento degli ambienti con un sistema di emissione in grado di fornire una temperatura ambiente con uniformità ed uguale nei vari ambienti e l'energia richiesta per il riscaldamento degli stessi ambienti con l'impianto di emissione reale [14]

Per avere un'indicazione sintetica del rendimento relativo alla climatizzazione invernale è stata generata la variabile **ETAH**, calcolata come prodotto dei quattro rendimenti sopra citati. La determinazione della classe energetica deve tenere in considerazione, oltre ai rendimenti, anche alcuni indici di fabbisogno, tra cui:

- **Fabbisogno di energia termica utile** ([kWh/m^2] o [kWh/m^3]): indica la quantità di energia primaria necessaria a mantenere la temperatura di progetto nel corso dell'anno negli ambienti riscaldati
- **Indice di prestazione energetica invernale ($\mathbf{EP_{H,nd}}$)** [kWh/m^{2}anno]: dato relativo all'involucro dato dal rapporto tra il fabbisogno di energia termica utile e la superficie utile
- **Indice di prestazione energetica invernale ($\mathbf{EP_H}$)**[kWh/m^{2}anno]: è dato dal rapporto tra l'indice di prestazione energetica invernale dell'involucro ($EP_{H,nd}$) e il rendimento medio stagionale dell'impianto di riscaldamento.

1.4.4 Dati sugli impianti

In questa sezione vengono elencati gli attributi relativi agli impianti tra cui:

- **Combustibile**: indica il tipo di combustibile utilizzato dall'impianto, come ad esempio il gas naturale, l'energia elettrica o le biomasse

- **Tipo impianto**: nei vecchi certificati questo campo indicava se l'impianto era centralizzato o autonomo, ma con il nuovo APE il campo è diventato libero e può perciò contenere sia il tipo di caldaia (a condensazione ad esempio) sia lo specifico modello

- **Potenza nominale** [kW]: se è presente indica la potenza dell'impianto espresso in KW

1.5 La classe energetica

La classe energetica esprime in maniera sintetica le prestazioni di un edificio attraverso l'assegnazione di un'etichetta e il suo calcolo avviene sulla base del fabbisogno globale di energia termica primaria. Il metodo è cambiato nel corso delgi anni, è quindi utile esaminare differentemente il caso dell'ACE e dell'APE.

1.5.1 Classe energetica: ACE

Prima di descrivere il metodo di determinazione della classe energetica per l'ACE, è utile definire l'**indice di prestazione energetica globale lordo ($\mathbf{EP_L}$)** come la somma tra l'indice di prestazione energetica per il riscaldamento invernale e quello per l'acqua calda sanitaria. Questi valori, nel caso del SICEE, tengono conto dei Gradi Giorno relativi alla localizzazione dell'edificio da certificare, in modo da permettere il confronto tra immobili distribuiti in zone diverse. Nella Tabella 1.4 viene descritto il metodo di determinazione della classe energetica, che avviene confrontando l'EP_L con dei valori limiti imposti dalla normativa *Dpr n. 412/93*.

Range	Classe
$EP_L < 27kWh/m^2$	A+
$27kWh/m^2 \leq EP_L < 44kWh/m^2$	A
$44kWh/m^2 \leq EP_L < 82kWh/m^2$	B
$82kWh/m^2 \leq EP_L < 143kWh/m^2$	C
$143kWh/m^2 \leq EP_L < 201kWh/m^2$	D
$201kWh/m^2 \leq EP_L < 249kWh/m^2$	E
$249kWh/m^2 \leq EP_L < 300kWh/m^2$	F
$300kWh/m^2 \leq EP_L < 436kWh/m^2$	G
$436kWh/m^2 \leq EP_L$	NC

Tabella 1.4: Valori limiti di EP_L secondo la normativa *Dpr n. 412/93* [9]

1.5.2 Classe energetica: APE

A differenza dell'ACE, la classe energetica viene determinata attraverso il confronto dell'indice di prestazione energetica globale non rinnovabile ($EP_{gl,nren}$) dell'edifico in esame con quello dell'edifico di riferimento[1] ($EP_{gl,nren,rif}$). L'$EP_{gl,nren}$ è definito dalla somma dei seguenti componenti [15]:

- $EP_{H,nren}$: indice di prestazione energetica non rinnovabile relativo alla climatizzazione invernale
- $EP_{W,nren}$: indice di prestazione energetica non rinnovabile relativo all'acqua calda sanitaria
- $EP_{C,nren}$: indice di prestazione energetica non rinnovabile relativo alla climatizzazione estiva
- $EP_{V,nren}$: indice di prestazione energetica non rinnovabile relativo alla ventilazione
- $EP_{L,nren}$: indice di prestazione energetica non rinnovabile relativo all'illuminazione
- $EP_{T,nren}$: indice di prestazione energetica non rinnovabile relativo al trasporto di persone o cose

Nella Tabella 1.5 viene illustrata la scala di classificazione degli edifici in base all'indice di prestazione energetica globale non rinnovabile $EP_{gl,nren}$:

[1] Edificio dotato dei requisiti minimi di legge e che ha elementi edilizi ed impianti standard stabiliti dal DM 26 giugno 2015

Range	**Classe**
$EP_{gl,nren} \leq 0{,}4\ EP_{gl,nren,rif}$	A4
$0{,}4\ EP_{gl,nren,rif} < EP_{gl,nren} \leq 0{,}6\ EP_{gl,nren,rif}$	A3
$0{,}6\ EP_{gl,nren,rif} < EP_{gl,nren} \leq 0{,}8\ EP_{gl,nren,rif}$	A2
$0{,}8\ EP_{gl,nren,rif} < EP_{gl,nren} \leq 1{,}0\ EP_{gl,nren,rif}$	A1
$1{,}0\ EP_{gl,nren,rif} < EP_{gl,nren} \leq 1{,}2\ EP_{gl,nren,rif}$	B
$1{,}2\ EP_{gl,nren,rif} < EP_{gl,nren} \leq 1{,}5\ EP_{gl,nren,rif}$	C
$1{,}5\ EP_{gl,nren,rif} < EP_{gl,nren} \leq 2{,}0\ EP_{gl,nren,rif}$	D
$2{,}0\ EP_{gl,nren,rif} < EP_{gl,nren} \leq 2{,}6\ EP_{gl,nren,rif}$	E
$2{,}6\ EP_{gl,nren,rif} < EP_{gl,nren} \leq 3{,}5\ EP_{gl,nren,rif}$	F
$EP_{gl,nren} > 3{,}5\ EP_{gl,nren,rif}$	G

Tabella 1.5: Scala di classificazione secondo il *Decreto 162/2015*

Capitolo 2

Framework

In questo capitolo viene presentato il *framework* (Figura 2.1) che è stato progettato e sviluppato per lavorare su un *dataset* reale contenente certificazioni energetiche.

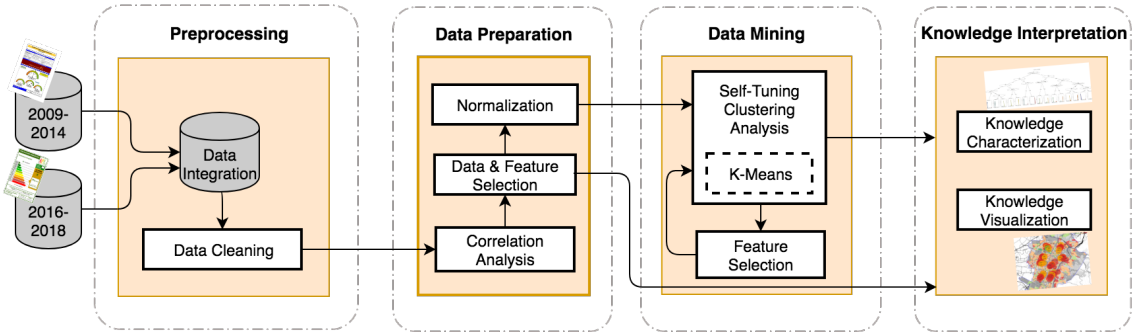

Figura 2.1: Framework sviluppato

Il *framework* è costituito da quattro blocchi principali, ognuno dei quali fa riferimento agli *step* presenti nel *Knowledge Discovery in Database* (KDD), ovvero:

1. **Data Preprocessing**: comprende la fase di *Data Integration* e di *Data Cleaning*
2. **Data Preparation**: comprende la *Correlation Analysis*, la *Data and Feature Selection* e la loro successiva *Normalization*
3. **Data Mining**: comprende il raggruppamento dei dati in cluster attraverso l'uso dell'algoritmo *K-Means* e la *Feature Selection*
4. **Knowledge Interpretation**: comprende la *Knowledge Visualization* per mezzo di mappe geolocalizzate e la *Knowledge Characterization* attraverso alberi di decisione

KDD

Il KDD è un termine coniato nel 1989 per enfatizzare il fatto che la conoscenza è il prodotto finale di un processo *data-driven*, definito come il processo non banale di identificazione di pattern validi, nuovi, potenzialmente utili e comprensibili nei dati. In altre parole l'obiettivo è quello di estrarre conoscenza ad alto livello da un elevato numero di dati di basso livello. Molte volte questo processo di scoperta viene chiamato erroneamente *data mining*, il quale invece indica l'applicazione di un certo tipo di algoritmo necessario ad estrarre un *pattern* ed è quindi uno *step* intermedio del KDD [16].

Il processo di KDD è un processo iterativo ed interattivo nel quale molte volte è l'utente il *decision maker* e può essere suddiviso in 9 step [17]:

1. Sviluppo e comprensione del dominio applicativo e dello stato dell'arte e identificazione dell'obiettivo del KDD dal punto di vista del cliente
2. Selezione di un *dataset* e di un sottoinsieme di attributi rilevanti per l'analisi
3. Pulizia dei dati e *preprocessing* degli stessi, attraverso l'eliminazione di dati che potrebbero introdurre problemi e gestione di eventuali dati mancanti
4. Riduzione della dimensionalità dei dati come scopo di definire le variabili utili a descrivere la conoscenza a seconda dell'obiettivo definito precedentemente
5. Trovare il matching tra l'obiettivo al punto 1 e un particolare metodo di *data mining* (classificazione, clustering, regressione, ecc.)
6. Scelta dell'algoritmo di *data mining* (e dei relativi parametri) usato per estarre pattern dai dati
7. Fase di *data mining* che include la ricerca di pattern interessanti e la definizione di una forma rappresentativa utile alla comprensione
8. Interpretazione dei risultati attraverso la loro visualizzazione ed analisi (possibile ripetizione degli step precedenti)
9. Consolidamento della conoscenza estratta ad esempio attraverso la sua formalizzazione su report

Questo processo viene solitamente ripetuto varie volte variando i parametri e gli algoritmi scelti, finchè non si giunge ad estrarre un tipo di conoscenza utile e che soddisfi l'obiettivo iniziale. Tuttavia il KDD può essere visto attraverso alcuni *step* base, illustrati nella Figura 2.2.

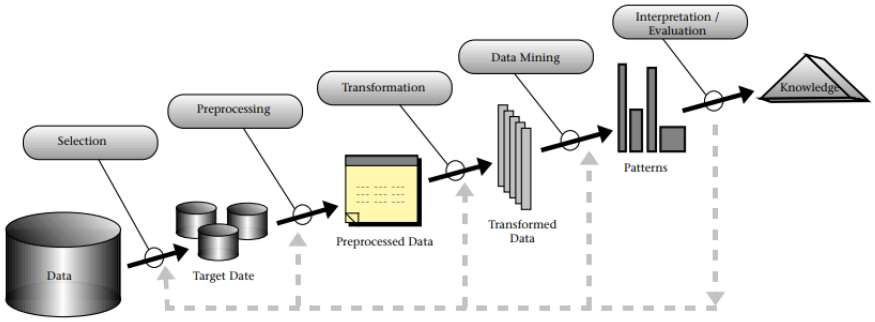

Figura 2.2: Step del processo di KDD ©Fayyad, Piatetsky-Shapiro, Smith [16]

2.1 Data preprocessing

Il blocco relativo al preprocessing, presente all'interno del *framework*, ha come scopo quello di limitare gli effetti dovuti alla presenza di dati non puliti e/o inconsistenti, i quali potrebbero impattare negativamente sulla qualità finale dei risultati. In generale diversi sono i motivi per cui i dati non sono affidabili, potrebbero infatti essere:

- Incompleti: attributi non valorizzati, mancanza di variabili rilevanti, presenza di soli dati aggregati
- Rumorosi: contengono errori o *outlier*
- Inconsistenti: vi sono ad esempio delle discrepanze tra i valori
- Duplicati: gli stessi dati sono presenti più volte

Il blocco di data prepocessing comprende al suo interno due fasi, tra cui (i) la *Data Integration* e (ii) il *Data Cleaning*. Di seguito vengono presentati in dettaglio le varie fasi.

2.1.1 Data Integration

I *dataset* a disposizione provengono da diverse fonti, in particolare sono presenti quelli contenenti informazioni sugli Attestati di Certificazione Energetica (dal 2009 al 2014) e quelli *dataset* che contengono informazioni sugli Attestati di Prestazione Energetica (dal 2016 al 2018), forniti dal CSI Piemonte[1] (Consorzio per il Sistema Informativo). In seguito

[1] http://www.csipiemonte.it/web/it/

si farà riferimento a questi DB rispettivamente come *DB old* e *DB new*. In questa fase i vari db messi a disposizione sono stati elaborati ed integrati in modo da creare un unico *dataset* contenente tutte le certificazioni energetiche. Il *DB old* era a sua volta suddiviso in 4 *dataset* (2009, 2010, 2013 e 2014) per cui è stato necessario concatenare queste informazioni, rinominando in maniera coerente i nomi delle variabili. I dati relativi al *DB new* sono invece stati forniti sottoforma di diversi file CSV, ognuno dei quali conteneva parti diverse di informazioni relative al certificato. Si è dovuto dunque effettuare un *merge* tra i vari file per ottenere un *dataset* utilizzabile per poi passare, dopo aver rinominato gli attributi relativi a quest'ultimo, alla concatenazione del *DB old* e del *DB new*. Prima di passare alla fase successiva si è provveduto all'eliminazione di certificazioni duplicate relative allo stesso appartamento attraverso la creazione di un unico gruppo per foglio, particella e subalterno e la successiva selezione del certificato più recente (data di caricamento maggiore). Uno schema semplificato della fase di *Data Integration* è proposto in Figura 2.3.

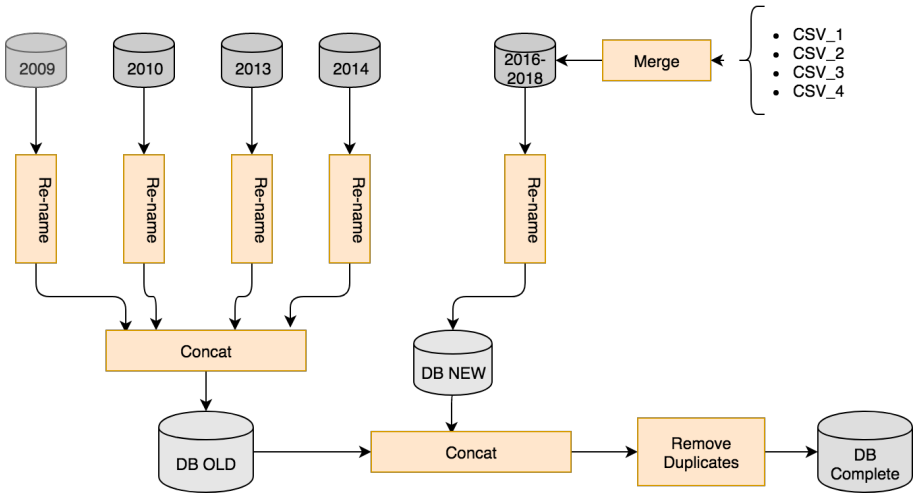

Figura 2.3: Step semplificato del processo di *Data Integration*

2.1.2 Data Cleaning

Dopo aver ottenuto un dataset integrato su cui poter effettuare le analisi, viene eseguita la fase di *Data Cleaning*. In questa fase viene dapprima svolta un'analisi di tipo univariata da parte dell'esperto di dominio su alcuni attributi ritenuti fondamentali per l'analisi, per poi passare alla fase di *Address Resolution* e all'*Outlier Detection* attraverso tecniche di tipo univariate e multivariate.

Expert-Driven univariate analysis

Il *framework* tiene conto dei suggerimenti forniti dall'esperto di dominio per attuare un'analisi di tipo univariata su attributi ritenuti molto importanti ai fini dell'analisi sulle certificazioni energetiche. In questa fase il *framework* applica dei filtri su alcune variabili termofisiche (*fattore forma, trasmittanza media delle superfici opache e trasparenti*) e riguardanti l'efficienza dei sottosistemi di riscaldamento (come il *rendimento di generazione* e quello di *distribuzione*). In questo modo viene fornita ai non esperti di dominio una configurazione che tiene già conto del significato fisico delle variabili, anche attraverso un'analisi di tipo *domain-driven* e non solo esclusivamente *data-driven*.

Address Resolution

Questa fase è fondamentale in quanto lo scopo finale è quello di visualizzare la conoscenza estratta tramite l'utilizzo di mappe energetiche. Il *framework* mette a disposizione un metodo che permette di pulire gli attributi geospaziali, in particolar modo l'indirizzo, il numero civico, il CAP, la latitudine e la longitudine. Dato che il campo relativo all'indirizzo è un campo di testo libero, è necessario prevedere la correzione automatica di errori di battitura o formato. Un metodo automatico e molto efficace consiste nell'utilizzo delle API (*Application Programming Interface*) messe a disposizione da Google[2], le quali dato un indirizzo in input riescono a risolverlo restituendo quello corretto ed altre informazioni aggiuntive come il CAP e le coordinate geografiche. Tuttavia vi è il grosso limite derivante dal numero di richieste che si possono fare al server in modalità free, non permettendo dunque l'uso esclusivo di questa soluzione per la pulizia delle coordiante geospaziali. Per questo motivo il *framework* affianca al servizio discusso in precedenza un nuovo metodo basato sul confronto dei dati presenti nel DB con quelli del viario di Torino[3]. Questo viario contiene infatti al suo interno la lista degli indirizzi presenti, ad ognuno dei quali sono associate le informazioni relative al CAP, alle coordinate e al numero civico. In particolare l'algoritmo sviluppato confronta l'indirizzo presente nel DB con quello presente nel viario, attraverso un'indice di similarità calcolato a partire dalla distanza di *Levenshtein* [18]. Essa esprime il minimo numero di modifiche (inserimento, cancellazione o sostituzione di un carattere) necessaria a trasformare la prima stringa nella seconda. In particolare la distanza di Levenshtein è definita come:

[2]developers.google.com/maps/documentation/geocoding/intro

[3]http://geoportale.comune.torino.it/web/

$$Levenshtein_{distance} = 2 * S + I + C$$

dove S è il numero di sostituzioni che servono a trasformare la prima stringa nella seconda, I è il numero di inserimenti e C il numero di cancellazioni. Definita Len_{FS}, la somma della lunghezza tra la prima e la seconda stringa, l'indice di similarità è calcolato come:

$$Similarity_{index} = \frac{Len_{FS} - Levenshtein_{distance}}{Len_{FS}}$$

Prima di procedere alla risoluzione degli indirizzi, il *framework* sviluppato effettua una pulizia dei caratteri non-ASCII, in modo da ridurre successivi problemi dovuti alla codifica. L'algoritmo calcola poi la distanza di Levenshtein tra le due stringhe in modo da ottenere l'indice di similarità, la quale può avere valori compresi nel range [0,1], dove 1 indica la completa similarità tra le stringhe (sono quindi uguali) e 0 la totale dissimilarità. Data una certa soglia t definita dall'utente, se l'indice di similarità è maggiore o uguale alla soglia definita l'indirizzo presente nel DB viene sostituito con quello presente nel viario, così come le altre informazioni ad esso connesse come il CAP, il numero civico e le coordinate. Qualora non sia possibile associare l'indirizzo originale con quelli presenti nel viario, ad esempio perchè l'indice di similarità calcolato è inferiore alla soglia t, viene utilizzato il servizio di geocoding messo a disposizione da Google. L'introduzione del metodo della risoluzione delle coordiante geospaziali utilizzando la distanza di Levenshtein ha dunque permesso di evitare l'utilizzo massivo del servizio di Google Geocoding, restringendone l'uso soltanto ad alcuni casi. In ultima analisi, se entrambi i metodi non riescono a risolvere l'indirizzo, il certificato viene eliminato [19].

Outlier detection

I dati provenienti dal mondo reale sono spesso inutilizzabili così come sono anche a causa della presenza di *outlier* [20], definito come un valore estremo il cui valore si discosta molto dalle altre osservazioni. Ciò può accadere per vari motivi come l'errato inserimento del dato nel sistema, un errore di rilevazione o una non corretta codifica del dato in fase di caricamento e di *export*. Per identificare la presenza di outlier e di valori inconsistenti è utile analizzare la distribuzione dei dati, attraverso l'uso di tecniche statistiche o grafiche. Il *framework* sfrutta due approcci (applicati in serie) per l'*outlier detection*: (i) *outlier detection univariata*, (ii) *outlier detection multivariata*.

Per l'*outlier detection* di tipo univariata il *framework* integra due metodologie in grado di individuare e rimuovere automaticamente gli outlier, che sono:

- **gESD (generalized Extreme Studentized Deviate)** [21]: è un metodo parametrico utilizzato per individuare uno o più outlier su dati che seguono approssimativamente una distribuzione normale. Affinchè questo test dia buoni risultati è necessario confermare che la precedente assunzione venga rispettata. Il limite di questo metodo è che bisogna settare a priori il numero massimo di outlier che si pensa possano essere presenti; settare in maniera non corretta questo valore puà impattare negativamente sul risultato del test. Dato il limite superiore k, il gESD esegue n test separati, calcolando le statistiche $R_1, ..., R_k$ riducendo via via la numerosità $(n, n-1, ..., n-k+1)$. La statistica R, per il set completo di dati, è definita come:

  $$R_1 = \frac{max|x_i - \bar{x}|}{s}$$

  dove:

  $$\bar{x} = \frac{\sum x_i}{n}; s^2 = \frac{\sum (x_i - \bar{x})^2}{n-1}$$

  Le altre $k-1$ statistiche vengono calcolate riducendo la numerosità attraverso l'eliminazione iterata dell'osservazione che massimizza $max|x_i - \bar{x}|$. Il gESD è definito per le seguenti ipotesi:

  - H_0: Non sono presenti outlier nel dataset
  - H_1: Ci sono fino a k outlier nel dataset

  I valori critici del test vengono determinati specificando il livello di significatività α e trovando β e $\lambda(\beta)$ per cui

  $$Pr[R_i > \lambda_i(\beta)|H_0] = \beta, i = 1, ..., k$$

  e

  $$Pr\left\{\bigcup_{i=1}^{k}[R_i > \lambda_i(\beta)|H_0]\right\} = \alpha$$

  Se tutte le statictiche R_i sono $\leq \lambda_i(\beta)$, allora non vi sono outlier mentre se qualche $R_i > \lambda_i(\beta)$ il metodo definisce il numero di outlier come pari al più grande valore di i per cui la precedente disuguaglianza è rispettata.

- **Percentile Outlier Detection**: metodo per l'eliminazione di *outlier* basato sul concetto di percentile [22]. Prendendo un insieme ordinato di dati, il percentile p è quel valore che è maggiore di una percentuale p dei dati e minore della restante percentuale $100 - p$, dove p è un numero nel range $[0 - 100]$. Questo metodo è utile per valutare la distribuzione dei dati, permettendo l'eliminazione di una coda dei dati.

Per l'outlier detection di tipo multivariata, il *framework* utilizza l'algoritmo **DBSCAN** *(Density-Based Spatial Clustering of Application with Noise)* [23]. Questo algoritmo, basandosi sul concetto di *density-reachability*, riesce ad individuare zone ad alta densità e zone a bassa densità (contenenti noise ed etichettabali come outlier). Sono due i parametri che devono essere settati affinchè si possa definire il concetto di densità, ovvero *minPoints* ed *Epsilon*. Dato un punto p, si definisce *Eps-neighborhood* ($N_{Eps}(p)$) come:

$$N_{Eps}(p) = \{q \in D | dist(p, q) \leq Eps\}$$

Ciò significa che un punto p appartiene ad un cluster D se esiste almeno un altro punto q ad una distanza minore o uguale ad *Eps* (raggio). In un cluster vi sono due tipi di punti, quelli dentro al cluster detti *core points* e punti che stanno sul bordo, detti *border points*. In generale, un Eps-neighborhood di un border point contiene molti meno punti di un *Eps-neighborhood* di un *core point*. Un punto p è *directly density-reachable* da un punto q se sono rispettate le seguenti condizioni:

$$p \in N_{Eps}(q)$$

$$|N_{Eps}(q)| \geq MinPoints$$

Tutti i punti che soddisfano l'ultima condizione sono *core point*, mentre un *border point* è semplicemente un punto che si trova entro l'*Eps-neighborhood* di un *core point* ma che non soddisfa la condizione di *Minpoints*. Il DBSCAN riesce ad individuare *noise points* (o outlier), come quei punti che non vengono asseganti a nessun cluster perchè non sono nè *core point* nè *border point*, come illustrato in Figura 2.4.

Un vantaggio di questo algoritmo sta nella sua capacità di individuare cluster anche di forma diversa tra loro (Figura 2.5) senza consoscerne a priori il numero e settando solo i due parametri *Eps* e *MinPoints*.

Esistono però casi in cui il DBSCAN non lavora bene (Figura 2.6), ovvero in presenza di cluster che presentano forma globulare o densità variabile (potrebbe essere necessario fare più *run* per avere un buon risultato). La scelta dei parametri *Eps* e *MinPoints* influenza notevolmente le prestazioni dell'algoritmo poichè questi, una volta scelti, vengono utilizzati per clusterizzare l'intero *dataset*. Esistono delle euristiche che permettono di impostare il *setting* corretto dei parametri, basati sul concetto del *k-nearest neighbors*, secondo il quale i punti che si trovano all'interno dello stesso cluster sono vicini tra loro mentre i *noise point* presentano *k-nearest neighbors* più distanti. Plottando le varie distanze, calcolate tra le varie coppie di punti, in maniera ordinata si riesce ad avere una visione della distribuzione di densità. Per specificare in maniera corretta i parametri, il *framework* proposto plotta

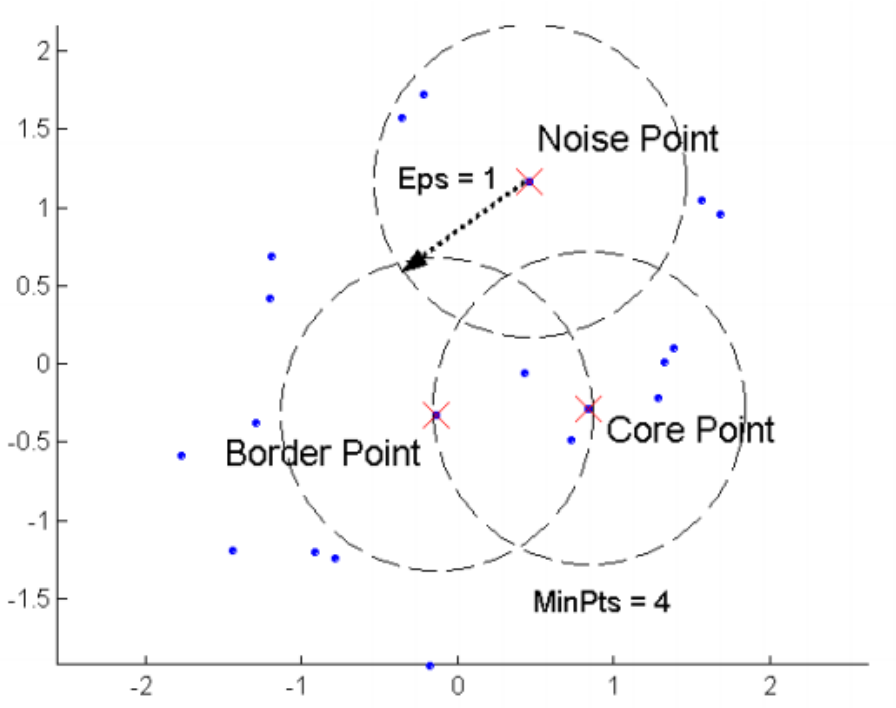

Figura 2.4: Punti Core, Border e Noise ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

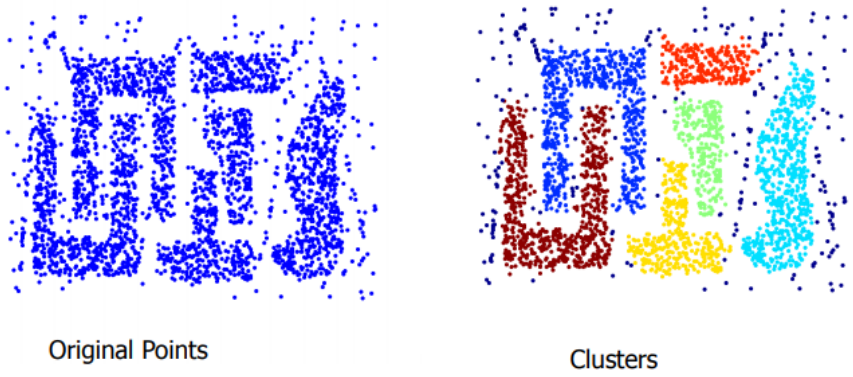

Figura 2.5: Situazione in cui il DBSCAN lavora bene ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

il *k-distance plot* ed estrae automaticamente i valori considerati migliori. In particolare il *k-distance plot* viene generato per diversi valori di *Eps* e *MinPoints* e viene selezionato come Minpoints il valore per cui la curva si stabilizza e come *Eps* il valore presente in

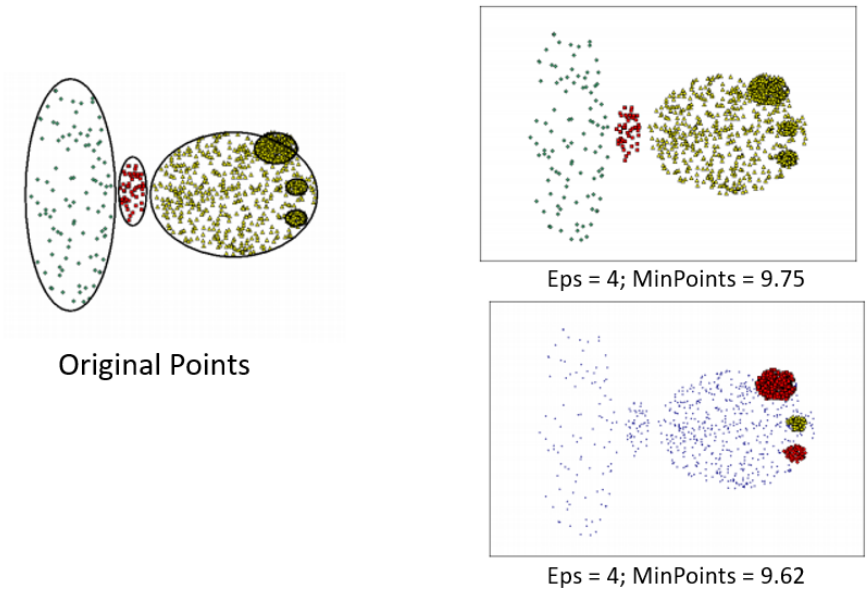

Figura 2.6: Situazione in cui il DBSCAN lavora male ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

corrispondenza del gomito individuato sulla curva [24]. In Figura 2.7 viene mostrato un esempio di *k-distance plot*, la cui curva si è stabilizzata per un valore di *MinPoints* pari a 5 e il valore di *Eps* di 0.28 è stato individuato sul gomito evidenziato dalla linea verde.

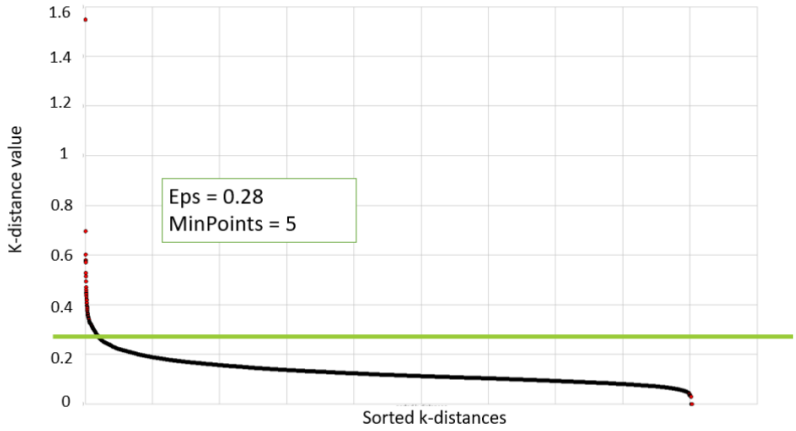

Figura 2.7: Esempio di k-distance plot con setting dei parametri

2.2 Data preparation

Il secondo blocco principale del *framework* è costituito dalla *Data Preparation*, la quale include la fase di *Correlation Analysis*, di *Data and Feature Selection* e di *Normalization*. Data la numerosità elevata degli attributi presenti all'interno del dataset è necessario (i) ridurre la dimensionalità dei dati e (ii) normalizzarli (operazione utile per l'utilizzo di algoritmi di clustering basati su distanze).

2.2.1 Correlation Analysis

Questa fase, precedente alla Data Preparation, consente di fornire informazioni riguardanti la correlazione tra le variabili presenti, fornendo un metodo per filtrare e scegliere gli attributi da utilizzare nel processo di Data Mining. Dati due attributi, un'analisi di correlazione permette di misurare quanto un attributo implichi l'altro, ovvero indica in che misura essi siano legati. Per gli attributi di tipo numerico, la correlazione viene misurata attraverso il *coefficiente di correlazione lineare*, anche detto *coefficiente di Pearson*, calcolato come:

$$r = \frac{\sum_i (x_i - \bar{x_i})(y_i - \bar{y_i})}{\sqrt{\sum_i (x_i - \bar{x_i})^2} \sqrt{\sum_i (y_i - \bar{y_i})^2}}$$

dove $\bar{x_i}$ è la media della variabile X e $\bar{y_i}$ la media di Y. Il valore di r è compreso tra -1 ed 1; se X e Y sono completamente correlate, r assume valore 1 o -1, mentre se r è pari a 0, significa che le due variabili sono indipendenti [25]. Lo scopo di questa fase è quello di capire quali attributi sono altamente correlati e quali invece sono scarsamente correlati in modo da includere nell'analisi solo quelli rilevanti, anche grazie alla loro visualizzazione attraverso una *correlation matrix*, di cui un esempio viene riportato in Figura 2.8.

2.2.2 Data and Feature Selection

In questa fase gli attributi rilevanti per l'analisi vengono presi in considerazione. La scelta viene guidata dalla matrice di correlazione generata in precedenza dal *framework*, attraverso un approccio di tipo data-driven. Come nella fase di outlier detection univariata, anche in questo caso il *framework* include alcune scelte fatte dall'esperto di dominio, in modo tale da assicurare che le variabili effettivamente scelte per fare Data Mining siano rilevanti dal punto di vista dell'obiettivo finale della caratterizzazione energetica degli edifici.

2.2.3 Normalization

Il processo di normalizzazione [26] (o standardizzazione) viene effettuato per evitare che l'unità di misura influisca sull'analisi, trasformando dei dati in un range comune come ad

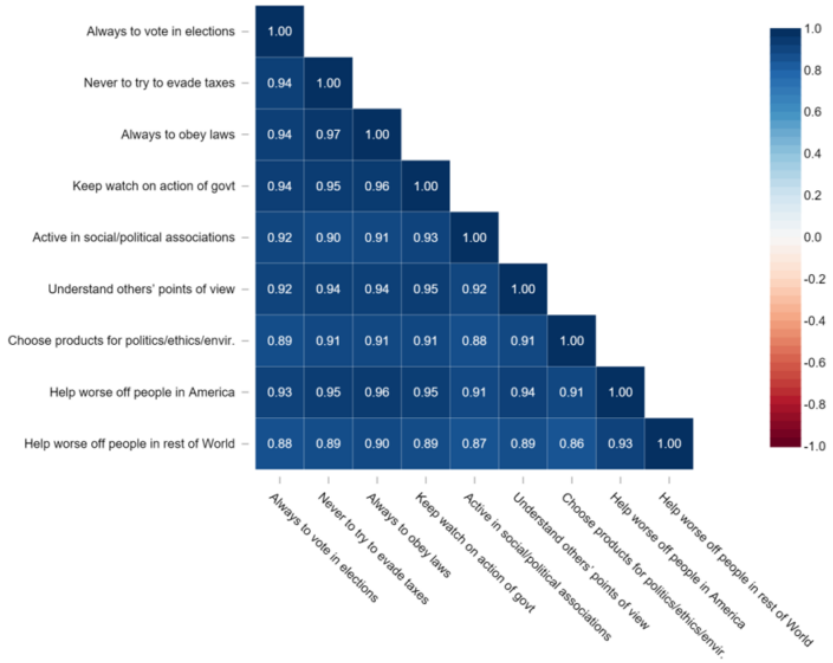

Figura 2.8: Esempio di correlation matrix ©displayr.com

esempio [0,1] o [-1,1]. Esistono diversi metodi di normalizzazione tra cui:

- **Max-min normalization**: esegue una trasformazione lineare sui dati. Preso in esame un certo attributo A, siano min_A e max_A rispettivamente il minimo e il massimo valore assunto dalla variabile in esame. Questo metodo trasforma un valore v_i di A in v_i' all'interno del range $[new\_min_A, new\_max_A]$ calcolando:

$$v_i' = \frac{v_i - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

  Il vantaggio sta nel fatto che questo tipo di normalizzazione mantiene la relazione presente nei dati originali ma è presente lo svantaggio dovuto all'elevata sensibilità alla presenza di outlier.

- **Z-score**: i valori di un attributo A vengono normalizzati basandosi sulla media e sulla deviazione standard. Il valore v_i di A è normalizzato in v_i', calcolando:

$$v_i' = \frac{v_i - \bar{A}}{\sigma_A}$$

dove $\bar{A}$ rappresenta la media e σ_A la deviazione standard dell'attributo A. Questo metodo, a differenza del max-min, è meno sensibile alla presenza di outlier ed è facilmente utilizzabile quando il valore massimo e quello minimo di un attributo non sono noti.

2.3 Data Mining

Dopo la fase di *Data Preparation*, che ha permesso di ottenere un dataset pulito, si passa all'esecuzione del terzo blocco principale del *framework*, ovvero quello di *Data Mining*. Questa fase corrisponde ad una parte dello step presente nel KDD in cui avviene la scelta dell'algoritmo di *data mining* da utilizzare (clustering, regole di associazione) e la sua successiva implementazione. Il *framework* proposto integra l'algoritmo di *clustering* **K-Means** [27], del quale si discuterà in dettaglio all'interno di questa sezione. Un obiettivo del *data mining* è quello della scoperta, ovvero il processo che permette di trovare conoscenza utile attraverso l'estrazione di *pattern*, il quale può essere a sua volta suddiviso in [16]:

1. **Predizione**: ovvero la scoperta di nuovi *pattern* che permettono la predizione di comportamenti futuri delle variabili d'interesse

2. **Descrizione**: ovvero la scoperta di *pattern* con lo scopo di rappresentare all'utente finale la conoscenza in modo comprensibile

Questi due obiettivi possono essere perseguiti utilizzando una serie di tecniche di *data mining*, le quali possono essere categorizzate in:

- **Classificazione**: consiste nel *trainare* un modello che permette di mappare (classificare) un certo dato con una delle classi che sono state definite

- **Regressione**: consistere nel trainare un modello che permette di mappare un certo dato su una variabile di tipo continuo

- **Clustering**: permette di raggruppare elementi simili tra loro ma diversi da altri presenti in altri raggruppamenti

- **Summarization**: comprende metodi che permettono di trovare una descrizione compatta dei dati

- **Dependency modeling**: consiste nel trovare un modello che descriva le dipendenze funzionali più significative tra le variabili

- **Change and deviation detection**: si concentra nel trovare i cambiamenti più significativi nei dati rispetto ad una precedente misurazione o normalizzazione dei valori

Di seguito verranno descritte le tecniche utilizzate all'interno del *framework*.

2.3.1 Clustering

Gli algoritmi di *clustering* fanno parte delle tecniche di tipo *unsupervised*, le quali non prevedono un tipo di conoscenza a priori sui dati ma cercano di estrarre della conoscenza utile attraverso la scoperta di *pattern* significativi. Lo scopo principale del *clustering* è quello di partizionare un insieme di dati in gruppi. Ogni gruppo rappresenta un cluster, formati in modo che gli oggetti presenti nel cluster siano simili tra di loro e diversi dagli oggetti presenti negli altri cluster. In altre parole, questo algoritmo cerca di minimizzare la distanza intra-cluster e di massimizzare quella inter-cluster, come mostrato in Figura 2.9.

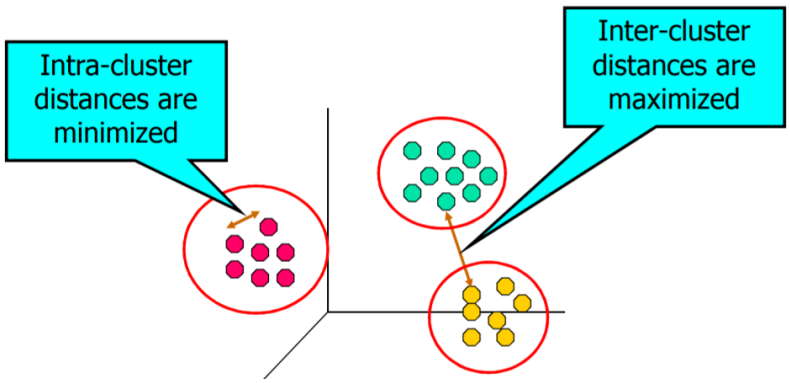

Figura 2.9: Obiettivo del clustering ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

Concetto di distanza

Prima di procedere con la caratterizzazione dei diversi algoritmi di clustering esistenti è utile definire il concetto di distanza citato in precedenza, necessario per poter capire quanto un oggetto è simile (vicino) ad un altro. Di seguito vengono descritte alcune misure utilizzate per calcolare la distanza [26]:

- **Euclidean Distance**: è la misura più diffusa ed è utilizzata per confrontare dati di tipo numerico. Definiti due oggetti $i = (x_{i1}, x_{i2}, ..., x_{ip})$ e $j = (x_{j1}, x_{j2}, ..., x_{jp})$ descritti da p attributi numerici, la distanza euclidea tra gli oggetti i e j è definita come:

$$d(i,j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + ... + (x_{ip} - x_{jp})^2}$$

- *Manhattan Distance (o city block)*: chiamata così perchè rappresenta la distanza in blocks (quartieri) tra due punti della città. Essa è definita come:

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + ... + |x_{ip} - x_{jp}|$$

- *Minkowski Distance*: è la generalizzazione della distanza Euclidea e della Manhattan. Definito h come un numero reale tale che $h \geq 1$, essa si calcola come:

$$d(i,j) = \sqrt{|x_{i1} - x_{j1}|^h + |x_{i2} - x_{j2}|^h + ... + |x_{ip} - x_{jp}|^h}$$

- *Cosine Distance*: è una misura calolata a partire dalla *Cosine Similarity*, usata principalmente con dati di tipo testuale. In particolare, dati due vettori x ed y, la *Cosine Similarity* è definita come:

$$sim(x,y) = \frac{x * y}{||x|| * ||y||}$$

  dove $||x||$ è la norma euclidea del vettore x definita come: $\sqrt{x_1^2 + x_2^2 + ... + x_p^2}$. Un valore pari a 0 indica che i due vettori sono ortogonali, mentre un valore vicino ad 1 indica un'elevata similarità tra i due vettori. Detto ciò, la *Cosine Distance* è calcolata come $1 - sim(x,y)$.

Categorie di algoritmi

In letteratura esistono diversi algoritmi di *clustering* e, per questo motivo, è difficile fare una netta categorizzazione dei vari metodi. In generale i più diffusi algoritmi possono essere divisi in quattro categorie [26]:

- **Partizionali**: dato un insieme di dati di dimensione n, un algoritmo di tipo partizionale (Figura 2.10) crea k partizioni di dati, dove ogni partizione rappresenta un cluster e contiene un sottoinsieme dei dati ($k \leq n$). La maggior parte dei metodi

partizionali effettuano una separazione dei cluster esclusiva, ovvero assegnano un'osservazione solo ed esclusivamente ad un cluster. Esistono però anche algoritmi partizionali di tipo *fuzzy*, per i quali ogni punto può appartenere contemporaneamente a più gruppi.

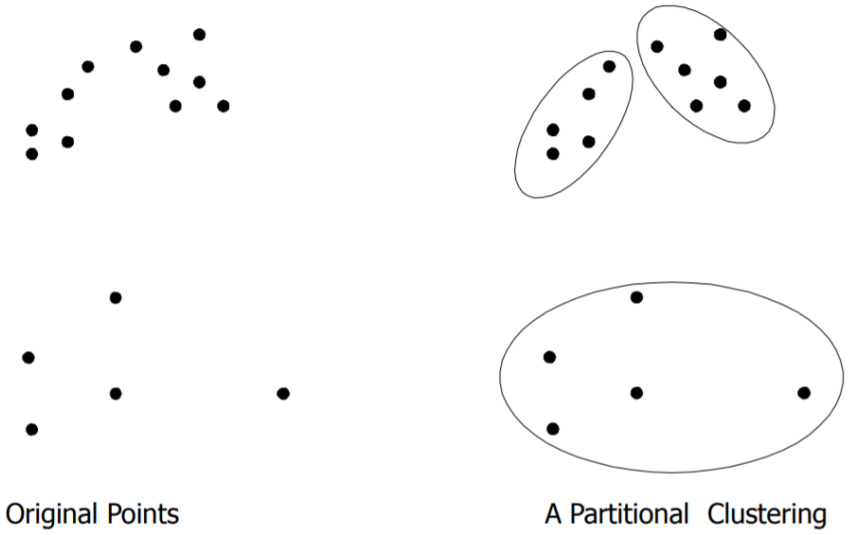

Figura 2.10: Esempio di clustering partizione con k=2 e k=1 ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

- **Gerarchici**: creano una decomposizione gerarchica del *dataset*, producendo un insieme di *cluster* annidati (Figura 2.11). Questi cluster possono essere visualizzati attraverso un dendrogramma, ovvero un diagramma che mostra la sequenza di aggregazioni (o *split*). Questi algoritmi possono essere ulteriormente classificati in:

  – Agglomerativi: iniziano individuando come *cluster* il singolo punto e aggregano ad ogni step le coppie di *cluster* più vicine finchè non si ottiene un unico *cluster*

  – Divisivi: iniziano individuando un unico *cluster* contenente l'insieme di tutti i dati per poi splittare ad ogni *step* i *cluster* finchè non si ottiene un numero di *cluster* pari al numero di dati disponibili

  Per capire quando fare il *merge* o lo *split* dei cluster, gli algoritmi di tipo gerarchico si basano sul concetto di similarità inter-cluster, la quale può essere calcolata utilizzando diversi metodi come il *max*, il *min*, la *group average*, il *ward* o altri [28]

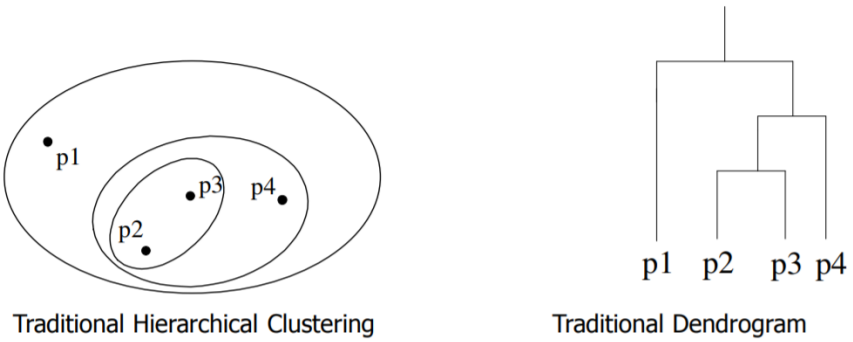

Figura 2.11: Esempio di clustering gerarchico e sua rappresentazione su dendrogramma ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

- **Density-based**: l'idea alla base di questo tipo di algoritmo è quella di continuare ad aumentare un certo cluster finchè la densità (numero di oggetti) in un intorno (neighborhood) supera una certa soglia. Esempio è il DBSCAN di cui si è discusso in una sezione precedente
- **Grid-based**: questi metodi dividono lo spazio in un numero finito di celle in modo da formare un sorta di griglia. Il vantaggio di questo approccio sta nella velocità di esecuzione, la quale è indipendente dal numero di oggetti presenti e dipendente solo dal numero di celle presenti

Una disamina sulle caratteristiche generali di queste quattro categorie di algoritmi è data nella Tabella 2.1.

Un algoritmo partizionale: il K-means

L'algoritmo partizionale **K-Means** [26] viene utilizzato all'interno del *framework* in modo da consentire l'individuazione di gruppi di certificazioni energetiche che abbiano caratteristiche comuni. Supponendo di avere un dataset D di numerosità n, i metodi partizionali come il K-Means distribuiscono gli oggetti in k cluster, $C_1, C_2, ..., C_k$ (con k definito a priori) in modo tale che $C_i \subset D$ e $C_i \cap C_j = \emptyset$ per $(1 \leq i, j \leq k)$. Per valutare la qualità del partizionamento deve essere definita una funzione obiettivo, la quale deve portare ad avere oggetti simili tra di loro all'interno del cluster ma dissimili dagli oggetti presenti negli altri cluster. In altre parole questa funzione obiettivo deve minimizzare la distanza (massimizzare la similarità) intra-cluster e massimiazzare quella inter-cluster (minimizzare la similarità). Il K-Means è un tipo di algoritmo partizionale del tipo *centroid-base*, ovvero utilizza il concetto di centroide per effettuare il partizionamento. Il centroide rappresenta

Metodo	Caratteristiche generali
Metodi partizionali	• Trovano cluster mutualmente esclusivi di forma sferica • Basati sul concetto di distanza • Utilizzano media (o altre misure) per rappresentare il centro del cluster • Efficaci per dataset medio-piccoli
Metodi gerarchici	• Generano cluster annidati • Non è possibile la correzione di merge o split sbagliati
Metodi denisity-based	• Trovano cluster di forma arbitraria • I cluster sono regioni dense di oggetti separate da zone a più bassa densità • *Densità del cluster*: ogni punto deve avere un numero minimo di punti entro il suo *neighborhood* • Possibile utilizzo per outlier detection
Metodi grid-based	• Utilizzano una struttura dati 'a griglia' • Tempi di processamento piccoli (indipendenti dalla dimensione del dataset)

Tabella 2.1: Prinicipali caratteristiche delle quattro categorie di clustering [26]

il punto centrale del cluster il quale può essere definito in diversi modi, ad esempio come la media dei punti assegnati al cluster. La differenza tra un certo punto p appartenente ad un cluster C_i e il suo centroide c_i è misurato dalla distanza euclidea ed indicato come $dist(p, c_i)$. La qualità di un cluster C_i può essere misurata osservando la somma dei quadrati degli errori (SSE), calcolata come:

$$SSE = \sum_{i=1}^{k} \sum_{p \in C_i} dist(p, c_i)^2$$

Dal punto di vista computazionale, ottimizzare l'SSE è un compito gravoso. È infatti possibile dimostrare che il problema è di tipo NP-hard ed in particolare la complessità è $O(n^{dk+1}log(n))$, dove k è il numero di cluster, d la dimensionalità dello spazio ed n il numero di punti presenti. Per sopperire a questa difficoltà computazionale, nella pratica si utilizzano altri approcci tra cui il più utilizzato è proprio il K-Means. L'algoritmo K-Means definisce il centroide di un cluster come il valore medio dei punti in esso contenuti. Inizialmente, definto k ovvero il numero di cluster in cui si vuole partizionare il dataset, l'algoritmo sceglie in maniera random k punti all'interno del dataset D, ognuno dei quali rappresenta il centroide di ogni cluster. Ciascun punto viene poi asseganto al cluster più vicino, in cui la vicinanza viene determinata attraverso il calcolo della distanza euclidea tra il punto e i k centroidi. Una volta conclusa questa prima fase, il K-Means iterativamente ripete il processo ricalcolando il nuovo centroide del cluster e riassegnando tutti i punti ai vari cluster nello stesso modo. Questo processo si ripete finchè il partizionamento risulta stabile, ovvero fino a quando vi sono pochi punti che si spostano da un cluster all'altro in due iterazioni successive. L'algoritmo può essere descritto sinteticamente nel seguente modo:

Algorithm 1 K-means

1: Seleziona K punti random come centroidi iniziali
2: **repeat**
3: Forma K cluster assegnando tutti i punti al centroide più vicino
4: Ricalcola il nuovo centroide per ogni cluster
5: **until** I centroidi non variano

La scelta iniziale del centroide avviene in maniera randomica; ciò significa che lanciando più volte l'algoritimo sugli stessi dati questo genera risultati differenti. Un esempio viene dato nelle Figure 2.12 e 2.13, in cui si mostra che eseguendo due volte lo stesso algoritmo sugli stessi dati si hanno risultati diversi proprio a causa della diversa scelta dei centroidi al primo passo. L'algoritmo K-Means non garantisce la convergenza verso un ottimo globale e spesso questo dà invece come risultato un ottimo locale (Figura 2.14). Anche questo risultato dipende dalla scelta iniziale dei centroidi e, nella pratica, l'algoritmo viene eseguito un diverso elevato di volte con diversi centroidi iniziali. A differenza di altri algortimi partizionali, il K-means è molto scalabile ed efficiente nel processare grandi collezioni di dati. La sua complessità è infatti dell'ordine di $O(nkt)$, dove n è il numero

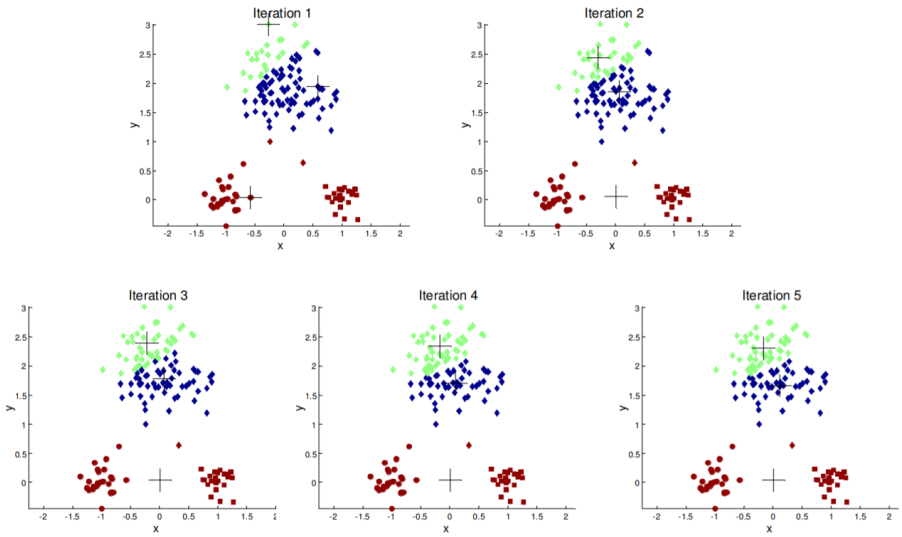

Figura 2.12: Esempio 1 - Effetto della scelta random dei centroidi ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

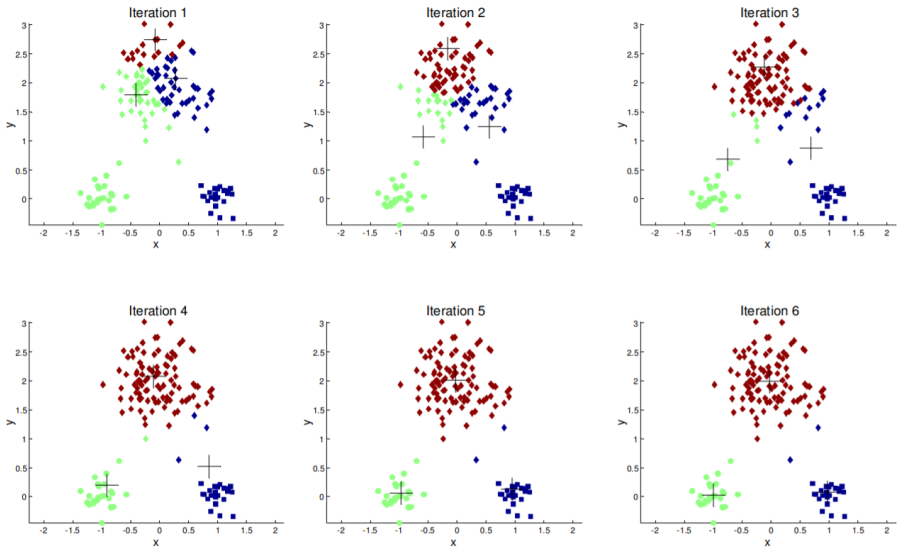

Figura 2.13: Esempio 2 - Effetto della scelta random dei centroidi ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

totali di oggetti presenti nel dataset, k il numero di cluster e t il numero di iterazioni e solitamente accade che $k << n$ e $t << n$.

Un altro svantaggio di questo algoritmo è dato dalla scarsa capacità di individuare

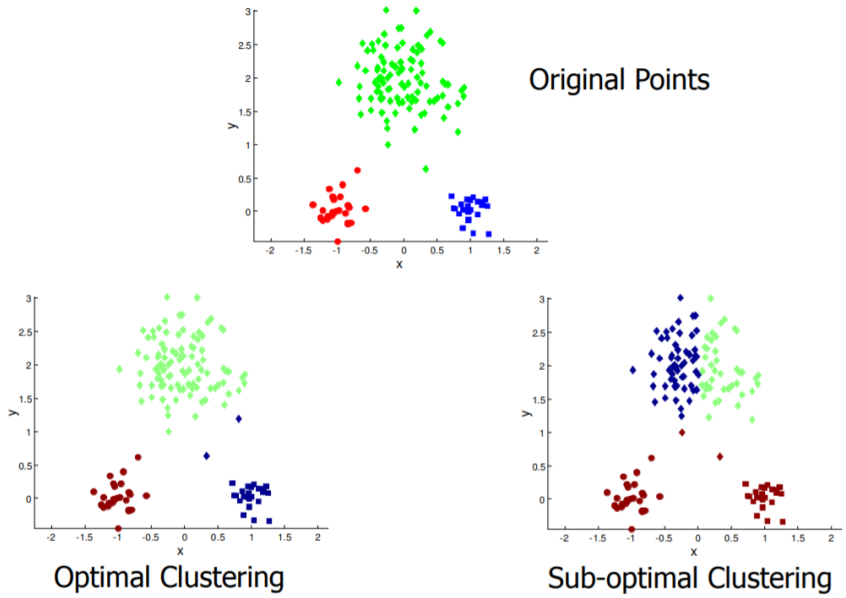

Figura 2.14: Confronto tra soluzione sub-ottimale ed ottimale ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

cluster di forma non globulare (Figura 2.16) o cluster di dimensioni differenti (Figura 2.15). Inoltre, essendo il centroide definito come la media dei punti appartenenti ad un cluster, l'algoritmo è fortemente influenzato dalla presenza di outlier e di rumore nei dati. Una possibile soluzione al problema è quella basata sull'idea del microclustering, che consiste nello scegliere un k elevato in modo da generare molti cluster, per poi riaggregarli tra loro. La necessità di stabilire a priori il numero di cluster k può essere visto come un

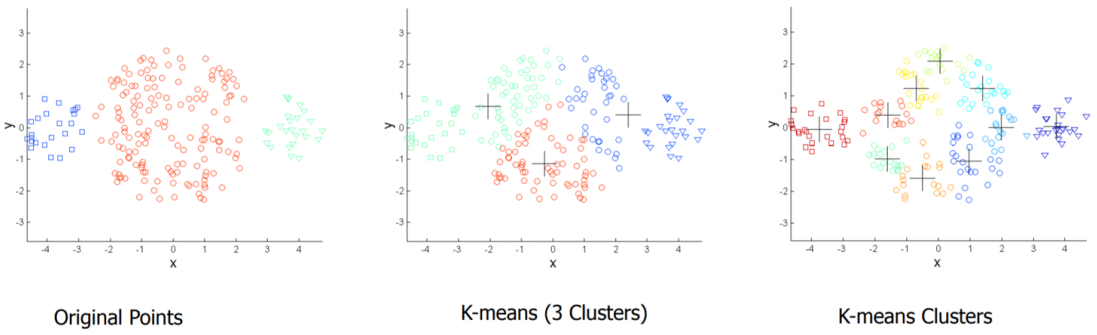

Figura 2.15: Clustering di dimensione diversa con k piccolo e con k più elevato ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

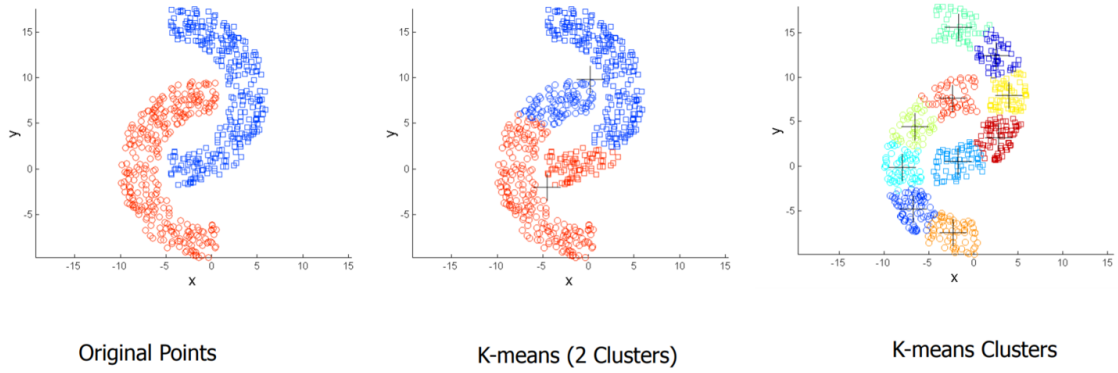

Figura 2.16: Clustering non globulari con k piccolo e con k più elevato ©Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

ulteriore svantaggio. Esistono in letteratura diverse tecniche che permettono di stimare quale sia il miglior valore di k, tra cui l'*Elbow Method* [29], il quale prende in considerazione la percentuale di variabilità spiegata in funzione del numero di cluster, attraverso una rappresentazione grafica di queste informazioni. Guardando all'esempio in Figura 2.17, si nota che i primi cluster ci danno molta informazione ma, aumentando man mano k, ad un certo punto il guadagno di informazioni in termini marginali decresce in maniera significativa, dando origine ad un 'gomito' (elbow) nel grafico. Il valore di k da scegliere si trova dunque in corrispondenza del gomito. La misura utilizzata per valutare la bontà del clustering è in questo caso l'SSE, il quale diminuisce all'aumentare di k.

2.3.2 Feature selection

Il *framework* è in grado di capire in maniera automatica se effettuare o meno più livelli di *clustering*, applicando l'algoritmo partizionale K-Means. Se alcuni *cluster* non sono abbastanza puri dal punto di vista del loro contenuto, il *framework* esegue un ulteriore clusterizzazione su questi, scegliendo in maniera automatica gli attributi più significativi da utilizzare facendo un *subset* di quelli utilizzati precedentemente alla prima iterazione. Per la scelta delle variabili è stato utilizzata la funzione *hclustvar* contenuta all'interno del pacchetto *ClustOfVar*[31] di *R*[32]. Questa funzione permette di effettuare un *clustering* di tipo gerarchico tra le variabili, le quali possono essere sia quantitative sia qualitative. Il criterio di aggregazione è basato sulla diminuzione dell'omogeneità dei *cluster* aggregati. L'omogeneità di un *cluster* è definita come la somma del *correlation ratio* e della *squared correlation* tra le variabili e il centro del *cluster*, identificato dalla prima componente principale del *PCAmix*. Il PCAmix include una normale *principal component analysis*

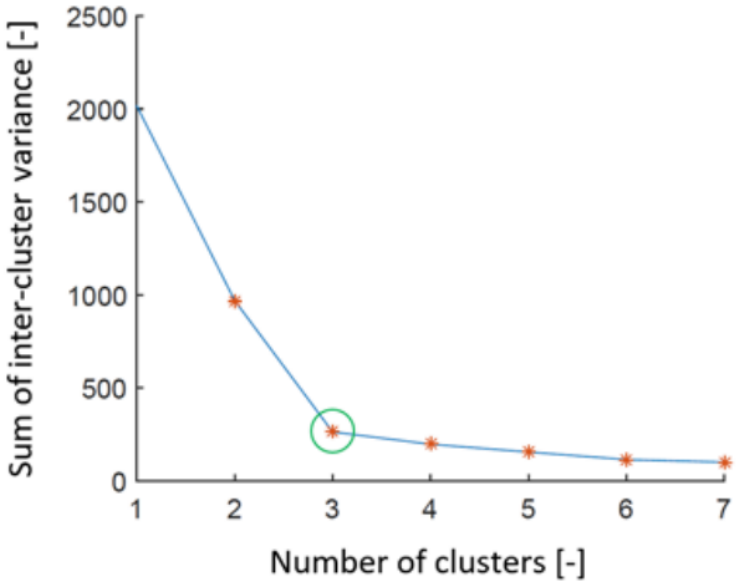

Figura 2.17: Elbow graph con k scelto pari a 3 ©Štrobl, Piorecký, Krajča [30]

(PCA) e una *multiple correspondence analysis* (MCA). Una volta che la funzione *hclustvar* ha generato il dendrogramma, per valutare quanti *cluster* bisogna tenere in considerazione è stato utilizzato lo *stability plot*. In particolare viene calcolato il *correct Rand-index* tra le varie coppie di *cluster* e plottato poi il valore in funzione del numero di *cluster*. Il numero di gruppi da considerare si ottiene in corrispondenza del valore massimo. Un esempio di dendrogramma e di *stability plot* viene illustrato in Figura 2.18. In questo esempio, dallo *stability plot* è stato determinato 5 come numero ottimale di *cluster* che si ottiene tagliando il dendrogramma all'altezza della linea tratteggiata. Se più variabili sono presenti all'interno dello stesso *cluster*, viene selezionata quella che presenta il valore di correlazione più elevato.

2.4 Knowledge Interpratation

In questa sezione viene descritta l'ultima parte principale del *framework*, ovvero quella relativa all'interpretazione della conoscenza ottenuta dal blocco precedente. In particolare è presente una fase dedicata alla *Knowledge Characterization*, il cui scopo principale è quello di caratterizzare attraverso alcuni strumenti la conoscenza, ed una fase dedicata alla *Knowledge Visualization*, attraverso una rappresentazione su mappa dei risultati ottenuti.

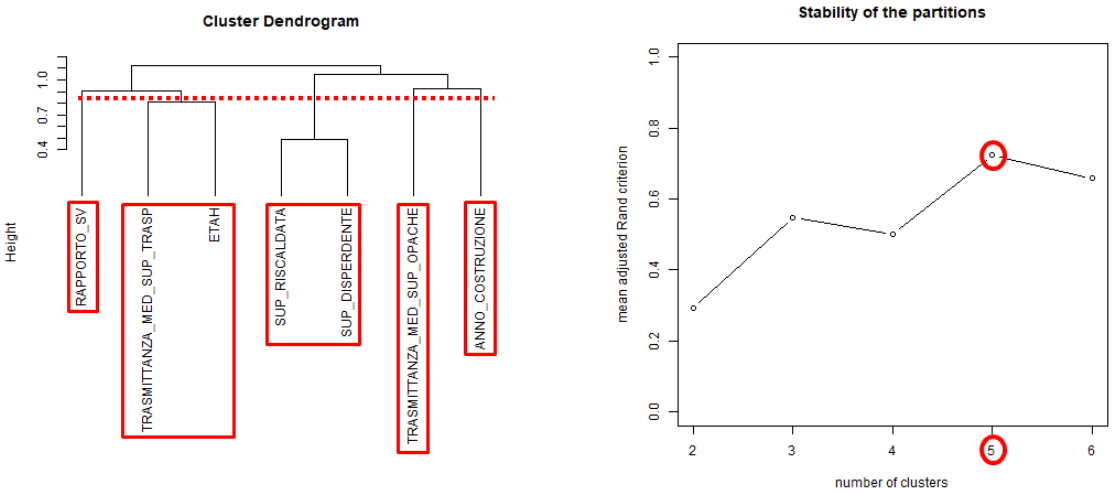

Figura 2.18: Dendrogramma e *stability plot* ottenuti con il pacchetto *ClustOfVar*

2.4.1 Knowledge Characterization

Il *framework* sviluppato utilizza gli *alberi di decisione*, ed in particolar modo il **CART (Classification and Regression Tree)** [33], a valle del clustering in modo da fornire una migliore caratterizzazione dei cluster generati. È infatti possibile utilizzare un *Decision Tree Classifier* non solo per classificare i dati ma anche per estrarre delle regole del tipo IF-THEN le quali sono facilmente comprensibili dai non esperti di dominio, consentendo di fare analisi multivariate. Il *framework* utilizza anche i *boxplot* [34] come metodo di caratterizzazione della conoscenza, i quali permettono agli esperti di dominio di analizzare la distribuzione delle variabili all'interno del cluster.

Viene di seguito fatta un'introduzione agli alberi di decisione e ai *boxplot*.

Alberi di decisione

Un albero di decisione è una struttura ad albero assimilabile ad un *flowchart*, nel quale ogni nodo interno denota un test su un attributo, ogni ramo dell'albero il risultato del test su quel nodo ed ogni nodo foglia (detto anche nodo terminale) contiene l'etichetta di classe; il primo nodo dell'albero è detto nodo radice. Per la costruzione dell'albero di decisione, sono necessari alcuni parametri come la partizione di dati D da utilizzare, la lista degli attributi che descrivono le varie tuple ed un'euristica per la selezione dell'attributo da utilizzare come test ad ogni nodo. Il CART, così come gli algoritmi ID3 e C4.5, utilizza un approccio *greedy*, in cui gli alberi di decisioni vengono costruiti in maniera ricorsiva

secondo una metodologia top-down. Dato un training set D, un algoritmo di generazione dell'albero viene di seguito presentato:

Algorithm 2 GeneraDecisionTree

1: Crea un nodo N
2: **if** le tuple in D appartengono tutte alla stessa classe C
3: **then** ritorna N come nodo foglia etichettato con la classe C
4: **if** la *listaAttributi* è vuota
5: **then** ritorna N come nodo foglia etichettato con la majority class in D
6: applica $MetodoSelezioneAttributi(D, listaAttributi)$ per trovare il migliore $CriterioDiSplit$
7: etichetta il nodo N con il $CriterioDiSplit$
8: **if** *AttributoDaSplittare* è di tipo discreto sono possibili più split
9: $ListaAttributi \leftarrow ListaAttributi - AttributoDaSplittare$
10: **for all** risultato j di $CriterioDiSplit$ **do**
11: sia D_j l'insieme delle tuple in D che soddisfano j
12: **if** D_j è vuoto
13: **then** attacca una foglia al nodo N etichettata con la majority class in D
14: **else** attacca il nodo generato da $GeneraDecisionTree(D_j, ListaAttributi)$ al nodo N
15: **end for**
16: ritorna N

La complessità computazionale di un algoritmo del genere è dell'ordine di $O(n * |D| * log(|D|))$. Un requisito fondamentale per un algoritmo decisionale è la scelta dell'attributo sui cui fare splitting ad ogni nodo. Possono essere utilizzate diverse euristiche per stabilire un criterio di splitting come l'*Information Gain*, il *Gain Ratio* o il *Gini Index*. Il CART (esempio in Figura 2.19) fa uso di quest'ultima misura, di cui viene di seguito proposto un approfondimento.

Gini Index

Sia D la partizione di dati su cui fare il modello. Supponiamo che l'etichetta di classe possa assumere m valori distinti, che definiscono m distinte classi $(C_1, ..., C_i, ...C_m)$. Sia $C_{i,D}$ l'insieme delle tuple di classe C_i in D, $|D|$ il numero di tuple in D e $|C_{i_D}|$ il numero di tuple in $C_{i,D}$, il *Gini Index* [35] è definito come:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2,$$

dove p_i rappresenta la probabilità che una tupla in D appartenga alla classe C_i, la quale è stimata come: $|C_{i,D}|/|D|$. Dato un attributo di tipo continuo, lo split point è dato

dal valore che minimizza il *Gini Index*. Per scegliere gli *split point* da confrontare, i valori dell'attributo A considerato vengono prima ordinati in maniera crescente e conseguentemente viene preso come possibile valore di *split* il punto medio calcolato tra ogni coppia di valori adiacenti:

$$ValoreDiSplit = \frac{a_i + a_{j+1}}{2},$$

Se v sono i valori assunti da A, allora verranno valutati $v - 1$ possibili split point.

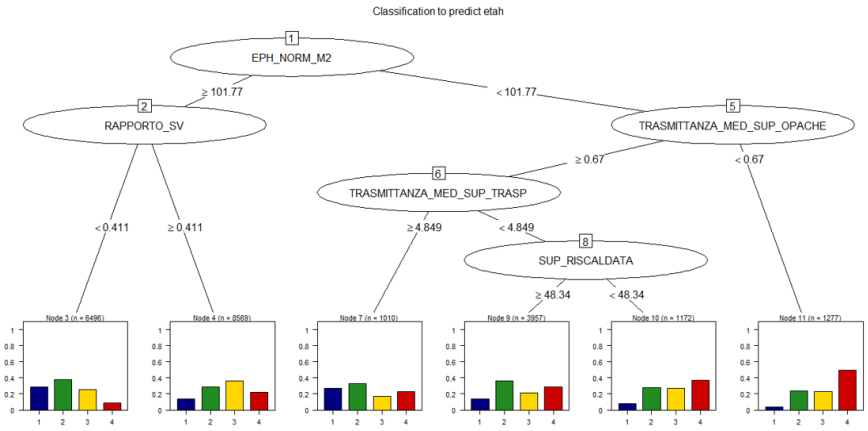

Figura 2.19: Esempio di albero di decisione (CART)

Scopo della classificazione

Lo scopo della classificazione è quello di predire una certa classe (o *label*), attraverso la costruzione di un classificatore. La *Data Classification* è un processo che si può suddividere in due fasi:

- Learning (o training): il classificatore viene costruito sulla base di un *training set*, ovvero sull'insieme di righe del dataset ad ognuna delle quali è associata una *label* (o etichetta di classe) categorica. Questa tecnica è dunque di tipo supervised in quanto l'etichetta di classe è conosciuta

- Classification: il classificatore costruito nella fase precedente viene utilizzato per predire l'etichetta di classe dato un insieme di record nuovo

Validazione del modello

Prima di utilizzare il modello come classificatore è necessario sapere quanto questo sia accurato e affidabile; bisogna quindi procedere alla sua validazione. Validare il modello sull'insieme di dati utilizzati per costruire il classificatore (ovvero sul *training set*), porterebbe a risultati sbagliati a causa della sua eccessiva specializzazione e, per questo motivo, si fa spesso uso di un *test set*. Solitamente il dataset disponibile viene infatti diviso in training set e test set, nel quale il primo viene utilizzato per costruire il modello e il secondo per validarlo.

Prima di procedere con la descrizione di queste tecniche, vengono di seguito presentate alcune definizioni utili alla comprensione. Data una certa classe di interesse (ad esempio *classe = 1*), la tuple contenenti questo tipo di informazione vengono dette positivi (P) mentre tutte le altre negativi (N); in questo modo è possibile confrontare l'etichetta di classe predetta dal classificatore con quella conosciuta e servirsi di una serie di misure per valutare l'accuratezza del modello. Si definisce:

- **Vero positivo** (TP): il numero di righe etichettate come P che sono state classificate in maniera corretta
- **Vero negativo** (TN): il numero di righe etichettate come N che sono state classificate in maniera corretta
- **Falso positivo** (FP): il numero di righe etichettate come N ma che sono state erroneamente classificate come P
- **Falso negativo** (FN): il numero di righe etichettate come P ma che sono state erroneamente classificate come N

Questi dati vengono riassunti nella cosiddetta matrice di confusione (esempio in Figura 2.20), attraverso la quale è possibile valutare l'affidabilità del modello. Guardando invece alla misure di performance si definisce l'accuratezza come:

$$Accuratezza = \frac{TP + TN}{P + N}$$

la quale dà un'indicazione sulla percentuale di tuple che sono state correttamente classificate dal classificatore. In alcune applicazioni l'utilizzo dell'accuratezza non è sufficiente, ma si affiancano altre due misure come la precisione e il richiamo:

$$Precisione = \frac{TP}{TP + FP}; Richiamo = \frac{TP}{TP + FN}$$

In particolare la precisione indica la percentuale di righe che sono state etichettate come P e che erano veramente tali, mentre il richiamo come la percentuale di righe etichettate come P ed assegnate come tali dal classificatore. Un altro modo di esaminare precisione e richiamo è quella di utilizzarli in maniera congiunta per definire un'unica misura chiamata F-score, definita come:

$$F = \frac{2 * precisione * richiamo}{precisione + richiamo}$$

Predicted class

Actual class		*yes*	*no*	Total
	yes	*TP*	*FN*	*P*
	no	*FP*	*TN*	*N*
	Total	P'	N'	$P + N$

Figura 2.20: Esempio di matrice di confusione ©Han, Kamber, Pei [26]

Per quanto detto precedentemente per misurare l'accuratezza del modello si utilizza un test set, composto da tuple con la rispettiva label associata che non sono state utilizzate per costruire il classificatore. Esistono diverse tecniche di partizionamento dei dati come:

- **Holdout**: i dati vengono partizionati in maniera randomica in due insiemi indipendenti, tipicamente il 75% dei dati va a finire nel training set e il restante 25% nel test set. Un esempio di holdout è riportato in Figura 2.21
  - Random subsampling: variante dell'holdout il quale viene ripetuto k volte. L'accuratezza è data dalla media delle accuratezze ottenute ad ogni iterazione
- **Cross-validation (k-fold)**: i dati vengono partizionati in maniera randomica in k fold (sottoinsiemi) disgiunti, mutualmente esclusivi e di dimensione approssimativamente uguale. Il training e il testing vengono eseguiti k volte, tenendo da parte un fold per il test ed utilizzando i rimanenti come training
  - Leave-one-out: caso speciale del k-fold in cui k è il numero iniziale di righe e ad ogni iterazione solo una tupla viene tenuta da parte per effettuare il test

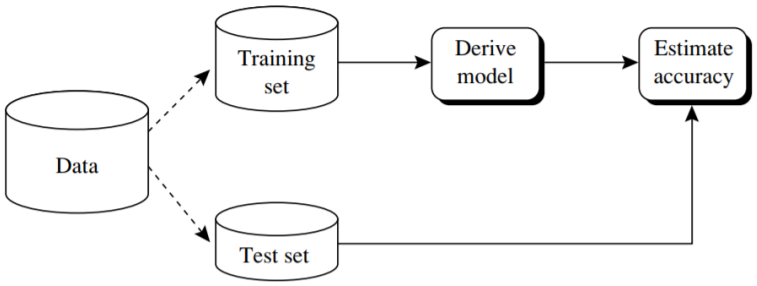

Figura 2.21: Esempio di validazione con holdout ©Han, Kamber, Pei [26]

Boxplot

Il *boxplot* [34] è un metodo grafico che permette di visualizzare la distribuzione dei dati attraverso i suoi quartili. In particolare sono indicati la posizione del primo (Q1) e del terzo quartile (Q3), corrispondenti rispettivamente al segmento inferiore e al segmento superiore del rattangolo mostrato in Figura 2.22 (il quale contiene il 50% dei dati), la mediana (Q2) e i valori massimi e minimi definiti rispettivamente come:

$$max(val) = Q3 + 1.5 * IQR; min(val) = Q1 - 1{,}5 * IQR$$

IQR rappresenta il cosiddetto range interquartile definito come: $Q3 - Q1$. Tutti quei valori che si trovano al di sotto del valore minimo o al di sopra del valore massimo sono considerati outlier.

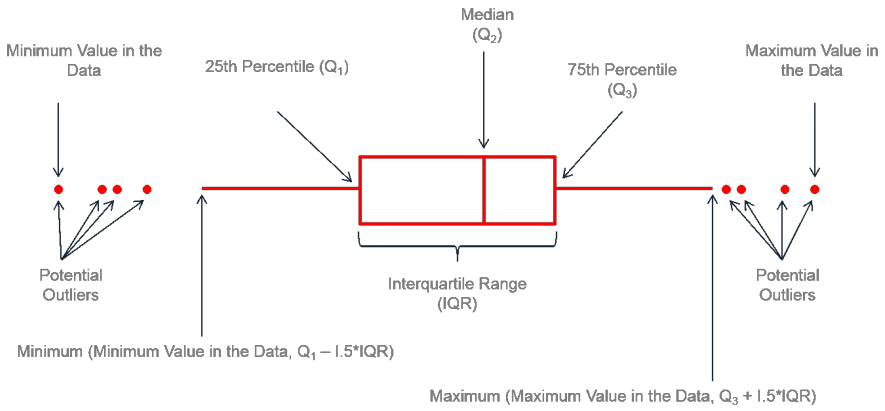

Figura 2.22: Esempio di boxplot ©leansigmacorporation.com

2.4.2 Knowledge Visualization

Lo scopo di questa fase è quello di permettere la visualizzazione della conoscenza estratta in modo da renderla facilmente fruibile anche ai non esperti di dominio, attraverso delle mappe consultabili a diversi livelli di granularità spaziale (città, circoscrizione, quartiere e singola unità abitativa). I dati presenti nel DB contengono tutte le informazioni rilevanti ai fini della geolocalizzazione come le coordinate (latitudine e longitudine), la circoscrizione di appartenenza, l'indirizzo e il numero civico dell'unità immobiliare oggetto di attestazione. Per la visualizzazione vengono utilizzate diverse mappe geolocalizzate come (i) le *mappe coropletiche*, (ii) le mappe di tipo *scatter* e (iii) le mappe di tipo *cluster-marker*. Queste tecniche di visualizzazione vengono utilizzate congiuntamente dal *framework* per fornire una visualizzazione dinamica della conoscenza estratta come risultato dell'applicazione dei blocchi precedenti. Vengono di seguito esposte nel dettaglio le varie tipologie di mappe.

Mappe coropletiche

Una mappa coropletica mostra delle aree geografiche che sono colorate in base al valore assunto da una certa variabile. La prima mappa di questo genere fu creata da Charles Dupin nella prima metà dell'800 [36], ma il termine *choroplet* fu introdotto per la prima volta da John Kirtland Wright circa un secolo dopo. Questo tipo di mappe presentano il vantaggio di rappresentare un gran numero di dati in modo semplice ed intuitivo anche se la visualizzazione realistica dei dati all'interno di una certa area viene compromessa [37]. Un esempio di mappa coropletica è mostrato in Figura 2.23. Per quanto riguarda le map-

Figura 2.23: Esempio di mappa coropletica

pe generate dal *framework*, ogni area viene colorata in base al valore medio dell'attributo

considerato per l'analisi; il colore verde indica una zona in cui sono presenti edifici che presentano dei buoni valori di performance mentre il colore rosso indica una cattiva performance. Inoltre se per una data area non è presente una certificazione, o se queste sono molto poche, la zona corrispondente viene colorata di grigio. Nel caso di studio analizzato le aree considerate sono le circoscrizioni e i quartieri della città di Torino.

Mappe scatter

Una mappa di tipo scatter permette di visualizzare dati geografici come punti sulla mappa. Nel caso in esame ogni punto rappresenta la singola unità abitativa oggetto della certificazione. Questo tipo di visualizzazione riesce a fornire un'informazione molto dettagliata riguardante la posizione dell'unità immobiliare ma, come accade nel datasatet a disposizione, la numerosità dei certificati non permette una visualizzazione agevole, causando un problema di *override informativo*. Un esempio di mappa scatter è riportato in Figura 2.24.

Figura 2.24: Esempio di mappa scatter contenete APE

Mappe cluster-marker

Questo tipo di mappa è simile ad una mappa coropletica ma i certificati vengono visualizzati in maniera aggregata attraverso dei *marker dinamici* il cui colore cambia in base al valore medio dei punti considerati per l'aggregazione. Le mappe considerate precedentemente sono molto utili per visualizzare la distribuzione geografica dei certificati limitatamente ai valori assunti da una sola variabile. La visualizzazione fatta utilizzando i market-cluster supera questo problema, consentendo di rappresentare più variabili allo

stesso tempo, in seguito al loro precedente raggruppamento in cluster. Oltre al colore anche la dimensione del marker-cluster riflette la cardinalità dei dati contenuti all'interno. La dinamicità di questi marker permette di risolvere il problema dovuto alla rappresentazione di un numero molto elevato di certificati su mappa, i quali vengono mostrati nel dettaglio solo quando si fa zoom su una particolare area d'interesse. Nella Figura 2.25 viene mostrata, a diversi livelli di dettaglio, la mappa ottenuta attraverso l'utilizzo dei marker dinamici.

Figura 2.25: Esempio di mappa cluster-marker. Dettaglio città (a sinistra), circoscrizione (in alto a destra) e quartiere (in basso a destra)

Capitolo 3

Risultati sperimentali

In questo capitolo vengono presentati i risultati sperimentali ottenuti nelle varie fasi di applicazione del *framework*. In particolare si analizzeranno i risultati e il setting dei parametri per ciascuno dei blocchi principali e per i relativi sotto-blocchi costituenti. Prima di procedere con una descrizione dettagliata, una disamina sugli strumenti di sviluppo utilizzati viene di seguito presentata.

Il *framework* è stato sviluppato in Python[1], attraverso l'utilizzo di PyCharm[2] come IDE (*Integrated Development Environment*) e di svariate librerie, di cui si riporta in Tabella 3.1 una breve descrizione.

Libreria	**Descrizione**
Pandas [38]	Lettura/modifica di CSV, per mezzo di DataFrame
Matplotlib [39]	Generazione di istogrammi, boxplot, ecc...
Scikit-Learn [40]	Data Mining (clustering e feature selection)
Folium [41]	Creazione delle mappe geolocalizzate

Tabella 3.1: Overview di alcune librerie incluse nel *framework*

Altri strumenti di supporto come *Excel*[3], *RapidMiner* [42] ed *R*[32] sono stati utilizzati.

[1] python.org

[2] jetbrains.com/pycharm/

[3] products.office.com/it-it/excel

3.1 Preprocessing

In questa sezione viene descritta la metodologia impiegata per guidare la fase di *Data Integration* e di *Data Cleaning* e i relativi risultati ottenuti.

3.1.1 Data Integration

Come riportato nel Capitolo 2, a causa dell'elevato numero di dataset disponibili, si è dovuto procedere ad un'integrazione degli stessi in modo da ottenere un unico DB su cui effetture le successive analisi. I dataset contenti APE presentavano informazioni distribuite su vari file CSV, di cui si riporta un dettaglio in Tabella 3.2.

File CSV	Attributi
Caratteristiche edificio reale	73
Caratteristiche edificio di riferimento	19
Consumi edificio reale	59
Energia edificio reale	78
Dati impianti	62
Raccomandazioni interventi migliorativi	61
Altre caratteristiche edificio reale	73

Tabella 3.2: File CSV forniti dal CSI Piemonte

Tra questi è stato escluso il file relativo alle *raccomandazioni sugli interventi migliorativi*. Sui file rimanenti si è proceduto ad effettuare il *join* utilizzando l'identificativo univoco del certificato, ovvero la terna (ID_CERTIFICATORE, PROGR_CERTIFICATO ed ANNO) ottenendo un dataset contenente all'incirca 79000 record e 350 attributi. Prima di procedere all'integrazione delle basi dati, un *rename* ed una *feature selection* è stata effettuata. In particolare si è effettuato un rename degli attributi in modo che risultassero coerenti al DB più recente e presi poi in considerazione solo quegli attributi in comune tra tutti i *dataset*. Per ridurre la complessità analitica e semplificare la successiva caratterizzazione dei risultati, l'analisi si è concentrata sulle unità abitative presenti nella città di *Torino* con *destinazione d'uso E1(1)*, cioè abitazioni adibite a residenza con carattere continuativo, che rappresentano quasi il 90% delle destinazioni d'uso presenti (Figura 3.1), e sui dati relativi alla climatizzazione invernale[4].

[4]L'insieme di funzioni atte ad assicurare, durante il periodo di esercizio dell'impianto termico, il benessere degli occupanti mediante il controllo, all'interno degli ambienti, della temperatura

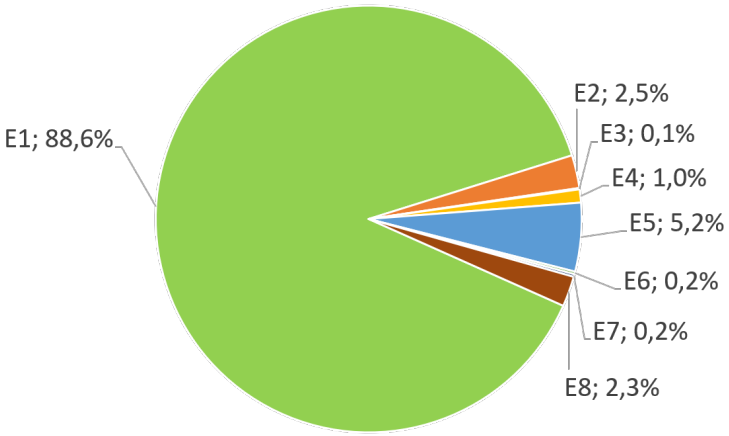

Figura 3.1: Distribuzione delle destinazioni d'uso all'interno del dataset

La concatenazione fra la base dati contenente gli ACE e quella contenente gli APE ha prodotto un *dataset* unico comprendente circa 50000 certificati, ognuno dei quali caratterizzato da più di 50 attributi. A questo punto il problema relativo alla presenza di più certificazioni relative allo stesso appartamento è stato risolto creando un unico gruppo per foglio, particella e subalterno (vedi sezione 2.1.1) e tenuto quello con data di caricamento maggiore, la cui applicazione ha portato a scartare circa il 2% delle certificazioni.

3.1.2 Address resolution

Come ribadito nella sezione 2.1.2, la correttezza degli attributi necessari a caratterizzare la posizione di un certo edificio è un aspetto critico nel caso in cui i certificati debbano essere visualizzati su mappa. Il *dataset* analizzato presentava diversi problemi relativi all'indirizzo e al CAP, il quale assumeva spesso il valore generico 10100 (non più in vigore). Per le certificazioni ACE, precedenti dunque al 2015, i campi relativi al CAP e alle coordinate non erano presenti. Per queste certificazioni i campi mancanti sono stati creati e successivamente popolati grazie all'applicazione dell'algoritmo di *Address Resolution*, il quale confronta l'indirizzo del *dataset* con quella presente all'interno del viario, permettendo di calcolare l'indice di similarità, contenuto all'interno del campo *Lev*. Se questo valore è maggiore di 0.9, l'indirizzo corrispondente del viario viene sostituito a quello presente nel DB, così come le altre informazioni relativa al CAP, alle coordinate e al numero civico. Considerando unicamente quei certificati che presentavano un valore per l'attributo CAP,

è possibile notare come l'applicazione dell'algoritmo abbia risolto il problema dovuto alla presenza del CAP generico (Figura 3.2).

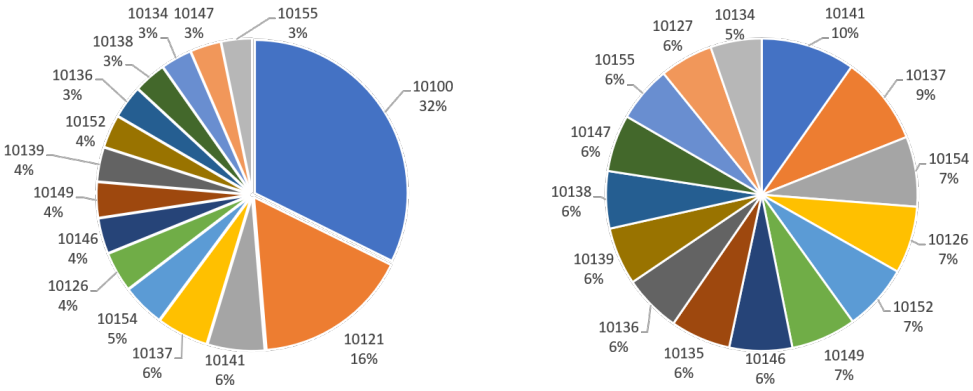

Figura 3.2: Distribuzione dei 15 CAP più frequenti prima della pulizia (a sinistra) e dopo la pulizia (a destra)

Un esempio di risoluzione degli indirizzi è presentato nell Tabella 3.3. Il campo *Indirizzo Input* e *Civico Input* indicano rispettivamente il campo indirizzo e il campo numero civico presenti all'interno del *database*. Grazie all'applicazione dell'algoritmo di *Address Resolution* oltre il 99% degli indirizzi sono stati risolti.

Indirizzo Input	**Civico Input**	**Lev**	**Indirizzo Output**	**Civico Output**
GABETTI 12	-	1	CORSO GIUSEPPE GABETTI	12
VIA SACCHI 2	-	1	VIA PAOLO SACCHI	2
VIA G. PRATI 1	1	0,941	VIA GIOVANNI PRATI	1
CORSO CIRIE'	2	0,952	CORSO CIRIE'	2
VIA BORSI GIOSUE'	88	0,965	VIA GIOSUE' BORSI	88
C.SO GROSSETO	94	0,916	CORSO GROSSETO	94

Tabella 3.3: Esempio di Address Resolution

3.1.3 Expert-driven outlier detection

Una volta ottenuto un *dataset* pulito dal punto di vista degli attributi geospaziali, il passo successivo è stato quello dell'applicazione delle tecniche di *outlier detection* guidate dall'esperto di dominio. L'applicazione di una metodologia *domain driven*, ha permesso di selezionare gli attributi critici per la determinazione della prestazione energetica degli edifici sui cui effettuare un'*outlier detection* specifica, ovvero (per i dettagli si rimanda alle sezioni 1.4.2 e 1.4.3):

- *Fattore Forma*
- *Trasmittanza media delle superfici trasparenti*
- *Trasmittanza media delle superfici opache*
- *Rendimento sottosistema di distribuzione*
- *Rendimento sottosistema di regolazione*
- *Rendimento sottosistema di generazione*
- *Rendimento sottosistema di emissione*

Prima di applicare i *range* definiti dall'esperto di dominio, uno *scaling* dei valori relativi ai rendimenti è stato effettuato. Numerosi attributi presentavano infatti un valore corretto ma espresso in una scala errata, ad esempio un rendimento poteva presentare un valore di 89 quando quello corretto sarebbe stato 0,89. Per questo motivo una divisione per un fattore 100 è stata applicata qualora il rendimento presentava un valore maggiore di 1. Caso particolare è stato il *rendimento di generazione*. In presenza di un impianto con pompa di calore (identificata dal COMBUSTIBILE = 'Energia Elettrica'), un valore del rendimento di generazione maggiore dell'unità e sull'ordine della decina, è ammissibile. Inoltre non è possibile distinguere il caso in cui il valore presente esprime il valore COP e si è dunque proceduto all'eliminazione di questi certificati. Per gli impianti di *teleriscaldamento* invece un valore fisso di 0,626 è stato imposto come rendimento di generazione, dato recuperato da un documento dell'IREN che esprime il *fattore di conversione in energia primaria dell'energia termica fornita ai punti di consegna della rete di teleriscaldamento di Torino* [43]. Dopo questa operazione, sono stati applicati i filtri suggeriti dall'esperto di dominio, di cui viene fornita una descrizione dettagliata nell Tabella 3.4.

Attributo	**Range di ammissibilità**
Fattore Forma	[0,1 - 2]
Rendimento Distribuzione	[0,75 - 1,25]
Rendimento Emissione	[0,85 - 1]
Rendimento Generazione	0,626 and [0,65 - 1,1]
Rendimento Regolazione	[0,6 - 1]
Trasmittanza Opaca	[0,1 - 3]
Trasmittanza Trasparente	[0,9 - 7]

Tabella 3.4: *Range* di ammissibilità definiti dall'esperto di dominio

Definiti i filtri, questi sono stati applicati in cascata. In particolare, in una prima fase sono stati applicati i filtri relativi alle *trasmittanze* e al *fattore forma* e poi quelli sui vari *rendimenti dei sottosistemi*. L'applicazione dei filtri relativi a *fattore forma, trasmittanza opaca e trasmittanza trasparente* ha avuto un impatto modesto sul *dataset*, in quanto sono stati scartati meno del 2% dei certificati presenti. In figura 3.3 è rappresentato un dettaglio della distribuzione delle trasmittanze prima dell'applicazione dei filtri, dove il *range* di ammissibilità è compreso fra le due linee verticali.

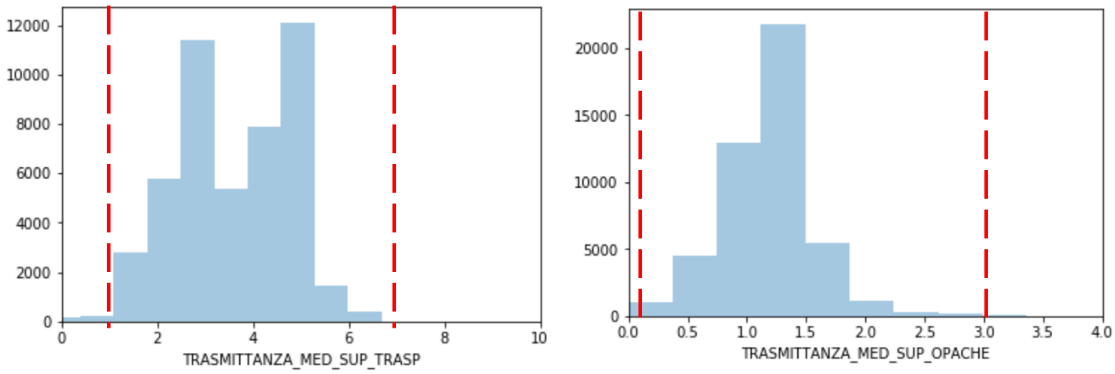

Figura 3.3: Distribuzione di Trasmittanza Trasparente (a sinistra) e di Trasmittanza Opaca (a destra)

Un impatto maggiore ha invece avuto l'applicazione dei filtri sui vari rendimenti dei sottosistemi dopo i quali circa il 66% dei certificati originari è rimasto a disposizione. Un dettaglio sulla distribuzione dei rendimenti (valori nulli esclusi) è mostrata in Figura 3.4, dove le linee rosse rappresentano i filtri definiti dall'esperto di dominio. Nel caso del *rendimento di generazione*, le linee verdi identificano i certificati che presentano il valore 0,626, che è stato imposto per il teleriscaldamento e che risulta quindi valido in termini di ammissibilità.

Questo numero elevato di certificazioni scartate è influenzato soprattutto da valori che sono fuori range, ma che presentano anche valori nulli. Dalla Figura 3.5 è possibile notare come i rendimenti di distribuzione, emissione e regolazione presentino circa il 25% di valori nulli (o non valorizzati) contro un più basso 13% del rendimento di generazione. La presenza di questi valori può essere dovuta ad errori di arrotondamento o di conversione del *software* oppure ad un errore di inserimento da parte del certificatore. Da questo si deduce che dei circa 15000 certificati esclusi a causa di un valore non ammissibile per i rendimenti, ben oltre il 70% contenga campi nulli (o non valorizzati).

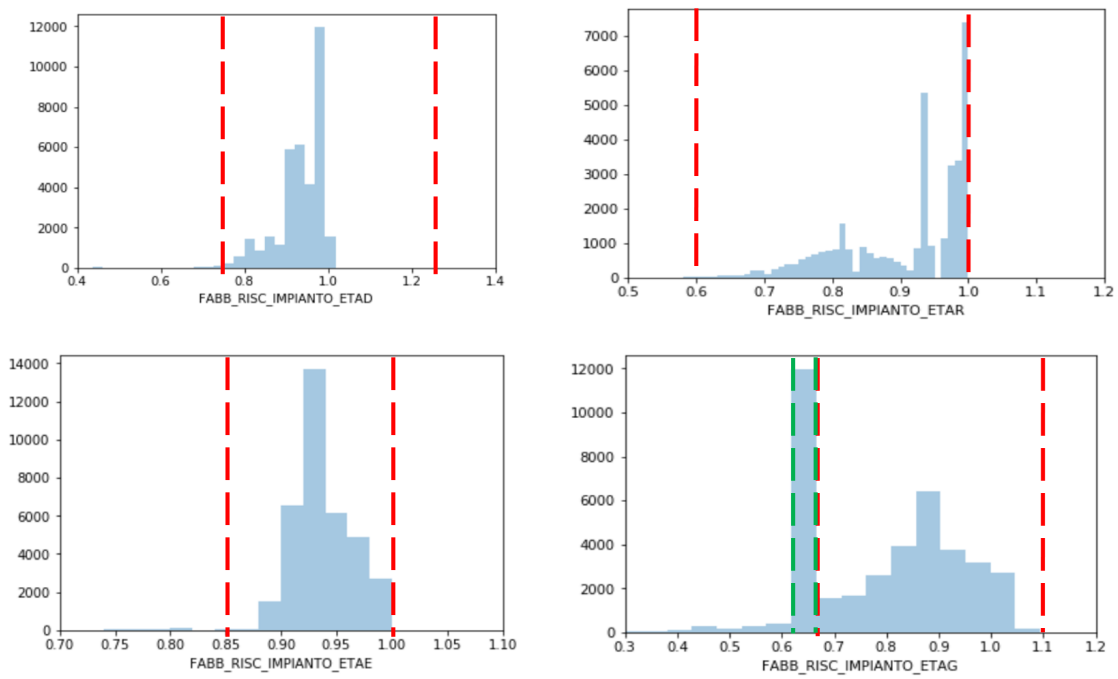

Figura 3.4: Distribuzione dei quattro rendimenti dei sottosistemi e filtri applicati

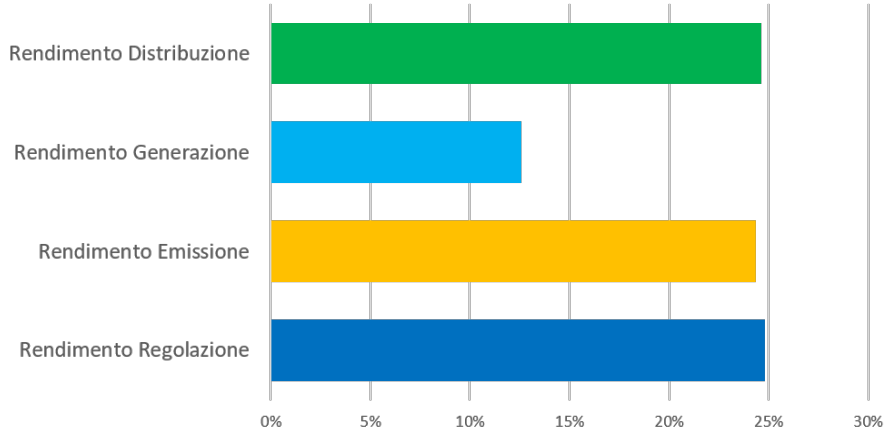

Figura 3.5: Percentuali di rendimenti nulli sul totale delle certificazioni

3.1.4 Univariate outlier detection

Su tutte le altre variabili considerate importanti per l'analisi, ma per cui non sono stati suggeriti dei range di ammissibilità da parte dell'esperto di dominio, si sono applicate le

tecniche di outlier detection presentate nella sezione 2.1.2, ovvero il gESD [21] e il boxplot.

gESD

Il gESD è stato applicato in parallelo su attributi ritenuti rilevanti per l'analisi ma che non sono stati già filtrati dai range suggeriti dall'esperto di dominio, tra cui:

- Superficie riscaldata
- Superficie disperdente
- Volume lordo riscaldato
- Emissioni gas serra
- Anno costruzione
- Fabbisogno di energia termica utile

Il parametro di MaxOLs (massimo numero di *outlier*) è stato posto in modo che fosse pari allo 0.5% del numero di certificazioni presenti nel *dataset* e al parametro *alpha* (significatività) è stato dato un valore pari all'1%. Il gESD è riuscito ad eliminare in maniera efficace gli *outlier*, come nell'esempio riportato in Figura 3.6 per l'attributo *indice di prestazione energetica globale non rinnovabile* e in Figura 3.7 per l'attributo *anno costruzione*. Alla fine di questa operazione di *outlier detection e removal*, il *dataset* è composto da circa 30000 attributi.

Percentile Outlier Detection

Dopo l'applicazione del gESD, un'ulteriore analisi sugli attributi è stata effettuata. In particolare per alcune variabili geometriche come la superficie riscaldata, la superficie disperdente e il volume lordo riscaldato, il gESD ha lavorato bene sul *range* superiore della distribuzione, mentre su quello inferiore alcuni valori poco realistici sono rimasti. Per evitare che dei certificati contenenti informazioni non ammissibili per le variabili sopra elencate influenzassero in maniera negativa i risultati, si è provveduto ad eliminare tutti quei valori presenti al di sotto del percentile [22] 1%, ovvero la coda della distribuzione. Questa operazione ha avuto un impatto su alcune centinaia di *record*. Facendo in questo modo, il valore minimo della superficie riscaldata, ad esempio, è passato da circa 6 a poco più di 20, mentre il resto della distribuzione è rimasto pressoché invariato.

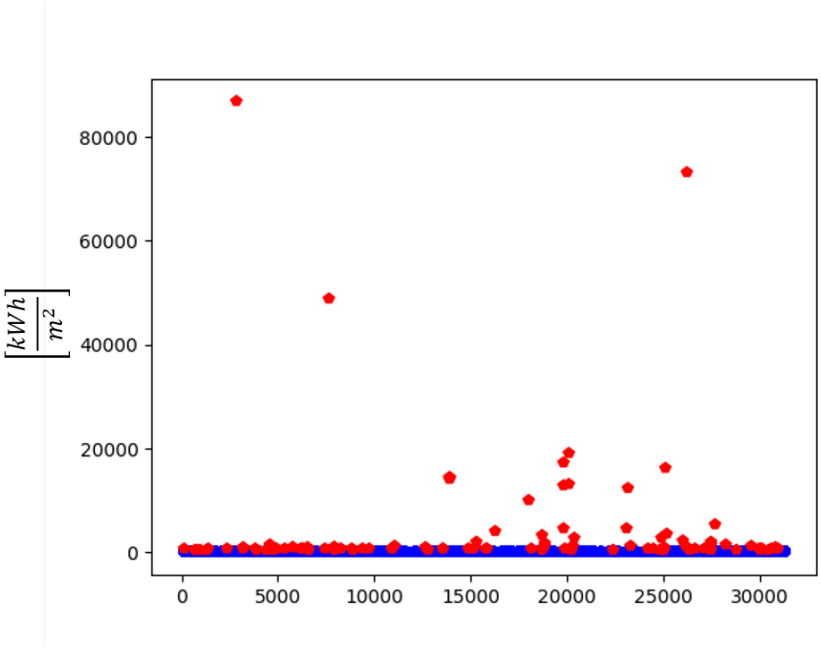

Figura 3.6: Effetto del gESD sull'indice di prestazione globale non rinnovabile

3.1.5 Multivariate outlier detection

Un'*outlier detection* di tipo univariata non è sufficiente ad assicurare che i dati analizzati siano privi di *outlier*. Nelle successiva fase di *Data Mining*, infatti, più attributi vengono considerati in maniera congiunta per effettuare operazioni di *clustering*. Può infatti accadere che nonostante le singole variabili siano prive di *outlier*, quando analizzate in maniera congiunta, queste presentino delle anomalie. Per questo motivo un'*outlier detection* di tipo multivariata è stata effettuata attraverso l'algoritmo DBSCAN. Questo algoritmo prevede il setting dei parametri *Eps* e *MinPoints*, utilizzando un *k-distance graph*. Il *plot* in Figura 3.8, rappresenta il momento in cui la curva si è stabilizzata ovvero con un valore di *MinPoints* pari a 5. Leggendo poi il valore in corrispondenza del gomito (intersezione con la linea verde), un valore di *Eps* pari a 0,28 è stato determinato. Scelti questi parametri è stato poi eseguito il DBSCAN sul *dataset*, il quale è riuscito ad isolare una zona a bassa densità, contenente un centinaio di punti, come cluster contente rumore (outliers) e un'altra molto densa contenente il resto dei dati. Dopo l'eliminazione di questi certificati presenti nel Cluster -1, poco meno di 30000 certificati sono rimasti all'interno del dataset.

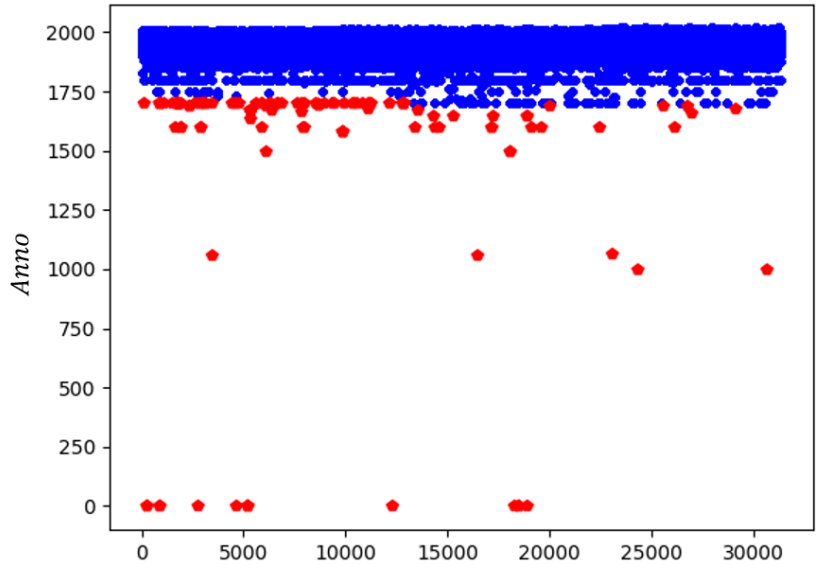

Figura 3.7: Effetto del gESD sull'anno di costruzione

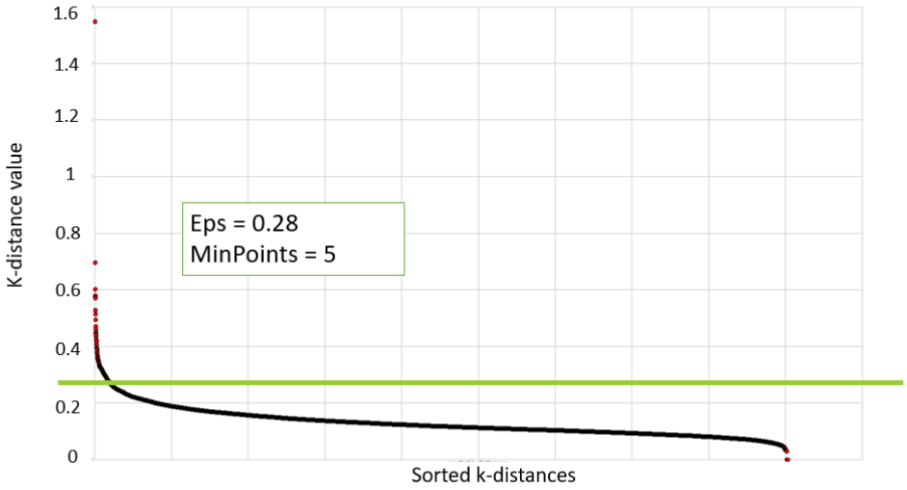

Figura 3.8: *K-distance plot* per il *setting* dei parametri del DBSCAN

3.2 Data Preparation

Conclusa la fase di preprocessing, il dataset risulta ora privo di outlier. Prima di poter fare delle analisi, è necessaria una selezione delle variabili.

3.2.1 Correlation Analysis

In supporto all'esperto di dominio è stata effettuata un'analisi di correlazione. La matrice di correlazione generata viene mostrata in Figura 3.9. I valori di correlazione di Pearson (in valore assoluto) possono essere letti a seconda dal colore assunto dalla cella. Una bassa correlazione tra le variabili è rappresentata da un colore grigio mentre un'elevata correlazione da un colore rosso intenso. Da questa analisi è facile notare come l'indice di prestazione energetica globale non rinnovabile (*EPGL_NREN*) sia altamente correlato (0,98) con l'*EPH*, così come la *superficie disperdente* con il *volume vordo riscaldato* (0.93). Una correlazione media è, ad esempio, presente tra la *superficie riscaldata* e il *volume lordo riscaldato* (0.73) oppure tra *superficie disperdente* e *superficie riscaldata* (0.69). Da notare inoltre che i vari rendimenti dei sottosistemi sono stati sostituiti dalla variabile ETAH, che tiene conto di questi contributi in maniera sintetica (vedi sezione 1.4.3).

3.2.2 Data and Feature Selection

L'analisi di correlazione svolta al passo precedente è venuta in supporto all'esperto di dominio per scegliere le variabili significative su cui effettuare le successive analisi. In particolare sono state selezionate le seguenti 7 variabili:

- **Fattore Forma**
- **Trasmittanza media delle superfici trasparenti**
- **Trasmittanza media delle superfici opache**
- **Superficie riscaldata**
- **Superficie disperdente**
- **Anno di costruzione**
- **Rendimento per la climatizzazione invernale ETAH**

Infine la variabile rappresentante l'energia primaria per la climatizzazione invernale, ovvero **EPH**, è stata utilizzata per valutare la bontà delle prestazioni energetiche degli edifici.

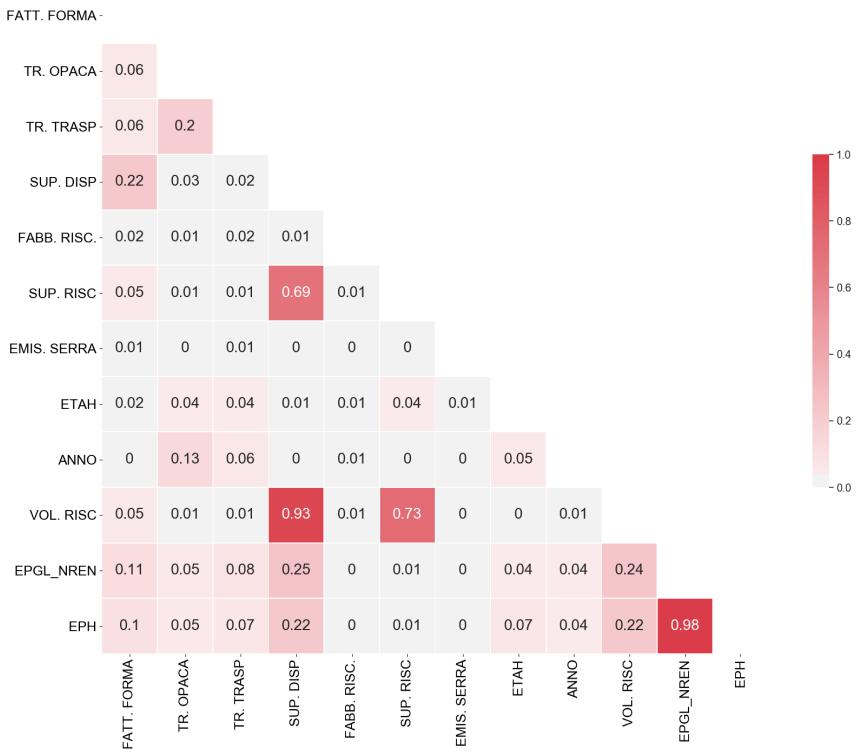

Figura 3.9: *Correlation matrix* utilizzata per l'analisi

3.2.3 Normalization

Dopo aver selezionato gli attributi rilevanti, un'operazione di normalizzazione è stata eseguita attraverso una normalizzazione del tipo *max-min*, descritta all'interno della sezione 2.2.3. La normalizzazione è fondamentale per la successiva applicazione delle tecniche di clustering attraverso l'uso dell'algoritmo K-Means, il quale si basa proprio sul concetto di distanza. È stato inoltre scelto questo tipo di normalizzazione perchè mantiene la relazione presente nei dati originali (facilitando di conseguenza la lettura dei risultati) e non vi sono problemi dovuti alla presenza di outlier, eliminati dal precedente blocco di *Data Cleaning*.

3.3 Data Mining

In questo blocco il *dataset* è pronto per essere utilizzato per estrarre della conoscenza. Come algoritmo di clustering il K-Means è stato utilizzato, il quale opera una clusterizzazione

a due livelli di tipo ricorsivo, arricchita da una *feature selection* automatica. Prima di procedere alla presentazione dei risultati, viene descritta la metodologia per la scelta dei parametri necessari per il funzionamento dell'algoritmo.

3.3.1 Setting dei parametri

Per utilizzare l'algoritmo K-Means, il valore del parametro k, ovvero del numero di cluster da generare, deve essere settato a priori. Al fine di scegliere un valore ottimale, l'euristica *Elbow Method* (descritto nella sezione 2.3.1) è stata applicata. In particolare l'SSE è stato calcolato per diversi k, all'interno del *range* $[2-30]$. Plottando poi il risultato, il grafico in Figura 3.10 è stato ottenuto. Da questo è possibile notare come non sia presente un gomito ben definito, ma piuttosto vengono individuati un insieme di valori possibili, compresi fra $k = 7$ e $k = 15$. Dopo aver effettuato una serie di prove sperimentali, il valore $k = 12$ è stato scelto per effettuare le successive analisi.

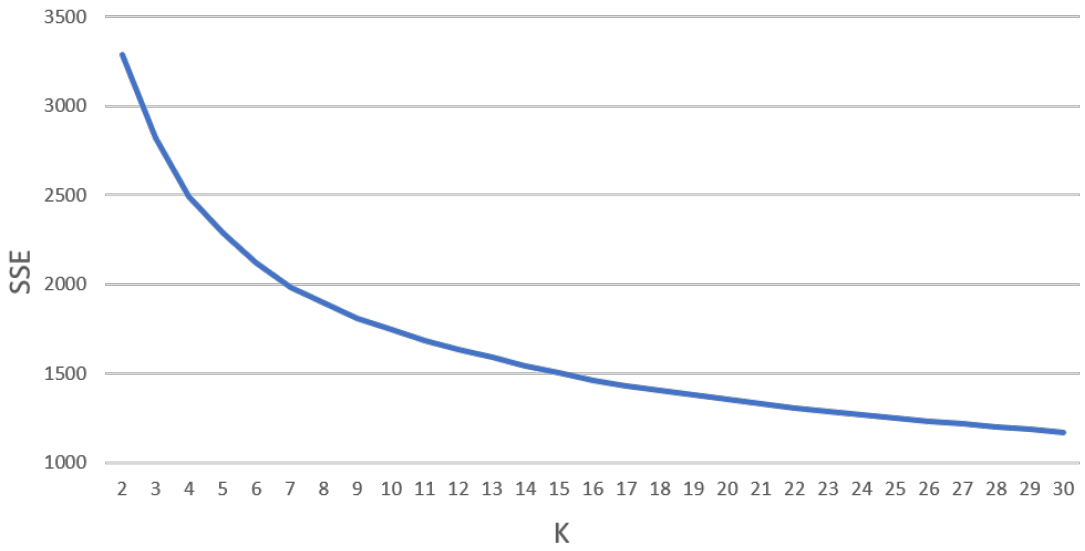

Figura 3.10: *Elbow graph* con k nel range $[2-30]$

3.3.2 Clustering - primo livello

Come ribadito precedentemente, la variabile relativa all'indice di prestazione energetica per la climatizzazione invernale (EPH) non è stata utilizzata come input dell'algoritmo di *clustering* ma per caratterizzare la bontà dei risultati generati da quest'ultimo. In particolare, l'EPH è stato discretizzato attraverso l'utilizzo di quattro classi, delimitate

dai valori relativi al primo quartile, alla mediana e al terzo quartile. Definiti questi valori, gli intervalli rappresentati in Tabella 3.5 sono stati generati.

Intervallo	Performance	Colore
$0 < EP_h \leq 85$	Molto efficiente	
$85 < EP_h \leq 138$	Efficiente	
$138 < EP_h \leq 205$	Poco efficiente	
$EP_h > 205$	Inefficiente	

Tabella 3.5: Discretizzazione della variabile EPH

Il passo successivo è stato quello di effettuare il K-Means con il k pari a 12 (scelto precedentemente) e come variabili di *input* quelle selezionate durante la fase di *Data and Feature Selection*, opportunamente normalizzate. Come metrica per il calcolo della distanza, quella euclidea è stata considerata. Il clustering ha dato origine a 12 gruppi la cui cardinalità è deducibile dall'analisi della Figura 3.11. I *cluster* 1 e 3 sono quelli più numerosi contenenti rispettivamente il 14% e il 13% dei dati, mentre quelli più piccoli sono il *cluster* 0 con il 3% e il *cluster* 2 con il 4% dei dati. Per valutare la bontà della

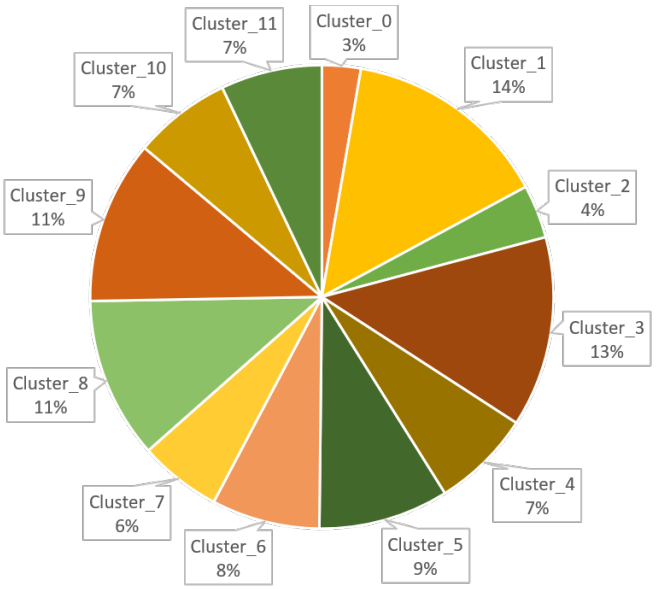

Figura 3.11: Cardinalità dei cluster ottenuti con $k = 12$

divisione dei gruppi, è stato utilizzato l'EPH. In particolare un cluster risulta ben separato dagli altri se contiene al suo interno almeno il 40% di certificati aventi un valore di EPH appartenente ad uno dei range descritti nella Tabella 3.5. Il risultato di questa

caratterizzazione è illustrato in Figura 3.12. Detto ciò, è possibile facilmente notare come i

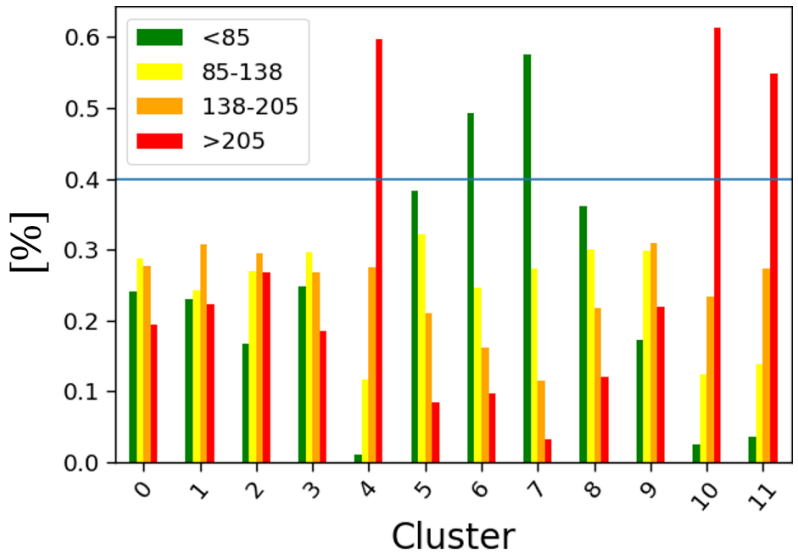

Figura 3.12: Caratterizzazione dei cluster mediante discretizzazione dell'EPH

cluster $4, 6, 7, 10$, ed 11 presentino almeno uno dei *range* al di sopra della soglia individuata dalla linea orizzontale azzurra. In particolare possiamo etichettare i *cluster* $4, 10$ ed 11 come cluster che contengono al loro interno edifici *inefficienti*, mentre i *cluster* 6 e 7 come gruppi contenenti edifici *molto efficienti*. Per gli altri cluster l'algoritmo non è riuscito a separare bene le varie caratteristiche. Per questo motivo è stato effettuata un'ulteriore clusterizzazione su questi, possibilità data anche dal fatto che la cardinalità di questi gruppi nella maggior parte dei casi è superiore al migliaio. Con questa prima clusterizzazione, quasi il 35% dei dati è stato etichettato immediatamente. Prima di procedere all'analisi dei dati risultati dalla clusterizzazione di secondo livello, un dettaglio sul contenuto dei cluster etichettati viene di seguito presentata, partendo dai gruppi inefficienti. Una caratterizzazione più dettagliata può essere ricavata dall'osservazione dei boxplot per ciascun cluster ma, per chiarezza espositiva, una descrizione del contenuto è proposta attraverso il *plot* dei centroidi su grafico *radar*, per ciascuna variabile e per ciascun cluster oggetto di analisi (Figura 3.13). Da questo grafico è possibile notare come gli edifici appartenenti a questi gruppi hanno approssimativamente la stessa dimensione in termini di *superfici* e presentano un elevato *fattore forma*. I *cluste*r 4 ed 1 presentano valori di *trasmittanza termica*, sia opaca che trasparente, molto elevati mentre il *cluster* 11 presenta un valore di *trasmittanza trasparente* più basso, ma la sua prestazione è inficiata dal *fattore forma* più alto delle media. Anche in termini di *efficienza termica*, i cluster presentano valori

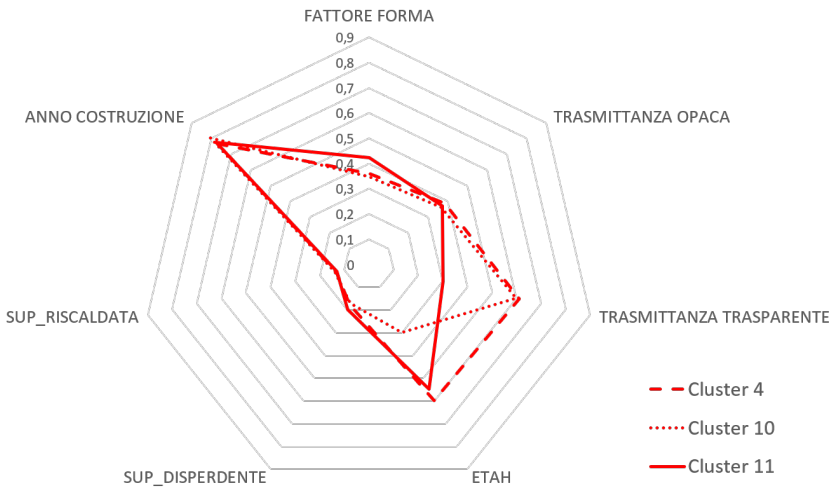

Figura 3.13: Radar dei tre cluster etichettati come inefficienti

medi (*cluster* 11 e 4) o molto bassi (*cluster* 10). Da questa analisi si può dedurre che edifici che presentano un elevato fattore forma sono inefficienti nonostante presentino valori nella media per quanto riguarda ETAH o trasmittanza trasparente. In conclusione, si può affermare che fattore forma, trasmittanza trasparente ed efficienza dell'impianto per la climatizzazione invernale rappresentano gli attributi che meglio riescono a discriminare i vari comportamenti tra i vari gruppi. Stessa analisi può essere fatta sui cluster che sono stati etichettati come *molto efficienti*, sempre attraverso l'osservazione dei vari centroidi su un grafico di tipo radar (Figura 3.14). Dall'analisi dei cluster etichettati como molto efficienti, si nota come questi abbiano valori di *trasmittanza*, sia opaca che trasparente, molto bassi. Per quanto riguarda il *cluster* 6, questo è caratterizzato da un *rendimento* dell'impianto di riscaldamento medio ma riesce comunque ad essere molto efficiente grazie al *fattore forma* molto contenuto. Viceversa il *cluster* 7 presenta un *fattore forma* leggermente più elevato ma un valore relativo al rendimento molto alto. Come nel caso precedente, gli attributi relativi alla trasmittanza, al rendimento dell'impianto e al fattore forma sono quelli che meglio riescono a caratterizzare i vari gruppi. La differenza è ancora più marcata se viene fatto un confronto fra *cluster* etichettati come molto efficienti e cluster contenenti edifici inefficienti. Dal *radar* in Figura 3.15, si può notare come i cluster in rosso (edifici inefficienti) possiedano valori ben più elevati di trasmittanza opaca, trasmittanza trasparente e fattore forma rispetto ai cluster in verde (molto efficienti).

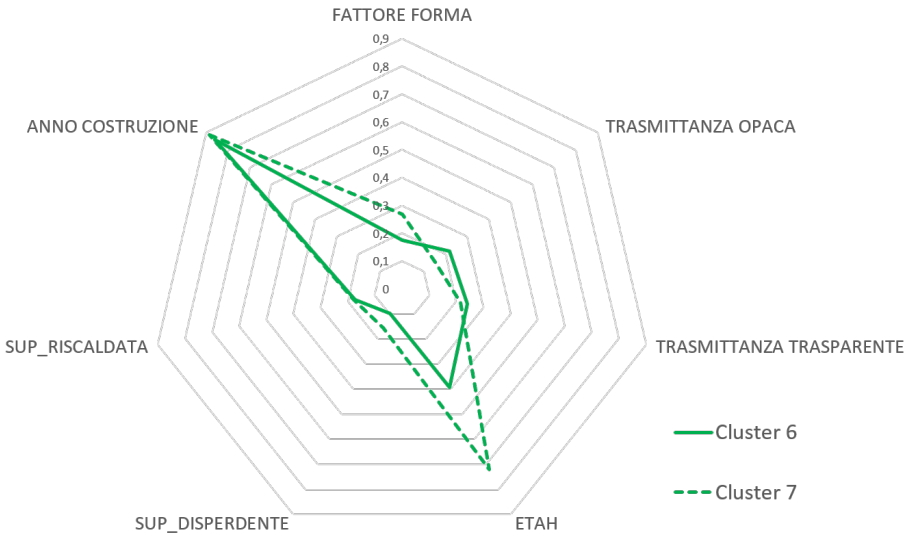

Figura 3.14: Radar dei due cluster etichettati come molto efficienti

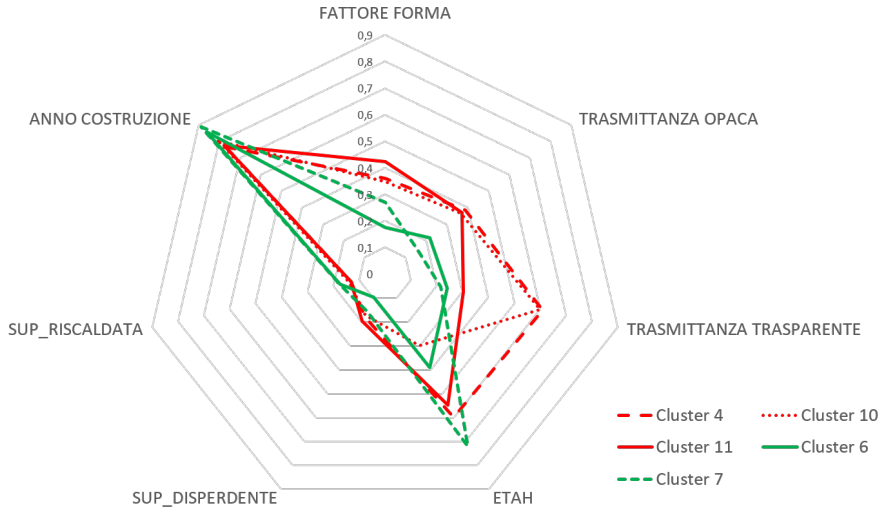

Figura 3.15: Confronto tra cluster molto efficienti (in verde) ed inefficienti (in rosso)

3.3.3 Clustering - secondo livello

Per tutti quei cluster che non sono stati etichettati al primo livello (ovvero i *cluster* 0,1,2,3,5,8,9), è stata effettuata un'altra operazione di clustering, utilizzando un valore di k pari a 5. Il *framework*, in maniera automatica, individua quei gruppi da clusterizzare una

seconda volta scegliendo ad ogni passo le variabili di input più adatte. Questa scelta avviene attraverso l'uso della funzione *hclustvar*[31] (vedi sezione 2.3.2), la quale permette di fare un *clustering* di tipo gerarchico sugli attributi. Una volta generato il dendrogramma, viene definito il livello di taglio attraverso il massimo valore assunto dallo *stability plot*. Se sono presenti più variabili all'interno dello stesso gruppo si sceglie quella che ha la correlazione più elevata con la variabile EPH. Selezionati gli attributi da generare, viene eseguito il *clustering* utilizzando il valore di k precedentemente specificato. In questo caso, essendo i gruppi meno numerosi, una soglia del 30% è stata imposta per etichettare il *cluster*. Viene riportato di seguito un esempio di partizionamento di secondo livello sul *cluster* 1, descrivendo la scelta delle variabili di input (Figura 3.16), la cardinalità rispetto alla situazione precedente (Figura 3.17) e la caratterizzazione della bontà del partizionamento per mezzo dell'istogramma in Figura 3.18. Per quanto riguarda la scelta delle variabili, dallo *stability plot* in Figura 3.16 è possibile notare come il massimo valore si trovi in corrispondenza del valore 6 e, tagliando il dendrogramma, si ottengono 6 partizioni di cui una contenente due variabili (*superficie riscaldata* e *superficie disperdente*). Tra questi due attributi è stata calcolata la correlazione con la variabile EPH e la *superficie disperdente* è stata selezionata poiché possedeva valore più elevato. Le variabili selezionate dall'algoritmo

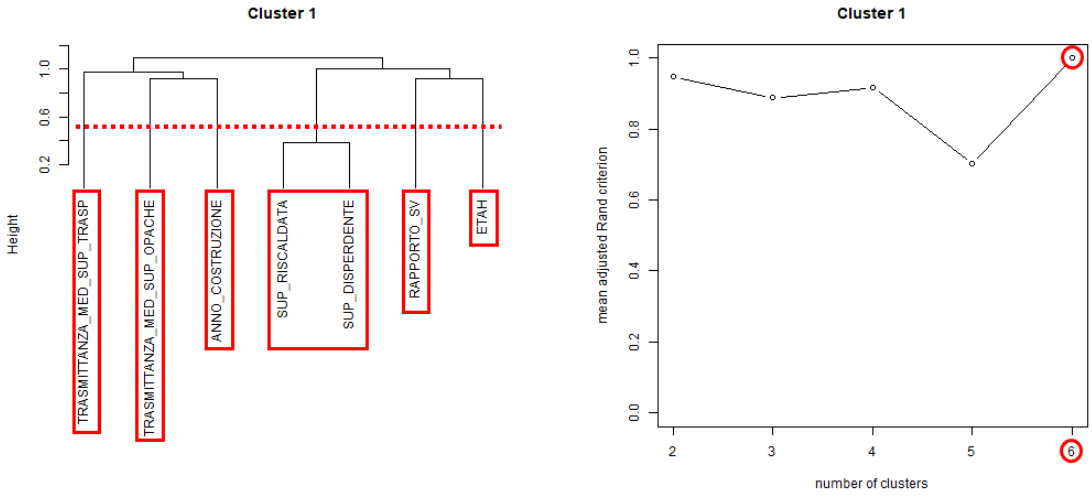

Figura 3.16: Dendrogramma e *stability plot* per il *cluster* 1

sono quindi:

- Fattore Forma
- Trasmittanza media delle superfici trasparenti

- Trasmittanza media delle superfici opache
- Superficie disperdente
- Anno di costruzione
- Rendimento per la climatizzazione invernale ETAH

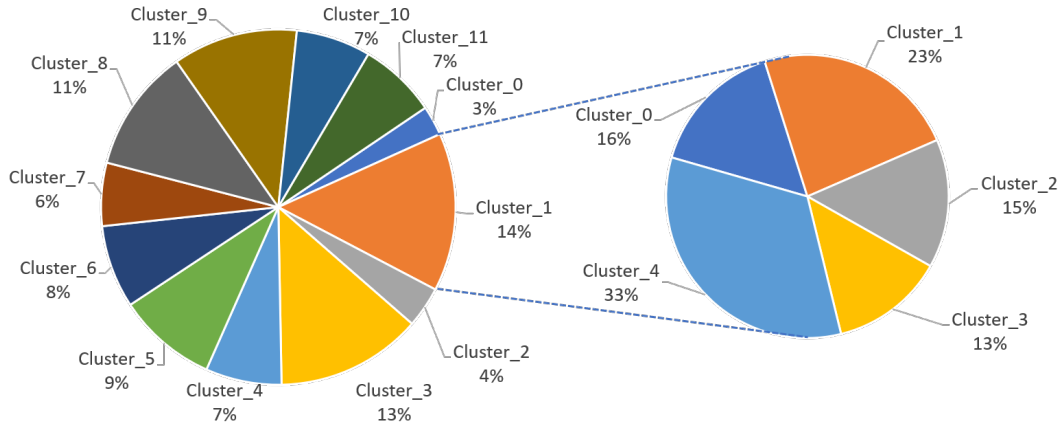

Figura 3.17: Dettaglio della cardinalità del *cluster* 1 dopo il clustering di secondo livello

Come si nota in Figura 3.17, il *cluster* 1 era quello più numeroso e, dopo il clustering di secondo livello, è stato partizionato in 5 sotto-gruppi di cardinalità pressoché paragonabile, ad eccezione del *sotto-cluster* 4. Se prima osservando l'istogramma (Figura 3.18) non vi era un comportamento ben definito dell'EPH che consentisse di etichettare il *cluster*, ora i gruppi sono ben separati, permettendo di classificarne efficacemente il 67%. Ripetendo poi questi passaggi su tutti i *cluster* individuati dall'algoritmo, si è passati da circa il 35% di *cluster* etichettati dal *clustering* di primo livello a circa il 92% (tra primo e secondo livello) dell'intero *dataset* analizzato. Nella tabella 3.6 viene presentato in dettaglio il numero di dati etichetatti, utilizzando gli stessi intervalli definiti per la generazione degli istogrammi.

3.4 Knowledge Interpretation

In questa sezione i risultati sperimentali relativi all'ultimo blocco del *framework* vengono presentati. In particolare un dettaglio sull'uso degli alberi di decisione e dei *boxplot* per la caratterizzazione dei risultati dei *clustering* ottenuti precedentemente verrà descritto, per poi passare ad una descrizione sulle mappe energetiche generate.

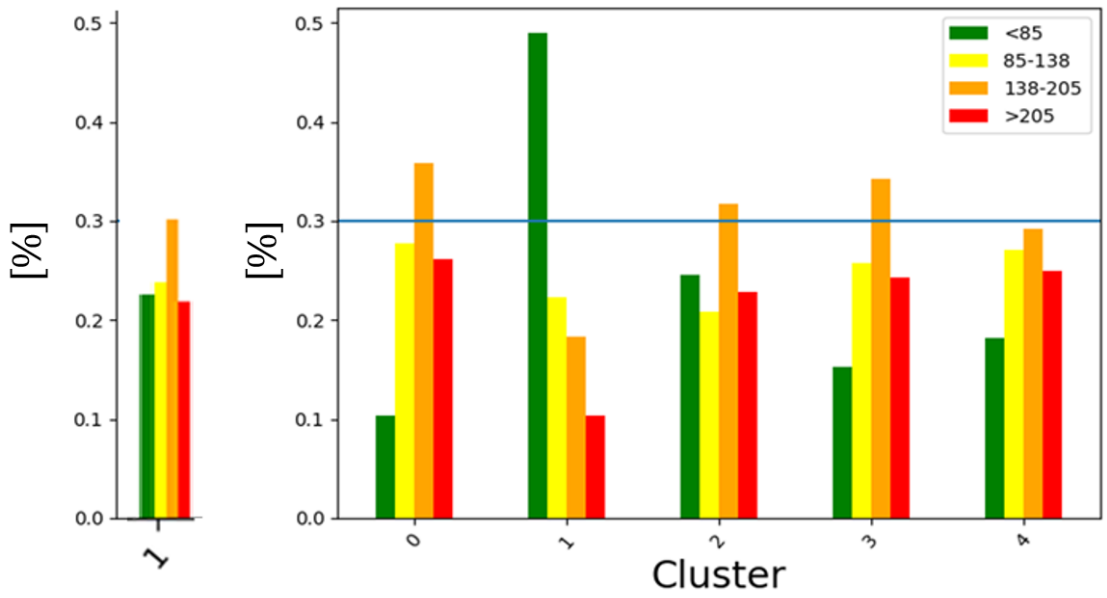

Figura 3.18: Distribuzione EPH per il *cluster* 1 al primo livello (a sinistra) e dopo il clustering di secondo livello (a destra)

Etichetta	Percentuale	Colore
Molto efficiente	28.51%	
Efficiente	21.48%	
Poco efficiente	16.93%	
Inefficiente	25.03%	
Non etichettato	8.05%	

Tabella 3.6: Cardinalità dei gruppi etichettati dal primo e dal secondo livello

3.4.1 Knowlwedge Characterization

Come descritto all'interno della sezione 2.4.1, il CART è stato utilizzato per caratterizzare il contenuto dei gruppi generati dall'algoritmo di clustering, attraverso l'estrazione di regole del tipo *IF-THEN*. Per la costruzione del CART è stato utilizzata la funzione *rpart*[5], presente all'interno del pacchetto *rpart* di R. Per la definizione dei parametri è stato fissato il *Complexity Parameter* (cp) a 0.01, analizzandone l'andamento in funzione dell'errore relativo commesso effettuando una 10-fold cross validation. In questo modo è stato possibile

[5]cran.r-project.org/web/packages/rpart/rpart.pdf

determinare se fare un *pruning* dell'albero decisionale in modo da diminuirne la complessità, mantenendo l'errore relativo all'interno della deviazione standard dell'errore commesso con cp pari a 0.01. Dal grafico in Figura 3.19 si nota che non è stato necessario effettuare un'operazione di *pruning* in quanto la configurazione ottenuta risultava già quella con il cp minimo e con l'errore minimo. Per facilità di lettura, una parte dell'albero di decisione

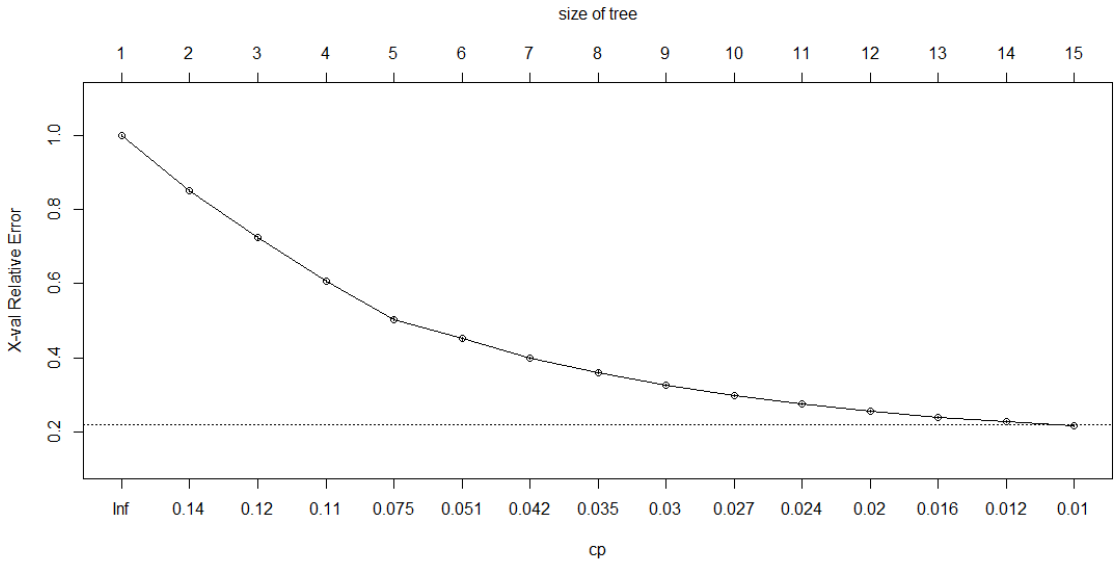

Figura 3.19: Andamento del *complexity parameter* in funzione dell'errore relativo (*10-fold cross validation*)

è presente in Figura 3.20. Dall'analisi dell'albero di decisione alcune regole possono essere estratte, di cui se ne riportano alcune nella Tabella 3.7.

IF	**THEN**	**Performance**
TR. TRASP. alta & ETAH medio & FATT. FORMA alto	Cluster 10	Inefficiente
TR. TRASP. alta & ETAH alto & FATT. FORMA alto	Cluster 4	Inefficiente
TR. TRASP. alta & ETAH basso & FATT. FORMA basso	Cluster 9	Poco Efficiente
TR. TRASP. bassa & ETAH medio & TR. OPACA bassa	Cluster 6	Molto Efficiente
TR. TRASP. bassa & ETAH alto & TR. OPACA bassa	Cluster 7	Molto Efficiente

Tabella 3.7: Alcune regole IF-THEN estratte dall'albero di decisione

Per quanto riguarda la validazione del CART, si è raggiunta un'accuratezza dell'81,8%. Un dettaglio sui valori di *precision* e *recall* è riportato in Tabella 3.8.

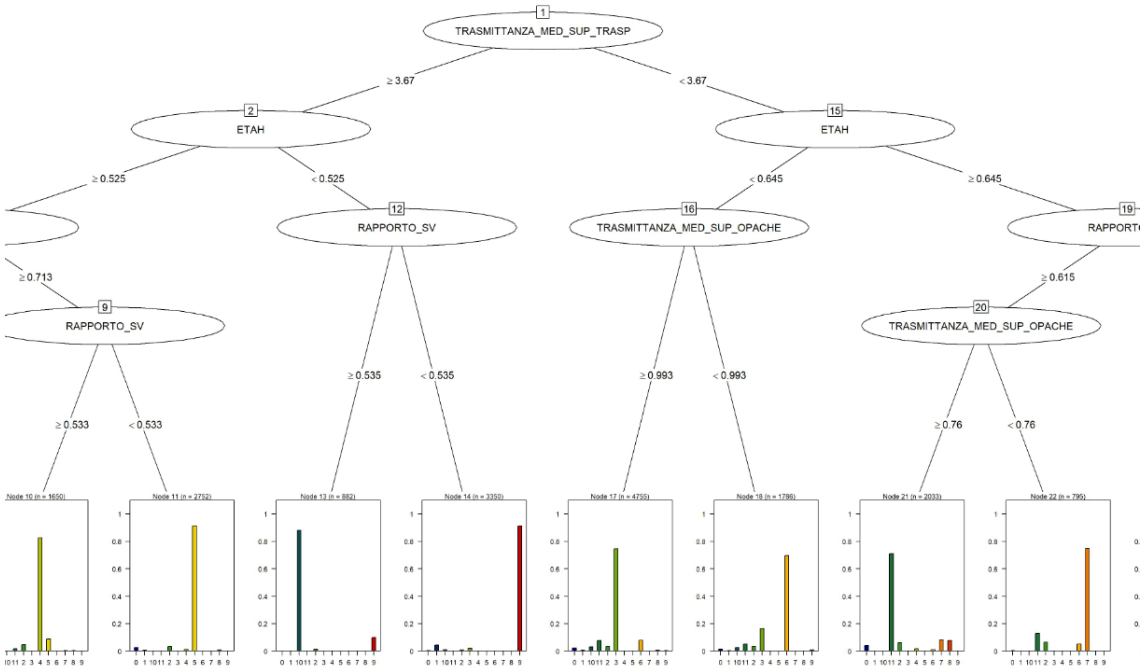

Figura 3.20: Dettaglio del CART generato per la caratterizzazione della label del clustering

Classe	Precision	Recall
0	0.871	0.374
1	0.857	0.929
2	-	-
3	0.747	0.891
4	0.821	0.924
5	0.913	0.923
6	0.709	0.725
7	0.763	0.809
8	0.817	0.881
9	0.912	0.894
10	0.887	0.840
11	0.711	0.682

Tabella 3.8: Matrice di confusione per il CART generato

Il nodo radice dell'albero è rappresentato dalla *trasmittanza trasparente*, la quale si conferma l'attributo che meglio riesce a separare gli appartamenti in cluster a diversa

prestazione energetica. I cluster che presentanto edifici molto efficienti presentano tipicamente una trasmittanza trasparente molto bassa, così come un fattore forma contenuto e un rendimento dell'impianto di riscaldamento medio-alto.

In questa fase il *boxplot* [34] viene utilizzato per fornire informazioni più dettagliate sulla distribuzione delle singole variabili all'interno di un cluster, informazione molto utile per un esperto di dominio. In particolare è possibile notare come nel *cluster* 1 siano presenti gli edifici vecchi che possiedono un anno di costruzione precedente al 1900 e, nel *cluster* 2, gli edifici con una superficie superiore rispetto alla media. Ciò può essere facilmente derivato dall'osservazione dei *boxplot* in Figura 3.21 e in Figura 3.22.

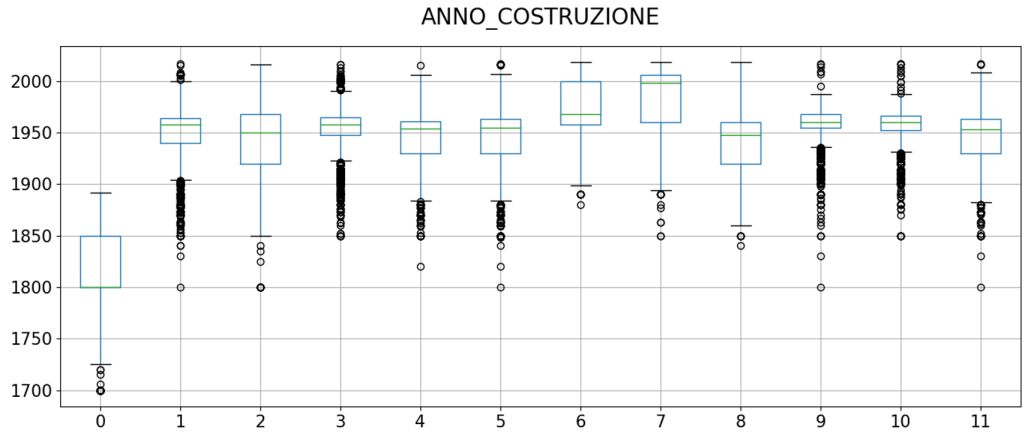

Figura 3.21: Boxplot relativo all'anno di costruzione, separatamente per ogni cluster

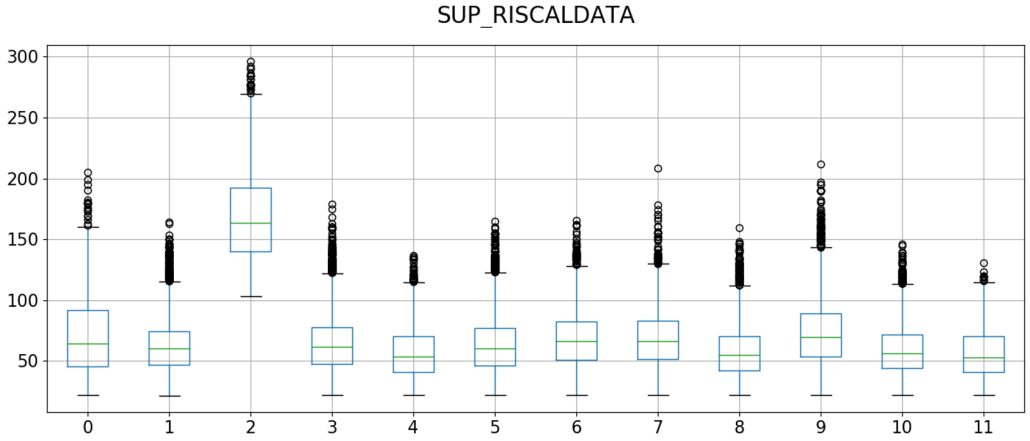

Figura 3.22: Boxplot relativo alla superficie riscaldata, separatamente per ogni cluster

3.4.2 Knowledge Visualization

Come riportato nella sezione 2.4.2, il *framework* permette la generazione di mappe energetiche, in modo da fornire un supporto sia agli esperti di dominio sia ai non esperti. In particolare questo tipo di visualizzazione potrebbbe essere di supporto a tre principali *stakeholder*:

1. **Il privato**: dare uno strumento orientativo durante l'acquisto o la locazione di un immobile
2. **La Pubblica Amministrazione (PA)**: individuare aree critiche in cui poter investire in termini di aumento dell'efficienza eneregetica, riqualificando in modo mirato alcune aree del territorio cittadino
3. **L'analista**: fornire elementi di conoscenza su quali siano i principali fattori che caratterizzano l'efficienza energetica di un edificio

Attraverso la mappa è infatti possibile visualizzare in maniera immediata come gli edifici, più o meno performanti, si distribuiscano all'interno della città. In particolare le mappe possono essere utilizzate sia per analizzare la distribuzione geografica dei certificati rispetto ad un certo tipo di variabile, sia per visualizzare i risultati dal punto di vista spaziale della conoscenza estratta nella fase di *Data Mining*.

Per la caratterizzazione delle variabili di tipo univariato, le mappe coropletiche e quelle *scatter* sono state utilizzate. In Figura 3.23 è riportato un esempio di una mappa coropletica riguardante la distribuzione del rendimento di distribuzione, mentre in Figura 3.24 un dettaglio sulla certificazione mediante *scatter plot*.

Per risolvere il problema della visualizzazione congiunta di più variabili sulla stessa mappa, sono stati implementati i *marker* dinamici. Questa soluzione si presta alla rappresentazione geografica dei cluster estratti dall'algoritmo K-Means. Grazie all'uso della mappa può essere visualizzata, a diverso livello di dettaglio, la distribuzione spaziale degli edifici appartenenti allo stesso cluster, consentendo di estrarre ulteriore conoscenza dagli stessi. Ad esempio è possibile capire che gli edifici appartenenti al *cluster* etichettato come molto efficiente (Figura 3.25), sono distribuiti in tutta la città, ad eccezione della circoscrizione 1 (Centro - Crocetta). In quest'ultima sono invece concentrati gli edifici appartenenti al *cluster* 0, ovvero quello contenente gli edifici più vecchi (Figura 3.26). Un ultimo esempio è dato dagli edifici appartenenti al *cluster* 2, i quali sono stati etichettati come edifici con superfici superiori alla media; dalla mappa (Figura 3.27) è possibile notare come questi siano distribuiti maggiormente a ridosso della zona collinare e nella zona centrale.

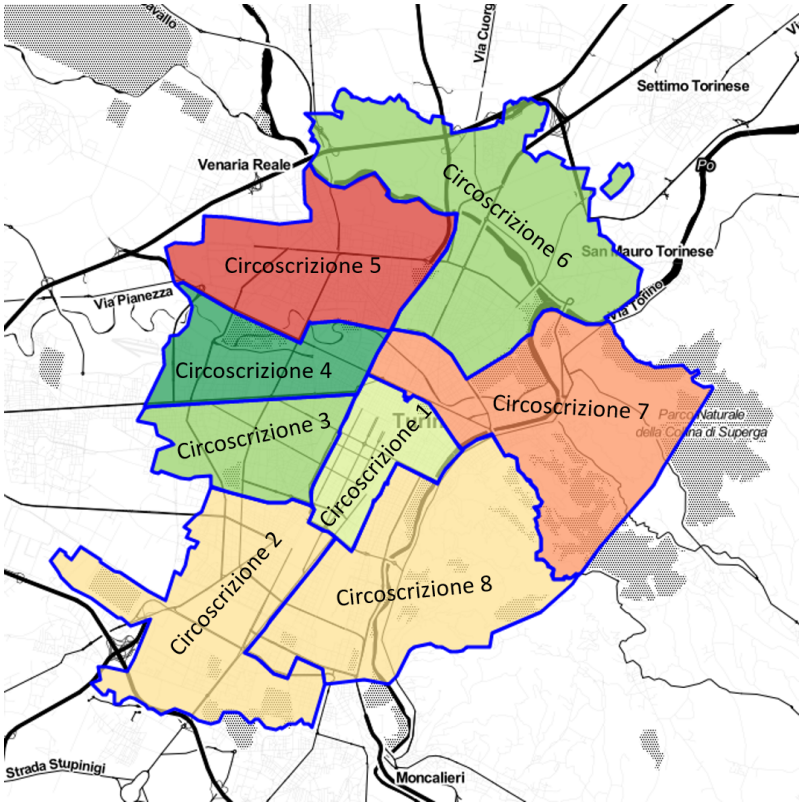

Figura 3.23: Mappa coropletica per il rendimento di distribuzione

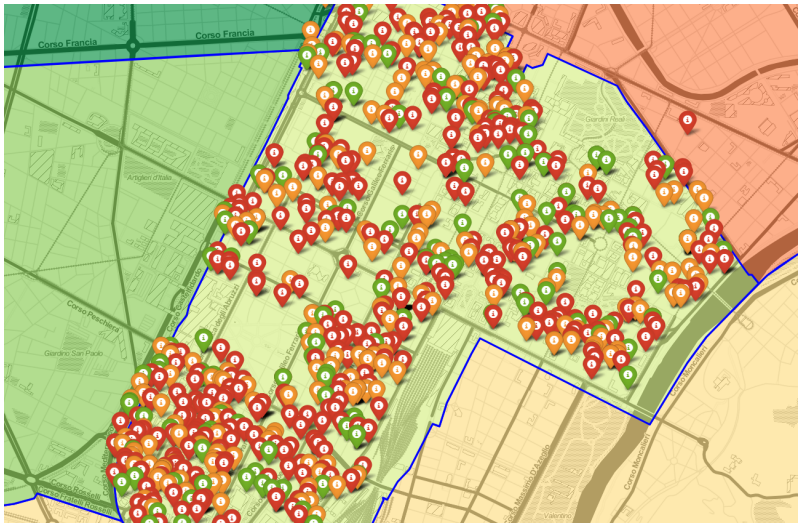

Figura 3.24: Mappa *scatter* per il rendimento di distribuzione (dettaglio Circoscrizione 1)

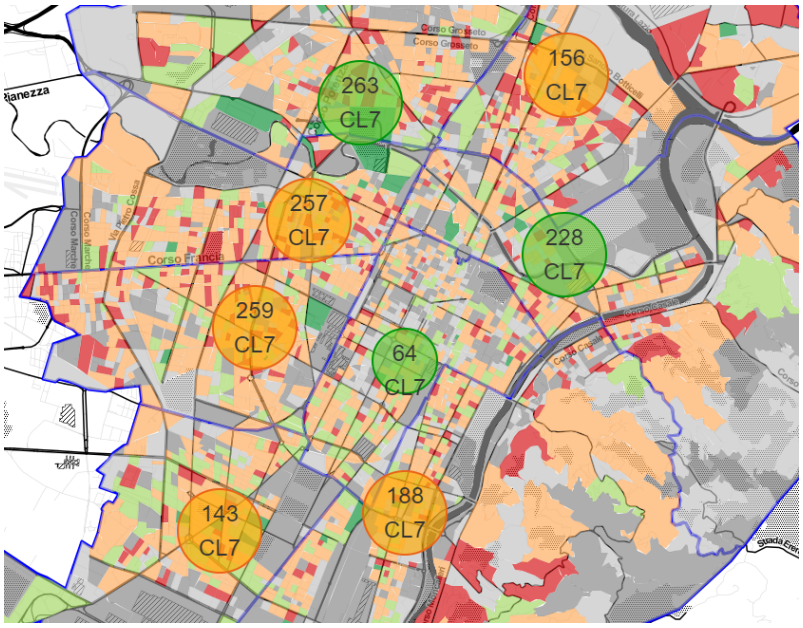

Figura 3.25: Mappa con *marker-cluster* per il cluster 7

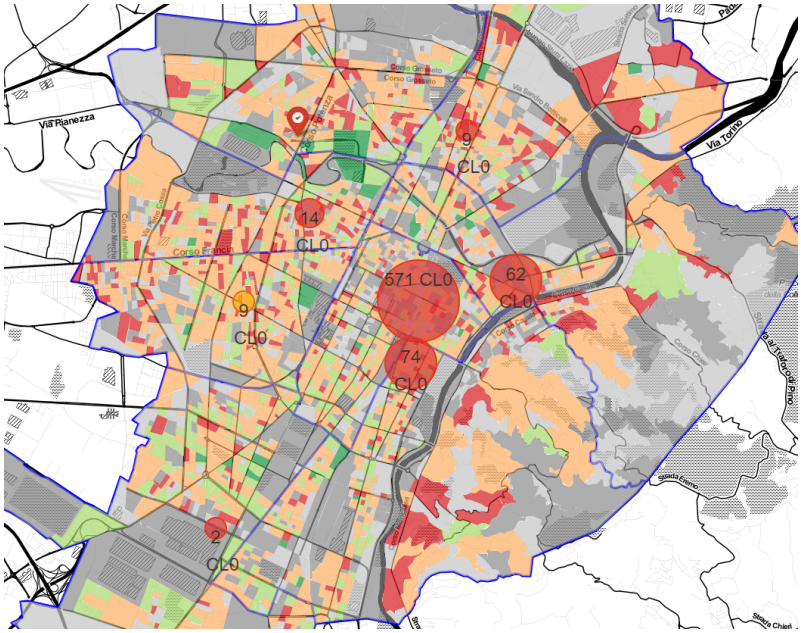

Figura 3.26: Mappa con *marker-cluster* per il cluster 0

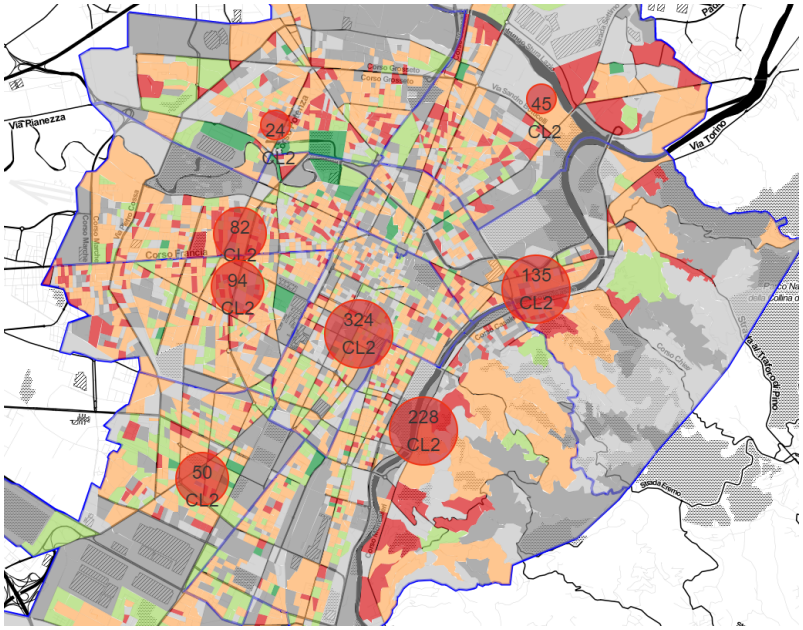

Figura 3.27: Mappa con *marker-cluster* per il cluster 2

Capitolo 4

Conclusioni e sviluppi futuri

In questa tesi è stato progettato e sviluppato un *framework* che permette di estrarre conoscenza utile da un ampio *dataset* contenente certificazioni energetiche. Attraverso la fase di *Data Integration* sono stati raccolti ed uniformati dati provenienti da diverse fonti *open*, puliti poi nella fase di *Data Cleaning* per ottenere un *dataset* il più possibile uniforme ed utilizzabile per analisi future. La fase di *Data Mining* ha permesso di estrarre la conoscenza e presentarla poi, in maniera semplificata, ai vari *stakeholder* attraverso la visualizzazione su mappe interattive. Questa conoscenza può essere infatti utilizzata da diversi attori in maniera diversa, adeguendola alle proprie esigenze. I privati potrebbero essere interessati all'analisi delle prestazioni energetiche di edifici in particolare aree della città, così come alle diverse *feature* che caratterizzano la geometria degli edifici, in modo da poter attuare una scelta oculata su dove comprare o affittare. La pubblica amministrazione potrebbe essere interessata nell'identificare zone scarsamente performanti dal punto di vista energetico ed attuare degli interventi migliorativi, aumentando il benessere della cittadinanza. E ancora, gli esperti di dominio potrebbero utilizzare il *framework* per caratterizzare, sia con tecniche *supervised* sia con tecniche *unsupervised*, gruppi di edifici con caratteristiche comuni, in modo da effettuare analisi comparative. In estrema sintesi, il *framework* permette di adattarsi alle esigenze dei vari *stakeholder*, fornendo un insieme di strumenti grafici che consentono di fruire facilmente della conoscenza nascosta all'interno dei dati, avendo al contempo modo di settare manualmente un sottinsieme di *feature* e parametri secondo le proprie necessità.

Diversi sono i possibili sviluppi futuri. In particolare si potrebbe pensare di arricchire i dati a disposizione con ulteriori informazioni riguardanti il catasto, le rilevazioni ambientali o estendere l'analisi anche edifici non residenziali. Un ulteriore miglioramento potrebbe essere l'arricchimento delle tecniche di *data mining* presenti, attraverso l'utilizzo di altri

algoritmi di clustering come il *Birch*, il *Gaussian Mixture Model* o il gerarchico, oppure l'estrazione di regole associative utilizzando algoritmi come l'*FP-growth*. Per quando riguarda la visualizzazione della conoscenza su mappa, è possibile aumentare l'interattività attraverso la creazione di *dashboard* dinamiche che diano informazioni puntuali sulla distribuzione di alcuni attributi all'interno di un'area. Un ultimo sviluppo possibile potrebbe essere quello relativo alla generalizzazione della conoscenza, a partire dai dati presenti ad oggi, in modo da predire i compartamenti di edifici non presenti nel *dataset* ma che possiedono caratteristiche comuni, sia per distribuzione geografica sia per appartenenza ad uno dei gruppi individuati durante la fase di *clustering*.

Si sta anche pensando di rilasciare il *framework* sviluppato per ottenere dei *feedback* dagli utenti finali (privati, pubblica amministrazione, esperti di dominio). In questo modo è possibile migliorare il *framework* ed introdurre altre funzioni e metodi di visualizzazione per supportare in maniera migliore il processo di *decision making* degli *stakeholder* di riferimento.

Bibliografia

[1] Livio de Santoli. *La gestione energetica degli edifici*. Dario Flaccovio Editore, 2010.

[2] Luis Pérez-Lombard, José Ortiz e Christine Pout. «A review on buildings energy consumption information». In: *Energy and Buildings* 40 (2008), pp. 394–398. URL: https://www.sciencedirect.com/science/article/pii/S0378778807001016.

[3] *Efficienza e attestazione della prestazione energetica degli edifici in Italia*. URL: https://www.cti2000.eu/.

[4] Giovanni Nidasio Roberto e Murano. «Certificazione energetica degli edifici». In: *Dossier UC* 5 (mag. 2016), pp. 17–36.

[5] *Fac-simile APE*. URL: https://www.certificato-energetico.it/Fac-simile-APE.pdf.

[6] *Legge regionale 28 maggio 2007 n.13 "Disposizioni in meteria di rendimento energertico"*. URL: https://www.cti2000.eu.

[7] *SICEE - Sistema Informativo Certificazione Energetica Edifici*. URL: https://www.agid.gov.it.

[8] *Delibera della Giunta Regionale del 21 settembre 2015 n.14-2119*. URL: https://www.cti2000.eu.

[9] *Decreto del Presidente della Repubblica 26 agosto 1993, n. 412*. URL: http://www.gazzettaufficiale.it/eli/id/1993/10/14/093G0451/sg.

[10] *Decreto del 26 giugno 2015*. URL: http://www.gazzettaufficiale.it/eli/id/2015/07/15/15A05198/sg.

[11] *Casa passiva*. URL: http://www.progettoenergetico.com.

[12] *Cos'è la trasmittanza termica*. URL: http://efficienzaenergetica.acs.enea.it/tecnici/trasmittanza.pdf.

[13] *Impianti termici: concetti innovativi della norma vigente*. URL: http://www.anima.it.

[14] *Cap13 - Sistemi impiantistici*. URL: http://manuali.cened.it.

[15] Angela Sanchini. *Manuale operativo redazione APE*. URL: http://www.insiel.it.

[16] Usama Fayyad, Gregory Piatetsky-Shapiro e Padharaic Smyth. «Knowledge Discovery and Data Mining: Towards a Unifying Framework». In: *aaai.org* (1996), pp. 82–88. URL: https://www.aaai.org/Papers/KDD/1996/KDD96-014.pdf.

[17] R. Brachman e T. Anand. «The Process of Knowledge Discovery in Database: A Human Centered Approach». In: *AKDDM, AAAI/MIT Press* (1996), pp. 37–58.

[18] Vladimir I Levenshtein. «Binary codes capable of correcting deletions, insertions, and reversal». In: *Soviet Physics-Doklady* (1966), pp. 707–710.

[19] Tania Cerquitelli et al. «Exploring energy performance certificates through visualization». In: *Workshop Proceedings of the EDBT/ICDT 2019 Joint Conference* (2019). URL: www.CEUR-WS.org.

[20] Charu C. Aggarwal e Philip S. Yu. «Outlier Detection for High Dimensional Data». In: *ACM Sigmod Record* (2001), pp. 37–46.

[21] Bernard Rosner. «Percentage points for a generalized ESD many-outlier procedure». In: *Technometrics* 25 (1983), pp. 165–172. URL: https://www.jstor.org/stable/pdf/1268549.pdf?refreqid=excelsior\%3Ae0c2e923ebe9e3b2c1f24644cdc22d67.

[22] Sheldon M. Ross. *Probabilità e statistica per l'ingegneria e le scienze*. Maggioli Editore, 2015.

[23] Martin Ester et al. «A density-based algorithm for discovering clusters in large spatial databases with noise». In: *KDD* (1996).

[24] Evelina Di Corso, Tani Cerquitelli e Daniele Apiletti. «METATECH: METerological Data Analysis for Thermal Energy CHaracterization by Means of Self-Learning Transparent Models». In: *Energies* (2018).

[25] Lei Yu e Huan Liu. «Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution». In: *Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)* (2003).

[26] Jiawi Han, Micheline Kamber e Jian Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2012.

[27] B.-H. Rabiner L.R. Juang. «The segmental K-Means algorithm for estimating parameters of hidden Markov models». In: *IEEE Transactions on Acoustics, Speech and Signal Processing* 38 (1990), pp. 1639–1641.

[28] Stephen C. Johnson. «Hierarchical Clustering Schemes». In: *Psychometrika* 32 (1967), pp. 241–254.

[29] Purnima Bholowalia e Arvind Kumar. «EBK-Means: A Clustering Technique based on Elbow Method and K-Means in WSN». In: *International Journal of Computer Applications* 105 (2014), pp. 17–24.

[30] Jan Štrobl1, Marek Piorecký e Vladimír Krajča. «Methods for automatic estimation of the number of clusters for k-means algorithm used on EEG Signal: Feasibility study». In: *Lekar a technika – Clinician and Technology* 47 (2017), pp. 81–87.

[31] *ClustOfVar Manual*. 2017. URL: cran.r-project.org/web/packages/ClustOfVar/ClustOfVar.pdf.

[32] R Project. «The R Foundation. Available: https://www.r-project.org/». In: ().

[33] Leo Breiman. *Classification and Regression Trees*. Routledge, 2017.

[34] John W Tukey. «Box-and-whisker plots». In: *Exploratory data analysis* (1977), pp. 39–43.

[35] Wenqian Shang et al. «A novel feature selection algorithm for text categorization». In: *Expert Systems with Applications* 33 (2005), pp. 1–5.

[36] Michael Friendly. «Milestones in the history of thematic cartography, statistical graphics, and data visualization». In: *yorku.ca record* (2009), pp. 1–79.

[37] Jeremy W. Crampton. «Rethinking maps and identity: Choropleths, clines, and biopolitics». In: *5274P RETHINKING MAPS-A/rev/lb.qxd* (2009), pp. 26–49.

[38] Wes McKinney. «pandas: a Foundational Python Library for Data Analysis and Statistics». In: (2011).

[39] John Hunter e Dale Darren. «The Matplotlib User's Guide». In: (2007).

[40] Fabian Pedregosa et al. «Scikit-learn: Machine Learning in Python». In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[41] *python-visualization/folium: v0.6.0*. 2018. URL: doi.org/10.5281/zenodo.1344457.

[42] Rapid Miner Project . «The Rapid Miner Project for Machine Learning. Available: http://rapid-i.com/ Last access on December 2015». In: ().

[43] *Fattori di conversione in energia primaria dell'energia termica fornita ai punti di consegna della rete di teleriscaldamento della rete di Torino*. 2017. URL: ttps://www.gruppoiren.it/documents/21402/69847/PEF-Torino-2016.pdf/01ff7f45-a025-40e0-9ed8-a39bd7659158.

[44] *Scikit-Learn Feature Selection Methods*. 2019. URL: scikit-learn.org/stable/modules/classes.html#module-sklearn.feature_selection.